

**А.М. Голиков**

**МОДУЛЯЦИЯ, КОДИРОВАНИЕ И МОДЕЛИРОВАНИЕ  
В ТЕЛЕКОММУНИКАЦИОННЫХ СИСТЕМАХ**

**Теория и практика**

**Учебное пособие**

**Томск**

Министерство образования и науки Российской Федерации  
Федеральное государственное бюджетное образовательное учреждение высшего  
профессионального образования  
**Томский государственный университет систем управления и радиоэлектроники**

**А.М. ГОЛИКОВ**

**МОДУЛЯЦИЯ, КОДИРОВАНИЕ И МОДЕЛИРОВАНИЕ  
В ТЕЛЕКОММУНИКАЦИОННЫХ СИСТЕМАХ**

**(Теория и практика)**

**Учебное пособие**

**2016**

УДК 621.39(075.8)

ББК 32.973(я73)

Г 60

**Голиков А.М.**

Модуляция, кодирование и моделирование в телекоммуникационных системах. Теория и практика: учебное пособие / А.М.Голиков. – Томск: Томск. гос. ун-т систем упр. и радиоэлектроники, 2016. – 516 с.: ил. — (Учебная литература для вузов)

Учебное пособие предназначено для направления подготовки магистров 11.04.02 "Инфокоммуникационные технологии и системы связи" по магистерской программе подготовки: "Радиоэлектронные системы передачи информации", магистерской программе: "Оптические системы связи и обработки информации" и магистерской программе: "Инфокоммуникационные системы беспроводного широкополосного доступа", и специалитета 210601.65 Радиоэлектронные системы и комплексы, а также бакалавриата 11.03.02 Инфокоммуникационные технологии и системы связи (Системы мобильной связи, Защищенные системы и сети связи, Системы радиосвязи и радиодоступа) и может быть полезна аспирантам.

Современные учебные курсы редко рассматривают комплексно вопросы модуляции и кодирования, а также их сигнально-кодовые конструкции. Мало учебников и для реализации современных модемов и кодеков. Актуальность пособия велика, так как в современных системах связи и телевидения, а также кабельных сетях применяются все более сложные виды модуляции и кодирования, обеспечивающие высокую помехоустойчивость.

Методология изучения курса состоит в закреплении теоретических знаний на примерах компьютерной реализации модемов и кодеков современных телекоммуникационных систем. Методическая новизна состоит в исследовании имитационных моделей модемов и кодеков, а также телекоммуникационных систем, представленных на рынке современной аппаратуры с использованием ПО MATLAB и LabVIEW.

**ОГЛАВЛЕНИЕ**

ВВЕДЕНИЕ.....	7
1. ЦИФРОВЫЕ ВИДЫ МОДУЛЯЦИИ И СИГНАЛЬНОГО КОДИРОВАНИЯ, ИХ СПЕКТРАЛЬНАЯ И ЭНЕРГЕТИЧЕСКАЯ ЭФФЕКТИВНОСТЬ.....	14
2. ПРОПУСКНАЯ СПОСОБНОСТЬ КАНАЛА СВЯЗИ. КОДИРОВАНИЕ ИСТОЧНИКА.....	69
3. ПОМЕХОУСТОЙЧИВОЕ КОДИРОВАНИЕ В ТЕЛЕКОММУНИКАЦИОННЫХ СИСТЕМАХ.....	122
4. СИГНАЛЬНО-КODOVЫЕ КОНСТРУКЦИИ В ТЕЛЕКОММУНИКАЦИОННЫХ СИСТЕМАХ.....	265
5. МОДЕМЫ И КОДЕКИ СОВРЕМЕННЫХ ТЕЛЕКОММУНИКАЦИОННЫХ СИСТЕМ.....	307
ЗАКЛЮЧЕНИЕ.....	514
ЛИТЕРАТУРА.....	516

## СОДЕРЖАНИЕ

ВВЕДЕНИЕ.....	7
1. ЦИФРОВЫЕ ВИДЫ МОДУЛЯЦИИ И СИГНАЛЬНОГО КОДИРОВАНИЯ, ИХ СПЕКТРАЛЬНАЯ И ЭНЕРГЕТИЧЕСКАЯ ЭФФЕКТИВНОСТЬ.....	14
1.1. Анализ цифровых методов модуляции.....	14
1.2. Модемы сотовой связи FSK, MSK GMSK и численный анализ вероятности символьной ошибки с использованием ПО LabVIEW.....	30
1.3. Модемы спутниковых систем связи M-QAM, M-PSK и численный анализ вероятности символьной ошибки с использованием ПО LabVIEW.....	47
2. ПРОПУСКНАЯ СПОСОБНОСТЬ КАНАЛА СВЯЗИ. КОДИРОВАНИЕ ИСТОЧНИКА.....	69
2.1. Пропускная способность канала связи. Объем сигнала и емкость канала связи, условия их согласования.....	69
2.2. Исследование кодирования источника дискретных сообщений методами Шеннона-Фано.....	71
2.3. Исследование алгоритмов Лемпеля - Зива.....	78
2.4. Фрактальные методы кодирования изображений.....	87
2.5. Вейвлет преобразования сигналов и изображений с использованием.....	99
3. ПОМЕХОУСТОЙЧИВОЕ КОДИРОВАНИЕ В ТЕЛЕКОММУНИКАЦИОННЫХ СИСТЕМАХ.....	122
3.1. Исследование кодов Хемминга, БЧХ (Боуза-Чоудхури-Хоквенгема), Рида- Соломона на базе MATLAB.....	122
3.2. Циклические избыточные коды CRC (Cyclic redundancy check).....	156
3.3. Сверточные коды. Декодирование сверточных кодов.....	170
3.4. Декодирование сверточных кодов по методу Витерби с использованием ПО MATLAB.....	191
3.5. Турбокодирование. Обобщенная схема турбокодера с параллельным каскади- рованием. Сверточные турбокоды. Декодирование турбокодов. Характеристики помехоустойчивости сверточных турбокодов. Исследование турбокодов с использованием ПО MATLAB.....	211
3.6. Низкоплотностные коды. Классификация LDPC-кодов. Методы построения проверочных матриц. Алгоритмы декодирования низкоплотных кодов. Оценка сложности алгоритмов декодирования на базе MATLAB и LabVIEW.....	225
3.7. Исследование каскадных кодов.....	243
4. СИГНАЛЬНО-КОДОВЫЕ КОНСТРУКЦИИ В ТЕЛЕКОММУНИКАЦИОННЫХ	

СИСТЕМАХ.....	265
4.1. Сигнально-кодовые конструкции на основе Треллис кодовой модуляции (TCM) и их анализ с использованием MATLAB.....	265
4.2. Исследование сигнально-кодовой конструкции на базе системы с ортогональным частотным мультиплексированием и пространственно-временным кодированием OFDM - MIMO с использованием NI LabVIEW.....	285
5. МОДЕМЫ И КОДЕКИ СОВРЕМЕННЫХ ТЕЛЕКОММУНИКАЦИОННЫХ СИСТЕМ.....	307
5.1. Имитационное моделирование системы мобильной связи стандарта GSM на базе ПО MATLAB.....	307
5.2. Имитационное моделирование системы мобильной связи стандарта CDMA на базе ПО MATLAB.....	326
5.3. Имитационное моделирование системы мобильной связи стандарта IEEE 802.11 (WiFi) на базе MATLAB 2015.....	350
5.4. Имитационное моделирование системы мобильной связи стандарта IEEE 802.15.4 ZigBee на базе ПО MATLAB.....	373
5.5. Имитационное моделирование системы мобильной связи стандарта IEEE 802.15.1 (Bluetooth) на базе ПО MATLAB.....	386
5.6. Имитационное моделирование системы мобильной связи стандарта IEEE 802.16 (WiMAX) на базе ПО MATLAB.....	400
5.7. Имитационное моделирование системы мобильной связи стандарта IEEE 802. 20 LTE на базе ПО MATLAB.....	427
5.8. Имитационное моделирование системы цифрового наземного телевизионного вещания DVB-T .....	456
5.9. Имитационное моделирование системы цифрового спутникового телевизионного вещания DVB-S и системы высокоскоростного цифрового спутникового ТВ-вещания DVB-S2.....	467
5.10. Имитационное моделирование системы цифрового кабельного телевизионного вещания DVB-C и системы высокоскоростного цифрового кабельного ТВ-вещания DVB-C2.....	474
5.11. Имитационное моделирование системы цифрового мобильного телевизионного вещания DVB-H и системы высокоскоростного цифрового мобильного ТВ-вещания DVB-H2.....	496
ЗАКЛЮЧЕНИЕ.....	514
ЛИТЕРАТУРА.....	516

## ВВЕДЕНИЕ

В зависимости от назначения и условий функционирования системы радиосвязи ее эффективность оценивается на основании тех или иных показателей (критериев), основными из которых являются энергетический и спектральный. Соответственно, важнейшими характеристиками любой системы радиосвязи являются энергетическая и спектральная эффективность, характеризующие, соответственно, энергетические затраты и полосу занимаемых частот, необходимые для передачи сообщений.

К сожалению, одновременное достижение предельных значений этих показателей эффективности оказывается невозможным, так что в каждом конкретном случае построения системы радиосвязи приходится руководствоваться компромиссными соображениями при оптимизации характеристик и режимов функционирования системы [1-4].

Эффективные методы модуляции и помехоустойчивого кодирования все шире используются в современных цифровых телекоммуникационных системах (ТКС). Переход в ТКС к ансамблям многопозиционных сигналов увеличивает информационную скорость и обеспечивает передачу больших потоков информации. Современная элементная база позволяет применять в ТКС достаточно сложные методы помехоустойчивого кодирования и обеспечивать тем самым высокую верность передачи информации. Исследованию методов модуляции и кодирования в ТКС посвящена обширная литература, отражающая как достижения теории, так и вопросы ее практических приложений [1-18].

Как известно, теория сигналов и теория кодирования длительное время развивались независимо. В последние годы значительно возрос интерес к новому перспективному направлению, возникшему на стыке этих наук. В работах отечественных и зарубежных авторов интенсивно исследуются возможности ТКС, в которых для передачи информации используются ансамбли многопозиционных сигналов в сочетании с помехоустойчивыми кодами, причем, процедуры модуляции/кодирования (демодуляции/декодирования) осуществляются *совместно*. При рациональном построении такие *сигнально-кодовые конструкции* (СКК) сочетают в себе положительные качества как многопозиционных ансамблей сигналов, так и помехоустойчивых кодов, допускают достаточно простые и реализуемые на практике алгоритмы декодирования и при использовании их в ТКС позволяют существенно продвинуться к теоретическим пределам эффективности. Вопросы синтеза таких систем модуляции/кодирования, анализа их структуры, помехоустойчивости и эффективности, демодуляции/декодирования составляют основное содержание нового перспективного направления в теории связи - *теории сигнально-кодовых конструкций*.

Один путь базируется на введенном А.Г. Зюко [1, 2] понятии эффективности систем связи. Анализ и сравнение эффективности систем с многопозиционными сигналами и систем с корректирующими кодами приводит в *информационной теории СКК* к идее *их комбинации в единой сигнально-кодовой конструкции*. Такой путь является наглядным, убедительным и излагается без строгих математических выкладок. Этот путь может быть использован для популярного изложения оснований теории СКК [1-4].

Статистическая теория связи предлагает большое количество вариантов построения телекоммуникационных систем. Как из этого множества выбрать вариант, наиболее целесообразный в заданных условиях? По каким критериям следует производить этот выбор? Насколько оправдано применение тех или иных новых систем и как совершенствовать существующие системы? Принципиальное решение этих вопросов в конечном итоге сводится к *оптимизации систем связи по критериям эффективности*.

Типовая структура одноканальной системы передачи дискретной информации (СПДИ) приведена на рис. 1.1. Источник вырабатывает сообщения, которые необходимо передавать по каналу СПДИ. Это могут быть последовательности *дискретных* сообщений (данные, телеграфные сообщения и т.д.) либо *непрерывные* сообщения (речь, телевидение, результаты телеизмерений и др.), преобразованные в *цифровую форму*.

Реальные сообщения содержат *избыточность* и для согласования источника с каналом передачи информации используют *кодер источника*. Совместно с *декодером* они образуют *кодек источника*. Методы кодирования источника изложены в работах [2, 3, 7, 9-11, 13-16, 18 ].

Основные требования к СПДИ формулируются достаточно просто:

1. ***верность,***
2. ***скорость,***
3. ***своевременность доставки информации от отправителя к получателю.***

В системах без помехоустойчивого кодирования верность и скорость зависят от вида используемых сигналов-переносчиков. Применение помехоустойчивого кодирования позволяет повысить верность передачи, но за счет снижения скорости. С другой стороны, выбором сигналов можно добиться повышения скорости передачи информации, но, зачастую, в ущерб верности передачи. Однако, следует учитывать, что применение корректирующих кодов вносит задержку в передачу цифровых данных, что ухудшает такой показатель, как своевременность доставки информации. В каждом конкретном случае телекоммуникационных систем (ТКС) могут быть сформулированы количественные выражения этих требований. Это зависит от назначения ТКС и вида передаваемого

сообщения. В общем случае показатели верности, скорости и своевременности передачи информации находятся в противоречивых соотношениях.

Кроме отмеченных выше, следует упомянуть о *дополнительных требованиях*, которые определяют применение того или иного метода передачи информации:

**Спектральная эффективность.** Современные ТКС работают в условиях дефицита частотного спектра, отводимого для передачи информации. Это и обуславливает важность методов формирования *компактных спектров сигналов*, передаваемых по линиям связи с ограниченной полосой частот.

**Сложность реализации методов модуляции-кодирования.** Теория информации указывает путь повышения помехоустойчивости - использование для передачи *длинных последовательностей сигналов и кодов*, для извлечения информации из которых на приемной стороне приходится применять достаточно сложные алгоритмы обработки. Поэтому *конструктивная теория кодирования* направлена на поиск корректирующих кодов, допускающих *реализуемые алгоритмы декодирования* [4].

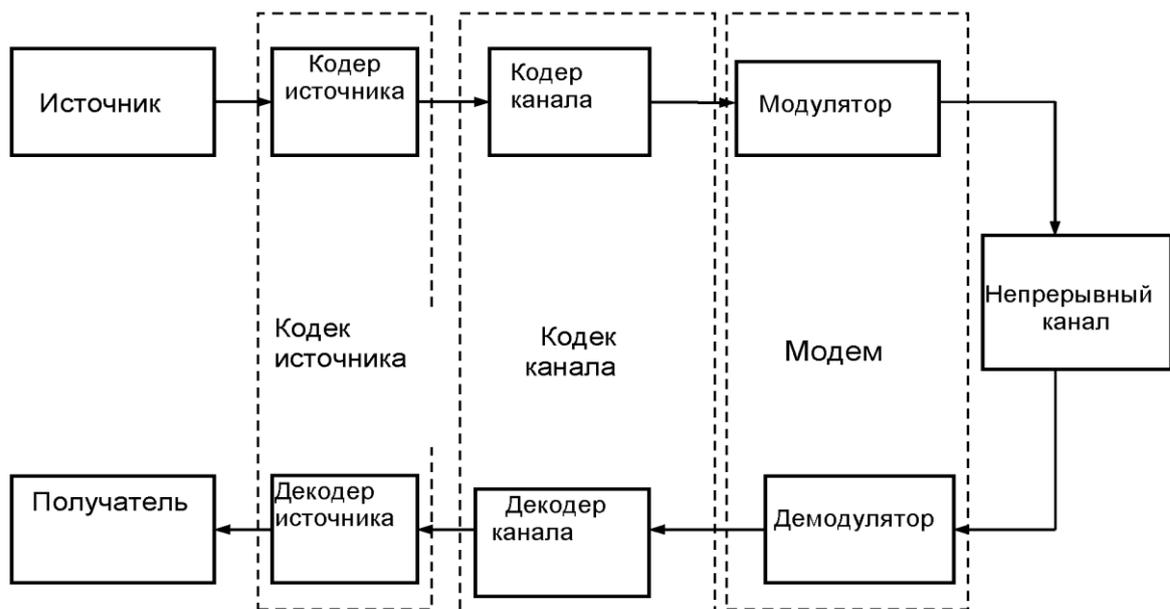


Рис. 1.1. Структурная схема СПДИ

Согласно известным теорем К. Шеннона, в принципе *возможно сколь угодно большое повышение верности передачи информации*, если скорость передачи по каналу  $R_{кан}$  не превышает пропускной способности канала  $C_K$ . Достигается это применением достаточно длинных *корректирующих кодов* (КК). С этой целью в структуру КК вводится *избыточность*.

*Кодек КК (кодер и декодер канала)* показан на рис. 1.1. В реальных условиях длина кода ограничена допустимой сложностью устройств кодирования и, прежде всего,

декодирования, поэтому эффект от применения корректирующих кодов зависит от параметров кода и ограничений на реализацию кодека канала.

Современная теория предлагает широкий набор корректирующих кодов, различных по структуре, принципам построения и корректирующей способности. *Кодем СКК* включает в себя сочетание кодека/модулятора и, соответственно, демодулятора/декодера.

В общем случае результат работы системы связи определяется количеством и качеством передаваемой информации. Количество оценивается скоростью передачи информации по каналу (бит/с), а качество - величиной ошибки. Согласно теоремам К. Шеннона, ошибка при соответствующем выборе метода передачи (модуляции/кодирования) может быть сделана произвольно малой. В то же время, скорость передачи не может быть выше некоторого информационного ресурса, называемого *пропускной способностью канала С*.

В работе А.Г. Зюко [2] было предложено считать одним из показателей эффективности системы величину средней скорости  $R_{кан}$  при которой обеспечивается заданная верность передачи информации. При этом могут быть определены следующие *показатели эффективности*:

- ***Информационная эффективность системы, определяющая степень использования пропускной способности канала относительной величиной***

$$\eta = \frac{R_{кан}}{C} \quad (1.1)$$

В реальных условиях показатель  $\eta$  всегда меньше единицы. Чем ближе  $\eta$  к единице, тем совершеннее система передачи информации. Достижение необходимых скорости и верности передачи сопровождается определенными затратами других важнейших ресурсов: ***мощности сигнала  $P_c$  и полосы частот канала  $F_k$*** . Такой подход позволил ввести показатели эффективности использования ресурсов системы, а именно:

- ***Энергетическая эффективность***

$$\beta = \frac{R_{кан}}{P_c / N_0} \quad (1.2)$$

где  $N_0$  - спектральная плотность мощности шума;

- ***Частотная эффективность***

$$\gamma = \frac{R_{кан}}{F} \quad (1.3)$$

где  $F$  - полоса частот канала.

Показатели  $\beta$  и  $\gamma$  имеют смысл удельных скоростей, а обратные величины  $\beta' = 1/\beta$  и  $\gamma' = 1/\gamma$  определяют *удельные расходы соответствующих ресурсов* на передачу информации с единичной скоростью (1 бит/с).

Для гауссовского канала с полосой  $F$ , отношением мощностей сигнала и шума  $\rho = \frac{P_c}{P_{ш}}$  и пропускной способностью  $C = F \log_2(1 + \rho)$  была установлена

связь между показателями эффективности

$$\eta = \frac{\gamma}{\log_2(1 + \frac{\gamma}{\beta})} \quad \text{и} \quad \gamma = \rho\beta$$

Для идеальной системы ( $\eta$ ) определена также предельная зависимость

$$\beta = \frac{\gamma}{2^\gamma - 1}. \quad (1.5)$$

Зависимость энергетической эффективности от удельной скорости удобно представить в виде кривой на плоскости  $\beta\gamma$  (рис. 1.2). Она отражает наилучший обмен между  $\beta$  и  $\gamma$  в непрерывном канале (НК-предел Шеннона). При этом частотная эффективность  $\gamma$  изменяется в пределах от 0 до  $\infty$  то время как энергетическая эффективность ограничена сверху, поскольку

$$\beta_{\max} = \lim_{\gamma \rightarrow 0} \beta = \lim_{\gamma \rightarrow 0} \frac{\gamma}{2^\gamma - 1} = \frac{1}{\ln 2} \approx 1,443(1,59\text{дБ}) \quad (1.6)$$

Аналогичные предельные кривые могут быть построены для других типов каналов [2] (ДНК-дискретно-непрерывный канал, ДСК-дискретный симметричный канал на рис. 1.2).

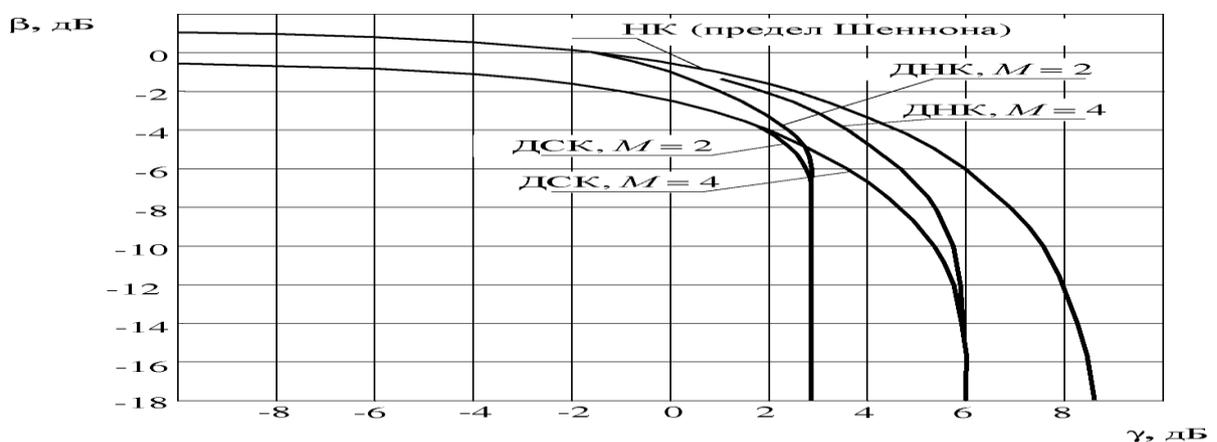


Рис. 1.2. Кривые предельной эффективности систем связи

Полосу частот, занимаемую сигналом конечной длительности не всегда можно определить однозначно. Поэтому далее для анализа и сравнения эффективности будем рассматривать следующую модель процесса передачи информации.

Полагаем, что на интервале  $T$  по каналу передается определенное количество независимых отсчетов  $N_T$  (по В. А. Котельникову  $N_T = 2FT$ ).

Тогда  $D = \frac{N_T}{T}$  число передаваемых отсчетов в единицу времени.

В этой модели могут быть определены следующие показатели:

**- Удельная скорость передачи информации**

$$RN = R_{\text{кан}} / D \text{ (бит / отсчет)}, \quad (1.7)$$

**- Удельная информационная емкость\* канала**

$$C_N = \frac{C}{D} \text{ (бит / отсчет)}, \quad (1.8)$$

**- Удельная средняя энергия сигнала, затрачиваемая на передачу одного отсчета**

$$EN = P_c / D$$

**- Удельная эффективность использования пропускной способности канал**

$$\eta_N = \frac{R_N}{C_N} = \eta, \quad (1.9)$$

**- Удельная энергетическая эффективность (удельная эффективность использования энергии сигнала)**

$$\beta_N = \frac{R_N}{E_N / N_0} = \beta, \quad (1.10)$$

**- Удельная скорость передачи информации**

$$\gamma_N = R_N. \quad (1.11)$$

Переход к удельным показателям эффективности означает, что процесс формирования сигналов рассматривается в ортогональном базисе. При этом удельные показатели

эффективности (1.9) и (1.10) точно совпадают с каноническими (1.4), (1.5), но удается избежать неопределенности при оценке полосы частот [1, 2].

В Главе 1 учебного пособия рассматриваются цифровые виды модуляции и сигнального кодирования, их спектральная и энергетическая эффективность. Приводятся краткие теоретические сведения и практическая их реализация на базе программного обеспечения NI LabVIEW. Даны результаты исследования вероятности ошибки на бит от отношения сигнал шум в канале для различных видов модуляции FSK, MSK и GMSK, а также QAM, M-QAM, PSK, M-PSK, созвездия для многопозиционных методов модуляции, спектры, глазковые диаграммы [11] и джиттер.

В Главе 2 рассматривается пропускная способность канала связи. Кодирование источника. Теорема К.Шеннона о кодировании в дискретном канале без помех. Проводится исследование кодирования источника дискретных сообщений методами Шеннона-Фано, исследование алгоритмов Лемпеля - Зива, Фрактальные методы кодирования изображений, а также вейвлет преобразования сигналов и изображений с использованием ПО MATLAB [13].

В Главе 3 исследуется помехоустойчивое кодирование в телекоммуникационных системах. Исследуются характеристики кодов Хемминга, БЧХ (Боуза-Чоудхури-Хоквенгема) и Рида-Соломона на базе MATLAB. Проведено исследование циклического избыточного кода (Cyclic redundancy check) CRC с использованием ПО MATLAB. Рассмотрены кодеры сверточных кодов и методы их декодирования, включая метод Витерби, исследованы их характеристики с использованием ПО MATLAB. Подробно исследовано турбокодирование, обобщенная схема турбокодера с параллельным каскадированием. Сверточные турбокоды. Декодирование турбокодов. Оценены характеристики помехоустойчивости сверточных турбокодов. Исследование турбокодов проведено с использованием ПО MATLAB. Рассмотрены низкоплотные коды, дана классификация LDPC-кодов, показаны методы построения проверочных матриц. Реализованы алгоритмы декодирования низкоплотных кодов, проведена оценка сложности алгоритмов декодирования на базе MATLAB и NI LabVIEW, исследованы каскадные коды на базе MATLAB Simulink 2015 [1, 3, 8-11, 16].

В Главе 4 рассмотрены сигнально-кодовые конструкции в телекоммуникационных системах, критерии эффективности систем передачи информации, информационная теория сигнально-кодовых конструкций, системы многочастотной OFDM модуляции - проведено имитационное моделирование в среде LabVIEW, реализованы Треллис кодовая модуляция (TCM) и проведен ее анализ с использованием MATLAB, рассмотрены методы

пространственно-временного кодирования (MIMO) и оценена их эффективность на базе NI LabVIEW [12, 16].

В Главе 5 рассмотрены модемы и кодеки современных телекоммуникационных систем, имитационное моделирование системы мобильной связи стандартов GSM на базе ПО MATLAB, имитационное моделирование системы мобильной связи стандарта CDMA на базе ПО MATLAB, имитационное моделирование системы мобильной связи стандарта IEEE 802.11 (WiFi) на базе MATLAB 2015, имитационное моделирование системы мобильной связи стандарта IEEE 802.15.4 ZigBee на базе ПО MATLAB, имитационное моделирование системы мобильной связи стандарта IEEE 802.15.1 (Bluetooth) на базе ПО MATLAB, имитационное моделирование системы мобильной связи стандарта IEEE 802.16 (WiMAX) на базе ПО MATLAB [16, 18], имитационное моделирование системы мобильной связи стандарта IEEE 802.20 LTE на базе ПО MATLAB, имитационное моделирование системы цифрового наземного телевизионного вещания DVB-T [16], имитационное моделирование системы цифрового спутникового телевизионного вещания DVB-S и системы высокоскоростного цифрового спутникового ТВ-вещания DVB-S2 [16], имитационное моделирование системы цифрового кабельного телевизионного вещания DVB-C и системы высокоскоростного цифрового кабельного ТВ-вещания DVB-C2 [16], имитационное моделирование системы цифрового мобильного телевизионного вещания DVB-H и системы высокоскоростного цифрового мобильного ТВ-вещания DVB-H2 [16, 19, 20].

## **ГЛАВА 1. ЦИФРОВЫЕ ВИДЫ МОДУЛЯЦИИ И СИГНАЛЬНОГО КОДИРОВАНИЯ, ИХ СПЕКТРАЛЬНАЯ И ЭНЕРГЕТИЧЕСКАЯ ЭФФЕКТИВНОСТЬ**

### **1.1. Анализ цифровых методов модуляции**

#### **Модулятор**

Известно [6, 7], что узкополосный модулированный сигнал с произвольным видом модуляции можно представить в виде:

$$s(t) = I(t) \cos(\omega t) - Q(t) \sin(\omega t), \quad (1.12)$$

где  $\omega$  - несущая частота радиосигнала,  $I(t)$  и  $Q(t)$  называются соответственно *синфазной* и *квадратурной* составляющими модулирующего сигнала.

Таким образом, для осуществления произвольного вида модуляции сигнала необходимо

выполнить две операции: (1) сформировать синфазную и квадратурную составляющие модулирующего сигнала (вид данных составляющих будет определять вид модуляции)

и (2) выполнить преобразование (1.12).

Выполнение операций (1) и (2) выполняется различными блоками передающего тракта. Операция (1) осуществляется в baseband-процессоре, а операция (2) в квадратурном (IQ) модуляторе.

*Baseband-модулятор* формирует низкочастотные (baseband) сигналы  $I(t)$  и  $Q(t)$  из закодированного информационного сигнала (последовательности нулей и единиц). Закон, по которому выполняется данное преобразование, определяет вид модуляции сигнала.

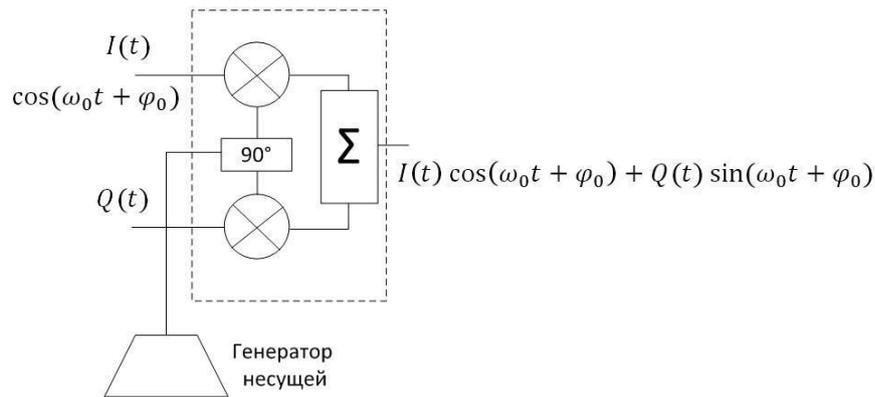


Рис. 1.3. IQ-модулятор

Для IQ-модулятора входными сигналами являются

$$I_{in}(t) = I(t); \quad Q_{in}(t) = Q(t) .$$

Сформированные baseband-процессором из исходного информационного сигнала и определяющие вид и свойства модулированного сигнала. На модулятор также поступает немодулированное несущее колебание вида  $\cos(\omega t)$ . На выходе модулятора образуется сигнал (1.12) посредством умножения сигнала  $I(t)$  на немодулированное несущее колебание,  $Q(t)$  на немодулированное несущее колебание, сдвинутое по фазе на  $90^\circ$ , и последующего суммирования.

Видно, что структура IQ-модулятора является инвариантной относительно вида модуляции. Таким образом, осуществление того или иного вида модуляции определяется программой, выполняемой baseband-процессором, а именно алгоритмом формирования квадратурных составляющих из закодированного информационного сигнала. IQ-модулятор работает на высокой (несущей) частоте и, как правило, является аналоговым устройством. Таким образом, структурно передающий тракт можно разделить на цифровую и аналоговую части, разделенные цифро-аналоговым преобразователем (ЦАП), формирующим аналоговый сигнал из последовательности поступающих на него отсчетов.

Возможна также и иная реализация, когда IQ-модулятор выполнен в цифровом виде и преобразование (1.12) осуществляется baseband-процессором. В этом случае на выходе

baseband-процессора формируется сигнал  $s(t)$  на промежуточной частоте, существенно более низкой, чем несущая. Данный сигнал преобразуется в аналоговый с помощью ЦАП и затем его спектр переносится на несущую частоту с помощью смесителя.

### Цифровые виды модуляции [7]

Цифровые виды модуляции (часто цифровая модуляция называется манипуляцией), как и аналоговые, могут быть амплитудными, фазовыми, частотными или комбинированными (например, амплитудно-фазовыми), в зависимости от того, какой из параметров немодулированного несущего колебания  $s(t) = A(t)\cos(\omega t) + p(t)$  изменяется в соответствии с изменением информационного сигнала. Так как значения цифрового информационного сигнала являются дискретными (например,  $\{0,1\}$ ), дискретным является также и возможный набор значений каждого из параметров. Однако если информационный сигнал проходит через baseband-фильтр для ограничения спектра, его значения уже не являются дискретными, поэтому реально переход от одного дискретного значения параметра колебания (например, изменение амплитуды или фазы) происходит гладко и непрерывно.

### Амплитудные виды модуляции (OOK, ASK, M-ASK)

Наиболее простым видом манипуляции сигнала является амплитудная манипуляция. Модулированный сигнал имеет вид:

$$s(t) = A(c(t) + B)\cos(\omega t + \varphi_0),$$

где  $c(t)$  - информационный цифровой сигнал,  $A$ ,  $B$  и  $\varphi_0$  - постоянные,  $B \geq 0$ ,  $\omega$  - несущая частота. Пусть множество возможных значений  $c(t)$   $\{0,1\}$ ,  $B=0$ . В этом случае модулированный сигнал имеет вид  $s(t) = Ac(t)\cos(\omega t + \varphi_0)$ , его амплитуда принимает значение 0 при нулевом значении информационного сигнала и  $A$  при единичном (рис.1.2). Такой тип манипуляции называется OOK (On-Off Keying, Включено-Выключено) и часто используется в системах сигнализации и охранных системах.

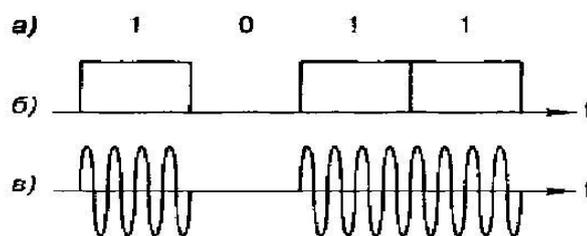


Рис. 1.4. Модуляция OOK: а - информационное сообщение; б - модулирующий цифровой сигнал; в - модулированный радиосигнал

Допустим теперь  $B = 1$ . В этом случае амплитуда модулированного сигнала принимает значение  $A$  при нулевом значении информационного сигнала и  $2A$  при единичном. Вид модуляции, для которого  $B > 0$ , носит название ASK (Amplitude Shift Keying -

амплитудная манипуляция). ООК является частным случаем ASK при  $B=0$ . Существует два основных критерия сравнения эффективности различных видов модуляции. Это *критерии спектральной и энергетической эффективности*. Спектральная эффективность характеризует полосу частот, необходимую для передачи информации с определенной скоростью. Энергетическая эффективность описывает мощность, необходимую для передачи информации с заданной достоверностью (вероятностью ошибки). Известно, что спектр модулированного сигнала на радиочастоте с точностью до постоянного множителя совпадает со спектром модулирующего (baseband) сигнала, однако, центр спектра радиосигнала размещен на несущей частоте, а не на нулевой. Поэтому, как правило, анализируются спектральные плотности модулирующих сигналов, центрированные относительно нулевой частоты. Спектральные плотности мощности ASK сигналов для различных baseband-фильтров приведены на рис.1.4. На рис.1.4 показаны соответствующие формы импульсов модулирующего сигнала после прохождения baseband-фильтра. Из сравнения рис.1.4 и рис.1.5 видно, что более гладкая форма импульса модулирующего сигнала приводит к расширению главного лепестка спектральной плотности мощности модулированного сигнала и более быстрому уменьшению амплитуды боковых лепестков.

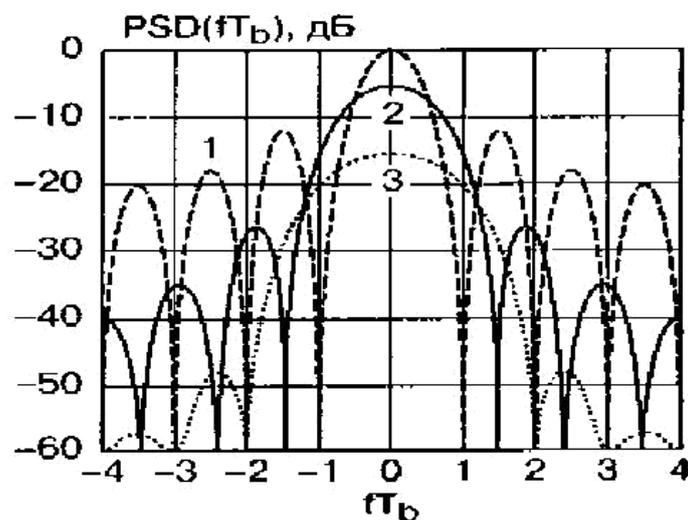


Рис. 1.5. Спектральная плотность мощности ASK-сигнала. Форма импульса модулирующего сигнала: 1 - прямоугольная, 2 - косинусоидальная, 3 - приподнятый косинус.

Множество возможных значений квадратурных компонент  $I(t)$  и  $Q(t)$  называется *сигнальным созвездием*. Как правило, данное множество отображают на декартовой плоскости, где по оси абсцисс отложены значения синфазной составляющей  $I(t)$ , а по оси ординат - квадратурной  $Q(t)$ . Точка на плоскости с координатами  $(x,y)$  соответствует состоянию сигнала, в котором синфазная составляющая равна  $x$ , квадратурная равна  $y$ .

Таким образом, сигнальное созвездие - это диаграмма возможных состояний сигнала. Используя общий вид модулированного радиосигнала, можно показать, что амплитуда модулированного радиосигнала в текущем состоянии равна

$$A(t) = \sqrt{I^2(t) + Q^2(t)},$$

а фаза равна углу вектора, указывающего в точку  $(I, Q)$ , отсчитываемого от оси абсцисс. Для модуляций OOK и ASK сигнальное созвездие изображено на рисунке 1.6.

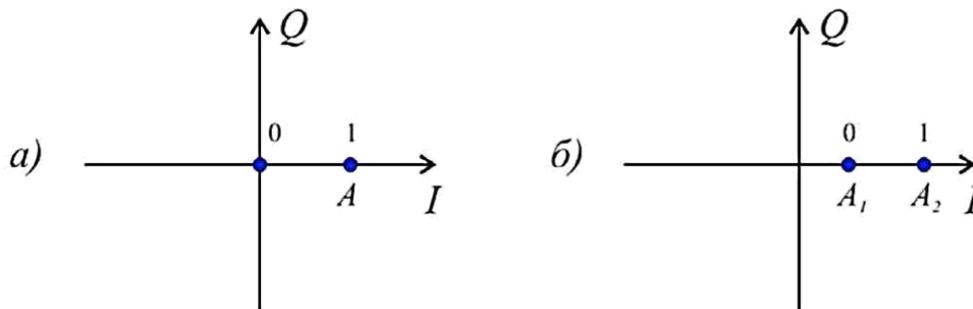


Рис. 1.6. Виды модуляции: а - сигнальное созвездие модуляции OOK, б - сигнальное созвездие модуляции ASK

### Многопозиционная амплитудная модуляция (M-ASK)

При модуляции ASK множество возможных значений амплитуды радиосигнала ограничивается двумя значениями (без учета сглаживания baseband-фильтром). Спектральная эффективность может быть существенно повышена, если использовать большее количество значений амплитуды радиосигнала.

Сгруппируем биты исходного информационного сообщения в пары. Каждая такая пара называется символом. Если каждый бит имеет множество значений  $\{0,1\}$ , то каждый символ имеет четыре возможных значения из множества  $\{00, 01, 10, 11\}$ . Сопоставим каждому из возможных значений символа значение амплитуды радиосигнала из множества  $\{0, A, 2A, 3A\}$ . Аналогичным образом можно группировать тройки, четверки и большее количество бит в одном символе. Получится многоуровневый (многопозиционный) сигнал M-ASK с размерностью множества возможных значений амплитуды сигнала  $M = \log_2 k$ , где  $k$  - число бит в одном символе.

Например, сигнал с модуляцией 256-ASK имеет 256 возможных значений амплитуды сигнала и 8 бит в одном символе.

На рисунке 1.7 изображена спектральная плотность мощности восьмиуровневого сигнала 8-ASK и спектральная плотность сигнала ASK с импульсами прямоугольной формы (без baseband-фильтрации). Многопозиционный сигнал имеет меньшую ширину главного лепестка (занимает меньшую полосу частот) и более низкий уровень боковых лепестков, т. е. имеет большую спектральную эффективность по сравнению с двухуровневым сигналом.

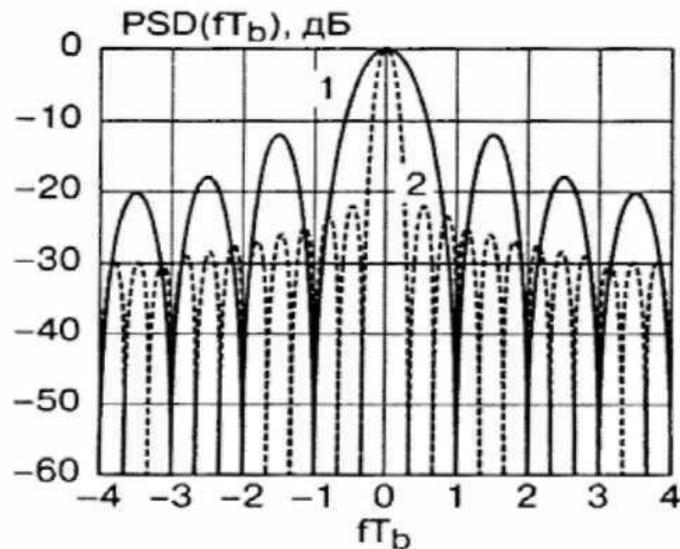


Рис. 1.7. Сравнение спектральных плотностей мощности двухуровневого и восьмиуровневого АМ-сигналов: 1 - сигнал АSK, 2 – сигнал 8-АSK

Амплитудные виды модуляции имеют невысокую энергетическую эффективность (так как средний уровень мощности существенно меньше максимального), требуют высокой линейности и большого динамического диапазона усилителя мощности. Ошибка в амплитуде сигнала из-за нелинейности усилителя приведет непосредственно к символьной ошибке, т. к. значение символа определяется амплитудой сигнала. Отношение максимальной амплитуды сигнала к минимальной достаточно высоко и требует усилителя с большим динамическим диапазоном. Влияние аддитивного шума или помехи непосредственно изменяет амплитуду сигнала, поэтому амплитудные виды модуляции не обладают высокой помехоустойчивостью. Однако они достаточно просты в реализации. Сигнальное созвездие для 8-ASK приведено на рисунок 1.8.

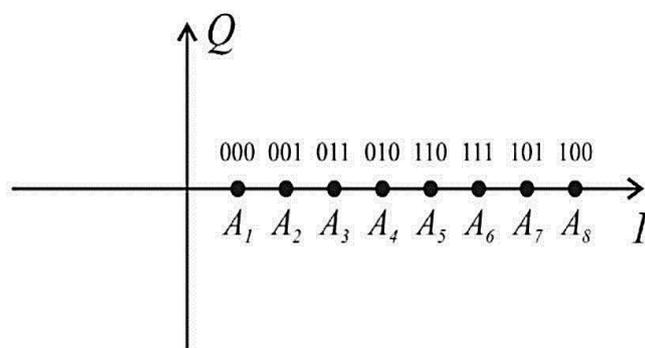


Рис. 1.8. Сигнальное созвездие модуляции 8-АSK

### Фазовые виды модуляции (BPSK, QPSK, M-PSK, QAM)

Фазомодулированный сигнал имеет вид:

$$s(t) = A \cos(\omega t + \varphi(t) + \varphi_0), \quad (1.13)$$

где  $A$  и  $\varphi_0$  - постоянные,  $\omega_0$  - несущая частота.

Информация кодируется фазой  $\varphi(t)$ . Так как при когерентной демодуляции в приемнике имеется восстановленная несущая  $S_C(t) = A \cos(\omega t + \varphi_0)$ , то путем сравнения сигнала с несущей вычисляется текущий сдвиг фазы  $\varphi(t)$ . Изменение фазы  $\varphi(t)$  взаимнооднозначно связано с информационным сигналом  $c(t)$ .

QPSK (Quadrature Phase Shift Keying) - Квадратурная фазовая модуляция (четырёхуровневая фазовая модуляция). В данной фазовой модуляции используются четыре значения фазы несущего колебания. В этом случае фаза сигнала должна принимать четыре значения:  $0^\circ$ ,  $90^\circ$ ,  $180^\circ$  и  $270^\circ$ . Однако чаще используются другие значения фаз:  $45^\circ$ ,  $135^\circ$ ,  $225^\circ$  и  $315^\circ$  с шагом, кратным  $\pi/2$ .

Соотношение между сдвигом фазы модулированного колебания из множества  $\{\pm\pi/4, \pm3\pi/4\}$  и множеством символов (дубитов) цифрового сообщения  $\{00, 01, 10, 11\}$  устанавливается в каждом конкретном случае стандартом на радиоканал и отображается сигнальным созвездием. Сигнальное созвездие – это представление манипулированных радиосигналов на комплексной плоскости.

Модулированный сигнал можно представить как вектор на графике в полярной системе координат; длина вектора соответствует амплитуде сигнала, а его ориентация — фазе. На рисунке 1.9 представлено сигнальное созвездие для QPSK модуляции.

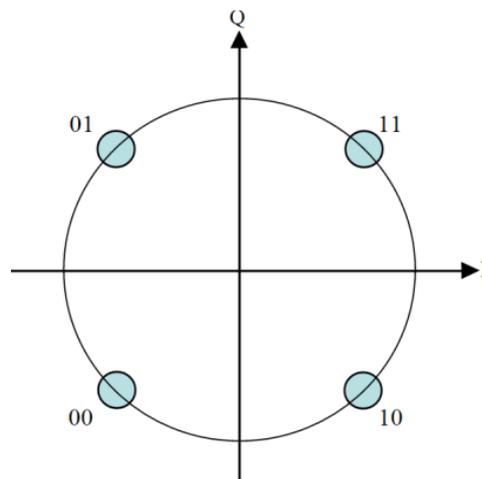


Рис. 1.9. Сигнальное созвездие для QPSK модуляции.

Фаза может переходить из одного состояния в другое. Такие переходы представлены на рисунке 1.10 чёрными стрелками.

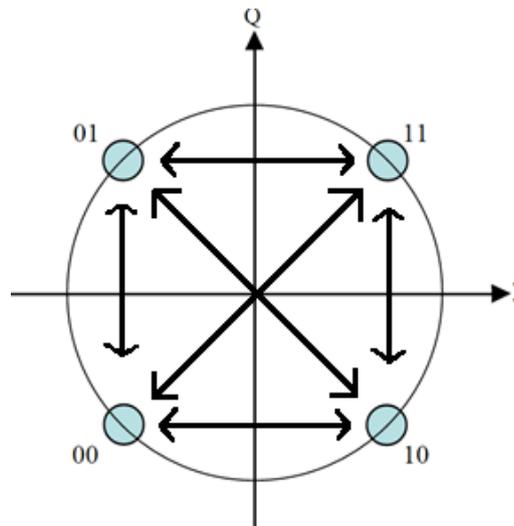


Рис. 1.10. Переходы из одного фазового состояния в другое

Из теории связи известно, что наивысшей помехоустойчивостью обладает двоичная фазовая модуляция BPSK. Однако в ряде случаев за счет уменьшения помехоустойчивости канала связи можно увеличить его пропускную способность.

Квадратурно-амплитудная модуляция - QAM (QAM – Quadrature Amplitude Modulation) может рассматриваться как расширенная многоуровневая ФМ, в которой два исходных сигнала генерируются независимо. Таким образом, здесь имеют место два полностью независимых квадратурных канала, включающие процессы кодирования и детектирования в основной полосе.

На рисунке 1.11 показано сигнально - точечное пространство для системы с 16-QAM и четырьмя уровнями в каждом квадратурном канале. Точки представляют составной сигнал, а штрихи на осях отмечают уровни амплитуды в каждом квадратурном канале.

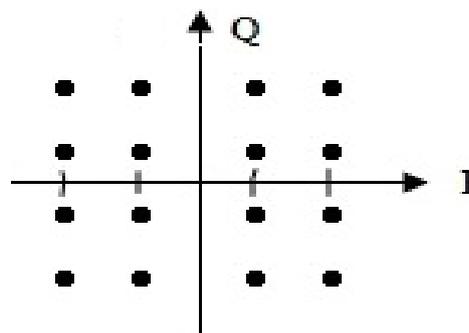


Рис. 1.11. Сигнально-точечное пространство модуляции  
для 16-QAM

В отличие от ФМ сигналов сигналы QAM не содержат постоянной огибающей. Наличие постоянной огибающей в ФМ объясняется поддержанием отношения уровней в квадратурных каналах. В QAM такие ограничения не вводятся ввиду того, что в каждом канале уровни независимы.

Характеристики ошибок систем QAM и ФМ модуляций сильно отличаются. При достаточно большом числе сигнальных точек системы QAM имеют, как правило, лучшие характеристики, чем системы с ФМ. Основная причина состоит в том, что расстояние между сигнальными точками на диаграмме для системы с QAM больше, чем для соответствующей системы с ФМ.

QAM имеет преимущество над системой ФМ при той же пиковой мощности.

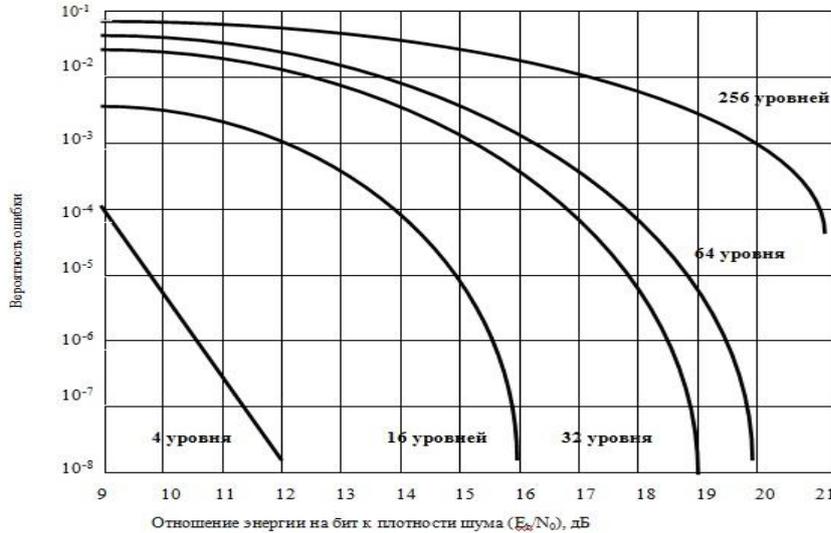


Рис. 1.12. Вероятности ошибок в системах QAM

Достоинство высоких значений номера QAM – это повышенная скорость передачи данных, поскольку таким образом большее количество битов информации может быть передано в течении одного цикла. Однако, с другой стороны, в этом случае большее число уровней амплитуды сигнала располагаются близко друг к другу, повышая тем самым вероятность неразличимости двух уровней, и как следствие – повышая чувствительность системы к шуму.

На рисунке 1.13 представлено отношение параметра SNR к параметру BER (Bit Error Rate – Отношение Бит/Ошибка).

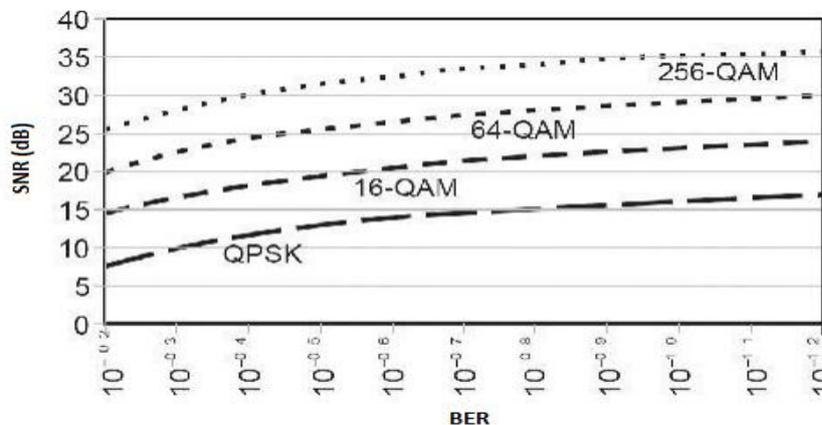


Рис. 1.13. Отношение параметра SNR к параметру BER

### Двоичная фазовая модуляция (BPSK - Binary Phase Shift Keying)

Множеству значений информационного сигнала  $\{0,1\}$  ставится в однозначное соответствие множество изменений фазы  $\{0, \pi\}$ . При изменении значения информационного сигнала фаза радиосигнала изменяется на  $180^\circ$ . Временная форма сигнала и его созвездие показаны на рисунке 1.14.

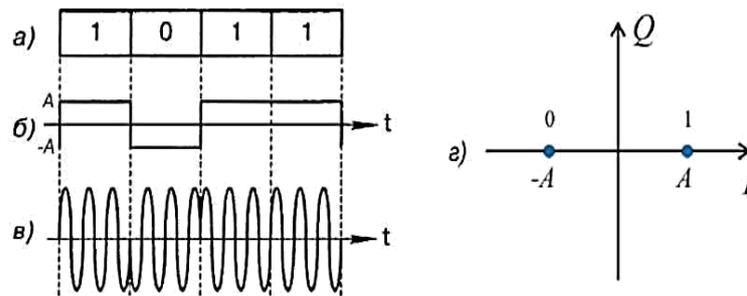


Рис. 1.14. Временная форма и сигнальное созвездие сигнала BPSK: а – цифровое сообщение; б - модулирующий сигнал; в - модулированное ВЧ-колебание; г - сигнальное созвездие

### Квадратурная фазовая модуляция (QPSK - Quadrature Phase Shift Keying)

Квадратурная фазовая модуляция является четырехуровневой фазовой модуляцией ( $M=4$ ), при которой фаза высокочастотного колебания может принимать 4 различных значения с шагом, кратным  $\pi/2$ .

Соотношение между сдвигом фазы модулированного колебания из множества  $\{\pm\pi/4, \pm3\pi/4\}$  и множеством символов (дибитов) цифрового сообщения  $\{00, 01, 10, 11\}$  устанавливается в каждом конкретном случае стандартом на радиоканал и отображается сигнальным созвездием, аналогичным рисунку 1.15. Стрелками показаны возможные переходы из одного фазового состояния в другое.

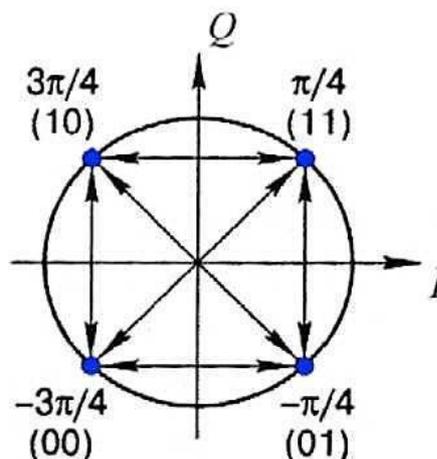


Рис. 1.15. Сигнальное созвездие модуляции QPSK

Из рисунка видно, что соответствие между значениями символов и фазой сигнала установлено таким образом, что в соседних точках сигнального созвездия значения соответствующих символов отличаются лишь в одном бите.

При передаче в условиях шума наиболее вероятной ошибкой будет определение фазы соседней точки созвездия. При указанном кодировании, несмотря на то, что произошла ошибка в определении значения символа, это будет соответствовать ошибке в одном (а не двух) бите информации. Таким образом, достигается снижение вероятности ошибки на бит. Указанный способ кодирования называется кодом Грея.

Каждому значению фазы модулированного сигнала соответствует 2 бита информации, и поэтому изменение модулирующего сигнала при QPSK - модуляции происходит в 2 раза реже, чем при BPSK-модуляции при одинаковой скорости передачи информации. Известно [5], что спектральная плотность мощности многоуровневого сигнала совпадает со спектральной плотностью мощности бинарного сигнала при замене битового интервала  $T_b$  на символьный интервал  $T_s = T_b \log_2 M$ . Для четырехуровневой модуляции  $M=4$  и, следовательно,  $T_s = 2T_b$ . Расстояние между первыми нулями спектральной плотности мощности сигнала QPSK равно  $Af = 1/T_b$ , что в 2 раза меньше, чем для сигнала BPSK. Другими словами, спектральная эффективность квадратурной модуляции QPSK в 2 раза выше, чем бинарной модуляции BPSK.

### Многопозиционная фазовая модуляция (M-PSK)

M-PSK формируется, как и другие многопозиционные виды модуляции, путем группировки  $k = \log_2 M$  бит в символы и введением взаимно-однозначного соответствия между множеством значений символа и множеством значений сдвига фазы модулированного колебания. Значения сдвига фазы из множества отличаются на одинаковую величину. Для примера на рисунке 1.16 приведено сигнальное созвездие для 8-PSK с кодированием Грея.

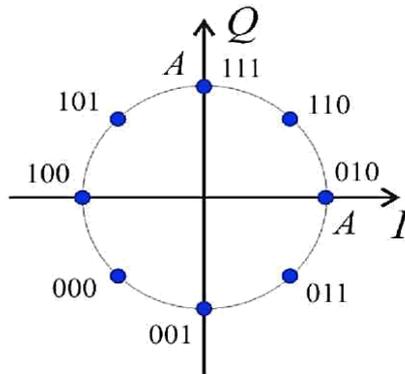


Рис. 1.16. Сигнальное созвездие модуляции 8-PSK

### Амплитудно-фазовые виды модуляции (QAM)

Модуляция, при которой происходит одновременное изменение двух параметров

несущего колебания - амплитуды и фазы - называется амплитудно-фазовой модуляцией.

Минимальный уровень символьных ошибок будет достигнут в случае, если расстояние между соседними точками в сигнальном созвездии будет одинаковым, т. е. распределение точек в созвездии будет равномерным на плоскости. Следовательно, сигнальное созвездие должно иметь решетчатый вид. Модуляция с подобным видом сигнального созвездия называется квадратурной амплитудной модуляцией (QAM - Quadrature Amplitude Modulation). QAM является многопозиционной модуляцией. При  $M=4$  она соответствует QPSK, поэтому формально считается для QAM  $M > 8$  (т.к. число бит на символ  $k = \log_2 M$ ,  $k \in \mathbb{N}$ , то  $M$  может принимать только значения степеней 2: 2, 4, 8, 16 и т.д.). Для примера на рисунке 1.17 приведено сигнальное созвездие 16-QAM с кодированием Грея.

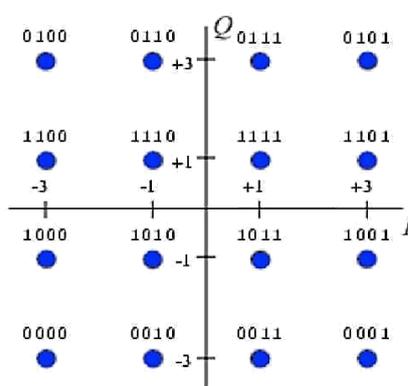


Рис. 1.17. Сигнальное созвездие модуляции 16-QAM

На практике используются большие значения  $M$ , вплоть до 1024-QAM. Такие виды модуляции позволяют достичь исключительно высокой спектральной эффективности. Однако, как видно из сигнального созвездия, так как информация кодируется в том числе амплитудой и изменения амплитуды велики, то QAM предъявляет высокие требования к линейности усилителя мощности и его динамическому диапазону, особенно для больших  $M$ . Практическое осуществление QAM-модуляции выполняется следующим образом. В памяти процессора хранится таблица значений квадратурных компонент  $I(t)$  и  $Q(t)$ , имеющих в сигнальном созвездии и расположенных в порядке возрастания значения соответствующего символа. Процессор анализирует входную последовательность битов, разбивает ее на символы и для каждого символа выбирает соответствующие значения квадратурных компонент из таблицы. Затем выполняется baseband-фильтрация сигналов  $I(t)$  и  $Q(t)$ .

### Сравнение различных видов модуляции

Как указывалось, основными критериями эффективности различных видов модуляции являются критерии спектральной и энергетической эффективности. Энергетическая эффективность характеризует энергию, которую необходимо затратить для передачи информации с заданной достоверностью (вероятностью ошибки). Спектральная

эффективность характеризует полосу частот, необходимую для того, чтобы передавать информацию с определенной скоростью. Кроме данных критериев, виды модуляции сравниваются по устойчивости к различным типам помех и искажений и сложности аппаратной реализации. Увеличение позиций (уровней) модуляции (модуляции M-ASK, M-PSK и M-QAM) увеличивает спектральную эффективность в  $k = \log_2 M$  раз. Также отмечено, что наибольшей спектральной эффективностью среди частотных видов модуляции обладает модуляция MSK. MSK является спектрально в 2.6 раза менее эффективной, чем QPSK и в 1.3 раза менее эффективной, чем BPSK. Сравним виды модуляции по критерию энергетической эффективности. Для этого оценим для каждого вида модуляции требуемую энергию для передачи информации с одинаковой вероятностью ошибки на бит. В таблице приводятся зависимости вероятности ошибки на бит от отношения  $E_b / N_0$  для различных видов модуляции.

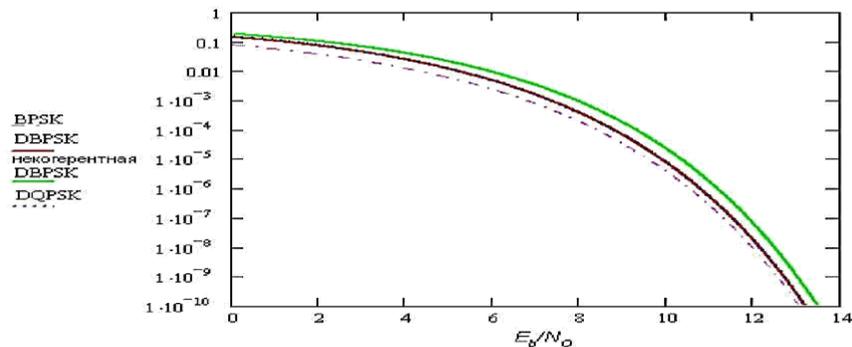


Рис. 1.18. Сравнение энергетической эффективности модуляций с относительным кодированием: DBPSK, DQPSK и некогерентной DBPSK.

Однако BER для BPSK и QPSK описываются одинаковыми формулами, при этом QPSK в 2 раза спектрально эффективнее, чем BPSK. Следовательно, QPSK всегда существенно эффективнее, чем BPSK, и, обыкновенно, имеет смысл использовать QPSK, а не BPSK.

Сравним виды модуляции по критерию энергетической эффективности. Для этого оценим для каждого вида модуляции требуемую энергию для передачи информации с одинаковой вероятностью ошибки на бит. В [2], [10] определены соотношения, связывающие вероятность битовой ошибки с величиной  $E_b / N_0$  для различных видов модуляции:

$$BER = f\left(\frac{E_b}{N_0}\right), \quad (1.14)$$

где  $BER$  - вероятность ошибки  $E_b$  - энергия, необходимая для передачи одного бита информации,  $N_0$  - спектральная плотность мощности белого шума в канале. Если мощность

передатчика равна  $P$ , то величина энергии, приходящаяся на один бит информации, равна  $E_b = PT_b$ , где  $T_b$  - длительность бита. В табл. 1.1 приводятся зависимости вероятности ошибки на бит от отношения  $E_b / N_0$  для различных видов модуляции [1].

Таблица 1.1. Вероятность ошибки для различных видов модуляции

Вид модуляции	Вероятность ошибки на бит (BER)
ООК	$Q(\sqrt{E_b / N_0})$
M-ASK код Грея	$\frac{2(M-1)}{M \log_2 M} Q\left(\sqrt{\frac{\log_2 M}{(M-1)^2} \frac{E_b}{N_0}}\right)$
BPSK	$Q(\sqrt{2E_b / N_0})$
Некогерентная DBPSK	$\frac{1}{2} \exp(-E_b / N_0)$
Когерентная DBPSK	$2Q\left(\sqrt{\frac{2E_b}{N_0}}\right)\left(1 - Q\left(\sqrt{\frac{2E_b}{N_0}}\right)\right)$
QPSK код Грея	$Q(\sqrt{2E_b / N_0})$
Когерентная DQPSK при $E_b / N_0 \gg 1$	$2Q(\sqrt{2E_b / N_0})$
M-PSK код Грея	$\frac{2}{\log_2 M} Q\left(\sqrt{\frac{2E_b \log_2 M}{N_0} \sin^2\left(\frac{\pi}{M}\right)}\right)$
FSK	$Q\left(\sqrt{\left[1 - \frac{\sin(2\pi \cdot m)}{2\pi \cdot m}\right] \frac{E_b}{N_0}}\right)$
MSK	$Q(\sqrt{E_b / N_0})$
M-MSK	$\frac{2(M-1)}{M \log_2 M} Q\left\{\sqrt{\log_2 M \frac{E_b}{N_0}}\right\}$
QAM код Грея	<p>для <math>k = \log_2 M</math>, <math>k</math> - четное:  <math>BER = \frac{2P_0 - P_0^2}{\log_2 M}</math>, где  <math>P_0 = \frac{2(\sqrt{M}-1)}{\sqrt{M}} Q\left(\sqrt{\frac{3 \log_2 M}{M-1} \frac{E_b}{N_0}}\right)</math></p> <p>для нечетных <math>k</math>:  <math>BER \leq \frac{1}{\log_2 M} \left[1 - \left(1 - 2Q\left(\sqrt{\frac{3 \log_2 M}{M-1} \frac{E_b}{N_0}}\right)\right)^2\right]</math></p>

В табл.  $Q(x) = \frac{1}{\sqrt{2\pi}} \int_x^\infty e^{-t^2/2} dt$  - интеграл ошибок (интеграл вероятности),

$M$  - число позиций для многопозиционных видов модуляции,  $m$  - индекс модуляции для частотной модуляции,  $BER$  - вероятность ошибки на бит.

Из табл. видно, что с увеличением позиционности модуляции, вероятность битовой ошибки увеличивается (см., например, формулы M-ASK и M-PSK,  $Q(x)$  является убывающей функцией аргумента). Таким образом, как правило, *при увеличении спектральной эффективности энергетическая эффективность уменьшается.*

Однако BER для BPSK и QPSK описываются одинаковыми формулами, при этом QPSK в 2 раза спектрально эффективнее, чем BPSK. Следовательно, *QPSK всегда существенно эффективнее, чем BPSK, и, обыкновенно, имеет смысл использовать QPSK, а не BPSK.* Физически, это объясняется тем, что в случае QPSK добавляется дополнительная степень свободы: квадратурная составляющая  $Q(t)$ . В случае BPSK используется только синфазная составляющая  $I(t)$ . Квадратурная форма когерентного фазового демодулятора приводит к тому, что два канала детектора обеспечивают *независимый* прием двух бинарных фазомодулированных сигналов. Аналогичное явление имеет место и при сравнении модуляций DBPSK и DQPSK (с относительным кодированием). Хотя выражения для BER несколько отличаются, с высокой степенью приближения они совпадают (рис.24). Модуляции с относительным кодированием имеют небольшой энергетический проигрыш по сравнению с обыкновенными BPSK и QPSK (0.3 - 0.9 дБ). DBPSK с некогерентным детектированием также имеет небольшой проигрыш по сравнению с DBPSK с когерентным детектированием (около 0.5 дБ), рис. 1.20. Под энергетическим выигрышем понимается разница в значении  $E_b \cdot N_0$  при одинаковом значении вероятности ошибки на бит.

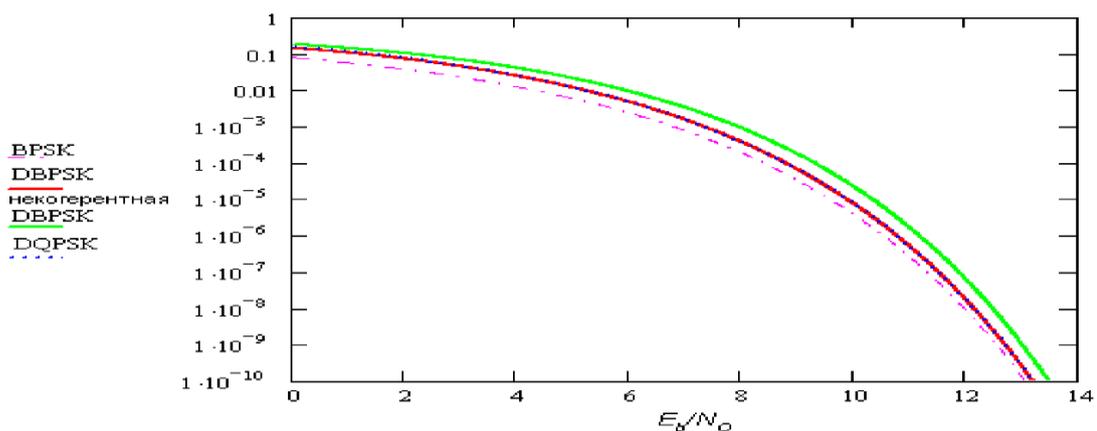


Рис. 1.20. Сравнение энергетической эффективности модуляций BPSK, DBPSK, DBPSK и DQPSK

Таким образом, имеет смысл сравнивать виды модуляции с одинаковым числом позиций. Сравним двухуровневые OOK, BPSK и MSK. Как видно из табл., OOK и MSK

имеют одинаковую эффективность и уступают BPSK (и, соответственно, QPSK) по энергетической эффективности приблизительно 3 дБ (рис.1.21).

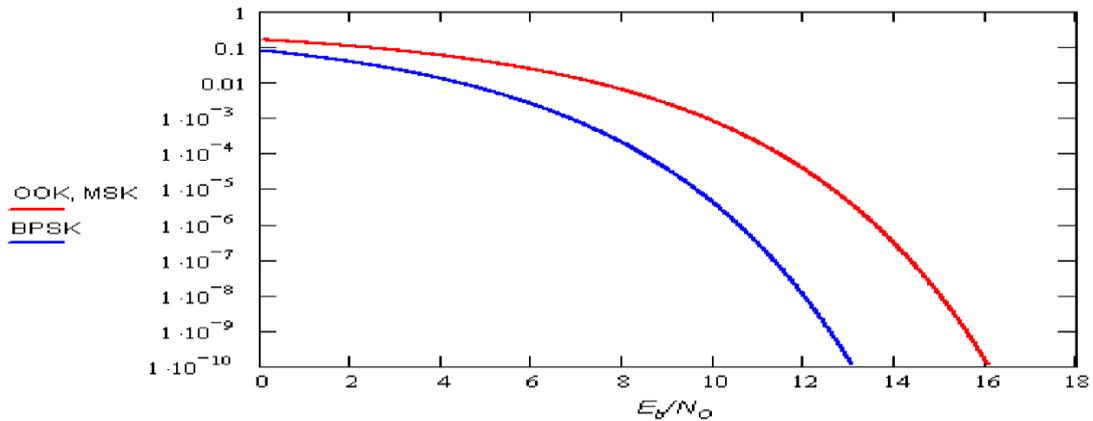


Рис. 1.21. Сравнение энергетической эффективности модуляций OOK, MSK и BPSK

По результатам данного сравнения можно сделать вывод о том, что при числе уровней до 4 включительно QPSK является спектрально и энергетически наиболее эффективным видом модуляции. Однако здесь следует сделать одно существенное замечание относительно модуляции GMSK. Ее спектральная эффективность ниже, чем QPSK, в системах с *линейным усилением*. GMSK, как частотный вид модуляции, позволяет использовать высокоэффективные нелинейные усилители и ограничители, что дает энергетический выигрыш. При прохождении QPSK через подобные устройства, ее спектр расширяется (происходит некоторое восстановление боковых лепестков). Поэтому, в некоторых случаях, GMSK может иметь большую эффективность, чем QPSK. В частности в стандарте GSM выбор сделан в пользу GMSK, а в CDMA - OQPSK. Однако, усовершенствованные виды модуляции QPSK (например, FQPSK) в любом случае превосходят GMSK.

Сравним теперь модуляции с числом уровней  $M > 4$ . На рис. 1.22 изображено сравнение энергетической эффективности для амплитудной, фазовой и амплитудно-фазовой манипуляции при  $M=16$  и  $M=64$ .

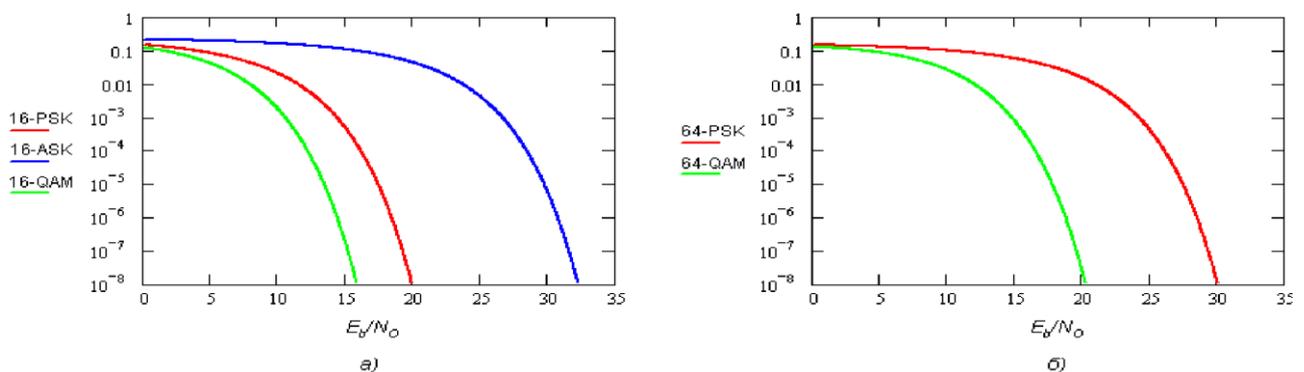


Рис. 1.22. Сравнение энергетической эффективности модуляций M-ASK, M-PSK и M-QAM: а - M=16, б - M=64

Как видно из рис. 1.22, амплитудная модуляция существенно (более 10 дБ при  $M=16$ ) уступает фазовой и амплитудно-фазовой, поэтому при  $M=64$  сравнение с ней не проводится. При сравнении M-PSK с M-QAM видно, что M-QAM превосходит по эффективности M-PSK, причем энергетический выигрыш M-QAM увеличивается с ростом  $M$ . Например, для  $M=16$  выигрыш составляет около 4 дБ, а при  $M=64$  около 10 дБ. Физически это объясняется тем, что расстояние между соседними точками в сигнальном созвездии M-PSK меньше, чем M-QAM. Сигнальное созвездие M-PSK представляет собой окружность с равномерно распределенными на ней точками, а созвездие M-QAM - квадрат с равномерно распределенными по его площади точками. Чем больше расстояние между точками в созвездии, тем менее вероятна ошибка в детектировании соседнего символа. Многопозиционная частотная модуляция используется гораздо реже, так как *при увеличении числа уровней и сохранении индекса модуляции ее спектр не сужается, а расширяется*, ввиду того, что вводятся новые частоты и ширина спектра растет по закону  $\frac{M}{\log_2 M}$ . Как видно из табл. 1.1, однако, *при увеличении числа уровней M-MSK, в отличие от всех других видов модуляции, вероятность ошибки на бит уменьшается*. Мы получаем выигрыш в энергетической эффективности за счет уменьшения спектральной эффективности.

Таким образом, при ограниченной полосе, при  $M < 4$  наиболее эффективной является модуляция QPSK, а при  $M > 4$  - QAM. QPSK является частным случаем QAM при  $M=4$ . Можно считать QAM наиболее эффективным видом модуляции при любом числе уровней. Еще больший выигрыш по сравнению с обыкновенными QPSK и QAM дают их усовершенствованные модификации, такие, как модификации Феера [11] (FQPSK, FQAM), модуляция с решетчатым кодированием (TCM), оптимизация формы сигнальных созвездий и использование многомерных сигнальных созвездий.

## **1.2. Модемы сотовой связи FSK, MSK GMSK и численный анализ вероятности символьной ошибки с использованием ПО LabVIEW**

### **Частотная манипуляция (FSK)**

Значениям «0» и «1» информационной последовательности соответствуют определённые частоты синусоидального сигнала при неизменной амплитуде. Частотная манипуляция весьма помехоустойчива, поскольку помехи телефонного канала искажают в основном амплитуду, а не частоту сигнала. Однако при частотной манипуляции неэкономно расходуется ресурс полосы частот телефонного канала. Поэтому этот вид модуляции

применяется в низкоскоростных протоколах, позволяющих осуществлять связь по каналам с низким отношением сигнал/шум.

Существует также подвид этой модуляции GFSK. Принцип работы модулятора GFSK похож на FSK, за исключением того, что сначала полоса импульсов (-1, 1) проходит через фильтр Гаусса для сглаживания, что обеспечивает уменьшения ширины его спектра, а уже после попадает в FSK. Фильтрация Гаусса — один из самых распространенных способов уменьшения ширины спектра.

На рисунке ниже приведен график двоичной бинарной последовательности нулей и единиц и, соответствующий ему, график частотно-манипулированного сигнала. Низкому уровню бинарного двоичного сигнала соответствует частота 1 КГц, а высокому - частота 0,5 КГц несущего сигнала синусоидального типа.

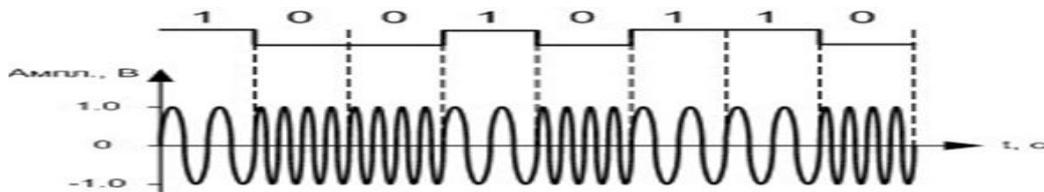


Рис. 1.23. Частотная манипуляция

В программной комплексе можно осуществить многопозиционную частотную модуляцию (MFSK) задав в поле M-FSK нужный уровень от 2 до 64. В бинарной FSK модуляции, т.е. при  $M=2$ , два двоичных числа представляются сигналами двух различных частот, расположенных около несущей. Бинарная частотная модуляция менее восприимчива к ошибкам, чем амплитудная модуляция.

Более эффективной, но и более подверженной ошибкам, является схема многочастотной модуляции (Multiple FSK - MFSK), в которой используется более двух частот. В этом случае каждая сигнальная посылка представляет более одного бита. Переданный сигнал MFSK (для одного периода передачи сигнальной посылки) можно определить следующим образом:

$$s_i = A \cos(2\pi f_i t), \quad 1 \ll i \ll M$$

Здесь

$$f_i = f_c + (2i - 1 - M)f_d,$$

где  $f_c$  - несущая частота;  $f_d$  - разностная частота;  $M$  - число различных сигнальных

посылок  $= 2^L$ ;  $L$  - количество битов на одну сигнальную посылку.

На рис. 1.24 представлен пример схемы MFSK с  $M=4$ . Входной поток битов кодируется по два бита, после чего передается одна из четырех возможных двухбитовых комбинаций.

Для уменьшения занимаемой полосы частот в модуляторах сигналов с фазовой модуляцией применяют сглаживающие фильтры. Применение сглаживающих фильтров приводит к увеличению эффективности использования полосы, но в то же время из-за сглаживания уменьшается расстояние между соседними сигналами, что приводит к снижению помехоустойчивости.

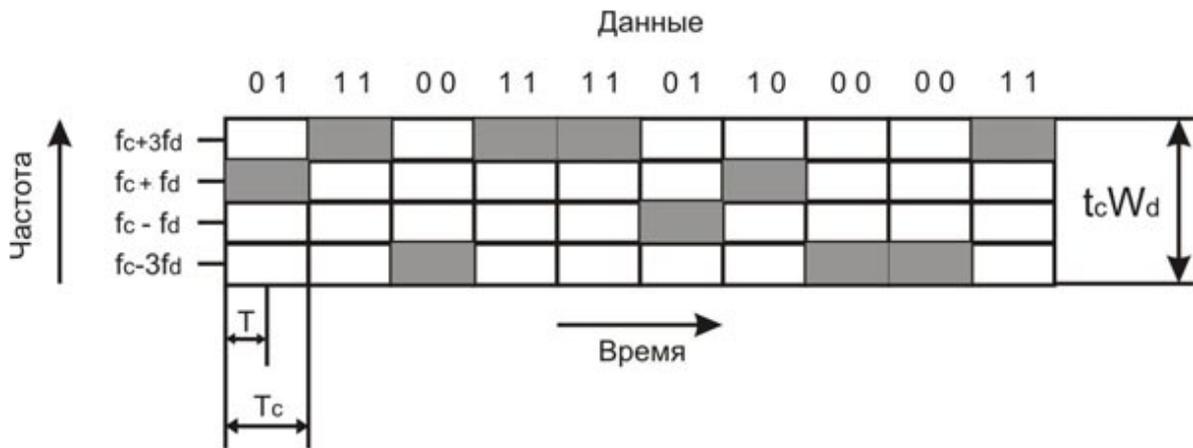


Рис. 1.24. Использование частоты схемой MFSK ( $M = 4$ )

Построение графиков глазковой диаграммы и ее вид дает много информации о джиттере сигнала, так же, как и о многих других его параметрах. Например, множество отдельных фронтов и спадов говорит о вероятном присутствии джиттера, зависящего от данных.

Глазковая диаграмма не просто предоставляет множество информации, она удобна простотой применения и тем, что может применяться для измерений в любой цепи с реальными данными.



Рис. 1.25. Параметры глазковой диаграммы

По индикаторной диаграмме можно выполнить ряд важных измерений:

- чем больше открыт глазок, тем легче различать логические 1 и 0;
- ширина открытия глазковой диаграммы (время между пересечениями линий логической 1 с логическим 0 и логического 0, с логической 1) показывает временной интервал, в течение которого сигнал может быть замерен без ошибки из-за межсимвольного влияния;
  - псевдослучайной последовательности битов и отображения сигналов на запоминающем осциллографе получается структура, которая называется индикаторной (глазковой) диаграммой (eye diagram). Типичная индикаторная диаграмма приведена на рис. 3.
  - высота открытия глазка измеряет запас помехоустойчивости на выходе приемника;
  - ширина линий глазка к точкам пересечения в углах глазка является мерой флуктуации в системе передачи. Флуктуации вызываются разбросом времени включения и выключения лазера; искажением импульса оптическим волокном и шумом. Флуктуации выражаются в пикосекундах, градусах или в процентах интервала бита;
  - толщина линий импульса наверху и внизу глазка пропорциональна шуму и искажениям в системе передачи; время перехода сигнала в схеме глазка с верхнего уровня (логического 0) в нижний (логическая 1) и наоборот указывает времена подъема и спада системы передачи. Они обычно замеряются между отметками 10 и 90%;
  - времена подъема и спада важны для оценки чувствительности системы к синхронизации замеров (sample timing). Чем больше времена подъема и спада сигнала, тем более чувствительна система к ошибкам синхронизации;
  - чтобы обеспечить Системе максимальную невосприимчивость к шуму, лучшим временем для замеров уровня сигнала является время, когда высота открытия индикаторной диаграммы максимальна.

Джиттер или фазовое дрожание цифрового сигнала данных — нежелательные фазовые и/или частотные случайные отклонения передаваемого сигнала. Возникают вследствие нестабильности задающего генератора, изменений параметров линии передачи во времени и различной скорости распространения частотных составляющих одного и того же сигнала.

В цифровых системах проявляется в виде случайных быстрых изменений местоположения фронтов цифрового сигнала во времени, что приводит к рассинхронизации и, как следствие, искажению передаваемой информации. Например, если фронт имеет малую крутизну или «отстал» по времени, то цифровой сигнал как бы запаздывает, сдвигается относительно значащего момента времени — момента времени, в который происходит оценка сигнала.

Джиттер является одной из основных проблем при проектировании устройств цифровой электроники, в частности, цифровых интерфейсов. Недостаточно аккуратный расчет джиттера может привести к его накоплению при прохождении цифрового сигнала по тракту и, в конечном счёте, к неработоспособности устройства.

Причины возникновения джиттера:

- Фазовые шумы петли ФАПЧ (фазовой автоподстройки частоты) устройства, синхронизируемого внешним сигналом. Джиттер, вызываемый ФАПЧ, проявляется при прослушивании материала с записывающего устройства, синхронизируемого от воспроизводящего устройства.
- АЦП. В современных цифровых системах звукозаписи и воспроизведения основным источником джиттера является АЦП. Нынешние полностью цифровые студийные синхронизаторы достаточно совершенны и часто вносят джиттер меньший, чем АЦП.

### Описание прибора с FSK модуляцией (манипуляцией)

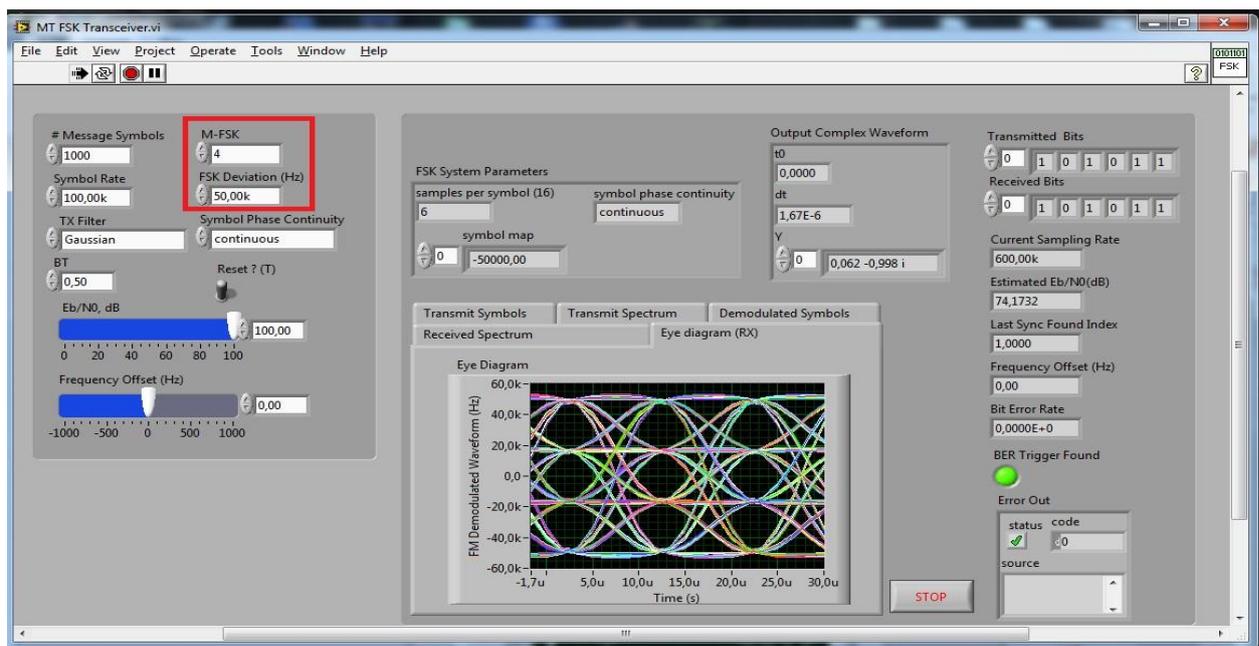


Рис. 1.26. Интерфейс прибора с FSK модуляцией

- Message Symbols – показывает количество передаваемых символов.
- M-FSK – показывает уровень модуляции. Доступны режимы: 2,4,8,16,32,64
- FSK Deviation (Гц) – отклонение по частоте частотной модуляции.
- Eb/N0- отношение сигнал/шум
- Frequency offset –задержка по частоте
- TX Filter – выбор фильтра для передатчика. Доступны следующие: Gaussian, Raised Cosine, Root Raised Cosine

- Symbol Rate – порядок символов.
- Symbol Phase Continuity – продолжительность фазы импульса. Соответственно режимы Continuous (Конечная), Discontinuous (Бесконечная).
- Transmitted Bits – переданные биты.
- Received Bits – принятые биты.
- BER Trigger Found – Вероятность ошибки
- $BT$  - безразмерная величина равная  $BT = B_{-3дБ} * T$ , где  $B_{-3дБ}$  - полоса фильтра Гаусса по уровню -3дБ,  $T = 1/B_r$  - длительность единичного импульса цифровой информации, передаваемой со скоростью  $B_r$  бит\с. Например  $B_r = 20 \text{ кбит/с}$ , тогда  $T = 50 \text{ мкс}$  и при полосе фильтра Гаусса по уровню -3 дБ  $B_{-3дБ} = 10 \text{ кГц}$  получаем  $BT = 10 \text{ кГц} * 50 \text{ мкс} = 0.5$ . Таким образом, параметр  $BT$  показывает во сколько раз полоса фильтра Гаусса  $B_{-3дБ}$  отличается от скорости передачи информации  $B_r$ , выраженной в единицах измерения частоты.

- Сокращения: TX – Передатчик, RX – Приемник

### Исследование линии передачи с FSK модуляцией

#### Предварительная фильтрация

Сравним спектры сигналов и их помехоустойчивость при использовании фильтра Гаусса и без фильтра

Без фильтра при соотношении сигнал/шум = 80

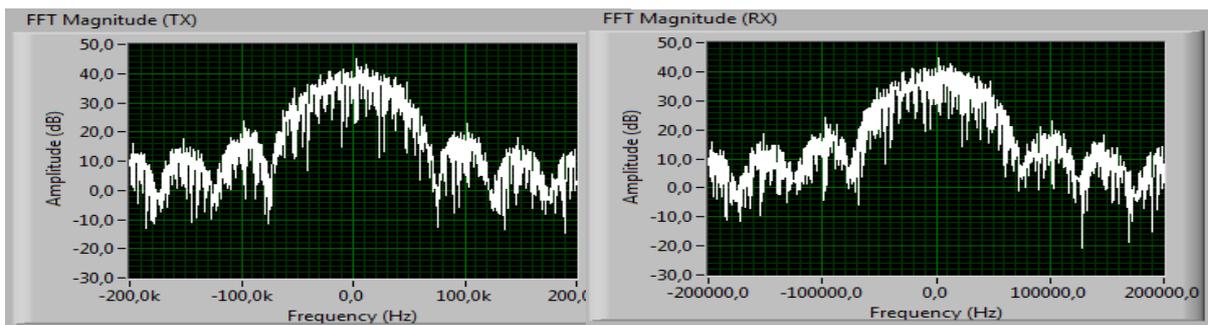


Рис. 1.27. Спектр переданного (слева) и принимаемого (справа) сигнала

Как видно из рисунка 1.27, спектр почти не изменился, но в нем присутствует высокий уровень боковых лепестков, что может повлиять на соседние каналы, также придется выделить более широкий канал в линии передачи, что не всегда является позволительным и выгодным.

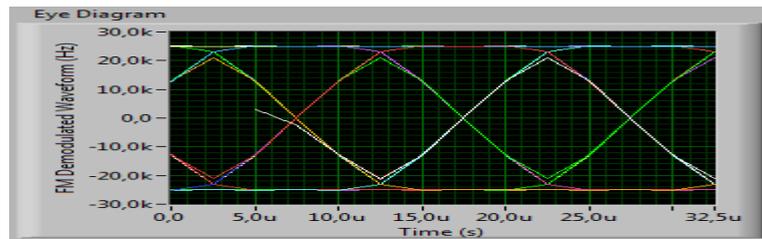


Рис. 1.28. Глазковая диаграмма

Без использования фильтра в сигнал не вносятся дополнительные искажения следовательно джиттер и помехи минимальны.

Применение фильтра Гаусса. При соотношении сигнал/шум = 80,  $BT = 0,5$

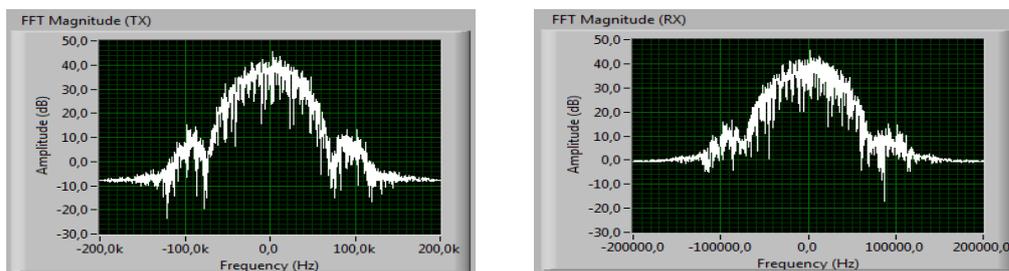


Рис. 1.29. Спектр переданного (слева) и принимаемого (справа) сигнала

Сужение полосы занимаемых частот удалось достигнуть за счет предварительной фильтрации модулирующего сигнала фильтром низкой частоты с Гауссовской импульсной характеристикой.

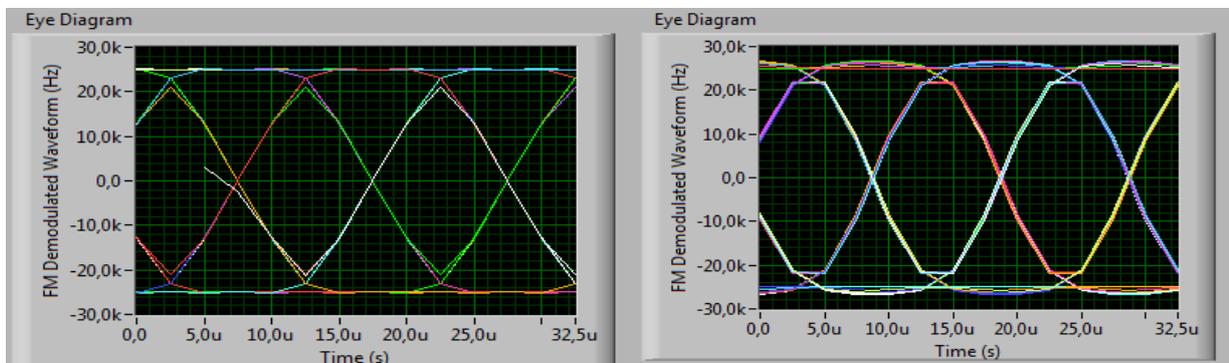


Рис. 1.30. Глазковая диаграмма без фильтра (слева) и с фильтром Гаусса (справа)

Сравнивая две глазковые диаграммы без использования фильтра и с фильтром Гаусса видно, что помехоустойчивость снизилась, появились помехи и искажения, а также увеличился джиттер.

## Графики

Фильтр raisedcosine : M-fsk 4 (красный) и 2 (синий)

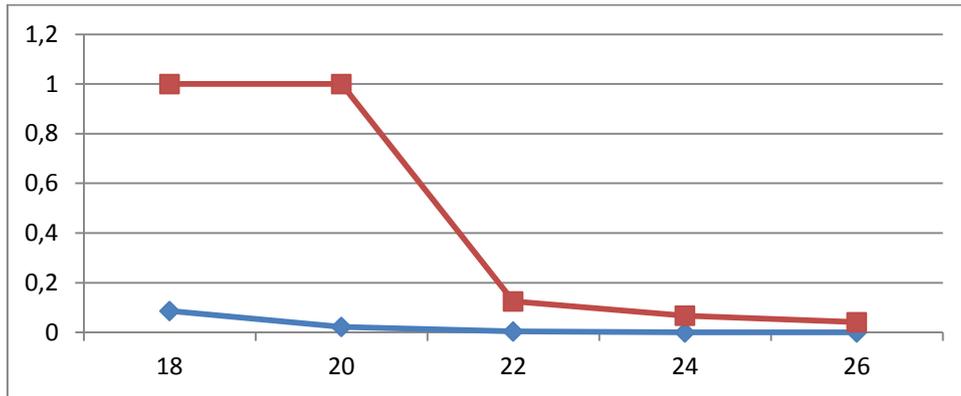


Рис. 1.31. Зависимость вероятности ошибки от отношения сигнал/шум

Фильтр rootraisedcos: M-fsk2 (красный) и 4 (синий)

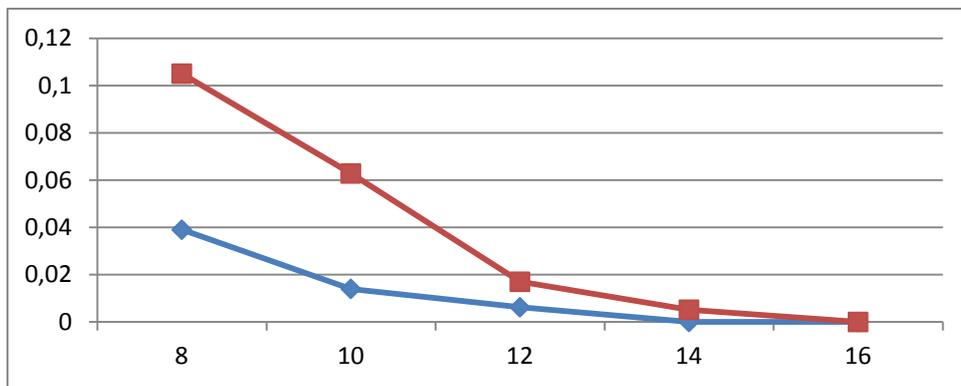


Рис. 1.32. Зависимость вероятности ошибки от отношения сигнал/шум

Фильтр Gaussian: M-fsk2 (красный) и 4 (синий)

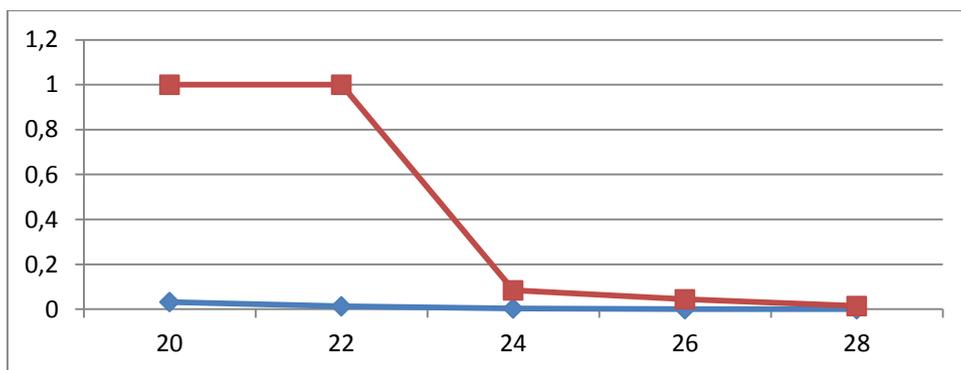


Рис. 1.33. Зависимость вероятности ошибки от отношения сигнал/шум

### Частотная манипуляция с минимальным сдвигом (MSK)

MSK это частный случай сигналов с частотной манипуляцией с непрерывной фазой CPFSK с минимальным индексом частотной манипуляции  $m=0.5$ , поэтому ширина главного

лепестка спектра MSK сигналов - минимальная из всех возможных сигналов с частотной манипуляцией.

Представляет собой способ модуляции, при котором не происходит скачков фазы и изменение частоты происходит в моменты пересечения несущей нулевого уровня. MSK уникальна потому, что значение частот соответствующих логическим «0» и «1» отличаются на величину равную половине скорости передачи данных. Другими словами, индекс модуляции равен 0,5.

Частотная модуляция с минимальным разносом частот MSK позволяет уменьшить ширину полосы частот, занимаемых цифровым радиосигналом в эфире. Однако даже этот вид модуляции не удовлетворяет всем требованиям, предъявляемым к современным радиосистемам мобильной связи. Обычно сигнал MSK в радиопередатчике дофильтровывают обычным фильтром. Именно поэтому появился еще один вид модуляции с еще более узким спектром радиочастот в эфире.

Сужение полосы занимаемых частот удалось достигнуть за счет предварительной фильтрации модулирующего сигнала фильтром низкой частоты с Гауссовской импульсной характеристикой. Ширина спектра сигнала GMSK определяется произведением длительности передаваемого символа на полосу пропускания Гауссовского фильтра ВТ. Именно полосой пропускания В и отличаются различные виды GMSK друг от друга. GMSK используется в стандарте GSM сотовой связи.

Формирование GMSK радиосигнала осуществляется таким образом, что на интервале одного информационного бита фаза несущей изменяется на  $90^\circ$ . Это наименьшее возможное изменение фазы, распознаваемое при данном типе модуляции. Непрерывное изменение фазы синусоидального сигнала дает в результате частотную модуляцию с дискретным изменением частоты. Применение фильтра Гаусса позволяет при дискретном изменении частоты получить "гладкие переходы". В стандарте GSM применяется GMSK-модуляция с величиной нормированной полосы ВТ - 0,3, где В - ширина полосы фильтра по уровню минус 3 дБ, Т - длительность одного бита цифрового сообщения. Основой формирователя GMSK-сигнала является квадратурный (1/Q) модулятор. Схема состоит из двух умножителей и одного сумматора. Задача этой схемы заключается в том, чтобы обеспечить непрерывную, очень точную фазовую модуляцию. Один умножитель изменяет амплитуду синусоидального, а второй косинусоидального колебания. Входной сигнал до умножителя разбивается на две квадратурные составляющие. Разложение происходит в двух обозначенных "sin" и "cos" блоках.

Модуляцию GMSK отличают следующие свойства, которые предпочтительны для

подвижной связи:

- постоянная по уровню огибающая, которая позволяет использовать эффективные
- передающие устройства с усилителями мощности в режиме класса С;
- компактный спектр на выходе усилителя мощности передающего устройства,
- обеспечивающий низкий уровень внеполосного излучения;
- хорошие характеристики помехоустойчивости канала связи.

Импульсная характеристика Гауссовского фильтра описывается следующей формулой:

$$h(t) = B \sqrt{\frac{2\pi}{\ln(2)} e^{-\frac{(B_t \pi)^2}{\ln(2)}}},$$

где  $B$  — полоса пропускания фильтра по уровню 3 дБ.

GMSK с фильтром Гаусса с  $BT = 1$ , вырождается в классическую MSK. Для формирования сигналов GSM стандартно используется фильтр Гаусса с  $BT = 0.3$ .

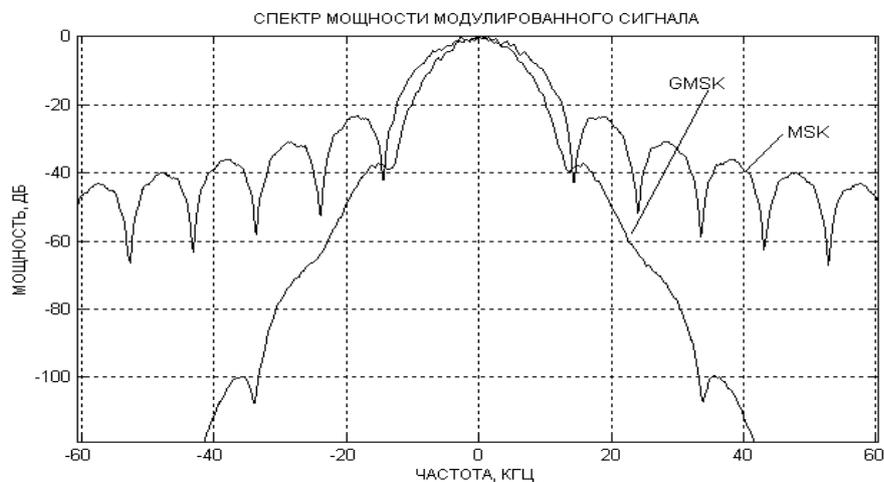


Рис. 1.34. Сравнение спектров двух видов модуляции с фильтром Гаусса(GMSK)

и без фильтра(MSK)

В точках взятия отсчетов сигнал GMSK зависит от предыдущих значений передаваемого сигнала. Это вызвано действием гауссовского фильтра, формирующего спектр сигнала GMSK. В результате помехоустойчивость сигнала GMSK ниже по сравнению даже с помехоустойчивостью сигнала MSK. Конкретное значение помехоустойчивости сигналов GMSK сильно зависит от произведения  $BT$ . Пример зависимости вероятности ошибки

приема сигнала GMSK в зависимости от отношения сигнал/шум на входе решающего устройства приведен на рисунке 1.35.

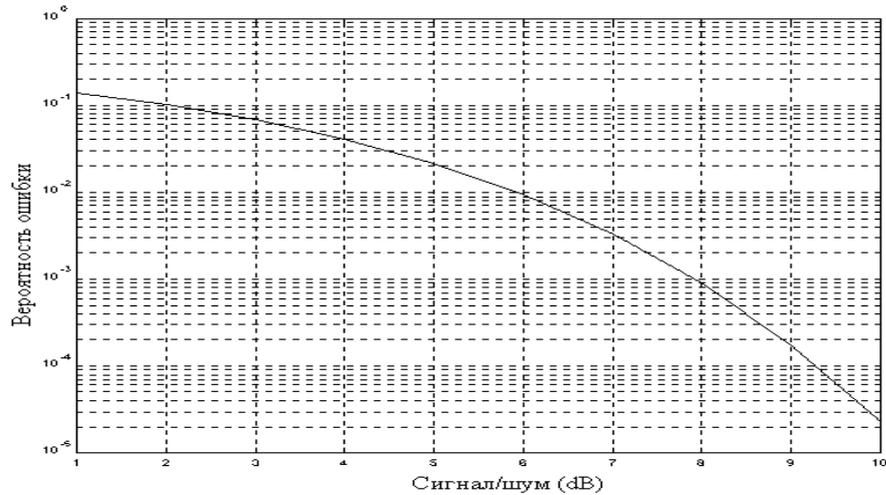


Рис. 1.35. График зависимости вероятности ошибки GMSK в зависимости от отношения сигнал/шум на входе решающего устройства

При индексе модуляции  $m = 0.5$  за время передачи одного символа фаза несущего колебания успевает измениться на угол  $\pm 90^\circ$ . Решетка переходов фазы в MSK сигнале на протяжении двух символьных периодов приведена на рисунке 1.36.

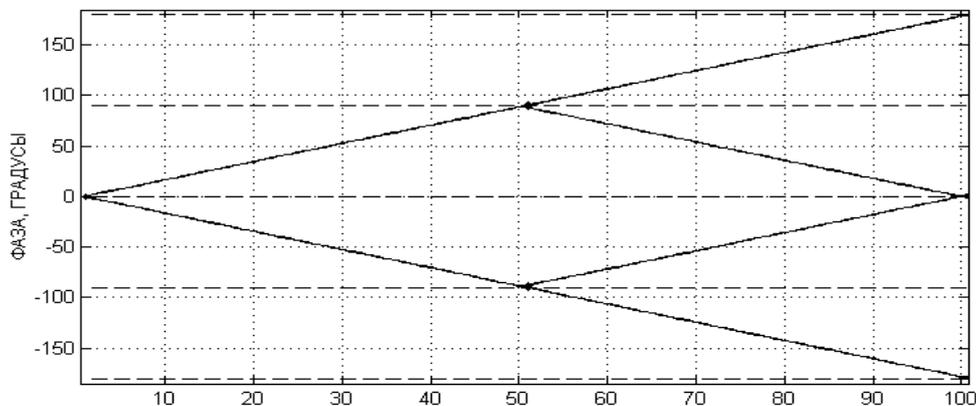


Рис. 1.36. Решетка переходов фазы в MSK сигнале

Два возможных значения фазы несущего колебания в отсчетной точке на одном временном интервале отличаются от двух возможных значений фазы несущего колебания отсчетной точке на соседнем интервале на  $90^\circ$ . Рассмотренная ситуация может быть проиллюстрирована на векторной диаграмме, приведенной на рисунке 1.37.

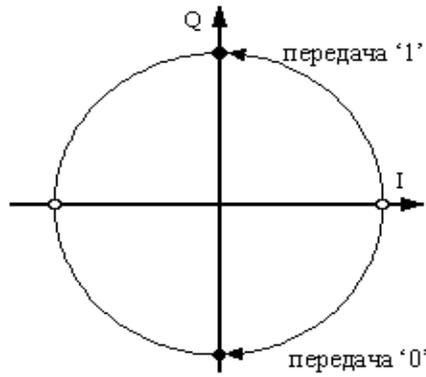


Рис. 1.37. Решетка переходов фазы в MSK сигнале.

Конец вектора, отражающий амплитуду несущего колебания на рисунке 4, в сигнале частотной модуляции может двигаться строго по окружности. На этом рисунке пара состояний сигнала показана пустыми кружочками, а пара состояний сигнала на соседнем символе — заполненными.

Рассмотренные диаграммы показывают, что при разработке радиоприемного устройства можно применить схему фазового детектора. Принимаемые двоичные символы в отсчетных точках будут отличаться друг от друга по фазе на  $180^\circ$ .

### Описание прибора с MSK модуляцией (манипуляцией)

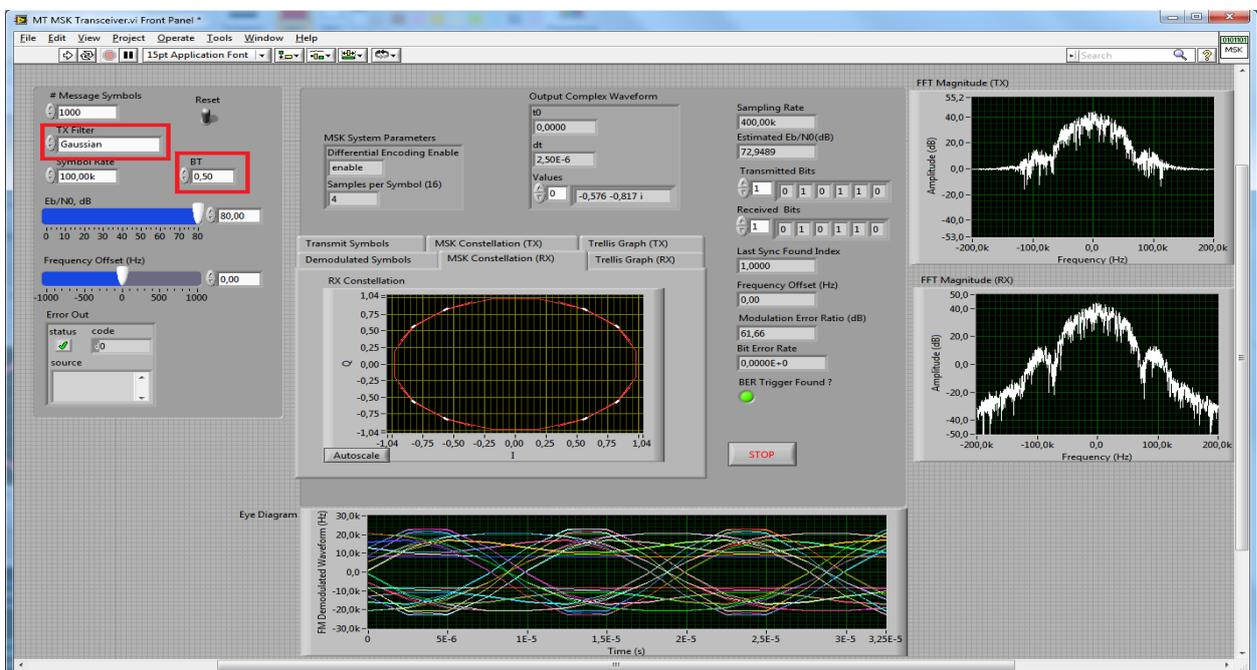


Рис. 1.38. Интерфейс программы с GMSK модуляцией

- Message Symbols – показывает количество передаваемых символов.
- $E_b/N_0$ - отношение сигнал/шум
- Frequency offset –задержка по частоте

- TX Filter – выбор фильтра для передатчика. Доступны следующие: Gaussian, Raised Cosine, Root Raised Cosine
- Symbol Rate – порядок символов.
- Transmitted Bits – переданные биты.
- Received Bits – принятые биты.
- BER Trigger Found – Вероятность шибки
- BT - безразмерная величина равная  $BT = 0,5$
- Сокращения: TX – Передатчик, RX – Приемник

### Исследование линии передачи с MSK модуляцией

В качестве характеристического параметра *GMSK* используют произведение  $BT$ , где  $B$  - ширина спектра импульса  $h(t)$  по уровню 3 дБ, а  $T$  - длительность одного бита. На рис. 1.39 изображена спектральная плотность мощности на выходе идеального

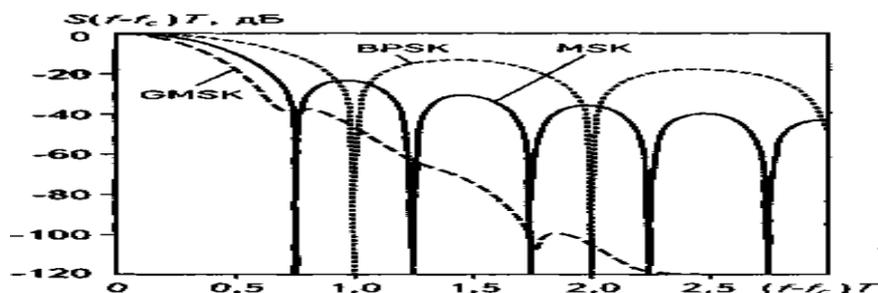


Рис. 1.39. Спектр сигнала с GMSK модуляцией

*GMSK*-модулятора ( $BT = 0,3$ ), нормализованного по отношению к периоду  $T$ . Сравнение этого графика со спектральной плотностью мощности *MSK* и *BPSK*, указывает на серьезное преимущество этой модуляции, прежде всего - в части скорости спада внеполосного излучения, т.е. скорости снижения уровня мощности боковых спектральных лепестков.

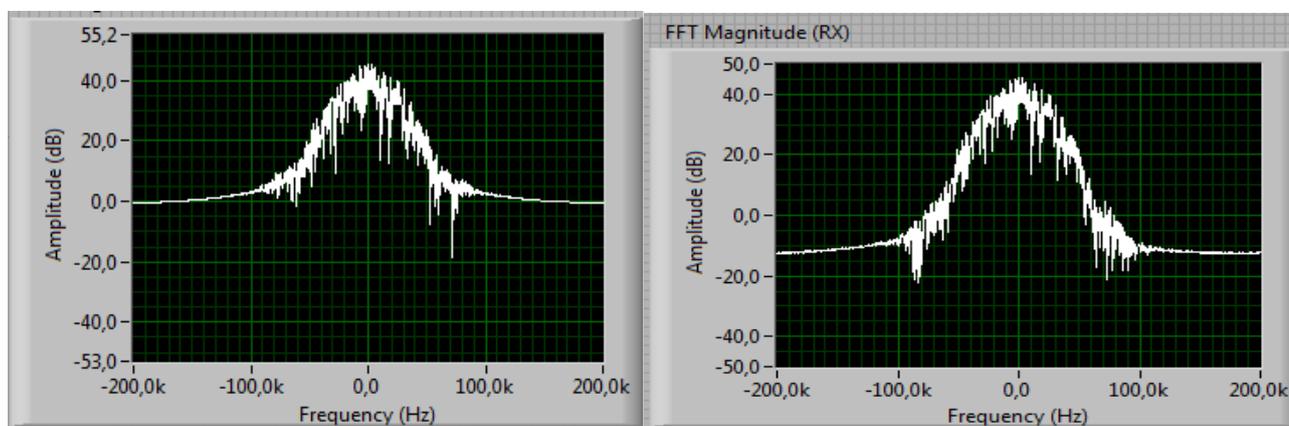


Рис. 1.40. Спектр передаваемого (слева) и принимаемого (справа) сигнала с GMSK модуляцией при  $BT = 0,2$ , сигнал/шум = 80

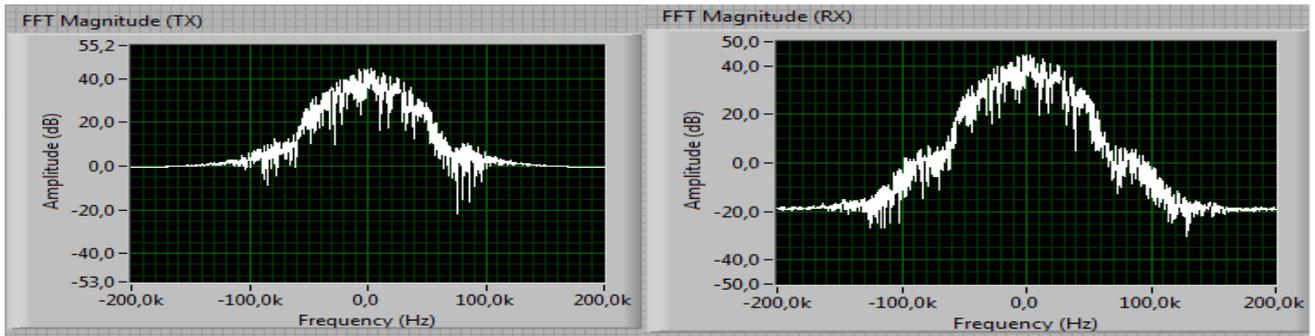


Рис. 1.41. Спектр передаваемого (слева) и принимаемого (справа) сигнала с GMSK модуляцией при  $BT = 0.3$ , сигнал/шум = 80

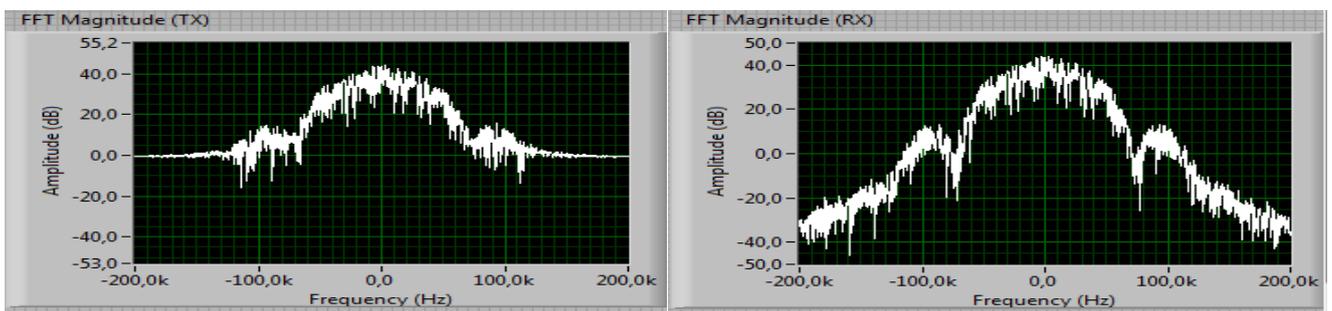


Рис. 1.42. Спектр передаваемого (слева) и принимаемого (справа) сигнала с GMSK модуляцией при  $BT = 0.5$ , сигнал/шум = 80

Можно видеть, что в спектре GMSK при уменьшении  $BT$  уменьшаются уровни боковых лепестков, кроме того значительно возрастает скорость убывания спектра. Так максимальный уровень бокового лепестка GMSK при  $BT = 0.3$  меньше, чем при MSK модуляции, а скорость убывания линейно зависит от частоты, что обусловлено применением

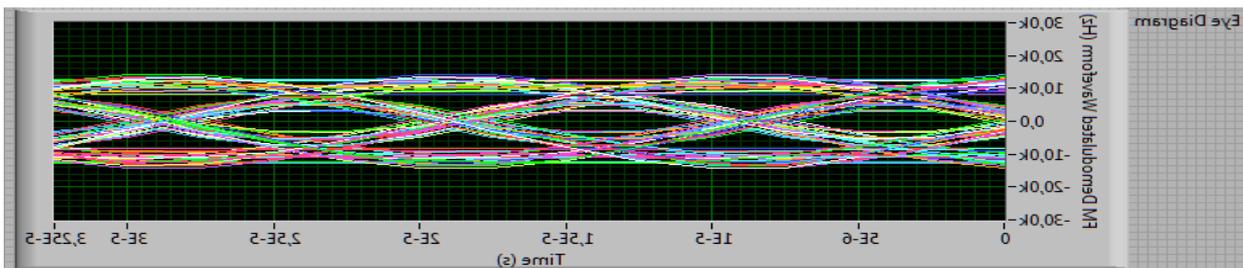


Рис. 1.43. Глазковая диаграмма сигнала с GMSK модуляцией при  $BT = 0.2$ , сигнал/шум = 80

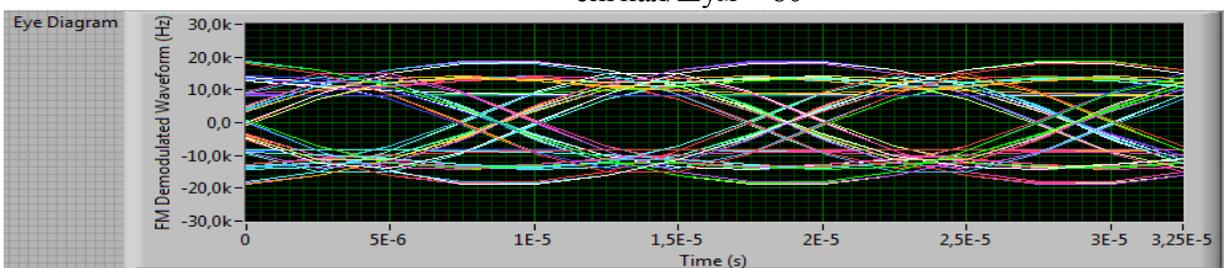


Рис. 1.44. Глазковая диаграмма сигнала с GMSK модуляцией при  $BT = 0.3$ ,  
сигнал/шум = 80

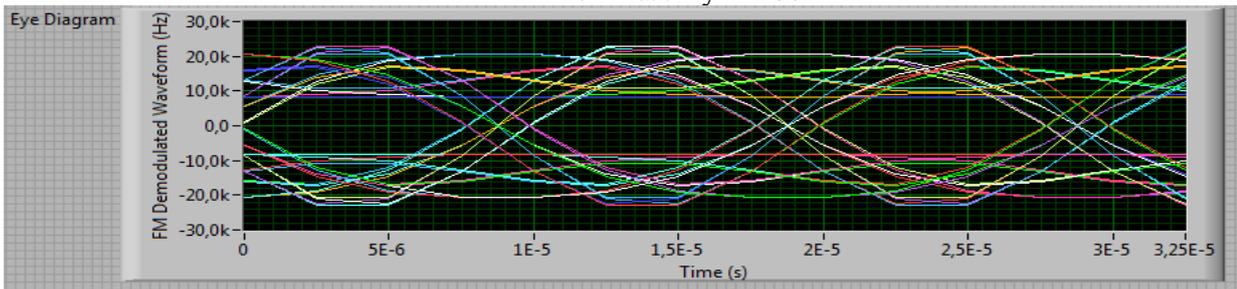


Рис. 1.45. Глазковая диаграмма сигнала с GMSK модуляцией при  $BT = 0.5$ ,  
сигнал/шум = 80

Из рисунков отчетливо видно, что с уменьшением межсимвольная интерференция усиливается ввиду расширения фильтра Гаусса. Фильтр Гаусса вносит межсимвольную интерференцию и позволяет снизить уровень боковых лепестков спектра, а также значительно увеличивает скорость убывания спектра GMSK сигнала по сравнению с MSK сигналами. На практике GMSK модуляция нашла применение в сотовой связи формата GSM при , ввиду наиболее эффективного использования радиочастотного ресурса. Дальнейшее уменьшение параметра приводит к существенному усложнению аппаратуры и сильно увеличивает вероятность ошибочного приема символа из-за недопустимо высокой межсимвольной интерференции.

Из полученных графиков можно сделать вывод что для обеспечения той же самой вероятности ошибки требуется различное отношение сигнал\шум, т.е. при использовании фильтра Гаусса требования к качеству принимаемого сигнала возрастают.

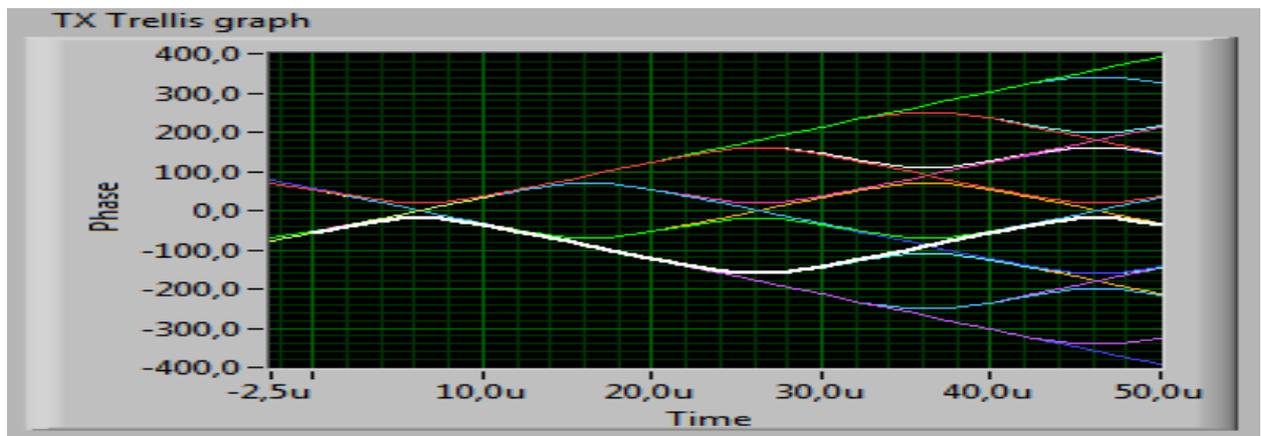


Рис. 1.46. Фазовая диаграмма передаваемого сигнала

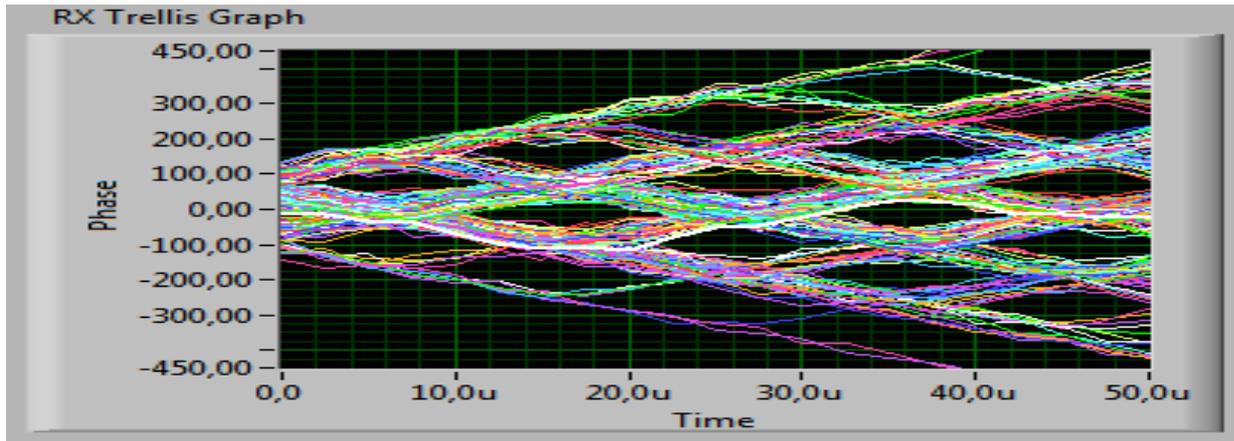


Рис. 1.47. Фазовая диаграмма полученного сигнала при  $BER=2.04 \cdot 10^{-3}$ ,  
сигнал/шум = 14

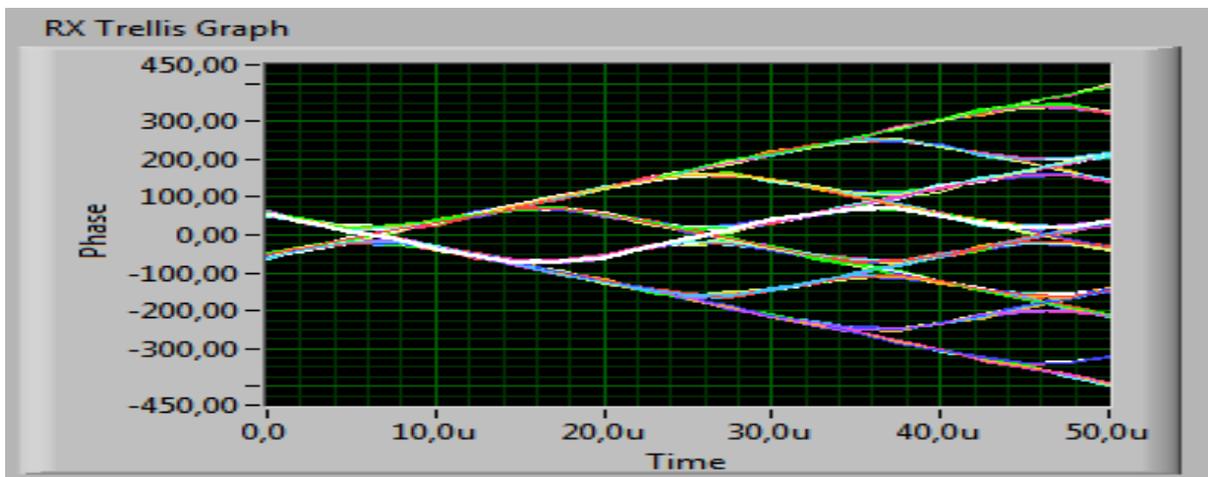


Рис. 1.48. Фазовая диаграмма полученного сигнала при  $BER=10^{-4}$ , сигнал/шум = 28

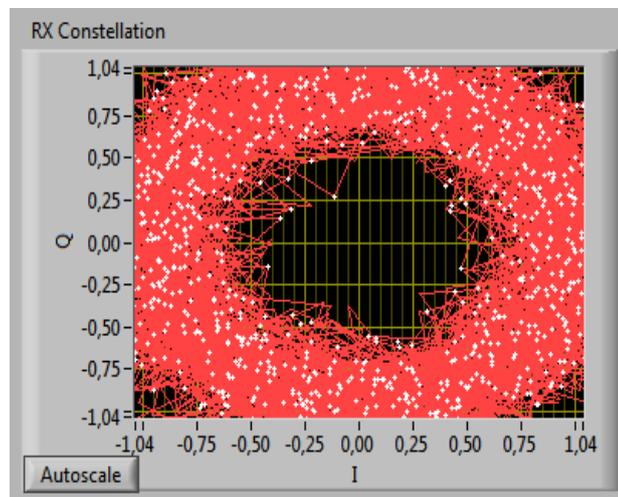


Рис. 1.49. Сигнальное созвездие сигнала при  $BER=2.04 \cdot 10^{-3}$ , сигнал/шум = 14

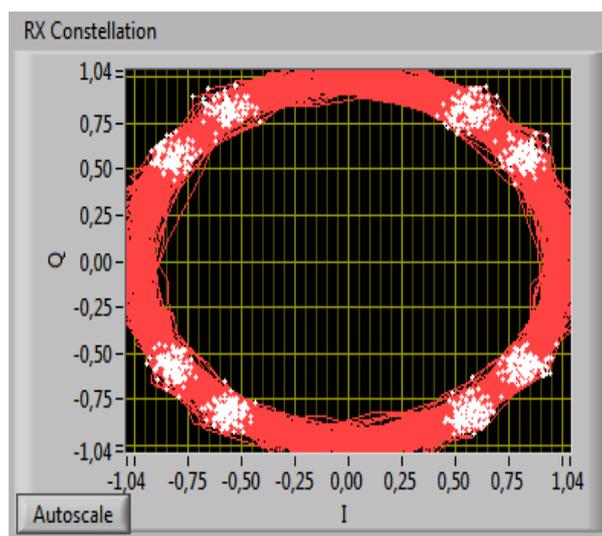


Рис. 1.50. Сигнальное созвездие сигнала при  $BER = 10^{-4}$ , сигнал/шум = 28

TX Filter Raised Cosine (синий), Root Raised Cos (красный), Gaussian (зеленый)

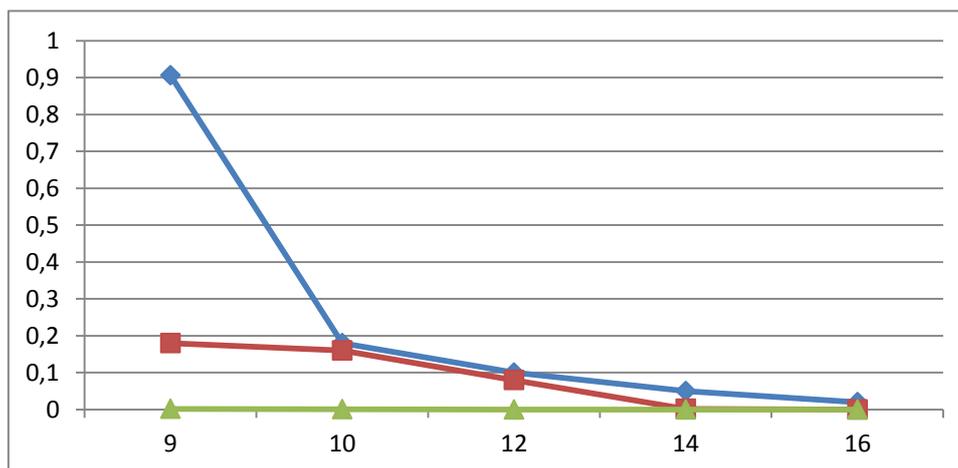


Рис. 1.51. Зависимость вероятности ошибки от отношения сигнал/шум

Таким образом, получаются следующие характеристики этих модуляций. FSK - простота реализации и относительно широкий спектр. GFSK - простота реализации меньший спектр, но и несколько худшая помехоустойчивость по отношению к FSK. MSK - очень высокая помехоустойчивость при относительно небольшом спектре, но сложная схема модулятора-демодулятора. GMSK - ширина спектра практически близка к теоретическому пределу  $B_T$  (скорость передачи цифровой информации), несколько худшая помехоустойчивость по сравнению с MSK, сложность модулятора-демодулятора такая же как у MSK.

При использовании программного обеспечения LabVIEW, были созданы две виртуальные измерительные лаборатории с различными видами модуляции сигнала FSK(Frequency Shift Keying) и MSK(Minimal Shift Keying) в каждой из которых можно

применить предварительные фильтрации и получить другие подвиды GMSK или GFSK, если использовать фильтр Гаусса.

### 1.3. Модемы спутниковых систем связи M-QAM, M-PSK и численный анализ вероятности символьной ошибки с использованием ПО LabVIEW

В данном разделе был разработан программный комплекс для изучения многоуровневых методов модуляции применяемых в спутниковых системах разработанная в ПО LabVIEW 2011 SP 1. В разработанном ПО можно исследовать такие виды манипуляции как BPSK, QPSK, 8-PSK, 16-PSK, 32-PSK, 64-PSK и M-QAM на то, как влияет отношение сигнал/шум на вид диаграммы, посмотреть глазковые диаграммы, спектр, оценить BER, посмотреть влияние фильтров на диаграмму.

Наиболее часто используют QPSK манипуляцию. Она имеет преимущество, поскольку может работать при мощности транспондера, близкой к насыщению, то есть эффективной отдачей энергии. В таблице 1 приведены теоретические значения отношения сигнал/шум, требуемые для достижения величины BER, равной  $10^{-10}$ , без кодирования канала связи для различных методов манипуляции.

Таблица 1.2. Сравнение методов цифровой манипуляции для BER, равной  $10^{-10}$

Метод манипуляции.	Сигнал/шум без применения кодирования, дБ.	Эффективность использования полосы пропускания, бит/с.
BPSK	13.06	1.0
QPSK	13.06	2.0
8-PSK	16.55	3.0
16-PSK	21.09	4.0
4-QAM	13.06	2.0
16-QAM	16.98	4.0
64-QAM	21.40	6.0

#### Квадратурная фазовая манипуляция (QPSK)

QPSK манипуляция строится на основе кодирования двух бит передаваемой информации одним символом. При этом символьная скорость в два раза ниже скорости передачи информации. На рис. 1.43 представлена векторная диаграмма QPSK.

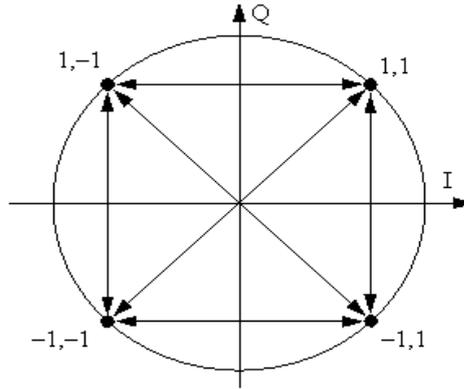


Рис. 1.52. Векторная диаграмма QPSK.

Кодирование осуществляется следующим образом: весь битовый поток разбивается на четные и нечетные биты, тогда  $I(t)$  будет кодировать четные биты, а  $Q(t)$  – нечетные. Два последовательно идущих друг за другом бита информации кодируются одновременно синфазным  $I(t)$  и квадратурным  $Q(t)$  сигналами. Это наглядно показано на осциллограммах, приведенных на рисунке 1.53 для информационного потока «1100101101100001».

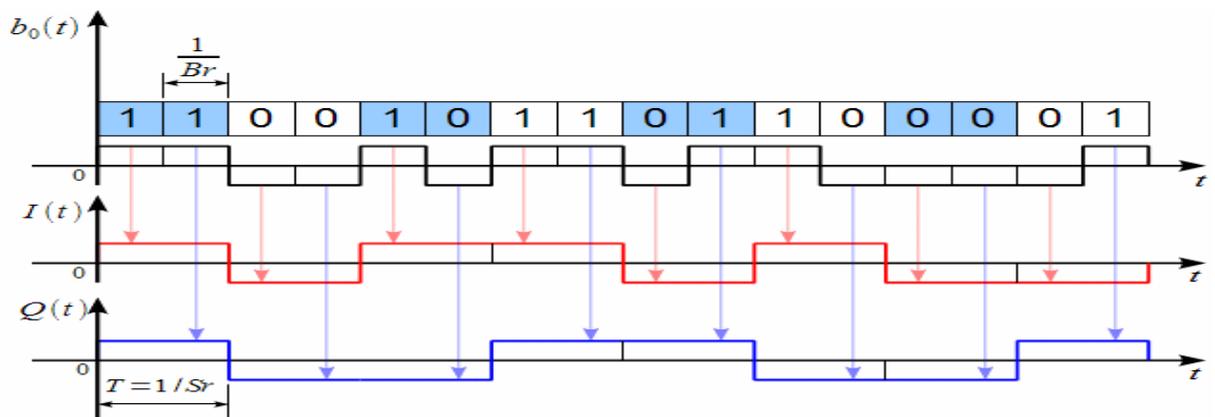


Рис. 1.53. Синфазная и квадратурная составляющие QPSK сигнала

На верхнем графике входной поток разделен на пары бит, соответствующих одной точке созвездия QPSK, показанного на рисунке 1.52. На втором графике показана осциллограмма  $I(t)$ , соответствующая передаваемой информации  $I(t) > 0$ . Если четный бит равен 1 (биты нумеруются с нуля, а не с единицы, поэтому первый в очереди бит имеет номер 0, а значит он четный по порядку), и  $I(t) < 0$  если четный бит 0 (т.е.  $b(t) < 0$ ). Аналогично строится квадратурный канал  $Q(t)$ , но только по нечетным битам. Длительность одного символа  $T = 1/S_r$  в два раза больше длительности одного бита исходной информации. Устройство выполняющее такое кодирование  $I(t)$  и  $Q(t)$  согласно созвездию, QPSK условно показано на рисунке 1.54.

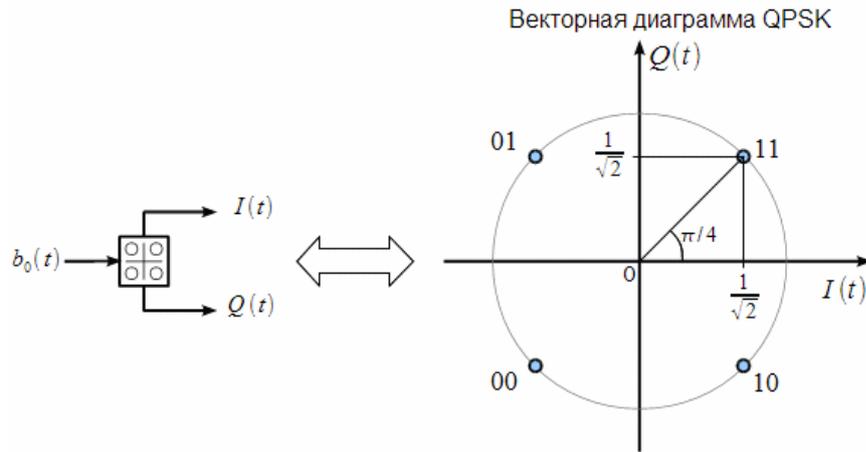


Рис. 1.54. Устройство кодирования синфазной и квадратурной составляющих на основе созвездия QPSK.

В зависимость от пары бит  $b(t)$  на входе на выходе получаем постоянные в пределах длительности этой пары бит сигналы  $I(t)$  и  $Q(t)$ , значение которых зависит от передаваемой информации.

#### Структурная схема QPSK модулятора.

Структурная схема QPSK модулятора на основе универсального квадратурного модулятора показана на рисунке 1.55.

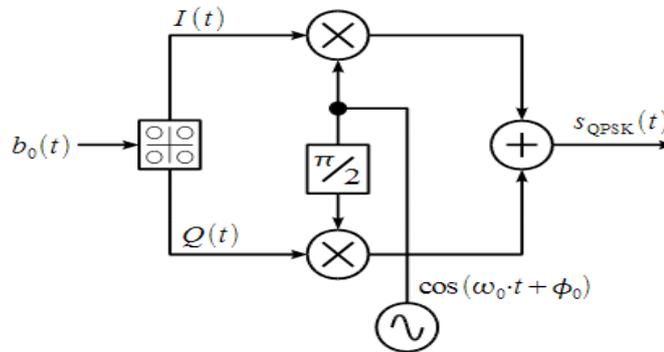


Рис. 1.55. Структурная схема QPSK модулятора.

Сигнал  $S_{QPSK}(t)$  имеет вид:

$$S_{QPSK}(t) = I(t) \cdot \cos(\omega_0 t + \varphi_0) - Q(t) \cdot \sin(\omega_0 t + \varphi_0)$$

Синфазная  $I(t)$  и квадратурная  $Q(t)$  составляющие это ничто иное, как реальная и мнимая части комплексной огибающей QPSK сигнала  $Z(t) = I(t) + j \cdot Q(t)$ , которые являются входными сигналами квадратурного модулятора. Тогда можно представить  $S_{QPSK}(t)$  через его комплексную огибающую  $Z(t)$ :  $S_{QPSK}(t) = R|Z(t) \cdot \exp(j\omega_0 t)|$ . Из комплексной огибающей можно выделить фазовую огибающую как:

$$\varphi(t) = \arctan\left(\frac{j[Z(t)]}{R[Z(t)]}\right) = \arctan\left(\frac{Q(t)}{I(t)}\right)$$

Важно отметить, что арктангенс должен вычисляться с учетом четверти комплексной плоскости (функции арктангенс 2). Вид фазовой огибающей  $\varphi(t)$  для информационного потока «1100101101100001» показан на рисунке 1.56.

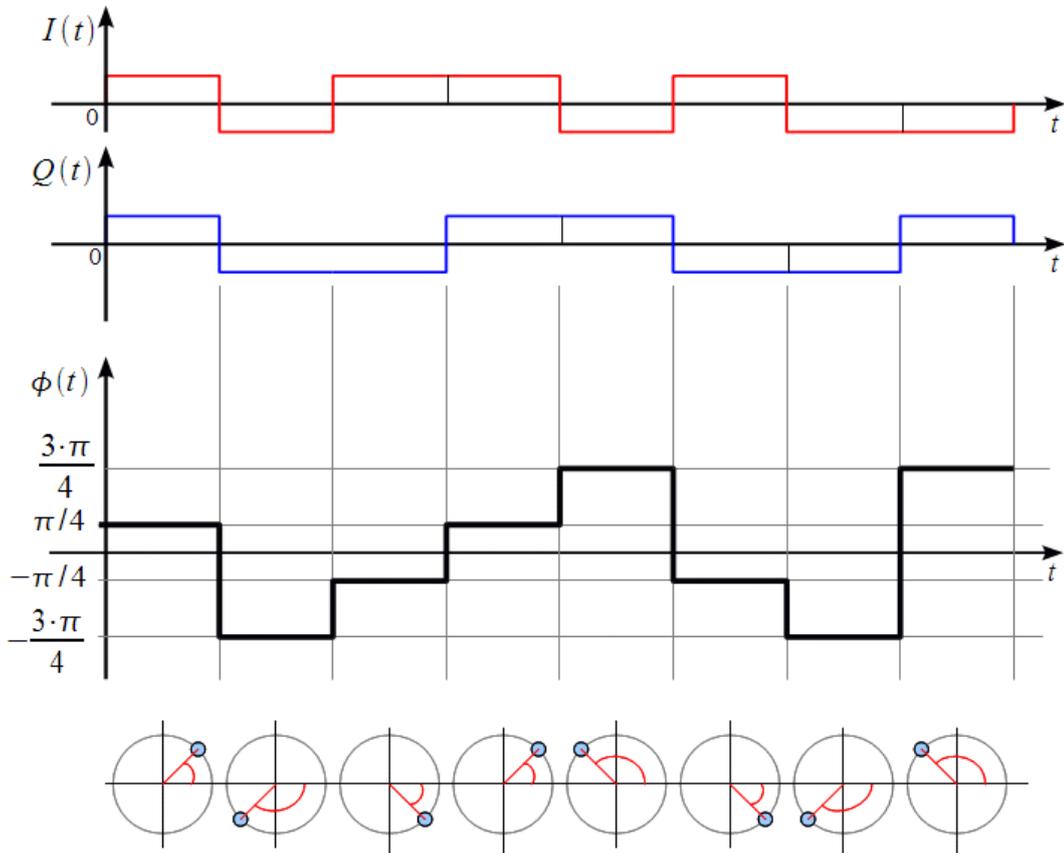


Рис. 1.56. Фазовая огибающая QPSK сигнала.

Фазовая огибающая представляет собой ступенчатую функцию времени, претерпевающую разрывы в моменты смены символа QPSK (один символ QPSK несет два бита информации). При этом в пределах одного символа векторная диаграмма QPSK находится всегда в одной точке созвездия, как это показано внизу, а при смене символа – скачкообразно переходит в точку соответствующую следующему символу. Поскольку у QPSK всего четыре точки в созвездии, то фазовая огибающая может принимать всего четыре значения:  $\mp \frac{\pi}{4}$  и  $\mp \frac{3\pi}{4}$ .

Амплитудная огибающая QPSK сигнала  $a(t)$  также может быть получена из комплексной огибающей  $Z(t)$ :

$$a(t) = \sqrt{I^2(t) + Q^2(t)}$$

Амплитудная огибающая QPSK сигнала равна единице всюду, за исключением моментов смены передаваемых символов, т. е. в моменты перескока фазы и перехода очередной точке созвездия.

Пример осциллограммы QPSK сигнала при входном битовом потоке «1100101101100001» при скорости передачи информации  $B_r=10$  кбит/с и несущей частоте 20 кГц показан на рисунке 1.57.

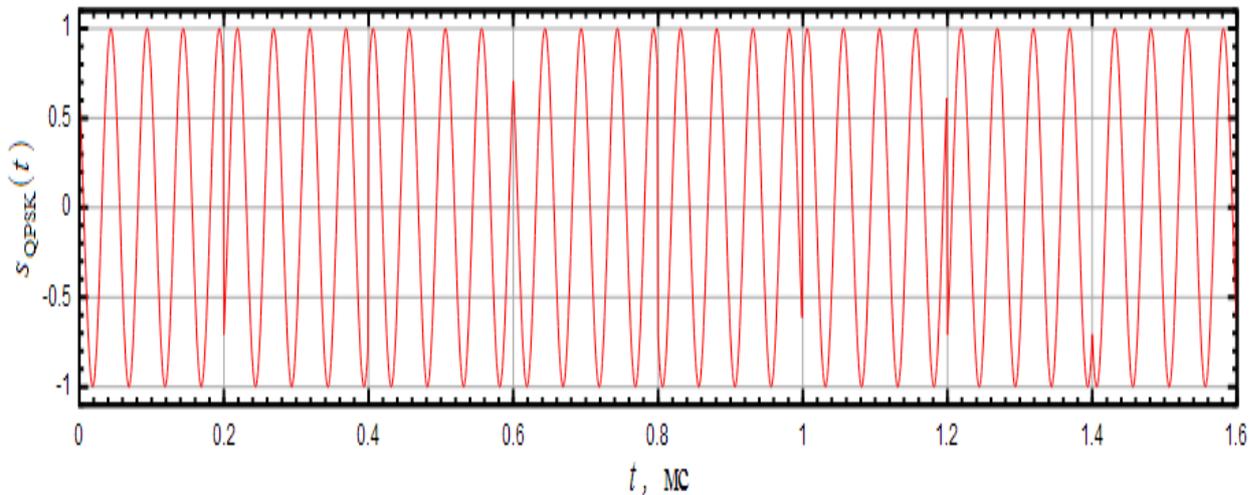


Рис. 1.57. Осциллограмма QPSK сигнала.

Фаза несущего колебания может принимать четыре значения:  $\mp \frac{\pi}{4}$  и  $\mp \frac{3\pi}{4}$  радиан. При этом фаза следующего символа относительно предыдущего может не измениться, или измениться на  $\mp \frac{\pi}{2}$  или на  $\mp \pi$  радиан. Также отметим, что при скорости передачи информации  $B_r=10$  кбит/с мы имеем символьную скорость  $S_r=B_r/2=5$  кбит/с, и длительность одного символа  $T=1/S_r=0.2$  мс, что отчетливо видно на осциллограмме (скачок фазы происходит через 0.2 мс).

#### Формирование спектра QPSK сигнала с помощью фильтров Найквиста.

На рисунке 1.58 показан спектр QPSK  $|S_{QPSK}(f)|^2$  сигналов при  $B_r=10$  кбит/с и несущей частоте 100 кГц. Можно заметить, что ширина главного лепестка, а также боковых лепестков QPSK сигнала вдвое меньше чем у BPSK сигнала при одной скорости передачи информации. Это обусловлено тем, что символьная скорость  $S_r$  QPSK сигнала вдвое меньше скорости передачи информации  $B_r$ .

Использование формирующих фильтров дает возможность передавать 0.5 символа в секунду на 1 Гц полосы, или 1 бит/с цифровой информации на 1 Гц полосы при использовании фильтра с АЧХ вида приподнятого косинуса.

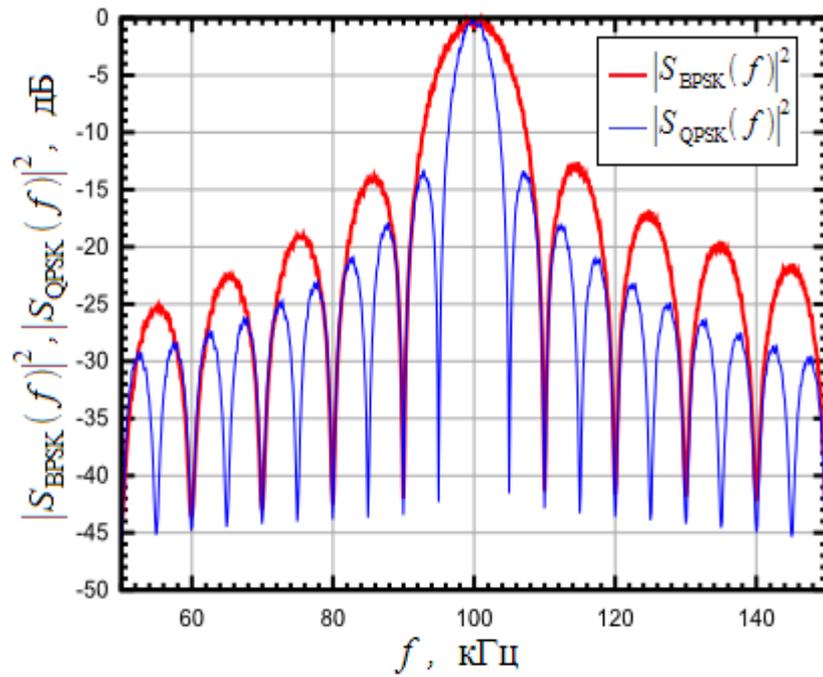


Рис. 1.58. Спектр QPSK сигнала.

На рисунке 1.58 черным показан спектр QPSK сигнала без использования формирующего фильтра. Видно, что применение фильтра Найквиста позволяет полностью подавить боковые лепестки в спектре QPSK сигналов. Структурная схема QPSK модулятора при использовании формирующего фильтра показана на рисунке 1.59.

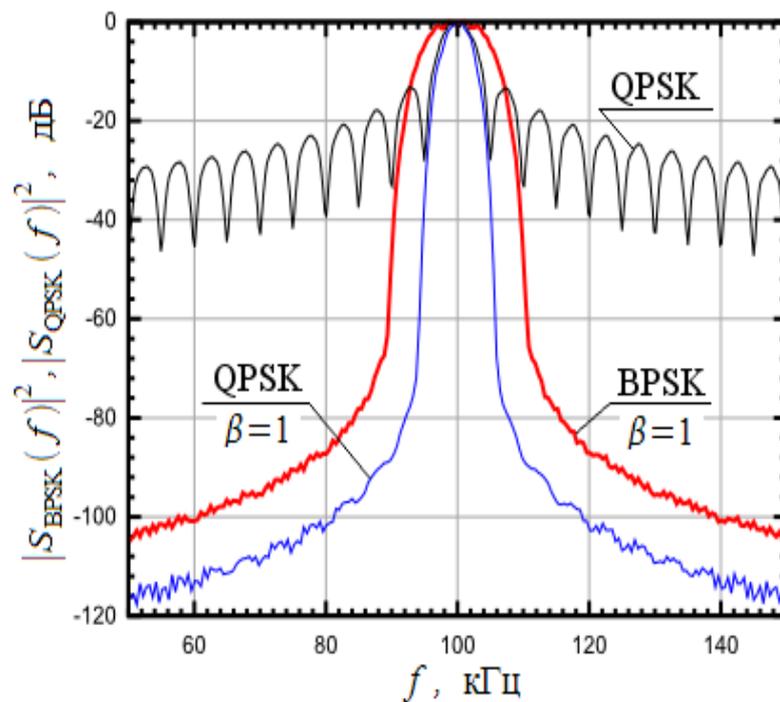


Рис. 1.59. Спектр QPSK сигнала с формирующим фильтром Найквиста

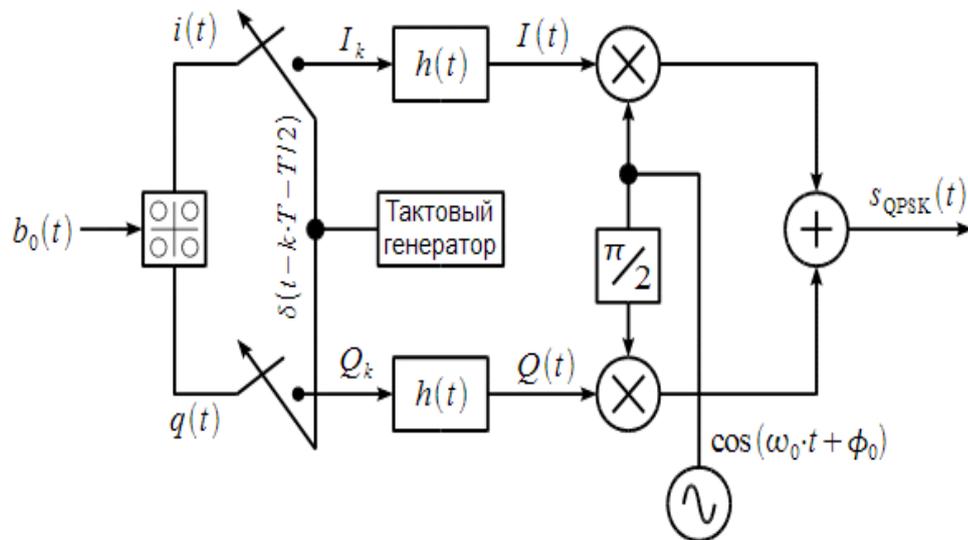


Рис. 1. 60. Структурная схема QPSK модулятора с использованием формирующего фильтра

Графики, поясняющие работу QPSK модулятора показаны на рисунке 1.61.

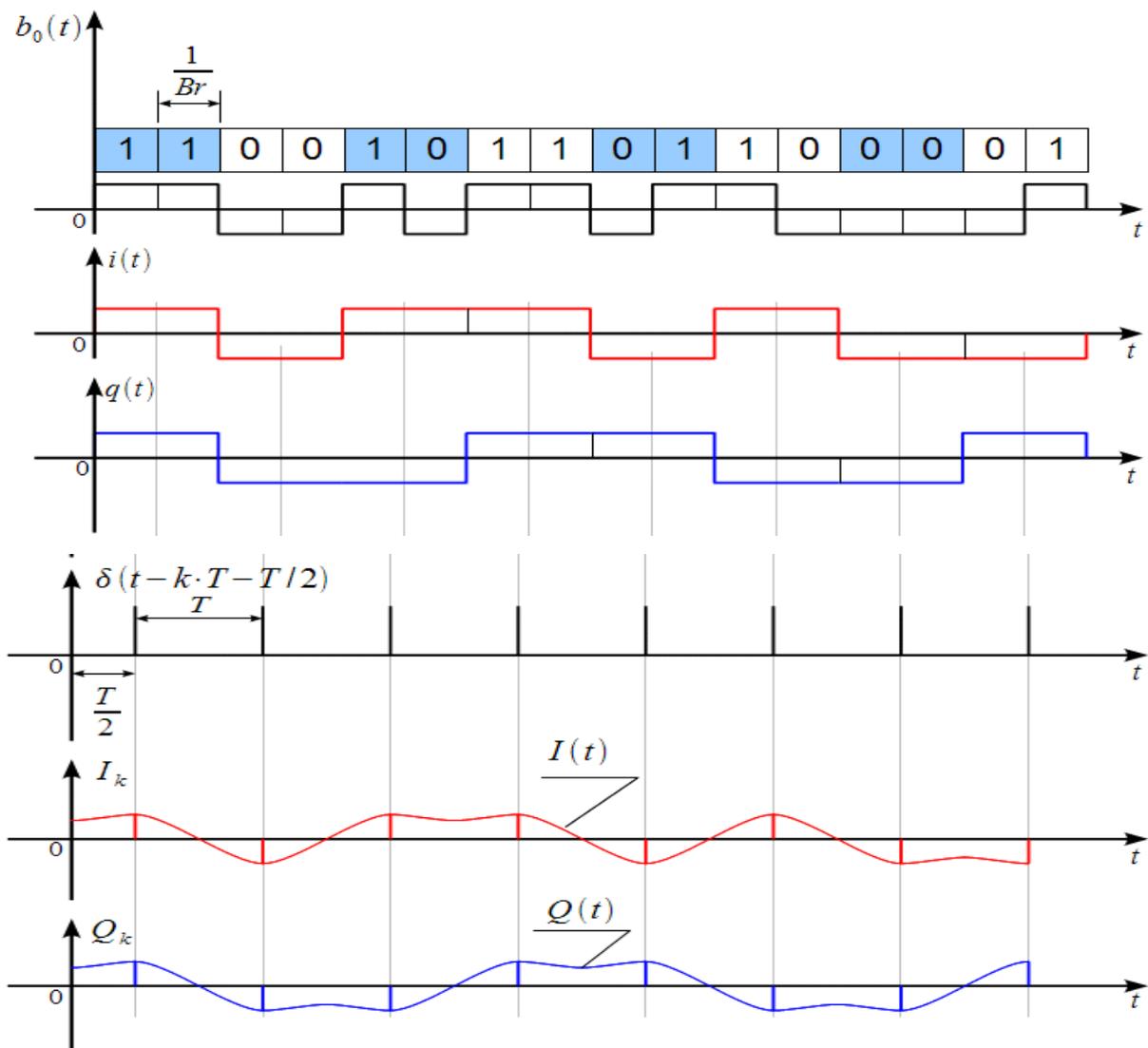


Рис. 1.61. Поясняющие графики.

Цифровая информация поступает со скоростью  $V_r$  и преобразуется в символы  $i(t)$  и  $q(t)$  в соответствии с созвездием QPSK, длительность одного передаваемого символа равна  $T=1/S_r=2/V_r$ . Тактовый генератор выдает последовательность дельта-импульсов с периодом  $T$ , но отнесенных к центру импульса  $i(t)$  и  $q(t)$ , как это показано на четвертом графике. Импульсы тактового генератора стробируют  $i(t)$  и  $q(t)$  при помощи ключей и получаем отсчеты  $I_k$  и  $Q_k$ , показанные на двух нижних графиках, которые возбуждают формирующий фильтр интерполятор с импульсной характеристикой  $h(t)$  и на выходе имеем синфазную  $I(t)$  и квадратурную  $Q(t)$  составляющие комплексной огибающей, которые подаются на универсальный квадратурный модулятор. На выходе модулятора получаем QPSK сигнал с подавлением боковых лепестков спектра.

Обратим внимание, что синфазная  $I(t)$  и квадратурная  $Q(t)$  составляющие становятся непрерывными функциями времени, в результате вектор комплексной огибающей QPSK уже не находится в точках созвездия, перескакивая во время смены символа, а непрерывно движется комплексной плоскости как это показано на рисунке 1.62 при использовании фильтра приподнятого косинуса с различными параметрами  $\beta$ .

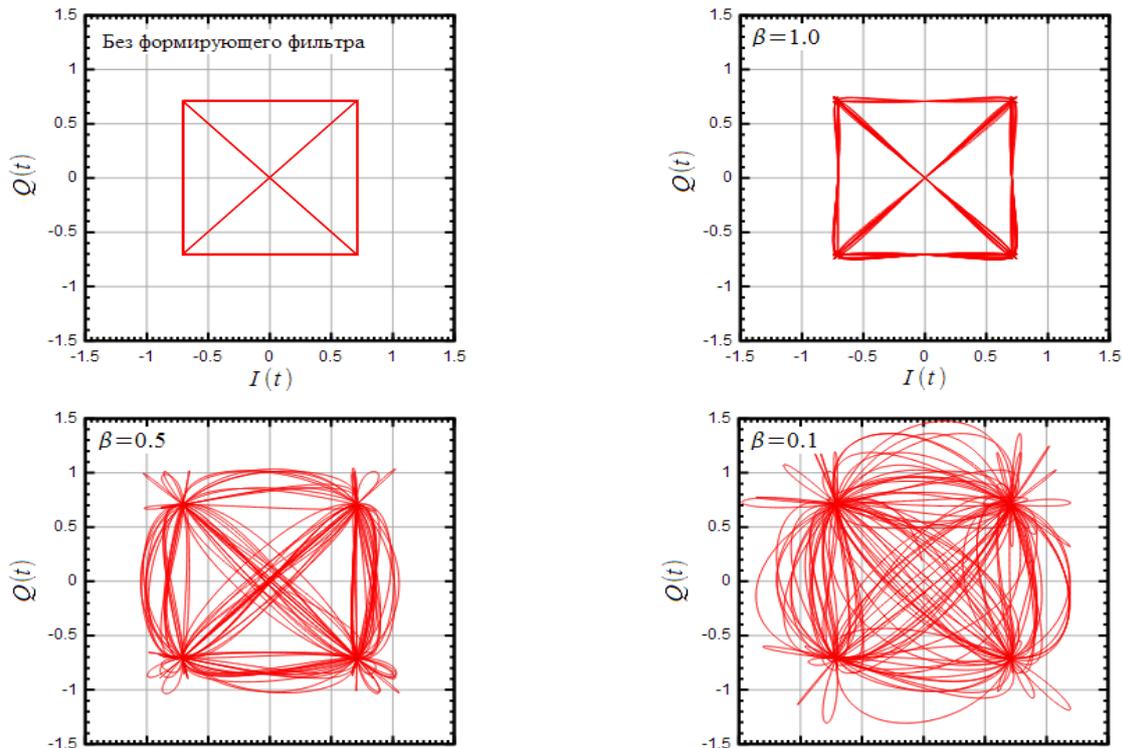


Рис. 1.62. Траектории движения вектора комплексной огибающей QPSK сигнала при различных параметрах формирующего фильтра Найквиста.

На верхнем левом графике показана векторная диаграмма при отсутствии формирующего фильтра при скачкообразном перемещении вектора комплексной огибающей. Если включить фильтр Найквиста, то при уменьшении  $\beta$  векторная диаграмма

превращается в «клубок». При  $\beta=1$  получаем наилучшее приближение к идеальной диаграмме.

При непрерывном движении вектора комплексной огибающей его амплитуда начинает меняться во времени, значит у QPSK сигнала при использовании формирующего фильтра начинает меняться амплитудная огибающая

$$a(t) = \sqrt{I^2(t) + Q^2(t)},$$

что наглядно демонстрируется осциллограммой QPSK сигнала, показанной на рисунке 1.63.

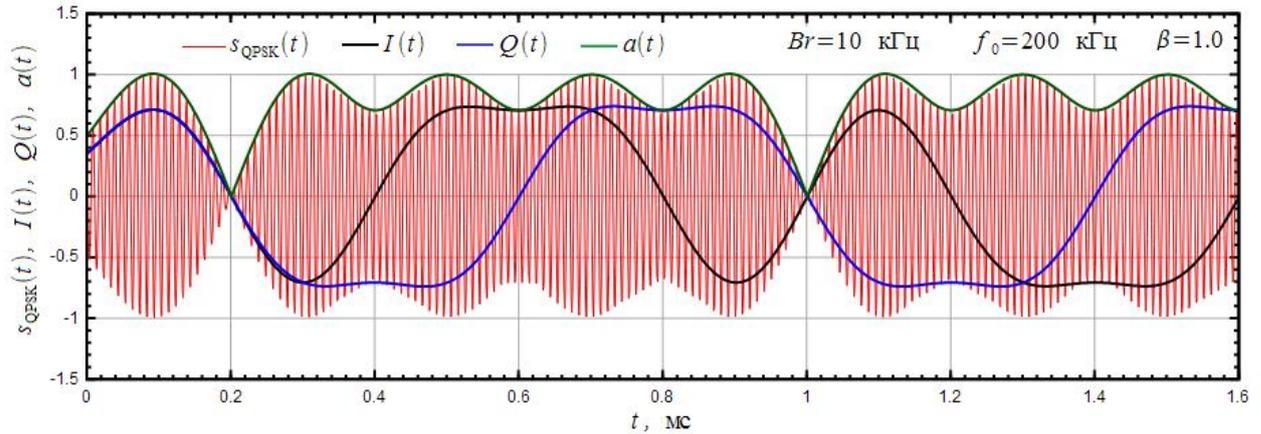


Рис. 1.63. Осциллограмма QPSK сигнала при использовании формирующего фильтра Найквиста

Видно, что фильтр Найквиста приводит к появлению паразитной амплитудной модуляции. При этом в точках когда и синфазная  $I(t)$  и квадратурная  $Q(t)$  составляющие комплексной огибающей равны нулю, амплитуда  $a(t)$  QPSK сигнала также падает до нуля, и фаза поворачивается на  $\pi$  радиан. Глубокая амплитудная модуляция — это негативный эффект, который устраняется офсетной QPSK (OQPSK) модуляции. Важно отметить, что при непрерывных  $I(t)$  и  $Q(t)$  фазовая огибающая также становится непрерывной функцией времени и перестает меняться скачкообразно, а плавно перетекает от символа к символу, что и приводит к сужению спектра QPSK сигнала при использовании формирующего фильтра.

### Практическая часть.

Внешний вид разработанного программного обеспечения представлен на рисунке 1.64.

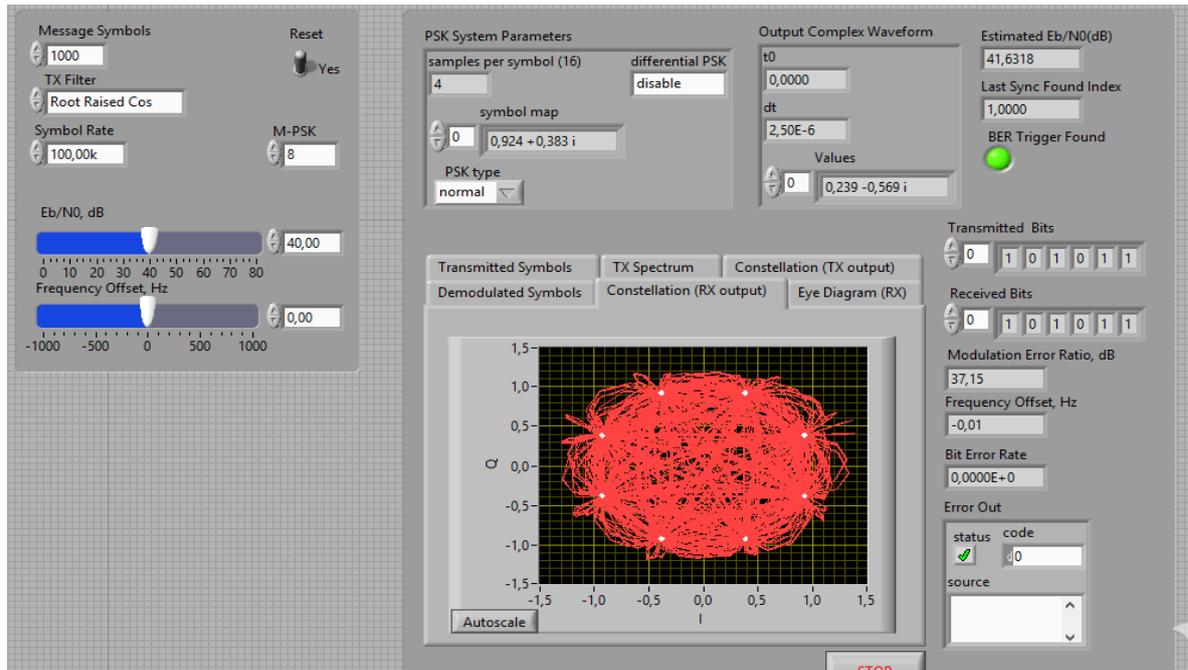


Рис. 1.64. Внешний вид разработанного ПО для исследования QPSK.

Программа имеет два блока:

1. Блок настройки (слева);
2. Блок отображения результата (справа).

В блоке настройки можно задавать количество символов в передаваемом сообщении (поле Message Symbols), чем больше символов в сообщении, тем дольше оно будет передаваться. На рисунке 1.65 показано как меняется диаграмма QPSK снятая с приемника в зависимости от количества символов сообщения.

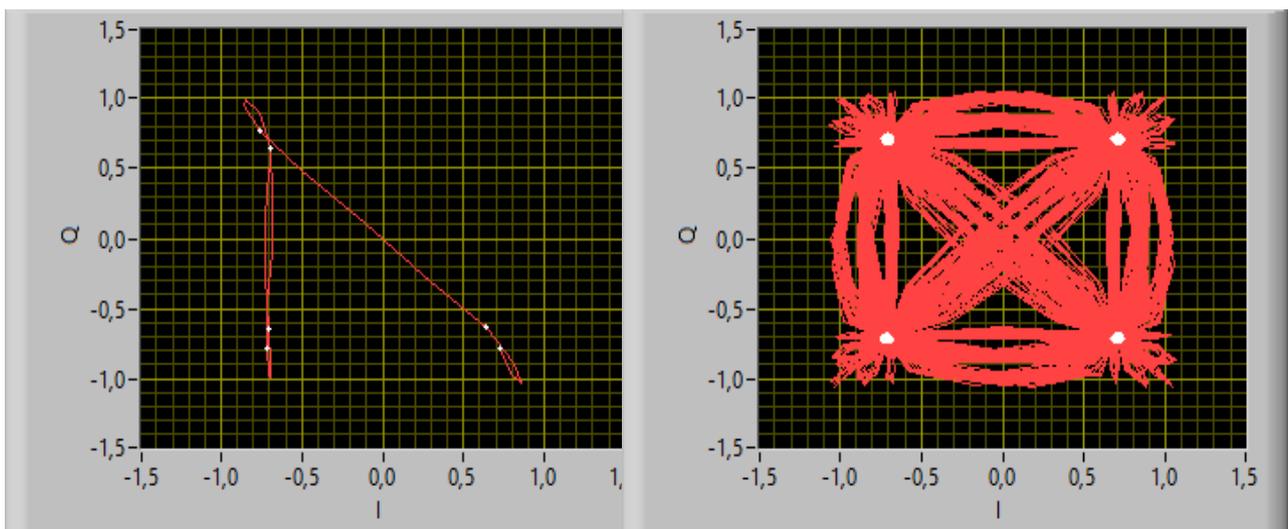


Рис. 1.65. Диаграмма QPSK. 10 символов в сообщении (слева), 1000 символов в сообщении (справа).

Можно задать значение отношения сигнал/шум (поле  $E_b/N_0$ , dB). На рисунке 16 представлена диаграмма QPSK для разных значений сигнал/шум.

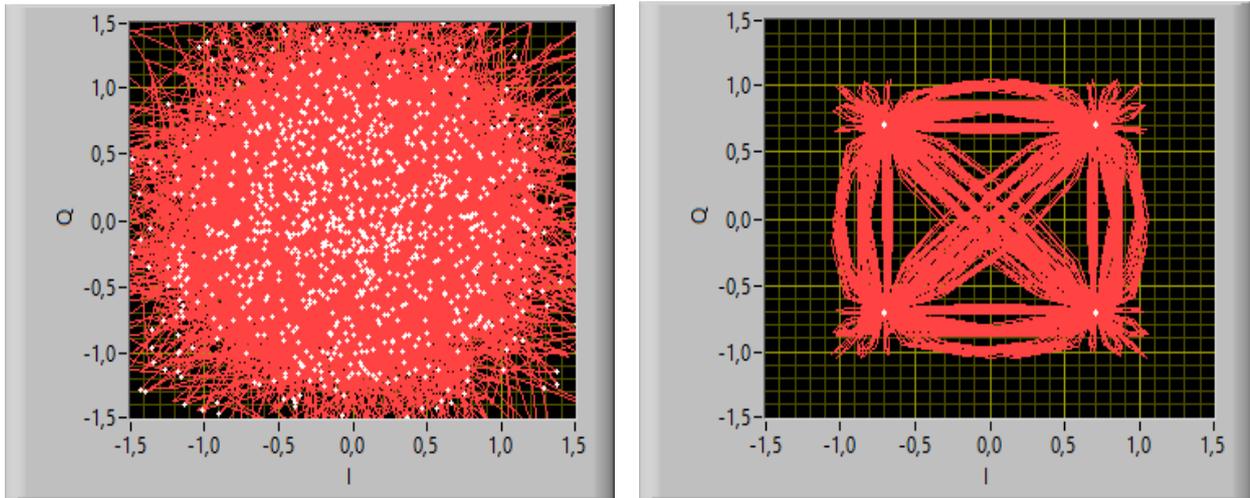


Рис. 1.66. Диаграмма QPSK. Отношение сигнал/шум 0 дБ (слева), Отношение сигнал/шум 60 дБ (справа).

Из рисунка 1.66 видно, что чем больше отношение сигнал/шум, тем четче выглядит диаграмма QPSK.

Так же можно установить:

1. Сдвиг по частоте (поле Frequency Offset, Hz) в диапазоне [-1000,1000] Гц с минимальным шагом 0.01 Гц;
2. Вид фильтра (поле TX Filter).

Можно задать следующие фильтры:

- None (без фильтра);
- Raised Cosine (Приподнятый косинус);
- Root Raised Cos();
- Gaussian (Гаусса).

3. Число точек  $M$  в созвездии (поле M-PSK). Можно задать следующие значения:

- 2 (BPSK);
- 4 (QPSK);
- 8 (8-BSK);
- 16 (16-BSK);
- 32 (32-BSK);
- 64 (64-BSK).

4. передаваемых бит (поле Transmitted Bits).

Последовательность

5. бит (поле Received Bits).

Последовательность принимаемых

6. Rate).

Скорость передачи (поле Symbol

На рисунке 1.60 показано как изменяется диаграмма QPSK в зависимости от выбранного значения  $M$ .

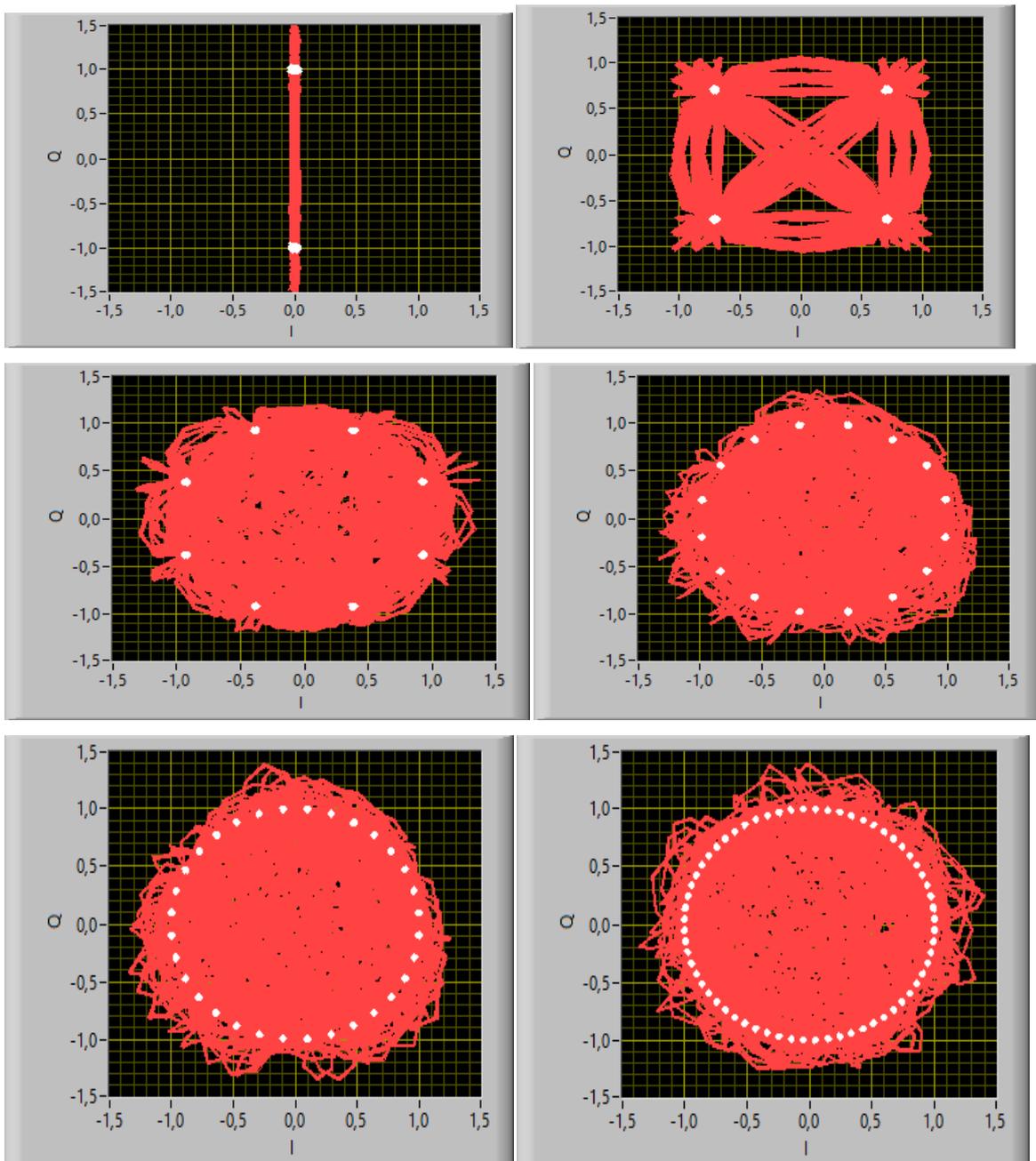


Рис. 1.67. Диаграмма QPSK. Слева на право сверху вниз - 2,4,8,16,32,64.

В разработанном программном обеспечении можно снимать спектр передатчика (вкладка TX Spectrum). На рисунке 1.67 представлены спектрограммы QPSK для разных фильтров.

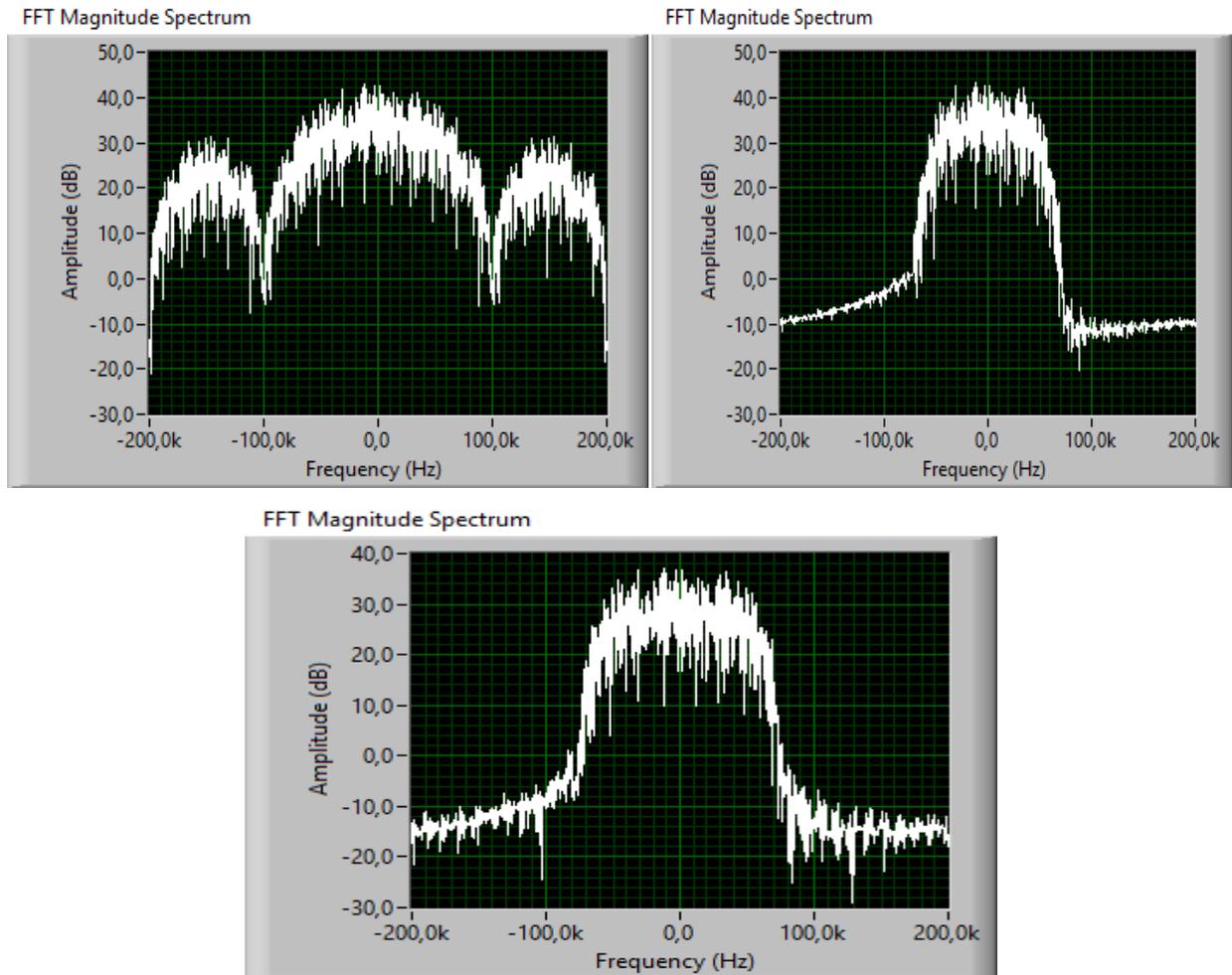


Рис. 1.68. Спектрограммы QPSK с выставленными значениями фильтра (слева на право - сверху вниз): без фильтра, приподнятый косинус, Root Raised Cos

В разработанном программном обеспечении можно снимать глазковую диаграмму с приемника (вкладка Eye Diagramm(RX) ). На рисунке 1.68 представлены глазковые диаграммы при разных значения М для оси Q, а на рисунке 1.69 представлены глазковые диаграммы при разных значения М для оси I

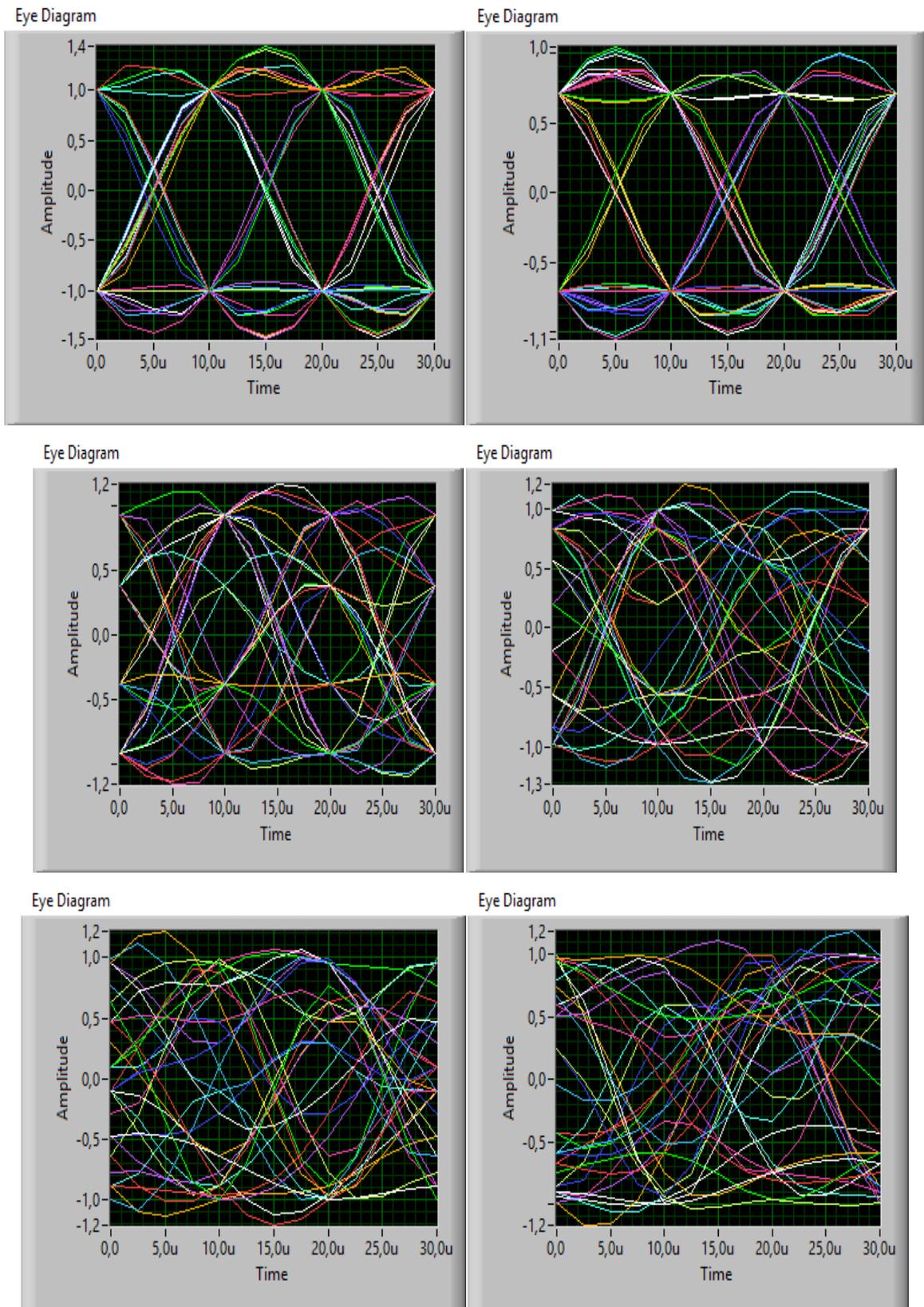


Рис. 1.69. Глазковые диаграммы для разных значений  $M$  на оси  $Q$  (слева на право – сверху вниз): 2,4,8,16,32,64

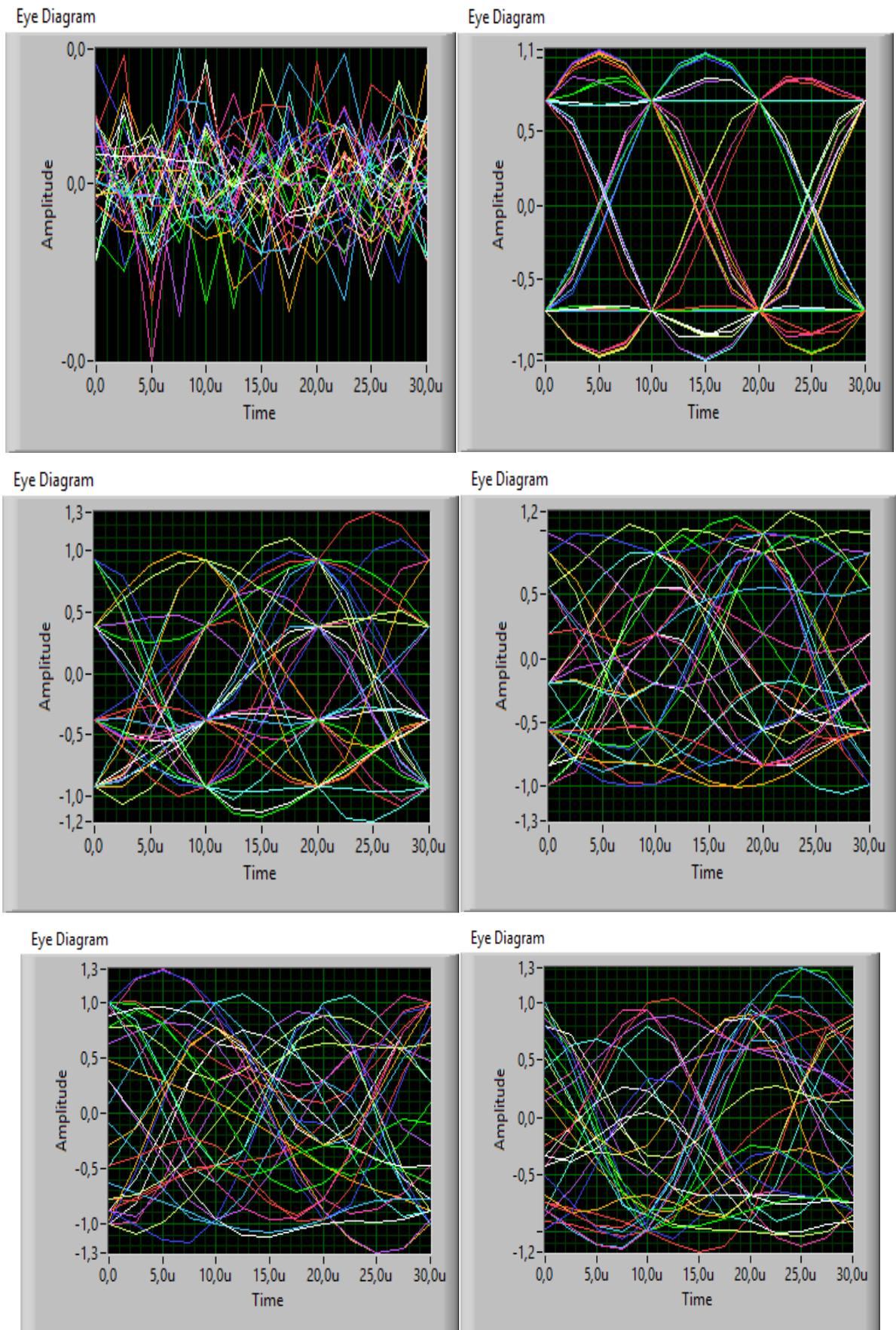


Рис. 1.70. Глазковые диаграммы для разных значений  $M$  на оси  $I$  (слева на право – сверху вниз): 2,4,8,16,32,64

Ось выбирается на вкладке Eye Diagramm (RX) в поле Display I/Q.

Так же программное обеспечение рассчитывает BER (Bit Error Rate – битовая вероятность ошибки). Это значение можно увидеть в поле Bit Error Rate.

Быстро определить есть ошибки или нет можно если посмотреть на зеленую кнопку с подписью BER Trigger Found, если она будет гореть значит битовая вероятность ошибки присутствует.

В правом нижнем углу есть информационное окно Error Out, в которое будет выводиться код и сообщение ошибки в случае не корректной работы программы.

На вкладках Transmitted Symbols и Demodulated Symbols можно посмотреть переданные и демодулированные символы соответственно.

**Квадратурно-амплитудная модуляция - QAM** (QAM – Quadrature Amplitude Modulation) может рассматриваться как расширенная многоуровневая ФМ, в которой два исходных сигнала генерируются независимо. Таким образом, здесь имеют место два полностью независимых квадратурных канала, включающие процессы кодирования и детектирования в основной полосе.

На рисунке показано сигнально - точечное пространство для системы с 16-QAM и четырьмя уровнями в каждом квадратурном канале. Точки представляют составной сигнал, а штрихи на осях отмечают уровни амплитуды в каждом квадратурном канале. Основная схема модулятор – демодулятора 16-QAM представлена на рисунке .

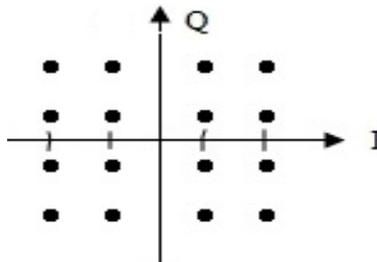


Рис.1.71. Сигнально-точечное пространство модуляции для 16-QAM

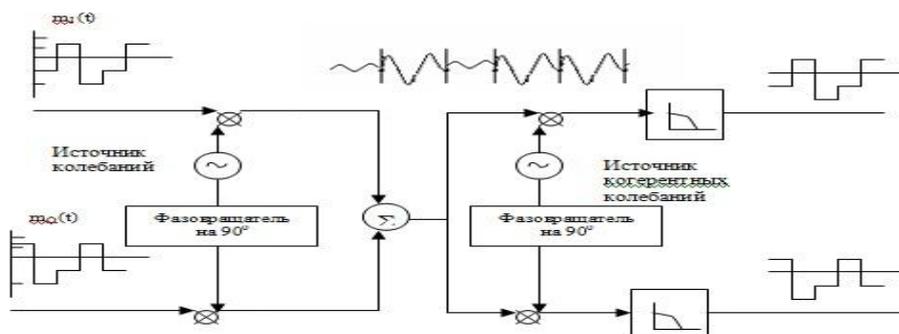


Рис. 1.72. Схема модулятора-демодулятора QAM

В отличие от ФМ сигналов сигналы QAM, показанные на рисунке не содержат постоянной огибающей. Наличие постоянной огибающей в ФМ объясняется поддержанием отношения уровней в квадратурных каналах. В QAM такие ограничения не вводятся ввиду того, что в каждом канале уровни независимы.

Характеристики ошибок систем QAM и ФМ модуляций сильно отличаются. При достаточно большом числе сигнальных точек системы QAM имеют, как правило, лучшие характеристики, чем системы с ФМ. Основная причина состоит в том, что расстояние между сигнальными точками на диаграмме для системы с QAM больше, чем для соответствующей системы с ФМ.

Расстояние  $d$  между соседними точками в системе QAM с нормированной к единице пиковой амплитудой и числом уровней  $L$  может быть представлено в виде:

$$d = \frac{\sqrt{2}}{L-1} \quad (1.14)$$

На рисунке представлено сравнение систем QAM-16 и ФМ-16 работающих на одинаковой пиковой мощности, по расстоянию между точками.

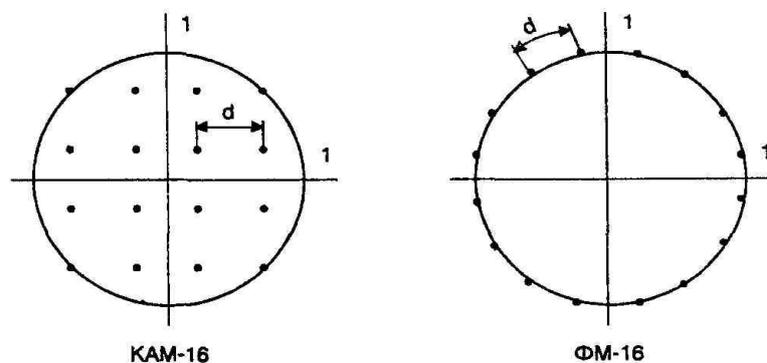


Рис. 1.73. Сравнение систем QAM-16 и ФМ-16 работающих на одинаковой пиковой мощности, по расстоянию между точками

QAM имеет преимущество над системой ФМ при той же пиковой мощности.

В настоящее время для передачи пользуются системами 256-QAM. Надо отметить, что надежное функционирование высокоплотных форматов модуляции, таких как 256-QAM требует строгой линейности усилителей, для возможности обработки широкого диапазона амплитуд сигналов. Соотношения для характеристик ошибок методов 4-, 16-, 64- и 256-QAM

в зависимости от отношения функции  $\frac{E_b}{N_0}$  приведены на рисунке.

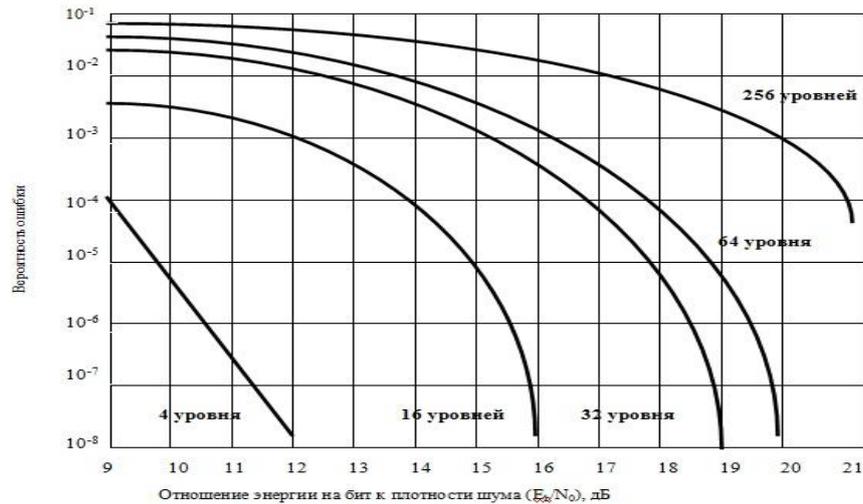


Рис. 1.74. Вероятности ошибок в системах QAM.

Достоинство высоких значений номера QAM – это повышенная скорость передачи данных, поскольку таким образом большее количество битов информации может быть передано в течении одного цикла. Однако, с другой стороны, в этом случае большее число уровней амплитуды сигнала располагаются близко друг к другу, повышая тем самым вероятность неразличимости двух уровней, и как следствие – повышая чувствительность системы к шуму. Таким образом, высокие значения номера QAM более требовательны к параметру SNR (Signal Noise Ratio – Отношение Сигнал/Шум).

### Практическая часть QAM

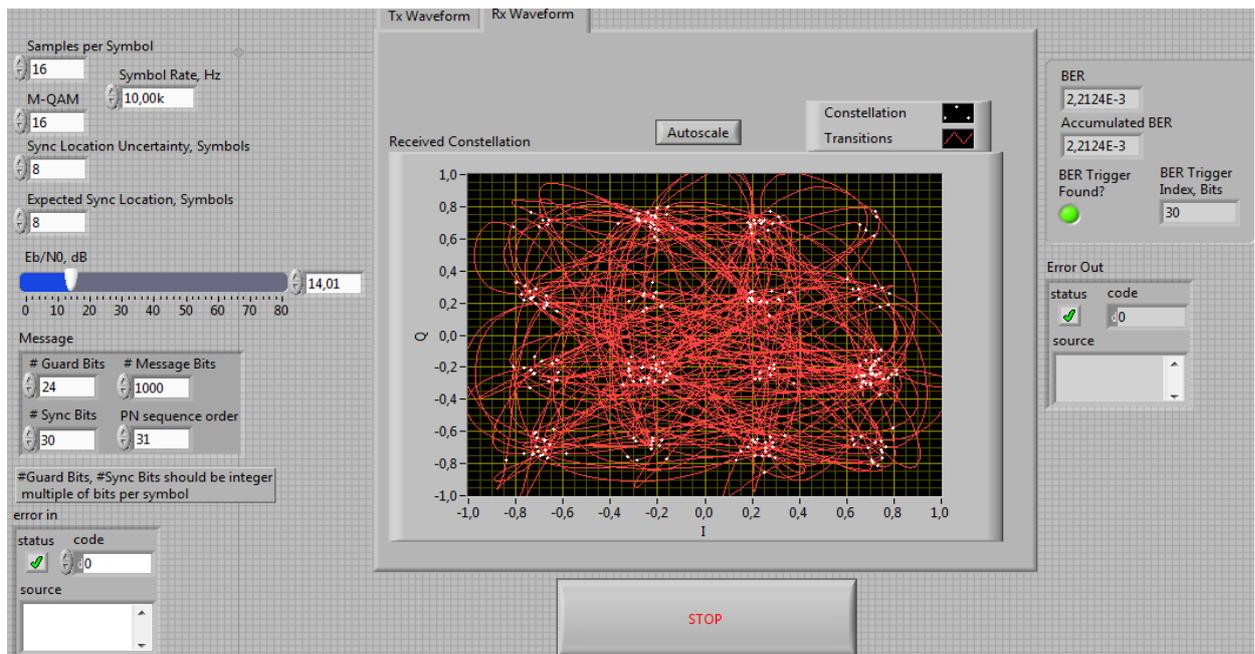


Рис. 1.75. Внешний вид разработанного ПО для исследования QAM

Число посылок 2000 и различные отношения сигнал/шум

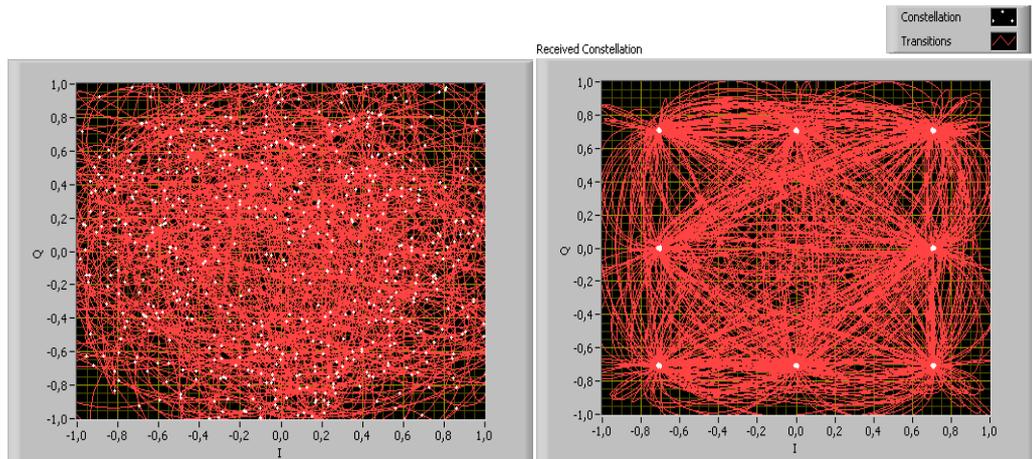


Рис. 1.76. Созвездия для QAM-8 передаваемого сигнала.

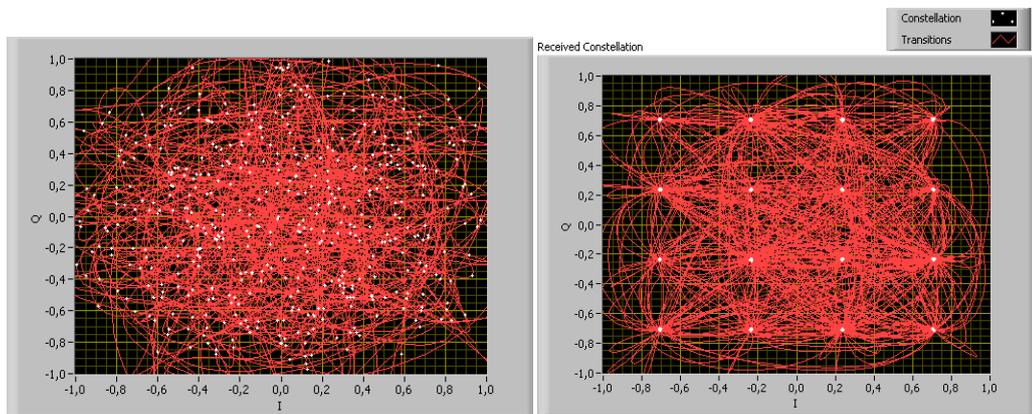


Рис. 1.77. Созвездия для QAM-16 передаваемого сигнала.

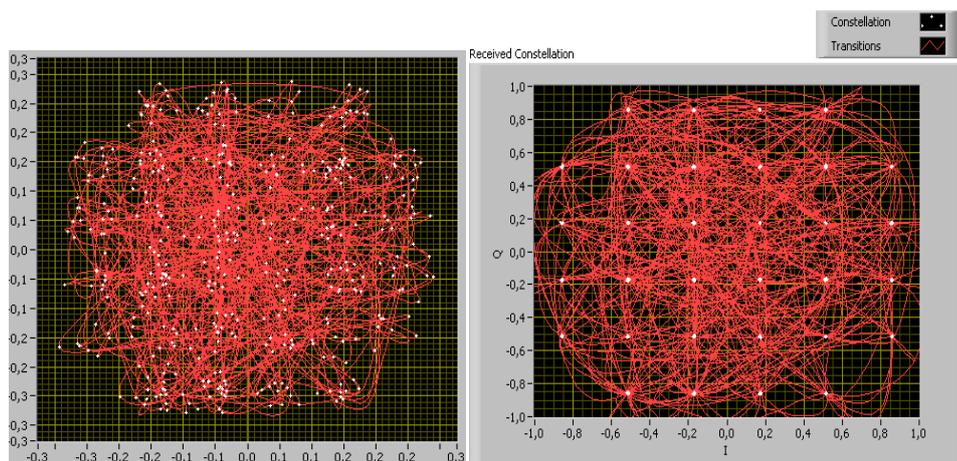


Рис. 1.78 Созвездия для QAM-32 передаваемого сигнала.

На рисунке 1.79. приведены глазковые диаграммы для QAM-8 сигнала.

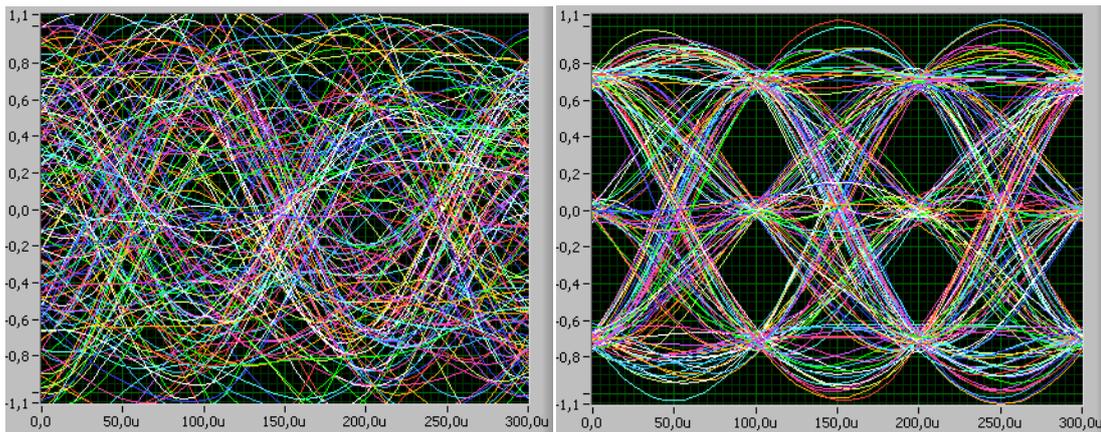


Рис. 1.79. Глазковые диаграммы при малом отношении сигнал/шум и при наилучшем отношении сигнал/шум

На рисунке 1.80. приведены глазковые диаграммы для QAM-16 сигнала.

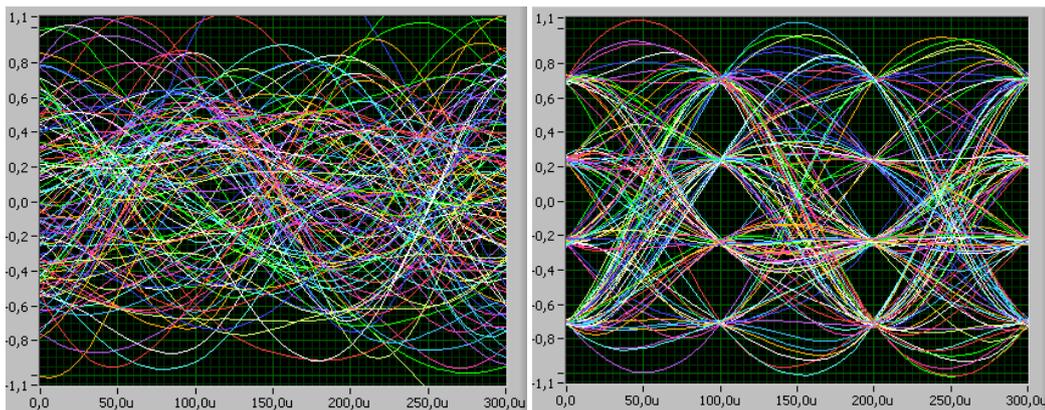


Рис. 1.81. Глазковые диаграммы при малом отношении сигнал/шум и при наилучшем отношении сигнал/шум

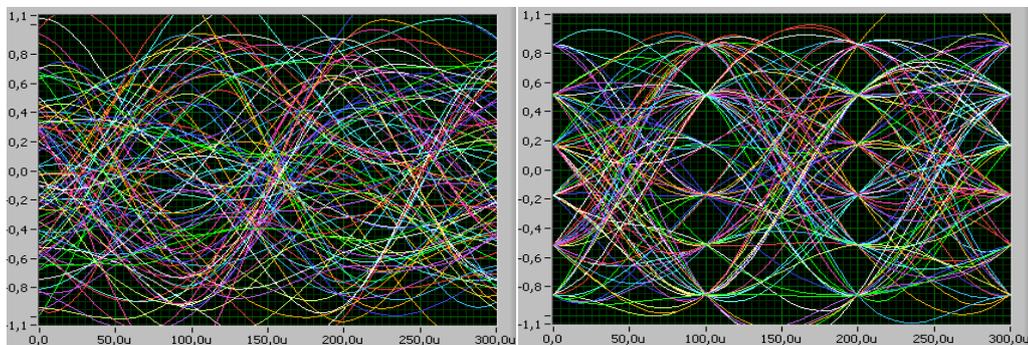


Рис. 1.82. Глазковые диаграммы при малом отношении сигнал/шум и при наилучшем отношении сигнал/шум

На рисунке 1.83 приведены спектрограммы для QAM-8 сигнала.

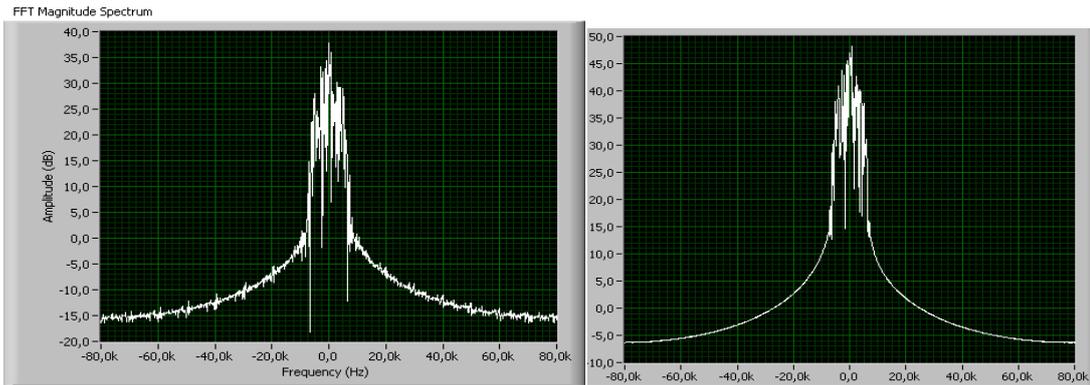


Рис. 1.83. Спектрограммы на входе и на выходе канала

На рисунке 1.84. приведены спектрограммы для QAM-16 сигнала.

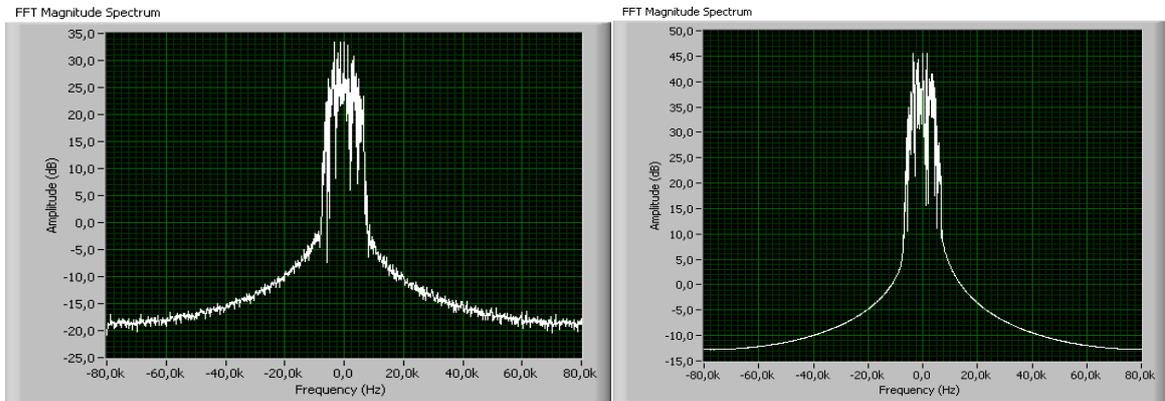


Рис. 1.85. Спектрограммы на входе и на выходе канала

На рисунке 1.86 приведены спектрограммы для QAM-32 сигнала.

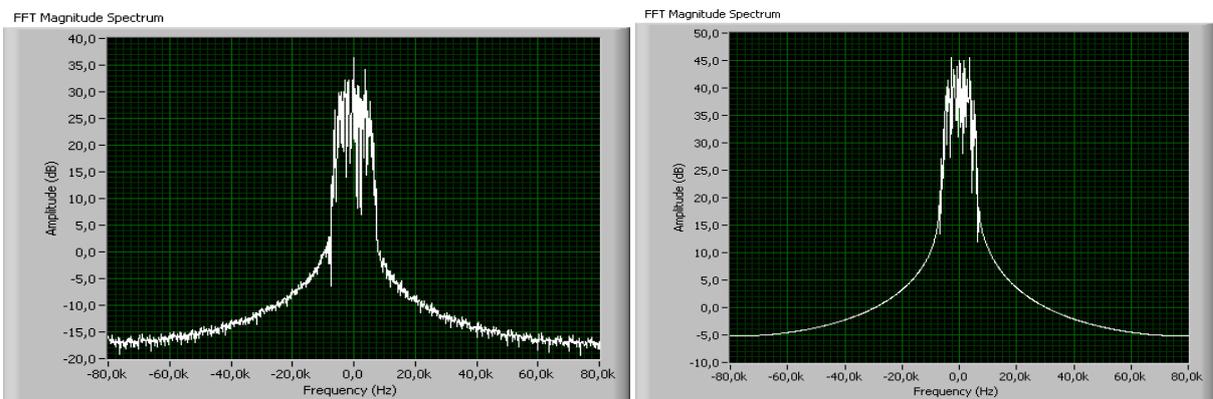


Рис. 1.86. Спектрограммы на входе и на выходе канала

Ползунком  $E_b/N_0$  устанавливается уровень отношения сигнал/шума, в поле BER отображается количество обнаруженных ошибок при передаче. Из полученных данных

можно построить график зависимости, сравнить показатели у различных видов модуляции и подтвердить/опровергнуть теорию, описанную выше.

В таблице ниже представлены данные для рассмотренных видов модуляций.

BER	1	1	1	1	1	1,00E+00	1,19E-02	3,97E-03	7,94E-03	1,98E-03
BER	1	1	1	1	4,51E-01	1,00E+00	1,72E-02	7,81E-03	2,08E-03	2,08E-03
BER	1	1	1	1	1,00E+00	2,27E-01	2,42E-02	2,10E-03	5,25E-04	5,25E-04
$E_b/N_0$ , dB	0	2	4	5	6	8	10	12	14	16

На рисунке представлен график зависимости параметра BER от  $E_b/N_0$  для 8-QAM, 16-QAM, 32-QAM

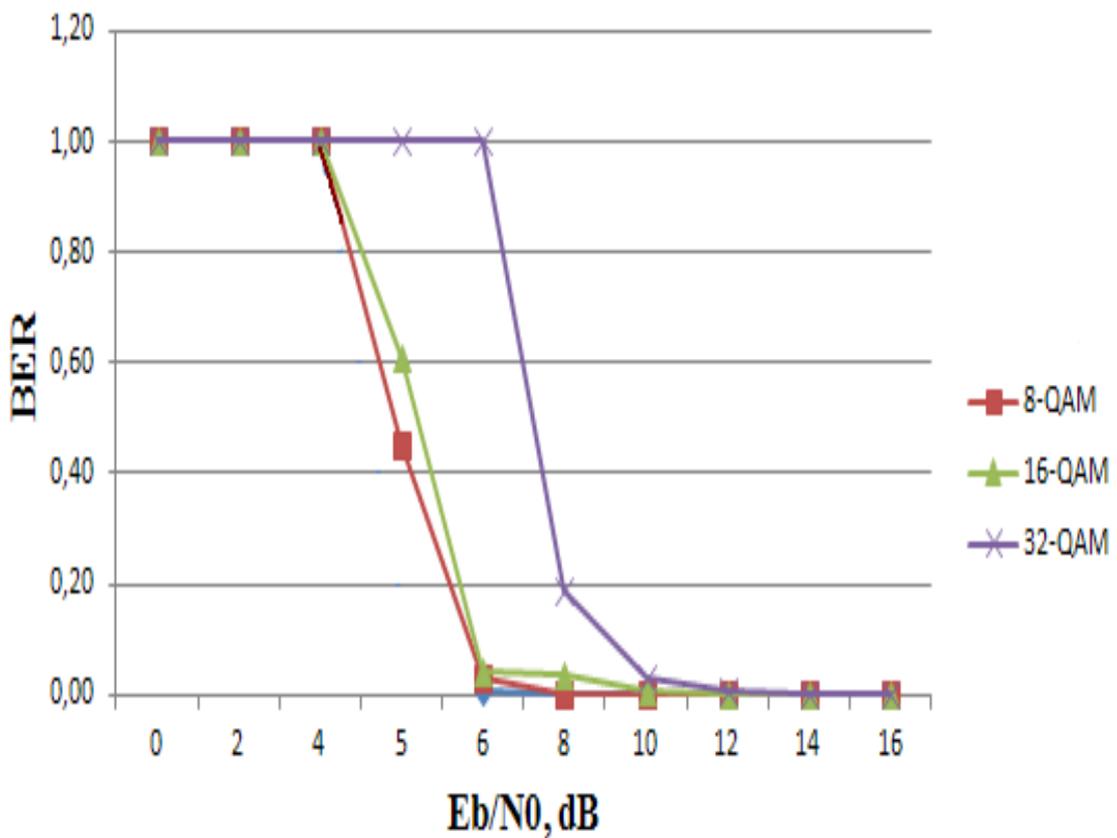


Рис. 1.87. График зависимости параметра BER от  $E_b/N_0$  для 8-QAM, 16-QAM, 32-QAM

## ГЛАВА 2. ПРОПУСКНАЯ СПОСОБНОСТЬ КАНАЛА СВЯЗИ. КОДИРОВАНИЕ ИСТОЧНИКА

### 2.1. Пропускная способность канала связи. Объем сигнала и емкость канала связи, условия их согласования

Рассматривается вопрос согласования дифференциальных характеристик источника дискретной информации (ИДИ) и предоставленного дискретного канала связи (КС) в терминах потока информации, выводимой из ИДИ, и пропускной способности КС. В данном разделе эта задача решается на уровне интегральных характеристик сигнала и канала в виде объема сигнала и емкости канала связи.

Сигнал как модель сообщения (информации) имеет «габаритные размеры», характеризующие объем сигнала, аналитически определяемый выражением

$$V_c = T_c F_c W_c = T_c F_c \log_2 \left(1 + \frac{P_c}{P_n}\right) \quad (2.1)$$

где  $T_c$  - временная длительность сигнала,  $F_c$  - эффективный спектр сигнала, определяемый эффективным спектра элементарного сигнала кода и типом модуляции. В (2.1) компонент  $W_c = \log_2 \left(1 + \frac{P_c}{P_n}\right)$  именуется логарифмическим превышением сигнала над помехой, в котором  $P_c$  - мощность сигнала,  $P_n$  - мощность помехи, сопровождающей процесс формирования сигнала.

Аналогичным образом канал связи как транспортная среда характеризуется емкостью канала связи, аналитически задаваемую выражением

$$V_k = T_k F_k W_k = T_k F_k \log_2 \left(1 + \frac{P_c}{P_n}\right), \quad (2.2)$$

где  $T_k$  - длительность интервала времени, на который предоставлен канал связи,  $F_k$  - эффективная полоса пропускания канала связи, которая может быть определена аналитически или экспериментально по амплитудной частотной характеристике четырехполюсника, который представляет собой канал связи. В (2.2) компонент  $W_k = \log_2 \left(1 + \frac{P_c}{P_n}\right)$  именуется логарифмическим превышением сигнала помехой, в котором  $P_c$  - мощность сигнала, фиксируемая в канальной среде,  $P_n$  - мощность помехи в канальной среде.

Нетрудно понять, что передача сигнала по предоставленному каналу связи возможна только тогда, когда размеры транспортируемого средства (сигнала) не превышают размеров транспортной среды (канала связи).

Таким образом необходимым условием согласования сигнала с предоставленным каналом связи является выполнение неравенства

$$V_c \leq V_k. \quad (2.3)$$

Достаточными условиями согласования сигнала с предоставленным каналом связи является выполнение неравенств

$$T_c \leq T_k, \quad F_c \leq F_k, \quad W_c \leq W_k, \quad (2.4)$$

И наконец, условием эффективного использования предоставленного канала связи является выполнение равенства

$$V_c = V_k. \quad (2.5)$$

Конструктивным инструментом согласования сигнала с предоставленным каналом связи путем уменьшения объема сигнала  $V_c$  за счет уменьшения компонента  $T_c$  является использование возможностей эффективного кодирования.

**Теорема К.Шеннона** об эффективном кодировании символов ИДИ (основная теорема К.Шеннона)

Пусть источник дискретной информации (ИДИ) генерирует алфавит

$X = \{x_i : p(x_i); i = \overline{1, n}\}$  составленный из  $n$  дискретных статистически независимых

$$H(X) = - \sum_{i=1}^n p(x_i) \log_2 p(x_i)$$

символов, характеризующийся энтропией, тогда:

1. существует такой способ кодирования символов, при котором среднее на символ

$$l_{cp}[K(X)] = \sum_{i=1}^n p(x_i) l[K(x_i)]$$

число двоичных разрядов кода будет сколь угодно близким к энтропии  $H(X)$  ИДИ;

2. не существует такого способа кодирования, при котором среднее на символ число двоичных разрядов кода  $l_{cp}[K(X)] = \sum_{i=1}^n p(x_i) l[K(x_i)]$  будет меньше энтропии  $H(X)$

ИДИ.

Наиболее распространенными алгоритмами эффективного кодирования, основанного на использовании основной теоремы К.Шеннона, являются:

- Алгоритм К.Шеннона - Р.Фано,
- Алгоритм Д. Хаффмэна.

Следует сказать, что содержательный момент этих алгоритмов, несмотря на процедурные различия, позволяющий построить эффективный двоичный код, средняя на символ длина которого максимально приближена к энтропии источника, состоит в том, что символам с большей вероятностью появления на выходе ИДИ ставятся в соответствие коды меньшей длины, а символам с меньшей вероятностью - коды большей длины. Таким образом, эффективное кодирование реализуется в классе неравномерных кодов.

Эффективные коды принято характеризовать двумя показателями кода: избыточность и эффективность

**Определение 2.1.** Избыточностью эффективного кода называется показатель  $D_k$ , определяемый выражением

$$D_k = \frac{l_{cp}[K(X)] - l_{\min}[K(X)]}{l_{cp}[K(X)]} = \frac{\sum_{i=1}^n p(x_i)l[K(x_i)] - H(X)}{\sum_{i=1}^n p(x_i)l[K(x_i)]}. \quad (2.6)$$

**Определение 2.2.** Эффективностью эффективного кода называется показатель  $\eta$ , определяемый выражением

$$\eta = \frac{l_{\min}[K(X)]}{l_{cp}[K(X)]} = \frac{H(X)}{\sum_{i=1}^n p(x_i)l[K(x_i)]}. \quad (2.7)$$

Нетрудно видеть, что избыточность и эффективность эффективного кода связаны соотношением

$$D_k = 1 - \eta. \quad (2.8)$$

## 2.2. Исследование кодирования источника дискретных сообщений методом

### Шеннона Фано

Для удобства расположим все имеющиеся  $p$  букв в один столбик в порядке убывания вероятностей. Затем все эти буквы следует разбить на две группы – верхнюю и нижнюю – так, чтобы суммарная вероятность первой группы была наиболее близка к суммарной вероятности второй группы. Для букв первой группы в качестве первой цифры кодового обозначения используется цифра 1, а для букв второй группы – цифра 0. Далее, каждую из двух групп подобным образом снова надо разделить на две части и в качестве второй цифры кодового обозначения мы будем использовать цифру 1 или 0 в зависимости от того, принадлежит ли наша группа к первой или ко второй из этих подгрупп. Затем, каждая

из содержащих более одной буквы групп снова делится на две части возможно более близкой суммарной вероятности и т.д.; процесс повторяется до тех пор, пока мы не придем к группам, каждая из которых содержит по одной единственной букве.

Например, если наш алфавит содержит всего шесть букв, вероятность которых (в порядке убывания) равны 0,4, 0,2, 0,2, 0,1, 0,05 и 0,05, то на первом этапе деления букв на группы мы отщепим лишь одну первую букву (1-я группа), оставив во второй группе все остальные. Далее, вторая буква составит 1-ю подгруппу 2-й группы; 2-я же подгруппа той же группы, состоящая из оставшихся четырех букв, будет и далее последовательно делиться на части так, что каждый раз 1-я часть будет состоять лишь из одной буквы.

Таблица 2.1.

№ буквы	вероятность	разбиение на подгруппы (римские цифры обозначают номера групп и подгрупп)					кодировочное обозначение	
1	0,4	} 1					1	
2	0,2	} 0	} 1				01	
3	0,2		} 1				001	
4	0,1		} 0	} 1				0001
5	0,05			} 0	} 0	} 1		
6	0,05		} 0			} 0	} 0	} 1
								} 0

Основной принцип, положенный в основу кодирования по методу Шеннона – Фано, заключается в том, что при выборе каждой цифры кодировочного обозначения мы стараемся, чтобы содержащееся в ней количество информации было наибольшим, т. е. чтобы независимо от значений всех предыдущих цифр, эта цифра принимала оба возможных для нее значения 0 и 1 по возможности с одинаковой вероятностью.

#### Процесс декодирования

Теперь рассмотрим алгоритм декодирования кодов Шеннона-Фано. Процесс усложняется тем, что невозможно, как в случае кодирования, заменять каждые 8 бит входного потока, кодом переменной длины. При восстановлении исходной последовательности необходимо провести обратные операции - заменить код переменной длины символом длиной 8 бит. В данном случае, лучше всего будет использовать бинарное дерево, ячейками которого будут являться символы.

#### Пример кодирования сообщений

Для наглядной иллюстрации алгоритма, сделаем сжатие предложения "Карлсон, который живет на крыше.". Вычислим, сколько раз встречается каждый символ, и занесем данные в таблицу 2.2.

Таблица 2.2.

Символ	Количество появлений
Пр	4
обел	3
р	3
о	2
к	2
а	2
е	2
т	2
ы	2
н	1
К	1
й	1
л	1
в	1
и	1
ш	1
ж	1
с	1
.	1
,	1

Далее, разделив таблицу на две части, таким образом, чтобы число появлений символов в верхней половине таблицы примерно равнялось общему числу в нижней половине. В предложении 32 символа, а значит, верхняя часть таблицы должна содержать около 16 появлений. Мы можем получить этот результат, поместив разделительную линию (далее р.л.) между строками "е" и "т". В итоге верхняя часть будет содержать появление 16 символов, а нижняя, соответственно, оставшиеся 16.

Далее проделаем то же с каждой из частей таблицы: размесим линии так, чтобы разделить 2 полученные части на приблизительно равные по количеству появлений.

Продолжаем процесс разделения до того момента, пока каждый символ не будет отделен от другого.

Результирующее дерево.

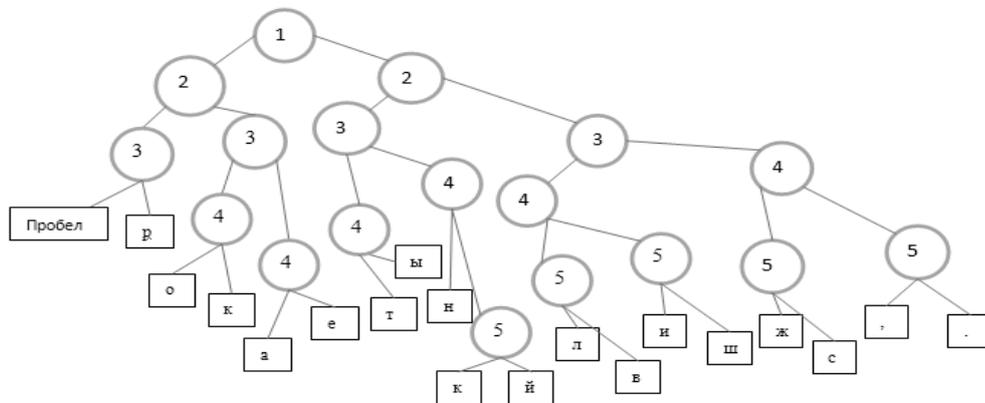


Рис.2.1. Результирующее дерево

Если принять, что перемещение влево эквивалентно нулевому биту, а вправо - единичному, можно создать таблицу кодирования.

Таблица 2.3

Символ	Кодировка
Пробел	000
р	0100
о	0101
к	0110
а	0111
е	1000
т	1001
ы	1010
н	10110
К	10111
й	11000
л	11001
в	11010
и	11011
ш	11100
ж	11101
с	11110
.	11111
,	

Таблица содержит 137 бит, а оригинальная фраза "Карлсон, который живет на крыше." занимает 256 бит, следовательно, у нас получается коэффициент сжатия 53%.

### Практическая часть

1. Запустить программный комплекс [6].

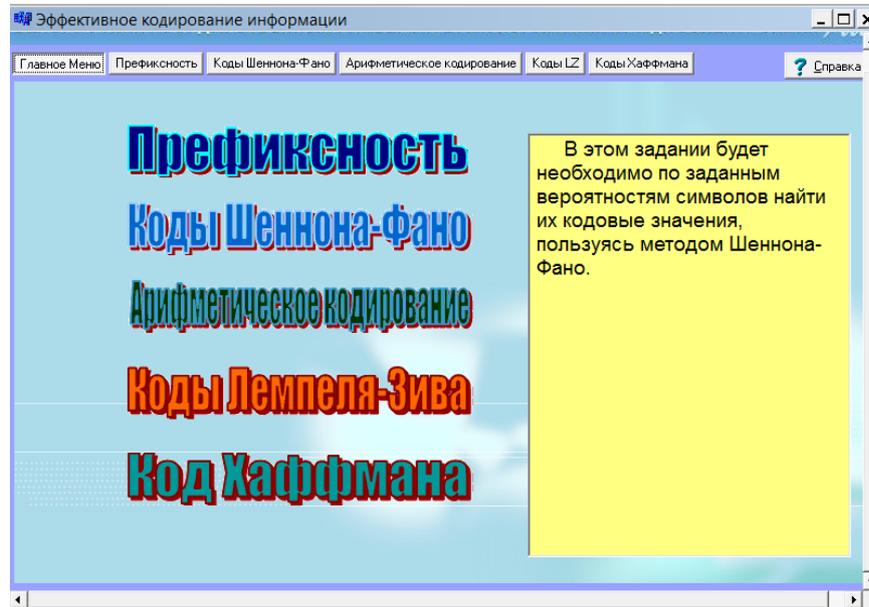


Рис. 2.2. Панель программного комплекса

2. Выбрали в меню пункт задания «Коды Шеннона-Фано». Изучили предложенную таблицу 16-символьного алфавита. Символы отсортированы в порядке убывания вероятностей их появления.

3. Пользуясь теоретическим материалом, найти двоичные коды для каждого из предложенных символов, записали их в графу Код кодовой таблицы. Следуя правилу приписывания очередного символа, указанному в окне программы над кодовой таблицей.

Закодировать приведенные символы алфавита кодом Шеннона-Фано. При разбиении суммы вероятностей в каждой из групп должны быть максимально одинаковыми.  
При разбиении на группы всем символам верхней половины в качестве первого бита приписывают 0, а всем нижним - 1

Символ	Вероятность	Код
А	0,11	000
Б	0,09	0010
В	0,08	0011
Г	0,08	010
Д	0,07	0110
Е	0,07	0111
Ж	0,07	1000
З	0,07	1001
И	0,06	1010
Й	0,06	1011
К	0,06	1100
Л	0,05	1101
М	0,05	1110
Н	0,04	11110
О	0,03	111110
П	0,01	111111

Проверить таблицу

Рис. 2.3. Кодовая таблица для данного сообщения

Закодируйте приведенные символы алфавита кодом Шеннона-Фано. При разбиении суммы вероятностей в каждой из групп должны быть максимально одинаковыми.

При разбиении на группы всем символам верхней половины в качестве первого бита приписывайте 0, а всем нижним - 1

Символ	Вероятность	Код
А	0,11	000
Б	0,09	0010
В	0,08	0011
Г	0,08	010
Д	0,07	0110
Е	0,07	0111
Ж	0,07	1000
З	0,07	1001
И	0,06	1010
Й	0,06	1011
К	0,06	1100
Л	0,05	1101
М	0,05	1110
Н	0,04	11110
О	0,03	111110
П	0,01	111111

Подсказка: Вы можете проверять результат построчно(посимвольно), для этого после каждой введенной вами строки (кода символа) нажимайте "Проверить таблицу". Программа сотрет строки результат которых введен не правильно.

Лабораторная работа

Верно! Вы отлично справились с заданием, продолжайте в том же духе.

OK

Проверить таблицу

Рис. 2.4. Процесс проверки кодовой таблицы

Закодируйте приведенные символы алфавита кодом Шеннона-Фано. При разбиении суммы вероятностей в каждой из групп должны быть максимально одинаковыми.

При разбиении на группы всем символам верхней половины в качестве первого бита приписывайте 0, а всем нижним - 1

Символ	Вероятность	Код
А	0,11	000
Б	0,09	0010
В	0,08	0011
Г	0,08	010
Д	0,07	0110
Е	0,07	0111
Ж	0,07	1000
З	0,07	1001
И	0,06	1010
Й	0,06	1011
К	0,06	1100
Л	0,05	1101
М	0,05	1110
Н	0,04	11110
О	0,03	111110
П	0,01	111111

Подсказка: Вы можете проверять результат построчно(посимвольно), для этого после каждой введенной вами строки (кода символа) нажимайте "Проверить таблицу". Программа сотрет строки результат которых введен не правильно.

Декодировать приведенную последовательность при помощи таблицы кода Шеннона-Фано.

001110100110011111110

**Результат:**

Проверить

Рис. 2.5. Закодированная последовательность

4. Декодировали при помощи составленной таблицы предложенной последовательности символов. Ответ записали в поле результат. По завершении проверили.

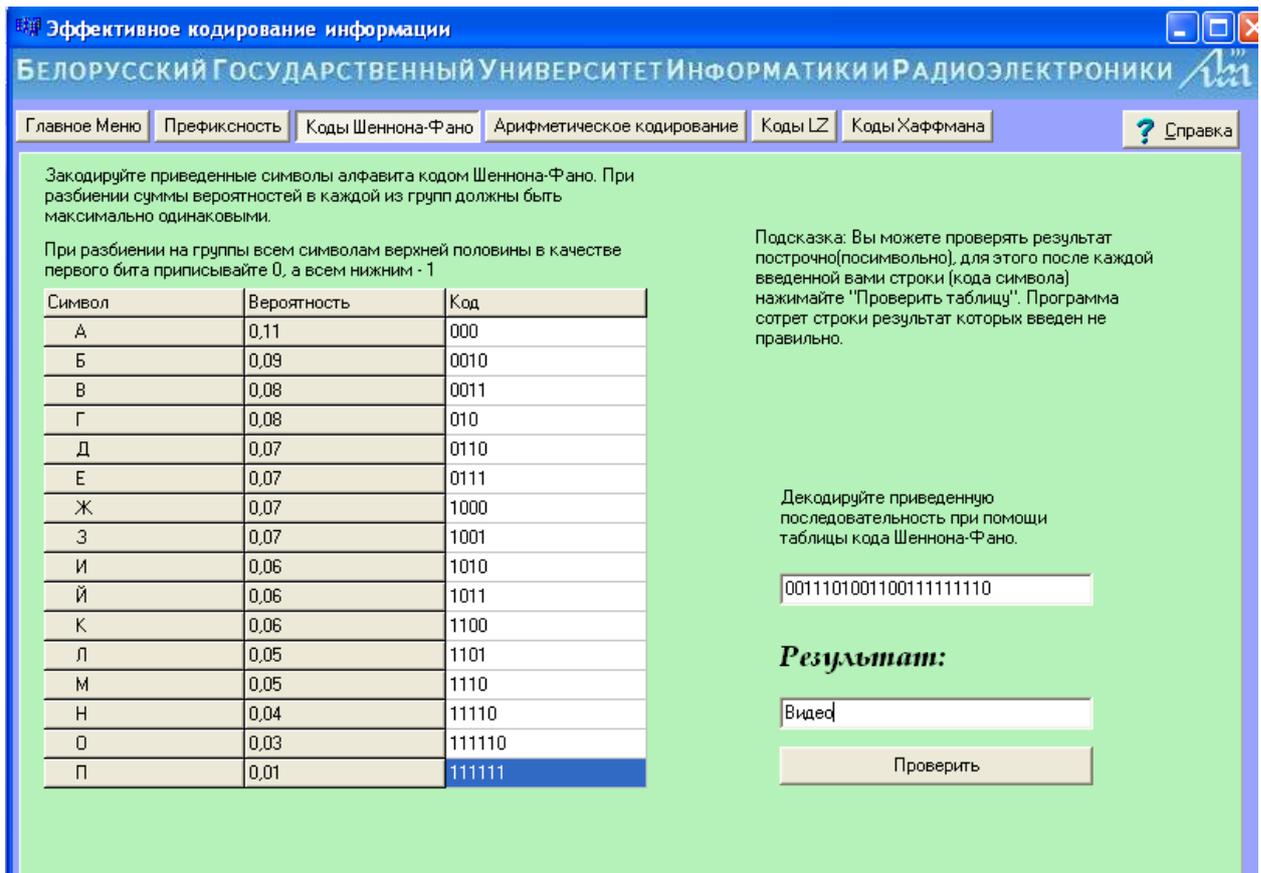


Рис. 2.6. Декодирование закодированной последовательности

Полученную последовательность в соответствии с таблицей символов декодировали самостоятельно и в результате получили слово: «Видео».

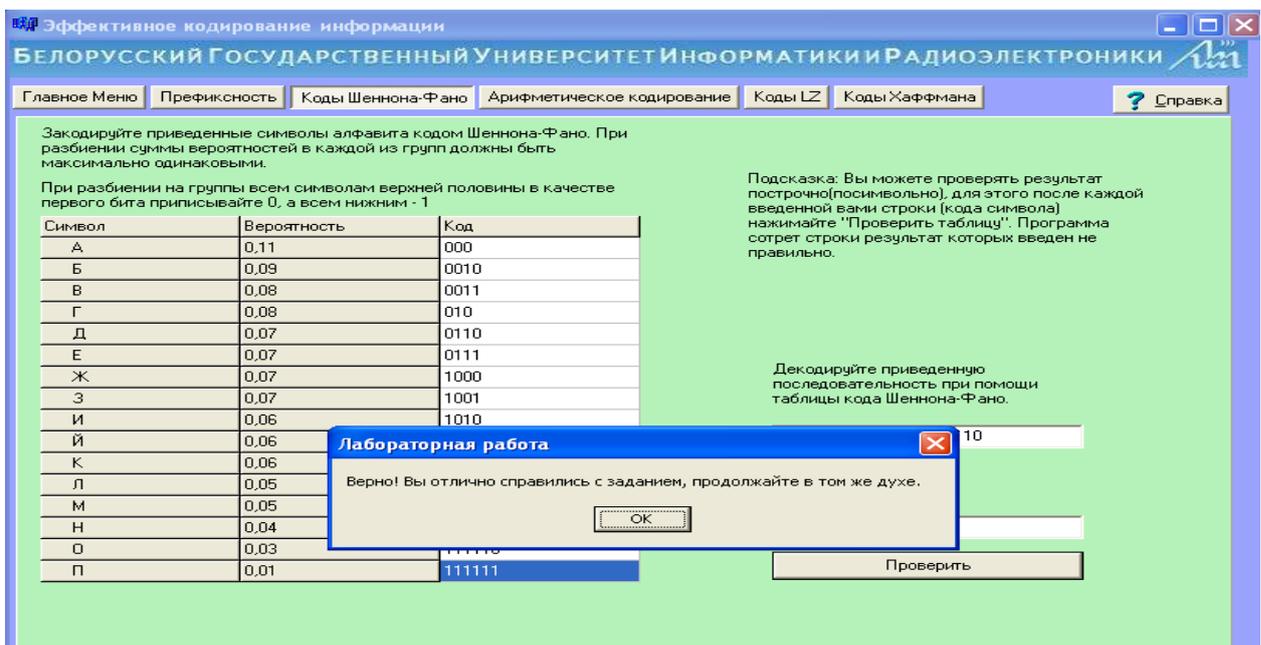


Рис. 2.7. Результат проверки

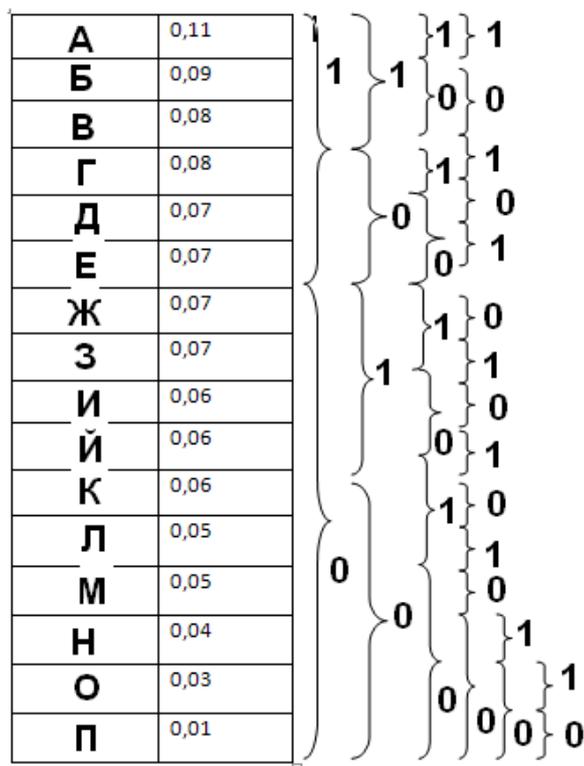


Рис. 2.8. Кодовое дерево

Преимуществом данного метода является его простота и, как следствие этого, высокая скорость кодирования и декодирования. Данный метод является простым в понимании, в реализации, а как следствие довольно эффективным. В настоящее время не используется, так как был усовершенствован Хаффманом, алгоритм которого в данный момент широко используется в связках с другими алгоритмами при кодировании/декодировании информации.

### 2.3. Исследование алгоритма Лемпеля - Зива

Основная идея алгоритма Лемпеля-Зива состоит в замене появления фрагмента в данных (группы байт) ссылкой на предыдущее появление этого фрагмента. Существуют два основных класса методов, названных в честь Якоба Зива (Jakob Ziv) и Абрахама Лемпеля (Abraham Lempel), впервые предложивших их в 1977 (алгоритм LZ77) и 1978 годах (алгоритм LZ78).

#### Алгоритм LZ77

LZ77 использует уже просмотренную часть сообщения как словарь. Чтобы добиться сжатия, он пытается заменить очередной фрагмент сообщения на указатель в содержимое словаря.

В качестве модели данных LZ77 использует “скользящее” по сообщению окно, разделенное на две неравные части. Первая, большая по размеру, включает уже

просмотренную часть сообщения. Вторая, намного меньшая, является буфером, содержащим еще не закодированные символы входного потока. Обычно размер окна составляет несколько килобайтов. Буфер намного меньше, обычно не более ста байтов. Алгоритм пытается найти в словаре фрагмент, совпадающий с содержимым буфера.

Алгоритм LZ77 выдает коды, состоящие из трех элементов:

- смещение в словаре относительно его начала подстроки, совпадающей с содержимым буфера;
- длина подстроки;
- первый символ в буфере, следующий за подстрокой.

Пример. Размер окна – 20 символов, словаря – 12 символов, а буфера -8. Кодировается сообщение «ПРОГРАММНЫЕ ПРОДУКТЫ ФИРМЫ MICROSOFT». Пусть словарь уже заполнен. Тогда он содержит строку «ПРОГРАММНЫЕ\_», а буфер строку «ПРОДУКТЫ». Просматривая словарь, алгоритм обнаружит, что совпадающей подстрокой будет «ПРО», в словаре она расположена со смещением 0 и имеет длину 3 символа, следующим символом в буфере является «Д». Таким образом, выходным кодом будет тройка (0, 3, «Д»). После этого алгоритм сдвигает влево все содержимое окна на длину совпадающей подстроки +1 и одновременно считывает столько же символов из входного потока в буфер. Получаем в словаре строку «РАММНЫЕ ПРОД», в буфере – «УКТЫ ФИР». В данной ситуации совпадающей подстроки обнаружить не удастся и алгоритм выдаст код (0, 0, «У»), после чего сдвинет окно на один символ. Затем словарь будет содержать «АММНЫЕ ПРОДУ», а буфер – «КТЫ ФИРМ». и т.д.

### **Проблемы LZ77**

Очевидно, что быстродействие кодера LZ77 сильно зависит от того, каким образом будет осуществляться поиск совпадающей подстроки в словаре. Если искать совпадение полным перебором всех возможных вариантов, то очевидно, что сжатие будет очень медленным. Причем при увеличении размеров окна для повышения степени сжатия скорость работы будет пропорционально уменьшаться. Для декодера это неважно, так как при декодировании не осуществляется поиск совпадения.

Быстродействие и кодера, и декодера зависит от того, как реализовано “скольжение” окна по содержимому сообщения. Рационально было бы для работы с окном пользоваться кольцевым буфером, организованным на физически сплошном участке памяти, а не реальным сдвигом содержимого окна. Хотя для поддержания кольцевого буфера необходимы дополнительные затраты на сохранение целостности индексов в нем, в целом это дает очень существенный выигрыш потому, что отсутствует постоянное сдвигание большого блока памяти. Если размер кольцевого буфера равен степени двойки, то

стандартная для кольцевого буфера операция “смещение по модулю размер”, может быть заменена побитовой логической операцией, что еще больше повышает эффективность.

Помимо проблем с быстродействием, у алгоритма LZ77 возникают проблемы с самим сжатием. Они появляются, когда кодер не может найти совпадающую подстроку в словаре и выдает стандартный 3-компонентный код, пытаясь закодировать один символ. Если словарь имеет длину  $4K$ , а буфер — 16 байтов, то код  $\langle 0, 0, \text{символ} \rangle$  будет занимать 3 байта. Кодирование одного байта в три имеет мало общего со сжатием и существенно понижает производительность алгоритма.

### **Упрощенный алгоритм LZ77**

пока (буфер\_предпросмотра не пуст) найти наиболее длинное соответствие (позиция\_начала, длина) в буфере\_предыстории и в буфере\_предпросмотра; если (длина > минимальной\_длины), то вывести в кодированные данные пару (позиция\_начала, длина); иначе вывести в кодированные данные первый символ буфера\_предпросмотра; изменить указатель на начало буфера\_предпросмотра и продолжить.

### **Алгоритм LZ78**

Этот алгоритм генерирует на основе входных данных словарь фрагментов, внося туда фрагменты данных (последовательности байт) по определенным правилам и присваивает фрагментам коды (номера). При сжатии данных (поступлении на вход программы очередной порции) программа на основе LZ78 пытается найти в словаре фрагмент максимальной длины, совпадающий с данными, заменяет найденную в словаре порцию данных кодом

фрагмента и дополняет словарь новым фрагментом. При заполнении всего словаря (размер словаря ограничен по определению) программа очищает словарь и начинает процесс заполнения словаря снова. Реализации этого метода различаются конструкцией словаря, алгоритмами его заполнения и очистки при переполнении.

Обычно, при инициализации словарь заполняется исходными (элементарными) фрагментами всеми возможными значениями байта от 0 до 255.

Это гарантирует, что при поступлении на вход очередной порции данных будет найден в словаре хотя бы однобайтовый фрагмент.

Алгоритм LZ78 резервирует специальный код, назовем его «Reset», который вставляется в упакованные данные, отмечая момент сброса словаря. Значение кода Reset обычно принимают равным 256.

Таким образом при начале кодирования минимальная длина кода составляет 9 бит. Максимальная длина кода зависит от объема словаря – количества различных фрагментов, которые туда можно поместить. Если объем словаря измерять в байтах (символах), то очевидно, что максимальное количество фрагментов равно числу символов, а,

следовательно, максимальная длина кода равна  $\log_2$  (объем словаря в байтах).

### Проведение эксперимента и обработка результатов

Кодирование фразы алгоритмом LZ77 со следующими размерами СЛОВАРЬ/БУФЕР

Таблица 2.3. Варианты расчетных заданий

ФРАЗА	СЛОВАРЬ/БУФЕР			
«IDI TYDA, NE ZNAU KUDA»	8/5	12/5	12/8	16/8

Используем ввод фразы с клавиатуры в программе LZ77.EXE для разных размеров словаря/буфера:

```

C:\DOCUME~1\Major\0016-1\1A1\11__3_-1\ZLZ77.EXE
Алгоритм Лемпела - Зива (LZ77)
Введите входной поток :IDI TYDA, NE ZNAU KUDA
Введите размер словаря:8
Введите размер буфера :5

!Словарь !Буфер!Код
!-----!IDI T!<0,0,"I">!
!-----!IDI TY!<0,0,"D">!
!-----!ID!I TYD!<6,1," ">!
!-----!IDI !TYDA,!<0,0,"T">!
!----!IDI T!YDA, !<0,0,"Y">!
!--!IDI TY!DA, N!<3,1,"A">!
!IDI TYDA!, NE !<0,0," ">!
!DI TYDA, ! NE Z!<2,1,"N">!
! TYDA, N!E ZNA!<0,0,"E">!
!TYDA, NE! ZNAU!<5,1,"Z">!
!DA, NE Z!NAU K!<4,1,"A">!
!, NE ZNA!U KUD!<0,0,"U">!
! NE ZNAU! KUDA!<0,1,"K">!
!E ZNAU K!UDA--!<5,1,"D">!
!ZNAU KUD!A----!<0,0,"A">!
Длина кода исходного сообщения:176
Длина кода полученного сообщения:210
  
```

Рис. 2.9. Ввод фразы с клавиатуры для размеров 8/5

```

C:\DOCUME~1\Major\0016-1\1A1\11__3_-1\ZLZ77.EXE
Алгоритм Лемпела - Зива (LZ77)
Введите входной поток :IDI TYDA, NE ZNAU KUDA
Введите размер словаря:12
Введите размер буфера :5

!Словарь      !Буфер!Код
!-----!IDI T!<0,0,"I">!
!-----!I !DI TY!<0,0,"D">!
!-----!ID!I TYD!<10,1," ">!
!-----!IDI !TYDA,!<0,0,"T">!
!-----!IDI T!YDA, !<0,0,"Y">!
!----!IDI TY!DA, N!<7,1,"A">!
!--!IDI TYDA!, NE !<0,0," ">!
!--!IDI TYDA, ! NE Z!<6,1,"N">!
!IDI TYDA, N!E ZNA!<0,0,"E">!
!IDI TYDA, NE! ZNAU!<3,1,"Z">!
!I TYDA, NE Z!NAU K!<8,1,"A">!
!TYDA, NE ZNA!U KUD!<0,0,"U">!
!YDA, NE ZNAU! KUDA!<4,1,"K">!
!A, NE ZNAU K!UDA--!<9,1,"D">!
! NE ZNAU KUD!A----!<0,0,"A">!
Длина кода исходного сообщения:176
Длина кода полученного сообщения:225
  
```

Рис. 2.10. Ввод фразы с клавиатуры для размеров 12/5

```

C:\DOCUME~1\Major\0016-1\1A1\11__3_-1\Z\Z77.EXE
Алгоритм Лемпела - Зива <LZ??>
Введите входной поток :IDI TYDA, NE ZNAU KUDA
Введите размер словаря:12
Введите размер буфера :8

!Словарь      !Буфер      !Код      !
!-----!
!IDI TYDA, NE ZNAU KUDA!<0,0,"I">!
!IDI TYDA, NE ZNAU KUDA!<0,0,"D">!
!IDI TYDA, NE ZNAU KUDA!<10,1," ">!
!IDI TYDA, NE ZNAU KUDA!<0,0,"T">!
!IDI TYDA, NE ZNAU KUDA!<0,0,"Y">!
!IDI TYDA, NE ZNAU KUDA!<7,1,"A">!
!IDI TYDA, NE ZNAU KUDA!<0,0," ">!
!IDI TYDA, NE ZNAU KUDA!<6,1,"N">!
!IDI TYDA, NE ZNAU KUDA!<0,0,"E">!
!IDI TYDA, NE ZNAU KUDA!<3,1,"Z">!
!IDI TYDA, NE ZNAU KUDA!<8,1,"A">!
!IDI TYDA, NE ZNAU KUDA!<0,0,"U">!
!IDI TYDA, NE ZNAU KUDA!<4,1,"K">!
!IDI TYDA, NE ZNAU KUDA!<9,1,"D">!
!IDI TYDA, NE ZNAU KUDA!<0,0,"A">!
Длина кода исходного сообщения:176
Длина кода полученного сообщения:240

```

Рис. 2.11. Ввод фразы с клавиатуры для размеров 12/8

```

C:\DOCUME~1\Major\0016-1\1A1\11__3_-1\Z\Z77.EXE
Алгоритм Лемпела - Зива <LZ??>
Введите входной поток :IDI TUDA, NE ZNAU KUDA
Введите размер словаря:16
Введите размер буфера :8

!Словарь      !Буфер      !Код      !
!-----!
!IDI TUDA, NE ZNAU KUDA!<0,0,"I">!
!IDI TUDA, NE ZNAU KUDA!<0,0,"D">!
!IDI TUDA, NE ZNAU KUDA!<14,1," ">!
!IDI TUDA, NE ZNAU KUDA!<0,0,"T">!
!IDI TUDA, NE ZNAU KUDA!<0,0,"U">!
!IDI TUDA, NE ZNAU KUDA!<11,1,"A">!
!IDI TUDA, NE ZNAU KUDA!<0,0," ">!
!IDI TUDA, NE ZNAU KUDA!<10,1,"N">!
!IDI TUDA, NE ZNAU KUDA!<0,0,"E">!
!IDI TUDA, NE ZNAU KUDA!<7,1,"Z">!
!IDI TUDA, NE ZNAU KUDA!<12,1,"A">!
!IDI TUDA, NE ZNAU KUDA!<5,1," ">!
!IDI TUDA, NE ZNAU KUDA!<0,0,"K">!
!IDI TUDA, NE ZNAU KUDA!<0,0,"U">!
Длина кода исходного сообщения:176
Длина кода полученного сообщения:224

```

Рис. 2.12. Ввод фразы с клавиатуры для размеров 16/8

Используем ввод текста из файла для разных размеров словаря/буфера:

Текст: «Полуадаптированное моделирование решает эту проблему, используя для каждого текста свою модель, которая строится еще до самого сжатия на основании результатов предварительного просмотра текста (или его образца). Перед тем, как окончено формирование сжатого текста, модель должна быть передана декодировщику. Несмотря на дополнительные затраты по передаче модели, эта стратегия в общем случае окупается благодаря лучшему соответствию модели тексту.

Адаптированное (или динамическое) моделирование уходит от связанных с этой передачей расходов. Первоначально и кодировщик, и декодировщик присваивают себе некоторую пустую модель, как если бы символы все были равновероятными. Кодировщик

использует эту модель для сжатия очередного символа, а раскодировщик - для его разворачивания. Затем они оба изменяют свои модели одинаковым образом (например, наращивая вероятность рассматриваемого символа). Следующий символ кодируется и достается на основании новой модели, а затем снова изменяет модель. Кодирование продолжается аналогичным раскодированию образом, которое поддерживает идентичную модель за счет применения такого же алгоритма ее изменения, обеспеченным отсутствием ошибок во время кодирования. Используемая модель, которую к тому же не нужно передавать явно, будет хорошо соответствовать сжатому тексту.

Адаптированные модели представляют собой элегантную и эффективную технику, и обеспечивают сжатие по крайней мере не худшее производимого неадаптированными схемами. Оно может быть значительно лучше, чем у плохо соответствующих текстам статичных моделей. Адаптированные модели, в отличие от полуадаптированных, не производят их предварительного просмотра, являясь поэтому более привлекательными и лучшесжимаемыми. Т.о. алгоритмы моделей, описываемые в подразделах, при кодировании и декодировании будут выполняться одинаково. Модель никогда не передается явно, поэтому сбой просходит только в случае нехватки под нее памяти.»

```

тэю, яю"Сюье ёсюц яЕюёЮюфшС Сююфью т ёмеурх эхТитрЕшь! <0,0,"ъ">
эю, яю"Сюье ёсюц яЕюёЮюфшС Сююфью т ёмеурх эхТитрЕшь! <1,1,"т">
. яю"Сюье ёсюц яЕюёЮюфшС Сююфью т ёмеурх эхТитрЕшь! <0,0,"т">
. яю"Сюье ёсюц яЕюёЮюфшС Сююфью т ёмеурх эхТитрЕшь! <8,2,"ь">
"Сюье ёсюц яЕюёЮюфшС Сююфью т ёмеурх эхТитрЕшь! <4,1,"з">
ьюе ёсюц яЕюёЮюфшС Сююфью т ёмеурх эхТитрЕшь! <0,0,"р">
ье ёсюц яЕюёЮюфшС Сююфью т ёмеурх эхТитрЕшь! <0,0,"х">
е ёсюц яЕюёЮюфшС Сююфью т ёмеурх эхТитрЕшь! <1,1,"э">
ёсюц яЕюёЮюфшС Сююфью т ёмеурх эхТитрЕшь! <29,1,"Т">
юц яЕюёЮюфшС Сююфью т ёмеурх эхТитрЕшь! <20,1,"р">
яЕюёЮюфшС Сююфью т ёмеурх эхТитрЕшь! <9,1,"ь">
ЕюёЮюфшС Сююфью т ёмеурх эхТитрЕшь! <6,1,"ъ">
ёЮюфшС Сююфью т ёмеурх эхТитрЕшь! <0,0,"я">
ЮюфшС Сююфью т ёмеурх эхТитрЕшь! <1,2,"ъ">
шС Сююфью т ёмеурх эхТитрЕшь! <19,2,"р">
Сююфью т ёмеурх эхТитрЕшь! <24,2,"р">
фью т ёмеурх эхТитрЕшь! <0,0,"ъ">
ью т ёмеурх эхТитрЕшь! <0,0,"ъ">
ю т ёмеурх эхТитрЕшь! <0,0,"ъ">
т ёмеурх эхТитрЕшь! <16,1,"ш">
т ёмеурх эхТитрЕшь! <0,0,"ъ">
ёмеурх эхТитрЕшь! <0,1,"ф">
ёмеурх эхТитрЕшь! <0,0,"ф">
ёмеурх эхТитрЕшь! <0,0,"ф">
Длина кода исходного сообщения :15544 буТ <1943 бауТ>
Длина кода полученного сообщения:15984 буТ <1998 бауТ>

```

Рис. 2.13. Ввод текста из файла для размеров 32/5

```

р эх яхЕхфрхСё тэю, яю"Сюье ёсюц яЕюёЮюфшС Сююфью т ёмеурх эхТитрЕшь! <17,1,"р">
э х я х Е х ф р х С ё тэю, яю"Сюье ёсюц яЕюёЮюфшС Сююфью т ёмеурх эхТитрЕшь! <10,1,"ъ">
. я х Е х ф р х С ё тэю, яю"Сюье ёсюц яЕюёЮюфшС Сююфью т ёмеурх эхТитрЕшь! <38,1,"ъ">
. х Е х ф р х С ё тэю, яю"Сюье ёсюц яЕюёЮюфшС Сююфью т ёмеурх эхТитрЕшь! <16,2,"ф">
. ф р х С ё тэю, яю"Сюье ёсюц яЕюёЮюфшС Сююфью т ёмеурх эхТитрЕшь! <51,3,"х">
. ё тэю, яю"Сюье ёсюц яЕюёЮюфшС Сююфью т ёмеурх эхТитрЕшь! <8,2,"р">
. тэю, яю"Сюье ёсюц яЕюёЮюфшС Сююфью т ёмеурх эхТитрЕшь! <11,1,"ъ">
. эю, яю"Сюье ёсюц яЕюёЮюфшС Сююфью т ёмеурх эхТитрЕшь! <7,1,"ш">
. яю"Сюье ёсюц яЕюёЮюфшС Сююфью т ёмеурх эхТитрЕшь! <0,0,"ъ">
. яю"Сюье ёсюц яЕюёЮюфшС Сююфью т ёмеурх эхТитрЕшь! <0,1,"ф">
. ю"Сюье ёсюц яЕюёЮюфшС Сююфью т ёмеурх эхТитрЕшь! <0,0,"ф">
Длина кода исходного сообщения :15544 буТ <1943 бауТ>
Длина кода полученного сообщения:15011 буТ <1876 бауТ>

```

Рис. 2.14. Ввод текста из файла для размеров 64/5

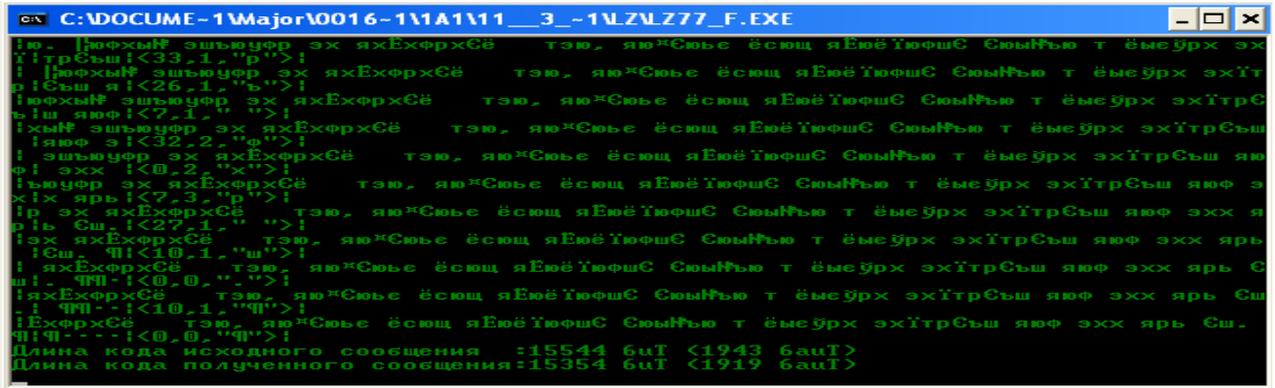


Рис. 2.15. Ввод текста из файла для размеров 80/5



Рис. 2.16. Ввод текста из файла для размеров 128/5

Таблица 2.4. Результаты измерений

D <sub>словарь</sub>	32	64	80	128
L <sub>вх</sub> /L <sub>вых</sub>	0,972	1,036	1,012	1,141

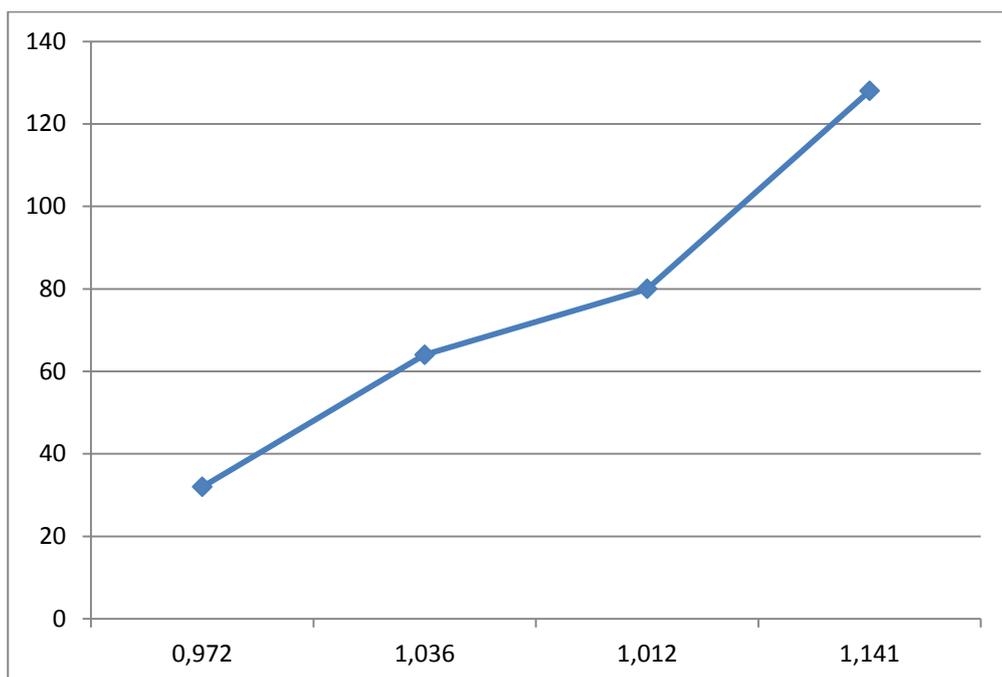


Рис. 2.17. График зависимости коэффициента сжатия от размера словаря  
 Кодирование для разной длины текста и одинакового размера словаря 64/5:

```

C:\DOCUMENT-1\Major\0016-1\1A1\11__3_-1\Z\Z77_F.EXE
>|-----Т хёе Еюшрёё! зюю!<50,1,"з
>|-----Т хёе Еюшрёё! зюю!<49,1,"ю
>|-----Т хёе Еюшрёё! зюю!<0,0,"з"
>|-----Т хёе Еюшрёё! зюю!<0,0,"з"
>|-----Т хёе Еюшрёё! зюю!<55,1,"
>|-----Т хёе Еюшрёё! зюю!<0,0,"т"
>|-----Т хёе Еюшрёё! зюю!<41,5,"
>|-----Т хёе Еюшрёё! зюю!<42,1,"э
>|-----Т хёе Еюшрёё! зюю!<53,2,"Е
>|-----Т хёе Еюшрёё! зюю!<37,1,"ё
>|-----Т хёе Еюшрёё! зюю!<0,0,"ы"
>|
Длина кода исходного сообщения :312 байт <39 байт>
Длина кода полученного сообщения:391 байт <49 байт>
  
```

Рис. 2.18. Ввод текста длиной 39 байт

```

C:\DOCUMENT-1\Major\0016-1\1A1\11__3_-1\Z\Z77_F.EXE
>|Ср ёррехуш т юс*хь ёмеурх юеярхСё сыр-уюфЕ меу°хье ёюСтхСёС!<16,1,"х
>|Ср ёррехуш т юс*хь ёмеурх юеярхСё сыр-уюфЕ меу°хье ёюСтхСёС!<35,2,"С
>|ёррехуш т юс*хь ёмеурх юеярхСё сыр-уюфЕ меу°хье ёюСтхСёС!тш||ь!<11,1,"ш
>|ёррехуш т юс*хь ёмеурх юеярхСё сыр-уюфЕ меу°хье ёюСтхСёС!||ьф!<0,0,"||"
>|ёррехуш т юс*хь ёмеурх юеярхСё сыр-уюфЕ меу°хье ёюСтхСёС!||ьфх!<7,1,"ь"
>|ехуш т юс*хь ёмеурх юеярхСё сыр-уюфЕ меу°хье ёюСтхСёС!ш||ьфхш!<36,2,"х
>|ш т юс*хь ёмеурх юеярхСё сыр-уюфЕ меу°хье ёюСтхСёС!ш||ьфх!ш Сх!<12,1,"ш
>|т юс*хь ёмеурх юеярхСё сыр-уюфЕ меу°хье ёюСтхСёС!ш||ьфхш! Схё!<0,1,"С"
>|юс*хь ёмеурх юеярхСё сыр-уюфЕ меу°хье ёюСтхСёС!ш||ьфхш С!хёёС!<4,1,"ь"
>|с*хь ёмеурх юеярхСё сыр-уюфЕ меу°хье ёюСтхСёС!ш||ьфхш Схё!ёС!<48,2,"ё
>|ь ёмеурх юеярхСё сыр-уюфЕ меу°хье ёюСтхСёС!ш||ьфхш СхёёС!<0,0,""
>|
Длина кода исходного сообщения :3616 байт <452 байт>
Длина кода полученного сообщения:3638 байт <455 байт>
  
```

Рис. 2.19. Ввод текста длиной 452 байта

```

C:\DOCUMENT-1\Major\0016-1\1A1\11__3_-1\Z\Z77_F.EXE
>|р эх яхЕхрхСё тэю, яю°Сюье ёсюц яЕюёЮфшС Сюмфью т ёмеурх эх!трСьш!<17,1,"р
>|эх яхЕхрхСё тэю, яю°Сюье ёсюц яЕюёЮфшС Сюмфью т ёмеурх эх!триСьш я!<10,1,"ь
>|яхЕхрхСё тэю, яю°Сюье ёсюц яЕюёЮфшС Сюмфью т ёмеурх эх!трСьш яюф!<38,1,"
>|хЕхрхСё тэю, яю°Сюье ёсюц яЕюёЮфшС Сюмфью т ёмеурх эх!трСьш |яюф э!<16,2,"ф
>|рхСё тэю, яю°Сюье ёсюц яЕюёЮфшС Сюмфью т ёмеурх эх!трСьш яюф! эхх !<51,3,"х
>|ё тэю, яю°Сюье ёсюц яЕюёЮфшС Сюмфью т ёмеурх эх!трСьш яюф эхх! ярь !<8,2,"р"
>|тэю, яю°Сюье ёсюц яЕюёЮфшС Сюмфью т ёмеурх эх!трСьш яюф эхх ярь !Сш. !<11,1,"
>|тэю, яю°Сюье ёсюц яЕюёЮфшС Сюмфью т ёмеурх эх!трСьш яюф эхх ярь !Сш. !<7,1,"ш"
>| яю°Сюье ёсюц яЕюёЮфшС Сюмфью т ёмеурх эх!трСьш яюф эхх ярь Сш!. !<0,0,".
>| яю°Сюье ёсюц яЕюёЮфшС Сюмфью т ёмеурх эх!трСьш яюф эхх ярь Сш!. !<0,1,"!"
>| ю°Сюье ёсюц яЕюёЮфшС Сюмфью т ёмеурх эх!трСьш яюф эхх ярь Сш. !<0,0,"!"
>|
Длина кода исходного сообщения :11904 байт <1488 байт>
Длина кода полученного сообщения:11475 байт <1434 байт>
  
```

Рис. 2.20. Ввод текста длиной 1488 байт

```

C:\> C:\DOCUME~1\Major\0016-11A1111__3_-1\Z\Z77_F.EXE
! яояем ЕэЧьш т эрёСю `хх тЕхь ьхСюфрьш, Срьшьш, эр-яЕшьхЕ, ьрь ёцрСшх !<17,1,"ц
!>!
! яояем ЕэЧьш т эрёСю `хх тЕхь ьхСюфрьш, Срьшьш, эр-яЕшьхЕ, ьрь ёцрСшх !<14,1,"С
!>!
! ем ЕэЧьш т эрёСю `хх тЕхь ьхСюфрьш, Срьшьш, эр-яЕшьхЕ, ьрь ёцрСшх ||ш!<7,1,"х"
!>!
! ЕэЧьш т эрёСю `хх тЕхь ьхСюфрьш, Срьшьш, эр-яЕшьхЕ, ьрь ёцрСшх! ||штр!<6,1,"||"
!>!
! эЧьш т эрёСю `хх тЕхь ьхСюфрьш, Срьшьш, эр-яЕшьхЕ, ьрь ёцрСшх ||штр-т!<3,1,"т"
!>!
! ьш т эрёСю `хх тЕхь ьхСюфрьш, Срьшьш, эр-яЕшьхЕ, ьрь ёцрСшх ||штр-трь!<40,2,"т"
!>!
! т эрёСю `хх тЕхь ьхСюфрьш, Срьшьш, эр-яЕшьхЕ, ьрь ёцрСшх ||штр-т!хьяхм!<14,2,"я"
!>!
! рёСю `хх тЕхь ьхСюфрьш, Срьшьш, эр-яЕшьхЕ, ьрь ёцрСшх ||штр-трь!хьяхмр. !<6,1,"м"
!>!
! Сю `хх тЕхь ьхСюфрьш, Срьшьш, эр-яЕшьхЕ, ьрь ёцрСшх ||штр-трь!хьяхмр. !<18,1,"."
!>!
! `хх тЕхь ьхСюфрьш, Срьшьш, эр-яЕшьхЕ, ьрь ёцрСшх ||штр-трь!хьяхмр. !<4,1,""
!>!
! хх тЕхь ьхСюфрьш, Срьшьш, эр-яЕшьхЕ, ьрь ёцрСшх ||штр-трь!хьяхмр. !<0,0,""
!>!
Длина кода исходного сообщения :34336 байт (4292 байт)
Длина кода полученного сообщения:33388 байт (4174 байт)

```

Рис. 2.21. Ввод текста длиной 4292 байта

Таблица 2.5. Результаты измерений

$L_{ВХ}$	39	452	1488	4292
$L_{ВХ}/L_{ВЫХ}$	0,795	0,993	1,037	1,028

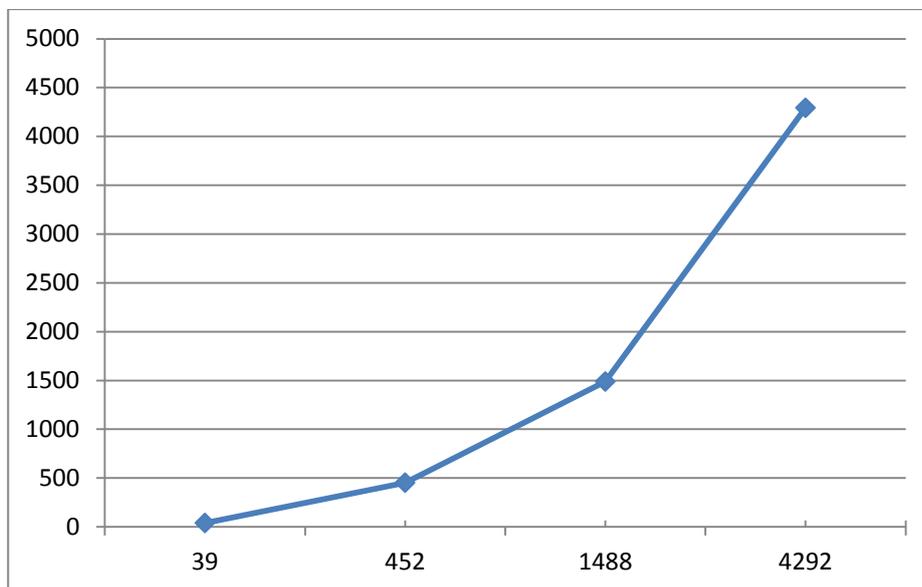


Рис. 2.22. График зависимости коэффициента сжатия от размера словаря

При кодировании коротких текстов коэффициент сжатия оказывается меньше 1, т.е. избыточность не удаляется, а вводится еще больше. При увеличении размера словаря коэффициент сжатия увеличивается. Большой размер буфера также ухудшает коэффициент сжатия. Если размер словаря кратен степеням двойки, то сжатие лучше. Однако, при

большом размере словаря снижается скорость кодирования. При увеличении размера текста коэффициент сжатия увеличивается.

#### **2.4. Фрактальные методы кодирования изображений**

Современные компьютеры весьма интенсивно применяют графику. Операционные системы с интерфейсом оконного типа используют картинки, например, для отображения директорий или папок. Некоторые совершаемые системой действия, например загрузку и пересылку файлов, также отображаются графически. Многие программы и приложения предлагают пользователю графический интерфейс (GUI), который значительно упрощает работу пользователя и позволяет легко интерпретировать полученные результаты. Компьютерная графика используется во многих областях повседневной деятельности при переводе сложных массивов данных в графическое представление.

Фрактал (лат. fractus — дроблёный, сломанный, разбитый) — геометрическая фигура, обладающая свойством самоподобия, то есть составленная из нескольких частей, каждая из которых подобна всей фигуре целиком. В математике под фракталами понимают множества точек в евклидовом пространстве, имеющие дробную метрическую размерность (в смысле Минковского или Хаусдорфа), либо метрическую размерность, отличную от топологической [14]. Слово «фрактал» может употребляться не только как математический термин. Фракталом в прессе и научно-популярной литературе могут называть фигуры, обладающие какими-либо из перечисленных ниже свойств: Обладает нетривиальной структурой на всех масштабах. В этом отличие от регулярных фигур (таких, как окружность, эллипс, график гладкой функции): если мы рассмотрим небольшой фрагмент регулярной фигуры в очень крупном масштабе, он будет похож на фрагмент прямой. Для фрактала увеличение масштаба не ведёт к упрощению структуры, на всех шкалах мы увидим одинаково сложную картину.

Является самоподобной или приближённо самоподобной. Обладает дробной метрической размерностью или метрической размерностью, превосходящей топологическую. Многие объекты в природе обладают фрактальными свойствами, например, побережья, облака, кроны деревьев, снежинки, кровеносная система и система альвеол человека или животных.

Фракталы, особенно на плоскости, популярны благодаря сочетанию красоты с простотой построения при помощи компьютера. Первые примеры самоподобных множеств с необычными свойствами появились в XIX веке (например, множество Кантора). Термин «фрактал» был введён Бенуа Мандельбротом в 1975 году и получил широкую популярность с выходом в 1977 году его книги «Фрактальная геометрия природы».

При фрактальном сжатии изображения для области меньшего размера подыскивается похожая на нее область большего размера того же изображения (рисунок 2.23).

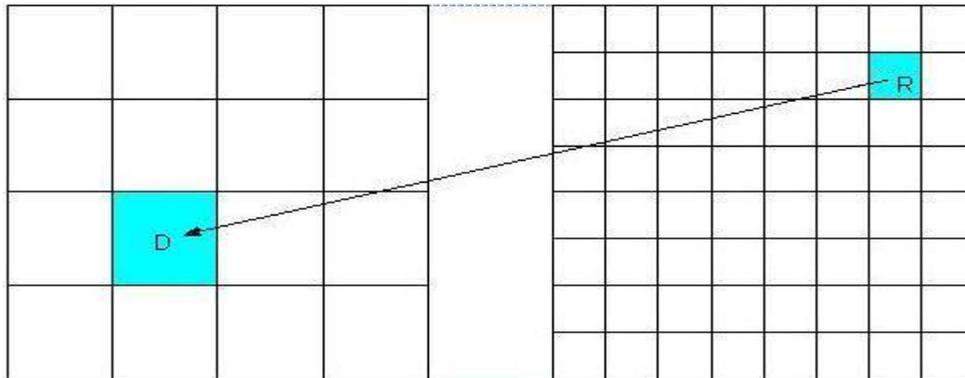


Рис. 2.23. Доменный и ранговый блоки

Область большого размера называется доменным блоком, а меньшего – ранговым блоком. Перевод доменного блока в ранговый осуществляется посредством аффинных преобразований, которые в случае изображения в оттенках серого цвета можно представить следующей системой уравнений (2.9):

$$\begin{bmatrix} x^* \\ y^* \\ z^* \end{bmatrix} = \begin{bmatrix} a & b & 0 \\ c & d & 0 \\ 0 & 0 & u \end{bmatrix} \begin{bmatrix} x \\ y \\ z \end{bmatrix} + \begin{bmatrix} e \\ f \\ v \end{bmatrix}, \quad (2.9)$$

где  $x$ ,  $y$ ,  $z$  – соответственно координаты и яркость пикселя доменного блока;  $x^*$ ,  $y^*$ ,  $z^*$  – соответственно координаты и яркость пикселя рангового блока;  $a$ ,  $b$ ,  $c$ ,  $d$ ,  $e$ ,  $f$  – коэффициенты преобразований координат на плоскости;  $u$  – коэффициент сжатия яркости;  $v$  – сдвиг по яркости.

В рассматриваемых алгоритмах фрактального сжатия и восстановления изображения введенные упрощения позволяют заменить матричные преобразования операциями изменения ориентации доменного блока и операциями расчета яркости пикселей. Изменения ориентации доменного блока осуществляется за счет его поворота на угол кратный  $90^\circ$  и зеркального отражения и поворота зеркального отражения.

Из всех возможных доменных блоков выбирается блок, ближайший (в выбранной метрике) к рассматриваемому ранговому блоку. Когда доменный блок найден, то запоминаются его номер и параметры преобразования в текущий ранговый блок, из которых и состоит решение задачи фрактального сжатия.

Ниже, на рисунке 2.32, представлен алгоритм фрактального сжатия.



Рис. 2.24. Алгоритм фрактального сжатия

#### Программное обеспечение в DELPHI 7

Приложение выполняет фрактальное сжатие / распаковку изображений с помощью классического алгоритма.

В нее можно загружать любые изображения, но размер должен быть не более 512x512 пикселей. Программа будет автоматически убирать цвет изображений.

Эти ограничения введены для того, чтобы существенно сократить время сжатия изображений на базе фрактального алгоритма.

Основные характеристики:

- Сжатие и декодирование изображений формата .bmp;
- Просмотр полученного результата;
- Просмотр размер полученного изображения.

На рисунке 2.35. представлено главное и единственное меню программы.

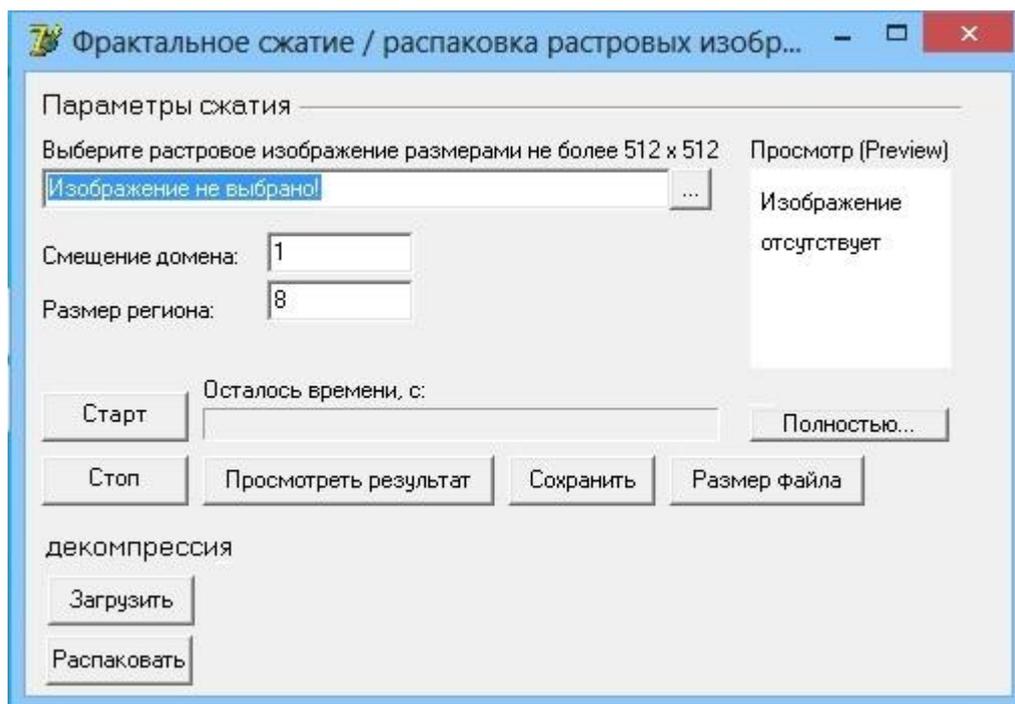


Рис. 2.25. Главное меню программы.

В программе доступны для изменения 2 следующих пункта: смещение домена и размер региона.

**Смещение домена:** Определяет шаг поиска участка в доменном изображении. Минимальный шаг равен 1. Чем больше шаг, тем быстрее выполняется поиск, но при этом могут быть пропущены важные детали изображения.

**Размер региона:** определяет размер области, на которое разбивается исходное изображение. При компрессии для каждой области осуществляется поиск подходящего домена с учетом трансформации (аффинных преобразований). Чем больше размер региона, тем хуже качество и при этом уменьшается размер IFS-изображения.

#### **Методика работы в программе и проведение исследования основных технических характеристик системы**

Для обеспечения фрактального сжатия была выбрана программа «Фрактальное сжатие / распаковка растровых изображений».

Для того чтобы начать работать с программой необходимо:

1. Загрузить изображение, которое удовлетворяет требованию программы;
2. Выставить значение параметров «Смещение домена» и «Размер региона»;
3. Нажать кнопку старт;

4. Дождаться окончания сжатия изображения;
5. Просмотреть полученный результат и, если он удовлетворил нас, сохранить его.

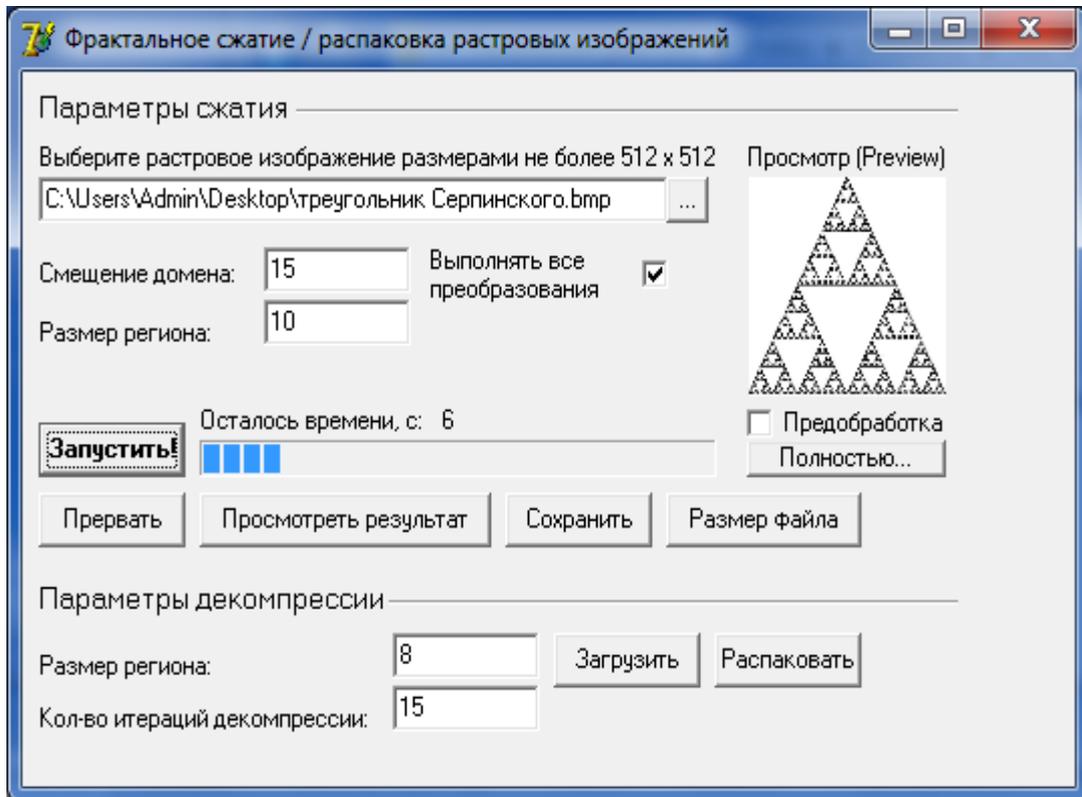


Рис. 2.26. Сжатие изображения в программе «Фрактальное сжатие / распаковка растровых изображений»

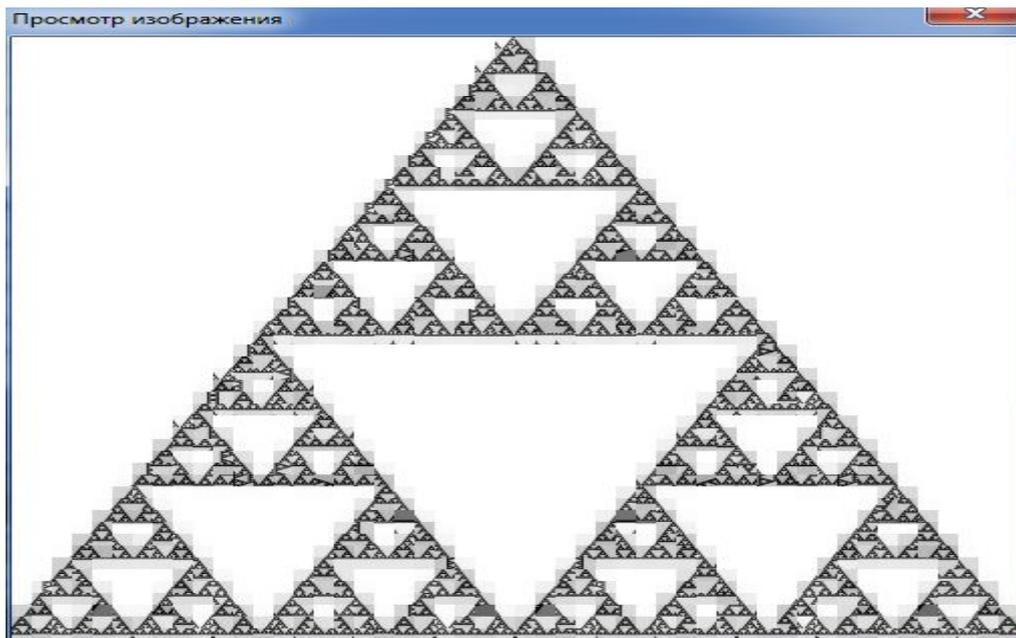


Рис. 2.27. Результат сжатия.

После сжатия изображение будет иметь формат .IFS.

Стандартные кодеки Windows не способны декодировать полученное изображение, поэтому в программе предусмотрена функция просмотра изображений формата .IFS.

Чтобы просмотреть нужное нам изображение необходимо:

1. В меню «декомпрессия» нажать на кнопку «Загрузить»;
2. Выбрать изображение формата .IFS;
3. В меню «декомпрессия» нажать на кнопку «Распаковать»

Далее рассмотрим сжатие изображений со спутника X-SAR Европейского космического агентства. На рисунке 2.36 представлено первоначальное изображение со спутника размером 435 Кб и разрешением 473x314 пикселей.



Рис. 2.28. Первоначальное изображение.

Далее, на рисунке 2.29, представлено это же изображение после обработки, при параметре смещение домена=1 и размер региона=8, время потраченное на сжатие  $t=703$  с, размер файла =11,4Кб.

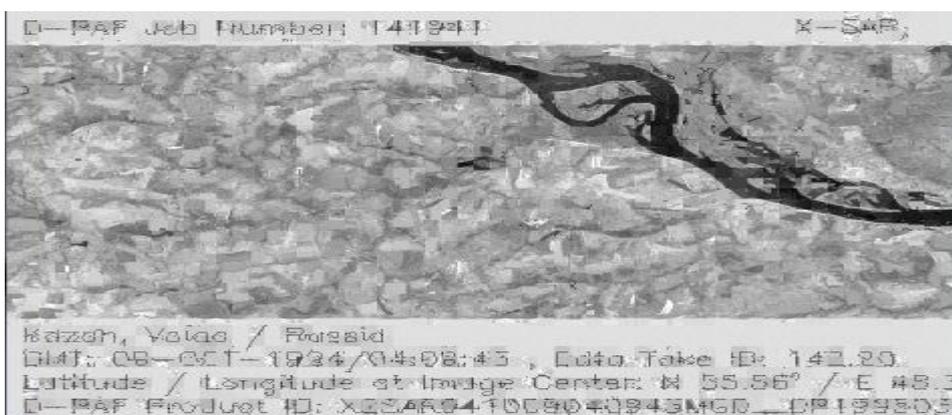


Рис. 2.29. Обработанное изображение.

Далее, на рисунке 2.30, представлено это же изображение после обработки, при параметре смещение домена=10 и размер региона=8, время потраченное на сжатие  $t=6$  с, размер файла =11Кб.

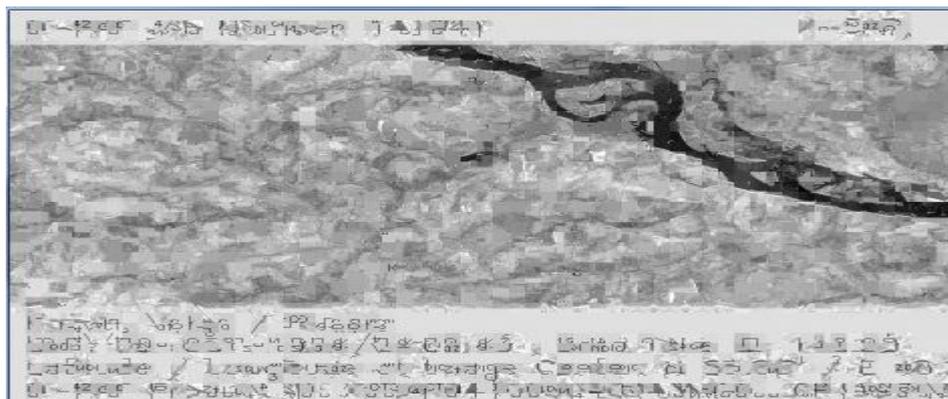


Рис. 2.30. Обработанное изображение.

Далее, на рисунке 2.31, представлено это же изображение после обработки, при параметре смещение домена=5 и размер региона=10, время потраченное на сжатие  $t=27$  с, размер файла =7,12Кб.

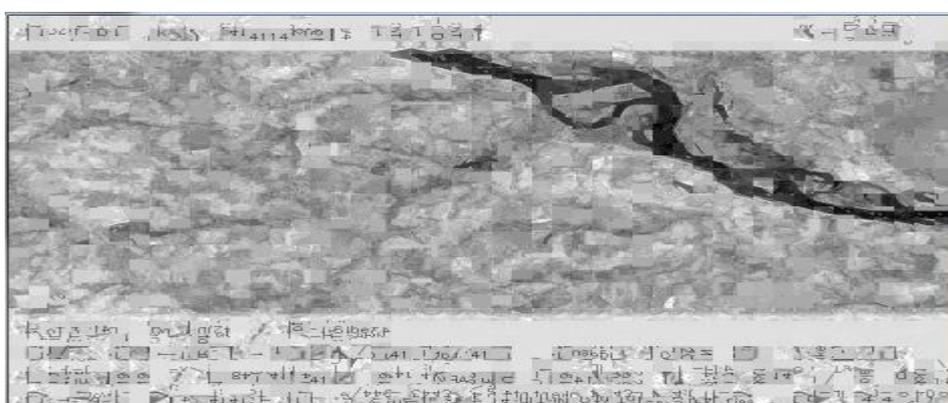


Рис. 2.31. Обработанное изображение.

Далее, на рисунке 2.32, представлено это же изображение после обработки, при параметре смещение домена=10 и размер региона=12, время потраченное на сжатие  $t=5$  с, размер файла =4,95Кб.

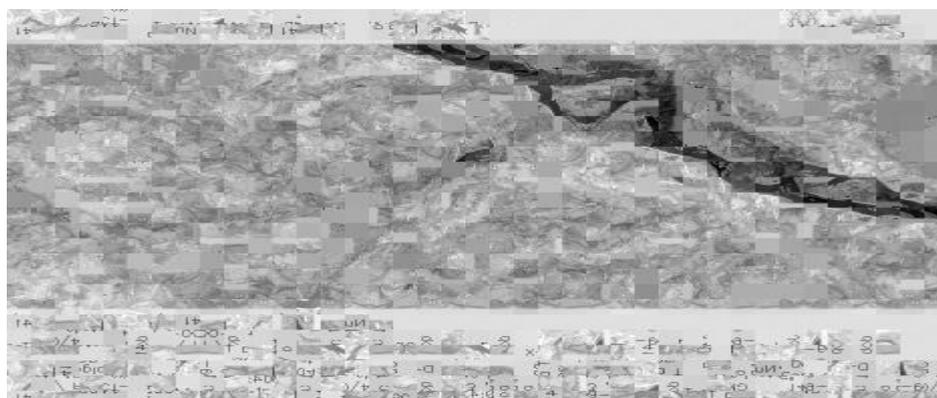


Рис. 2.32. Обработанное изображение.

Далее, на рисунке 2.33, представлено это же изображение после обработки, при параметре смещение домена=1 и размер региона=15, время потраченное на сжатие  $t=566$  с, размер файла =3,04Кб.

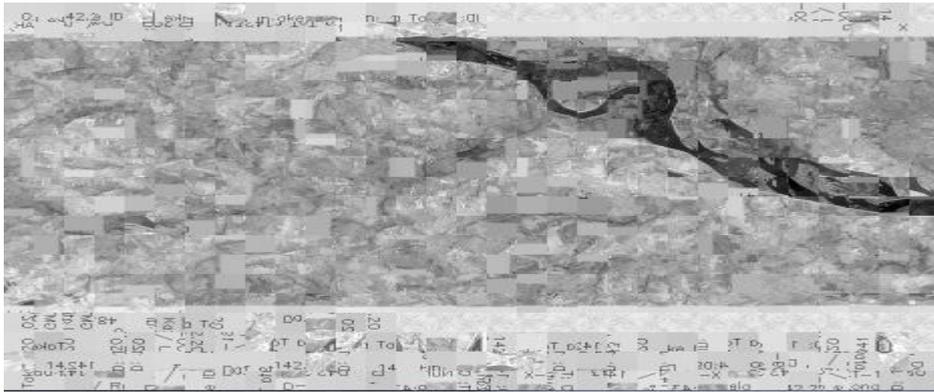


Рис. 2.33. Обработанное изображение.

Далее, на рисунке 2.34, представлено это же изображение после обработки, при параметре смещение домена=10 и размер региона=14, время потраченное на сжатие  $t=4$  с, размер файла =3,55Кб.

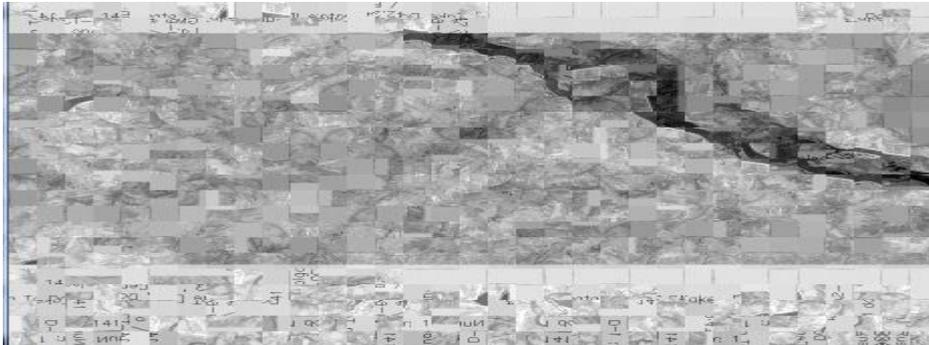


Рис. 2.34. Обработанное изображение.

Далее, на рисунке 2.35, представлено это же изображение после обработки, при параметре смещение домена=20 и размер региона=10, время потраченное на сжатие  $t=1$  с, размер файла =4,12Кб.

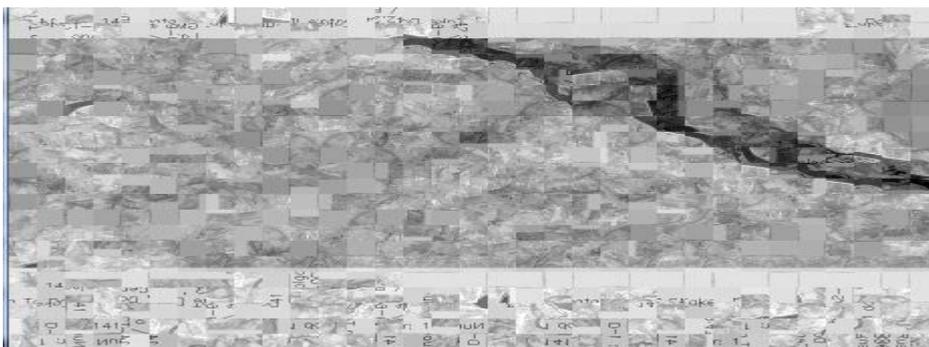


Рис. 2.35. Обработанное изображение.

Ниже, на рисунке 2.36 представлен график зависимости времени сжатия от размера изображения.

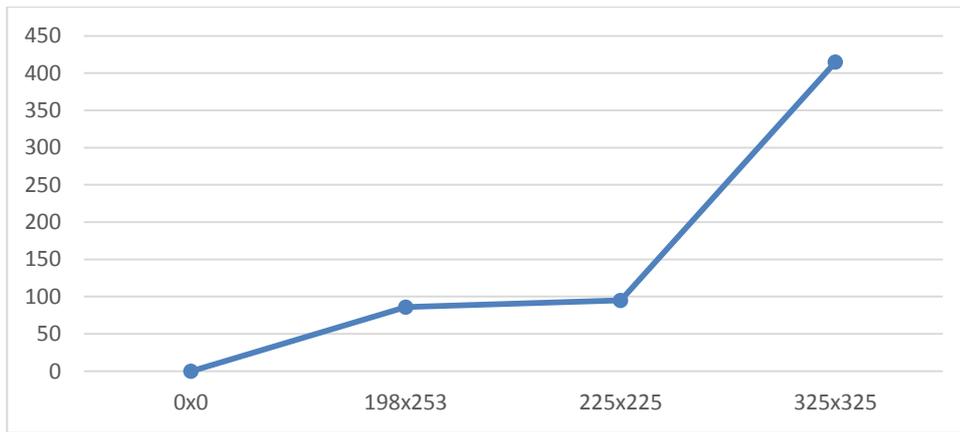


Рис. 2.36. График зависимости времени сжатия от размера изображения.

На рисунке 2.37 представлен график зависимости времени сжатия от параметра «смещение домена»

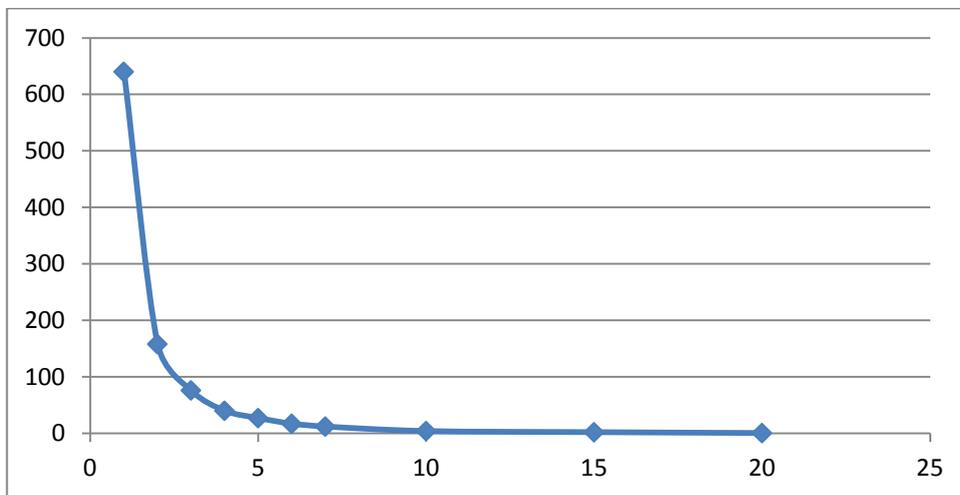


Рис. 2.37. График зависимости времени сжатия от параметра «смещение домена».

На рисунке 2.38 представлен график зависимости времени сжатия от параметра «размер региона».

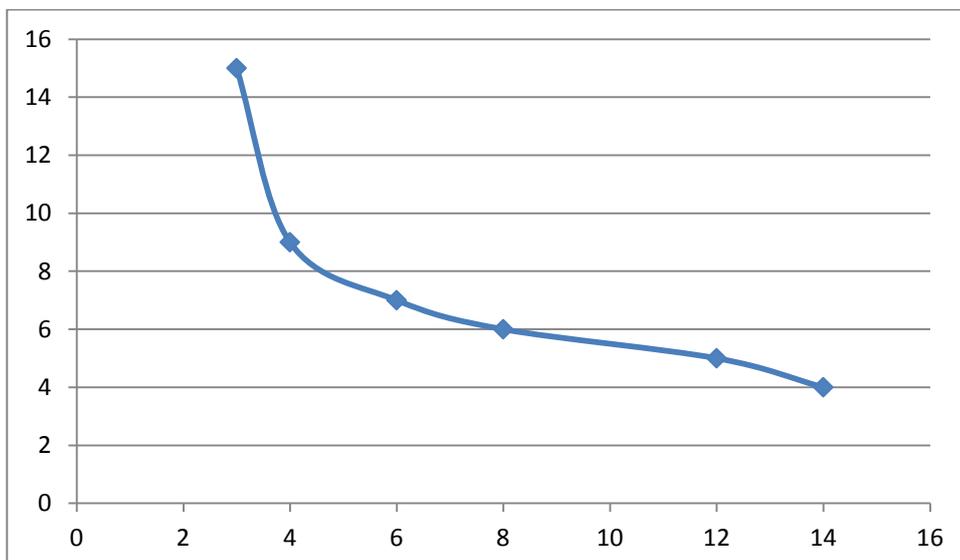


Рис. 2.38. График зависимости времени сжатия от параметра «размер региона».

На рисунке 2.39 представлен график зависимости размера изображения (в КБайт) от параметра «размер региона»

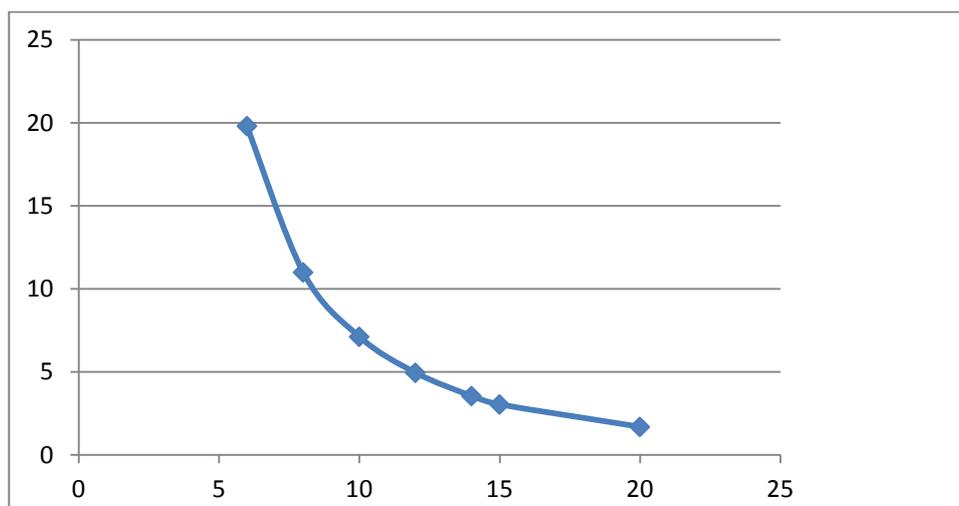


Рис. 2.39. График зависимости размера изображения (в КБайт) от параметра «размер региона»

Ниже, в таблице 2.6 представлена зависимость размера от коэффициента сжатия.

Таблица 2.6. Зависимость размера изображения от коэффициента сжатия.

Коэф-т сжатия	143	122	105	88	61	40
Размер изображения, Кб	3,04	3,55	4,12	4,95	7,12	11

Как видно из таблицы, при максимально достигнутом коэффициенте сжатия равному 143, размер изображения уменьшился с 435Кб до 3,04Кб.

Из представленных выше графиков можно сделать вывод, что:

1. Чем больше изображение, тем больше время сжатия;
2. Чем больше параметр «смещение домена», тем меньше время сжатия;
3. Чем меньше параметр «размер региона», тем больше время сжатия;
4. Чем меньше параметр «размер региона», тем больше размер изображения.

Рассмотрим таблицу 2.4, в которой сводятся воедино параметры различных алгоритмов сжатия изображений [13].

Таблица 2.7. Алгоритмы сжатия

Алгоритм	К-ты сжатия	На что ориентирован	Потери
RLE	32, 2, 0.5	3,4-х битные	Нет
LZW	1000, 4, 5/7	1-8 битные	Нет
Хаффмана	8, 1.5, 1	8 битные	Нет
СCITT-3	213(3), 5, 0.25	1-битные	Нет
JBIG	2-30 раз	1-битные	Нет
Lossless JPEG	2 раза	24-битные, серые	Нет
JPEG	2-20 раз	24-битные, серые	Да
Рекурсивное сжатие	2-200 раз	24-битные, серые	Да
Фрактальный	2-2000 раз	24-битные, серые	Да

Использование сжатия с потерями предоставляет возможность за счет потерь регулировать качество изображений. Коэффициенты сжатия у фрактальных алгоритмов варьируются в пределах 2-2000 раз. Причем большие коэффициенты достигаются на реальных изображениях, что нетипично для предшествующих алгоритмов. Ниже представлен график зависимости размера изображения от коэффициента сжатия

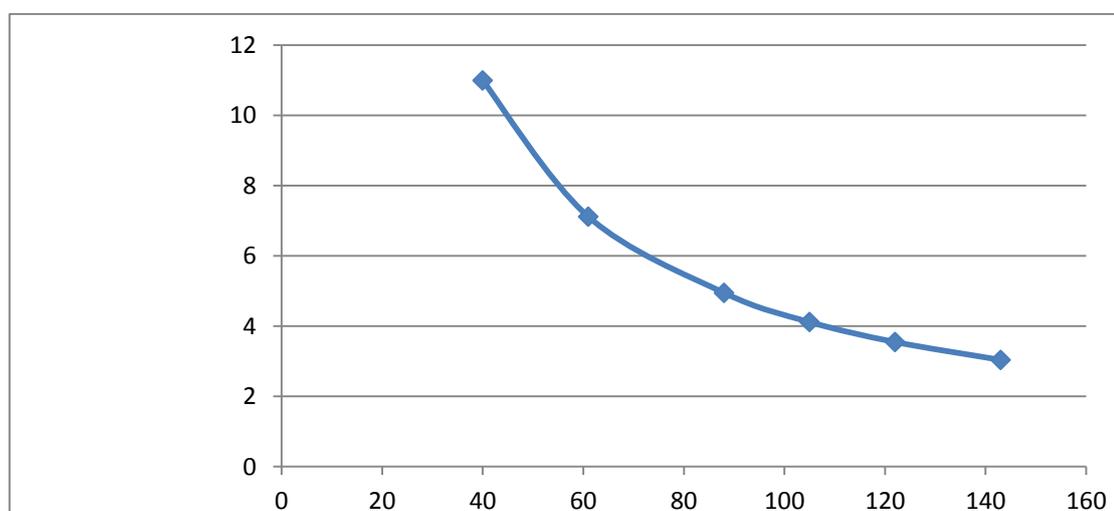


Рис. 2.40. Зависимость размера изображения по оси ординат, от коэффициента сжатия по оси абсцисс

Ниже, для наглядности, приведены 2 рисунка, первый – исходное изображение; второй – изображение с максимально достигнутом коэффициентом сжатия равным 143.

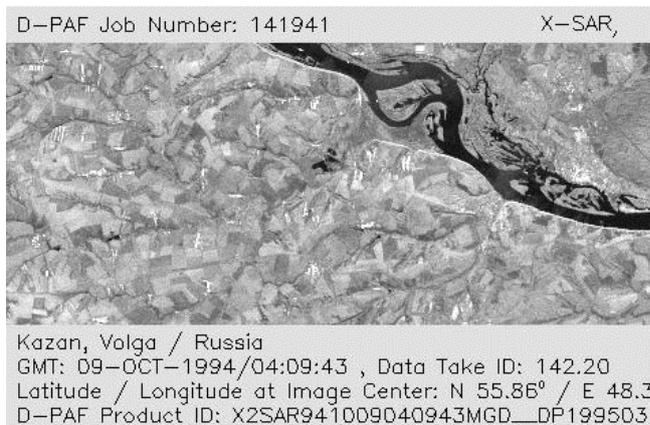


Рис. 2.41. Исходное изображение

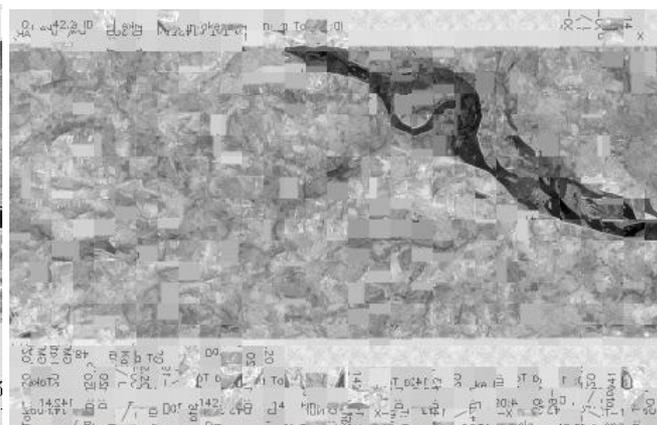


Рис. 2.42. Изображение с коэффициентом сжатия 143

Как видно из рисунков, при сжатии текстовой информации она становится нечитабельная, хотя как всё изображение в целом остаётся узнаваемым.

Недостатком этого алгоритма является потребность в больших вычислительных мощностях при архивации. Фактически это первый существенно несимметричный алгоритм. Причем, если у всех предшествующих алгоритмов коэффициент симметричности (отношение времени архивации ко времени разархивации) не превышает 3, то у фрактального алгоритма он колеблется от 1000 до 10000.

## 2.5. Вейвлет преобразования сигналов и изображений [15]

В настоящее время вейвлет-анализ является одним из наиболее мощных и при этом гибких средств исследования данных: помимо возможностей сжатия и фильтрации данных, анализ в базисе вейвлет-функций позволяет решать задачи идентификации, моделирования, аппроксимации стационарных и нестационарных процессов, исследовать вопросы наличия разрывов в производных, осуществлять поиск точек склеивания данных, удалять в данных тренд, отыскивать признаки фрактальности информации. Стоит отметить, что в основе подобных возможностей, обеспечивающих вейвлет-анализу весьма перспективное будущее, лежит природа его многомасштабности. Иначе говоря, гармонический анализ не способен конкурировать с вейвлет-анализом.

Вейвлет-преобразование широко используется для анализа сигналов. Помимо этого, оно находит большое применение в области сжатия данных. В дискретном вейвлет-преобразовании наиболее значимая информация в сигнале содержится при высоких амплитудах, а менее полезная — при низких. Сжатие данных может быть получено за счет отбрасывания низких амплитуд. Вейвлет-преобразование позволяет получить высокое

соотношение сжатия в сочетании с хорошим качеством восстановленного сигнала. Вейвлет-преобразование было выбрано для стандартов сжатия изображений JPEG2000 и ICER. Однако, при малых сжатиях вейвлет-преобразование уступает по качеству в сравнении с оконным Фурье-преобразованием, которое лежит в основе стандарта JPEG.

Выбор конкретного вида и типа вейвлетов во многом зависит от анализируемых сигналов и задач анализа. Для получения оптимальных алгоритмов преобразования разработаны определенные критерии, но их еще нельзя считать окончательными, так как они являются внутренними по отношению к самим алгоритмам преобразования и, как правило, не учитывают внешних критериев, связанных с сигналами и целями их преобразований. Отсюда следует, что при практическом использовании вейвлетов необходимо уделять достаточное внимание проверке их работоспособности и эффективности для поставленных целей по сравнению с известными методами обработки и анализа.

Изобретение вейвлетов было напрямую связано с необходимостью более глубокого анализа сигналов, чем анализ сигнала с помощью преобразования Фурье. Вейвлеты используют в тех случаях, когда результат анализа некоторого сигнала должен содержать не только простое перечисление его характерных частот (масштабов), но и сведения об определенных локальных координатах, при которых эти частоты проявляют себя. Анализ и обработка нестационарных (во времени) или неоднородных (в пространстве) сигналов разных типов представляют собой основное поле применения вейвлет-анализа.

Вейвлеты – новые системы базисных функций, используемых для представления, фильтрации, сжатия, хранения и т.д. любого из «сигналов»

$$f : R^n \rightarrow C. \quad (2.10)$$

Если  $n = 1$ , переменная  $t$  представляет собой время, и мы работаем с временными сигналами.  $f : R \rightarrow C$ . Случай  $n=2$  относится к обработке изображений. То есть, вейвлет представляет собой функцию от времени, которая используется для анализа одномерного или двумерного сигнала.

В своих работах Малл определяет вейвлет как функцию  $\phi$  с нулевым средним значением:

$$\int_{-\infty}^{\infty} \phi(t) dt = 0. \quad (2.11)$$

Вейвлетом называется волновое колебание с начальным значением амплитуды, равным нулю, затем увеличивающимся значением и затем снова уменьшающейся до нулевого значения. Это выражается в небольшом колебании исследуемого сигнала. Такое поведение этих функций позволяет записать и исследовать сейсмические волны или колебания сердца.

В общем случае, вейвлеты представляют собой функции, имеющие специфические свойства, позволяющие эффективно обрабатывать сигналы. Вейвлеты могут комбинироваться (с применением операций сдвига, умножения и суммирования), с выборками изучаемого сигнала для получения соответствующей информации.

Одна из основополагающих идей использования вейвлетов для представления сигналов заключается в разбивке приближения к сигналу на две составляющие: грубую (аппроксимирующую) и утонченную (детализирующую), с последующим их уточнением итерационным методом. Каждый шаг такого уточнения соответствует определенному уровню декомпозиции и восстановления сигнала. Это возможно как во временной, так и в частотной областях представления сигнала вейвлетами. Вейвлеты применяются во многих областях науки; главные области применения – обработка и анализ сигналов и сжатие изображений.

Для обработки сигнала с помощью вейвлетов сначала выбирается анализирующий (материнский) вейвлет. Обозначим его  $x \mapsto \phi(x)$ . Как правило, они определены на компактном носителе (на промежутке  $[0, L]$ ). Под носителем функции понимается ее область определения.

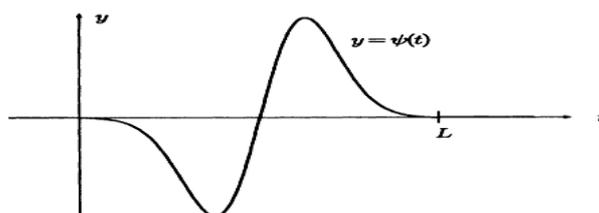


Рис. 2.43. Материнский вейвлет

Растянутые и сдвинутые копии вейвлета  $\phi$  называются вейвлетными функциями. Для них принято следующее обозначение:

$$\varphi_{a,b} : \mathbb{R} \rightarrow \mathbb{C}, t \mapsto \frac{1}{|a|^{1/2}} \psi\left(\frac{t-b}{a}\right). \quad (2.12)$$

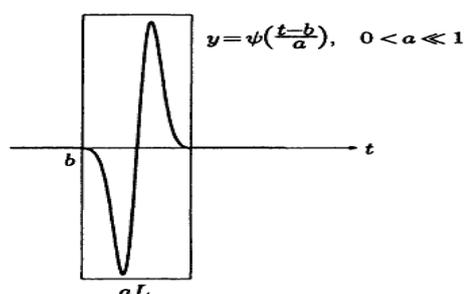


Рис. 2.44. Общий вид совокупности волновых функций (вейвлет-пакетов)

Параметр  $a$  называется масштабирующим параметром, а параметр  $b$  – параметром сдвига.

Результат выполнения вейвлет-преобразования (массив данных, получающийся в результате) полностью зависит от материнского вейвлета.

На практике используются следующие типы вейвлетов:

- Вейвлет Хаара.
- Мексиканская шляпа.
- Модулированная гауссова кривая.
- Производная гауссовой кривой.
- Вейвлеты Добеши, Грассмана, Мейера.
- Вейвлеты Бэттла-Лемарье.
- Симлеты.
- Койфлеты.

### Математический аппарат вейвлета Хаара

Вейвлет Хаара представляет собой следующую функцию:

$$\psi(x) = \begin{cases} 1, (0 \leq x < \frac{1}{2}) \\ -1, (\frac{1}{2} \leq x < 1) \end{cases} \quad (2.13)$$

Используя в качестве материнской функции вейвлет Хаара, можно определить вейвлетные функции.

$$\psi_{r,k} = 2^{-r/2} \psi_{haar} \left( \frac{t - k \cdot 2^r}{2^r} \right), r, k \in \mathbb{Z} \quad (2.14)$$

В качестве носителя данная функция имеет интервал длины  $2^r$ :

$$I_{r,k} = [k \cdot 2^r, (k+1) \cdot 2^r] \quad (2.15)$$

Большим значениям  $r$  соответствуют большие интервалы  $L$ , следовательно, вейвлетные функции имитируют большие волны. Одним из примеров, иллюстрирующих применение вейвлетов, является разложение сигнала с помощью выбранного преобразования [5].

Масштабирующая функция имеет вид:

$$\phi(x) = \begin{cases} 1, 0 \leq x < 1 \\ 0, x \notin [0,1) \end{cases} \quad (2.16)$$

Ее интеграл:  $\int_{-\infty}^{\infty} \phi(x) dx = 1$ .

Масштабирующая функция определяет аппроксимацию сигнала (позволяет исследовать числовые характеристики и качественные свойства объекта, сводя задачу к изучению более простых или более удобных объектов).

Построение масштабирующей функции и функции вейвлета Хаара представлено на рисунок 2.45.

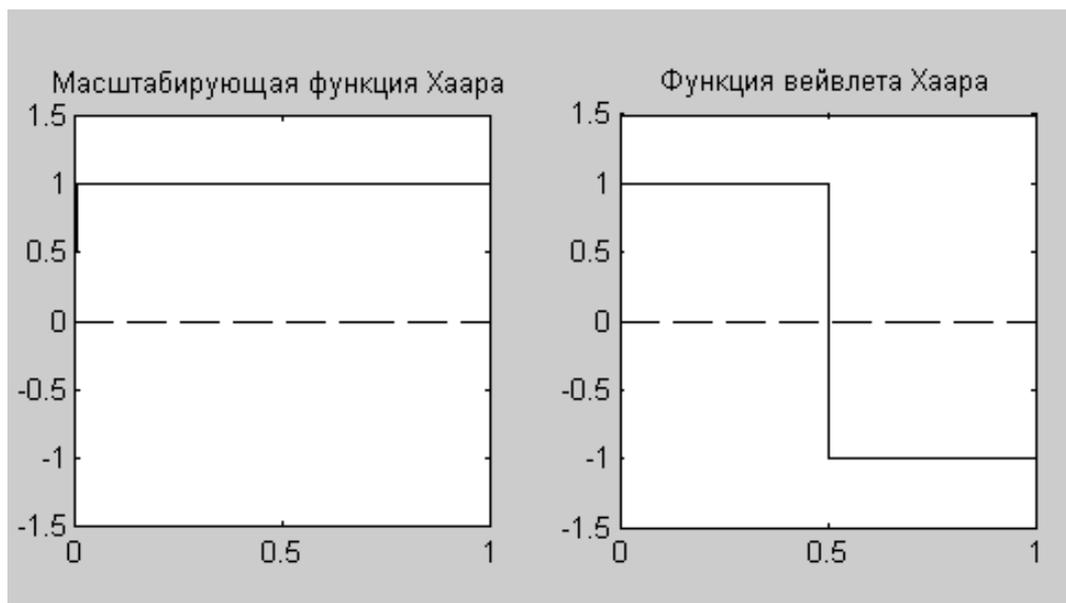


Рис. 2.45. Масштабирующая функция и функция вейвлета Хаара

Преобразование Хаара использует комбинацию этих функций. Чтобы понять, как выполняется преобразование, выполним сначала преобразование одномерного массива.

Пример. Выполним преобразование Хаара следующего массива:

$$A = (6, 7, 8, 9, 10, 6, 7, 8).$$

Вычислим сначала средние и полуразности элементов:

1 шаг:

$$\text{Средние: } (6+7)/2 = 13/2 = 6.5; (8+9)/2 = 8.5; (10+6)/2 = 8; (7+8)/2 = 7.5.$$

$$\text{Полуразности: } (6-7)/2 = -0.5; (8-9)/2 = -0.5; (10-6)/2 = 2; (7-8)/2 = -0.5$$

Таким образом, получим массив из восьми элементов:

$$A_1 = (6.5, 8.5, 8, 7.5, -0.5, -0.5, 2, 0.5)$$

Обратим внимание на последние четыре элемента – их значения гораздо меньше, чем до выполнения преобразования. Поэтому их оставим без изменений. Средние значения называются аппроксимирующими коэффициентами, в которых накапливается максимальная информация, а значения полуразностей позволяют осуществлять полное восстановление

исходного сигнала. На следующем шаге мы будем работать с первыми четырьмя элементами, а вторую половину массива оставим без изменений.

2 шаг:

Средние:  $(6.5+8.5)/2 = 7.5$ ;  $(8+7.5)/2 = 7.75$ .

Полуразности:  $(6.5-8.5)/2 = -1$ ;  $(8-7.5)/2 = 0.25$ .

Новый массив:  $A_2 = (7.5, 7.75, -1, 0.25, -0.5, -0.5, 2, -0.5)$

3 шаг:

Средние:  $(7.5+7.75)/2 = 7.625$ ;  $(-1+0.25) = 0.625$ .

Полуразности:  $(7.5-7.75)/2 = -0.125$ ;  $(-1-0.25)/2 = -0.625$ .

Новый массив  $A_3 = (7.625, 0.625, -0.125, -0.625, -0.5, -0.5, 2, -0.5)$

Из за постоянного вычисления полуразностей происходит постоянное уменьшение значений исходных элементов (пикселей изображения), поэтому после выполнения преобразования, можно к получившемуся массиву дополнительно применить один из алгоритмов сжатия (например, Хаффман или арифметическое кодирование). Если необходимо использовать сжатие с потерями, то здесь используется операция квантования или удаления наименьших значений полуразностей: например, может быть записано:  $A_3 = (7.625, 0.6250, 0, 0, 0, 2, 0)$ . Тогда исходные данные восстановятся с искажениями.



Рис. 2.46. Результат применения вейвлета Хаара

### **Математический аппарат вейвлета Добеши**

В настоящее время вейвлет Добеши 4-ого порядка является наиболее используемым. Функция данного вейвлета и его масштабирующая функция представлены на рисунке 2.47.

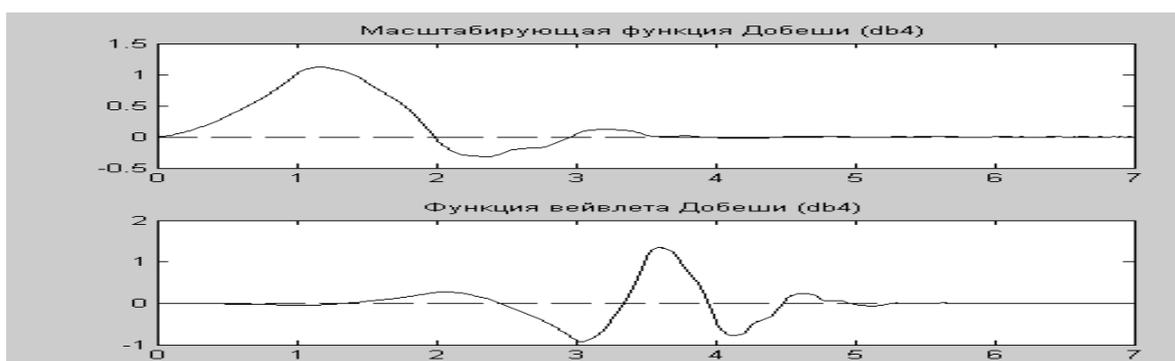


Рис. 2.47. Масштабирующая функция и функция вейвлета Добеши 4-го порядка

Масштабирующая функция и функция вейвлета Добеши задаются рекурсивно. Кроме того, функции удовлетворяют следующему требованию: масштабирующая функция имеет компактный носитель – отрезок  $[0, 3]$ , а также должна быть равна нулю вне этого отрезка. Эту функцию, как и вейвлет, можно задать рекурсивно, определив начальные условия.

Начальные значения:

$$\varphi(0) = 0, \varphi(1) = \frac{1 + \sqrt{3}}{2}, \varphi(2) = \frac{1 - \sqrt{3}}{2}, \varphi(3) = 0 \quad (2.17)$$

И рекурсивное соотношение:

$$\begin{aligned} \varphi(r) = & \frac{1 + \sqrt{3}}{4} \varphi(2r) + \frac{3 + \sqrt{3}}{4} \varphi(2r - 1) + \frac{3 - \sqrt{3}}{4} \varphi(2r - 2) + \\ & \frac{1 - \sqrt{3}}{4} \varphi(2r - 3) = \end{aligned} \quad (2.18)$$

$$\begin{aligned} h_0 \varphi(2r) + h_1 \varphi(2r - 1) + h_2 \varphi(2r - 2) + h_3 \varphi(2r - 3) = \\ (h_0, h_1, h_2, h_3) \cdot (\varphi(2r), \varphi(2r - 1), \varphi(2r - 2), \varphi(2r - 3)). \end{aligned}$$

Сумма начальных значений равна единице.

$$\varphi(0) + \varphi(1) + \varphi(2) + \varphi(3) = 0 + \frac{1 + \sqrt{3}}{2} + \frac{1 - \sqrt{3}}{2} + 0 = 1 \quad (2.19)$$

Далее функция вычисляется по шагам. Для этого на шаге 1 применяются условие компактного носителя, начальные значения и рекурсивные соотношения. На последующих шагах также применяется рекурсия. Для построения вейвлета также используется рекурсивное соотношение.

$$\psi(r) = -\frac{1 + \sqrt{3}}{4} \varphi(2r - 1) + \frac{3 + \sqrt{3}}{4} \varphi(2r) - \frac{3 - \sqrt{3}}{4} \varphi(2r + 1) + \frac{1 - \sqrt{3}}{4} \varphi(2r + 2) \quad (2.20)$$

Использование рекурсивных соотношений при построении вейвлетов и масштабирующих функций способствует тому, что построенные кривые являются

самоподобными, то есть обладают фрактальными свойствами. Это делает возможным применение вейвлетов к анализу фрактальных последовательностей.

Фильтры  $\{h_n\}_{n \in \mathbb{Z}}$  и  $\{g_n\}_{n \in \mathbb{Z}}$  вейвлетов  $\phi(x)$  и  $\psi(x)$  есть коэффициенты масштабирующих уравнений  $\phi(x) = \sqrt{2} \sum_n h_n \phi(2x - n)$  и  $\psi(x) = \sqrt{2} \sum_n g_n \psi(2x - n)$ , где  $g_n = (-1)^n \overline{h_{N-n}}$  (черта обозначает комплексное сопряжение). Фильтр  $h$  называется низкочастотным, а фильтр  $g$  – высокочастотным. Для восстановления используются фильтры  $\{h_n\}$  и  $\{g_n\}$  вейвлетов  $\phi$  и  $\psi$ , а для восстановления с помощью фильтров – сопряженные (транспонированные) фильтры.

### Математический аппарат непрерывного вейвлет-преобразования

Запись непрерывного вейвлет-преобразования (НВП) выглядит следующим образом:

$$Wf(a, b) = \int_{-\infty}^{\infty} f(t) \frac{1}{\sqrt{s}} \psi^* \left( \frac{t-b}{a} \right) dt \quad (2.21)$$

Областью определения вейвлет-преобразования  $Wf$  является множество:

$$\mathbb{R}^2_- = \{(a, b) \mid a \in \mathbb{R}^*, b \in \mathbb{R}\} \quad (2.22)$$

Параметр  $a$  определен в комплексной плоскости, параметр  $b$  – в области действительных чисел. В теории вейвлетов ось  $a$  масштабируется вертикально, а ось  $b$  – горизонтально. Часто область определения  $Wf$  ограничивается положительными значениями параметра  $a$ .

Главной задачей является нахождение вейвлет-коэффициентов. Использование свойств вейвлет-преобразования позволяет осуществить анализ сигналов. Например, слишком большие амплитуды значений вейвлет-коэффициентов указывают на положение перепадов, которые имеют резкие изменения интенсивности изображения. Различные масштабы описывают контуры структуры изображения меняющихся размеров. Такое выделение перепадов эффективно для распознавания образов при компьютерной визуализации [4].

Представляется необходимым продемонстрировать действие НВП на примере вейвлета Хаара.

Вейвлет Хаара может быть записан как:

$$\psi \left( \frac{t-b}{a} \right) = \begin{cases} 1, & \left( b \leq t \leq b + \frac{a}{2} \right) \\ -1, & \left( b + \frac{a}{2} \leq t < b + a \right) \\ 0 & \end{cases} \quad (2.23)$$

Вейвлет-преобразование определяется как:

$$Wf(a,b) = \frac{1}{\sqrt{a}} \left( \int_b^{b+a/2} f(t) dt - \int_{b+a/2}^{b+a} f(t) dt \right) = \frac{\sqrt{a}}{2} \left( \frac{2}{a} \int_b^{b+a/2} f(t) dt - \frac{2}{a} \int_{b+a/2}^{b+a} f(t) dt \right). \quad (2.24)$$

Это означает, что за исключением нормирующего множителя значение  $Wf(a,b)$  представляет собой разность между двумя средними значениями функции  $f$ , эти средние значения берутся по двум соседним интервалам длины  $a/2$  в окрестности точки  $b$ .

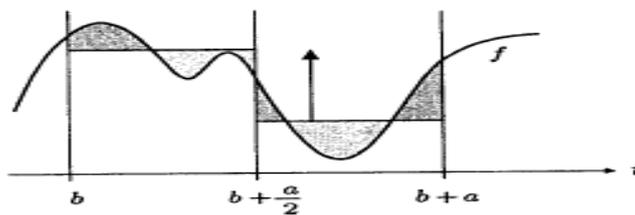


Рис. 2.48. Результат НВП в случае вейвлета Хаара

Величину  $Wf(a,b)$  можно рассматривать с другой точки зрения:

$$Wf(a,b) = \frac{1}{\sqrt{a}} \int_a^{b+a/2} \left( f(t) - f\left(t + \frac{a}{2}\right) \right) dt = -\frac{1}{\sqrt{a}} \int_b^{b+a/2} \left( \int_t^{t+a/2} f'(r) dr \right) dt = \dots = -\frac{1}{\sqrt{a}} \int_{a/2}^{a/2} \left( \frac{a}{2} - |r| \right) f'\left(b + \frac{a}{2} + r\right) dr \quad (2.25)$$

В этой форме величина  $Wf(a,b)$  проявляется как взвешенное среднее производной  $f'$  на интервале  $[b, b+a]$ .

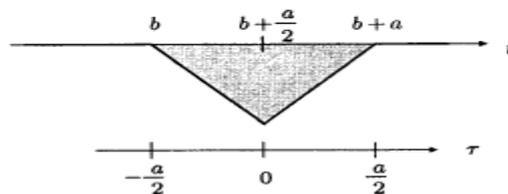


Рис. 2.49. Другая интерпретация величины  $Wf$

Подставляя вместо  $f(t)$  конкретную функцию, можно вычислить ее НВП.

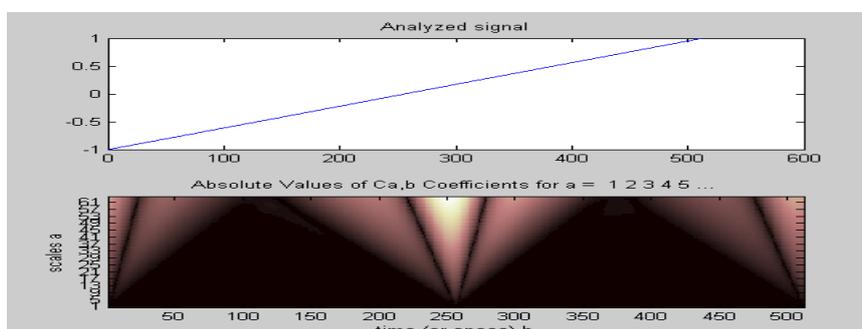


Рис. 2.50. Результат НПВ сигнала

По полученной спектрограмме можно судить о характере сигнала, например, наблюдать максимумы «всплесков энергии» в различные промежутки времени (показатель  $b$ ) и при различных масштабах (показатель  $a$ ).

### Математический аппарат дискретного вейвлет-преобразования

Для обработки и просмотра сигналов и изображений при различных разрешениях также используется дискретное вейвлет-преобразование.

В общем виде прямое дискретное преобразование изображения  $f(x,y)$  размерности  $M \times N$  можно выразить в общем виде следующим образом:

$$T(u, v, \dots) = \sum_{x, y} f(x, y) g_{u, v, \dots}(x, y) \quad (2.26)$$

Здесь  $x$  и  $y$  – пространственные переменные (координаты пикселей изображения), а  $u, v$  – переменные в частотной области, характеризующие закономерность распределения пикселей изображения. Зная  $T(u, v, \dots)$ , можно восстановить функцию  $f(x, y)$  с помощью обратного преобразования:

$$f(x, y) = \sum_{u, v, \dots} T(u, v, \dots) h_{u, v, \dots}(x, y). \quad (2.27)$$

Члены последовательностей  $g$  и  $h$  называются прямыми и обратными ядрами преобразования, определяющими природу, сложность и эффективность пары преобразований. Дискретное вейвлет-преобразование, в отличие от преобразования Фурье, обозначает целый класс преобразований, которые различаются не только своими ядрами (коэффициентами), но и свойствами этих коэффициентов (*с каким классом вейвлетов мы будем работать*) и способом применения этих коэффициентов (*сколько требуется вычислить различных решений*).

Можно охарактеризовать каждое ДВП с помощью ядра преобразования или, основываясь на множестве параметров, которые однозначно определяют пару ядер. Все преобразования являются родственными (их функции разложения представляют собой «маленькие волны» или «вейвлеты»), которые имеют переменную частоту колебаний и ограниченную длительность.

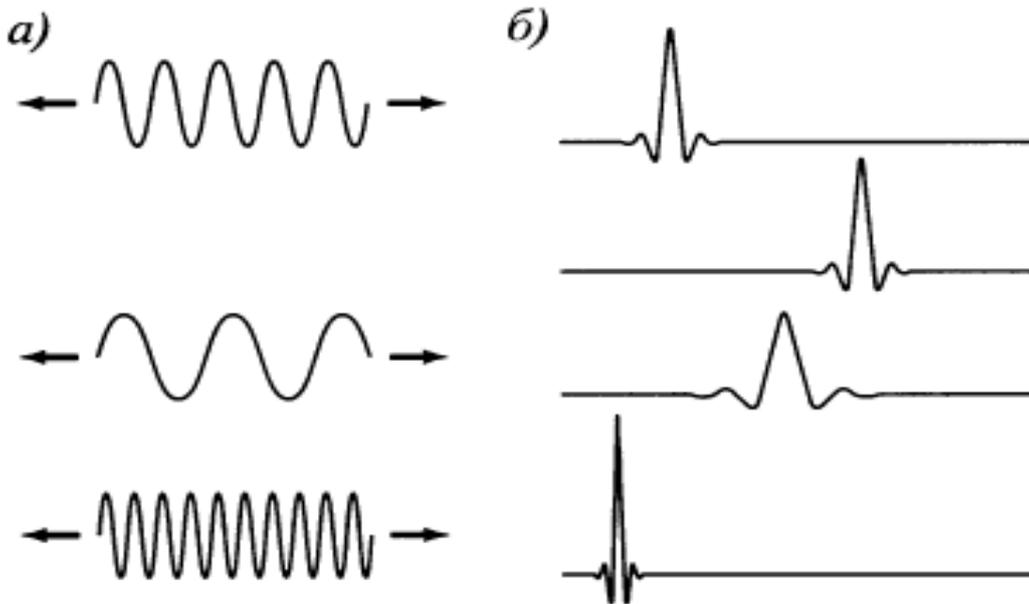


Рис. 2.51. а) Множество функций разложения Фурье – синусоиды с переменной частотой и бесконечной длительностью б) Функции разложения ДВП – «маленькие волны» с переменной частотой и конечной длительностью

Свойства «ядер», являющиеся общими для всех типов вейвлетов.

Свойство 1. *Разделимость, масштабируемость и переносимость.* Ядра можно представить в виде трех разделимых двумерных вейвлетов:

$$\begin{aligned}\psi^H(x, y) &= \psi(x)\varphi(y), \\ \psi^V(x, y) &= \varphi(x)\psi(y), \\ \psi^D(x, y) &= \psi(x)\psi(y).\end{aligned}\tag{2.28}$$

Здесь  $\psi^H(x, y), \psi^V(x, y), \psi^D(x, y)$  называются горизонтальными, вертикальными и диагональными вейвлетами, а двумерная функция:

$$\varphi(x, y) = \varphi(x)\varphi(y)\tag{2.29}$$

Называется масштабирующей функцией. Каждая из этих двумерных функций является произведением двух одномерных масштабирующих функций [5].

$$\begin{aligned}\varphi_{j,k}(x) &= 2^{j/2}\varphi(2^j x - k), \\ \psi_{j,k}(x) &= 2^{j/2}\psi(2^j x - k).\end{aligned}\tag{2.30}$$

Параметры  $i$  и  $j$  являются целыми числами. Число  $k$  определяет положение одномерных функций на оси  $x$ , масштаб  $j$  – ширину по оси  $x$ , множитель  $2^{j/2}$  отвечает за высоту и амплитуду.

Свойство 2. *Кратномасштабная совместимость.* Свойство оговаривает требования к выбранной масштабирующей функции.

а) Каждая функция  $\varphi_{i,j}(x)$  ортогональная любому своему целочисленному сдвигу.

б) Множество функций, которые можно представить в виде рядов разложения по  $\varphi_{j,k}(x)$  при малых масштабах ( $j$  мало), принадлежит множеству функций, представимых в более высоких значениях.

в) Каждую масштабирующую функцию можно представить с произвольной точностью при  $j \rightarrow \infty$ .

Свойство 3. Ортогональность. Функции разложения  $\{\varphi_{i,j}(x)\}$  образуют ортогональный или биортогональный базис (отсюда и название еще одного типа вейвлетов – ортогональные и биортонональные) в пространстве одномерных измеримых, суммируемых в квадрате функций. Базис обладает тем свойством, что каждая представимая в этом базисе функция имеет единственный набор коэффициентов. Для вещественных ортогональных ядер имеет место равенство  $g = h$ . В биортогональном случае

$$\langle h_r, g_s \rangle = \delta_{rs} = \begin{cases} 1, & r = s. \\ 0 & \end{cases} \quad (2.31)$$

Для практического применения важно следствие из приведенных свойств – функцию вейвлета и масштабирующую функцию можно представить *в виде линейной комбинации своих же копий с удвоенным разрешением*:

$$\varphi(x) = \sum_n h_\varphi(n) \sqrt{2} \varphi(2x - n), \quad \psi(x) = \sum_n h_\psi(n) \sqrt{2} \psi(2x - n) \quad (2.32)$$

В среде MATLAB дискретное вейвлет-преобразование реализуется с помощью процедур `dwt` и `fwt` (быстрое вейвлет-преобразование). Можно выполнять разложение как двумерных, так и одномерных сигналов.



Рис. 2.52. Результат дискретного вейвлет-преобразования

### Алгоритм быстрого вейвлет-преобразования

Обобщая преобразование на двумерный и многомерный уровень, можно представить схему преобразования (быстрого вейвлет-преобразования).

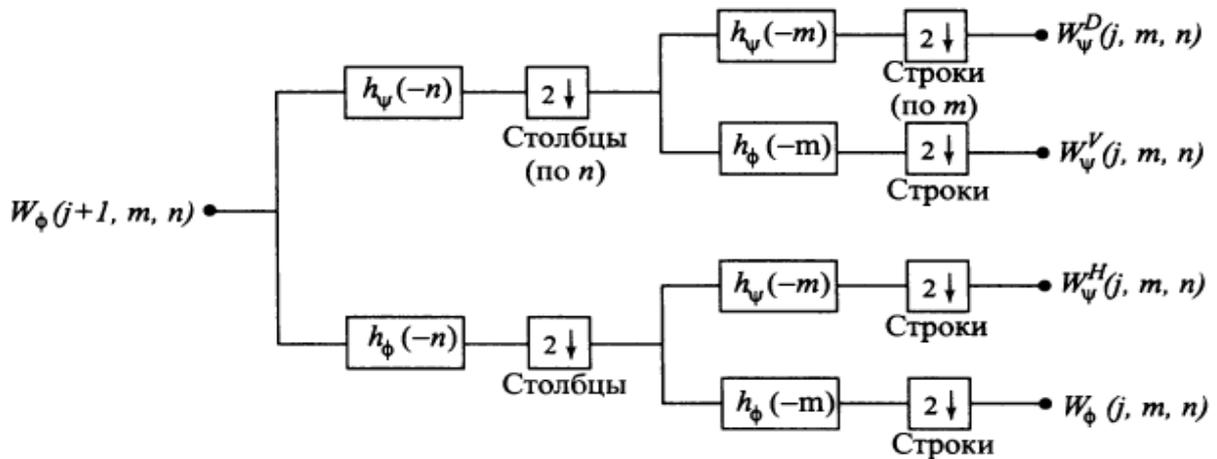


Рис. 2.53. Схема быстрого вейвлет-преобразования

Сначала обрабатываются столбцы, а затем строки. Используются при этом фильтры низких и высоких частот.

Каждый проход через блок фильтров на рис. 10 разлагает входные данные на четыре компоненты меньшего диапазона (или масштаба). Коэффициенты  $W_\varphi$  преобразования получаются двумя проходами низкочастотной фильтрации (т. е. с фильтром  $h_\varphi$ ), и поэтому они называются коэффициентами приближения; коэффициенты  $W_\varphi^i, i = H, V, D$  называются, соответственно, коэффициентами горизонтальных, вертикальных и диагональных деталей. Поскольку само изображение  $f(x, y)$  представлено в наивысшем разрешении, то оно становится входом  $W_\varphi(j+1, m, n)$  первой итерации процедуры. Кроме того, здесь имеется три переменные преобразованного пространства: масштаб  $j$ , горизонтальная трансляция  $p$  и вертикальная трансляция  $t$ . Эти переменные соответствуют обозначениям  $u, v, \dots$  в первых двух уравнениях (8), (9). Далее посмотрим примеры применения двумерного вейвлет-преобразования (dwt2, fwt) при обработке изображений.

При выполнении этапов преобразования используется операция свертки сигналов. Слово «свертка» обозначает совместное сворачивание двух величин или функций [2]. Дискретная свертка векторов  $f$  и  $g$  обозначается  $f * g$  и вычисляется с помощью соотношения:

$$(f * g)_i = \sum_j f_j g_{i-j} \quad (2.33)$$

В результате выполнения операции получается один вектор, охватывающий диапазон входного сигнала. Поэтому вейвлет-преобразования называют *поддиапазонными*. Дискретная свертка двух числовых последовательностей  $f(i)$  и  $g(i)$  задается неравенством:

$$h(i) = f(i) * g(i) = \sum_j f(i)g(i - j)$$

Свертка сигналов применяется, например, для удаления шума или сглаживания сигналов.

Использование приложения Wavelet Toolbox Matlab для сжатия информации путем вейвлет-преобразований [12].

Пакет MATLAB Wavelet Toolbox представляет собой совокупность программ, позволяющих выполнять вейвлет-анализ и обработку сигналов средствами графического интерфейса пользователя. Пакет позволяет продемонстрировать практические приложения теории вейвлетов и вейвлет-преобразований. Остановимся на следующих пунктах:

1. Просмотр вейвлетов.
2. Одномерный дискретный вейвлет-анализ.
  - 2.1. Вейвлет-разложение сигнала.
  - 2.2. Интерпретация статистических характеристик сигнала.
  - 2.3. Гистограммы коэффициентов аппроксимации и детализации.
3. Сжатие одномерных сигналов.
4. Удаление шума из сигналов.

Главное меню пакета вызывается из MATLAB с помощью команды `wavemenu`.

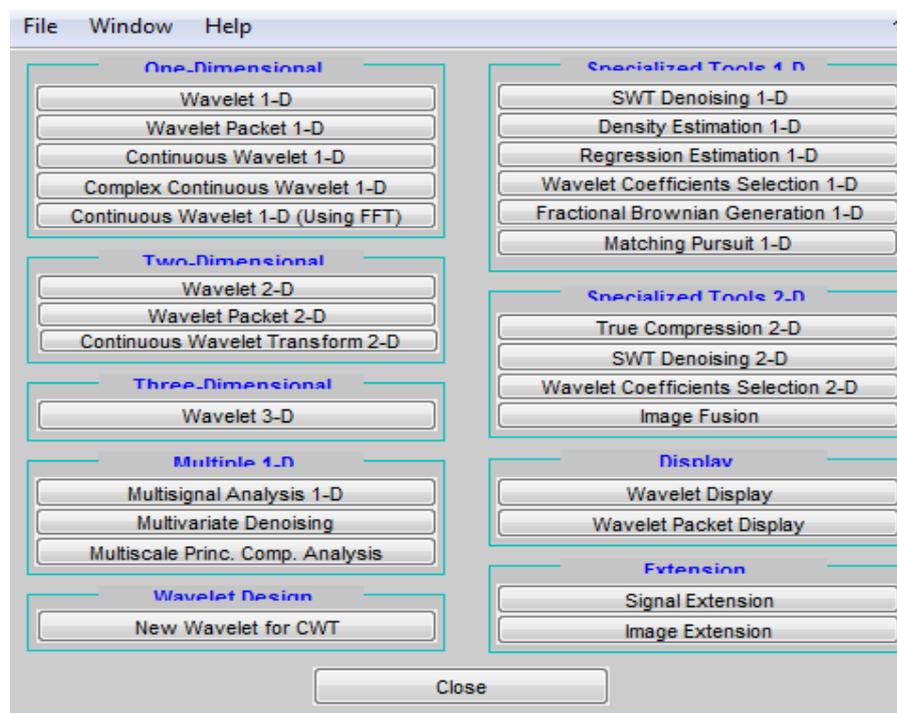


Рис. 2.54. Главное окно Wavelet Toolbox

### Способы просмотра вейвлетов в Wavelet Toolbox

Способы просмотра вейвлетов – группа Display (кнопка Wavelet Display). Откроется следующее окно:

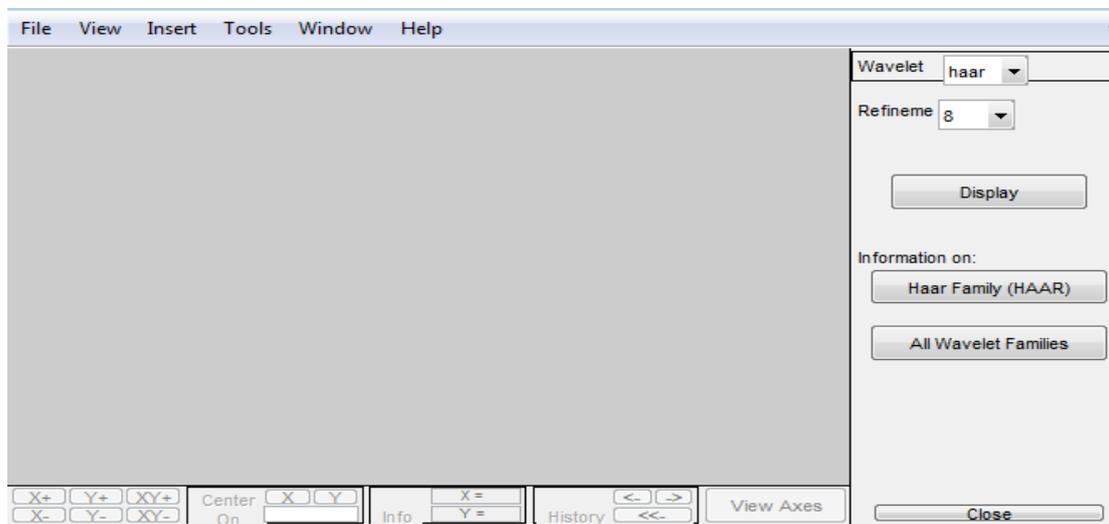


Рис. 2.55. Окно раздела Wavelet Display после запуска

Вверху расположено обычное меню, справа и внизу – меню, позволяющие управлять отображением нужной информации.

В правом меню можно из раскрывающегося списка тип вейвлета, задать уровень разрешения (параметр *Refinement*)

– определить шаг вычисления значений вейвлетов (этот параметр равен  $1/2^N$ ,  $N$  по умолчанию равно 8).

Также можно посмотреть информацию по выбранному типу вейвлета.

Выбрав из списка тип вейвлета 'haar', следует нажать кнопку *Display* ( $Refinement = 8$ ).

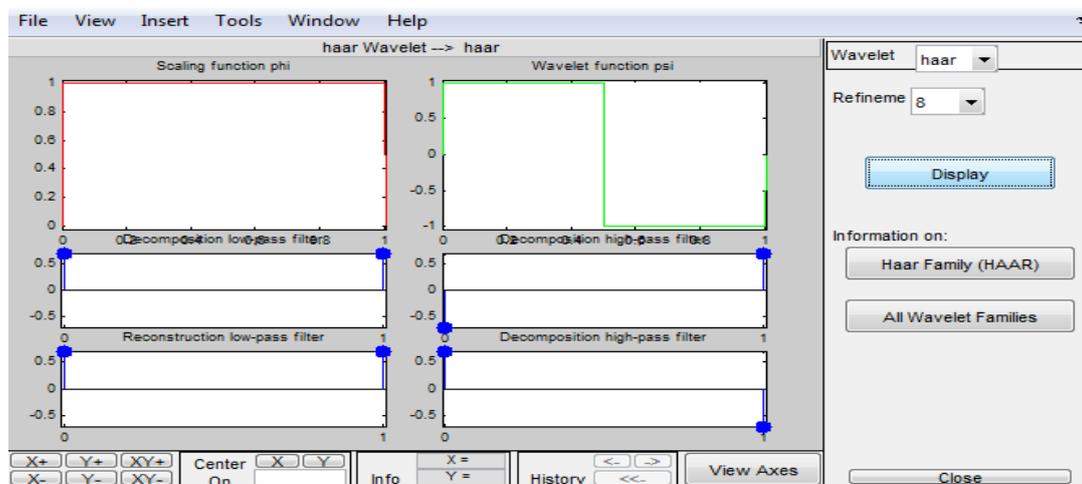


Рис. 2.56. Вейвлет и масштабирующая функция Хаара, фильтры восстановления и разложения

Вверху расположены графики масштабирующей функции и функции вейвлета. Ниже изображены фильтры восстановления и разложения вейвлета (низкочастотные и высокочастотные). Для восстановления используются фильтры (коэффициенты масштабирующих уравнений)  $h$  и  $g$  функций  $\varphi(x)$  и  $\psi(x)$ , а для вейвлет-разложения – сопряженные (транспонированные) фильтры. Это хорошо видно на графиках фильтров. Для вейвлета Хаара фильтры простые, так как базовыми операциями являются нахождение средних и полуразностей. У других вейвлетов фильтры более сложные по своей структуре.

Например, если выбрать вейвлет Добеши 7-ого порядка, то получится следующее изображение:

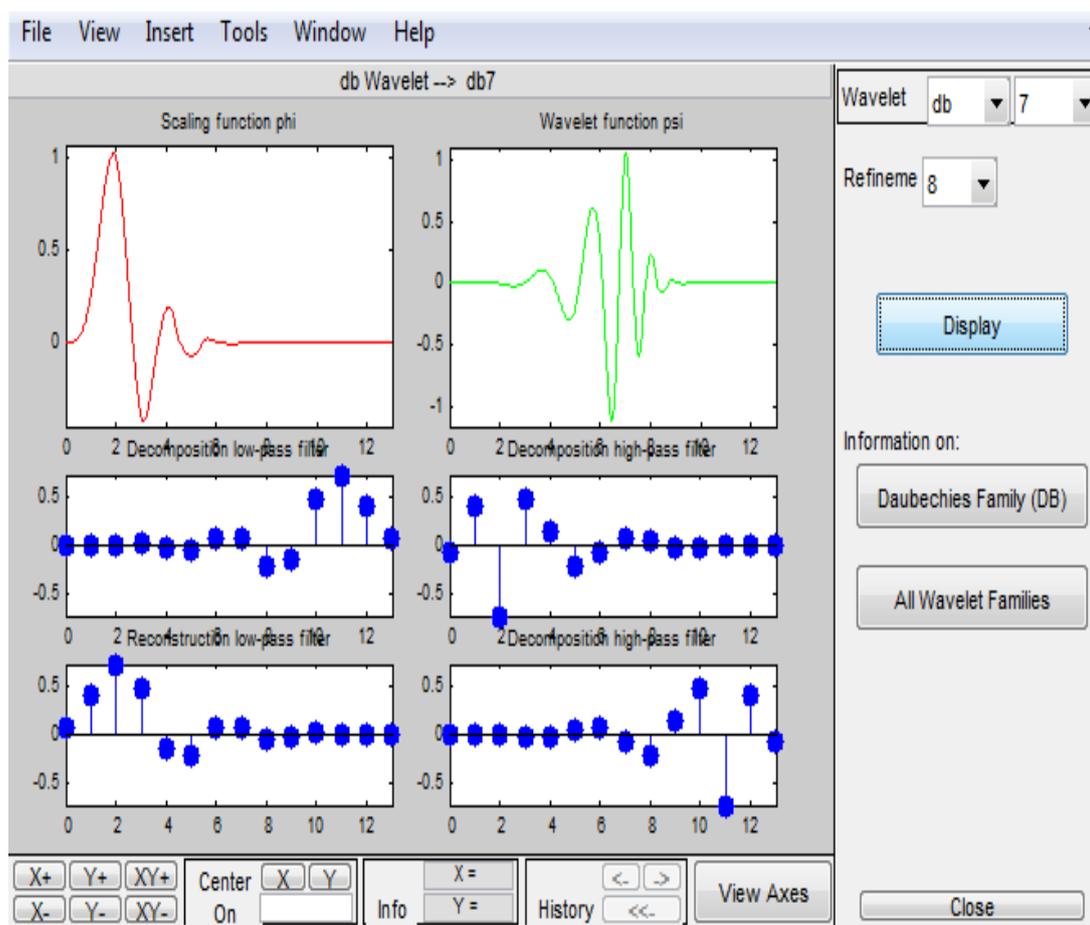


Рис. 2.57. Вейвлет 7-го порядка и масштабирующая функция Добеши, фильтры восстановления и разложения

### Сжатие сигналов при помощи вейвлет-преобразований в Wavelet Toolbox

В главном окне необходимо нажать кнопку Wavelet 1-D, появится диалоговое окно, интерфейс которого позволяет осуществлять обработку одномерных сигналов.

Загрузка сигнала Frequency Breakdown (Load → Example Analysis → Basic Signals → Frequency Breakdown):

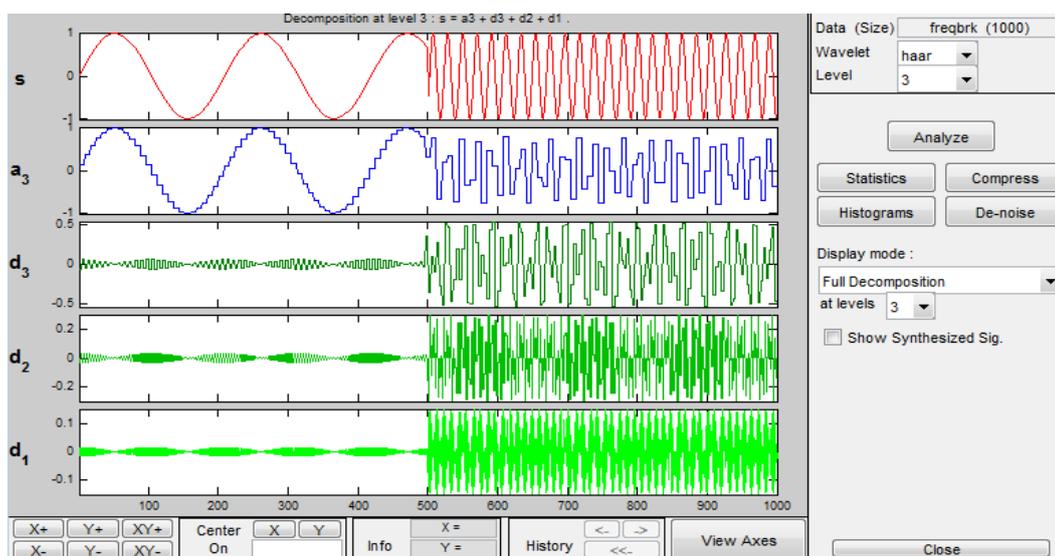
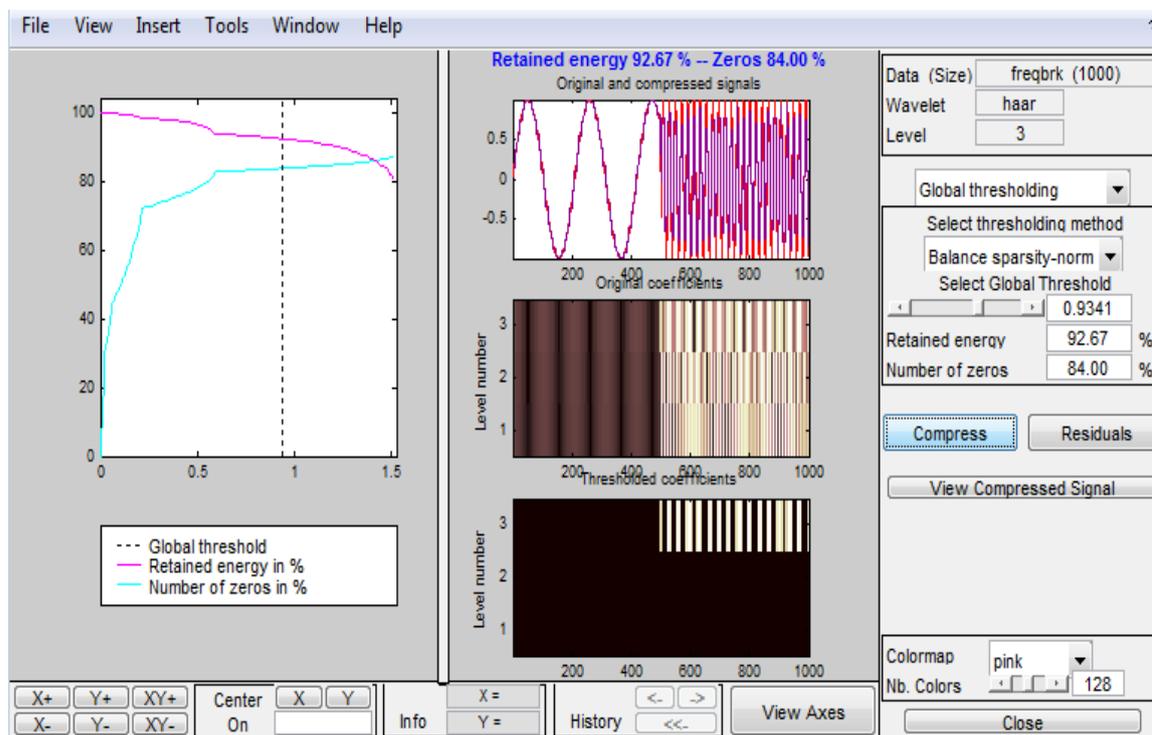


Рис. 2.58. Разложение сигнала

Кнопка **compress** позволяет осуществить сжатие одномерного сигнала.

Рис. 2.59. Интерфейс окна, вызываемого нажатием кнопки **Compress**

При сжатии одномерных сигналов используется метод глобальной пороговой обработки детализирующих коэффициентов. Управлять сжатием можно группой компонентов справа, устанавливая ползунком значение. Задав порог (процент сохраняемых коэффициентов), в результате получается сжатый сигнал.

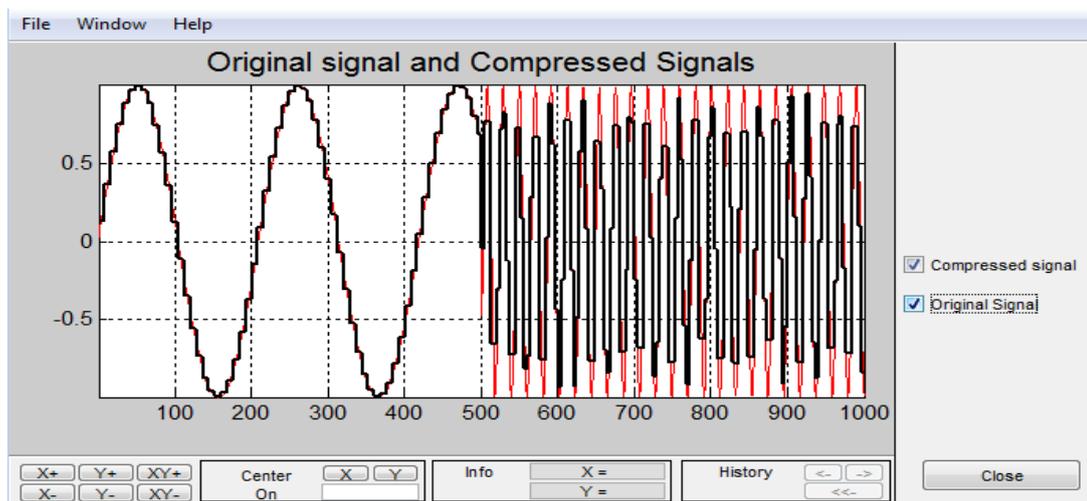


Рис. 2.60. Исходный и сжатый сигнал при помощи вейвлет-преобразования Хаара третьего уровня

### Сжатие одномерного сигнала при помощи различных вейвлет-преобразований

На сегодняшний день существует большое количество математических алгоритмов вейвлет-преобразований, данный раздел содержит рассмотрение трех основных и сравнительный анализ их характеристик.

Вейвлет Хаара пятого уровня. Разложение исследуемого сигнала при помощи вейвлета Хаара пятого уровня представлено на рисунке 2.61

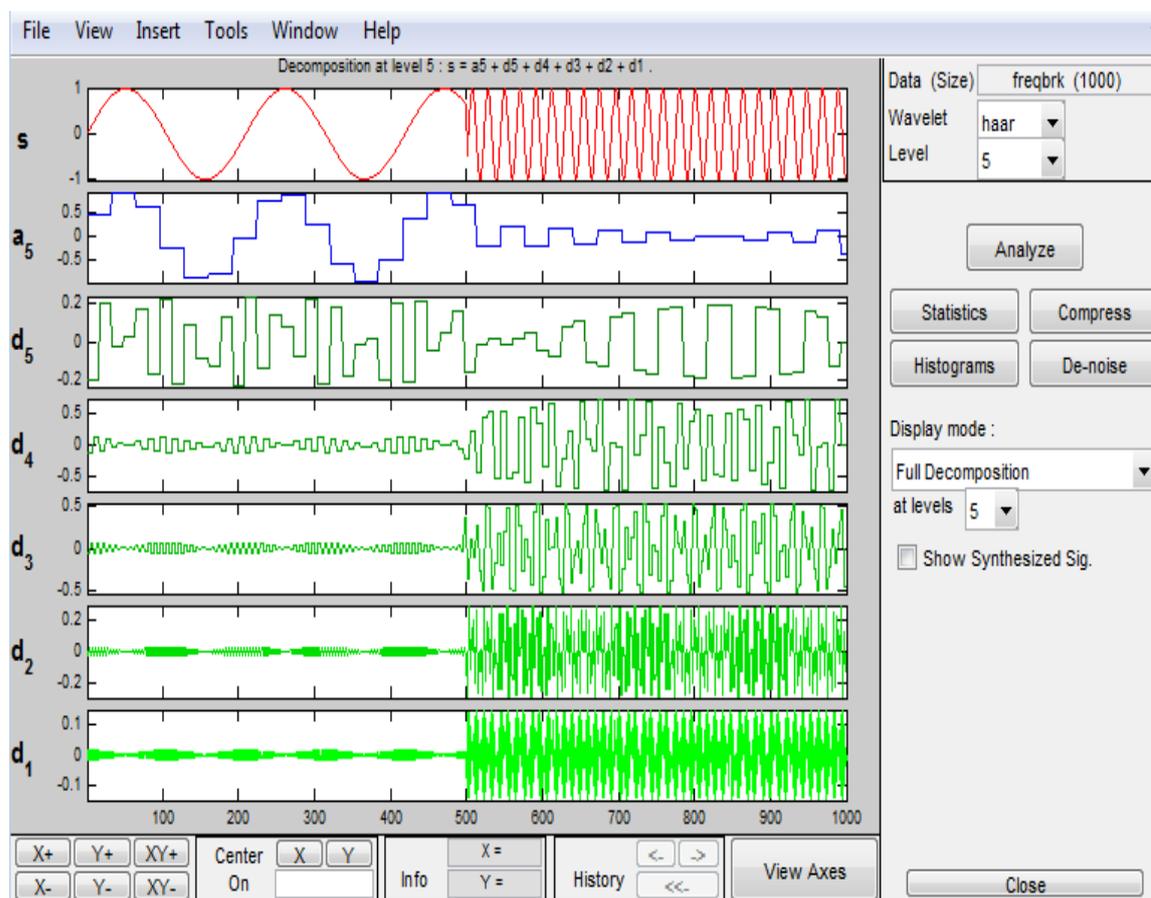


Рис. 2.61. Разложение сигнала при помощи вейвлета Хаара пятого уровня

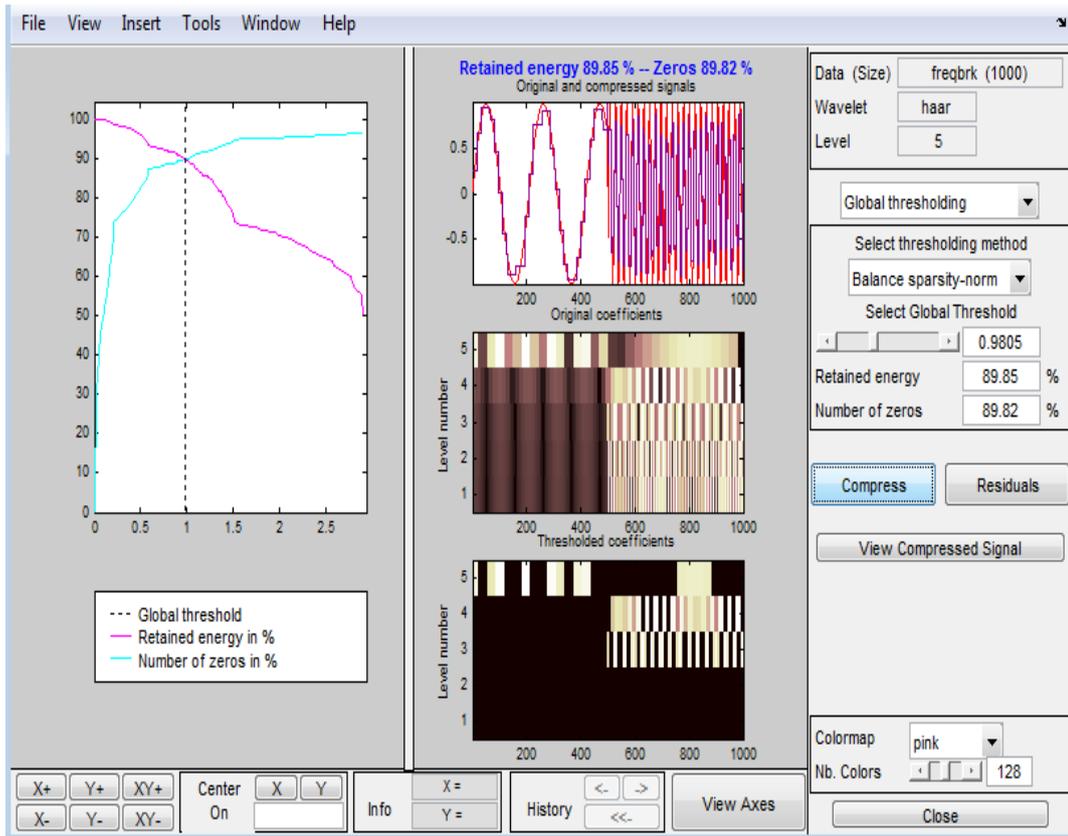


Рис. 2.62. Сжатие сигнала при помощи вейвлета Хаара пятого уровня

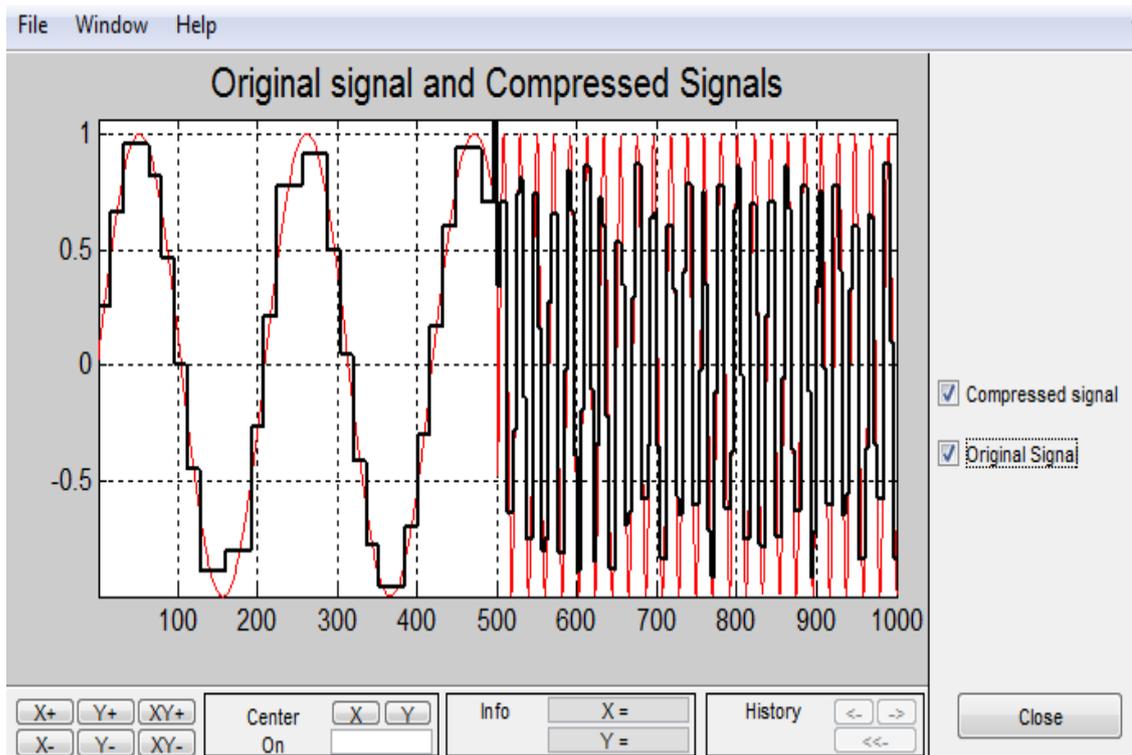


Рис. 2.63. Исходный и сжатый сигнал при помощи вейвлет-преобразования Хаара пятого уровня при коэффициенте сжатия 0.98

Далее необходимо проанализировать зависимость количества сохраненной энергии сигнала от коэффициента сжатия, результаты представлены в таблице 2.7.

Таблица 2.7. Зависимость количества сохраненной энергии от коэффициента сжатия для вейвлет-преобразования Хаара пятого порядка

$K_{сж}$	0,5	1	1,5	2	2,5
$Q_{сохр}$	95,93	89,47	75,37	70,36	65,32

Данную зависимость удобно представить в виде графика (рисунок 2.64):

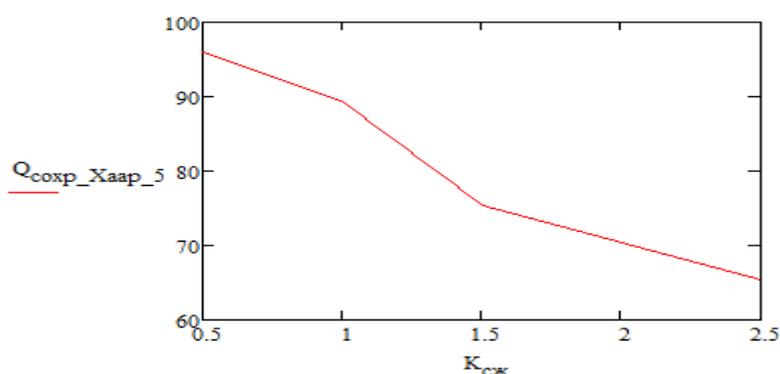


Рис. 2.64. Зависимость количества сохраненной энергии от коэффициента сжатия для вейвлет-преобразования Хаара пятого уровня

Вейвлет Добеши пятого уровня и пятого порядка. Разложение исследуемого сигнала при помощи вейвлета Добеши пятого уровня и пятого порядка представлено на рисунке 2.65.

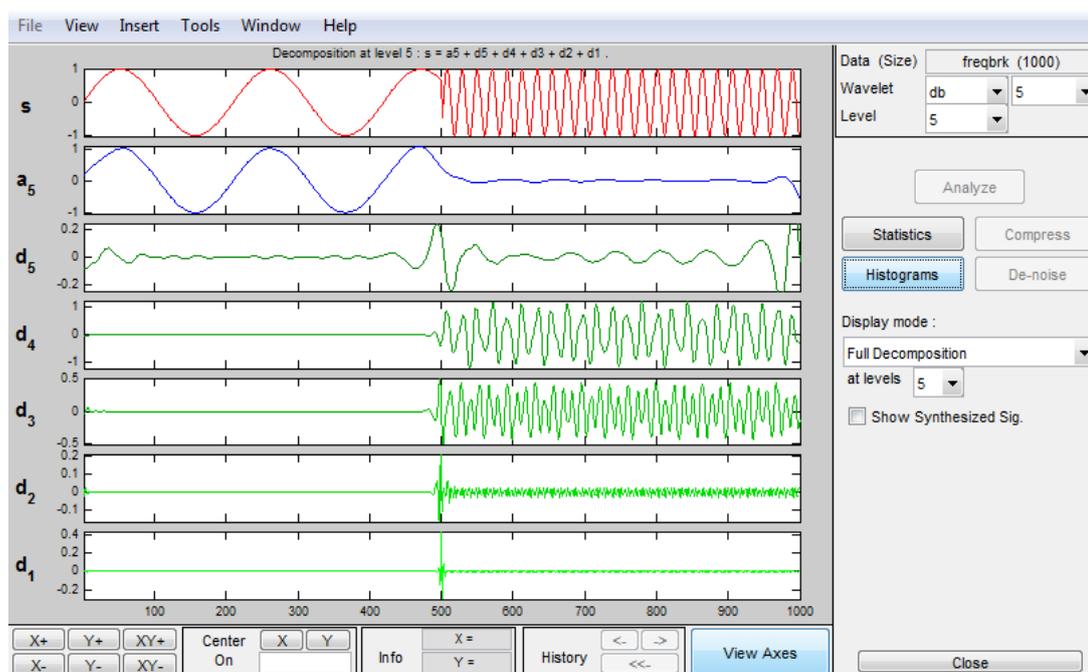


Рис. 2.65. Разложение сигнала при помощи вейвлета Добеши пятого уровня и пятого порядка

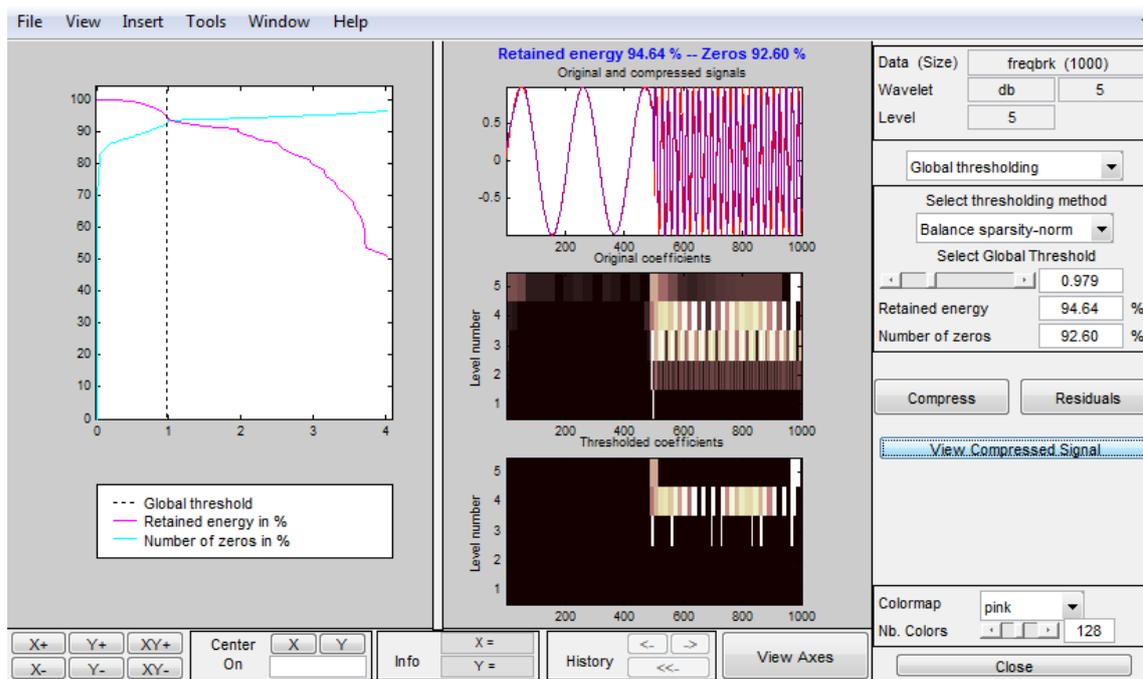


Рис. 2.66. Сжатие сигнала при помощи вейвлета Добеши пятого уровня и пятого порядка

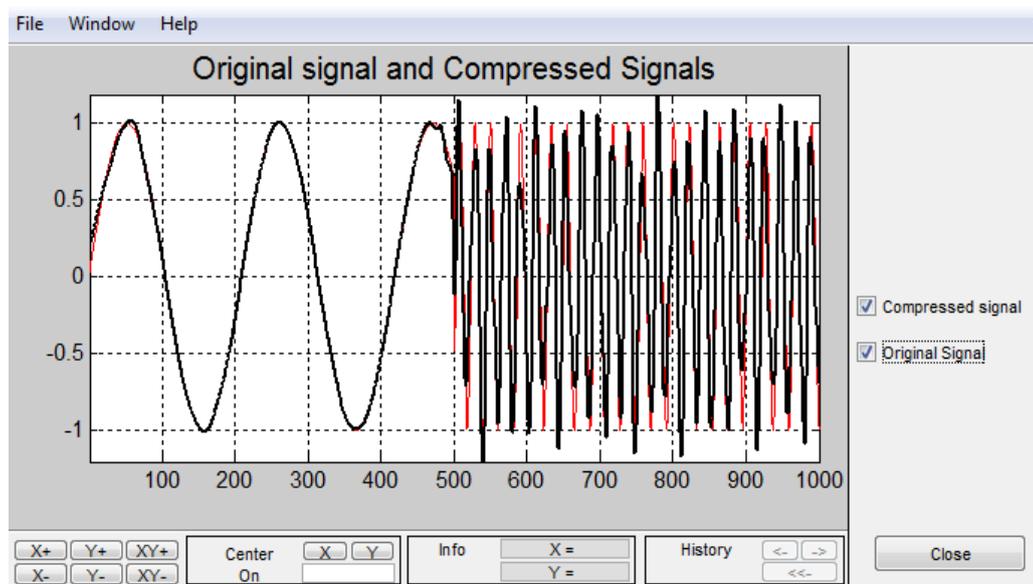


Рис. 2.67. Исходный и сжатый сигнал при помощи вейвлет-преобразования Добеши пятого уровня и пятого порядка при коэффициенте сжатия 0.98

Далее необходимо проанализировать зависимость количества сохраненной энергии сигнала от коэффициента сжатия, результаты представлены в таблице 2.7

Таблица 2.7 – Зависимость количества сохраненной энергии от коэффициента сжатия для вейвлет-преобразования Добеши пятого порядка

$K_{сж}$	0,5	1	1,5	2	2,5
$Q_{сохр}$	99,35	93,61	91,81	89,95	86,41

Данную зависимость удобно представить в виде графика ( рисунок 2.68)

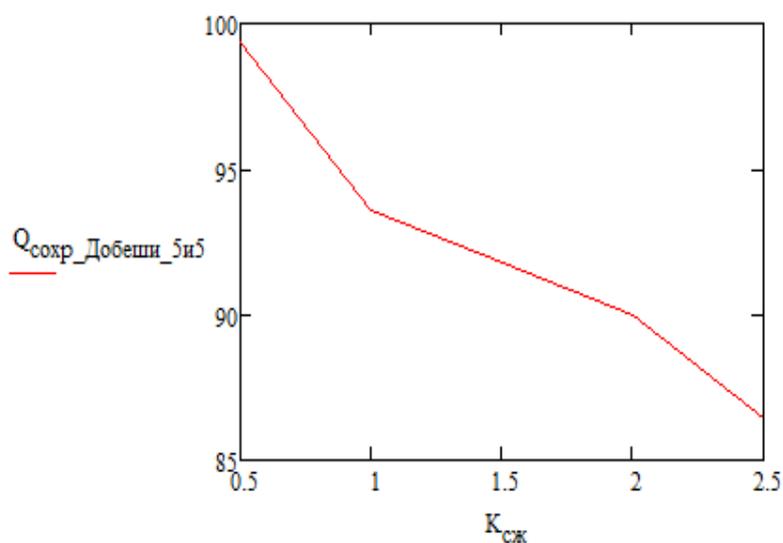


Рис. 2.68. Зависимость количества сохраненной энергии от коэффициента сжатия для вейвлет-преобразования Добеши пятого уровня и пятого порядка

Биортогональное преобразование пятого уровня и порядка 5,5. Разложение исследуемого сигнала при помощи биортогонального преобразования пятого уровня и порядка 5,5 представлено на рисунке 2.70.

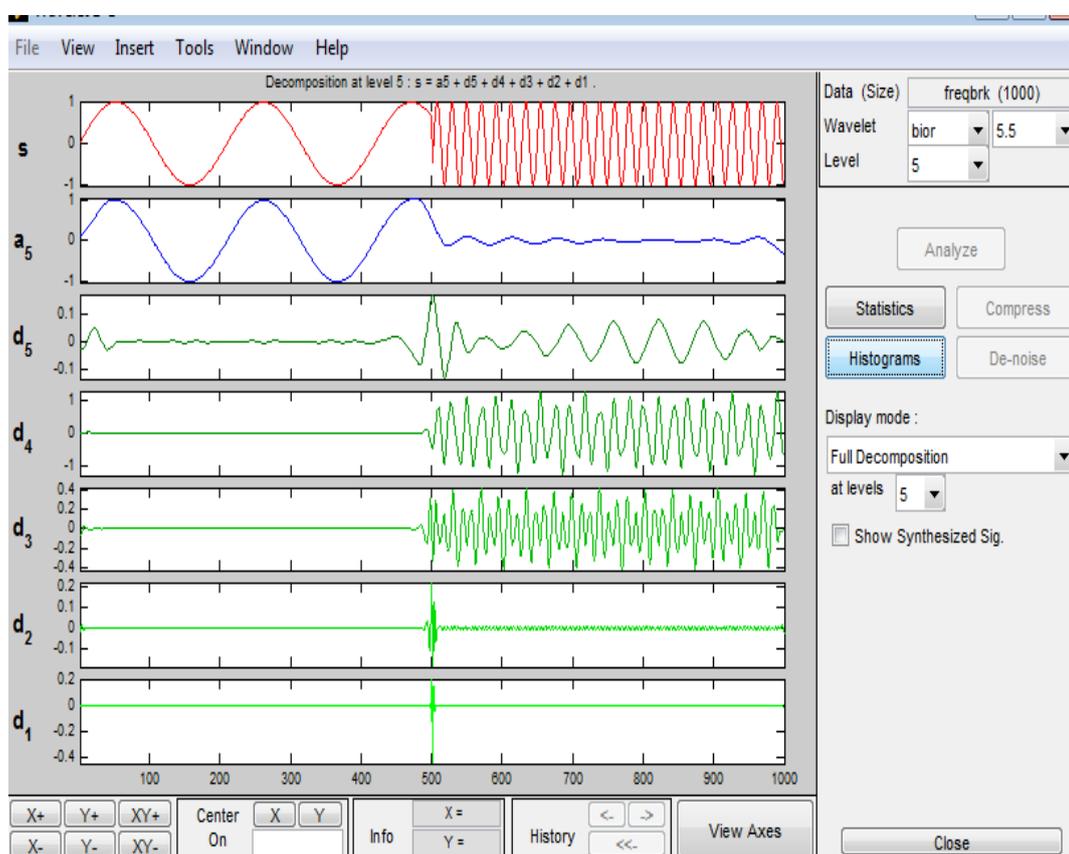


Рис. 2.70. Разложение сигнала при помощи биортогонального преобразования пятого уровня и порядка 5,5

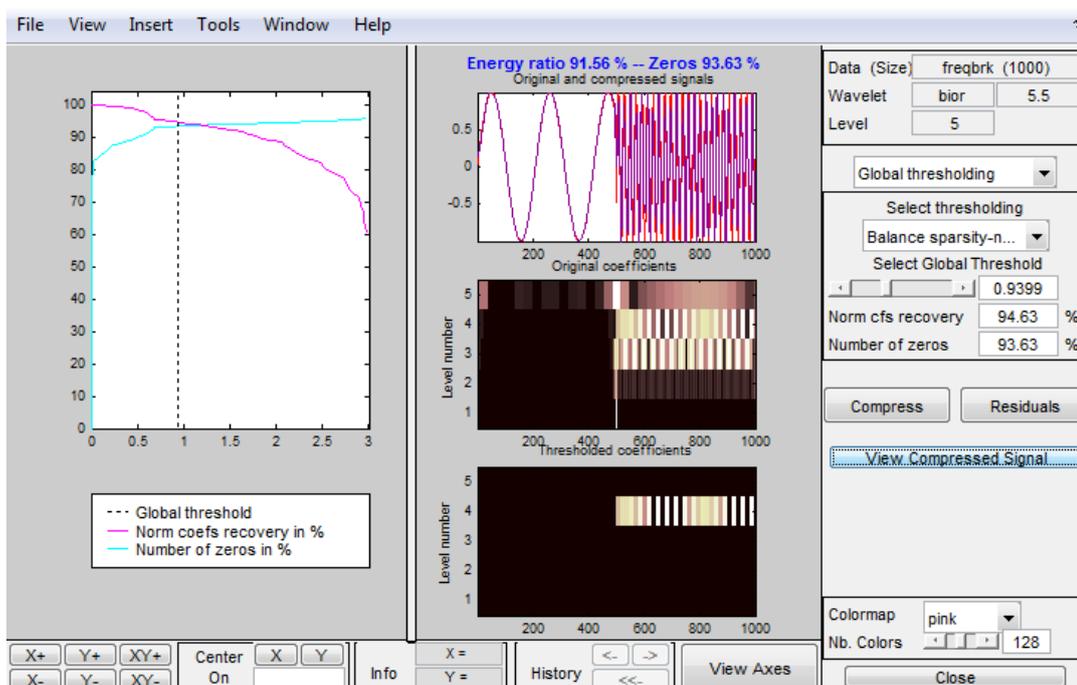


Рис. 2.71. Сжатие сигнала при биортогональном вейвлет-преобразовании пятого уровня и порядка 5,5

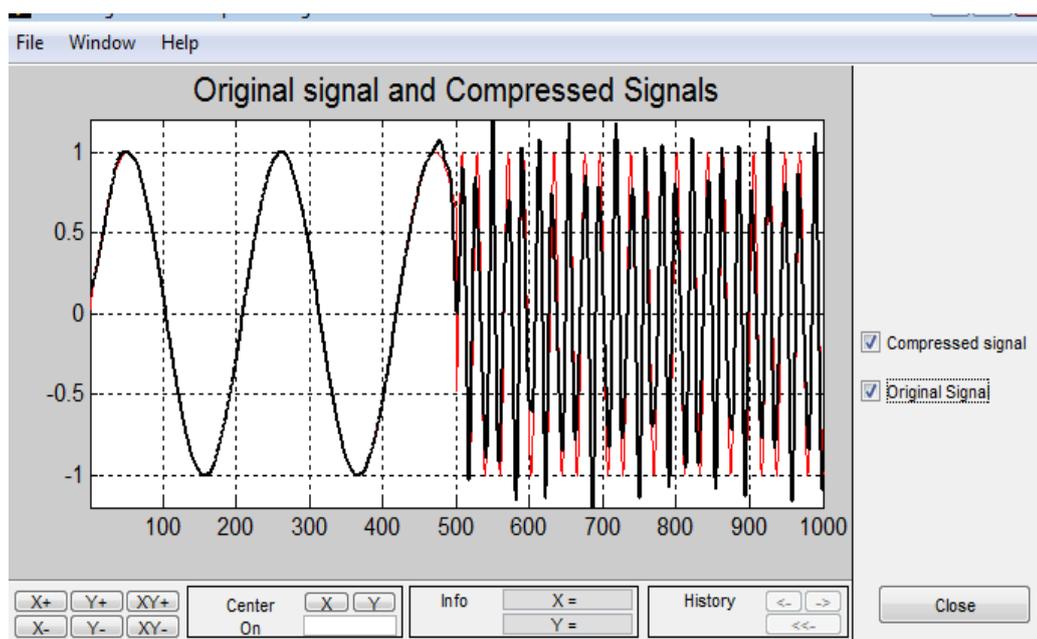


Рис. 2.72. Исходный и сжатый сигнал при помощи биортогонального вейвлет-преобразования пятого уровня и 5,5 порядка при коэффициенте сжатия 0.98

Далее необходимо проанализировать зависимость количества сохраненной энергии сигнала от коэффициента сжатия, результаты представлены в таблице 2.8.

Таблица 2.8. Зависимость количества сохраненной энергии от коэффициента сжатия для вейвлет-преобразования Хаара пятого порядка

$K_{сж}$	0,5	1	1,5	2	2,5
$Q_{сохр}$	98,68	94,63	92,71	88,65	82,28

Данную зависимость удобно представить в виде графика ( рисунок 2.73):

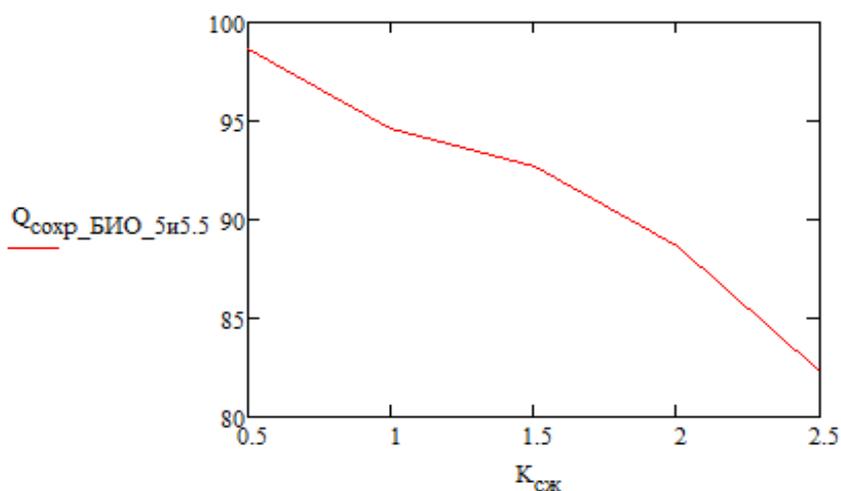


Рис. 2.73. Зависимость количества сохраненной энергии от коэффициента сжатия для биортогонального пятого уровня и 5,5 порядка

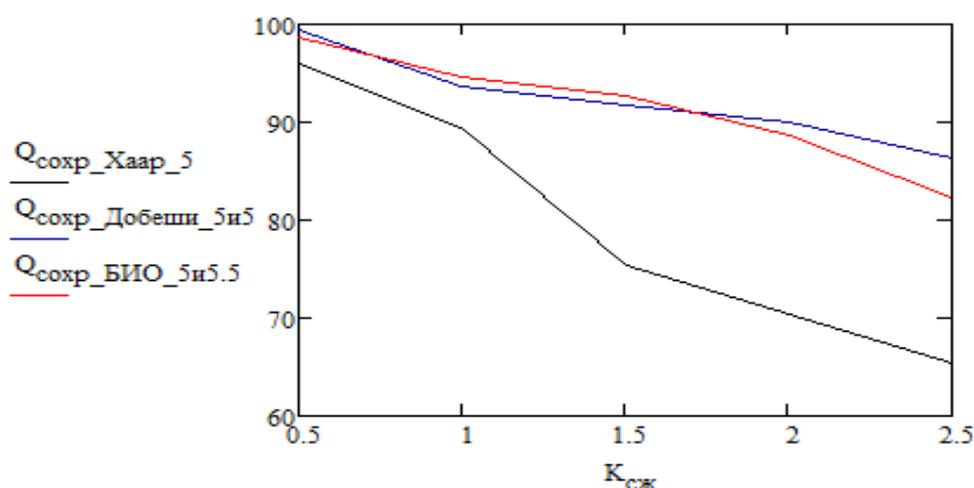


Рис. 2.74. Сравнение энергетических характеристик сжатых сигналов при различных методах вейвлет-преобразований

В результате выполнения данного индивидуального задания был проведен теоретический обзор современных алгоритмов вейвлет-преобразований для сжатия одномерных сигналов.

Было установлено, что использование вейвлет-преобразований является эффективным средством для преобразования информации для передачи по каналам связи, так в современном телекоммуникационном оборудовании предусмотрено использование помехоустойчивых алгоритмов кодирования с использованием избыточности.

Так же, были приобретены практические навыки работы с приложением Wavelet Toolbox Matlab, которое позволяет наглядно продемонстрировать принцип действия сжатия информации путем вейвлет-преобразований.

Были построены графики зависимостей количества сохраненной энергии сигнала от коэффициента сжатия для различных алгоритмов вейвлет-преобразований.

В том числе, было установлено, что использование вейвлет-преобразований Добеши и Хаара эффективнее использования биортогонального вейвлет-преобразования с точки зрения сохранной энергии сигнала.

Данная работа имеет практическое значения, так как была разработана методика для выполнения лабораторной работы по сжатию информации с использованием вейвлет-преобразований при помощи программного обеспечения Matlab 2015.

Таким образом, можно сделать обоснованный вывод о том, что вейвлет-анализ и обработка сигналов на сегодняшний день быстро развивающаяся область прикладной науки, которая в будущем может стать ведущим способом обработки информации. На основе вейвлет-преобразования могут быть построены высокоэффективные алгоритмы сжатия информации, что актуально в наши дни.

### **ГЛАВА 3. ПОМЕХОУСТОЙЧИВОЕ КОДИРОВАНИЕ В ТЕЛЕКОММУНИКАЦИОННЫХ СИСТЕМАХ**

#### **Корректирующие коды в телекоммуникационных системах**

В теории современных телекоммуникационных систем значительное внимание уделяется изучению методов кодирования информации. В технике электрической связи широко используют результаты теории кодирования.

**Кодирование** - операция отождествления символов или групп символов одного кода с символами или группами символов другого кода. Необходимость кодирования возникает, прежде всего, из потребности приспособить форму сообщения к данному каналу связи или к какому-либо другому устройству, предназначенному для преобразования или хранения информации.

Типичная структурная схема системы передачи дискретной информации (СПДИ) приведена на рис. 3.1. Источник вырабатывает сообщения, которые необходимо передавать по каналу СПДИ. Это могут быть последовательности дискретных сообщений (данные, телеграфные сообщения и т.д.) либо непрерывные сообщения (речь, телевидение, результаты телеизмерений и др.), преобразованные в цифровую форму.



Рис. 3.1. Структурная схема системы передачи дискретной информации

Реальные сообщения содержат избыточность и для согласования источника с каналом передачи информации используют кодер источника. Совместно с декодером они образуют кодек источника. Методы кодирования источников изучались в модуле 2. Основная задача любой СДПИ - передача информации с заданными верностью и скоростью передачи. Эти требования находятся в противоречии, причем, повышение скорости передачи информации в реальных СПДИ приводит к снижению помехоустойчивости и верности передачи.

Согласно известных теорем К. Шеннона, в принципе возможно сколь угодно большое повышение верности передачи информации, если скорость передачи по каналу  $R_{\text{кан}}$  не превышает пропускной способности канала  $C_{\text{к}}$ . Достигается это применением достаточно длинных корректирующих кодов (КК).

**Корректирующие коды** - это коды, позволяющие обнаруживать или исправлять ошибки, возникающие при передаче сообщений по каналам связи.

С этой целью в структуру КК вводится **избыточность**.

**Кодек КК** (кодер и декодер канала) приведены на рис.3.1. В реальных условиях длина кода ограничена допустимой сложностью устройств кодирования и, прежде всего, декодирования, поэтому эффект от применения корректирующих кодов зависит от параметров кода и ограничений на реализацию кодека канала.

Современная теория предлагает широкий набор корректирующих кодов, различных по структуре, принципам построения и корректирующей способности. В последующих разделах рассмотрены классы кодов, для которых разрабо-

таны достаточно простые и эффективные алгоритмы кодирования/декодирования и которые наиболее перспективны для использования в каналах телекоммуникационных систем.

## Классификация корректирующих кодов

В теории и технике помехоустойчивого кодирования известно множество корректирующих кодов, которые могут быть классифицированы по различным признакам. Классификация кодов приведена на рис. 3.2.

По способу формирования КК подразделяются на блочные и непрерывные. Формирование *блочных кодов* предусматривает разбиение передаваемых цифровых последовательностей на отдельные блоки, которые подаются на вход кодера. Каждому такому блоку на выходе кодера соответствует блок кодовых символов, работа кодера определяется правилом, или *алгоритмом кодирования*. Формирование *непрерывных кодов* осуществляется непрерывно во времени, без разделения на блоки, что и определяет наименование этого класса кодов. Блочные коды исторически были предложены и изучены ранее, на заре развития теории кодирования.

В классе непрерывных кодов следует отметить *сверточные коды*, которые по характеристикам превосходят блочные коды, и, по этой причине, находят широкое применение в телекоммуникационных системах. Многие коды носят имена ученых, которые их предложили и исследовали. Таким примером является непрерывный код Финка-Хагельбаргера, предложенный советским ученым Л.М. Финком и немецким специалистом Р. Хагельбаргером. Длительное время этот код служил в литературе показательным примером непрерывного кода с простым алгоритмом кодирования/декодирования, но после открытия сверточных кодов уступил им место.

Для описания процедур кодирования/декодирования как блочных, так и сверточных кодов используют адекватный математический аппарат. Для описания *линейных кодов* используется хорошо разработанный аппарат линейной

алгебры. Формирование нелинейных кодов производится с применением нелинейных процедур. Такой подход позволяет в некоторых случаях получить *нелинейные коды* с рядом специальных свойств. В теории и технике кодирования важной является проблема сложности реализации процедур кодирования/декодирования и, в особенности, процедур декодирования. Поэтому некоторые классы кодов (коды Хемминга, циклические коды Боуза-Чоудхури-Хоквингема, Рида-Соломона, Файра и др.) были разработаны совместно с алгоритмами декодирования, связанными со структурными свойствами этих кодов. И, наоборот, разработка новых алгоритмов декодирования сверточных кодов (алгоритм А. Витерби, последовательное декодирование, пороговое декодирование) инициировала поиски соответствующих сверточных кодов. Отличительные преимущества корректирующих кодов (как блочных, так и сверточных) побуждали поиски новых подходов к реализации путей

повышения помехоустойчивости и эффективности телекоммуникационных систем. На рис. 3.2 отмечены, соответственно, *новые методы кодирования*: сигнально-кодовые конструкции, турбокоды, пространственно-временные коды и т.п.

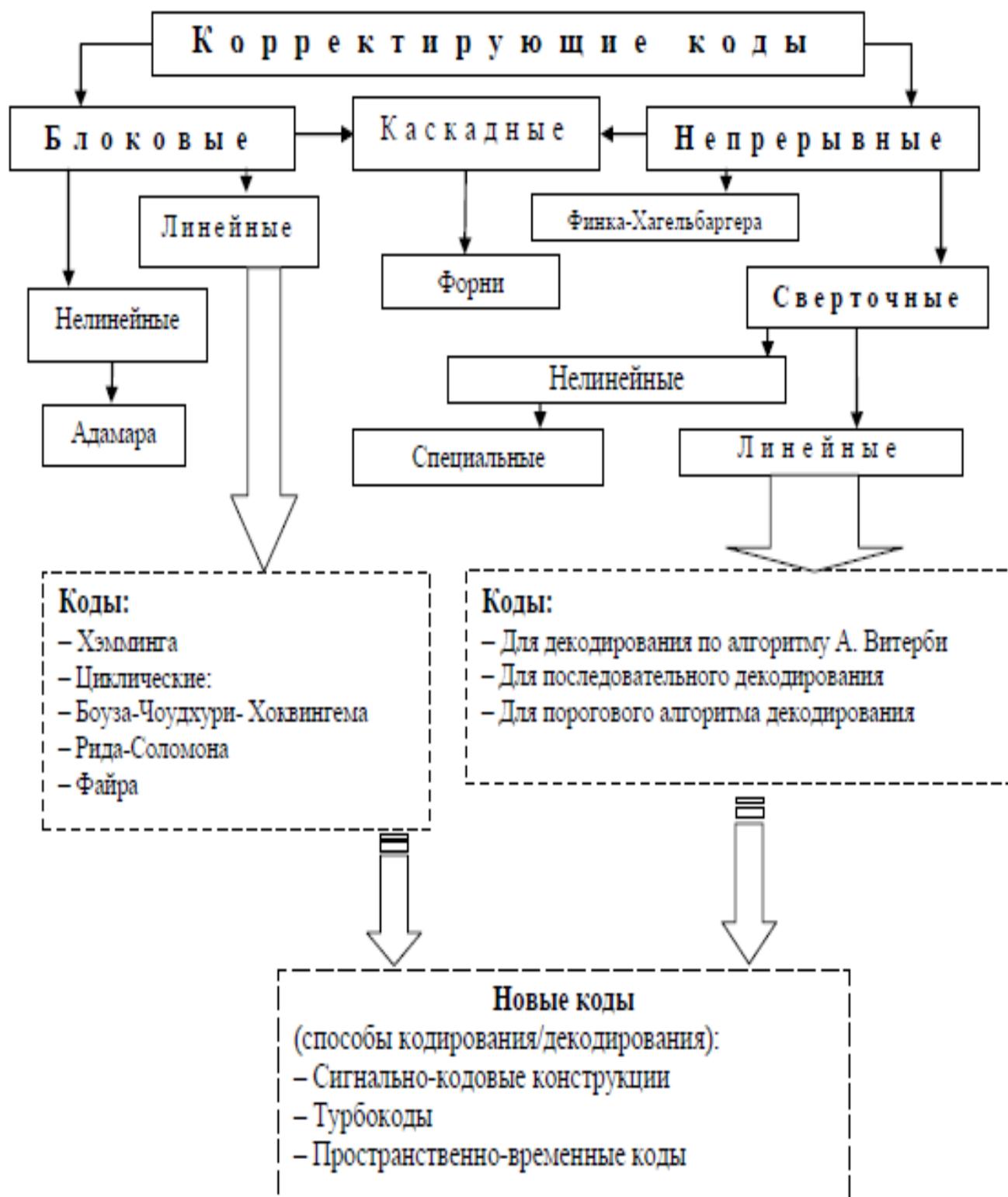


Рис. 3.2. Классификация корректирующих кодов

### 3.1 Исследование кодов Хемминга, БЧХ (Боуза-Чоудхури-Хоквенгема), Рида-Соломона на базе MATLAB 2015

#### Код Хэмминга

Наиболее известным линейным кодом является **блоковый код Хэмминга**. В данном пункте, будет рассмотрен (7,4)-код Хэмминга. Линейный блочный код, для которого выполняется неравенство Хэмминга, является кодом Хэмминга, то есть код (7,4) не единственный код, который является данным кодом.

Введем некоторые обозначения, которые составляют основу помехоустойчивого кодирования:

$k$  – количество информационных бит

$n$  – количество бит на выходе кодера

$r$  – количество проверочных(избыточных) бит

Вместо  $k$  бит информационного вектора в канал передается  $n$  бит кодового вектора. В этом случае говорят об избыточном кодировании со скоростью:  $R = \frac{n}{k}$ .

Чем ниже скорость, тем больше избыточность кода, и тем большими возможностями для защиты от ошибок он обладает. Однако, следует учитывать, что с увеличением избыточности затраты на передачу информацию также возрастают.

В кодировании и декодировании участвуют генераторная и проверочная матрицы, структура которых следующая:

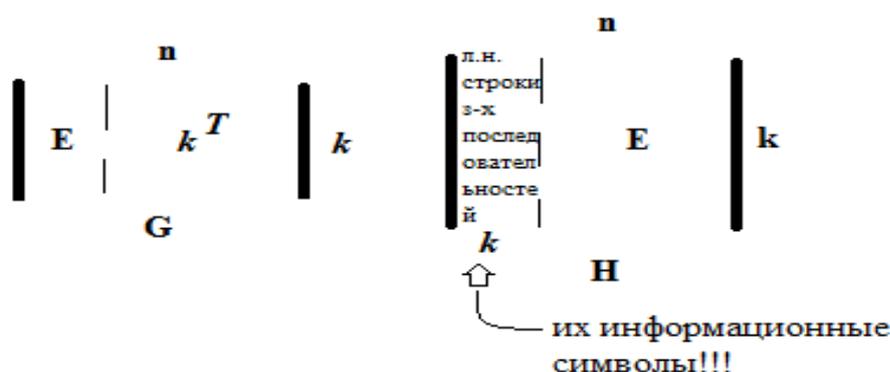


Рис. 3.3. Генераторная и проверочная матрицы

Также, код Хэмминга имеет минимальное кодовое расстояние в пространстве равное трем, из этого следует, что данный код умеет обнаруживать любые двукратные ошибки и исправлять любые однократные.

Рассмотрим код Хэмминга (7,4) в теории.

Для этого кода генераторная и проверочная матрицы уже заготовлены и имеют следующий вид:

$$G_{4 \times 7} = \begin{pmatrix} 1 & 1 & 0 & 1 & 0 & 0 & 0 \\ 0 & 1 & 1 & 0 & 1 & 0 & 0 \\ 1 & 1 & 1 & 0 & 0 & 1 & 0 \\ 1 & 0 & 1 & 0 & 0 & 0 & 1 \end{pmatrix} \quad H_{3 \times 7} = \begin{pmatrix} 1 & 0 & 0 & 1 & 0 & 1 & 1 \\ 0 & 1 & 0 & 1 & 1 & 1 & 0 \\ 0 & 0 & 1 & 0 & 1 & 1 & 1 \end{pmatrix}$$

Рис. 3.4. Структура генераторной и проверочной матрицы для кода (7,4)

Кодирование осуществляется следующим образом:

Кодовое слово  $n$  и информационное слово  $k$  связаны соотношением:

$$n = v = k * G$$

где  $G$  — порождающая матрица, структура которой была описана выше.

Например, информационный вектор  $k = (1010)$  отобразится в кодовый вектор следующим образом:

$$v = (1010) \times \begin{pmatrix} 1 & 1 & 0 & 1 & 0 & 0 & 0 \\ 0 & 1 & 1 & 0 & 1 & 0 & 0 \\ 1 & 1 & 1 & 0 & 0 & 1 & 0 \\ 1 & 0 & 1 & 0 & 0 & 0 & 1 \end{pmatrix} = (0011010)$$

Рис. 3.5. Кодирование кода (7,4)

Легко заметить, что последние четыре разряда кодового вектора совпадают с информационным вектором. Это свойство называется систематичностью кода.

Здесь первые три бита последовательности являются проверочными (избыточными).

Декодирование осуществляется на получении **синдрома** и производится это все следующим образом:

Система проверочных уравнений выглядит:

$$s = n = v * H^T$$

Вектор  $s$  принято называть **синдромом**. Таким образом, ошибка будет обнаружена, если хотя бы одна из компонент  $s$  не равна нулю.

При передаче информационного слова  $a = (1010)$  по каналу без шумавыходной вектор был  $n = (0011010)$ . Можем убедиться, что в этом случае синдром равен 0.

$$s = (0011010) \times \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \\ 1 & 1 & 0 \\ 0 & 1 & 1 \\ 1 & 1 & 1 \\ 1 & 0 & 1 \end{pmatrix} = (000)$$

Рис. 3.6. Получение синдрома для кода (7,4)

Если, например, в кодовом слове произошла одиночная ошибка на четвертой позиции ( $r = (0010010)$ ), то синдромом является четвертая строка транспонированной проверочной матрицы.

$$s = (0010010) \circ \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \\ 1 & 1 & 0 \\ 0 & 1 & 1 \\ 1 & 1 & 1 \\ 1 & 0 & 1 \end{pmatrix} = (110)$$

Рис. 3.7. Нахождение ошибочного бита для кода (7,4)

Перебрав все возможные позиции одиночной ошибки, получим полную таблицу синдромов однократных ошибок - таблицу соответствий номера ошибочного разряда получающемуся при этом синдрому. Таким образом, производится декодирование. Если ошибки найдены, то они соответственно исправляются.

### Проведение эксперимента и обработка результатов в MATLAB 2015

Задание:

1. Собрать схему
2. Подготовить схемы для реализации кодов Хэмминга (7,4) и (15,11) основываясь на примере, представленном в отчете.

При коде (7,4) в параметрах кодера и декодера MessageLengthK, от M-degreeprimitivepolynomial: устанавливается gfprimfd(3,'min'), а при коде (15,11) gfprimfd(4,'min').

3. Для полученных кодов изменять вероятность ошибки в пределах от 0 до 1 (не менее 4-х точек) и снимать с дисплеев полученные входные последовательности, закодированные последовательности, декодированные последовательности и ошибки.

4. Построить графики зависимости числа обнаруженных ошибок от вероятности ошибки для кодов (7,4) и (15,11).

В рабочем поле необходимо собрать схему для работы кода Хэмминга (7,4). Схема представлена на рисунке 3.8.

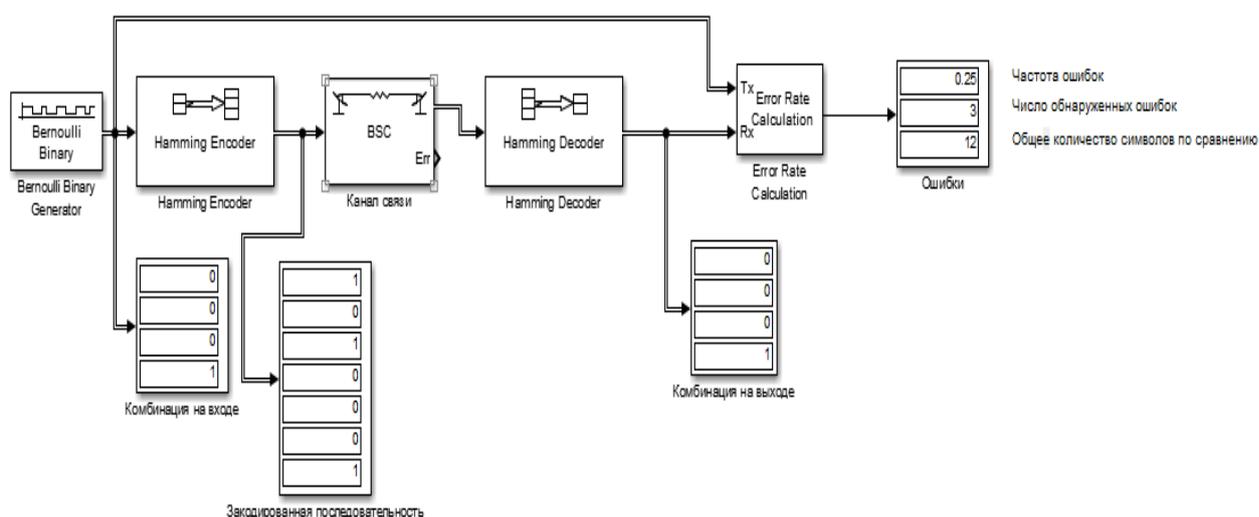


Рис. 3.8. Линия передачи с применением кода Хэмминга

В состав линии с кодированием входят:

1. BernoulliBinaryGenerator
2. HammingEncoder
3. Binary Symmetric Channel (каналпередачи)
4. HammingDecoder
5. Error RateCalculation (анализаторошибок)
6. Display

Устанавливаем характеристики блоков для кода (7,4)

**Bernoulli Binary Generator**

Generate a Bernoulli random binary number.  
To generate a vector output, specify the probability as a vector.

Parameters

Probability of a zero:

Initial seed:

Sample time:

Frame-based outputs

Samples per frame:

Output data type:

Рис. 3.9. Параметры Bernoulli Binary Generator

**Hamming Encoder (mask) (link)**

Create a Hamming code with message length  $K$  and codeword length  $N$ . The number  $N$  must have the form  $2^M - 1$ , where  $M$  is an integer greater than or equal to 3.  $K$  must equal  $N - M$ .

The input must contain exactly  $K$  elements. If it is frame-based, then it must be a column vector.

Parameters

Codeword length  $N$ :

Message length  $K$ , or  $M$ -degree primitive polynomial:

Рис. 3.10. Параметры Hamming Encoder

**Binary Symmetric Channel (mask) (link)**

Add binary errors to the input signal.

Parameters

Error probability:

Initial seed:

Output error vector

Output data type:

Рис. 3.11. Параметры Binary Symmetric Channel

Hamming Decoder (mask) (link)

Recover a binary message vector from a binary Hamming codeword vector. The message is of length K and the codeword is of length N, where N has the form  $2^M - 1$ , for some integer M greater than or equal to 3. K must equal N-M.

The input must contain exactly N elements. If it is frame-based, then it must be a column vector.

Parameters

Codeword length N:

Message length K, or M-degree primitive polynomial:

Рис. 3.12. Параметры Hamming Decoder

Error Rate Calculation (mask) (link)

Compute the error rate of the received data by comparing it to a delayed version of the transmitted data. The block output is a three-element vector consisting of the error rate, followed by the number of errors detected and the total number of symbols compared. This vector can be sent to either the workspace or an output port.

The delays are specified in number of samples, regardless of whether the input is a scalar or a vector. The inputs to the 'Tx' and 'Rx' ports must be scalars or column vectors.

The 'Stop simulation' option stops the simulation upon detecting a target number of errors or a maximum number of symbols, whichever comes first.

Parameters

Receive delay:

Computation delay:

Computation mode:

Output data:

Рис. 3.13 Параметры Error Rate Calculation



Рис. 3.14. Комбинация на входе, Закодированная последовательность, Комбинация на выходе, Ошибки (вероятность ошибок равна 0,2)

Анализируя рисунок выше, можно сделать вывод, что комбинация на входе совпадает с комбинацией на выходе, таким образом, передача осуществилась удачно. Что касается

ошибок, то их частота равна 0,25, число обнаруженных ошибок равно 3, общее количество символов по сравнению равно 12. Кодирование и декодирование здесь осуществляется методом описанном в выше.

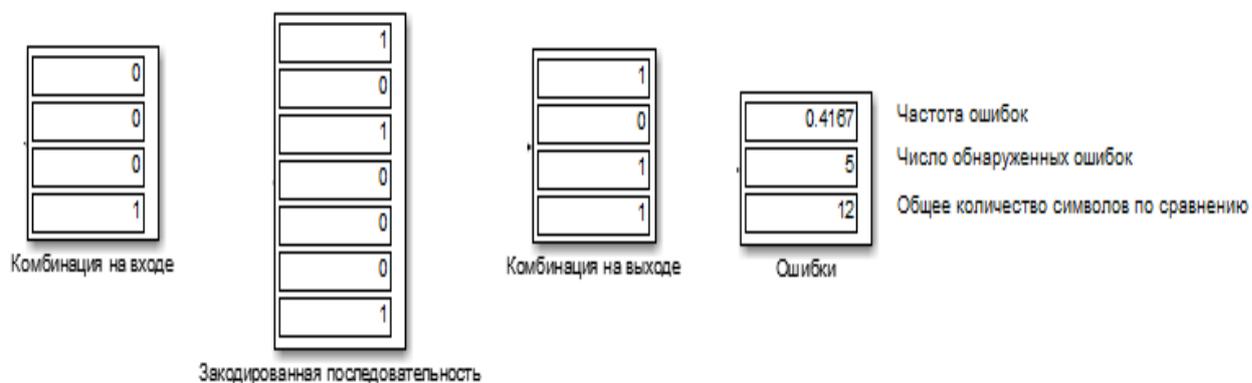


Рис. 3.15. Комбинация на входе, Закодированная последовательность, Комбинация на выходе, Ошибки (вероятность ошибок равна 0,4)

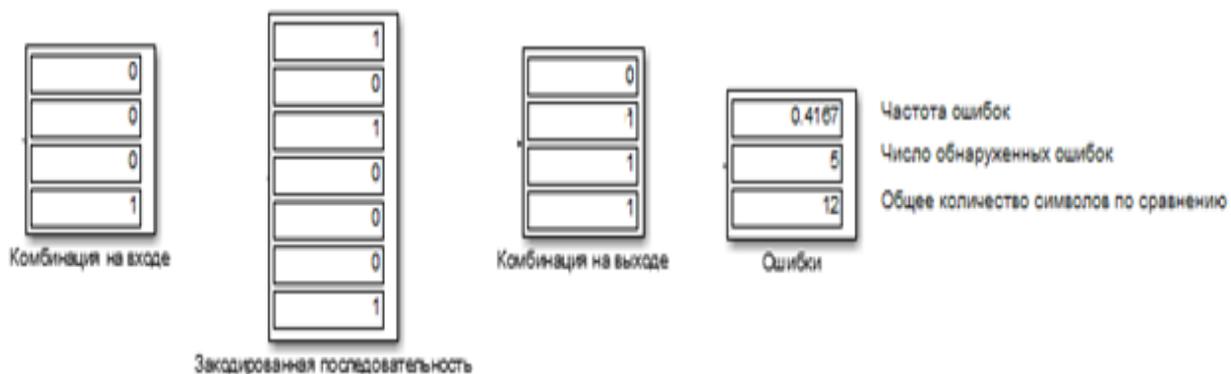


Рис. 3.16. Комбинация на входе, Закодированная последовательность, Комбинация на выходе, ошибки (вероятность ошибок равна 0,6)



Рис. 3.17. Комбинация на входе, Закодированная последовательность, Комбинация на выходе, Ошибки (вероятность ошибок равна 0,8)



Рис. 3.18. График зависимости числа ошибок (OY) от вероятности ошибки (OX) для кода Хэмминга (7,4)

Устанавливаем характеристики блоков для кода (15,11)

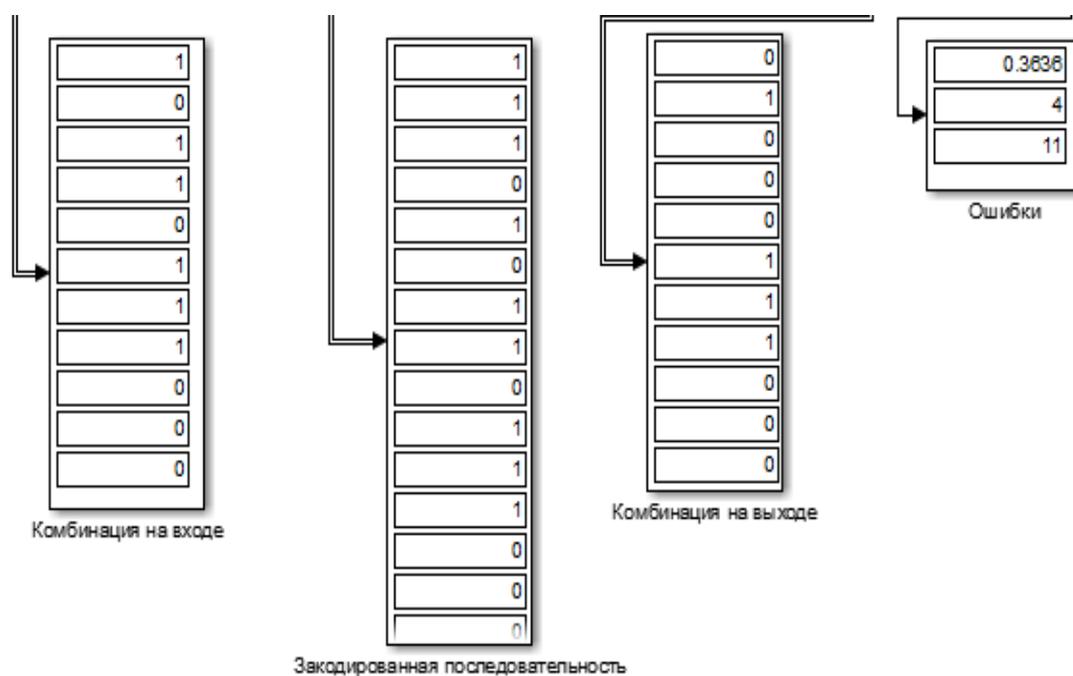


Рис. 3.19. Комбинация на входе, Закодированная последовательность, Комбинация на выходе, Ошибки (вероятность ошибок равна 0,2)

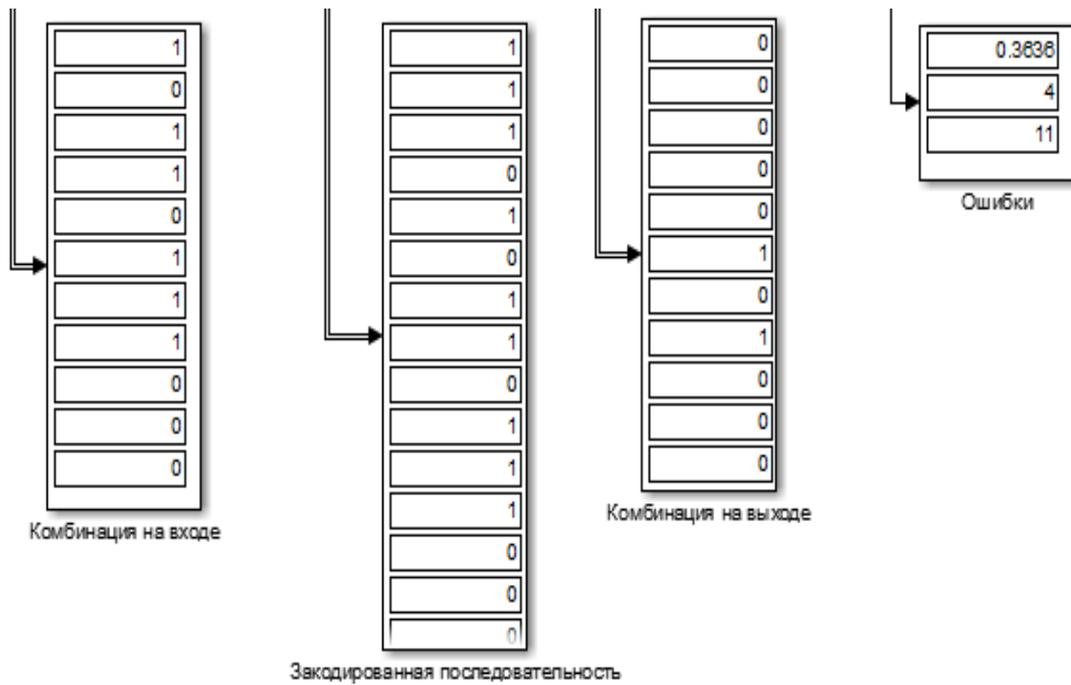


Рис. 3.20. Комбинация на входе, Закодированная последовательность, Комбинация на выходе, Ошибки (вероятность ошибок равна 0,4)

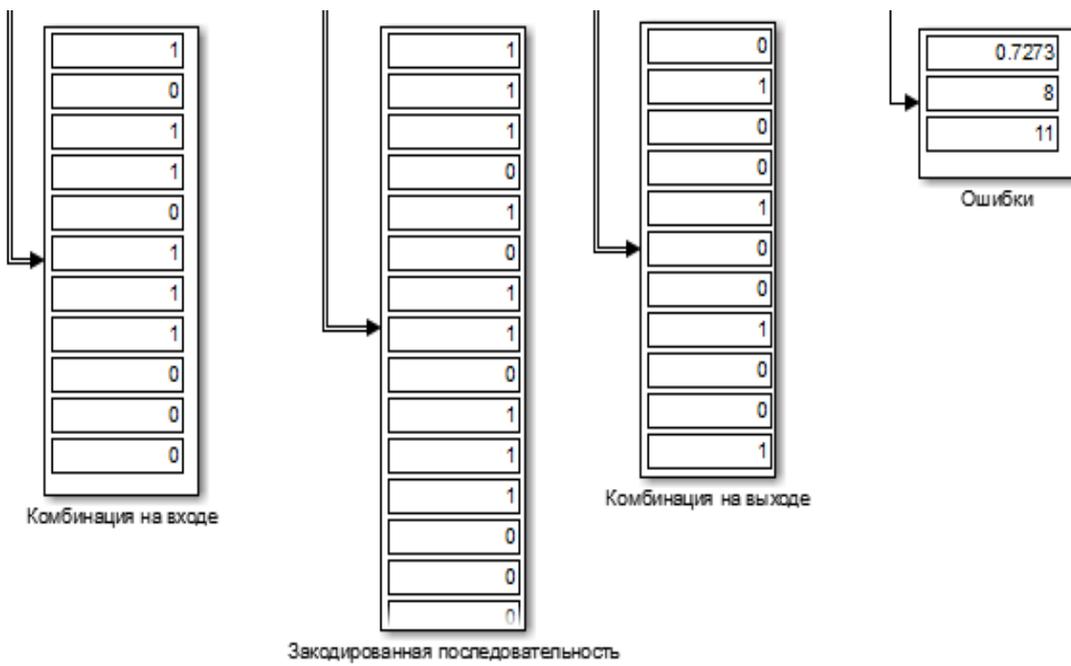


Рис. 3.21. Комбинация на входе, Закодированная последовательность, Комбинация на выходе, Ошибки (вероятность ошибок равна 0,6)

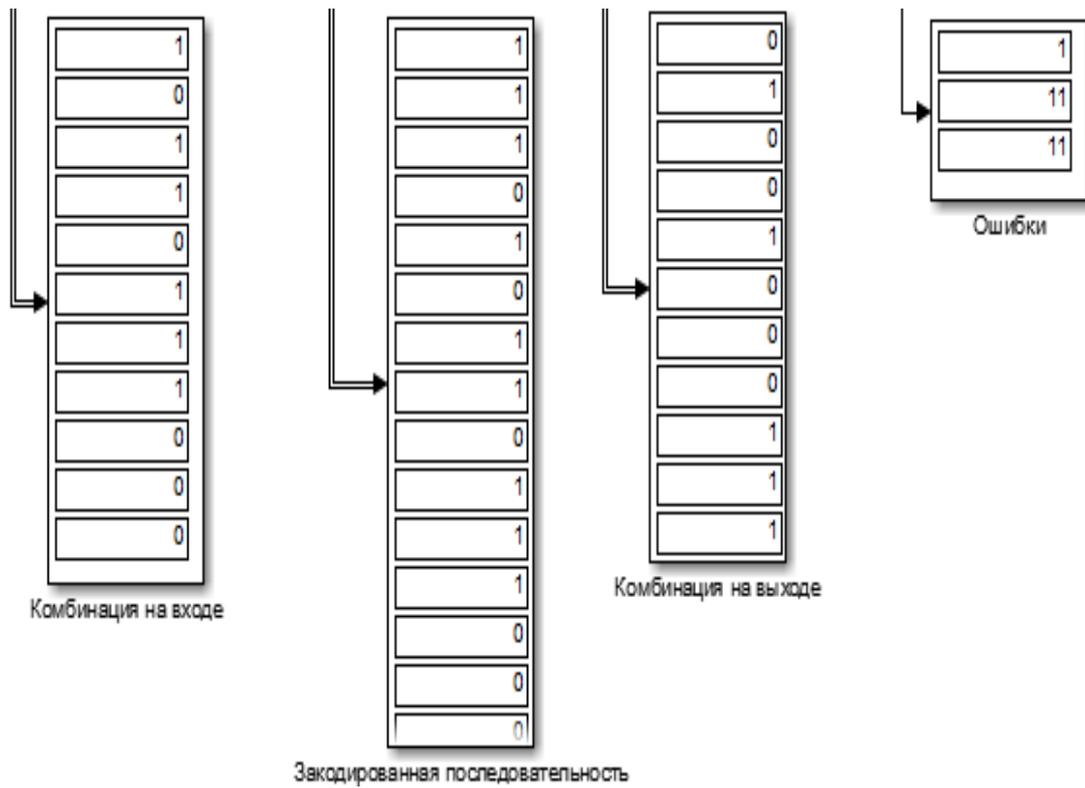


Рис. 3.22. Комбинация на входе, Закодированная последовательность, Комбинация на выходе, Ошибки (вероятность ошибок равна 0,8)

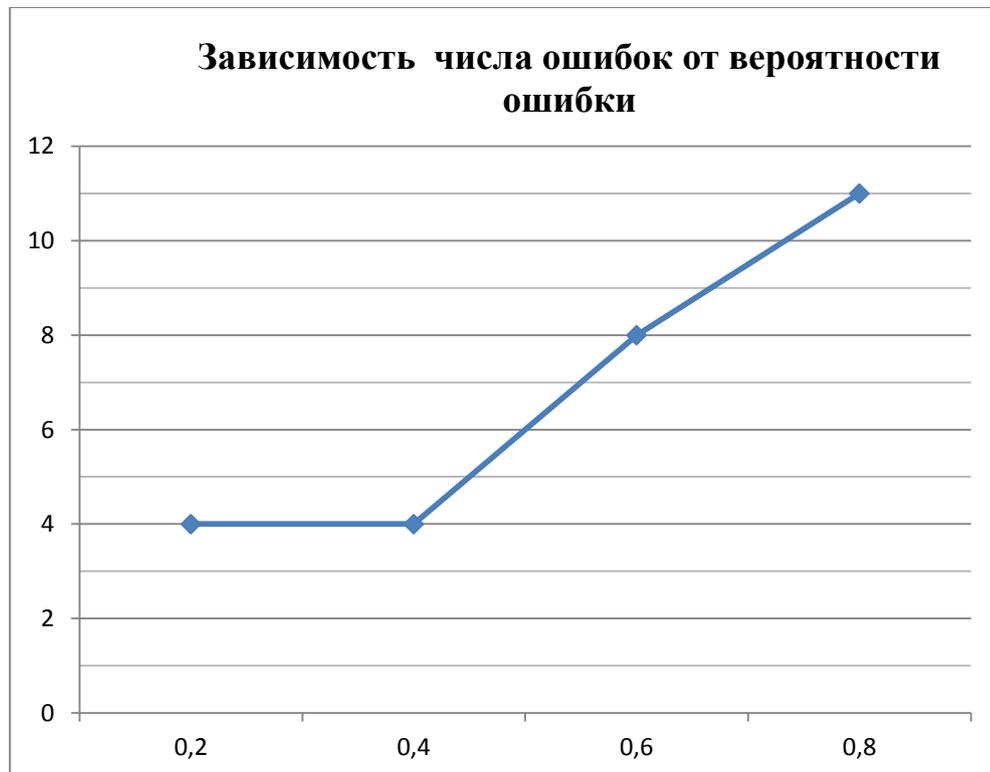


Рис. 3.23. График зависимости числа ошибок (OY) от вероятности ошибки (OX) для кода Хэмминга (15,11)

В результате проверки построена схема линии передачи с кодированием Хэмминга в среде Simulink. Построены графики зависимостей числа ошибок на выходе декодера от вероятности ошибки в канале связи для кодов (7,4) и (15,11).

Из графиков (рисунок 3.18 и рисунок 3.23) видно, что число ошибок увеличивается с ростом вероятности ошибок.

### Код БЧХ (Боуза-Чоудхури-Хоквенгема)

Код БЧХ является циклическим кодом. С циклическим кодом Хэмминга у них много чего общего, а именно алгоритм кодирования, который отличается только нахождением генераторного полинома, а процесс декодирования полностью схож. Необходимо начать с небольшого введения в код БЧХ. Многочлен  $g(x)$  степени называется *примитивным*, если  $x^j + 1$  делится на  $g(x)$  без остатка для  $j = 2^k - 1$  и не делится ни для какого меньшего значения (где  $k$  – количество информационных бит). Например, многочлен  $1 + x^2 + x^3$  примитивен: он делит  $x^7 + 1$ , но не делит  $x^j + 1$  при  $j < 7$ . Примитивен также многочлен  $1 + x^3 + x^4$  – он делит  $x^{15} + 1$ , но не делит  $x^j + 1$  при  $j < 15$  (для кода (7,15)).

Кодирующий многочлен  $g(x)$  для БЧХ-кода, длина кодовых слов которого  $n$ , строится так. Находится примитивный многочлен минимальной степени такой, что  $n \leq 2^q - 1$  или  $n \geq \log_2(n + 1)$ . Пусть  $\alpha$  – корень этого многочлена, тогда рассмотрим кодирующий многочлен  $g(x) = \text{НОК}(m_1(x), \dots, m_{d-1}(x))$ , где  $m_1(x), \dots, m_{d-1}(x)$  – многочлены минимальной степени, имеющие корнями соответственно  $\alpha, \alpha^2, \dots, \alpha^{d-1}$ .

Построенный кодирующий многочлен производит код с минимальным расстоянием между кодовыми словами, не меньшим  $d$ , и длиной кодовых слов  $n$ .

### Кодирование:

Например, нужно построить БЧХ-код с длиной кодовых слов  $n=15$  и минимальным расстоянием между кодовыми словами  $d=5$ . Степень примитивного многочлена равна  $q = \log_2(n + 1) = 4$  и сам он равен  $1 + x^3 + x^4$  (примитивный многочлен 4-ой степени для кода (7,15)). Пусть  $\alpha$  – его корень, тогда  $\alpha^2$  и  $\alpha^4$  – также его корни. Минимальным многочленом для  $\alpha^3$  будет  $1 + x + x^2 + x^3 + x^4$ . Следовательно,

$$\begin{aligned} g(x) &= \text{НОК}(1 + x^3 + x^4, 1 + x + x^2 + x^3 + x^4) = (1 + x^3 + x^4)(1 + x + x^2 + x^3 + x^4) \\ &= 1 + x + x^2 + x^4 + x^8 \end{aligned}$$

Степень полученного многочлена равна 8, построенный BCH-код будет (7,15) кодом. Слово 1000100 или  $a(x) = x^4 + 1$  будет закодировано кодовым словом  $a(x)g(x) = x^{12} + x^6 + x^5 + x^2 + x + 1$  или 111001100000100. На практике будет рассмотрен код BCH (15,7) и BCH (15,11).

### Декодирование:

Декодирование производится путем деления закодированной последовательности на генераторный полином, который использовался при кодировании. Полученная последовательность при делении и будет декодированной последовательностью. В лучшем случае, остатка при делении не будет, это значит, что ошибок не выявлено.

Если же остаток есть, называется он не иначе как **синдром**, в таком случае ошибки присутствуют в закодированной последовательности. В этом случае поступают так. Закодированную последовательность складывают с вектором ошибок по модулю два, и исправляют ошибочный бит. Вектор ошибок формируется с помощью специальных схем, которые анализируя закодированную последовательность, формируют данный вектор.

Еще одним способ исправления ошибок является следующий метод. На основании полученного генераторного полинома строится схема и на каждом такте ее работы определяется **синдром**. В данном случае необходимо получить два синдрома, один из них это остаток, от деления полученный при декодировании, а второй это комбинация 100. Далее производится вычитание номера такта комбинации 100 и номера такта остатка. Полученная разность и является номером бита в закодированной комбинации.

## Проведение эксперимента и обработка результатов

Задание:

1. Собрать схему
2. Подготовить схемы для реализации кодов BCH (15,7) и (15,11) основываясь на примере, представленном в отчете.
3. Для полученных кодов изменять вероятность ошибки в пределах от 0 до 1 (не менее 4-х точек) и снимать с дисплеев полученные входные последовательности, закодированные последовательности, декодированные последовательности и ошибки.
4. Построить графики зависимости числа обнаруженных ошибок от вероятности ошибки для кодов (15,7) и (15,11).
5. Все поэтапное исследование представить в отчете.

В рабочем поле необходимо собрать схему для работы кода BCH (15,7). Схема представлена на рисунке 3.24.

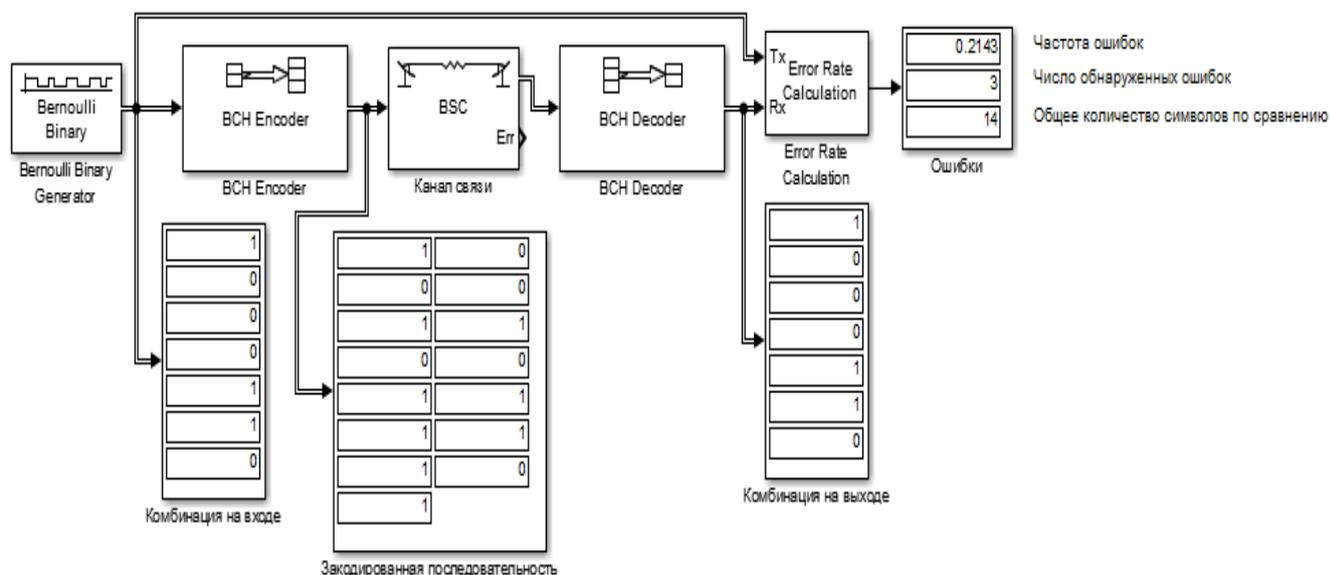


Рис. 3.24. Линия передачи с применением кода БЧХ (15,7)

В состав линии с кодированием входят:

1. BernoulliBinaryGenerator
2. BCHEncoder
3. Binary Symmetric Channel (каналпередачи)
4. BCHDecoder
5. Error RateCalculation (анализаторошибок)
6. Display

Устанавливаем характеристики блоков для кода (15,7)

**Bernoulli Binary Generator**

Generate a Bernoulli random binary number.  
To generate a vector output, specify the probability as a vector.

**Parameters**

Probability of a zero:

Initial seed:

Sample time:

Frame-based outputs

Samples per frame:

Output data type:

Рис. 3.25. Параметры Bernoulli Binary Generator

BCH Encoder (mask) (link)

Encode the message in the input vector using an (N,K) BCH encoder with the narrow-sense generator polynomial. This block accepts a column vector input signal with an integer multiple of K elements. Each group of K input elements represents one message word to be encoded. The values of N and K must produce a valid narrow-sense BCH code.

Shorten the code by setting the shortened message length parameter S. In this case, use full length N and K values to specify the (N, K) code that is shortened to an (N - K + S, S) code.

Parameters

Codeword length, N:  
15

Message length, K:  
7

Рис. 3.26. Параметры BCH Encoder

Binary Symmetric Channel (mask) (link)

Add binary errors to the input signal.

Parameters

Error probability:  
0.2

Initial seed:  
71

Output error vector

Output data type: boolean

Рис. 3.27. Параметры Binary Symmetric Channel

BCH Decoder (mask) (link)

Decode the message in the input vector using an (N,K) BCH decoder with the narrow-sense generator polynomial. This block accepts a column vector input signal with an integer multiple of N elements. Each group of N input elements represents one codeword to be decoded. The values of N and K must produce a valid narrow-sense BCH code.

Shorten the code by setting the shortened message length parameter S. In this case, use full length N and K values to specify the (N, K) code that is shortened to an (N - K + S, S) code.

Parameters

Codeword length, N:  
15

Message length, K:  
7

Рис. 3.28 Параметры BCH Decoder

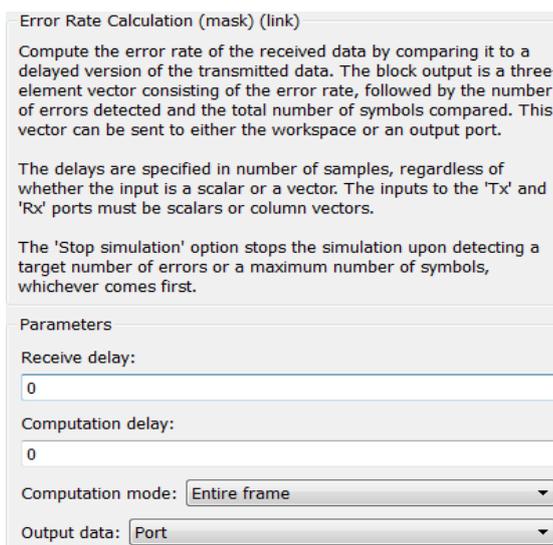


Рис. 3.29. Параметры Error Rate Calculation

Представим полученные результаты

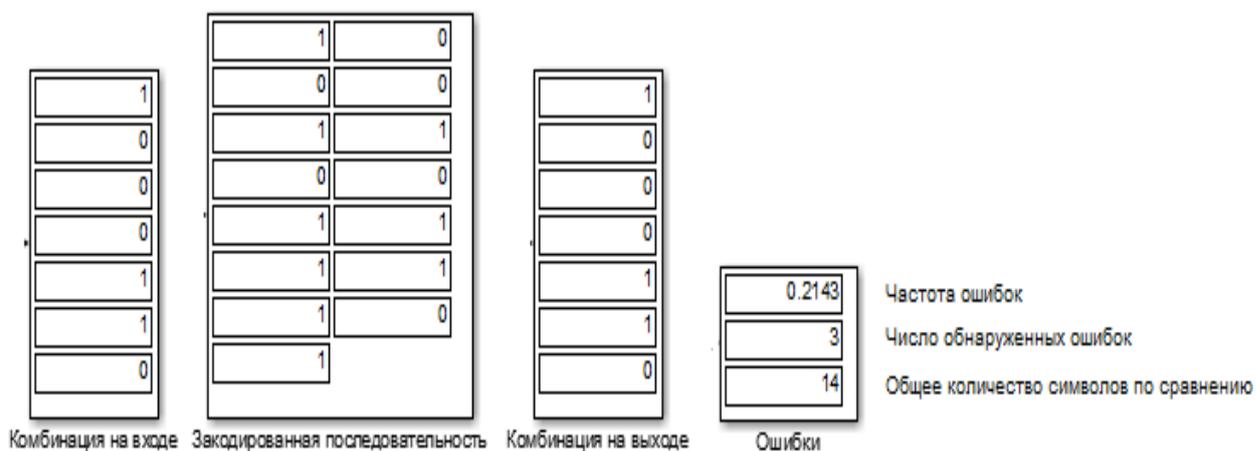


Рис. 3.30. Комбинация на входе, Закодированная последовательность, Комбинация на выходе, Ошибки (вероятность ошибок равна 0,2)

Можно сделать вывод, что комбинация на входе совпадает с комбинацией на выходе, таким образом, передача осуществилась удачно. Что касается ошибок, то их частота равна 0,2143, число обнаруженных ошибок равно 3, общее количество символов по сравнению равно 14. Кодирование и декодирование здесь осуществляется методом описанном выше.

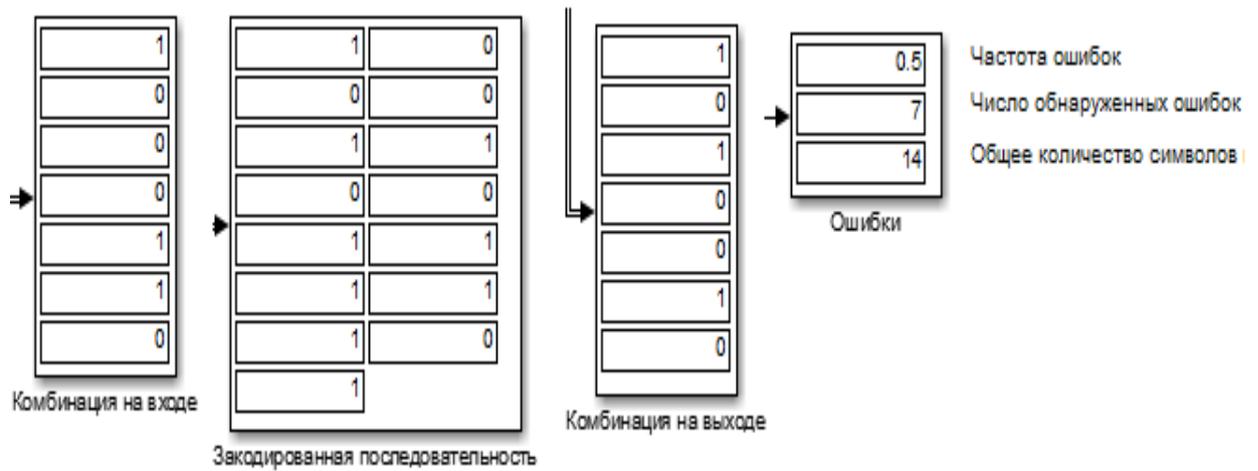


Рис. 3.31. Комбинация на входе, Закодированная последовательность, Комбинация на выходе, Ошибки (вероятность ошибок равна 0,4)

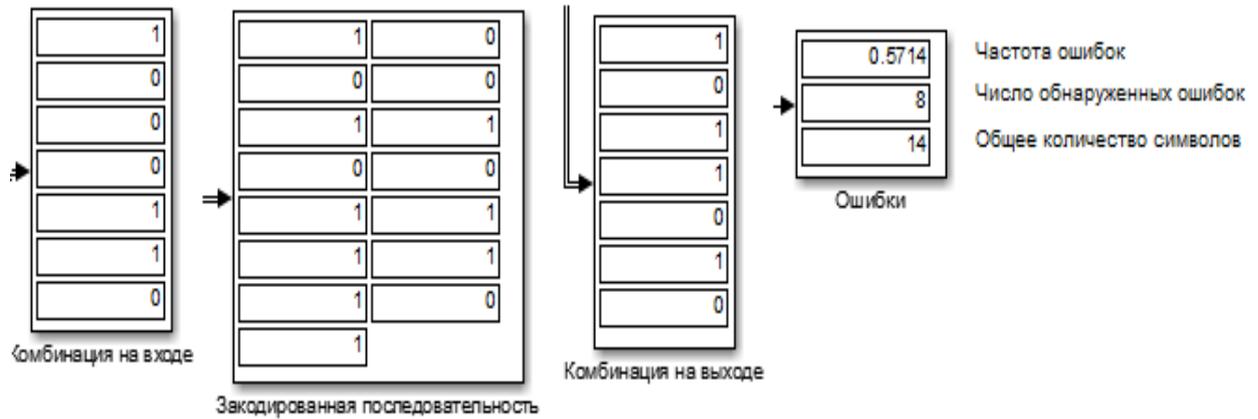


Рис. 3.32. Комбинация на входе, Закодированная последовательность, Комбинация на выходе, Ошибки (вероятность ошибок равна 0,6)

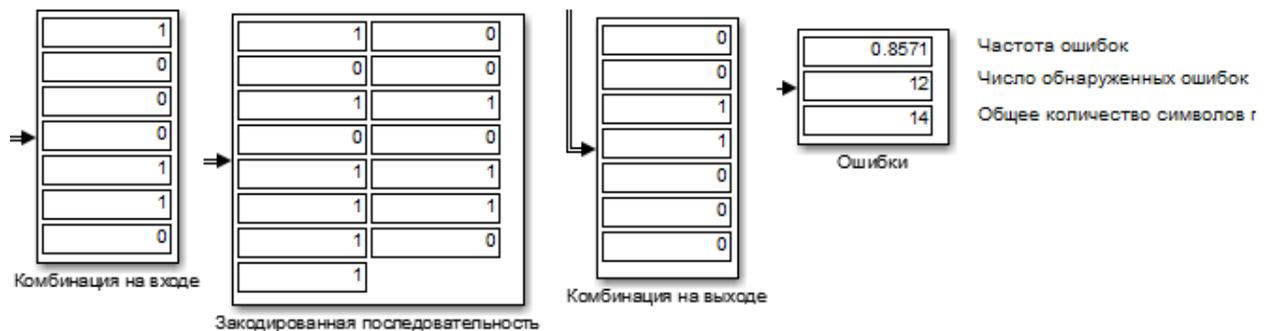


Рис. 3.33. Комбинация на входе, Закодированная последовательность, Комбинация на выходе, Ошибки (вероятность ошибок равна 0,8)

Устанавливаем характеристики блоков для кода (15,11)

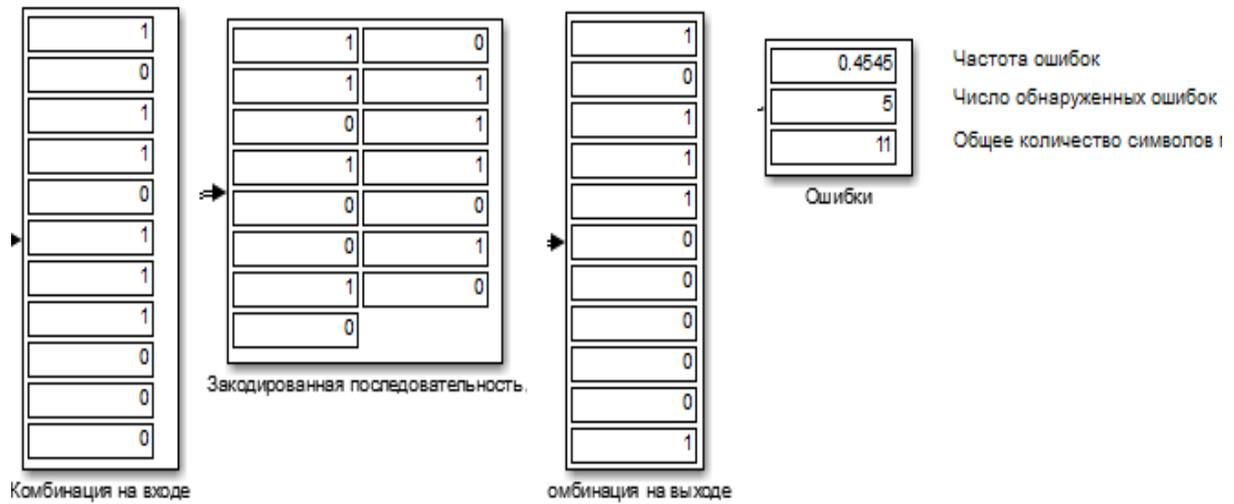


Рис. 3.34. Комбинация на входе, Закодированная последовательность, Комбинация на выходе, Ошибки (вероятность ошибок равна 0,2)

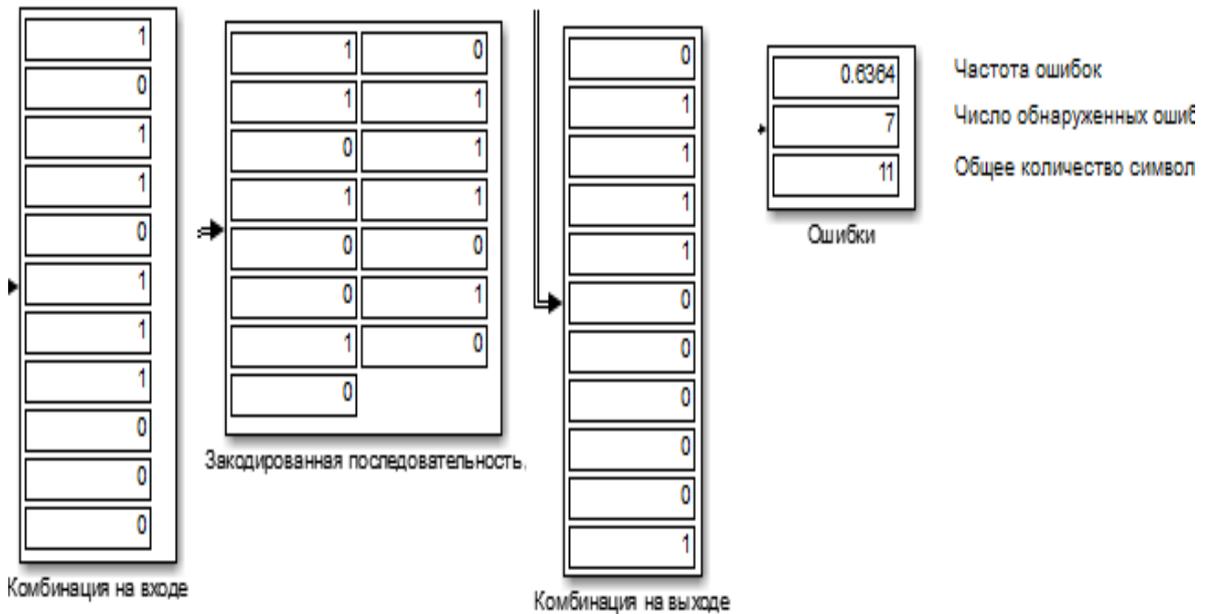


Рис. 3.35. Комбинация на входе, Закодированная последовательность, Комбинация на выходе, Ошибки (вероятность ошибок равна 0,4)

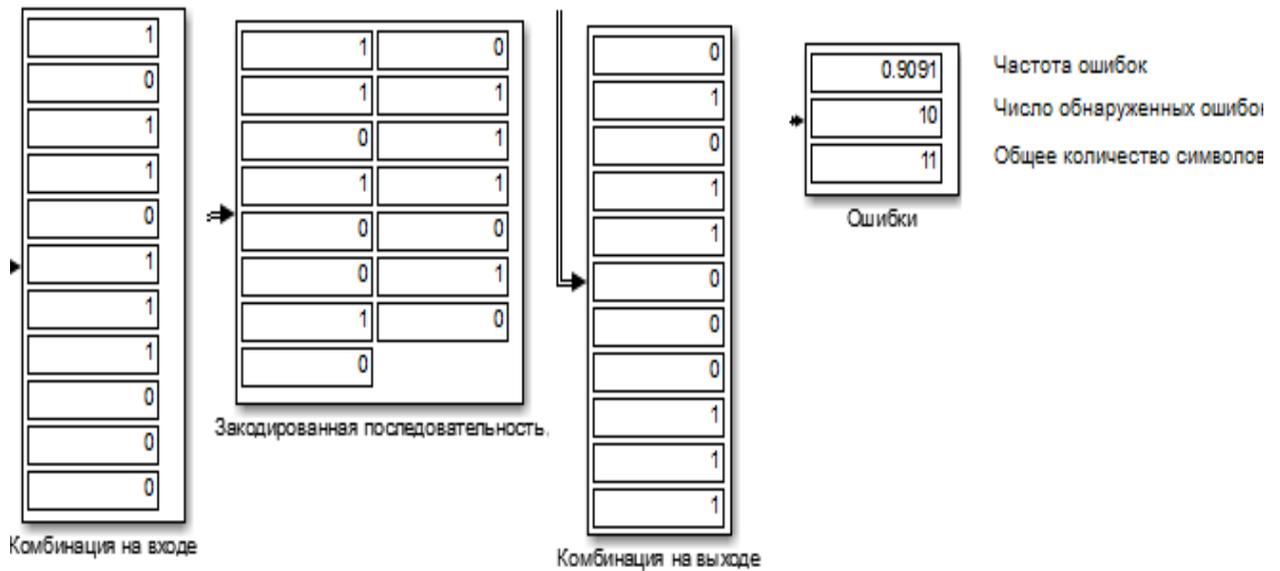


Рис. 3.36 Комбинация на входе, Закодированная последовательность, Комбинация на выходе, Ошибки (вероятность ошибок равна 0,6)

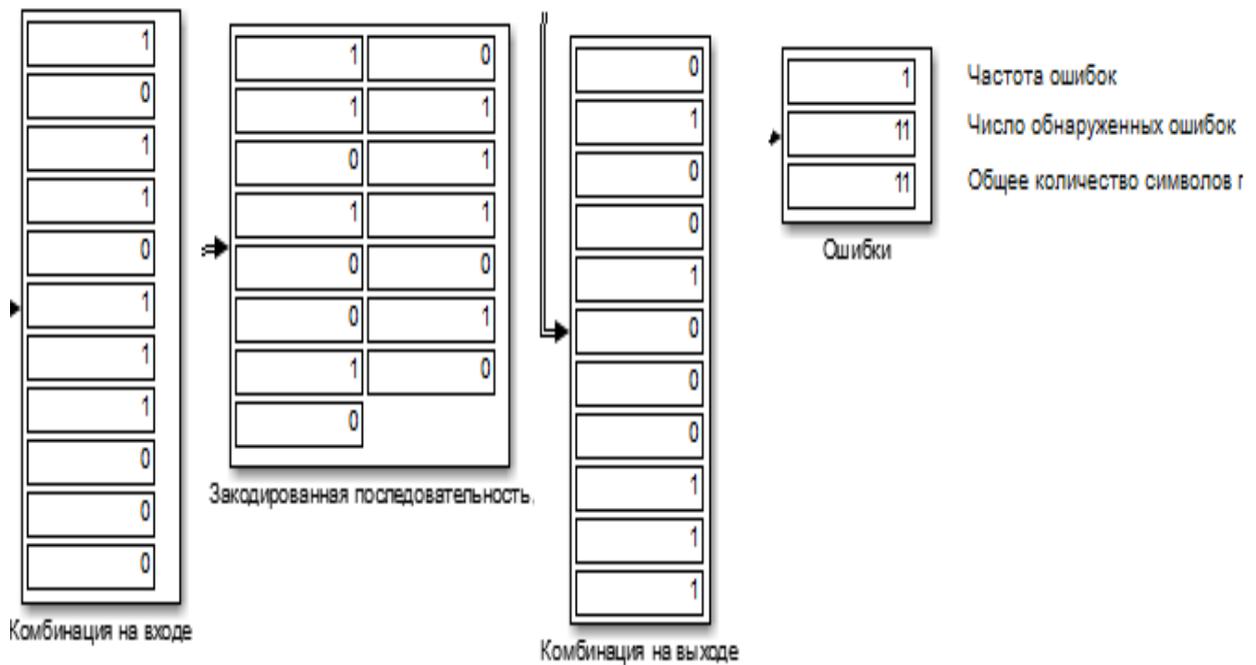


Рис. 3.37. Комбинация на входе, Закодированная последовательность, Комбинация на выходе, Ошибки (вероятность ошибок равна 0,8)

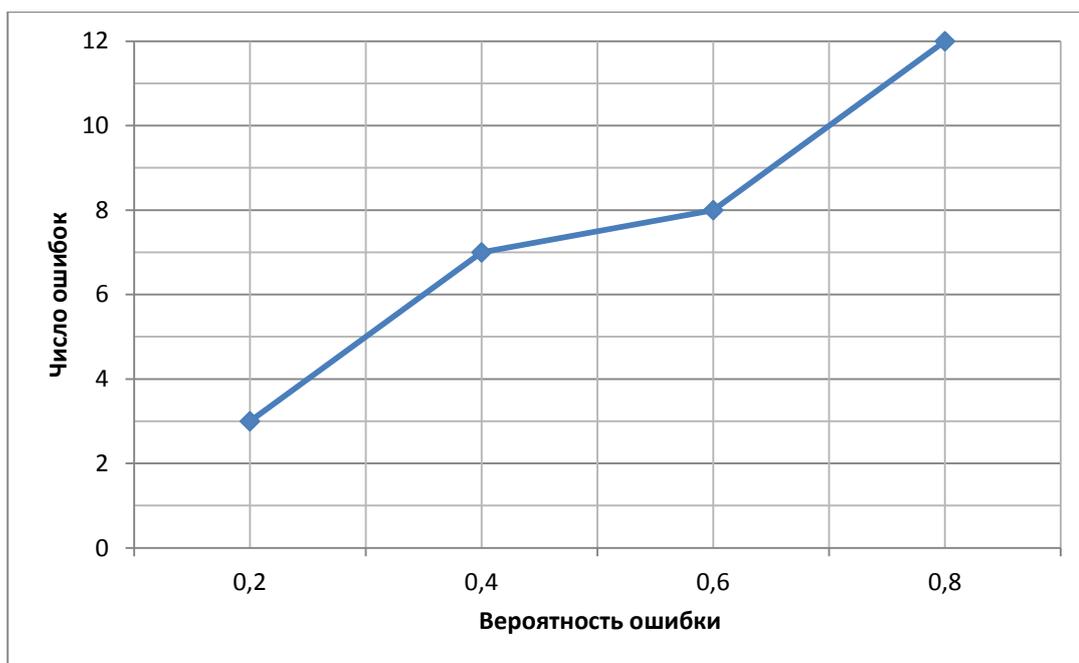


Рис. 3.38. График зависимости числа ошибок от вероятности ошибки для кода БЧХ (15,7)

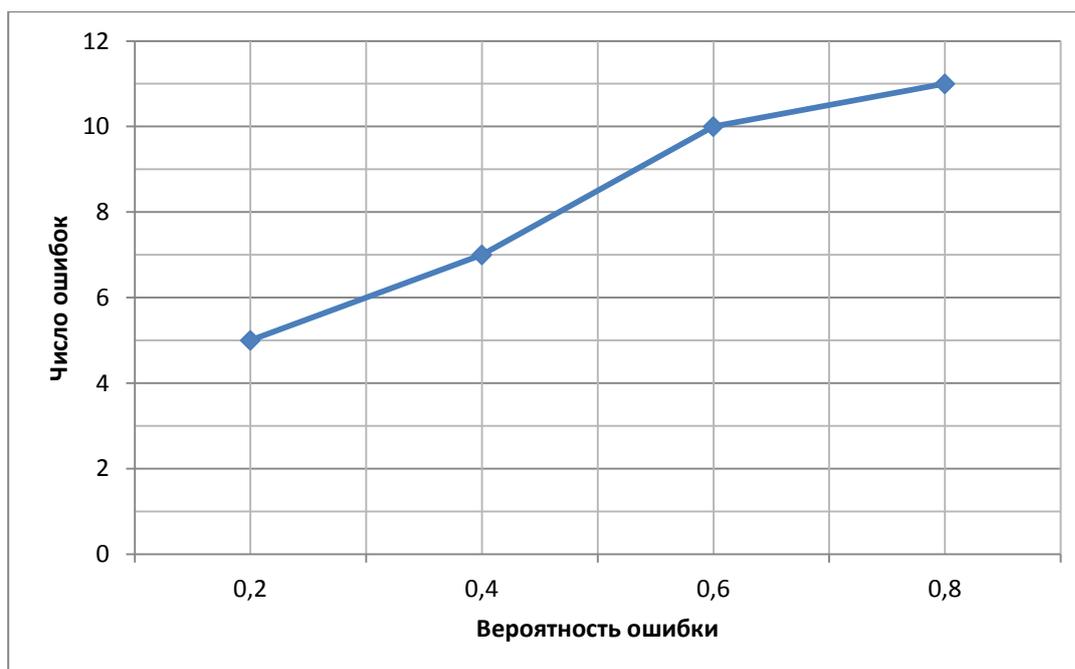


Рис. 3.39. График зависимости числа ошибок от вероятности ошибки для кода БЧХ (15,11)

В результате практической работы построена схема линии передачи с кодированием БЧХ в среде Simulink. Построены графики зависимостей числа ошибок на выходе декодера от вероятности ошибки в канале связи для кодов (15,7) и (15,11). Из графиков (рисунок 3.38 и

рисунок 3.39) видно, что число ошибок увеличивается с ростом вероятности ошибок. При этом код (15,7) оказался лучше по способности обнаружения ошибок.

### Код Рида-Соломона

Кодировщик Рида-Соломона берет блок цифровых данных и добавляет дополнительные "избыточные" биты. Ошибки происходят при передаче по каналам связи или по разным причинам при запоминании (например, из-за шума или наводок, царапин на CD и т.д.). Декодер Рида-Соломона обрабатывает каждый блок, пытается исправить ошибки и восстановить исходные данные. Число и типы ошибок, которые могут быть исправлены, зависят от характеристик кода Рида-Соломона.

#### Свойства кодов Рида-Соломона

Коды Рида-Соломона являются подмножеством кодов БЧХ и представляют собой линейные блочные коды. Код Рида-Соломона специфицируется как RS (n,k) s - битных символов.

Это означает, что кодировщик воспринимает  $k$  информационных символов по  $s$  битов каждый и добавляет символы четности для формирования  $n$  символьного кодового слова. Декодер Рида-Соломона может корректировать до  $t$  символов, которые содержат ошибки в кодовом слове, где  $2t = n - k$ .

Диаграмма, представленная ниже, показывает типовое кодовое слово Рида-Соломона:

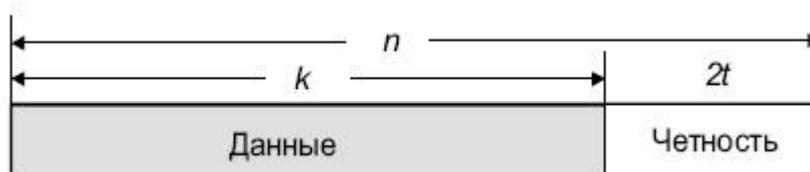


Рис. 3.40. Кодовое слово для кода Р-С

Популярным кодом Рида-Соломона является RS (255, 223) с 8-битными символами. Каждое кодовое слово содержит 255 байт, из которых 223 являются информационными и 32 байтами четности. Для этого кода  $n = 255$ ,  $k = 223$ ,  $s = 8$ ,  $2t = 32$ ,  $t = 16$ .

Декодер может исправить любые 16 символов с ошибками в кодовом слове: то есть ошибки могут быть исправлены, если число искаженных байт не превышает 16.

При размере символа  $s$ , максимальная длина кодового слова  $n$  для кода Рида-Соломона равна  $n = 2^s - 1$ .

Например, максимальная длина кода с 8-битными символами ( $s=8$ ) равна 255 байтам.

Коды Рида-Соломона могут быть в принципе укорочены путем обнуления некоторого числа информационных символов на входе кодировщика (передать их в этом случае не нужно). При передаче данных декодеру эти нули снова вводятся в массив.

Код (255, 223), описанный выше, может быть укорочен до (200, 168). Кодировщик будет работать с блоком данных 168 байт, добавит 55 нулевых байт, сформирует кодовое слово (255, 223) и передаст только 168 информационных байт и 32 байта четности.

Объем вычислительной мощности, необходимой для кодирования и декодирования кодов Рида-Соломона, зависит от числа символов четности. Большое значение  $t$  означает, что большее число ошибок может быть исправлено, но это потребует большей вычислительной мощности по сравнению с вариантом при меньшем  $t$ .

#### Ошибки в символах

Одна ошибка в символе происходит, когда 1 бит символа оказывается неверным или когда все биты неверны.

Код RS(255,223) может исправить до 16 ошибок в символах. В худшем случае, могут иметь место 16 битовых ошибок в разных символах (байтах). В лучшем случае, корректируются 16 полностью неверных байт, при этом исправляется  $16 * 8 = 128$  битовых ошибок.

Коды Рида-Соломона особенно хорошо подходят для корректировки кластеров ошибок (когда неверными оказываются большие группы бит кодового слова, следующие подряд).

#### Декодирование

Алгебраические процедуры декодирования Рида-Соломона могут исправлять ошибки и потери. Потерей считается случай, когда положение неверного символа известно. Декодер может исправить до  $t$  ошибок или до  $2t$  потерь. Данные о потере (стирании) могут быть получены от демодулятора цифровой коммуникационной системы, т.е. демодулятор помечает полученные символы, которые вероятно содержат ошибки.

Когда кодовое слово декодируется, возможны три варианта.

1. Если  $2s + r < 2t$  ( $s$  ошибок,  $r$  потеря), тогда исходное переданное кодовое слово всегда будет восстановлено. В противном случае:
2. Декодер детектирует ситуацию, когда он не может восстановить исходное кодовое слово. Или же:
3. Декодер некорректно декодирует и неверно восстановит кодовое слово без какого-либо указания на этот факт.

Вероятность каждого из этих вариантов зависит от типа используемого кода Рида-Соломона, а также от числа и распределения ошибок.

#### Архитектура кодирования и декодирования кодов Рида-Соломона

Кодирование и декодирование Рида-Соломона может быть выполнено аппаратно или программно.

### Арифметика конечного поля Галуа

Коды Рида-Соломона базируются на специальном разделе математики – полях Галуа (GF) или конечных полях. Арифметические действия (+, -, x, / и т.д.) над элементами конечного поля дают результат, который также является элементом этого поля. Кодировщик или декодер Рида-Соломона должны уметь выполнять эти арифметические операции. Эти операции для своей реализации требуют специального оборудования или специализированного программного обеспечения.

#### Образующий полином

Кодовое слово Рида-Соломона формируется с привлечением специального полинома. Все корректные кодовые слова должны делиться без остатка на эти образующие полиномы. Общая форма образующего полинома имеет вид

$$g(x) = (x - a^1)(x - a^{i+1}) \dots (x - a^{i+1+2t})$$

а кодовое слово формируется с помощью операции

$$c(x) = g(x) * i(x)$$

где  $g(x)$  является образующим полиномом,  $i(x)$  представляет собой информационный блок,  $c(x)$  – кодовое слово, называемое простым элементом поля.

#### Архитектура кодировщика

$2t$  символов четности в кодовом слове Рида-Соломона определяются из следующего соотношения:

$$p(x) = i(x) * x^{n-k} \bmod g(x)$$

Ниже показана схема реализации кодировщика для версии RS(255,249):

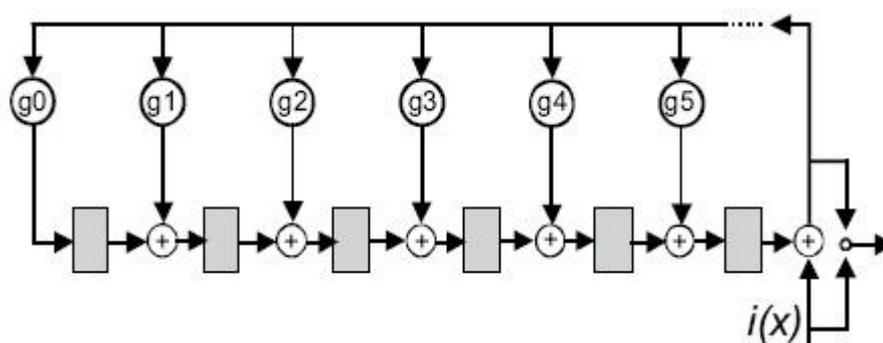


Рис. 3. 41. Схема кодировщика P-C

Каждый из 6 регистров содержит в себе символ (8 бит). Арифметические операторы выполняют сложение или умножение на символ как на элемент конечного поля.

#### Архитектура декодера

Общая схема декодирования кодов Рида-Соломона показана ниже на рис. 3.42.

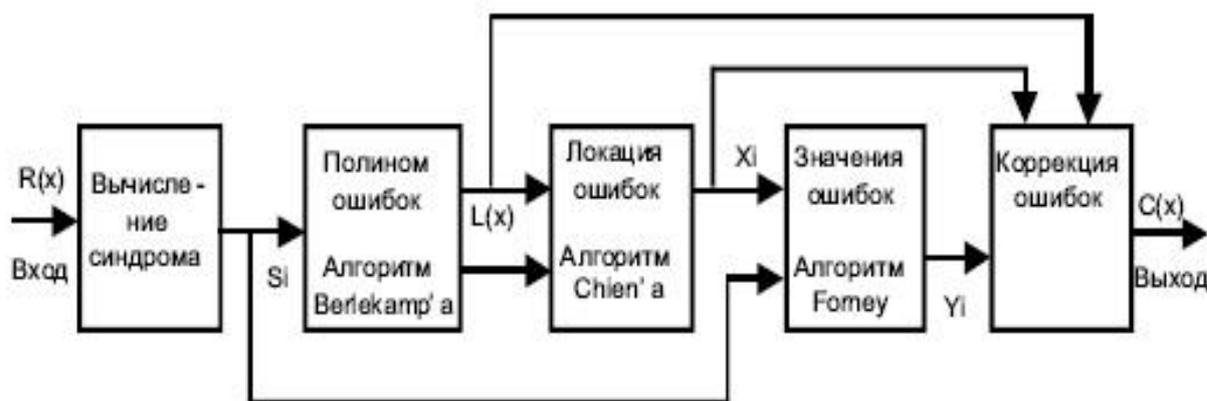


Рис. 3.42. Схема работы с кодами Рида-Соломона

Обозначения:

- $r(x)$  – Полученное кодовое слово
- $S_i$  – Синдромы
- $L(x)$  – Полином локации ошибок
- $X_i$  – Положения ошибок
- $Y_i$  – Значения ошибок
- $c(x)$  – Восстановленное кодовое слово
- $v$  – Число ошибок

Полученное кодовое слово  $r(x)$  представляет собой исходное (переданное) кодовое слово  $c(x)$  плюс ошибки:

$$r(x) = c(x) + e(x)$$

Декодер Рида-Соломона пытается определить позицию и значение ошибки для  $t$  ошибок (или  $2t$  потерь) и исправить ошибки и потери.

**Вычисление синдрома**

Вычисление синдрома похоже на вычисление четности. Кодовое слово Рида-Соломона имеет  $2t$  синдромов, это зависит только от ошибок (а не передаваемых кодовых слов). Синдромы могут быть вычислены путем подстановки  $2t$  корней образующего полинома  $g(x)$  в  $r(x)$ .

**Нахождение позиций символьных ошибок**

Это делается путем решения системы уравнений с  $t$  неизвестными. Существует несколько быстрых алгоритмов для решения этой задачи. Эти алгоритмы используют особенности структуры матрицы кодов Рида-Соломона и сильно сокращают необходимую вычислительную мощность. Делается это в два этапа.

1. Определение полинома локации ошибок.

Это может быть сделано с помощью алгоритма Berlekamp-Massey или алгоритма Эвклида. Алгоритм Эвклида используется чаще на практике, так как его легче реализовать,

однако алгоритм Berlekamp-Massey позволяет получить более эффективную реализацию оборудования и программ.

2. Нахождение корней этого полинома. Это делается с привлечением алгоритма поиска Chien.

Нахождение значений символьных ошибок

Здесь также нужно решить систему уравнений с  $t$  неизвестными. Для решения используется быстрый алгоритм Forney.

Реализация кодировщика и декодера Рида-Соломона. Аппаратная реализация

Существует несколько коммерческих аппаратных реализаций. Имеется много разработанных интегральных схем, предназначенных для кодирования и декодирования кодов Рида-Соломона. Эти ИС допускают определенный уровень программирования (например, RS (255, k), где  $t$  может принимать значения от 1 до 16).

Программная реализация

До недавнего времени программные реализации в "реальном времени" требовали слишком большой вычислительной мощности практически для всех кодов Рида-Соломона. Главной трудностью в программной реализации кодов Рида-Соломона являлось то, что процессоры общего назначения не поддерживают арифметические операции для поля Галуа. Однако оптимальное составление программ в сочетании с возросшей вычислительной мощностью позволяют получить вполне приемлемые результаты для относительно высоких скоростей передачи данных.

### **Практическая часть исследования**

На данном этапе необходимо реализовать и проанализировать описанные выше алгоритмы кодирования на практике в программной среде Matlab 15. Необходимо построить соответствующие схемы, которые будут в себя включать источник сообщения, кодер, декодер, канал передачи, анализатор ошибок и соответственно устройства визуализации полученных результатов. Так же необходимо сравнить ошибки передачи с линией, которая не будет никак преобразовывать передаваемую информацию, а просто передавать ее по каналу связи и при увеличении вероятности ошибки в канале до 0,3. Коды, которые подлежат реализации это: код Хэмминга (7,4), код BCH (15,7), и код Рида-Соломона (255,223). Всю задачу выполнить в среде Matlab 15. Представить проделанную работу в отчете.

Для начала необходимо запустить саму программу. Происходит это следующим образом:

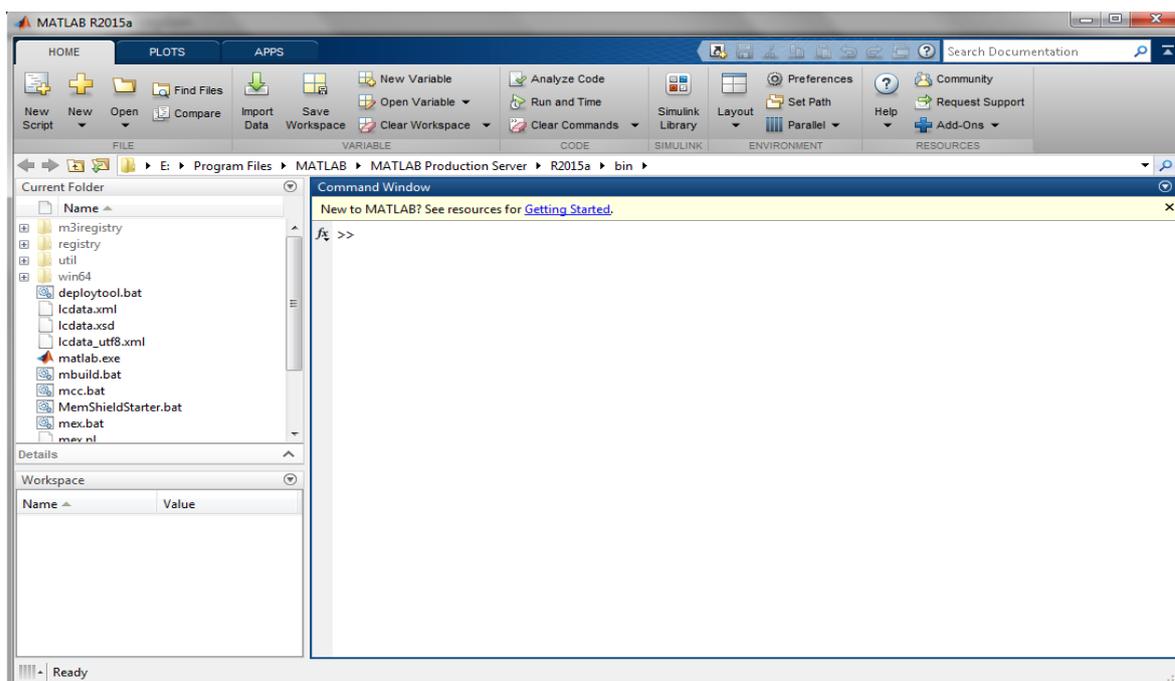


Рис. 3.43. Окно Matlab 15

Далее необходимо нажать на вкладку «Simulink Library», которая находится на панели инструментов в открывшемся окне. Откроется следующее окно:

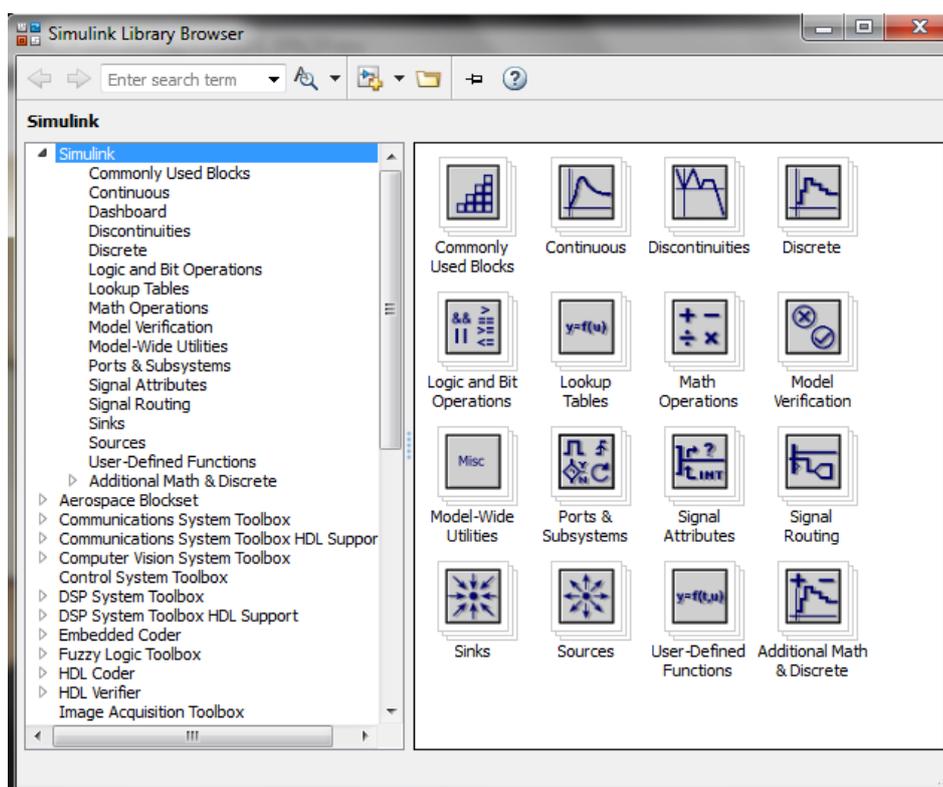


Рис. 3.44. Окно «Simulink Library»

Далее создаем новую модель (вкладка на панели инструментов) и приступаем к реализации описанных выше алгоритмов.

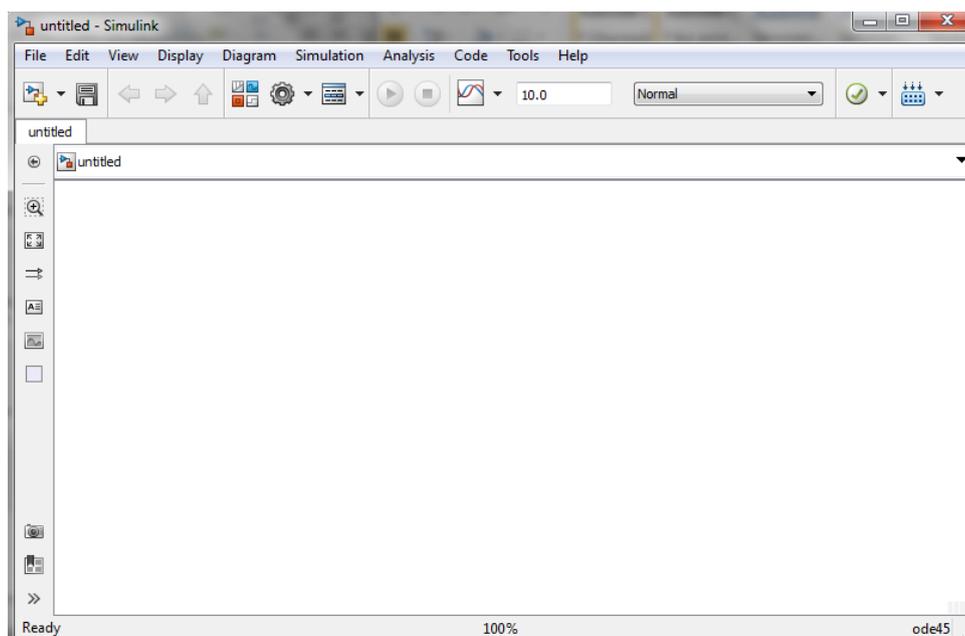


Рис. 3.45. Рабочее поле

Окно, представленное на рисунке 3.44, необходимо для поиска необходимых компонентов для реализации данной работы (кодеры, декодеры и т.д.)

Реализуем канал передачи такой же кодовой комбинации без преобразований.

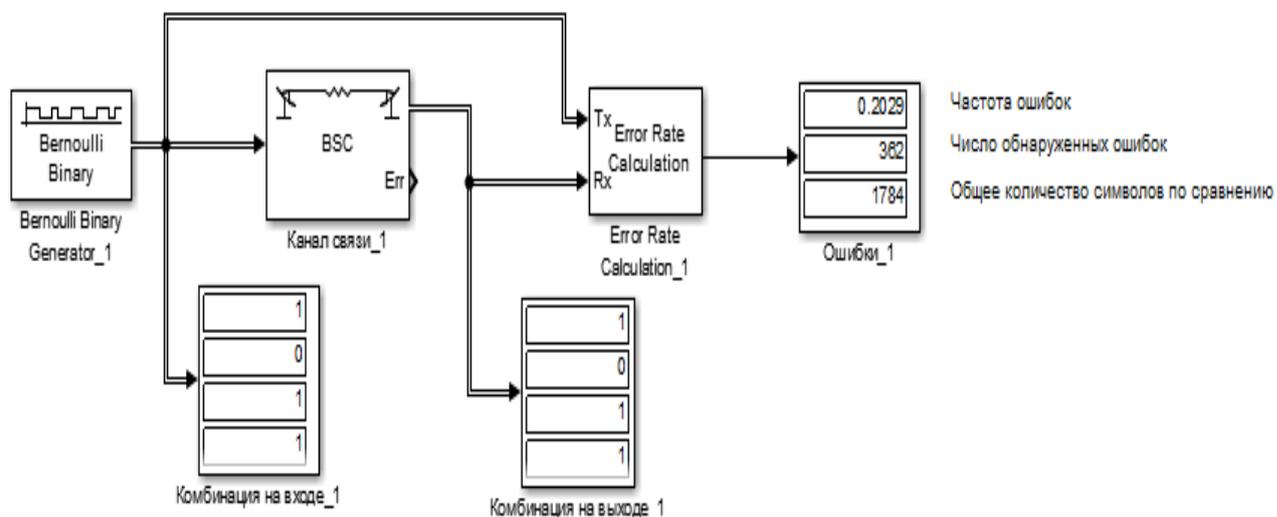


Рис. 3.46. Линия передачи без преобразований

В состав линии без преобразований входят:

1. BernoulliBinaryGenerator
2. Binary Symmetric Channel (каналпередачи)
3. Error RateCalculation (анализаторошибок)
4. Display

Характеристики блоков выставить следующие:

**Bernoulli Binary Generator**

Generate a Bernoulli random binary number.  
To generate a vector output, specify the probability as a vector.

**Parameters**

Probability of a zero:

Initial seed:

Sample time:

Frame-based outputs

Samples per frame:

Output data type:

Рис. 3.47. Bernoulli Binary Generator

**Binary Symmetric Channel (mask) (link)**

Add binary errors to the input signal.

**Parameters**

Error probability:

Initial seed:

Output error vector

Output data type:

Рис. 3.48. Binary Symmetric Channel

**Error Rate Calculation (mask) (link)**

Compute the error rate of the received data by comparing it to a delayed version of the transmitted data. The block output is a three-element vector consisting of the error rate, followed by the number of errors detected and the total number of symbols compared. This vector can be sent to either the workspace or an output port.

The delays are specified in number of samples, regardless of whether the input is a scalar or a vector. The inputs to the 'Tx' and 'Rx' ports must be scalars or column vectors.

The 'Stop simulation' option stops the simulation upon detecting a target number of errors or a maximum number of symbols, whichever comes first.

**Parameters**

Receive delay:

Computation delay:

Computation mode:

Output data:

Рис. 3.49. ErrorRateCalculation (анализаторошибок)

Представим полученные результаты:

Входная последовательность:

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

Комбинация на входе\_1

Рис. 3.50. Входная последовательность

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

Комбинация на выходе\_1

Рис. 3.51. Выходная последовательность

0.2029	Частота ошибок
362	Число обнаруженных ошибок
1784	Общее количество символов по сравнению

Ошибки\_1

Рис. 3.52. Счетчик ошибки

Анализируя рисунок выше, можно сделать вывод, что комбинация на входе совпадает с комбинацией на выходе, таким образом, передача осуществилась удачно. Что касается ошибок, то их частота равна 0,2029, число обнаруженных ошибок равно 362, общее

количество символов по сравнению равно 1784. В данной схеме осуществляется передача информации без преобразований. Если сравнивать этот результат с результатом с кодированием при вероятности ошибки 0,2, то они идентичны. При увеличении вероятности ошибки до 0,3 в декодированной последовательности были найдены ошибки.

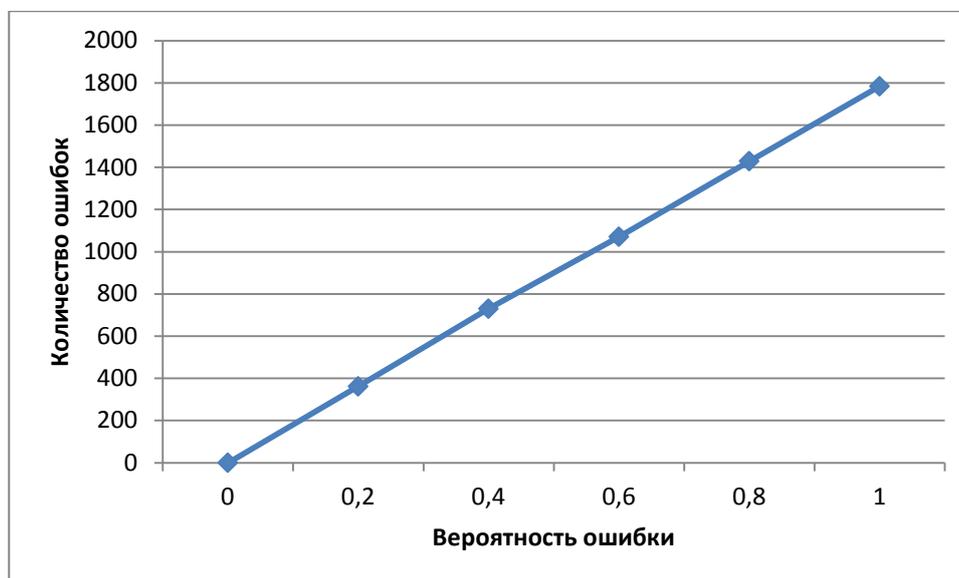


Рис. 3. 53. График зависимости числа обнаруженных ошибок от вероятности ошибки для кода (255,223)

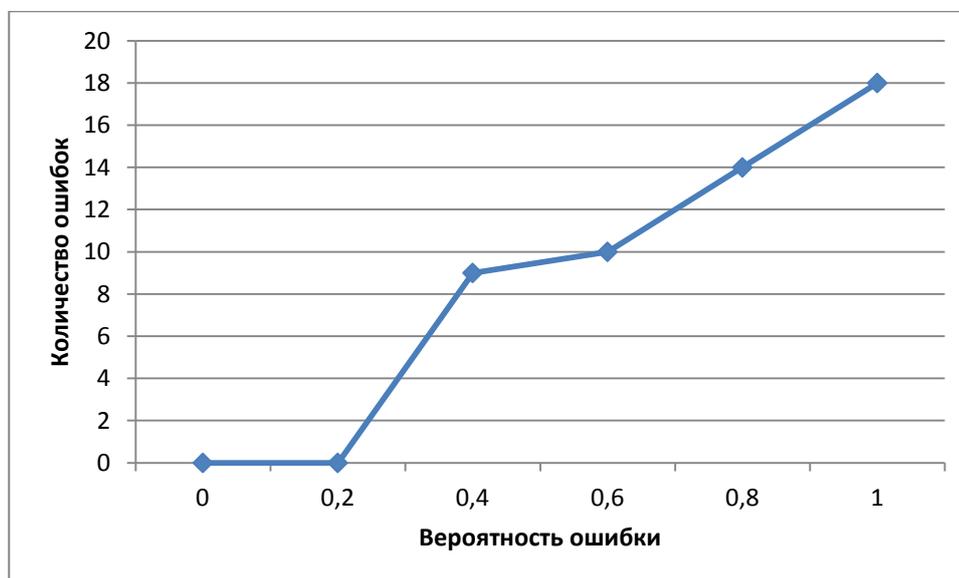


Рис. 3.54. График зависимости числа обнаруженных ошибок от вероятности ошибки для кода (7,3)

Были рассмотрены особенности кода Рида-Соломона. Построили графики зависимости числа обнаруженных ошибок от вероятности ошибки для кодов (255,223) и (7,3).

Изучена структура кодера и декодера.

1. Код Рида-Соломона — код с 8-битными символами и проверкой на чётность. В коде (255,223) 223 байта информационных символов, 32 байта проверки на четность.

2. Объем вычислительной мощности, необходимой для кодирования и декодирования кодов Рида-Соломона, зависит от числа символов четности. Большое значение  $t$  означает, что большее число ошибок может быть исправлено, но это потребует большей вычислительной мощности по сравнению с вариантом при меньшем  $t$ .

3. Код RS(255,223) может исправить до 16 ошибок в символах. В худшем случае, могут иметь место 16 битовых ошибок в разных символах (байтах). В лучшем случае, корректируются 16 полностью неверных байт, при этом исправляется  $16 * 8 = 128$  битовых ошибок.

4. В результате анализа графиков рисунков 2.19-2.20 можно сделать вывод, что код с длинными последовательностями исправляет больше ошибок.

В разделе начало взяло аналитика выбора тех или иных параметров кодов, таких как, выбор размерности кодов, вероятность ошибки в канале и т.д. Сначала были выполнен предварительный анализ, позволившие выбрать и обосновать структурные схемы кодов по исходным данным технического задания.

Путем проведения компьютерной симуляции, была проверена достоверность расчетов. В программе Matlab 15 были собраны схемы:

- 1) Кода Хэмминга
- 2) Кода БЧХ
- 3) Кода Рида-Соломона

С помощью компьютерной симуляции была дана оценка ошибкам при передаче информации по каналу.

Также я познакомился с различным программным обеспечением, для построения различного вида схем.

Подводя итог своего индивидуального задания, можно сказать следующее, то, что я сделал, является основополагающим делом к дальнейшим, более трудным вещам. Индивидуальная работа была весьма увлекательна и полезна. С поставленными целями справился успешно.

Таблица 3.1. Результаты

	Вер-ть ошибки 0,2	Вер-ть ошибки 0,3
Код	Частота ошибок: 0,25	Частота ошибок: 0,4167
Хэмминга	Число обнаруженных ошибок:	Число обнаруженных ошибок: 5

(7,4)	3 Общее кол-во символов по сравнению: 12	Общее кол-во символов по сравнению: 12
Код BCH (15,7)	Частота ошибок: 0,2143 Число обнаруженных ошибок: 3 Общее кол-во символов по сравнению: 14	Частота ошибок: 0,4286 Число обнаруженных ошибок: 6 Общее кол-во символов по сравнению: 14
Код P-C (255,223)	Частота ошибок: 0,2029 Число обнаруженных ошибок: 362 Общее кол-во символов по сравнению: 1784	Частота ошибок: 0,31 Число обнаруженных ошибок: 553 Общее кол-во символов по сравнению: 1784

### 3.2. Циклические избыточные коды CRC (Cyclic redundancy check) [16]

**Циклический избыточный код (CRC).** Наиболее известными из методов обнаружения ошибок передачи данных являются [1]:

- *Посимвольный контроль чётности*, называемый также поперечным, подразумевает передачу с каждым байтом дополнительного бита, принимающего единичное значение по чётному или нечётному количеству единичных бит в контролируемом байте. Посимвольный контроль чётности прост как в программной, так и в аппаратной реализации, но его вряд ли можно назвать эффективным методом обнаружения ошибок, так как искажение более одного бита исходной последовательности резко снижает вероятность обнаружения ошибки передачи. Этот вид контроля обычно реализуется аппаратно в устройствах связи.

- *Поблочный контроль чётности*, называемый продольным. Схема данного контроля подразумевает, что для источника и приёмника информации заранее известно, какое число передаваемых символов будет рассматриваться ими как единый блок данных. В этой схеме контроля для каждой позиции разрядов в символах блока (поперёк блока) рассчитываются свои биты чётности, которые добавляются в виде обычного символа в конце блока. По сравнению с посимвольным контролем чётности, поблочный контроль чётности обладает большими возможностями по обнаружению и даже корректировке ошибок передачи, но всё равно ему не удаётся обнаруживать определённые типы ошибок.

- *Вычисление контрольных сумм.* В отличие от предыдущих методов, для метода контрольных сумм нет чёткого определения алгоритма. Каждый разработчик трактует понятие контрольной суммы по-своему. В простейшем виде контрольная сумма – это арифметическая сумма двоичных значений контролируемого блока символов. Но этот метод обладает практически теми же недостатками, что и предыдущие, самый главный из которых – нечувствительность контрольной суммы к чётному числу ошибок в одной колонке и самому порядку следования символов в блоке.

- *Контроль циклически избыточным кодом – CRC.* Это гораздо более мощный и широко используемый метод обнаружения ошибок передачи информации. Он обеспечивает обнаружение ошибок с высокой вероятностью. Кроме того, этот метод обладает рядом других полезных моментов, которые могут найти своё воплощение в практических задачах.

Циклический избыточный код (англ. Cyclic redundancy code, CRC) – алгоритм вычисления контрольной суммы, предназначенный для проверки целостности передаваемых данных. Алгоритм CRC обнаруживает все одиночные ошибки, двойные ошибки и ошибки в нечётном числе бит. Понятие циклических кодов достаточно широкое, однако на практике его обычно используют для обозначения только одной разновидности, использующей циклический контроль (проверку) избыточности.

Главная особенность значения CRC состоит в том, что оно однозначно идентифицирует исходную битовую последовательность и поэтому используется в различных протоколах связи, а также для проверки целостности блоков данных, передаваемых различными устройствами. Благодаря относительной простоте алгоритм вычисления CRC часто реализуется на аппаратном уровне.

При передаче пакетов по сетевому каналу могут возникнуть искажения исходной информации вследствие разных внешних воздействий: электрических наводок, плохих погодных условий и многих других. Сущность методики в том, что при хороших характеристиках контрольной суммы в подавляющем числе случаев ошибка в сообщении приведёт к изменению его контрольной суммы. Если исходная и вычисленная суммы не равны между собой, принимается решение о недостоверности принятых данных, и можно запросить повторную передачу пакета.

Основная идея алгоритма CRC состоит в представлении сообщения в виде огромного двоичного числа, делении его на другое фиксированное двоичное число и использовании остатка этого деления в качестве контрольной суммы [2]. Получив сообщение, приёмник может выполнить аналогичное действие и сравнить полученный остаток с «контрольной суммой».

На рисунках 2.1-2.2 изображено графическое представление кодирования и декодирования CRC-кода [3]:

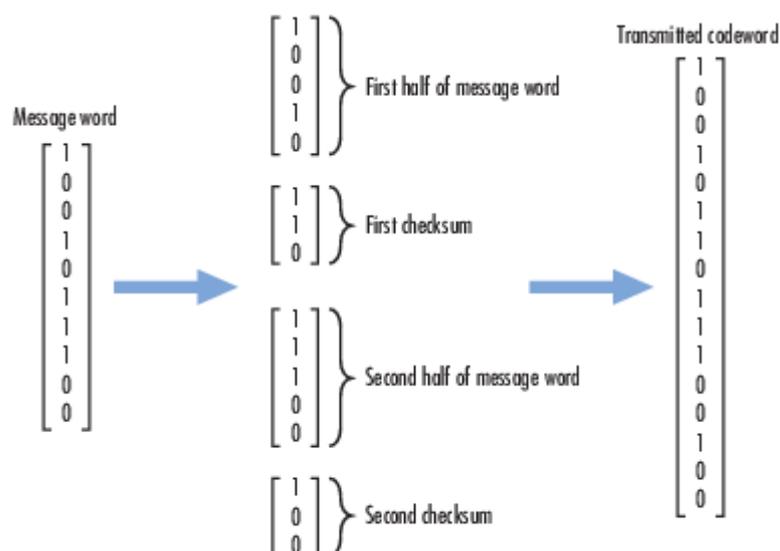


Рис. 3.55. Принцип работы кодера CRC

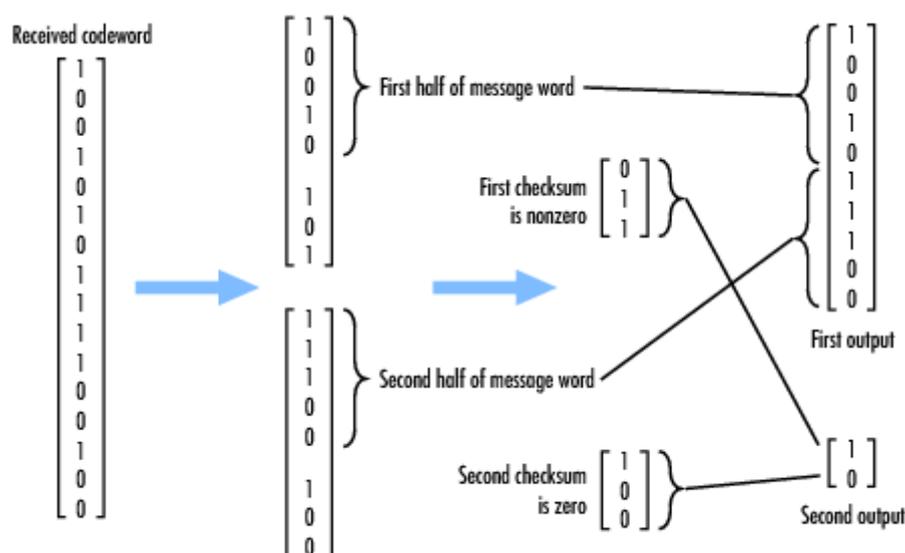


Рис. 3.56. Принцип работы декодера CRC

Степень CRC-полинома  $W$  называют позицию самого старшего единичного бита. Например, степень полинома  $10011_2$  равна 4.

Для вычисления CRC используют полиномиальную арифметику. Вместо представления делителя, делимого (сообщения), частного и остатка в виде положительных целых чисел, можно представить их в виде полиномов с двоичными коэффициентами или в виде строки бит, каждый из которых является коэффициентом полинома.

Например, десятичное число 23 в 16-ричной и 2-ичной системах будет иметь вид  $23_{10}=17_{16}=10111_2$ , что совпадает с полиномом:  $1 \cdot x^4 + 0 \cdot x^3 + 1 \cdot x^2 + 1 \cdot x^1 + 1 \cdot x^0$  или упрощённо:  $x^4 + x^2 + x^1 + 1$ .

И сообщение, и делитель могут быть представлены в виде полиномов, с которыми можно выполнять любые арифметические действия без переносов.

Как правило, контрольная сумма добавляется к исходному сообщению и полученное расширенное сообщение передаётся через канал связи.

На другом конце канала приёмник может сделать одно из возможных действий (оба варианта совершенно равноправны):

1. Выделить текст полученного сообщения, вычислить для него контрольную сумму и сравнить её с переданной.
2. Вычислить контрольную сумму для всего переданного сообщения, и посмотреть, получится ли в результате нулевой остаток.

Поскольку исходное сообщение может быть очень большим (до нескольких Мбайтов) и так же из-за того, что для получения CRC используется CRC-арифметика, использовать обычную компьютерную операцию деления нельзя.

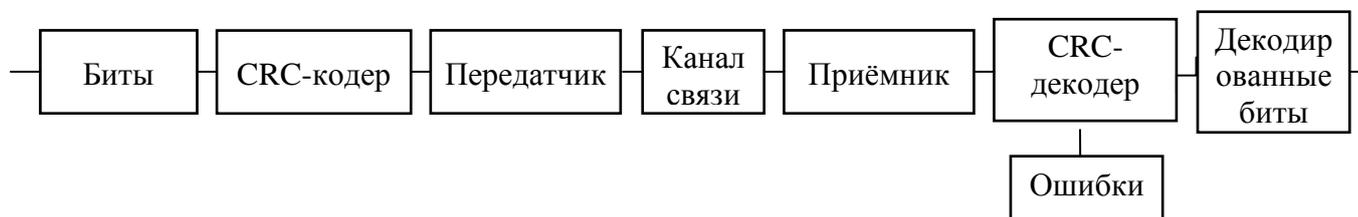


Рис. 3.57. Обобщенная структурная схема исследования CRC кодов

Самый популярный и рекомендуемый IEEE полином для CRC-32 используется в Ethernet, FDDI; также этот многочлен является генератором кода Хемминга. Использование другого полинома — CRC-32C — позволяет достичь такой же производительности при длине исходного сообщения от 58 бит до 131 кбит, а в некоторых диапазонах длины входного сообщения может быть даже выше — поэтому в наши дни он тоже пользуется популярностью. К примеру, стандарт ITU-T использует CRC-32C с целью обнаружения ошибок в полезной нагрузке.

Ниже в таблице перечислены наиболее распространённые многочлены — генераторы CRC [4]:

Таблица 3.2. Распространённые полиномы CRC кодов

Название	Полином
CRC-1	$x+1$ (используется в аппаратном контроле ошибок, также известен как бит чётности)
CRC-4-ITU	$x^4+x+1$
CRC-5-ITU	$x^5+x^4+x^2+1$
CRC-5-USB	$x^5+x^2+1$
CRC-6-ITU	$x^6+x+1$
CRC-7	$x^7+x^3+1$ (системы телекоммуникации, ITU-T G.707, ITU-T G.832, MMC, SD)
CRC-8	$x^8+x^7+x^6+x^4+x^2+1$
CRC-16-IBM	$x^{16}+x^{15}+x^2+1$ (Bisync, Modbus, USB, ANSI X3.28)
CRC-16-CCITT	$x^{16}+x^{12}+x^5+1$ (X.25, HDLC, XMODEM, Bluetooth, SD)
CRC-30	$x^{30}+x^{29}+x^{21}+x^{20}+x^{15}+x^{13}+x^{12}+x^{11}+x^8+x^7+x^6+x^2+x+1$ (CDMA)

## Программная реализация виртуальных моделей кодирования

### Описание реализации циклического избыточного кода (CRC)

Виртуальная модель передачи данных с обнаружением ошибок при помощи CRC-кода была реализована в среде Simulink Matlab. Модель демонстрирует работу CRC-кодера и декодера, позволяет исследовать обнаруживающую способность кода для разных генераторных полиномов.

На рисунке 3.58 приведена разработанная модель:

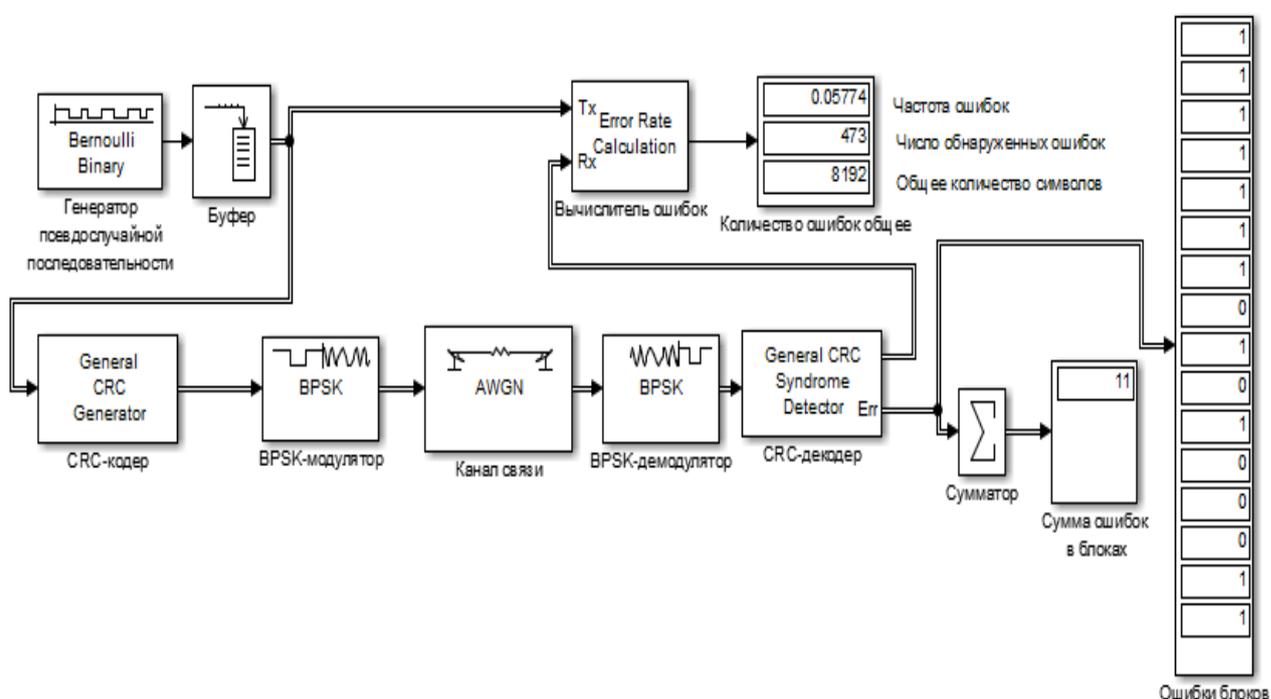


Рис. 3.58. Разработанная модель исследования CRC-кодов

В её основу положены следующие элементы, встроенные в библиотеку Simulink:

- Bernoulli Binary Generator;
- General CRC Generator;
- BPSK Modulator Baseband;
- AWGN Channel;
- BPSK Demodulator Baseband;
- General CRC Syndrome Detector;
- Error Rate Calculation;
- Buffer;
- Add;
- Display (Дисплей, отражающий ошибки).

Далее представлено описание основных блоков:

Bernoulli Binary Generator (генератор псевдослучайной последовательности) – генерирует случайную бинарную последовательность (рисунок 3.59).

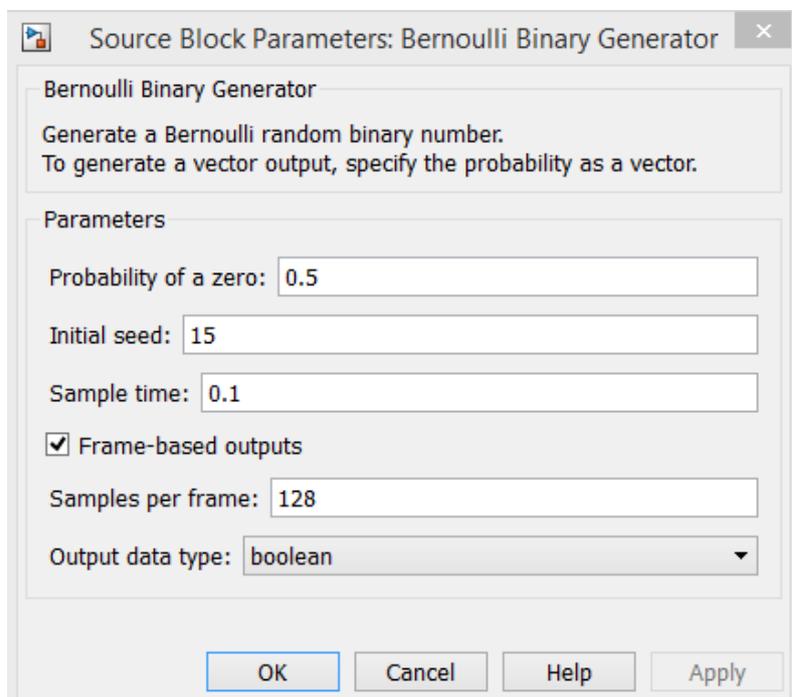


Рис. 3.59. Параметры блока «Bernoulli Binary Generator»

«Probability of a zero» - вероятность появления нуля;

«Initial seed» - начальное значение для генерации;

«Sample time» - длительность сэмпла;

«Samples per frame» - размер фрейма.

General CRC Generator (CRC-кодер) – циклический избыточный кодер (рисунок 3.3).

«Generator polynomial» - генераторный полином, может быть задан в 3 формах:

1) В обычной записи, например:  $x^3 + x^2 + x + 1$ .

2) в виде матрицы-строки с указанием степеней с ненулевыми коэффициентами, например:  $[4 \ 1 \ 0] = x^4 + x + 1$ .

3) в виде матрицы-строки с указанием нулевых и ненулевых коэффициентов, например:  $[1 \ 1 \ 0 \ 1 \ 1] = x^4 + x^3 + x + 1$ .

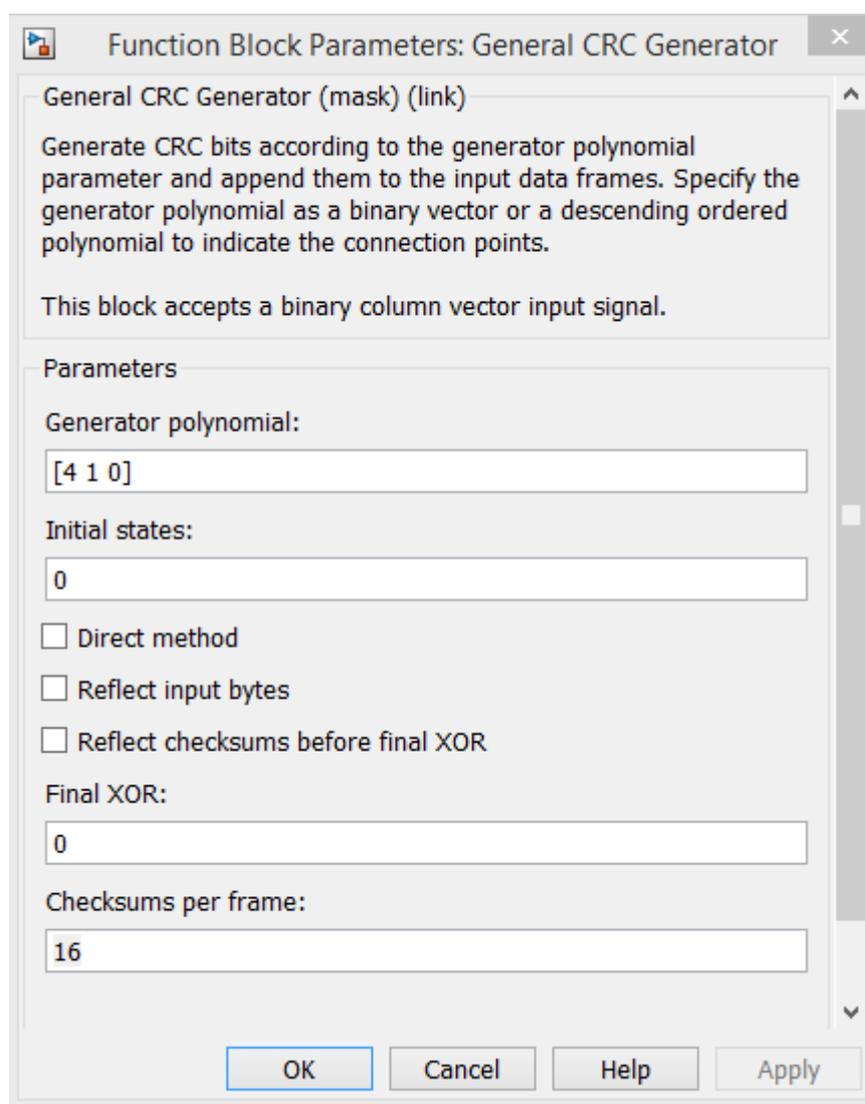


Рис. 3.60. Параметры блока «General CRC Generator»

«Initial states» - начальное состояние сдвиговых регистров.

«Direct method» - включение прямого метода вычисления CRC, иначе работает по табличному методу.

«Reflect input bytes» - инвертировать входной поток.

«Reflect checksums before final XOR» - инвертировать контрольные суммы перед конечной операцией XOR.

«Final XOR» - Выполнить операцию XOR в конце кодирования.

«Checksums per frame» - количество контрольных сумм во фрейме.

BPSK Modulator Baseband – BPSK модулятор.

BPSK Demodulator Baseband – BPSK демодулятор.

AWGN Channel (Канал связи) – добавляет «белый» гауссовский шум в канале (рисунок 3.61).

«SNR» - задаёт отношение сигнал/шум в канале.

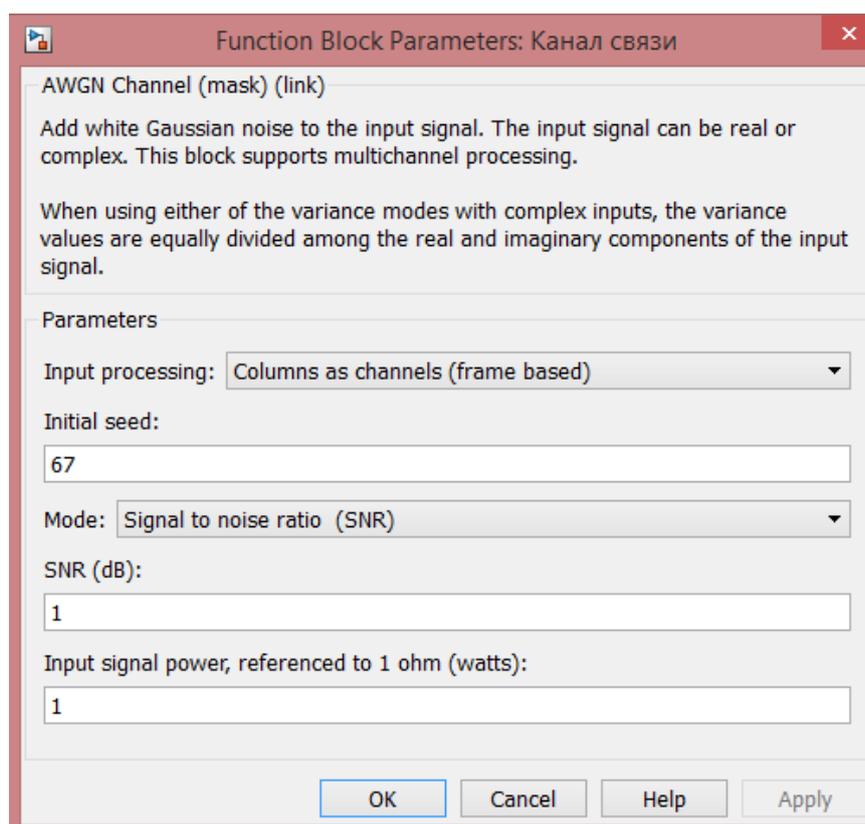


Рис. 3.61. Параметры блока «AWGN»

General CRC Syndrome Detector - циклический избыточный декодер. Все параметры декодера задаются аналогично параметрам блока «General CRC Generator» (рисунок 3.3).

Error Rate Calculation – вычислитель ошибок между переданной и принятой последовательностью.

Buffer – буфер. Переводит последовательность бит в один блок.

Add (сумматор) – суммирует ошибки от CRC-декодера.

Display - дисплей, отражающий ошибки.

### Результаты моделирования

#### Исследование циклического избыточного кода

*Модель циклического избыточного кода (срс), разработанная представленная на рисунке 3.58, позволяет исследовать обнаруживающую способность CRC кодов с различными полиномами.*

Задаём одинаковый генераторный полином в блоки CRC-кодер и CRC-декодер (рисунок 3.62):

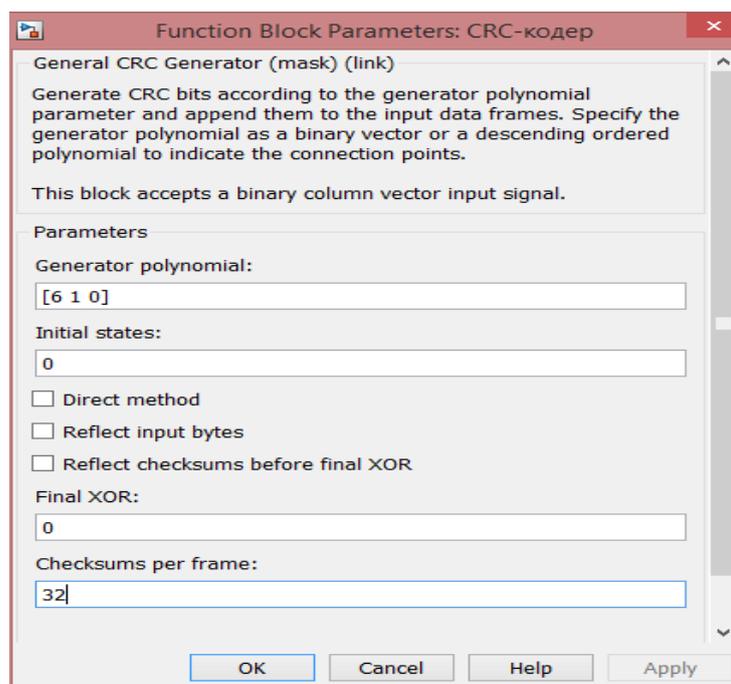


Рис. 3.62. Параметры блока CRC-кодер

Общее число передаваемых символов составляет 8192. Количество контрольных сумм изменяется от 2 до 8192, с увеличением каждого предыдущего значения в 2 раза (2, 4, 8, 16...8192).

Значение SNR в блоке «Канал связи» установлено в 1 дБ. Таким образом, битовая вероятность ошибки (BER) составит 0,05786.

На рисунке 4.2 представлен график зависимости числа обнаруженных ошибок от числа контрольных сумм для различных полиномов CRC-кода.

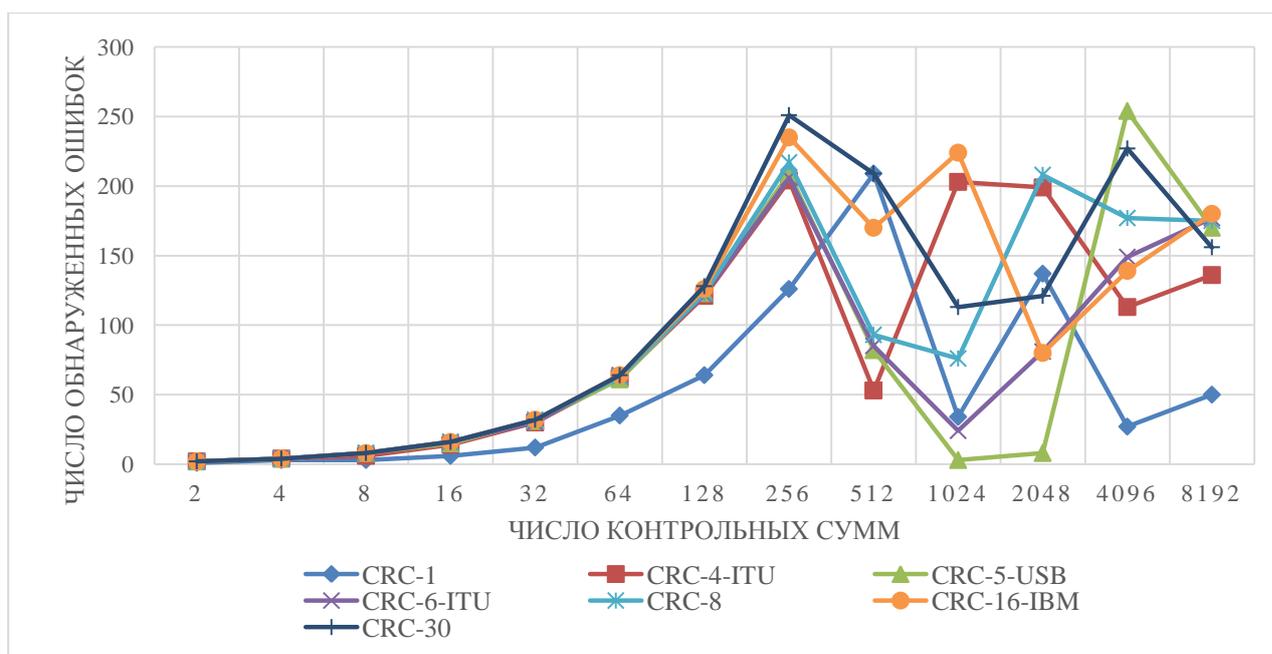


Рис. 3.63. График зависимости числа обнаруженных ошибок от числа контрольных сумм для различных полиномов CRC-кода

В данном разделе проведено исследование модели циклического избыточного кода (CRC).

Модель позволяет исследовать CRC-коды с возможностью задания любого генераторного полинома и изменении количества контрольных сумм во фрейме.

Получены следующие результаты и сделаны следующие выводы:

- 1) чем выше степень полинома, тем лучше его обнаруживающая способность;
- 2) для каждого полинома есть такое число контрольных сумм в блоке, при котором его обнаруживающая способность максимальна, причём у всех полиномов эти точки различны.

Однако, при выборе полинома CRC-кода также необходимо учитывать и другие факторы:

- 1) увеличение степени полинома приводит к усложнению реализации кодера и декодера;
- 2) чем выше частота вычисления контрольных сумм, т.е. чем больше контрольных сумм добавляется в блок данных, тем меньше пропускная способность канала;
- 3) CRC-коды используют для обнаружения ошибок, что означает наличие канала переспроса. При выборе между кодом CRC/каналом переспроса и помехоустойчивым кодированием, необходимо учитывать характеристики канала связи. При большом числе ошибок передача данных будет невозможна.
- 4) Выбор полинома зависит от размера передаваемого блока данных, чем больше блок – тем выше степень полинома необходимо подбирать. Таким образом, существует ограничение на размер блока данных, иначе в любом блоке на приёмном конце будет обнаруживаться ошибка.

### **3.3. Сверточные коды. Декодирование сверточных кодов**

Современная теория кодов достаточно развита и содержит детальную классификацию. Все применяемые коды можно разбить на две большие группы: блочные, в которых кодирование и декодирование производится в пределах определенного участка кодовой последовательности- блока, и древовидные, в которых обработка символов производится непрерывно, без разделения на блоки. Часть кодов относится к разряду линейных, в которых кодовые последовательности представлены как элементы линейного векторного пространства. Можно применить также разбиение на коды, исправляющие независимые случайные ошибки, и коды, исправляющие пакетные ошибки.

В отличие от блочных, свёрточные коды обладают следующими преимуществами:

- сверточные коды позволяют производить кодирование и декодирование потоков данных непрерывно во времени;
- сверточные коды не нуждаются в блоковой синхронизации;

- применение сверточных кодов позволяет достичь очень высокой надежности передаваемой информации.

Сверточные коды используются при низком отношении сигнал-шум, когда исправляющей способности блочных кодов при разумной длине блока оказывается недостаточно.

Сверточное кодирование, удобнее всего описывать, характеризуя действие соответствующего кодирующего устройства. *Сверточный кодер* представляет собой устройство воспринимающее за каждый такт работы в общем случае  $k$  входных информационных символов и выдающее на выход за тот же такт  $n$  выходных символов, подлежащих передаче по каналу связи. Параметром сверточного кода, характеризующим его помехоустойчивость, является минимальное свободное расстояние —  $d_c$ , определяемое как минимальное расстояние по Хэммингу между последовательностями сверточного кода на длине кодовых ограничений по выходу. Кодовое ограничение по выходу — это число символов на выходе кодера, в формировании которых участвует один входной бит. Эффективность сверточного кода определяется в основном тем, каким образом соединены сумматоры с ячейками регистра сдвига.

Отношение  $R = k/n$  называют относительной скоростью кода. Выходные символы, создаваемые кодером на данном такте, зависят от  $k$  информационных символов, поступивших на этом и предыдущем тактах. Таким образом, выходные символы сверточного кодера однозначно определяются его входным сигналом и состоянием, зависящим от  $m-k$  предыдущих информационных символов.

Основными элементами сверточного кодера являются: регистр сдвига, сумматоры по модулю 2 и коммутатор.

*Регистр сдвига* является динамическим запоминающим устройством (рисунок 3.55), в котором хранятся двоичные символы 0 или 1. Число триггерных ячеек  $m$  в регистре сдвига и определяет память кода. В момент поступления на вход регистра нового информационного символа символ, хранящийся в крайнем правом разряде, выводится из регистра и сбрасывается. Каждый из остальных, хранящихся в регистре символов перемещается на один разряд вправо, освобождая тем самым крайний левый разряд, куда и поступает новый информационный символ. [6]

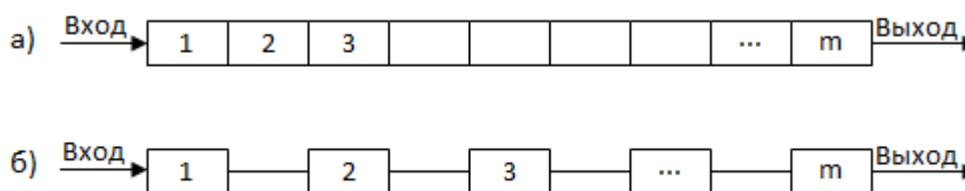


Рис. 3.64. Регистр сдвига

Используются два различных изображения регистров сдвига: с состыкованными впритык ячейками (рисунок 3.64, а) и с непосредственными последовательными связями между ячейками (рисунок 3.64, б), что дает возможность на схемах встраивать между соответствующими ячейками добавочные элементы (схемы суммирования по модулю 2).

*Сумматор по модулю 2* осуществляет сложение поступающих на его входы символов 0 и 1. Правило сложения по модулю 2 следующее: сумма двоичных символов равна 0, если число единиц среди поступающих на входы символов четно, и равна 1, если это число нечетно.

*Коммутатор* осуществляет последовательное считывание поступающих на его входы (контакты) символов и устанавливает на выходе очередность посылки кодовых символов в канал связи.

По аналогии с блоковыми кодами, сверточные коды можно классифицировать на *систематические* и *несистематические*.

*Систематическим* сверточным кодом является такой код, для которого в выходной последовательности кодовых символов содержится без изменения породившая ее последовательность информационных символов. В противном случае сверточный код является несистематическим.

Сверточный код создается прохождением передаваемой информационной последовательности через линейный сдвиговый регистр с конечным числом состояний. В общем, регистр сдвига состоит из  $M$  ( $m$ -битовых) ячеек и линейного преобразователя, состоящего из  $n$  функциональных генераторов и выполняющего алгебраические функции, как показано на рисунке 3.65.

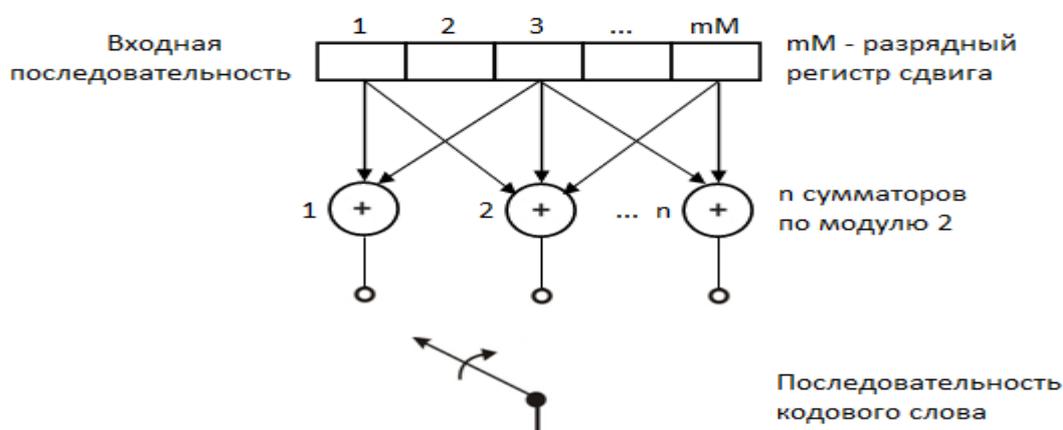


Рис. 3.65. Общий вид сверточного кодера

Входные данные к кодеру, которые считаются двоичными, продвигаются вдоль регистра сдвига по  $k$  бит за раз. Число выходных битов для каждой  $k$ -битовой входной последовательности равно  $n$ . Следовательно, кодовая скорость, определённая как  $R = k/n$ , согласуется с определением скорости блокового кода.

## Представление сверточных кодов

### Графическое представление

Рассмотрим свёрточный кодер со скоростью кода  $1/2$ , его графическое представление показано на рисунке 3.57. В этом кодере каждый раз, информационный бит поступает на вход регистров сдвига, а на выходе генерируется два бита.

В этом кодере каждый раз, информационный бит поступает на вход регистров сдвига, а на выходе генерируется два бита.

В качестве примера рассмотрена ситуация, когда на вход кодера подаётся некая последовательность информационных символов  $V = 1\ 1\ 0\ 1\ 0\ 1$  на выходе имеем последовательность  $U = 11\ 10\ 00\ 10\ 00\ 01$ . Процесс образования выходных символов легко воспроизвести в уме глядя на рисунок.

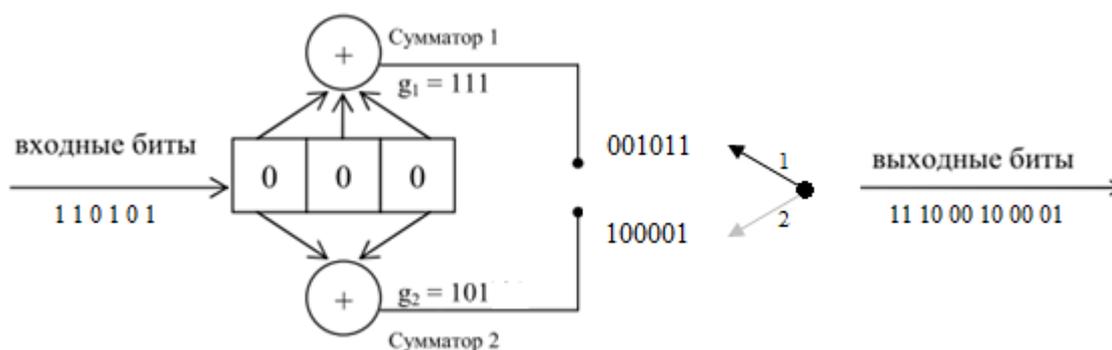


Рис. 3.66. Свёрточный кодер с  $K=3$ ,  $k=1$ ,  $n=2$

Старшие разряды входных информационных бит для удобства располагаются справа и поступают в регистр сдвига первыми. При поступлении на вход регистра одного бита, на выходах сумматора 1 и сумматора 2 образуется по одному биту, которые в дальнейшем поочередно считываются коммутатором и образуют выходную последовательность.

Более подробно весь процесс кодирования информационного потока 00...101 приведен в таблице 3.3.

Таблица 3.3. Процесс кодирования информационного потока

№ такта	Входной информационный бит	Состояние регистра сдвига	Сумматор 1	Сумматор 2	Выходные кодовые комбинации
0	-	0000			-
1	1	1000	$1 \oplus 0 \oplus 0 = 1$	$1 \oplus 0 \oplus 0 = 1$	11
				$0 \oplus 0 = 1$	

2	0	0100	$0 \oplus 0 \oplus 0 = 0$	$0 \oplus$ $1 \oplus 0 = 1$	01
3	1	1010	$1 \oplus 1 \oplus 0 = 0$	$1 \oplus$ $0 \oplus 0 = 1$	01
4	1	1101	$1 \oplus 0 \oplus 1 = 0$	$1 \oplus$ $1 \oplus 1 = 1$	01
5	0	0110	$0 \oplus 1 \oplus 0 = 1$	$0 \oplus$ $1 \oplus 0 = 1$	11
6	1	1011	$1 \oplus 1 \oplus 1 = 1$	$1 \oplus$ $0 \oplus 1 = 0$	10

В данной таблице видно, как меняется состояние регистра сдвига с приходом нового информационного бита, показан и процесс образования выходных кодовых комбинаций образуемых в результате считывания коммутатором символов с обоих сумматоров.

### Полиномиальное представление

Иногда связи кодера описываются с помощью *полиномиального генератора*, аналогичного используемому для описания реализации обратной связи регистра сдвига циклических кодов. Сверточный кодер можно представить в виде набора из  $n$  полиномиальных генераторов, по одному для каждого из  $n$  сумматоров по модулю 2. Каждый полином имеет порядок  $K - 1$  или меньше и описывает связь кодирующего регистра сдвига с соответствующим сумматором по модулю 2, почти так же как и вектор связи. Коэффициенты возле каждого слагаемого полинома порядка  $(K - 1)$  равны либо 1, либо 0, в зависимости от того, имеется ли связь между регистром сдвига и сумматором по модулю 2. [7].

Для кодера на рисунке 3.57, можно записать полиномиальный генератор  $g_1(X)$  для верхних связей и  $g_2(X)$  - для нижних.

$$g_1(X) = 1 + X + X^2, \quad g_2(X) = 1 + X^2$$

Здесь слагаемое самого нижнего порядка в полиноме соответствует входному разряду регистра. Выходная последовательность находится следующим образом:  $U(X) = m(X)g_1(X)$  чередуется с  $m_1(X)g_2(X)$ .

Прежде всего, необходимо выразить вектор некоего сообщения  $m = 1 \ 0 \ 1$  в виде полинома, т.е.  $m(X) = 1 + X^2$ . Для очистки регистра предполагается использование нулей, следующих за битами сообщения. Тогда выходящий полином  $U(X)$ , или выходящая

последовательность  $U$  кодера (рисунок 3.57) для входного сообщения  $m$  может быть найдена следующим образом:

$$m(X)g_1(X) = (1+X^2)(1+X+X^2) = 1 + X + X^3 + X^4$$

$$\underline{m(X)g_2(X) = (1+X^2)(1+X^2) = 1+X^4}$$

$$m(X)g_1(X) = 1+X+0X^2+X^3+X^4$$

$$m(X)g_2(X) = 1+0X+0X^2+0X^3+X^4$$

$$U(X) = (1,1) + (1,0)X + (0,0)X^2 + (1,0)X^3 + (1,1)X^4$$

$$U = 11\ 10\ 00\ 10\ 11$$

В этом примере кодер был представлен в виде *полиномиальных генераторов*, с помощью которых описываются циклические коды [7]. *Полиномиальные генераторы кодера изображенного на рисунке 2.57 так же можно представить в виде векторов:*

$$g_1 = [111] \text{ и } g_2 = [101].$$

#### **Представление в виде кодовых деревьев, решеток и диаграмм состояний**

Помимо графического и полиномиального представления имеются три альтернативных метода, которые часто используются для описания свёрточного кода. Это древовидная диаграмма, решетчатая диаграмма и диаграмма состояний. Для примера, древовидная диаграмма для свёрточного кодера, показанного на рисунке 3.57 иллюстрируется на рисунке 3.58.

Рассмотрим свёрточный код со скоростью  $1/2$ ,  $K=3$ , описанный ранее и показанный на рисунке 3.57. Первый входной бит может быть 0 или 1. Соответствующие выходные биты – 00 или 11. Когда следующий бит входит в кодер, первый бит передвигается в следующую ячейку. Соответствующие выходные биты зависят от бита, переместившегося во вторую ячейку и нового входного бита. Следовательно, древовидная диаграмма для этого кода, показанная на рисунке 3.58, имеет две ветви на узел, соответствующие двум возможным входным символам. Жирной линией на рисунке 3.58 обозначен путь, по которому бы шел кодер, если бы входной комбинацией была последовательность 101.

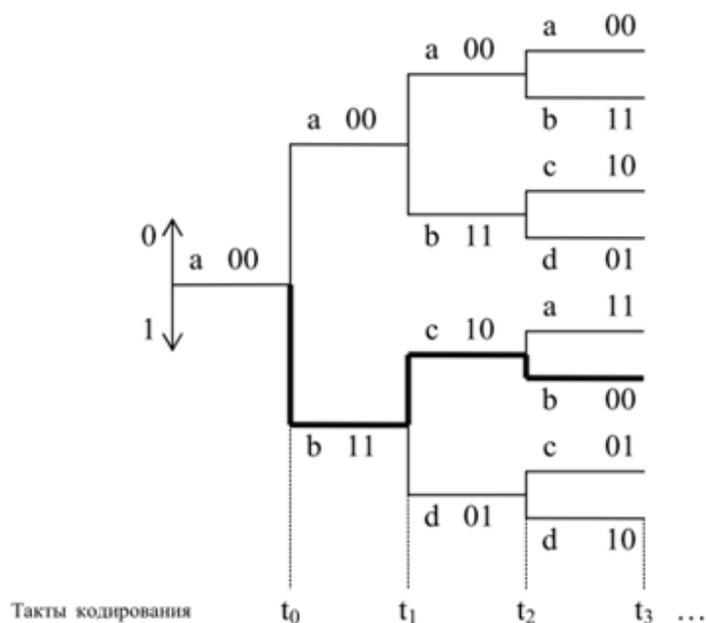


Рис. 3.67. Древоподобная диаграмма для сверточного кода, имеющего параметры  $K=3$ ,  $k=1$ ,  $n=2$

Поскольку кодовое ограничение по входу кодера  $K=3$ , а значит дерево начнёт повторяться после третьего шага. Как показано на рисунке 3.58, все ветви, исходящие из узла, обозначенного  $a$  (состояния  $a$ ) дают идентичные выходы. Путем слияния узлов, имеющих одинаковое название, мы получаем решётку, показанную на рисунке 3.68. На нём жирной линией обозначен путь по которому бы шел кодер, если бы входной комбинацией была последовательность 1100, в таком случае на выходе имелась бы комбинация 11 01 01 11.

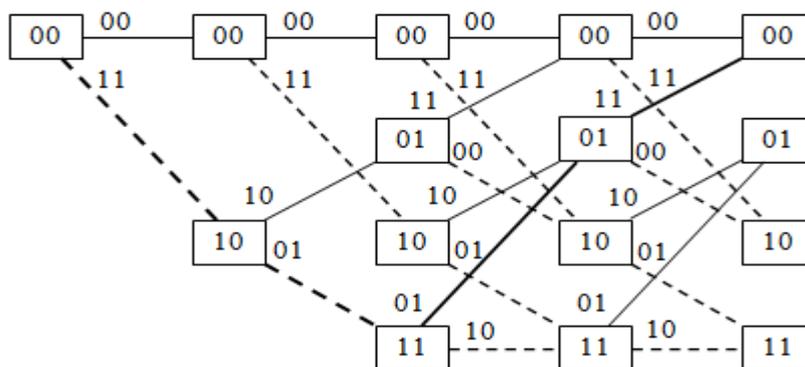


Рис. 3.68. Решётчатая диаграмма для сверточного кода, имеющего параметры  $K=3$ ,  $k=1$ ,  $n=2$

Поскольку выход кодера определяется входом и состоянием кодера, ещё более компактной, чем решётка, является диаграмма состояний. Диаграмма состояний — это граф возможных состояний кодера и возможных переходов из одного состояния в другое. Итоговая диаграмма состояний для этого кодера показана на рисунке 3.60.

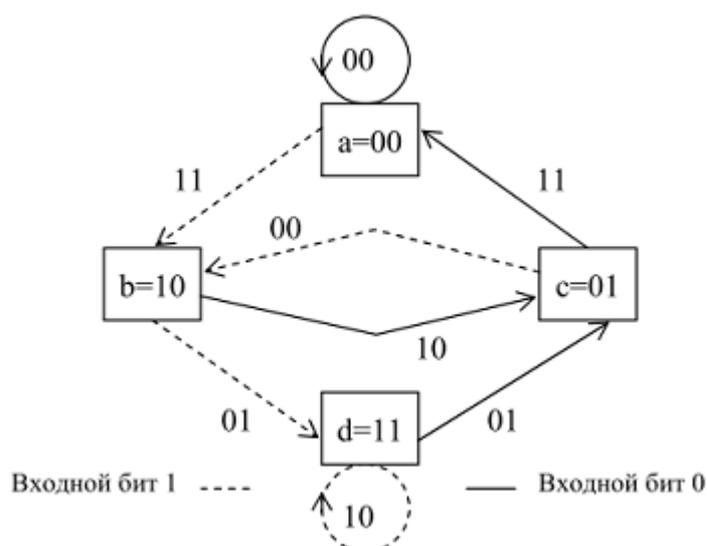


Рис. 3.69. Диаграмма состояний для свёрточного кода, имеющего параметры

$$K=3, k=1, n=2$$

Для обобщения стоит отметить, что свёрточный код со скоростью  $k/n$  и кодовым ограничением  $K$  характеризуется  $2^k$  ветвями, уходящими от каждого узла на древовидной диаграмме. Решетка и диаграмма состояний имеют (каждая из них)  $2^{k(K-1)}$  возможных состояний. Имеются  $2^k$  ветвей, входящих в каждое состояние, и  $2^k$  ветвей, покидающих каждое состояние (для решётки и дерева это верно после наступления установившегося режима) [1, 10].

### Методы декодирования свёрточных кодов

#### Метод порогового декодирования

При пороговом декодировании свёрточных кодов вычисляются синдромы (признаки места ошибочных символов), затем эти синдромы или последовательности, полученные посредством линейного преобразования синдромов, подаются на входы порогового элемента, где путем “голосования” (мажоритарный метод) и сравнения его результатов с порогом выносится решение о значении декодируемого символа. Основное достоинство этого метода декодирования – простота реализации. Однако он не полностью реализует потенциальные корректирующие способности свёрточного кода. Кроме того, не все свёрточные коды могут быть декодированы этим методом. Пороговое декодирование, как правило, применяется для систематических кодов. [1]. Общая схема декодера для свёрточного кода ( $R = 1/2$ ) представлена на рисунке 3.70.



Рис. 3.70. Общая схема декодера для сверточного кода

Декодер содержит аналог кодера, в котором по принимаемым информационным символам в сдвигающем регистре формируется копия проверочной последовательности. С этой целью синхронизатор декодера с помощью ключей 1 и 2 “расфасовывает” входную последовательность символов на 2 потока – информационный и проверочный, синхронизатор управляет работой всего декодера.

В формирователе синдрома (сумматоре по модулю 2) образуется последовательность синдромов  $S$ , которая поступает на вход синдромного регистра. В отсутствие в канале ошибок последовательности на входах формирователя синдрома всегда совпадают, и синдромная последовательность состоит из одних нулей. Различным наборам ошибок соответствуют определенные конфигурации синдромных последовательностей, в которых на определенных позициях появляются единичные символы. Закон формирования проверочных символов выбирается таким образом, чтобы по структуре синдромной последовательности можно было определить искаженные символы.

Логическая схема определяет по синдрому правильность записанного в информационном регистре блока информационных символов. Если имеется комбинация ошибок, которая может быть исправлена, то логическая схема исправляет ошибки в этом блоке путем подачи единичных символов на выходную схему суммирования по модулю 2 в моменты выхода из информационного регистра искаженных символов.

Ошибки, исправляемые в очередном информационном блоке, могут влиять на символы синдромов, соответствующих последующим блокам, поскольку сверточные коды непрерывны. Для того чтобы декодер смог полностью реализовать свои корректирующие возможности, следует исключить влияние этих ошибок. Это может быть достигнуто за счет обратной связи, которая на схеме (рисунок 3.62) представлена пунктирной линией.

Обратная связь преобразует синдромный регистр прямого действия в нелинейный регистр сдвига с обратной связью. Это может привести к явлению, называемому

размножением ошибок. Неисправимые ошибки в канале могут вызвать переход синдромного регистра в такое состояние, что и при отсутствии аддитивных ошибок в канале декодер будет продолжать всегда декодировать неправильно. Причина этого состоит в том, что выход нелинейного регистра сдвига с обратной связью, когда на его вход поступает нулевая последовательность, а начальное состояние – ненулевое, может быть периодическим [6].

### **Последовательный алгоритм декодирования**

При рассмотрении алгоритмов последовательного декодирования удобно представлять сверточный код в виде кодового дерева. Как уже отмечалось ранее, исходному нулевому состоянию сдвигающего регистра кодера соответствует начальный узел дерева. Если входной информационный символ, поступающий в регистр равен 1, то ему приписывается линия (ребро дерева), идущая, как принято на этом рисунке, вниз, а если информационный символ равен 0, – то вверх. Тем самым получаем два новых узла, соответствующие следующему такту работы кодера, для каждого из которых дерево строится далее аналогичным образом, и так далее. Над каждым ребром дерева записываются кодовые символы, получаемые при этом на выходе кодера. Совокупность нескольких последовательных ребер, соединяющих какие-либо два узла, составляет ветвь дерева.

Каждая последовательность кодируемых информационных символов порождает определенный путь по кодовому дереву. Очевидно, задача декодера заключается в отыскании истинного (правильного) пути, то есть того пути, который в действительности был порожден кодером [1].

Таким образом, при алгоритмах последовательного декодирования декодер определяет наиболее правдоподобный путь по решетке, что позволяет исключить из анализа большую часть остальных путей, имеющих меньшее правдоподобие.

Для этого, сначала необходимо передвигаться поочередно вдоль каждого пути кодовой решетки, сравнивая принятую последовательность со значением, соответствующим пути, по которому происходит движение. Если при этом удастся обнаружить некоторый путь, значение которого, совпадает с принятой последовательностью, то естественно считать, что эта последовательность и передавалась.

Если быть точнее, происходит сравнение принятой комбинации с комбинациями возможными для данного шага декодирования. Путь считается наиболее правдоподобным, если метрика между принятой последовательностью и последовательностью соответствующей данному переходу минимальна.

Резюмируем: на каждом шаге декодирования мы имеем два исходящих из узла пути. Сначала мы вычисляем расстояние Хемминга между принятой комбинацией и комбинацией соответствующей одному исходящему пути, затем вычисляем расстояние Хемминга между

принятой комбинацией и комбинацией соответствующей второму пути. После чего выбираем тот путь, значение расстояния Хемминга которого меньше. Данный процесс показан на рисунке 3.71. В данном случае переход вниз по решетке будет считаться наиболее правдоподобным, так как расстояние Хемминга между значением соответствующему этому переходу и принятой комбинацией является наименьшим.

Принятая последовательность: 11

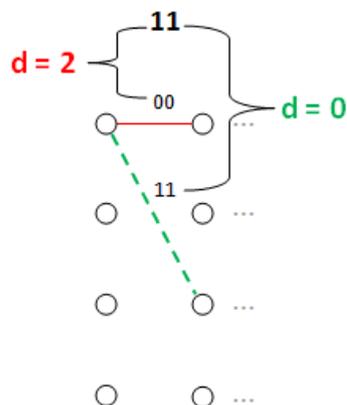


Рис. 3.71. Выбор пути с наименьшим расстоянием Хэмминга

Декодер выполняет операцию выбора пути с наименьшей метрикой до тех пор пока не столкнётся с неоднозначностью, при которой принять решение о том какой путь является кратчайшим не представляется возможным из-за того что расстояния Хемминга между принятой последовательностью и значениями двух возможных путей одинаковы. В этой ситуации декодер вынужден проверить оба возможных пути, и по следующим метрикам принять решение касаясь неопределённости.

В нашем же случае, если расстояния Хемминга для обоих путей одинаковы, то прибегаем к жесткому методу принятия решений и пользуемся следующим правилом:

- Если мы находимся в узле 00 или 11, и пришла комбинация 00 или 01 – считаем что передавался ноль, 10 или 11 – единица.
- Если мы находимся в узле 01 или 10, и пришла комбинация 00 или 01 – считаем что передавалась единица, 10 или 11 – ноль.

Аналогичным образом декодер продолжает углубляться в решетку и принимать решения, касающиеся информационных битов, устраняя все пути, кроме одного.

На рисунке 3.72 изображена ситуация когда на вход декодера поступает последовательность, в которой биты выделенные цветом заведомо искажены так, как это бывает в реальных каналах связи.

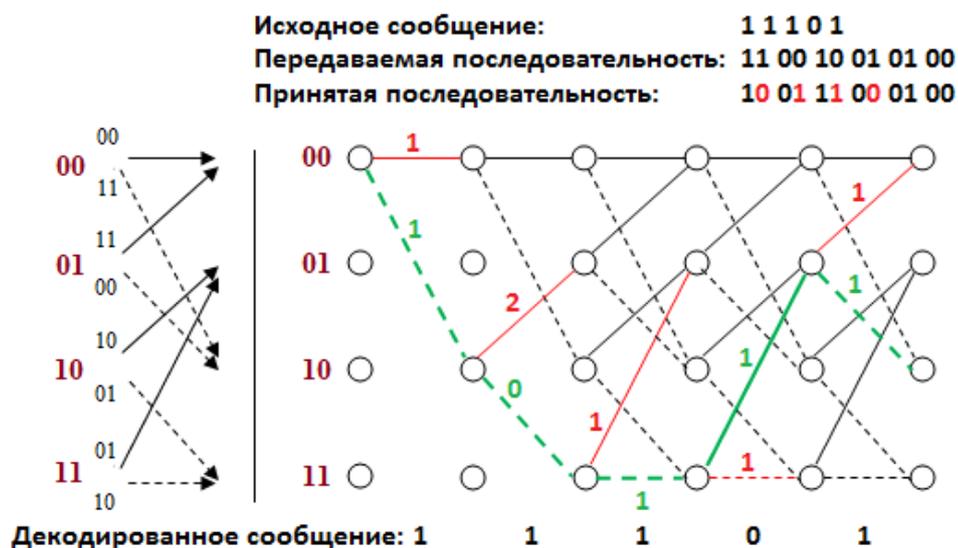


Рис. 3.72. Решетчатая диаграмма последовательного декодера

### Сравнение алгоритма Витерби с последовательным алгоритмом

Принципиальное отличие алгоритма Витерби от последовательного алгоритма декодирования заключается в том, что алгоритм Витерби принимает решение по суммарным метрикам путей, причём в узлах, где эти пути сходятся. Этот шаг в процессе декодирования показан на рисунке 3.65. (метрики обозначены данным образом для наглядности).

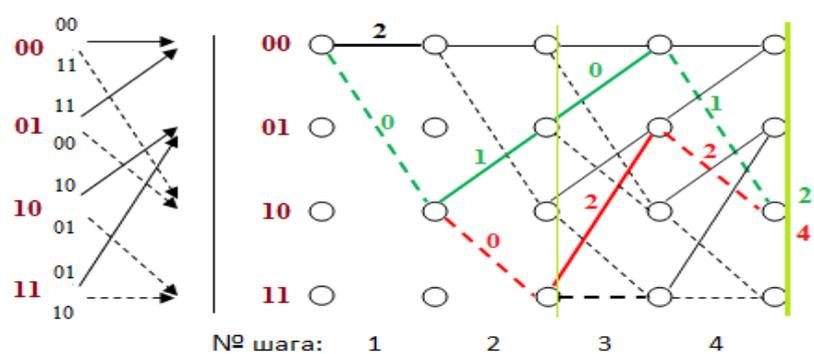


Рис. 3.73. Решетчатая диаграмма декодера Витерби

Таким образом, если бы в ситуации, представленной на рисунке 3.64, мы принимали решение на основе последовательного алгоритма, то на первом и втором шаге декодирования мы приняли бы решение идти вниз по решетке, выбирая кратчайшие метрики Хемминга. Однако в таком случае мы бы не учли метрики последующих ветвей нашего, на первый взгляд кратчайшего пути, которые на шаге три и четыре сильно возрастают, а значит, выбрали бы не самый короткий путь при декодировании. Поэтому, алгоритм Витерби является оптимальным. Однако он сложнее последовательного как в понимании, так и в реализации. В качестве метрики выступает расстояние Хемминга.

## Программная реализация сверточного кодера

### Подготовка данных к кодированию

Сначала, символы, введенные с клавиатуры ассоциируются со своими десятичными значениями в таблице ASCII. Затем, ASCII-код символа переводится в восьмеричную систему исчисления.

Ранее был объявлен массив «Bit\_Array()», который имеет 8 ячеек, в которых содержится двоичное представление чисел от 0 до 7. Так как восьмеричная система имеет всего восемь цифр (0-7) то перебирается массив в цикле от 0 до 7, и находится равенство между индексом массива (счетчиком цикла) и восьмеричной цифрой ASCII-кода, после чего вместо нее подставляется двоичное представление из ячейки массива.

К примеру, цифра 0, будет иметь десятичный ASCII код - 48, в восьмеричный - 60. Что в двоичной системе исчисления будет 110000. Если ASCII-код - двузначное число, впереди дописывается 00, чтобы двоичные числа имели одинаковую длину. Формируется строка бит сообщения 00110000.

После того как символы исходного текста ассоциированы с кодами таблицы кодов ASCII и переведены в двоичный вид, осуществляется свёрточное кодирование.

### Функция свёрточного кодирования

Разрабатываемый свёрточный кодер является несистематическим, без обратной связи и изображен на рисунке 3.74. Функциональную схему кодера (см. Приложении Б).

Кодер называется систематическим, если в каждом кодовом подблоке длины  $s$  имеется  $b$  символов, которые совпадают с символами соответствующего информационного подблока. В противном случае кодер - несистематический.

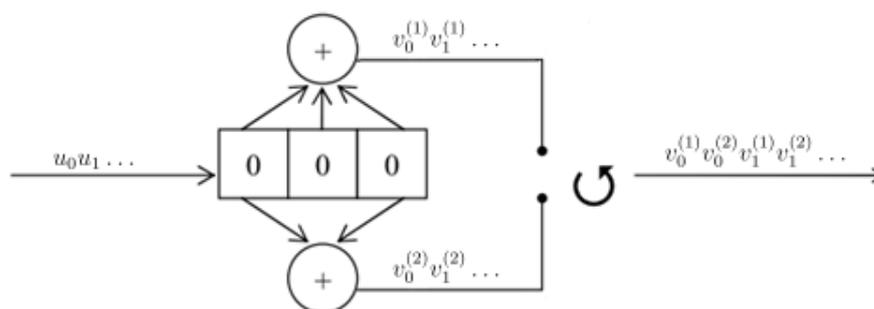


Рис. 3.74. Двоичный сверточный кодер скорости  $R = 1/2$ .

Бесконечная последовательность информационных символов  $u = u_0 u_1 \dots$  поступает в регистр сдвига (в нашем примере памяти  $m = 3$ , то есть с двумя элементами задержки). Кодер генерирует две выходные последовательности  $v^{(1)} = v_0^{(1)} v_1^{(1)} \dots$  и  $v^{(2)} = v_0^{(2)} v_1^{(2)} \dots$ , которые поступают в коммутатор, формирующий выходную (кодую) последовательность  $v = v_0^{(1)} v_0^{(2)} v_1^{(1)} v_1^{(2)} \dots$ , передаваемую по каналу. Поскольку на каждый

входящий в кодер информационный символ приходится два кодовых символа, кодовая скорость равна  $R = 1/2$ . Если одна из кодовых последовательностей  $v^{(1)}$  или  $v^{(2)}$  совпадает с информационной последовательностью  $u$ , то, как отмечалось ранее, кодер называется систематическим. В противном случае кодер называется несистематическим [9].

Генераторы свёрточного кодера изображенного на рисунке 3.75. можно описать при помощи полиномов:

$$G_1 = 1+X+X^2$$

$$G_2 = 1+X^2$$

Таким образом, порождающая матрица этого свертчного кодера , в терминах оператора задержки имеет вид:

$$G(X) = (1+X+X^2 \ 1+X^2).$$

Как уже отмечалось ранее, кодер – это автомат. В данном случае имеется классический алгоритм автомата Мили — конечный автомат, выходная последовательность которого (в отличие от автомата Мура) зависит от состояния автомата и входных сигналов. Это означает, что в графе состояний каждому ребру соответствует некоторое значение (выходной символ). В вершины графа автомата Мили записываются выходящие сигналы, а дугам графа приписывают условие перехода из одного состояния в другое, а также входящие сигналы. Диаграмма состояний свёрточного кодера представленного на рисунке 3.66 изображена на рисунке 3.75.

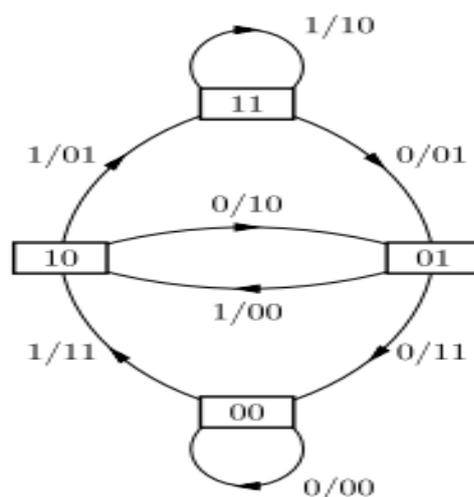


Рис. 3.75. Диаграмма состояний свертчного кодера скорости  $R = 1/2$

Свёрточное кодирование кодером со скоростью  $R = 1/2$  в программном комплексе выполняет функция Fun\_BitCoder. После объявления массива и его целочисленных переменных обнуляются регистры сдвига, описывается цикл, осуществляющий побитное чтение информационных символов, их сдвиг в регистрах и суммирование по модулю два

согласно установленным правилам. Элемент исходного кода программы, отвечающий за реализацию свёрточного кодирования, представлен на рисунке 3.76.

```

R1 = 0
R2 = 0
For i = 0 To length_array 'цикл обработки сообщения автоматом Мили
X = S_mess_chr(i).ToString
Y1 = X Xor R1 'сумма по модулю 2(исключающее или)
Y2 = X Xor R2 '
R2 = R1 'сдвиг битов в регистрах
R1 = X '
Y1 = Y1 Xor R2 'сумма по модулю 2(исключающее или)
s_res = s_res + Y1.ToString + Y2.ToString 'результат кодирования
Next
Return s_res
End Function

```

Рис. 3.76. Элемент программного кода функции свёрточного кодирования

В роли сумматоров выступают переменные Y1 и Y2, а в роли ячеек регистра переменные R1 и R2. Коммутатором является переменная s\_res. Для осуществления поочередной коммутации бит с выходов кодера символы переводятся в строковый тип.

### Функция имитации канала связи

По мере прохождения по каналу связи сигнал подвергается воздействию помех, именно этот процесс имитирует данная функция. Помехи - случайные воздействия, искажающие передаваемый сигнал. Воздействие помехи на сигнал может быть двояким. Если помеха  $\zeta(t)$  складывается с сигналом  $s(t)$  и на вход приёмника действует их сумма  $x(t) = s(t) + \zeta(t)$  такую помеху называют аддитивной. Если результирующий сигнал равен произведению помехи и передаваемого сигнала  $x(t) = s(t) \cdot \zeta(t)$ , то помеху называют мультипликативной.

В канале с мультипликативной помехой наблюдаются всплески искажений, причинами которых могут быть, например, коммутационные помехи, быстрые замирания радиосигнала. Выражается мультипликативная помеха в изменении характеристик линии связи (сопротивление линии связи (ЛС), частота среза ЛС, нелинейность характеристик ЛС).

Чтобы приблизить модель симметричного канала к действительности, вводится понятие пакета ошибок.

Пакетом ошибок длины L называется вектор ошибок E, все ненулевые компоненты которого расположены на отрезке из подряд следующих позиций, причем в начале и конце отрезка расположены ненулевые компоненты.

Аддитивная помеха не зависит от ЛС и определяется внешними воздействиями на среду передачи сигналов. В качестве модели аддитивной помехи обычно принимают белый шум, то есть стационарный гауссовский случайный процесс, имеющий нулевое математическое

ожидание и равномерный и бесконечно широкий спектр плотности мощности. При воздействии помех на передаваемый сигнал, имеющих аддитивный характер,  $L=1$  имеем пакет длиной в один символ, то есть однократную ошибку.

В разрабатываемом программном комплексе имеется возможность пронаблюдать эффективность работы последовательного декодера свёрточных кодов как при наличии однократных ошибок, так и пачек длиной  $L > 1$ .

На рисунке 3.77 приведена блок-схема программно реализованного алгоритма имитации канала связи с белым шумом и канала с многолучевым распространением.

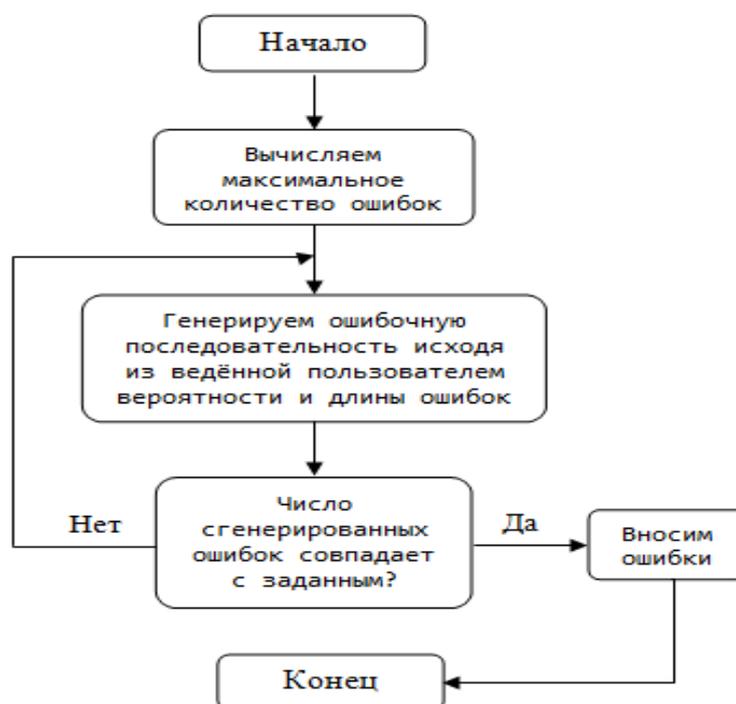


Рис. 3.77. Блок-схема для алгоритма имитации канала связи

За имитацию канала связи отвечает функция «Fun\_SimulatorChannelCom». В ней, первым делом, определяется максимальное количество генерируемых ошибок:

$$\max\_mistake\_dou = \text{length\_array} * (\text{CInt}(\text{probability\_mistake}) / 100),$$

где:  $\max\_mistake\_dou$  – максимальное количество ошибок

$\text{length\_array}$  – длина сообщения

$\text{probability\_mistake}$  – вероятность ошибки

Например, если длина сообщения равна 80 бит, а битовая вероятность ошибки – 0.1 то генератор сгенерирует от 0 до 8 ошибок.

Далее, активируется обычный генератор случайных чисел в диапазоне от 1 до 100:  
`Dim value As Integer = CInt(Int((100 * Rnd()) + 1))`

Функция CInt приводит выражение в тип Integer и округляет его до ближайшего целого. После чего число сгенерированных ошибок сверяется с заданным числом ошибок,

если ошибок меньше чем задано, цикл продолжается. Проверяется также заданная вероятность с сгенерированным числом, если оно меньше или равно, то в сообщение вносится ошибка. Внесение ошибки осуществляется путем инвертирования 0 и 1. Элемент исходного кода программы, отвечающий за реализацию функции имитации канала связи представлен на рисунке 3.79.

```

If (max_mistake_int > 0) And (i <= (length_array_1 - 1)) And (remainder > 0) Then ' пока
кол-во введенных ошибок не достигло максимума
    If (value <= CInt(probability_mistake)) Then 'если случайное число или
равно, заданной вероятности ошибки
        j = 1 ' выставляем флаг ошибки

If (S_mess_chr(f).ToString = "0") Then ' заменяем биты в сообщении
    S_mess_chr(f) = "1"
Else
    S_mess_chr(f) = "0"
End If

```

Рис. 3.78. Элемент программного кода функции имитации канала связи

Если были заданы двойные ошибки, инвертируются по два соседних бита.

В конечном итоге, после того как подсчет ошибок осуществлен, и их количество и кратность соответствует заданным параметрам, сообщение с ошибками выводится на блок индикации, где искаженные биты выделяются цветом.

### Функция вычисления метрик путей

Важным этапом разработки алгоритма декодирования является выбор алгоритма вычисления метрик путей. В качестве такой метрики будет принято расстояние Хэмминга. Расстояние Хэмминга — число позиций, в которых соответствующие символы двух слов одинаковой длины различны. Согласно выбранному алгоритму вычисления метрик путей, биты сообщения суммируются по модулю два со значениями возможных путей и полученному результату подставляется его десятичное значение, соответствующее величине расстояния Хемминга между слагаемыми. За это отвечает функция «Fun\_Metric», программная реализация которой приведена на рисунке 3.70, где:

s\_1 – биты сообщения;

s\_2 – возможные значения путей решетки;

res – результат суммирования по модулю два бит сообщения со значениями возможных путей решетки;

res\_1 - десятичное значение, соответствующее величине расстояния Хемминга между слагаемыми.

```

Public Function Fun_Metric(ByVal f_s_mess_1 As String, ByVal f_s_mess_2 As String)
    Dim s_1, s_2 As Integer
    Dim res, res_1 As Integer
    s_1 = CInt(f_s_mess_1)
    s_2 = CInt(f_s_mess_2)
    res = s_1 Xor s_2 ' сумма по модулю два
    Select Case res
        Case "00"
            res_1 = 0
        Case "01"
            res_1 = 1
        Case "10"
            res_1 = 1
        Case "11"
            res_1 = 2
    End Select
    Return res_1
End Function

```

Рис. 3.79. Программный код функции вычисления метрик путей

Из таблицы 3.4 видно, что значение 00 может стать результатом суммирования только двух слов, биты которых не отличаются ни в одной из позиций, а это значит, что расстояние Хемминга между такими словами равно 0. Значения  $res = 01$  и  $res = 10$  возможны лишь в том случае, если по модулю 2 суммировались слова, соответствующие биты которых отличаются друг от друга в одной позиции, то есть расстояние Хемминга между ними равно 1. Что же касается значения  $res = 11$ , то оно может получиться только в результате сложения двух слов, биты которых в обеих позициях отличаются, и расстояние Хемминга между этими словами равно 2.

Таблица 3.4. Вычисление метрик путей

№ такта	Входной информационный бит	Состояние регистра сдвига	Сумматор 1	Сумматор 2	Выходные кодовые комбинации
0	-	0000			-
1	1	1000	$1 \oplus 0 \oplus 0 = 1$	$1 \oplus$ $0 \oplus 0 = 1$	11
2	0	0100	$0 \oplus 0 \oplus 0 = 0$	$0 \oplus$ $1 \oplus 0 = 1$	01
3	1	1010	$1 \oplus 1 \oplus 0 = 0$	$1 \oplus$ $0 \oplus 0 = 1$	01
4	1	1101	$1 \oplus 0 \oplus 1 = 0$	$1 \oplus$	01

				$1 \oplus 1 = 1$	
5	0	0110	$0 \oplus 1 \oplus 0 = 1$	$0 \oplus$ $1 \oplus 0 = 1$	11
6	1	1011	$1 \oplus 1 \oplus 1 = 1$	$1 \oplus$ $0 \oplus 1 = 0$	10

### Функция декодирования последовательным алгоритмом

Учитывая топологию решетки, первый шаг декодирования был реализован отдельно. В элементе исходного кода приведенного на рисунке 3.71 происходит проверка и выбор возможного пути, зависимости от битов сообщения (проверяются биты массива `S_str_Array()`).

Как мы уже знаем из теории, на первом шаге декодирования состояние декодера – 00, возможны только два пути, к узлам 00 и 10, которым соответствуют кодовые символы с выхода кодера 00 и 11.

Однако на вход декодера могут придти целых четыре комбинации бит сообщения, поэтому, на первом шаге декодирования, было решено связать приход кодовой комбинации 00 и 01 к переходу в узел 00, а кодовые комбинации 10 и 11 к переходу в узел 10. Так мы избавляемся от неопределённостей, заставляя декодер принять решение, даже если пришла не разрешенная комбинация.

```

If ((S_str_Array(i) = "00") Or (S_str_Array(i) = "01")) Then
    state_decoder = "00"
    state_metric = Fun_Metric(state_decoder, S_str_Array(i))
    s_mess = "0"
    f_x1 = Arr_XY(0, f)
    f_y1 = 284
    ...
ElseIf ((S_str_Array(i) = "10") Or (S_str_Array(i) = "11")) Then
    state_decoder = "10"
    state_metric = Fun_Metric(state_decoder, S_str_Array(i))
    s_mess = "1"
    f_x1 = Arr_XY(2, f)
    f_y1 = 384
    ...
End If
GoTo End_1_step

```

Рис. 3.80. Элемент программного кода функции декодирования последовательным алгоритмом

После того как путь выбран, в переменную `state_metric` записывается значение метрики пути, вычисленное описанной функцией `Fun_Metric`. Далее ставится метка 0 или 1,

указывающая на то какой символ передавался, после чего задаются конечные координаты по горизонтали и вертикали отрезка для выбранного пути.

На этом заканчивается первый шаг декодирования и осуществляется безусловный переход на метку End\_1\_step. Сюда же переходит программа и после отработки всех остальных шагов.

Далее происходит аналогичные операции, только количество возможных путей увеличивается до 4-х (00, 01, 10, 11), осуществляется переход ко второму шагу декодирования. Граф автомата, каковым является наш декодер, принимает вид показанный на рисунке 3.82

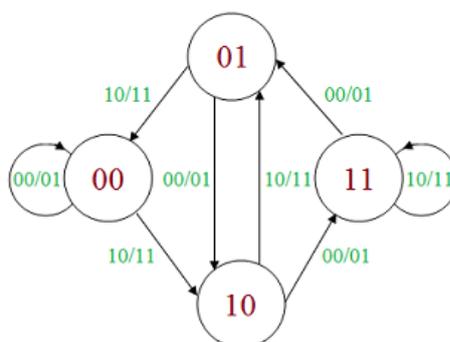


Рис. 3.81. Граф реализуемого последовательного декодера

Стратегия представленная на рисунке 3.82 реализуется при помощи алгоритма подобному тому, что был использован на первом шаге декодирования, теперь декодер так же выбирает путь с кратчайшей метрикой, однако в случае если расстояния Хемминга для обоих путей одинаковы, то прибегает жесткому методу принятия решений:

- Если мы находимся в узле 00 или 11, и пришла комбинация 00 или 01 – считаем что передавался ноль (двигаемся вверх), 10 или 11 – передавалась единица (двигаемся вниз).
- Если мы находимся в узле 01 или 10, и пришла комбинация 00 или 01 – считаем что передавалась единица (двигаемся вниз), 10 или 11 – передавался ноль (двигаемся вверх).

Процесс происходит до тех пор пока не закончится сообщение. Полный исходный код функции декодирования приведен в приложении (см. Приложение И).

### Подготовка к выводу результатов декодирования

Для того чтобы в конечном итоге получить символьный результат организуем цикл реверсирования массива символов сообщения, иначе мы получим строку в зеркальном отражении. Организуем цикл перебора массива символов сообщения, разделяем их на пачки по 8 бит и каждую пачку преобразуем в десятичное число по принципу:  $2^7 + 2^6 + 2^5 + 2^4 +$

$2^3 + 2^2 + 2^1 + 2^0$  а затем получаем символ из его ASCII-кода. Повторяем операции пока не закончится сообщение, после чего результаты выводятся в соответствующих окнах.

### Сверточное декодирование. Практическая часть

#### ЗАДАНИЕ №1

##### Вариант 1

Сверточный код:  $g_1(X) = 1 + X^2 + X^3$ ,  $g_2(X) = 1 + X + X^3$ .

Входная последовательность:  $V = 10110$ .

На рисунке 3.73 представлен сверточный кодер.

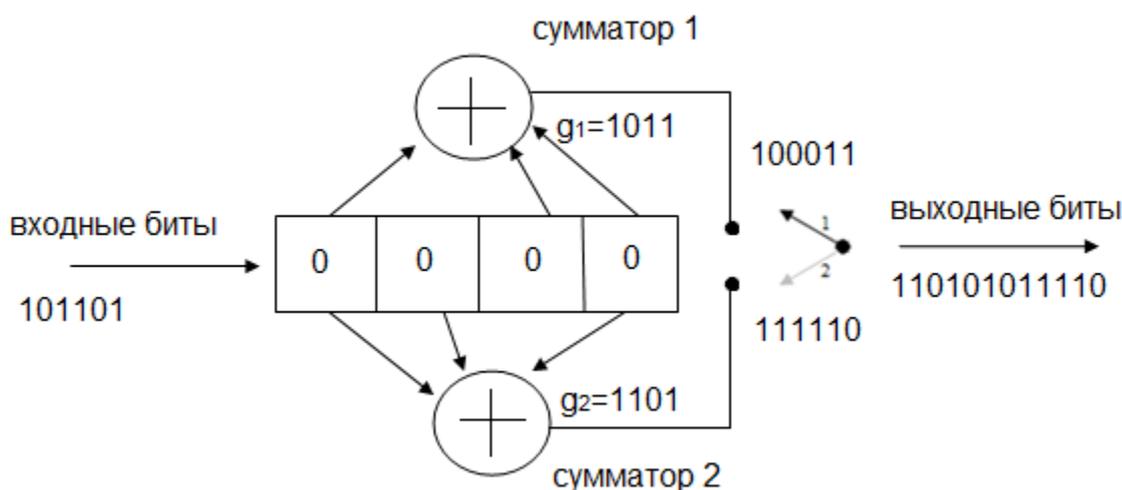


Рис. 3.82. Сверточный кодер

Произвели кодирование входной последовательности  $V$  данным сверточным кодером и нарисовали таблицу кодирования информационного потока. Данные представлены в таблице 3.5.

Таблица 3.5. Процесс кодирования информационного потока

№ такта	Входной информационный бит	Состояние регистра сдвига	Сумматор 1	Сумматор 2	Выходные кодовые комбинации
0	-	0000			-
1	1	1000	$1 \oplus 0 \oplus 0 = 1$	$1 \oplus 0 \oplus 0 = 1$	11
2	0	0100	$0 \oplus 1 \oplus 0 = 1$	$0 \oplus 1 \oplus 0 = 1$	01

				$1 \oplus 0 = 1$	
3	1	1010	$1 \oplus 1 \oplus 0 = 0$	$1 \oplus$ $0 \oplus 0 = 1$	01
4	1	1101	$1 \oplus 0 \oplus 1 = 0$	$1 \oplus$ $1 \oplus 1 = 1$	01
5	0	0110	$0 \oplus 1 \oplus 0 = 1$	$0 \oplus$ $1 \oplus 0 = 1$	11
6	1	1011	$1 \oplus 1 \oplus 1 = 1$	$1 \oplus$ $0 \oplus 1 = 0$	10

Построили решетку кодера:

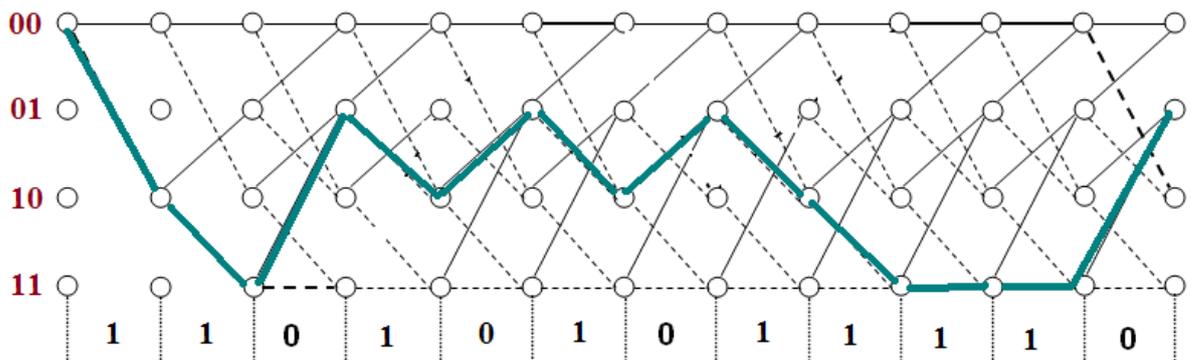


Рис. 3.83. Решетка кодера

Изобразили решетку декодера и произвели декодирование:

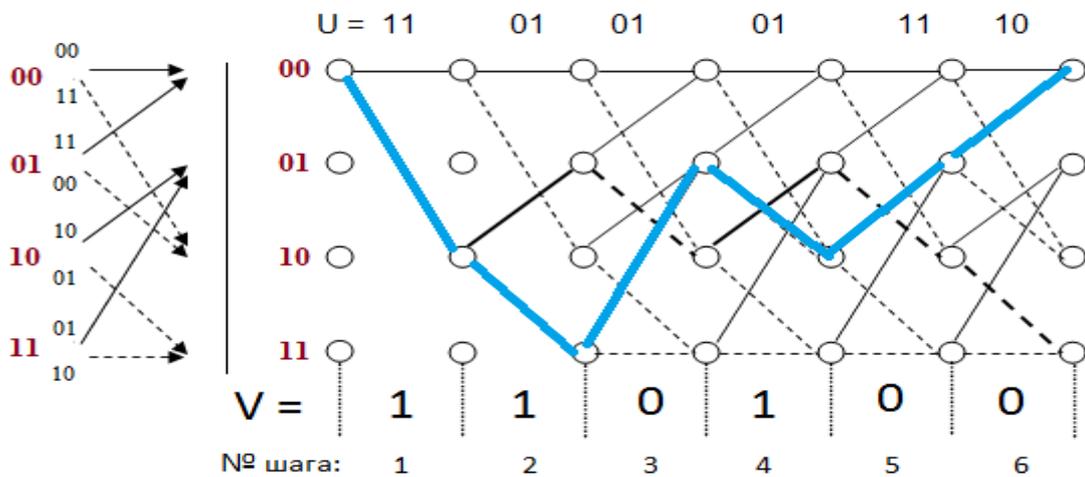


Рис. 3.84. Решетка декодера

В результате были выявлены 6 ошибок: После декодера был получен результат: 11 10 11 01 10 01. Подчеркиванием выделены места, где были совершены ошибки.

### Вариант 2

Сверточный код:  $g_1(X) = X + X^2 + X^3$ ,  $g_2(X) = 1 + X + X^2$ .

Входная последовательность:  $V = 010100$ .

На рисунке 3.85 представлен сверточный кодер.

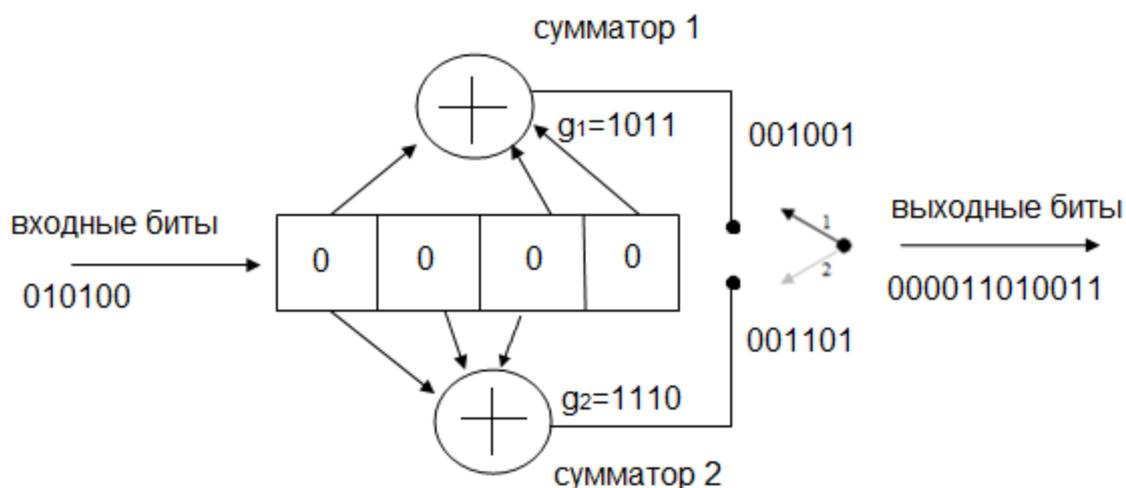


Рис. 3.85. Свёрточный кодер

Произвели кодирование входной последовательности  $V$  данным свёрточным кодером и нарисовали таблицу кодирования информационного потока. Данные представлены в таблице 3.6.

Таблица 3.6. Процесс кодирования информационного потока

№ такта	Входной информационный бит	Состояние регистра сдвига	Сумматор 1	Сумматор 2	Выходные кодовые комбинации
0	-	0000			-
1	0	0000	$0 \oplus 0 \oplus 0 = 0$	$0 \oplus 0 \oplus 0 = 0$	00
2	0	0000	$0 \oplus 0 \oplus 0 = 0$	$0 \oplus 0 \oplus 0 = 0$	00
3	1	1000	$1 \oplus 0 \oplus 0 = 1$	$1 \oplus 0 \oplus 0 = 1$	11

				$0 \oplus 0 = 1$	
4	0	0100	$0 \oplus 0 \oplus 0 = 0$	$0 \oplus$ $1 \oplus 0 = 1$	01
5	1	1010	$1 \oplus 1 \oplus 0 = 0$	$1 \oplus$ $0 \oplus 1 = 0$	00
6	0	0101	$0 \oplus 0 \oplus 1 = 1$	$0 \oplus$ $1 \oplus 0 = 1$	11

Построили решетку кодера:

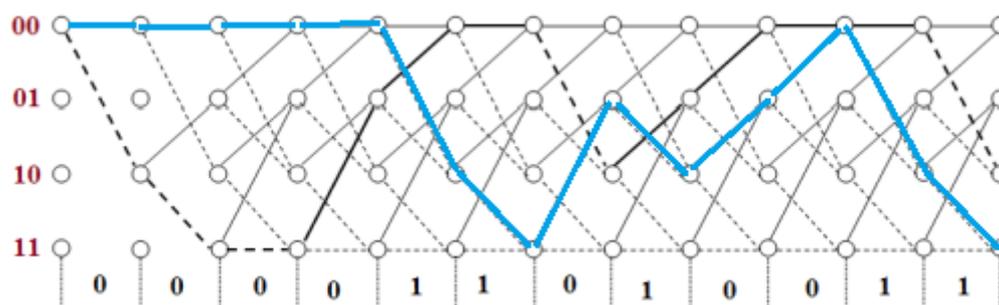


Рис. 3.86. Решетка кодера

Изобразили решетку декодера и произвели декодирование:

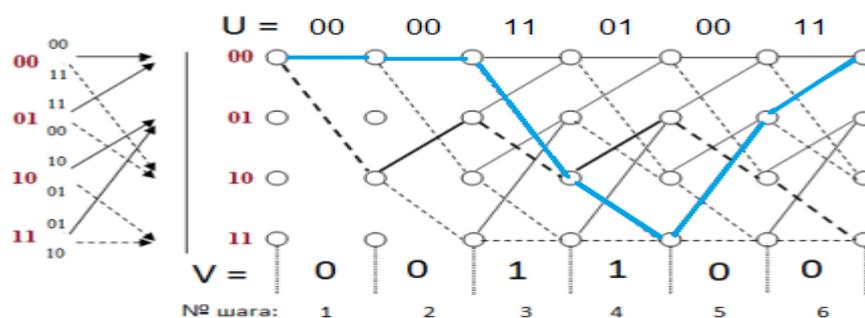


Рис. 3.87. Решетка декодера

В результате были выявлены 5 ошибок: После декодера был получен результат: 00 00 10 11 01 00. Подчеркиванием выделены места, где были совершены ошибки.

## ЗАДАНИЕ №2

Построить графики зависимости числа ошибок от вероятности ошибок для одиночной и двойной длины ошибок.

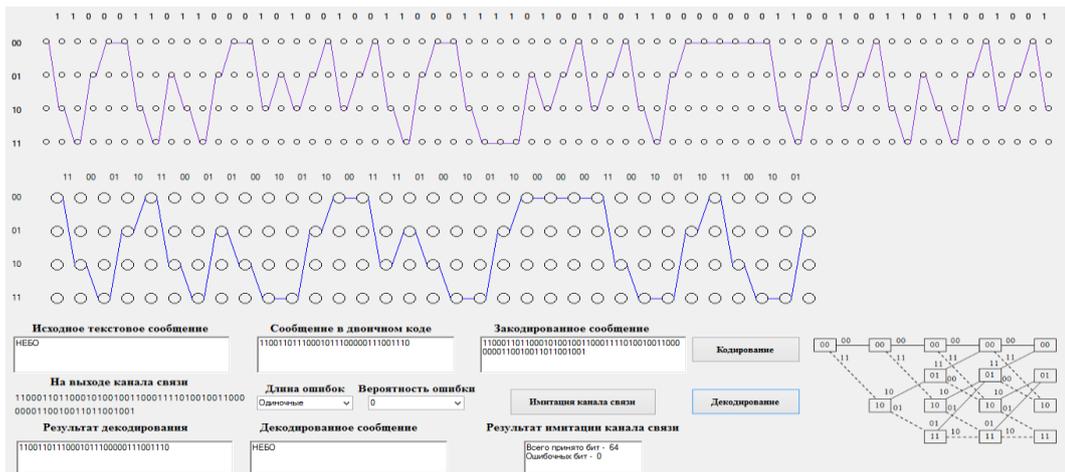


Рис. 3.88. Процесс кодирования и декодирования

График зависимости числа ошибок от вероятности ошибок для одиночной длины ошибок представлен на рисунке 3.89, а для двойной длины ошибок на рисунке 3.90.

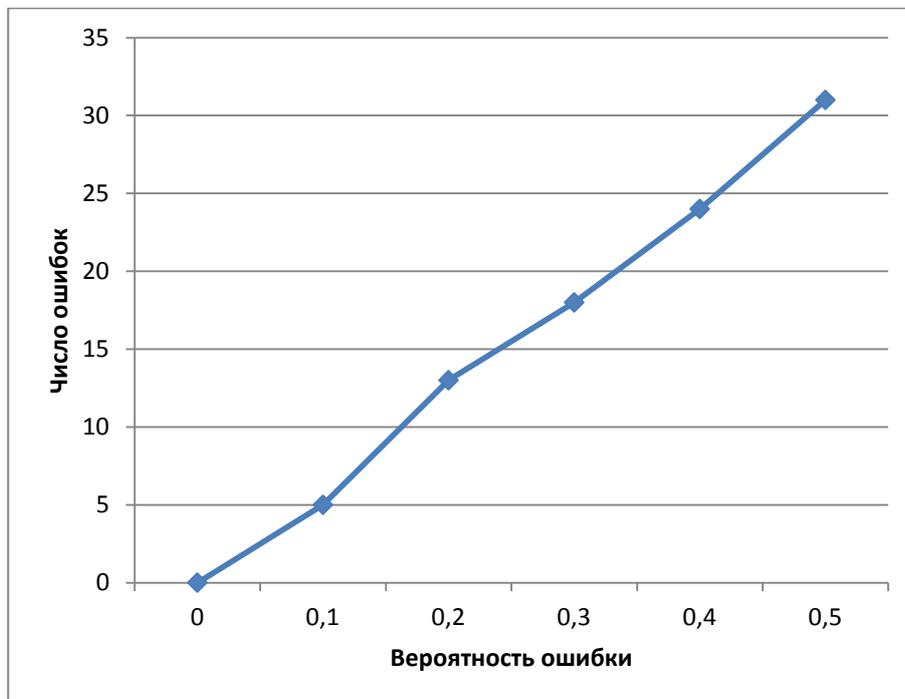


Рис. 3.89. График зависимости числа ошибок от вероятности ошибок для одиночной длины ошибок

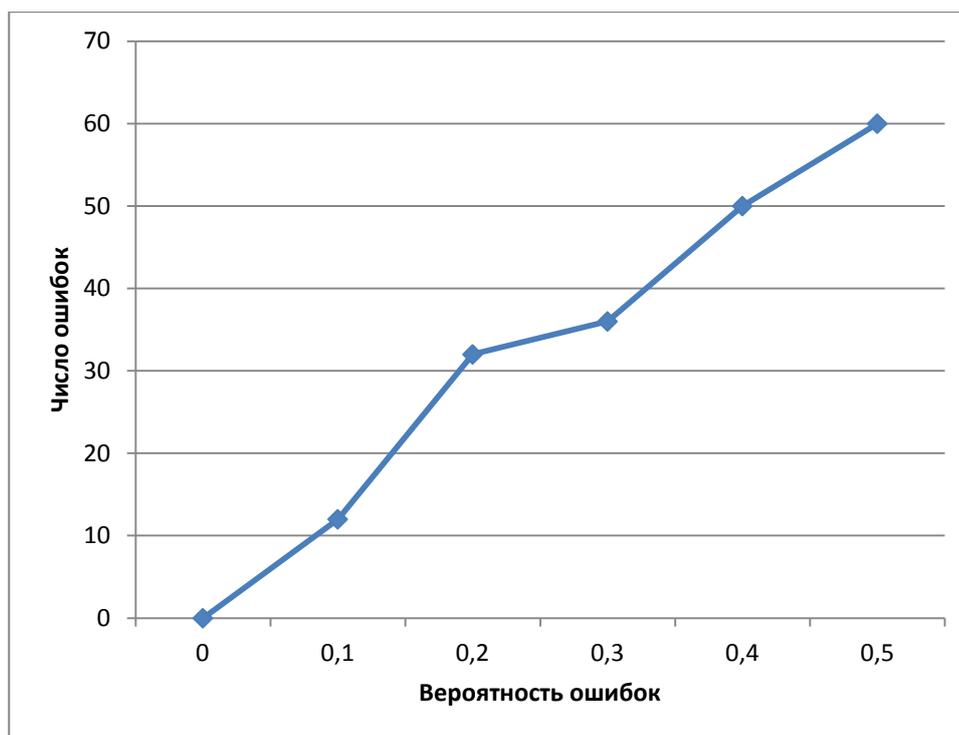


Рис. 3.90. График зависимости числа ошибок от вероятности ошибок для двойной длины ошибок

В процессе выполнения индивидуальной работы была осуществлена работа по созданию программного комплекса для визуализации и исследования метода свёрточного декодирования на основе последовательного алгоритма. Программный комплекс выполнен в полном соответствии с техническим заданием, и отвечает всем заявленным в нём требованиям.

Были произведены лабораторные испытания программного комплекса, в ходе которых была отмечена высокая эффективность его использования в качестве макета наглядно демонстрирующего процессы свёрточного кодирования и последовательного алгоритма декодирования.

### **3.4. Декодирование сверточных кодов по методу Витерби с использованием ПО MATLAB**

#### **Алгоритм свёрточного декодирования Витерби [10]**

В 1967 году Витерби разработал и проанализировал алгоритм, в котором, по сути, реализуется декодирование, основанное на принципе максимального правдоподобия; однако в нем уменьшается вычислительная нагрузка за счет использования особенностей структуры конкретной решетки кода. Преимущество декодирования Витерби, по сравнению с декодированием по методу "грубой силы", заключается в том, что сложность декодера Витерби не является функцией количества символов в последовательности кодовых слов.

Алгоритм включает в себя вычисление *меры подобия* (или *расстояния*), между сигналом, полученным в момент времени  $t$ , и всеми путями решетки, входящими в каждое состояние в момент времени  $t_i$ . В алгоритме Витерби не рассматриваются те пути решетки, которые, согласно принципу максимального правдоподобия, заведомо не могут быть оптимальными. Если в одно и то же состояние входят два пути, выбирается тот, который имеет лучшую метрику; такой путь называется *выживающим*. Отбор выживающих путей выполняется для каждого состояния. Таким образом, декодер углубляется в решетку, принимая решения путем исключения менее вероятных путей. Предварительный отказ от маловероятных путей упрощает процесс декодирования. В 1969 году Омуре (Omura) показал, что основу алгоритма Витерби составляет оценка максимума правдоподобия. Отметим, что задачу отбора оптимальных путей можно выразить как выбор кодового слова с *максимальной метрикой правдоподобия* или *минимальной метрикой расстояния*.

Алгоритм Витерби является достаточно мощным алгоритмом декодирования, при этом сложность его аппаратной реализации невысока. В настоящее время алгоритм Витерби применяется во многих стандартах беспроводной связи, таких как IEEE 802.11a/g, WiMAX, DAB/DVB, WCDMA, GSM.

Наиболее подходящей элементной базой для аппаратной реализации декодера Витерби являются программируемые логические интегральные схемы (ПЛИС).

### **Работа декодера Витерби**

Алгоритм Витерби находит широкое применение и реализует *поиск максимально правдоподобного пути* на кодовой решетке с отбрасыванием части наименее правдоподобных вариантов путей на каждом шаге декодирования.

Алгоритм Витерби характеризуется постоянством вычислительной работы, однако *сложность* декодера Витерби растет, как при всех переборных алгоритмах, *по экспоненциальному закону* от длины кодового ограничения сверточного кода. Поэтому алгоритм Витерби используется для декодирования коротких сверточных кодов.

*Алгоритм заключается в повторении одного основного шага*. На каждой из последующих диаграмм рис. 3.91 этот шаг изображен подробно.

*Метрика пути* (МП) есть сумма метрик ветвей, образующих некоторый путь на решетчатой диаграмме. Путь конечной длины оканчивается в определенном состоянии. Метрика состояния (МС) равна метрике пути, который заканчивается в данном состоянии.

*Шаг декодирования* состоит в обработке декодером принимаемых из канала данных в интервале между двумя соседними отсчетами.

На рисунке показано развитие процесса декодирования символов СК со скоростью  $R_{\text{код}} = 1/2$  и длиной кодирующего регистра  $K = 3$ . На вход декодера поступают пары символов из канала: (...11,10,00,11,01...) (декодирование с жестким решением). Цифрами около ветвей обозначены метрики ветвей, цифры в кружках обозначают метрики состояний.

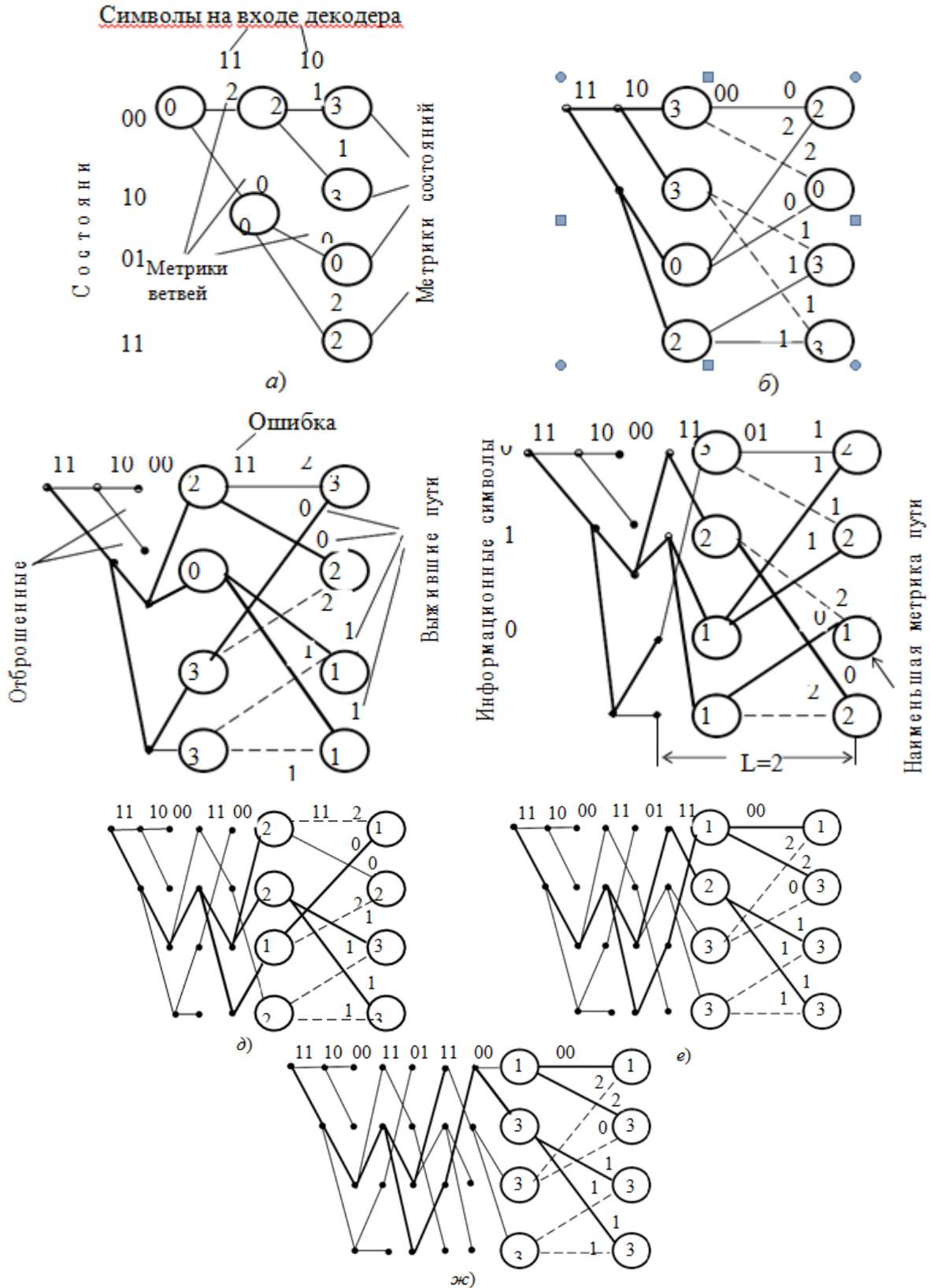


Рис. 3.91. Декодирование кода(7,5)

К каждому новому состоянию ведут два пути. К примеру, к состоянию 00 ведут пути из предыдущих состояний 00 и 01. На  $i$ -м шаге декодирования декодер вычисляет метрики путей как суммы метрик предыдущих состояний и метрик входящих ветвей.

Далее производится попарное сравнение метрик путей, входящих в каждое из состояний (пары показаны фигурными скобками). В результате сравнения *выбирается меньшая метрика*, и она считается метрикой данного состояния для последующего шага декодирования. *Путь*, входящий в данное состояние с меньшей метрикой, считается *выжившим*. На рис. 3.91 отрезки выживших путей показаны сплошной линией. Пути, входящие в состояния с большими метриками, считаются отмершими (оборванными). Они показаны на решетчатой диаграмме пунктиром.

Таким образом, на каждом шаге декодирования в соответствии с алгоритмом Витерби, *в каждом из состояний* решетчатой диаграммы производятся *однотипные операции*:

1) *Сложение метрик* предыдущих состояний с метриками соответствующих ветвей.

2) *Сравнение метрик* входящих путей.

3) *Выбор путей* с наименьшими метриками, величины которых используются как метрики состояний на последующем шаге декодирования. Если метрики сравниваемых путей одинаковы, то выбор одного из двух путей производится случайным образом.

На каждом шаге декодирования половина возможных продолжений путей отбрасывается. Другая половина образует продолжения путей для следующего шага декодирования, на котором вновь появляются два варианта продолжения каждого пути. Это обеспечивает *постоянство количества вычислений*, производимых на каждом шаге. *Декодер прослеживает* по кодовой решетке *путь*, имеющий *минимальное расстояние* от пути, который порождает кодер.

Таким образом, декодер, выбирающий на решетчатой диаграмме путь с наименьшей метрикой, *минимизирует вероятность ошибки*. Поскольку при декодировании анализу подвергаются последовательности конечной длины  $L$ , алгоритм не является строго оптимальным. Результаты расчетов и моделирования показывают, что при соответствующем выборе величины  $L > (6...7)v$  можно получить результаты декодирования, достаточно близкие к оптимальным. *Сложность реализации алгоритма Витерби* для декодирования СК можно оценить по количеству ветвей кодовой решетки,

обрабатываемых декодером на длине декодирования  $L$ , с учетом сложности каждого шага решетки.

Декодер состоит из АЦП в каналах  $X$  и  $Y$ , вычислителя метрик ветвей, процессора, в котором производятся операции сложения, сравнения и выбора, устройства памяти путей, которые выжили, и мажоритарного элемента МЭ, в котором выбирается путь с наибольшей метрикой. Оптимальное значение шага квантования зависит от отношения сигнал/шум на входе АЦП. При восьми уровнях квантования минимум потерь обеспечивается при отношении размаха сигнала к шагу квантования, равном (4,5...5,5).

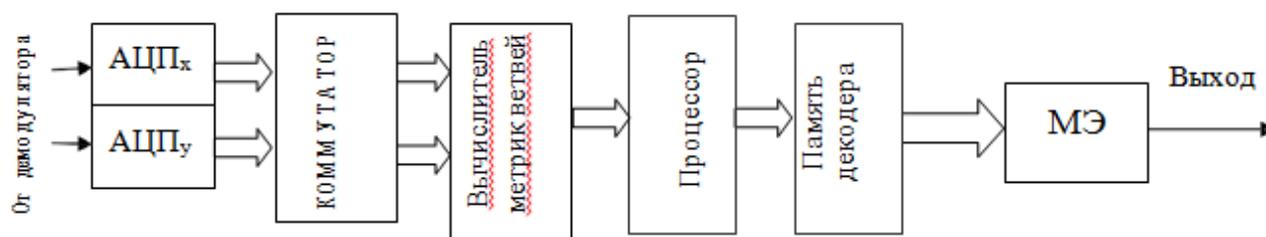


Рис. 3.92. Структурная схема декодера Витерби

### **Разработка программного интерфейса для исследования характеристик и визуализации основных преобразований.**

На данном этапе необходимо реализовать и проанализировать описанный выше алгоритмы декодирования Витерби на практике в программной среде Matlab 2015. Необходимо построить соответствующие схемы, которые будут в себя включать источник сообщения, кодер, декодер, канал передачи, анализатор ошибок и соответственно устройства визуализации полученных результатов.

**Индивидуальности сверточного кодировки.** Библиотека Communications Blockset поддерживает разомкнутые либо бинарные замкнутые коды, которые могут быть описаны структурой сетки либо рядом порождающих полиномов. Это употребляет метод Viterbi, чтоб выполнить твердое и мягкое декодирование. Библиотека также включает декодер, максимизирующий апостериорную возможность, который может употребляться для мягкого декодирования сверточных кодов.

**Характеристики сверточного кодировки.** Процессы сверточного кодировки употребляют блоки Convolutional Encoder, Viterbi Decoder, и/либо APP Decoder из библиотеки Convolutional. Если параметр маски требуется и в кодирующем устройстве и в декодере, используйте одно и то же значение в обоих блоках. Блоки в подбиблиотеке Convolutional подразумевают, что используется один из 2-ух разных представлений сверточного кодера: если проектируется кодирующее устройство, используя диаграмму со

сдвигowymi регистрами и сумматорами по модулю 2, можно вычислить порождающую полиномиальную матрицу кода и потом применять функцию `poly2trellis` (из Communications Toolbox), чтоб произвести подобающую решетчатую (trellis) диаграмму маски параметра автоматом. Если проектируется кодирующее устройство, используя решетчатую диаграмму, можно выстроить решетчатую (trellis) диаграмму в MATLAB и применять его в качестве параметра маски.

**Внедрение полиномиального описания в блоках.** Для использования полиномиального описания в блоках Convolutional Encoder, Viterbi Decoder, либо APP Decoder используется сервисная функция `poly2trellis` из Communications Toolbox. Эта функция воспринимает полиномиальное описание и преобразовывает это в решетчатое описание. К примеру, последующая команда вычисляет решетчатое описание кодирующего устройства, длина ограничения которого равна 5 и чьи порождающие полиномы 35 и 31: `trellis = poly2trellis (5, [35 31])`

**Определение характеристик кодировки.** Блоки Convolutional Encoder и Viterbi Decoder могут выполнить этот код, если у их характеристик установлены надлежащие значения. Длина ограничения кодирующего устройства - вектор длины 2, потому кодирующее устройство имеет два входа. Элементы этого вектора указывают на число сохраненных битов в каждом сдвиговом регистре, включая текущие входные биты. Подсчет ячеек памяти в каждом сдвиговом регистре на диаграмме и добавлении того для текущих входов приводят к длине ограничения, равного [5 4]. Определяя порождающую матрицу кода как матрицу размера 2 X 3 восьмеричных чисел, используйте элемент в  $i$  - ой строке и  $j$  - ом столбце, чтоб указать, как  $i$  - ый вход повлияет на  $j$  - ый выход. К примеру, чтоб вычислить элемент во 2-ой строке и 3-ем столбце, заметьте, что последнее левое и два самых правых элемента во 2-м сдвиговом регистре диаграммы влияют на сумму, что сформировывает 3-ий выход. Эта информация в двоичном коде равна 1011, что эквивалентно восьмеричному числу 13. Полная порождающая матрица кода равна [27 33 0; 0 5 13]. Для использования длины кодового ограничения и характеристик порождающей матрицы кода в блоках Convolutional Encoder and Viterbi Decoder, примените функцию `poly2trellis`, чтоб преобразовать эти характеристики в решетчатую диаграмму.

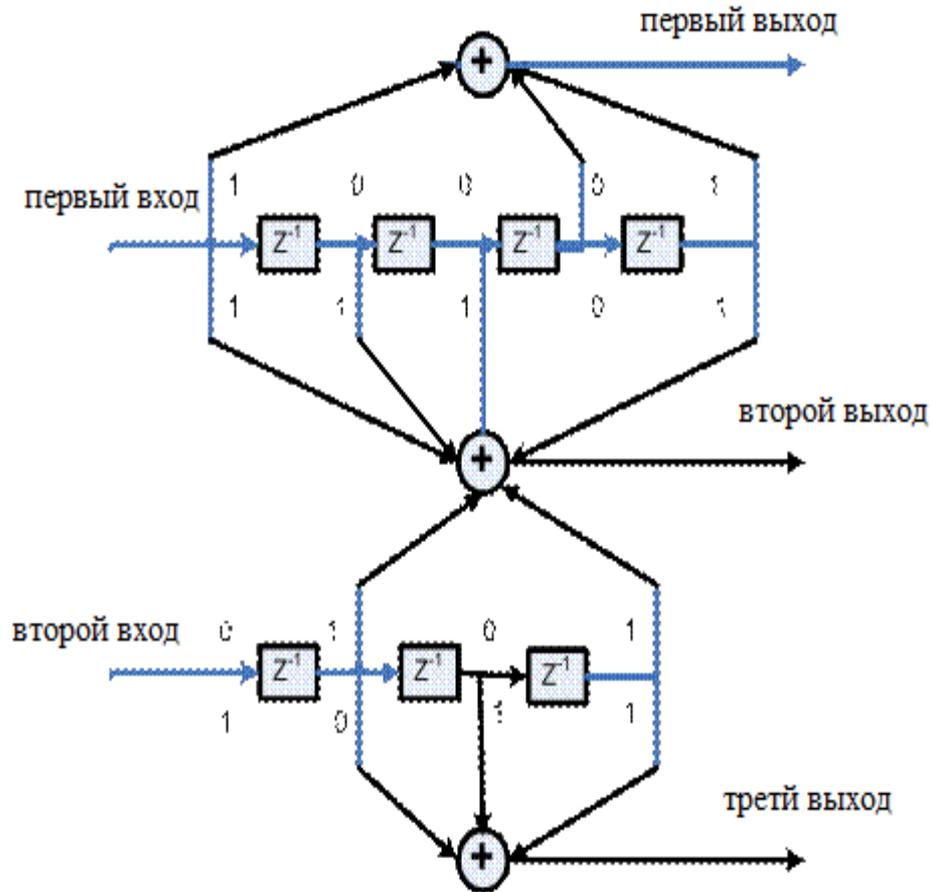


Рис. 3.93. Модель кодирующего устройства

### Моделирование кодера Витерби

На рисунке представлена модель кодирующего устройства.

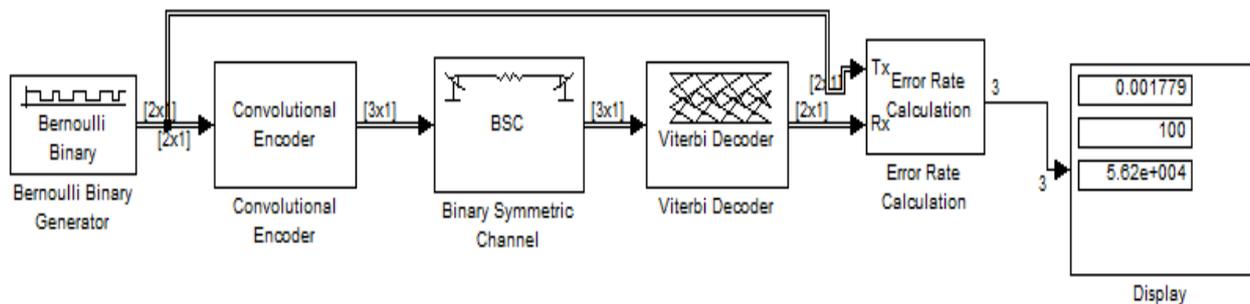
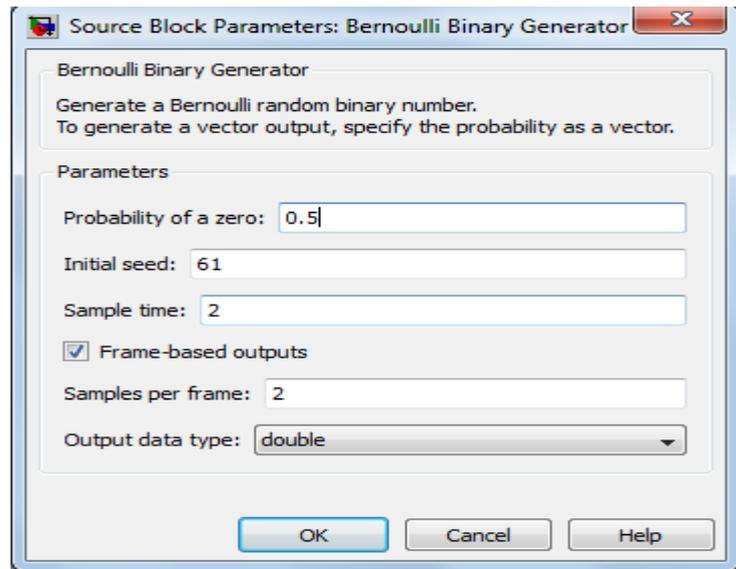


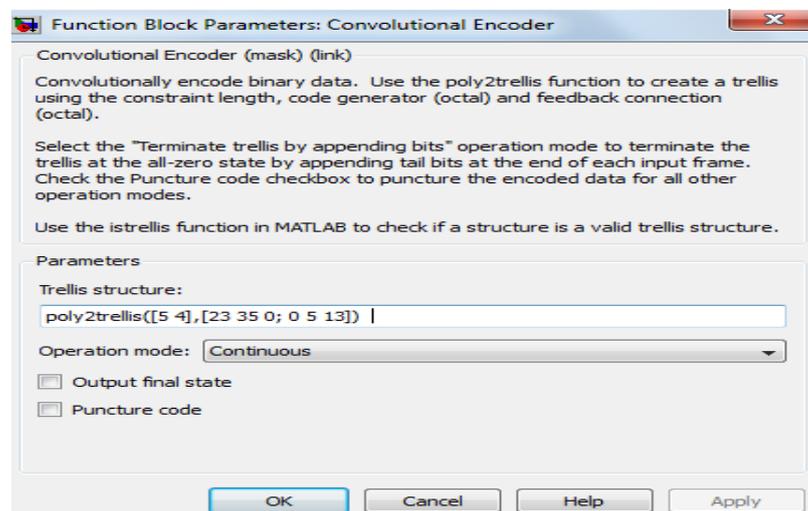
Рис. 3.94. Линия передачи с применением декодера Витерби

Чтоб выстроить модель, соберите и конфигурируйте блоки.

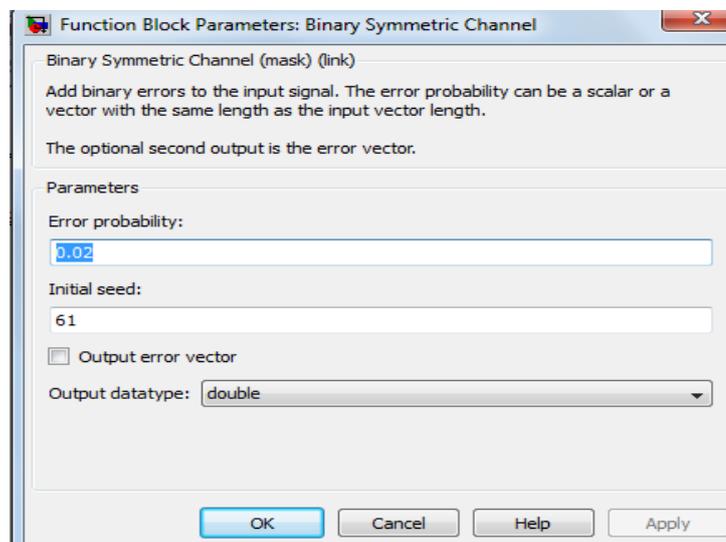
1. Блок Bernoulli Binary Generator из библиотеки Comm Sources.



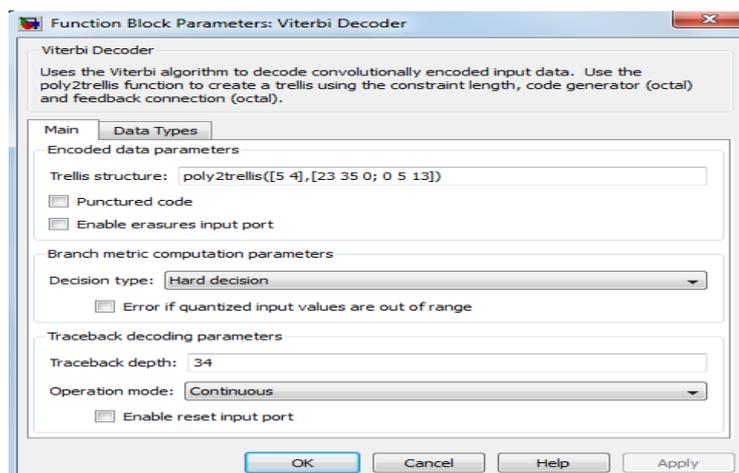
## 2. Блок Convolutional Encoder



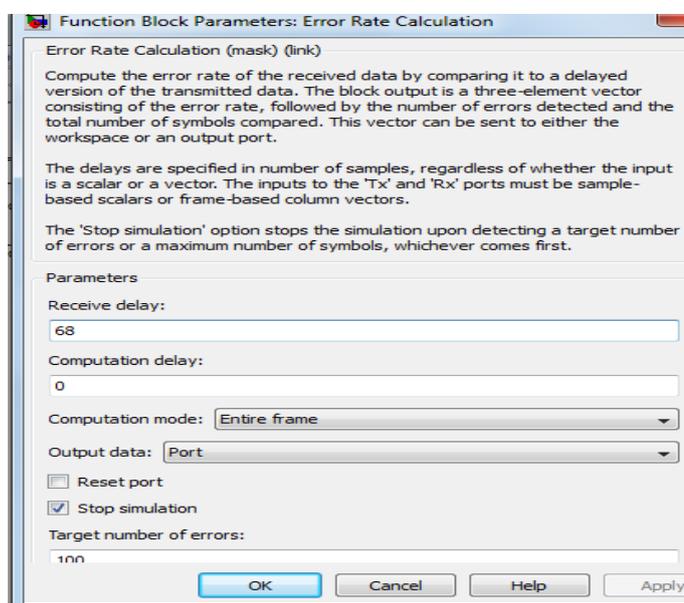
## 3. Блок Binary Symmetric Channel из библиотеки Channels



## 4. Блок Viterbi Decoder



## 5. Блок Error Rate Calculation из библиотеки Comm Sinks.



6. Блок Display из библиотеки Simulink Sinks. Потяните базисный край знака, чтоб сделать показ, довольно огромным для 3-х записей. Соедините блоки как на рисунке. Из меню Simulation, выберите Configuration parameters. В диалоговом окне установите параметр Stop time, равным inf.

Инструкции по размерам матрицы возникают на соединительных линиях, лишь если выбрать параметр Signal Dimensions подменю Port/signal displays из меню Format модели.

Кодирующее устройство воспринимает вектор кадра размера  $2 \times 1$  и производит вектор кадра размера  $3 \times 1$ , в то время как декодер делает обратное. Параметр Samples per frame parameter в блоке Bernoulli Binary Generator принят равным 2, так как блок должен произвести слово сообщения длины 2. Параметр Receive delay в блоке Error Rate Calculation принят равным 68, который является длиной вектора (2) восстановленного сообщения значения Traceback depth (34) в блоке Viterbi Decoder. Если исследуется переданные и

приобретенные сигналы, как матрицы в рабочем пространстве MATLAB, Вы видите, что 1-ые 34 строчки восстановленного сообщения состоят из нулей, в то время как следующие ряды - расшифрованные сообщения. Таким образом, задержка приобретенного сигнала - 34 вектора длины 2, либо 68 отсчетов. Пуск модели производит выход монитора, состоящий из 3-х чисел: скорость роста ошибок, общее количество ошибок и общее количество ошибок сравнений, которые делает блок Error Rate Calculation во время моделирования. (1-ые два числа меняются в зависимости от Ваших значений Initial seed в блоках Bernoulli Binary Generator и Binary Symmetric Channel). Остановка моделирования после 100 ошибок происходит, так как параметр Target number of errors установлен равным 100, в блоке Error Rate Calculation. Скорость роста ошибки обязана быть меньше 0,02, как Error probability в блоке Binary Symmetric Channel.

**Мягкое декодирование.** Моделирование делает случайный сигнал двоичного сообщения, кодирует сообщение в сверточный код, модулирует код, используя двоичную манипуляцию со смещением фазы, (BPSK) и добавляет белый Гауссовский шум к модулированному сигналу, чтоб моделировать шум канала. Тогда, моделирование готовит полученную информацию для блока декодирования. В конце концов, моделирование ассоциирует декодированную информацию с сигналом начального сообщения, чтоб вычислить скорость роста ошибок. Конец моделирования наступает после обработки 100 битов ошибок либо 107 битов сообщения, после прибытия первого.

Длина ограничения кодирующего устройства - скаляр, потому что у кодирующего устройства есть один вход. Значение длины ограничения - число битов, сохраненных в сдвиговом регистре, включая текущий вход. Есть 6 регистров памяти, и текущий вход составляет один бит. Таким образом, длина ограничения кода - 7. Кодовый генератор является матрицей размера 1 x 2 матрица восьмеричных чисел, так как кодирующее устройство имеет вход и два выхода, 1-ый элемент в матрице показывает на значение входа, влияющего на 1-ый выход, и 2-ой элемент в матрице показывает, на значение входа, влияющего на 2-ой выход.

К примеру, 1-ый выход в диаграмме кодирующего устройства – сумма по модулю 2 самого правого и 4 последних левых частей в массиве входных значений диаграммы. Двоичное число с семью цифрами 1111001 отражает эту информацию и эквивалентно восьмеричному числу 171. Восьмеричное число 171, таким образом, становится первым входом порождающей кодовой матрицы. Тут, любая тройка употребляет последний левый бит как самый значимый бит. 2-ой выход соответствует двоичному числу 1011011, который эквивалентен восьмеричному номеру 133. Порождающий генератор кода потому [171 133].

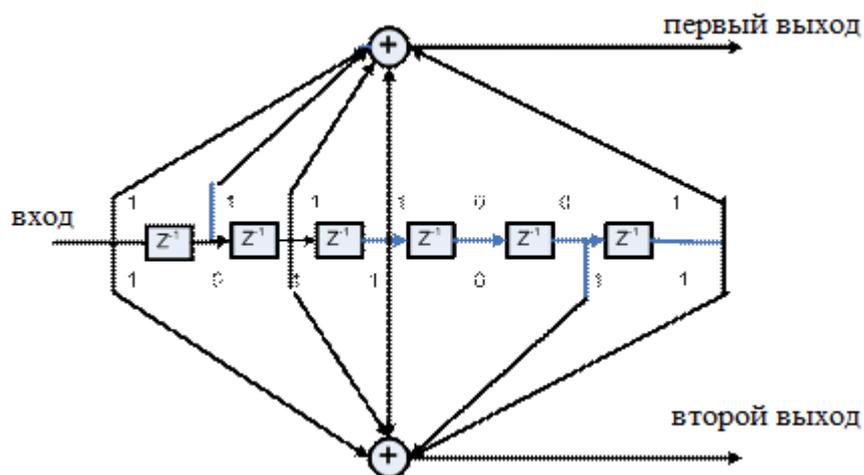


Рис.3.95. Модель кодирующего устройства

Параметр Trellis structure в блоке Convolutional Encoder говорит о том, какой блок, кодирует, когда поступают данные. В этом варианте, функция poly2trellis, в Communications Toolbox, преобразовывает длину кодового ограничения и пару восьмеричных чисел в решетчатую диаграмму. В то время как данные о сообщении, входящие в блок Convolutional Encoder, являются скалярным потоком битов, закодированные данные являются потоком бинарных векторов длиной 2.

**Карта приобретенных данных.** Приобретенные данные, другими словами, выход блока Convolutional Encoder, состоят из комплексных чисел, которые находятся в спектре меж -1 и 1. Чтоб вернуть оригинальное двоичное сообщение, приемная часть модели обязана декодировать сверточный код. Блок Viterbi Decoder в данной модели ждет, что ее входные данные будут целые числа меж 0 и 7. Демодулятор, рядовая подсистема в данной модели, преобразовывает приобретенные данные в формат, который блок Viterbi Decoder может интерпретировать подабающим образом. Наиболее точно, подсистема демодулятора: Конвертирует приобретенные данные в настоящий сигнал, удаляя его мнимую часть. Уместно представить, что мнимая часть приобретенных данных не содержит существенную информацию, так как мнимая часть переданных данных – есть ноль (игнорирование малых ошибок округления) и так как шум канала не чрезвычайно мощнейший. Восстанавливает приобретенные данные, деля на его текущее обычное отклонение и потом умножая на -1. Квантует нормализованные данные, используя три бита. Композиция данной карты и решения карты блока Viterbi Decoder на сто процентов изменяет модуляцию BPSK, чтоб блок BPSK Modulator Baseband выступил на передающей стороне данной модели. Декодирование сверточного кода. После получения данных подабающим образом нанесенных на карту векторов длиной 23-битовых значений решения, блок Viterbi Decoder

декодирует его. Блок употребляет метод мягенького декодирования с 23 разными входами поэтому, что параметр Decision type - есть Soft Decision и параметр Number of soft decision bits равен 3. Интерпретация данных мягенького решения. Когда характеристики Decision type установлен в состояние Soft Decision, блок Viterbi Decoder просит входных значений меж 0 и  $2b-1$ , где  $b$  – есть параметр Number of soft decision bits. Блок интерпретирует 0 как самое достоверное решение того, что бит кодового слова, является 0 и интерпретирует  $2b-1$  как самое достоверное решение того, что бит кодового слова, является 1. Значения меж этими крайностями представляют наименее достоверные решения.

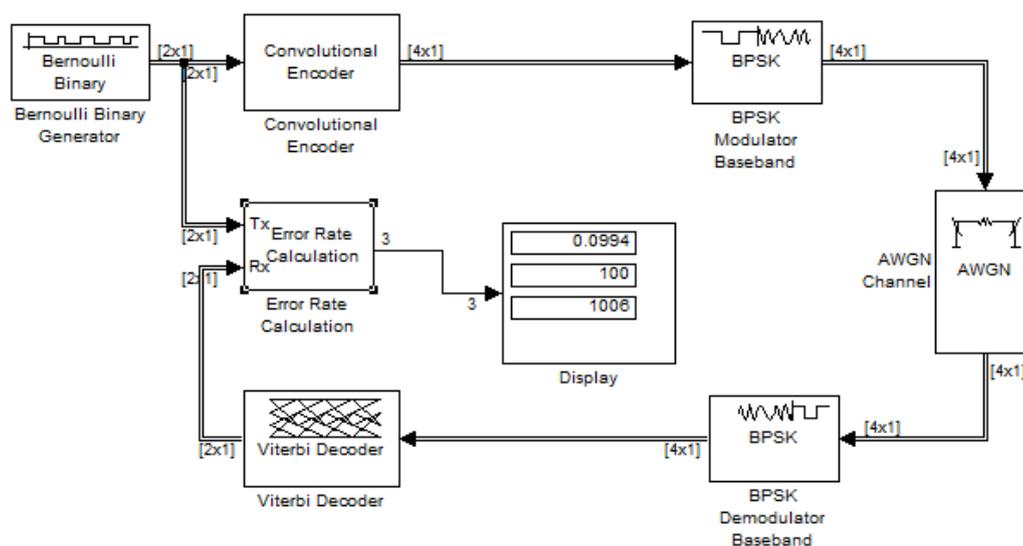


Рис. 3.96. Схема декодера Витерби в MATLAB

	Вер-ть ошибки 0,02	Вер-ть ошибки 0,03
Код (5, [35 31])	скорость роста ошибок: 0.5435 общее количество ошибок: 100 общее количество ошибок сравнений: 184	скорость роста ошибок: 0.5435 общее количество ошибок: 100 общее количество ошибок сравнений: 184
Код ([5 4],[23 35 0; 0 5 13])	скорость роста ошибок: 0,001779 общее количество ошибок: 100 общее количество ошибок сравнений: 5.62e+0.04	скорость роста ошибок: 0,01489 общее количество ошибок: 100 общее количество ошибок сравнений: 6714

### Блоки программы декодера Витерби, реализованного в Matlab [21]

В блоке формирования узлов сетки выполняются следующие функции:

1. Формируется матрица фрагмента решетчатой диаграммы размера  $n \times m$ , в которой количество столбцов соответствует числу состояний кодера (узлов), а количество строк равно числу путей, подходящих к каждому узлу. Каждому элементу матрицы соответствуют ответвляемые слова при переходах между состояниями. Переходы между состояниями повторяются на протяжении всей решетки.

2. Формируется матрица переходов между состояниями.

3. Каждому переходу к конкретному узлу соответствует определенный входной бит (единица, либо ноль). Формируем матрицу размера  $(n \times m)$  состоящую только из нулей и единиц .

4. Для каждого входного дибита (слова) декодера вычисляются все возможные значения метрик по фрагменту решетчатой диаграммы кодера, сформированной в предыдущем блоке. В каждый момент времени происходит операция сложения по модулю два входного дибита с соответствующим данному моменту времени ответвляемым словом. Для исключения нижней части решетчатой диаграммы в начальные моменты времени мы искусственно увеличиваем значение метрик переходов нижней части решетки.

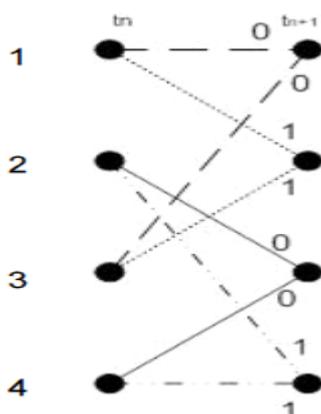


Рис. 3.97. Фрагмент решетчатой диаграммы декодера

Блок удаления путей:

1. В каждом узле, сравнивая метрики входных путей, обнуляем путь с наибольшей метрикой, тем самым убирая его из рассмотрения, но при этом могут возникнуть «тупиковые» пути.

2. Для удаления «тупиковых» путей, поочередно перебирая возможные  $m$  состояний в данный момент времени ищем отсутствие какого-либо элемента матрицы переходов состояний, если элемент отсутствует, то пути в предыдущих фрагментах решетки (матрицы) удаляются.

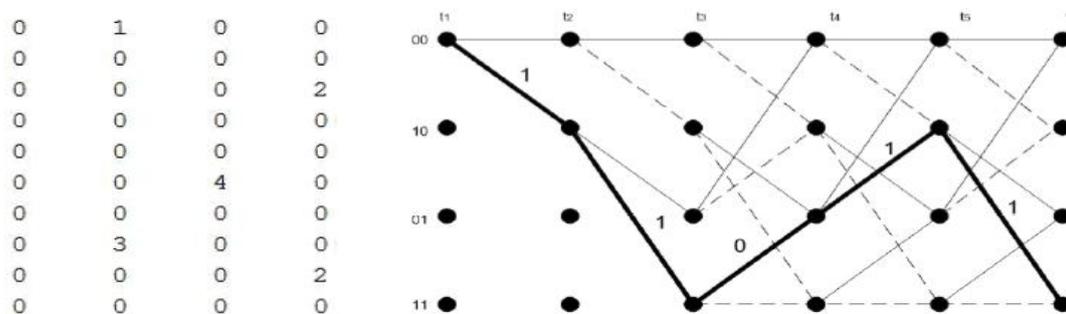


Рис. 3.98. Декодирование по выделенному пути

Блок декодирования:

Последовательно проверяем количество ненулевых переходов в  $n$  строках. Если такой переход единственный, то находим позицию этого элемента, и определяем какой бит был подан на вход кодера

### Вопрос 1. Исследование методов построения кодеров непрерывных кодов.

В рабочем поле необходимо собрать схему для работы сверточного кода (5, [35 31]).  
Схема представлена на рисунке 3.99

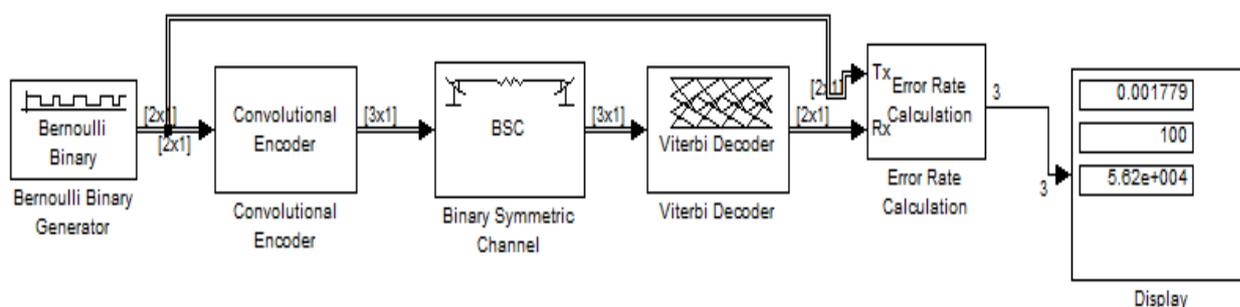


Рис. 3.99. Линия передачи с применением декодера Витерби.

В состав линии с кодированием входят:

1. BernoulliBinaryGenerator
2. Convolutional Encoder
3. Binary Symmetric Channel (каналпередачи)
4. Viterbi Decoder

5. Error Rate Calculation (анализаторошибок)

6. Display

Устанавливаем характеристики блоков для кода (5, [35 31])

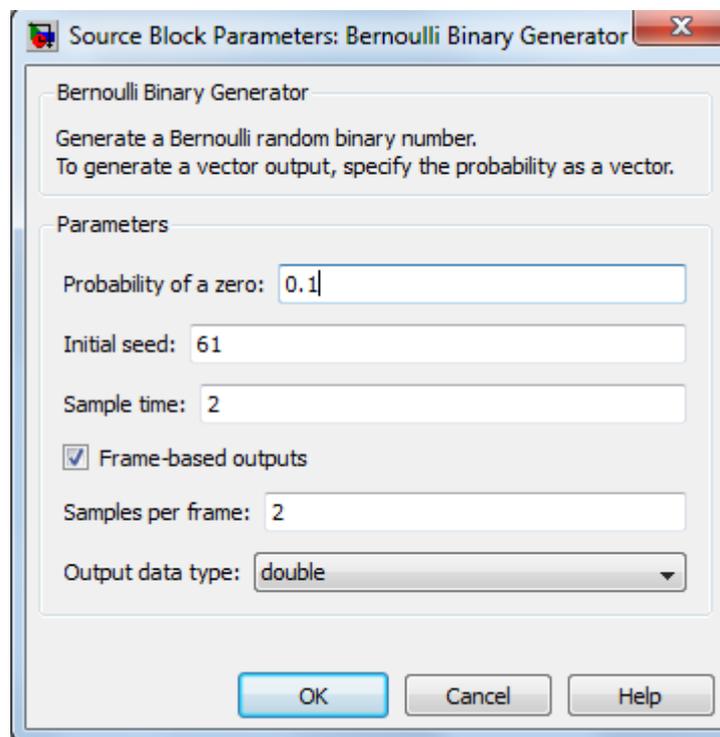


Рис. 3.102. Параметры Bernoulli Binary Generator

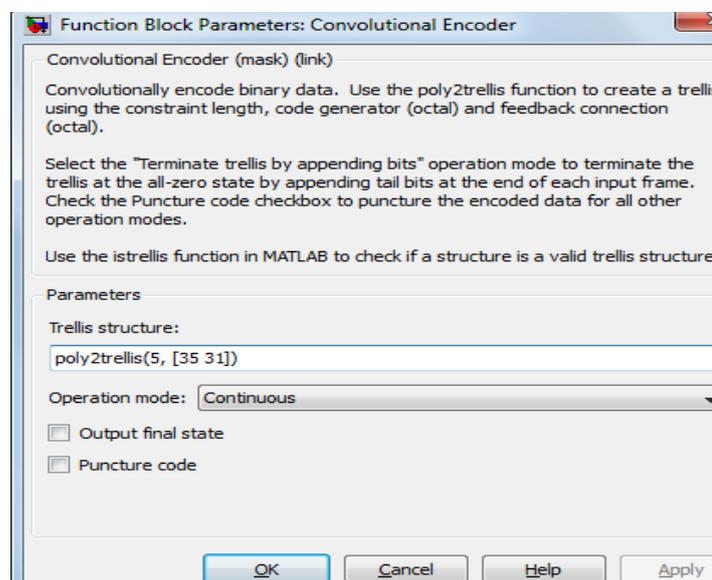


Рис. 3.103. Convolutional Encoder

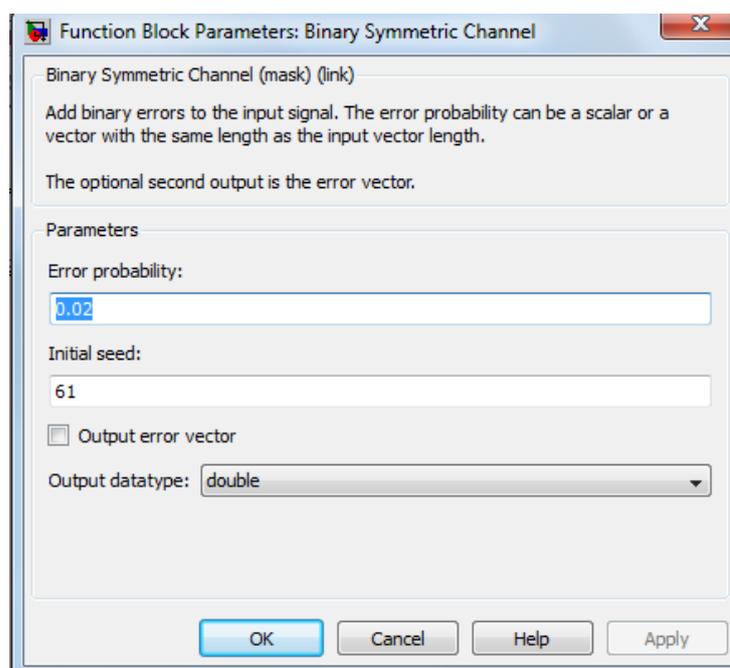


Рис. 3.104. Binary Symmetric Channel

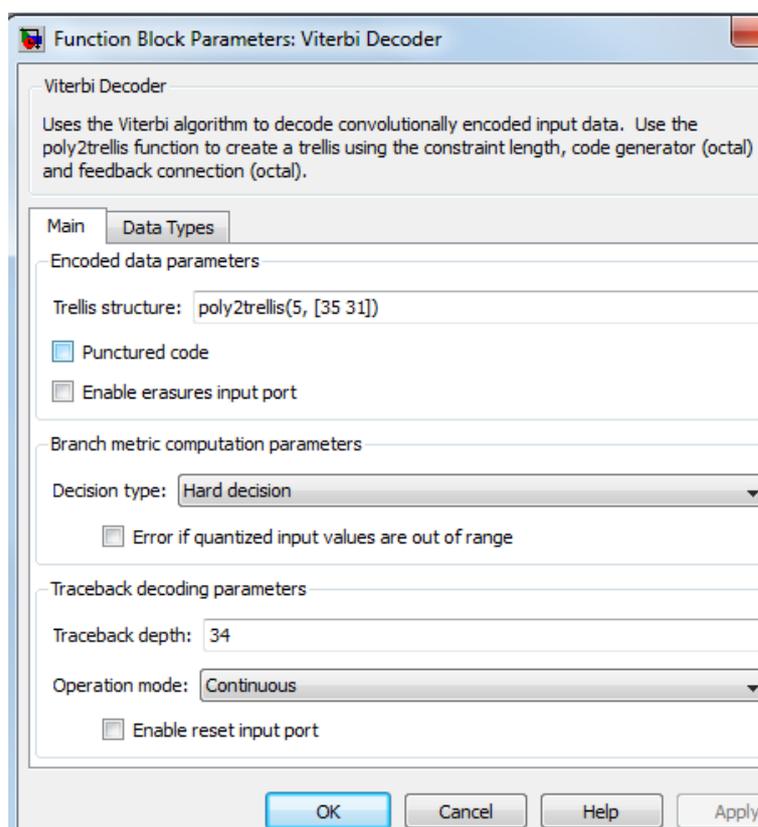


Рис. 3.100 Viterbi Decoder

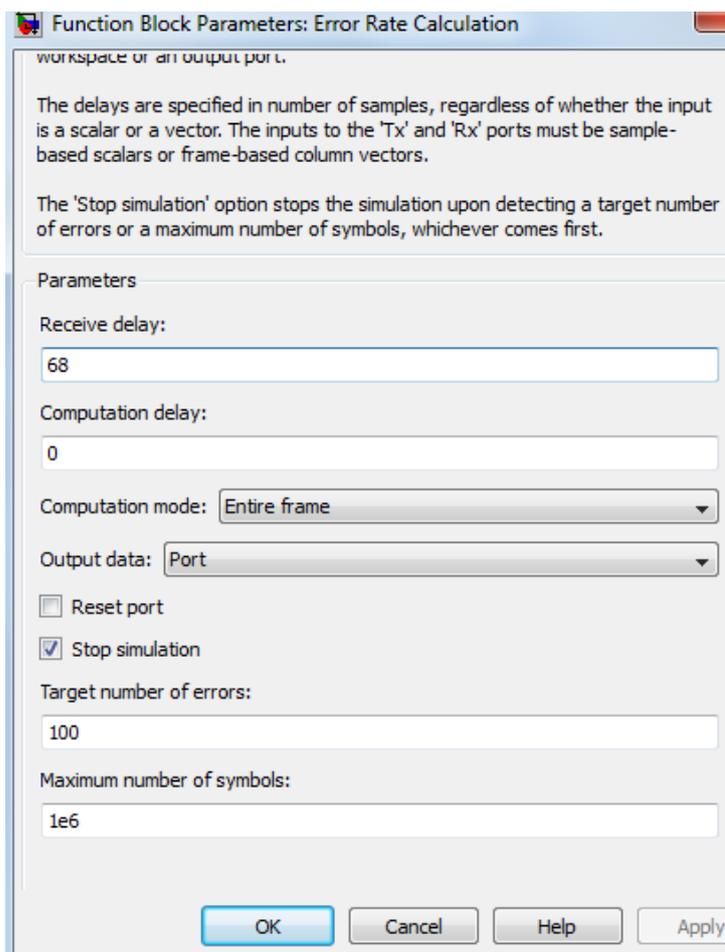


Рис. 3.101. Параметры Error Rate Calculation



Рис. 3.102. Комбинация на входе, Закодированная последовательность, Комбинация на выходе, Ошибки (вероятность ошибок равна 0,02)

Анализируя рисунок выше, можно сделать вывод, что комбинация на входе совпадает с комбинацией на выходе, таким образом, передача осуществилась удачно. Что касается

ошибок, то их частота равна 0,1799, число ошибок равно 100, общее количество символов по сравнению равно 556.

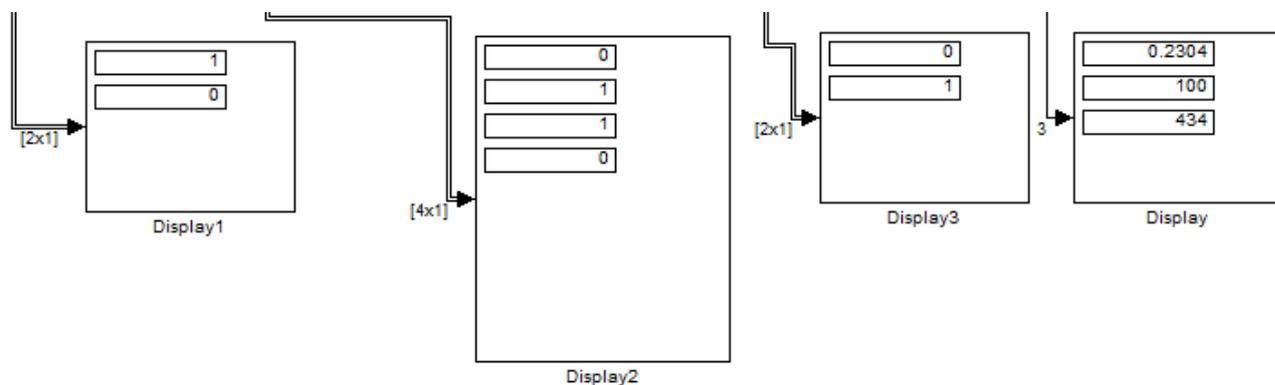


Рис. 3.103. Комбинация на входе, Закодированная последовательность, Комбинация на выходе, Ошибки (вероятность ошибок равна 0,08)

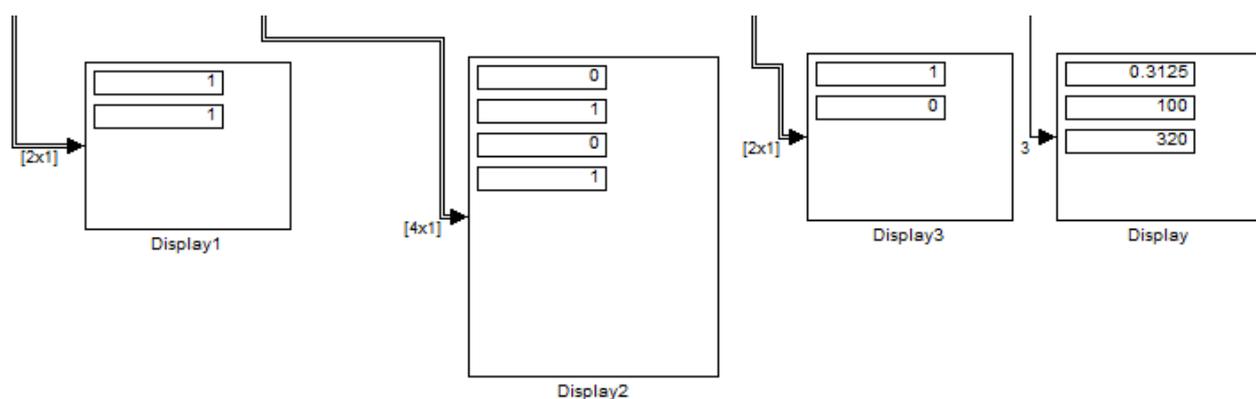


Рис. 3.104. Комбинация на входе, Закодированная последовательность, Комбинация на выходе, Ошибки (вероятность ошибок равна 0,12)

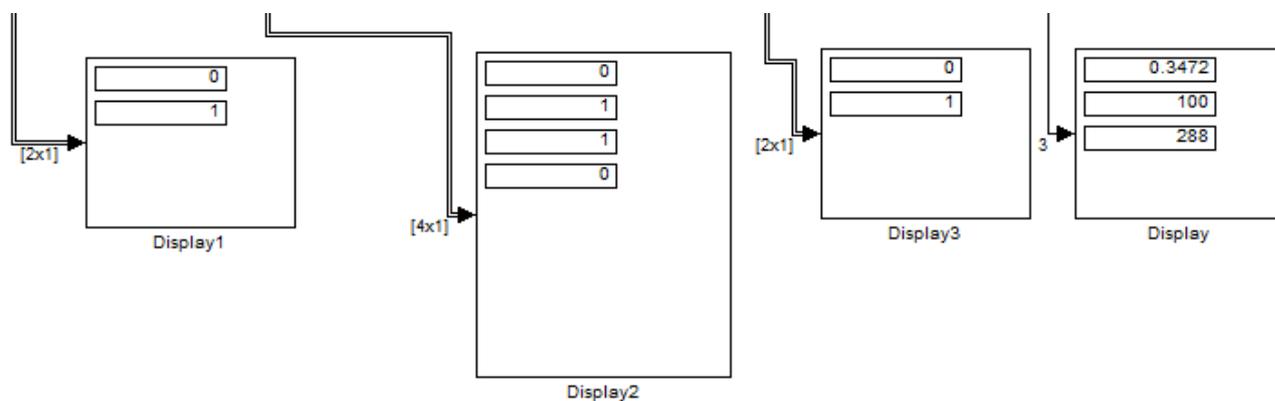


Рис. 3.105. Комбинация на входе, Закодированная последовательность, Комбинация на выходе, Ошибки (вероятность ошибок равна 0,14)



Рис. 3.106. График зависимости общего количества ошибок сравнений от вероятности ошибки для кода (5, [35 31])

Устанавливаем характеристики блоков для кода ([5 4],[27 33 0; 0 5 13])

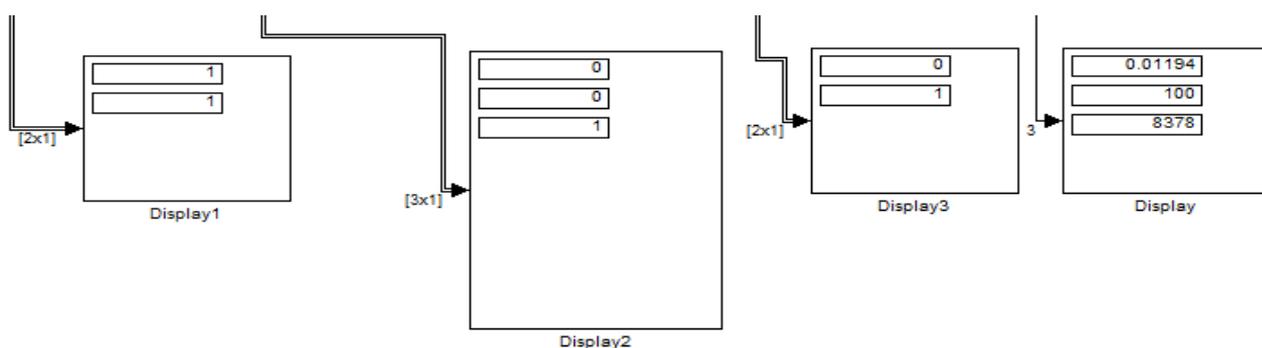


Рис. 3.107. Комбинация на входе, Закодированная последовательность, Комбинация на выходе, Ошибки (вероятность ошибок равна 0,02)

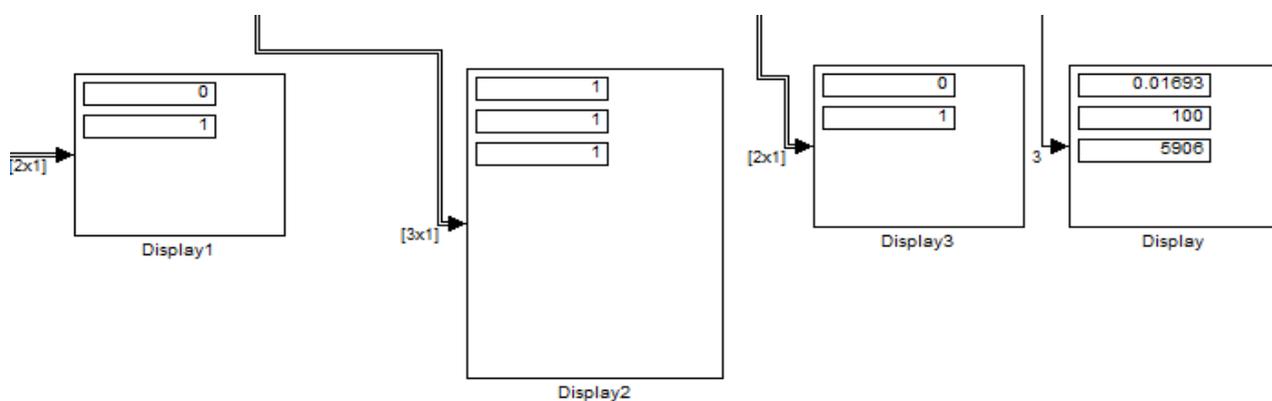


Рис. 3.108. Комбинация на входе, Закодированная последовательность, Комбинация на выходе, Ошибки (вероятность ошибок равна 0,025)

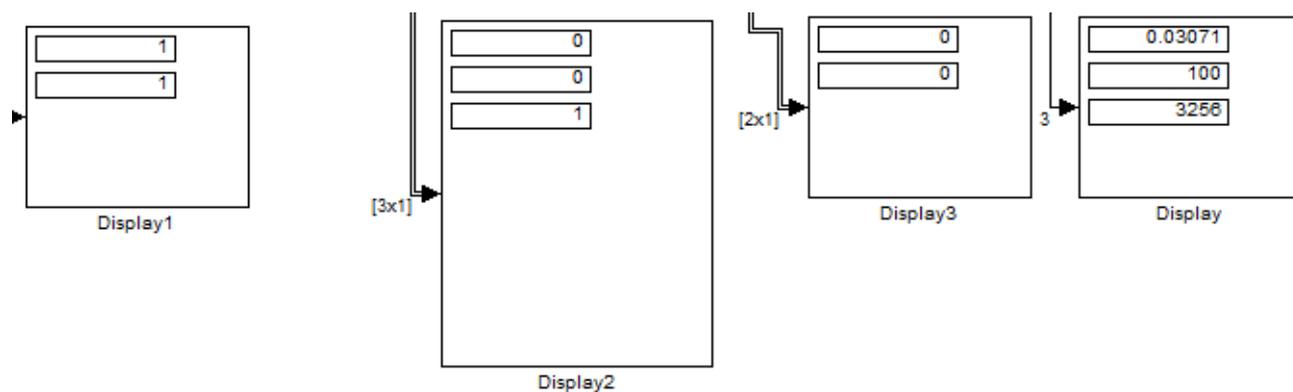


Рис. 3.109. Комбинация на входе, Закодированная последовательность, Комбинация на выходе, Ошибки (вероятность ошибок равна 0,03)

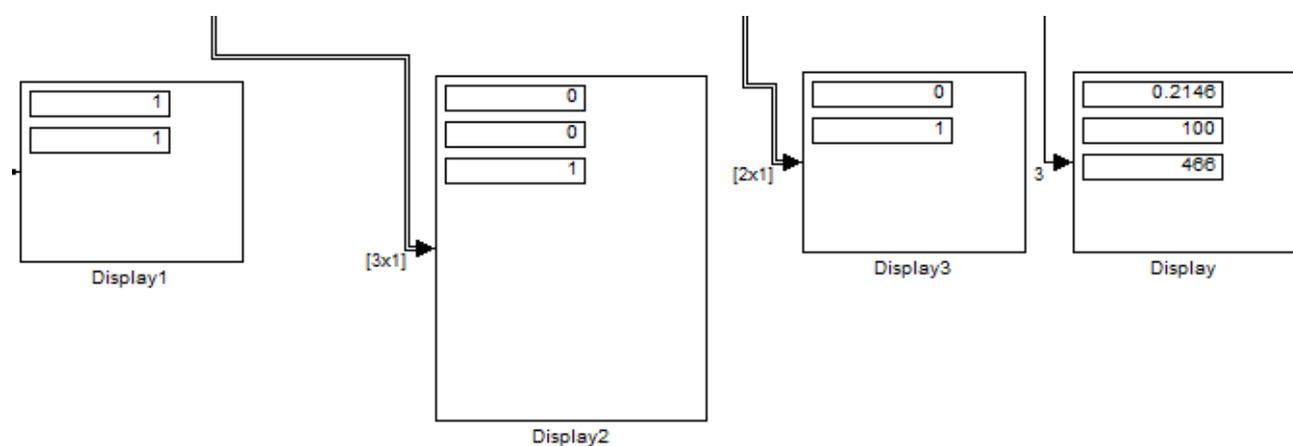


Рис. 3.110. Комбинация на входе, Закодированная последовательность, Комбинация на выходе, Ошибки (вероятность ошибок равна 0,035)

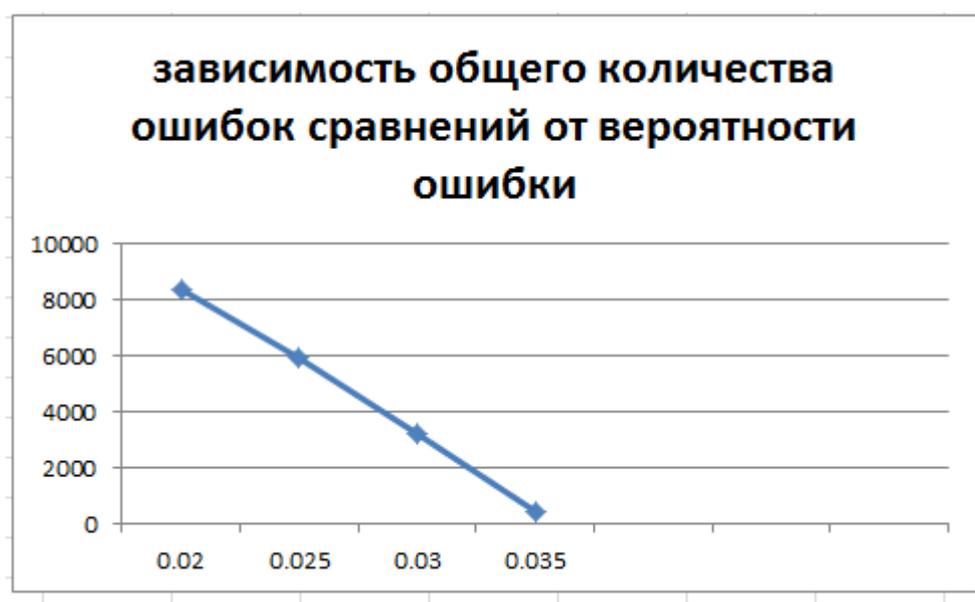


Рис. 3.111. График зависимости числа ошибок от вероятности ошибки для ([5 4], [27 33 0; 0 5 13])

## Вопрос 2. Построение схемы декодирования непрерывных кодов в канале с Гауссовскими помехами.

В рабочем поле необходимо собрать схему для работы сверточного кода (7, [171 133]).  
Схема представлена на рисунке 3.112

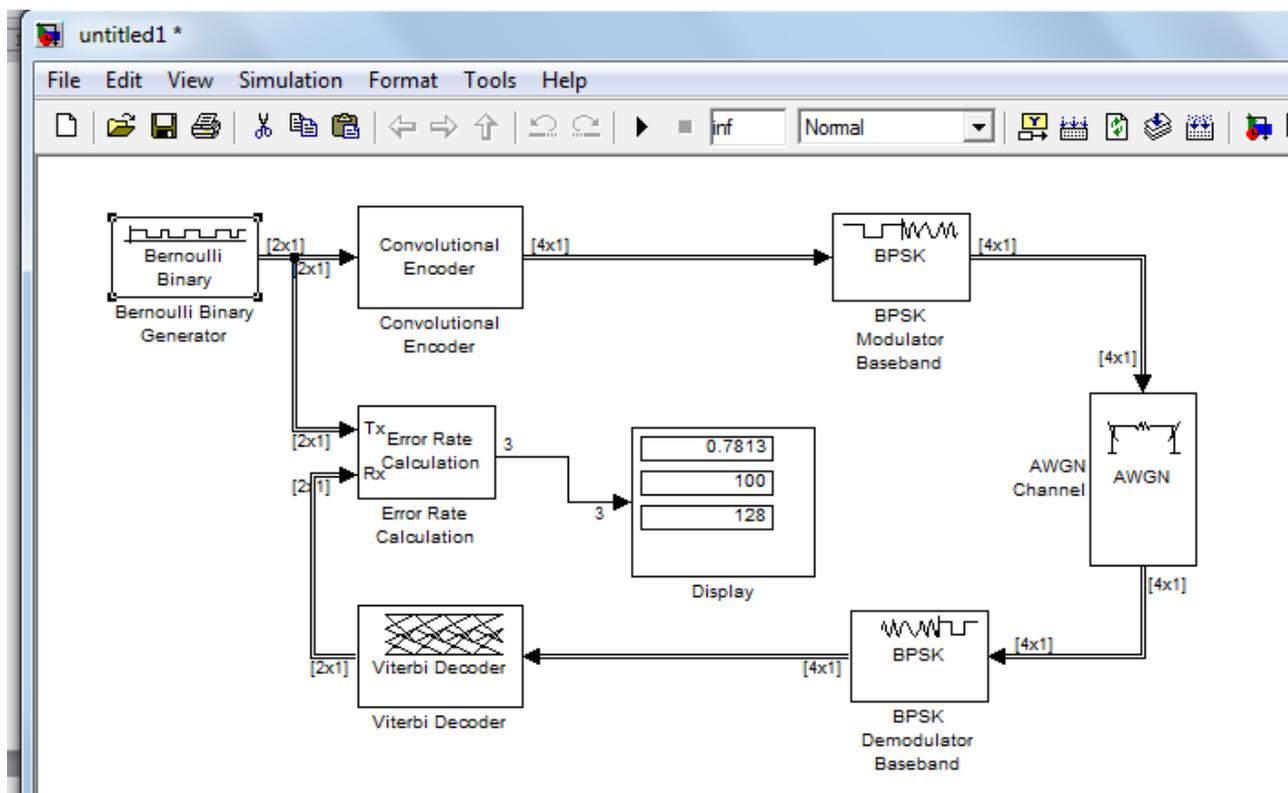


Рис. 3.112. Линия передачи с применением декодера Витерби

В результате практической работы построена схема линии передачи с декодированием Витерби в среде Simulink. Построены графики зависимостей общего количества ошибок сравнений на выходе декодера от вероятности ошибки в канале связи для кодов(5, [35 31]) и ([5 4],[27 33 0; 0 5 13]). Из графиков видно, что число ошибок сравнений уменьшается с ростом вероятности ошибок.

**3.5. Турбокодирование. Обобщенная схема турбокодера ТСС с параллельным каскадированием. Сверточные турбокоды. Декодирование турбокодов. Характеристики помехоустойчивости сверточных турбокодов ТСС. Исследование турбокодов с использованием ПО MATLAB.**

### Принципы построения турбо-кодов

В данном разделе приведены основные теоретические данные одного из важнейших классов итеративно декодируемых кодов – турбо-кодов (turbo codes) и с их обобщением –

многократными турбо-кодов (multiple turbo codes). Турбо-кодирование основано на двух фундаментальных идеях: построение кодов с кодовыми словами, обладающими квазислучайными свойствами и построение декодеров, основанных на легко реализуемых алгоритмах итеративного декодирования [1]. Турбо-коды обладают хорошими характеристиками, особенно при больших длинах кодов и умеренных требованиях к вероятности ошибки на бит.

### Параллельное каскадирование двух сверточных кодов

Турбо-код можно определить как параллельное каскадирование (parallel concatenation) двух сверточных кодов. Кодер состоит из двух параллельных систематических кодеров скорости  $R=1/2$  и памяти  $m=2$  (компонентные кодеры (constituent encoders)) с обратной связью (recursive encoders) имеющих рациональные порождающие подматрицы:

$$G = \left(1 + \frac{1 + D^2}{1 + D + D^2}\right) \quad (3.1)$$

и простого блочного перемежителя (ПБП). Входом турбо-кодера служит двоичная информационная последовательность длины  $K$

$$u = u_0 u_1 \dots u_{K-1} \quad (3.2)$$

Входом первого составного кодера и перемежителя, ассоциированного с вторым составным кодером, служит последовательность  $u$  длины  $K = K + m$ ,

$$u = u_0 u_1 \dots u_{K-1} \quad (3.3)$$

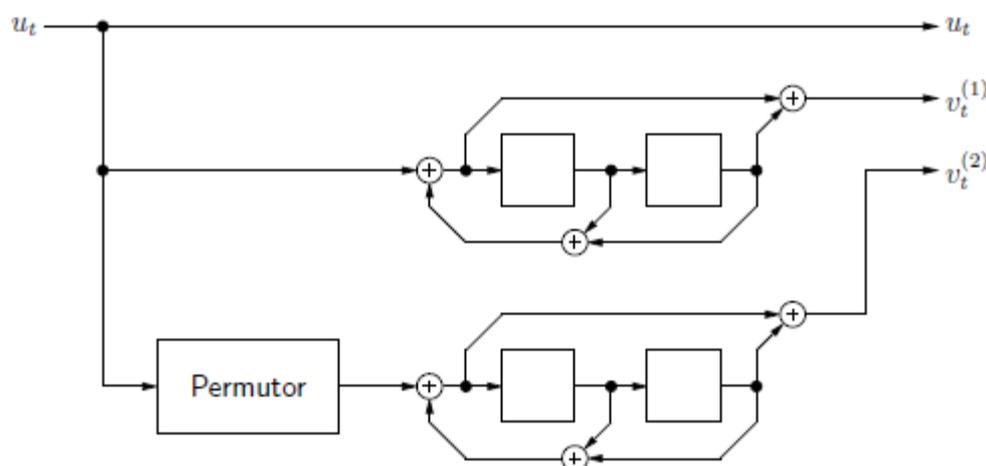


Рис. 3.113. Систематический сверточный кодер скорости  $R = 1/3$ , основанный на параллельном каскадировании двух сверточных кодеров

где  $m$  память составного кодера. Первые  $K$  символов этой последовательности – это символы информационной последовательности  $u$ , последние  $m$  символов образуют хвост,

приводящий первый компонентный кодер в нулевое состояние. Поскольку кодирование систематическое, входная последовательность (включая хвост) имеет вид:

$$v(0) = v_0^{(0)} v_1^{(0)} \dots v_{K-1}^{(0)} = u. \quad (3.4)$$

Первый компонентный кодер генерирует проверочную последовательность:

$$v(1) = v_0^{(1)} v_1^{(1)} \dots v_{K-1}^{(1)}. \quad (3.5)$$

Поскольку хвост заставляет первый компонентный кодер вернуться в нулевое состояние, мы можем использовать БКВР алгоритм апостериорно вероятностного декодирования кодовой последовательности первого компонентного кодера.

Параллельно последовательность  $u$  поступает в ПБП, определенный перестановочной матрицей  $P = (p_{ik})$ ,  $i = 0, 1, \dots, K-1$ ,  $k = 0, 1, \dots, K-1$  размеров  $(K \times K)$ . Выходная последовательность  $u(2) = uP$  ПБП поступает во второй компонентный кодер, генерирующий проверочную последовательность

$$v(2) = v_0^{(2)} v_1^{(2)} \dots v_{K-1}^{(2)} \quad (3.6)$$

Вообще говоря, второй кодер не переходит в конце кодирования в нулевое состояние. В этом случае нельзя использовать для декодирования БКДР алгоритм, но можно использовать однопутевой алгоритм апостериорно вероятностного декодирования. При больших длинах  $K$  это не сильно отражается на характеристиках декодирования и вызывает лишь их небольшую деградацию.

ПБП, использованный во втором кодере, перестановочной матрицей  $P$ . С помощью введения вектора перестановки индексов (indexpermutation vector) длины  $K$  можно дать альтернативное определение ПБП:

$$\pi = (\pi_0 \pi_1 \dots \pi_{K-1}) \quad (3.7)$$

где  $0 \leq \pi_i' \leq K-1$ ,  $i = 0, 1, \dots, K-1$ ,  $\pi_i' \neq \pi_j'$  если  $i \neq j$  и  $\pi_i'$  таково, что  $p_{\pi_i' i} = 1$ , то есть  $p_{\pi_i' i}$  – это единственный ненулевой элемент в  $i$ -ом столбце перестановочной матрицы  $P$ .

Комбинация трех последовательностей,  $v^{(0)}$ ,  $v^{(1)}$ , и  $v^{(2)}$  дает полную передаваемую последовательность (кодовое слово)

$$v = v_0 v_1 \dots v_{K-1} \quad (3.8)$$

где  $v_n = v_n^{(0)} v_n^{(1)} v_n^{(2)}$ ,  $n = 0, 1, \dots, K-1$ , такая что общая длина блока кода  $N = 3K$ . Поскольку, как правило  $K \gg m$ , скорость кода  $R \approx 1/3$ .

Как было отмечено выше, второй кодер, вообще говоря, не возвращается в нулевое состояние. Важным исключением является турбо-кодер, у которого оба компонентных кодера – это рекурсивные сверточные кодеры памяти  $m = 1$  с порождающей матрицей.

$$G = \left(1 + \frac{1}{1+D}\right) \quad (3.9)$$

В этом случае хвост состоит из одного символа  $u_{k-1}$ , равному 0, если вес информационной последовательности четный, и равен 1, если вес нечетный. Такая входная последовательность приводит оба компонентных сверточных кодера памяти  $m = 1$  в нулевое состояние. Поэтому решетки компонентных сверточных кодеров начинаются и заканчиваются в нулевом состоянии и мы можем использовать БКДР алгоритм для декодирования обоих составных кодов.

Важной особенностью обычных турбо-кодов является слабый рост минимального расстояния  $d_{\min}$  с ростом длины кода.

### Параллельное каскадирование трех и более сверточных кодов

Обычные турбо-коды имеют относительно плохое минимальное расстояние. При относительно высоких отношениях сигнал-шум это вызывает эффект “зависания” вероятности ошибки декодирования, при котором вероятность ошибки очень медленно убывает с ростом отношения сигнал-шум. Этот участок графика называется “полом” (floor).

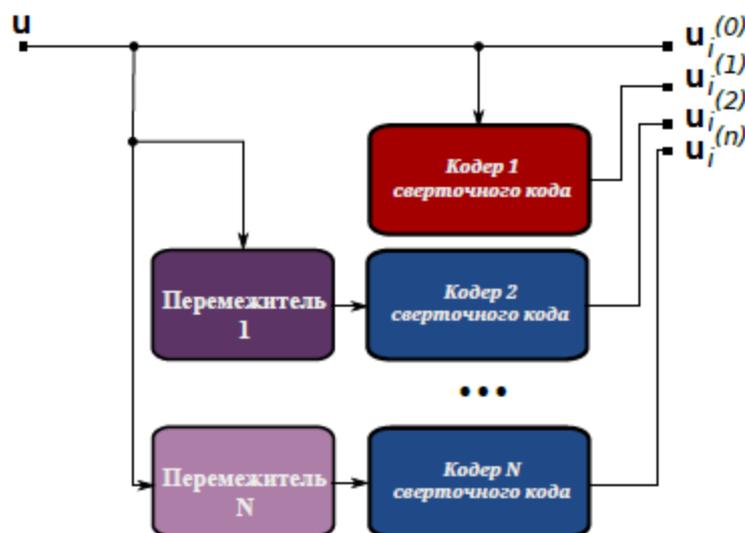


Рис. 3.114. Структура  $J$ -размерного сверточного турбокодера

Хотя минимальное расстояние турбо-кода может быть улучшено выбором хорошего переключателя, оно не может расти быстрее, чем логарифмически с ростом длины блока. Поведение минимального расстояния МТК существенно лучше.

Кодер  $J$ -мерного МТК состоит из  $J$ ,  $J \geq 3$ , параллельных компонентных сверточных кодеров и  $J$  единичных блоковых переключателей  $P(j)$ ,  $j = 1, 2, \dots, J$ , размеров  $K \times K$ , как показано на рис. 7.2. ( Определим формально нулевой переключатель  $P(0)$  как единичную матрицу размеров  $K \times K$ .) Предполагается, что все компонентные кодеры – это

систематические рекурсивные кодеры скорости  $R = 1/2$  и памяти  $m$ . Двоичная последовательность длины  $K$

$$u = u_0 u_1 \dots u_{K-1} \quad (3.10)$$

это входная последовательность МТК. Это может быть информационная последовательность; в этом случае все компонентные кодеры – это коды с нейтрализацией хвоста. Аналогично обычным турбо-кодам следует рассмотреть случай, когда  $u$  состоит из  $K$  информационных символов плюс хвост из  $m$  двоичных символов, так что  $K = K + m$ . В частности, если все компонентные кодеры – это систематические рекурсивные сверточные кодеры памяти 1, хвост состоит из 1 символа.

Последовательность  $u$  поступает на входы  $J$  ПБП, описываемым перестановочными матрицами  $P(j) = (p(j)_{ik})$ , где  $j = 1, 2, \dots, J$ ,  $i, k = 0, 1, \dots, K - 1$ , размеров  $K \times K$ . За перемежителями следуют компонентные кодеры, такие что -ая переставленная версия  $u$ , то есть  $u(j) = uP(j)$ ,  $j = 1, 2, \dots, J$ , поступает на вход  $j$ -ого компонентного кодера.

Выходная последовательность -ого компонентного кодера (проверочная последовательность) – это последовательность:

$$v^{(j)} = v_0^{(j)} v_1^{(j)} \dots v_{K-1}^{(j)}, j = 1, 2, \dots, J. \quad (3.11)$$

Входная последовательность МТК,

$$u = v^{(0)} = v_0^{(0)} v_1^{(0)} \dots v_{K-1}^{(0)},$$

перемешивается с выходными последовательностями  $J$  компонентных кодеров и передается по каналу. Кодовое слово многократного турбо-кода,

$$v = v_0 v_1 \dots v_{K-1} \quad (3.12)$$

где  $v_n = v_n^{(0)} v_n^{(1)} \dots v_n^{(j)}$ ,  $n = 0, 1, \dots, K - 1$ , так что длина блока кода равна  $N = (J+1)K$  и результирующая скорость равна  $R = 1/(J+1)$ . Кодовая скорость может быть увеличена выкалыванием символов.

Комбинация тождественного перемежителя и остальных  $J$  перемежителей называется  $(J + 1)$ -мерным перемежителем. Он может быть представлен в виде  $(J + 1)$ -мерной таблицы, сконструированной так, что проекция ненулевого элемента на плоскость, образованную 0-ой и -ой,  $j = 1, 2, \dots, J$ , осями давало бы матрицу перестановок  $P(j)$ ,  $j = 1, 2, \dots, J$ .

### Итеративное декодирование турбо-кодов

В данном разделе описано декодирование турбо-кодов. Предположим, что кодовая последовательность  $v = v_0 v_1 \dots v_{K-1}$ , сконструированная из трех последовательностей,  $v^{(0)} = u$ ,  $v^{(1)}$ , and  $v^{(2)}$ , передается по дискретному каналу без памяти. Пусть

$$r = r_0 r_1 \dots r_{K-1} \quad (3.13)$$

где  $r_n = r_n^{(0)} r_n^{(1)} r_n^{(2)}$ ,  $n = 0, 1, \dots, K-1$ , означает принятую последовательность и пусть

$$\pi_n^{(0)}(0) = P(u_n = 0) \quad (3.14)$$

означает априорную вероятность, что информационный символ  $u_n = 0$ ,  $n = 0, 1, \dots, K-1$ .

На практике обычно  $P(u_n = 0) = 1/2$ .

Итеративный алгоритм декодирования турбо-кодов похож на итеративный алгоритм распространения доверия кодирования МППЧ кодов. Декодер состоит из двух компонентных апостериорно-вероятностных декодеров. Процесс декодирования состоит из  $I$  итераций; каждая итерация, в свою очередь, состоит из двух фаз. В результате  $e$ -ой фазы,  $e = 1, 2$ ,  $i$ -ой итерации  $i = 1, 2, \dots, I$ , вычисляется последовательность:

$$\pi^{(e)}(i) = \pi_0^{(e)}(i) \pi_1^{(e)}(i) \dots \pi_{K-1}^{(e)}(i) \quad (3.15)$$

где  $\pi^{(e)}(i)$  апостериорная вероятность 2, что информационный символ  $u_n = 0$ ,  $n = 0, 1, \dots, K-1$ . Удобно вместо  $\pi^{(e)}(i)$  оперировать с соответствующими логарифмическими отношениями правдоподобия, то есть,

$$z_n^{(e)} = \log \frac{\pi_n^{(e)}}{1 - \pi_n^{(e)}} \quad (3.16)$$

и вместо  $\pi^{(e)}(i)$  оперировать с последовательностью:

$$z^{(e)}(i) = z_0^{(e)}(i) z_1^{(e)}(i) \dots z_{K-1}^{(e)}(i) \quad (3.17)$$

Можно использовать обозначение:

$$z_n^{(0)}(0) = \log \frac{P(u_n = 0 | r_n^{(0)})}{P(u_n = 1 | r_n^{(0)})} = \log \frac{\pi_n^{(0)}}{1 - \pi_n^{(0)}} + \log \frac{P(u_n = 0 | r_n^{(0)})}{P(u_n = 1 | r_n^{(0)})} \quad (3.18)$$

для логарифмического отношения апостериорных вероятностей значений символа  $u_n$ , при условии, что принят символ  $r_n$ . Оно называется внутренней информацией (intrinsic information) об информационном символе  $u_n = v^{(0)}(n)$ . Предполагается, что оба компонентных декодера хранят в памяти последовательность  $z^{(0)}(0) = z_0^{(0)}(0) z_1^{(0)}(0) \dots z_{K-1}^{(0)}(0)$  и используют ее на каждом шаге итеративного процесса декодирования.

Пусть  $r^{(0)}$ ,  $r^{(1)}$ , and  $r^{(2)}$  означают последовательности принятых символов, соответствующих переданной информационной последовательности  $v^{(0)} = u$ , первой проверочной последовательности  $v^{(1)}$ , и второй проверочной последовательности  $v^{(2)}$ , соответственно. Пусть, далее,  $\hat{r}_n^{(0)}$  означает последовательность принятых символов, соответствующих последовательности  $u$  за исключением принятого символа  $r_n^{(0)}$ , то есть,

$$\hat{r}_n^{(0)} = r_0^{(0)} r_1^{(0)} \dots r_{n-1}^{(0)} r_{n+1}^{(0)} \dots \quad (3.19)$$

Аналогично, пусть  $\pi(e)n(i)$  означает последовательность апостериорных вероятностей для символов последовательности  $u$  за исключением  $\pi(e)n(i)$  после  $i$ -ой фазы  $i$ -ой итерации, то есть

$$\pi_n^{(e)}(i) = \pi_0^{(e)}(i)\pi_1^{(e)}(i) \dots \pi_{n-1}^{(e)}(i)\pi_{n+1}^{(e)}(i) \dots \quad (3.20)$$

На первой фазе первой итерации первый компонентный декодер, используя  $r^{(0)n}$ ,  $r(1)$  и

$$\pi_n^{(0)}(0) = \pi_0^{(0)}(0)\pi_1^{(0)}(0) \dots \pi_{n-1}^{(0)}(0)\pi_{n+1}^{(0)}(0) \dots \quad (3.21)$$

вычисляет логарифмическое отношение правдоподобия

$$y_n^{(1)}(1) = \log \frac{P(r_n^{(0)}, r^{(1)} | r_n^{(0)} u_n = 0)}{P(r_n^{(0)}, r^{(1)} | u_n = 1)} \quad (3.22)$$

для информационных символов  $u_n = v_n^{(0)}$ ,  $n = 0, 1, \dots, K-1$ .

Для вычисления  $y_n^{(1)}(1)$  декодер использует БКВР алгоритм апостериорновероятностного декодирования описанный в параграфе 4.5. Величина  $y_n^{(1)}(1)$  называется внешней информацией (extrinsic information) первого компонентного декодера об информационном символе  $u_n$  при первой итерации. Первый компонентный декодер вычисляет последовательность:

$$z^{(1)}(1) = z_0^{(1)}(1)z_1^{(1)}(1) \dots z_{K-1}^{(1)}(1) \quad (3.23)$$

используя формулу

$$z_n^{(1)}(1) = z_n^{(0)}(0) + y_n^{(1)}(1) \quad (3.24)$$

и посылает ее второму компонентному декодеру.

На второй фазе первой итерации второй компонентный декодер, зная последовательность  $z_n^{(1)}(1)$ , вычисляет апостериорные вероятности  $\pi_n^{(1)}(1)$ ,  $n = 0, 1, \dots, K-1$ , и, следовательно,  $\pi_n^{(1)}(1)$ . Используя  $r_n^{(0)}$ ,  $r^{(2)}$ , и  $\pi_n^{(1)}(1)$ , декодер, вычисляет внешнюю информацию  $y_n^{(2)}(1)$ ,

$$y_n^{(2)}(1) = \log \frac{P(r_n^{(0)}, r^{(2)} | r_n^{(0)} u_n = 0)}{P(r_n^{(0)}, r^{(2)} | u_n = 1)} \quad (3.25)$$

Затем декодер вычисляет новое логарифмическое отношение правдоподобия и посылает:

$$z^{(2)}(1) = z_0^{(2)}(1)z_1^{(2)}(1) \dots z_{K-1}^{(2)}(1) \quad (3.26)$$

первому компонентному декодеру. Это завершает первую итерацию.

Рассмотрим теперь  $i$ -ую итерацию,  $i = 2, 3, \dots, I$ . На первой фазе  $i$ -ой итерации первый компонентный декодер, зная последовательность логарифмических отношений правдоподобия  $z^{(2)}(i-1)$ , вычисляет последовательность апостериорных вероятностей  $\pi^{(2)}(i-1)$  и, следовательно,  $\pi_n^{(2)}(i-1)$ . Используя  $r_n^{(0)}$ ,  $r^{(1)}$ , и  $\pi_n^{(2)}(i-1)$ , первый

компонентный декодер вычисляет  $u_n^{(1)}(i), n = 0, 1, \dots, K-1$ , и  $z_n^{(1)}(i), n = 0, 1, \dots, K-1$ . Затем он посылает  $z^{(1)}(i)$  второму компонентному декодеру и т.д.. Идея состоит в том, что на каждой фазе итеративного процесса декодирования активный компонентный декодер использует (аппроксимированные) апостериорные вероятности, вычисленные другим компонентным декодером на предыдущем шаге декодирования в качестве априорных вероятностей и вычисляет новую аппроксимацию апостериорных вероятностей.

Декодер чередует эти фазы декодирования в течение  $I$  итераций. Решение  $u_n, n = 0, 1, \dots, K-1$ , получается сравнением финального логарифмического отношения  $z_n^{(2)}(I)$  с порогом 0:

$$u_n = \begin{cases} 0, & \text{если } z_n^{(2)}(I) \succ 0; \\ 1, & \text{если } z_n^{(2)}(I) \prec 0. \end{cases} \quad (3.27)$$

(Если  $z_n^{(2)}(I) = 0$ , тогда декодер использует правило бросания монеты и принимает решение  $u_n = 0$  или  $u_n = 1$  с вероятностью  $1/2$ .) Итеративный декодер иллюстрируется на рис. 2.3.1, где  $u = u_0 u_1 \dots u_{K-1}$ . Итеративное декодирование многократных турбо-кодов аналогично итеративному декодированию обычных турбо-кодов. Декодирование многократных турбокодов с  $J$  компонентными кодами состоит из  $J$  итераций, каждая итерация аналогична итерациям декодирования обычных турбо-кодов.

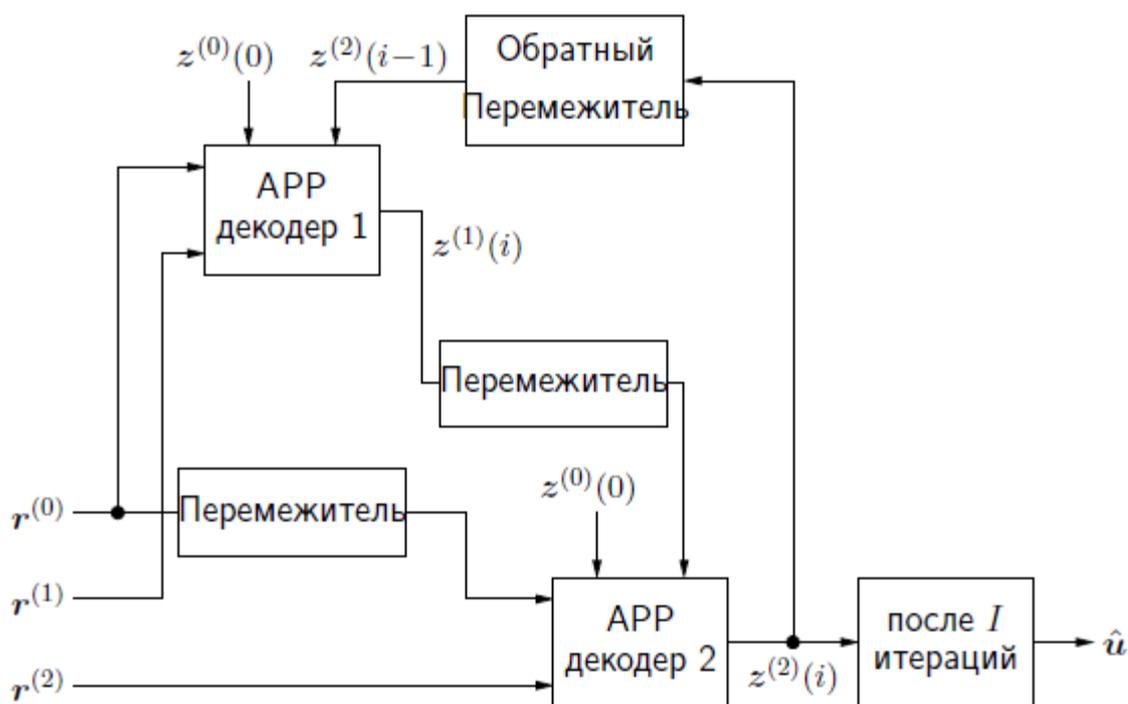


Рис. 3.115. Итеративный декодер для турбо-кодов

## Разработка модели системы связи с использованием турбо-кодов

В диалоговом окне приложения Simulink необходимо собрать схему, представленную на рисунке 3.116.

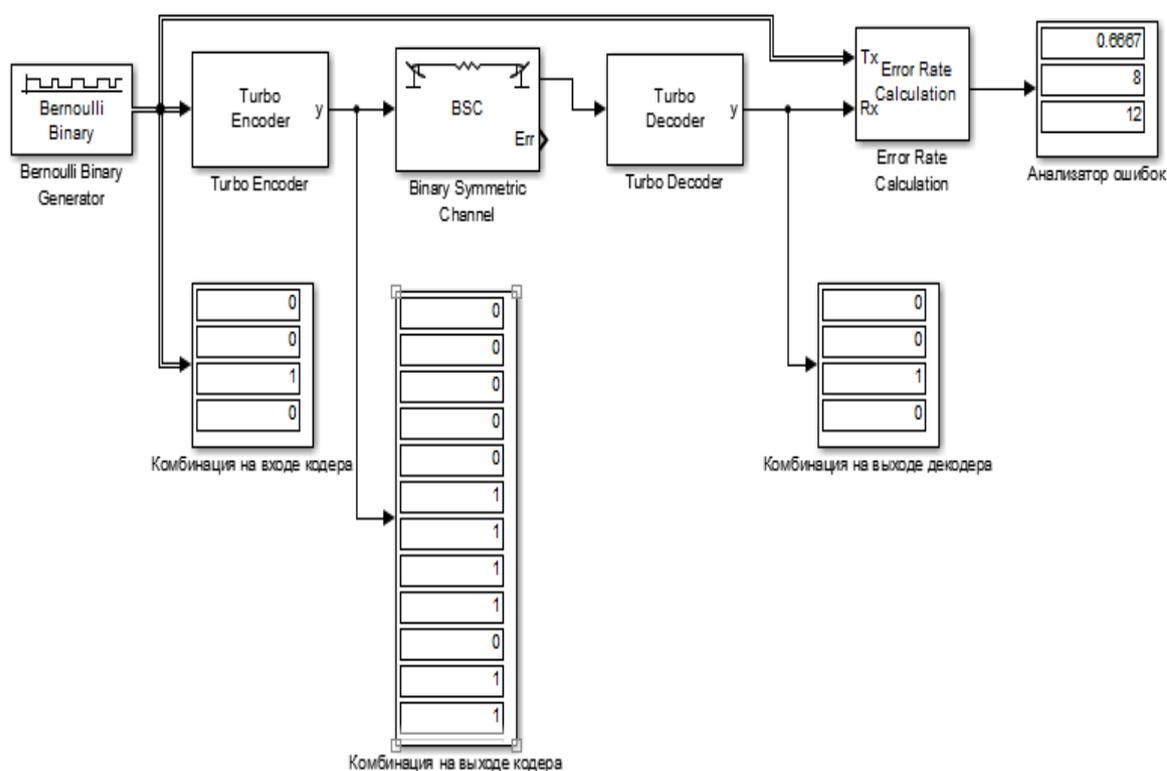


Рис. 3.116. Модель системы связи с использованием турбо-кодов

В состав проектируемой системы связи входят следующие элементы:

1. Bernoulli Binary Generator;
2. Turbo Encoder;
3. Binary Symmetric Channel (канал передачи);
4. Turbo Decoder;
5. Error Rate Calculation (анализатор ошибок);
6. Дисплей, отображающий комбинацию на входе кодера;
7. Дисплей, отображающий комбинацию на выходе кодера;
8. Дисплей, отображающий комбинацию на выходе декодера;
9. Дисплей анализатора ошибок, отображающий частоту ошибок, число обнаруженных ошибок, количество символов по сравнению.

На следующем этапе моделирования необходимо произвести настройку функциональных элементов исследуемой системы связи.

Сначала необходимо выполнить настройку генератора бинарных последовательностей

Бернулли (Bernoulli Binary Generator) (рис. 3.110):

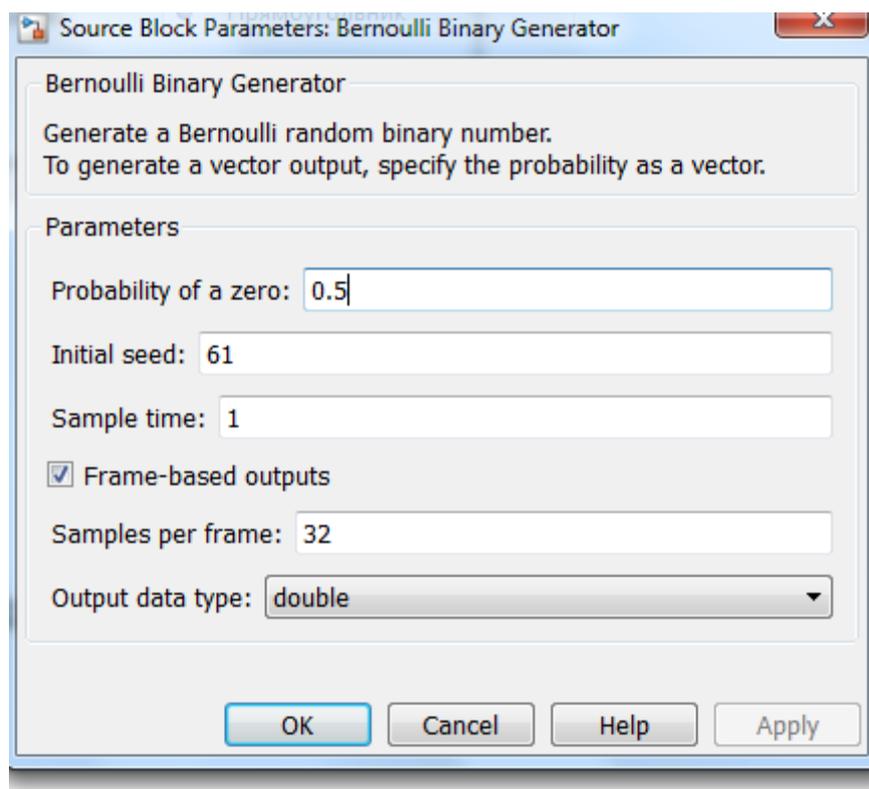


Рисунок 3.110. Настройка Bernoulli Binary Generator для 32 бит

В представленном случае будет использоваться входная последовательность, состоящая из 32 бит

Далее необходимо выполнить настройку турбо-кодера и декодера (рис. 3.111 и рис. 3.112):

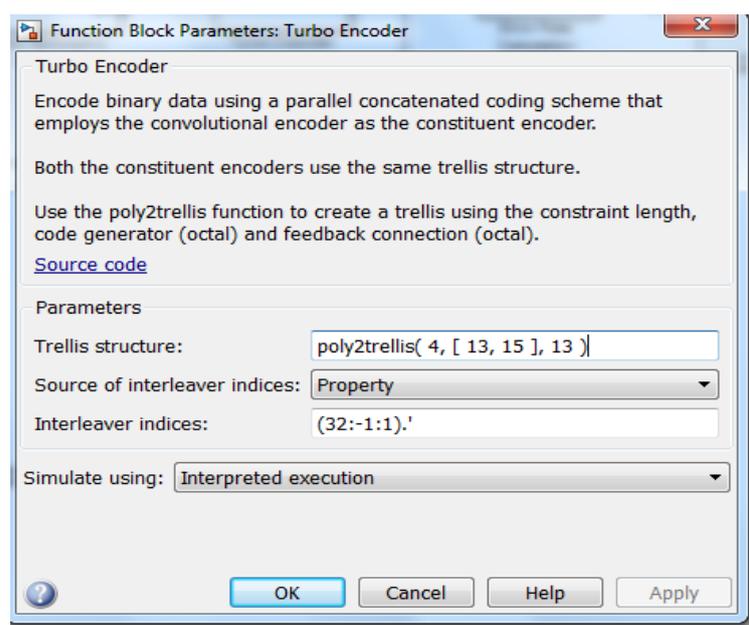


Рис. 3.111. Настройка Turbo Encoder

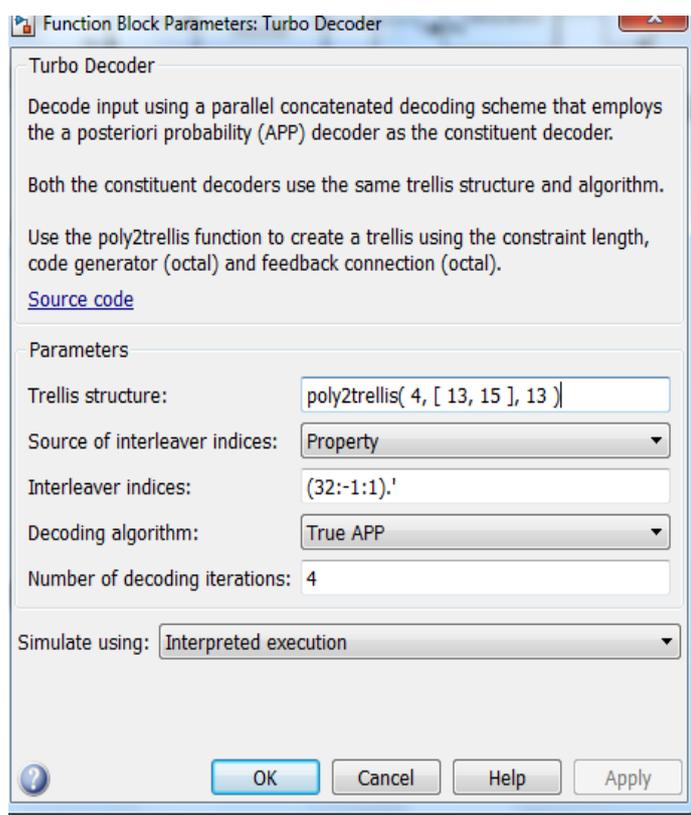


Рис. 3.112. Настройка Turbo Decoder

На следующем шаге необходимо выполнить настройку двоичного канала передачи информации (Binary Symmetric Channel) (рис. 3.113):

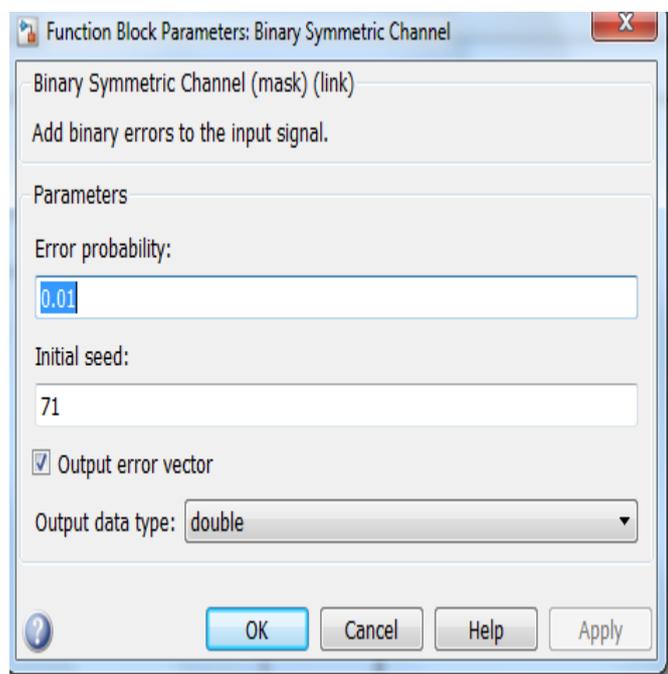


Рис. 3.113. Настройка Binary Symmetric Channel

Стоит отметить, что на данном этапе параметр «вероятность ошибки» необходимо задать малым, для того чтобы проверить работоспособность системы в целом (вероятность ошибки выступает аналогом отношения сигнал/шум в реальном аналоговом канале связи).

Далее настройка анализатора ошибок (Error Rate Calculation) (рис.3.6).

Parameters

Receive delay:  
0

Computation delay:  
0

Computation mode: Entire frame

Output data: Port

Reset port

Stop simulation

Рис. 3.114. Настройка Error Rate Calculation

Заключительный шаг – запуск работы всей системы. После запуска спроектированной системы будут получены следующие результаты:

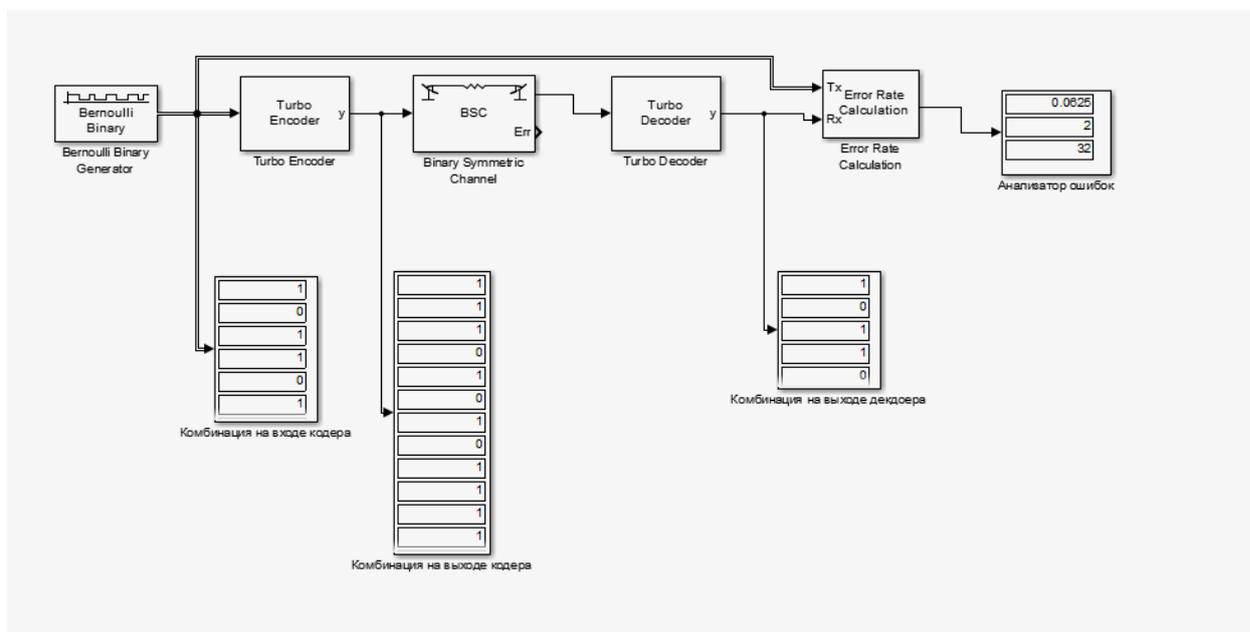


Рис. 3.115. Результаты работы системы связи на основе турбо-кодера

На основе полученных результатов можно сделать вывод, что операции кодирования и декодирования выполнены правильно, следовательно, спроектированная система исправна и работоспособна.

Одной из основных характеристик, характеризующих помехоустойчивость кода, является зависимость количества обнаруженных ошибочных бит от вероятности ошибки. Вероятность ошибки – является аналогом шумов в реальном аналоговом канале передачи. Таким образом для исследования помехоустойчивости турбо-кода представляется целесообразным построить вышеупомянутую зависимость.

Для турбо кода, с входной последовательностью, состоящей из четырех бит и со скоростью кодирования  $R=1/3$  зависимость количества ошибок от вероятности ошибок представлена в таблице 3.7:

Таблица 3.7. Зависимость вероятности ошибок от количества ошибочных бит для турбо кода

P	0	0	0	0	0	0,5	0,7	0,8	0,9	1
ош	,01	,1	,2	,3	,4	6	1	5	2	0,562
N	0	0	0	0	0	0,4	0,53	0,62	0,56	0,562
bits	,062	,281	,343	,375	,437	37	406	1	5	2

Для удобства обработки полученной информации, представляется необходимым и целесообразным представить таблицу 3.7 в виде графика:

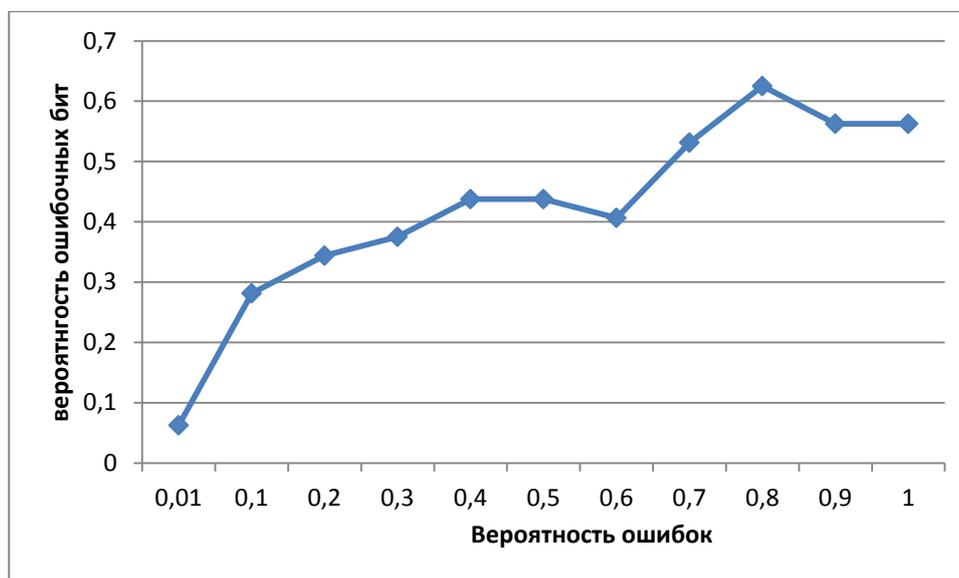


Рис. 3.116. Зависимость вероятности ошибок от количества ошибочных бит

Для турбо кода, с входной последовательностью, состоящей из 64 бит и со скоростью кодирования  $R=1/3$  зависимость количества ошибок от вероятности ошибок представлена в таблице 3.8:

Таблица 3.8. Зависимость вероятности ошибок от количества ошибочных бит для турбо кода

P	0	0	0	0	0	0	0	0	0	0	0	1
ош	,01	,1	,15	,2	,3	,4	,5	,6	,7	,8	,9	
N	0	0	0	0	0	0	0	0	0	0	0	0
bits	,250	,375	,390	,468	,5	,5	,5	,51	,54	,57	,56	,54

Для удобства обработки полученной информации, представляется необходимым и целесообразным представить таблицу 3.8 в виде графика:



Рис. 3.117. Зависимость вероятности ошибок от количества ошибочных бит

Построим зависимости количества ошибочных битов от вероятности возникновения ошибки (рис. 3.118):

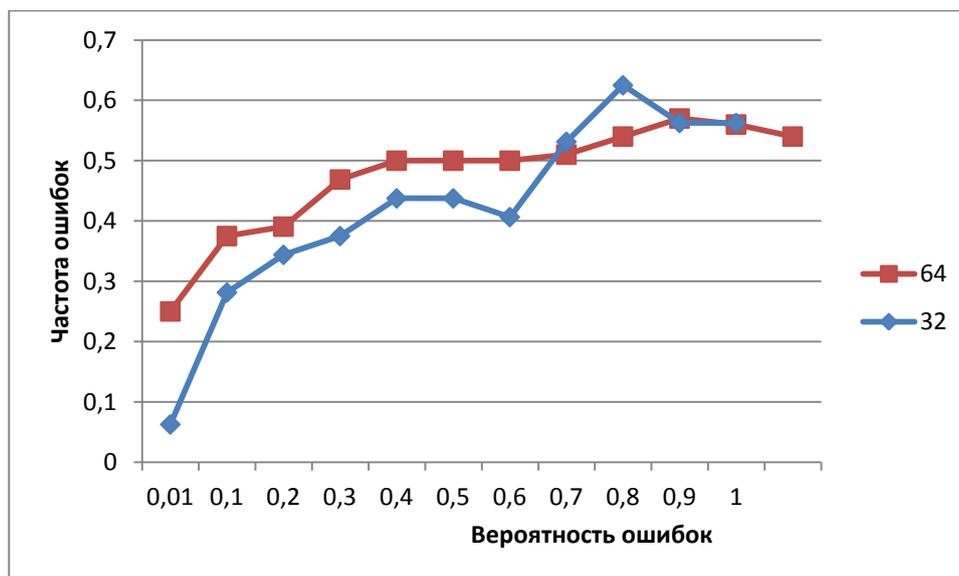


Рис. 3.118. Зависимость количества ошибочных бит

В результате выполнения данной лабораторной работы был проведен теоретический обзор современных алгоритмов использующих для передачи информации по каналам связи турбо-коды.

В ходе работы была разработана модель системы передачи информации с использованием турбо кодов. Кодер представляет собой классический турбо-кодер со скоростью кодирования  $R = 1/3$  реализованный путем параллельного каскадирования двух сверточных кодеров. Декодер представляет собой итеративный декодер турбо-кодов. Основным его достоинством его является то, что на каждой итерации с помощью простых процедур декодирования анализируются данные, принадлежащие простым итерационным кодам.

При помощи разработанной модели было установлено, что турбо-коды с большей длиной кодовых комбинацией обладают лучшей помехоустойчивостью, что соответствует теории.

### 3.6. Низкоплотностные коды. Классификация LDPC-кодов. Методы построения проверочных матриц. Алгоритмы декодирования низкоплотных кодов.

#### Оценка сложности алгоритмов декодирования на базе MATLAB и LabVIEW

#### Низкоплотностные коды. Основные положения [16]

В 1948 году Клод Элвуд Шеннон опубликовал свою работу по теории передачи информации. Одним из ключевых результатов работы считается теорема о передаче информации для канала с шумами, которая говорит о возможности свести вероятность ошибки передачи по каналу к минимуму при выборе достаточно большой длины ключевого слова — единицы информации передаваемой по каналу.



Рис. 3.119. Упрощенная схема передачи информации по каналу связи.

При передаче информации её поток разбивается на блоки определённой (чаще всего) длины, которые преобразуются кодером (кодируются) в блоки, называемыми ключевыми словами. Ключевые слова передаются по каналу, возможно с искажениями. На принимающей стороне декодер преобразует ключевые слова в поток информации, исправляя (по возможности) ошибки передачи.

Теорема Шеннона утверждает, что при определённых условиях вероятность ошибки декодирования (то есть невозможность декодером исправить ошибку передачи) можно уменьшить, выбрав большую длину ключевого слова. Однако, данная теорема (и работа вообще) не показывает, как можно выбрать большую длину, а точнее как эффективно организовать процесс кодирования и декодирования информации с большой длиной ключевых слов.

Если предположить, что в кодере и декодере есть некие таблицы соответствия между входным блоком информации и соответствующим кодовым словом, то такие таблицы будут занимать очень много места. Для двоичного симметричного канала без памяти (если говорить упрощённо, то на вход кодера поступает поток из нулей и единиц) количество различных блоков составляет  $2^n$ , где  $n$  — количество бит (нулей или единиц) которые будут преобразовываться в одно кодовое слово. Для 8 бит это 256 блоков информации, каждый из которых будет содержать в себе соответствующее кодовое слово. Причём кодовое слово обычно большей длины, так как содержит в себе дополнительные биты для защиты от ошибок передачи данных. Поэтому одним из способов кодирования является использование проверочной матрицы, которые позволяют за одно математическое действие (умножение строки на матрицу) выполнить декодирование кодового слова.

Аналогичным образом каждой проверочной матрице соответствует порождающая матрица, аналогичным способом одной операцией умножения строки на матрицу генерирующей кодовой слово.

Таким образом, для сравнительно коротких кодовых слов кодеры и декодеры могут просто содержать в памяти все возможные варианты, или даже реализовывать их в виде полупроводниковой схемы. Для большего размера кодового слова эффективнее хранить порождающую и проверочную матрицу. Однако, при длинах блоков в несколько тысяч бит хранение матриц размером, соответственно, в мегабиты, уже становится неэффективным.

Одним из способов решения данной проблемы становится использования **кодов с малой плотностью проверок на чётность**, когда в проверяющей матрице количество единиц сравнительно мало, что позволяет эффективнее организовать процесс хранения матрицы или же напрямую реализовать процесс декодирования с помощью полупроводниковой схемы.

**Коды с низкой плотностью проверок на чётность (LDPC)** – это класс линейных блоковых кодов, позволяющих получить превосходную эффективность с относительно малыми вычислительными затратами на их декодирование. Эти коды были предложены

Робертом Галлагером еще в 1963-ем году, однако были забыты на сорок лет в связи со сложностью реализации алгоритмов их декодирования.

Развитие цифровой техники позволило преодолеть многие проблемы и в конце 20-го столетия исследования в области LDPC – кодов получили новый импульс.

LDPC-коды становятся востребованными в системах передачи информации, требующих максимальной скорости передачи при ограниченной полосе частот. Основным конкурентом LDPC-кодов на данный момент являются турбо-коды, которые нашли свое применение в системах спутниковой связи, ряде стандартов цифрового телевидения и мобильных системах связи третьего поколения. Однако LDPC-коды по сравнению с турбо-кодами имеют ряд преимуществ.

Во-первых, LDPC-коды обгоняют турбо-коды по скорости декодирования.

Во-вторых, LDPC-коды более предпочтительны в каналах с меньшими вероятностями ошибок. С развитием методов передачи информации каналы передачи улучшаются, что дает хорошую перспективу для развития LDPC-кодов.

Применение методов итеративного декодирования к данным кодам позволяет практически вплотную приблизиться к пропускной способности канала при относительно небольшой сложности реализации. В связи с этим во многих новых стандартах передачи различного рода данных (DVB-S2, 802.11n, 802.16e) именно LDPC- коды рекомендованы для исправления ошибок. LDPC - коды представляют собой линейные блочные коды, задаваемые с помощью проверочной матрицы  $H$ , характеризуемой относительно малым ( $<10$ ) числом единиц в строках и столбцах.

В 1981-ом году Р.М. Таннером было предложено использовать двудольные неориентированные графы, впоследствии названные графами Таннера, для описания структуры итеративно декодируемых кодов. В принципе любой блочный код размерности  $(M,N)$ , где  $N$  – число битов, а  $M$  – число проверок в кодовом слове, можно представить в виде двудольного графа Таннера. Например, на рисунке 3.131 изображен такой граф для кода Хэмминга  $(7,4)$ , проверочная матрица которого имеет вид:

$$H = \begin{pmatrix} 1 & 1 & 1 & 0 & 1 & 0 & 0 \\ 0 & 1 & 1 & 1 & 0 & 1 & 0 \\ 1 & 1 & 0 & 1 & 0 & 0 & 1 \end{pmatrix}$$

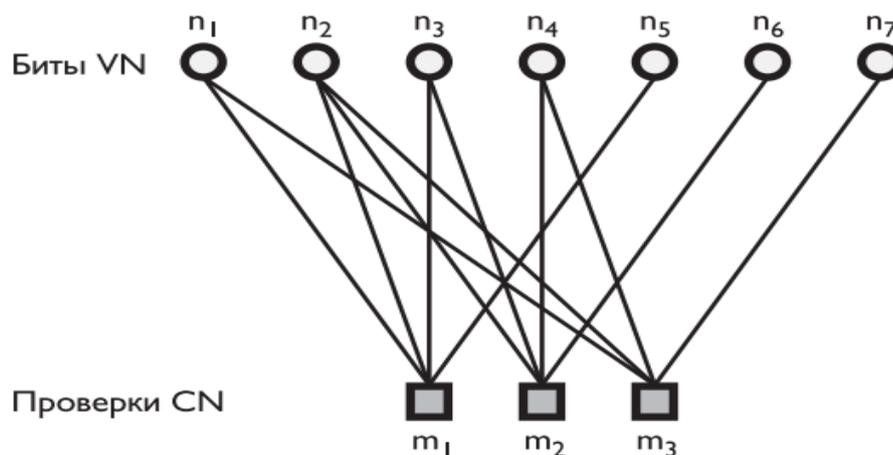


Рис. 3.120. Граф Таннера для кода Хэмминга (7,4)

Вершины графа называются проверочными (check nodes - CN) и битовыми узлами (variable nodes - VN), они обозначены на рисунке  $m$  и  $n$  соответственно.

При помощи графа Таннера большинство алгоритмов декодирования LDPC-кодов можно представить в виде процессов последовательного обмена сообщениями между соединенными ребрами вершинами. Для проверочных и битовых узлов графа вводится понятие степени – величины, показывающей число рёбер, входящих в рассматриваемый узел. Степени битовых и проверочных узлов обозначаются  $d_c$  и  $d_r$  соответственно. Если  $d_c$  и  $d_r$  фиксированы для всех узлов, такой код называют регулярным, а если хотя бы один из этих параметров изменяется от узла к узлу – нерегулярным. Для описания нерегулярных кодов вводится ряд распределения степеней, показывающий долю узлов, имеющих конкретную степень.

В 1996-ом году вышла в свет первая после Р.Галлагера работа, посвященная использованию LDPC – кодов в качестве кодов, способных вплотную приблизиться к границе Шеннона при достаточно большой длине кодового слова. Появление этой статьи породило целую волну исследований, посвященных поиску новых, более эффективных структур LDPC – кодов, а также альтернативных алгоритмов их декодирования с различными соотношениями эффективность/производительность.

В последнее время LDPC – коды получили широкое распространение благодаря превосходной эффективности. Использование LDPC – кодов предусматривает большинство современных стандартов передачи данных (например, стандарты IEEE 802.11, IEEE 802.16), стандартов цифрового вещания (например, стандарты DVB-S2, DVB-T2, DVB-C2).

### Классификация LDPC - кодов.

По определению, данному Р. Галлагером, низкоплотный код — это линейный код, проверочная матрица  $H$  которого размерности  $(M \times N)$  содержит  $d_c \ll M$  единиц в каждом столбце и  $d_r \ll N$  единиц в каждой строке. Причем распределение единиц по столбцам и строкам в общем случае случайно.

На практике случайное распределение единиц крайне неудобно — для кодирования и декодирования приходится хранить проверочные и генераторные матрицы, что достаточно накладно, особенно при больших длинах кодов.

Очевидным средством борьбы с этой проблемой является переход к низкоплотным кодам, проверочная матрица которых обладает какой-то структурой. Простейший вариант структуризации проверочной матрицы — использование циклических кодов.

Формально проверочная матрица такого кода представляет собой циркулянтную матрицу размерности  $N \times N$ , в которой каждая строка получается циклическим сдвигом вправо предыдущей строки. Значение влияния цикличности проверочной матрицы на сложность декодера LDPC-кода сложно переоценить, поскольку каждая из строк матрицы однозначно определяется предыдущей строкой, в связи с чем реализация декодера может быть существенно упрощена по сравнению со случайной структурой проверочной матрицы. Как известно, кодер циклического кода достаточно просто реализовать с использованием сдвигового регистра и набора сумматоров.

К недостаткам циклических кодов можно отнести фиксированный, для всех скоростей кодирования, размер проверочной матрицы  $N \times N$ , что подразумевает более сложный декодер, а также высокий Хэммингов вес строк, что усложняет структуру декодера. Дополнительно стоит заметить, что циклический код — всегда регулярный.

К достоинствам помимо упрощения кодирования/декодирования следует отнести большое минимальное расстояние и очень низкий порог при итеративном декодировании.

Желание преодолеть недостатки циклических LDPC-кодов привело к появлению квазициклических LDPC-кодов. Квазициклические коды также имеют хорошую структуру, позволяющую упростить кодер и декодер. В дополнение к этому они позволяют более гибко подойти к разработке кода, в частности позволяют синтезировать нерегулярные коды. Проверочная матрица такого кода представляет собой не что иное, как набор циркулянтных подматриц:

$$H = \begin{bmatrix} A_{11} & \cdots & A_{1N} \\ \vdots & & \vdots \\ A_{M1} & \cdots & A_{MN} \end{bmatrix}$$

Очевидно, что для получения низкоплотного кода, циркулянтные матрицы должны быть разреженными, что на практике означает использование в качестве циркулянтов единичных матриц. Для того чтобы получить нерегулярный код, какие-то подматрицы просто объявляются нулевыми.

### Методы построения проверочных матриц

Методы построения LDPC-кодов также можно разбить на классы. К первому классу относятся все алгоритмические способы и способы, использующие вычислительную технику. А ко второму — способы, основанные на теории графов, математике конечных полей, алгебре и комбинаторике.

При этом стоит заметить, что первый класс методов позволяет получать как случайные, так и структурированные LDPC-коды, в то время как второй нацелен на получение только структурированных LDPC-кодов, хотя бывают и исключения.

В отличие от других линейных блочных кодов, таких как БЧХ или кодов Рида–Соломона, имеющих строгий алгоритм синтеза кодов с заданными параметрами, для LDPC-кодов существует множество способов построения кодов.

Существуют способы построения LDPC-кодов, предложенные Галлагером и МакКеем, о которых ниже и пойдет речь.

Для начала будет рассмотрен метод, предложенный Р. Галлагером.

Р. Галлагер предложил следующий алгоритм построения низкоплотных кодов. Пусть проверочная матрица кода имеет вид:

$$H = \begin{bmatrix} H_1 \\ H_2 \\ \vdots \\ H_{d_c} \end{bmatrix}$$

В ней подматрицы  $H_a$ ,  $a = 1, 2, \dots, d_c$  имеют структуру, которая может быть описана следующим образом.

Для любых двух целых  $\mu$  и  $d_r$ , больших 1, каждая подматрица  $H_a$  имеет размерность  $(\mu \times \mu \cdot d_r)$ , при этом веса строк этой подматрицы -  $d_r$ , а столбцов - 1. Подматрица  $H_1$  имеет специфическую форму: для  $i=0,1,\dots, \mu-1$   $i$ -ая строка содержит все  $d_r$  единиц на позициях с  $i$  от  $d_r$  до  $(i+r) \cdot r - 1$ . Очевидно, что результирующая матрица  $H$  - регулярная матрица размерности  $\mu \cdot d_c \times \mu \cdot d_r$  с весами строк и столбцов  $d_r$  и  $d_c$  соответственно.

Важной характеристикой матрицы LDPC-кода является отсутствие циклов определенного размера (кратности). Под циклом кратности 4 понимается наличие в двух разных столбцах проверочной матрицы ненулевых элементов на совпадающих позициях. Отсутствие цикла кратности 4 определяется вычислением скалярного произведения столбцов матрицы: если всевозможные скалярные произведения всех столбцов матрицы не превосходят 1, то это означает отсутствие в матрице циклов кратности 4. Цикл кратности 4 является минимально возможным и встречается существенно чаще циклов большей длины (6, 8, 10 и т. д.). Присутствие в матрице LDPC-кода циклов любой кратности свидетельствует о заложенной в структуру матрицы избыточности, не приводящей к улучшению помехоустойчивых свойств кода. Пример циклов кратности 4 приведен на рисунке 3.121

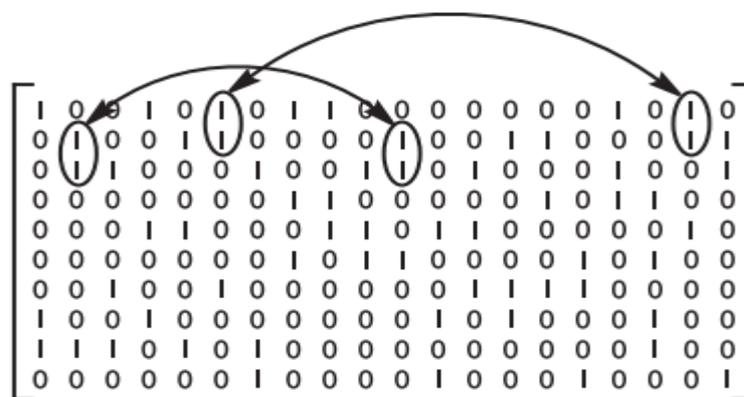


Рис. 3.121. Циклы кратности 4 в матрице LDPC - кода

Рассмотренный выше алгоритм не гарантирует отсутствие циклов кратности 4, однако они могут быть удалены впоследствии. Галлагер показал, что ансамбль таких кодов обладает прекрасными свойствами. Также была показана возможность реализации достаточно простых кодеров, поскольку проверочные биты такого кода могут быть найдены по проверочной матрице кода как функция информационных узлов.

По прошествии тридцати пяти лет МакКей, будучи незнакомым с работой Галлагера, повторно открыл преимущества кодов с разреженными матрицами, и был первым, кто при помощи компьютерного моделирования показал возможность этих кодов вплотную приблизиться к границе Шеннона как для двоичного симметричного канала, так и для канала с аддитивным белым гауссовским шумом. МакКей предложил несколько компьютерных алгоритмов построения проверочных матриц низкоплотностных кодов. Приведем некоторые из них в порядке увеличения сложности реализации.

Проверочная матрица  $H$  синтезируется путём случайного генерирования столбцов веса  $d_c$  и настолько это возможно, равномерным распределением весов строк  $d_r$ .

Проверочная матрица  $H$  синтезируется путём случайного генерирования столбцов веса  $d_c$  и строк веса  $d_r$ , с дополнительной проверкой на отсутствие циклов кратности 4.

Проверочная матрица  $H$  синтезируется по алгоритму 2, с дополнительным удалением циклов кратности 4.

Проверочная матрица  $H$  синтезируется по алгоритму 3, с дополнительным условием, что проверочная матрица имеет вид  $H = [H_1 H_2]$ , где  $H_2$  – обратимая матрица.

Недостатком алгоритмов МакКея является отсутствие какой-либо структуры в проверочных матрицах, что усложняет процесс кодирования.

Кодирование осуществляется приведением матрицы  $H$  к виду  $H = [P^T I]$ , из которого можно получить генераторную матрицу в систематической форме  $G = [PI]$ . Проблема при кодировании по матрице  $G$  заключается в том, что подматрица  $P$ , в общем случае, не является разреженной. То есть для кодов, представляющих интерес, сложность кодирования оказывается достаточно высокой.

### **Декодирование LDPC – кодов.**

Наибольший интерес для исследователей представляет процедура декодирования, ввиду того что она является более время затратной и ресурсоемкой.

Декодирование – это процедура поиска и исправления ошибки, наложенной каналом на кодовое слово, по принятому из канала вектору или собственно поиск кодового слова по вектору, принятому из канала.

Декодирование по максимуму правдоподобия кода  $C$  обозначает нахождение по заданному принятому вектору  $y$  такого кодового слова  $c$  из  $C$  (множества всех кодовых слов), которое максимизирует вероятность того, что передавалось слово  $c$  при условии принятия вектора  $y$ . Задача декодирования по максимуму правдоподобия является NP-полной.

Для оценки качества работы различных декодеров используется оценка вероятности ошибки декодирования (BER) на информационный бит, вычисляемая как отношение количества ошибочных информационных бит после декодирования к общему количеству переданных информационных бит. итеративные схемы декодирования кодов с низкой плотностью проверок на четность не являются декодерами по максимуму правдоподобия, но

позволяют получить разумный баланс по сложности и вероятности ошибки декодирования по сравнению с декодированием по максимуму правдоподобия. итеративное декодирование подразумевает, что нахождение кодового слова будет производиться не за один проход, а за несколько, с последовательным уточнением результата на каждом шаге. Применяются следующие основные схемы декодирования: «жёсткое» декодирование, быстрое декодирование, многопороговое декодирование.

«Жесткое» декодирование – это схема декодирования для двоичного симметричного канала при небольшом количестве ошибок в канале. «Жесткое» декодирование инвертированием битов – самая простая схема декодирования кодов с низкой плотностью проверок на четность.

Под проверкой понимается любая строка  $h = \{h_0, h_1, \dots, h_{N-1}\}$  из проверочной матрицы кода с низкой плотностью проверок на чётность. Будем говорить, что проверка для некоторого вектора  $y = \{y_0, y_1, \dots, y_{N-1}\}$  выполняется тогда, когда скалярное произведение вектора  $y$  на проверку даёт нуль. Будем говорить, что элемент  $y_i$  принятого вектора  $y$  участвует в проверке  $h = \{h_0, h_1, \dots, h_{N-1}\}$  тогда, когда соответствующий элемент проверки  $h_i$  не равен нулю.

Одна итерация «жёсткого» декодирования инвертированием битов производится следующим образом:

1. Для принятого вектора вычисляются все проверки.
2. Если некоторый бит принятого вектора участвовал более чем в половине не выполнившихся проверок, бит инвертируется.
3. После такого анализа всех символов принятого вектора вектор проверяется на принадлежность коду. Если вектор является кодовым словом, декодирование заканчивается, в противном случае выполняется следующая итерация алгоритма.

Такая процедура декодирования применима для кодов с низкой плотностью проверок на четность потому, что большинство проверок в таком случае будут содержать одну ошибку или не будут содержать ошибок вообще и тогда невыполнение большого количества проверок для символа принятого слова будет обозначать наличие в нем ошибки.

Сложность одной итерации «жесткого» декодирования инвертированием бит является линейной, количество итераций декодирования обычно выбирается около  $\log_2(N)$ , где  $N$  – длина кодового слова.

Декодирование по вероятностям является «мягким» декодированием, т.е. декодированием на основе вектора, состоящего не из дискретных значений (0 и 1), а из вещественных величин, полученных на выходе канала путем пересчета вероятностей (англ. belief propagation decoding).

На основе принятого из канала вектора формируются два (для двоичного случая) вектора вероятностей того, что в принятом векторе на данной позиции находился заданный символ.

Каждому ненулевому элементу проверочной матрицы кода с низкой плотностью проверок на чётность приписываются две величины:  $q_{i,j}^x$  и  $r_{i,j}^x$ . Величина  $q_{i,j}^x$  является вероятностью того, что  $j$ -ый символ принятого вектора имеет значение  $x$  по информации, полученной из всех проверок, кроме  $i$ -й. Величина  $r_{i,j}^x$  является вероятностью того, что проверка  $i$  выполняется, если  $j$ -ый символ принятого вектора равен  $x$ , а все остальные символы проверок имеют распределение вероятностей, заданное величинами  $\{q_{i,j}^x: j \text{ из } N(i)/j\}$ , где  $N(i)$  – множество символов, входящих в  $i$ -ую проверку.

Перед началом работы алгоритму требуется инициализация, далее алгоритм работает по принципу пересчета вероятностей символов принятого вектора (belief propagation), используя для пересчета вероятностей правило Байеса для апостериорной вероятности события. Одна итерация алгоритма представляет собой следующую последовательность действий:

1. Для всех проверок вычисляются величины  $\Delta r_{i,j}^x$  и пересчитываются вероятности  $r_{i,j}^x$  для  $x = \{0,1\}$ .
2. Для всех символов принятого вектора пересчитываются вероятности  $q_{i,j}^x$ .
3. Формируются векторы псевдоапостериорной вероятности  $q_j^0$  и  $q_j^1$ .
4. Формируется вектор решения  $c'$  по следующему правилу:  $c'_j=1$ , если  $q_j^1 > S$ , иначе 0.

Если вектор  $c'$  является кодовым словом, декодирование заканчивается, в противном случае выполняется следующая итерация алгоритма.

Сложность данного алгоритма выше, чем сложность «жесткого» декодирования инвертированием битов, но качество декодирования повышается за счет использования дополнительной информации на выходе канала. Однако точность работы такого алгоритма зависит от инициализации: чем точнее она произведена, тем точнее будет конечный результат. Для канала с гауссовским шумом инициализация может быть произведена при

помощи информации о дисперсии шума в канале. Для других распределений шума в канале или при неизвестных характеристиках шума точная инициализация алгоритма может оказаться сложной задачей.

Несмотря на то что декодирование пересчетом вероятностей является эффективным методом для каналов с непрерывным выходом, тот факт, что сложность его значительно выше, чем сложность «жесткого» декодирования, создает предпосылки для поиска более быстрых алгоритмов декодирования, обладающих приемлемым качеством.

Среди известных алгоритмов быстрого декодирования кодов с низкой плотностью проверок на четность для каналов с непрерывным выходом наиболее известен алгоритм «min-sum», являющийся упрощением декодера «belief propagation», а также алгоритм UMP (Uniformly Most Powerful).

Сложность декодера UMP (быстрого декодирования по надежностям) значительно ниже, чем сложность декодера, пересчитывающего вероятности, за счет того, что пересчет надежностей выполняется по упрощенной схеме (схеме «взвешенного» мажоритарного голосования, в качестве «весов» используется надежность проверок), а также за счет возможности использования исключительно целочисленных операций сложения и сложения по модулю два. также к достоинствам быстрого декодера по надежностям можно отнести то, что декодеру не требуется знать характеристики шума в канале (дисперсию и т. д.), следовательно, такой декодер может работать в любом симметричном канале с двоичным входом.

Недостатком быстрого декодера по надежностям является оценка вероятности ошибки декодирования, которая для канала с аддитивным гауссовским шумом оказывается на 0,5 дБ хуже, чем вероятность ошибки декодирования вероятностного декодера.

Далее будет рассмотрено многопороговое декодирование.

Основная идея многопорогового декодирования по надежностям состоит в том, чтобы изменять значения порогов инвертирования символов от одной итерации к другой следующим образом: на первых итерациях порог инвертирования символов выбирается так, чтобы количество инвертированных символов было минимальным (вплоть до инвертирования только одного символа на первой итерации); на последующих итерациях пороги инвертирования постепенно повышаются.

При многопороговом декодировании, если на первой итерации была исправлена хотя бы одна ошибка, декодирование на последующих итерациях становится значительно проще и общее качество декодирования улучшается. По-прежнему для работы декодеру не требуется информация о шуме в канале, достаточно лишь задать надежности.

Декодер, работающий по многопороговой схеме, позволяет получить вероятность ошибки декодирования на 0,1–0,4 дБ лучше, чем обеспечивает быстрый декодер по надежности UMP, практически приближаясь к вероятности ошибки, получаемой при вероятностном декодировании кодов с низкой плотностью проверок на четность. Помимо независимости от характеристик канала многопороговый декодер обладает свойством декодеров кодов с низкой плотностью проверок на четность, а именно универсальностью и применимостью для любой конструкции таких кодов.

Следует отметить, что эффективность нерегулярных LDPC-кодов оказывается выше эффективности регулярных кодов. Это объясняется тем, что в нерегулярных кодах из-за различного числа единиц в строках и столбцах информационные символы защищены по-разному. В результате при декодировании проявляется так называемый эффект волны, когда более защищенные биты декодируются быстрее и затем как бы помогают при декодировании менее защищенных бит.

### Экспериментальная часть

Задание на лабораторную работу:

1. Изучить работу приложения «LDPC Кодер-Декодер»;
2. Установить параметры матрицы, выставив значения  $n=100$ ,  $m=50$ ,  $j=3$ , Message Bits = 1000;
3. Меняя значение параметра Noise выписать показания контроллеров BER с использованием кодирования и без него
4. По полученным данным построить соответствующие графики.
5. Привести рисунки, демонстрирующие исходное сообщение, закодированное сообщение, декодированное сообщение и разницу между исходным и декодированным сообщением.

В таблице 3.7 приведены данные, которые были получены в результате эксперимента.

Таблица 3.9. Данные эксперимента

		0,								
Noise		02	0,04	0,06	0,08	0,1	0,12	0,14	,16	,18

BER(coding)	0,001	0,007064	0,018164	0,030273	0,076832	0,087948	0,125126		
BER(NO_coding)	0,0242	0,0565	0,0777	0,0843	0,1129		1	1	

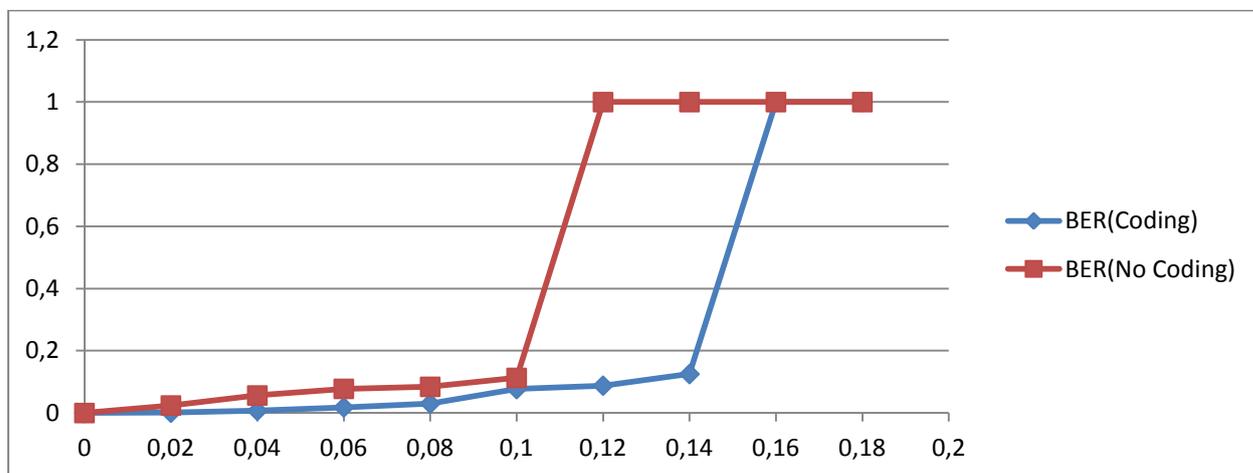


Рис. 3.122. График зависимости BER от Noise

На данном графике видно, что быстрее единицы достиг счётчик, который считал BER в канале без кодирования LDPC.

На рисунках 3.123, 3.124, 3.125 и 3.126 представлены примеры работы LDPC кодера-декодера [17].

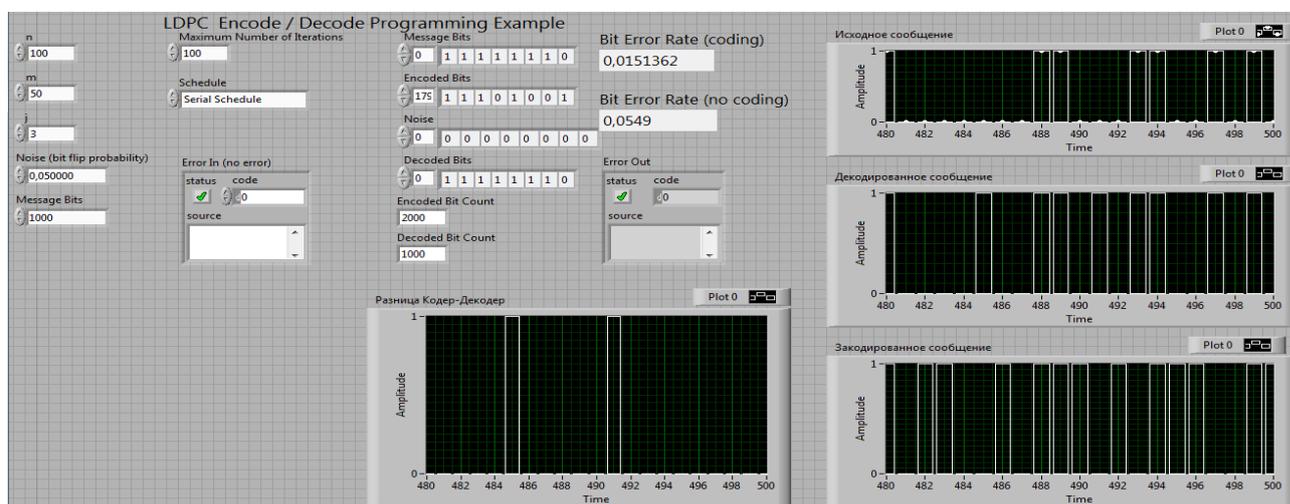


Рис.3.123. Панель управления аппаратно-программного комплекса LDPC кодера-декодера

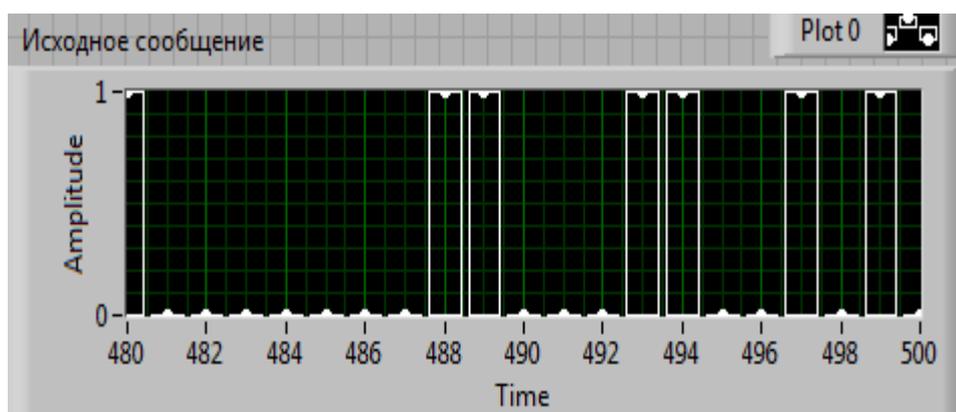


Рис. 3.124. Исходное сообщение

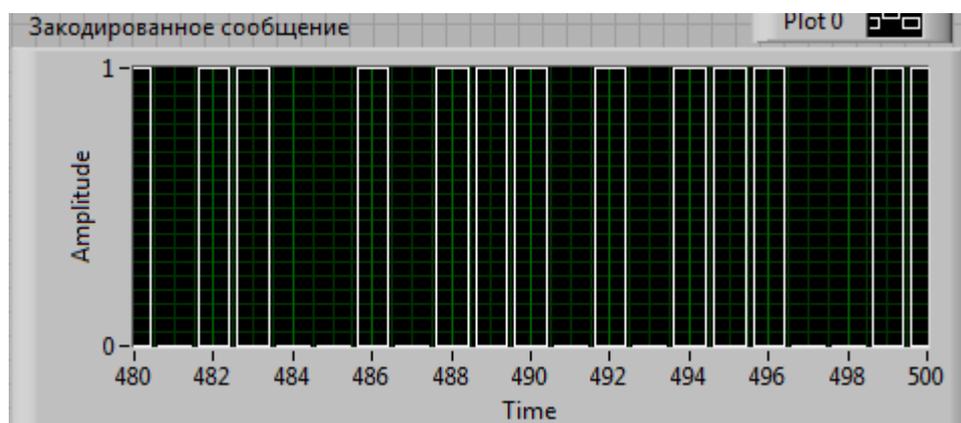


Рис. 3.125. Закодированное сообщение

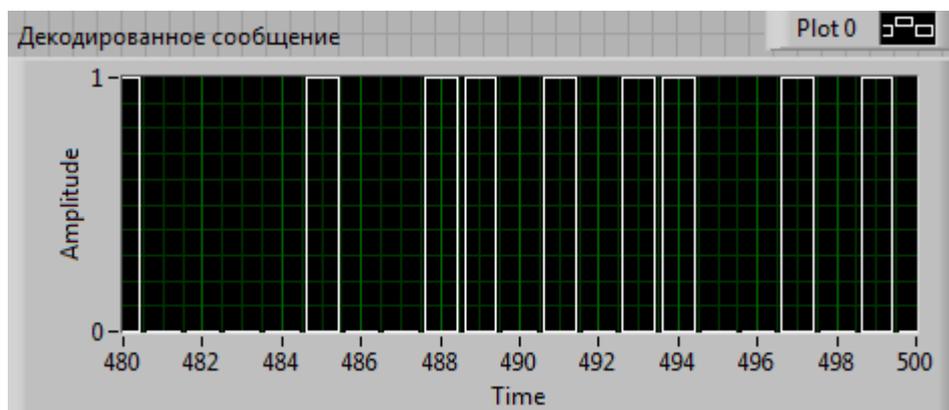


Рис. 3.126. Декодированное сообщение

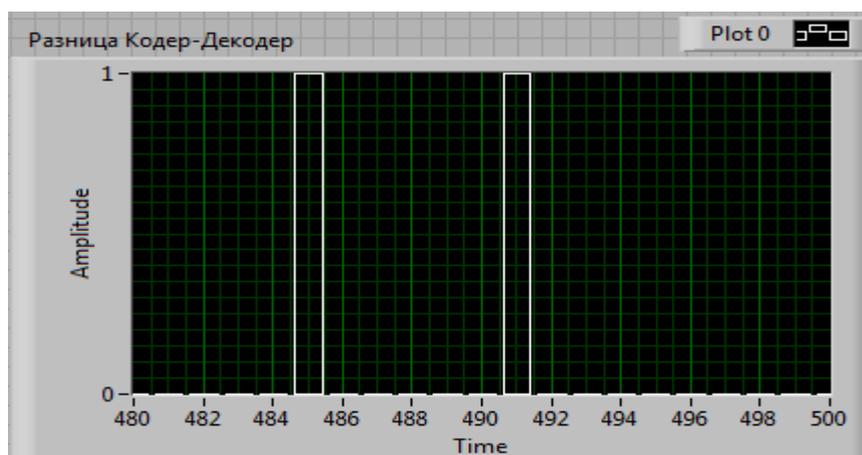


Рис. 3.127. Разница между исходным и декодированным сообщением

В ходе данной лабораторной работы были закреплены навыки работы с ПО LabVIEW, изучен учебный аппаратно-программный комплекс для визуализации и исследования методов канального кодирования/декодирования в беспроводных системах цифрового вещания и связи. Данный комплекс позволяет проводить исследование кодов LDPC. Были сделаны выводы о том, что в канале с кодированием LDPC стопроцентная ошибка возникнет лишь при значении шума, равном 0,16. Тогда как в канале без кодирования стопроцентная ошибка достигается уже при 0,1.

### Моделирование LDPC кодов в MATLAB Simulink

В рабочем поле необходимо собрать схему для работы кода LDPC. Схема представлена на рисунке 3.128 [21]

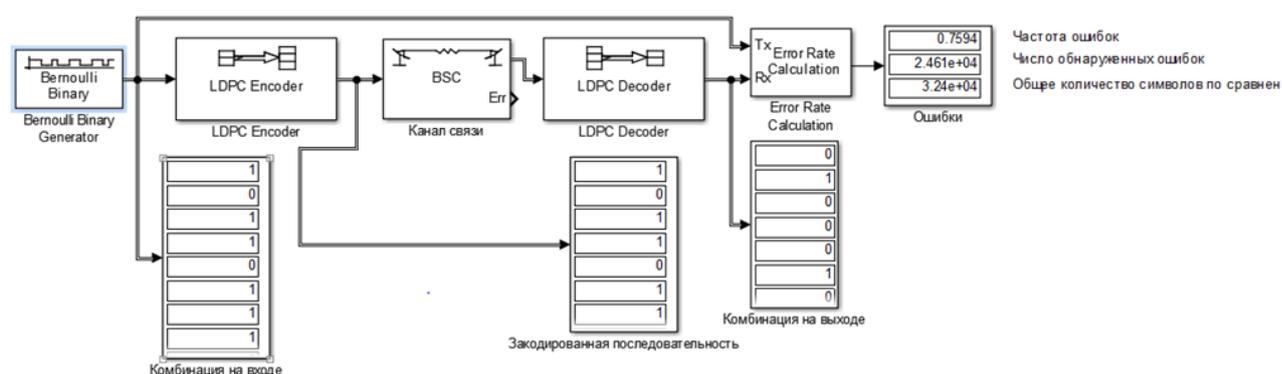


Рис.3.128 Линия передачи с применением код LDPC

В состав линии с кодированием входят:

7. Bernoulli Binary Generator
8. LDPC Encoder
9. Binary Symmetric Channel (канал передачи)

10.LDPC Decoder

11.Error Rate Calculation (анализатор ошибок)

12.Display

Данные блоки необходимо найти в окне «Simulink Library», представленной на рисунке

11. Для упрощения поиска, можно воспользоваться окошком поиска элементов.

Характеристики блоков выставить следующие:

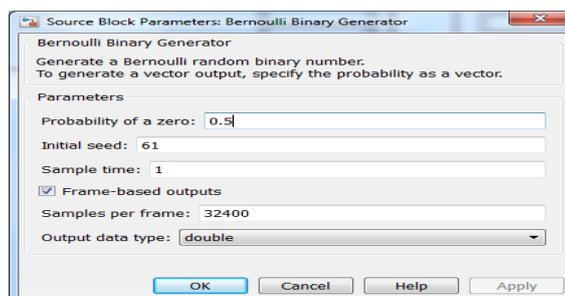


Рис.3.129 Bernoulli Binary Generator

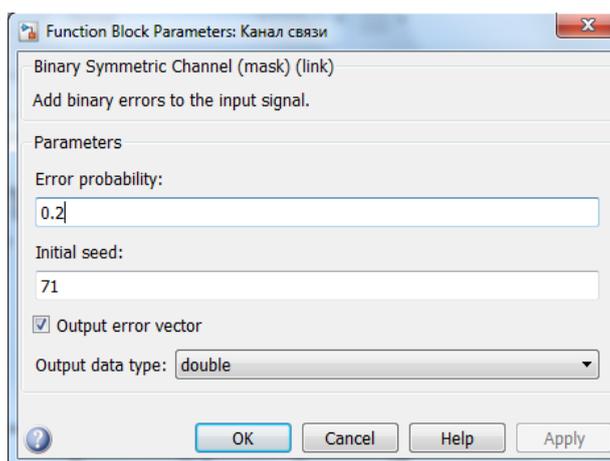


Рис.3.130. LDPC Encoder

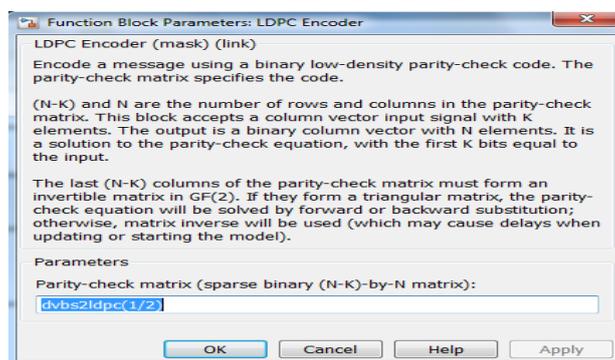


Рис.3.131. Binary Symmetric Channel

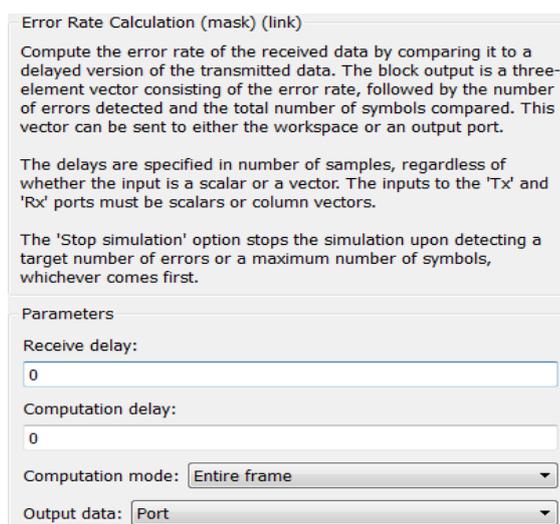


Рис.3.132. BCH Decoder

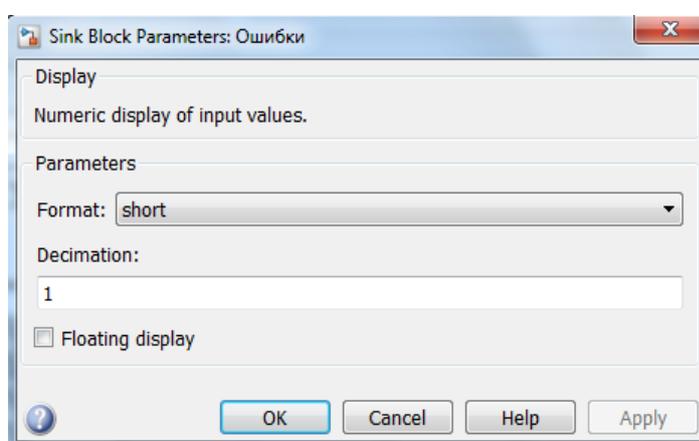


Рис.3.133. Error Rate Calculation (анализатор ошибок)

Представим полученные результаты:

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

Рис. 3.134. Комбинация на входе



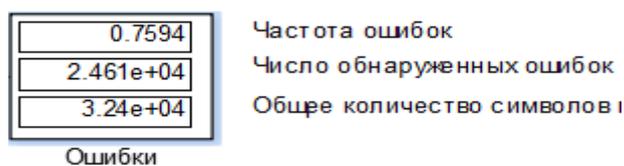


Рис. 3.137. Работа блока оценки ошибки в канале

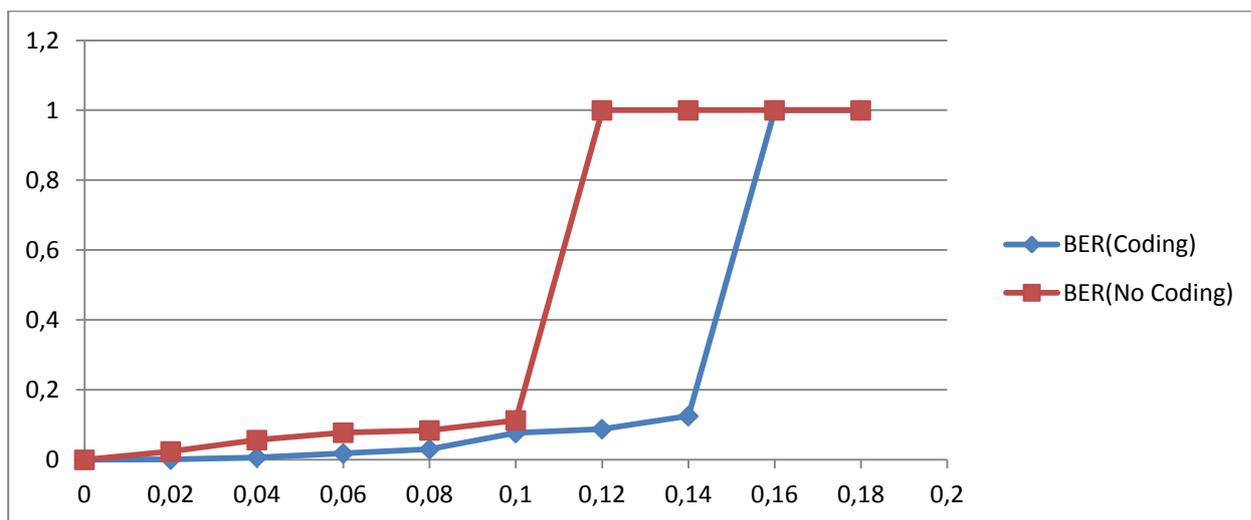


Рис. 3.138 График зависимости BER от Noise

### 3.7. Исследование каскадных кодов

Каскадными называют коды, в которых кодирование осуществляется в два уровня; имеется внутренний и внешний коды, с помощью которых и достигается желаемая надёжность передачи сообщений. Внутренний код связан с модулятором. Демодулятор, как правило, настраивается для исправления большинства канальных ошибок. Внешний код, чаще всего высокоскоростной (с низкой избыточностью), снижает вероятность появления ошибок до заданного значения. Основной причиной использования каскадного кода является низкая степень кодирования и общая сложность реализации, меньшая той, которая потребовалась бы для осуществления отдельной процедуры кодирования.

В одной из наиболее популярных систем каскадного кодирования для внутреннего кода применяется сверточное кодирование по алгоритму Витерби, а для внешнего — код Рида-Соломона с чередованием между двумя этапами кодирования [1]. Функционирование таких систем при  $E_b/N_0$ , находящемся в пределах от 0,2 до 2,5 дБ, для достижения  $P_B = 10^{-5}$  реально достижимо в прикладных задачах. В этой системе демодулятор выдает мягко квантованные кодовые символы на внутренний свёрточный декодер, который, в свою

очередь, выдает жестко квантованные кодовые символы с пакетными ошибками на декодер Рида-Соломина.

Внешний код Рида-Соломона образуется из  $m$ -битовых сегментов двоичного потока данных. Производительность такого (недвоичного) кода Рида-Соломона зависит только от числа *символьных ошибок* в блоке. Код не искажается пакетами ошибок внутри  $m$ -битового символа. Иными словами, для данной символьной ошибки производительность кода Рида-Соломона такова, как если бы символьная ошибка была вызвана одним битом или  $m$  бит. Тем не менее производительность каскадных систем несколько ухудшается за счет коррелирующих ошибок в последовательных символах. Поэтому чередование между кодированиями нужно выполнять на уровне символов (а не битов). Работа [22] представляет собой обзор каскадных кодов, которые были разработаны для дальней космической связи.

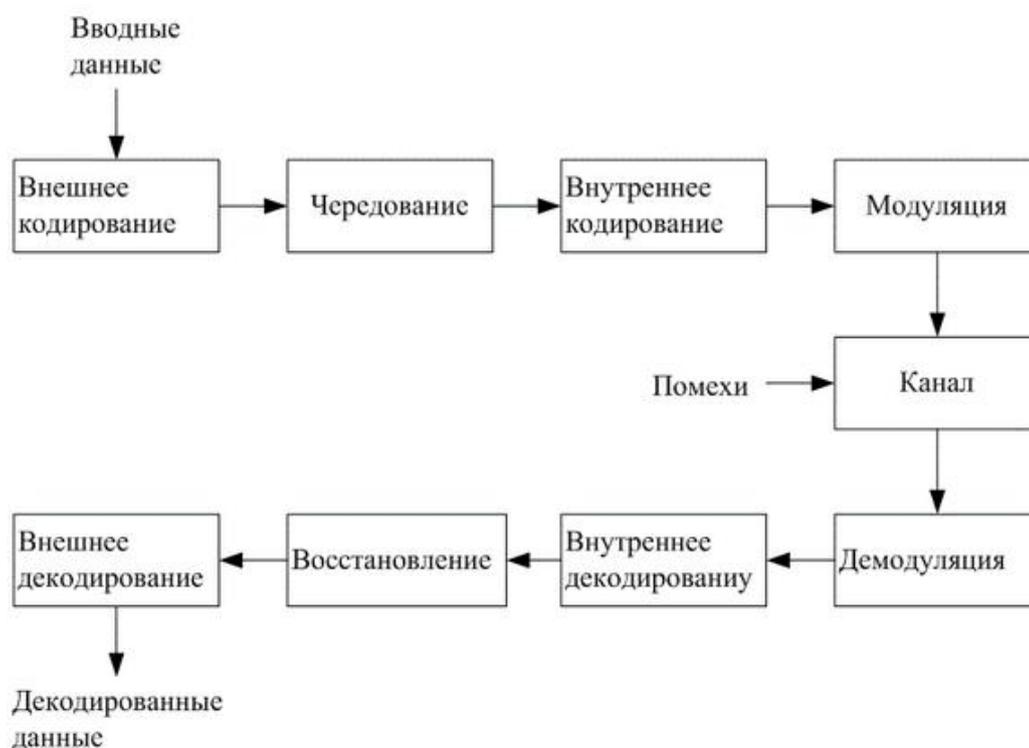


Рис. 3.139. Обобщенная структурная схема исследования каскадных кодов

Однако в данной работе исследуется случай, когда в качестве внутреннего кодирования используются турбо-коды.

**Турбо-код** – параллельный каскадный блочный систематический код, способный исправлять ошибки, возникающие при передаче информации.

Схема каскадного кодирования впервые была предложена Форни [23] как метод получения высокоэффективного кода посредством комбинаций двух или более компонентных кодов (иногда называемых составными). В результате, такие коды могут корректировать

ошибки в значительно более длинных кодах и имеют структуру, которая позволяет относительно легко осуществить декодирование средней сложности. Последовательные каскадные коды часто используются в системах с ограничением мощности, таких как космические зонды. Самая распространенная из этих схем содержит внешний код Рида-Соломона (выполняется первым, убирается последним), который следует за сверточным внутренним кодом (выполняется последним, убирается первым). Турбо-код можно считать обновлением структуры каскадного кодирования с итеративным алгоритмом декодирования связанной кодовой последовательности.

Турбо-коды впервые были введены в 1993 году Берру, Главье и Цитимаджимой. В описываемой схеме достигалась вероятность появления ошибок  $10^{-5}$  при степени кодирования  $1/2$  и модуляции BPSK в канале с белым аддитивным гауссовым шумом с  $E_b/N_0$ , равным 0,7 дБ. Коды образуются посредством компоновки двух или более составных кодов, являющихся разными вариантами чередования одной и той же информационной последовательности. Тогда как для сверточных кодов на финальном этапе декодер выдает жестко декодированные биты (или в более общем случае — декодированные символы), в каскадной схеме, такой как турбо-код, для хорошей работы алгоритм декодирования не должен ограничивать себя, подавая на декодеры жесткую схему решений. Для лучшего использования информации, получаемой с каждого декодера, алгоритм декодирования должен применять, в первую очередь, мягкую схему декодирования, вместо жесткой. Для систем с двумя составными кодами концепция, лежащая в основе турбо-декодирования, заключается в том, чтобы передать мягкую схему принятия решений с выхода одного декодера на вход другого и повторять эту процедуру до тех пор, пока не будут получены надежные решения.

Турбокоды представляют собой сравнительно новый тип кодов для исправления ошибок, возникающих при передаче цифровой информации по каналам связи с шумами. Впервые они были введены в рассмотрение в 1993 году и сразу же привлекли к себе пристальное внимание специалистов в области помехоустойчивого кодирования. Причина этому — уникальная способность турбокодов обеспечивать характеристики помехоустойчивости передачи информации, близкие к теоретически достижимым значениям при умеренной сложности реализации кодеков. Разработка турбокодов развивается по двум направлениям: свёрточные турбокоды, образованные путём параллельного соединения двух или более свёрточных кодеров, и блочные турбокоды, образованные путём последовательного соединения двух или более блочных кодеров. Как показали исследования, блочные турбокоды являются более эффективными при относительно высоких кодовых скоростях.

## Кодирование

На рисунке 3.140 представлена структурная схема турбо-кодера:

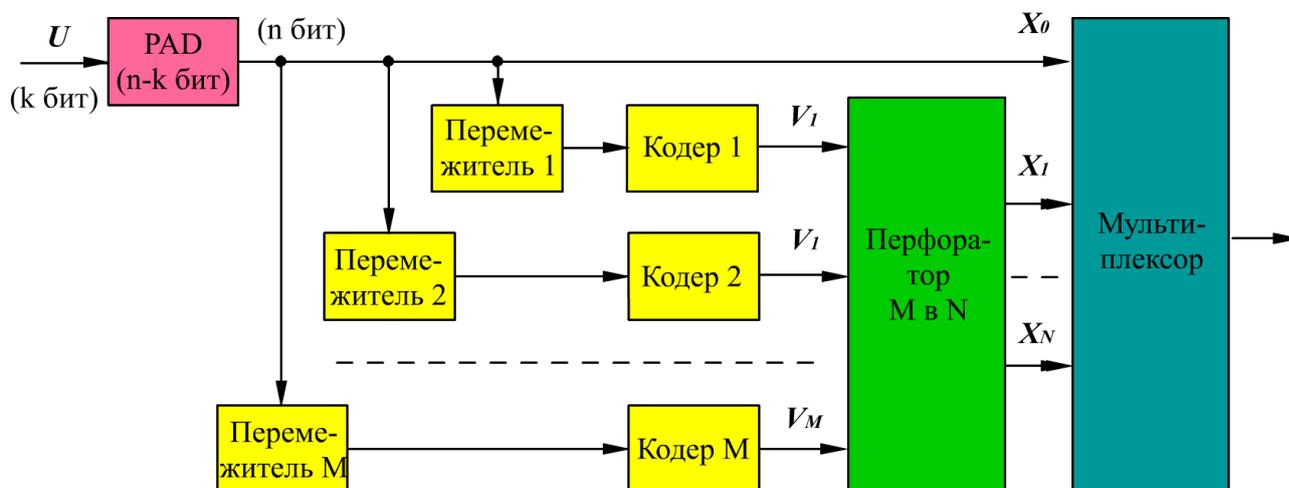


Рис. 3.140. Структурная схема турбо-кодера

Сначала на вход формирователя пакетов (PAD) поступает блок данных  $U$  длиной  $k$  бит. В формирователе пакетов к данным прибавляется ещё  $(n-k)$  дополнительных бит служебной информации, соответствующих используемому стандарту формирования пакета и включающих в себя символы его начала и окончания. То есть получается пакет  $X_0$ , состоящий из  $n$  бит.

Далее последовательность бит  $X_0$  поступает параллельно на  $M$  ветвей, содержащих последовательно соединённые перемежитель и компонентный кодер. Таким образом  $X_0$  используется в качестве входных данных сразу всеми компонентными кодерами.

В перемежителях по псевдослучайному закону происходит перемешивание поступающих бит. В отличие от посимвольного прямоугольного перемежителя, используемого в кодах Рида-Соломона, в турбо-кодах используется перемежение отдельных бит, которое подобно случайным перестановкам. Причём впоследствии, при операциях декодирования этот закон перемежения будет считаться известным. Полученные последовательности поступают на входы кодеров.

Задача перемежителя — преобразовать входную последовательность так, чтобы комбинации бит  $X_0$ , соответствующие кодовым словам с низким весом (весом называется число ненулевых бит кодового слова) на выходе первого кодера, были преобразованы в комбинации, дающие кодовые слова с высоким весом на выходах остальных кодеров. Таким образом кодеры получают на выходе кодовые слова с различными весами. При кодировании

формируются кодовые слова так, чтобы получалось максимально возможное среднее расстояние между ними (расстоянием между двумя кодовыми словами называется число бит, в которых они различаются). Из-за того, что кодовые блоки формируются из почти независимых частей, на выходе турбо-кодера среднее расстояние между кодовыми словами больше, чем минимальное расстояние для каждого компонентного кодера, а, следовательно, растёт эффективность кодирования.

Кодовая скорость — отношение длины кодового блока на входе к длине преобразованного кодового блока на выходе кодера.

В отсутствие перфоратора исходная последовательность  $X_0$  мультиплексируется с последовательностями проверочных бит, образуя кодовое слово, подлежащее передаче по каналу.

Для увеличения кодовой скорости применяется выкалывание (перфорация) определённых проверочных битов выходной последовательности. Таким образом кодовая скорость возрастает.

Если учесть, что турбо-коды оперируют с блоками большой длины с  $k > 10000$ , то  $k \approx n$ .

С помощью перфоратора, выкалывая разное число проверочных бит, возможно регулирование кодовой скорости. То есть можно построить кодер, адаптирующийся к каналу связи. При сильном зашумлении канала перфоратор выкалывает меньше бит, чем вызывает уменьшение кодовой скорости и рост помехоустойчивости кодера. Если же канал связи хорошего качества, то выкалывать можно большое число бит, вызывая рост скорости передачи информации.

### **Алгоритм декодирования по максимуму апостериорной вероятности**

Алгоритм Бала даёт «мягкую» оценку достоверности декодированного бита. То есть предьявляет на выходе степень доверия результату декодирования. В противоположность «жёсткой» структуре, при которой на выходе декодера формируется лишь наиболее вероятное значение декодированного бита («0» или «1»), при вынесении «мягкого» решения используется более подробная дискретизация выходного сигнала, характеризующая вероятность корректного приема бита. Благодаря использованию «мягких» решений в турбо-декодерах оказывается эффективным использование нескольких итераций декодирования. Апостериорная информация, полученная о кодовом слове на выходе первой итерации декодирования, поступает на вход блока следующей итерации и является для него уже априорной вероятностью. Такой подход позволяет улучшать качество декодирования от итерации к итерации. Таким образом, изменяя число итераций декодирования, можно

адаптировать декодер к текущему состоянию канала передачи и достичь требуемой вероятности ошибки на бит.

На рисунке 3.141 представлена структурная схема одной итерации турбо-декодера при двухкаскадном кодировании:

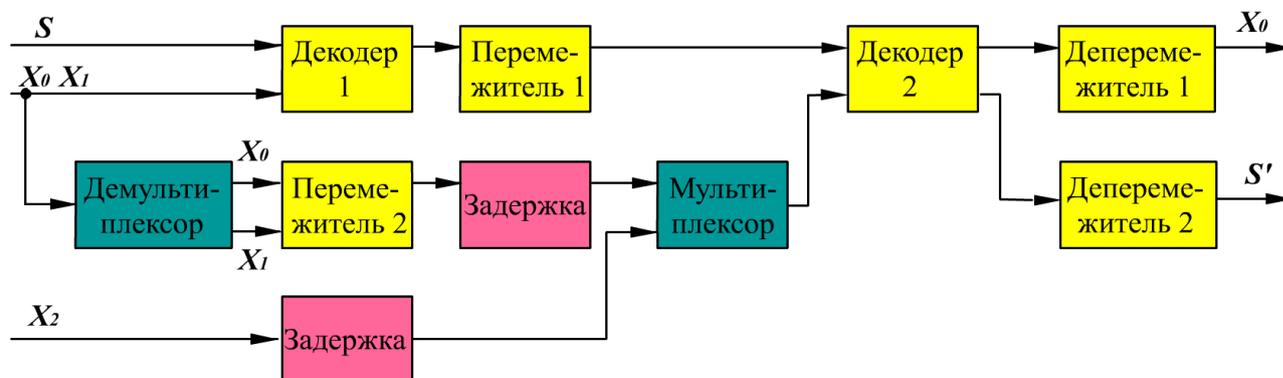


Рисунок 3.141. Структурная схема одной итерации турбо-декодера при двухкаскадном кодировании

На рис. 3.141 для простоты понимания представлен вариант схемы одной итерации турбо-декодирования при двухкаскадном кодировании. Эта схема несложно обобщается на случай произвольного количества каскадов кодирования.

Декодер для одной итерации содержит каскадное соединение двух элементарных декодеров, каждый из которых, основываясь на критерии максимума апостериорной вероятности, выносит «мягкое» решение о переданном символе. На первый декодер первой итерации с выхода демодулятора поступают «мягкие» решения символов последовательностей  $X_0$  и  $X_1$ . Таким образом, на выходе первого декодера появляется оценка информационного символа, которая после последующего перемежения попадает на вход второго декодера и является для него априорной информацией. Используя «мягкое» решение о последовательности  $X_2$ , второй декодер формирует свою оценку.

На рисунке 3.142 представлена структурная схема трёх итераций турбо-декодера при двухкаскадном кодировании:

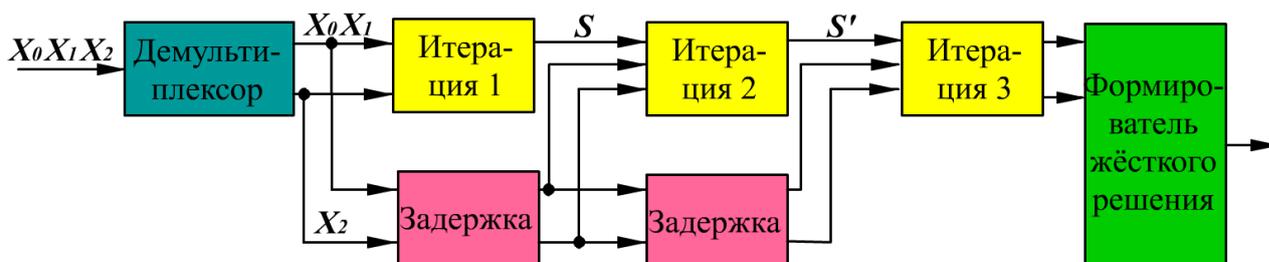


Рис. 3.142. Структурная схема трёх итерации турбо-декодера при двухкаскадном кодировании

С выхода каждой итерации решение переходит на вход следующей. От итерации к итерации происходит уточнение решения. При этом каждая итерация работает с «мягкими» оценками и на выход отдает также «мягкие». Поэтому такие схемы получили название декодеров с мягким входом и мягким выходом. Процесс декодирования прекращается либо после выполнения всех итераций, либо когда вероятность ошибки на бит достигнет требуемого значения. После декодирования из полученного «мягкого» решения производится окончательное «жёсткое».

### Преимущества и недостатки турбо-кодов

#### *Преимущества*

Среди всех практически используемых современных методов коррекции ошибок турбо-коды и коды с низкой плотностью проверок на чётность наиболее близко подходят к границе Шеннона, теоретическому пределу максимальной пропускной способности зашумленного канала. Турбо-коды позволяют увеличить скорость передачи информации, не требуя увеличения мощности передатчика, или они могут быть использованы для уменьшения требуемой мощности при передаче с заданной скоростью. Важным преимуществом турбо-кодов является независимость сложности декодирования от длины информационного блока, что позволяет снизить вероятность ошибки декодирования путём увеличения его длины.

#### *Недостатки*

Основной недостаток турбо-кодов — это относительно высокая сложность декодирования и большая задержка, которые делают их неудобными для некоторых применений. Но, например, для использования в спутниковых каналах этот недостаток не является определяющим, так как длина канала связи сама по себе вносит задержку, вызванную конечностью скорости света.

Ещё один важный недостаток турбо-кодов — сравнительно небольшое кодовое расстояние (то есть минимальное расстояние между двумя кодовыми словами в смысле выбранной метрики). Это приводит к тому, что, хотя при большой входной вероятности ошибки (то есть в плохом канале) эффективность турбо-кода высока, при малой входной вероятности ошибки эффективность турбо-кода крайне ограничена. Поэтому в хороших каналах для дальнейшего уменьшения вероятности ошибки применяют не турбо-коды, а LDPC-коды. Хотя сложность используемых алгоритмов турбо-кодирования и недостаток открытого программного обеспечения препятствуют внедрению турбо-кодов, в настоящее время многие современные системы используют турбо-коды.

### **Применение турбо-кодов**

Компании France Telecom и Telediffusion de France запатентовали широкий класс турбо-кодов, что ограничивает возможность их свободного применения и, в то же время, стимулирует развитие новых методов кодирования таких, как, например, LDPC.

Турбо-коды активно применяются в системах спутниковой и мобильной связи, беспроводного широкополосного доступа и цифрового телевидения. Турбо-коды утверждены в стандарте спутниковой связи DVB-RCS. Турбо-коды также нашли широкое применение в мобильных системах связи третьего поколения (стандарты CDMA2000 и UMTS).

### **Моделирование каскадных кодов в MATLAB Simulink [21]**

Виртуальная модель передачи данных с исправлением ошибок при помощи каскадного кода была реализована в среде Simulink Matlab. Модель демонстрирует работу кодера Рида-Соломона (внешний код) и кодера Турбо-кодов (внутренний код), позволяет исследовать исправляющую способность кодов для разных видов модуляции и сравнить её характеристики с работой указанных выше кодеров в отдельности.

На рисунке 3.143 приведена разработанная модель:

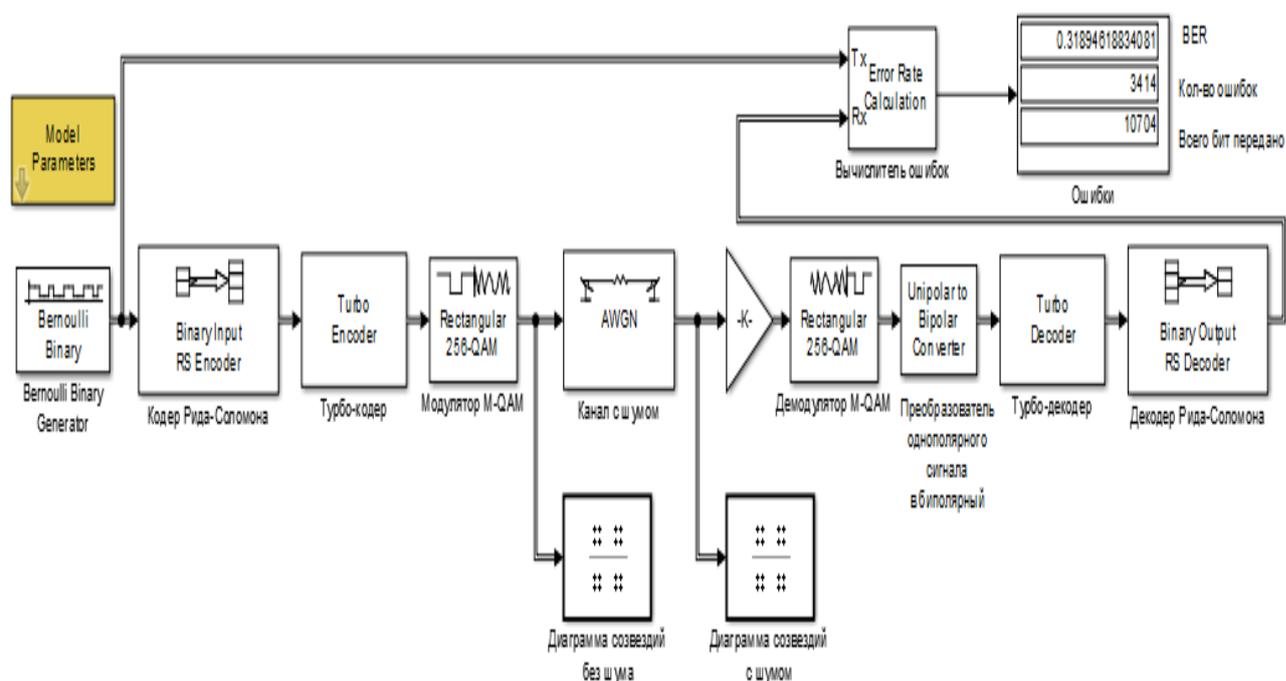


Рис. 3.143. Разработанная модель исследования каскадных кодов

В её основу положены следующие элементы, встроенные в библиотеку Simulink:

- Bernoulli Binary Generator
- Binary Input RS Encoder
- Turbo Encoder
- Rectangular QAM Modulator Baseband
- AWGN Channel
- Rectangular QAM Demodulator Baseband
- Unipolar to Bipolar Converter
- Turbo Decoder
- Binary Output RS Decoder
- Error Rate Calculation
- Discrete Time Scatter Plot Scope
- Gain
- Display (Дисплей, отражающий ошибки)

Далее представлено описание основных блоков:

**Bernoulli Binary Generator** (генератор псевдослучайной последовательности) – генерирует случайную бинарную последовательность (рисунок 3.144).

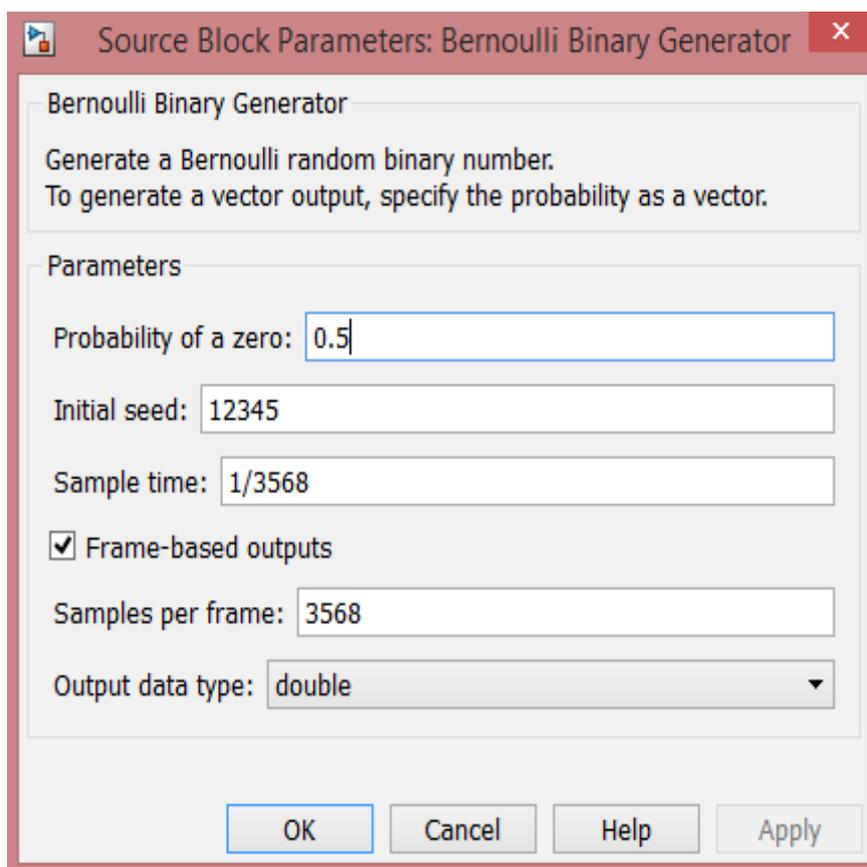


Рис. 3.144. Параметры блока «Bernoulli Binary Generator»

«Probability of a zero» - вероятность появления нуля;

«Initial seed» - начальное значение для генерации;

«Sample time» - длительность сэмпла;

«Samples per frame» - размер фрейма.

Binary Input RS Encoder – кодер Рида-Соломона (рисунок 3.7).

«Codeword length N» - общее количество бит;

«Message length K» - количество информационных бит.

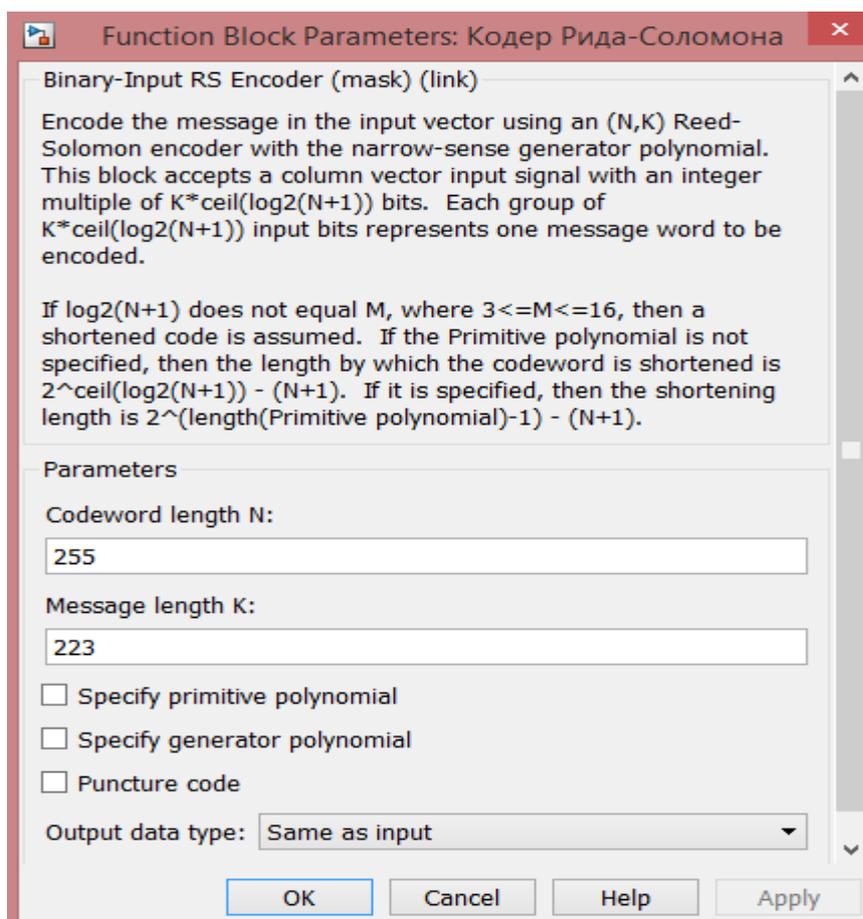


Рис. 3.145. Параметры блока «Binary Input RS Encoder»

Данный блок имеет следующую структуру внутри (рисунок 3.8):

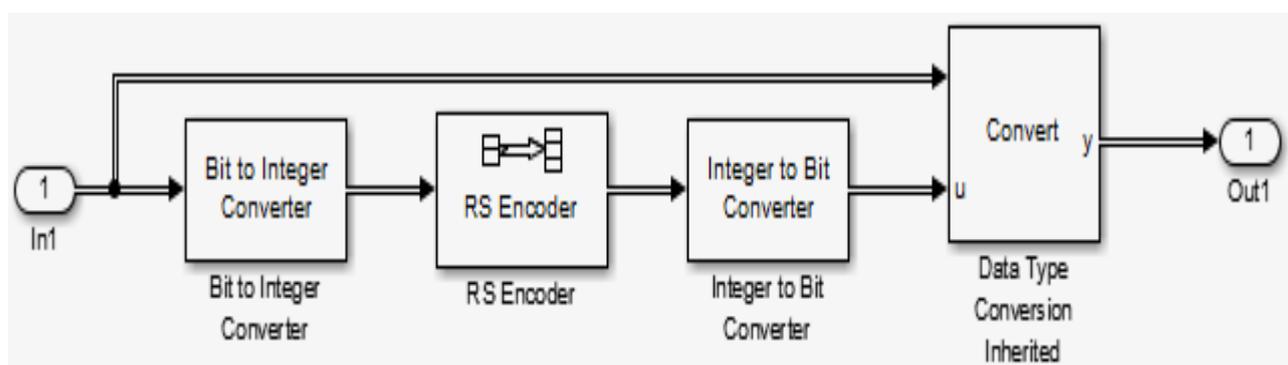


Рис. 3.146. Состав блока «Binary Input RS Encoder»

Таким образом, информационные биты, поступающие со входа генератора преобразуются в тип «Integer», кодируются и преобразуются обратно, затем полученные биты конвертируются в тот тип данных, который изначально был на входе кодера.

Turbo Encoder – Турбо-кодер (рисунок 3.147).

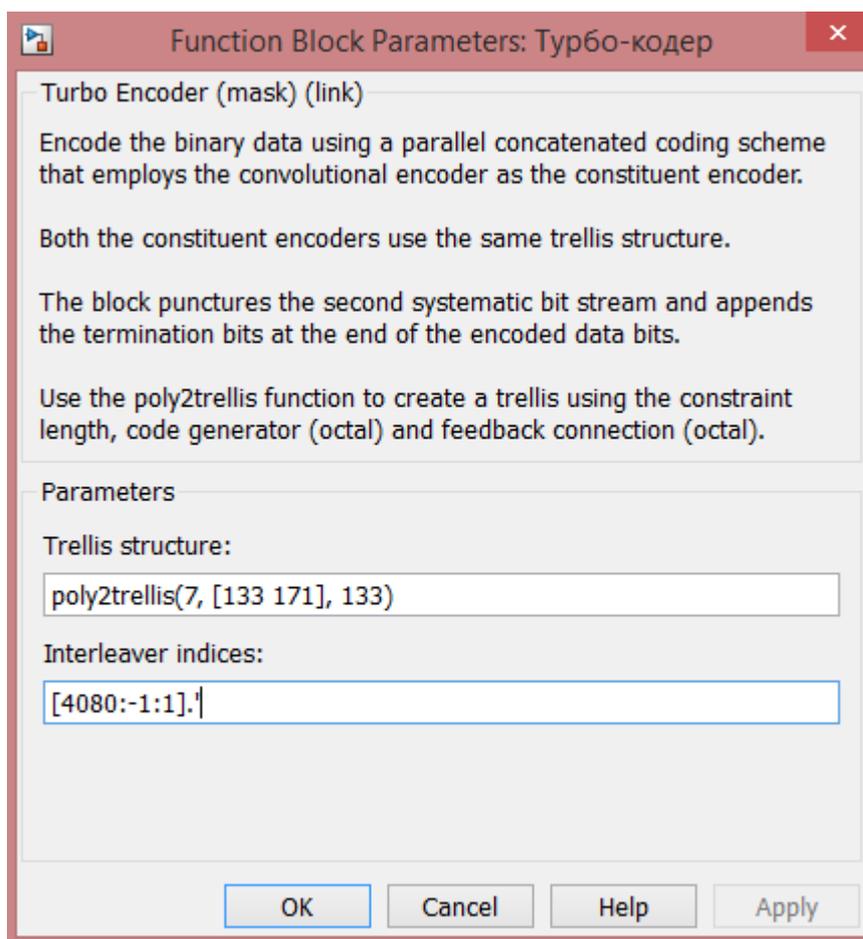


Рис. 3.147. Параметры блока «Turbo Encoder»

«Trellis structure» - структура треллис-модуляции.

Треллис-модуляция (TCM – Trellis Coded Modulation) представляет собой способ, который позволяет обеспечить повысить скорость передачи сообщения с сохранением уровня помехоустойчивости. Этот способ отличается тем, что помехоустойчивое кодирование и тип модуляции используются совместно. Выбранная соответствующим образом пара помехоустойчивый код – способ модуляции часто также носит название сигнально-кодовая конструкция (СКК).

Для кодирования использован один из наиболее часто употребляемых свёрточных кодов – код (171,133,7), который кодирует последовательность со скоростью 1/2.

«Interleaver indices» - входные параметры перемежителя.

Схема турбокодера имеет следующую структуру (рисунок 3.148):

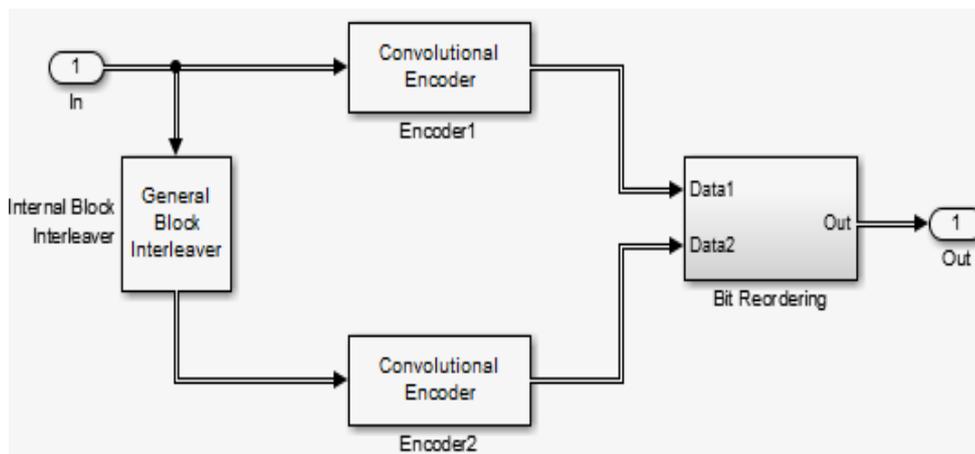


Рис. 3.148. Структура блока «Turbo Encoder»

Поток бит распараллеливается на два. В первом случае биты поступают на свёрточный кодер (133, 171), а во втором потоке биты сначала проходят перемежитель, затем поступают на аналогичный свёрточный кодер. Блок «Bit Reordering» выстраивает биты в последовательный поток.

Rectangular QAM Modulator Baseband – модулятор QAM-M (рисунок 3.149).

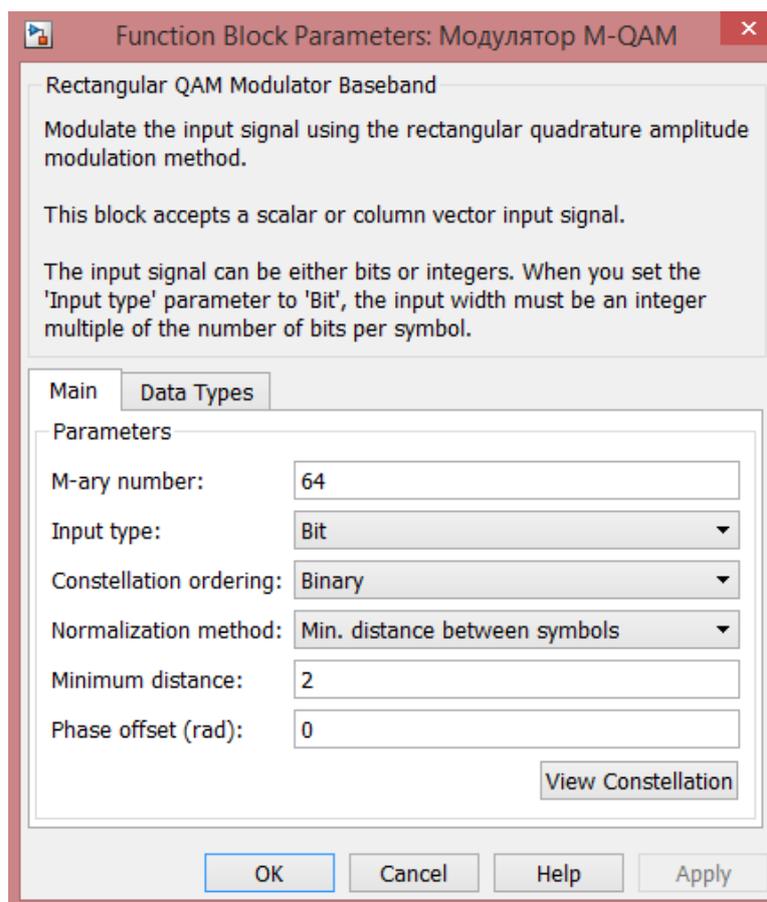


Рис. 3.149. Параметры блока «Turbo Encoder»

«M-ary number» - количество позиций в QAM-M;

«Input type» - тип входных данных;

«Constellation ordering» - порядок построения созвездия (рисунок 3.150).

Остальные параметры по умолчанию.

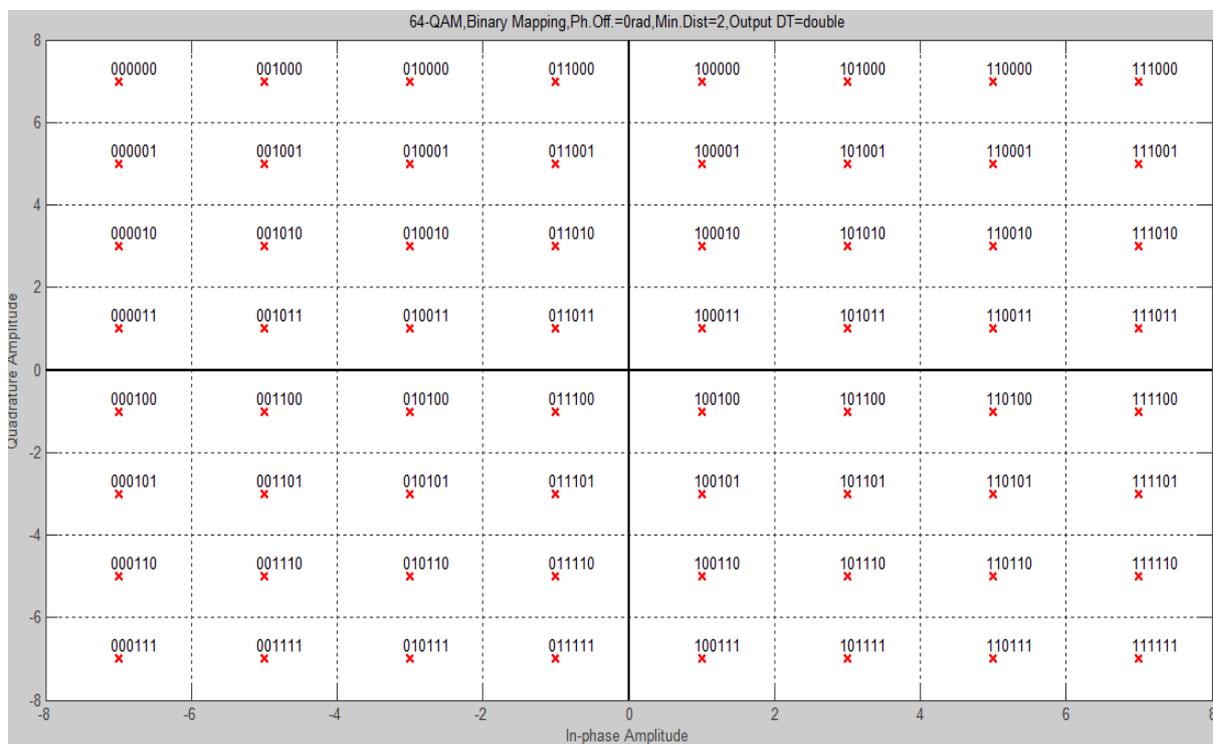


Рис. 3.150. Диаграмма построения созвездий

AWGN Channel (Канал связи) – добавляет «белый» гауссовский шум в канал (рисунок 3.13).

«Variance» - считывает параметр  $E_b/N_0$  из блока Model Parameters.

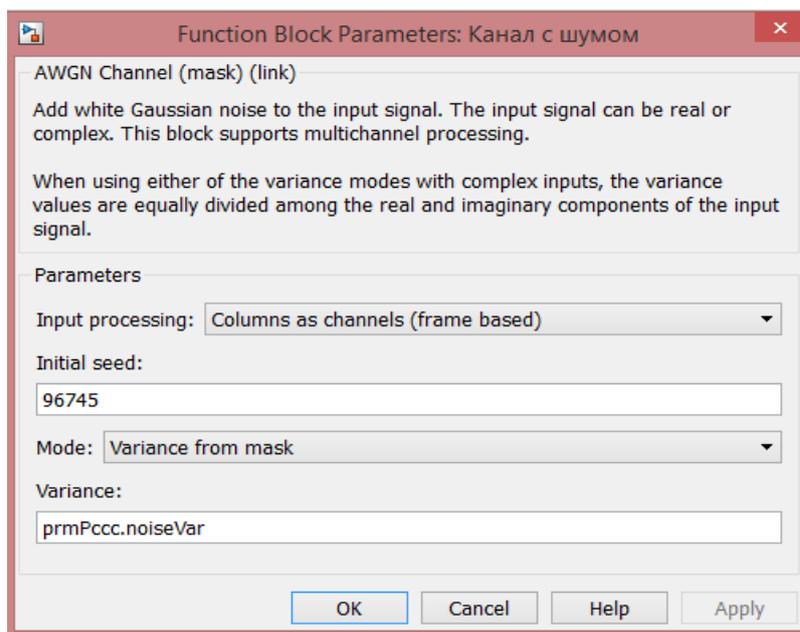


Рис. 3.151. Параметры блока «AWGN»

Discrete Time Scatter Plot Scope – Блок для отражения диаграммы созвездий.

Rectangular QAM Demodulator Baseband – демодулятор QAM-M, обладает теми же параметрами, что и модулятор.

Unipolar to Bipolar Converter – Преобразователь сигнала из однополярного в биполярный. На вход турбо-декодера необходимо подавать биполярный сигнал.

Turbo Decoder – Турбо-декодер (рисунок 3.152).

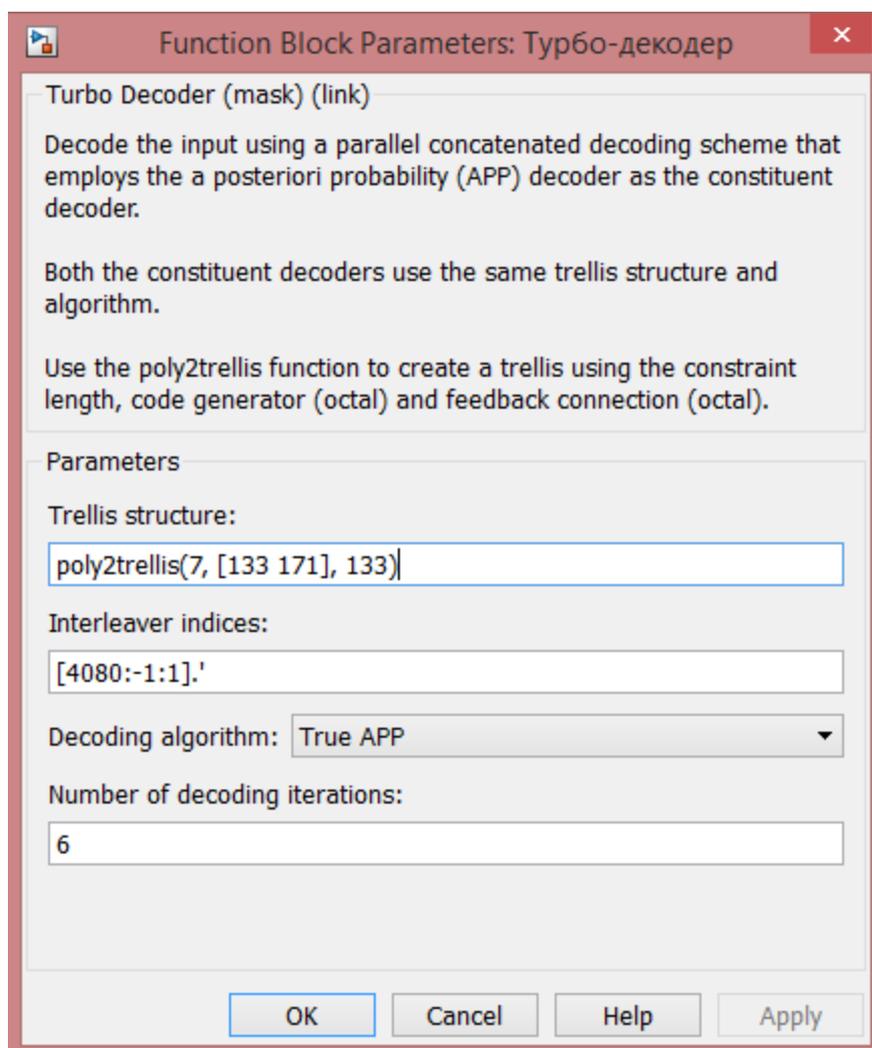


Рис. 3.152. Параметры блока «Turbo decoder»

Первые два параметра задаются аналогично параметрам кодера.

«Number of decoding iterations» – количество итераций декодирования. Декодирование в турбо-декодере происходит в несколько итераций. Чем больше итераций, тем точнее декодирование. Однако, большое количество не даёт результата, а лишь увеличивает длительность вычислений и может даже ухудшить помехоустойчивость.

Схема блока имеет следующую структуру (рисунок 3.153):

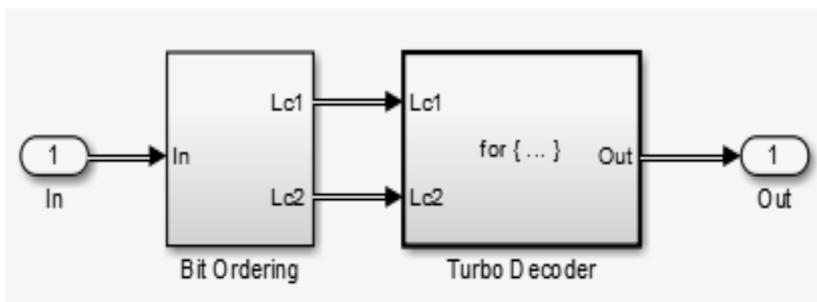


Рисунок 3.153. Структура блока «Turbo-decoder»

«Bit Ordering» - Выстраивание потока бит в параллельный поток.

Схема самого турбо-декодера представлена на рисунке 3.154:

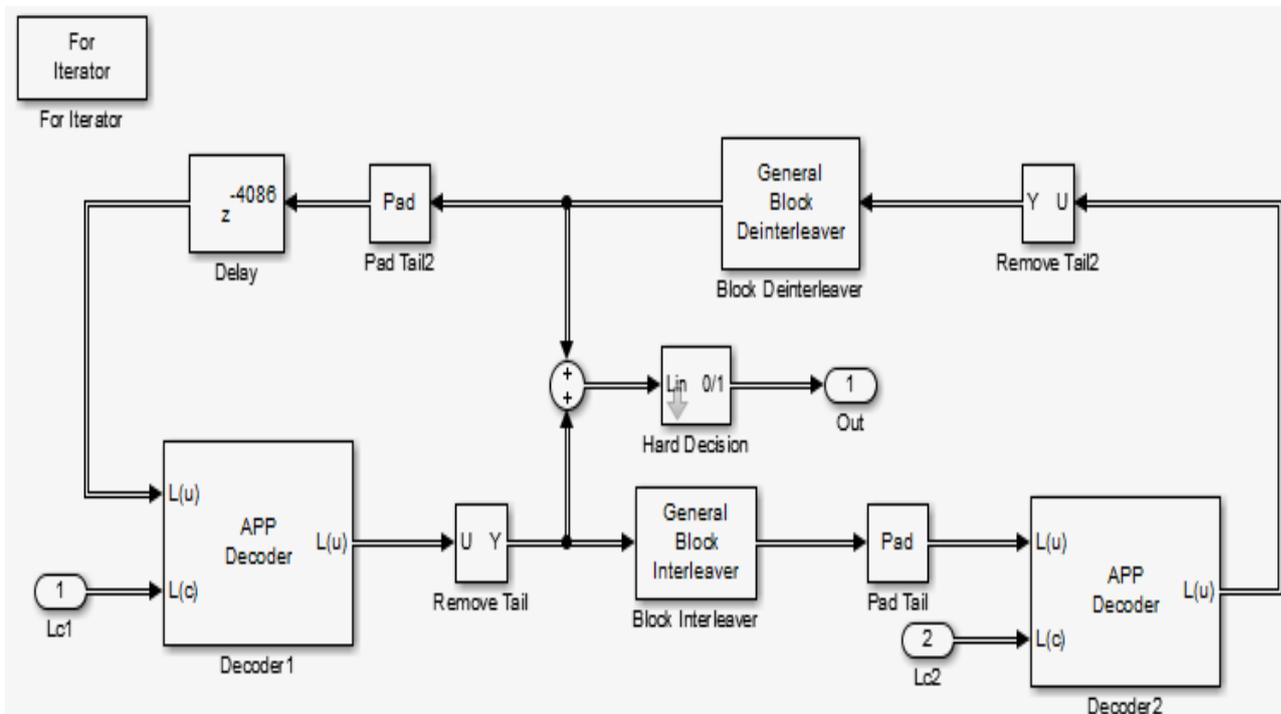


Рисунок 3.154. Схема турбо-декодера

Декодер имеет сложную структуру. Параллельный поток приходит на входы (1) и (2). Данные с входа (2) декодируются и поступают на вход блока на деперемежитель. Затем данные поступают на сумматор и через задержку на декод «Decoder1». Выход этого декодера поступает на сумматор и на перемежитель, данные с которого поступают на второй декодер. Таким образом, декодеры влияют друг на друга и помехоустойчивость и сумма их выходных значений поступает на блок принятия жёстких решений «Hard Decision». Операция декодирования повторяется столько раз, сколько указано в блоке турбо-декодера в параметре «количество итераций».

Binary Input RS Decoder – декодер Рида-Соломона. Параметры и структура аналогична блоку кодера.

Error Rate Calculation – вычислитель ошибок между переданной и принятой последовательностью.

Display - дисплей, отражающий ошибки.

### Исследование каскадных кодов

1. Спроектированная модель передачи данных демонстрирует работу каскадного кодирования (рисунок 3.143). Данная модель позволяет исследовать применение последовательно-параллельного кодирования на примере использования кодера Рида-Соломона (внешний код) и Турбо-кодера (внутренний код), а также позволяет исследовать исправляющую способность кодов для разных видов модуляции и сравнить её характеристики с работой указанных выше кодеров в отдельности.

2. В качестве турбо-кода используются два параллельных свёрточных кодера с треллис-модуляцией (для ускорения передачи данных).

На рисунке 3.155 представлен график зависимости битовой вероятности ошибки (BER) от отношения сигнал/шум в канале (SNR) для разных видов модуляции:

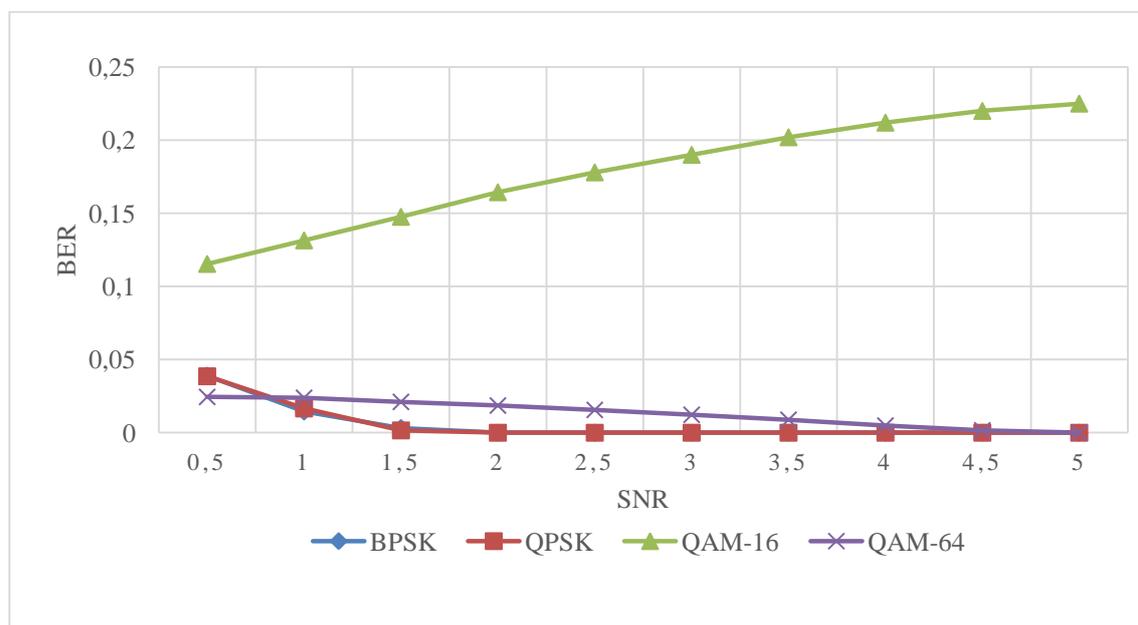


Рис. 3.155. График зависимости BER от SNR для разных видов модуляции

Из рисунка 3.155 можно заметить, что зависимость BER от SNR является неправильной в явном виде. Предположительно, данное явление связано с ошибочным программным кодом самого блока QAM-Modulator, поэтому данную зависимость рассматривать не будем. На

рисунке 3.156 представлен график зависимости BER от SNR в каскадных кодах для видов модуляции BPSK, QPSK и QAM-64:

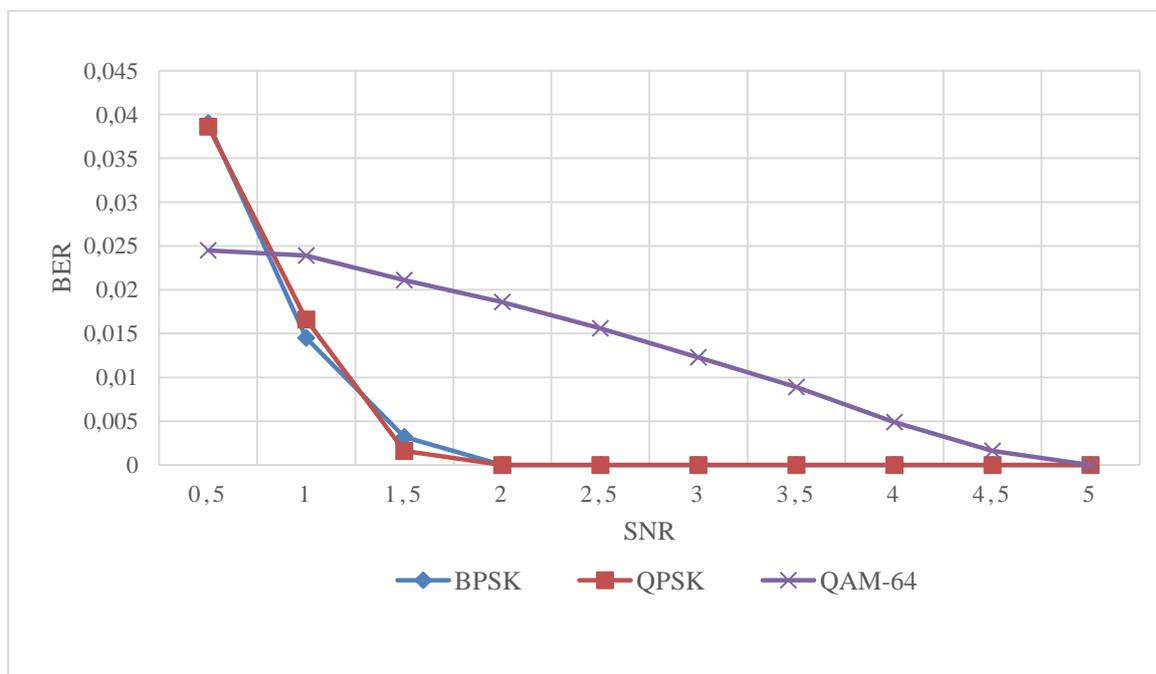


Рис. 3.156 – График зависимости BER от SNR для разных видов модуляции

На рисунке 3.157 представлена диаграмма созвездий QAM-64 сигнала на выходе передатчика:

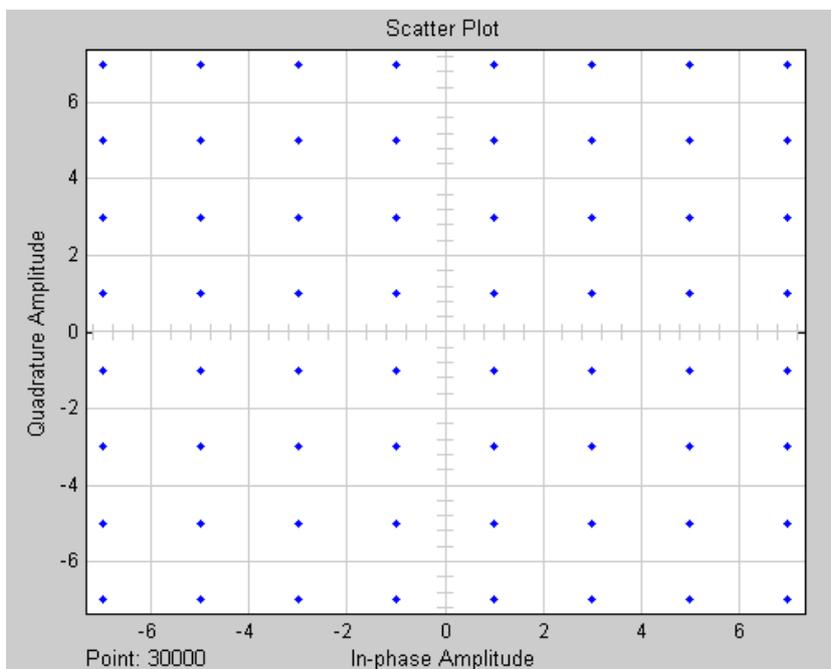


Рис. 3.158. Диаграмма созвездий QAM-64 сигнала на выходе передатчика

На рисунке 3.159 представлена диаграмма созвездий QAM-64 сигнала на приёмном конце после канала с шумом ( $\text{SNR} = 3$  дБ) :

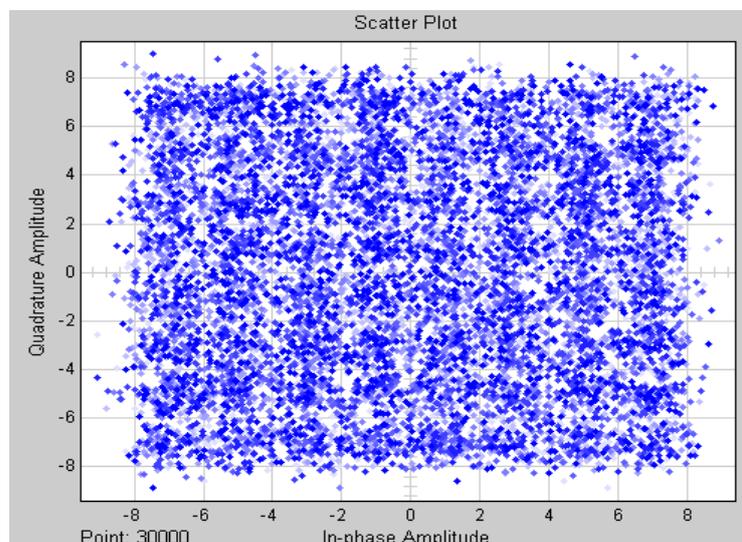


Рис. 3.160. Диаграмма созвездий QAM-64 сигнала на входе приёмника

На рисунке 4.7 представлена диаграмма созвездий QAM-64 сигнала после исправления ошибок каскадным декодером ( $\text{SNR} = 3$  дБ) :

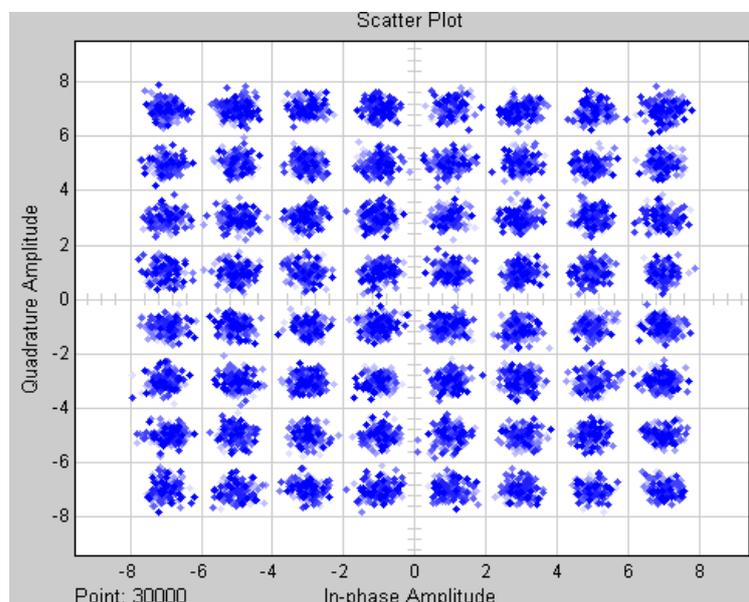


Рис. 3.161. Диаграмма созвездий QAM-64 сигнала после декодирования

На рисунке 3.162 представлены временные формы сигнала QAM-64 с каскадным кодированием ( $\text{SNR} = 3$  дБ) :

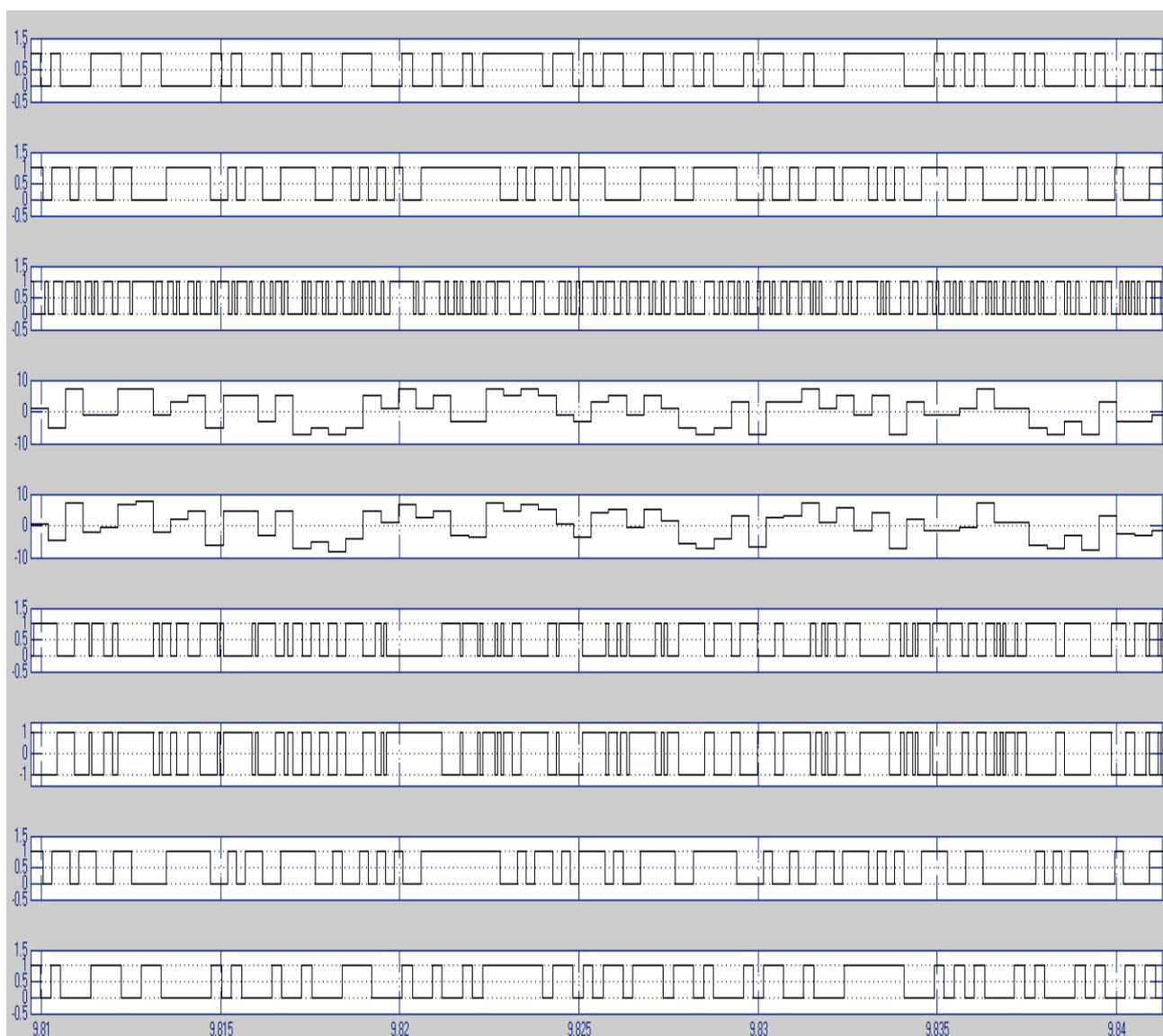


Рис. 3.162. Временные формы сигналов (сверху вниз):

- 1) на выходе генератора псевдослучайной последовательности;
- 2) на выходе внешнего кодера (РС);
- 3) на выходе внутреннего кодера (турбо-кодера);
- 4) на выходе QAM-модулятора;
- 5) на входе QAM-демодулятора;
- 6) на выходе QAM-демодулятора;
- 7) на выходе преобразователя сигнала из однополярного в биполярный;
- 8) на выходе внутреннего декодера (турбо-декодера);
- 9) на выходе внешнего декодера (РС).

На рисунке 3.163 представлен график зависимости BER от SNR при модуляции QAM-64 для каскадного кода, внешнего кода и внутреннего кода:

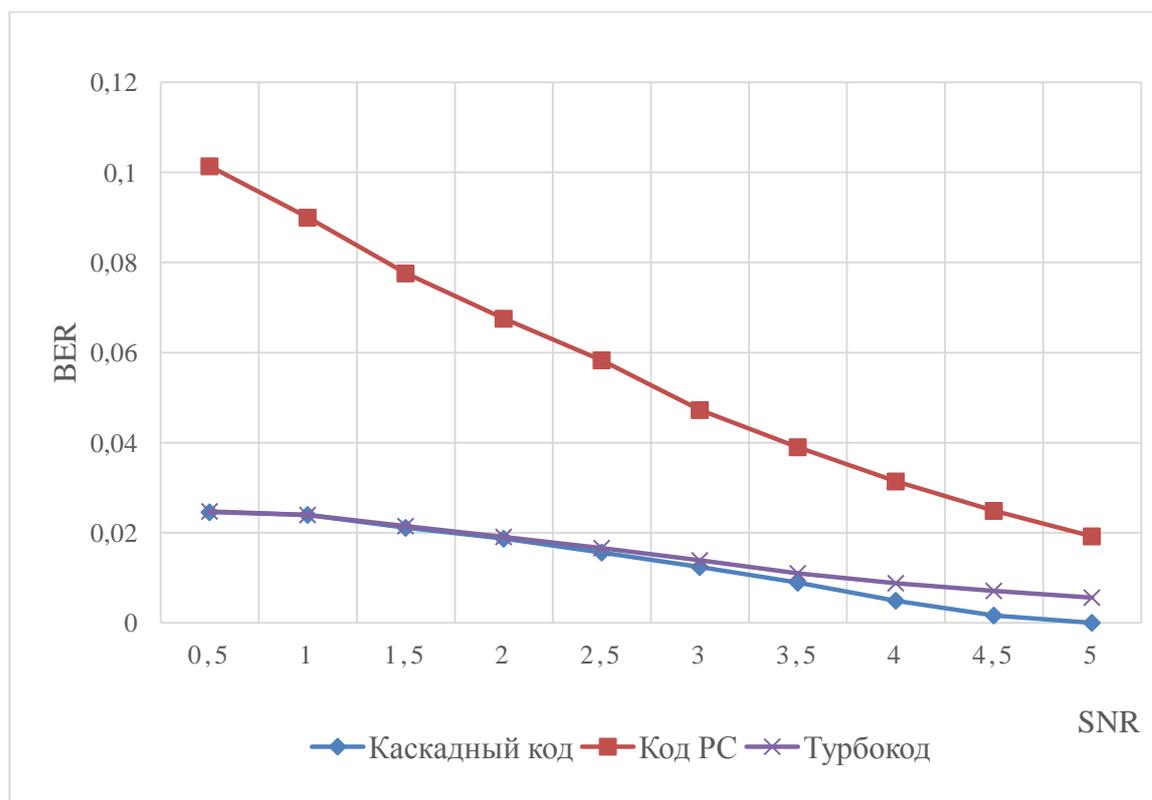


Рис. 3.163. График зависимости BER от SNR для каскадного кода, кода Рида-Соломона и турбокода

Как видно из рисунка 4.9, применение каскадного кодирования неоправданно по сравнению с применением простого турбокодирования, однако он имеет гораздо лучшую характеристику, чем применение простого помехоустойчивого кодирования (РС).

Также данная модель позволяет исследовать исправляющую способность каскадного кода в зависимости от количества итераций декодирования (рисунок 4.10).

Как видно из графика, повышение количества итераций декодирования не улучшает помехоустойчивость, а даже делает её чуть хуже, и, к тому же, приводит к повышению времени декодирования каждой посылки, кратное количеству этих итераций.

Можно сделать предположение, что повышение количества итераций необходимо при увеличении размера фрейма.

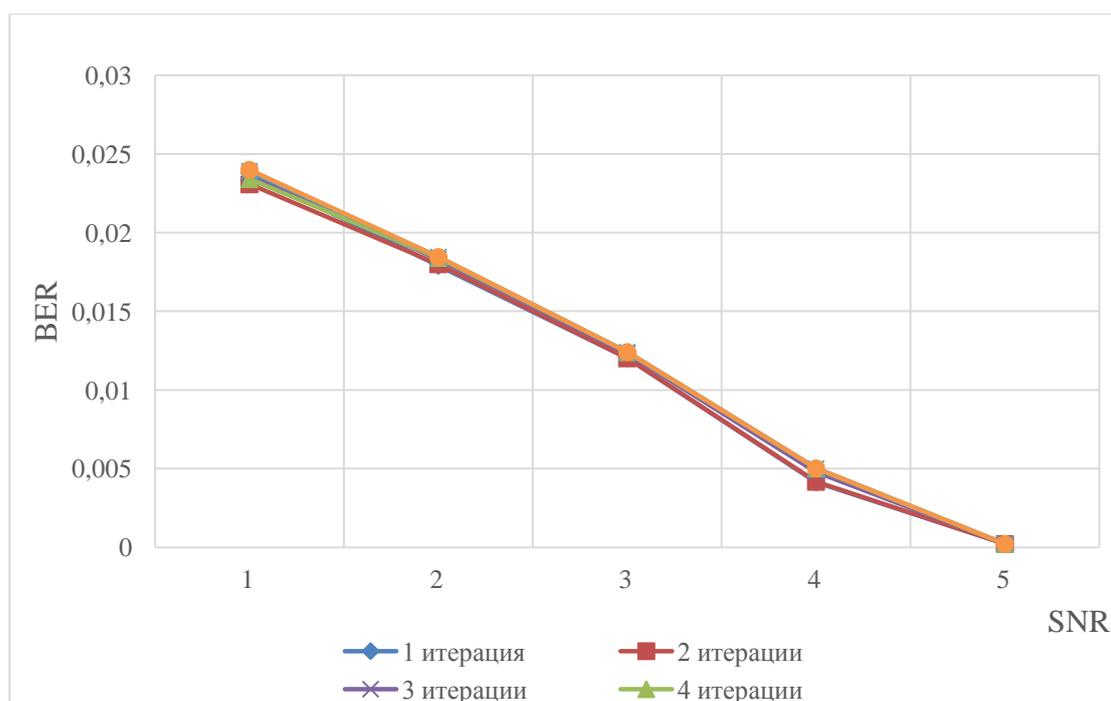


Рис. 3.164. График зависимости BER от SNR для разных значений количества итераций декодирования при количестве переданных символов  $\sim 75\ 000$ .

В результате работы спроектированы модель исследования каскадных кодов.

Модель позволяет исследовать работу каскадных кодов. В качестве внешнего кода используется код Рида-Соломона, в качестве внутреннего – Турбо-код на базе свёрточного кодирования и треллис-модуляции. Данная модель позволяет исследовать зависимость битовой вероятности ошибки (BER) системы от отношения сигнал/шум (SNR) в канале.

Получены следующие результаты и выводы:

Как видно из рисунка 4.9, применение каскадного кодирования неоправданно по сравнению с применением простого турбо-кодирования.

Повышение количества итераций декодирования не улучшает помехоустойчивость при одинаковой характеристике канала, и, к тому же, приводит к повышению времени декодирования каждой посылки, кратное количеству этих итераций.

Повышение количества итераций необходимо при изменении параметров канала.

Эффективность от применения каскадного кодирования заметна лишь при значительном размере фрейма ( $k > 10000$ ).

Методические указания позволяют использовать данные модели на лабораторных работах студентами для исследования помехоустойчивых кодов.

## **4. СИГНАЛЬНО-КODOVЫЕ КОНСТРУКЦИИ В ТЕЛЕКОММУНИКАЦИОННЫХ СИСТЕМАХ**

### **4.1. Сигнально-кодovые конструкции на основе Треллис кодовой модуляции (TCM) и их анализ с использованием MATLAB [16]**

В цифровых системах связи обычно повышают скорость передачи данных путем уменьшения энергетической емкости бита, т.е. количества энергии сигнала, приходящейся на один бит информации. Но чем меньше энергия, тем больше вероятность того, что бит будет искажен в канале при передаче. Поэтому при повышении скорости передачи разработчики всегда сталкиваются со снижением уровня помехоустойчивости.

Для повышения помехоустойчивости канала передачи данных в цифровых системах применяются коды, исправляющие ошибки. Однако действие таких кодов не всегда эффективно, так как снижается скорость передачи данных.

Треллис-модуляция (TCM – Trellis Coded Modulation) представляет собой способ, который позволяет обеспечить повысить скорость передачи сообщения с сохранением уровня помехоустойчивости. Этот способ отличается тем, что помехоустойчивое кодирование и тип модуляции используются совместно. Выбранная соответствующим образом пара помехоустойчивый код – способ модуляции часто также носит название сигнально-кодovая конструкция (СКК).

В данной работе описан способ включения сверточного кодека, используемого для передачи данных с помощью радиорелейных систем связи, в режим треллис-модуляции. Такой способ повышает скорость передачи в два раза при сохранении уровня помехоустойчивости.

В цифровых системах связи обычно повышают скорость передачи данных путем уменьшения энергетической емкости бита, т.е. количества энергии сигнала, приходящейся на один бит информации. Но чем меньше энергия, тем больше вероятность того, что бит будет искажен в канале при передаче. Поэтому при повышении скорости передачи разработчики всегда сталкиваются со снижением уровня помехоустойчивости.

Для повышения помехоустойчивости канала передачи данных в цифровых системах применяются коды, исправляющие ошибки. Однако действие таких кодов не всегда эффективно, так как снижается скорость передачи данных.

Треллис-модуляция (TCM – Trellis Coded Modulation) представляет собой способ, который позволяет обеспечить повысить скорость передачи сообщения с сохранением уровня помехоустойчивости. Этот способ отличается тем, что помехоустойчивое кодирование и тип модуляции используются совместно. Выбранная соответствующим образом пара помехоустойчивый код – способ модуляции часто также носит название

сигнально-кодовая конструкция (СКК).

В данной работе описан способ включения сверточного кодера, используемого для передачи данных с помощью радиорелейных систем связи, в режим треллис-модуляции. Такой способ повышает скорость передачи в два раза при сохранении уровня помехоустойчивости.

### **Корректирующие коды**

Наряду с многопозиционными сигналами для повышения эффективности системы электрической связи (СЭС) широко используются помехоустойчивые коды. Применение корректирующих кодов позволяет повысить верность передачи сообщений или при заданной верности повысить энергетическую эффективность системы. Это особенно важно для систем с малой энергетикой, например, систем спутниковой связи.

На практике используются как блочные, так и непрерывные коды. На рис. 4.1 приведены кривые эффективности для циклического кода Боуза-Чоудхури-Хоквингема (БЧХ) и для сверточного кода (СК) с декодированием по алгоритму Витерби.

Применение циклического кода позволяет получить энергетический выигрыш  $\Delta\beta = 2...4$  дБ, а сверточного кода  $\Delta\beta = 4...6$  дБ в обмен на снижение частотной эффективности примерно в 2 раза (3 дБ).

Энергетический выигрыш  $\Delta\beta$  от применения помехоустойчивого кодирования тем больше, чем выше требуемая верность передачи. Для непрерывного канала с белым гауссовским шумом при требуемой вероятности ошибки  $10^{-5}$  предельный энергетический выигрыш кодирования по сравнению с ФМн-2 без кодирования при оптимальном когерентном приеме составляет примерно 10 дБ. Расчетные кривые на рис. 9.2 показывают, что применение циклического кода в канале с фазовой манипуляцией (ФМн) или сверточного кода в канале с АФМ позволяет повысить одновременно энергетическую, так и частотную эффективность. Построение таких высокоэффективных систем на основе сигнально-кодовых конструкций ведет к неизбежному увеличению сложности системы. Не пропускная способность, а сложность является ограничивающим фактором при построении высокоэффективных систем. Задача состоит в том, чтобы построить систему, удовлетворяющую высоким показателям эффективности, при допустимой сложности.

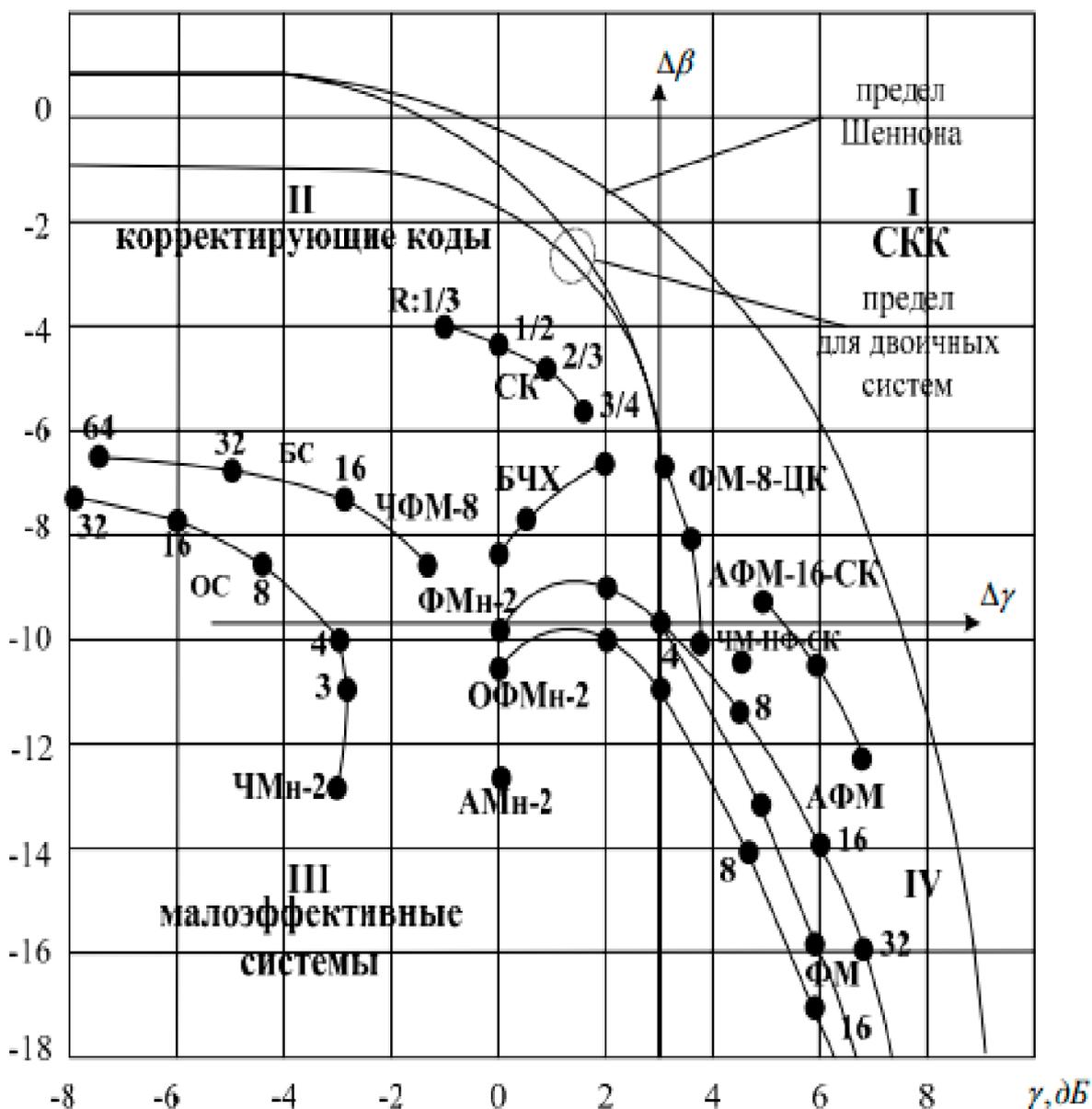
$\beta, \text{дБ}$ 

Рис.4.1. Графики зависимости энергетической и частотной эффективности систем связи

При современной элементной базе затраты на реализацию кодирующих и декодирующих устройств значительно сократились. В то же время стоимость энергетики канала практически не изменилась. Таким образом, «цена» выигрыша  $\Delta\beta$  за счет кодирования может быть существенно меньше цены того же выигрыша, полученного за счет увеличения энергетики канала (мощности сигнала или размеров антенн).

Отметим, что выбор способов кодирования и модуляции зависит от характеристик

канала. Улучшение этих характеристик, например, путем адаптации к помехам и оценивания искажений сигнала и их последующей компенсации, снижает потери в канале и создает лучшие условия для применения корректирующих кодов.

### **Оптимизация систем связи**

Повышение таких важнейших показателей систем электрической связи, как скорость и верность передачи, связано со значительными частотными и энергетическими затратами. Сравнение между собой различных СЭС осуществляется по степени использования ими основных ресурсов канала связи (пропускной способности, мощности, занимаемой полосы частот), выражаемой через показатели информационной, энергетической и частотной эффективности. Создание СЭС, в которых достигаются близкие к предельным показатели эффективности, требует совместного согласования кодека и модема с учетом статистических свойств непрерывного канала.

### **Согласование методов модуляции и кодирования**

Эффективный путь повышения удельной скорости передачи информации заключается в увеличении числа используемых сигналов  $m$  на интервале  $T$ . Однако увеличение  $m$  приводит к уменьшению расстояния между ближайшими сигналами ансамбля и снижению энергетической эффективности.

При высоких требованиях к верности передачи целесообразным становится применение помехоустойчивых кодов, которые позволяют повысить энергетическую эффективность за счет снижения частотной. Помехоустойчивое кодирование позволяет снизить необходимую величину мощности сигнала, поскольку расстояние между кодовыми комбинациями увеличивается. Одновременное требование большой скорости и верности передачи в условиях ограниченного частотного и энергетического ресурса может быть выполнено при использовании многопозиционных сигналов и помехоустойчивых кодов.

При многопозиционной модуляции, когда по каналам связи передается блок из  $n$  кодовых символов, важно также правильно выбрать манипуляционный код, определяющий правило сопоставления с каждым передаваемым сигналом определенного блока кодовых символов. Общий принцип заключается в том, что большему расстоянию по Хэммингу между кодовыми блоками должно соответствовать большее расстояние по Евклиду между отображающими их сигналами.

Создание СЭС, в которых достигаются близкие к предельным показатели эффективности, требует совместного согласования кодека и модема с учетом статистических свойств непрерывного канала. Это означает, что кодирование и модуляцию необходимо рассматривать как единый процесс формирования сигнала, а демодуляцию и декодирование – как процесс оптимального приема сигнально-кодированного блока в целом.

Согласование модуляции и кодирования сводится к поиску такого заполнения сигнального пространства, при котором обеспечивается высокая удельная скорость (сигналы расположены достаточно плотно) и одновременно высокая помехоустойчивость (сигналы достаточно далеко друг от друга).

Комбинирование различных ансамблей  $m$ -ичных сигналов, помехоустойчивых и манипуляционных кодов порождает множество конструкций. Однако только согласованные варианты обеспечивают повышение частотно - энергетической эффективности СЭС.

Эти варианты называют **сигнально - кодовыми конструкциями (СКК)**. Рассмотрим обобщенную схему передачи дискретных сообщений, приведенную на рис.4.2.

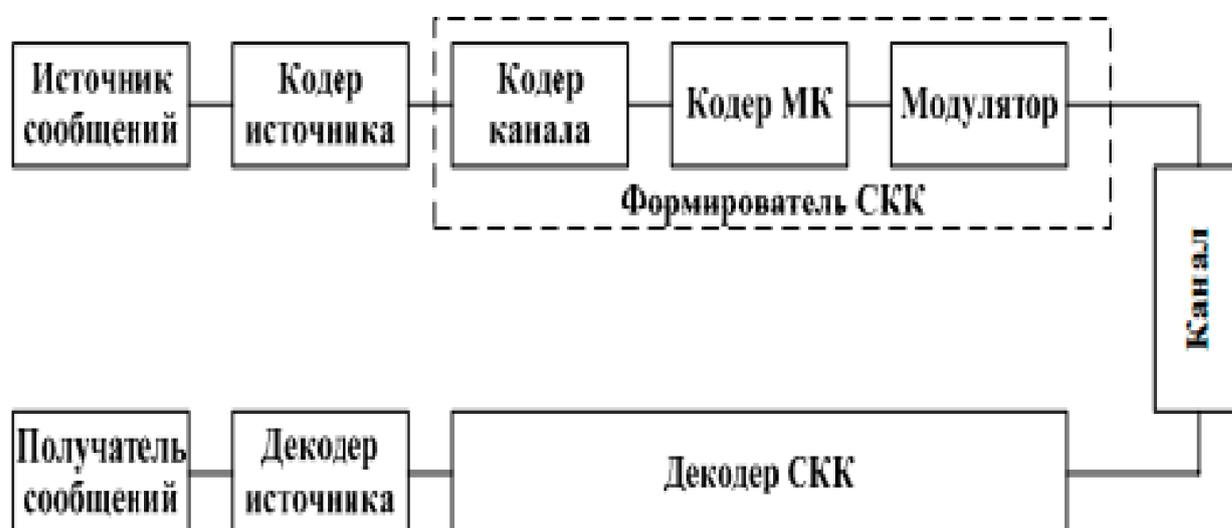


Рис. 4.2. Обобщенная схема передачи дискретных сообщений

Последовательность символов с выхода кодера канала разбивается на блоки по  $n$  символов. Отображение блоков в сигналы, формируемые модулятором, осуществляется по правилу манипуляционного кодирования, т.е. манипуляционный код определяет правило соответствия блоков кодовых символов  $m$ -ичным сигналам. Например, в случае двоичного канального кода каждому из  $m = 2^n$  кодовых блоков ставится в соответствие один из  $2^n$  сигналов.

Оптимальная процедура приема СКК заключается в обработке сигнально - кодового блока в целом. Поэтому демодулятор, декодеры канала и манипуляционный декодер рассматриваются как единое устройство – декодер СКК (рис.4.2).

Декодер СКК строится так, чтобы минимизировалась вероятность ошибки приема. Оптимальный декодер реализует принцип максимального правдоподобия. При белом

гауссовском шуме выбирается кодовое слово, находящееся на минимальном евклидовом расстоянии от принятого.

Декодирование МК можно рассматривать как последний этап обработки сигнально – кодового блока оптимальным декодером СКК. При этом декодер канала работает в метрике Евклида с сигналами, а не с их двоичными представлениями по правилу манипуляционного кода. Схема поэлементного приема, наоборот, ориентирована на применение декодера канала в метрике Хемминга, т.е. обработку двоичных величин после декодера манипуляционного кода.

Следовательно, достижение наибольшей эффективности возможно при декодировании по алгоритму максимального правдоподобия сигнально - кодового блока в целом». Необходимо отметить, что в принципе любой сигнальный ансамбль на выходе последовательно соединенных кодека и модема может быть отнесен к СКК. Введение понятия СКК отражает подход к модуляции и кодированию как процессу объединения сигналов и кодов в единую эффективную конструкцию.

#### **Классификация сигнально – кодовых конструкций**

В основе формирования СКК лежат операции отображения информационной последовательности в кодовую, путем внесения избыточности и кодовой последовательности в канальную заданием манипуляционного кода. Помехоустойчивое кодирование, повышающее энергетическую эффективность СЭС, является одной из важнейших операций формирования СКК. Получаемый при этом энергетический выигрыш от кодирования зависит от степени увеличения минимального сигнального расстояния между разрешенными кодовыми блоками. В качестве сигнального для канала АБГШ используется расстояние Евклида. Асимптотический энергетический выигрыш определяется формулой [21]:

$$\text{ЭВК}[ДБ] = 20 \lg \left( \frac{d_{ef}}{d_e} \right), \quad (4.1)$$

где,  $d_{ef}$  – минимальное евклидово расстояние между разрешенными кодовыми блоками;  $d_e$  – минимальное евклидово расстояние между различными некодированными последовательностями канальных символов одинаковой мощности с кодированными символами.

Согласно (4.1), для получения больших величин энергетического выигрыша при построении СКК необходимо подбирать коды, максимизирующие минимальное евклидово расстояние между разрешенными кодовыми комбинациями. В основу классификации СКК можно положить отличительные особенности по типам

помехоустойчивого кода, по типам ансамблей сигналов и по способам согласования модуляции и кодирования.

По типу помехоустойчивых кодов все СКК могут быть поделены на два больших класса: СКК на основе блочных кодов и СКК на основе непрерывных кодов. Кроме того, отдельный класс составляют СКК на основе каскадных кодов, в которых применяются одновременно блочные и непрерывные коды. Каждый из классов делится на группы по конкретным видам кода.

Среди блочных наиболее употребляемыми являются коды Хэмминга, Голея, БЧХ, Рида - Соломона, Рида-Маллера и др. Непрерывные коды на практике представлены сверточными кодами, которые обладают дополнительными свойствами линейности, и постоянства во времени.

При использовании сверточного кода практически удобным является случай, когда при объеме ансамбля сигналов  $m = 2^{k+1}$  скорость сверточного кода выбирается равной  $R_{\text{СКК}} = k/k+1$ . Тогда частотная эффективность у системы с кодированием и без него одна и та же. Поскольку каждый кодовый блок длиной  $(k + 1)$  переносится одним двумерным сигналом, то и СКК считается также двумерной.

Декодирование СКК ведется обычно по алгоритму Витерби, реализующему принцип максимального правдоподобия. Одно из важнейших преимуществ СКК заключается в простоте применения алгоритма Витерби для мягкого решения на выходе демодулятора.

Любая СКК вне зависимости от способа согласования модуляции и кодирования представляет собой каскадный код с ансамблем сигналов на внутренней ступени и одним или несколькими помехоустойчивыми кодами на внешней. При использовании нескольких помехоустойчивых кодов говорят о построении СКК на основе обобщенного каскадного кода.

По типу ансамблей сигналов СКК делятся на конструкции с одномерными, двумерными и многомерными сигналами. Многомерные сигналы состоят из более простых (одномерных, двумерных) сигналов. При использовании в качестве составляющих двумерных сигналов число позиций  $M$ , соответствующих каждому  $n$ -мерному сигналу, определяется выражением  $M = m^n/2$ , где  $m$  – позиционность двумерного сигнала.

Каждый  $n$ -мерный сигнал в этом случае образуется последовательностью  $n/2$  двумерных сигналов. Например, для получения многомерного сигнала с  $n = 6$  требуется последовательность из трех двумерных сигналов, например ФМн-4.

Способы согласования модуляции и кодирования условно можно разделить на две группы: согласование кодом Грея и согласование на основе разбиения ансамбля на вложенные подансамбли.

Сигнально-кодовые конструкции, принадлежащие первой группе, представляют собой результат согласования известных двоичных помехоустойчивых кодов с многопозиционным ансамблем сигналов путем использования кода Грея в качестве манипуляционного кода. Поскольку ошибки происходят за счет переходов в области соседних сигналов, то кодовые блоки, соответствующие соседним сигналам, должны различаться наименьшим числом двоичных символов.

Вторая группа включает в себя достаточно большое число типов СКК, различающихся модификациями методов согласования. Разбиение осуществляется таким образом, что подансамбли содержат равное количество сигналов, расстояния между соседними сигналами подансамблей одинаковы, а минимальные расстояния между сигналами подансамблей увеличиваются с каждым шагом разбиения. Широкое практическое применение получило согласование путем разбиения ансамбля на вложенные подансамбли, когда внешними кодами являются сверточные коды. В основе синтеза СКК со сверточными кодами лежит поиск кодов, максимизирующих евклидово расстояние, причем обычно эти коды не являются оптимальными в метрике Хэмминга. У решетчатой диаграммы, описывающей сверточные коды в метрике Евклида, переходы между состояниями промаркированы не двоичными блоками, а сигнальными точками.

Таким образом, достижение близких к предельным показателей частотно - энергетической эффективности цифровых систем связи предполагает согласование кодека и модема с учетом статистических свойств непрерывного канала. Одно из решений подобного согласования представляют сигнально-кодовые конструкции сверточного кодирования. Мягкое декодирование по алгоритму Витерби обеспечивает энергетический выигрыш порядка 3...7 дБ без расширения занимаемой полосы частот.

**Характеристики основных типов СКК. Согласование канала кодом Грея**  
Рассмотрим СКК, представляющие собой результат согласования известных двоичных помехоустойчивых кодов с многопозиционным ансамблем сигналов путем использования в качестве манипуляционного кода Грея (табл. 4.1).

Комбинации кода в табл. 4.1 получены по следующему правилу. Кодовая комбинация натурального кода складывается по модулю 2 с такой же комбинацией, сдвинутой на один разряд вправо, при этом младший разряд сдвинутой комбинации отбрасывается.

Таблица 4.1.

Результат согласования двоичных помехоустойчивых кодов с кодом Грея

Десятичное число	Натуральный двоичный код	Код Грея
0	000	000
1	001	001
2	010	011
3	011	010
4	100	110
5	101	111
6	110	101
7	111	100

Поскольку ошибки чаще происходят за счет переходов в области соседних сигналов, то кодовые блоки, соответствующие соседним сигналам, должны различаться наименьшим числом двоичных символов.

На рис. 4.3 приведены примеры кода Грея для ансамблей одномерных (АМ-4) и двумерных (ФМ-4, КАМ-16) сигналов.

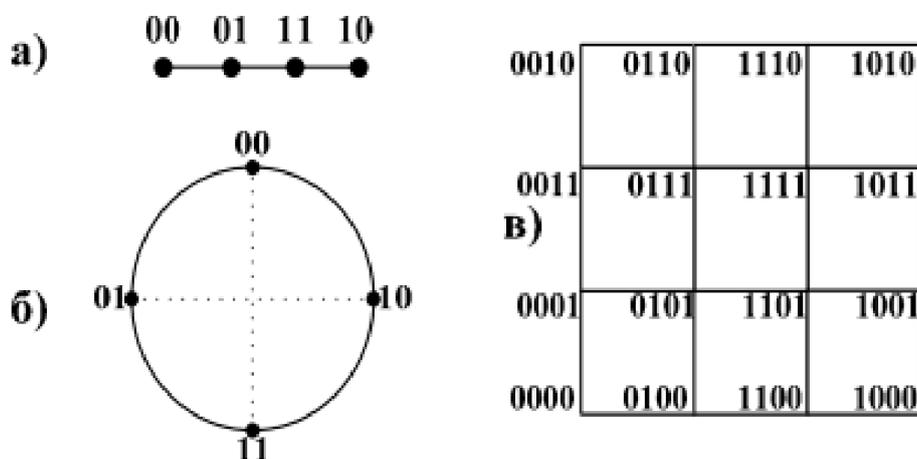


Рис. 4.3. Коды Грея ансамблей: одномерных: а) АМ-4; б) ФМ-4; двумерных: КАМ-16

Несмотря на достаточно высокие показатели энергетической эффективности при мягком решении в демодуляторе и декодировании алгоритмом Витерби, согласование кодом Грея не является оптимальным.

Двоичные коды, оптимальные по критерию максимума хэммингова расстояния, будут оптимальны и по критерию максимума свободного евклидова расстояния, если при отображении двоичных подблоков в сигнальные точки ансамбля выполняется принцип:

большему расстоянию Хэмминга  $dh \max$ , соответствует большее расстояние по Евклиду  $de \max$ .

Простейшие ансамбли сигналов АМн-2, ФМн-2, ФМн-4 этому условию для кода Грея удовлетворяют. В табл. 4.4 показаны комбинации (подблоки) двоичного кода длиной  $n = 3$ , а также расстояния  $dh$  и  $de$  при использовании кода Грея для ФМн-8 (см. рис. 4.4).

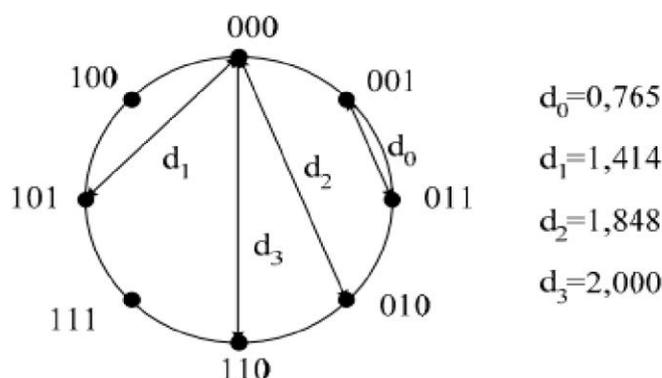


Рис. 4.4. Расстояние между сигнальными точками ФМн-8

Как следует из таблицы, сформулированный принцип соответствия большому расстоянию Хэмминга большому расстоянию Евклида для всех вариантов отображения не выполняется. Например, для комбинации 111 большому расстоянию Хэмминга  $dh = 3$  соответствует не самое большое расстояние Евклида  $dh = 1,848$ , и т.д.

#### Соответствие расстояний Хэмминга и Евклида для сигналов ФМн-8

Кодовые комбинации	000	001	011	010	110	111	101	100
$d_h$	0	1	2	1	2	3	2	1
$d_e$	0	0,765	1,414	1,848	2,000	1,848	1,414	0,765

Таким образом поскольку манипуляционный код Грея для сложных сигналов не обеспечивает оптимального согласования кодека и модема, необходимо найти методы дальнейшего повышения свободного евклидова расстояния  $def$  и, соответственно, энергетической эффективности  $\beta$ .

#### Согласование на основе разбиения ансамбля на вложенные подансамбли

В начале 80 х гг. Унгербок (Ungerboeck G.) опубликовал статью, в которой, анализируя СКК на базе ансамбля ФМн-8 и сверточного кода со скоростью  $R_{кк} = k / k + 1$ ,

сформулировал ряд правил построения СКК. Поэтому СКК построенные по этим правилам (Trellis-Coded Modulation – TCM), часто называют СКК Унгербоека.

По способу согласования модуляции и кодирования СКК Унгербоека относятся к конструкциям, полученным на основе разбиения ансамбля сигналов на вложенные подансамбли. Разбиение осуществляется таким образом, что подансамбли содержат равное количество сигналов, расстояния  $de$  между соседними сигналами подансамблей одинаковы, минимальные расстояния  $de_{\min}$  между сигналами подансамблей увеличиваются с каждым шагом разбиения; при этом левая ветвь разбиения кодируется символом «0», а правая «1». Считывание кодовой комбинации, соответствующей сигнальной точке на амплитудно-фазовой плоскости, осуществляется снизу вверх. Разбиение для ансамбля сигналов ФМн-8 представлено на рис. 4.5.

Как следует из рис. 4.5, исходный ансамбль разбивается на подансамбли при максимальном увеличении наименьших расстояний  $de_{\min}$  между сигналами внутри подансамблей  $d_0 < d_1 < d_2 < d_3$ . Разбиение осуществляется поэтапно.

В данном примере три этапа, заключающихся в разбиении каждого из подансамблей пределы предыдущего этапа на 2 равноэлементных подансамбля.

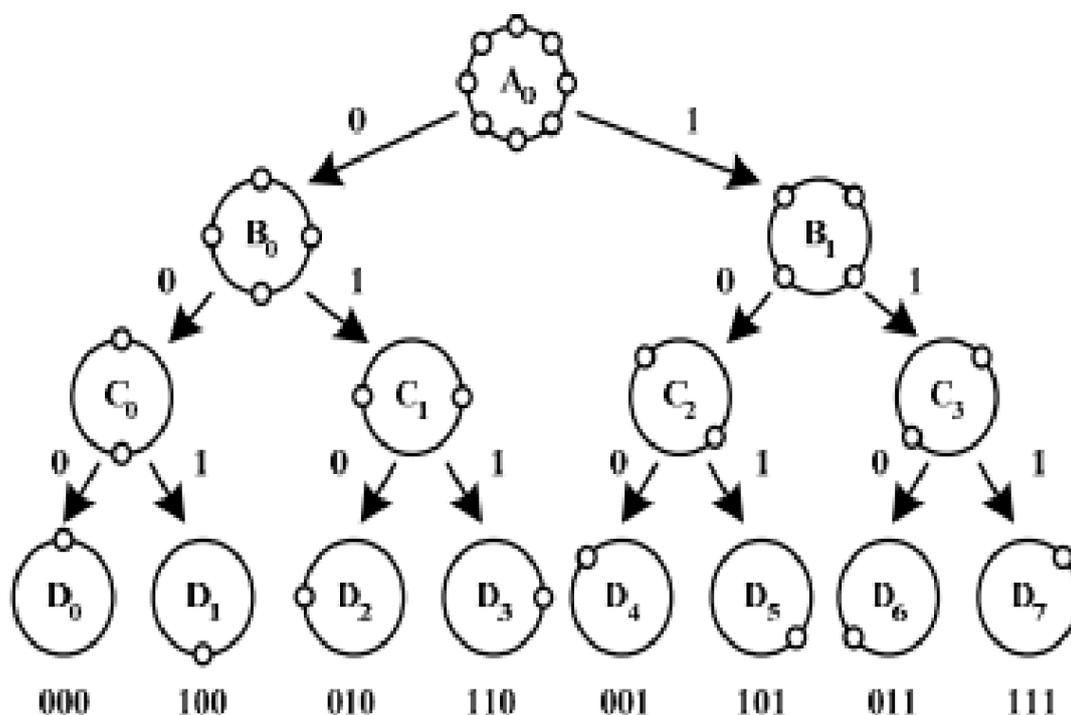


Рис.4.5. Разбиение ансамбля сигналов ФМн-8

В общем случае количество этапов  $i$  полного разбиения ансамбля из  $m$  сигналов на вложенные подансамбли определяется выражением:  $i_{\max} = \log_2 m$ , (4.6), т. е. совпадает кратностью ансамбля  $n$ . В ансамбле из  $m$  сигналов кратности  $n$  каждой сигнальной точке соответствует блок двоичных символов  $b = [b_{n-1}, b_{n-2}, \dots, b_0]$ . Соответствие между

кодовым блоком  $b$  и сигнальной точкой на плоскости определяет манипуляционный код.

Достижение наибольшей помехоустойчивости непосредственно связано с увеличением евклидова расстояния между передаваемыми сигнальными последовательностями. Решетчатая диаграмма сверточного кода (5.6.3), ребра которой промаркированы сигнальными точками, полностью отображает весь набор разрешенных сигнальных последовательностей.

Таким образом, величина свободного евклидова расстояния  $def$  зависит от маркировки ребер решетчатой диаграммы сигнальными точками (канальными символами).

Унгербок на примере ансамбля сигналов ФМн-8 (см. рис. 4.6) сформулировал четыре необходимых правила маркировки ребер сигнальными точками: все сигнальные точки используемого ансамбля сигналов должны встречаться с одинаковой частотой и с определенной степенью регулярности и симметричности;

переходы из одного и того же состояния соответствуют сигналам из подансамблей  $B_0$  или  $B_1$ ;

переходы в одно и то же состояние соответствуют сигналам из подансамблей  $B_0$  или  $B_1$ ;

параллельные переходы между состояниями соответствуют сигналам из подансамблей  $C_0$  или  $C_1$ , или  $C_2$ , или  $C_3$ . Как показывает анализ, СКК Унгербока имеют несколько более высокие частотно-энергетические характеристики по сравнению с традиционными СКК, при той же сложности реализации. Это определило их бурное внедрение в технике связи. Но известные правила построения СКК Унгербока, хотя и снижают размерность переборной задачи синтеза, но не обеспечивают гарантированное построение СКК с максимальными частотно-энергетическими характеристиками. В то же время, основной целью работ в области синтеза систем сигналов и СКК является поиск таких способов их формирования и обработки, которые при заданных ограничениях на сложность устройств формирования и приема, временные задержки, позволяли бы приблизиться к известной шенноновской границе.

При построении многомерных СКК возникает проблема выбора манипуляционного кода, поскольку известные методы его построения (правила построения кодов Грея и разбиения ансамбля на вложенные подансамбли Унгербока) не всегда позволяют согласовать евклидовы и хемминговы расстояния. Именно с этим связаны многие проблемы построения многомерных СКК.

Синтез многопозиционных ансамблей сигналов и СКК, построенных на их основе, является одним из направлений решения более общей задачи статистического согласования

вероятностных характеристик передаваемого информационного сигнала и вероятностных характеристик канала. В рамках этих традиционных задач, такое согласование осуществляется на уровне канальных символов или их блоков (супербукв канала). При этом подходы к построению алфавита таких супербукв (ансамблей сигналов и СКК) могут существенно отличаться между собой, но направлены на решение этой общей проблемы.

Известно, что ансамбль сигналов, соответствующий полному двоичному коду длины  $n$  в пространстве соответствующей размерности  $n$ , построенный заменой «1» на «-1», а «0» на «+1», соответственно, обладает практически идеальным манипуляционным кодом. Минимальным хемминговым расстояниям таких ансамблей соответствуют ребра  $n$ -мерного куба, которые характеризуются и минимальными евклидовыми расстояниями.

Кодовые комбинации и соответствующие им координаты сигнальных векторов приведены в табл. 4.2; графическое изображение ансамбля представлено на рис. 9.13. При приеме сигналов такого ансамбля минимальная ошибка (ошибочный прием одной координаты сигнальной точки) приводит к неправильному приему одного бита информации. Ошибочный прием двух координат сигнальной точки приводит к искажению двух бит информации и так далее. Однако, если рассмотреть зависимость между хемминговыми  $dh(i, j)$  и евклидовыми  $de(i, j)$  расстояниями для такого ансамбля, то можно выявить следующую закономерность, связывающую эти две величины :

$$de(i, j) = 2r \sqrt{dh(i, j)}, \quad (4.2)$$

где  $r$  – радиус сферы.

Таблица 4.2

Взаимосвязь кодовых комбинаций манипуляционного кода и координат сигнальных векторов

Манипуляционный код	Координаты сигнальных векторов
000	+1,+1,+1
001	+1,+1,-1
010	+1,-1,+1
011	+1,-1,-1
100	-1,+1,+1
101	-1,+1,-1
110	-1,-1,+1
111	-1,-1,-1

Таким образом, взаимосвязь между евклидовыми и хемминговыми расстояниями в многомерном ансамбле сигналов нелинейная, хотя большему хемминговому расстоянию

будет соответствовать большее евклидово расстояние. Если мощность и энергия сигналов являются постоянными величинами, не зависящими от номера, то ансамбли таких сигналов считают сигналами поверхностно-сферической упаковки.

В противном случае ансамбли сигналов рассматривают как объемные упаковки. Сохранение манипуляционного кода, принятого для простого трехмерного куба, в значительной мере сохраняет пропорциональность между евклидовыми и хемминговыми расстояниями и поэтому будет наилучшим и для наиболее плотного ансамбля. Для других комбинаций манипуляционных кодов для сигнальных векторов изначально не будет соблюдаться взаимная пропорциональность между евклидовыми и хемминговыми расстояниями.

Таким образом, практически невозможно создать идеальный манипуляционный код и, следовательно, целесообразно строить манипуляционные коды, у которых хотя бы частично выполняется взаимосвязь между евклидовыми и хемминговыми расстояниями.

### Практическая часть работы в MATLAB 2015 [21]

Перед началом выполнения работы, необходимо с папки TCM скопировать код в командную строку, который подгрузит необходимую схему для треллис модуляции.

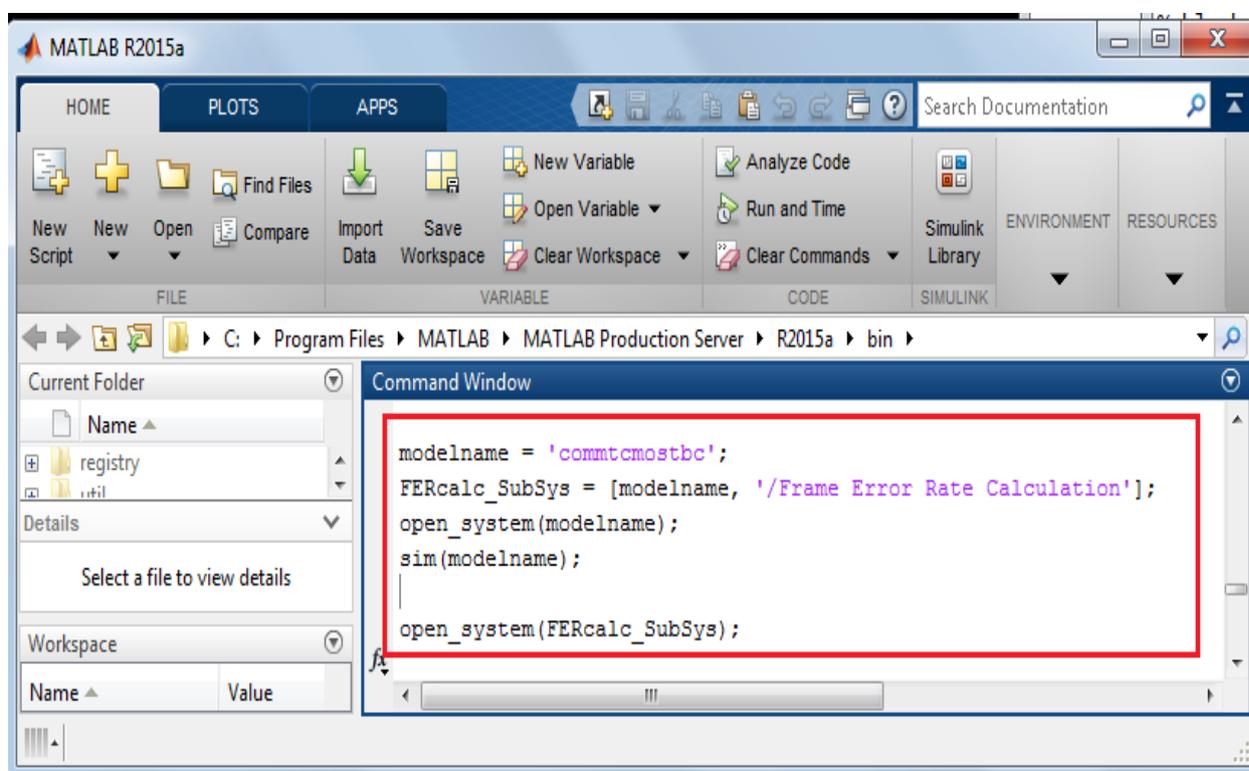


Рис. 4.6. Панель MATLAB 2015

После, откроется следующая схема:

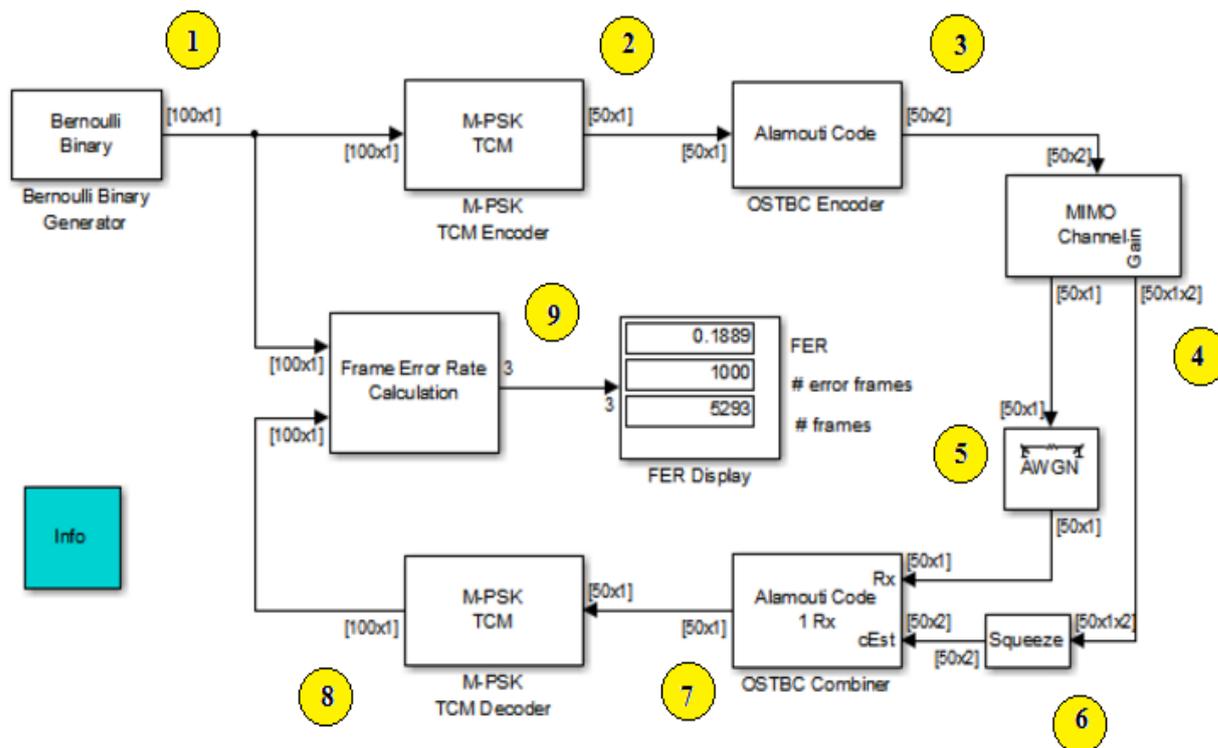


Рис. 4.7. Схема для исследования TCM в Simulink MATLAB 2015

1. Блок Двоичный генератор Бернулли генерирует случайные двоичные числа, используя распределение Бернулли. Выставьте ниже представленные параметры:

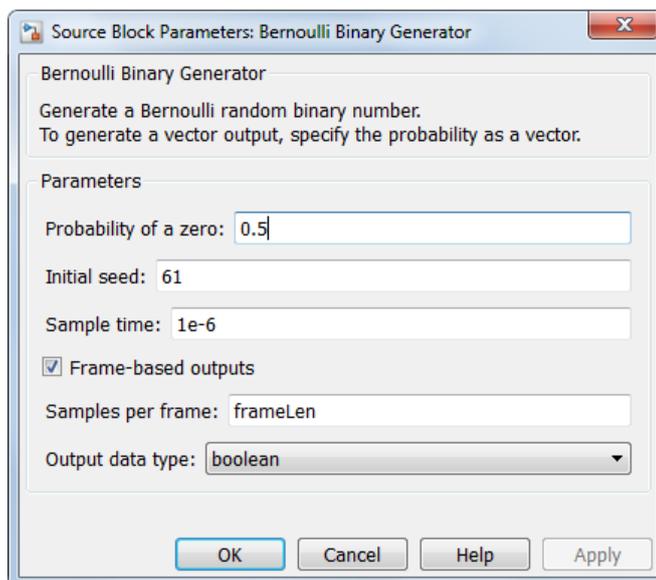


Рис. 4.8. Параметры источника (Source Block Parameters)

2. Блок M-PSK TCM кодер преобразовывает данные, полученные от блока Бернулли в PSK созвездия с заданной его средней энергией. В этом примере мы используем схему TCM 8 - PSK созвездия для 8 решетчатых состояний. Выставьте ниже представленные параметры:

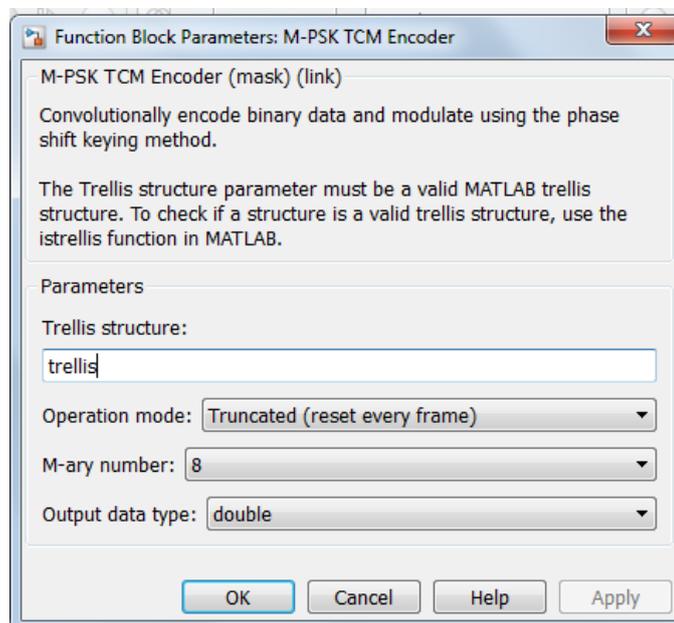


Рис. 4.9. Параметры блока: M-PSK TCM Encoder

После 2го блока нужно добавить диаграмму созвездий, который расположен в панели программы Matlab. Выставьте ниже представленные параметры:

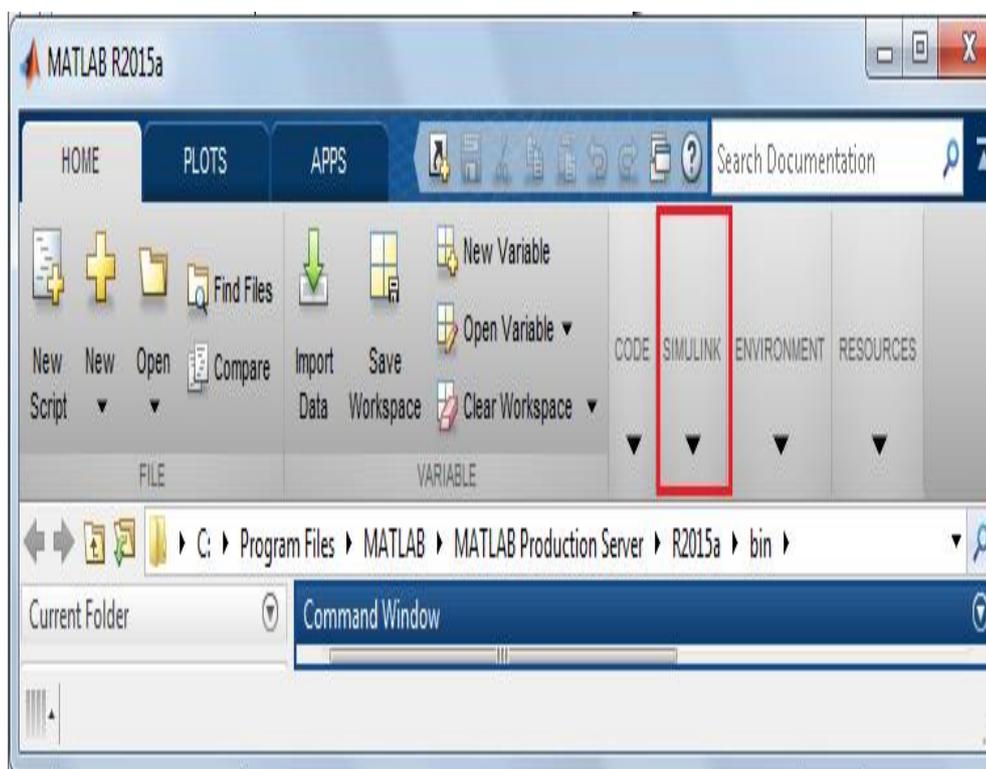


Рис. 4.10. Панель MATLAB

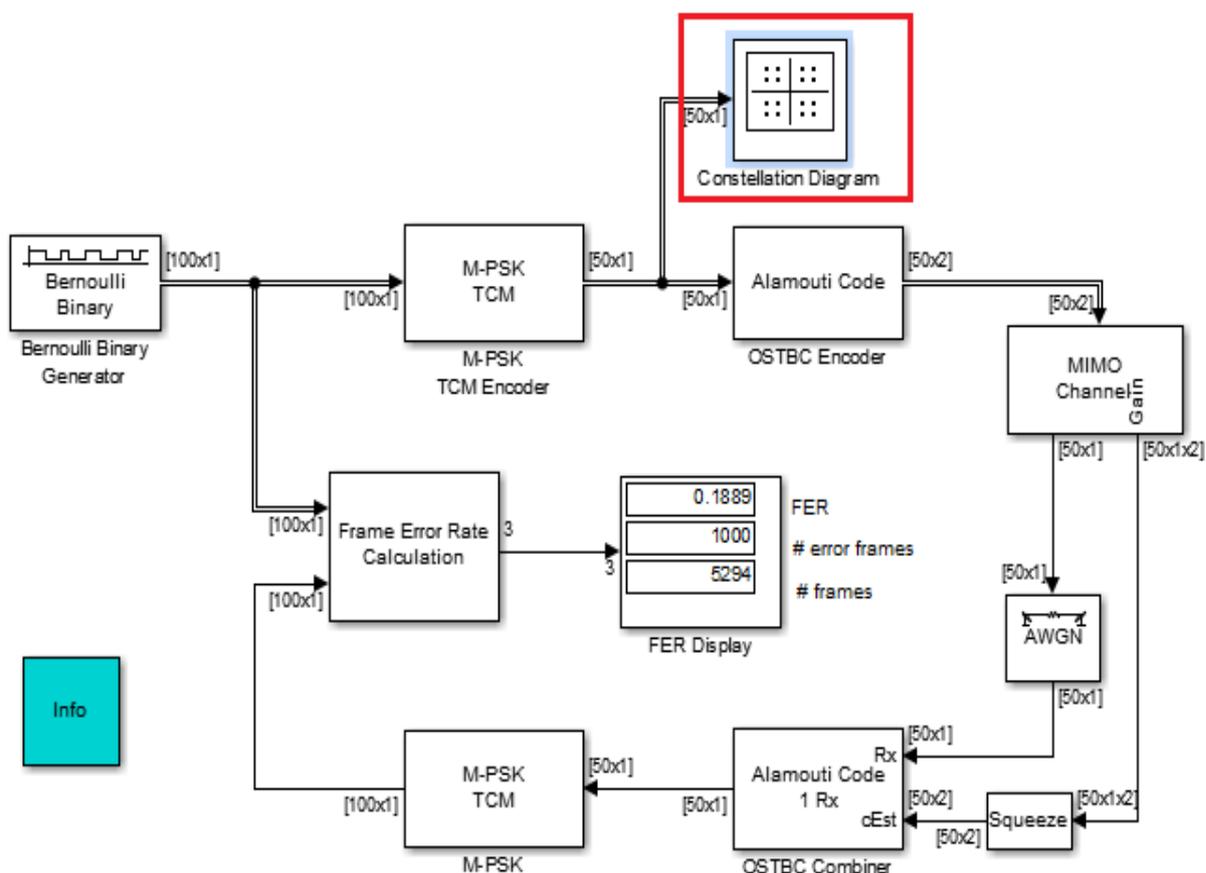


Рис. 4.11. Схема для исследования TCM в Simulink MATLAB 2015

3. Блок OSTBC (ортогонально пространственно-временные блочные коды) кодирует информационные символы из TCM кодировщика, используя код Alamouti для 2х передающих антенн. Выходом этого блока является матрица размером  $50 \times 2$ , элементы которой соответствуют данным колонки, передаваемых по одной антенне.

Блок OSTBC сочетает в себе полученные сигналы от приемной антенны с информацией о состоянии канала (CSI), которые затем подают в M-PSK TCM декодера. CSI известно на стороне приемника. Выставьте ниже представленные параметры:

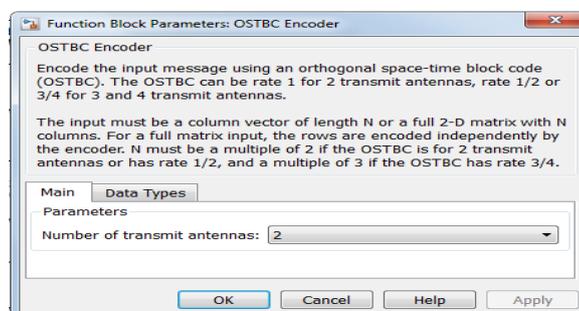


Рис. 4.12. Параметры блока: OSTBC Encoder

4. MIMO канал разносит передающие и приёмные антенны так, чтобы корреляция между соседними антеннами была слабой. Выставьте ниже представленные параметры:

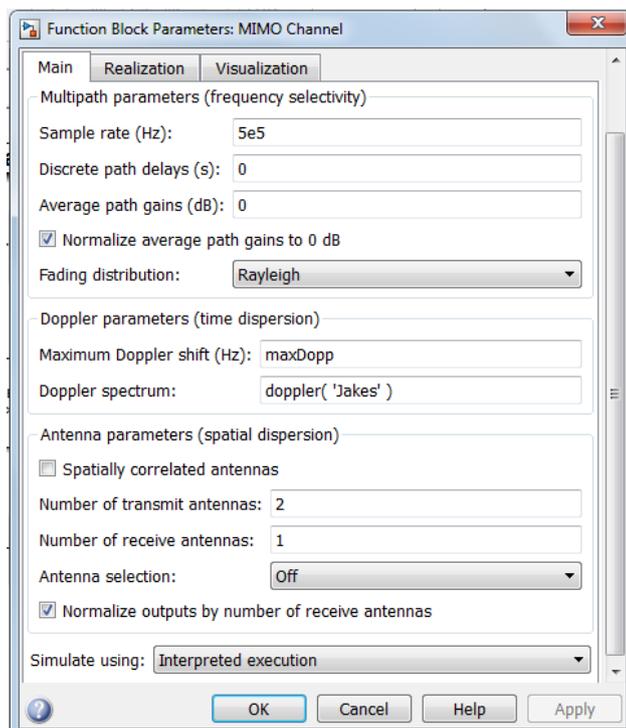


Рис. 4.13. Параметры блока: MIMO Channel

5. Блок AWGN добавляет белый гауссовский шум на приемной стороне. Выставьте ниже представленные параметры:

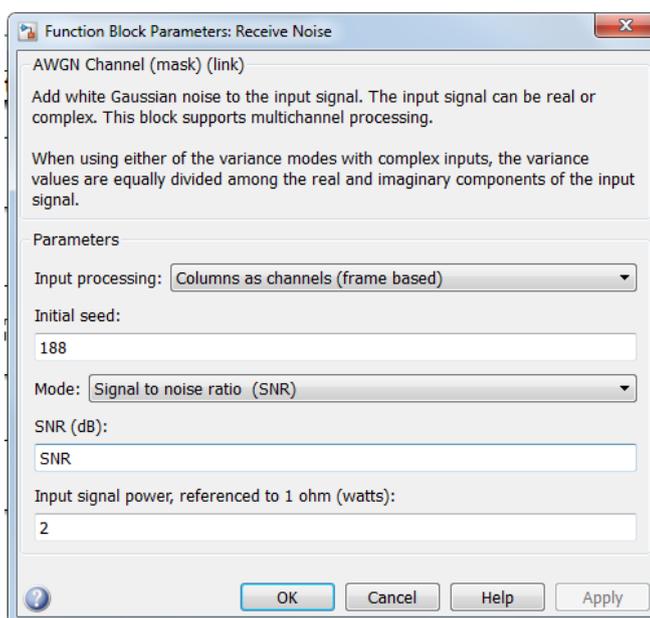


Рис. 4.14. Параметры блока: Receive Noise

6. Блок Сжатия. Выставьте ниже представленные параметры:

7. Блок OSTBC дает оценку канала сигнала. Выставьте ниже представленные параметры:

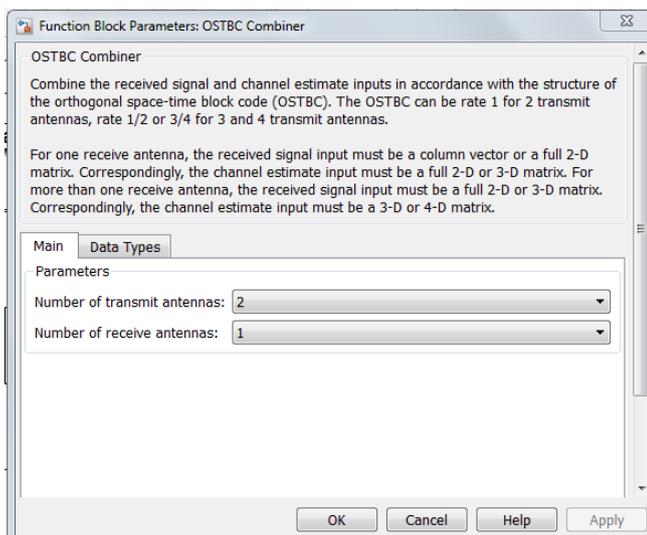


Рис. 4.15. Параметры блока: OSTBC Combiner

8. Блок M-PSK TCM декодер выполняет декодирование сигнала ранее модулированного с использованием PSK созвездия на входе. Для декодирования используется алгоритм Витерби. Выставьте ниже представленные параметры:

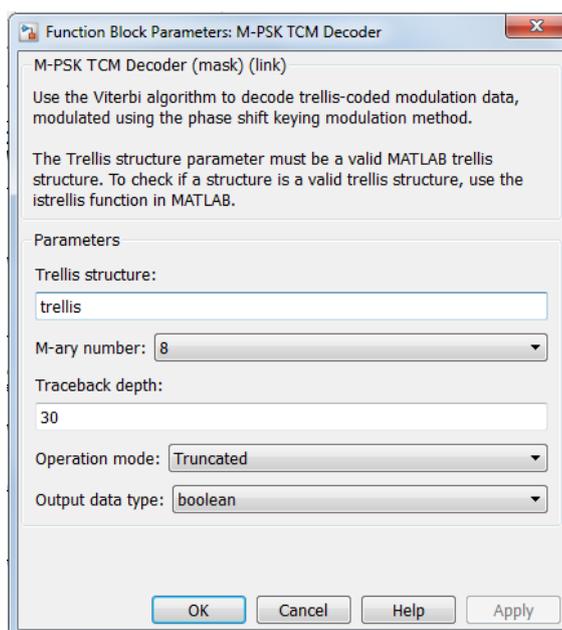


Рис. 4.16. Параметры блока: M-PSK TCM Decoder

9. Блок FER Display сравнивает декодированные биты с исходными битами в кадре для обнаружения ошибок и в онлайн режиме обновляет показатели. Этот блок состоит из трех показателей: количество ошибок в кадре, количество наблюдаемых кадров и количество обработанных ошибок в кадре. Выставьте ниже представленные параметры:

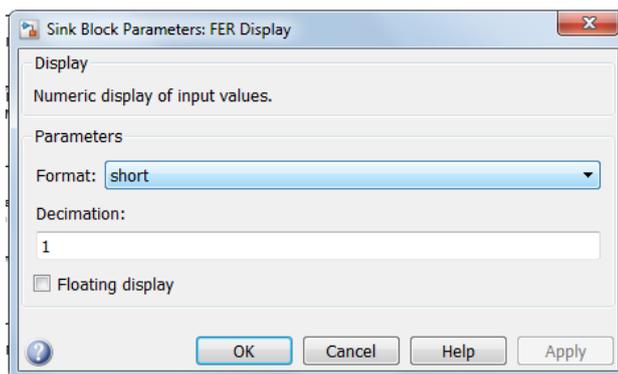


Рис. 4.17. Параметры блока: FER Display

Таблица 4.3 – Результаты работы

FER	1	4	0,99	0,88	0,62	0,28	0,18		
SNR	0	2	73	89	13	8	10	12	14

Таблица 4.4 – Результаты работы

FER	1	5	0,99	0,88	0,61	0,27	9	0,08	0,04
SNR	0	2	2	4	6	8	10	12	14

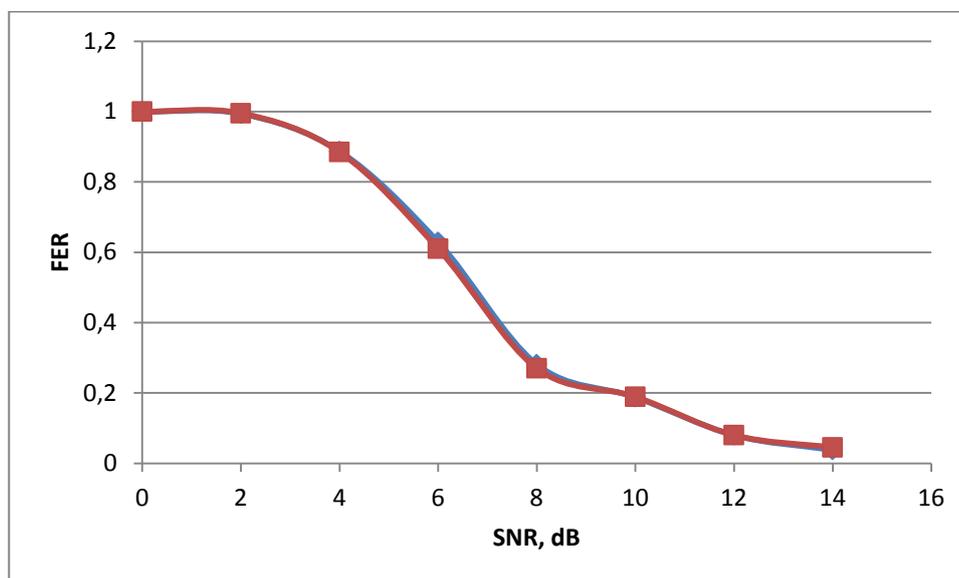


Рис. 4.18. Зависимость FER от отношения сигнал/шум

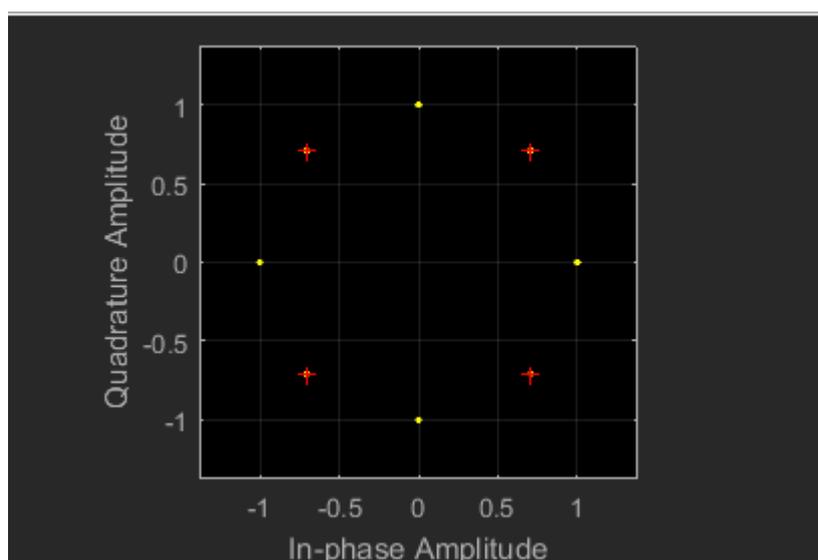


Рис. 4.19. Диаграмма созвездия 8-PSK

Таким образом, в данной работе на примере сверточного кодирования были рассмотрены преимущество и технология применения треллис-модуляции для повышения скорости передачи данных.

Переход в режим треллис-модуляции позволил сохранить тот же уровень помехоустойчивости (3 дБ) при увеличении скорости передачи информации в 2 раза. Это достигается с помощью совместного использования сверточного кода со скоростью  $2/3$  и фазовой манипуляции 8PSK. Использование треллис-модуляции является простым и эффективным решением для повышения скорости передачи в цифровых высокоскоростных системах связи.

#### **4.2. Исследование сигнально-кодовой конструкции на базе системы с ортогональным частотным мультиплексированием и пространственно-временным кодированием OFDM - MIMO с использованием NI LabVIEW [18]**

В последнее время все активнее ощущается рост беспроводных систем связи. Развитие технологий мобильных устройств, беспроводных локальных сетей (WLAN) и стремительный рост Интернет вызывают все возрастающую потребность в увеличении емкости мобильных сетей. Также наблюдается все большая интеграция сотовых сетей с сетями передачи данных, например GPRS в GSM сетях, а также сети 3G. Однако существующие технологии не могут удовлетворить новых потребностей по емкости сети, скорости передачи и стоимости услуг.

OFDM - ортогональное частотное мультиплексирование – это схема модуляции, которая позволяет быстро и эффективно передавать данные даже в каналах с многолучевым распространением сигнала. Передача ведется одновременно на большом количестве несущих частот. Эти несущие имеют небольшое разнесение по частоте и их спектры образуют групповой спектр OFDM сигнала. Частотное разнесение и синхронизация подобраны так,

чтобы несущие были ортогональны между собой, то есть не оказывали влияния друг на друга, несмотря на перекрытие по спектру. Другой важный подход, позволяющий существенно улучшить, по сравнению с традиционными системами, спектральную эффективность и помехоустойчивость системы - применение разнесения как на передающей, так и на приемной стороне путем использования нескольких передающих и нескольких приемных антенн (MIMO). Эта технология уже применяется в некоторых современных стандартах, например IEEE 802.16 и LTE.

### Особенности распространения сигнала

В мобильной радиосвязи параметры канала изменяются во времени, поскольку перемещение абонента в пространстве приводит к изменению условий распространения сигнала. Скорость изменения условий распространения определяет скорость замираний. В большинстве случаев мобильный канал связи характеризуется отсутствием прямой видимости между передатчиком и приемником, особенно в условиях плотной городской застройки. Если имеется большое число многократно отраженных лучей и отсутствует прямая видимость, огибающая полученного сигнала может быть статистически описана с помощью релевской функции плотности вероятности. Характер замираний в этом случае называют релевским. На Рис. 4.20 показан многолучевой характер распространения сигнала вследствие отражений.

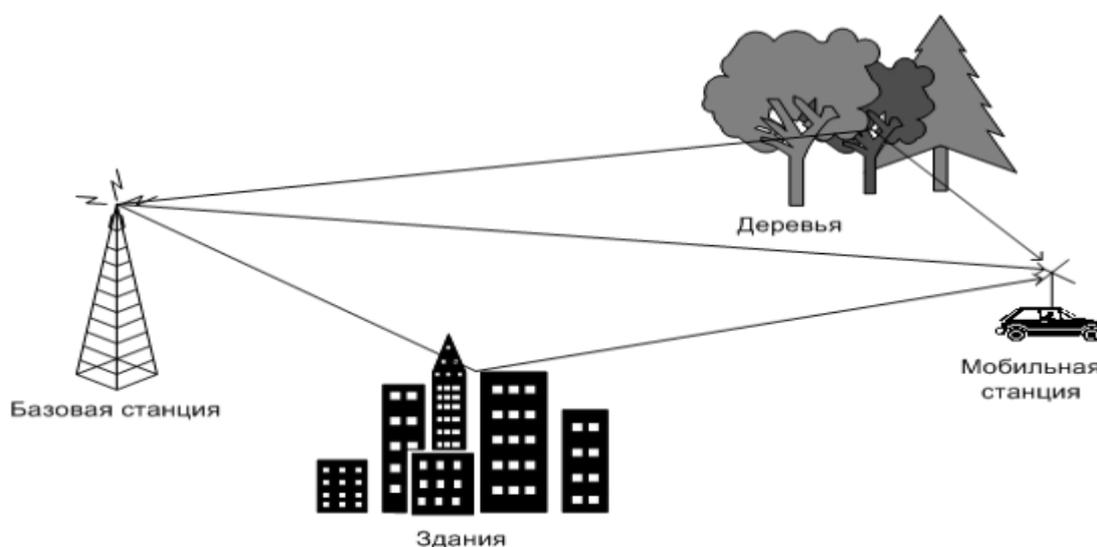


Рис. 4.20. Многолучевой характер распространения сигнала в мобильном канале связи.

В мобильном канале связи сигнал подвергается воздействию следующих основных типов искажений:

- Многолучевое распространение возникает в результате отражения, рассеяния и дифракции электромагнитных волн при взаимодействии с различными объектами в

пространстве. Таким образом, сигнал в приемной антенне содержит сумму волн с различными задержками, амплитудами и фазами. Суперпозиция этих волн приводит к

изменению амплитуды и фазы принимаемого сигнала.

- Допплеровский сдвиг возникает из-за перемещения абонента в пространстве. Возникает изменение амплитуды и фазы принимаемого сигнала во времени. Даже небольшие перемещения на расстояния, соизмеримые с длиной волны передаваемого сигнала, могут вызывать существенные изменения параметров принимаемого сигнала. Изменения параметров сигнала во времени, вызванные движением абонента и многолучевым распространением радиоволн называются быстрыми замираниями. Величина доплеровского сдвига пропорциональна частоте передачи и скорости движения. Чем меньше разнесение между несущими в сигнале OFDM, тем более восприимчива система к доплеровскому сдвигу частоты. Различные режимы передачи позволяют получить компромисс между уровнем восприимчивости к межсимвольной интерференции и доплеровскому сдвигу частоты.

- Затенение вызывается объектами, такими как здания, холмы, деревья и т.п., оказывающимися на пути сигнала и ограничивающими прямую видимость между передатчиком и приемником. Изменения параметров сигнала во времени, вызванные затенением обычно относят к медленным замираниям.

- Потери в тракте характеризуются как зависимость падения средней мощности сигнала от расстояния между передатчиком и приемником. В открытом пространстве средняя мощность уменьшается пропорционально квадрату расстояния между передатчиком и приемником. В мобильном радиоканале, где прямая видимость часто отсутствует, эта характеристика часто представляется как функция степени от 3 до 5.

Искажения, вызванные потерями в тракте и затенением (медленные замирания) обычно компенсируются с помощью систем управления мощностью. Борьба с быстрыми замираниями, вызванными движением и многолучевым распространением волн, является более сложной задачей и требует применения сложной обработки сигнала как на приемной, так и на передающей стороне.

Следствием многолучевого распространения радиоволн является искажение формы принимаемого сигнала. Многолучевая интерференция присуща любому типу сигналов, но особенно негативно она сказывается на широкополосных сигналах. Дело в том, что при использовании широкополосного сигнала в результате интерференции определенные частоты складываются синфазно, что приводит к увеличению уровня сигнала, а некоторые, наоборот, противофазно, вызывая ослабление сигнала на данной частоте.

Говоря о многолучевой интерференции, возникающей при передаче сигналов, различают два крайних случая. В первом из них максимальная задержка между различными сигналами не превышает времени длительности одного символа и интерференция возникает в пределах одного передаваемого символа. Во втором случае максимальная задержка между различными сигналами больше длительности одного символа, в результате интерференции складываются сигналы, представляющие разные символы, и возникает так называемая межсимвольная интерференция (InterSymbol Interference, ISI).

Межсимвольная интерференция наиболее существенно влияет на искажение сигнала. Поскольку символ - это дискретное состояние сигнала, характеризующееся значениями частоты несущей, амплитуды и фазы, то для различных символов меняются амплитуда и фаза сигнала, а значит, восстановить исходный сигнал крайне сложно. Чтобы хотя бы частично компенсировать эффект многолучевого распространения, используются частотные эквалайзеры, однако по мере роста скорости передачи данных либо вследствие увеличения символьной скорости, либо из-за усложнения схемы кодирования эффективность использования эквалайзеров падает.

В традиционных системах с одной несущей борьба с межсимвольной интерференцией обычно ведется путем адаптивного выравнивания. Этот процесс использует адаптивную фильтрацию для аппроксимации импульсного отклика канала. Затем инверсный фильтр используется для воссоздания копий искаженных символов. Этот процесс, однако, довольно сложный, ввиду высокой сложности адаптивного эквалайзера. В случаях, когда межсимвольная интерференция становится довольно высокой, процесс также теряет эффективность.

### **Модель системы связи с ортогональным частотным мультиплексированием (OFDM)**

При высоких скоростях передачи применяется метод передачи данных, который состоит в том, что поток передаваемых данных распределяется по множеству частотных подканалов и передача ведется параллельно на всех этих подканалах. При этом высокая скорость передачи достигается именно за счет одновременной передачи данных по всем каналам, а скорость передачи в отдельном подканале вполне может быть невысокой. Поскольку в каждом из частотных подканалов скорость передачи можно сделать не слишком высокой, это создает предпосылки для эффективного подавления межсимвольной интерференции.

При частотном разделении каналов необходимо, чтобы ширина отдельного канала была, с одной стороны, достаточно узкой для минимизации искажения сигнала в пределах

отдельного канала, а с другой - достаточно широкой для обеспечения требуемой скорости передачи. Кроме того, для экономного использования всей полосы канала, разделяемого на подканалы, желательно как можно более плотно расположить частотные подканалы, но при этом избежать межканальной интерференции, чтобы обеспечить полную независимость каналов друг от друга. Частотные каналы, удовлетворяющие перечисленным требованиям, называются ортогональными. Несущие сигналы всех частотных подканалов (а точнее, функции, описывающие эти сигналы) ортогональны друг другу. Важно, что хотя сами частотные подканалы могут частично перекрывать друг друга, ортогональность несущих сигналов гарантирует независимость каналов друг от друга, а следовательно, и отсутствие межканальной интерференции.

Рассмотренный способ деления широкополосного канала на ортогональные частотные подканалы называется ортогональным частотным мультиплексированием (OFDM). Сигнал в системе с ортогональным частотным мультиплексированием имеет разбиение на множество несущих, что обеспечивает небольшое количество символов на одну несущую и снижает межсимвольную интерференцию. На рисунке 4.21 представлен спектр сигнала при модуляции OFDM.

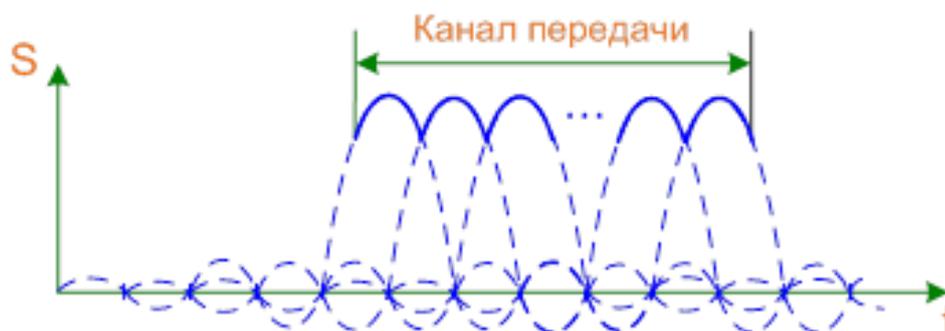


Рис. 4.21. Спектр сигнала при модуляции OFDM

Дополнительно применяется защитный интервал - циклический префикс, добавляемый в начало каждого символа. Для эффективной работы системы, использующей такой подход, максимальная задержка в канале не должна превышать длину циклического префикса. Высокая эффективность систем OFDM при работе в каналах с многократными отражениями делает их пригодными для высокоскоростных систем передачи данных в наземных системах связи.

На рисунке 4.22 представлена структура OFDM символа.

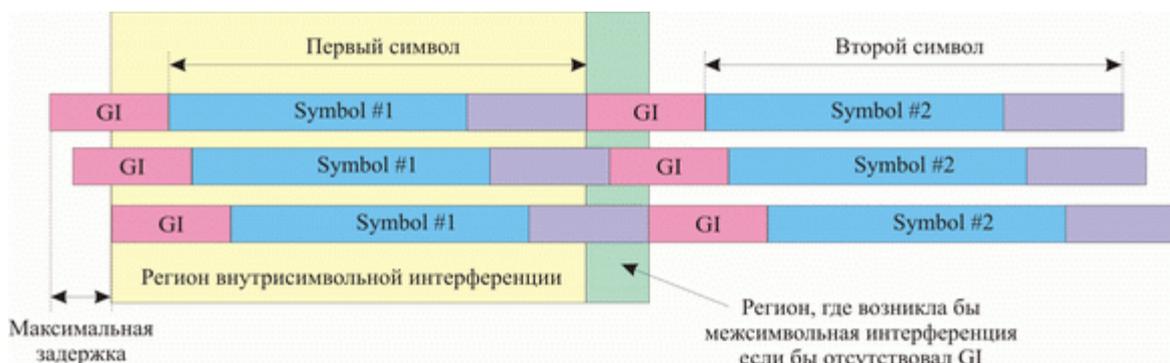


Рис. 4.22. Структура OFDM символа.

На рисунке 4.24 показана обобщенная структура системы связи с ортогональным частотным мультиплексированием. Данные пользователя вначале поступают в блок помехоустойчивого кодирования, а затем на модулятор. После модуляции данные в виде комплексных символов поступают в блок Обратного быстрого преобразования Фурье (ОБПФ), где происходит формирование сигнала OFDM с использованием  $M$  поднесущих. На рисунке 4.23 представлена временная форма сигнала после прохождения ОБПФ.

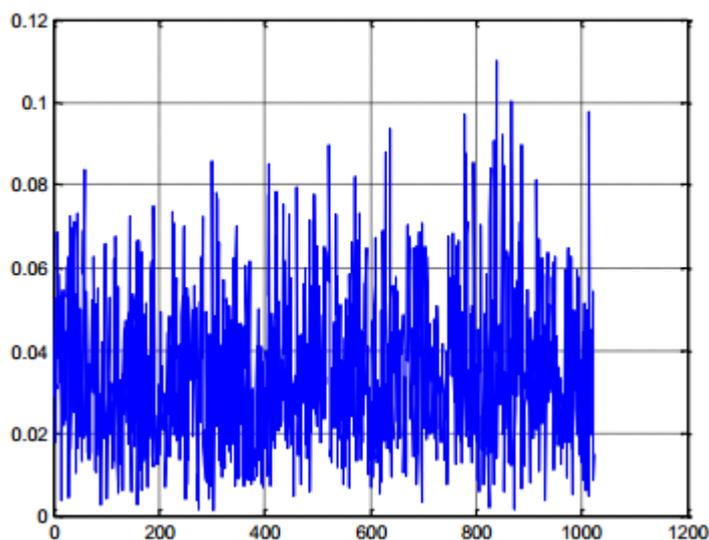
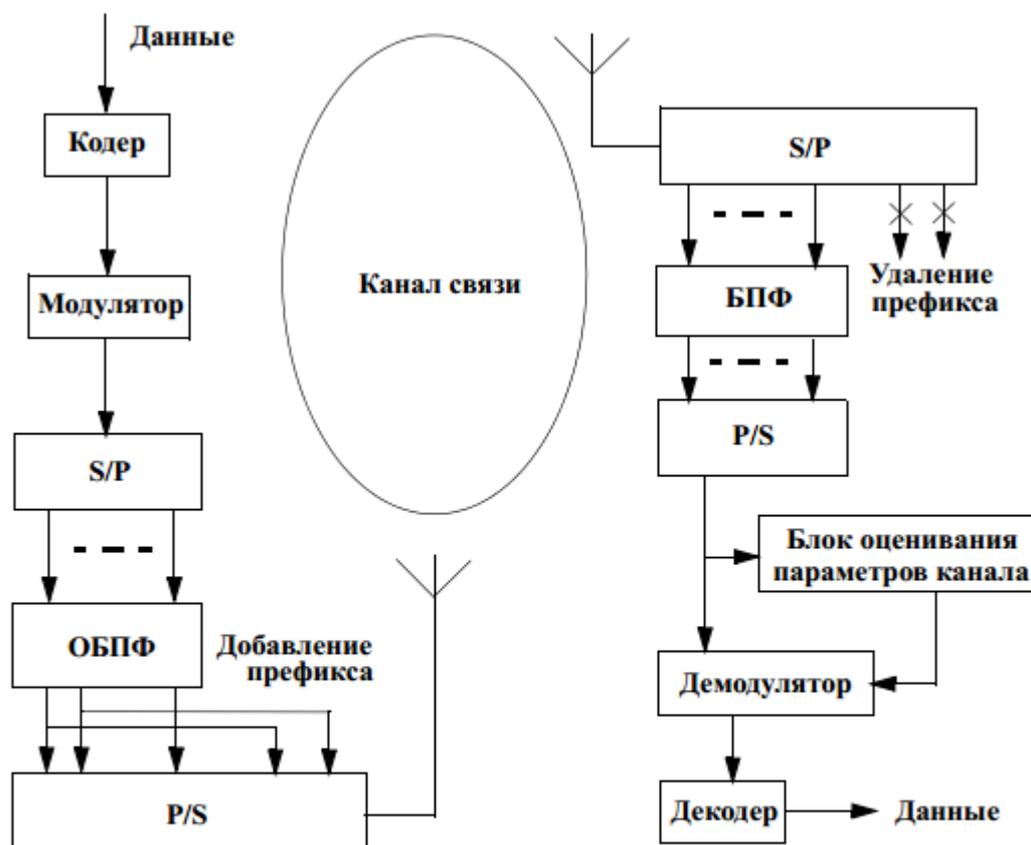


Рис. 4.23. Временная форма сигнала

Символ OFDM состоит из основной информационной части и циклического префикса, который формируется путем копирования последних  $L$  отсчетов в начало кадра. Далее последовательность символов OFDM преобразуется в аналоговый сигнал и передается по каналу связи. Длительность циклического префикса выбирается таким образом, чтобы быть больше, чем длительность импульсного отклика канала связи. Таким образом, символ OFDM имеет длительность  $(M+L)T_0$ , где  $T_0$  - период дискретизации в системе.



ОБПФ - блок обратного быстрого преобразования Фурье  
 БПФ - блок быстрого преобразования Фурье  
 S/P - последовательно-параллельное преобразование потока символов  
 P/S - параллельно-последовательное преобразование потока символов

Рис. 4.24. Структурная схема системы связи с ортогональным частотным мультиплексированием

На приемной стороне, сигнал после дискретизации и удаления префикса поступает в блок Быстрого преобразования Фурье (БПФ), где осуществляется демодуляция сигнала OFDM. После преобразования параллельного потока символов в последовательный, символы поступают в демодулятор, а также в блок оценивания параметров канала. Оценки параметров канала необходимы для работы демодулятора, поэтому выход блока оценивания параметров канала соединен с демодулятором. Дальнейшие преобразования сигнала зависят от структуры конкретной системы и не относятся к основной концепции OFDM. В реальных условиях работы системам беспроводной подвижной связи с FDM приходится работать в условиях, когда в канале связи сигнал подвергается воздействию замираний. Поэтому для получения приемлемой помехоустойчивости требуется использование помехоустойчивого кодирования.

## Модель связи с использованием технологии ММО [4]

ММО (Multiple Input, Multiple Output — множественный вход, множественный выход) предусматривает применение пространственного мультиплексирования с целью одновременной передачи нескольких информационных потоков по одному каналу, а также многолучевое отражение, которое обеспечивает доставку каждого бита информации соответствующему получателю с небольшой вероятностью влияния помех и потерь данных. Именно возможность одновременной передачи и приема данных определяет высокую пропускную способность устройств, работающих под технологией ММО.

Радиосигнал редко идет по кратчайшему пути между передатчиком и приемником из-за стен, дверей и других препятствий. Большинство подобных окружений имеют много различных поверхностей, которые отражают радиосигнал (электромагнитную волну) подобно зеркалу, отражающему свет. После переотражения образуются множественные копии исходного сигнала. Когда множественные копии сигнала перемещаются различными путями от передатчика к приемнику сигнал, шедший кратчайшим путем, будет первым, а следующие копии (или переотраженное эхо сигнала) придут чуть позже из-за более длинных путей. Это называют многолучевым распространением сигнала (multipath). Условия множественного распространения постоянно меняются, т.к. устройства приёма и передачи часто перемещаются (смартфон с Wi-Fi в руках пользователя), движутся вокруг различные объекты, создавая помехи (люди, машины и т.п.). В случае прибытия сигналов в разное время и под разными углами это может вызывать искажения и возможное затухание сигнала.

Как уже отмечалось выше, для организации технологии ММО необходима установка нескольких антенн на передающей и на приемной стороне. Обычно устанавливается равное число антенн на входе и выходе системы, т.к. в этом случае достигается максимальная скорость передачи данных.

Для работы технологии ММО необходимы некоторые изменения в структуре передатчика по сравнению с обычными системами. В первую очередь, на передающей стороне необходим делитель потоков, который будет разделять данные, предназначенные для передачи на несколько низкоскоростных подпотоков, число которых зависит от числа антенн. Далее каждый их данных потоков должен быть передан через свою антенну. Обычно, антенны на передаче устанавливаются с некоторым пространственным разнесением, чтобы обеспечить как можно большее число побочных сигналов, которые возникают в результате переотражений. В одном из возможных способов организации

технологии MIMO сигнал передается от каждой антенны с различной поляризацией, что позволяет идентифицировать его при приеме. Однако в простейшем случае каждый из передаваемых сигналов оказывается промаркированным самой средой передачи (задержкой во времени, затуханием и другими искажениями). На приемной стороне несколько антенн принимают сигнал из радиоэфира. Причем антенны на приемной стороне также устанавливаются с некоторым пространственным разнесением, за счет чего обеспечивается разнесенный прием. Принятые сигналы поступают на приемники, число которых соответствует числу антенн и трактов передачи. Причем на каждый из приемников поступают сигналы от всех антенн системы. Каждый из таких сумматоров выделяет из общего потока энергию сигнала только того тракта, за который он отвечает. Делает он это либо по какому-либо заранее предусмотренному признаку, которым был снабжен каждый из сигналов, либо благодаря анализу задержки, затухания, сдвига фазы, т.е. набору искажений или «отпечатку» среды распространения. На рисунке 4.22. представлен принцип организации технологии MIMO.

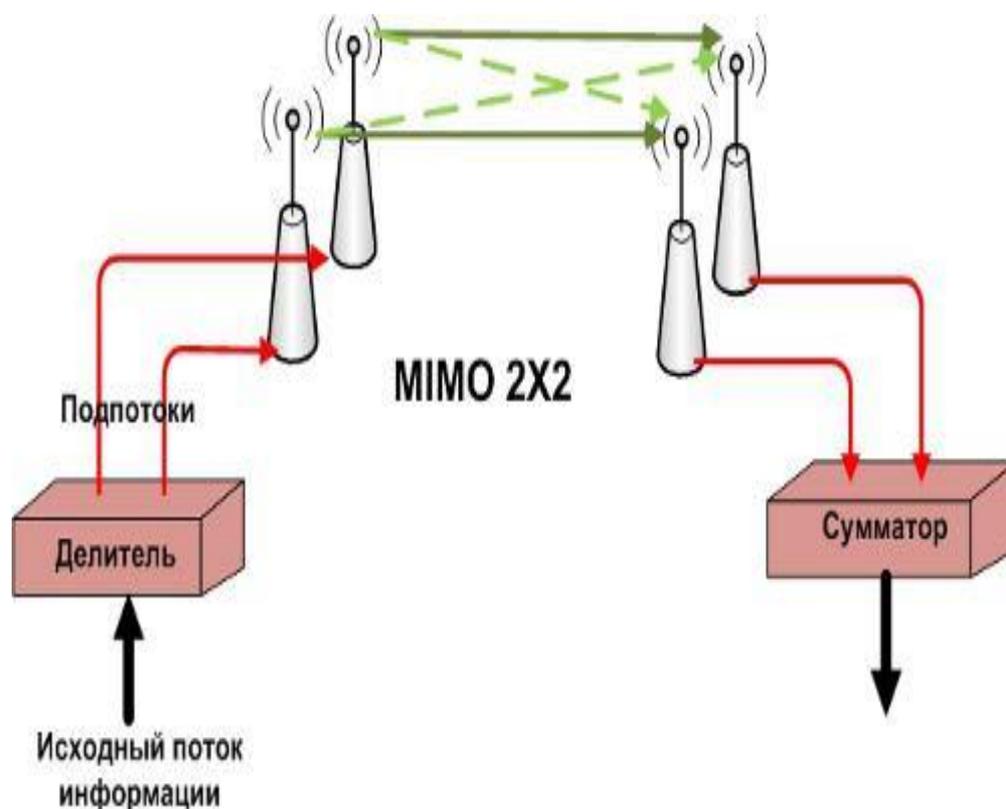


Рис. 4.25. Принцип организации технологии MIMO

Структурная схема программного комплекса

Структурная схема программного комплекса представлена на рисунке 4.26.

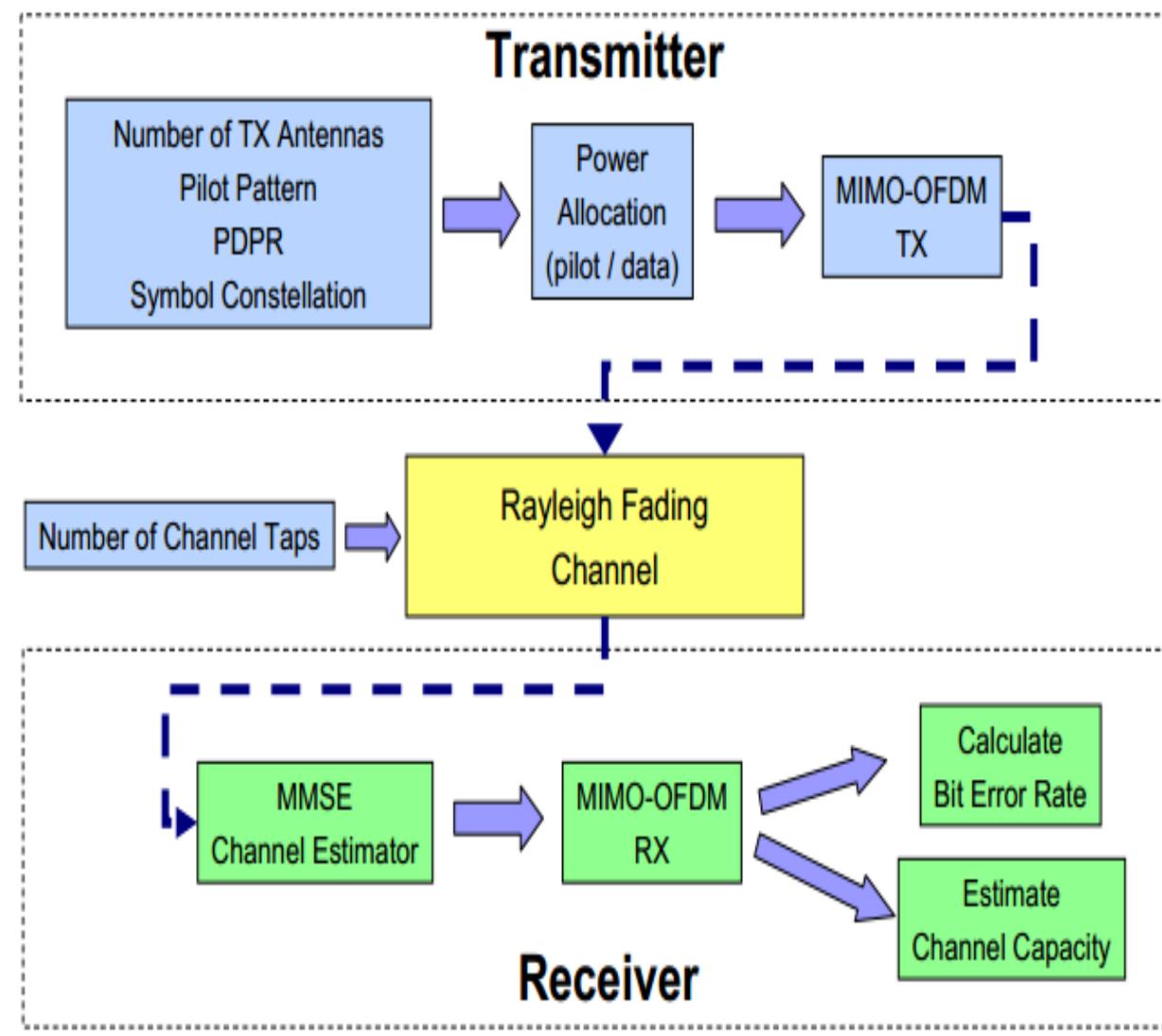


Рис. 4.26. Структурная схема передатчика MIMO-OFDM

В блоках передатчика устанавливаются необходимые для передачи параметры, после чего по каналу Релеевского затухания идет передача на приёмник. В нём происходит приём и последующее оценивание результатов моделирования.

### Описание учебного аппаратно-программного комплекса

С помощью ПО LabVIEW 2012, включающим в себя набор Modulation Toolkit от компании National Instruments было разработано и протестировано приложение для исследования MIMO-OFDM [17].

Передняя панель программного комплекса представлена на рисунке 4.24.

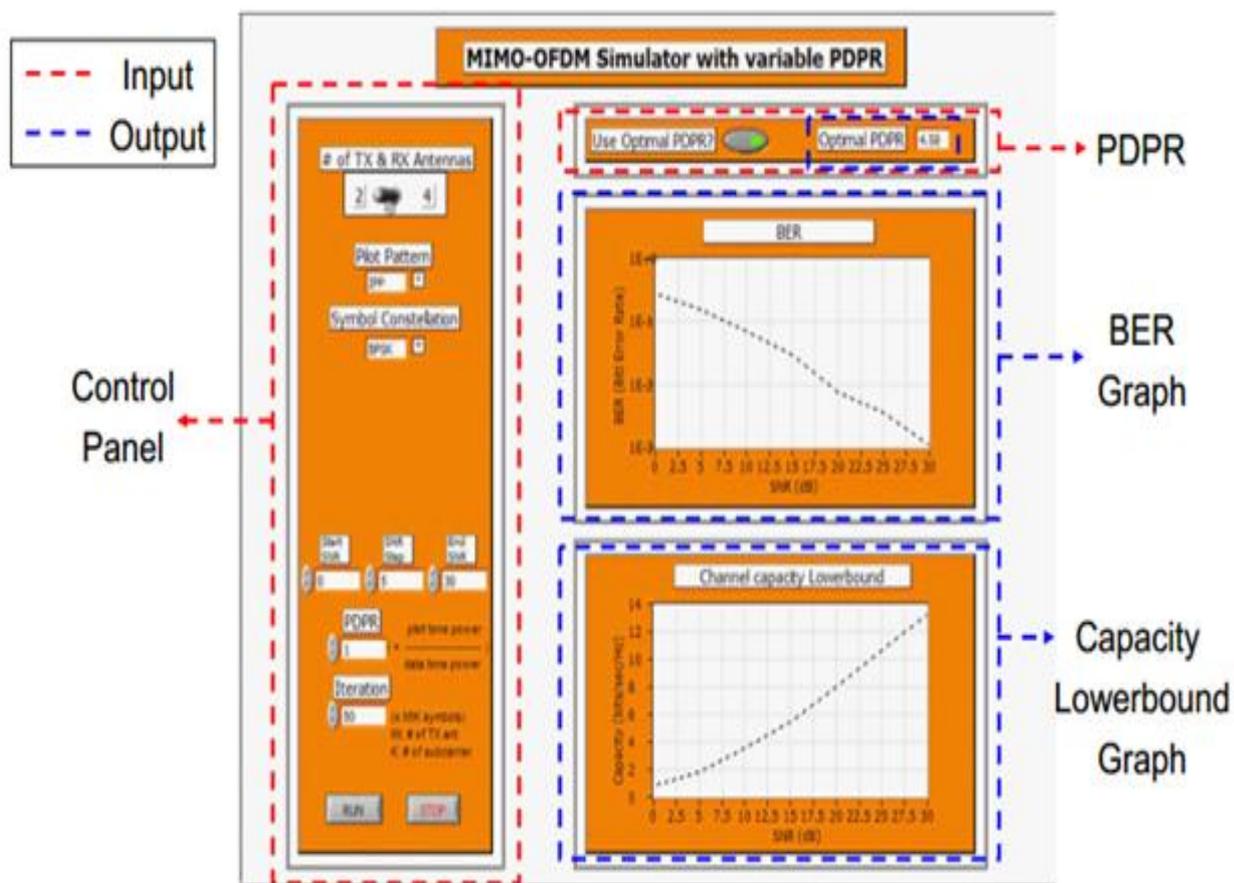


Рис. 4.27. Передняя панель программного комплекса

Панель комплекса имеет простой и удобный вид. Красной пунктирной линией отмечены элементы, задействованные при передаче, синей – элементы, представляющие выходные данные. Ниже каждый элемент будет рассмотрен подробнее.

На рисунке 4.28 представлена главная панель управления комплексом.

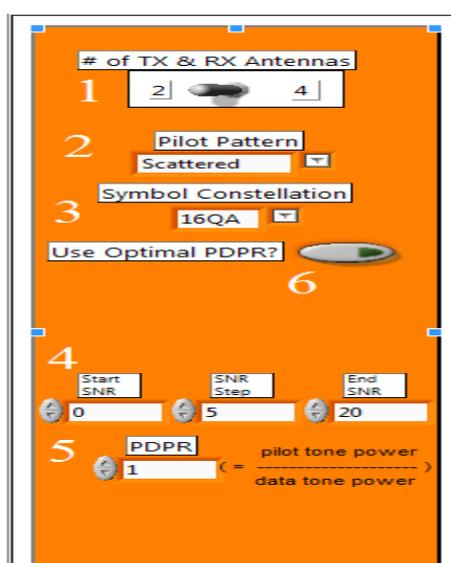


Рис. 4.28. Главная панель управления комплексом

На данной панели представлены следующие варианты настройки:

1. Число приёмно-передающих антенн. В данном комплексе число передающих и приёмных антенн одинаково.
2. Варианты вставки пилот-сигналов в OFDM символ. Возможен выбор вариантов Independent, Scattered, Orthogonal. Представленный ниже рисунок поясняет различие между этими вариантами.

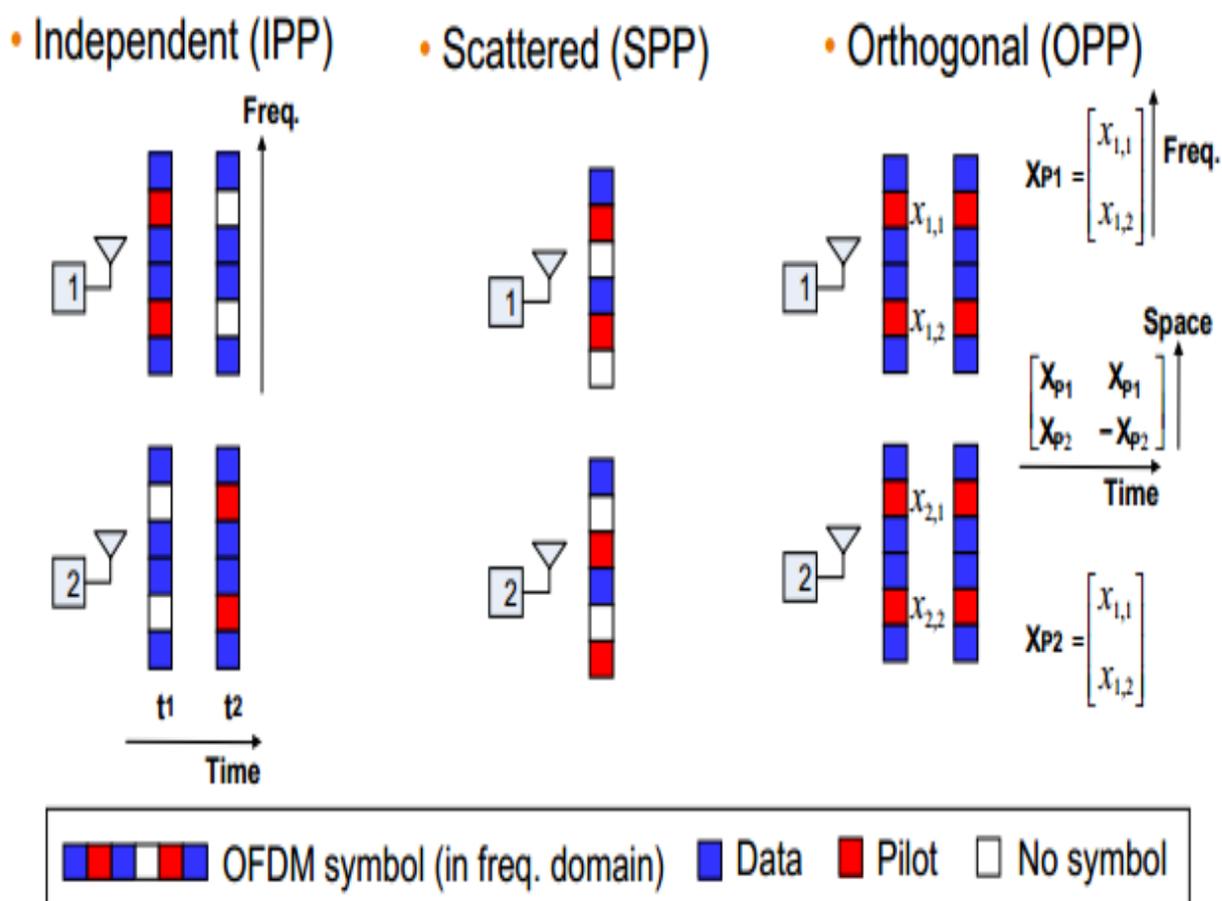


Рис. 4.29. Варианты расположения пилот-сигналов в OFDM символе

3. Выбор варианта модуляции: BPSK, QPSK, 16QAM, 64QAM.
4. Определяет отношение сигнал/шум в процессе работы схемы.
5. PDPR (Pilot to data power ratio) – отношение величины мощности пилотных символов к мощности символов данных.
6. Использовать оптимальное соотношение PDPR. Если данная функция будет включена, то программа автоматически подберет оптимальное соотношение PDPR.

На рисунке 4.30 представлено окно результатов и оценки.

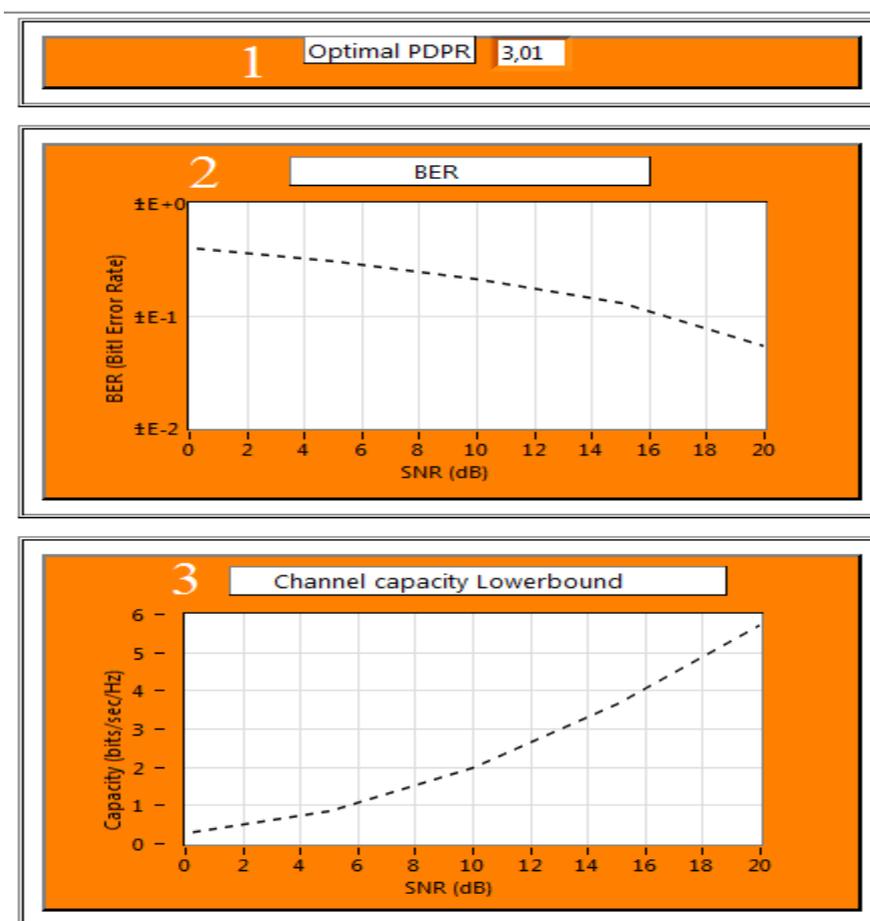


Рисунок 4.30. Окно результатов и оценки.

1. Если функция оптимального PDPR включена, то в данном окне будет указана величина PDPR, выбранная автоматически.
2. График зависимости BER от SNR.
3. График зависимости пропускной способности канала от SNR.

С помощью данного окна видны легко можно получить информацию о влиянии SNR, количества антенн и др. на качество передачи.

### **Проведение исследования с помощью разработанного программного комплекса**

На данном программном комплексе было проведено исследование влияния вида модуляции на качество передачи.

Исходные данные:

Число антенн: 2.

Варианты вставки пилот-сигналов в OFDM символ: Scattered.

Модуляция: QPSK, 16QAM, 64QAM.

Включена функция оптимального PDPR.

Отношение сигнал/шум: от 0 до 20.

В результате были получены следующие графики.

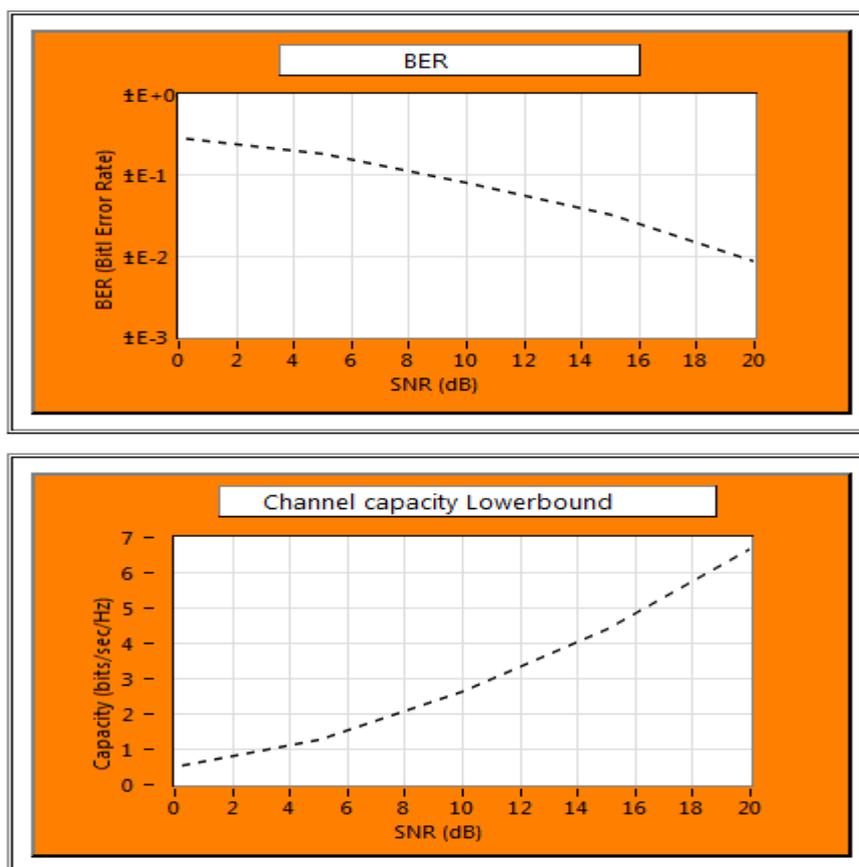


Рис. 4.28. Результаты работы при модуляции QPSK

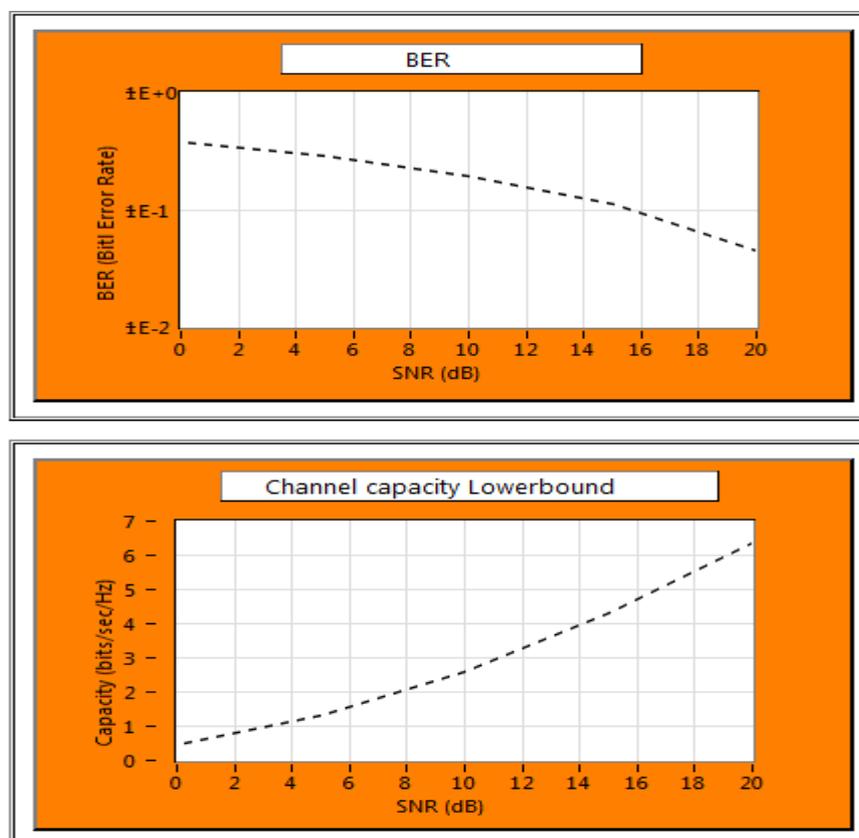


Рис. 4.31. Результаты работы при модуляции 16QAM

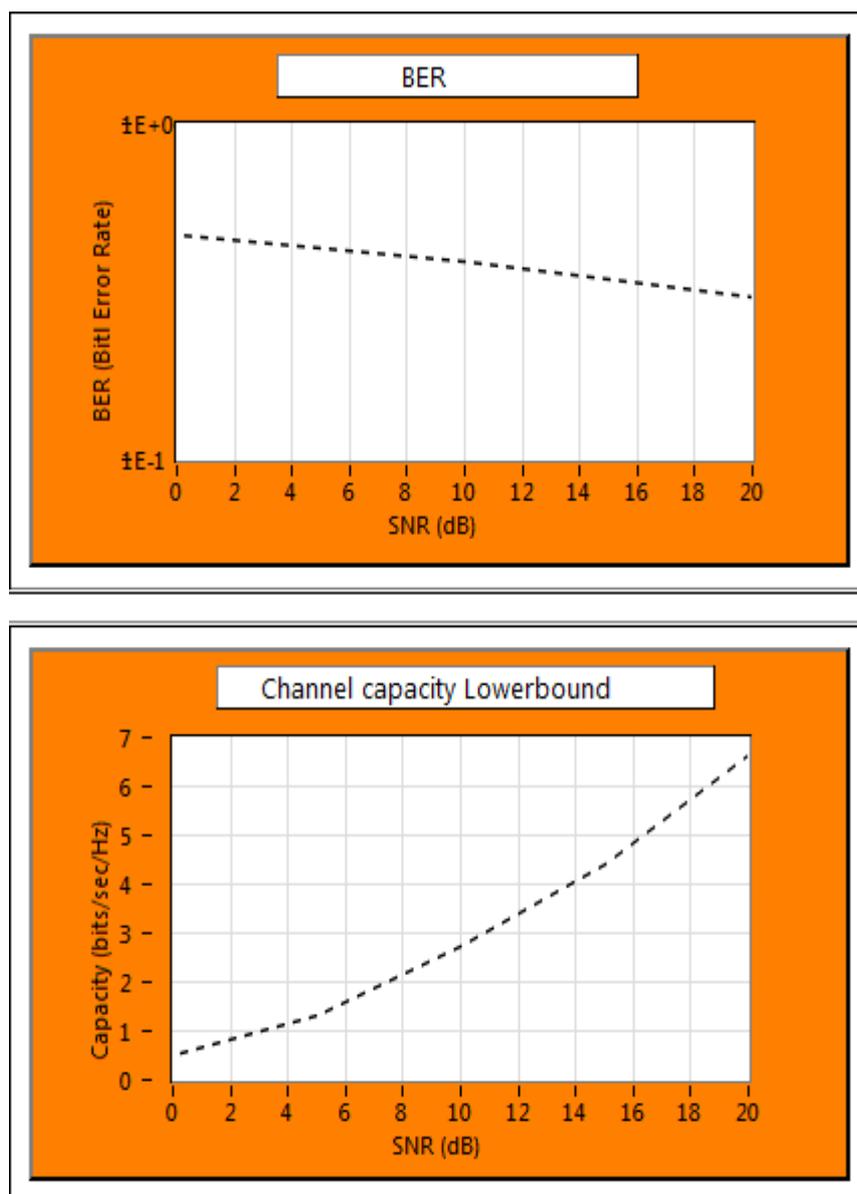


Рис. 4.32. Результаты работы при модуляции 64QAM

По данным графикам можно отметить, что при модуляции 64QAM BER изменяется медленно и имеет значение, близкое к единице даже при отношении сигнал/шум, равном 20. Когда при других видах модуляции BER уменьшается резко и имеет значение, близкое к 0.01.

Пропускная ёмкость канала имеет практически одинаковые значения.

## **Методика проведения исследования с помощью разработанного программного комплекса**

Задание на лабораторную работу:

1. Изучить работу приложения «MIMO-OFDM»;
2. Установить параметры макета, выставив значения: Количество антенн:2, Pilot Pattern: Scattered, Symbol Constellation: BPSK, функцию Optimal PDPR включить, значение SNR выставить от 0 до 20 с шагом 5.
3. Меняя виды модуляции получить различные графики зависимости отношения BER от SNR и пропускной способности канала от SNR, проанализировать полученные результаты.
4. Увеличить число антенн до четырёх и повторить п.3
5. По полученным данным сделать соответствующие выводы.

В данном разделе разработан учебный аппаратно-программный комплекс для визуализации и исследования технологии MIMO-OFDM.

Была изучена методика написания приложений в ПО LabVIEW, изучены и проанализированы методы пространственно-временного кодирования.

Разработанное приложение может послужить комплексом для проведения лабораторных работ на тему MIMO-OFDM по различным дисциплинам. В том числе данное приложение можно использовать и в качестве тренажера для написания программных решений в LabVIEW.

### **Методика проведения исследований с помощью схемы MIMO-OFDM**

С помощью разработанного комплекса можно провести сравнение использования 2-х и 4-х антенной системы MIMO-OFDM. Исходные данные к работе представлены ниже.

1. Число антенн: 2;
2. Pilot Pattern: Independent;
3. Symbol Constellation: BPSK;
4. Кнопка «Use Optimal PDPR» нажата.

Выставив подобные исходные данные на панели управления, необходимо запустить схему. На рисунке 4.31 представлен результат работы схемы.

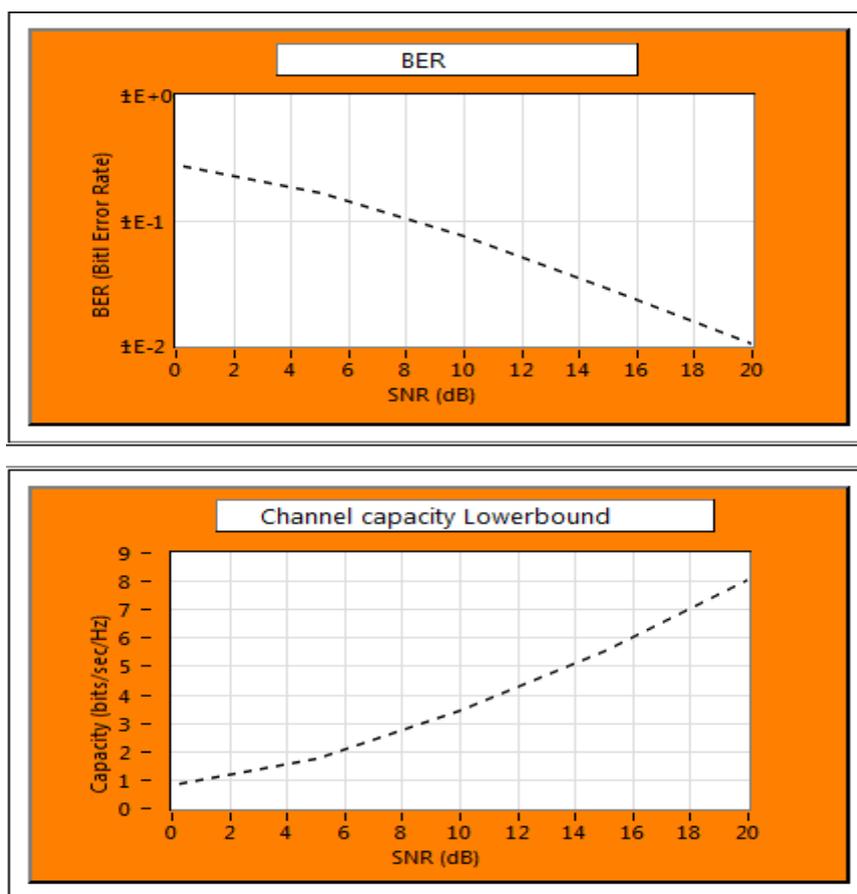


Рис. 4.33. Результат работы схемы при первом измерении

Далее необходимо поменять число антенн на 4 и повторить измерение. На рисунке 4.32 представлен результат работы схемы.

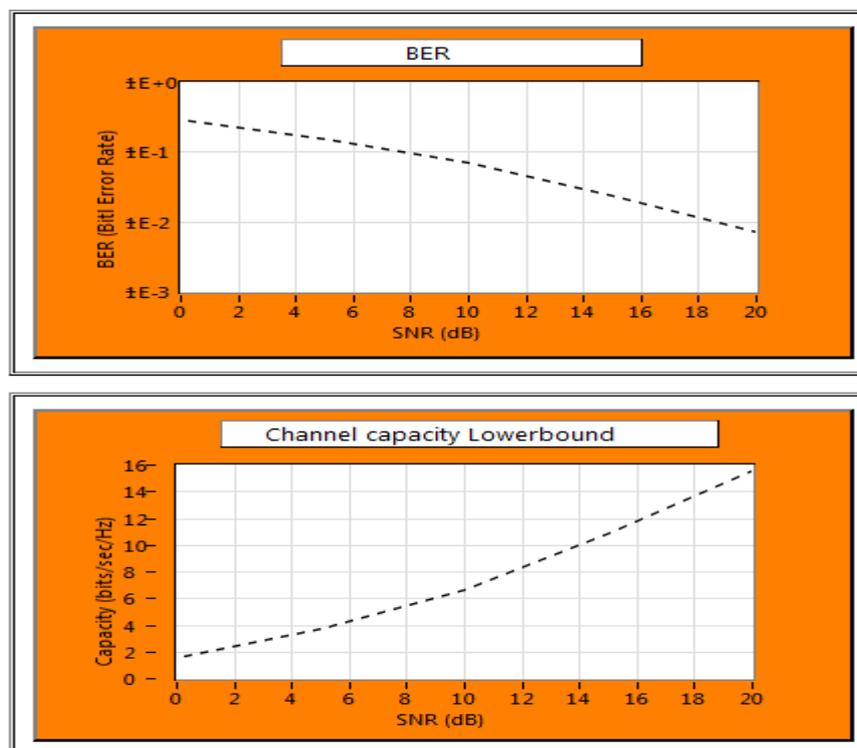


Рис. 4.34. Результат работы схемы при втором измерении

По полученным графикам можно сделать следующие выводы:

- Пропускная способность канала (а следовательно и максимальная скорость передачи) при использовании 4-х антенн увеличивается в два раза;
- При использовании 4-х антенн количество ошибок при увеличении отношения сигнал/шум уменьшается заметно быстрее и в итоге их число меньше, чем в ситуации использования 2-х антенн.

Далее будет исследована модуляция QPSK. Для этого необходимо выставить на панели управления модуляцию QPSK и переключить тумблер на 2 антенны. На рисунке 4.35 представлен результат работы схемы.

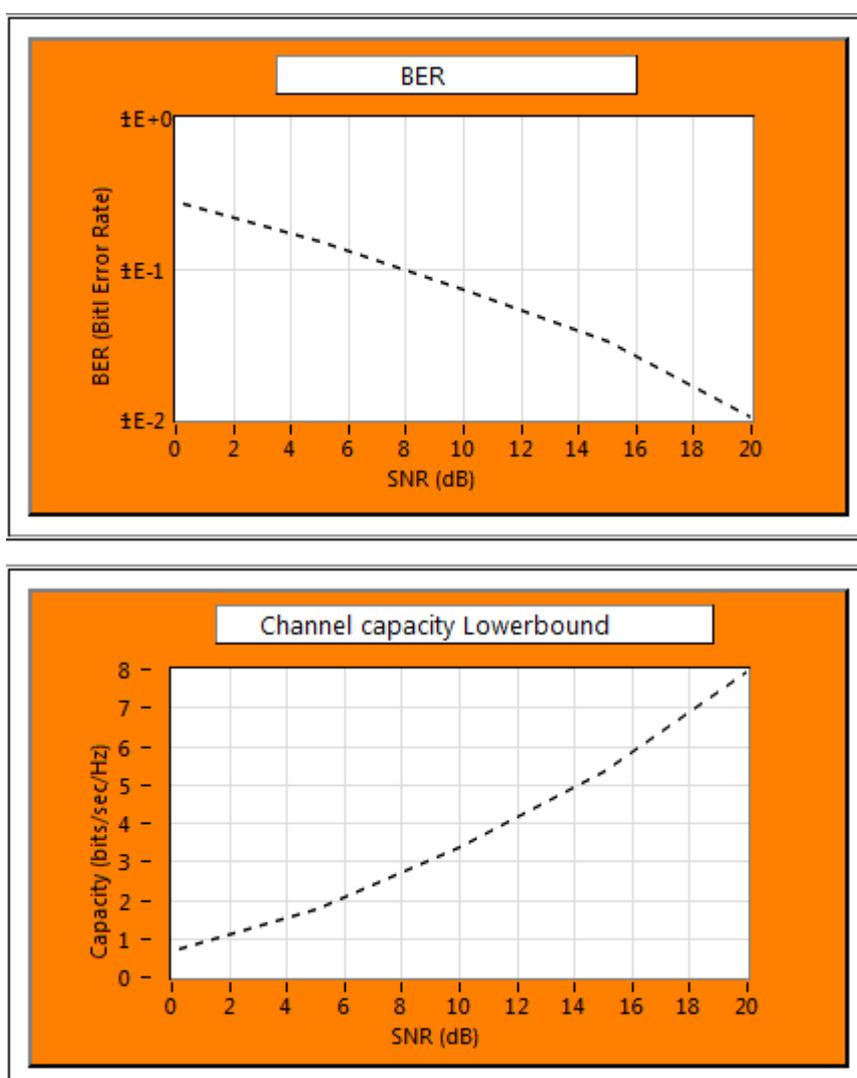


Рис. 4.35. Результат работы схемы при третьем измерении

Повторим измерение, используя 4 антенны. На рисунке 4.36 представлен результат работы схемы.

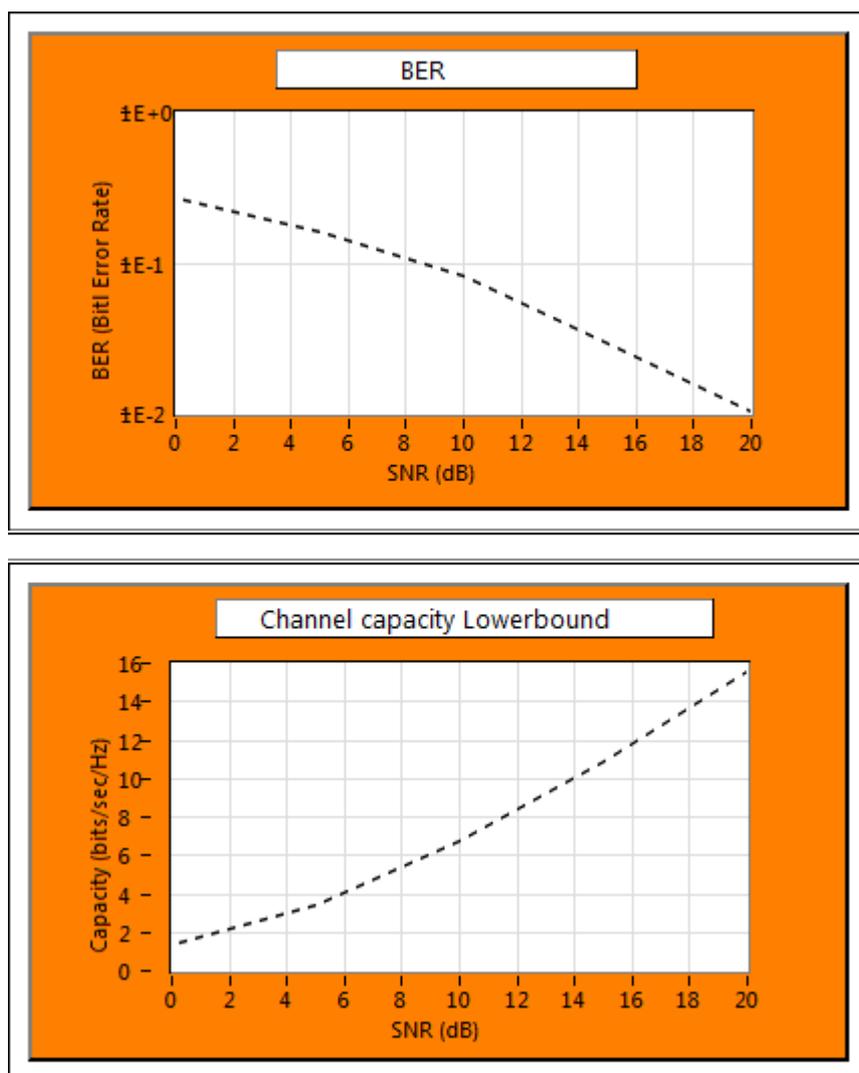


Рис.4.36. Результат работы схемы при четвертом измерении

Из полученных графиков можно сделать следующие выводы:

- Как и в предыдущем случае пропускная способность канала при использовании 4-х антенн увеличивается в два раза. Однако на этот раз скорость нарастания зависимости пропускной способности канала от SNR чуть выше;
- Крутизна функции зависимости BER от SNR различается. При использовании 4-х антенн функция падает быстрее.

Далее проведем исследование модуляции 16-QAM, выставив соответствующее значение на панели управления. На рисунке 4.37 представлен результат работы схемы.

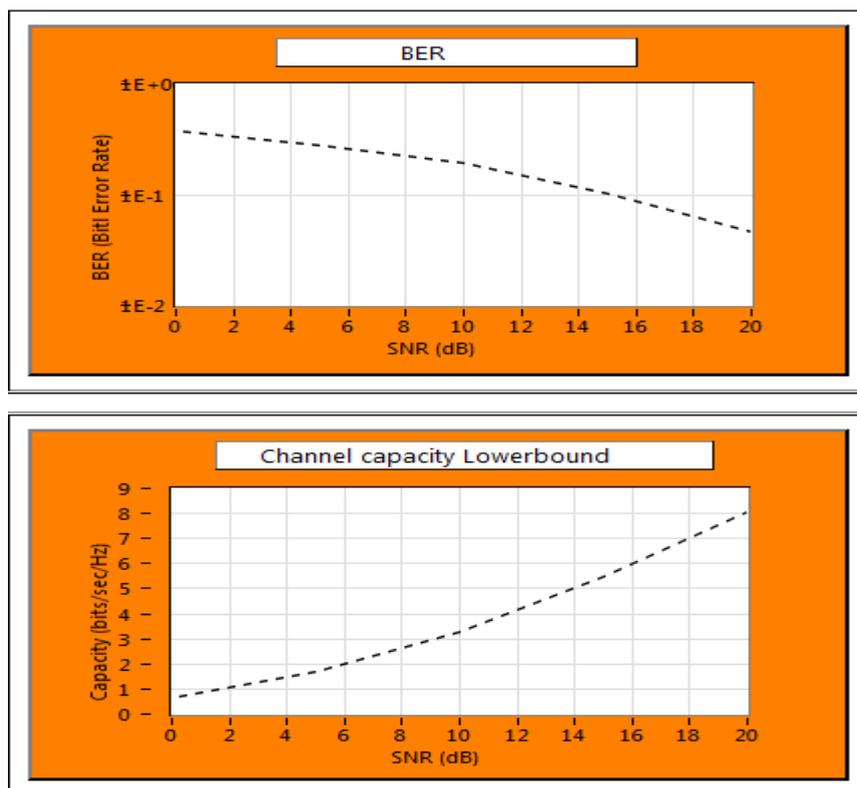


Рис. 4.37. Результат работы схемы при пятом измерении

Повторим измерение, используя 4 антенны. На рисунке 4.38 представлен результат работы схемы.

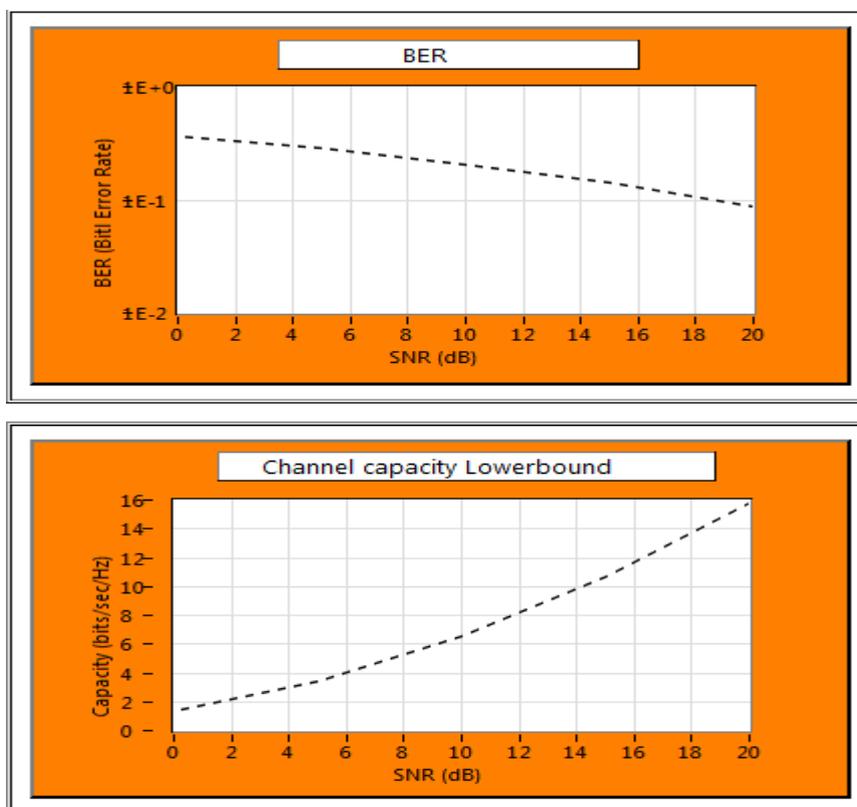


Рис. 4.38. Результат работы схемы при шестом измерении

Из полученных графиков можно сделать следующие выводы:

- Функция зависимости SNR от BER при использовании 2-х антенн выглядит заметно лучше, чем при использовании 4-х антенн;
- Разница пропускной способности канала составляет те же 2 раза, но на этот раз при использовании 2-х антенн функция быстрее стремится вверх.

Далее исследуем влияние PDPR на передачу. Для этого произведем измерения с активированной кнопкой «Optimal PDPR» и с выставленным вручную значением PDPR.

На панели управления выставим модуляцию – BPSK, выберем число антенн – 2 и активируем кнопку «Optimal PDPR».

На рисунке 4.39 представлен результат работы схемы.

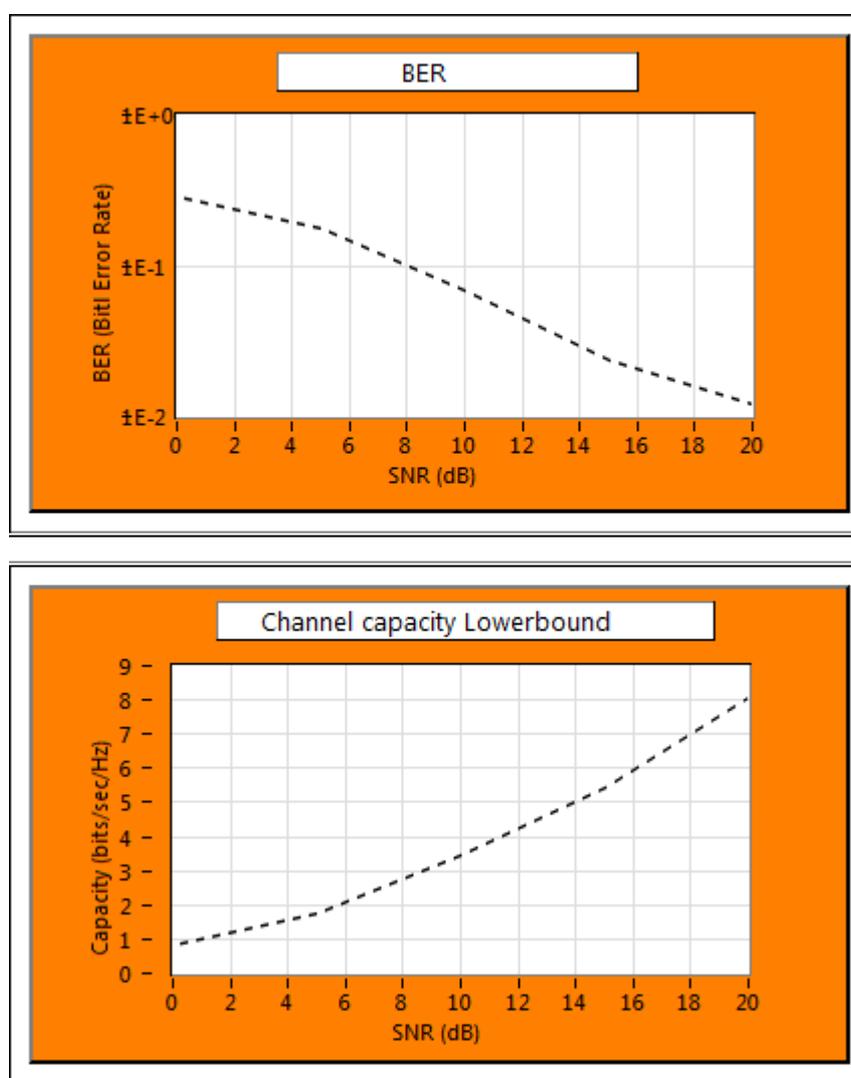


Рис. 4.39. Результат работы схемы при седьмом измерении

А теперь выставим значение PDPR вручную, равное = 0,005. Не забудем при этом выключить кнопку «Optimal PDPR».

На рисунке рис. 4.40 представлен результат работы схемы.

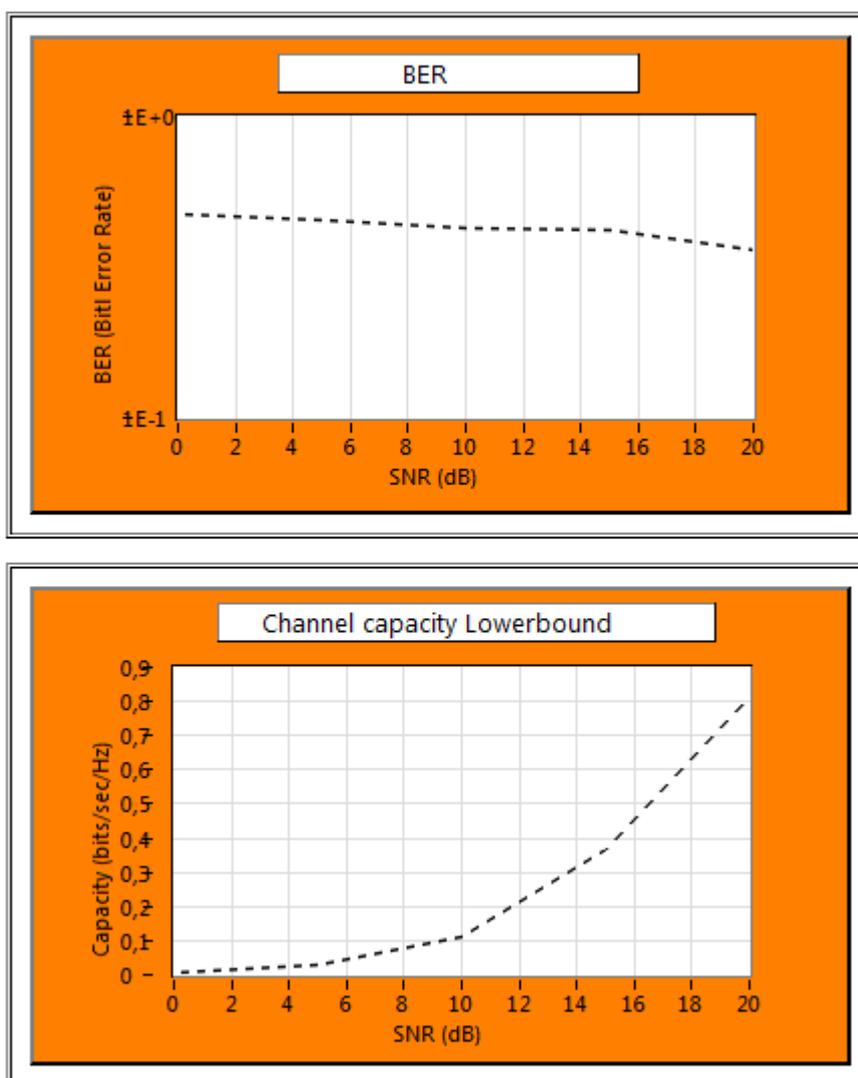


Рис. 4.40. Результат работы схемы при восьмом измерении

Из полученных графиков можно сделать следующие выводы:

- При оптимальном соотношении PDPR пропускная способность канала составляет около 8-ми бит/сек, тогда как при низком значении PDPR, равном 0,005 она составляет около 0,8 бит/сек. Т.е. уменьшение составляет в 10 раз;
- Зависимость BER от SNR при низком значении PDPR заметно хуже, нежели при оптимальном PDPR;
- При оптимальном значении PDPR функция зависимости SNR от BER круче падает вниз и число ошибок резко уменьшается, тогда как при низком значении PDPR ошибки практически не уменьшаются, а сама функция падает очень медленно.

В ходе данной работы были исследованы различные виды модуляций при различных режимах работы – 2 или 4 антенны. Были сделаны соответствующие выводы относительно каждого случая.

Наилучшие показатели были замечены при использовании модуляции QPSK в режиме работы 4-х антенн. Функция зависимости SNR от BER падает быстрее, а пропускная способность растёт быстрее. Кроме того, пропускная способность составляет около 16-ти бит/сек, что позволяет добиться очень высоких скоростей при передаче.

Необходимо относиться серьёзно к выбору отношения PDPR, т.к. неверно выбранное значение в десятки раз уменьшит пропускную способность, а, следовательно, и скорость передачи. Также, низкие показатели PDPR не позволят системе избавляться от ошибок даже при высоких значениях отношения сигнал/шум.

Данная работа позволяет проводить измерения схемы MIMO-OFDM, проводить лабораторные работы по дисциплинам, изучающим передачу и приём сигналов в современных системах передачи данных.

## **5. МОДЕМЫ И КОДЕКИ СОВРЕМЕННЫХ ТЕЛЕКОММУНИКАЦИОННЫХ СИСТЕМ**

### **5.1. Имитационное моделирование системы мобильной связи стандарта GSM**

#### **на базе ПО MATLAB**

GSM относится к сетям второго поколения (2 Generation) (1G — аналоговая сотовая связь, 2G — цифровая сотовая связь, 3G — широкополосная цифровая сотовая связь, коммутируемая многоцелевыми компьютерными сетями, в том числе Интернет).

Мобильные телефоны выпускаются с поддержкой 4 частот: 850 МГц, 900 МГц, 1800 МГц, 1900 МГц.

В зависимости от количества диапазонов, телефоны подразделяются на классы и вариацию частот в зависимости от региона использования.

- Однодиапазонные — телефон может работать в одной полосе частот. В настоящее время не выпускаются, но существует возможность ручного выбора определённого диапазона частот в некоторых моделях телефонов, например Motorola C115, или с помощью инженерного меню телефона.

- Двухдиапазонные (Dual Band) — для Европы, Азии, Африки, Австралии 900/1800 и 850/1900 для Америки и Канады.

- Трёхдиапазонные (Tri Band) — для Европы, Азии, Африки, Австралии 900/1800/1900 и 850/1800/1900 для Америки и Канады.

- Четырехдиапазонные (Quad Band) — поддерживают все диапазоны 850/900/1800/1900.

В стандарте GSM применяется GMSK-модуляция с величиной нормированной полосы  $BT = 0,3$ , где  $B$  — ширина полосы фильтра по уровню минус 3 дБ,  $T$  — длительность одного бита цифрового сообщения.

GSM на сегодняшний день является наиболее распространённым стандартом связи. По данным ассоциации GSM (GSMA) на данный стандарт приходится 82 % мирового рынка мобильной связи, 29 % населения земного шара использует глобальные технологии GSM. В GSMA в настоящее время входят операторы более чем 210 стран и территорий.

### **История развития**

GSM сначала означало *Groupe Spécial Mobile*, по названию группы анализа, которая создавала стандарт. Теперь он известен как *Global System for Mobile Communications* (Глобальная Система для Мобильной Связи), хотя слово «Связь» не включается в сокращение. Разработка GSM началась в 1982 году группой из 26 Европейских национальных телефонных компаний. *Европейская конференция почтовых и телекоммуникационных администраций* (CEPT), стремилась построить единую для всех европейских стран сотовую систему диапазона 900 МГц. Достижения GSM стали «одними из наиболее убедительных демонстраций какое сотрудничество в Европейской промышленности может быть достигнуто на глобальном рынке».

В 1989 году *Европейский Телекоммуникационный Институт Стандартов* (ETSI) взял ответственность за дальнейшее развитие GSM. В 1990 году были опубликованы первые рекомендации. Спецификация была опубликована в 1991 году.

Коммерческие сети GSM начали действовать в Европейских странах в середине 1991 г. GSM разработан позже, чем аналоговая сотовая связь и во многих отношениях была лучше спроектирована. Северо-Американский аналог — PCS, вырос из своих корней стандарты включая цифровые технологии TDMA и CDMA, но для CDMA потенциальное улучшение качества обслуживания так и не было никогда подтверждено.

### **GSM Phase 1**

1982 (Groupe Spécial Mobile) — 1990 г. Global System for Mobile Communications. Первая коммерческая сеть в январе 1992 г. Цифровой стандарт, поддерживает скорость передачи данных до 9,6 кбит/с. Полностью устарел, производство оборудования под него прекращено.

В 1991 году были введены услуги стандарта GSM «ФАЗА 1».

В них входят:

- Переадресация вызова (Call forwarding). Возможность перевода входящих звонков на другой телефонный номер в тех случаях, когда номер занят или абонент не отвечает; когда

телефон выключен или находится вне зоны действия сети и т. п. Кроме того, возможна переадресация факсов и данных.

- **Запрет вызова (Call barring).** Запрет на все входящие/исходящие звонки; запрет на исходящие международные звонки; запрет на входящие звонки, за исключением внутрисетевых.

- **Ожидание вызова (Call waiting).** Эта услуга позволяет принять входящий вызов во время уже продолжающегося разговора. При этом первый абонент или по-прежнему будет находиться на связи, или разговор с ним может быть завершён.

- **Удержание вызова (Call Holding).** Эта услуга позволяет, не разрывая связь с одним абонентом, позвонить (или ответить на входящий звонок) другому абоненту.

- **Глобальный роуминг (Global roaming).** При посещении любой из стран, с которой ваш оператор подписал соответствующее соглашение, вы можете пользоваться своим сотовым телефоном GSM без изменения номера.

### **GSM Phase 2**

Стандарт GSM Phase 2 принят в 1993 г.<sup>[3]</sup> Цифровой стандарт, поддерживает скорость передачи данных до 9,6 кбит/с. С 1995 г. включает диапазон 1900 МГц. Вторым этапом развития GSM — GSM «Фаза 2», который завершился в 1997 г., предусматривает такие услуги:

- **Определение номера вызывающей линии (Calling Line Identification Presentation).** При входящем звонке на экране высвечивается номер вызывающего абонента.

- **Антиопределитель номера (Calling Line Identification Restriction).** С помощью этой услуги можно запретить определение собственного номера при соединении с другим абонентом.

- **Групповой вызов (Multi party).** Режим телеконференции или конференц-связи позволяет объединить до пяти абонентов в группу и вести переговоры между всеми членами группы одновременно.

- **Создание закрытой группы до десяти абонентов (Closed User Group).** Позволяет создавать группу пользователей, члены которой могут связываться только между собой. Чаще всего к этой услуге прибегают компании, предоставляющие терминалы своим служащим для работы.

- **Информация о стоимости разговора.** Сюда входят таймер, который считает время на линии, и счётчик звонков. Также благодаря этой услуге можно проверять оставшийся на счёте кредит. Возможна и другая услуга: «Совет по оплате» (Advice of Charge). По требованию пользователя происходит проверка стоимости и длительности разговора в то время, когда аппарат находится на связи.

- Обслуживание дополнительной линии (Alternative Line Service). Пользователь может приобрести два номера, которые будут приписаны к одному модулю SIM. В этом случае связь выполняется по двум линиям, с предоставлением двух счетов, двух голосовых ящиков и т. п.

- Короткие текстовые сообщения (Short Message Service). Возможность приёма и передачи коротких текстовых сообщений (до 160 знаков).

- Система голосовых сообщений (Voice Mail). Услуга позволяет автоматически переводить входящие звонки на персональный автоответчик (голосовая почта). Пользоваться этим можно только в том случае, если у абонента активизирована услуга «переадресация вызовов».

Стандарт GSM Phase 2 считается устаревшим; но так как стандарт GSM подразумевает обратную совместимость, то старое оборудование базовых станций и телефоны могут работать (и работают) в современных сетях.

### **GSM Phase 2+**

Следующий этап развития сетей стандарта GSM «ФАЗА 2+» не связан с конкретным годом внедрения. Новые услуги и функции стандартизируются и внедряются после подготовки и утверждения их технических описаний. Все работы по этапу «Фаза 2+» проводились *Европейским институтом стандартизации электросвязи (ETSI)*. Количество уже внедрённых и находящихся в стадии утверждения услуг превышает 50. Среди них можно выделить следующие:

- улучшенное программное обеспечение SIM-карты;
- улучшенное полноскоростное кодирование речи EFR (Enhanced Full Rate);
- возможность взаимодействия между системами GSM и DECT;
- повышение скорости передачи данных благодаря пакетной передаче данных GPRS (General Packet RadioService) или за счёт системы передачи данных по коммутируемым каналам HSCSD (High Speed Circuit Switched Data).

### **Стандарты и радиointерфейс**

Стандарты GSM создаются и публикуются Европейским институтом телекоммуникационных стандартов. Документы обозначаются GSM nn.nn, например широко известен стандарт на GSM SIM-карточки GSM 11.11.

На сегодняшний день разработано множество различных стандартов сотовой связи. Существенная часть из них уже и морально, и физически устарела, часть не нашла распространения, а другие, напротив, распространились по всему миру и нашли сотни миллионов пользователей. Вот список самых распространенных стандартов:

\* AMPS

- \* DAMPS
- \* NMT-450
- \* GSM 900,1800,1900
- \* CDMA
- \*DECT

Наибольшее распространение, благодаря отличным функциональным возможностям (передача SMS, MMS, EMS, факсов, возможность доступа в интернет по GPRS, система GPS и т.д.), нашли полностью цифровые стандарты GSM и CDMA.

### **GSM-900**

Цифровой стандарт мобильной связи в диапазоне частот от 890 до 915 МГц (от телефона к базовой станции) и от 935 до 960 МГц (от базовой станции к телефону). Количество *реальных* каналов связи гораздо больше чем написано выше в таблице, т.к. присутствует еще и временное разделение каналов TDMA, т.е. на одной и той же частоте могут работать несколько абонентов с разделением во времени.

В некоторых странах диапазон частот GSM-900 был расширен до 880—915 МГц (MS -> BTS) и 925—960 МГц (MS <- BTS), благодаря чему максимальное количество каналов связи увеличилось на 50. Такая модификация была названа **E-GSM** (extended GSM).

### **GSM-1800**

Модификация стандарта GSM-900, цифровой стандарт мобильной связи в диапазоне частот от 1710 до 1880 МГц.

Особенности:

- Максимальная излучаемая мощность мобильных телефонов стандарта GSM-1800 — 1 Вт, для сравнения у GSM-900 — 2 Вт. Больше время непрерывной работы без подзарядки аккумулятора и снижение уровня радиоизлучения.
- Высокая ёмкость сети, что важно для крупных городов.
- Возможность использования телефонных аппаратов, работающих в стандартах GSM-900 и GSM-1800, одновременно. Такой аппарат функционирует в сети GSM-900, но, попадая в зону GSM-1800, переключается — вручную или автоматически. Это позволяет оператору рациональнее использовать частотный ресурс, а клиентам — экономить деньги за счёт низких тарифов. В обеих сетях абонент пользуется одним номером. Но использование аппарата в двух сетях возможно только в тех случаях, когда эти сети принадлежат одной компании, или между компаниями, работающими в разных диапазонах, заключено соглашение о роуминге.

Сеть GSM 900-1800 — это единая сеть, с общей структурой, логикой и мониторингом в которой телефон никуда не переключается. Вручную можно только запретить использовать один из диапазонов в тестовых или очень старых аппаратах.

Проблема состоит в том, что зона охвата для каждой базовой станции значительно меньше, чем в стандартах GSM-900, AMPS/DAMPS-800, NMT-450. Необходимо большее число базовых станций. Чем выше частота излучения, тем хуже проникающая способность радиоволн в городской застройке.

Дальность связи в GSM лимитирована задержкой сигнала Timing advance и составляет до 35 км. При использовании режима extended cell возрастает до 75 км. Практически достижимо только в море, пустыне и горах.

### **CDMA**

Тип стандарта: цифровой

Полоса частот: 1,23 МГц

Статус: Активно эксплуатируется

Краткое описание: Технология CDMA (система множественного доступа с кодовым разделением) изначально разработана для военных целей США, но, благодаря отличным показателям, нашла после модернизации широкое применение и в гражданской связи.

Особенности:

\* Сигнал каждого абонента модулируется псевдослучайным, уникальным кодом (шумоподобным сигналом, отправляемым клиенту в начале разговора). Несущая частота сигнала меняется, согласно этому случайному правилу, в результате чего узкополосный информационный сигнал каждого пользователя расширяется во всю ширину частотного спектра (1,23 МГц в случае CDMA). В приемнике сигнал демодулируется с помощью идентичного кода, в результате чего восстанавливается изначальный сигнал. Но в то же время сигналы остальных пользователей для данного приемника продолжают оставаться расширенными и воспринимаются им лишь как шум, незначительно мешающий нормальной работе приемника.

\* Отличные показатели шумоустойчивости, как следствие - снижение стоимости развертывания CDMA-сетей.

\* Высокое качество передачи речи при низких показателях излучаемой мощности.

\* Большая, по сравнению с GSM, емкость сети.

\* Высокое качество связи в зданиях.

### **NMT-450**

Тип стандарта: аналоговый

Частотный диапазон: 453-468 МГц

Статус: устарел и морально, и физически

Краткое описание: NMT-450 (Nordic Mobile Telephone) разработан скандинавскими учеными. Первые сотовые сети в России строились именно на базе этого стандарта - федеральная сеть "СОТЕЛ" работала именно на NMT.

Особенности:

\* Большая площадь покрытия одним ретранслятором, а значит, меньшие затраты на организацию сети.

\* Малое затухание сигнала на открытом пространстве, что для России с ее плотностью заселения - огромный плюс.

\* Сигнал ретранслятора может добивать на 100 километров!

\* Благодаря тому, что стандарт - аналоговый, обеспечивается более высокое качество передачи речи - отсутствует грубая дискретизация голосовых отсчетов.

\* Плохая помехоустойчивость из-за используемых частот. Уровень промышленных помех в этом диапазоне значительно выше, чем, скажем, на 800, 900 и 1800 МГц.

\* Отсутствие секретности разговоров - их можно слушать УКВ-приемником.

\* Низкая емкость сетей, что не позволяет массово использовать стандарт в крупных городах.

\* Список дополнительных услуг издевательски пуст.

\* NMT-трубки весят в несколько раз больше своих цифровых собратьев и крайне расточительны в плане электроэнергии и здоровья владельца.

## **AMPS**

Тип стандарта: аналоговый

Частотный диапазон: 825-890 МГц

Статус: устарел и морально, и физически

Краткое описание: В конце восьмидесятых американские специалисты разработали специально для своей страны стандарт AMPS (Advanced Mobile Phone Service - усовершенствованная мобильная телефонная система). Завоевав популярность в других странах, в 1993 стандарт пришел в Россию. Такие сети по сей день эксплуатируются в 55 регионах, часть из них работает в аналоговом стандарте AMPS, часть - в усовершенствованном цифровом D-AMPS.

Особенности:

\* Более высокая, чем у NMT-450, емкость сетей.

\* Низкий уровень промышленных и атмосферных помех благодаря используемому частотному диапазону.

\* Более надежная, чем у NMT-450, связь в помещениях.

\* Меньшая зона устойчивой связи для одной базовой станции, что вынуждает операторов ставить их ближе друг к другу - большие затраты.

\* Почти не распространен в Европе и Азии.

AMPS уже давным-давно морально устарел, и в 1990 г. в США был разработан D-AMPS.

### **D-AMPS**

Тип стандарта: цифровой

Частотный диапазон: 825-890 МГц

Статус: устарел морально

Краткое описание: Когда AMPS морально устарел - а это произошло довольно быстро, в 1990 году - в Штатах был разработан D-AMPS.

Особенности:

\* Емкость сетей на несколько порядков выше, чем у NMT-450 и AMPS.

\* Возможность эксплуатации мобильных аппаратов как в цифровом, так и в аналоговом режимах.

\* Расширенный спектр дополнительных услуг.

\* Емкость DAMPS-сетей ниже, чем в полностью цифровых системах, но выше, чем в аналоговых.

### **GPRS**

Главным недостатком стандарта GSM на сегодня является низкая скорость передачи данных - максимум 9,6 Кбит/с, да и сам процесс реализован довольно убого - под данные выделяется один голосовой канал; оплата услуги, соответственно, осуществляется исходя из времени соединения, причем по тарифам, весьма схожим с речевыми. Для решения этой проблемы и был разработан стандарт передачи данных GPRS (General Packet Radio Service - услуга пакетной передачи данных по радиоканалу).

Новая система предложила пользователям мобильной связи уже совсем другие условия - максимальная скорость соединения составляет 171,2 Кбит/с, а оплата осуществляется исходя из количества реально переданной информации, трафика.

В GSM-сетях, оборудованных GPRS-модулями, более рационально распределяется радиочастотный ресурс. Не вдаваясь в сложные технические детали, можно сказать, что выигрыш в скорости достигается за счет одновременного использования для передачи данных нескольких свободных в настоящий момент каналов. Тут следует отметить, что скорость передачи информации определяется не столько теоретическими возможностями сетевого и абонентского оборудования, сколько загрузкой сети - так, из собственного опыта могу сказать, что скорость соединения в России в ближайшие несколько лет у тебя не превысит 5-6 Кбит/с.

Благодаря тому, что пакеты данных имеют значительно меньший приоритет, по сравнению с голосовой информацией, внедрение систем GPRS не приводит к ухудшению качества услуг передачи речи.

Система GPRS состоит из двух основных модулей: SGSN (Serving GPRS Support Node - узел поддержки GPRS) и GGSN (Gateway GPRS Support Node - шлюзовой узел GPRS). В некотором смысле SGSN можно назвать аналогом коммутатора сети GSM. SGSN обеспечивает доставку пакетов информации пользователям, взаимодействует с реестром абонентов, проверяет, разрешены ли запрашиваемые услуги, ведет мониторинг пользователей, организует регистрацию вновь прибывших абонентов и т.п.

Назначение GGSN легко понять из расшифровки названия - это шлюз между сотовой сетью (вернее, SGSN) и внешними информационными сетями (интернетом, провайдерскими Intranet-сетями и т.д.).

Основной задачей GGSN, таким образом, является маршрутизация (обычно совмещенная с NAT'ом) пакетов, генерируемых абонентом через SGSN. Вторичными функциями GGSN являются: динамическая выдача IP-адресов (а-ля DHCP-сервер :)), отслеживание информации о внешних сетях, подсчет трафика, тарификация и т.д.

Благодаря хорошей масштабируемости системы GPRS, оператор может увеличивать число SGSN и GGSN по мере роста числа пользователей и их суммарного трафика.

Как известно, для работы с GPRS необходимо иметь специальный телефон, поддерживающий эту технологию.

Основная характеристика такого телефона - так называемый класс GPRS. Это максимальное количество каналов, которое может задействовать аппарат для передачи данных - напомним, что один канал обеспечивает передачу данных со скоростью до 13,4 Кбит/с.

Самым первым производителем телефонов с GPRS стала французская фирма Sagem - на проходящей в Женеве выставке Telecom'99 она представила телефон Sagem MC-850, имеющий 3 канала на прием и 1 на передачу данных.

Современные телефоны способны использовать десять и более каналов для передачи данных, что, теоретически, обеспечивает отличную скорость соединения - до 20 килобайт в секунду.

В стандарте GSM определены 4 диапазона работы (ещё есть пятый):

**900/1800 МГц (используется в Европе, Азии)**

Характеристики	GSM-900	GSM-1800
Частоты передачи MS и приёма BTS (uplink), МГц	890 — 915	1710 — 1785
Частоты приёма MS и передачи BTS (downlink), МГц	935 — 960	1805 — 1880
Дуплексный разнос частот приёма и передачи, МГц	45	95
Количество <b>частотных</b> каналов связи с шириной 1 канала связи в 200 кГц	124	374
Ширина полосы канала связи, кГц	200	200

**850/1900 МГц (используется в США, Канаде, отдельных странах Латинской Америки и Африки)**

Характеристики	GSM-850	GSM-1900
Частоты передачи MS и приёма BTS, МГц	824 — 849	1850 — 1910
Частоты приёма MS и передачи BTS, МГц	869 — 894	1930 — 1990
Дуплексный разнос частот приёма и передачи, МГц	45	80

**Структура GSM**

Система GSM состоит из трёх основных подсистем:

- подсистема базовых станций (BSS — Base Station Subsystem),
- подсистема коммутации (NSS — Network Switching Subsystem),
- центр технического обслуживания (OMC — Operation and Maintenance Centre).

В отдельный класс оборудования GSM выделены терминальные устройства — подвижные станции (MS — Mobile Station), также известные как мобильные (сотовые) телефоны.

### Подсистема базовых станций



Рис.5.1. Антенны трех базовых станций на мачте

BSS состоит из собственно базовых станций (BTS — Base Transceiver Station) и контроллеров базовых станций (BSC — Base Station Controller). Область, накрываемая сетью GSM, разбита на условные шестиугольники, называемые *сотами* или *ячейками*. Диаметр каждой шестиугольной ячейки может быть разным — от 400 м до 50 км. Максимальный теоретический радиус ячейки составляет 120 км, что обусловлено ограниченной возможностью системы синхронизации к компенсации времени задержки сигнала. Каждая ячейка покрывается находящейся в её центре одной базовой станцией, при этом ячейки частично перекрывают друг друга, тем самым сохраняется возможность передачи обслуживания без разрыва соединения при перемещении абонента из одной соты в другую. Естественно, что на самом деле сигнал от каждой станции распространяется, покрывая площадь в виде круга, а не шестиугольника, последний же является лишь упрощением представления зоны покрытия. Каждая базовая станция имеет шесть соседних в связи с тем, что в задачи планирования размещения станций входила минимизация стоимости системы. Меньшее количество соседних базовых станций приводило бы к большему перехлёсту зон покрытия с целью избегания "мёртвых зон", что в свою очередь потребовало бы более плотного расположения базовых станций. Большее количество соседних базовых станций приводило бы к излишним расходам на дополнительные станции, в то время как выигрыш от уменьшения зон перехлёста был бы уже весьма незначительным.

Базовая станция (BTS) обеспечивает приём/передачу сигнала между MS и контроллером базовых станций. BTS является автономной и строится по модульному принципу. Направленные антенны базовых станций могут располагаться на вышках, крышах зданий и т. д.

Контроллер базовых станций (BSC) контролирует соединения между BTS и подсистемой коммутации. В его полномочия также входит управление очередностью соединений, скоростью передачи данных, распределение радиоканалов, сбор статистики, контроль различных радиоизмерений, назначение и управление процедурой Handover.

### **Подсистема коммутации**

NSS состоит из нижеследующих компонентов.

Центр коммутации (MSC — Mobile Switching Center)

MSC контролирует определённую географическую зону с расположенными на ней BTS и BSC. Осуществляет установку соединения к абоненту и от него внутри сети GSM, обеспечивает интерфейс между GSM и ТфОП, другими сетями радиосвязи, сетями передачи данных. Также выполняет функции маршрутизации вызовов, управление вызовами, эстафетной передачи обслуживания при перемещении MS из одной ячейки в другую. После завершения вызова MSC обрабатывает данные по нему и передаёт их в центр расчётов для формирования счета за предоставленные услуги, собирает статистические данные. MSC также постоянно следит за положением MS, используя данные из HLR и VLR, что необходимо для быстрого нахождения и установления соединения с MS в случае её вызова.

Домашний регистр местоположения (HLR — Home Location Registry)

Содержит базу данных абонентов, приписанных к нему. Здесь содержится информация о предоставляемых данному абоненту услугах, информация о состоянии каждого абонента, необходимая в случае его вызова, а также Международный Идентификатор Мобильного Абонента (IMSI — International Mobile Subscriber Identity), который используется для аутентификации абонента (при помощи AUC). Каждый абонент приписан к одному HLR. К данным HLR имеют доступ все MSC и VLR в данной GSM-сети, а в случае межсетевого роуминга — и MSC других сетей.

Гостевой регистр местоположения (VLR — Visitor Location Registry)

VLR обеспечивает мониторинг передвижения MS из одной зоны в другую и содержит базу данных о перемещающихся абонентах, находящихся в данный момент в этой зоне, в том числе абонентах других систем GSM — так называемых роумерах. Данные об абоненте удаляются из VLR в том случае, если абонент переместился в другую зону. Такая схема позволяет сократить количество запросов на HLR данного абонента и, следовательно, время обслуживания вызова.

Регистр идентификации оборудования (EIR — Equipment Identification Registry)

Содержит базу данных, необходимую для установления подлинности MS по IMEI (International Mobile Equipment Identity). Формирует три списка: белый (допущен к использованию), серый (некоторые проблемы с идентификацией MS) и чёрный (MS,

запрещённые к применению). У российских операторов (и большей части операторов стран СНГ) используются только белые списки, что не позволяет раз и навсегда решить проблему кражи мобильных телефонов.

#### Центр аутентификации (AUC — Authentication Center)

Здесь производится аутентификация абонента, а точнее — SIM (Subscriber Identity Module). Доступ к сети разрешается только после прохождения SIM процедуры проверки подлинности, в процессе которой с AUC на MS приходит случайное число RAND, после чего на AUC и MS параллельно происходит шифрование числа RAND ключом Ki для данной SIM при помощи специального алгоритма. Затем с MS и AUC на MSC возвращаются «подписанные отклики» — SRES (Signed Response), являющиеся результатом данного шифрования. На MSC отклики сравниваются, и в случае их совпадения аутентификация считается успешной.

#### Подсистема OMC (Operations and Maintenance Center)

Соединена с остальными компонентами сети и обеспечивает контроль качества работы и управление всей сетью. Обрабатывает аварийные сигналы, при которых требуется вмешательство персонала. Обеспечивает проверку состояния сети, возможность прохождения вызова. Производит обновление программного обеспечения на всех элементах сети и ряд других функций.

#### Преимущества и недостатки

##### Преимущества стандарта GSM:

- Меньшие по сравнению с аналоговыми стандартами (NMT-450, AMPS-800) размеры и вес телефонных аппаратов при большем времени работы без подзарядки аккумулятора. Это достигается в основном за счёт аппаратуры базовой станции, которая постоянно анализирует уровень сигнала, принимаемого от аппарата абонента. В тех случаях, когда он выше требуемого, на сотовый телефон автоматически подаётся команда снизить излучаемую мощность.

- Хорошее качество связи при достаточной плотности размещения базовых станций.
- Большая ёмкость сети, возможность большого числа одновременных соединений.
- Низкий уровень промышленных помех в данных частотных диапазонах.
- Улучшенная (по сравнению с аналоговыми системами) защита от подслушивания и нелегального использования, что достигается путём применения алгоритмов шифрования с разделяемым ключом.

- Эффективное кодирование (сжатие) речи.EFR-технология была разработана фирмой Nokia и впоследствии стала промышленным стандартом кодирования/декодирования для технологии GSM (см. GSM-FR, GSM-HR и GSM-EFR)

- Широкое распространение, особенно в Европе, большой выбор оборудования.
- Возможность роуминга. Это означает, что абонент одной из сетей GSM может пользоваться сотовым телефонным номером не только у себя «дома», но и перемещаться по всему миру переходя из одной сети в другую не расставаясь со своим абонентским номером. Процесс перехода из сети в сеть происходит автоматически, и пользователю телефона GSM нет необходимости заранее уведомлять оператора (в сетях некоторых операторов, могут действовать ограничения на предоставление роуминга своим абонентам, более детальную информацию можно получить обратившись непосредственно к своему GSM оператору)

### Недостатки стандарта GSM

- Искажение речи при цифровой обработке и передаче.
- Связь возможна на расстоянии не более 120 км от ближайшей базовой станции даже при использовании усилителей и направленных антенн. Поэтому для покрытия определённой площади необходимо большее количество передатчиков, чем в NMT-450 и AMPS.

### Моделирование канала стандарта GSM в MATLAB Simulink [21]

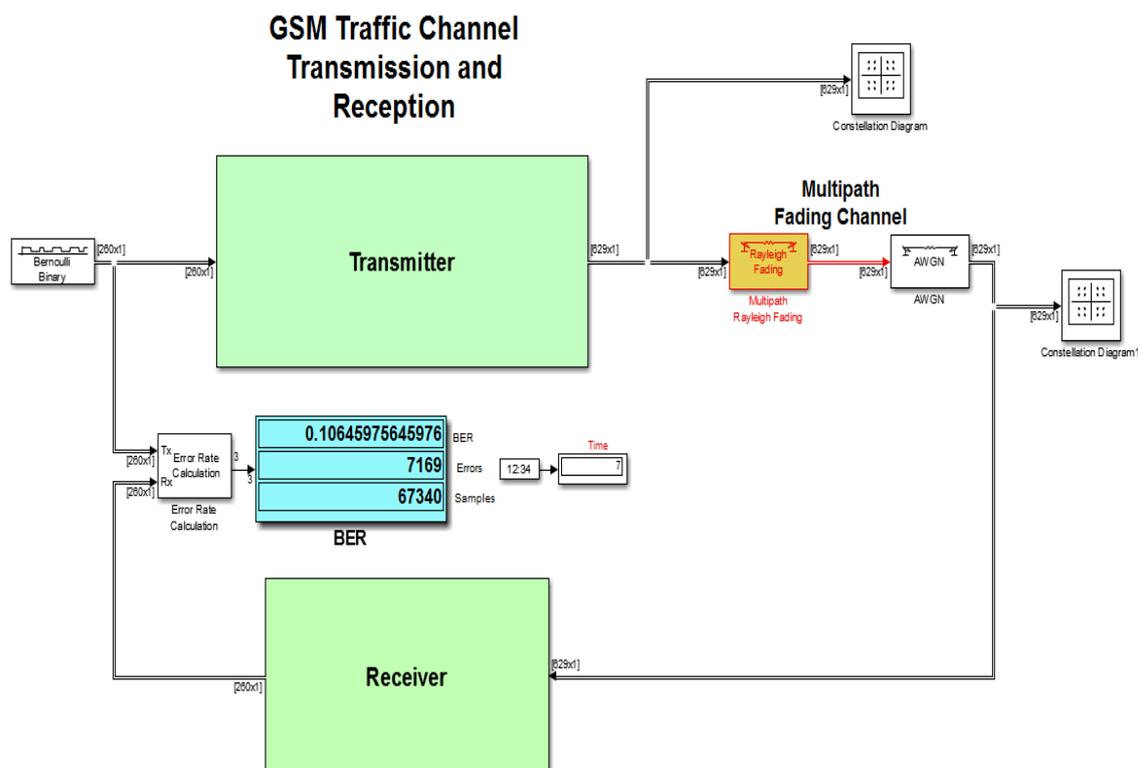


Рис. 5.2. Модель GSM в Simulink MATLAB 2015

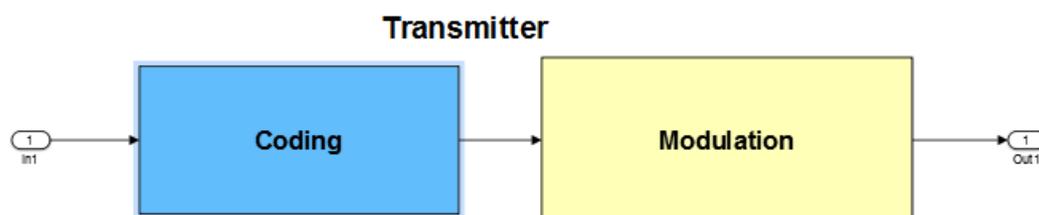


Рис. 5.3. Схема передатчика

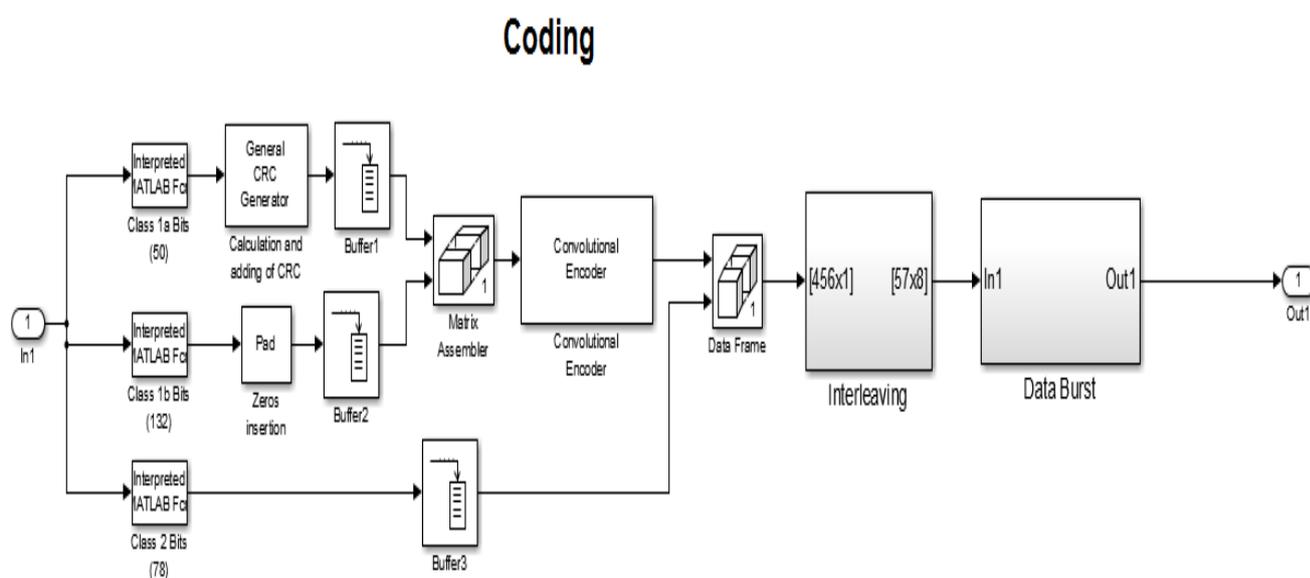


Рис. 5.4. Схема кодера

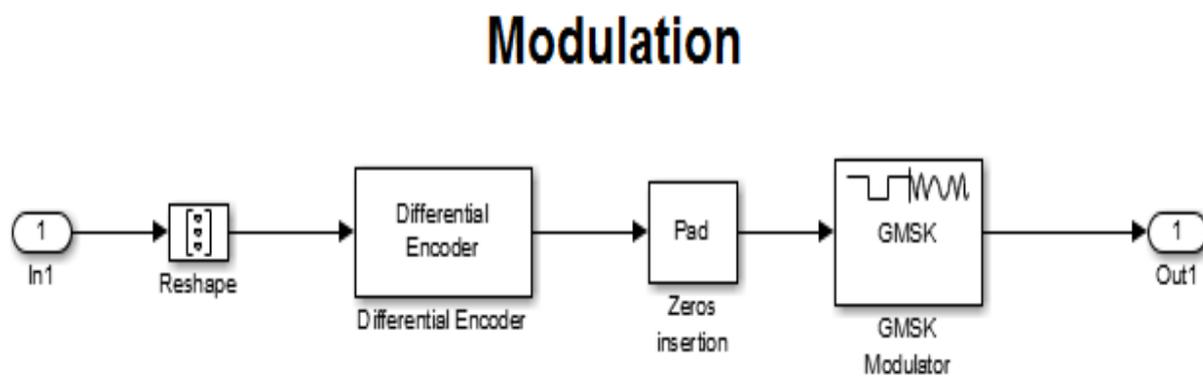


Рис. 5.5. Схема модулятора

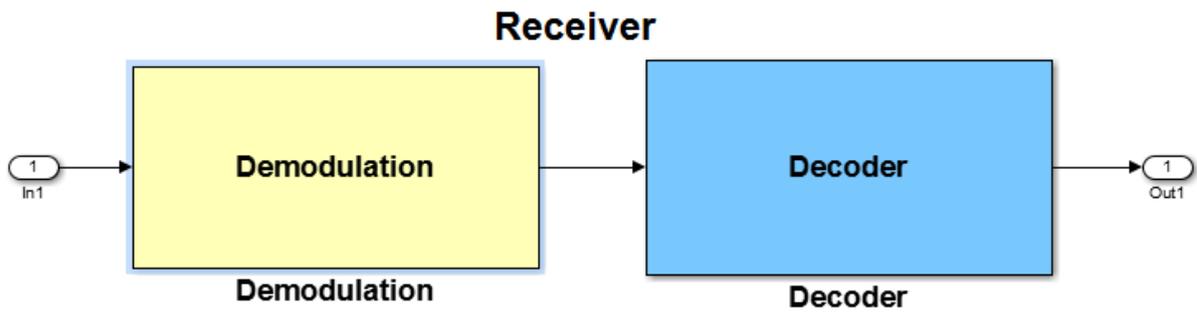


Рис. 5.6. Схема приемника

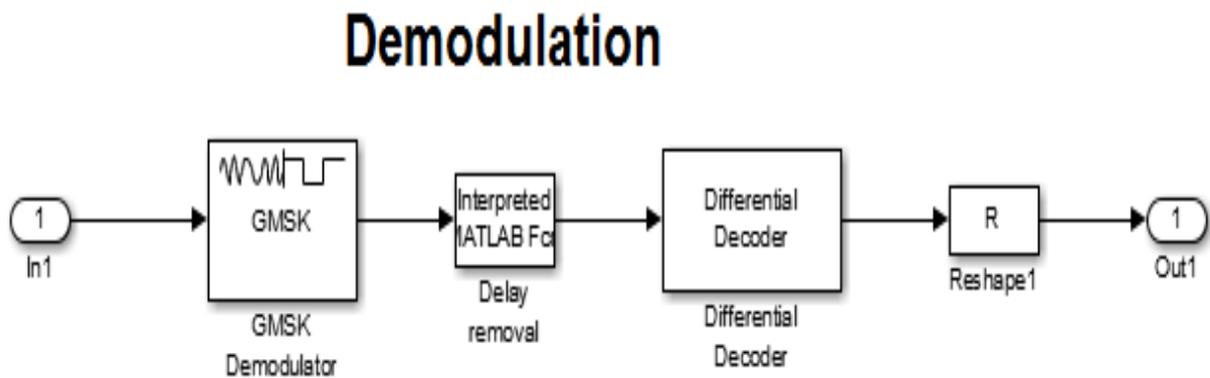


Рис. 5.7. Схема демодулятора

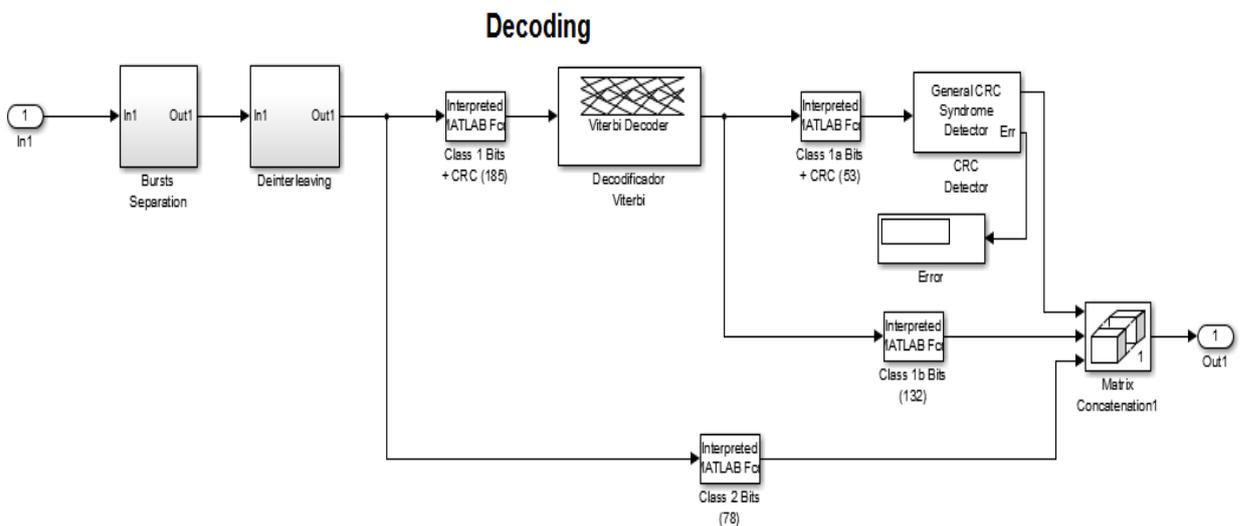


Рис. 5.8. Схема декодера

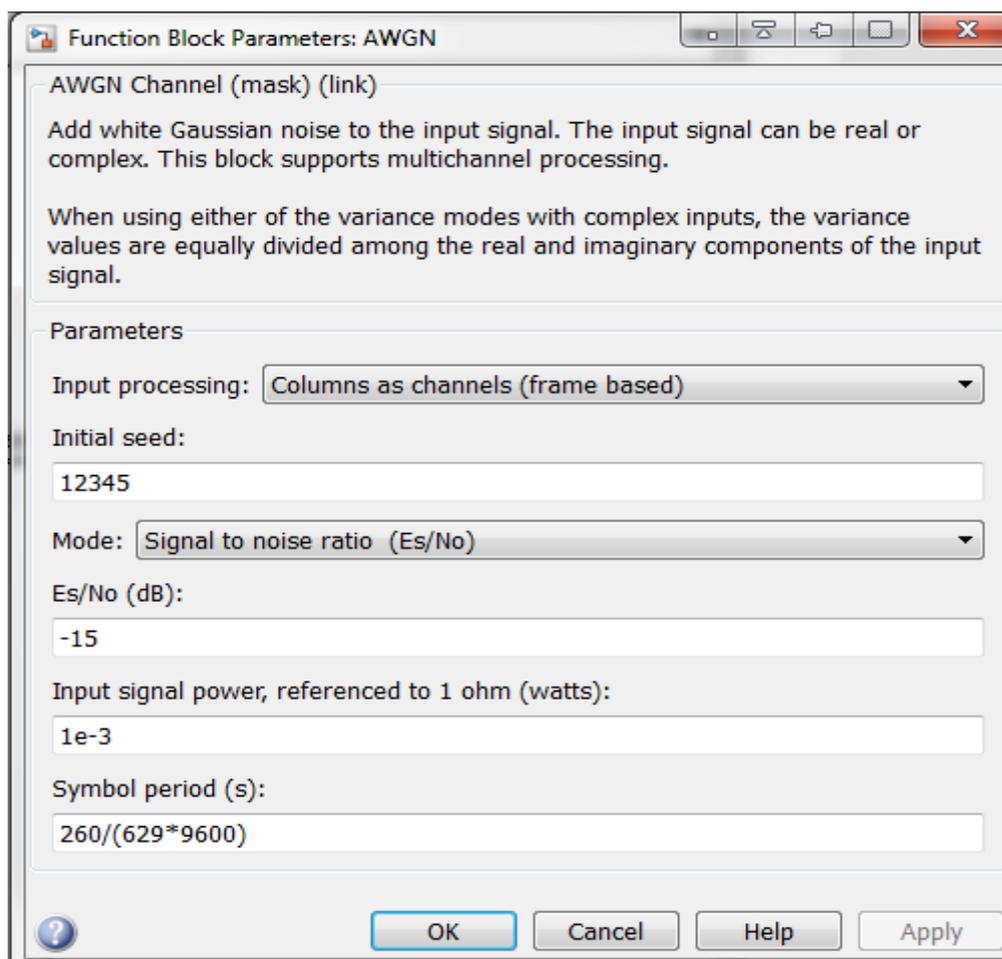


Рис. 3.9. Изменение отношения сигнал/шум.

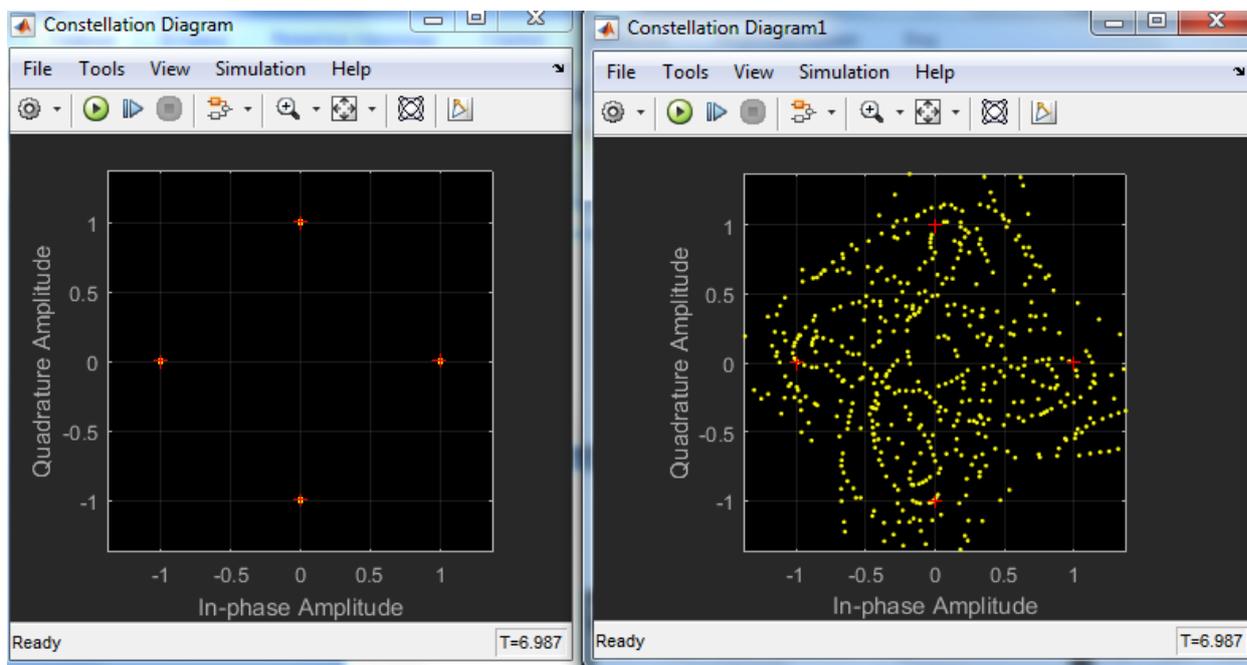


Рис. 3.10. Сравнение передаваемого созвездия и принятого, при отношении С/Ш – 20 Дб.

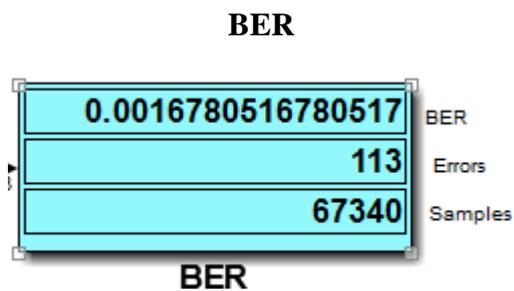


Рис. 3.11. BER при отношении С/Ш 15 Дб.

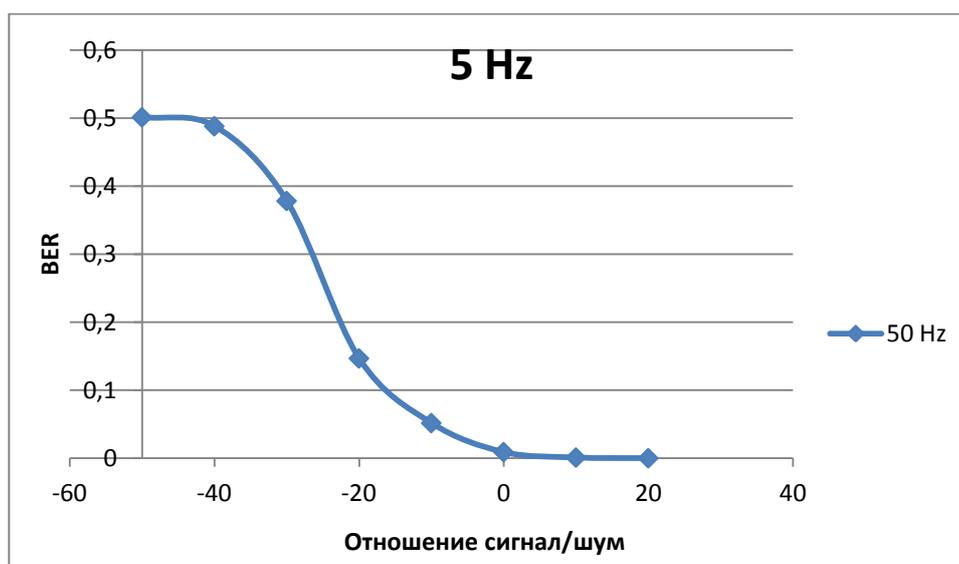


Рис. 3.12. Зависимость BER от отношения сигнал/шум при доплеровском сдвиге 5 Гц

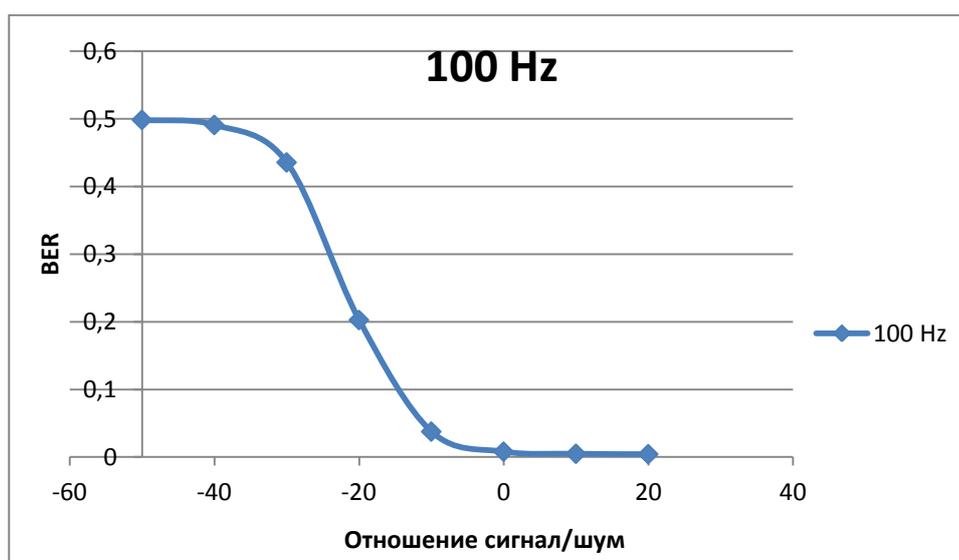


Рис. 3.13. Зависимость BER от отношения сигнал/шум при доплеровском сдвиге

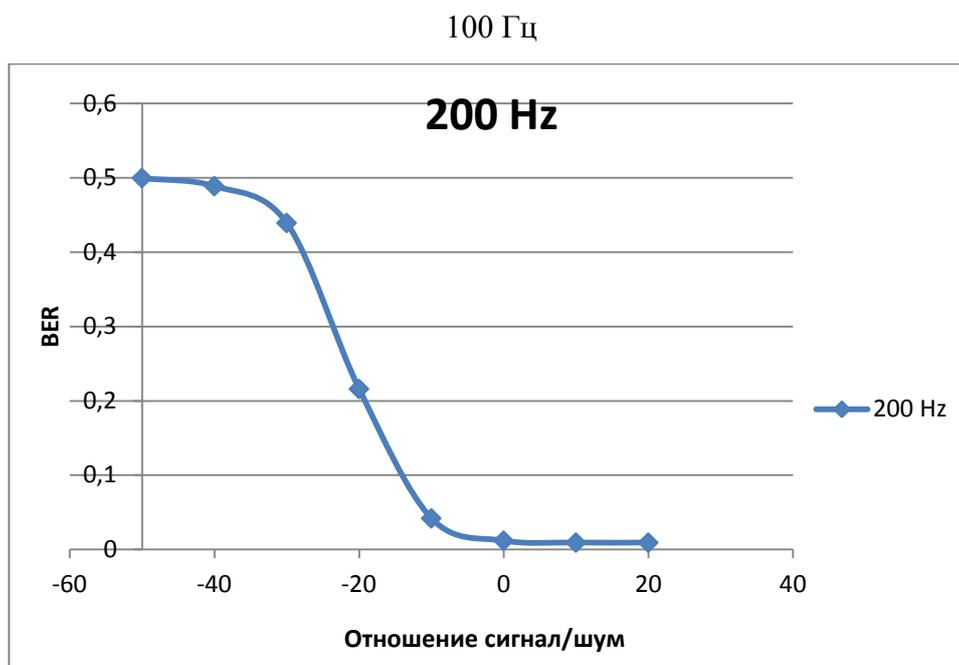


Рис. 3.14. Зависимость BER от отношения сигнал/шум при доплеровском сдвиге

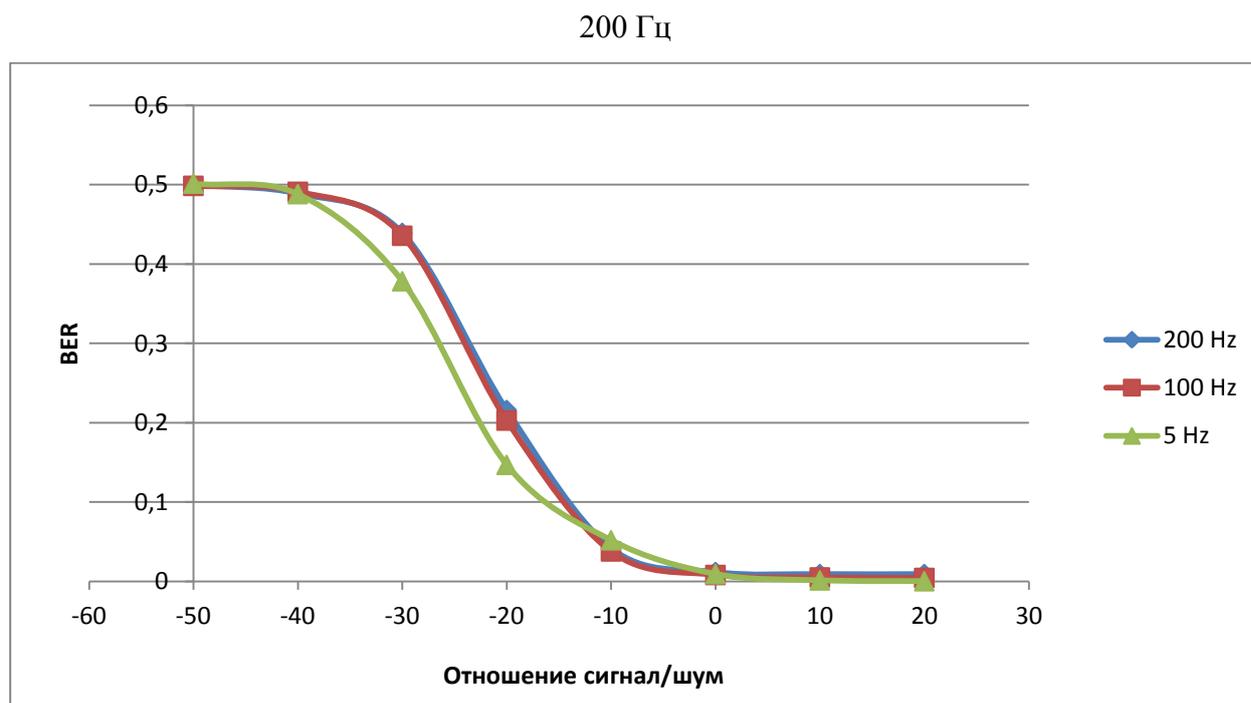


Рис. 3.15. Зависимость BER от отношения сигнал/шум при доплеровском сдвиге

5, 100, 200 Гц

В ходе данной работы мы исследовали стандарт GSM. Он позволяет производить эффективную передачу сигнала при довольно малом соотношении Сигнал Шум. Сигнал GSM зависим от Доплеровского эффекта, но учитывая, что скорость ЭМ волн крайне большая, по сравнению со скоростью объектов связи, то этот эффект нормализуется.

## **5.2. Имитационное моделирование системы мобильной связи стандарта CDMA на базе ПО MATLAB [21]**

В настоящее время развиваются системы мобильной связи, так как каждый год осуществляется рост числа абонентов, что приводит к загруженности сети, необходимости улучшения качества связи, улучшения емкости базовых станций, а также увеличения зоны покрытия сот. Но необходимо улучшать и безопасность мобильной связи, так как злоумышленники могут осуществить перехват информационного сигнала.

Новые поколения сотовой связи появляются достаточно быстро, но их внедрение требует значительных временных ресурсов, поэтому до сих пор основополагающими считаются технологии CDMA и GSM, но технология CDMA работает не только как отдельный стандарт, эта технология используется, например, в LTE.

CDMA - система множественного доступа с кодовым разделением - стала, возможно, самой многообещающей системой, появившейся на мировом рынке. Десятилетия назад эта технология использовалась в военной связи (США), а сегодня известна всем как глобальный цифровой стандарт для коммерческих систем коммуникаций. Технология использования CDMA была протестирована, стандартизирована, лицензирована и запущена в производство большинством поставщиков беспроводного оборудования и применяется во всем мире. В отличие от других методов доступа абонентов к сети, где энергия сигнала концентрируется на выбранных частотах или временных интервалах, сигналы CDMA распределены в непрерывном частотно-временном пространстве. Фактически метод манипулирует и частотой, и временем, и энергией.

CDMA применяется в 32 странах Азии и Океании, 2 странах Северной Америки, 14 странах Европы и 45 странах Африки.

История технологии CDMA берёт своё начало в 30-е годы прошлого (XX) столетия. В 1935 году в СССР академик Агеев Дмитрий Васильевич издал небольшим тиражом брошюру под странным названием "Кодовое разделение каналов". В ней были определены основы ортогонального разделения сигналов, разделения сигналов по форме. В то время реально существовал только один способ разделения каналов связи – частотный. И относилось это, в основном, к каналам радиосвязи. При таком методе каждый канал занимает некоторую свою полосу в общем спектре частот. Эти полосы относительно узки и разделены между собой защитными интервалами. Частотный диапазон ещё не был так перегружен как сегодня, поэтому использование такого способа разделения каналов связи считалось достаточно простым и логичным, поскольку осуществлялась манипуляция только одним параметром сигнала – частотой. Однако учёные, работавшие в области разработок новейших систем

связи, в общем, и радиосвязи, в частности, понимали, что такая идиллия не будет долгой. Кроме того, узкополосные радиосигналы очень чувствительны к селективным замираниям. Требовалось разработать методику, минимизирующую потери полезного сигнала за счёт селективных замираний и позволяющую бережнее относиться к используемому диапазону частот.

Несколько позже, примерно в одно и то же время, появляются работы «Математическая теория связи» Клода Шеннона (США) и «Теория потенциальной помехоустойчивости» Владимира Александровича Котельникова (СССР).

Впервые радиооборудование, использующее кодовое разделение каналов, появилось в США где-то в конце 50-х годов. Технология CDMA нашла применение в военных системах, где успешно отработала более двух десятков лет. Во второй половине 80-х годов военное ведомство США рассекретило данную технологию и разрешило ее использование в гражданских средствах радиосвязи (диапазон 800 МГц).

В сентябре 1995 года в Гонконге фирма HUTCHISON начала развертывание первой в мире коммерческой сети CDMA, используя базовое оборудование Motorola (базовые станции SC 9600 и коммутирующее оборудование EMX 2500) и мобильные телефоны Qualcomm. На конец 1996 года эта сеть насчитывала 113 сот, работала на одном частотном канале с полосой 1,25 МГц и обслуживала более 40.000 абонентов. Правда, соты CDMA были наложены на существующую сеть AMPS и мобильные терминалы работали в дуалмодовом режиме, т.е. при сбое в CDMA-сети абонентский терминал автоматически переключался в сеть AMPS (FDMA). В Корее в январе 1996 года фирма KMT, используя оборудование Gold Star, начала коммерческую эксплуатацию CDMA-сети. А в апреле Shinsengi Telecom начала создавать новую сеть на базе оборудования Samsung, Sony, Qualcomm. На конец 1996 года эти сети обслуживали более 200.000 клиентов. Корея приняла IS-95 в качестве национального стандарта сотовой связи. В США развертыванием CDMA-сетей занимаются такие фирмы, как Air Touch (Сан-Диего, Лос-Анджелес), BANM (Трентон, Нью-Джерси), 360-Communications (Лас-Вегас, Невада). Они используют базовое оборудование Qualcomm, Lucent Technologies, Motorola, а также абонентские терминалы фирм Qualcomm, Sony, Nortel. В Австралии, в канун Олимпийских игр, были построены сети сотовой мобильной радиотелефонной связи в Сиднее и Мельбурне на базе оборудования CDMA-one (IS-95) производства фирмы Samsung.

Кроме вышеназванного стандарта (IS-95) в 1999 году был разработан и широкополосный вариант - W-CDMA (Ericsson, Швеция), функционирующий в диапазоне 1800 МГц. Он предназначался для использования в районах с высокой плотностью населения, так как обладал ещё большей пропускной способностью.

## **Стандарты CDMA**

В CDMA системах каждый голосовой поток отмечен своим уникальным кодом и передается на одном канале одновременно со многими другими кодированными голосовыми потоками. Принимающая сторона использует тот же код для выделения сигнала из шума. Единственное отличие между множественными голосовыми потоками это уникальный код. Канал, как правило, очень широк и каждый голосовой поток занимает целиком всю ширину диапазона. Эта система использует наборы каналов шириной 1.23МГц. Голос кодируется на скорости 8.55кбит/с, но определение голосовой активности и различные скорости кодирования могут урезать поток данных до 1200бит/с. В системах CDMA могут устанавливаться очень прочные и защищенные соединения, несмотря на экстремально низкую величину мощности сигнала, теоретически - сигнал может быть слабее, чем уровень шума

### **Стандарт CDMAOne**

Стандарт cdmaOne, существует в вариациях IS-95a, IS-95b (cellular по американской терминологии, 800 МГц) и J-STD-008 (PCS, диапазон 1900). Аббревиатура IS (interim standard - временной стандарт) используется для учета в Ассоциации телекоммуникационной промышленности TIA (Telecommunications Industry Association). Как правило, в сетях cdmaOne используется IS-95a, он обеспечивают передачу сигнала со скоростью 9,6 кбит/с (с кодированием) и 14,4 кбит/с (без кодирования). Версия IS-95b основана на объединении нескольких каналов CDMA, организуемых в прямом направлении (от базовой станции к мобильной). Скорость может увеличиваться до 28,8 кбит/с (при объединении двух каналов по 14,4 кбит/с) или до 115,2 кбит/с (8 каналов по 14,4 кбит/с). Собственно, кроме IS-95 сети cdmaOne используют еще целый набор протоколов и стандартов, их список можно найти в любой достаточно глубокой статье по этой теме. Прямой и обратный каналы располагаются соответственно в диапазонах 869,040-893,970 и 824,040-848,860 МГц. Используются 64 кода Уолша и несущие в 1.25 МГц.

### **Стандарт WCDMA**

WCDMA (Wideband Code Division Multiple Access - широкополосный CDMA) - технология радиointерфейса избранная большинством операторов сотовой связи Японии и (в январе 1988 года) институтом ETSI (European Telecommunications Standards Institute) для обеспечения широкополосного радиодоступа с целью поддержки услуг третьего поколения.

Технология оптимизирована для предоставления высокоскоростных мультимедийных услуг типа видео, доступа в Интернет и видеоконференций; обеспечивает скорости доступа вплоть до 2 Мбит/с на коротких расстояниях и 384 Кбит/с на больших с полной мобильностью. Такие величины скорости передачи данных требуют широкую полосу частот, поэтому ширина полосы WCDMA составляет 5 МГц. Технология может быть добавлена к существующим сетям GSM и PDC, что делает стандарт WCDMA наиболее перспективным с точки зрения использования сетевых ресурсов и глобальной совместимости.

WCDMA (широкополосный множественный доступ с кодовым разделением каналов) представляет собой технологию, использующую расширенную полосу пропускания и разновидность принципа DMA. Это технология мобильной радиосвязи третьего поколения, обеспечивающая значительно более высокие скорости передачи данных, чем стандарт GSM. WCDMA поддерживает передачу голоса, изображений, данных и видео в сетях мобильной связи на скорости до 2 Мбит/с (локальный доступ) или 384 кбит/с (глобальный доступ). WCDMA используется в основном в Европе при переходе от стандарта GSM к стандарту UMTS.

### **Стандарт CDMA2000**

Стандарт cdma2000 является дальнейшим развитием стандарта 2 поколения cdmaOne. Дальнейшим развитием cdmaOne должен был стать IS-95c, и именно это обозначение очень часто используется производителями. Официальным обновлением стандарта, разработанным компанией Qualcomm и утвержденным ITU (Международный союз электросвязи, International Telecommunication Union), является cdma2000. В документах Lucent Technologies встречается обозначение IS-2000. Наконец, международный союз электросвязи (МСЭ) отобрал из десяти предложенных проектов пять радиоинтерфейсов третьего поколения IMT-2000 (International Mobile Telecommunications System - 2000 - Международная система мобильной связи - 2000), в их числе - IMT-MC (Multi Carrier), который представляет собой модификацию многочастотной системы cdma2000, в которой обеспечивается обратная совместимость с оборудованием стандарта cdmaOne (IS-95).

Еще один из пяти стандартов IMT-2000 - IMT-DS (Direct Spread) - построен на базе проектов W-CDMA и взят за основу европейской системы UMTS.

На начало 2003г. из 127 миллионов пользователей CDMA почти 15 миллионов использовали технологию cdma2000. В течение первых семи месяцев 2002 года, в Азии и Америке было запущено 11 сетей CDMA2000 и общее количество этих сетей составляло 18.

Это - 99% рынка 3G, на IMT-МС приходилось 14.8 миллионов абонентов, на UMTS - 0.13 миллиона. Однако, стоит отметить, что реализованная фаза cdma2000 1X все же не является полноценным 3G, ибо не дотягивает до обязательных двух мегабит. Поэтому ее чаще называют 2.5G.

Изначально cdma2000 (IMT-МС) разделили на две фазы - 1X и 3X. Именно к первой фазе применяется название IS-95C. А вторую позже назвали 1X-EV (evolution), разделив ее на две фазы - cdma2000 1X EV-DO (data only) и cdma2000 1X EV-DV (data & voice).

И именно стандарт cdma2000 1X EV-DO подразумевается под 3G IMT-МС. Стандарт 1x-EV-DO был принят TTA в октябре 2000 года и предусматривает следующую схему функционирования: аппарат одновременно производит поиск сети 1x и 1xEV, передачу данных осуществляет с помощью 1xEV, голоса - с помощью 1x. Стандарт 1xEV-DV полностью соответствует всем требованиям 3G.

Следует отметить, что стандарты семейства cdma2000 не требуют организации отдельной полосы частот и в ходе их эволюционного развития от cdmaOne могут быть реализованы во всех частотных диапазонах, используемых системами сотовой подвижной связи (450, 700, 800, 900, 1700, 1800, 1900, 2100 МГц).

## Структура и формирование сигналов

### Схема кодирования в прямом канале (от базовой станции к абоненту).

Базовая скорость передачи данных в канале составляет 9,6 кбит/с, что достигается добавлением дополнительных корректирующих двоичных символов к цифровому потоку вокодера 8,55 кбит/с.

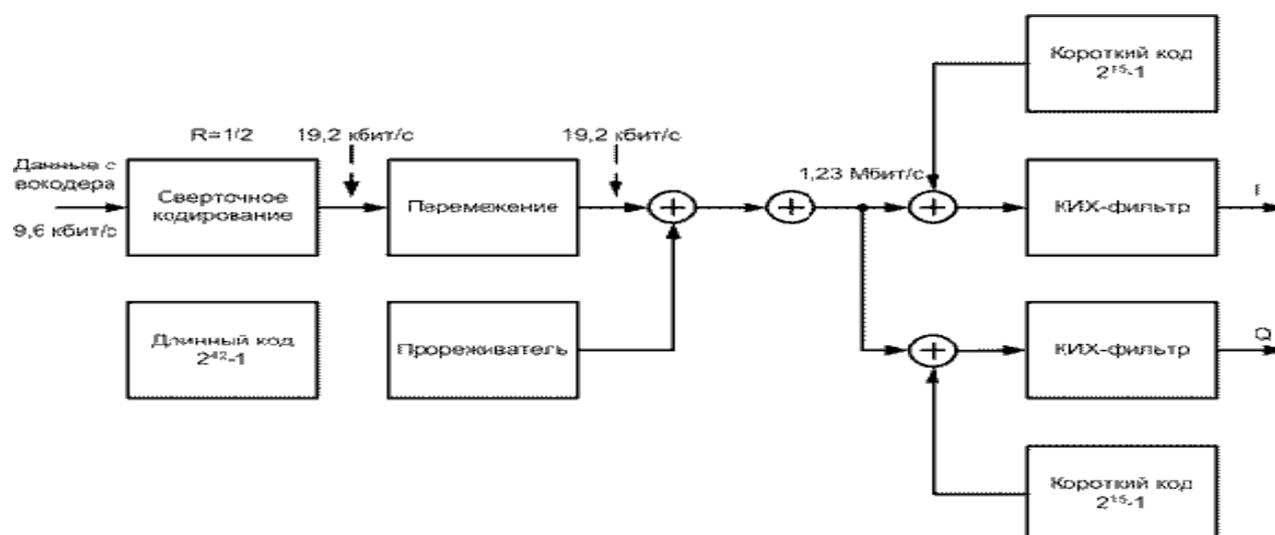


Рис. 5.16. Схема кодирования в прямом канале

Для реализации на приемной стороне прямой коррекции ошибок (без повторной передачи сообщения) в канале используется избыточное кодирование. Для этого базовый цифровой поток разбивается на пакеты длительностью по 20 мс и подается на сверточный кодер с половинной скоростью. На его выходе число битов удваивается. Затем данные перемежаются, т. е. перемешиваются во временном интервале 20 мс. Это делается для того, чтобы равномерно распределить в потоке данных (после обратного перемежения) потерянные во время передачи биты. Известно, что ошибочно принятые символы обычно формируют группы. В то же время, схема прямой коррекции ошибок работает наилучшим образом, когда ошибки распределены равномерно во времени. Это происходит после осуществления на приемной стороне процедуры, обратной перемежению при передаче. После перемежения цифровой поток преобразуется с помощью длинного кода и логической операции "исключающее ИЛИ" (сложение по модулю два). По определению, длинными кодами (кодами максимальной длины - M-последовательностями) являются коды, которые могут быть получены с помощью регистра сдвига или элемента задержки заданной длины.

Максимальная длина двоичной последовательности, которая может быть получена с помощью генератора, построенного на основе регистра сдвига, равна  $2^n - 1$  двоичных символов, где  $n$  - число разрядов регистра сдвига. В аппаратуре стандарта CDMA длинный код формируется в результате нескольких последовательных логических операций с псевдослучайной двоичной последовательностью, генерируемой в 42-разрядном регистре сдвига, и двоичной 32-битовой маской, которая определяется индивидуально для каждого абонента. Такой регистр сдвига применяется во всех базовых станциях этого стандарта для обеспечения режима синхронизации всей сети. Длина M-последовательности при этом составляет 4 398 046 511 103 бит и если ее элементы формируются с тактовой частотой, например, 450 МГц, то период повторения будет составлять  $9773,44 \text{ с} = 2 \text{ ч } 43 \text{ мин}$ . Это значит, что если даже удастся засинхронизировать приемник в случае несанкционированного перехвата, то чтобы определить структуру сигнала-носителя необходимо вести наблюдение в течение почти 3-х часов, а с применением индивидуальной 32-битовой маски "подслушивание" практически исключено. Так как информационный поток имеет скорость 19,2 Кбит/с, то в прямом канале используется только каждый 64-й символ длинного кода. Следующий этап преобразования сообщения - кодирование с помощью кодов Уолша. Любая строка матрицы Уолша ортогональна другой строке. Матрица Уолша размером 2 имеет вид:

$$W_2 = \begin{pmatrix} 1 & 1 \\ 1 & -1 \end{pmatrix}$$

Матрицы больших размеров образуются следующим образом:

$$W_{2N} = \begin{pmatrix} W_N & W_N \\ W_N & -W_N \end{pmatrix}$$

т.е., например,

$$W_8 = \begin{pmatrix} 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 \\ 1 & -1 & 1 & -1 & 1 & -1 & 1 & -1 \\ 1 & 1 & -1 & -1 & 1 & 1 & -1 & -1 \\ 1 & -1 & -1 & 1 & 1 & -1 & -1 & 1 \\ 1 & 1 & 1 & 1 & -1 & -1 & -1 & -1 \\ 1 & -1 & 1 & -1 & -1 & 1 & -1 & 1 \\ 1 & 1 & -1 & -1 & -1 & -1 & 1 & 1 \\ 1 & -1 & -1 & 1 & -1 & 1 & 1 & -1 \end{pmatrix}$$

Можно показать, что строки матрицы Уолша ортогональны. Ортогональность строк  $x$  и  $y$  длиной  $N$  определяется следующим условием:

$$\sum_{i=1}^N x_i y_i = 0$$

По сути в этом случае вычисляется значение ВКФ двух различных строк при временном сдвиге равном нулю.

Один ряд матрицы Уолша ставится в соответствие каналу связи между абонентом и базовой станцией. Если на входе кодера "0", то посылается соответствующий ряд матрицы (код Уолша), если "1" - посылается последовательность, сформированная путем логического отрицания соответствующего ряда матрицы (кода Уолша). При точном совпадении начала пришедшей последовательности и имеющейся (строка матрицы  $W_{64}$ ) наблюдаются пики корреляционной функции положительной и отрицательной полярностей - в зависимости от передаваемого бита. В случае обработки "чужого" сигнала на выходе в момент окончания сигнала не будет ничего, т.е. происходит разделение каналов при приеме абонентской станцией. Кодирование по Уолшу повышает скорость информационного потока с 19,2 Кбит/с до 1,2288 Мбит/с. Соответственно расширяется и спектр сигнала. На заключительном этапе двоичный поток разделяется между синфазным и квадратурным каналами (I- и Q-каналами) для последующей передачи с использованием квадратурной фазовой манипуляции (QPSK). До подачи на смесители цифровой поток в каждом из каналов преобразуется с помощью короткого кода и операции сложения по модулю два.

Короткий код представляет собой псевдослучайную двоичную последовательность длиной 32768 двоичных символов, генерируемую со скоростью 1,3288 Мбит/с. Эта

последовательность является общей для всех базовых и подвижных станций в сети. Короткий код формируется в 15-разрядном регистре сдвига с линейной обратной связью. Результирующий двоичный поток в каждом канале проходит через цифровой фильтр с конечной импульсной характеристикой (КИХ-фильтр), что позволяет ограничить полосу излучаемого сигнала. Частота среза фильтра составляет около 615 кГц. Полученные аналоговые сигналы поступают на соответствующие входы I/Q-модулятора. Ряд информационных сигналов образуется путем слияния I- и Q-каналов.

Поскольку все пользователи получают объединенный сигнал, то для выделения информации необходимо передавать опорный сигнал по каналу, получившему название пилотного. В пилотном канале передается нулевой информационный сигнал, код Уолша для этого канала формируется из нулевого ряда матрицы Уолша (все единицы). Другими словами, в пилотном канале передается только короткий код. Обычно на нем излучается около 20% общей мощности. Опорный сигнал необходим для последующей фазовой демодуляции. Короткий код позволяет многократно использовать в каждой ячейке один и тот же набор кодов Уолша. Каждая базовая станция имеет свой временной сдвиг при формировании кода и поэтому может быть однозначно определена в сети. Основано это на уже описанном свойстве псевдослучайных двоичных последовательностей: значение АКФ близко к нулю для всех временных смещений более одной длины бита.

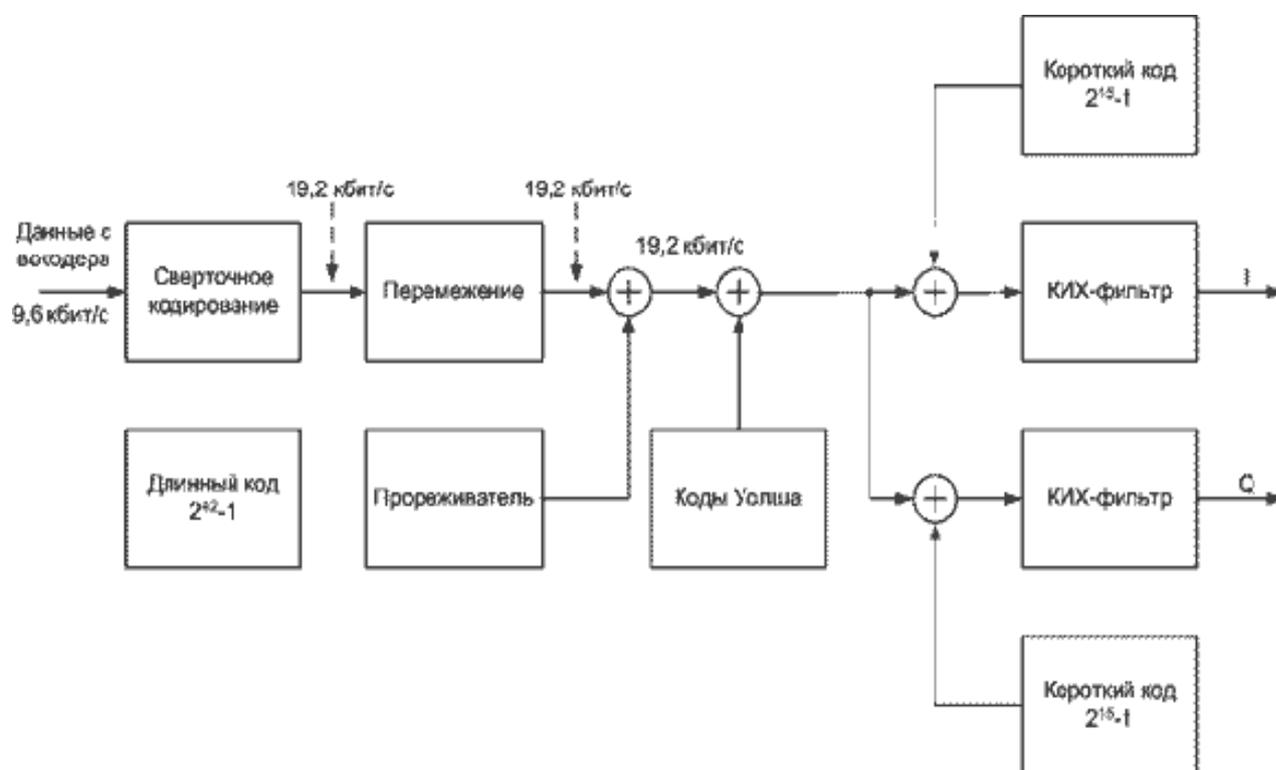


Рис. 5.17. Схема кодирования в обратном канале.

В обратном канале (от абонента к базовой станции) применяется другая схема кодирования. Подвижная станция не может использовать преимуществ трансляции опорного сигнала. В этом случае необходимо было бы передавать два сигнала, что значительно усложнило бы демодуляцию в приемнике базовой станции. В обратном канале применяется такой же, как и в прямом, вокодер и сверточное кодирование со скоростью  $1/3$ , что повышает скорость передачи данных с базовой  $9,6$  до  $28,8$  кбит/с, и перемежение в пакете длительностью  $20$  мс. После перемежения выходной поток разбивается на слова по шесть битов в каждом. Шестибитовому слову можно поставить в соответствие один из  $64$  кодов Уолша. Таким образом, каждый абонентский терминал использует весь их набор. После этой операции скорость потока данных повышается до  $307,2$  Кбит/с. Далее поток преобразуется с помощью длинного кода, аналогичного используемому базовой станцией. На этом этапе происходит разделение пользователей. Абонентская емкость системы определяется обратным каналом. Для ее увеличения применяется регулирование мощности в обратном канале, методы пространственного разнесения приема на базовой станции и др. Окончательное формирование потоков данных происходит таким же образом, как и в базовой станции, за исключением дополнительного элемента задержки на  $1/2$  длительности символа в Q-канале для реализации, смещенной QPSK.

В системе CDMA применяются квадратурная фазовая манипуляция (QPSK) в базовой и смещенная QPSK в подвижных станциях. При этом информация извлекается путем анализа изменения фазы сигнала, поэтому фазовая стабильность системы - критичный фактор при обеспечении минимальной вероятности появления ошибки в сообщениях. Применение смещенной QPSK позволяет снизить требования к линейности усилителя мощности подвижной станции, так как амплитуда выходного сигнала при этом виде модуляции изменяется значительно меньше. До того, как интерференционные помехи будут подавлены методами цифровой обработки сигналов, они должны пройти через высокочастотный тракт приемника и не вызвать насыщения малошумящего широкополосного усилителя (МШУ) и смесителя. Это заставляет разработчиков системы искать баланс между динамическими и шумовыми характеристиками приемника.



Рис. 5.18. Структурная схема CDMA

## Моделирование CDMA2000 1xRTT

Модель состоит из трех основных блоков:

1. Базовая станция (передатчик);
2. Канал;
3. Мобильная станция (приемник).

Канал имеет три режима работы:

1. Нет канала;
2. Канал с шумами;

### Канал с многолучевым распространением.

Мобильный приемник состоит из декодера и приемника, которые выполняют все операции необходимые декодирования сигнала [21].

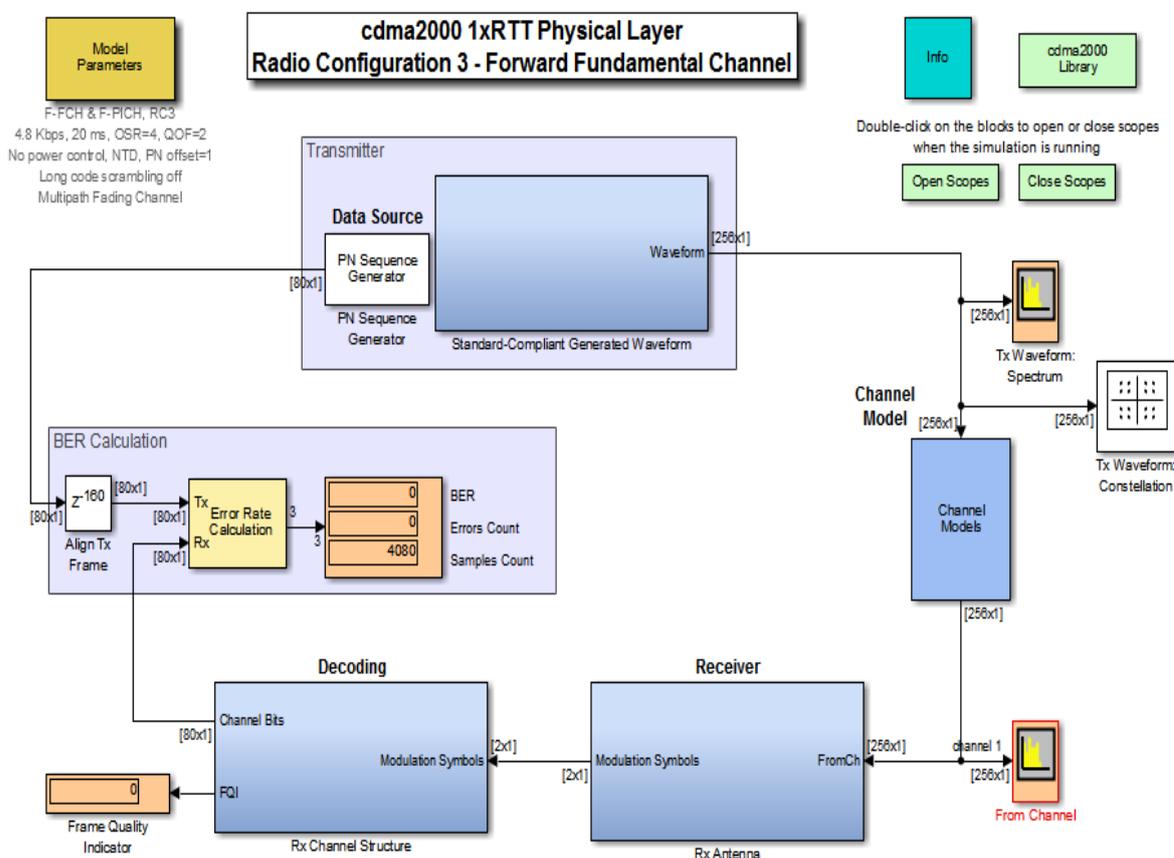


Рис.5.19. Модель CDMA2000 1xRTT в MATLAB R2015b

Развернутая модель передатчика представлена на рисунке 5.20.

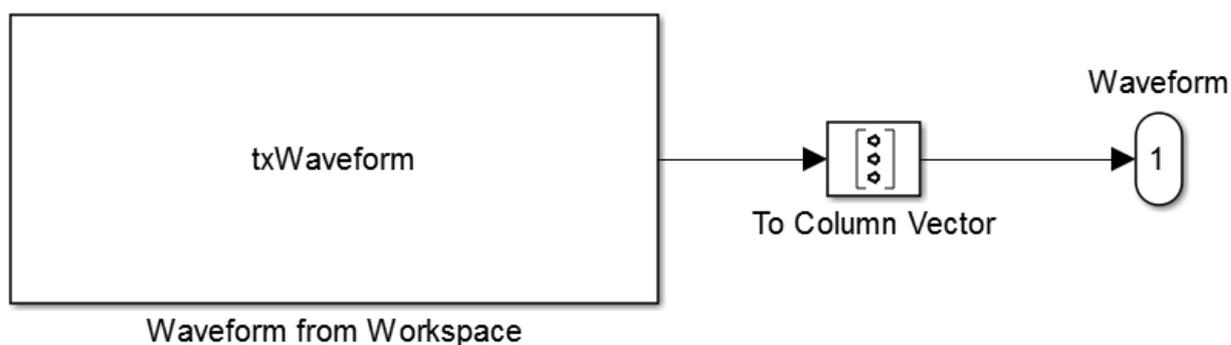


Рис. 5.20. Модель передатчика

Блок txWaveform содержит в себе длинный программный код посредством которого и генерируется сигнал, далее этот сигнал формируется в вектор с помощью блока To Column Vector. Этот вектор передается по каналу и затем поступает в приемник. Развернутая модель приемника представлена на рисунке 5.21.

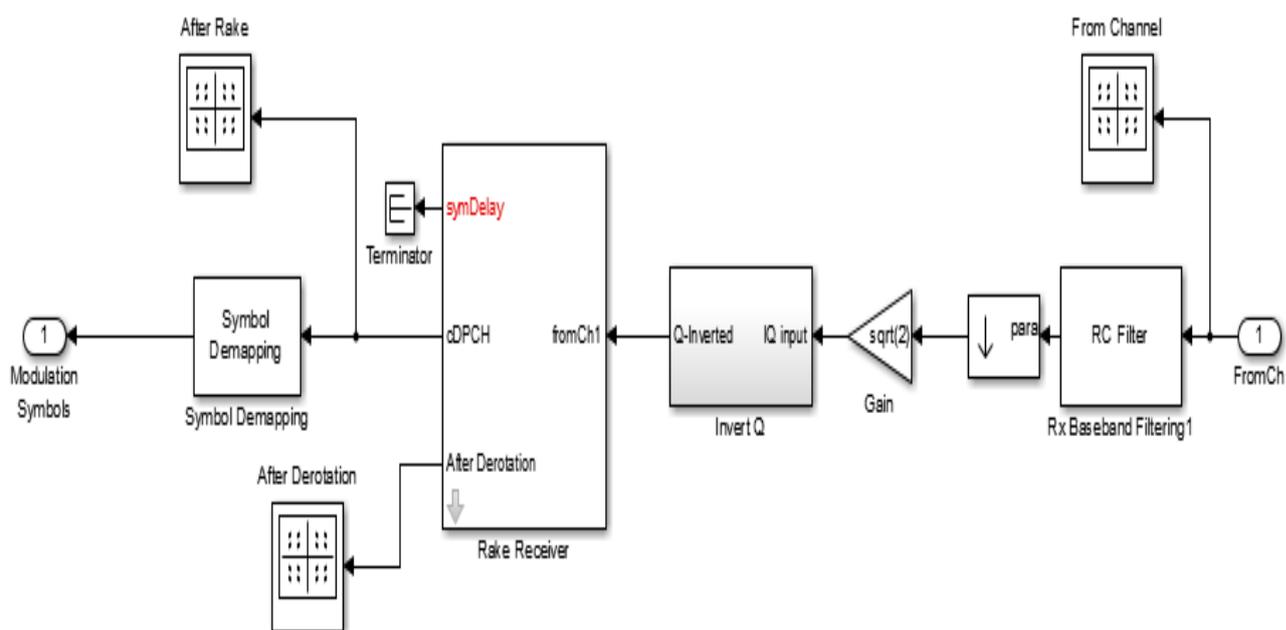


Рис. 5.21. Модель приемника.

Принятый сигнал поступает на фильтр RC Filter, АЧХ которого представлена на рисунке 6, и затем усиливается с помощью блока Gain в корень из двух раз, после чего сигнал поступает в блок Invert Q, который разделяет его на реальную и мнимую части, умножает мнимую часть на -1 и затем объединяет реальную и мнимую части обратно. Далее восстанавливается созвездие с помощью блока Rake Receiver, после чего сигнал поступает в

блок Symbol Demapping для демодуляции. Полученные символы модуляции поступают на декодер, развернутая модель которого представлена на рисунке 5.22.

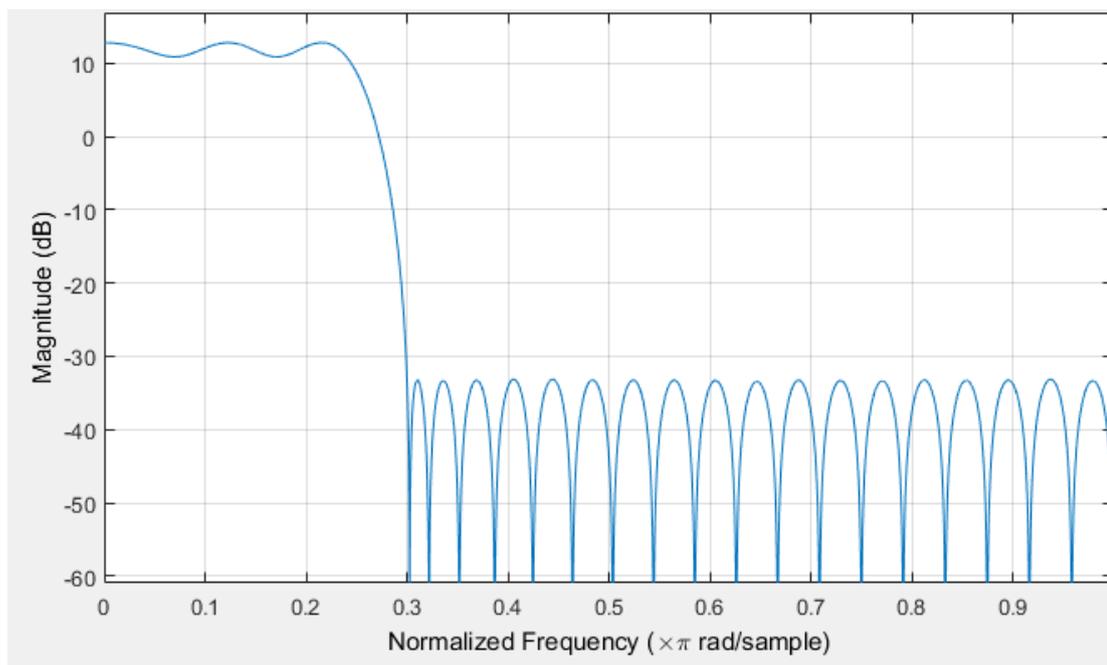


Рис. 5.22. АЧХ фильтра приемника

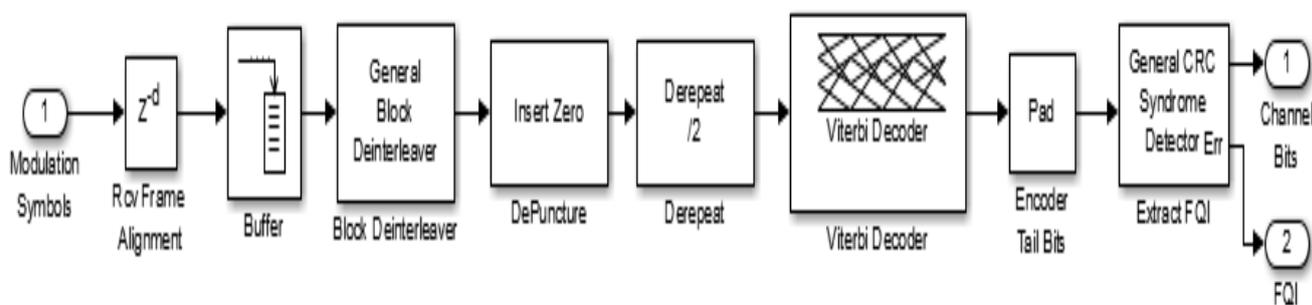


Рис. 5.23. Модель декодера.

Полученные символы модуляции поступают в блок Rcv Frame Alignment, который представляет собой задержку на 768 тактов, далее символы поступают в блок Buffer для накопления 768 символов. Накопленные символы поступают в блок Block Deinterleaver для обратного перемежения, далее данные поступают в блок Insert Zero, который возвращает последовательности нулей, замененных на специальные символы, далее данные поступают в блок Derepeat, обратное преобразование кодов с повторением с коэффициентом повторения 2, далее данные поступают на декодер Витерби и наконец в блок Encoder Tail Bits, который добавляет нули или урезает число бит если оно не равно 80.

## Параметры модели

Модель позволяет изменять такие настройки как скорость потока и вид канала. В зависимости от вида канала можно задавать значение отношения сигнал/шум, а также параметры многолучевого распространения сигнала: максимальное Доплеровское отклонение частоты, вектор задержки и вектор ослабления/усиления. Длины векторов определяют количество лучей в канале.

## Результаты моделирования

Компонент расчета BER сравнивает декодированный сигнал и сигнал, сгенерированный базовой станцией. Если BER равен нулю, то сигнал не подвергся каким-либо изменениям либо ошибки удалось исправить. Сигнал с базовой станции перед попаданием в блок расчета BER проходит через задержку для того что бы выровнять фреймы.

Для того что бы отобразить все возможные графики необходимо два раза кликнуть по кнопке Open Scores в правом верхнем углу. В результате чего отобразятся следующие графики:

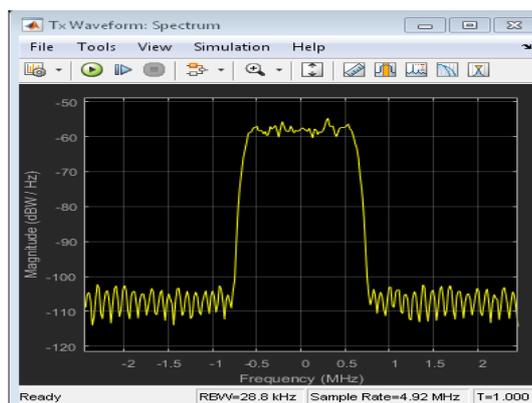


Рис. 5.24. Спектр сигнала сгенерированного базовой станцией.

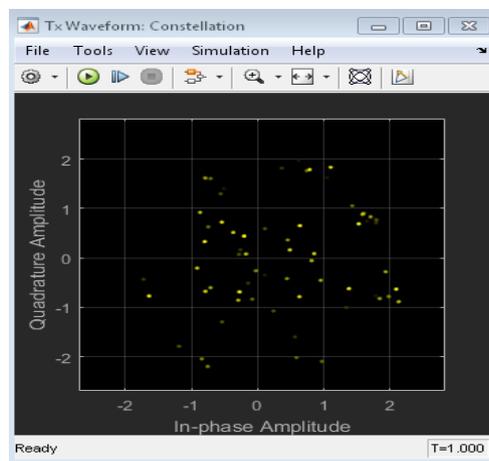


Рис. 5.25. Сгенерированный базовой станцией сигнал на I-Q диаграмме

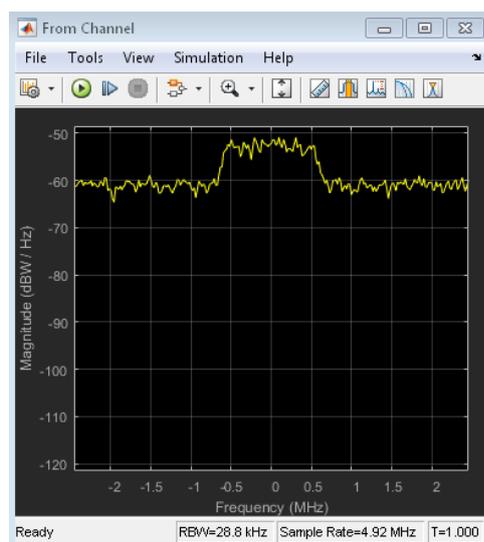


Рис. 5.26. Спектр принимаемого мобильной станцией сигнала после прохождения через канал.

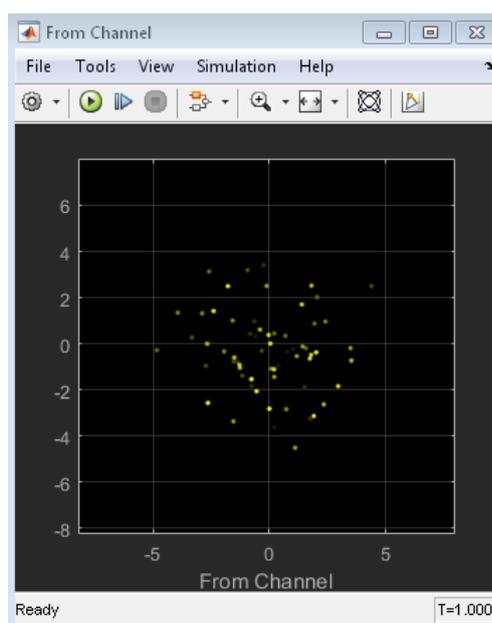


Рис. 5.27. Сигнал принимаемый мобильной станцией после прохождения через канал на I-Q диаграмме.

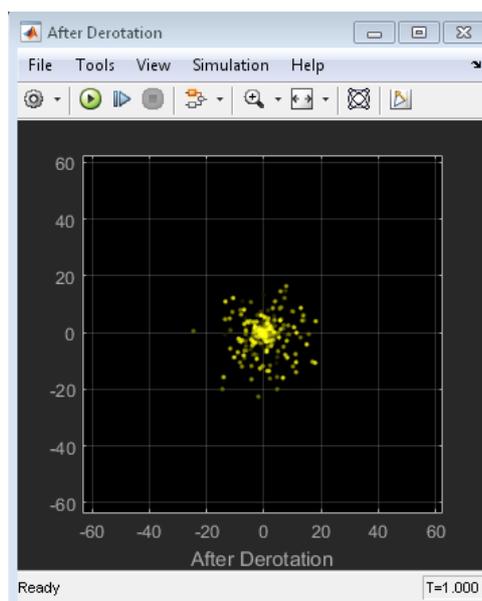


Рис. 5.28. Сигнал, принятый мобильной станцией

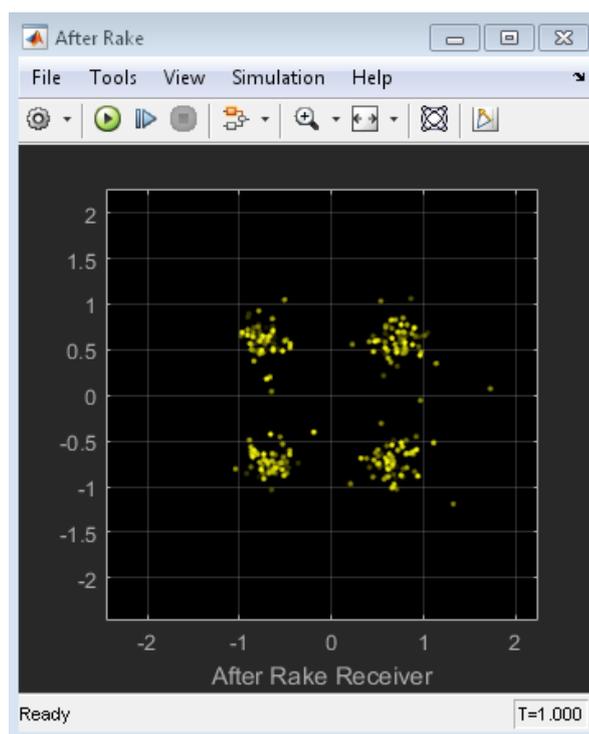


Рис. 5.29. Сигнал, декодированный мобильной станцией, на I-Q диаграмме.

### Исследование модели

В блоке Model Parameters во вкладке Channel Settings выберем Channel Model: No Channel.

Результат моделирования:

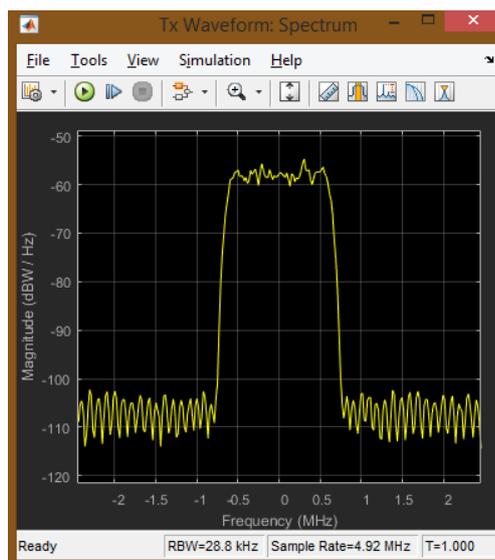


Рис. 5.30. Спектр сигнала сгенерированного базовой станцией

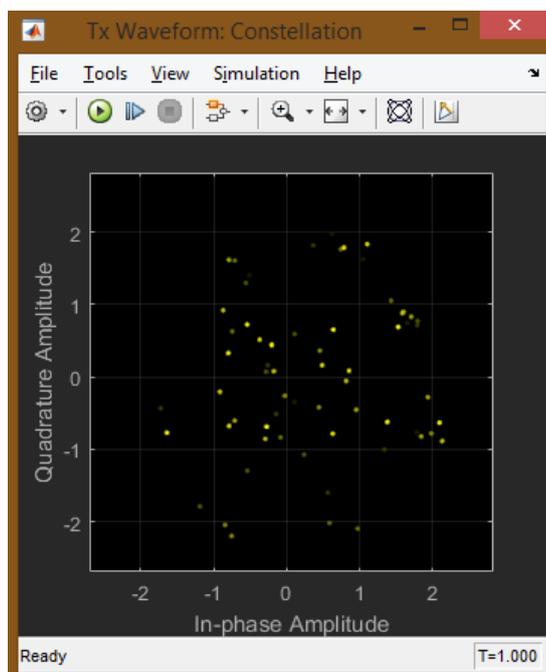


Рис. 5.31. Сгенерированный базовой станцией сигнал на I-Q диаграмме.

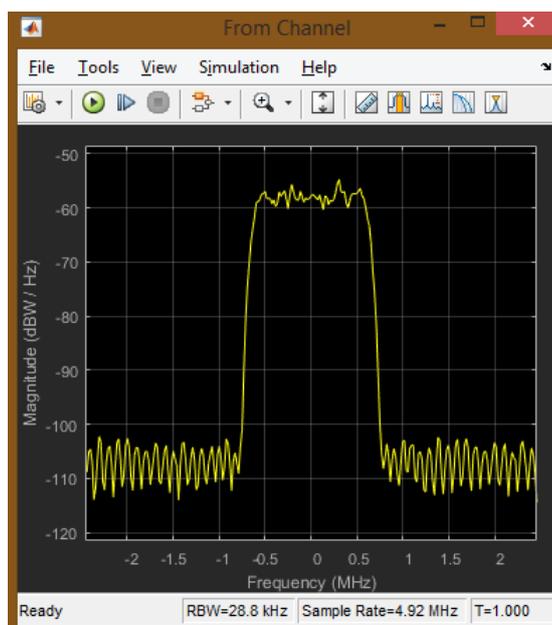


Рис. 5.32. Спектр сигнала после канала

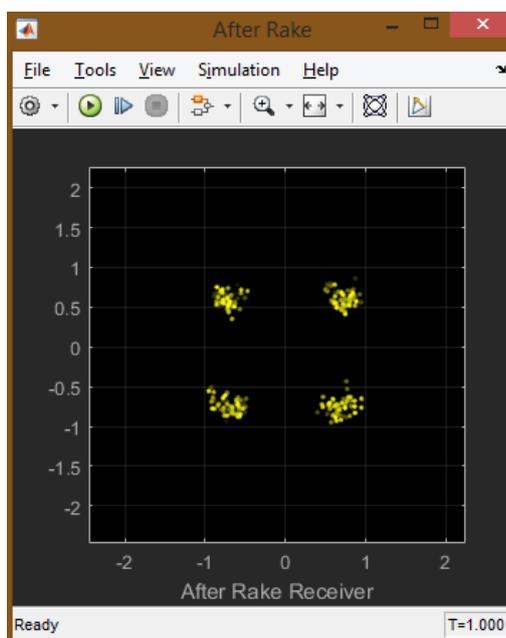


Рис. 5.33. Сигнал, декодированный мобильной станцией, на I-Q диаграмме

Видно, что спектр сигнала не изменился, так как в канале не было потерь. По результатам моделирования BER равен нулю.

В блоке Model Parameters во вкладке Channel Settings выберем Channel Model: AWGN Channel.

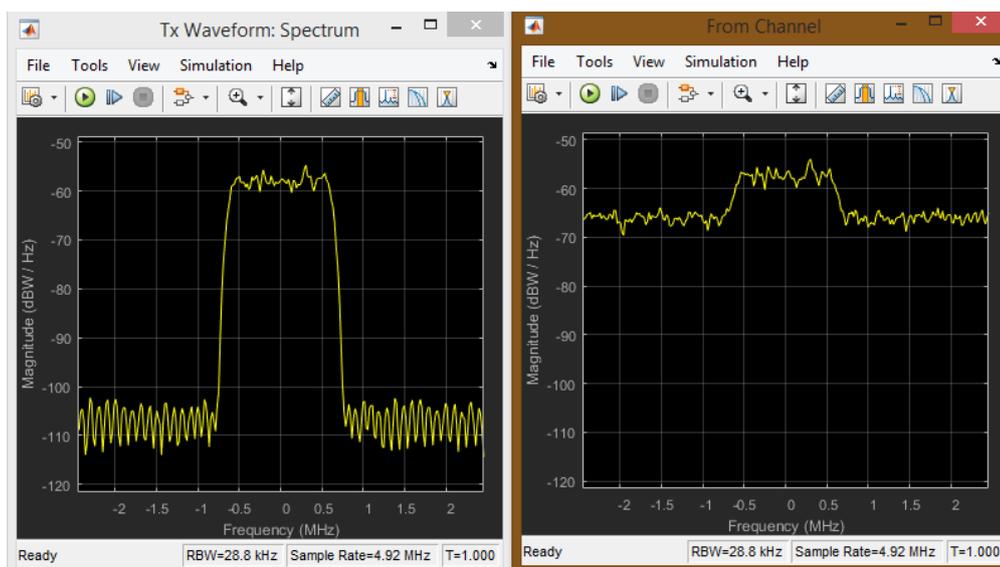


Рис. 5.34. Спектр сигнала до и после канала при отношении сигнал/шум 5 дБ

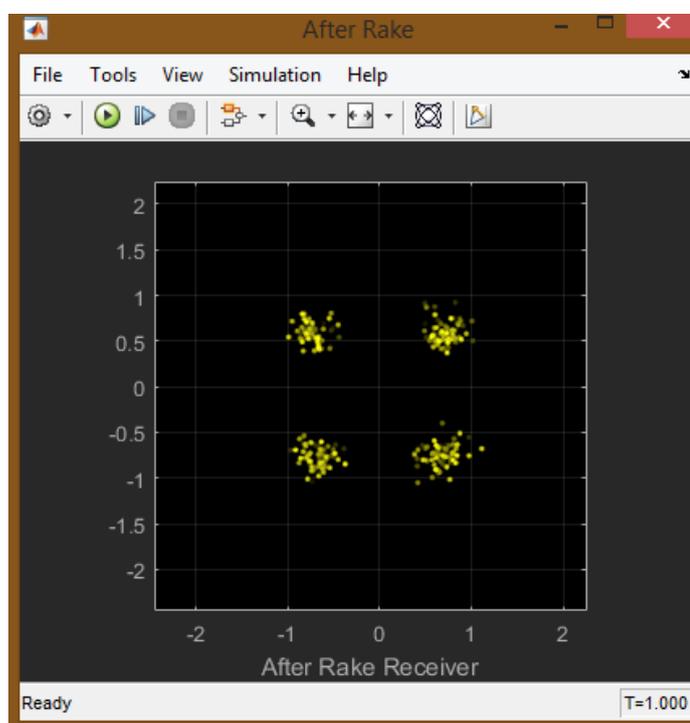


Рис. 5.35. Сигнал, декодированный мобильной станцией, на I-Q диаграмме.

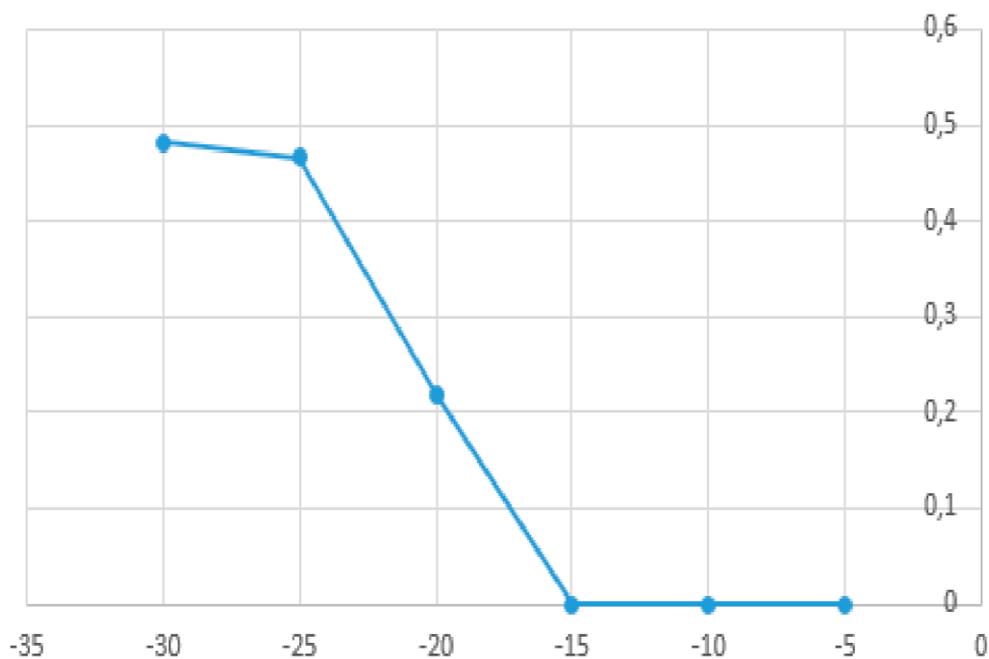


Рис. 5.36. Зависимость BER от SNR в канале с шумами.

Таблица 5.1. Зависимость BER от SNR в канале с шумами.

SNR	-30	-25	-20	-15	-10	-5
BER	0,4814	0,4662	0,2186	0	0	0

В блоке Model Parameters во вкладке Channel Settings выберем Channel Model: Multipath Fading Channel.

И установим следующие параметры

Maximum Doppler Frequency shift (in Hz):

450

Multipath Profile - Delay Vector (s):

[0 260e-9 521e-9 781e-9]

Multipath Profile - Gain Vector (dB):

[0 -3 -6 -9]

Рис. 5.37. Заданные параметры канала с многолучевым распространением.

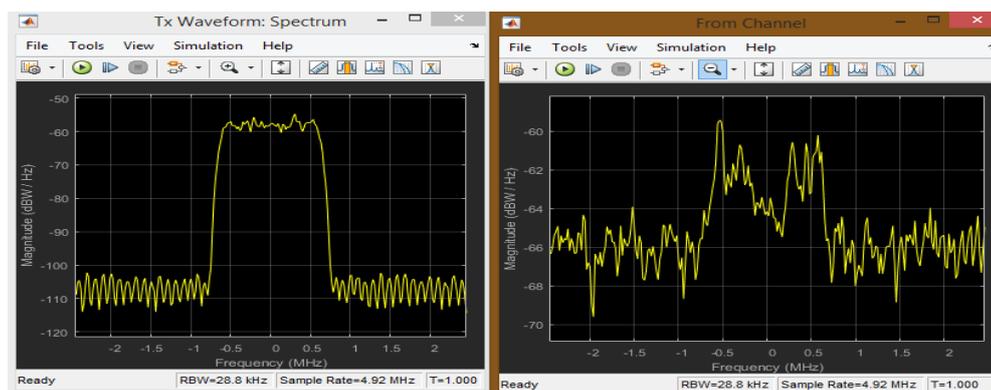


Рис. 5.38. Спектры сигнала до и после канала при отношении сигнал/шум 5 дБ.

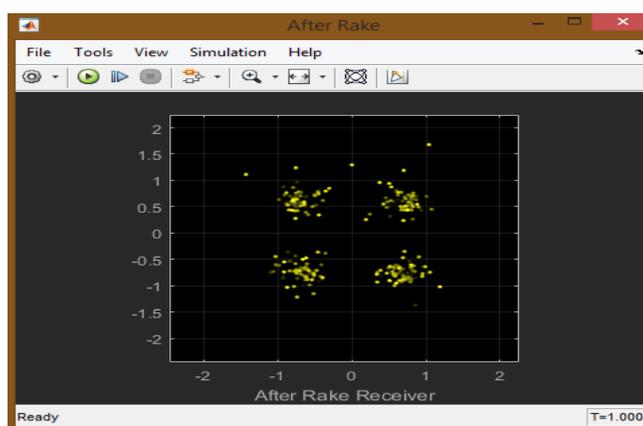


Рис. 5.39. Сигнал, декодированный мобильной станцией, на I-Q диаграмме.

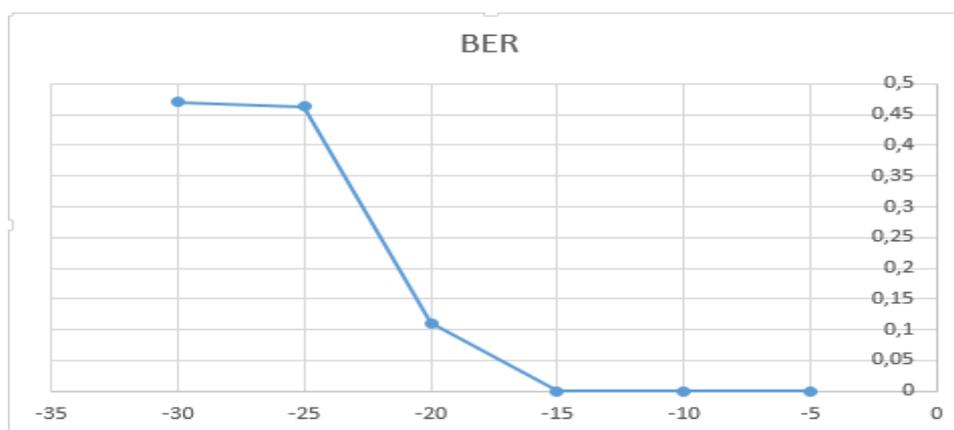


Рис. 5.40. Зависимость BER от SNR в канале с многолучевым распространением.

Таблица 5.2. Зависимость BER от SNR в канале с многолучевым распространением

SNR	-30	-25	-20	-15	-10	-5
BER	0,4708	0,4637	0,1105	0	0	0

В блоке Model Parameters во вкладке Channel Settings выберем Channel Model: Multipath Fading Channel.

И установим следующие параметры

Maximum Doppler Frequency shift (in Hz):

600

Multipath Profile - Delay Vector (s):

[0 280e-9 541e-9 801e-9]

Multipath Profile - Gain Vector (dB):

[0 -4 -7 -10]

Рис. 5.41. Заданные параметры канала с многолучевым распространением

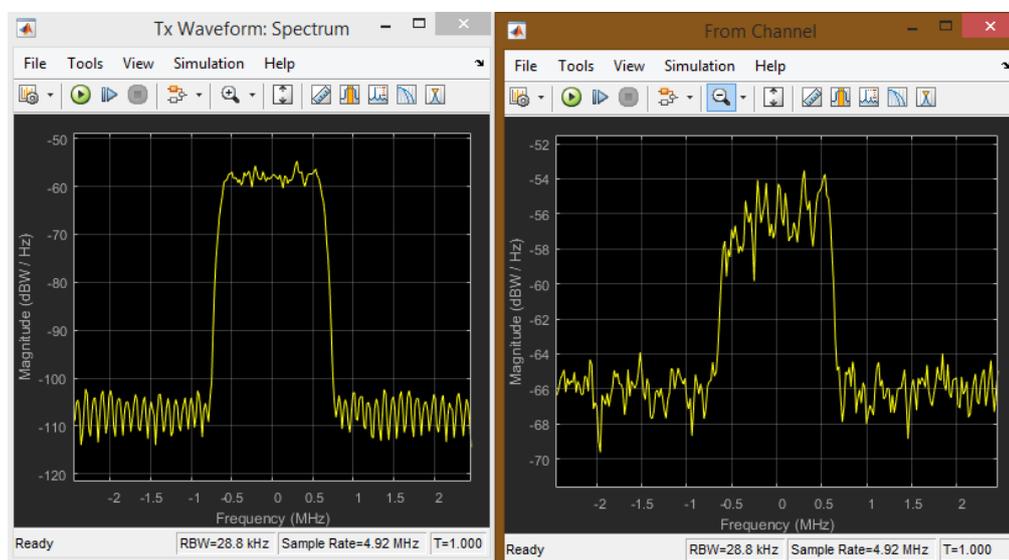


Рис. 5.42. Спектры сигнала до и после канала при отношении сигнал/шум 5 дБ

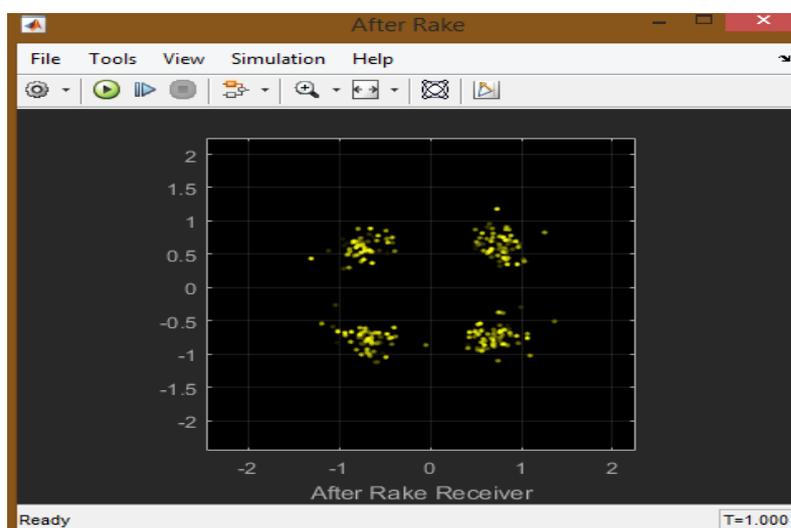


Рис. 5.43. Сигнал, декодированный мобильной станцией, на I-Q диаграмме

Таблица 5.3. Зависимость BER от SNR в канале с многолучевым распространением

SNR	-30	-25	-20	-15	-10	-5
BER	0,5007	0,4657	0,1532	0	0	0

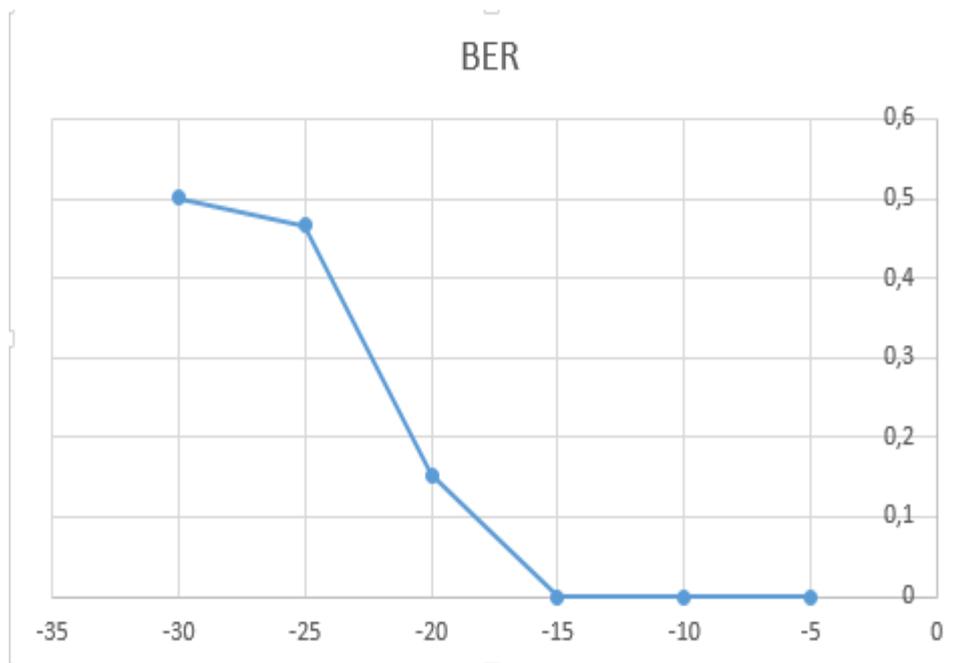


Рис. 5.44. Зависимость BER от SNR в канале с многолучевым распространением

Таким образом, в разделе было сделано:

1. Проведен аналитический обзор существующих методов и средств систем мобильной связи с кодовым разделением канала CDMA;
2. Разработана структурная схема DownLink канала CDMA2000 и приведена в приложении Б;
3. Приведена модель DownLink канала CDMA2000 реализованная в MATLAB R2015b;
4. Приведено исследование данной модели, а также методика проведения исследования, представленная в приложении А. Данную методику можно использовать для проведения учебных лабораторных работ.

На основе проведенного исследования можно сделать следующие выводы:

1. Система мобильной связи CDMA2000 обладает рядом преимуществ: возможность декодировать сигналы при отношении сигнал/шум меньше единицы, т.е. уровень передаваемого сигнала ниже уровня шума, что делает сигнал скрытным, а значит более защищенным.

2. Формируемый сигнал возможно принять и декодировать без ошибок даже при наличии многолучевости, однако при большом Доплеровском отклонении частоты и больших задержках, например, 1МГц и 1 мкс ошибки будут даже при высоком отношении сигнал/шум, например, 40 дБ. Но такие плохие характеристики канала довольно редки.

3. Для большей защищенности в аппаратуре стандарта CDMA длинный код формируется в результате нескольких последовательных логических операций с псевдослучайной двоичной последовательностью, генерируемой в 42-разрядном регистре сдвига, и двоичной 32-битовой маской, которая определяется индивидуально для каждого абонента. Такой регистр сдвига применяется во всех базовых станциях этого стандарта для обеспечения режима синхронизации всей сети. Длина М-последовательности при этом составляет 4 398 046 511 103 бит и если ее элементы формируются с тактовой частотой, например, 450 МГц, то период повторения будет составлять  $9773,44 \text{ с} = 2 \text{ ч } 43 \text{ мин}$ . Это значит, что если даже удастся засинхронизировать приемник в случае несанкционированного перехвата, то чтобы определить структуру сигнала-носителя необходимо вести наблюдение в течение почти 3-х часов, а с применением индивидуальной 32-битовой маски "подслушивание" практически исключено.

Таблица 5.4. Характеристики CDMA2000

Характеристика	Значение
Базовая скорость передачи данных в канале	9.6 кбит/с
Длительность пакетов, на которые разбивается базовый поток	20 мс
Цифровая модуляция DownLink	QPSK
Цифровая модуляция UpLink	OQPSK
Размер матрицы Адамара	64x64
Разрядность регистра сдвига для формирования длинного кода	42
Длина М-последовательности длинного кода	4 398 046 511 103

Количество бит в индивидуальной маске пользователя	32
Разрядность регистра сдвига для формирования короткого кода	15
Длина M-последовательности короткого кода	32768
Частота среза КИХ-фильтра	615 кГц

Методика проведения измерений работы:

1. Запустить MATLAB R2015b от имени администратора;
2. В командной строке ввести команду «cdma2000SimulinkExample»;
3. Два раза кликнуть левой кнопкой мыши по блоку Model Parameters;
4. Во вкладке Channel Settings выбрать Channel Model: No Channel;
5. Два раза кликнуть левой кнопкой мыши по блоку Open Scopes;
6. Запустить моделирование;
7. После отображения всех графиков сохранить полученные данные и убедиться, что спектр сигнала, до и после канала, не изменился;
8. Не закрывая окна с графиками два раза кликнуть левой кнопкой мыши по блоку Model Parameters;
9. Во вкладке Channel Settings выбрать Channel Model: AWGN Channel и изменяя значение отношения сигнал/шум построить зависимость BER от SNR, и сохранить полученные диаграммы хотя бы для одного измерения;
10. Не закрывая окна с графиками два раза кликнуть левой кнопкой мыши по блоку Model Parameters;
11. Во вкладке Channel Settings выбрать Channel Model: Multipath Fading Channel и изменяя значение отношения сигнал/шум построить зависимость BER от SNR, и сохранить полученные диаграммы хотя бы для одного измерения;
12. Не закрывая окна с графиками два раза кликнуть левой кнопкой мыши по блоку Model Parameters;
13. Во вкладке Channel Settings изменить параметры доплеровского отклонения частоты (Maximum Doppler Frequency shift), вектора задержки (Multipath Profile – Delay Vector), вектора усиления (Multipath Profile – Gain Vector) и повторить пункт 11. Длины векторов задержки и усиления должны совпадать!

### **5.3. Имитационное моделирование системы мобильной связи стандарта IEEE 802.11 (WiFi) на базе MATLAB 2015 [21]**

На современном этапе развития сетевых технологий, технология беспроводных сетей Wi-Fi является наиболее удобной в условиях, требующих мобильность, простоту установки и использования. Как правило, технология Wi-Fi используется для организации беспроводных локальных компьютерных сетей, а также создания так называемых горячих точек высокоскоростного доступа в Интернет.

Беспроводные сети обладают, по сравнению с традиционными проводными сетями, немалыми преимуществами, главным из которых, конечно же, является:

- Простота развёртывания;
- Гибкость архитектуры сети, когда обеспечивается возможность динамического изменения топологии сети при подключении, передвижении и отключении мобильных пользователей без значительных потерь времени;
- Быстрота проектирования и реализации, что критично при жестких требованиях к времени построения сети;

В то же время беспроводные сети на современном этапе их развития не лишены серьёзных недостатков. Прежде всего, это зависимость скорости соединения и радиуса действия от наличия преград и от расстояния между приёмником и передатчиком. Один из способов увеличения радиуса действия беспроводной сети заключается в создании распределённой сети на основе нескольких точек беспроводного доступа. При создании таких сетей появляется возможность превратить здание в единую беспроводную зону и увеличить скорость соединения вне зависимости от количества стен (преград). Аналогично решается и проблема масштабируемости сети, а использование внешних направленных антенн позволяет эффективно решать проблему препятствий, ограничивающих сигнал.

**В соответствии с техническим заданием основными задачами данной работы являлись:**

1. Аналитический обзор существующих методов и средств;
2. Разработка структурной схемы программного комплекса;
3. Разработка алгоритма программы;
4. Разработка программного интерфейса для исследования характеристик и визуализации основных преобразований;
5. Разработка методики и проведение исследования основных технических характеристик, анализ результатов исследования.

Полученная в результате разработка позволяет исследовать беспроводные сети на базе стандарта 802.11b.

## **История развития**

В 1990 г. Комитет по стандартам IEEE 802 (Institute of Electrical and Electronic Engineers) сформировал рабочую группу по стандартам для беспроводных локальных сетей 802.11. Это группа занялась разработкой всеобщего стандарта для радиооборудования и сетей, работающих на частоте 2.4 ГГц со скоростями 1 и 2 Мбит/с. Работа по созданию стандарта были завершены через семь лет, и в июне 1997 г. была ратифицирована первая спецификация 802.11 [1].

Стандарт IEEE 802.11 стал первым стандартом для продуктов WLAN от независимой международной организации. Однако к моменту выхода стандарта в свет первоначально заложенная в нем скорость передачи данных оказалась недостаточной. Это послужило причиной последующих доработок, поэтому сегодня можно говорить о группе стандартов.

## **Методы построения радиосигнала в WiFi-сетях**

В настоящее время при разработке аппаратуры для беспроводных сетей используются два метода построения сигнала:

1. С непосредственной модуляцией несущей частоты (Direct-Sequence Spread Spectrum – DSSS).

Информационный сигнал домножается на псевдослучайный код (Pncode – Pseudo Random Noise Code). Полученный результат используют для модуляции несущей. В приемнике полученный сигнал умножают на тот же код и выделяют полезный сигнал.

Основной проблемой, возникающей при использовании метода прямой последовательности, является эффект близко расположенного передатчика, т.е. уровень сигнала мешающего передатчика гораздо выше уровня нужного передатчика, что может привести к потере связи.

2. Со скачкообразной перестройкой частоты (Frequency-Hopping Spread Spectrum – FHSS).

Частота несущей изменяется согласно уникальной последовательности. Для реализации этого метода необходим скоростной синтезатор частот.

Недостаток: сложность получения высокого значения базы сигнала, что необходимо для увеличения числа пользователей, помехоустойчивости, повышения конфиденциальности.

Достоинство: меньшая подверженность эффекту близкого передатчика.

Оба метода основаны на принципе приемопередачи с «расширенным спектром», который обеспечивает защиту от помех и конфиденциальность передаваемой информации. Обычно при выборе сетевого продукта учитывают следующие факторы: скорость передачи данных, дальность устойчивой связи, соответствие стандартам, эксплуатационные характеристики и стоимость. Выбор типа аппаратуры для беспроводной сети определяется как условиями

эксплуатации, так и стоимостью изделия. Следует отметить, что устройства, работающие по методу FHSS, можно получить миниатюрный и недорогой адаптер для портативного ПК [2].

### **Описание стандарта**

Из всех существующих стандартов беспроводной передачи данных IEEE 802.11 на практике чаще всего используются всего три стандарта, определенные Инженерным институтом электротехники и радиоэлектроники (IEEE): 802.11b, 802.11a и 802.11g.

В стандарте IEEE 802.11b благодаря высокой скорости передачи данных (до 11 Мбит/с), практически эквивалентной пропускной способности обычных проводных локальных сетей Ethernet, а также ориентации на диапазон 2,4 ГГц, этот стандарт завоевал наибольшую популярность у производителей оборудования для беспроводных сетей.

Поскольку оборудование, работающее на максимальной скорости 11 Мбит/с, имеет меньший радиус действия, чем на более низких скоростях, стандартом 802.11b предусмотрено автоматическое снижение скорости при ухудшении качества сигнала.

Стандарт IEEE 802.11a имеет большую ширину полосы из семейства стандартов 802.11 при скорости передачи данных до 54 Мбит/с.

В отличие от базового стандарта, ориентированного на область частот 2,4 ГГц, спецификациями 802.11a предусмотрена работа в диапазоне 5 ГГц. В качестве метода модуляции сигнала выбрано ортогональное частотное мультиплексирование (OFDM).

К недостаткам 802.11a относятся более высокая потребляемая мощность радиопередатчиков для частот 5 ГГц, а также меньший радиус действия.

Стандарт IEEE 802.11g является логическим развитием 802.11b и предполагает передачу данных в том же частотном диапазоне. Кроме того, стандарт 802.11g полностью совместим с 802.11b, то есть любое устройство 802.11g должно поддерживать работу с устройствами 802.11b. Максимальная скорость передачи в стандарте 802.11g составляет 54 Мбит/с, поэтому на сегодня это наиболее перспективный стандарт беспроводной связи.

При разработке стандарта 802.11g рассматривались две отчасти конкурирующие технологии: метод ортогонального частотного разделения OFDM и метод двоичного пакетного сверточного кодирования PBCC, опционально реализованный в стандарте 802.11b. В результате стандарт 802.11g содержит компромиссное решение: в качестве базовых применяются технологии OFDM и ССК, а опционально предусмотрено использование технологии PBCC.

### **Физические уровни стандарта**

Основное назначение физических уровней стандарта 802.11 - обеспечить механизмы беспроводной передачи для подуровня MAC, а также поддерживать выполнение вторичных функций, таких как оценка состояния беспроводной среды и сообщение о нем подуровню MAC. Уровни MAC и PHY разрабатывались так, чтобы они были независимыми. Именно независимость между MAC и подуровнем PHY и позволила использовать дополнительные высокоскоростные физические уровни, описанные в стандартах 802.11b, 802.11a и 802.11g.

Каждый из физических уровней стандарта 802.11 имеет два подуровня:

- Physical Layer Convergence Procedure (PLCP). Процедура определения состояния физического уровня.
- Physical Medium Dependent (PMD). Подуровень физического уровня, зависящий от среды передачи.

На [рис.1](#) показано, как эти подуровни соотносятся между собой и с вышестоящими уровнями в модели взаимодействия открытых систем (Open System Interconnection - OSI).

Подуровень PLCP по существу является уровнем обеспечения взаимодействия, на котором осуществляется перемещение элементов данных протокола MAC (MAC Protocol Data Units - MPDU) между MAC-станциями с использованием подуровня PMD, на котором реализуется тот или иной метод передачи и приема данных через беспроводную среду. Подуровни PLCP и PMD отличаются для разных вариантов стандарта 802.11.

Перед тем как приступить к изучению физических уровней, рассмотрим одну из составляющих физического уровня, до сих пор не упомянутую, а именно - скремблирование.

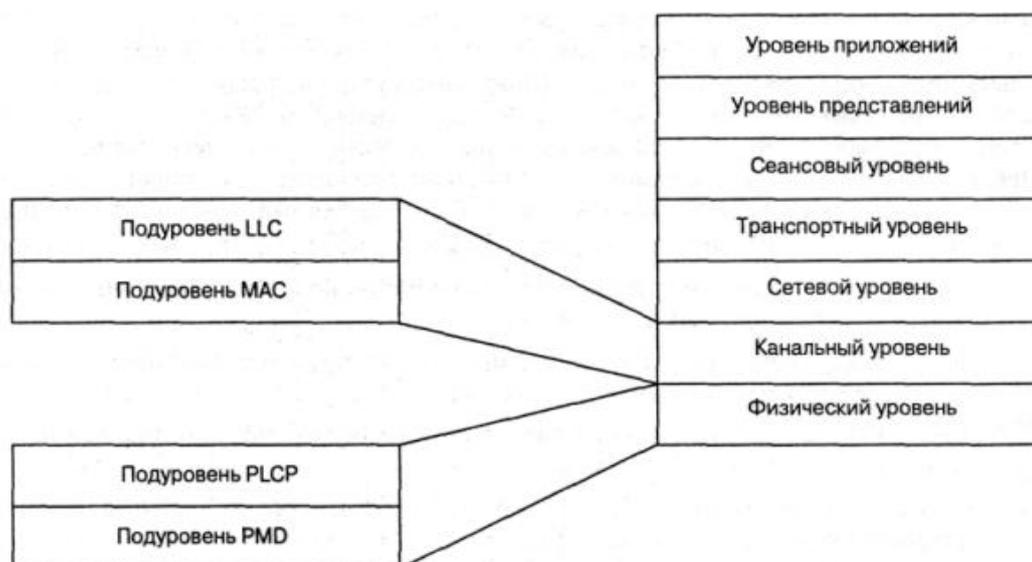


Рис. 5.45. Подуровни уровня PHY

Одна из особенностей, лежащих в основе современных передатчиков, благодаря которой данные можно передавать с высокой скоростью, - это предположение о том, что данные, которые предлагаются для передачи, поступают, с точки зрения передатчика, случайным образом. Без этого предположения многие преимущества, получаемые за счет применения остальных составляющих физического уровня, остались бы нереализованными.

Однако бывает, что принимаемые данные не вполне случайны и на самом деле могут содержать повторяющиеся наборы и длинные последовательности нулей и единиц.

Скрэмблирование (перестановка элементов) - это метод, посредством которого принимаемые данные делаются более похожими на случайные; достигается это путем перестановки битов последовательности таким образом, чтобы превратить ее из структурированной в похожую на случайную. Эту процедуру иногда называют "отбеливанием потока данных". Дескрэмблер приемника затем выполняет обратное преобразование этой случайной последовательности с целью получения исходной структурированной последовательности. Большинство способов скрэмблирования относится к числу самосинхронизирующихся; это означает, что дескрэмблер способен самостоятельно синхронизироваться со скрэмблером.

### **IEEE 802.11**

Исходный стандарт 802.11 определяет три метода передачи на физическом уровне:

- Передача в диапазоне инфракрасных волн.
- Технология расширения спектра путем скачкообразной перестройки частоты (FHSS) в диапазоне 2,4 ГГц.
- Технология широкополосной модуляции с расширением спектра методом прямой последовательности (DSSS) в диапазоне 2,4 ГГц.

#### **Передача в диапазоне инфракрасных волн**

Средой передачи являются инфракрасные волны диапазона 850 нм, которые генерируются либо полупроводниковым лазерным диодом, либо светодиодом (LED). Так как инфракрасные волны не проникают через стены, область покрытия LAN ограничивается зоной прямой видимости. Стандарт предусматривает три варианта распространения излучения: ненаправленную антенну, отражение от потолка и фокусное направленное излучение. В первом случае узкий луч рассеивается с помощью системы линз. Фокусное направленное излучение предназначено для организации двухточечной связи, например между двумя зданиями.

#### **Беспроводные локальные сети со скачкообразной перестройкой частоты (FHSS)**

Беспроводные локальные сети FHSS поддерживают скорости передачи 1 и 2 Мбит/с. Устройства FHSS делят предназначенную для их работы полосу частот от 2,402 до 2,480 ГГц

на 79 неперекрывающихся каналов (это справедливо для Северной Америки и большей части Европы). Ширина каждого из 79 каналов составляет 1 МГц, поэтому беспроводные локальные сети FHSS используют относительно высокую скорость передачи символов - 1 МГц - и намного меньшую скорость перестройки с канала на канал.

Последовательность перестройки частоты должна иметь следующие параметры: частота перескоков не менее 2,5 раз в секунду как минимум между шестью (6 МГц) каналами. Чтобы минимизировать число коллизий между перекрывающимися зонами покрытия, возможные последовательности перескоков должны быть разбиты на три набора последовательностей, длина которых для Северной Америки и большей части Европы составляет 26. В [таблице 1](#) представлены схемы скачкообразной перестройки частоты, обеспечивающие минимальное перекрытие.

По сути, схема скачкообразной перестройки частоты обеспечивает неторопливый переход с одного возможного канала на другой таким образом, что после каждого скачка покрывается полоса частот, равная как минимум 6 МГц, благодаря чему в многосотовых сетях минимизируется возможность возникновения коллизий.

Набор	Схема скачкообразной перестройки частоты
1	{0,3,6,9,12,15,18,21,24,27,30,33,36,39,42,45,48,51,54,57,60,63,66,69,72,75}
2	{1,4,7,10,13,16,19,22,25,28,31,34,37,40,43,46,49,52,55,58,61,64,67,70,73,76}
3	{2,5,8,11,14,17,20,23,26,29,32,35,38,41,44,47,50,53,56,59,62,65,68,71,74,77}

После того как уровень MAC пропускает MAC-фрейм, который в локальных беспроводных сетях FHSS называется также служебным элементом данных PLCP, или PSDU (PLCP Service Data Unit), подуровень PLCP добавляет два поля в начало фрейма, чтобы сформировать таким образом фрейм PPDU (PPDU - элемент данных протокола PLCP). На [рис.2](#) представлен формат фрейма FHSS подуровня PLCP.

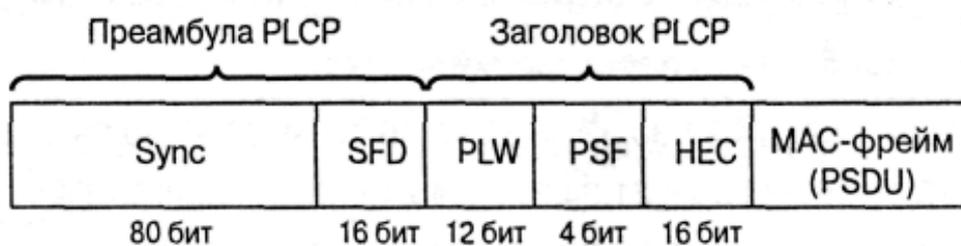


Рис. 5.46. Формат фрейма FHSS подуровня PLCP

Преамбула PLCP состоит из двух подполей:

- Подполе Sync размером 80 бит. Строка, состоящая из чередующихся 0 и 1, начинается с 0. Приемная станция использует это поле, чтобы принять решение о выборе антенны при наличии такой возможности, откорректировать уход частоты (frequency offset) и синхронизировать распределение пакетов (packet timing).

- Подполе флага начала фрейма (Start of Frame Delimiter, SFD) размером 16 бит. Состоит из специфической строки (0000 1100 1011 1101, крайний слева бит первый) в обеспечение синхронизации фреймов (frame timing) для приемной станции.

Заголовок фрейма PLCP состоит из трех подполей:

- Слово длины служебного элемента данных PLCP (PSDU), PSDU Length Word (PLW) размером 12 бит. Указывает размер фрейма MAC (PSDU) в октетах.
- Сигнальное поле PLCP (Signaling Field PLCP - PSF) размером 4 бит. Указывает скорость передачи данных конкретного фрейма.
- НЕС (Header Error Check). Контрольная сумма фрейма.

Служебный элемент данных PLCP (PSDU) проходит через операцию скремблирования с целью отбеливания (рандомизации) последовательности входных битов. Получившийся в результате PSDU представлен на [рис.3](#). Заполняющие символы вставляются между всеми 32-символьными блоками. Эти заполняющие символы устраняют любые систематические отклонения в данных, например, когда единиц больше, чем нулей, или наоборот, которые могли бы привести к нежелательным эффектам при дальнейшей обработке.

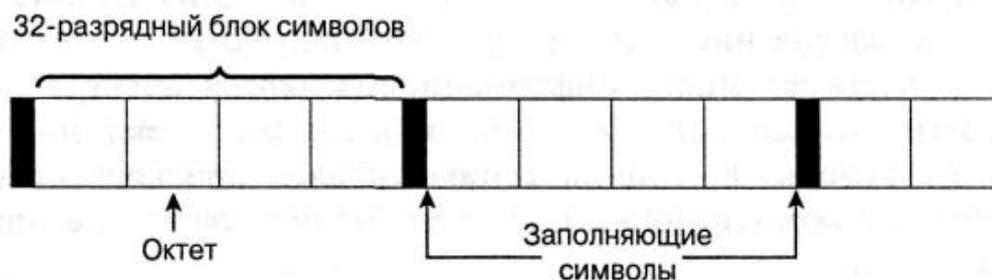


Рис. 5.47. Скремблированный PSDU в технологии FHSS

Подуровень PLCP преобразует фрейм в поток битов и передает его на подуровень PMD. Подуровень PMD технологии FHSS модулирует поток данных с использованием модуляции, основанной на гауссовой частотной модуляции (Gaussian Frequency Shift Keying - GFSK).

### **Беспроводные локальные сети, использующие широкополосную модуляцию DSSS с расширением спектра методом прямой последовательности**

В спецификации стандарта 802.11 оговорено использование и другого физического уровня - на основе технологии широкополосной модуляции с расширением спектра методом

прямой последовательности (DSSS). Как было указано в стандарте 802.11 разработки 1997 года, технология DSSS поддерживает скорости передачи 1 и 2 Мбит/с.

Аналогично подуровню PLCP, используемому в технологии FHSS, подуровень PLCP технологии DSSS стандарта 802.11 добавляет два поля во фрейм MAC, чтобы сформировать PPDU: преамбулу PLCP и заголовок PLCP. Формат фрейма представлен на рис.5.4.

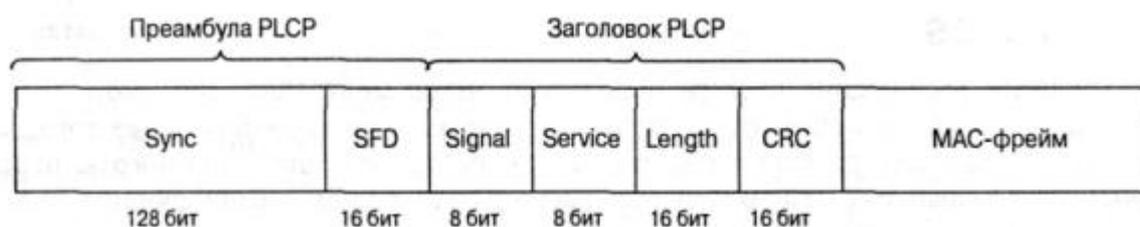


Рис. 5.48. Формат фрейма DSSS подуровня PLCP

Преамбула PLCP состоит из двух подполей:

- Подполе Sync шириной 128 бит, представляющее собой строку, состоящую из единиц. Задача этого подполя - обеспечить синхронизацию для приемной станции.
- Подполе SFD шириной 16 бит; в нем содержится специфичная строка 0xF3A0; его задача - обеспечить тайминг (timing) для приемной станции.

Заголовок PLCP состоит из четырех подполей:

- Подполе Signal шириной 8 бит, указывающее тип модуляции и скорость передачи для данного фрейма.
- Подполе Service шириной 8 бит зарезервировано. Это означает, что во время разработки спецификации стандарта оно осталось неопределенным; предполагается, что оно пригодится в будущих модификациях стандарта.
- Подполе Length шириной 16 бит, указывающее количество микросекунд (из диапазона 16-216), необходимое для передачи части MAC-фрейма.
- Подполе CRC. 16-битная контрольная сумма.

Подуровень PLCP преобразует фрейм в поток битов и передает данные на подуровень PMD. Весь PPDU проходит через процесс скремблирования с целью рандомизации данных.

Скремблированная преамбула PLCP всегда передается со скоростью 1 Мбит/с, в то время как скремблированный фрейм MPDU передается со скоростью, указанной в подполе Signal. Подуровень PMD модулирует отбеленный поток битов, используя следующие методы модуляции:

- Двоичная относительная фазовая модуляция (Differential Binary Phase Shift Keying - DBPSK) для скорости передачи 1 Мбит/с.

- Квадратурная относительная фазовая модуляция (Differential Quadrature Phase Shift Key - DQPSK) для скорости передачи 2 Мбит/с.

### IEEE 802.11b

На физическом уровне к MAC-кадрам (MPDU) добавляется заголовок физического уровня, состоящий из преамбулы и собственно PLCP-заголовка (рис.5.5).

Преамбула содержит стартовую синхропоследовательность (SYNC) для настройки приемника и 16-битный код начала кадра (SFD) - число F3A016. PLCP-заголовок включает поля SIGNAL (информация о скорости и типе модуляции), SERVICE (дополнительная информация, в том числе о применении высокоскоростных расширений и PBCC-модуляции) и LENGTH (время в микросекундах, необходимое для передачи следующей за заголовком части кадра). Все три поля заголовка защищены 16-битной контрольной суммой CRC.

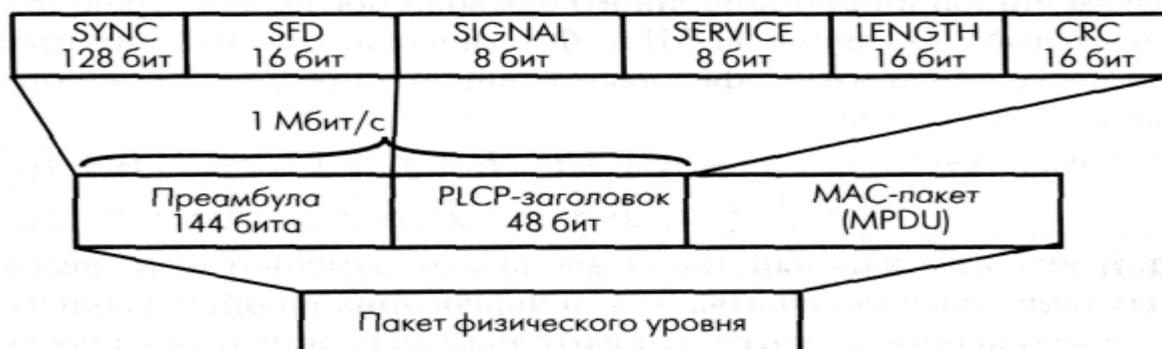


Рис. 5.49. Структура кадров сети IEEE 802.11b физического уровня

В стандарте IEEE 802.11b предусмотрено два типа заголовков: длинный и короткий (51).



Рис. 5.50. Короткий заголовок кадров сети 802.11b

Они отличаются длиной синхропоследовательности (128 и 56 бит), способом ее генерации, а также тем, что символ начала кадра в коротком заголовке передается в обратном порядке. Кроме того, если все поля длинного заголовка передаются со скоростью 1 Мбит/с, то при коротком заголовке преамбула транслируется на скорости 1 Мбит/с, другие поля заголовка - со скоростью 2 Мбит/с. Остальную часть кадра можно передавать на любой

из допустимых стандартом скоростей передачи, указанных в полях SIGNAL и SERVICE. Короткие заголовки физического уровня предусмотрены спецификацией IEEE 802.11b для увеличения пропускной способности сети.

Из описания процедур связи сети IEEE 802.11 видно, что "накладные расходы" в этом стандарте выше, чем в проводной сети Ethernet. Поэтому крайне важно обеспечить высокую скорость передачи данных в канале. Повысить пропускную способность канала с заданной шириной полосы частот можно, разрабатывая и применяя новые методы модуляции. По этому пути пошла группа разработчиков IEEE 802.11b.

Напомним, что изначально стандарт IEEE 802.11 предусматривал работу в режиме DSSS с использованием так называемой Баркеровской последовательности (Barker) длиной 11 бит:  $B1 = (10110111000)$ . Каждый информационный бит замещается своим произведением по модулю 2 (операция "исключающее ИЛИ") с данной последовательностью, т. е. каждая информационная единица заменяется на B1, каждый ноль - на инверсию B1. В результате бит заменяется последовательностью 11 чипов. Далее сигнал кодируется посредством дифференциальной двух- или четырехпозиционной фазовой модуляции (DBPSK или DQPSK, один или два чипа на символ соответственно). При частоте модуляции несущей 11 МГц общая скорость составляет в зависимости от типа модуляции 1 и 2 Мбит/с.

Стандарт IEEE 802.11b дополнительно предусматривает скорости передачи 11 и 5,5 Мбит/с. Для этого используется так называемая ССК-модуляция (Complementary Code Keying - кодирование комплементарным кодом).

Хотя механизм расширения спектра, используемый для получения скоростей 5,5 и 11 Мбит/с с применением ССК, относится к методам, которые применяются для скоростей 1 и 2 Мбит/с, он по-своему уникален. В обоих случаях применяется метод расширения, но при использовании модуляции ССК расширяющий код представляет собой код из 8 комплексных чипов, в то время как при работе со скоростями 1 и 2 Мбит/с применяется 11-разрядный код. 8-чиповый код определяется или 4, или 8 битами - в зависимости от скорости передачи данных. Скорость передачи чипов составляет 11 Мчип/с, т.е. при 8 комплексных чипах на символ и 4 или 8 битов на символ можно добиться скорости передачи данных 5,5 и 11 Мбит/с.

Для того чтобы передавать данные со скоростью 5,5 Мбит/с, нужно сгруппировать скремблированный поток битов в символы по 4 бита ( $b_0, b_1, b_2$  и  $b_3$ ). Последние два бита ( $b_2$  и  $b_3$ ) используются для определения 8 последовательностей комплексных чипов, как показано в таблице 6, где  $\{c_1, c_2, c_3, c_4, c_5, c_6, c_7, c_8\}$  представляют чипы

последовательности. В таблицн  $\underline{b}_j$  представляет мнимое число, корень квадратный из  $-1$ , и откладывается по мнимой, или квадратурной, оси комплексной плоскости.

(b2, b3)	1	2	3	4	5	6	7	C8
00				1			1	1
01	j	1	j				j	1
10	j		j	1	j			1
11		1			j			1

Теперь, имея последовательность чипов, определенную битами (b2, b3), можно использовать первые два бита (b0, b1) для определения поворота фазы, осуществляемого при модуляции по методу DQPSK, который будет применен к последовательности (таблица 3). Вы должны также пронумеровать каждый 4-битовый символ PSDU, начиная с 0, чтобы можно было определить, преобразуете вы четный либо нечетный символ в соответствии с этой таблицей. Следует помнить, что речь идет об использовании DQPSK, а не QPSK, и поэтому представленные в таблице изменения фазы отсчитываются по отношению к предыдущему символу или, в случае первого символа PSDU, по отношению к последнему символу предыдущего DQPSK-символа, передаваемого со скоростью 2 Мбит/с.

(b0, b1)	Изменение фазы четных символов	Изменение фазы нечетных символов
00	0	$\pi$
01	$\pi/2$	$-\pi/2$
11	$\pi$	0
10	$-\pi/2$	$\pi/2$

Это вращение фазы применяется по отношению к 8 комплексным чипам символа, затем осуществляется модуляция на подходящей несущей частоте.

Чтобы передавать данные со скоростью 11 Мбит/с, скремблированная последовательность битов PSDU разбивается на группы по 8 символов. Последние 6 битов выбирают одну последовательность, состоящую из 8 комплексных чипов, из числа 64 возможных последовательностей, почти так же, как использовались биты (b2, b3) для выбора одной из четырех возможных последовательностей. Биты (b0,b1) используются таким же образом, как при модуляции ССК на скорости 5,5 Мбит/с для вращения фазы последовательности и дальнейшей модуляции на подходящей несущей частоте.

В чем достоинство ССК-модуляции? Дело в том, что чипы символа определяются на основе последовательностей Уолша-Адамара. Последовательности Уолша-Адамара хорошо изучены, обладают отличными автокорреляционными свойствами. Что немаловажно, каждая такая последовательность мало коррелирует сама с собой при фазовом сдвиге - очень полезное свойство при борьбе с переотраженными сигналами. Нетрудно заметить, что теоретическое операционное усиление ССК-модуляции - 3 дБ (в два раза), поскольку без кодирования QPSK-модулированный с частотой 11 Мбит/с сигнал может транслировать 22 Мбит/с. Как видно, ССК-модуляция представляет собой вид блочного кода, а потому достаточно проста при аппаратной реализации. Совокупность этих свойств и обеспечила ССК место в стандарте IEEE 802.11b в качестве обязательного вида модуляции.

На практике важно не только операционное усиление. Существенную роль играет и равномерность распределения символов в фазовом пространстве - они должны как можно дальше отстоять друг от друга, чтобы минимизировать ошибки их детектирования. И с этой точки зрения ССК-модуляция не выглядит оптимальной, ее реальное операционное усиление не превышает 2 дБ. Поэтому изначально прорабатывался другой способ модуляции - пакетное бинарное сверточное кодирование PBCC (Packet Binary Convolutional Coding). Этот метод вошел в стандарт IEEE 802.11b как дополнительная (необязательная) опция. Механизм PBCC (5.51) позволяет добиваться в сетях IEEE 802.11b пропускной способности 5,5, 11 и 22 Мбит/с.

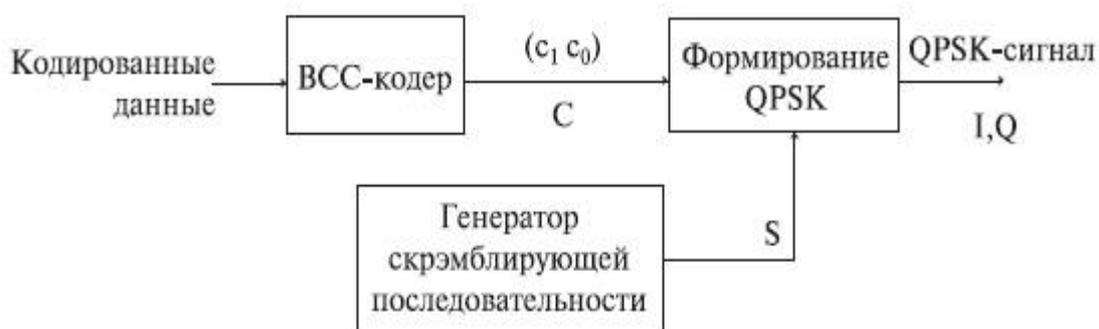


Рис. 5.51. Общая схема PBCC-модуляции

Как следует из названия, метод основан на сверточном кодировании. Для скоростей 5,5 и 11 Мбит/с поток информационных битов поступает в шестизрядный сдвиговый регистр с сумматорами (5.52). В начальный момент времени все триггеры сдвигового регистра инициализируют нулем. В результате каждый исходный бит  $d$  заменяется двумя битами кодовой последовательности ( $c_0$ ,  $c_1$ ). При скорости 11 Мбит/с  $c_0$  и  $c_1$  задают один символ четырехпозиционной QPSK-модуляции. Для скорости 5,5 Мбит/с используют двухпозиционную BPSK-модуляцию, последовательно передавая кодовые биты  $c_0$  и  $c_1$ . Если же нужна скорость 22 Мбит/с, схема кодирования усложняется (рис.5.9): три кодовых бита ( $c_0$ - $c_2$ ) определяют один символ в 8-позиционной 8-PSK-модуляции.

После формирования PSK-символов происходит скремблирование. В зависимости от сигнала  $s$  (5.51) символ остается без изменений ( $s = 0$ ), либо его фаза увеличивается на  $\pi/2$  ( $s = 1$ ). Значение  $s$  определяет 256-битовая циклически повторяющаяся последовательность  $S$ . Она формируется на основе начального вектора  $U = 338Bh$ , в котором равное число нулей и единиц.

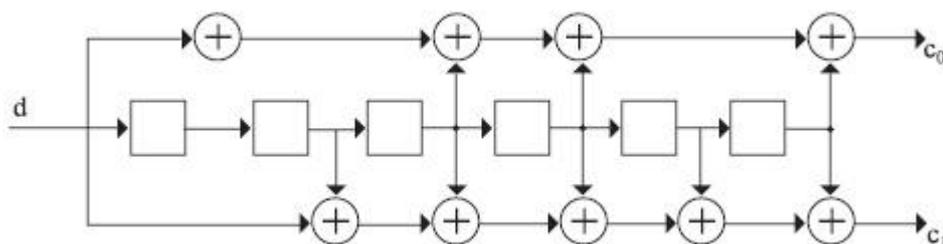


Рис. 5.52. Сверточное кодирование с двумя битами кодовой последовательности

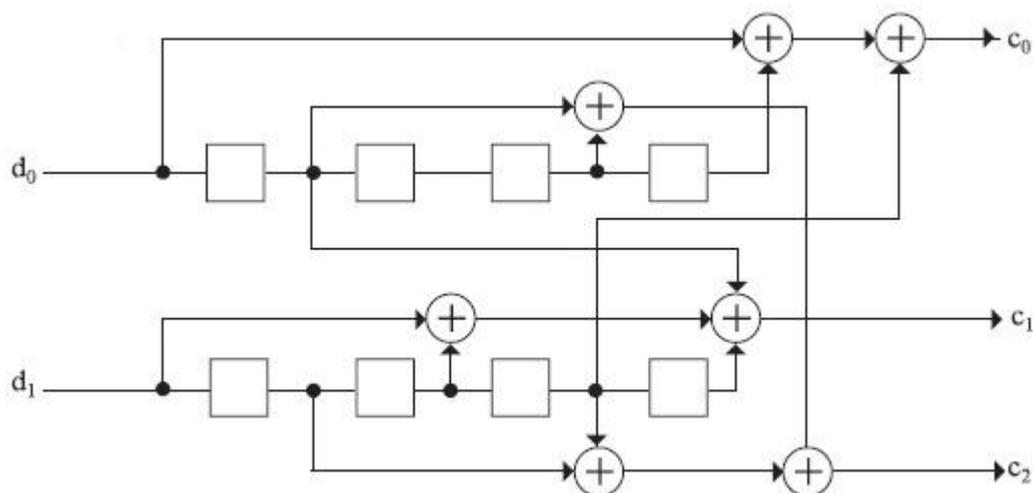


Рис. 5.53. Сверточное кодирование с тремя битами кодовой последовательности

У шестиразрядного сдвигового регистра, применяемого в РВСС для скоростей 11 и 5,5 Мбит/с, 64 возможных выходных состояния. Так что при модуляции РВСС информационные биты в фазовом пространстве оказываются гораздо дальше друг от друга, чем при ССК-модуляции. Поэтому РВСС и позволяет при одном и том же соотношении "сигнал-шум" и уровне ошибок вести передачу с большей скоростью, чем в случае ССК. Однако плата за более эффективное кодирование - сложность аппаратной реализации данного алгоритма.

### IEEE 802.11a

Стандарт IEEE 802.11a появился практически одновременно с IEEE 802.11b, в сентябре 1999 года. Эта спецификация была ориентирована на работу в диапазоне 5 ГГц и основана на принципиально ином, чем описано выше, механизме кодирования данных - на частотном мультиплексировании посредством ортогональных несущих (OFDM).

Стандарт 802.11a определяет характеристики оборудования, применяемого в офисных или городских условиях, когда распространение сигнала происходит по многолучевым каналам из-за множества отражений.

В IEEE 802.11a каждый кадр передается посредством 52 ортогональных несущих, каждая с шириной полосы порядка 300 КГц (20 МГц/64). Ширина одного канала - 20 МГц. Несущие модулируют посредством BPSK, QPSK, а также 16- и 64-позиционной квадратурной амплитудной модуляции (QAM). В совокупности с различными скоростями кодирования (1/2 и 3/4, для 64-QAM - 2/3 и 3/4) образуется набор скоростей передачи 6, 9, 12, 18, 24, 36, 48 и 54 Мбит/с. В [таблице 5.8](#) показано, как необходимая скорость передачи данных преобразуется в соответствующие параметры узлов передатчика OFDM.

Таблица 5.8. Параметры передатчика стандарта 802.11a

Скорость передачи данных (Мбит/с)	Модуляция	Скорость сверточного кодирования	Число канальных битов на поднесущую	Число канальных битов на символ	Число битов данных на символ OFDM
6	BPSK	1/2	1	48	24
9	BPSK	3/4	1	48	36
12	QPSK	1/2	2	96	48
18	QPSK	3/4	2	96	72

24	16-QAM	1/2	4	192	96
36	16-QAM	3/4	4	192	144
48	64-QAM	2/3	6	288	192
54	64-QAM	3/4	6	288	216

Из 52 несущих 48 предназначены для передачи информационных символов, остальные 4 - служебные. Структура заголовков физического уровня отличается от принятого в спецификации IEEE 802.11b, но незначительно (рис.5.54).

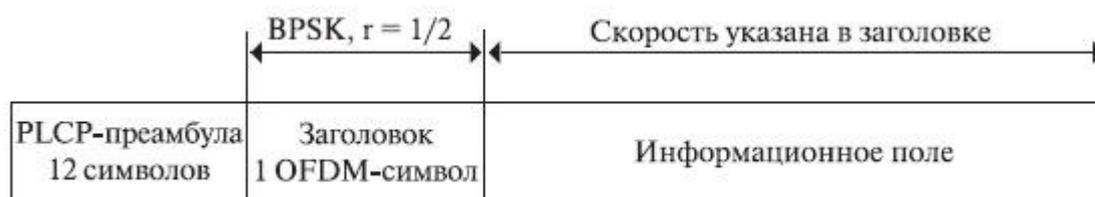


Рис. 5.54. Структура заголовка физического уровня стандарта IEEE 802.11a

Кадр включает преамбулу (12 символов синхропоследовательности), заголовок физического уровня (PLCP-заголовок) и собственно информационное поле, сформированное на MAC-уровне. В заголовке передается информация о скорости кодирования, типе модуляции и длине кадра. Преамбула и заголовок транслируются с минимально возможной скоростью (BPSK, скорость кодирования  $r = 1/2$ ), а информационное поле - с указанной в заголовке, как правило, максимальной, скоростью, в зависимости от условий обмена. OFDM-символы передаются через каждые 4 мкс, причем каждому символу длительностью 3,2 мкс предшествует защитный интервал 0,8 мкс (повторяющаяся часть символа). Последний необходим для борьбы с многолучевым распространением сигнала - отраженный и пришедший с задержкой символ попадет в защитный интервал и не повредит следующий символ.

Естественно, формирование/декодирование OFDM-символов происходит посредством быстрого преобразования Фурье (обратного/прямого, ОБПФ/БПФ). Функциональная схема трактов приема/передачи (рис. 5.54) достаточно стандартна для данного метода и включает сверточный кодер, механизм перемежения/перераспределения (защита от пакетных ошибок) и процессор ОБПФ. Фурье-процессор, собственно, и формирует суммарный сигнал, после чего к символу добавляется защитный интервал, окончательно формируется OFDM-символ и посредством квадратурного модулятора/конвертера переносится в заданную частотную область. При приеме все происходит в обратном порядке.

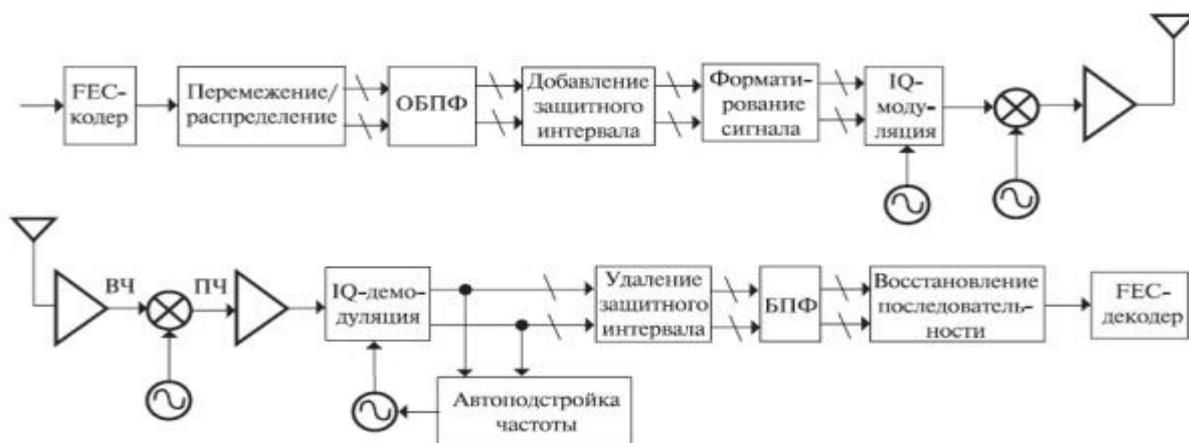


Рис. 5.55. Функциональная схема трактов приема/передачи стандарта IEEE 802.11a

### IEEE 802.11g

Стандарт IEEE 802.11g по сути представляет собой перенесение схемы модуляции OFDM, прекрасно зарекомендовавшей себя в 802.11a, из диапазона 5 ГГц в область 2,4 ГГц при сохранении функциональности устройств стандарта 802.11b. Это возможно, поскольку в стандартах 802.11 ширина одного канала в диапазонах 2,4 и 5 ГГц схожа - 22 МГц.

Одним из основных требований к спецификации 802.11g была обратная совместимость с устройствами 802.11b. Действительно, в стандарте 802.11b в качестве основного способа модуляции принята схема ССК (Complementary Code Keying), а в качестве дополнительной возможности допускается модуляция PBCC (Pocket Binary Convolutional Coding).

Разработчики 802.11g предусмотрели ССК-модуляцию для скоростей до 11 Мбит/с и OFDM для более высоких скоростей. Но сети стандарта 802.11 при работе используют принцип CSMA/CA - множественный доступ к каналу связи с контролем несущей и предотвращением коллизий. Ни одно устройство 802.11 не должно начинать передачу, пока не убедится, что эфир в его диапазоне свободен от других устройств. Если в зоне слышимости окажутся устройства 802.11b и 802.11g, причем обмен будет происходить между устройствами 802.11g посредством OFDM, то оборудование 802.11b просто не поймет, что другие устройства сети ведут передачу, и попытается начать трансляцию. Последствия очевидны.

Чтобы не допустить подобной ситуации, предусмотрена возможность работы в смешанном режиме - ССК-OFDM. Информация в сетях 802.11 передается кадрами. Каждый информационный кадр включает два основных поля: преамбулу с заголовком и информационное поле (рис.5.56).



Рис. 5.56. Кадры IEEE 802.11g в различных режимах модуляции

Преамбула содержит синхропоследовательность и код начала кадра, заголовок - служебную информацию, в том числе о типе модуляции, скорости и продолжительности передачи кадра. В режиме ССК-OFDM преамбула и заголовок модулируются методом ССК (реально - путем прямого расширения спектра DSSS посредством последовательности Баркера, поэтому в стандарте 802.11g этот режим именуется DSSS-OFDM), а информационное поле - методом OFDM. Таким образом, все устройства 802.11b, постоянно "прослушивающие" эфир, принимают заголовки кадров и узнают, сколько времени будет транслироваться кадр 802.11g. В этот период они "молчат". Естественно, пропускная способность сети падает, поскольку скорость передачи преамбулы и заголовка - 1 Мбит/с.

Видимо, данный подход не устраивал лагерь сторонников технологии PBCC, и для достижения компромисса в стандарт 802.11g в качестве дополнительной возможности ввели, так же как и в 802.11b, необязательный режим - PBCC, в котором заголовок и преамбула передаются так же, как и при ССК, а информационное поле модулируется по схеме PBCC и передается на скорости 22 или 33 Мбит/с. В результате устройства стандарта 802.11g должны оказаться совместимыми со всеми модификациями оборудования 802.11b и не создавать взаимных помех. Диапазон поддерживаемых им скоростей отражен в таблице 5.9, зависимость скорости от типа модуляции - на рис.5.57.

Скорость, Мбит/с	Тип модуляции	
	Обязательно	Допустимо
1	Последовательность Баркера	

2	Последовательность Баркера	
5,5	ССК	PBCC
6	OFDM	OFDM
9		OFDM, ССК-OFDM
11	ССК	PBCC
12	OFDM	ССК-OFDM
18		OFDM, ССК-OFDM
22		PBCC
24	OFDM	ССК-OFDM
33		PBCC
36		OFDM, ССК-OFDM
48		OFDM, ССК-OFDM
54		OFDM, ССК-OFDM

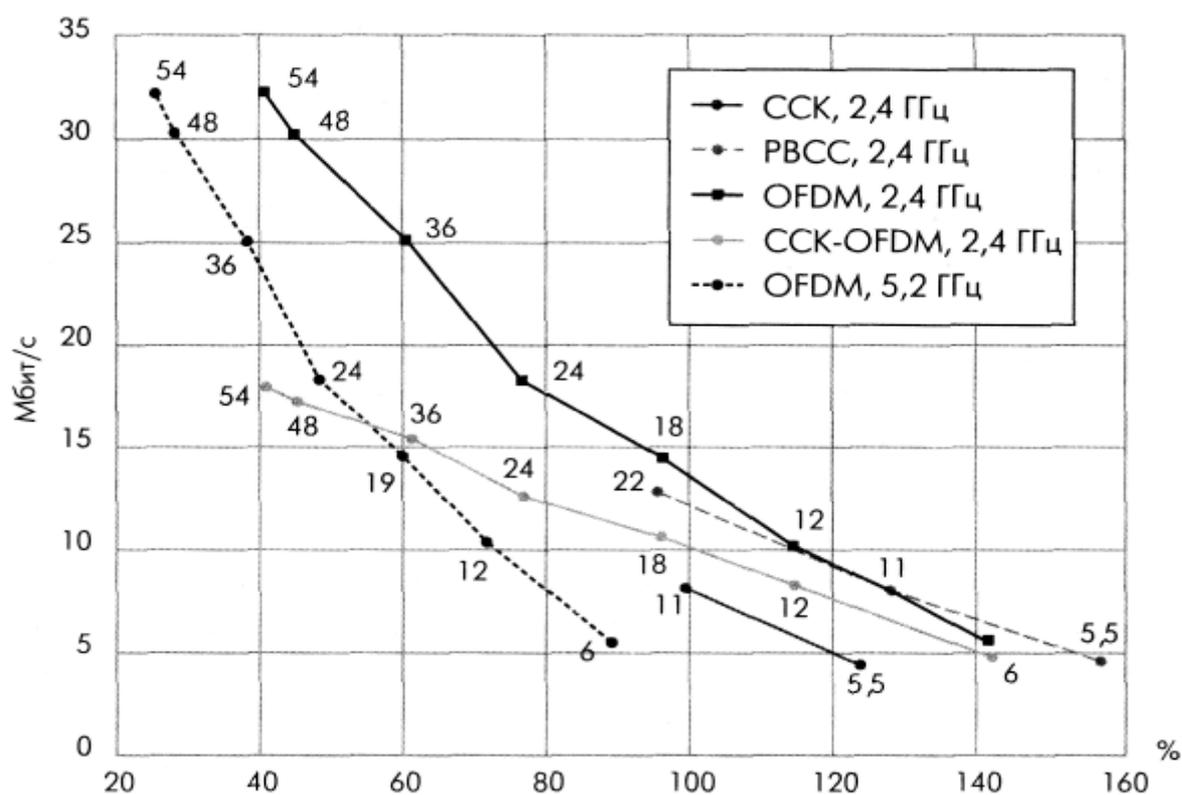


Рис. 5.57. Зависимость скорости передачи от расстояния для различных технологий передачи. Расстояние приведено в процентах, 100% - дальность передачи с модуляцией ССК на скорости 11 Мбит/с

Очевидно, что устройствам стандарта IEEE 802.11g достаточно долго придется работать в одних сетях с оборудованием 802.11b. Также очевидно, что производители в массе своей не будут поддерживать режимы CCK-OFDM и PBCC в силу их необязательности, ведь почти все решает цена устройства. Поэтому одна из основных проблем данного стандарта - как обеспечить бесконфликтную работу смешанных сетей 802.11b/g.

Основной принцип работы в сетях 802.11 - "слушать, прежде чем вещать". Но устройства 802.11b не способны услышать устройства 802.11g в OFDM-режиме. Ситуация аналогична проблеме скрытых станций: два устройства удалены настолько, что не слышат друг друга и пытаются обратиться к третьему, которое находится в зоне слышимости обоих. Для предотвращения конфликтов в подобной ситуации в 802.11 введен защитный механизм, предусматривающий перед началом информационного обмена передачу короткого кадра "запрос на передачу" (RTS) и получение кадра подтверждения "можно передавать" (CTS). Механизм RTS/CTS применим и к смешанным сетям 802.11b/g. Естественно, эти кадры должны транслироваться в режиме CCK, который обязаны понимать все устройства. Однако защитный механизм существенно снижает пропускную способность сети.

Таблица 5.10. Стандарты физического уровня

Параметр	802.11 DSSS	802.11 FHSS	802.11b	802.11a	802.11g
Частотный диапазон (ГГц)	2,4	2,4	2,4	5	2,4
Максимальная скорость передачи данных (Мбит/с)	2	2	11	54	54
Технология	DSSS	FHSS	CCK	OFDM	OFDM
Тип модуляции (для максимальной скорости передачи)	QPSK	GFSK	QPSK	64-QAM	64-QAM
Число неперекрывающихся каналов	3	3	3	15	3

## Создание модели радиointерфейса WiFi 802.11 [21]

### IEEE 802.11b

Чтобы открыть модель необходимо в командном окне ввести (Command Window):  
commwlan80211b. Появится модель, изображенная на рис.5.58.

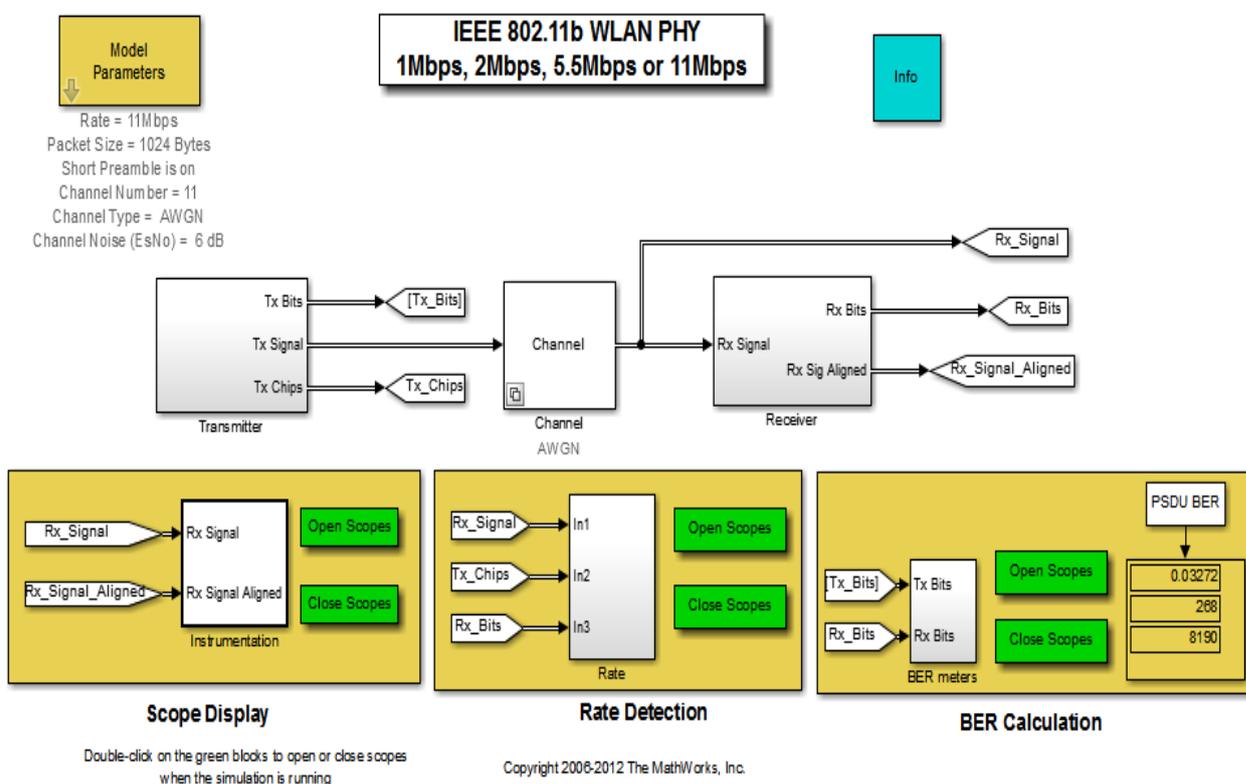


Рис. 5.58. Модель радиointерфейса 802.11b Simulink MATLAB 2015b

С помощью двойного щелчка на элемент Model Parameters можно устанавливать желаемые параметры моделируемой сети:

- скорость передачи данных (Rate),
- размер пакета (Packet Size),
- число каналов (Channel Number),
- тип канала (Channel Type),
- уровень шумов в канале (Channel EsNo).

Двойным щелчком по передатчику, приемнику или каналу передачи можно посмотреть их структурные схемы. Они представлены на рис.5.59, рис.5.60, рис.5.61.

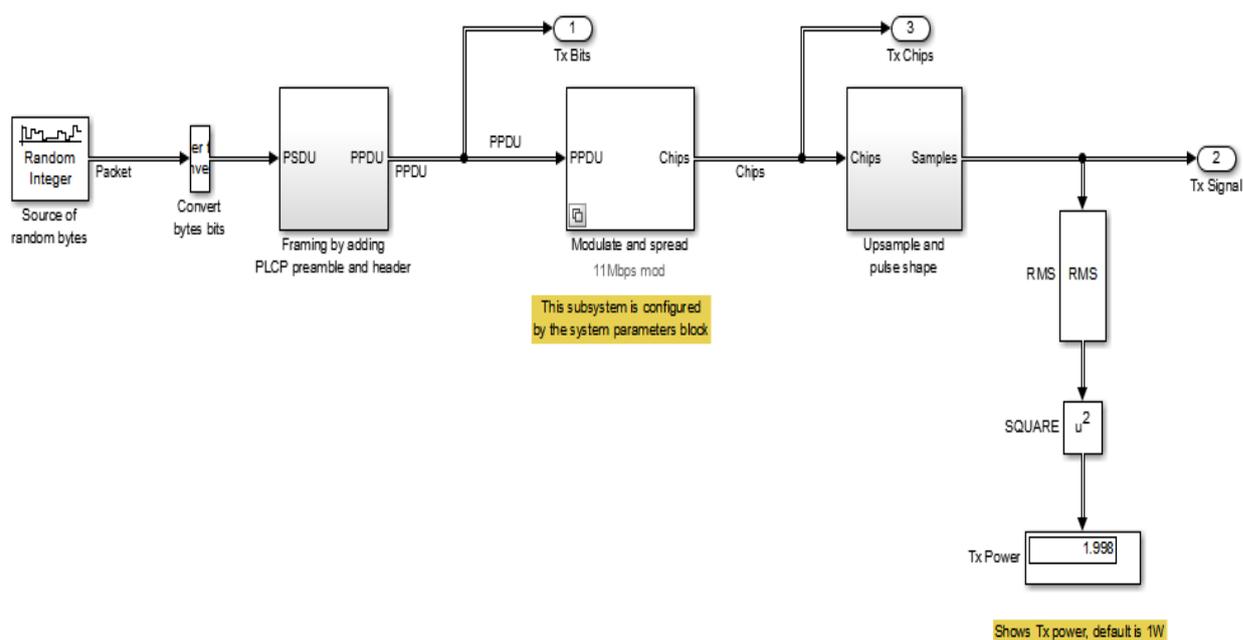


Рис. 5.59. Структурная схема передатчика IEEE 802.11b

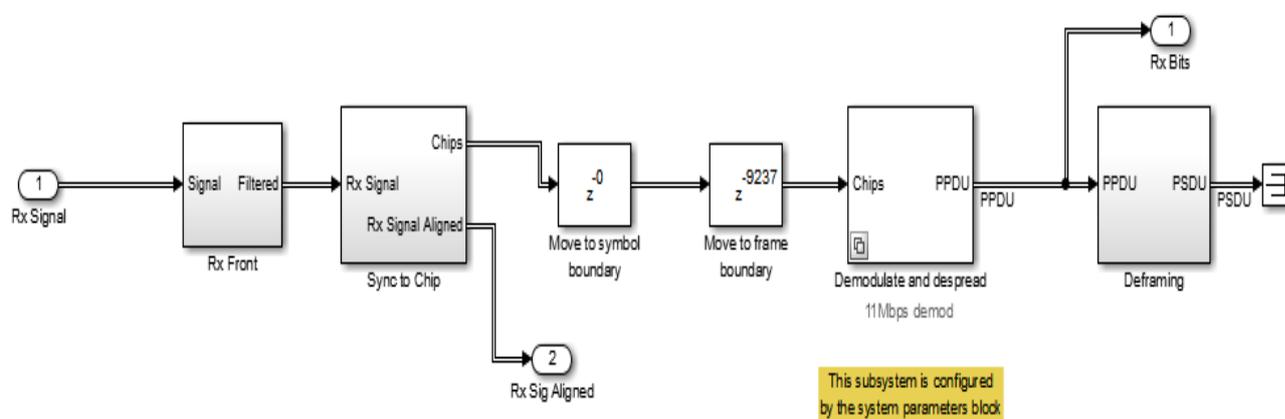


Рис. 5.60. Структурная схема приемника IEEE 802.11b

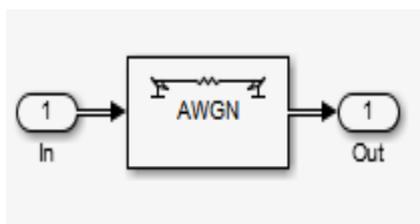


Рис. 5.61. Структурная схема канала передачи IEEE 802.11b

Пример частотной характеристика представлен на рис.5.62, а диаграмма созвездий на рис.5.63.

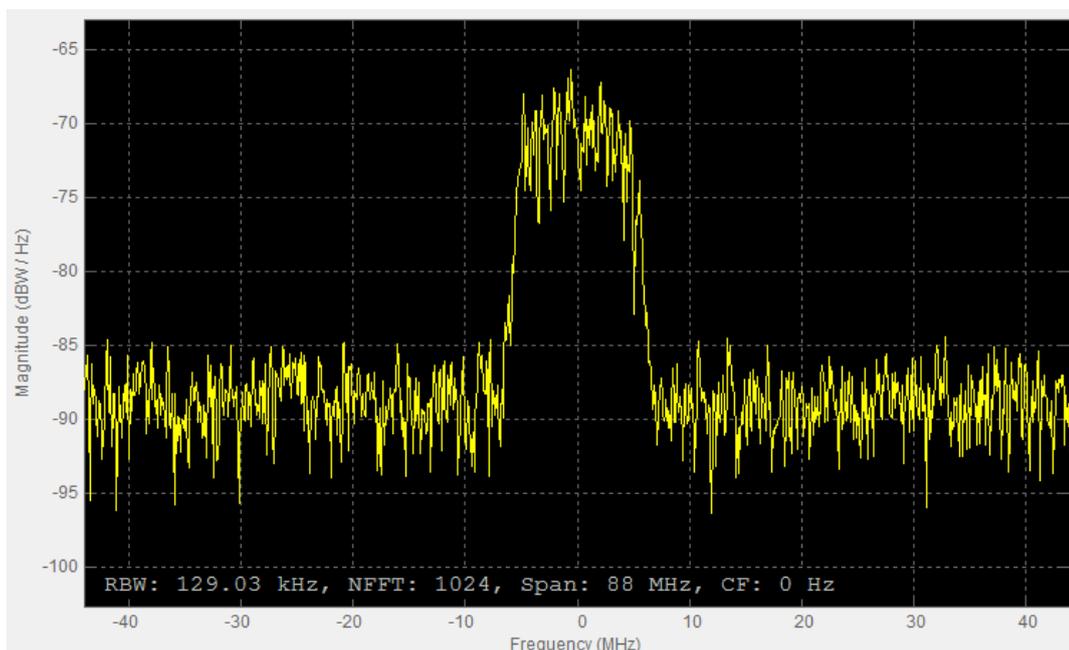


Рис. 5.62. Частотная характеристика для скорости передачи 11 Мбит/с

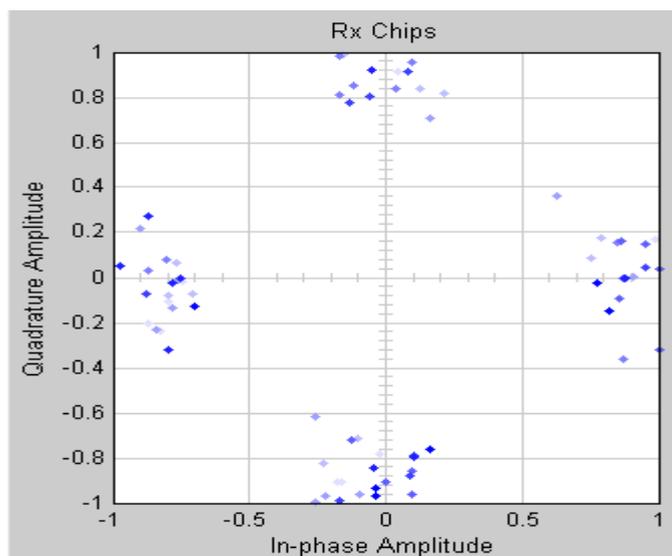


Рис. 5.63. Диаграмма созвездий для скорости передачи 11 Мбит/с

### Исследование влияния ошибок BER

BitErrorRate (BER) - коэффициент ошибок, отношение числа неверно принятых битов (0 вместо 1 и наоборот) к полному числу переданных битов при передаче по каналу связи.

Чтобы получить зависимость BER от отношения сигнал/шум необходимо изменять уровень шумов в канале (0-14) и снимать показания в блоке BER Calculation в верхнем дисплее.

Протестировав систему таким образом, были получены зависимости, представленные на рис.5.64.

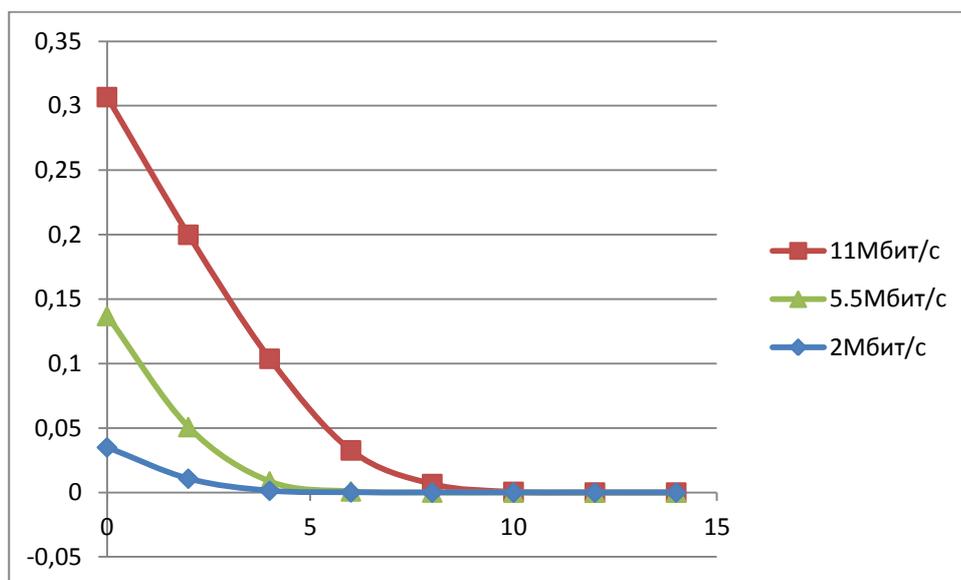


Рис. 5.64. Зависимость ошибки BER от отношения сигнал/шум для различных скоростей IEEE 802.11b

По полученным результатам можно сделать следующие выводы:

- 1) Большим скоростям соответствует большая вероятность появления ошибки
- 2) Для уменьшения ошибки необходимо увеличивать отношение сигнал/шум
- 3) Большим скоростям необходимо более высокое значение отношения сигнал шум для устранения возможных ошибок.

В результате работы изучены стандарты IEEE 802.11.

Рассмотрены и протестированы модели данных стандартов, реализованные в среде Simulink Matlab. Получены графики зависимостей вероятности ошибки (BER) от отношения сигнал/шум для разных скоростей.

## Методические указания к моделированию [21]

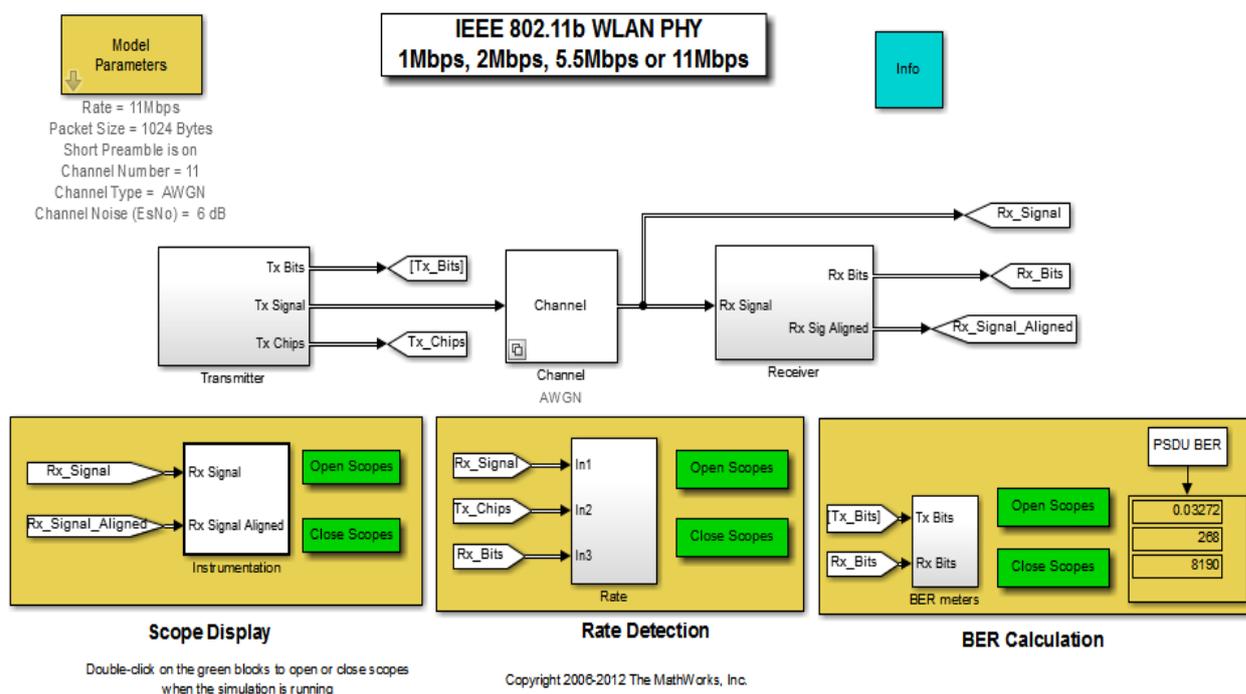


Рис. 5.65. Схем сети IEEE 802.11b MATLAB

1. Запустить модель командой `commwlan80211b` и изучить ее параметры. Сделать скриншоты полной схемы, а также передатчика, приёмника и канала.
2. Снять частотную характеристику, а также диаграмму созвездий для 4-х вариантов максимальной скорости передачи (1, 2, 5, 11 Мбит/с). Параметр «тип канала» (Channel Type) – none.
3. Для каждой скорости изменяя отношение сигнал/шум в канале от 0 до 14дБ снять зависимость BER от Channel EsNo. Параметр «тип канала» (Channel Type) – AWGN.
5. Сделать выводы по проделанной работе.

### 5.4. Имитационное моделирование системы мобильной связи стандарта IEEE 802.15.4 ZigBee на базе ПО MATLAB [21]

Среди наиболее известных беспроводных технологий можно выделить: Wi-Fi, Wi-Max, Bluetooth, Wireless USB и относительно новую технологию — ZigBee, которая изначально разрабатывалась с ориентацией на промышленные применения.

Каждая из этих технологий имеет свои уникальные характеристики, которые определяют соответствующие области применения.

Стандарт	802.15.4 ZigBee™		802.15.1 Bluetooth	802.15.3 High Rate WPAN, WiMedia	802.15.3a* UWB	802.11b Wi-Fi	
Приложения	Мониторинг, управление, сети датчиков, домашняя/промышленная автоматика		Голос, данные, замена кабелей	Потоковое мультимедиа, замена кабелей аудио/видеосистем		Данные, видео, ЛВС	
Преимущества	Цена, энергосбережение, размеры сети, менее загруженные диапазоны	Цена, энергосбережение, размеры сети, глобальный диапазон	Цена, энергосбережение, передача голоса, переключения частоты	Высокая скорость, энергосбережение		Скорость, гибкость	
Частота, ГГц	0,868	0,915	2,4		3,1 – 10,6	2,4	
Макс. скорость	20 Кбит/с	40 Кбит/с	250 Кбит/с	1 Мбит/с	22 Мбит/с (доп. 11, 33, 44, 55 Мбит/с)	110 Мбит/с (10 м), 200 Мбит/с (4 м) (доп. 480 Мбит/с)	11 Мбит/с
Выходная мощность (ном.), дБм	0		0 (класс 3) 4 (класс 2) 20 (класс 1)	0	< 20 (110 Мбит/с) < 24 (200 Мбит/с)	20	
Дальность, м	10 – 100		10 (класс 3) 100 (класс 1)	5 – 50	10 (110 Мбит/с) 4 (200 Мбит/с)	100	
Чувствительность (спецификация, дБм)	-92	-85	-70	-75	-	-76	
Размер стека, Кбайт	4 – 32		> 250	-		> 1000	
Срок службы батареи, дней	100 – 1000+		1 – 7	теоретически более 1000		0,5 – 5	
Размер сети	65536 (16-битные адреса), 2 <sup>64</sup> (64-битные адреса)		мастер +7	до 127 на хост		32	

Рис. 5.66. Основные характеристики популярных стандартов беспроводной связи

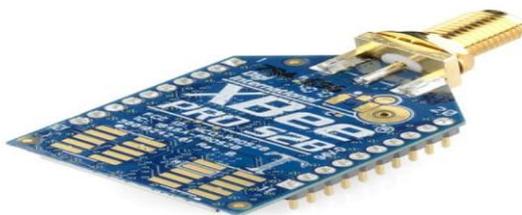


Рис. 5.67. Внешний вид микросхемы ZigBee

Анализ беспроводных технологий показывает, что высокоскоростные технологии Wi-Fi, Wi-Max, Bluetooth, Wireless USB предназначены в первую очередь для обслуживания компьютерной периферии и устройств мультимедиа. Они оптимизированы для передачи больших объемов информации на высоких скоростях, работают в основном по топологии

«точка-точка» или «звезда» и малопригодны для реализации сложных разветвленных промышленных сетей с большим количеством узлов. Напротив, технология ZigBee имеет достаточно скромные показатели скорости передачи данных и расстояния между узлами, но обладает следующими важными, с точки зрения применения в промышленности, преимуществами:

- Она ориентирована на преимущественное использование в системах распределенного мульти-микропроцессорного управления со сбором информации с интеллектуальных датчиков, где вопросы минимизации энергопотребления и процессорных ресурсов являются определяющими.
- Предоставляет возможность организации самоконфигурируемых сетей со сложной топологией, в которых маршрут сообщения автоматически определяется не только числом исправных или включенных/выключенных на текущий момент устройств (узлов), но и качеством связи между ними, которое автоматически определяется на аппаратном уровне.
- Обеспечивает масштабируемость — автоматический ввод в работу узла или группы узлов сразу после подачи питания на узел.
- Гарантирует высокую надежность сети за счет выбора альтернативного маршрута передачи сообщений при отключениях/сбоях в отдельных узлах.
- Поддерживает встроенные аппаратные механизмы шифрации сообщений AES-128, исключая возможность несанкционированного доступа в сеть.

### **Организация сети ZigBee**

ZigBee — относительно новый стандарт беспроводной связи, который изначально разрабатывался как средство для передачи небольших объемов информации на малые расстояния с минимальным энергопотреблением. Фактически этот стандарт описывает правила работы программно-аппаратного комплекса, реализующего беспроводное взаимодействие устройств друг с другом.

Стек протоколов ZigBee представляет собой иерархическую модель, построенную по принципу семиуровневой модели протоколов передачи данных в открытых системах OSI (OpenSystemInterconnection). Стек включает в себя уровни стандарта IEEE 802.15.4, отвечающие за реализацию канала связи, и программные сетевые уровни и уровни поддержки приложений, определенные спецификацией ZigBee. Модель реализации стандарта связи ZigBee представлена на рисунке 5.69.



Рис. 5.68. Многоуровневая модель стандарта связи ZigBee

Стандарт IEEE 802.15.4 определяет два нижних уровня стека: уровень доступа к среде (MAC) и физический уровень передачи данных в среде распространения (PHY), то есть нижние уровни протокола беспроводной передачи данных. Альянс определяет программные уровни стека ZigBee от уровня канала передачи данных (DataLinkControl) до уровня профилей устройств (ZigBeeProfiles). Прием и передача данных по радиоканалу осуществляется на физическом уровне PHY, определяющем рабочий частотный диапазон, тип модуляции, максимальную скорость, число каналов. Уровень PHY осуществляет активацию-деактивацию приемопередатчика, детектирование энергии принимаемого сигнала на рабочем канале, выбор физического частотного канала, индикацию качества связи при получении пакета данных и оценку свободного канала. Важно понимать, что стандарт 802.15.4 — это физическое радио (микросхема радио-приемопередатчика), а ZigBee — это логическая сеть и программный стек, обеспечивающие функции безопасности и маршрутизации.

Далее в структуре стека ZigBee следует уровень контроля доступа к среде IEEE 802.15.4 MAC, осуществляющий вход и выход из сети устройств, организацию сети, формирование пакетов данных, реализацию различных режимов безопасности (включая 128-битное шифрование AES), 16- и 64-битную адресацию.

Уровень MAC обеспечивает различные механизмы доступа в сеть, поддержку сетевых топологий от «точка-точка» до «многочейковая сеть», гарантированный обмен данными (ACK, CRC), поддерживает потоковую и пакетную передачи данных.

Для предотвращения нежелательных взаимодействий возможно использование временного разделения на основе протокола CSMA-CA (протокол множественного доступа к среде с контролем несущей и предотвращением коллизий).

Временное разделение ZigBee базируется на использовании режима синхронизации, при котором подчиненные сетевые устройства, большую часть времени находящиеся в «спящем» состоянии, периодически «просыпаются» для приема сигнала синхронизации от сетевого координатора, что позволяет устройствам внутри локальной сетевой ячейки знать, в какой момент времени осуществлять передачу данных. Данный механизм, основанный на определении состояния канала связи перед началом передачи, позволяет существенно сократить (но не устранить) столкновения, вызванные передачей данных одновременно несколькими устройствами. Стандарт 802.15.4 основывается на полудуплексной передаче данных (устройство может либо передавать, либо принимать данные), что не позволяет использовать метод CSMA-CA для обнаружения коллизий — только для их предотвращения.

В спецификации стека предусмотрены три типа устройств: координатор, маршрутизатор и конечное устройство.

**Координатор** инициализирует сеть, управляет ее узлами, хранит информацию о настройках каждого узла, задает номер частотного канала и идентификатор сети PAN ID, а в процессе работы может являться источником, приемником и ретранслятором сообщений.

**Маршрутизатор** отвечает за выбор пути доставки сообщения, передаваемого по сети от одного узла к другому, и в процессе работы также может являться источником, приемником или ретранслятором сообщений. Если маршрутизаторы имеют соответствующие возможности, они могут определять оптимизированные маршруты к определенной точке и хранить их для последующего использования в таблицах маршрутизации.

**Оконечное устройство** не участвует в управлении сетью и ретрансляции сообщений, являясь только источником/приемником сообщений.

Среди свойств ZigBee следует особо выделить поддержку сложных топологий сетей. Именно за счет этого, при относительно малой максимальной дальности связи двух близлежащих устройств, возможно расширить зону покрытия сети в целом. Также этому способствует 16-битная адресация, позволяющая объединять в одну сеть более 65 тыс. устройств.

#### **Спецификация стандарта IEEE 802.15.4**

Спецификация ZigBee-стека определяет сетевой уровень, уровни безопасности и доступа к приложению и может использоваться совместно с решениями на базе стандарта 802.15.4 для обеспечения совместимости устройств.

Таблица 5.11. Спецификация стандарта IEEE 802.15.4

Стандарт	802.15.4 ZigBee™		
Частота	868 МГц	915 МГц	2,4 ГГц
Число каналов/шаг	1/-	10/2 МГц	16/5 МГц
География распространения	Европа	Америка	Весь мир
Макс. скорость, модуляция	20 кбит/с, BPSK	40 кбит/с, BPSK	250 кбит/с, O- QPSK
Выходная мощность, ном.	0 dBm (1 мВт)	0 dBm (1 мВт)	0 dBm (1 мВт)
Дальность	10–100м		
Чувствительность (спецификация)	-92dBm	-92dBm	-85dBm
Размер стека	4–32 кбайт		
Срок службы батареи	От 100 до 1000 и более дней		
Размер сети	65536 (16-битные адреса), $2^{64}$ (64-битные адреса)		

### Практическая часть

Задание:

6. Собрать схему
7. Подготовить схемы для реализации Стандарта ZigBee 802.15.4 основываясь на примере, представленном в отчете.
8. Изменять SNR в пределах от 1 до 100 (не менее 4-х точек)
9. Построить графики зависимости SNR от BER
10. Все поэтапное исследование представить в отчете.

В рабочем поле необходимо собрать схему для работы стандарта ZigBee 802.15.4. Схема представлена на рисунке 5.70.

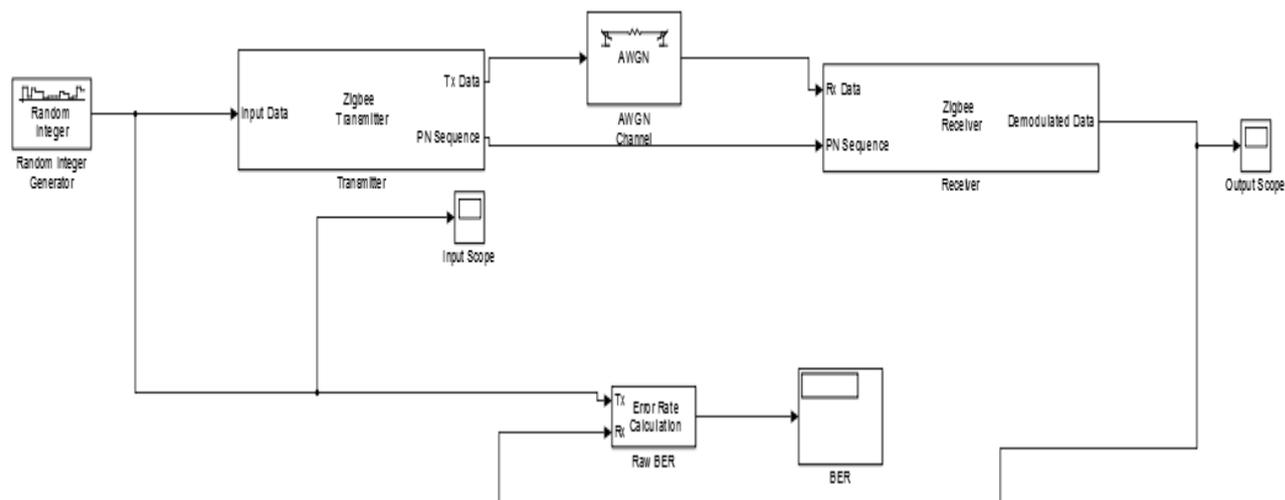


Рис. 5.69. Схема стандарта ZigBee 802.15.4 Simulink MATLAB 2015b

В состав схемы входят:

1. RandomIntegerGenerator
2. ZigBeeTransmitter
3. AWGN Channel (каналпередачи)
4. ZigBeeReciever
5. ErrorRateCalculation (анализаторошибок)
6. Display

Рассмотрим каждый блок отдельно. Все значения, заданные в блоках, помимо отношения Сигнал/шум в канале, остаются неизменными.

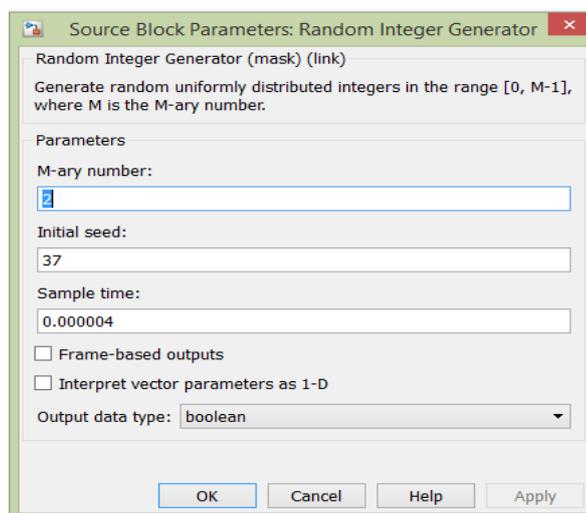


Рис. 5.70. Параметры блока Random Integer Generator

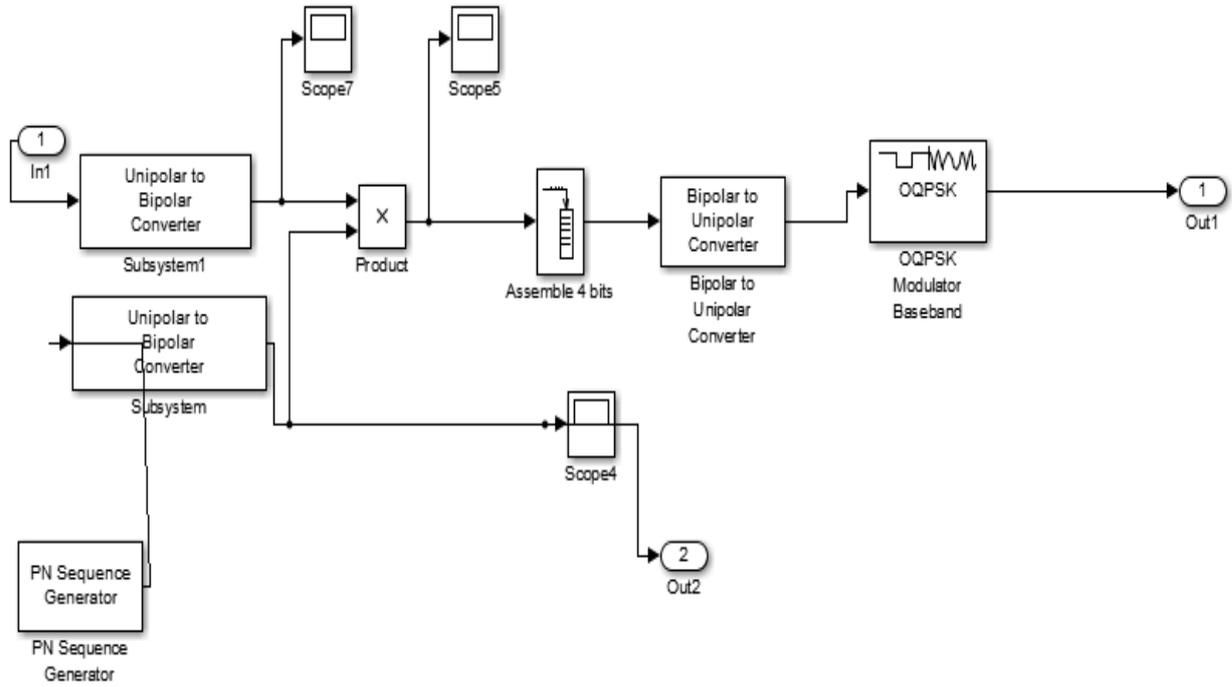


Рис. 5.71. Схема ZigBee Transmitter

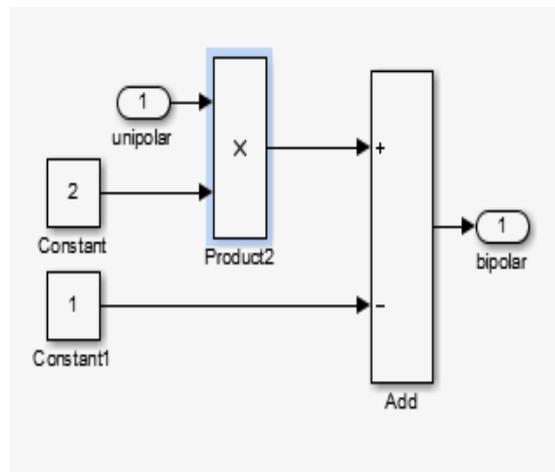


Рис. 5.72. Unipolar to bipolar converter

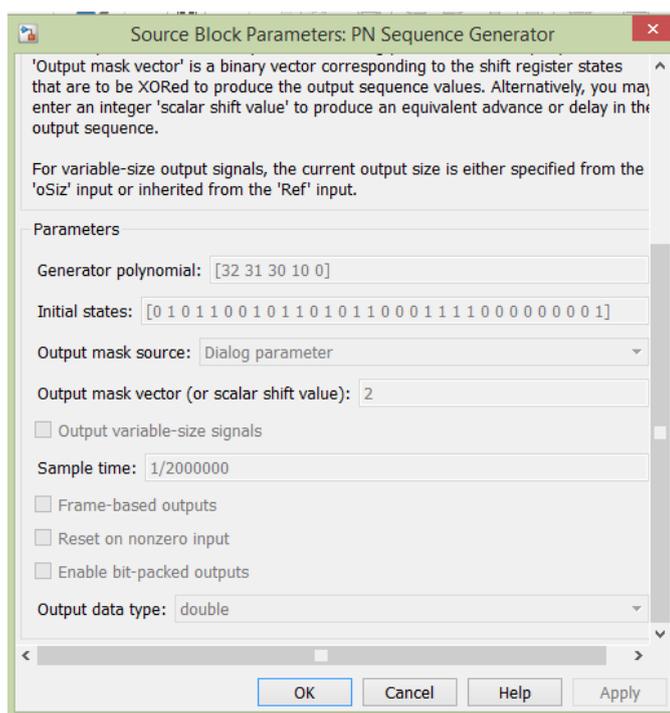


Рис. 5.73. Параметры блока PN sequence generator

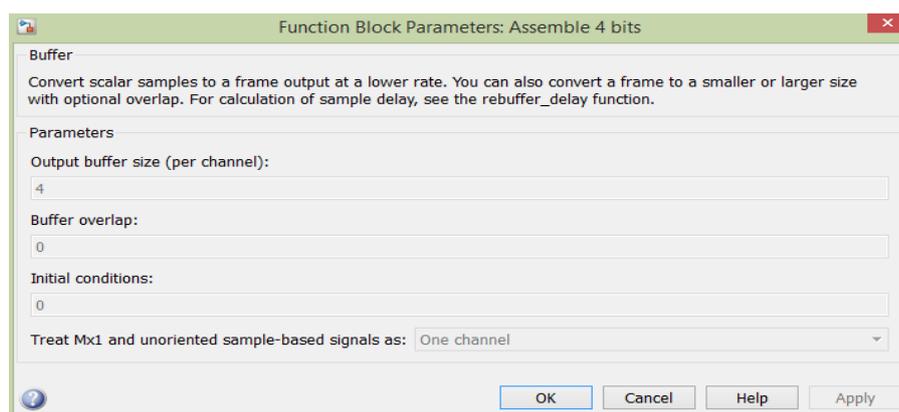


Рис. 5.74. Параметры блока Function block parameters: Assemble 4 bits

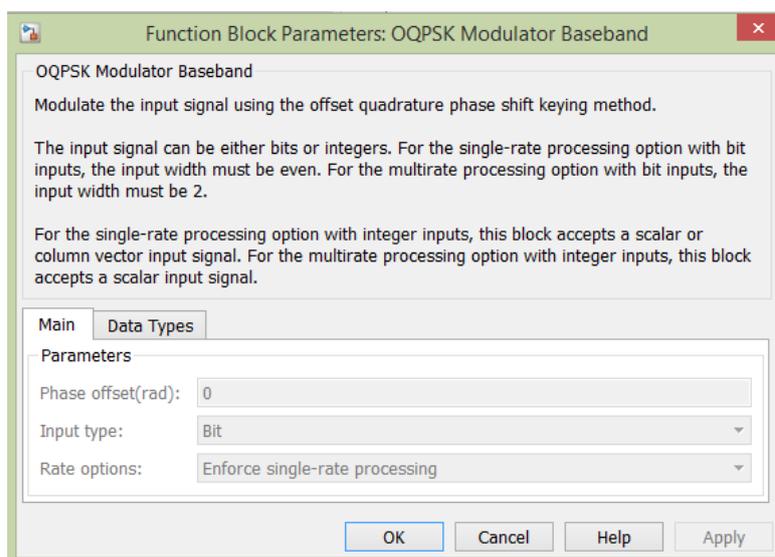


Рис. 5.75. Параметры блока OQPSK modulator baseband

ОQPSK - четырехпозиционная фазовая модуляция со сдвигом квадратур (**ОQPSK**), где битовые потоки, подаваемые на модуляторы квадратур I и Q, сдвинуты друг относительно друга на длительность одного бита (половина символьного интервала).

Рассмотрим блок канала с БГШ. В данном блоке необходимо изменять значения в строчке  $E_b/N_0$  от 1 до 100.

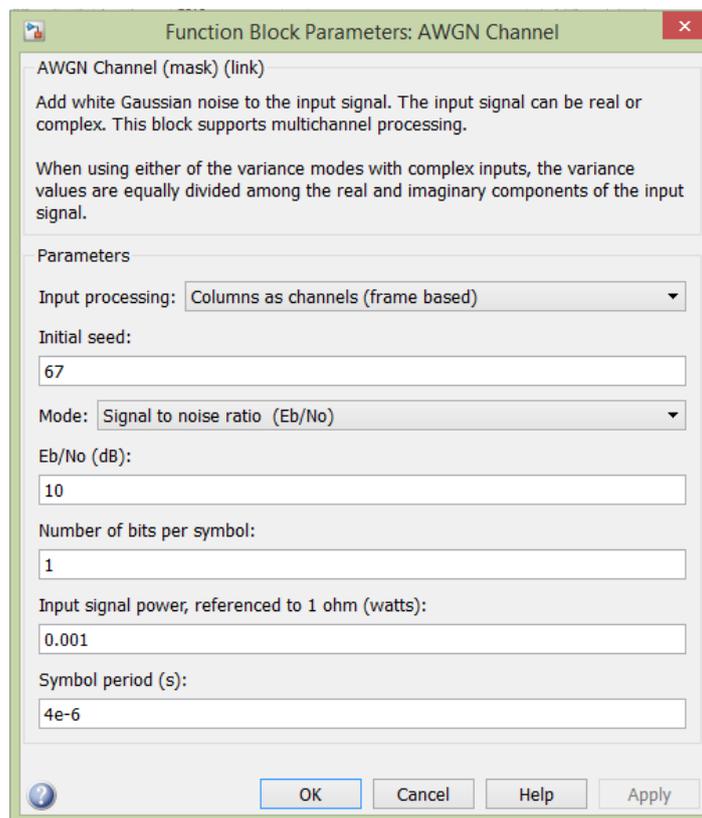


Рис. 5.76. Параметры блока AWGNchannel

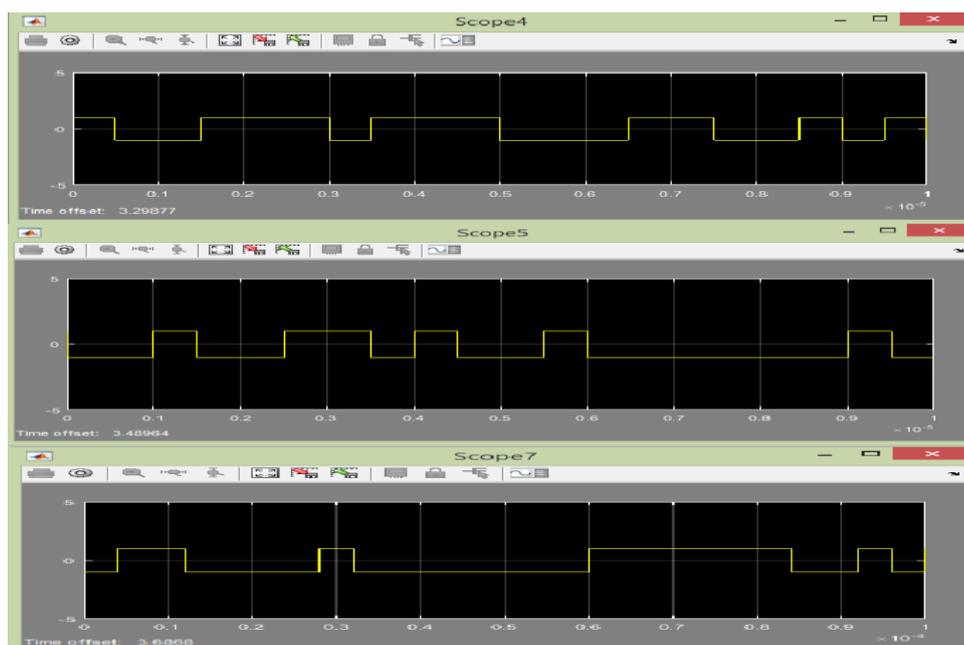


Рис. 5.77. Вид сигнала на осциллографах 4, 5, 7

Рассмотрим подробнее блок ZigBee Receiver.

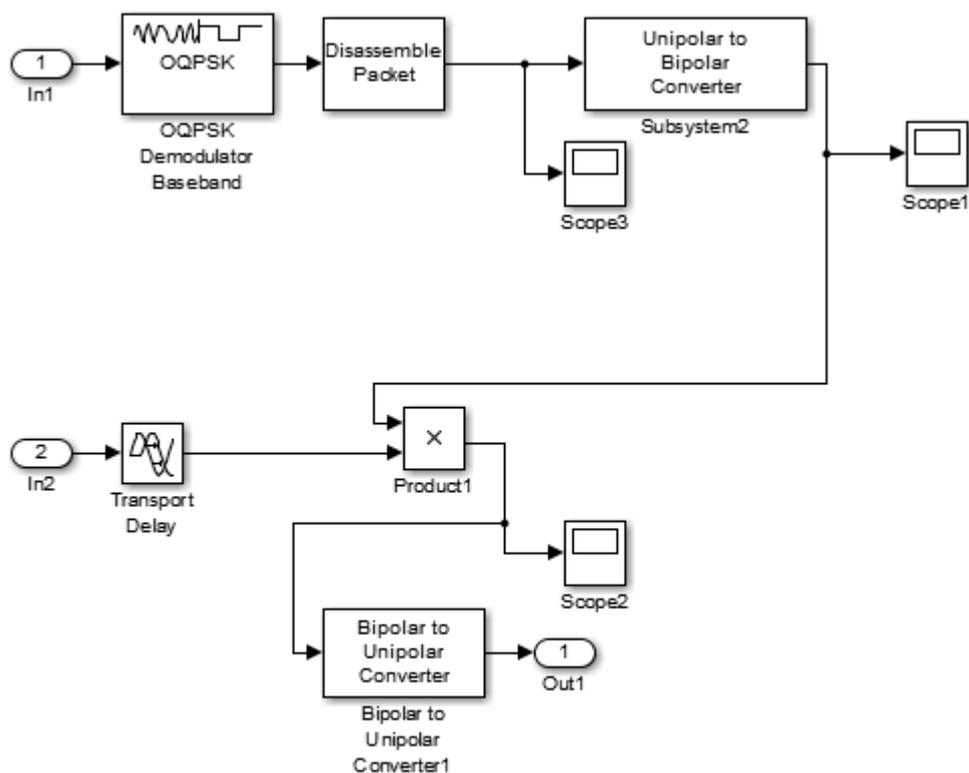


Рис. 5.78. Схема приемника стандарта ZigBee

Рассмотрим каждый блок отдельно. Единственным незнакомым элементом является блок TransportDelay.

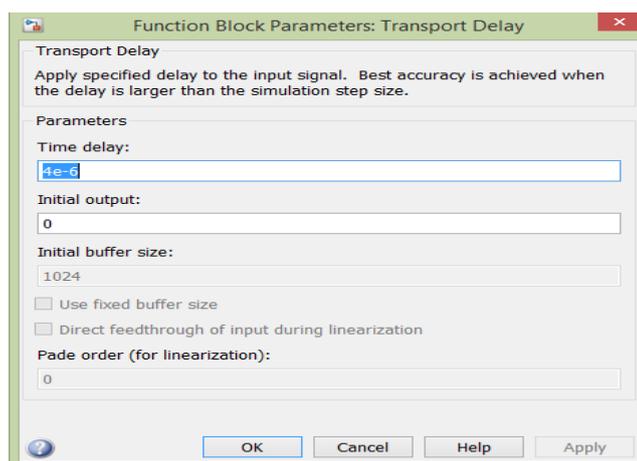


Рис. 5.79. Параметры блока TransportDelay

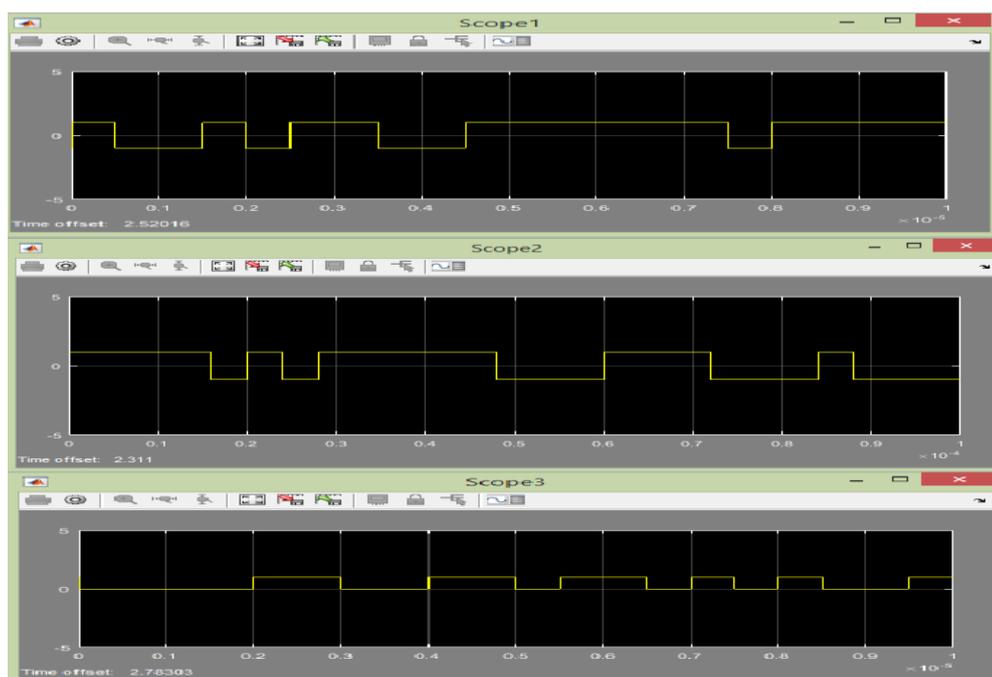


Рис. 5.80. Вид сигнала на осциллографах 1, 2, 3

Вернемся к общей схеме стандарта ZigBee.

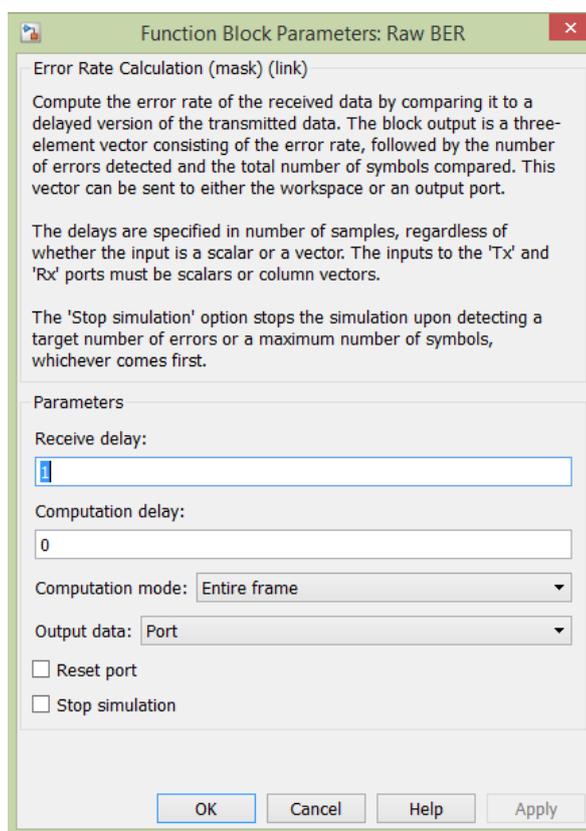


Рис. 5.81. Параметры блока RawBER

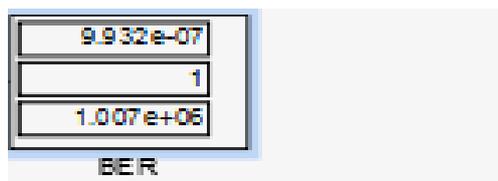


Рис. 5.82. Счетчик ошибок

Для построения графика зависимости BER от SNR, необходимо из счетчика брать первую строку.

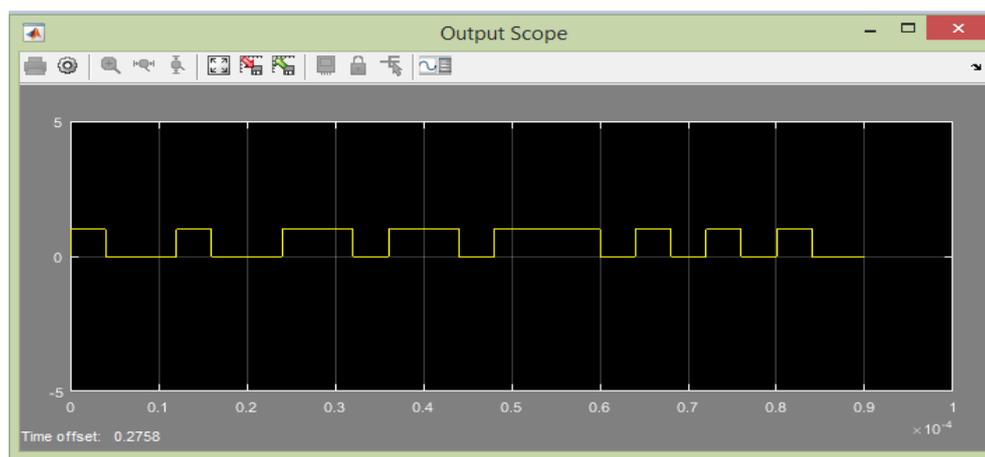


Рис. 5.83. Вид сигнала на выходе

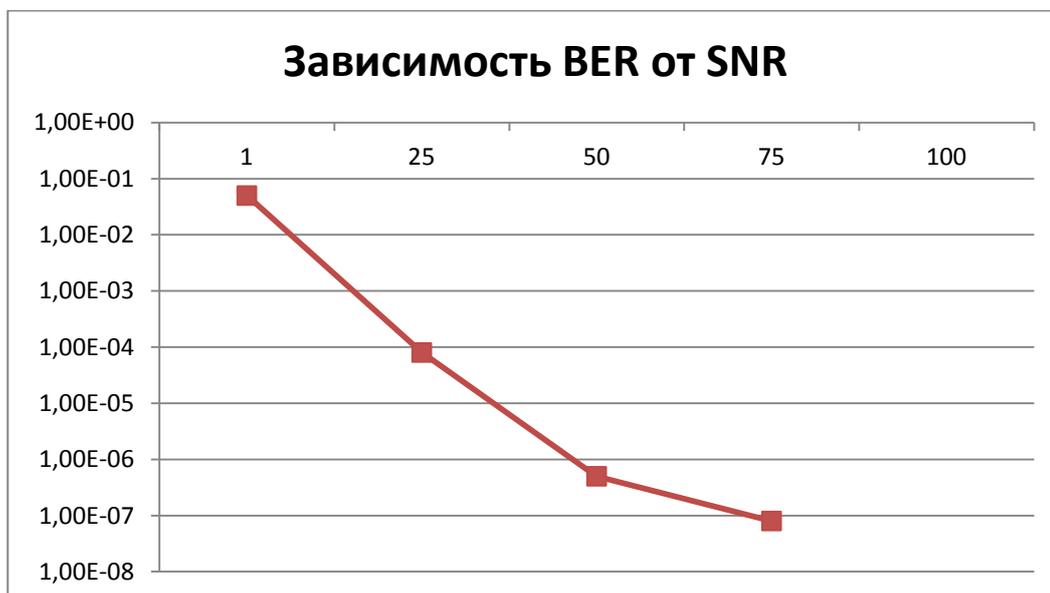


Рис. 5.84. График зависимости BER от SNR

В разделе построена схема стандарта ZigBee 802.15.4 в среде Simulink. Построен график зависимостей зависимости BER от SNR. Из графика (рисунок 5.85) видно, что при увеличении значения сигнал/шум, снижается количество ошибок.

## 5.5. Имитационное моделирование системы мобильной связи стандарта IEEE 802.15.1 (Bluetooth) на базе ПО MATLAB [16, 21]

### Bluetooth

Стандарт Bluetooth является компромиссным с точки зрения соотношения параметров экономичность/дальность/скорость. По своей функциональности и возможности применения в различных приложениях он имеет наибольшее число пересечений с другими стандартами группы Short Range RF. Поэтому для начала рассмотрим именно его.

Основная идея Bluetooth заключалась в создании универсального, надежного и очень дешевого радиоинтерфейса беспроводного доступа. Технология Bluetooth позволяет обеспечить сопряжение с различным профессиональным и бытовым оборудованием в режимах передачи речи, данных и мультимедиа, при этом гарантируется его электромагнитная совместимость с другим домашним или офисным оборудованием. Как было указано в таблице, существует всего три класса устройств Bluetooth, если градировать их по излучаемой мощности: 1-й — до 100 метров (до 100 мВт); 2-й — до 10 метров (до 2,5 мВт); 3-й — до 1 метра (до 1 мВт).

Для определения модели поведения при установлении соединения между различными типами устройств в технологии Bluetooth введено понятие профиль. Этим термином обозначается набор функций и возможностей, которые использует Bluetooth в качестве механизма транспортировки. Профили гарантируют возможность обмена информацией между устройствами разных производителей. Bluetooth SIG определяет 15 стандартных профилей:

- Generic Access Profile (GAP);
- Service Discover Application Profile (SDAP);
- Serial Port Profile (SPP);
- Dial-up Networking Profile (DUNP);
- Generic Object Exchange Profile (GOEP);
- Object Push Profile (OPP);
- File Transfer Profile (FTP);
- Synchronization Profile (SP);
- AV Control, Headset Profile (HSP);
- Advanced Audio Distribution Profile (A2DP);
- Basic Imaging Profile (BIP);

- Handsfree Profile (HFP);
- Human Interface Device Profile (HID);
- LAN Access Profile (LAP);
- Sim-Card Access Profile (SAP).

По характеру взаимодействия со внешними устройствами и приложениями архитектура всех существующих модулей Bluetooth может быть разделена на три вида (рис. 1). Модули с двухпроцессорной архитектурой (рис. 1а) не содержат в себе программного высокоуровневого стека Bluetooth с поддержкой стандартных профилей. Это значит, что необходимые профили Bluetooth должны быть реализованы на внешнем процессоре. Взаимодействие внешнего процессора с модулем происходит через виртуальный интерфейс HCI (Host Controller Interface). В частном случае HCI может быть реализован через аппаратный интерфейс SPI или UART.

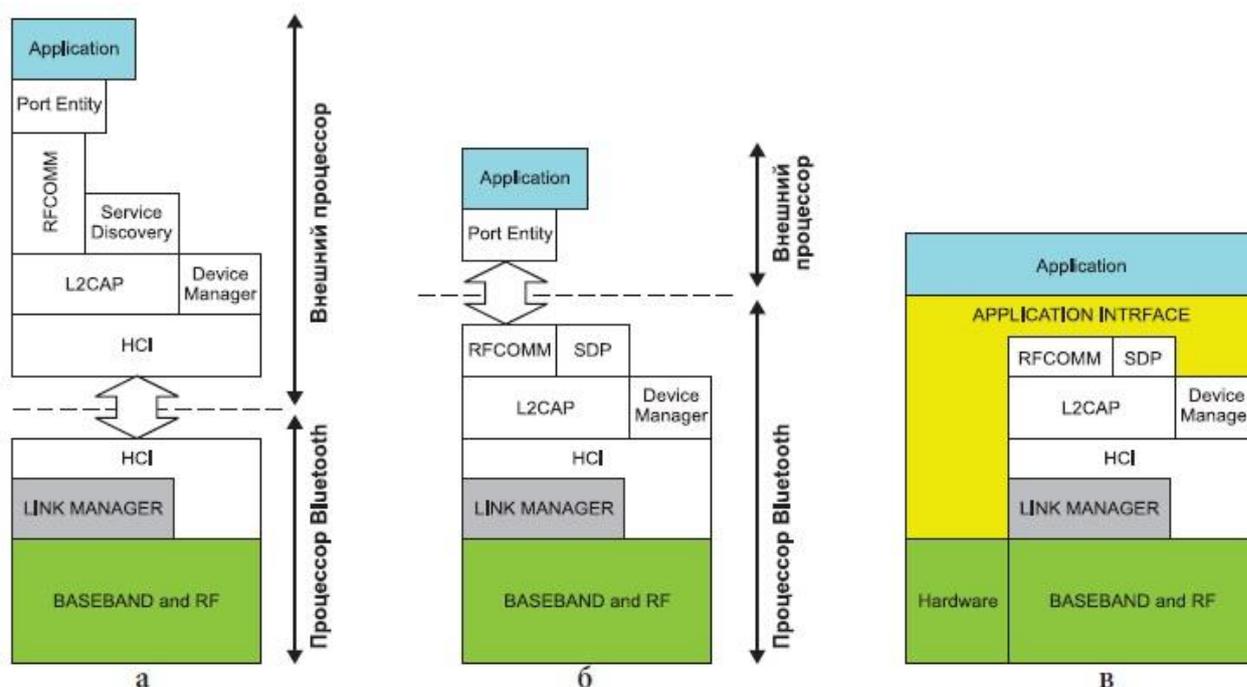


Рис. 5.85. Разновидности архитектуры модулей стандарта Bluetooth: а) двухпроцессорная; б) встроенная двухпроцессорная; в) однопроцессорная

Модули Bluetooth со встроенной двухпроцессорной архитектурой (рис. 5.85б) являются наиболее распространенными. Данная разновидность архитектуры подразумевает наличие стека Bluetooth высокого уровня с поддержкой стандартных профилей непосредственно во внутреннем процессоре модуля. В этом случае приложение, работающее на внешнем процессоре, взаимодействует с модулем Bluetooth через аппаратные интерфейсы.

Однопроцессорная архитектура (рис. 5.85в) является наименее распространенной. Для ее реализации разработчик должен создать специальное приложение, которое будет работать на внутреннем процессоре модуля Bluetooth. В этом случае модуль превращается в автономное устройство, доступ к которому через внешние аппаратные интерфейсы закрыт.

Принадлежность модуля к той или иной архитектуре может определяться как его аппаратной реализацией, так и внутренним программным обеспечением. Например, в частном случае один и тот же модуль Bluetooth может быть отнесен к любой из трех разновидностей архитектуры в зависимости от типа прошивки, загруженной во внутренний процессор модуля. Такой подход пользуется наибольшей популярностью среди зарубежных производителей.

Чтобы получить наиболее полное представление о роли Bluetooth среди других представителей группы Short Range RF, обратимся к истории (рис. 5.86). Развитие Bluetooth с самого начала шло по пути увеличения скорости обмена данными, снижения энергопотребления, повышения безопасности и надежности соединения. Вплоть до версии 3.0 сохранялась обратная совместимость всех версий Bluetooth между собой. До сих пор в эксплуатации встречаются устройства Bluetooth версий 1.1 и 1.2, которые успешно используются совместно с 2.0 и 2.1.

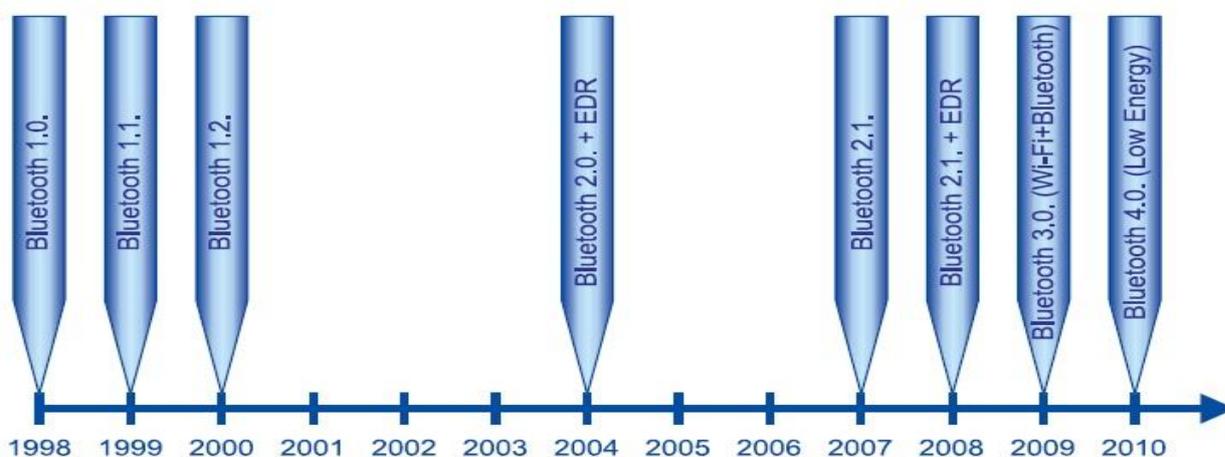


Рис. 5.86. Хронология развития стандарта Bluetooth

Bluetooth 3.0 является чем-то средним между Bluetooth и Wi-Fi. Модули с ее поддержкой соединяют в себе две радиосистемы: первая обеспечивает передачу данных в 3 Мбит/с (стандартная для Bluetooth 2.0) и имеет низкое энергопотребление; вторая совместима со стандартом 802.11 (Wi-Fi) и обеспечивает возможность передачи данных со скоростью до 24 Мбит/с (сравнима со скоростью сетей Wi-Fi). Выбор радиосистемы для передачи данных зависит от размера передаваемого файла. Это один из наиболее ярких примеров объединения

двух разных технологий для завоевания новых сегментов рынка. Правда, успеха эта попытка не имела: распространения Bluetooth 3.0 не получил.

Bluetooth 4.0 не имеет обратной совместимости с предыдущими версиями. Сверхнизкое энергопотребление достигается за счет использования специального алгоритма работы. Передатчик включается только на время отправки данных, что обеспечивает возможность работы от одной батарейки типа CR2032 в течение нескольких лет. Стандарт предоставляет скорость передачи данных в 1 Мбит/с при размере пакета 8–27 байт. В новой версии два Bluetooth-устройства смогут устанавливать соединение менее чем за 5 мс и поддерживать его на расстоянии до 100 м. Для этого используется усовершенствованная коррекция ошибок, а необходимый уровень безопасности обеспечивает 128-битное шифрование.

Предполагается, что Bluetooth 4.0 будет конкурировать и вытеснять ZigBee в классе малопотребляющих радиочастотных устройств с поддержкой сложных сетей. Это также является ярким примером пересечения двух разных технологий, в данном случае — ZigBee и Bluetooth.

Проанализировав современное состояние технологии Bluetooth, можно обозначить плюсы и минусы. К достоинствам стандарта относятся:

- высокий уровень стандартизации и совместимость между устройствами Bluetooth разных производителей;
- защита передаваемых данных;
- низкая стоимость;
- высокая дальность действия (до 1000 м);
- универсальность и большое разнообразие модулей под разные задачи.

Среди недостатков отметим:

- Относительно высокое энергопотребление (работа от автономных источников питания не всегда возможна). Предполагается, что этого недостатка будет лишена новая версия спецификации Bluetooth 4.0.
- Относительно невысокая скорость обмена данными (до 1 Мбит/с). Как правило, реальная скорость обмена данными ограничивается пропускной способностью внешних аппаратных интерфейсов модуля.

Одно из основных преимуществ стандарта Bluetooth заключается в его высоком уровне стандартизации и широчайшем распространении в составе пользовательских электронных устройств. Это позволяет в ряде случаев практически в два раза сэкономить время и затраты на разработку при проектировании некоторой системы сбора данных, телеметрии или

управления на основе Bluetooth, поскольку в качестве одной из сторон беспроводного обмена данными может выступать, например, обычный серийно выпускаемый ноутбук или коммуникатор с поддержкой данной технологии.

Исходя из характерных особенностей модулей Bluetooth, сформировались их области применения в России и за рубежом:

- Автомобильная электроника. Модули Bluetooth могут использоваться в бортовых автомобильных системах контроля и управления. Эта область применения характерна для России.
- Системы удаленного управления и телеметрии. Здесь устройства Bluetooth могут использоваться наряду с модулями технологий Wi-Fi, ZigBee, Short Range RF 434/868 МГц. Данная область применения в равной степени актуальна как для России, так и для зарубежных стран.

### **Bluetooth**

Ноутбуки, сотовые телефоны, смартфоны, торговые терминалы со встроенной функцией Bluetooth. Bluetooth - это современная технология беспроводной передачи данных, позволяющая соединять друг с другом практически любые устройства: мобильные телефоны, ноутбуки, принтеры, цифровые фотоаппараты и даже холодильники, микроволновые печи, кондиционеры. Соединить можно все, что соединяется (то есть имеет встроенный микрочип Bluetooth). Технология стандартизирована, следовательно, проблемы несовместимости устройств от конкурирующих фирм быть не должно.

Bluetooth - это маленький чип, представляющий собой высокочастотный (2.4 - 2.48 ГГц) приёмопередатчик, работающий в диапазоне ISM (Industry, Science and Medicine; промышленный, научный и медицинский). Для использования этих частот не требуется лицензия (исключения рассмотрим ниже). Скорость передачи данных, предусмотриваемая стандартом, составляет порядка 720 Кбит/с в асимметричном режиме и 420 Кбит/с в полнодуплексном режиме. Обеспечивается передача трех голосовых каналов, но не видеосигнала. Энергопотребление (мощность передатчика) не должно превышать 10 мВт. Изначально технология предполагала возможность связи на расстоянии не более 10 метров. Сегодня некоторые фирмы предлагают микросхемы Bluetooth, способные поддерживать связь на расстоянии до 100 метров. Как радиотехнология, Bluetooth способна "обходить" препятствия, поэтому соединяемые устройства могут находиться вне зоны прямой видимости. Соединение происходит автоматически, как только Bluetooth-устройства оказываются в пределах досягаемости, причем не только по принципу точка - точка (два устройства), но и по принципу точка - много точек (одно устройство работает с несколькими

другими). Естественно, для реализации технологии Bluetooth на практике необходимо определенное программное обеспечение (ПО). Кстати, в новую версию операционной системы MS Windows Whistler встроена поддержка Bluetooth [16].

### Передача данных Bluetooth

В стандарте Bluetooth предусмотрена дуплексная передача на основе разделения времени (Time Division Duplexing - TDD). Основное устройство передает пакеты в нечетные временные сегменты, а подчиненное устройство – в четные.

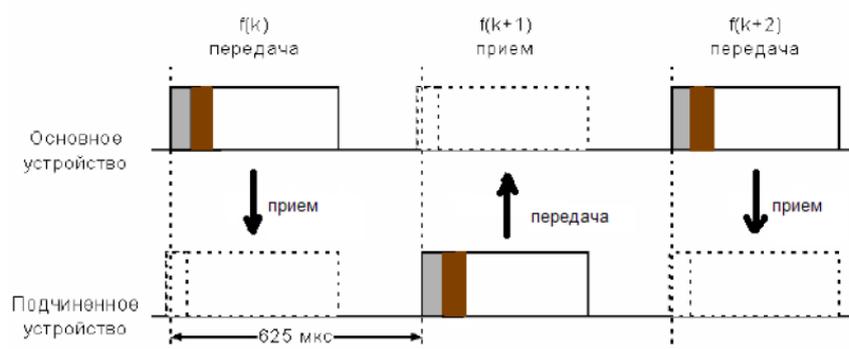


Рис. 5.87. Дуплексная передача с временным разделением

Пакеты в зависимости от длины могут занимать до пяти временных сегментов. При этом частота канала не меняется до окончания передачи пакета.

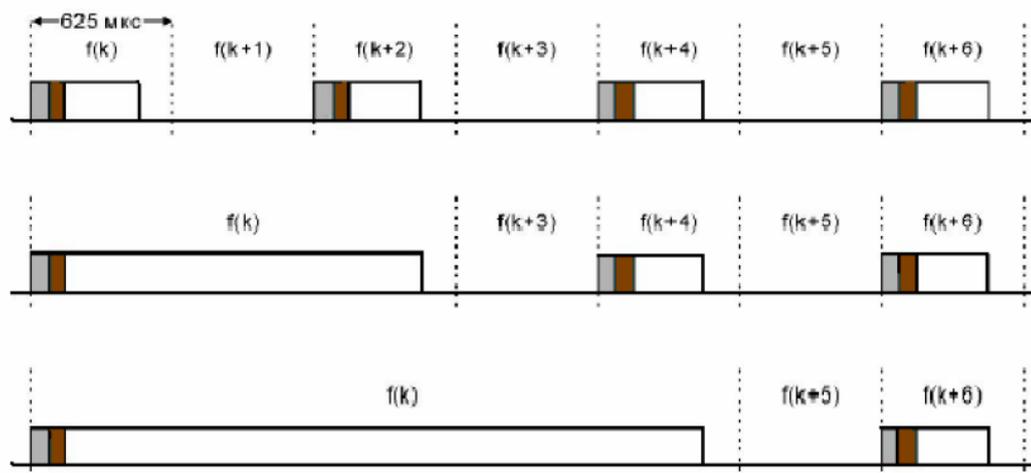


Рис. 5.88. Передача пакетов различной длины

Протокол Bluetooth может поддерживать асинхронный канал данных, до трех синхронных (с постоянной скоростью) голосовых каналов или канал с одновременной асинхронной передачей данных и синхронной передачей голоса. Скорость каждого голосового канала – 64 Кбит/с в каждом направлении, асинхронного в асимметричном режиме – до 723,2 Кбит/с в прямом и 57,6 кбит/с в обратном направлениях или до 433,9 Кбит/с в каждом направлении в симметричном режиме.

## Структура пакета

Стандартный пакет Bluetooth содержит код доступа длиной 72 бита, 54-битный заголовок и информационное поле длиной не более 2745 бит. Однако пакеты могут быть различных типов. Так, пакет может состоять только из кода доступа (в этом случае его длина равна 68 битам) или кода доступа и заголовка.

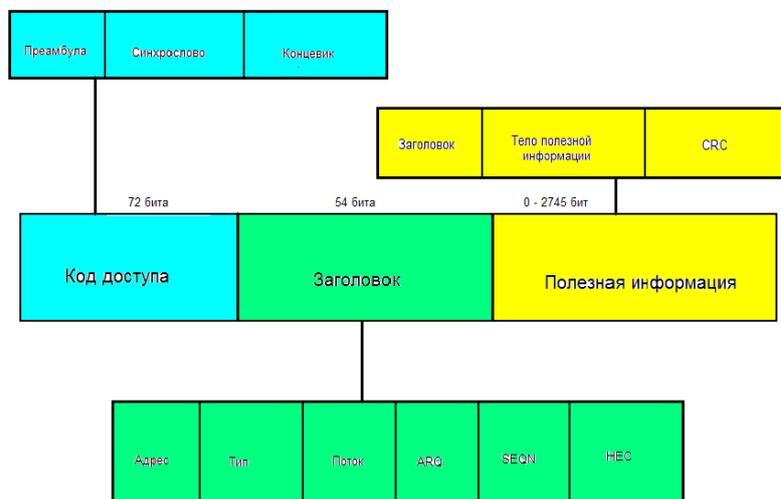


Рис. 5.89. Структура пакета

**Код доступа** идентифицирует пакеты, принадлежащие одной пикосети, а также используется для синхронизации и процедуры запросов. Он включает преамбулу (4 бита), синхрослово (64 бита) и концевик – 4 бита контрольной суммы.

**Заголовок** содержит информацию для управления связью и состоит из шести полей:

- Адрес (3 бита) - адрес активного элемента;
- Тип (4 бита) - код типа данных;
- Поток (1 бит) - управление потоком данных, показывает готовность устройства к приему;
- ARQ (1 бит) - подтверждение правильного приема;
- SEQN (1 бит) - служит для определения последовательности пакетов;
- HEC (8 бит) - контрольная сумма.

Заключительной частью общего формата пакета является **полезная информация**. В этой части есть два типа полей: поле голоса (синхронное) и поле данных (асинхронное). ACL пакеты имеют только поле данных, а SCO пакеты – только поле голоса. Исключением является пакет данных и голоса (Data Voice - DV), который имеет оба поля. Поле данных состоит из трех сегментов: заголовок полезной информации, тело полезной информации и возможно, CRC (Cyclic Redundancy Check) код.

- Заголовок полезной информации (8 бит). Только поля данных имеют заголовок полезной информации. Он определяет логический канал, управление потоком в логических каналах, а также имеет указатель длины полезной информации.

- Тело полезной информации (0-2721 бит). Тело полезной информации включает пользовательскую информацию. Длина этого сегмента указана в поле длины заголовка полезной информации.

- CRC (16 бит). От передаваемой информации вычисляется 16-битный циклический избыточный код (CRC), после чего он прикрепляется к информации.

Существует 4 типа контрольных пакетов: NULL, POLL, FHS, ID. Они одинаковые как для ACL, так и для SCO.

- ID-пакеты имеют длину 68 бит и применяются для пейджинга и запросов. Состоит из поля Код Доступа .

- NULL-пакеты (126 бит) состоят только из полей Код Доступа и Заголовок, играя роль подтверждений установления соединения или получения данных

- Тип POLL (126 бит) аналогичен предыдущему за исключением того, что POLL-пакеты обязывают получателя ответить.

- Пакеты FHS (366 бит) содержат информацию об адресе, классе устройства и тактовой частоте его передатчика

### Работа Bluetooth

Есть два основных состояния для устройств Bluetooth: Соединение (Connection) и Режим ожидания (Standby). Предусмотрено семь субсостояний, которые используются для добавления клиента или подключения к пикосети: **page**, **page scan**, **inquiry**, **inquiry scan**, **master response**, **slave response** и **inquiry response**.

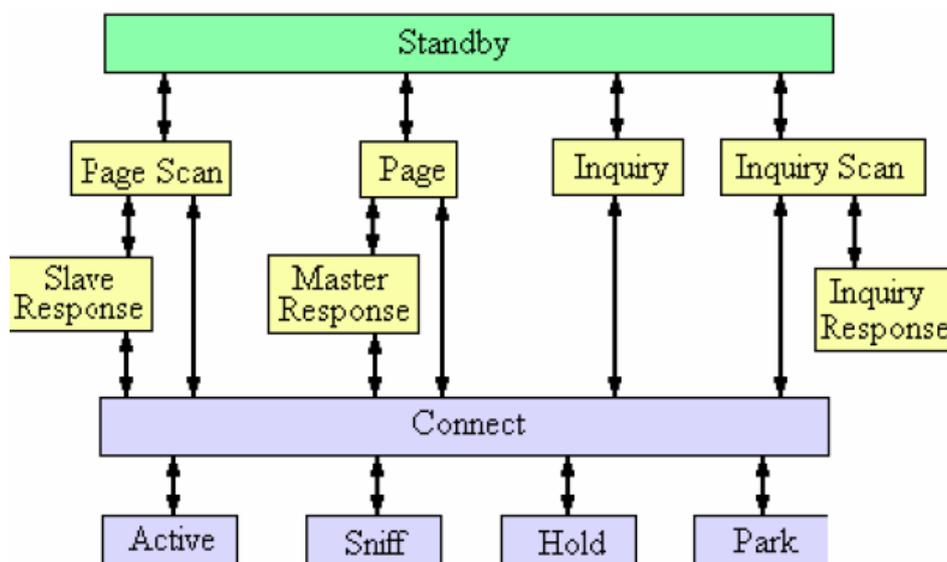


Рис. 5.60. Состояние соединений

Состояние Standby по умолчанию является режимом с пониженным энергопотреблением, работает только внутренний задающий генератор. В состоянии Соединения основной узел (master) и подчиненный (slave) могут обмениваться пакетами, используя код доступа к каналу.

Соединение между устройствами присходит так - если об удаленном устройстве ничего не известно, то используются процедуры inquiry и page. Если некоторая информация о устройстве все-таки есть, то достаточно процедуры page.

### Этап 1

Процедура **inquiry** позволяет устройству определить, какие приборы доступны, выяснить адреса и осуществить синхронизацию.

1.1 Посылаются пакеты inquiry и получаются отклики.

1.2 Если адресат, получивший пакет inquiry, находится в состоянии inquiry scan , тогда он способен принимать такие пакеты

1.3 Получатель переходит в состояние inquiry response и посылает отправителю пакет-отклик.

После того как процедура inquiry завершена, соединение может быть установлено с помощью процедуры paging.

### Этап 2

Процедура **paging** реализует соединение. Для осуществления этой процедуры необходим адрес. Устройство, выполняющее процедуру paging, автоматически становится хозяином этого соединения.

2.1 Посылается пакет paging

2.2 Адресат получит этот пакет (находится в состоянии page Scan)

2.3 Получатель посылает отправителю пакет-отклик (находится в состоянии Slave Response)

2.4 Инициатор посылает адресату пакет FHS (находится в состоянии Master Response).

2.5 Получатель посылает отправителю второй пакет-отклик (находится в состоянии Slave Response)

2.6 Получатель и отправитель устанавливают параметры канала заданные инициатором (находятся в состоянии Master Response & Slave Response)

После установления соединения основной узел (master) посылает пакет POLL, чтобы проверить, синхронизовал ли клиент свои часы и настроился ли на коммутацию частот. Клиент при этом может откликнуться любым пакетом. После успешного обнаружения устройств новое Bluetooth устройство получает набор адресов доступных Bluetooth устройств, после чего выясняет имена всех доступных Bluetooth устройств из списка. У

каждого Bluetooth устройства есть свой глобально уникальный адрес, но на уровне пользователя обычно используется не этот адрес, а имя устройства, которое может быть любым, и ему не обязательно быть глобально уникальным. Имя Bluetooth устройства может быть длиной до 248 байт, и использовать кодировку в соответствии с Unicode UTF-8 (при использовании UCS-2, имя может быть укорочено до 82 символов). Также у Bluetooth есть возможность автоматического подключения Bluetooth устройств к службам, предоставляемым другими Bluetooth устройствами. Поэтому, после того как имеется список имён и адресов, выполняется поиск доступных услуг, предоставляемых различными устройствами. Для поиска возможных услуг используется специальный протокол обнаружения услуг (Service Discovery Protocol - SDP).

Устройство Bluetooth при установлении соединения может работать в четырех режимах: **Active** (активный), **Hold** (удержание), **Sniff** (прослушивание) и **Park** (пассивный).

Таблица 5.12. Режимы работы Bluetooth

Название режима	Описание
Active	В активном режиме устройство Bluetooth участвует в работе канала. Основной узел (master) диспетчеризует обмены на основе запросов трафика, поступающих от участников. Кроме того, этот режим предусматривает регулярные обмены с целью синхронизации клиентов. Активные клиенты прослушивают домены master-to-slave пакетов. Если к активному клиенту нет обращений, он может пребывать в пассивном состоянии (sleep) до очередной передачи со стороны главного узла
Sniff	Устройства синхронизованные в рамках пикосети могут перейти в режим экономного расходования энергии, когда их активность понижается. В режиме <b>SNIFF</b> , подчиненное устройство прослушивает пикосеть с пониженной частотой. Этот режим имеет наивысшую скважность рабочего цикла (наименьшая экономия энергии) из 3 экономичных режимов ( <b>sniff</b> , <b>hold</b> и <b>park</b> )
Hold	Устройства синхронизованные в рамках пикосети могут перейти в режим экономного расходования энергии, когда их активность понижается. Основной узел пикосети может перевести клиента в режим <b>HOLD</b> , когда работает только внутренний таймер. Подчиненное устройство может запросить перевода в режим

	HOLD. Передача данных возобновляется мгновенно, когда устройство выходит из режима HOLD. Клиент имеет промежуточную скважность (промежуточный уровень экономии энергии) из указанных 3 режимов (sniff, hold и park)
Park	В режиме PARK, устройство еще синхронизовано в рамках пикосети, но не принимает участия в обменах. Пассивные устройства отказываются от своих MAC-адресов, прослушивают трафик главного модуля с целью ресинхронизации и отслеживают широковещательные сообщения. Данный режим имеет минимально возможную скважность (максимальная экономия энергии) из указанных 3 режимов (sniff, hold и park). Устройства, находящиеся в режиме park, должны посылать пакеты широковещательно, так как лишены собственного активного адреса.

### "Частотный конфликт"

Тот факт, что частотный диапазон 2.4 ГГц свободен от лицензирования, вносит определенные сложности в использование Bluetooth-устройств. В этом диапазоне работают также различные медицинские приборы, бытовая техника, беспроводные телефоны, беспроводные локальные сети стандарта IEEE. Вполне логично предположить, что они могут "конфликтовать" друг с другом. Во избежание интерференции с другими беспроводными устройствами Bluetooth работает по принципу скачкообразной перестройки частоты (1600 скачков в секунду). Переход с одной частоты на другую происходит по псевдослучайному алгоритму. Это позволяет "освободить" нужные другим устройствам частоты[3].

### Моделирование Bluetooth

Модель состоит из трех основных блоков:

- 1 Передатчик;
- 2 Канал;
- 3 Приемник.

Канал имеет три режима работы:

- 1 Нет канала;
- 2 AWGN канал;

Также имеется генератор сигнала стандарта 802.11, который как раз может конфликтовать с сигналами Bluetooth, для чего и применяется скачкообразная перестройка частоты.

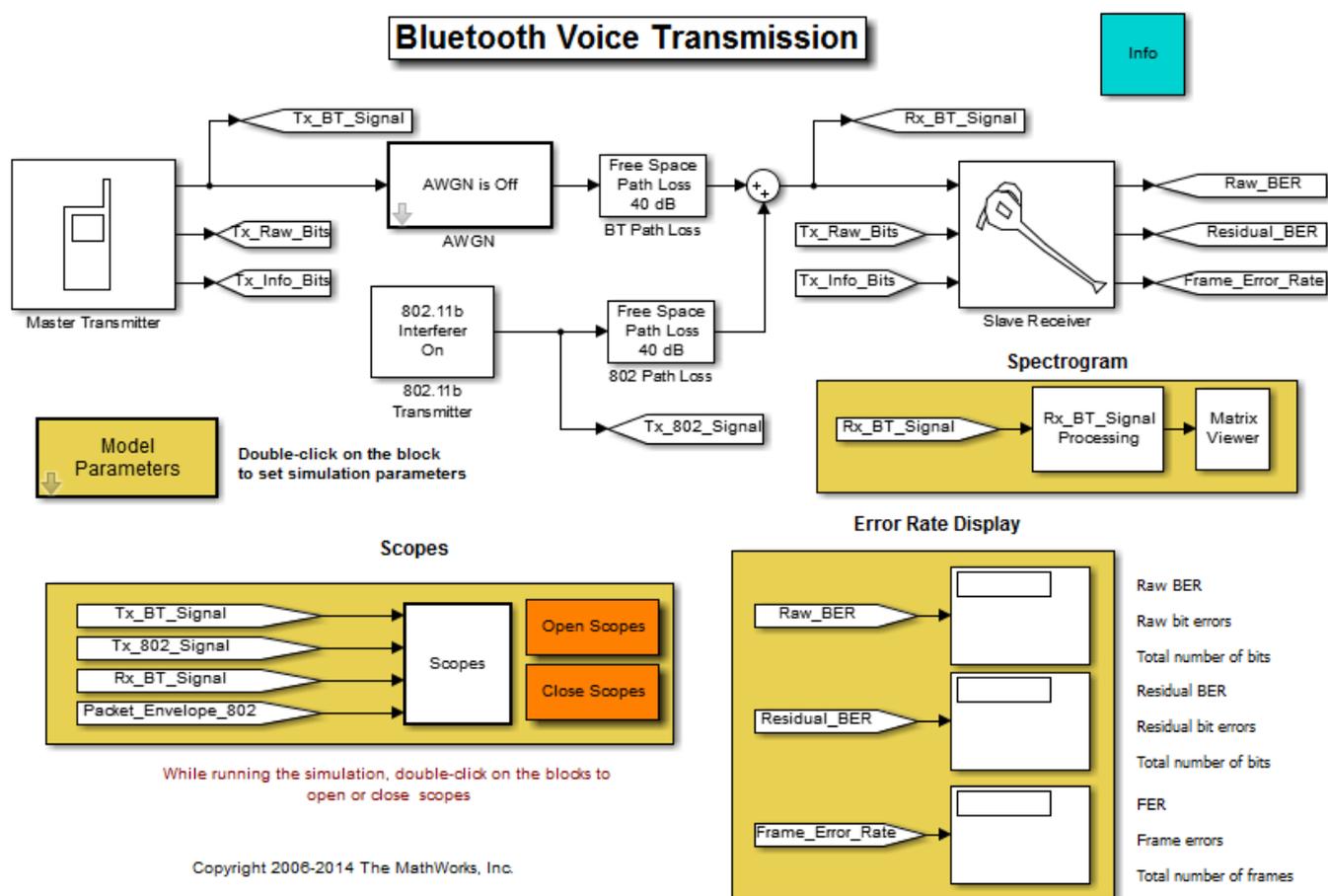


Рис. 5.61. Модель Bluetooth в MATLAB R2015b

### Результаты моделирования.

В результате моделирования данной схемы система строит три графика: спектр сигнала, временную форму сигнала и зависимость изменения рабочей частоты во времени (скачкообразная перестройка). На графике ниже представлен спектр Bluetooth сигнала в один из моментов времени. Одним из минусов метода перестройки частоты в системе Bluetooth являются задержки, которые хорошо видны на данной диаграмме при моделировании, также о них будет сказано ниже.

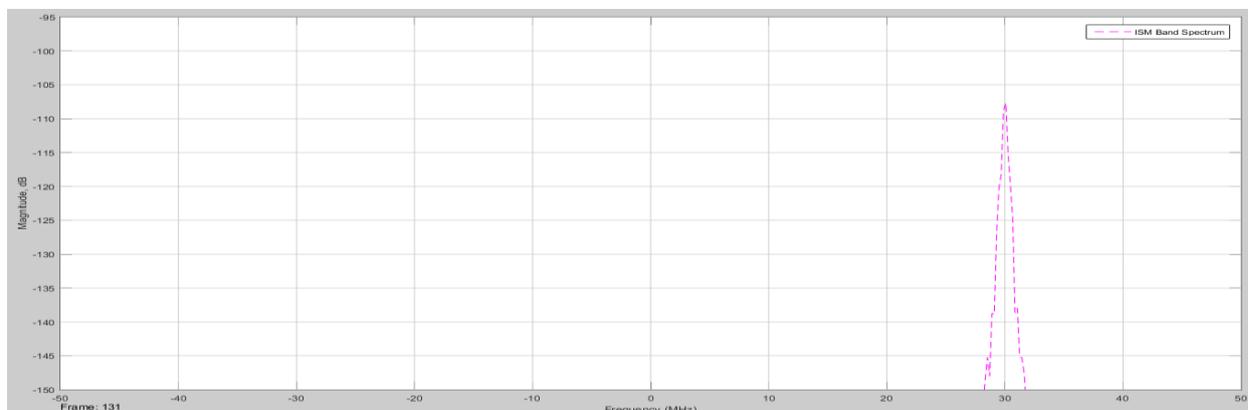


Рис. 5.61. Спектр Bluetooth без мешающего сигнала 802.11

Временная форма сигнала представляет просто набор битов, как и во многих современных системах связи. О значениях каждого бита(структуре кадра) была сказано ранее.

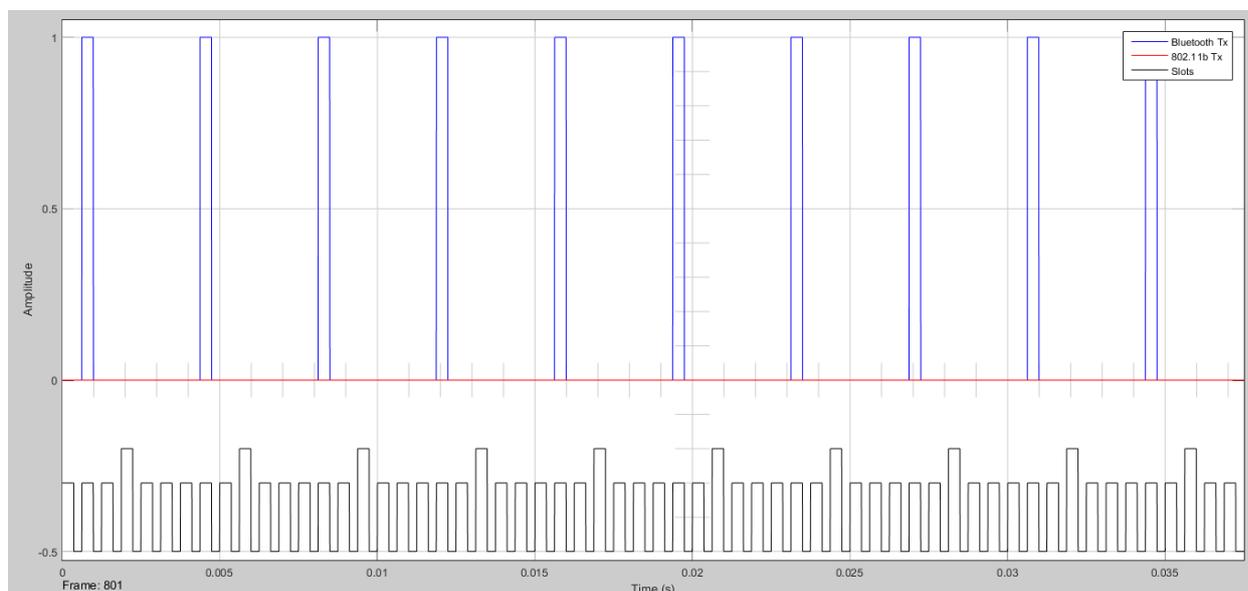


Рис. 5.62. Временная диаграмма Bluetooth без мешающего сигнала 802.11

На рисунке 5.63 хорошо видно изменение частоты от времени. На рисунке на оси абсцисс представлена частота, а на оси ординат время. Видно, что по оси времени перестройка с одной частоты на другую занимает определенное время, что относят к недостаткам системы Bluetooth.

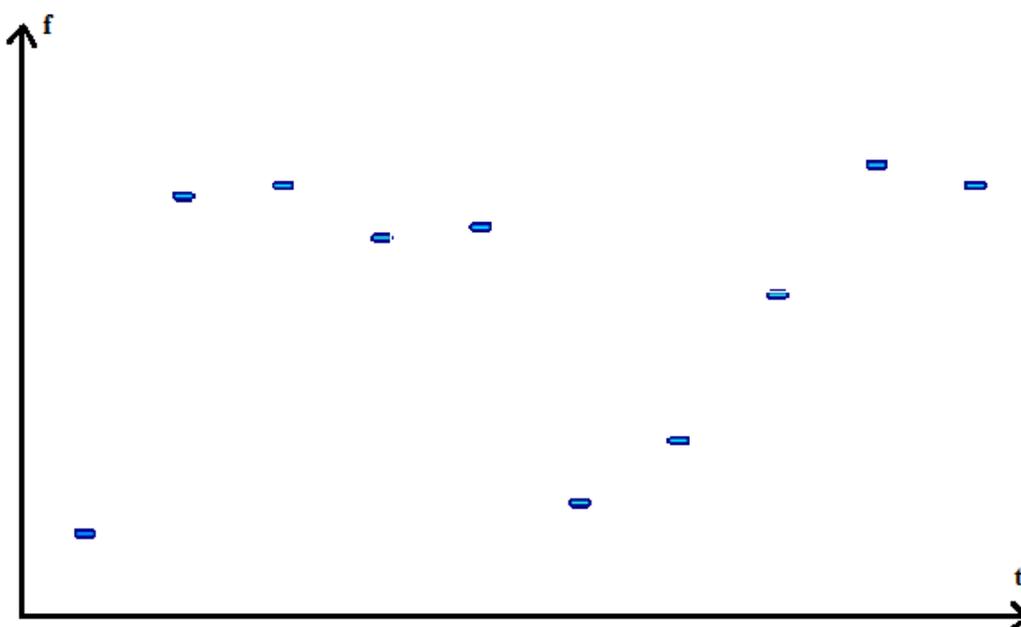


Рис. 5.63. Пример скачков частоты Bluetooth во времени без мешающего сигнала 802.11(WiFi)

На рисунке 5.64 представлен спектр вместе с мешающим сигналом. Здесь прекрасно видно, почему для построения системы Bluetooth был выбран алгоритм FHSS, который позволяет ему работать в одном диапазоне частот со стандартом 802.11 не мешая друг другу.

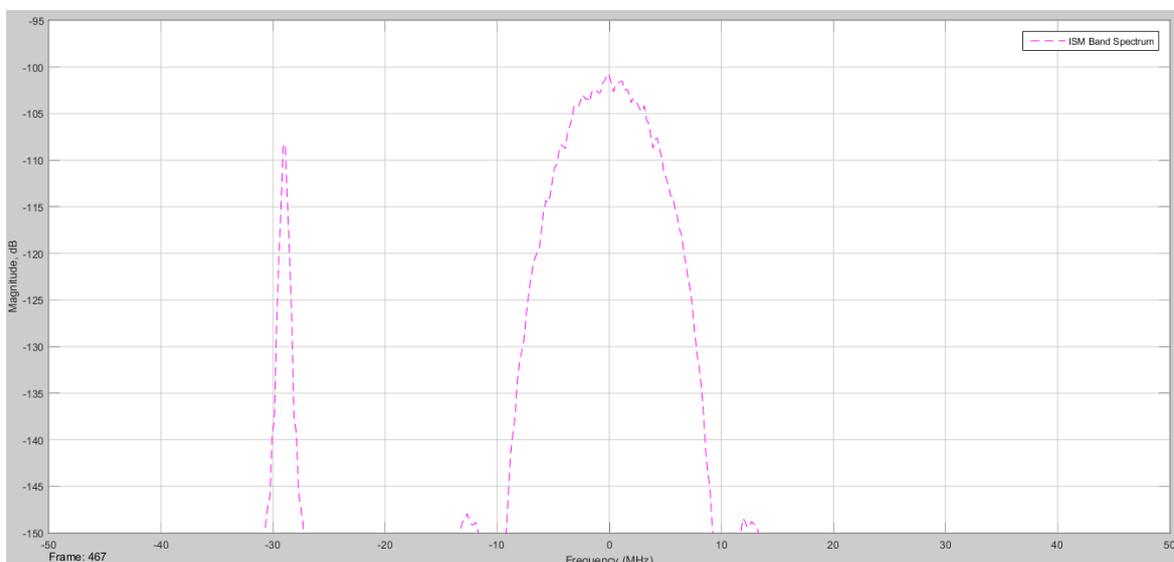


Рис. 5.64. Спектр Bluetooth с мешающим сигналом 802.11

Благодаря тому, что спектры сигналов разнесены в частотной области перекрытие их во временной, не играет большой роли, т.к. сигналы можно без проблем разделить.

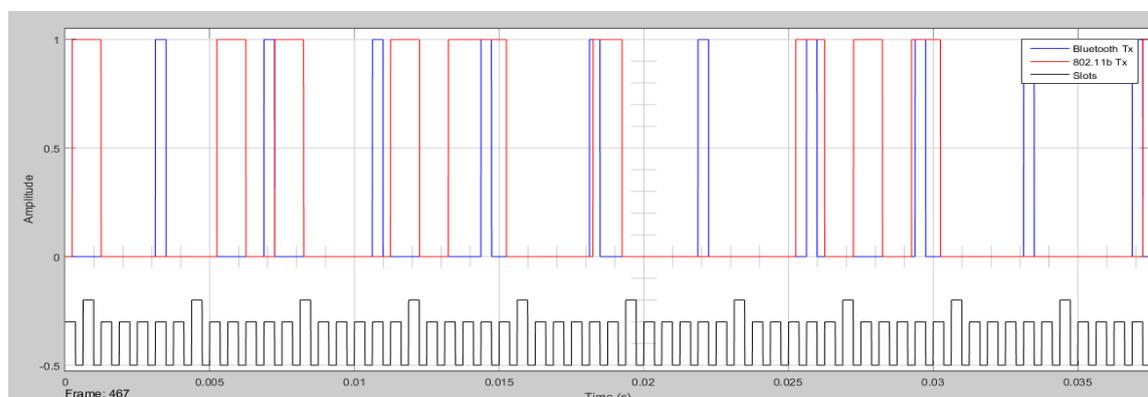


Рис. 5.65. Временная диаграмма Bluetooth с мешающим сигналом 802.11

Из рисунка ниже прекрасно видно, что во время работы устройства стандарта 802.11 рабочая частота системы Bluetooth находится достаточно далеко по спектру, а в некоторые моменты занимает свободный диапазон стандарта 802.11

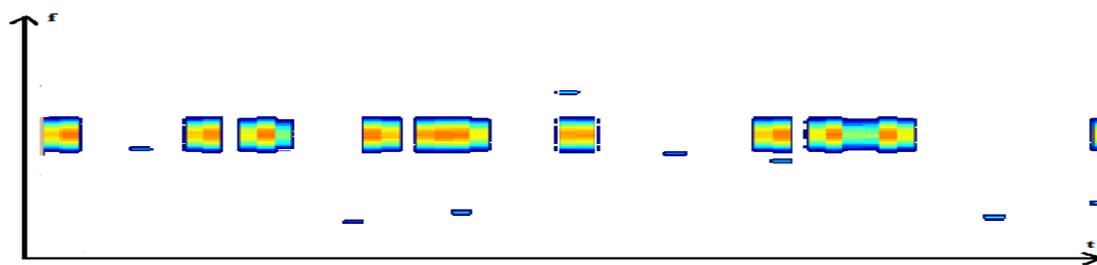


Рис. 5.66. Пример скачков частоты Bluetooth во времени с мешающим сигналом 802.11

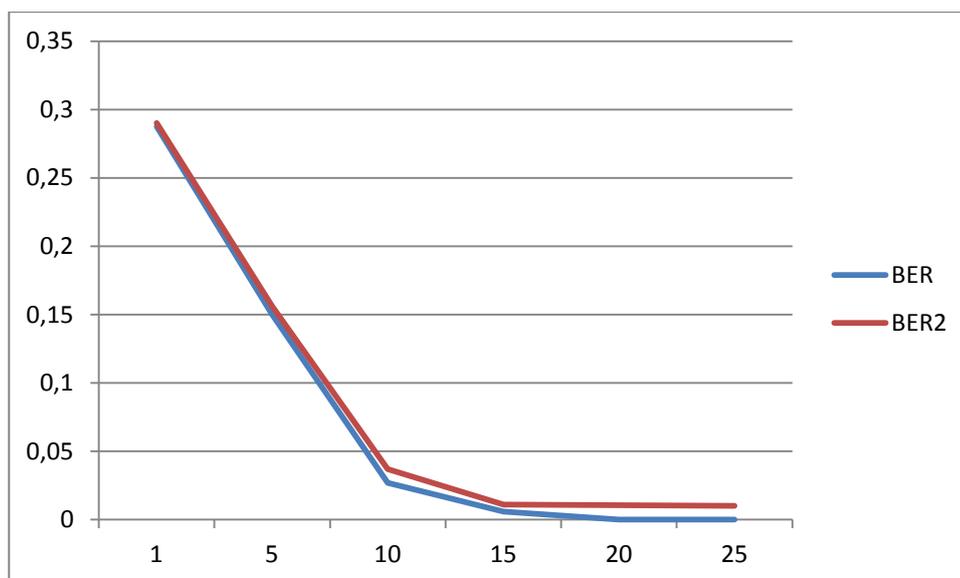


Рис. 5.67. Зависимость BER от SNR. Красным цветом (верхняя кривая) выделен график при включенном мешающем сигнале 802.11

Благодаря алгоритму FHSS система не сильно подвержена влиянию других стандартов передачи данных, работающих в том же диапазоне частот.

В разделе пока не технология передачи данных 802.15.1 Bluetooth, а также использована модель передачи звука по такой системе в системе Simulink.

С помощью модели были построены временная диаграмма сигнала, спектр и FHSS спектр сигнала BLUETOOTH при воздействии мешающего сигнала и без него. Также была построена зависимость BER от SNR.

На основе графиков зависимости BER от SNR (рисунок 5.67) видно, что мешающий сигнал 802.11 оказывает незначительное влияние на передачу данных. На рисунке 5.66 видно, что во время передачи сигнала 802.11, сигнал Bluetooth совершает скачок на другую частоту, что также хорошо видно на рисунке 5.64.

## 5.6. Имитационное моделирование системы мобильной связи стандарта IEEE 802.16 (WiMAX) на базе ПО MATLAB [16, 21]

Существующие системы проводной цифровой связи уже не могут в полной мере удовлетворять растущим потребностям высокоскоростного широкополосного доступа. Важнейшими их недостатками являются длительные сроки прокладки, сложности расширения, высокие затраты, проблема "последней мили". Основной и является так называемая проблема "последней мили". Высокоскоростные цифровые соединительные линии DSL (Digital Subscriber Line) не снимают этой проблемы.

Технология WiMAX позволяет разрешить эту проблему в кратчайшие сроки, так как не требует прокладки соединительных линий к зданиям. Значительно проще развернуть по

городу сеть базовых станций (наподобие сети станций сотовой связи). Каждая базовая станция в типовом варианте покрывает зону радиусом 6—8 км (возможны зоны радиусом до 30—50 км). В этой зоне каждая базовая станция (BS) по схеме "точка-многоточка" способна передавать/принимать сигналы от сотен зданий, внутри которых находится телекоммуникационное оборудование пользователей.

Под аббревиатурой WiMAX (Worldwide Interoperability for Microwave Access) понимается технология операторского класса с высоким качеством сервиса, которая основана на семействе стандартов IEEE 802.16, разработанных международным институтом инженеров по электротехнике и электронике (IEEE). Обеспечивает мультисервисность, гибкое распределение частот, задание приоритетов различным видам трафика, возможность обеспечения разного уровня качества (QoS), поддержка интерфейсов IP. Эта технология позволяет параллельно передавать голос, мультимедийную информацию и цифровые данные по одному каналу связи. Важным преимуществом является возможность быстро наращивать емкость и расширять территорию связи.

Технология WiMAX представляет прекрасную возможность обеспечивать беспроводной доступ всем пользователям цифрового оборудования, включая оборудование беспроводных локальных сетей, технологии Wi-Fi, к глобальным сетям, являясь связующим звеном между локальными сетями и глобальными сетями.

### **Общие принципы построения сетей WiMAX**

#### **Стандарты IEEE 802.16. Форум WiMAX.**

При переходе к созданию систем широкополосного радиодоступа с интеграцией услуг стало понятно, что основополагающие принципы, заложенные в беспроводные системы на предыдущих этапах, нуждаются в существенной корректировке. На сигнальном уровне первостепенное значение приобрело оптимальное использование спектрального ресурса радиоканала при любых соотношениях "скорость - помехоустойчивость". На уровне протоколов стало необходимым обеспечивать заданный уровень качества обслуживания каждому абоненту сети.

Основным преимуществом сетей WiMAX по сравнению с другими технологиями, призванными решать аналогичные задачи, является относительно быстрое развертывание систем на достаточно больших территориях без проведения работ по прокладке кабеля и предоставление конечным пользователям каналов связи в единицы Мбит/с, что особенно актуально для мест с неразвитой сетевой инфраструктурой. Основным конкурентом сетей WiMAX являются системы связи четвертого поколения LTE E UTRA.

На сегодняшний день беспроводные сети городского масштаба представлены следующими стандартами:

- IEEE 802.16e-2005, 2009 (WiMAX);
- ETSI HiperMAN;
- IEEE 802.20 (WBWA).

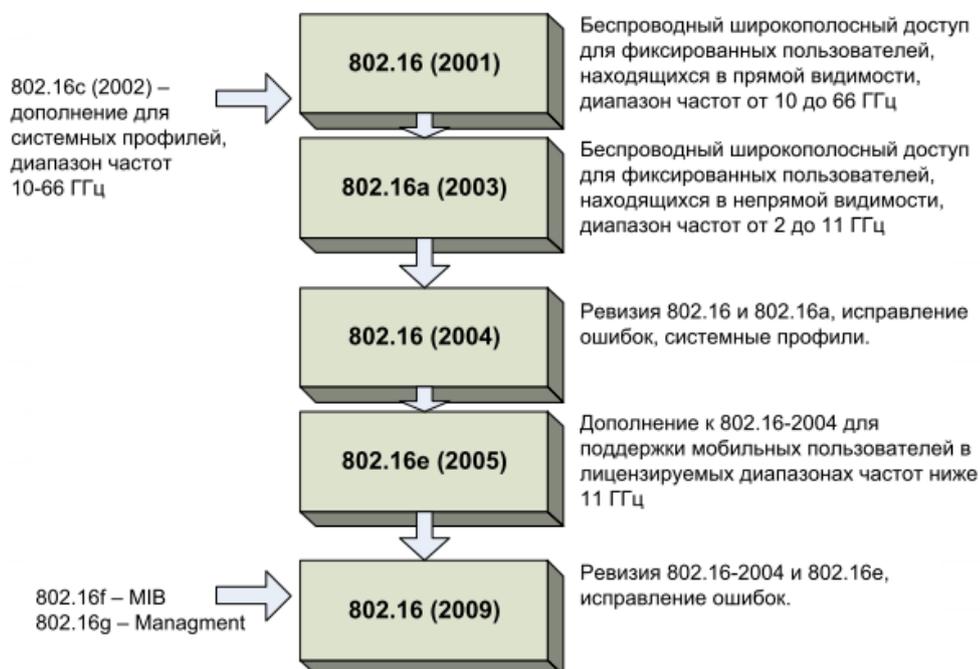


Рис. 5.68. Эволюция стандартов IEEE 802.16

Необходимо различать стандарты связи серии IEEE 802.16 (рисунок 5.68) и форум WiMAX (рисунок 5.69). Стандарты серии IEEE 802.16 — это множество стандартов, определяющих беспроводные сети городского масштаба (WMAN — Wireless Metropolitan Area Network), разработаны для обеспечения беспроводным широкополосным доступом стационарных и мобильных пользователей. Форум WiMAX является некоммерческой организацией для продвижения и сертификации устройств беспроводного широкополосного доступа, основанных на согласованном стандарте IEEE 802.16/ETSI HiperMAN. Сотрудничает с поставщиками услуг, производителями оборудования, производителями тестового оборудования, сертификационными лабораториями и поставщиками программно-аппаратных ресурсов для обеспечения соответствия ожиданиям заказчика и государственным стандартам.

Стандарты серии IEEE 802.16 определяет радиointерфейс для систем широкополосного беспроводного доступа (уровни MAC и PHY, рисунок 5.70) с фиксированными и мобильными абонентами в диапазоне частот 1-66 ГГц, рассчитанных на внедрение в городских распределенных беспроводных сетях операторского класса. Сети, построенные на основе этих стандартов, займут промежуточное положение между локальными сетями (IEEE 802.11x) и региональными сетями (WAN), где планируется применение разрабатываемого стандарта IEEE 802.20. Указанные стандарты совместно со

стандартом IEEE 802.15 (PAN — Personal Area Network) и IEEE 802.17 (мосты уровня MAC) образуют иерархию стандартов беспроводной связи.

Для обеспечения работоспособности систем в диапазоне 10-66 ГГц, вследствие относительно малой длины волны, требуется наличие прямой видимости между передатчиком и приемником. В таких условиях при анализе канала связи многолучевостью среды можно пренебречь. Данные передаются на одной несущей. Ширина полосы частот одного канала составляет 25 или 28 МГц, скорость передачи данных до 120 Мбит/с.



Рис. 5.69. Стандарты серии IEEE 802.16 и форум WiMAX

В диапазоне частот 1-11 ГГц за счет увеличения длины волны возможен сценарий взаимодействия передатчика и приемника в условиях отсутствия прямой видимости. При этом необходимо применять более сложные (по сравнению с системами, функционирующими в диапазоне частот 10-66 ГГц) методы регулировки мощности, различные способы борьбы с межсимвольной интерференцией. Для передачи данных используется одна или множество несущих (сигналы с OFDM).

Структура стандартов IEEE 802.16 представлена на рисунке 5.70. Стандарты описывают MAC- и PHY- уровни семиуровневой эталонной модели взаимодействия открытых систем (ЭМОС). При этом уровень MAC делится на подуровни конвергенции, общей части и безопасности.

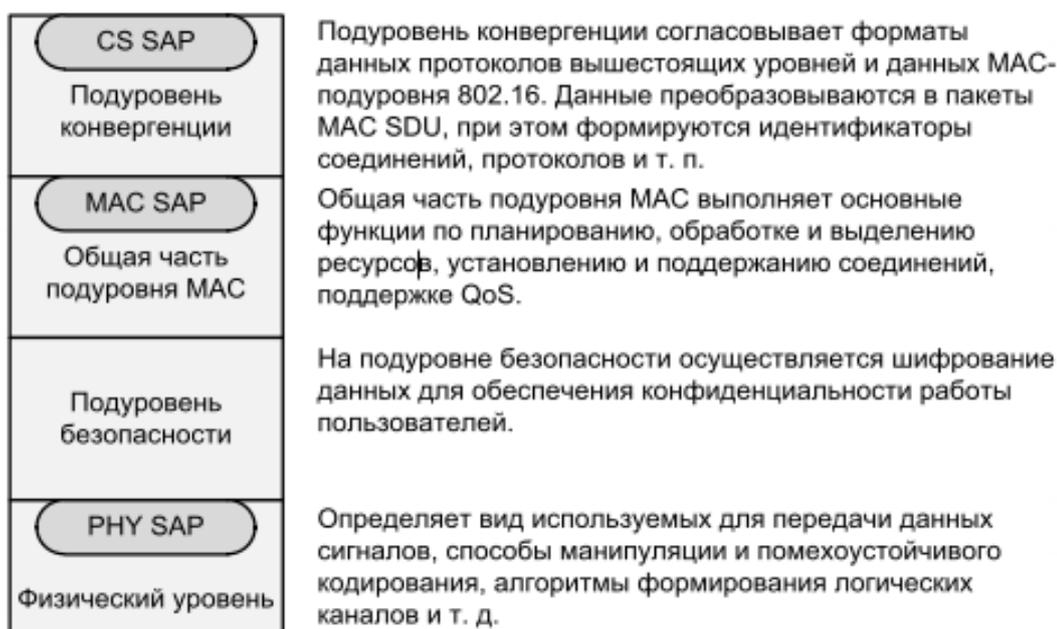


Рис. 5.70. Структура стандартов IEEE 802.16

## Сетевой уровень систем WiMAX

### Архитектура сетей WiMAX IEEE 802.16

Базовая станция (БС, BS — Base Station) размещается в здании или на вышке и осуществляет связь с абонентскими станциями (АС, SS — Subscriber Station) по схеме — «точка – мультиточка» (Point to Multipoint — PMP). Возможен сеточный режим связи (Mesh — сетка связей — «точка – точка» — PTP), когда любые клиенты (АС) могут осуществлять связь между собой непосредственно, а антенные системы, как правило, являются ненаправленными. БС предоставляет соединение с основной сетью и радиоканалы к другим станциям. Радиус действия БС может достигать 30 км (в случае прямой видимости) при типовом радиусе сети 6–8 км. АС может быть радиотерминалом или повторителем, который используется для организации локального трафика. Трафик может проходить через несколько повторителей, прежде чем достигнет клиента. Антенны в этом случае являются направленными.

Канал связи предполагает наличие двух направлений передачи: восходящий канал (АС – БС, uplink) и нисходящий (БС – АС, downlink). Эти два канала используют разные неперекрывающиеся частотные диапазоны при частотном дуплексе и различные интервалы времени при временном дуплексе.

Простейший способ представления архитектуры сетей WiMAX заключается в их описании как совокупности БС, которые располагаются на крышах высотных зданий или вышках, и клиентских приемо-передатчиков (рисунок 5.71).



Рис. 5.71. Схематическое изображение сети WiMAX

Радиосеть обмена данными между БС и АС работает в СВЧ-диапазоне от 2 до 11 ГГц. Такая сеть в идеальных условиях может обеспечить техническую скорость передачи информации до 75 Мбит/с и не требует того, чтобы БС находилась на расстоянии прямой видимости от пользователя.

Диапазон частот от 10 до 66 ГГц используется для установления соединения между соседними базовыми станциями при условии, что они располагаются в зоне прямой видимости друг от друга. Так как в городской среде это условие может оказаться невыполнимым, связь между базовыми станциями иногда организуют посредством прокладки кабелей.

При более детальном рассмотрении сеть WiMAX можно описать как совокупность беспроводного и базового (опорного) сегментов. Первый описывается в стандарте IEEE 802.16, второй определяется спецификациями WiMAX Forum. Базовый сегмент объединяет все аспекты, не относящиеся к абонентской радиосети, то есть связь базовых станций друг с другом, связь с локальными сетями. Базовый сегмент основывается на IP-протоколе и стандарте IEEE 802.3-2005 (Ethernet). Однако само описание архитектуры в части, не относящейся к беспроводной клиентской сети, содержится в документах WiMAX Forum, объединенных под общим названием – "Network Architecture".

Таблица 5.13. Основные режимы для стандарта IEEE 802.16 в РФ

Диапазон частот, ГГц	Разрешенные полосы частот, МГц	Общая ширина выделенных полос, МГц	Тип беспроводного доступа
2,5	2500 – 2530 2560 – 2570 2620 – 2630 2660 – 2670 2680 – 2690	70	мобильный
3,5	3400 – 3450 3500 – 3550	100	фиксированный
5	5150 – 5350 5650 – 5725 5725 – 6425	975	фиксированный

В этих спецификациях к сетям WiMAX предъявляются такие требования, как независимость архитектуры от функций и структуры транспортной IP-сети. В то же время, должны обеспечиваться услуги, основанные на применении IP-протокола, а также мобильная телефония на основе VoIP и мультимедийные услуги. Обязательным является условие поддержки архитектурой протоколов IPv4 и IPv6. Сети WiMAX должны быть легко масштабируемыми и гибко изменяемыми и основываться на принципе декомпозиции (строиться на основе стандартных логических модулей, объединяемых через стандартные интерфейсы). Свойства масштабируемости и гибкости необходимо обеспечивать по таким эксплуатационным характеристикам, как плотность абонентов, географическая протяженность зоны покрытия, частотные диапазоны, топология сети, мобильность абонентов. Сети WiMAX должны поддерживать взаимодействие с другими беспроводными или проводными сетями. Большое значение имеет способность обеспечивать различные уровни качества обслуживания QoS.

#### **Базовая модель сетей WiMAX IEEE 802.16 – 2009**

Базовая модель (БМ) сети WiMAX – это логическое представление сетевой архитектуры WiMAX. Термин "логическое" в данном случае означает, что модель рассматривает набор стандартных логических функциональных модулей и стандартных интерфейсов (точек сопряжения этих модулей).

БМ включает три основных элемента: множество абонентских (мобильных) станций (АС), совокупность сетей доступа (ASN, Access Service Network) и совокупность сетей

подключения (CSN, Connectivity Service Network). Кроме того, в БМ входят так называемые базовые точки (R1...R8), через которые происходит сопряжение функциональных модулей. Базовая модель WiMAX сети изображена на рисунке 5.72.

Сеть (сети) ASN принадлежит провайдеру сети доступа (NAP, Network Access Provider) — организации, предоставляющей доступ к радиосети для одного или нескольких сервис-провайдеров WiMAX (NSP, Network Service Provider). В свою очередь, сервис-провайдер WiMAX — организация, предоставляющая IP-соединения и услуги WiMAX конечным абонентам. В рамках данной модели именно сервис-провайдеры WiMAX заключают соглашения с интернет-провайдерами, операторами других сетей доступа, соглашения о роуминге. Сервис-провайдеры по отношению к абоненту могут быть домашними и гостевыми, каждый со своей сетью CSN.

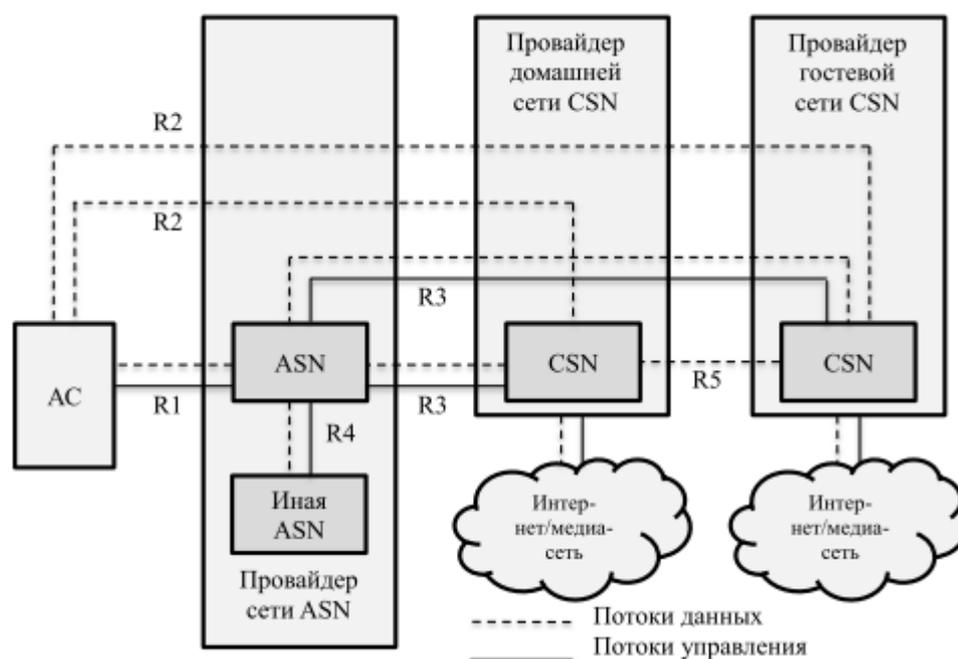


Рис. 5.72. Базовая модель WiMAX-сети

Сеть доступа ASN представляет собой множество базовых станций беспроводного доступа по стандарту IEEE 802.16 и шлюзов для связи с транспортной IP-сетью. Логическая модель доступа представлена на рисунке 5.73. Фактически эта сеть связывает радиосеть IEEE 802.16 и IP-сеть. ASN включает как минимум одну БС и как минимум один ASN-шлюз. Но и базовых станций, и шлюзов в одной ASN может быть несколько, причем одна БС может быть логически связана с несколькими шлюзами. БС в рамках данной модели — это логическое устройство, поддерживающее набор протоколов IEEE 802.16 и функции внешнего сопряжения.

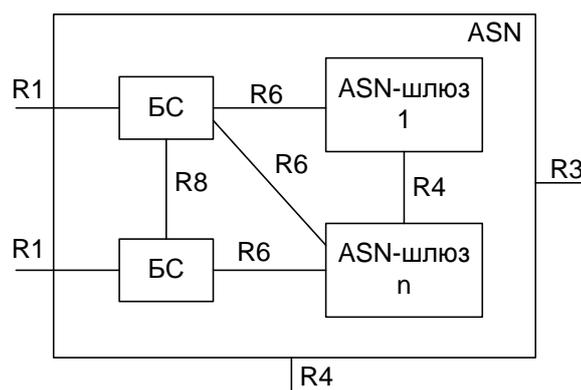


Рис. 5.73. Логическая модель сети доступа ASN

Сети ASN отвечают за передачу данных AC на втором логическом уровне и передают служебные и информационные сообщения AC, поступающие от сетей CSN. Состояния готовности к подключению и обеспечение целостного включения в сеть WiMAX для каждой AC накладывает следующие функциональные требования к ASN:

- обнаружение сети и выбор предпочитаемого абонентом WiMAX NSP;
- подключение AC к сети в соответствии с правилами второго логического уровня IEEE 802.16;
- реализация транслирующей функции для организации подключений AC на третьем логическом уровне (выделение IP-адресов);
  - управление радиоресурсами сети;
  - управление групповыми и ширококвещательными рассылками;
  - поддержка мобильности с привязкой к ASN;
  - поддержка внешних агентов для мобильности с привязкой к ASN;
  - поддержка туннелей ASN – CSN;
  - управление персональными вызовами и службами для определения местоположения абонентов;
    - продвижение данных;
    - авторизация сервисных потоков;
    - поддержка уровней QoS;
    - управление доступом и контроль соблюдения правил.

Перечисленные функции сетей ASN распределяются между БС и шлюзами ASN в соответствии с профилями, определенными в документе — «Release 1.0» форума WiMAX. На сегодняшний момент описано три таких профиля: А, В и С. На рисунке 5.107 представлена архитектура профиля С. Все профили обеспечивают функциональную совместимость между CSN и другими ASN через специальные опорные точки.

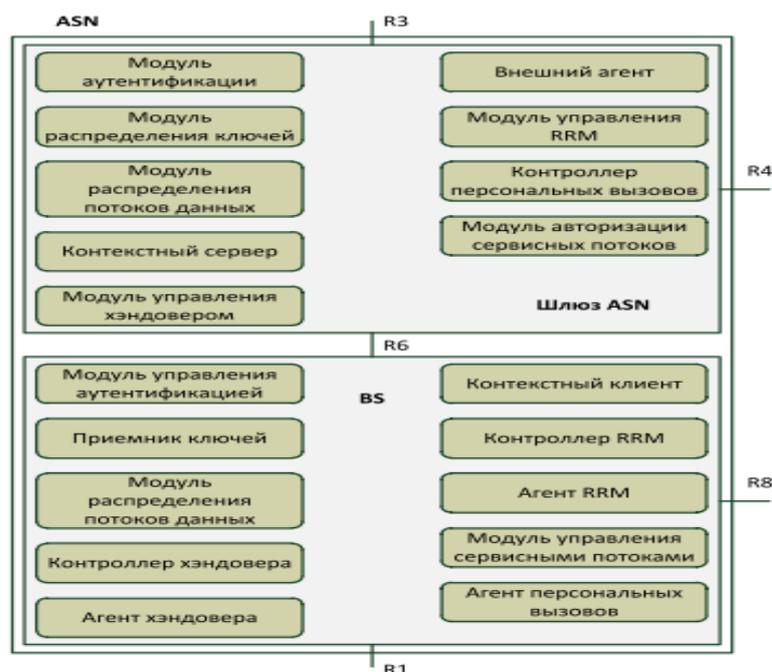


Рис. 5.74. Архитектура профиля С

Профиль В подразумевает полную свободу производителя, ему соответствует как концентрация всех функций в одном устройстве, так и их произвольное распределение. Профили А и С чрезвычайно похожи. Различие состоит в том, что в профиле А функции контроллера радиоресурсов и управления хэндовером отнесены к ASN-шлюзу, а в профиле С — к базовой станции. Несмотря на это незначительное различие, профиль А был официально закрыт в 2007 г. на сессии WiMAX Forum, а общепризнанным стандартом стал профиль С.

Шлюз ASN (шлюз радиоподсети) — это логическое устройство, связывающее базовые станции одной ASN с другими сетями доступа и с сетью подключения CSN. Шлюз ASN обеспечивает связность как на уровне каналов передачи данных, так и на уровне управления. Шлюз ASN может быть представлен как совокупность двух групп функциональных элементов: блока решения (DP, Decision Point) и блока исполнения (EP, Enforcement Point).

Блок решения шлюза ASN выполняет функции управления инвариантно к виду используемых радио интерфейсов. Шлюз ASN может содержать модуль, управляющий аутентификацией и распределением ключей, для реализации инфраструктуры AAA (Authentication, Authorisation, Accounting). Инфраструктура AAA проверяет полномочия пользователей в процессе их подключения или переподключения к сети по алгоритму EAP и создает безопасную среду с разделяемыми между АС и БС ключами. Инфраструктура AAA также отвечает за ведение статистики по абонентам.

Шлюз ASN реализует управление текущим профилем АС и осуществляет контроль за соблюдением правил CSN. Управление профилем АС включает в себя определение и

сохранение аутентификационных данных, параметров безопасности, а также настроек АС, необходимых при пере-ключении АС на другую БС.

В процессе хэндовера (процесс передачи абонента во время вызова или сессии передачи данных от одной базовой станции к другой) шлюз ASN осуществляет переключение канала передачи данных к требуемой БС. Шлюз ASN также обеспечивает целостность передаваемых данных для минимизации задержек распространения и снижения количества потерянных пакетов. Шлюз ASN содержит внешний агент для поддержки хэндовера на третьем логическом уровне и взаимодействия с домашним агентом в случае перехода АС от одной сети ASN к другой. Контроллер персональных вызовов и регистр местоположения в шлюзе ASN осуществляют поддержку персональных вызовов и работы АС в режиме ожидания.

Блок исполнения шлюза ASN отвечает за фильтрацию пакетов, организацию туннелей, управлением доступом, контроль за соблюдением правил, поддержку уровней QoS и продвижение данных. Блок исполнения может поддерживать протоколы IPv4/v6 и протоколы групповой и индивидуальной маршрутизации. Шлюз ASN является центральным элементом сетей ASN, выполняет функции интерфейса к внешним сетям и реализует различные службы IP.

Последним рассматриваемым компонентом ASN является устройство, называемое внешним агентом (FA, Foreign Agent). Это — маршрутизатор, отслеживающий принадлежность АС к той или иной БС в каждый момент времени для распределения информационных потоков

Сеть подключения CSN — это сеть оператора WiMAX. Именно в ней реализуются функции управления авторизацией, аутентификацией и доступом, подключение абонентов WiMAX к глобальным IP-сетям, предоставление таких услуг, как IP-телефония, доступ к телефонным сетям общего пользования, доступ в интернет и частные сети. Базовая модель сети WiMAX допускает, что одной сетью доступа ASN могут пользоваться несколько сервис-провайдеров, каждый со своей CSN. Напротив, одна CSN может подключаться к сетям доступа разных провайдеров доступа.

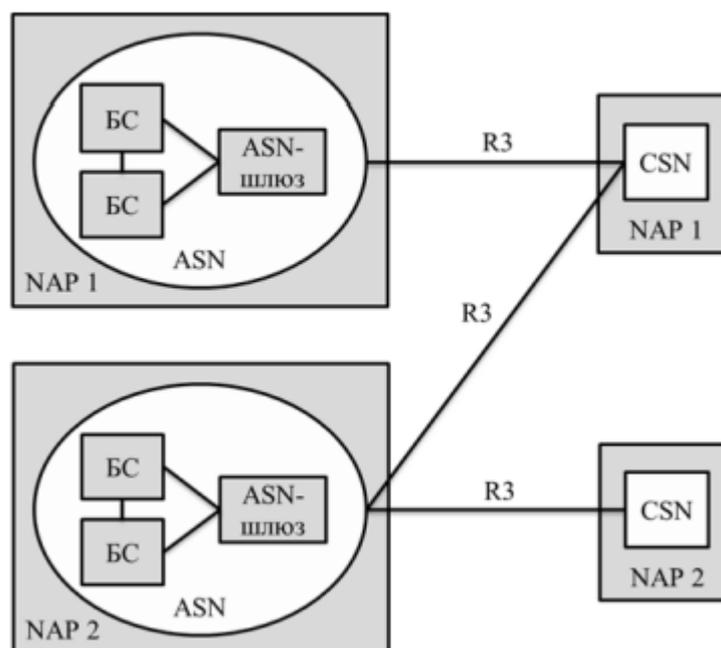


Рис. 5.75. Модель взаимодействия сетей доступа и сетей подключения различных провайдеров

Сеть CSN представляет собой набор сетевых функций, которые обеспечивают абонента WiMAX различными услугами IP. CSN может выполнять следующие функции:

- выделение IP-адресов;
- предоставление доступа к интернету;
- реализацию сервера или прокси-сервера для инфраструктуры AAA;
- управление доступом и контроль за соблюдением правил на основе профилей пользовательской подписки;
- поддержка туннелей ASN – CSN;
- учет оплаты абонентских услуг;
- поддержка туннелей между CSN для роуминга;
- поддержка перемещений абонента между различными ASN;
- поддержка специальных служб WiMAX таких, как службы, основанные на определении местоположения абонента (LBS — location based services), одноранговые службы, подключение спецслужб, групповые широковещательные рассылки.

Процедуры аутентификации, авторизации и контроля доступа выполняет устройство, называемое AAA-сервером. При этом под аутентификацией пользователя понимается проверка его подлинности и возможности доступа в сеть, под авторизацией — выделение ему ресурсов сети в соответствии с услугами, на которые он подписан и под контролем доступа — подсчет потребленных пользователем ресурсов (количество времени или размер переданных данных) для формирования счета за пользование сетью. Также каждая CSN

содержит домашний агент (НА, Home Agent), который является роутером-шлюзом, расположенным на границе WiMAX сети и внешних сетей.

Опорные точки в рамках базовой модели сети WiMAX — это каналы связи между базовыми модулями. Они определяют собой стандартные интерфейсы, причем не обязательно физические, если соединяемые опорной точкой модули конструктивно находятся в одном устройстве. Назначение опорных точек описано в таблице 2.2. Опорные точки необходимы для того, чтобы улучшить функциональную совместимость устройств, выпускаемых различными производителями.

БС WiMAX является логическим объектом, который реализует радио интерфейс и интерфейс к IP-сетям. БС реализует физический уровень и MAC-подуровень стандарта IEEE 802.16, а также одну или несколько функций ASN для взаимодействия со шлюзом ASN и другими БС. Экземпляр БС стандарта IEEE 802.16 поддерживает один сектор с одной несущей частотой, одна БС может взаимодействовать с несколькими шлюзами ASN для балансировки нагрузки и/или обеспечения избыточности передаваемых данных. Физически БС может включать в себя несколько объектов БС, так как БС определена как логический объект. Основным элементом БС является планировщик задач, осуществляющий управление частотно-временными ресурсами сети при организации обмена в нисходящем и восходящем каналах связи.

Таблица 5.14. Назначение опорных точек БМ сети WiMAX IEEE 802.16

Опорная точка	Назначение
R1	Канал связи между мобильной станцией и сетью доступа ASN, беспроводный интерфейс стандарта IEEE 802.16.
R2	Канал между МС и шлюзом ASN или CSN, включает процедуры аутентификации, авторизации АС и IP-конфигурирования. Данная точка представляет собой логический интерфейс, который обычно используется для аутентификации, авторизации конфигурирования IP и управления мобильность АС.
R3	Содержит набор протоколов для взаимодействия ASN и CSN для реализации процедур AAA и контроля за выполнением правил сети. Данная опорная точка применяется, когда АС находится в гостевой сети и обменивается информацией с домашней сетью.

R4	Канал связи между ASN-шлюзами различных ASN-сетей или одной ASN-сети. Данная опорная точка используется при перемещении АС между различными ASN. В отличие от предыдущих точек, между точками R4 могут создаваться защищенные соединения, в том числе, с использованием протоколов IP-Sec или SSL VPN.
R5	Канал связи между сетью домашнего и гостевого сервис-провайдера.
R6	Интерфейс между БС и ASN.
R7	Виртуальный канал между блоками DP и EP в ASN-шлюзах.
R8	Канал связи непосредственно между БС. Данная опорная точка применяется для передачи сигнализации в процессе хэндовера и при балансировке нагрузки между несколькими БС. Так же, как и для точек R4, между точками R8 могут создаваться защищенные соединения, в том числе, с использованием IP-Sec или SSL VPN.

### Поддержка мобильности в сетях WiMAX IEEE 802.16

Одно из главных направлений в области описания и стандартизации сетей WiMAX IEEE 802.16 — обеспечение глобальной мобильности абонентов WiMAX, их свободы перемещаться между различными сетями во всем мире, постоянно оставаясь — «на связи». Особенно актуально это замечание для мобильных сетей WiMAX, абоненты которых являются обладателями небольших пользовательских терминалов, которые чаще всего берутся с собой в командировки, деловые поездки и на отдых.

В связи с поддержкой мобильности в базовой модели сети WiMAX введены понятия домашних и гостевых сервис-провайдеров — соответственно, H-NSP и V-NSP (рисунок 2.9). Домашний NSP — это оператор, заключивший договор об обслуживании с абонентом WiMAX. Именно он реализует функции AAA. Для поддержки роуминга домашний сервис-провайдер WiMAX заключает роуминговые соглашения с другими NSP. Гостевой NSP — это оператор, который предоставляет WiMAX-абоненту услуги роуминга. Прежде всего, V-NSP обеспечивает для такого абонента функции AAA, а также полный или частичный доступ ко всем услугам WiMAX-сети.

WiMAX-сетях выделяют два вида мобильности: микро- и макромобильность. Метод обеспечения макромобильности применяется в случае движения АС в пределах одной ASN. Все функции выполняет сама ASN с помощью маршрутизатора FA. При этом методе CSN не участвует в процедуре передачи абонента от одной БС к другой (хэндовер) и не осведомлена

о ней. IP-адрес АС остается неизменным. Процедура хэндовера на радиointерфейсе определена стандартом IEEE 802.16.

### Мобильность с привязкой к ASN (микромобильность)

В процессе перемещения АС меняет точку привязки. В случае, когда точка привязки (БС) принадлежит той же ASN, АС не меняет свой IP-адрес. Такой тип мобильности называется мобильностью с привязкой к ASN. Существует также мобильность с привязкой к CSN, при которой АС перемещается между зонами действия БС, принадлежащих разным ASN.

Управление перемещением АС необходимо для снижения количества потерянных пакетов, уменьшения задержек при хэндовере и сохранения порядка следования пакетов при требуемом уровне безопасности.



Рис. 5.76. Модель взаимодействия операторов сервисных сетей WiMAX IEEE 802.16, сетей доступа и абонентов

### Мобильность с привязкой к CSN (макромобильность)

В случае, когда АС перемещается в зону покрытия БС, принадлежащей другой ASN, имеет место мобильность на основе IP, что приводит к перепривязке текущего внешнего агента к новому внешнему агенту.

Метод обеспечения макромобильности необходим при перемещении АС между различными ASN. Его суть состоит в переключении пути данных с одного интерфейса R3 на другой. Метод основан на использовании MIP (Mobile IP) — интернет-протокола, обеспечивающего передвижение абонента между сетями без изменения его IP-адреса. Таким образом, устройство, с которым абонент обменивается информацией, не замечает этого передвижения. MIP подразумевает, что у каждого устройства есть два IP-адреса — основной (HoA), присвоенный ему в домашней сети, и дополнительный (CoA).

MIP использует следующие функциональные компоненты: MIP-клиент (MIP client), FA, HA. MIP клиент может располагаться в AC (CMIP, Client MIP) — в этом случае AC должен быть присвоен глобальный CoA. Этот адрес AC сообщает HA домашней сети, который перехватывает все сообщения по HoA и направляет их по CoA (как правило, в режиме туннелирования и инкапсуляции). Также MIP клиент может быть расположен в FA ASN, который будет действовать от имени MS, от которой не требуется поддержки протокола MIP. При этом для всех сетей, лежащих за пределами ASN, в роли CoA мобильной станции выступает IP-адрес внешнего агента. Настоящий CoA AC присваивается FA и используется для маршрутизации пакетов внутри ASN. Основным IP-адресом мобильной станции для этих целей не используется из-за опасности его совпадения с IP-адресом другой AC, работающей внутри этой же ASN. В сетях WiMAX IEEE 802.16 обязательна поддержка обоих случаев расположения MIP-клиента.

HA содержит таблицу соответствия IP-адреса AC (основного или дополнительного, в зависимости от того, является ли данный NSP домашним по отношению к абоненту) IP-адресу FA, к которому относится эта AC. Таким образом, HA имеет информацию о том, к какому FA отправлять пакеты, предназначенные данной AC.

### **Физический уровень WiMAX**

На физическом уровне в стандарте IEEE 802.16-2004 определены три метода передачи данных: метод модуляции одной несущей (SC), метод ортогонального частотного мультиплексирования (OFDM) и метод множественного доступа на основе такого мультиплексирования (OFDMA) [2].

Спецификация физического уровня WirelessMAN-OFDM является наиболее интересной с точки зрения практической реализации. Она базируется на технологии OFDM, что значительно расширяет возможности оборудования, в частности, позволяет работать на относительно высоких частотах в условиях отсутствия прямой видимости. Кроме того, в нее включена поддержка топологии «каждый с каждым» (mesh) [3], при которой абонентские устройства могут одновременно функционировать и как базовые станции, что сильно упрощает развертывание сети и помогает преодолеть проблемы прямой видимости.

### **Модуляция OFDM**

При формировании OFDM-сигнала [4] цифровой поток данных делится на несколько подпотоков, и каждая поднесущая связывается со своим подпоток данных. Амплитуда и фаза поднесущей вычисляются на основе выбранной схемы модуляции. Согласно стандарту, отдельные поднесущие могут модулироваться с использованием бинарной фазовой манипуляции (BPSK), квадратурной фазовой манипуляции (QPSK) или квадратурной амплитудной манипуляции (QAM) порядка 16 или 64. В передатчике амплитуда как функция

фазы преобразуется в функцию от времени с помощью обратного быстрого преобразования Фурье (ОБПФ). В приемнике с помощью быстрого преобразования Фурье (БПФ) осуществляется преобразование амплитуды сигналов как функции от времени в функцию от частоты.

Применение преобразования Фурье позволяет разделить частотный диапазон на поднесущие, спектры которых перекрываются, но остаются ортогональными. Ортогональность поднесущих означает, что каждая из них содержит целое число колебаний на период передачи символа. Как видно на рисунке 5.77, спектральная кривая любой из поднесущих имеет нулевое значение для «центральной» частоты смежной кривой. Именно эта особенность спектра поднесущих и обеспечивает отсутствие между ними интерференции.

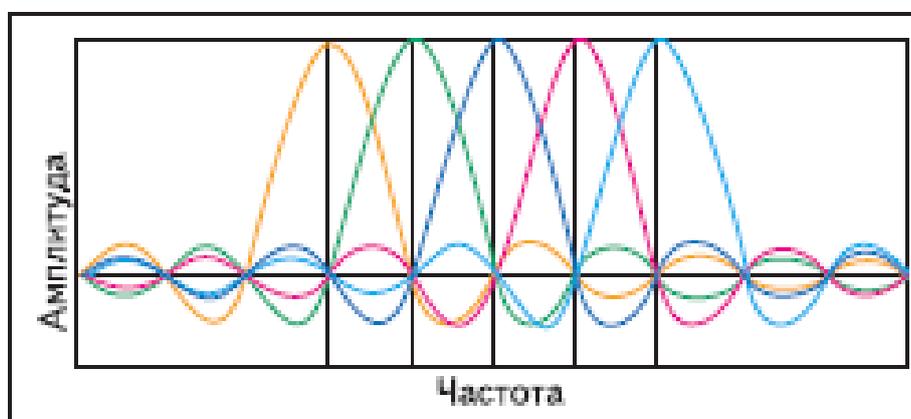


Рис. 5.77. Ортогональные поднесущие

Одним из главных преимуществ метода OFDM является его устойчивость к эффекту многолучевого распространения. Эффект вызывается тем, что излученный сигнал, отражаясь от препятствий, приходит к приемной антенне разными путями, вызывая межсимвольные искажения. Этот вид помех характерен для городов с разноэтажной застройкой из-за многократных отражений радиосигнала от зданий и других сооружений. Для того чтобы избежать межсимвольных искажений, перед каждым OFDM-символом вводится защитный интервал, называемый циклическим префиксом. Циклический префикс представляет собой фрагмент полезного сигнала, что гарантирует сохранение ортогональности поднесущих (но только в том случае, если отраженный сигнал при многолучевом распространении задержан не больше, чем на длительность циклического префикса). Кроме того, циклический префикс позволяет выбрать окно для преобразования Фурье в любом месте временного интервала символа (рисунок 5.78).

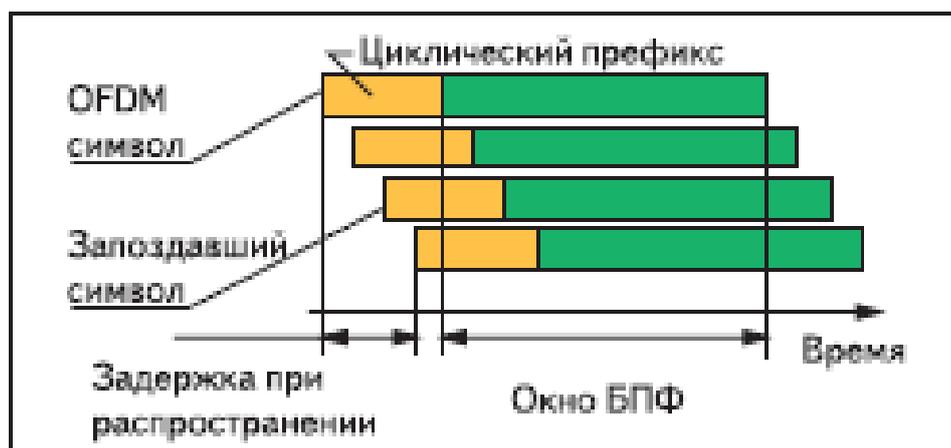


Рис. 5.79. Обработка OFDM-символа при многолучевом распространении

### Канальное кодирование данных

Многолучевое распространение радиосигнала может приводить к ослаблению и даже полному подавлению некоторых поднесущих вследствие интерференции прямого и задержанного сигналов. Для решения этой проблемы используется помехоустойчивое кодирование. В стандарте IEEE 802.16-2004 предусмотрены как традиционные технологии помехоустойчивого кодирования, так и относительно новые методы. К традиционным относится сверточное кодирование с декодированием по алгоритму Витерби и коды Рида-Соломона. К относительно новым — блочные и сверточные турбокоды. Для увеличения эффективности кодирования без снижения скорости кода применяется перемежение данных. Перемежение увеличивает эффективность кодирования, поскольку пакеты ошибок дробятся на мелкие фрагменты, с которыми справляется система кодирования.

### Уровень доступа к среде сетей WiMAX

#### Подуровни стандарта IEEE 802.16

Физический уровень стандарта IEEE802.16 организует непосредственную доставку потоков данных между БС и АС. Все задачи, относящиеся к формированию структур этих данных, а также контроль передачи потоков производятся на MAC-уровне (Medium Access Control). Оборудование стандарта IEEE 802.16 создаёт транспортную среду для трансляции потоков различных услуг (сервисов). Поток услуги (service flow) – центральная концепция MAC-протокола.

Обобщенная задача уровня MAC – это поддержка механизма различных услуг верхнего уровня. Разработчики стандарта стремились реализовать единый для всех приложений протокол MAC-уровня, независимо от особенностей физического канала, что позволяет сопрягать терминалы конечных пользователей с кабельными сетями передачи.

Физически среды передачи в разных соединениях сети могут быть различны, но структура данных одинакова. В одном канале могут работать (не одновременно) десятки

различных абонентских терминалов. Абоненты являются потребителями самых разных сервисов (приложений) в виде соединений по протоколу IP. Качество услуг (QoS) каждого отдельного соединения не должно изменяться при передаче информации через сети IEEE 802.16e. Алгоритмы и механизмы доступа MAC-уровня должны решать эти задачи.

MAC-уровень IEEE 802.16 включает три подуровня:

- подуровень преобразования потоков услуг (CS - Convergence Sublayer);
- основной подуровень (CPS - Common Part Sublayer);
- подуровень безопасности (PS - Privacy Sublayer).

В таблице 5.15 приведены операции, реализуемые на отдельных подуровнях уровня MAC.

Таблица 5.15. Основные операции на уровне MAC

Подуровень преобразования	Общая часть MAC	Подуровень безопасности
<ul style="list-style-type: none"> <li>• Упаковка PDU для нижестоящего уровня</li> <li>• Распаковка PDU для вышестоящего уровня</li> </ul>	<ul style="list-style-type: none"> <li>• Ввод и подавление заголовков</li> <li>• Режим запроса повторной передачи</li> <li>• Фрагментация</li> <li>• Установление соединения/разъединения</li> <li>• Управление качеством (QoS)</li> <li>• Многопользовательские услуги</li> <li>• Соединение/разъединение с сетью</li> <li>• Управление предоставляемой полосой частот</li> </ul>	<ul style="list-style-type: none"> <li>• Поддержка режима шифрации</li> <li>• Обмен данными при переходе к шифрации</li> <li>• Обмен ключом авторизации</li> <li>• Взаимная аутентификация</li> </ul>

На подуровне преобразования потоков услуг осуществляется преобразование потоков данных протоколов верхних уровней для передачи по сети WiMAX. Стандарт предусматривает свой механизм трансформации для каждого типа приложений верхних уровней. При этом можно реализовать различные протоколы пакетной передачи данных: ATM, PPP, IEEE 802.3 (Ethernet).

### Общий подуровень доступа к среде

В общей части подуровня MAC реализованы методы множественного доступа, управления ресурсами, установки соединений и функции поддержки работоспособности системы. В ядре MAC - подуровня также обеспечивается работа системы безопасности WiMAX и предоставляется точка входа на физический уровень сети для передачи данных по радиointерфейсу.

### Подуровень безопасности

Подуровень безопасности осуществляет криптозащиту данных и механизмы аутентификации/предотвращения несанкционированного доступа. С этой целью реализованы наборы алгоритмов криптозащиты и протокол управления ключами шифрования.

### Описание экспериментальной установки и методики измерений

В ходе данной работы были изучены существующие программа для симуляции физического уровня системы связи на основе технологии WiMAX. Для запуска в командную строку MATLAB надо ввести "commwman80216dstbc" и нажать Enter. Произойдет симуляция физического уровня системы связи на основе технологии WiMAX (см. рис.

Построенная схема исследуемой системы приведена на рисунке 5.80 [21].

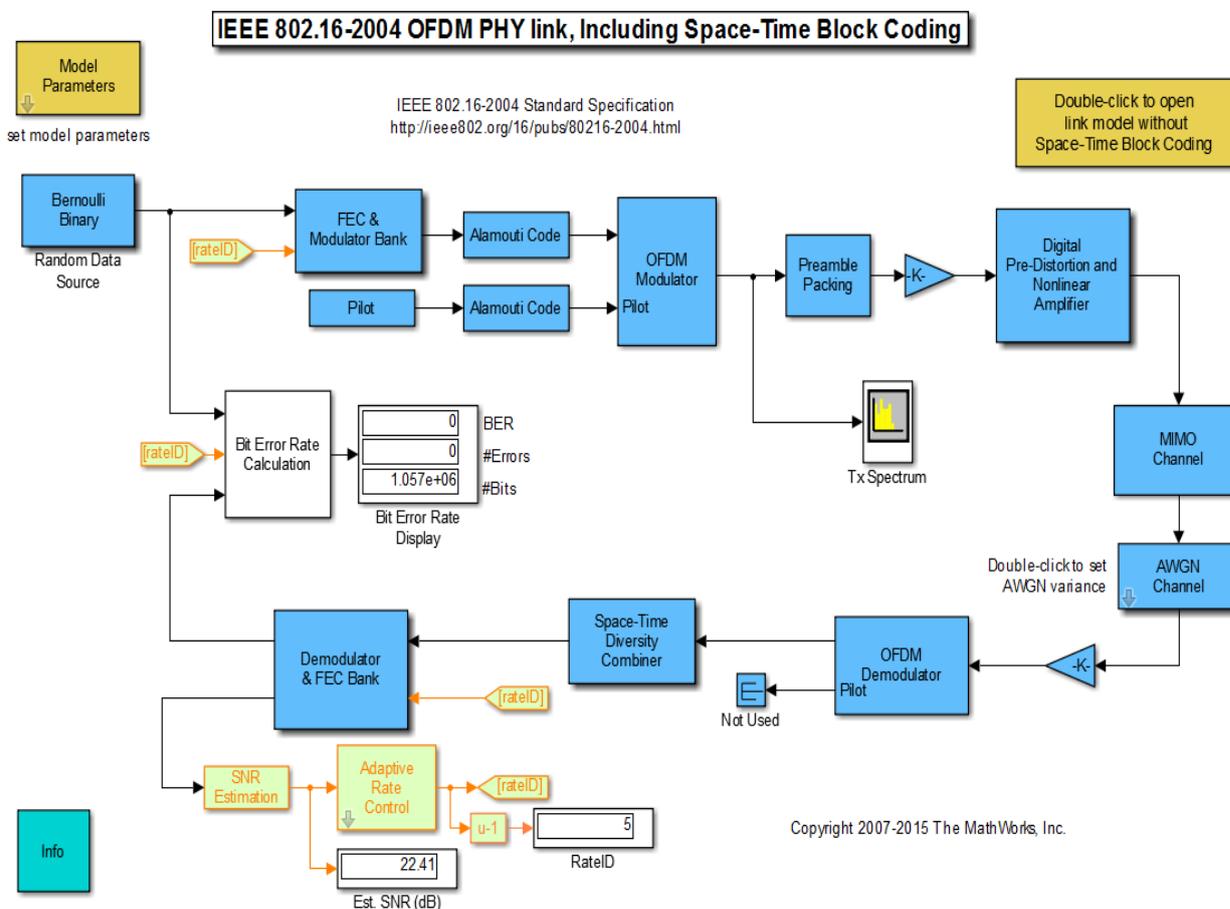


Рис. 5.80. Модель IEEE 802.16-2004 OFDM в MATLAB 2015b

При проведении симуляции существует возможность изменения ряда параметров системы в следующих блоках:

Общие параметры модели (блок «Model Parameters», рисунок 5.81).

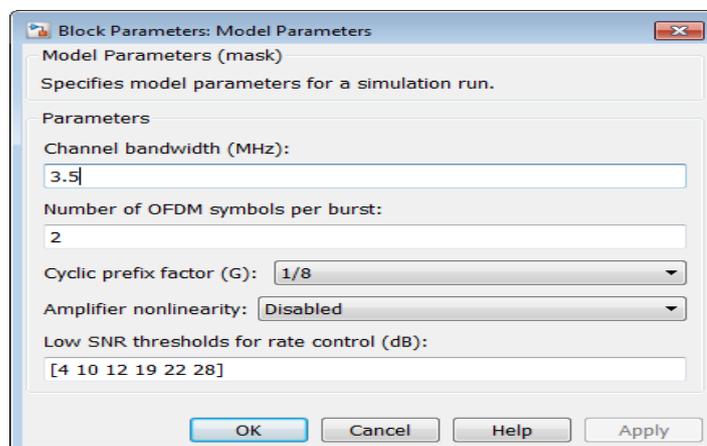


Рис. 5.81. Параметры системы, изменяемые в блоке «Model Parameters»

Параметры OFDM-модулятора (блок «OFDM Modulator», рисунок 5.82).

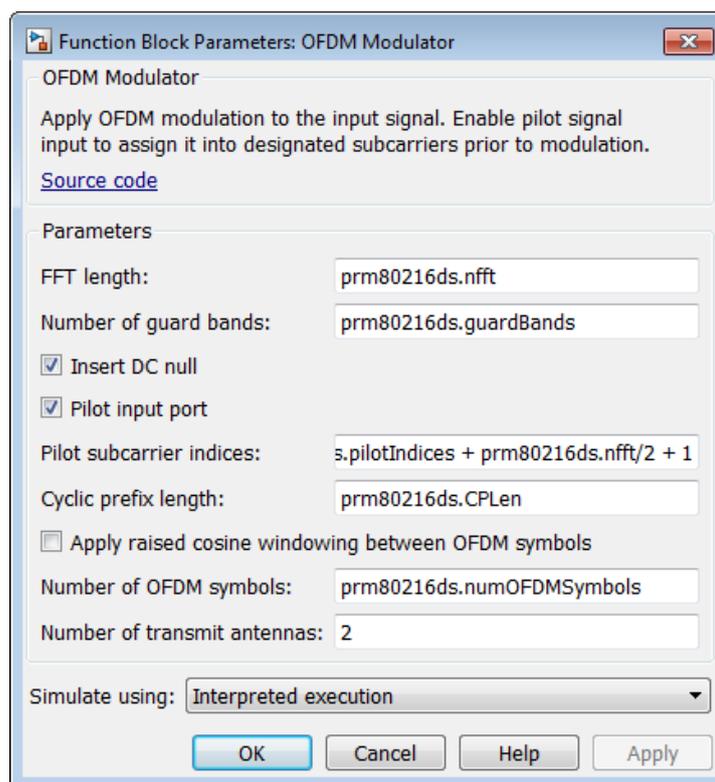


Рис. 5.82. Параметры системы, изменяемые в блоке «OFDM Modulator»

Параметры OFDM-демодулятора (блок «OFDM Demodulator», рисунок 5.83).

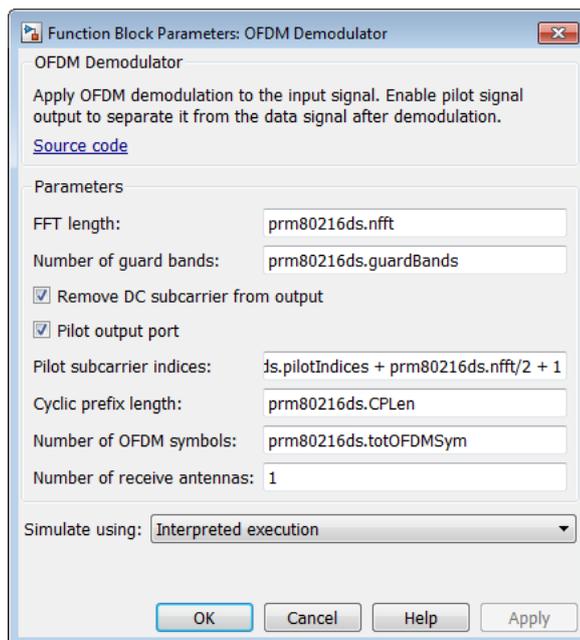


Рис. 5.83. Параметры системы, изменяемые в блоке «OFDM Demodulator»  
Параметры канала MIMO (блок «MIMO Channel», рисунок 5.84).

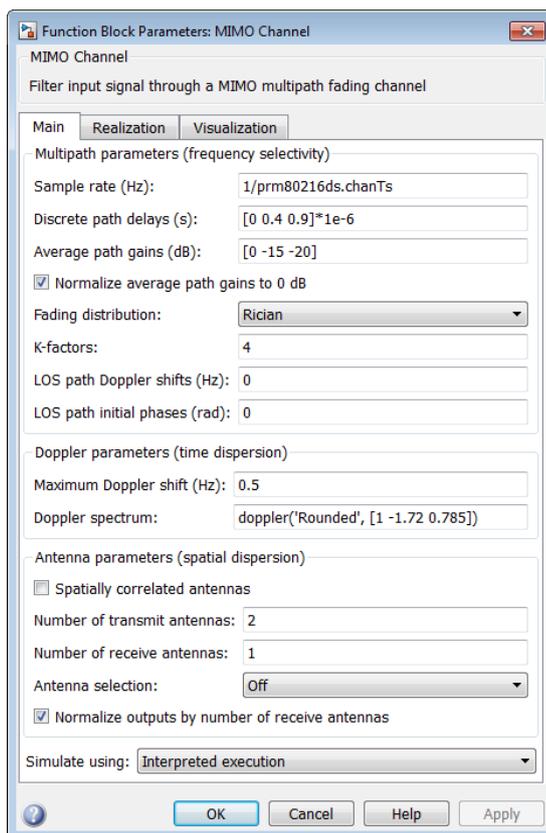


Рис. 5.84. Параметры системы, изменяемые в блоке «MIMO Channel»

Параметры канала AWGN (блок «AWGN Channel», рисунок 5.85).

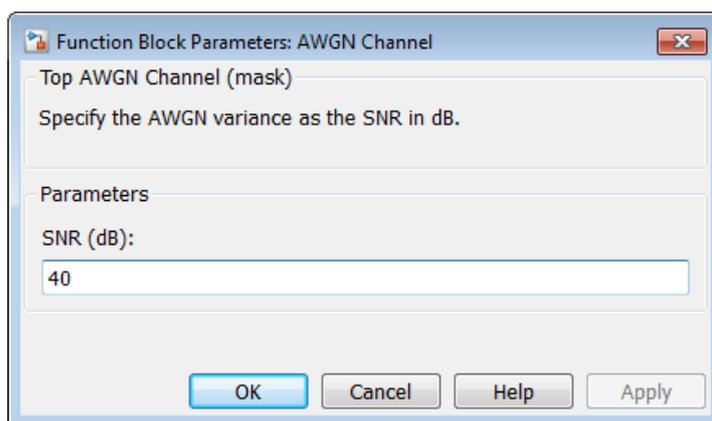


Рис. 5.85. Параметры системы, изменяемые в блоке «AWGN Channel»

### Результаты работы и их анализ

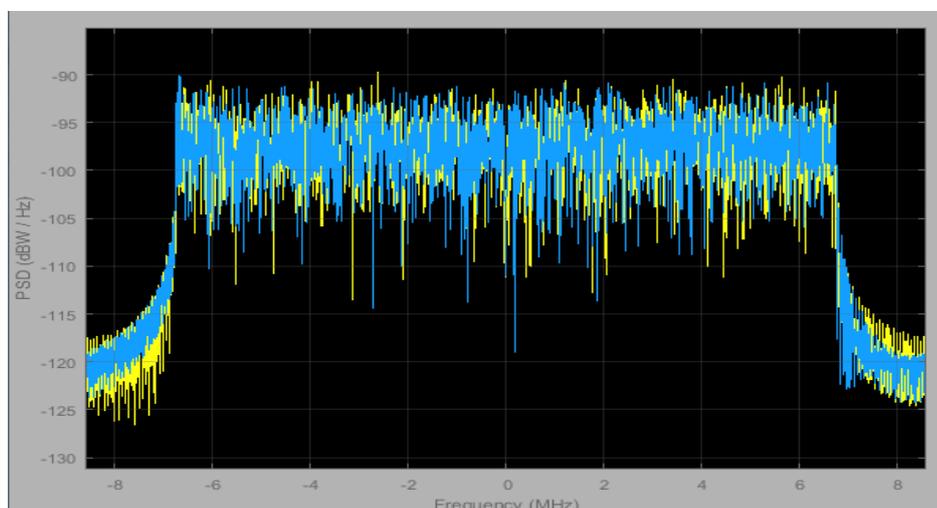


Рис. 5.86. Спектр передаваемого сигнала

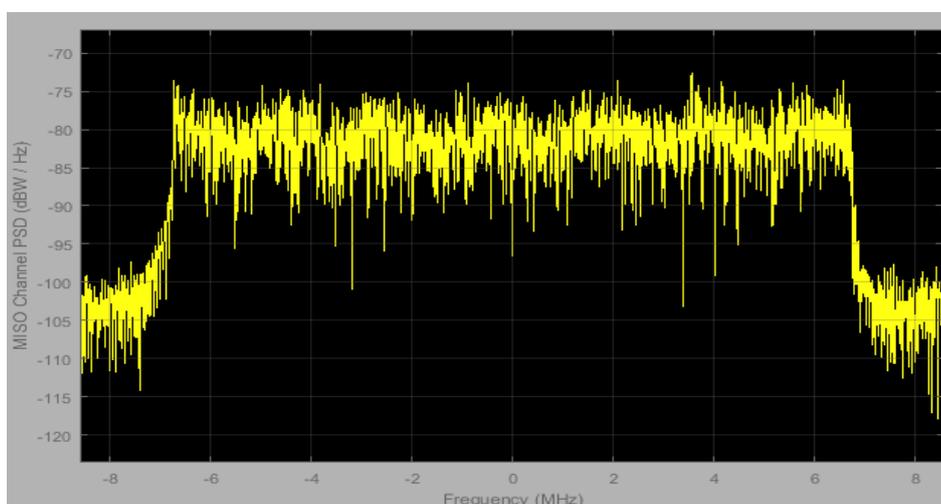


Рис. 5.87. Спектр принимаемого сигнала

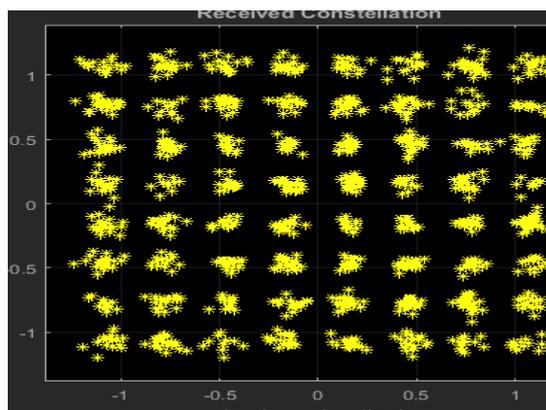


Рис. 5.88. Созвездие принимаемого сигнала

5. По данным блока «Bit Error Rate Display» построить график зависимости  $BER = f(SNR)$ .

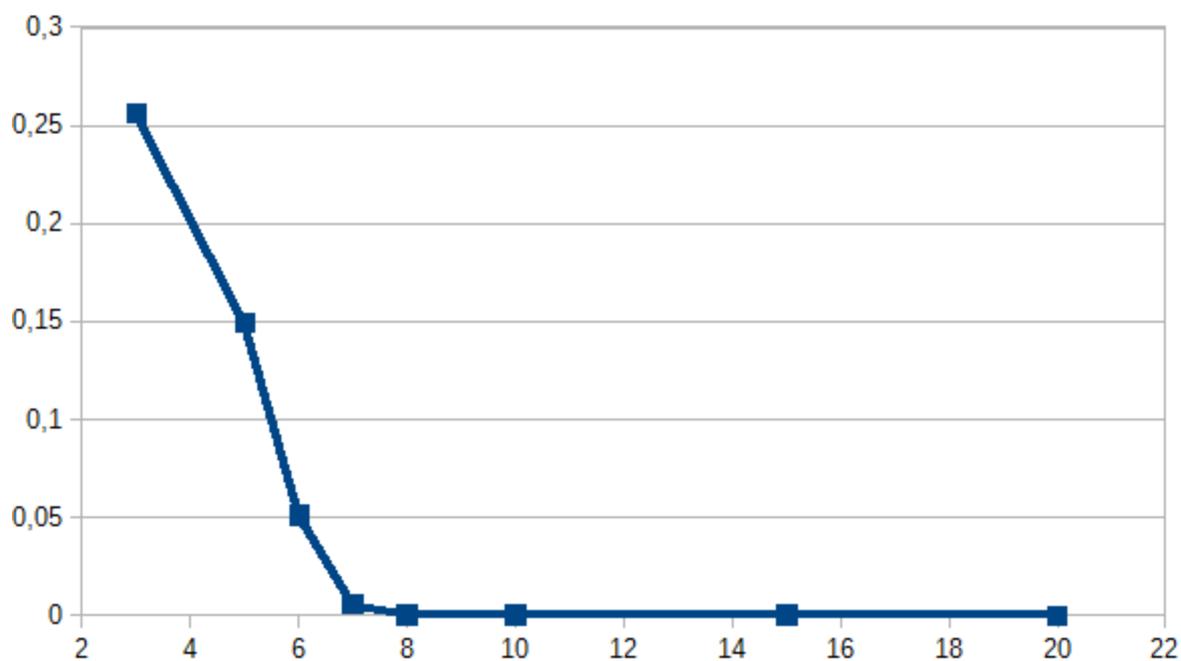


Рис. 5.89. График зависимости  $BER = f(SNR)$

График зависимости количества ошибок от отношения сигнал/шум приведён на рисунке 5.90.

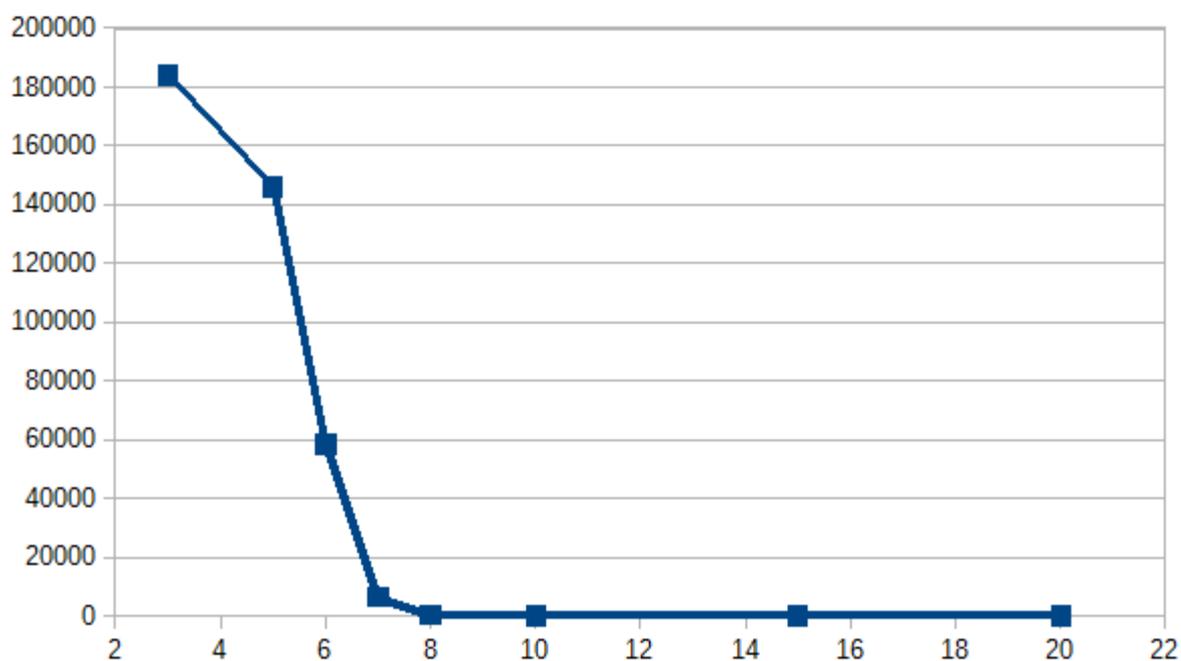


Рис. 5.90. График зависимости  $ERRORS = f(SNR)$

6. Как было выяснено ранее, система адаптируется к каналу связи, изменяя вид модуляции сигнала. Необходимо посмотреть при каком уровне SNR, какой вид модуляции. Записать значения в виде таблицы и представить на графиках.

В результате симуляции были получены следующие данные:

Созвездие принимаемого сигнала (BPSK):

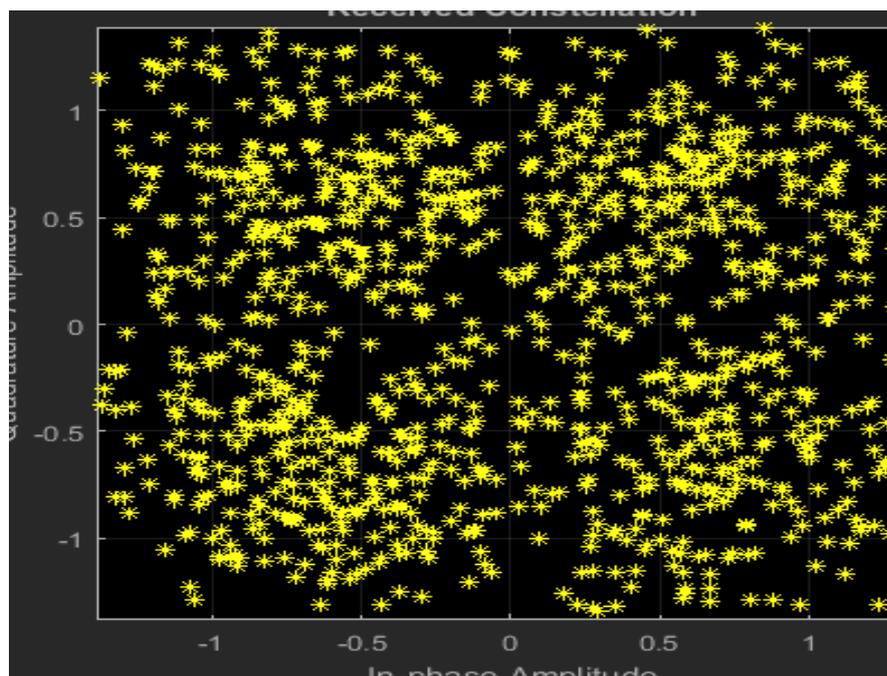


Рис. 5.91. Созвездие принимаемого сигнала (SNR = 6)

Созвездие принимаемого сигнала (QAM-4):

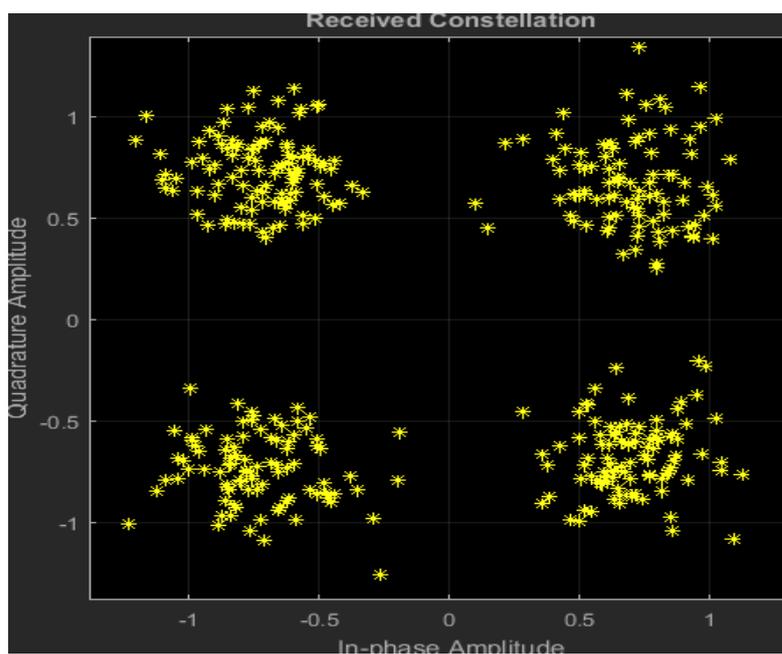


Рис. 5.92. Созвездие принимаемого сигнала (SNR = 15)

Созвездие принимаемого сигнала (QAM-16):

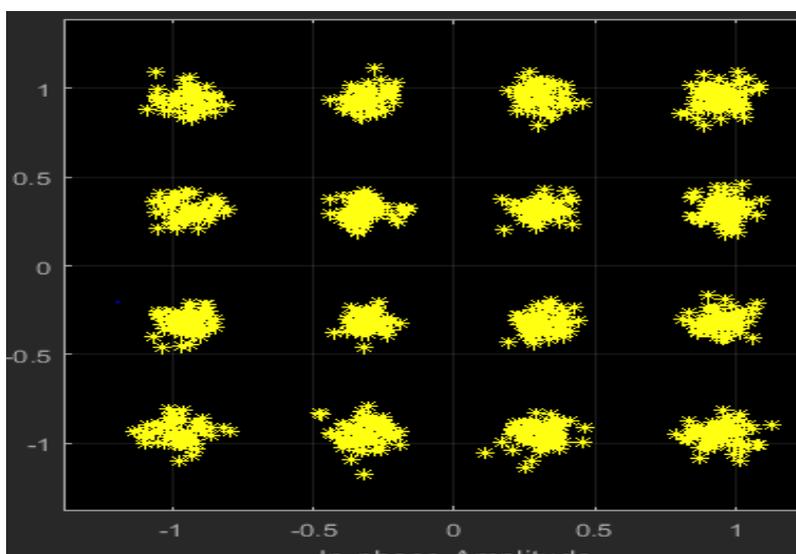


Рис. 5.93. Созвездие принимаемого сигнала (SNR = 20)

Созвездие принимаемого сигнала (QAM-64):

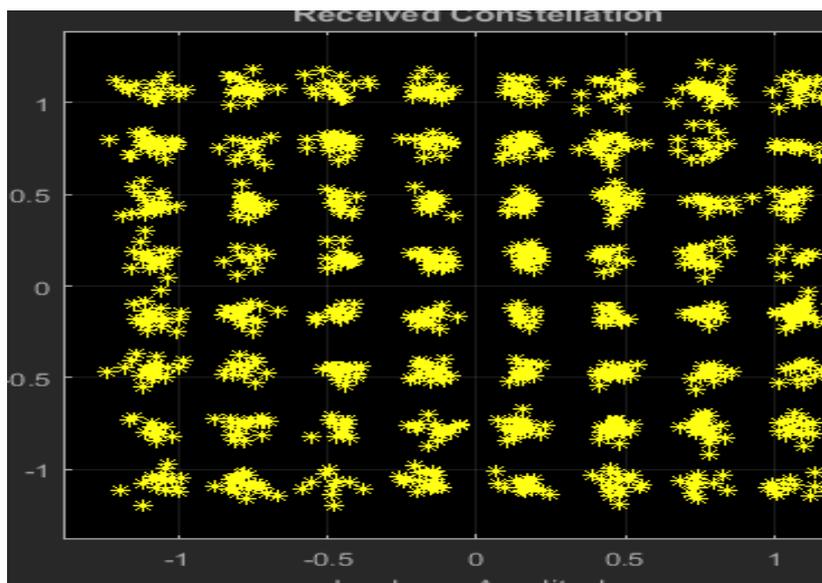


Рис. 5.94. Созвездие принимаемого сигнала (SNR = 30)

В результате работы была изучена программа для моделирования системы связи в соответствии со стандартом WiMAX (802.16).

1. Можно сделать вывод, что система адаптируется к каналу связи, изменяя вид модуляции сигнала: при SNR = 5 дБ – BPSK, SNR = 14 дБ – QAM-4, SNR = 25 дБ – QAM-16, SNR = 40 дБ – QAM-64.

2. Принимаемый сигнал является суммой передаваемых сигналов (подвергнутых зашумлению).

3. При увеличении SNR значение BER уменьшается. Выбросы при SNR = 5 дБ и 14 дБ обусловлены переходами системы на другой вид модуляции при низком SNR, при SNR = 4. производится модуляция BPSK, при SNR = 8 – QAM-4, при SNR = 14 – QAM-16. Смена вида модуляции осуществляется путём оценки SNR в канале связи (блоки «SNR Estimation» и «Adaptive Rate Control»).

5. График зависимости оценки SNR от реального SNR, можно сделать вывод, что система работает наиболее стабильно (зависимость линейна) на участке 6...24 дБ. При SNR > 24 дБ более точная оценка канала не требуется (выбирается наименее помехоустойчивый метод модуляции – QAM-64 (в рамках стандарта)). При SNR < 6 дБ выбирается наиболее помехоустойчивый метод модуляции – BPSK.

## **5.7. Имитационное моделирование системы мобильной связи стандарта IEEE 802. 20 (LTE) на базе ПО MATLAB**

Целью раздела является приобретение и закрепление навыков организации и реализации в программной среде системы мобильной связи стандарта LTE, подробное изучение схем входящих в состав стандарта и программного обеспечения с которыми предстоит работать при выполнении курсового проекта, умения выбрать необходимые решения на основе требований технического задания.

Помимо теоритической части, задачей курсового проектирования является построение в программной среде схемы передачи информации от базовой станции (БС) к мобильной станции (МС) и ее анализ. Схема будет включать в свой состав: генератор бинарной последовательности, кодек, модулятор/демодулятор, канал связи, анализатор ошибок и т.д.

Основным отличием стандарта LTE от предыдущих стандартов сетей связи является применение «плоской» более упрощённой IP-архитектуры, которая способствует уменьшению задержек при установленной Интернет-сессии. В стандарте LTE использовано два принципиально новых метода увеличения пропускной способности. Первый заключается в применении технологии MIMO (Multiple Input Multiple Output), где передача и приём сигнала осуществляется одновременно через несколько передающих и приёмных антенн. Таким образом, повышается скорость передачи данных в беспроводных сетях. Второй метод заключается в применении OFDM (Orthogonal frequency division multiplexing) модуляции, использующей несколько поднесущих. Преимущество данного метода заключается также в том, что системы связи с LTE могут работать в отсутствии прямой видимости.

### **Стандарты 2G и 3G**

#### **Стандарт 2G (GSM)**

Разработка стандарта GSM началась еще в 1982 году организацией по стандартизации CEPT (European Conference of Postal and Telecommunications Administrations) . В 1991 году в Финляндии была введена в эксплуатацию первая в мире сеть GSM. Уже к концу 1993 года число абонентов, использующих этот стандарт, перевалило за миллион. К этому времени сети GSM были развернуты в 73 странах мира.

Сети стандарта GSM позволяют предоставлять широкий перечень услуг:

- Голосовые соединения
- Услуги передачи данных (до 384 кбит/сек благодаря технологии EDGE (дополнение технологии GPRS, в результате появилась передача данных с пакетной коммутацией, т.е. пакетный трафик отделяется от голосового))
- Передача коротких текстовых сообщений (SMS)

- Передача факсов
- Голосовая почта
- Конференцсвязь и мн. др.

Итак, рассмотрим основные элементы, входящие в состав системы GSM:

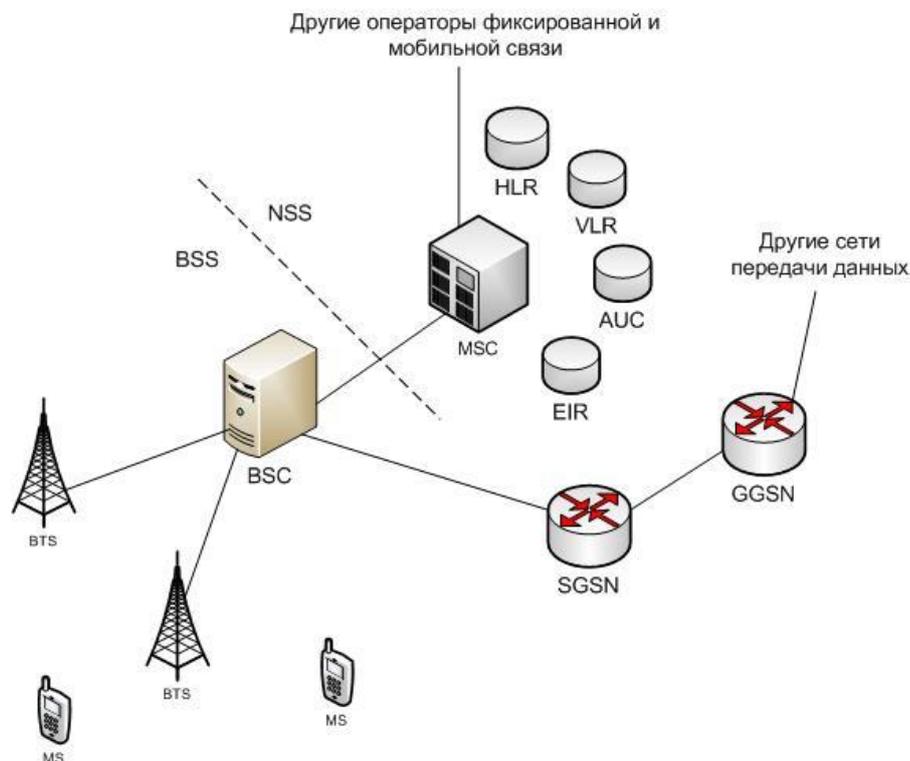


Рис. 5.95. Структура сети стандарта 2G (GSM)

Сеть GSM делится на 2 системы. Каждая из этих систем включает в себя ряд функциональных устройств, которые, в свою очередь являются компонентами сети мобильной радиосвязи.

Данными системами являются:

- Система коммутации – Network Switching System (NSS).
- Система базовых станций - Base Station System (BSS).

Система NSS выполняет функции обслуживания вызовов и установления соединений, а также отвечает за реализацию всех назначенных абоненту услуг. NSS включает в себя следующие функциональные устройства:

- Центр коммутации мобильной связи (MSC).
- Домашний регистр местоположения (HLR).
- Визитный регистр местоположения (VLR).
- Центр аутентификации (AUC).

- Регистр идентификация абонентского оборудования (EIR).

Система BSS отвечает за все функции, относящиеся к радиointерфейсу. Эта система включает в себя следующие функциональные блоки:

- Контроллер базовых станций (BSC).
- Базовую станцию (BTS).

MS (т.е. телефон абонента (мобильная станция)) не принадлежит ни к одной из этих систем, но рассматривается как элемент сети.

Элементы сети, относящиеся к пакетной передаче данных:

- SGSN – узел обслуживания абонентов.
- GGSN – шлюзовой узел.

### Стандарт 3G (UMTS)[2]

Разработка стандарта UMTS началась в 1992 году организацией по стандартизации ИМТ-2000. Впоследствии разработка этого стандарта была поручена [3GPP](#). Первая сеть UMTS была запущена в коммерческую эксплуатацию 1 декабря 2001 года в Норвегии. К маю 2010 года число абонентов переваливает за 540 миллионов по всему миру.

Скорость передачи данных для сетей UMTS может достигать 2Мбит/сек. Благодаря технологии [HSDPA](#)-High Speed Downlink Packet Access (3.5G), которая была внедрена в 2006 году максимальная скорость возросла до 14 Мбит/сек. Эти и другие преимущества UMTS позволяют предоставлять абонентам широкий перечень услуг: [ВИДЕОЗВОНКИ](#), [ВИДЕОКОНФЕРЕНЦИИ](#), высококачественные голосовые звонки, загрузка файлов с высокой скоростью, сетевые игры, мобильная коммерция и мн. др.

Рассмотрим структуру системы UMTS и ее основные отличия от стандарта второго поколения [GSM](#).

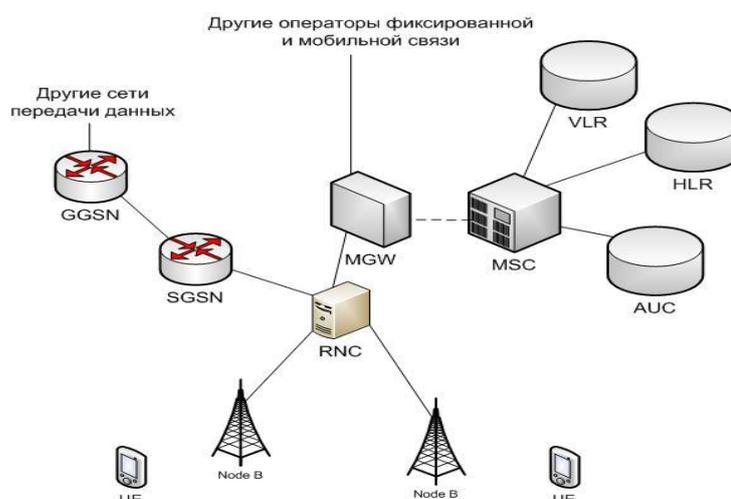


Рис. 5.96. Структура сети стандарта 3G (UMTS)

**Подсистема коммутации:**

В первых релизах стандарта UMTS (R99, R4) подсистема коммутации не отличалась по своей структуре от той же подсистемы сетей второго поколения. В нее входили MSC – Mobile Switching Centre, который выполнял функции коммутации, установления соединения, тарификации и др., а также ряд регистров HLR, VLR, AUC, которые предназначены для хранения абонентских данных. В более поздних релизах (R5, R6, R7,R8) функции MSC были разделены между двумя устройствами: MSC-Server и MGW (Media gateway). MSC-Server отвечает за установление соединений, тарификацию, выполняет некоторые функции аутентификации. MGW представляет собой коммутационное поле, подчиненное MSC-Server.

**Подсистема базовых станций:**

В сети UMTS по сравнению с сетью GSM наибольшие изменения претерпела подсистема базовых станций. Отмеченные выше преимущества достигаются в первую очередь за счет новой технологии передачи информации между базовой станцией и телефоном абонента.

Итак, рассмотрим основные элементы, входящие в подсистему базовых станций:

RNC (Radio Network Controller) – контроллер сети радиодоступа системы UMTS. Он является центральным элементом подсистемы базовых станций и выполняет большую часть функций: контроль радиоресурсов, шифрование, установление соединений через подсистему базовых станций, распределение ресурсов между абонентами и др. В сети UMTS контроллер выполняет гораздо больше функций, нежели в системах сотовой связи второго поколения.

NodeB – базовая станция системы сотовой связи стандарта UMTS. Основной функцией NodeB является преобразование сигнала, полученного от RNC в широкополосный радиосигнал, передаваемый к телефону. Базовая станция не принимает решений о выделении ресурсов, об изменении скорости к абоненту, а лишь служит мостом между контроллером и оборудованием абонента, и она полностью подчинена RNC.

Оборудование абонента получило название UE (User Equipment (мобильная станция)). Тем самым подчеркивается, что в отличие от предшествующих стандартов в UMTS может быть не только обычный телефон, но и смартфон, ноутбук, стационарный компьютер и т.п.

Пакетные данные в сети UMTS передаются от MGW к известному нам по системе GSM элементу SGSN (узел обслуживания абонентов), после чего через GGSN (шлюзовой узел) поступают к другим внешним сетям передачи данных, например Internet. Как правило, SGSN и GGSN сети GSM применяются для тех же целей и в сети UMTS. Производится только коррекция программного обеспечения данных элементов.

**Стандарт LTE и его отличие от предыдущих стандартов**

Стандарты третьего поколения позволяют предоставить широкий перечень мультимедийных услуг и поддерживают скорости передачи данных до 14Мбит/сек. Это

вполне соответствует запросам абонентов в настоящее время. Однако, объемы передаваемой информации в телекоммуникационных сетях растут с каждым днем. Чтобы удовлетворить потребности пользователей по скорости передачи данных и набору услуг, хотя бы на 20 лет вперед необходим новый стандарт, уже четвертого поколения.

Работа над первым стандартом четвертого поколения - LTE (Long Term Evolution) началась в 2004 году организацией 3GPP. Главными требованиями, которые предъявлялись в процессе работы над стандартом были следующие:

- Скорость передачи данных выше 100 Мбит/сек.
- Высокий уровень безопасности системы.
- Высокая энергоэффективность.
- Низкие задержки в работе системы.
- Совместимость со стандартами второго и третьего поколений.

В конце 2009 года в Швеции была запущена в коммерческую эксплуатацию первая сеть стандарта LTE.

Сети LTE поддерживают скорости передачи данных до 326,4 Мбит/сек. К примеру, загрузка фильма в хорошем качестве займет менее одной минуты. Таким образом, верхняя планка по скорости передачи данных практически снимается.

Рассмотрим структуру сети LTE:

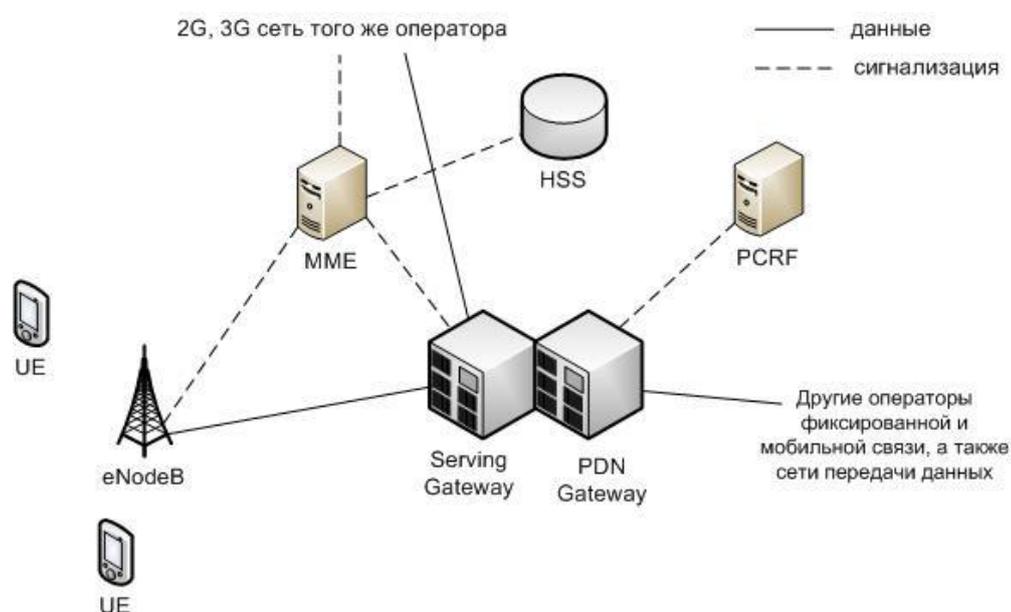


Рис. 5.97. Структура сети стандарта LTE

Из схемы сети LTE, представленной выше, уже видно, что структура сети сильно отличается от сетей стандартов 2G и 3G. Существенные изменения претерпела и подсистема базовых станций, и подсистема коммутации. Была изменена технология передачи данных между оборудованием пользователя и базовой станцией. Также подверглись изменению и

протоколы передачи данных между сетевыми элементами. Вся информация (голос, данные) передается в виде пакетов. Таким образом, уже нет разделения на части обрабатывающие либо только голосовую информацию, либо только пакетные данные.

Можно выделить следующие основные элементы сети стандарта LTE:

**Serving SAE Gateway** или просто **Serving Gateway (SGW)** – обслуживающий шлюз сети LTE. Предназначен для обработки и маршрутизации пакетных данных поступающих из/в подсистему базовых станций. По сути, заменяет MSC (выполняет функции коммутации, установления соединения, тарификации), MGW (представляет собой коммутационное поле) и SGSN (узел обслуживания абонентов пакетной сети передачи данных) сети UMTS (3G). SGW имеет прямое соединение с сетями второго и третьего поколений того же оператора, что упрощает передачу соединения в/из них по причинам ухудшения зоны покрытия, перегрузок и т.п.

**Public Data Network (PDN) SAE Gateway** или просто **PDN Gateway (PGW)** – шлюз к/от сетей других операторов. Если информация (голос, данные) передаются из/в сети данного оператора, то они маршрутизируются именно через PGW.

**Mobility Management Entity (MME)** – узел управления мобильностью. Предназначен для управления мобильностью абонентов сети LTE.

**Home Subscriber Server (HSS)** – сервер абонентских данных. HSS представляет собой объединение VLR (гостевой регистр местоположения), HLR (домашний регистр местоположения), AUC (центр аутентификации абонентов) выполненных в одном устройстве.

**Policy and Charging Rules Function (PCRF)** – узел выставления счетов абонентам за оказанные услуги связи.

Все перечисленные выше элементы относятся к системе коммутации сети LTE. В системе базовых станций остался лишь один знакомый нам элемент – базовая станция, которая получила название **eNodeB**. Этот элемент выполняет функции и базовой станции, и контроллера базовых станций сети LTE. За счет этого упрощается расширение сети, т.к. не требуется расширение емкости контроллеров или добавления новых. Мобильная станция представлена – UE.

### **Интерфейсы между узловыми элементами в сетях стандарта LTE**

Структура сети стандарта LTE претерпела значительные изменения по сравнению с сетями предыдущих поколений. Это повлияло также и на изменение интерфейсов между узлами сети. На рисунке ниже представлена общая модель сети стандарта LTE и ее основные интерфейсы.

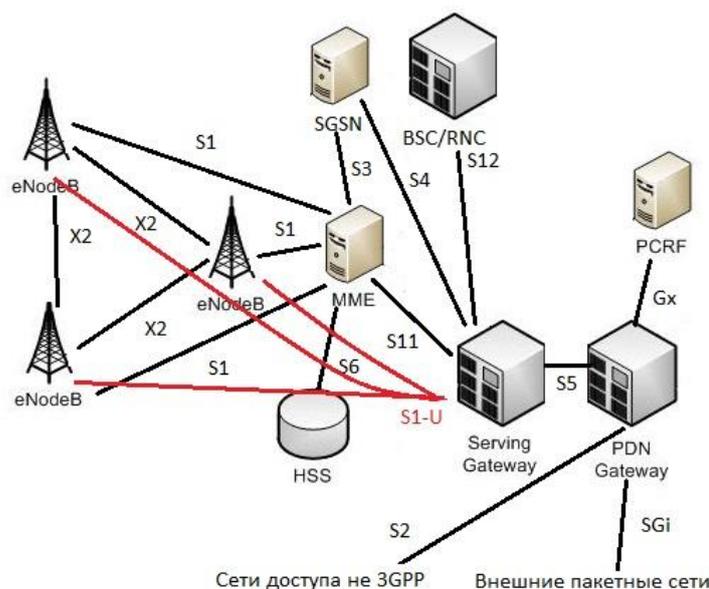


Рис. 5.98. Интерфейсы сети стандарта LTE

Рассмотрим основные интерфейсы сети LTE:

- X2 - интерфейс между eNodeB. Базовые станции в сети LTE соединены по принципу «каждый с каждым».
- S1 – интерфейс связывающий подсистему базовых станций E-UTRAN и MME. По данному интерфейсу передаются данные управления.
- S1-U – интерфейс между E-UTRAN и SAE, по которому передаются пользовательские данные.
- S2 – интерфейс для организации соединения между PDN-Gateway и сетями доступа, которые не разрабатывались 3GPP.
- S3 – интерфейс, предоставляющий прямое соединение SGSN и MME. Он служит для передачи данных управления для обеспечения мобильности между LTE и 2G/3G сетями.
- S4 – интерфейс, связывающий SAE и SGSN. Он служит для передачи пользовательских данных для обеспечения мобильности между LTE и 2G/3G сетями.
- S5 – интерфейс между SAE и PDN-Gateway. S5 предназначен для передачи пользовательских данных между SAE и PDN-Gateway.
- S6 – интерфейс между MME и HSS. Он используется для передачи данных абонентского профиля, а также осуществления процедур аутентификации в сети LTE.
- Gx – интерфейс между PDN-Gateway и PCRF. Gx предназначен для передачи правил тарификации от PCRF к PDN-Gateway.
- SGi - интерфейс между PDN-Gateway и внешними IP-сетями.

### Принципы построения радиointерфейса LTE в Downlink (от БС к МС)

Одной из главных отличительных особенностей стандарта LTE, которая позволяет достигать высоких скоростей передачи данных является изменение принципов построения интерфейса от eNodeB (БС) до UE (МС) на линии «вниз». Рассмотрим главные особенности этого интерфейса и постараемся выделить основные качественные отличия, которые отличают этот стандарт от других.

В сетях связи стандарта LTE в Downlink (DL) используется модуляция OFDM – Orthogonal Frequency Devision Multiplexing– ортогональная частотная модуляция. Этот тип модуляции определяет и принцип доступа OFDMA - Orthogonal Frequency Devision Multiple Access – множественный доступ с ортогональным частотным разделением каналов. Суть его заключается в том, что все частотно-временное поле, выделенное для работы оператора, разделяется на небольшие блоки. Причем они небольшие как по частоте (15 кГц), так и по времени (0,5 мс). Сеть распределяет эти блоки между абонентами в зависимости от их потребностей и возможностей сети. Таким образом, обеспечивается максимально эффективное использование ресурсов.

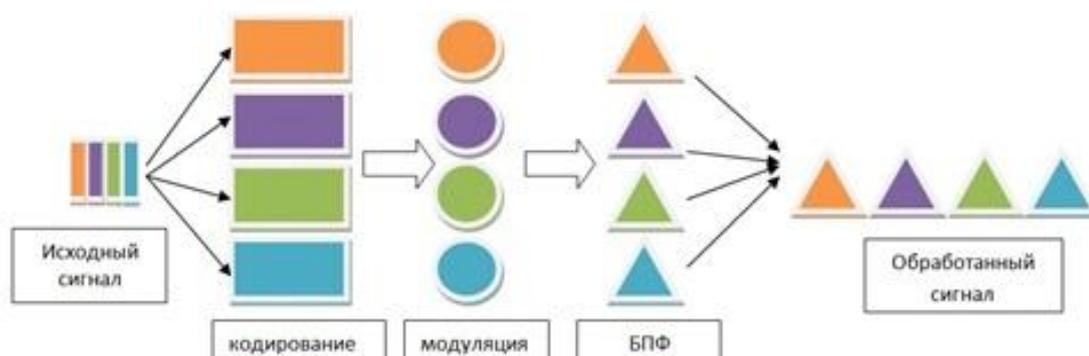


Рис. 5.99. OFDM – модулятор

Ниже перечислены главные шаги преобразования сигнала в OFDM модуляторе.

- 1) Разделение исходного потока бит на параллельные потоки.
- 2) Кодирование помехоустойчивым кодом, в процессе которого значительно увеличивается число символов в отдельных потоках.
- 3) Манипуляция выбранным в данный конкретный момент способом модуляции: QPSK, 16QAM, 64QAM.
- 4) Перемножение полученной последовательности каждого потока на свою поднесущую. Эта операция является ключевой и будет рассмотрена ниже.
- 5) Объединение сигналов и передача в эфир.

Умножение сигнала на свою поднесущую перемещает сигнал в нужное частотное пространство. Также на этом этапе происходит преобразование сигнала из временной области в частотную. Это выполняется благодаря БПФ – быстрому преобразованию Фурье. Эти две процедуры позволяют добиться максимально близкого размещения сигналов в частотной области и сократить до минимума защитные интервалы. Это достигается благодаря тому, что поднесущие выбираются ортогональными (на практике квазиортогональными), и отдельные потоки относительно легко выделить на приемной стороне.

Кроме использования OFDMA в LTE – есть еще одно важное новшество: обязательное (в отличие от UMTS) использование MIMO - Multiple Input Multiple Output – множественный вход множественный выход. При этом информационный поток направляется между сторонами обмена информации несколькими «путями», что обеспечивает более эффективное использование частотно-временного ресурса.

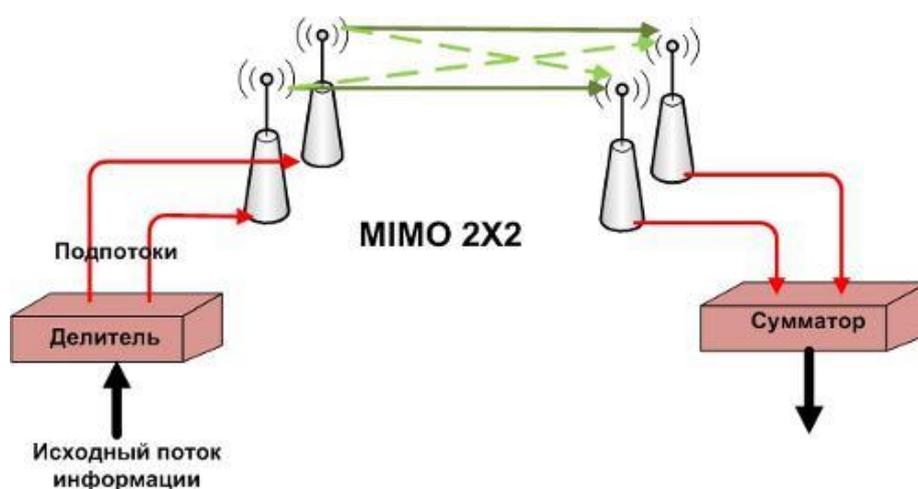


Рис. 5.100. MIMO 2x2

Эти два важных изменения позволяют добиться скорости передачи данных в Downlink свыше 100 Мбит/сек. Задержки передачи данных не превышают 20 мс. Для сравнения в UMTS скорости передачи данных редко поднимаются свыше 20 Мбит/сек, а задержки могут колебаться от 40 до 100 мс.

### **Принципы построения радиointерфейса LTE в Uplink (от МС к БС)**

В сетях связи стандарта LTE скорость передачи данных в направлении от UE (МС) к eNodeB (БС) может достигать 50 Мбит/сек, а задержки не превышают 10мс. Эти показатели на много превышают значения в сетях третьего поколения и практически сравнялись с проводными выделенными каналами связи. Рассмотрим главные особенности построения радиointерфейса Uplink в стандарте LTE.

В отличие от радиointерфейса Downlink, где информация одного пользователя может передаваться на разных поднесущих, в Uplink данные каждого пользователя передаются в одной полосе частот, причем в одно и то же время. Однако это не означает, что информационные потоки накладываются друг на друга и необратимо искажаются. Это обеспечивается благодаря использованию множественного доступа с частотным разделением с единственной несущей частотой SC-FDMA (Single Carrier Frequency Devision Multiple Access). Рассмотрим основные принципы работы SC-FDMA – модулятора.

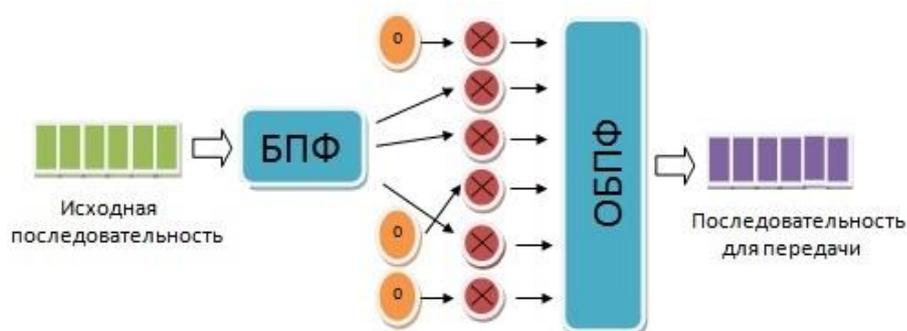


Рис. 5.101. SC-FDMA – модулятор

Первым этапом исходная информационная последовательность, предназначенная для передачи от абонента, преобразуется в частное представление с помощью быстрого преобразования Фурье (БПФ). Далее, в зависимости от скорости потока от данного абонента, сеть выделяет UE (MC) несколько поднесущих, среди которых распределяются преобразованный поток. Те поднесущие, которые используют другие пользователи не занимают в данном абонентском терминале, а соответствующие поднесущие перемножаются с «0». После обратного быстрого преобразования Фурье (ОБПФ) модулированные потоки объединяются и переводятся обратно во временную область. Несмотря на то, что данные передаются от разных устройств в сети в одно и то же время в одной и той же полосе частот, на приемной стороне после обратных сказанным выше процедур, можно выделить информационные потоки от отдельных UE (MC).

Благодаря использованию SC-FDMA в системе LTE удалось достигнуть трехкратного увеличения спектральной эффективности на линии «вверх», по сравнению с сетями 3G.

### Логические каналы на радиointерфейсе в LTE

Одной из важнейших составляющих радиointерфейса любой подвижной системы связи, которая обеспечивает заданные характеристики ее работы, является структура логических, транспортных и физических каналов. Рассмотрим логические параметры сети связи LTE.

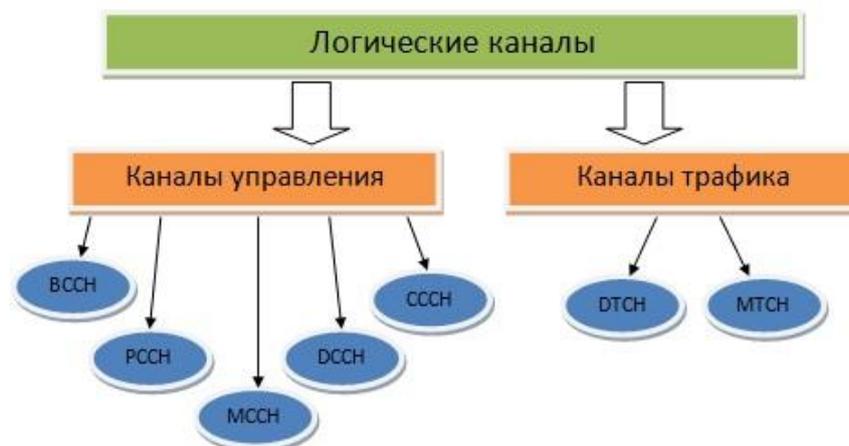


Рис. 5.102. Логические каналы LTE

Логические каналы подразделяются по типам передаваемой информации на каналы управления и на трафиковые каналы.

**К каналам управления относятся:**

□ BCCH (Broadcast Control Channel) – вещательный канал управления – служит для передачи системной служебной информации в Downlink.

□ PCCH (Paging Control Channel) – пейджинговый канал управления – предназначен для передачи пейджинговых сообщений к UE (MC) от eNodeB (BC).

□ MCCH (Multicast Control Channel) – многопользовательский канал управления – необходим для передачи служебной информации одновременно к нескольким абонентским устройствам.

□ DCCH (Dedicated Control Channel) – выделенный канал управления – служит для передачи служебной информации между конкретным абонентским устройством и сетью.

□ CCCH (Common Control Channel) – общий канал управления – предназначен для обмена служебной информацией между UE (MC) и сетью в процедурах начального доступа UE (MC) в сеть до организации выделенного канала.

**К трафиковым каналам относятся:**

□ DTCH (Dedicated Traffic Channel) – выделенный трафиковый канал – основной канал для передачи пользовательских данных между одним конкретным UE (MC) и сетью.

□ MTCH (Multicast Traffic Channel) – многопользовательский трафиковый канал – служит для передачи широковещательной трафиковой информации. Хорошим примером использования этого канала может служить трансляция радио или ТВ-программ.

**Транспортные каналы на радиointерфейсе в LTE**

На радиointерфейсе в сети стандарта LTE применяется стек каналов для передачи данных между абонентским терминалом и сетью. Низший уровень в этом стеке образуют

физические каналы. По ним передаются транспортные, которые в свою очередь несут логические каналы.



Рис. 5.103. Транспортные каналы LTE

Рассмотрим виды транспортных каналов на радиointерфейсе сети стандарта LTE. Все транспортные каналы можно классифицировать по направлению передачи: Uplink (от UE (MC) к eNodeB (BC)) и Downlink (от eNodeB (BC) к UE (MC)).

**К транспортным каналам в Downlink относятся:**

- BCH (Broadcast Channel) – широковещательный канал.
- PCH (Paging Channel) – канал для пейджинга.
- DL-SCH (Downlink Shared Channel) – общий канал для передачи данных вниз.
- MCH (Multicast Channel) – многопользовательский канал.

**К транспортным каналам в Uplink относятся:**

- RACH (Random Access Channel) – канал случайного доступа.
- UL-SCH (Downlink Shared Channel) – общий канал для передачи данных вверх.

Как было сказано выше, транспортные каналы передаются в логических каналах. На рисунке ниже представлена связь между логическими и транспортными каналами в LTE.

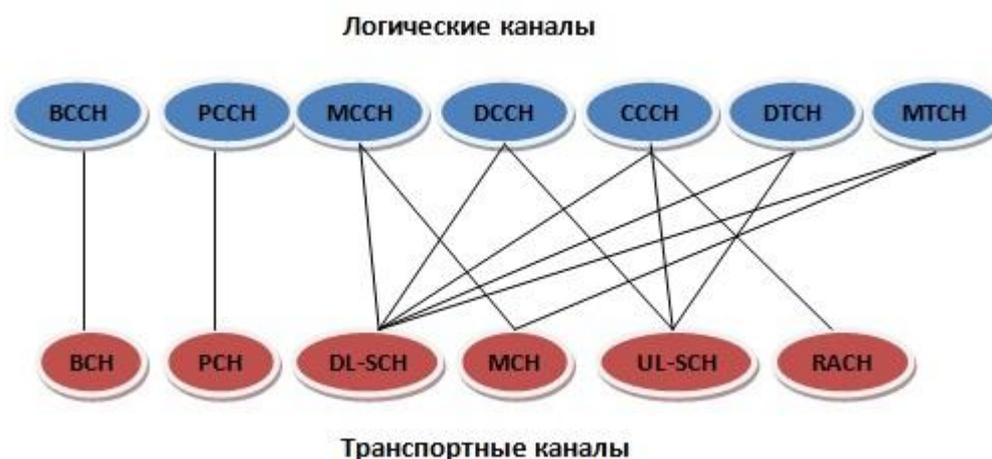


Рис. 5.104. Связь логических и транспортных каналов

### Физические каналы на радиointерфейсе в LTE

Информация между UE (МС) и eNodeB (БС) передается не произвольным образом, а через четко организованную структуру каналов. Рассмотрим классификацию, виды и назначение физических каналов в сети LTE.



Рис. 5.105. Физические каналы LTE

Физические каналы можно классифицировать по направлению передачи информации: Downlink и Uplink.

#### К физическим каналам в Downlink относятся:

- PDSCH (Physical Downlink Shared Channel) - физический распределенный канал в направлении «вниз» - служит для высокоскоростной передачи мультимедийной информации.
- PDCCH (Physical Downlink Control Channel) – физический канал управления в направлении «вниз» - предназначен для передачи информации для управления конкретным UE (МС).
- CCPCH (Common Control Physical Channel) – общий физический канал управления – необходим для передачи общей для всех информации.

#### К физическим каналам в Uplink относятся:

- PRACH (Physical Random Access Channel) – физический канала произвольного доступа – служит для первичного доступа в сеть.
- PUCCH (Physical Uplink Control Channel) – физический канал управления в направлении «вверх» - необходим для передачи служебной информации от конкретной UE (МС) к eNodeB (БС).
- PUSCH (Physical Uplink Shared Channel) – физический распределенный канал в направлении «вверх» - предназначен для высокоскоростной передачи данных в Uplink.

Связь между транспортными и физическими каналами представлена на рисунке ниже.

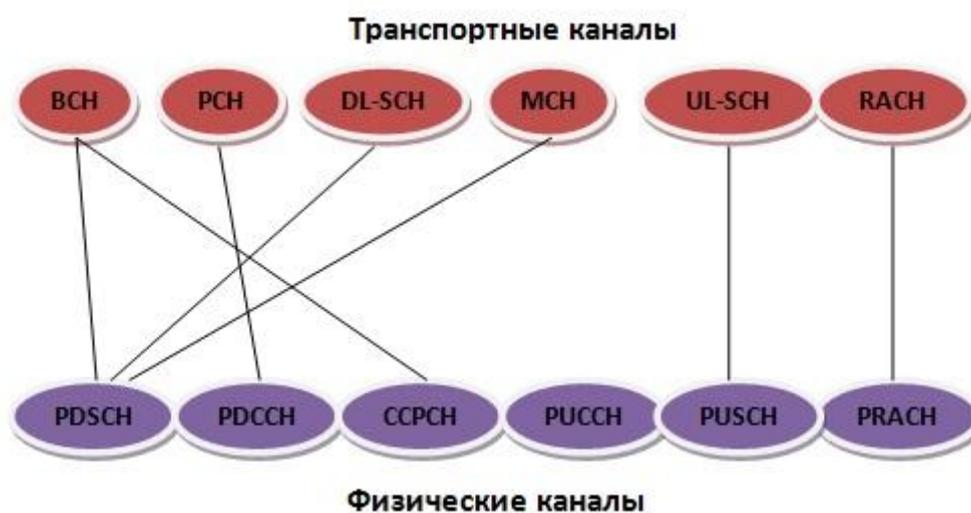


Рис. 5.106. Связь физических и транспортных каналов

### Основные параметры LTE

Таблица 5.16. Основные параметры LTE

Название параметра	Параметр
Uplink (UL): восходящее соединение	SC-FDMA
Downlink (DL): нисходящее соединение	OFDMA
Ширина частотного диапазона, МГц	1,4; 3, 5; 10; 15; 20
Минимальный интервал между кадрами, мс	1
Шаг (частотный интервал) между поднесущими, кГц	15
Стандартная длина префикса CP, мкс	4,7
Увеличенная длина префикса CP, мкс	16,7
Схемы модуляции (Uplink)	BPSK, QPSK, 8PSK, 16QAM

Схемы модуляции (Downlink)	QPSK, 16QAM, 64QAM
Пространственное мультиплексирование	Один канал для Uplink-трафика на каждый абонентский терминал; До 4 каналов для Downlink-трафика на каждый абонентский терминал; MU-MIMO с поддержкой для восходящего (Uplink) и нисходящего (Downlink) соединений

### Практическая реализация

Как было сказано выше, на практике будет реализован канал Downlink системы мобильной связи стандарта LTE. Структура данного канала представлена на рисунке 5.107 [21].

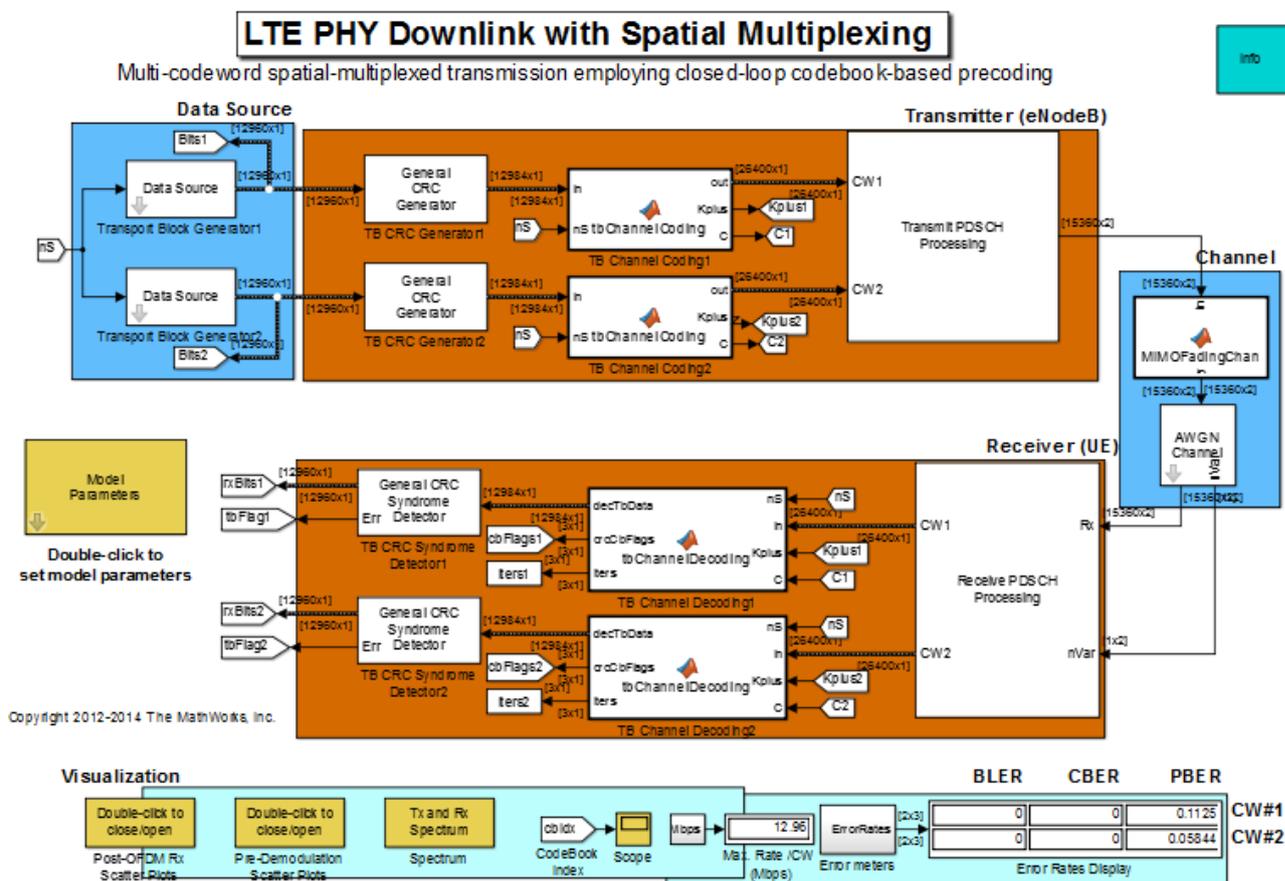


Рис. 5.108. Канал Downlink LTE Simulink MATLAB 2015b

Рассмотрим более подробно данный канал.

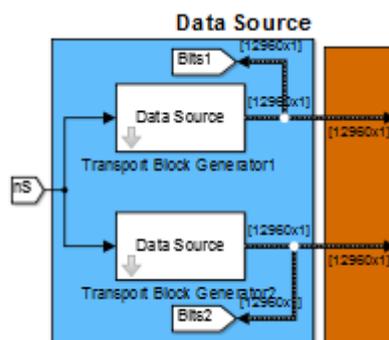


Рис. 5.109. Разделение исходного потока бит на параллельные потоки

- Кодирование помехоустойчивым кодом, в процессе которого значительно увеличивается число символов в отдельных потоках. В данной схеме используется код CRC.

Каждый отдельный параллельный поток кодируется данным кодом с заданным полиномом.

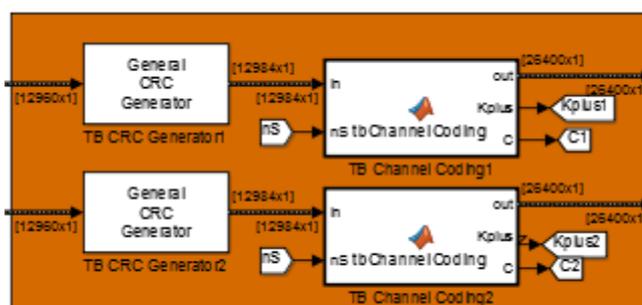


Рис. 5.110. Кодирование помехоустойчивым кодом

General CRC Generator (mask) (link)

Generate CRC bits according to the generator polynomial parameter and append them to the input data frames. Specify the generator polynomial as either a string expressing the polynomial in algebraic form, a hexadecimal string, or as a binary or integer row vector with coefficients in descending order of powers.

This block accepts a binary column vector input signal.

Parameters

Generator polynomial:

[1 1 0 0 0 0 1 1 0 0 1 0 0 1 1 0 0 1 1 1 1 1 0 1 1] % qCRC24A

Initial states:

0

Direct method

Reflect input bytes

Reflect checksums before final XOR

Final XOR:

0

Checksums per frame:

1

Рис. 5.111. Параметры CRC кодера

• Манипуляция выбранным в данный конкретный момент способом модуляции. В канале Downlink используются методы манипуляции: QPSK, 16QAM, 64QAM. Далее перемножение полученной последовательности каждого потока на свою поднесущую и БПФ (так называемая OFDM – модуляция). Где в результате получаем один сложный сигнал.

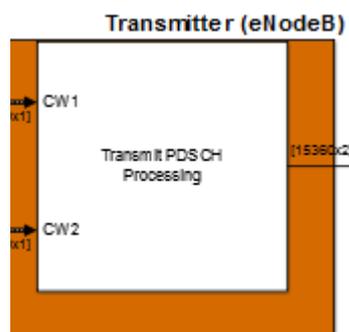


Рис. 5.112. Манипуляция выбранным в данный конкретный момент способом модуляции

Структура этого блока имеет следующий вид:

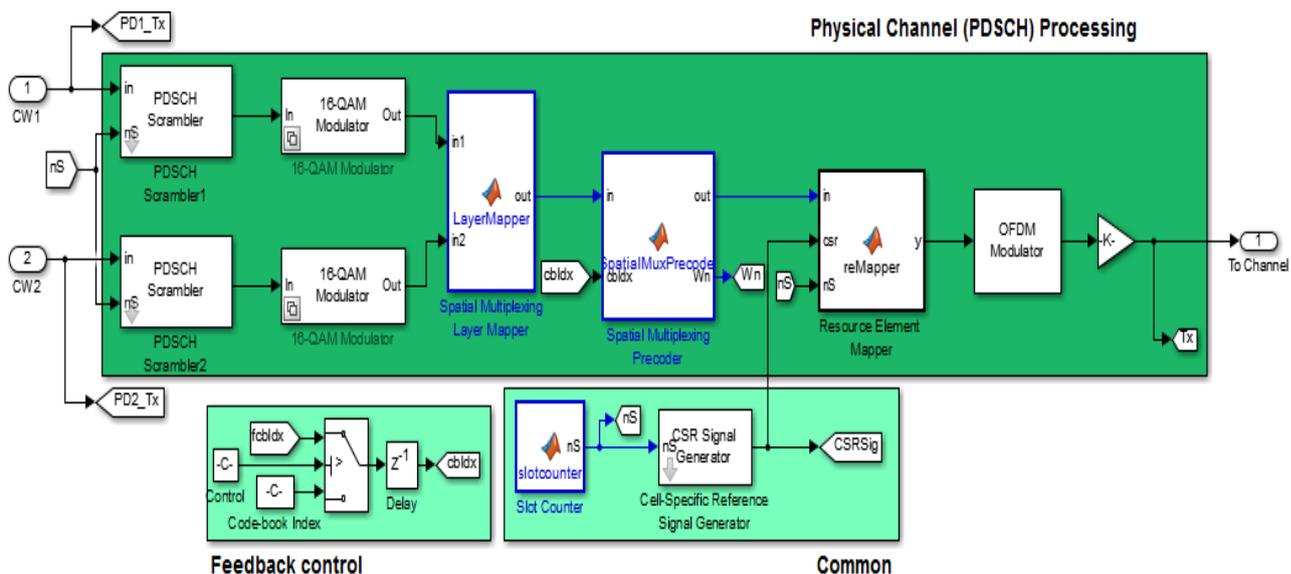


Рис. 5.113. Квадратурная манипуляция и получение OFDM символов

• Передача в эфир. Для этого используется технология MIMO 2x2 или 4x4 приемных/передающих антенн. Где один общий поток (сигнал) разделяется на 2 потока (2x2 антенна) или 4 потока (4x4 антенна).

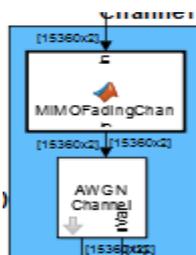


Рис. 5.114. Передача в эфир



Рис. 5.115. Характеристики блока БГШ (AWGN)

Далее подпотоки ММО объединяются в один поток, который приходит на мобильную станцию под воздействием помех.

Далее мобильная станция производит обратные преобразования, реализованные выше, а именно, получаем паралельные потоки. Потом производится обратное быстрое преобразование Фурье (ОБПФ). Затем производится демодуляция.

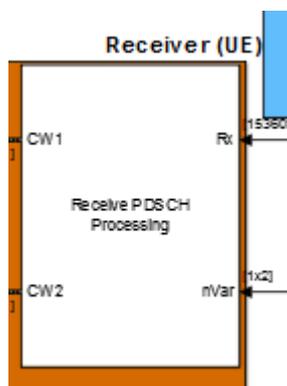


Рис. 5.116. Паралельные потоки-ОБПФ-демодуляция

Схема, входящая в данный блок:

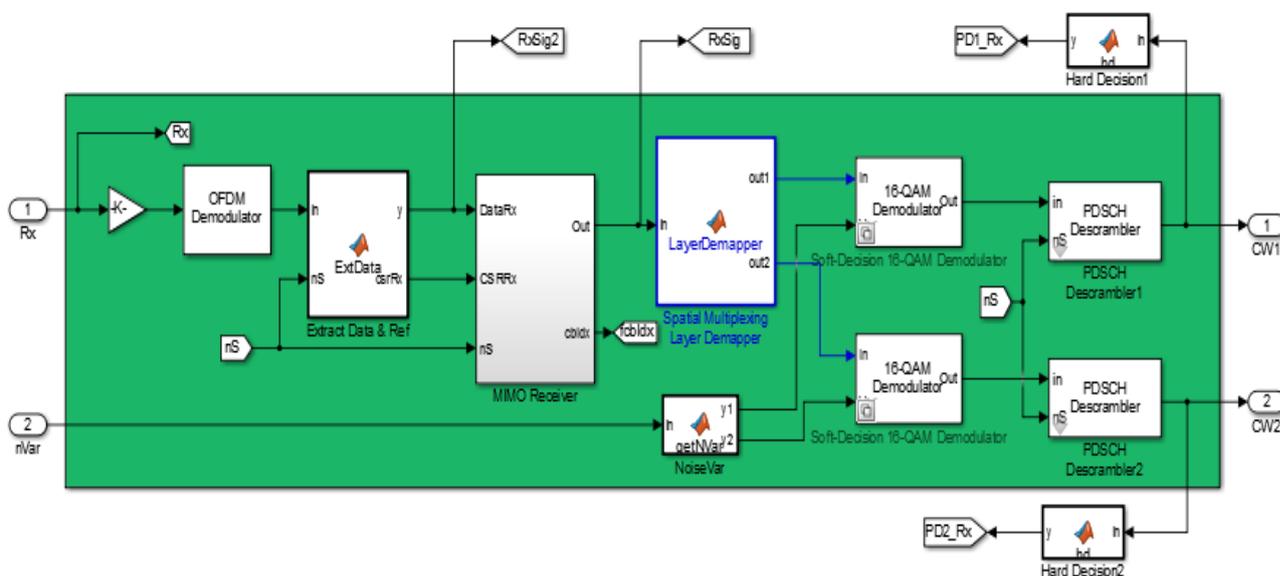


Рис. 5.117. Паралельные потоки-ОДПФ-демодуляция

Далее производится декодирование по соответствующему алгоритму CRC:

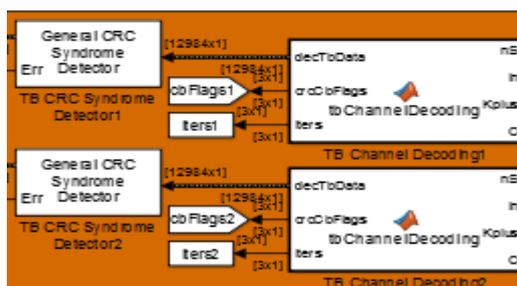


Рис. 5.118. Декодирование CRC

General CRC Syndrome Detector (mask) (link)

Detect errors in the input data frames according to the generator polynomial parameter. Specify the generator polynomial as either a string expressing the polynomial in algebraic form, a hexadecimal string, or as a binary or integer row vector with coefficients in descending order of powers.

The first output is the data frame with the CRC bits removed and the second output indicates if an error was detected in the data frame.

This block accepts a binary column vector input signal.

Parameters

Generator polynomial:

`[1 1 0 0 0 0 1 1 0 0 1 0 0 1 1 0 0 1 1 1 1 1 0 1 1] % gCRC24A`

Initial states:

0

Direct method

Reflect input bytes

Reflect checksums before final XOR

Final XOR:

0

Рис. 5.119. Характеристики декодера CRC

После декодирования производится преобразование параллельных потоков в один исходный поток:

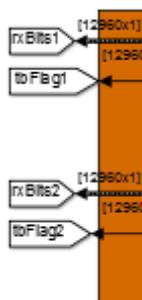


Рис. 5.120. Получение исходного потока

Данная схема позволяет формировать характеристики передачи данных по этому каналу, а именно это ширина спектра, количество антенн в ММО, вид модуляции, отношение сигнал/шум:

Model Parameters (mask)  
Specifies model parameters for a simulation run.

Parameters

Channel bandwidth (MHz) : 10

Control region (number of OFDM symbols per subframe):  
2

Antenna configuration: 2x2

PDSCH modulation type: 16QAM

Target coding rate:  
1/2

Fading channel model: EPA 0Hz

SNR (dB):  
12.1

Enable PMI feedback

Maximum decoding iterations:  
8

Disable transport-block level early termination

Рис. 5.121. Характеристики канала

В результате работы схемы можно получить некоторые зависимости:

1. Спектр передаваемого и принятого сигнала.
2. Диаграмму созвездий передаваемого и принятого сигнала (для каждой из антенн ММО).
3. Итерации декодера в зависимости от времени и кодовых слов для каждого параллельного потока.

Также можно построить зависимость битовой вероятности ошибки при заданном отношении сигнал/шум каждого параллельного потока отдельно, меняя значения отношения сигнал/шум.

BLER	CBER	PBER	
0	0	0.1125	CW#1
0	0	0.05844	CW#2

Error Rates Display

Рис. 5.122. Информация о битовой вероятности ошибки параллельных потоков  
В качестве примера зададим следующие характеристики передачи данных:

- Ширина спектра - 10 МГц.
- Количество антенн MIMO – 4x4.
- Модуляция – QPSK.
- Отношение сигнал/шум – 1 дБ.

Model Parameters (mask)  
Specifies model parameters for a simulation run.

Parameters

Channel bandwidth (MHz) : 10

Control region (number of OFDM symbols per subframe):  
2

Antenna configuration: 4x4

PDSCH modulation type: QPSK

Target coding rate:  
1/2

Fading channel model: EPA 0Hz

SNR (dB):  
1

Enable PMI feedback

Maximum decoding iterations:  
8

Рис. 5.123. Характеристики передачи данных

В результате получим следующие зависимости:

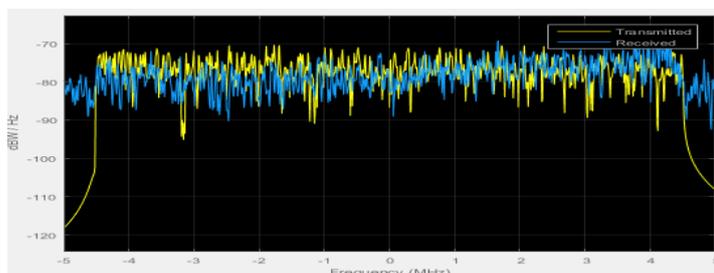


Рис. 5.124. Спектр входного (желтым) и выходного (синим) сигналов

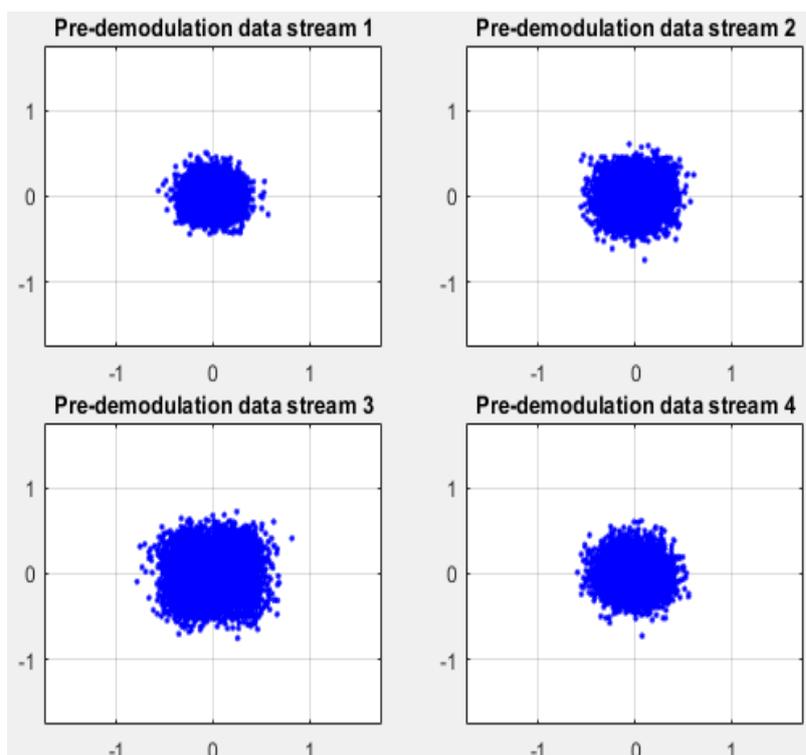


Рис. 5.125. Диаграмма созвездий переданного сигнала для каждой из антенн ММО

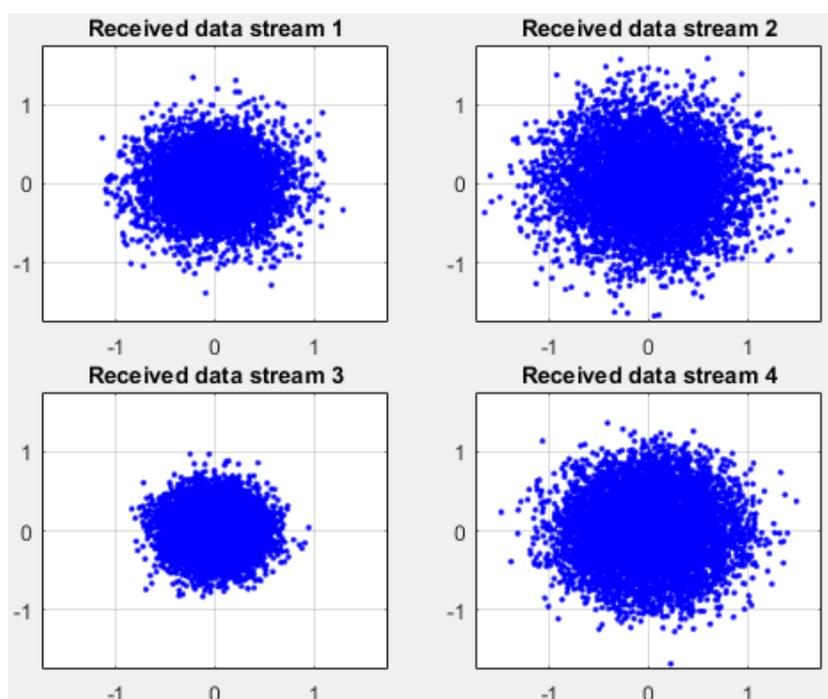


Рис. 5.126. Диаграмма созвездий принятого сигнала для каждой из антенн ММО

На основании полученных значений, построим зависимость.

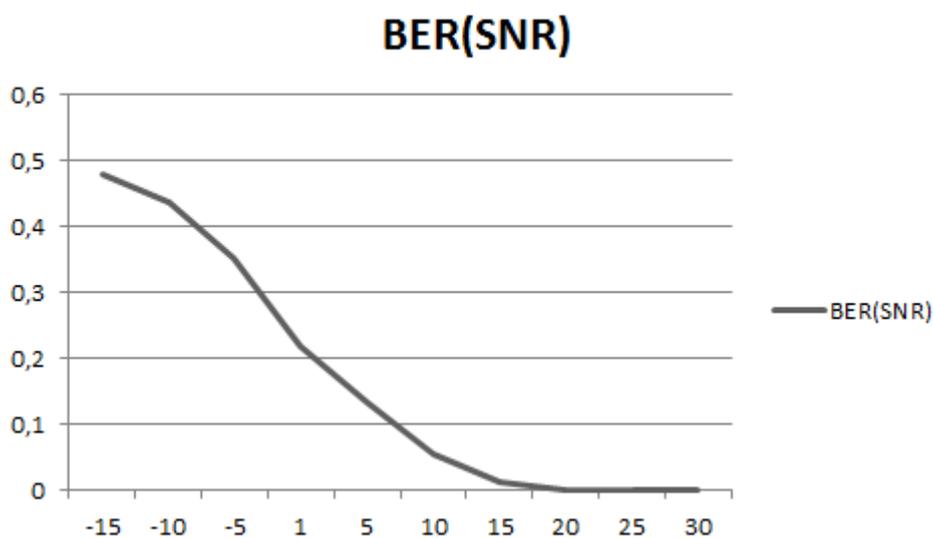


Рис. 5.127. Зависимость битовой вероятности ошибки от отношения сигнал/шум для первого потока

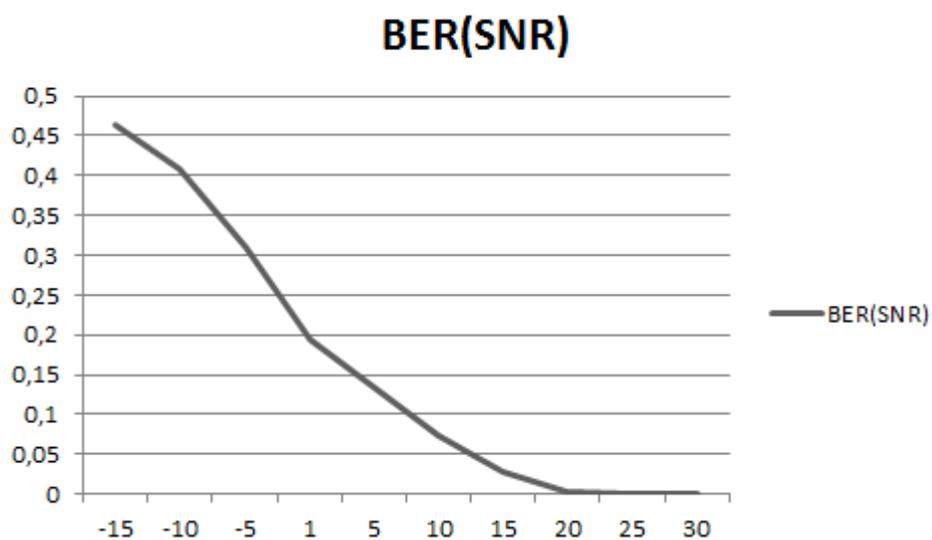


Рис. 5.128. Зависимость битовой вероятности ошибки от отношения сигнал/шум для второго потока

В качестве еще одного примера зададим следующие характеристики передачи данных:

- Ширина спектра - 10 МГц.
- Количество антенн ММО – 2x2.
- Модуляция – QPSK.
- Отношение сигнал/шум – 1 дБ.

**Model Parameters (mask)**  
Specifies model parameters for a simulation run.

**Parameters**

Channel bandwidth (MHz) : 10

Control region (number of OFDM symbols per subframe):  
2

Antenna configuration: 2x2

PDSCH modulation type: QPSK

Target coding rate:  
1/2

Fading channel model: EPA 0Hz

SNR (dB):  
1

Enable PMI feedback

Maximum decoding iterations:  
8

Disable transport-block level early termination

Рис. 5.129. Характеристики передачи данных

В результате получим следующие зависимости:

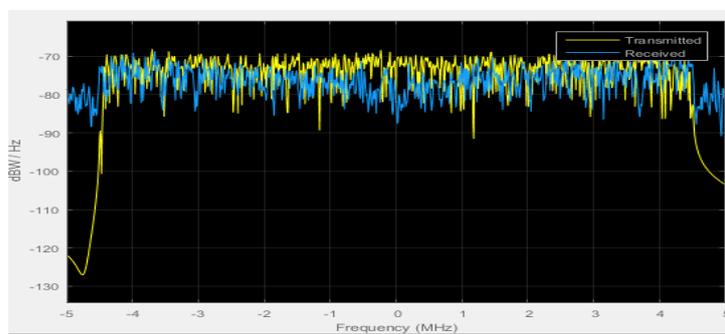


Рис. 5.130. Спектр входного (желтым) и выходного (синим) сигналов

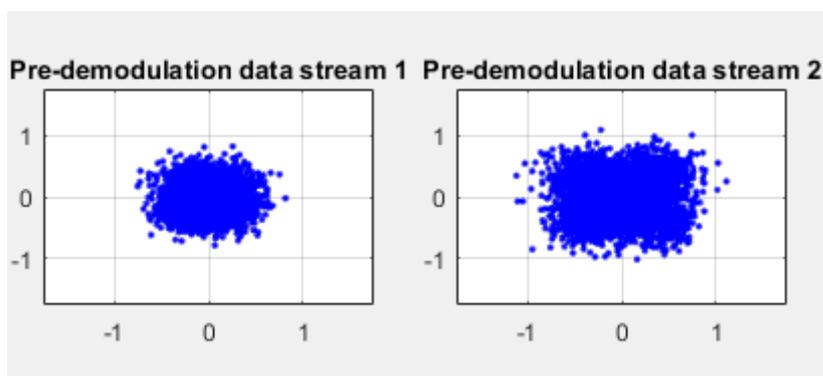


Рис. 5.131. Диаграмма созвездий переданного сигнала для каждой из антенн MIMO

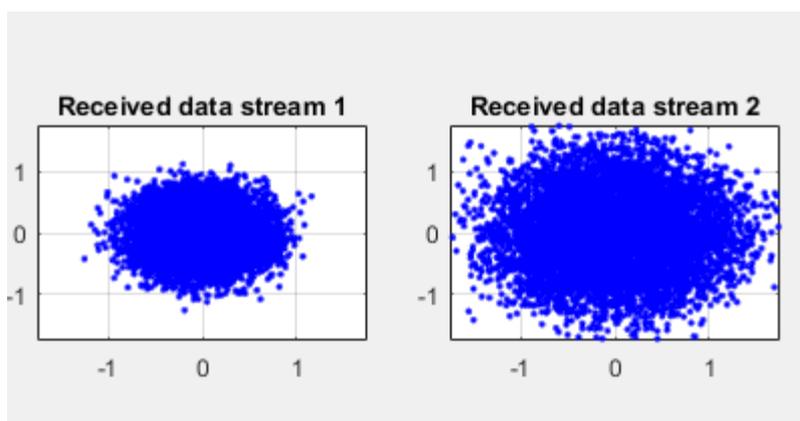


Рис. 5.132. Диаграмма созвездий принятого сигнала для каждой из антенн ММО

Изменим отношение сигнал/шум – -15, -10, -5, 1, 5, 10, 15, 20, 25 и 30 дБ и построим зависимость битовой вероятности ошибки от отношения сигнал/шум для десяти точек для обоих параллельных потоков

На основании полученных значений, построим зависимость.

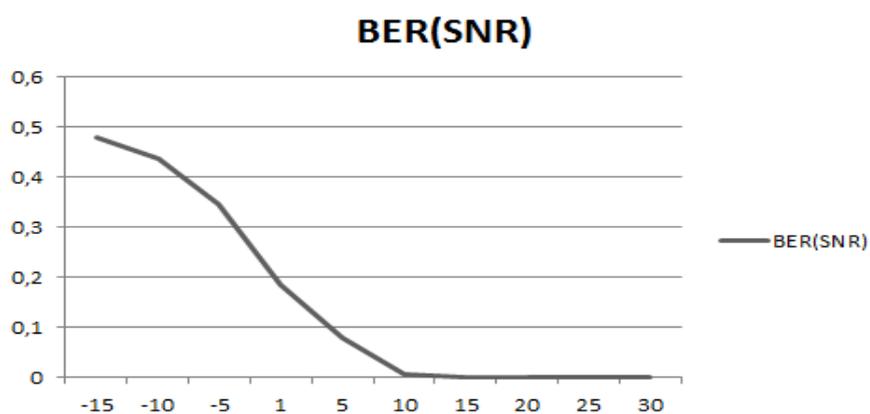


Рис. 5.133. Зависимость битовой вероятности ошибки от отношения сигнал/шум для первого потока

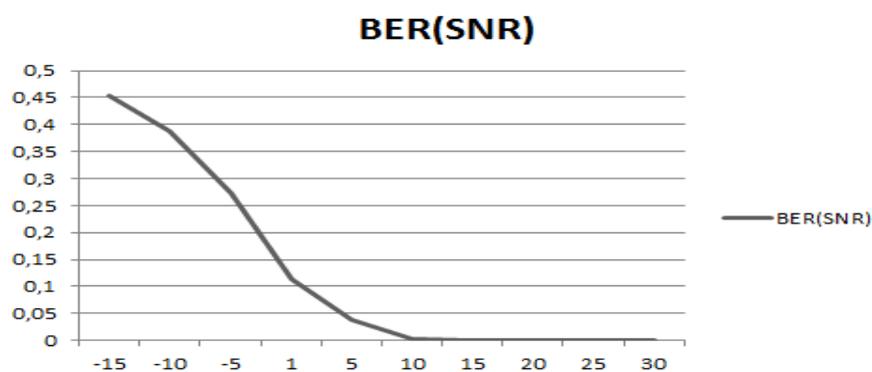


Рис. 5.134. Зависимость битовой вероятности ошибки от отношения сигнал/шум для второго потока

## Методика и проведение исследования канала Downlink

Запустить Matlab 15 от имени администратора (обязательно).

В результате запуска на экране монитора появится следующее окно:

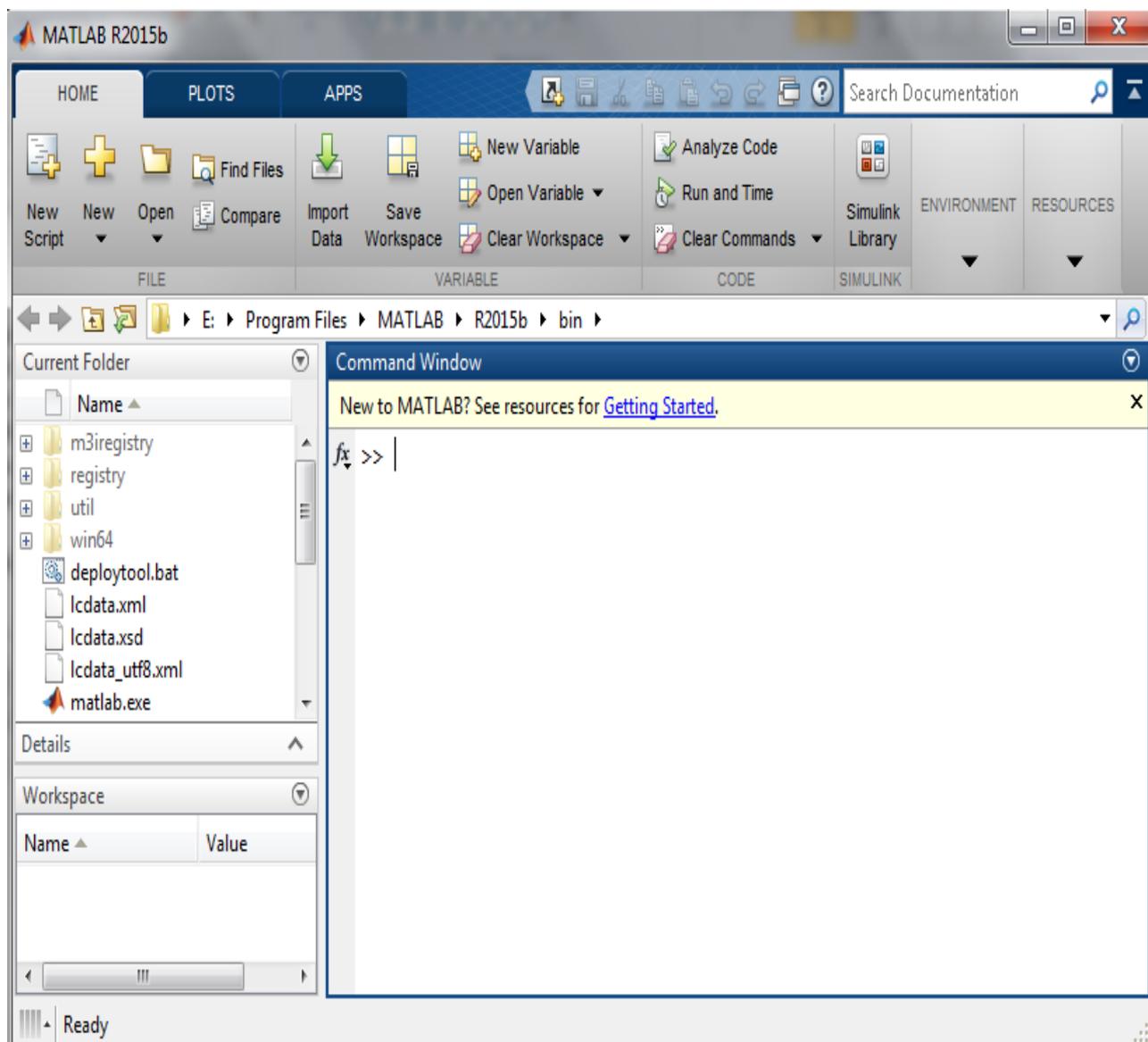


Рис. 5.135. Диалоговое окно Matlab 15

1. В командной строке программы прописать: `cd ../` (при пропуске данного пункта могут возникнуть проблемы при компиляции).
2. В командной строке программы прописать `LTEDownlinkExample`, в результате откроется окно со схемой в программе, которое имеет следующий вид:

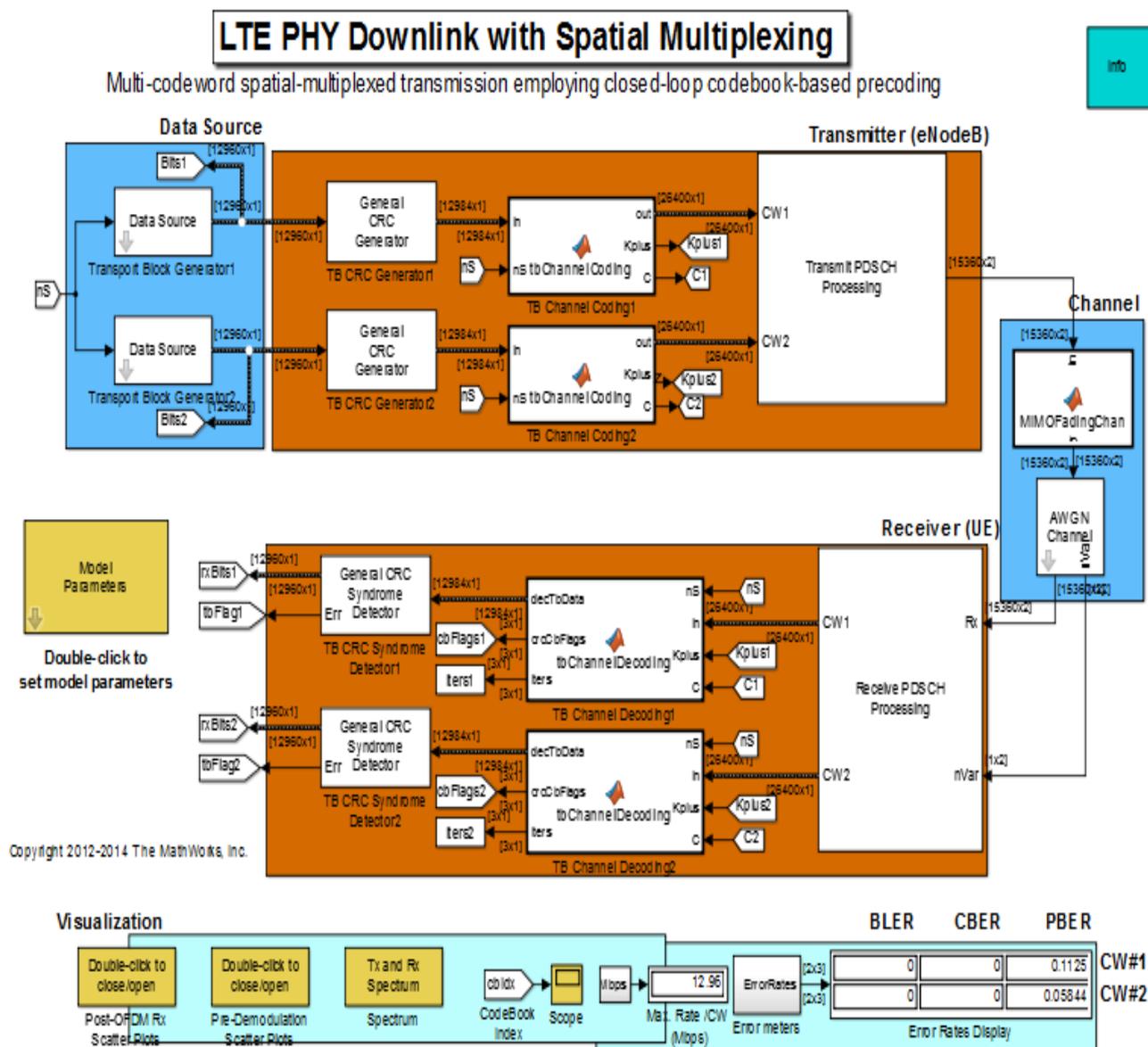


Рис. 5.136. Схема канала Downlink

3. Задать следующие характеристики передачи данных:

- Модуляция – QAM-16.
- Количество антенн MIMO 2x2.
- Ширина спектра – 10 МГц.

4. Изменять отношение сигнал/шум – -15, -10, -5, 1, 5, 10, 15, 20, 25, 30 дБ (SNR) и на каждом его значении фиксировать в отчете выходные зависимости, а именно спектр переданного и принятого сигналов, диаграммы созвездий на каждой из антенн MIMO для переданного и принятого сигналов, итерации декодера в зависимости от времени и кодовых слов первого потока и второго потока. Так же на каждом шаге фиксировать значение битовой вероятности ошибки (BER) обоих параллельных

потоков. И после окончания исследования построить зависимости BER от SNR обоих параллельных потоков.

#### 5. Содержание отчета

- Титульный лист.
- Цель работы.
- Теория канала Downlink.
- Исследуемая схема канала Downlink.
- Результаты работы по пунктам 6 и 7.
- Заключение.

В результате выполнения в разделе были выполнены следующие мероприятия:

1. Проведен теоритический анализ стандарта мобильной связи стандарта LTE. Проведен анализ сравнения данного стандарта с уже устаревающими стандартами на данный момент – UMTS (3G) и GSM (2G). Также было проведено аналитическое исследование физических каналов стандарта – Downlink (от БС к МС) и Uplink (от МС к БС), а также логические и транспортные каналы. Приведены обобщенные схемы формирования данных каналов.

2. Путем проведения компьютерной симуляции, была проверена достоверность теоритического исследования. В программе Matlab 15 были собрана схема канала Downlink.

3. С помощью компьютерной симуляции были получены различного рода зависимости при передаче информации по каналу. Самая важная из них это зависимость битовой вероятности ошибки от отношения сигнал/шум. В результате получились следующие значения:

Таблица 5.17. Зависимость BER от SNR при MIMO 4x4

<b>сигнал/шум (дБ)</b>	<b>1 поток (BER)</b>	<b>2 поток (BER)</b>
<b>-15</b>	0,4766	0,4537
<b>-10</b>	0,4347	0,3876
<b>-5</b>	0,3456	0,2729
<b>1</b>	0,1836	0,114
<b>5</b>	0,0774	0,039
<b>10</b>	0,0076	0,0027

<b>15</b>	0,000001	0,000002
<b>20</b>	<b>0</b>	<b>0</b>
<b>25</b>	<b>0</b>	<b>0</b>
<b>30</b>	<b>0</b>	<b>0</b>

Таблица 5.18. Зависимость BER от SNR при MIMO 2x2

<b>сигнал/шум (дБ)</b>	<b>1 поток (BER)</b>	<b>2 поток (BER)</b>
<b>-15</b>	0,4772	0,4645
<b>-10</b>	0,4354	0,4073
<b>-5</b>	0,3512	0,3096
<b>1</b>	0,2178	0,1933
<b>5</b>	0,1347	0,1345
<b>10</b>	0,056	0,074
<b>15</b>	0,0137	<b>0,0282</b>
<b>20</b>	0,0003	0,0037
<b>25</b>	<b>0</b>	0,000003
<b>30</b>	<b>0</b>	<b>0</b>

4. Анализируя полученные значения таблицы 3, можно сделать следующий вывод, что при увеличении отношения сигнал/шум, битовая вероятность ошибки стремится к нулю быстрее в MIMO 4x4, нежели в MIMO 2x2. Таким образом, использование большего числа прямо-передающих антенн, дает меньшие ошибки.

5. Была написана методика исследования канала Downlink.

б. Также я познакомился с различным программным обеспечением, для построения различного вида схем.

Подводя итог своего курсового проекта, можно сказать следующее, то, что я сделал, является основополагающим делом к дальнейшим, более трудным вещам. Курсовой проект был весьма увлекательным и полезным. С поставленными целями справился успешно.

### **5.8. Имитационное моделирование системы цифрового наземного телевизионного вещания DVB-T [16, 20]**

Объектом исследования является система цифрового телевидения на базе стандарта DVB-T. Цель лабораторной работы – закрепленных знаний полученных при изучении дисциплины «Основы проектирования защищенных телекоммуникационных систем». Задача лабораторной работы – исследование основных характеристик системы цифрового наземного телевидения стандарта DVB-T.

В 1993 году группа ведущих европейских компаний-производителей вещательного оборудования образовала некоммерческую организацию по разработке стандартов цифрового телевизионного вещания, получившую название DVB Project.

Для каждой транспортной среды был разработан стандарт обработки и передачи транспортно потока, учитывающий ее специфику и в то же время максимально инфицированный со смежными стандартами. Для упрощения взаимного обмена программами выбраны такие параметры обработки, чтобы пропускная способность и число передаваемых ТВ программ во всех случаях оставались бы примерно одинаковыми. Документ для спутникового вещания получил сокращенное наименование DVB-S, для сетей кабельного телевидения - DVB-C, для наземного (эфирного) телевидения DVB-T.

#### **Концепция стандарта DVB-T**

Одним из первых решений данной организации было решение принять за основу всех разработок стандарт цифрового сжатия MPEG-2. Однако, данный стандарт не охватывает передачу цифрового сигнала по каналам связи и его необходимо дополнить документами, регламентирующими обработку сигнала перед подачей в канал.

Второе важное решение – использование общего MPEG-2 мультиплекса во всех средах распространения и максимальная унификация методов помехоустойчивого кодирования и модуляции. Во всех случаях используется код Рида-Соломона с единым размером блока, и в тех случаях где это необходимо, - сверточный код с единым набором относительных скоростей. Очень важна для широкого круга концепция «контейнера данных» - создание

универсального цифрового канала, переносящего видео, аудио, данные пользователя в любых пропорциях и с высокими показателями качества обслуживания.

Передаваемая информация в системе DVB-T представляет собой пакеты транспортного потока MPEG-2. Для рассматриваемой системы содержание контейнера не имеет значение, она лишь приспособливает данные транспортного мультиплексора MPEG-2 к свойствам и характеристикам канала передачи наземного телевизионного вещания, стремясь наиболее эффективно донести их приемнику. Иными словами, стандарт DVB-T определяет только структуру передаваемого потока данных, систему канального кодирования и модуляции.

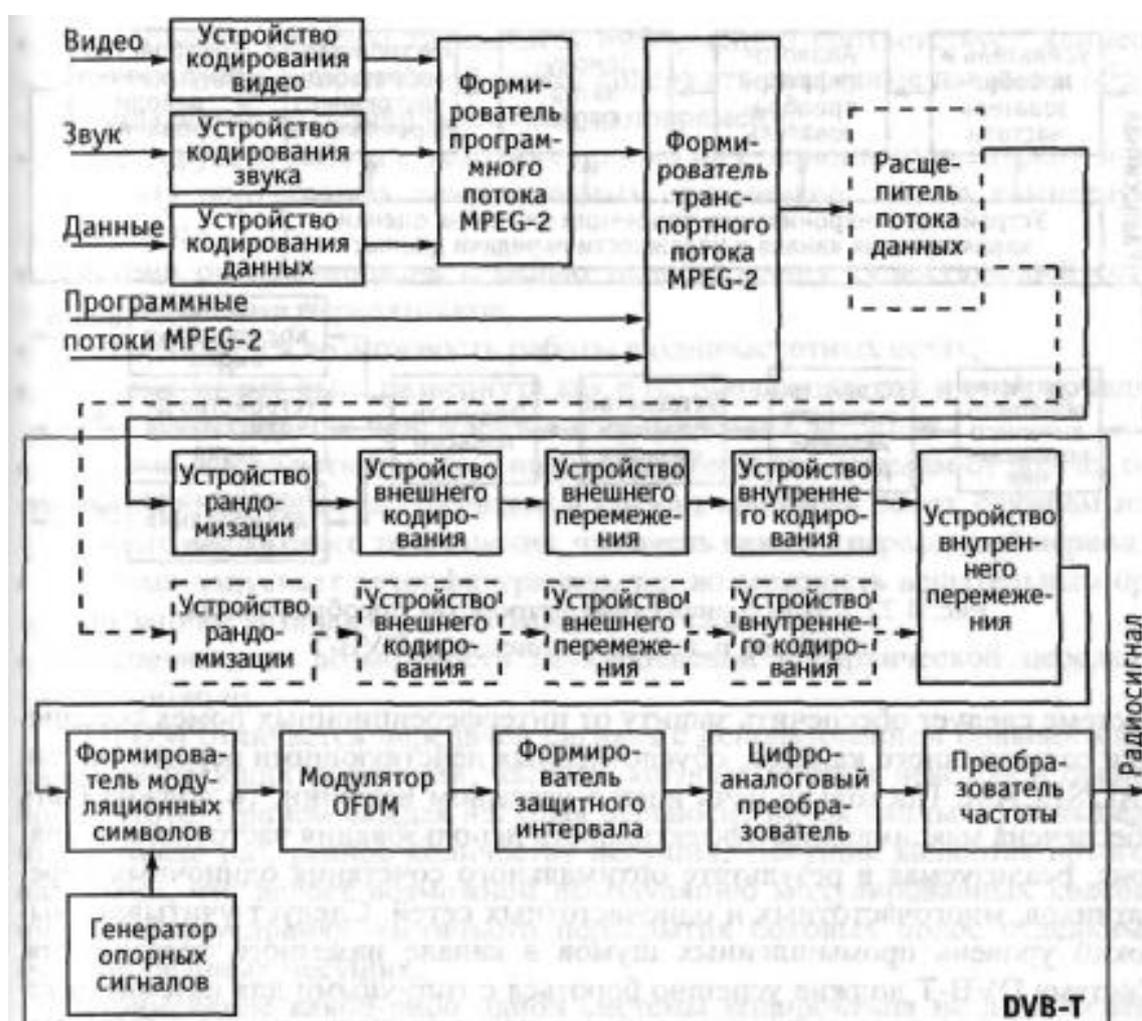


Рис. 5.137. Структурная схема передатчика системы DVB-T

На приемной стороне выполняются операции, обратные операциям производимым в приемнике.



Рис. 5.138. Структурная схема приемника системы DVB-T

### Обработка данных и сигналов в системе DVB-T

#### Адаптация транспортных пакетов MPEG-2 в системе DVB-T. Рандомизация

Адаптация транспортных пакетов MPEG-2 в исследуемой системе осуществляется путем включения в общий поток информационных 187 байт одного байта синхронизации.



Рис. 5.139. Адаптация транспортных пакетов в системе MPEG-2

Рандомизация данных является первой операцией, выполняемой в системе DVB-T. Ее цель – превратить цифровой сигнал в квазислучайный и тем самым решить две важные задачи. Первая – обеспечение возможности выделения из него тактовых импульсов (самосинхронизация). Вторая – приведение более равномерного энергетического спектра излучаемого радиосигнала. Рандомизация осуществляется путем сложения по модулю 2, то есть посредством логической операции «исключающее ИЛИ» цифрового потока данных и двоичной псевдослучайной последовательности.



Рис. 5.140. Структурная схема устройства рандомизации данных

### Внешнее кодирование и перемежение

Как было отмечено выше, в системе внешнего кодирования для защиты всех 188 байт транспортного пакета (включая байт синхронизации) используется код Рида-Соломона (204, 188). В процессе кодирования к этим 188 байтам добавляется 16 проверочных байт. Стоит отметить, что при декодировании на приемной стороне это позволяет исправлять до восьми ошибочных байт в пределах каждого кодового слова длиной 204 байта.

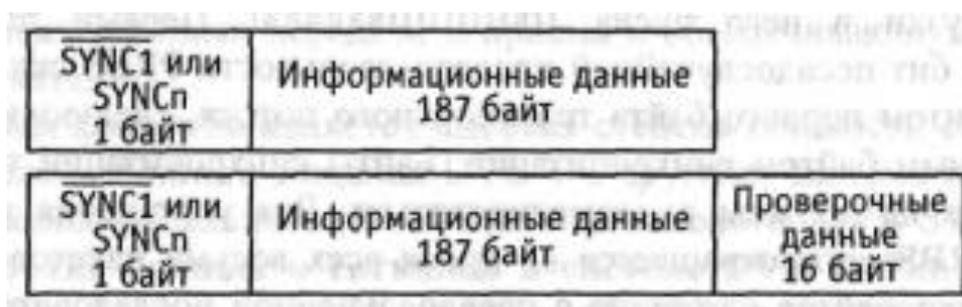


Рис. 5.141. Структурная схема устройства рандомизации данных

Внешнее перемежение осуществляется путем изменения порядка следования байт в пакетах, защищенных от ошибок. В соответствии со схемой представленной на слайде перемежение выполняется путем последовательного циклического подключения источника и получателя данных к двенадцати ветвям, причем за одно подключение в ветвь направляется и снимается 1 байт данных. В одиннадцати ветвях включены регистры сдвига, содержащие разное количество ячеек и создающие увеличивающиеся от ветви к задержку. Первый же синхробайт поступает в 0 ветвь, которая не содержит задержки, что не создает проблем синхронизации.



Рис. 5.142. Функциональная схема внутреннего перемежителя данных

### Внутреннее кодирование

Функциональная и структурная схема кодера/декодера сверточного кода используемого в системе DVB-T может быть представлена в следующем виде:

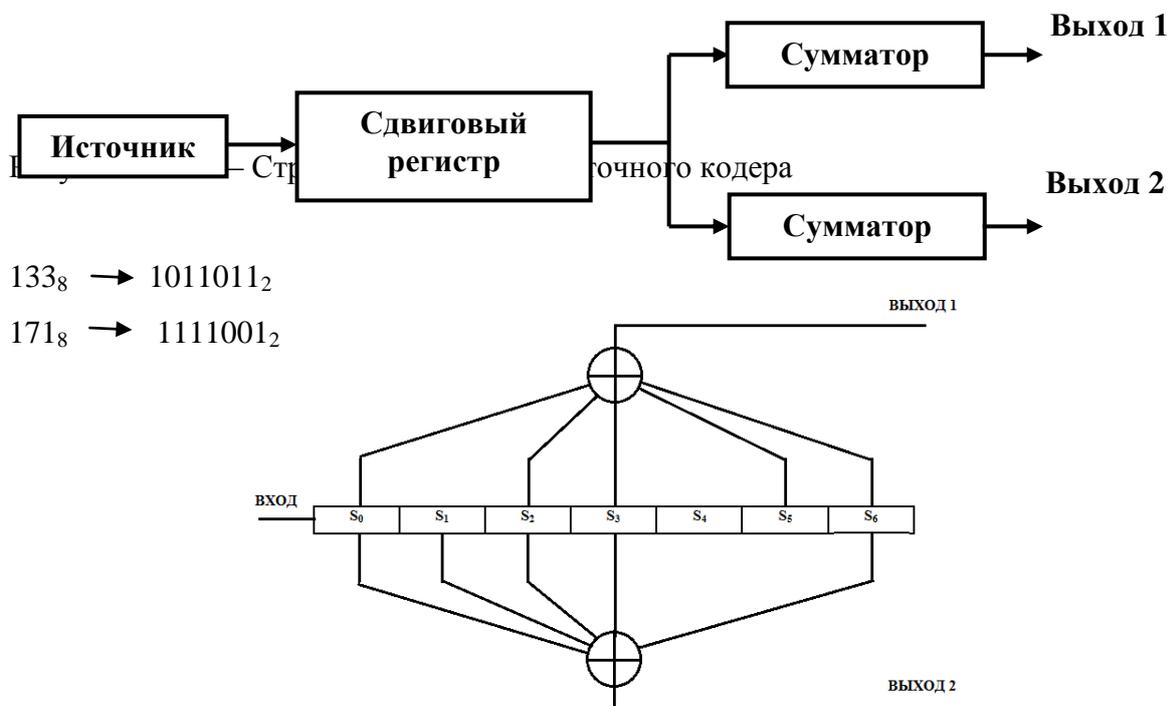


Рис. 5.143. Функциональная схема сверточного кодера 133,171

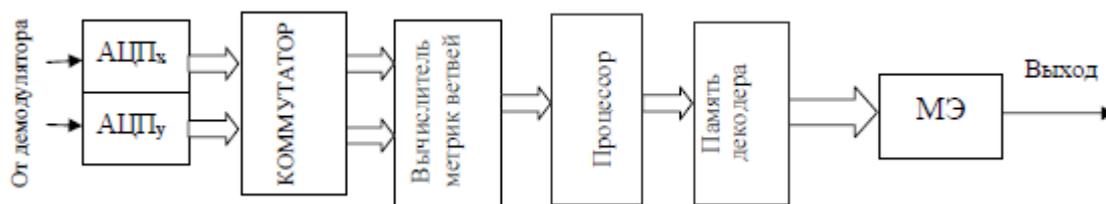


Рис. 5.144. Структурная схема декодера Витерби



Рис. 5.145. Функциональная схема декодера Витерби

### Внутреннее перемежение

Как показано на слайде внутреннее перемежение осуществляется в два этапа. На первом этапе цифровой поток с выхода сверточного кодера разделяется на  $m$  парциальных потоков. Каждый из потоков делится на блоки длиной 126 битов и поступает на отдельный блоковый перемежитель битов с поразрядным перемежением. Функция перемежения представлена на слайде.

Выходные потоки перемежителей группируются по одному биту с каждого выхода, образуя  $m$ -битовые кодовые слова, поступающие на вход символьного перемежителя.

$$\begin{aligned}
 H_0(w) &= w; \\
 H_1(w) &= (w + 63) \bmod 126; \\
 H_2(w) &= (w + 105) \bmod 126; \\
 H_3(w) &= (w + 42) \bmod 126; \\
 H_4(w) &= (w + 21) \bmod 126; \\
 H_5(w) &= (w + 84) \bmod 126.
 \end{aligned}$$

Рис. 5.146. Функция битового перемежения для QAM-64

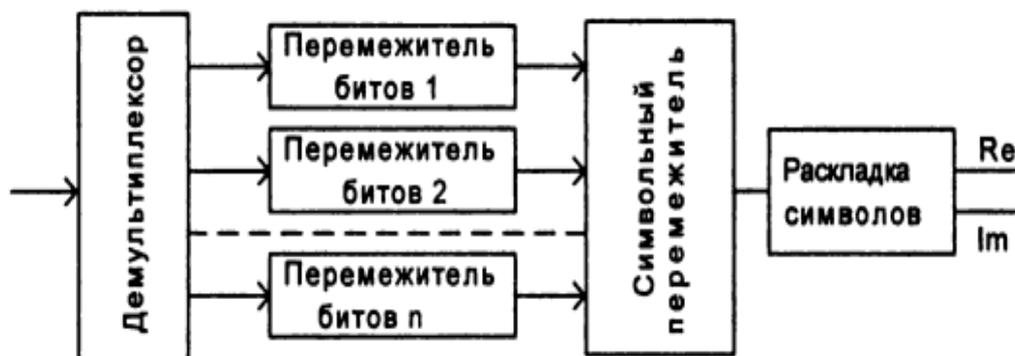


Рис. 5.147. Структурная схема внутреннего перемежителя системы DVB-T

### Модуляция в системе DVB-T

В системе цифрового наземного телевизионного вещания используются следующие виды модуляции: QPSK, QAM-16 и QAM-64. Диаграммы созвездий проиллюстрированы на рисунке 5.178.

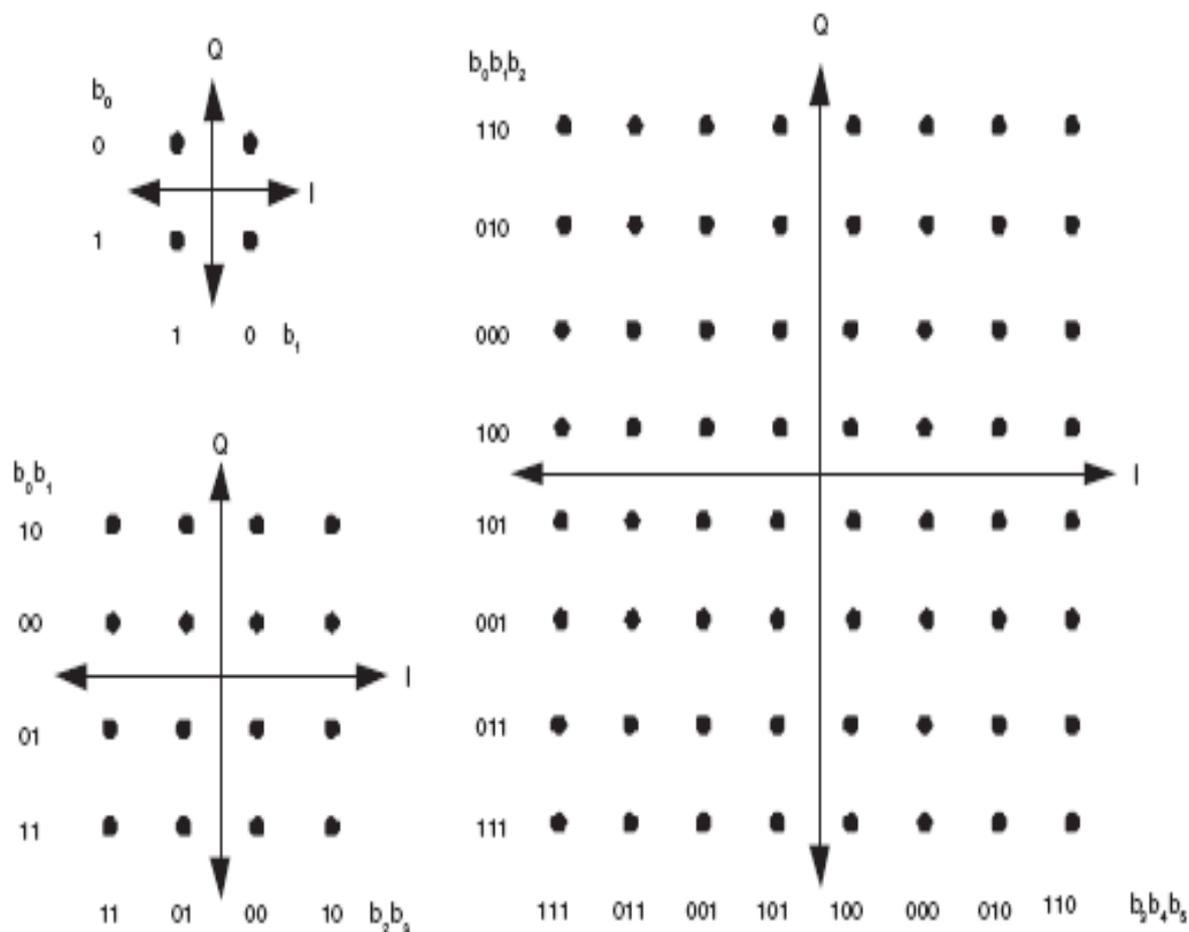


Рис. 5.148. Виды модуляции используемые в системе DVB-T

С внутреннего перемежителя биты поступают на модулятор QAM-64, затем происходит распределение по поднесущим с добавлением пилот-сигналов. К сформированному спектру применяется операция обратного быстрого преобразования Фурье (IFFT), добавляется циклический префикс. Далее полученный символ передается через канал с шумом и затем производятся обратные операции в приёмнике.

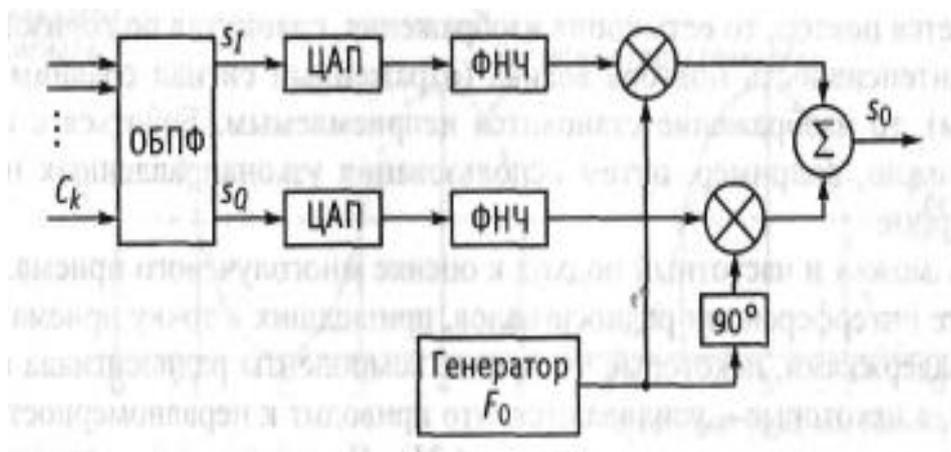


Рис. 5.149. Структурная схема формирователя OFDM-символа в системе DVB-T



Рис. 5.150. Структура OFDM-символа в системе DVB-T

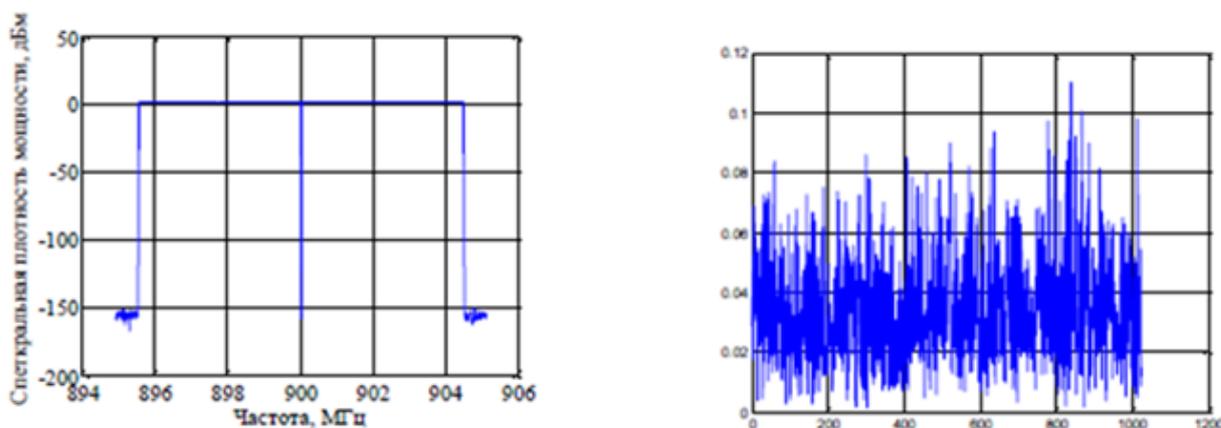


Рис. 5.151. Спектральное и временное представление OFDM-сигнала

## Практическая часть [21]

В первую очередь была запущена модель системы DVB-T в программе Matlab следующим образом: Matlab R2015b – Simulink Library Browser – Open – dvbt.slx.

### – Simulink

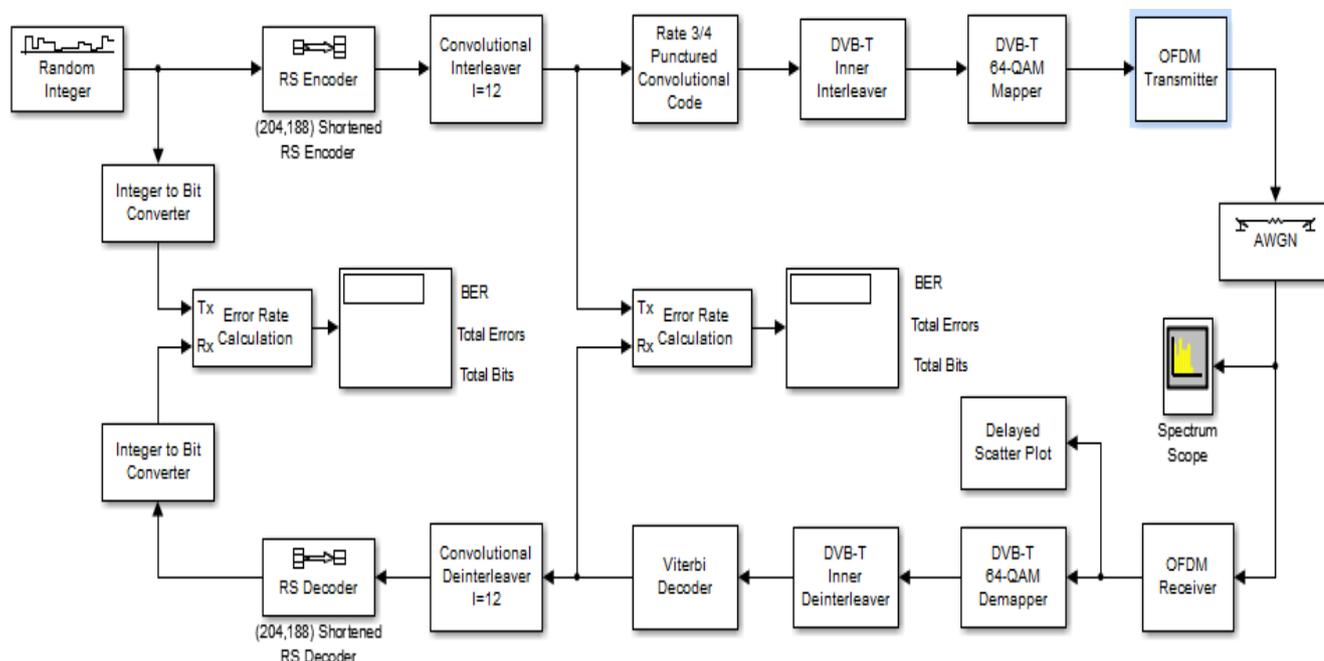


Рис. 5.152. Функциональная схема системы DVB-T реализованная в Matlab R2015b

Затем была исследована зависимость битовой вероятности ошибки (BER) от отношения сигнал/шум (SNR), путем изменения параметра SNR в блоке AWGN в диапазоне от 1 дБ до 25 дБ с шагом 4 дБ.

Результаты измерений представлены в таблице 5.21.

Таблица 5.21 – Зависимость BER от SNR для системы DVB-T

<b>SNR,</b> <b>дБ</b>	1	5	9	13	17	21	25
<b>BER</b>	0.5	0.49	0.47	0.29	0.001	$5 \times 10^{-6}$	0

На основании данных представленных в таблице 5.1 был построен график зависимости битовой вероятности ошибки от отношения сигнал/шум.

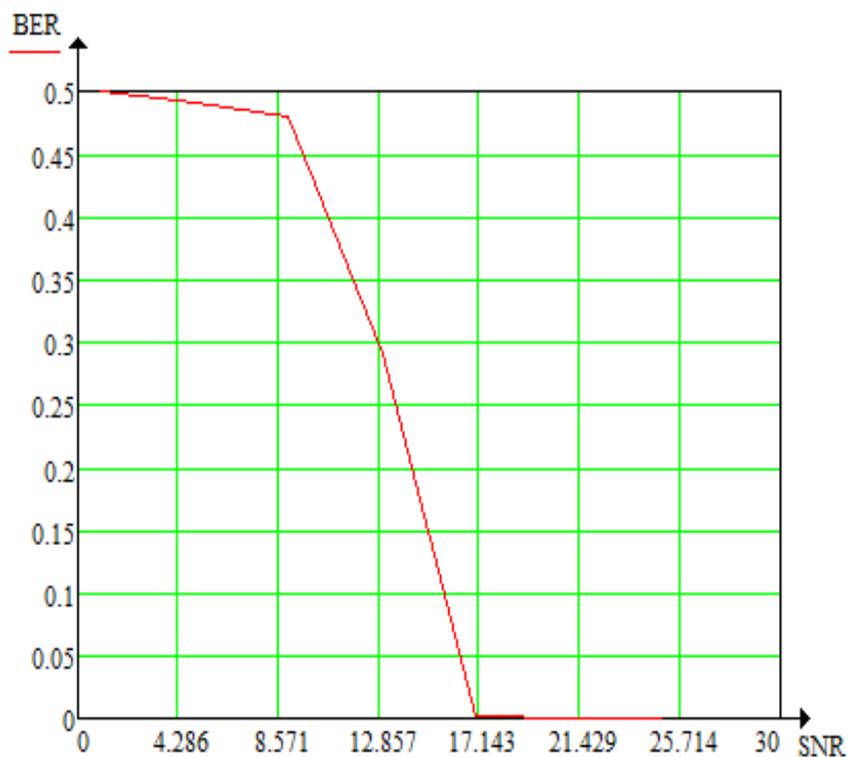


Рисунок 5.153. Зависимость BER от SNR для системы DVB-T при использовании 64-QAM

При исследовании зависимости битовой вероятности битовой ошибки от отношения сигнал/шум рассматриваемой системы телевизионного вещания были сняты изображения спектра передаваемого сигнала и диаграммы созвездий 64-QAM исследуемой системы при SNR равном 1 дБ, 13 дБ и 25 дБ.

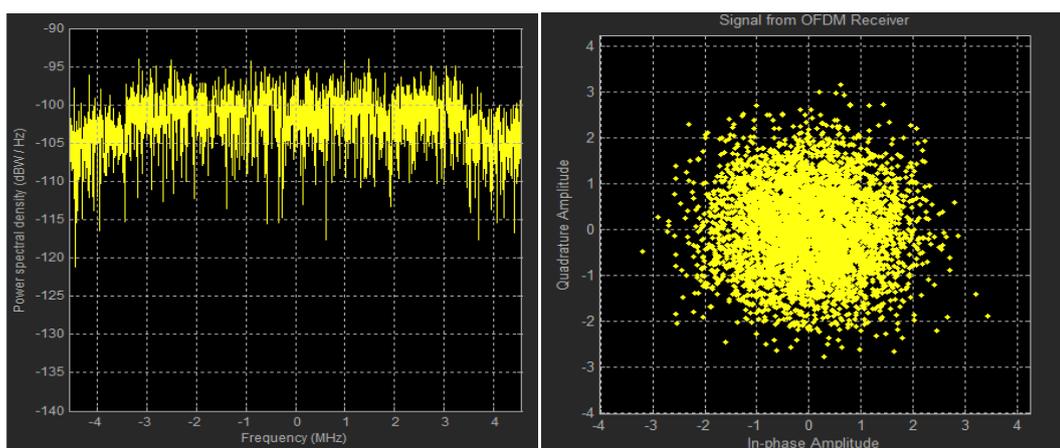


Рис. 5.154. Спектр OFDM-сигнала и диаграмма созвездий 64-QAM при SNR=1 дБ

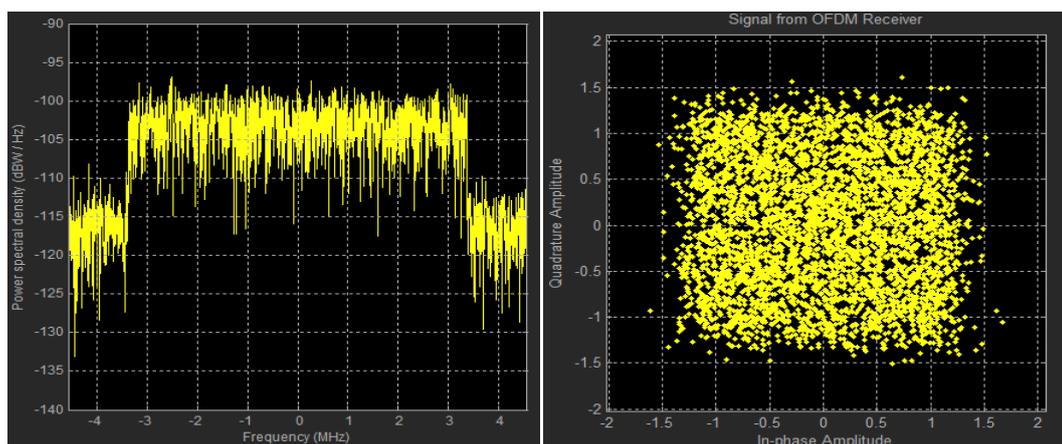


Рис. 5.155. Спектр OFDM-сигнала и диаграмма созвездий 64-QAM при SNR=13 дБ

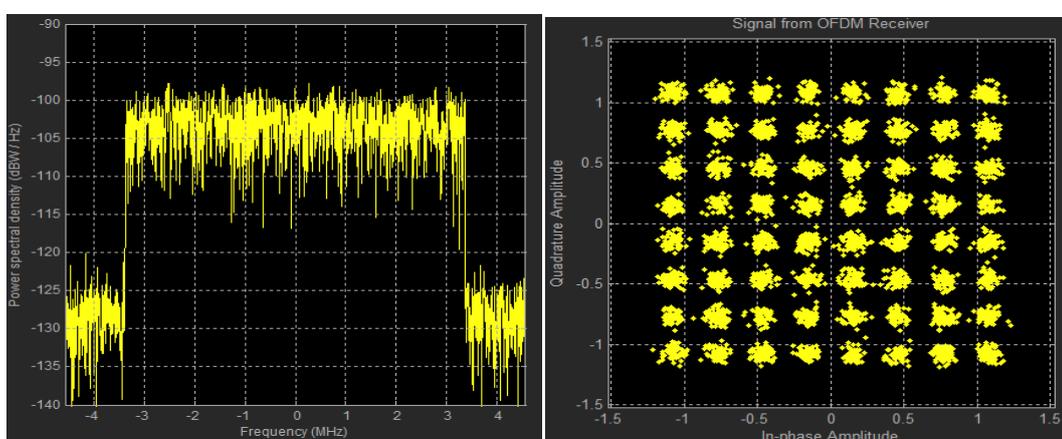


Рис. 5.156. Спектр OFDM-сигнала и диаграмма созвездий 64-QAM при SNR=25 дБ

В процессе выполнения данной работы были изучены основные теоретические аспекты системы цифрового наземного телевизионного стандарта DVB-T.

При выполнении практической части работы была построена зависимость битовой вероятности ошибки от отношения сигнал/шум, результат представлен в виде графика (рисунок 5.155). Полученная в ходе моделирования зависимость соответствует теоретическим данным.

Вместе с этим, были сняты изображения спектра OFDM-символа и диаграммы созвездий 64-QAM при прохождении сигнала в канале с аддитивным белым гауссовским шумом (АБГШ).

Полученные в результате моделирования данные позволяют сделать вывод о том, что безошибочная передача данных по каналу связи в системе DVB-T возможна при отношении сигнал/шум не менее 17 дБ.

### 5.9. Имитационное моделирование системы цифрового спутникового телевизионного вещания DVB-S и системы высокоскоростного цифрового спутникового ТВ-вещания DVB-S2 [16, 21]

Методы модуляции и канального кодирования DVB-S используются для первичного и вторичного распределения спутникового цифрового многопрограммного ТВ/ТВЧ в полосах системы стационарной спутниковой связи (FSS — Fixed Satellite Service) и системы спутникового вещания (BSS — Broadcast Satellite Service). Система предназначена для обеспечения сервиса «непосредственно-на-дом» (Direct To Home — DTH) с использованием потребительского интегрированного приемника-декодера (IRD — Integrated Receiver Decoder), а также для систем коллективного приема (SMATV — Satellite Master Antenna Television) и головных станций кабельного телевидения с возможностью повторной модуляции.

Таблица 5. 19. Максимальная скорость битового потока при ширине полосы телевизионного канала 8 МГц

Тип модуляции	Скорость кодирования	Рекомендуемая максимальная скорость, Мбит/с	Длина T2-кадра, OFDM-символов	Число кодовых слов в кадре
QPSK	1/2	7,4442731	60	50
	3/5	8,9457325		
	2/3	9,9541201		
	3/4	11,197922		
	4/5	12,948651		
	5/6	12,456553		
16-QAM	1/2	15,037432	60	101
	3/5	18,07038		
	2/3	20,107323		
	3/4	22,619802		
	4/5	24,136276		
	5/6	25,162236		
64-QAM	1/2	22,481705	60	151
	3/5	27,016112		
	2/3	30,061443		
	3/4	33,817724		
	4/5	36,084927		
	5/6	37,618789		
256-QAM	1/2	30,074863	60	202
	3/5	36,140759		
	2/3	40,214645		
	3/4	45,239604		
	4/5	48,272552		
	5/6	50,524472		

В системе применена модуляция QPSK и защита от ошибок на основе сверточного кода и сокращенного кода Рида-Соломона. Система может быть использована в спутниковых ретрансляторах с различной шириной полосы.

На рис. 5.189 приведена функциональная структурная схема передающей части системы DVB-S. Система непосредственно совместима с телевизионными сигналами, закодированными по стандарту MPEG-2.

Техника помехоустойчивого кодирования, принятая в системе, разработана в целях достижения «квазибезошибочного» (QEF — Quasi-Error-Free) режима работы, при котором возможно возникновение менее одного случая неисправимой ошибки на час передачи, что соответствует уровню ошибки (BER — Bit Error Ratio)  $10^{-10}$ - $10^{-11}$  на входе демультимплексора MPEG-2.

При адаптации сигнала к спутниковому каналу связи осуществляются следующие операции:

- адаптация транспортного мультимплексирования с использованием статистического кодирования, аналогичного используемому в системе DVB-T ;

- внешнее кодирование с использованием кода Рида-Соломона RS(204,188,t = 8);

- сверточное перемежение;

- внутреннее кодирование с использованием сверточного кода с выкалыванием;

- система предусматривает сверточное кодирование со скоростями кода 1/2, 2/3, 3/4, 5/6 и 7/8;

- формирование сигнала в основной полосе частот;

- модуляция QPSK.

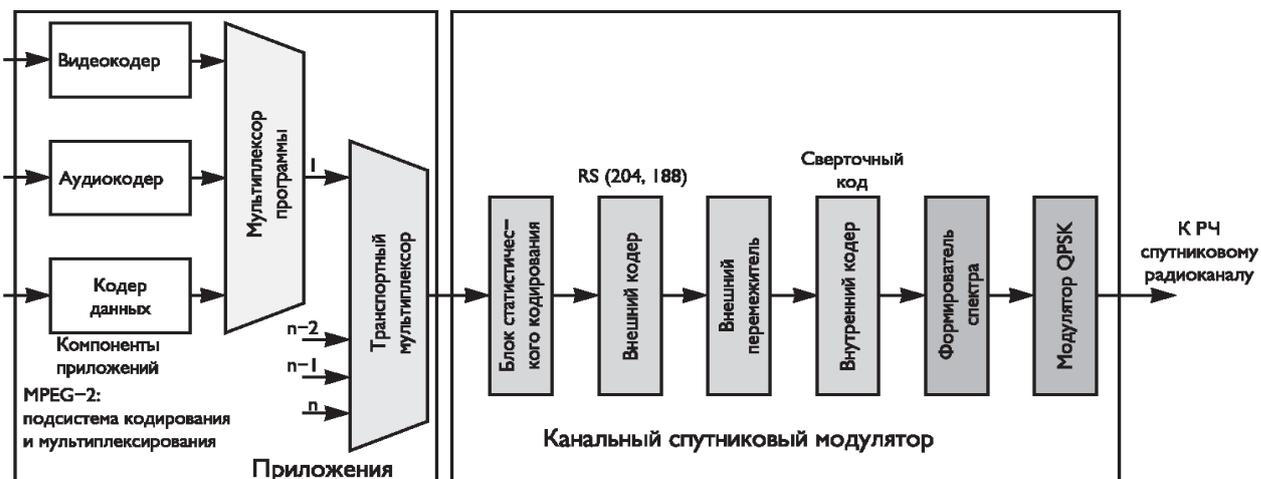


Рис. 5.187. Структурная схема передающей части системы DVB-S

DVB-S, принятый еще 1994 году, определяет структуру транспортных пакетов, канальное кодирование и схемы модуляции при передаче по спутниковым каналам сетей непосредственного вещания (DTH). Стандарт DVB-DSNG, появившийся на три года позже, выполняет те же задачи для профессиональных сетей, то есть для сетей передачи сигнала на пункты ретрансляции и спутниковых сетей сбора новостей. Второй стандарт отличается от первого, в основном тем, что рассчитан на более слабые передатчики, не вводящие спутниковый ретранслятор в режим насыщения и поэтому допускающие использование более высоких уровней модуляции – 8PSK и 16QAM.

### Система высокоскоростного цифрового спутникового ТВ-вещания DVB-S2

DVB-S2 призван покрыть обе эти области, а также должен решить ряд задач, с которыми имеющиеся стандарты справляются плохо.

#### Схемы модуляции и способы помехозащитного кодирования

Новый стандарт предусматривает четыре возможных схемы модуляции (рис 5.189).

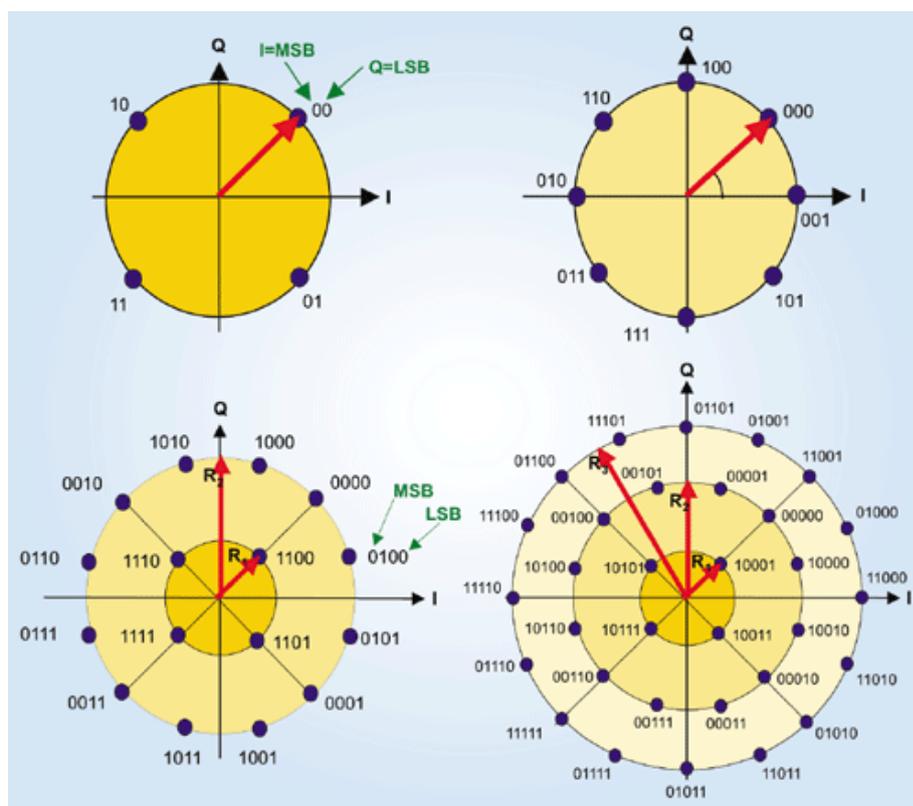


Рис. 5.188. Четыре схемы модуляции, применяемых в стандарте DVB-S2:

QPSK, 8PSK, 16APSK, 32APSK

Первые две, QPSK и 8PSK, предназначены для использования в вещательных сетях. Передатчики транспондеров работают там в режиме, близкому к насыщению, что не позволяет модулировать несущую по амплитуде. Более скоростные схемы модуляции, 16 APSK и 32 APSK, ориентированы на профессиональные сети, где часто используются более слабые наземные передатчики, не вводящие бортовые ретрансляторы в нелинейный режим работы, а на приемной стороне устанавливаются профессиональные конвертеры (LNB), позволяющие с высокой точностью оценить фазу принимаемого сигнала. Эти схемы модуляции можно использовать и в системах вещания, но этом случае каналообразующее оборудование должно поддерживать сложные варианты предвыскажений, а на приемной стороне должен быть обеспечен более высокий уровень отношения сигнал/шум.

### Практическая часть индивидуального задания [21]

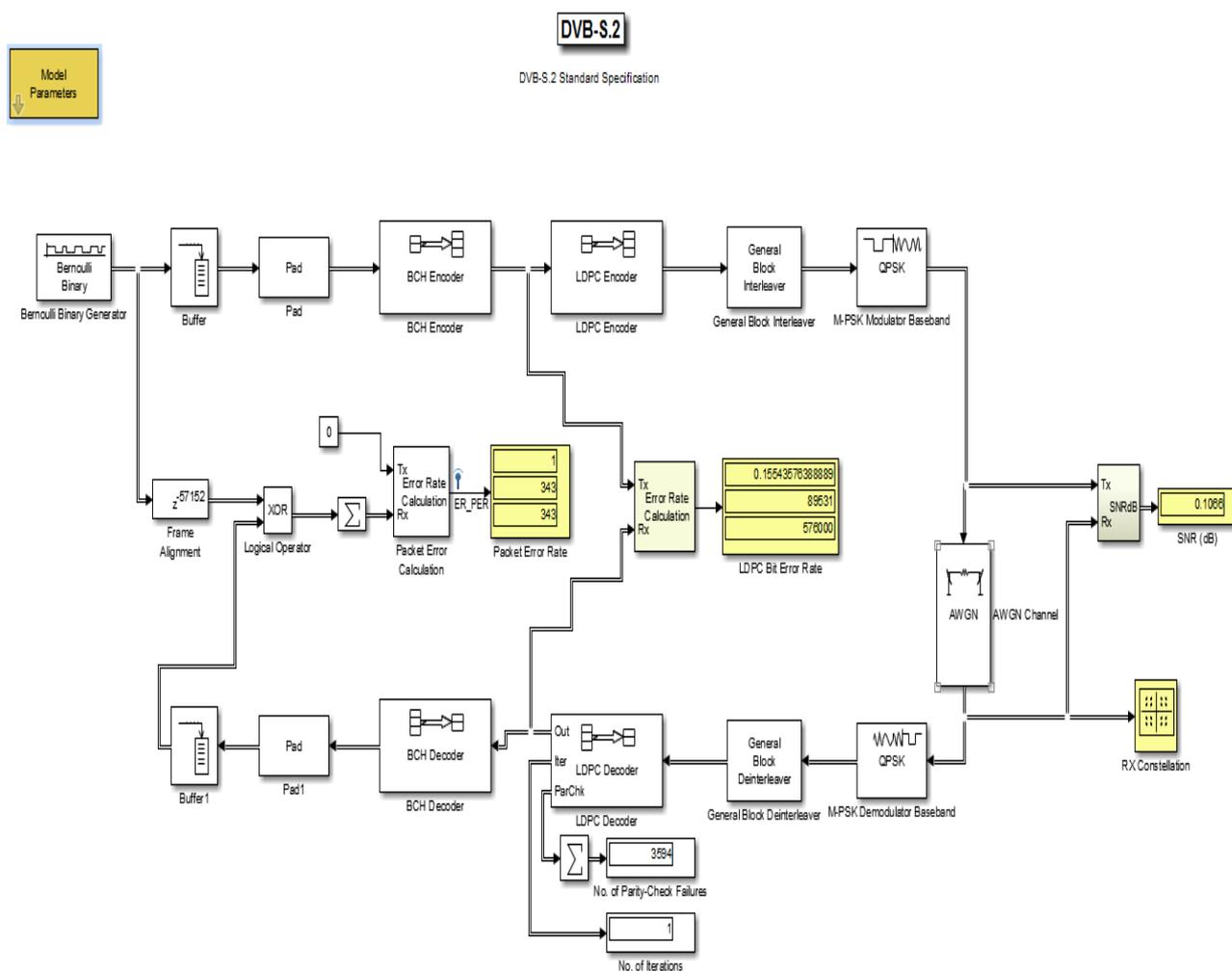


Рис. 5.189. Структура DVB-S.2 в Simulink MATLAB 2015b

Структура модема и кодека DVB-S2 состоит из следующих модулей:

1. **Bernoulli sequence generator** - Первый блок отвечает за генерацию сбалансированных, с точки зрения вероятности инцидентов, случайной двоичной последовательности. Последовательность Бернулли распределена нулей и единиц вероятностями  $p$  и  $(1-p)$  соответственно. В этой модели,  $p = 0,5$  в результате равной вероятностью происходит 0 и 1. Выходной сигнал этого кадра имеет тот же размер, как пакет MPEG-TS, который содержит 188 байт по 8 бит, то есть 1504 бит.

2. **BBFRAME buffering/unbuffering**. С выхода генератора пакеты буферизуются, создавая базовый диапазон кадра (BBFRAME). Размер этого кадра зависит от скорости кодирования, чтобы BCH был равен размерам входного сигнала, на входе кодера. Информационные биты (DFL) могут быть рассчитаны по формуле:

$$\text{DataField} = K_{\text{BCH}} - 80$$

Где  $K_{\text{BCH}}$  является размер внешнего кода FEC кодер BCH, и размер заголовка равен 80 BBFrame. Структура BBFRAME показана на рисунке 4.

3. BCH encoder/decoder - Одним из DVB-S2 достижений является прямое исправление ошибок, которые развернуты, чтобы уменьшить BER в передаче используется исправление ошибок BCH. Выход BBFrame буферизации блока на стороне отправителя, являются кадры бит, где BCH исправление ошибок с исправлением власти  $t$  будет применяться к ним. Для каждого из 11 скорости кодирования представлены в стандартных значений  $K_{\text{bch}}$  и  $n_{\text{МПБ}}$  определяются в том числе  $T$ -коррекции ошибок параметра. В таблицах 1 и 2 эти значения приведены для нормальных и коротких кадров, соответственно.

4. LDPC encoder/coder – Кодирование с проверкой четности. Отношение в скорости показывает, на сколько бит информации приходится бит с проверкой четности. Например  $1/4$  имеет высокую степень проверки четности, и малую скорость, а  $9/10$  высокую скорость, но слабую проверку на четность. На стороне приемника, LDPC-декодер проверяет принятую последовательность до проверки четности

## Ход работы

### Вид модуляции Qpsk 1/4

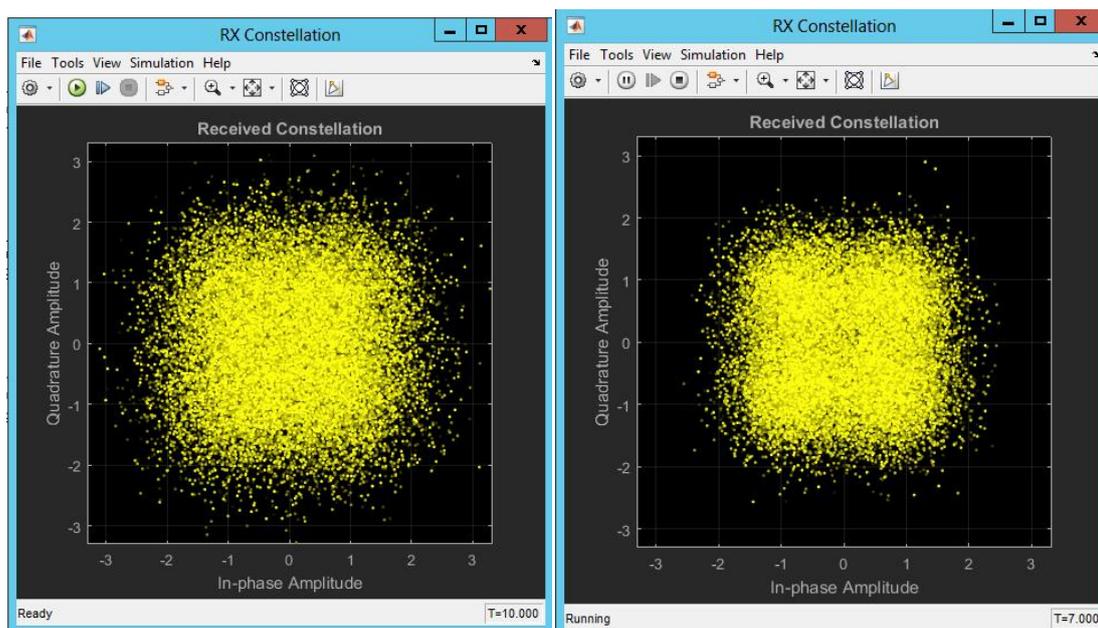


Рис. 5.190. Созвездие при  $E_b/N_0 = 0.5$  и  $E_b/N_0 = 3.5$

### Вид модуляции Qpsk 3/4

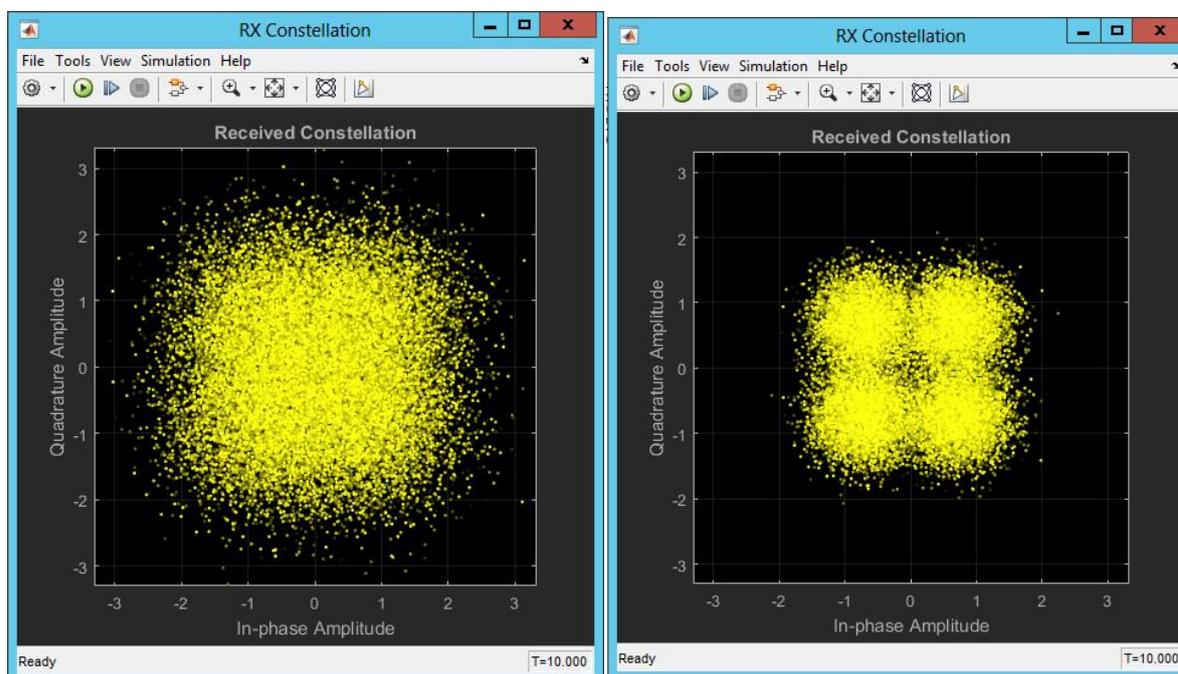
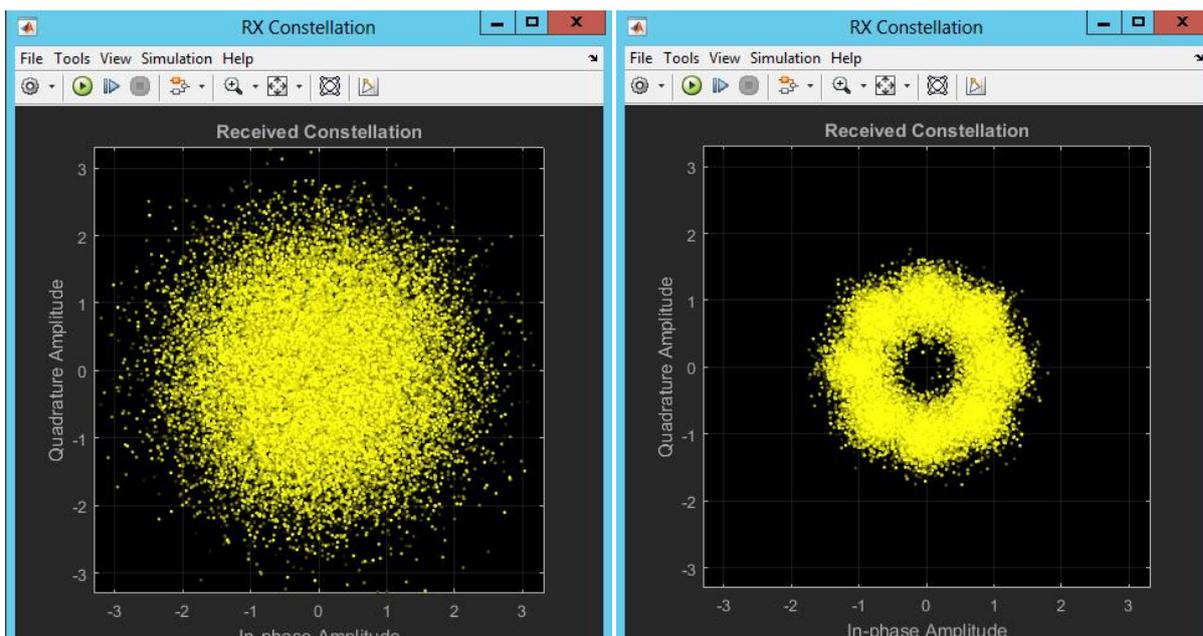


Рис. 5.191. Созвездие при  $E_b/N_0 = 0,5$  и  $E_b/N_0 = 5$

## Вид модуляции - 8psk

Рис. 5.192. Созвездие  $E_b/N_0 = 0,5$  и  $E_b/N_0 = 12$ 

## Вид модуляции - 8psk9/10

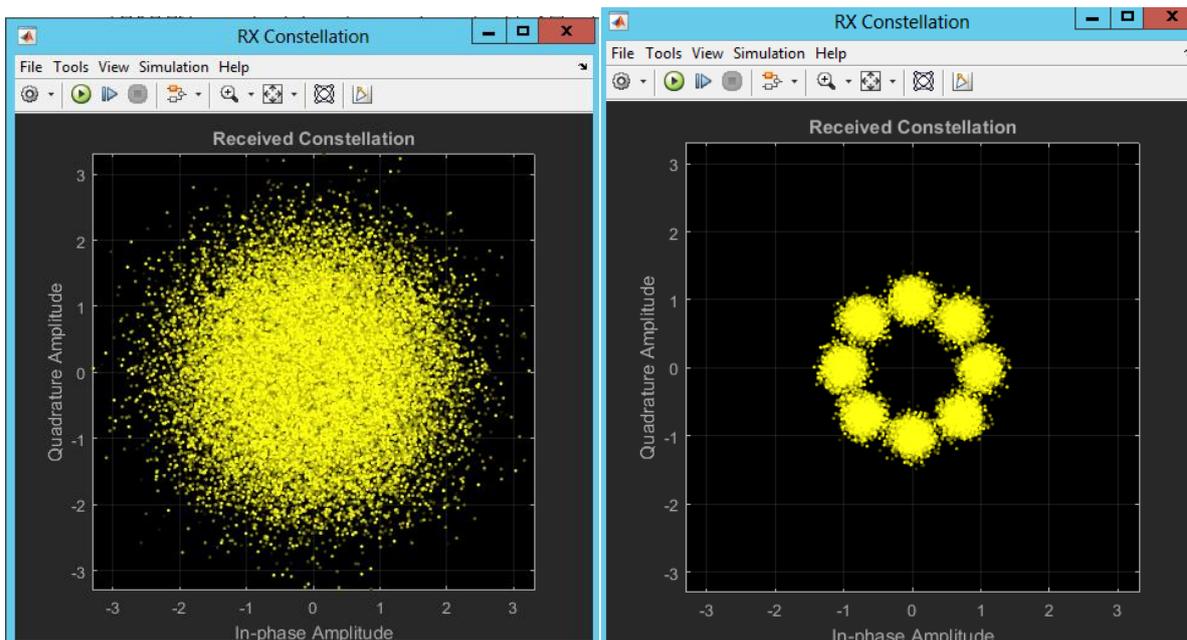
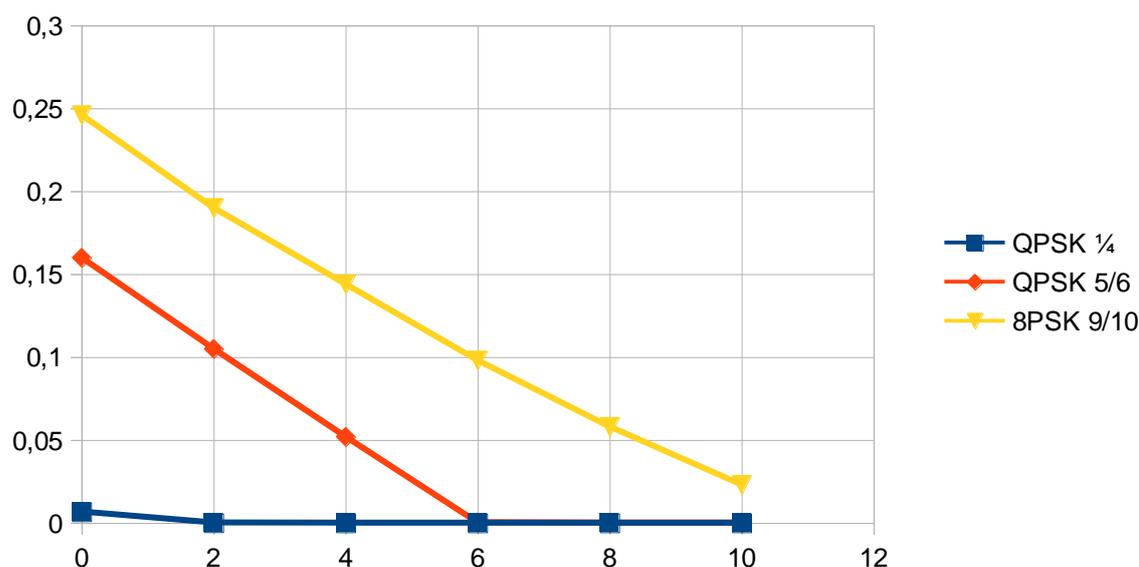
Рис. 5.193. Созвездие при  $E_b/N_0 = 0.5$  и  $E_b/N_0 = 6$

График зависимости  $E_b/N_0$  от BERРис. 5.194. График зависимости BER от  $E_b/N_0$ 

В ходе работы был изучен алгоритм стандарта DVB-S2, создан его рабочий макет, позволяющий увидеть получаемые созвездия, увидеть разность появляющихся ошибок при передаче на разных скоростях. Стандарт DVB-S2 являлся промежуточным звеном между DVB-S и DVB-C2, и не был реализован в полной мере, по сравнению с форматом DVB-S.

#### 5.10. Имитационное моделирование системы цифрового кабельного телевизионного вещания DVB-C и системы высокоскоростного цифрового кабельного ТВ-вещания DVB-C2 [16, 21]

DVB-C – стандарт цифрового телевизионного вещания, который производится по кабелю. В основе стандартов DVB-C лежит стандарт кодирования движущихся изображений и звукового сопровождения MPEG-2.

##### Система цифрового кабельного телевизионного вещания DVB-C.

Система цифрового кабельного телевидения определяется как функциональный блок оборудования, выполняющий адаптацию ТВ-сигналов к характеристикам кабельного канала. Система DVB-C максимально гармонизирована со спутниковой системой DVB-S и может использовать источник местных ТВ-программ.

В связи с высокой помехозащищенностью кабельных каналов связи в системе

DVB-C не используется сверточное кодирование, но применяется многопозиционная QAM-модуляция — от 16-QAM до 256-QAM.

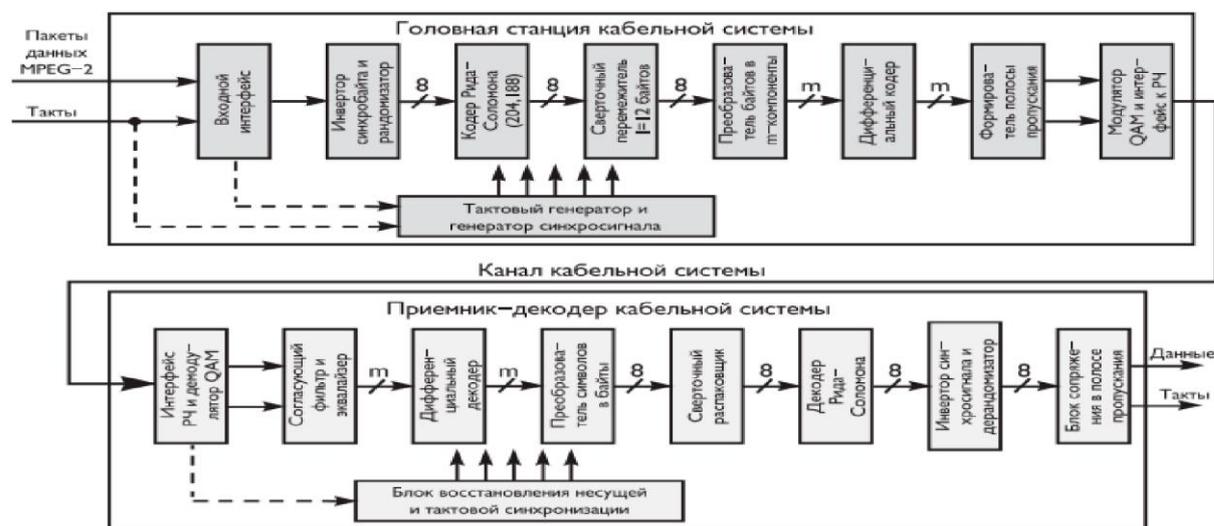


Рис. 5.195. Структурная схема системы цифрового кабельного телевидения DVB-C

В качестве входных сигналов на головной станции используются транспортные пакеты MPEG-2, поступающие через входной интерфейс на модуль, и рандомизирующий поток поступающих данных. Схема рандомизатора/дерандомизатора аналогична используемой в стандарте DVB-T.

Вслед за процессом рандомизации распределения энергии выполняется систематическое сокращенное кодирование Рида-Соломона каждого рандомизированного транспортного пакета MPEG-2, при  $t = 8$ , что обеспечивает возможность корректировки 8 ошибочных байтов в каждом транспортном пакете. Данный процесс добавляет 16 байтов четности к транспортному пакету MPEG-2 для получения кодового слова (204, 188).

Затем следует сверточный перемежитель состоящий из  $l = 12$  звеньев, циклически присоединенных к входному потоку байтов с помощью коммутатора. Каждое звено представляет собой регистр FIFO размером  $M \cdot j$  ячеек ( $M = 17 = N/l$ ,  $N = 204$  — длина защищенного от ошибок кадра,  $l = 12$  — глубина перемежения,  $j$  — индекс звена). Ячейки FIFO содержат 1 байт, а работа коммутаторов входа и выхода синхронизирована. Притом неинвертированные и инвертированные синхронизирующие байты должны быть всегда адресованы в нулевое звено компоновщика, соответствующее нулевой задержке.

После сверточного перемежения производится точное перекодирование байтов в символы. Перекодирование должно быть основано на использовании границ байтов в системе модуляции. Длина символов  $m = \log_2 M$ , где  $M$  — число позиций QAM-созвездия. Циклическая задача отображения для одного цикла определяется соотношением:  $8k = n \cdot m$ , (1) где  $k$  и  $n$  — числа преобразуемых байтов и последовательности двоичных символов, соответственно (см. табл. 5.20.).

Таблица 5.20. Коэффициенты преобразования байтов в последовательности символов.

Модуляция	$m$	$n$	$k$	$8k$
16-QAM	4	2	1	8
32-QAM	5	8	5	40
64-QAM	6	4	3	24
128-QAM	7	8	7	56
256-QAM	8	1	1	8

Для устранения потерь из-за скачков фазы несущей применяется дифференциальное кодирование двух старших битов ( $A_k$  и  $B_k$ ) последовательности символов. Эти два старших бит каждого символа должны быть дифференциально закодированы для получения инвариантного относительно фазового сдвига на  $\pi/2$  созвездия QAM. Дифференциальное кодирование двух старших битов (MSB) должно осуществляться согласно следующему Булеву выражению:

$$I_k = \overline{(A_k \oplus B_k)} \cdot (A_k \oplus I_{k-1}) \vee (A_k \oplus B_k) \cdot (A_k \oplus Q_{k-1});$$

$$Q_k = \overline{(A_k \oplus B_k)} \cdot (A_k \oplus Q_{k-1}) \vee (A_k \oplus B_k) \cdot (A_k \oplus I_{k-1}).$$

На рис. 5.197. приведен пример реализации преобразования байтов в символы.



Рис. 5.196. Пример реализации преобразования байта в  $m$ -мерный вектор и дифференциального кодирования двух старших битов

Квадратурные сигналы  $I$  и  $Q$  с выхода схемы отображения перед модуляцией подвергаются фильтрации с помощью фильтра, характеристика которого соответствует соотношению при коэффициенте  $\alpha = 0,15$ .

В табл. 2 приведены примеры расчетных значений символьной и информационной скоростей при разных кратностях модуляции в канале с полосой 8 МГц. Максимальная скорость достигает 38,1 Мбит/с, что соответствует пропускной

способности ствола спутникового ретранслятора с полосой 33 МГц в типовом режиме  $F_{\text{симв}} = 27,5$  Мсим/с,  $R = 3/4$ .

Таблица 5.21. Примеры расчетных значений символьной и информационной скоростей при использовании стандарта DVB-C.

Полезная информационная скорость (транспортный уровень MPEG-2), Мбит/с	Общая скорость, включая RS (204,188), Мбит/с	Кабельная символьная скорость, Мбод/с	Занимаемая полоса частот, МГц	Вид модуляции
38,1	41,34	6,89	7,92	64-QAM
31,9	34,61	6,92	7,96	32-QAM
25,3	27,34	6,84	7,86	16-QAM
18,9	20,52	3,42	3,93	64-QAM
16,0	17,40	3,48	4,00	32-QAM
12,8	13,92	3,48	4,00	16-QAM
9,6	10,44	1,74	2,00	64-QAM
8,0	8,70	1,74	2,00	32-QAM
6,4	6,96	1,74	2,00	16-QAM

### Система высокоскоростного цифрового кабельного телевизионного вещания DVB-C2.

Стандарт кабельного цифрового телевизионного вещания DVB-C2 максимально унифицирован с новыми стандартами, обслуживающими спутниковую (DVB-S2) и эфирную (DVB-T2) транспортные среды.

На рис. 3 а–г приводится достаточно подробная структурная схема передающей части DVB-C2. Как в DVB-S2 и DVB-T2, в новом кабельном стандарте внутри одного физического канала предусмотрено выделение транспортных PLP физических каналов, которые могут обрабатывать и переносить обычный поток MPEG-2 TS или использоваться для передачи IP с применением GSE-протокола.

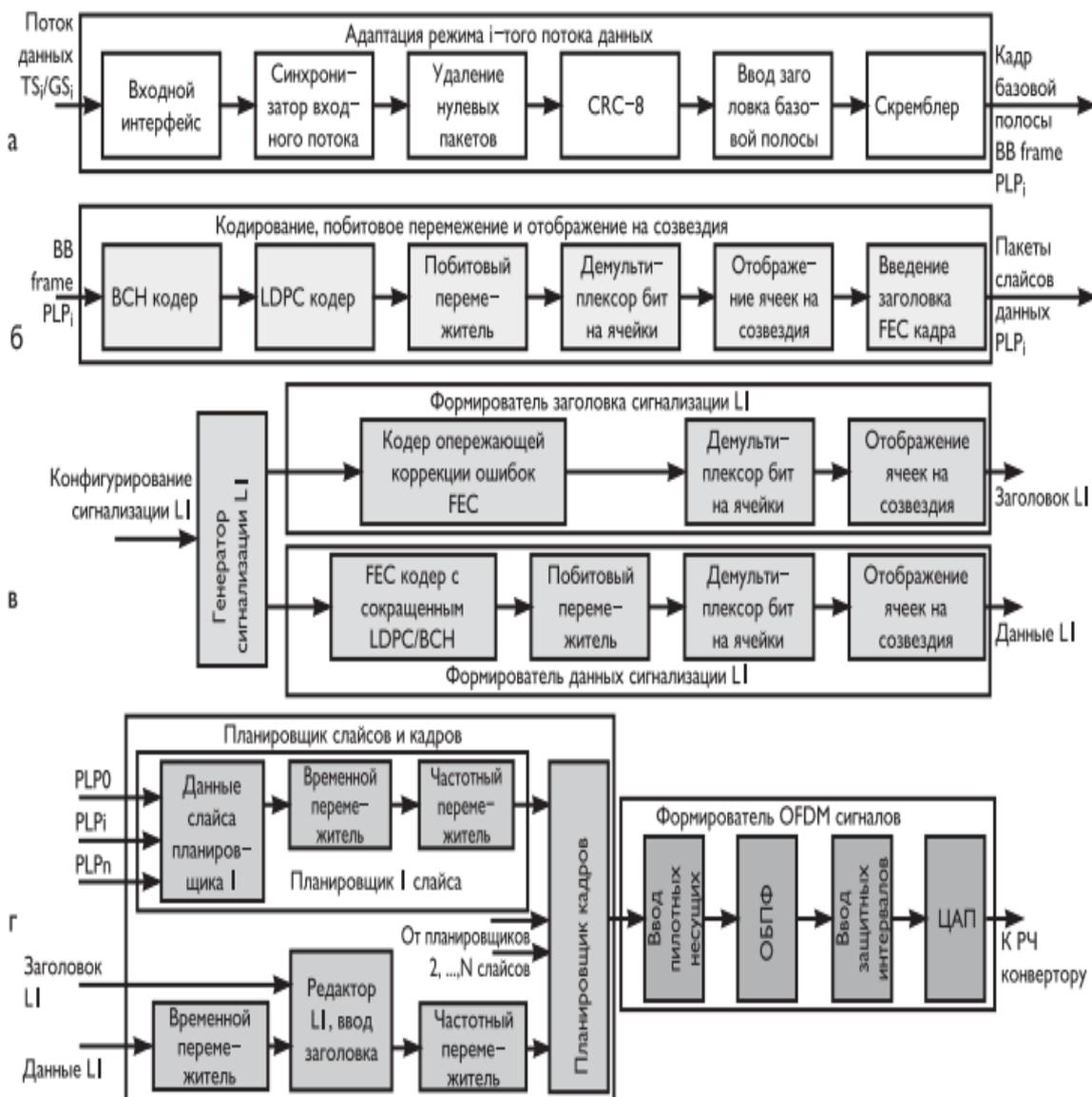


Рис. 5.197. Структурная схема передающей части системы DVB-C2

Вход системы DVB-C2 состоит из одного или из множества логических потоков данных. По одной магистрали физического уровня (PLP) передается один логический поток данных. Модули адаптации режима, по отдельности обрабатывающие содержимое каждой PLP, разбивают входной поток данных на поля данных, которые после адаптации потока должны сформировать кадры базовой полосы (BBFrame). Модуль адаптации режима включает в себя входной интерфейс, за которым следуют три опциональные подсистемы (синхронизатор входного потока, модуль удаления нулевых пакетов и кодер CRC-8), после которых он разбивает входной поток данных на поля данных и выполняет вставку заголовка базовой полосы (BBHeader) в начале каждого поля данных. Подсистема входного интерфейса обеспечивает преобразование входного сигнала во внутренний логически-битовый формат для каждой единичной магистрали физического уровня (PLP, рис.3а). Длина поля данных DFL на выходе интерфейса должна находиться в

пределах:  $0 < DFL < (Kbch - 80)$ , где  $Kbch$  — количество битов в поле данных, защищенном кодами BCH и LDPC; 10-байтовый (80 битов) заголовок BBHeader присоединяется к началу поля данных и также защищен кодами LDPC и BCH.

Синхронизатор входного потока формирует поле, состоящее из 2 или 3 байтов (поле ISSY - Input Stream Synchronisation), в котором передается значение счетчика, тактируемого с тактовой частотой модулятора ( $1/T$ , где  $T = 7/64$  мкс или  $T = 7/48$  мкс для каналов с полосой пропускания 8 МГц или 6 МГц, соответственно), используемого приемником для восстановления точной синхронизации восстановленного выходного потока. Содержание поля ISSY зависит от формата входного потока и режимов обычной или повышенной эффективности, указанных в заголовке базовой полосы.

Установленные для передачи транспортных потоков требования предусматривают, чтобы скорости битовых потоков на выходе мультиплексора передатчиков и на входе демупльтиплексора приемников были постоянными на протяжении длительных периодов времени и сквозная задержка также была постоянной. Во входных транспортных потоках может присутствовать большая доля нулевых пакетов для адаптации сервисов с переменной скоростью битового потока в транспортных потоках с постоянной скоростью. В таком случае, во избежание излишних накладных расходов при передаче, нулевые пакеты TS должны быть удалены. Процесс выполняется таким образом, чтобы удаленные нулевые пакеты могли быть повторно вставлены в приемнике в точности на то же самое место, где они находились первоначально.

Кодирование CRC-8, как и в системах DVB-T2 и DVB-S2, применяется для детектирования ошибок на уровне пользовательского пакета, а 10-байтовый заголовок базовой полосы (BBHeader) фиксированного размера вводится перед полем данных для описания формата поля данных.

Перед поступлением на вход системы помехоустойчивого кодирования (см. рис.3б) цифровой поток базовой полосы скремблируется сдвиговым регистром с обратной связью. Порождающий полином последовательности PRBS —  $1 + x^{14} + x^{15}$  — с иницируемой в начале каждого кадра BBFrame загрузкой в регистр кода 100101010000000.

Структура формата кадра с опережающей коррекцией ошибок BCH и LDPC для основного размера 64 800 битов и сокращенного размера 16 200 битов могут быть для LDPC кодов —  $2/3$ ,  $3/4$ ,  $4/5$ ,  $5/6$  и  $9/1$  и для LDPC кодов —  $1/2$ ,  $2/3$ ,  $3/4$ ,  $4/5$ ,  $5/6$  и  $8/9$ , соответственно.

Сигнал с выхода кодера LDPC подвергается побитовому перемежению, которое состоит из перемежения проверочных битов, за ним следует перемежение со сдвигом начала

столбцов в соответствии с правилом системы DVB-C2 (информационные биты не перемежаются). Значения параметра  $Q_{ldpc}$  определены в табл. 5.22.

Таблица 5.22. Значения  $Q_{ldpc}$  для основных и сокращенных кадров.

Скорость кода	$N_{ldpc} = 64\ 800$	$N_{ldpc} = 16\ 200$
1/2	–	25
2/3	60	15
3/4	45	12
4/5	36	10
5/6	30	8
8/9	–	5
9/10	18	–

Конфигурация перемежения со сдвигом начала столбцов для каждого формата модуляции определена в табл. 5.23.

Таблица 5.23. Структура побитового перемежителя

Модуляция	Строки $N_r$		Столбцы $N_c$
	$N_{ldpc} = 64\ 800$	$N_{ldpc} = 16\ 200$	
16-QAM	8100	2025	8
64-QAM	5400	1350	12
256-QAM	4050	–	16
	–	2025	8
1024-QAM	3240	810	20
4096-QAM	5400	–	12
	–	675	24

При перемежении со сдвигом начала столбцов биты данных  $u_i$  с перемежителя проверочных битов последовательно записываются в перемежитель со сдвигом начала столбцов по столбцам, и последовательно считываются по строкам (старший бит заголовка считывается первым). Запись стартовой позиции каждого столбца сдвигается на величину  $t_c$  в соответствии с табл. 5.24.

Таблица 5.24

Модуляция	16-QAM		64-QAM		256-QAM		1024-QAM		4096-QAM	
$N_{ldpc}$	64 800	16 200	64 800	16 200	64 800	16 200	64 800	16 200	64 800	16 200
Столбцы $N_c$	8		12		16	8	20		12	24
Сдвиг $t_c$	0	0	0	0	0	0	0	0	0	0
	1	0	0	0	0	2	0	1	0	0
	2	2	0	2	0	2	0	3	0	2
	3	4	1	2	2	2	1	4	2	2
	4	4	7	3	2	2	7	5	2	3
	5	5	20	4	2	3	20	6	2	4
	6	7	20	4	3	7	20	6	2	4
	7	7	21	5	3	15	21	9	2	5
	8	–	–	5	3	16	–	13	5	5
	9	–	–	7	6	20	–	14	5	7
	10	–	–	8	7	22	–	14	5	8
	11	–	–	9	7	22	–	16	5	9
	12	–	–	–	–	27	–	21	5	–
	13	–	–	–	–	27	–	21	7	–
	14	–	–	–	–	28	–	23	7	–
	15	–	–	–	–	32	–	25	7	–
	16	–	–	–	–	–	–	25	7	–
	17							26	8	–
	18							28	8	–
	19							30	10	–
	20									10
	21									10
	22									10
	23									11

Каждый кадр FECFRAME преобразовывается в кодированный и модулированный FEC блок с опережающей коррекцией ошибок. Для этого входные биты сначала демультиплексируются на параллельные модулирующие значения ячеек, и затем эти модулирующие значения отображаются на значения созвездия. Количество ячеек выходных данных и эффективное количество битов на ячейку  $\eta_{mod}$  заданы в табл. 5.25.

Таблица 5.25. Параметры побитового отображения на созвездия.

Длина блока LDPC-кода ( $N_{ldpc}$ )	Режим модуляции	$\eta_{mod}$	Число выходных ячеек данных
64 800	4096-QAM	12	5400
	1024-QAM	10	6480
	256-QAM	8	8100
	64-QAM	6	10 800
	16-QAM	4	16 200
16 200	4096-QAM	12	1350
	1024-QAM	10	1620
	256-QAM	8	2025
	64-QAM	6	2700
	16-QAM	4	4050
	QPSK	2	8100

Битовый поток  $v_{di}$ , от побитового перемежителя демультиплексируется на  $N_{\text{substreams}}$  подпотоков (табл. 5.26).

Таблица 5.26. Количество подпотоков в демультиплексоре.

Модуляция	$N_{\text{dps}}$	Количество подпотоков $N_{\text{substreams}}$
QPSK	Любое	2
16-QAM	Любое	8
64-QAM	Любое	12
256-QAM	64 800	16
	16 200	8
1024-QAM	Любое	20
4096-QAM	64 800	12
	16 200	24

Демультиплексирование определяется как отображение подвергнутых побитовому перемежению входных битов,  $v_{di}$ , на выходные биты  $b_{e,do}$  на выходе демультиплексора  $d_i \bmod N_{\text{substreams}}$  — число входных битов;  $do = d_i \div N_{\text{substreams}}$  — число битов в заданном потоке на выходе демультиплексора;  $e$  — количество демультиплексированных битовых потоков, ( $0 \leq e < N_{\text{substreams}}$ ), зависящее от  $d_i$ .

Каждое модулирующее значение ячеек, поступающее от демультиплексора, модулируется с использованием созвездий одного из типов: QPSK, 16-QAM, 64-QAM, 256-QAM, 1024-QAM или 4096-QAM. Созвездия и детализация применяемого к ним отображения реализуются в соответствии с кодом Грея.

Точки созвездий  $z_{qj}$  для каждого входного модулирующего значения ячеек  $[y_{0,q} \dots y_{\eta} \bmod -1,q]$  нормализуются в соответствии с табл. 5.27.

Таблица 5.28. Параметры демультиплексирования битов на подпотоки для всех кодовых скоростей (за исключением 2/3)

QPSK																									
Вход	0 1																								
Выход	0 1																								
16-QAM																									
Вход	0	1	2	3	4	5	6	7																	
Выход	7	1	4	2	5	3	6	0																	
64-QAM																									
Вход	0	1	2	3	4	5	6	7	8	9	10	11													
Выход	11	7	3	10	6	2	9	5	1	8	4	0													
256-QAM ( $N_{\text{ldpc}} = 64\ 800$ )																									
Вход	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15									
Выход	7	2	9	0	4	6	13	3	14	10	15	5	8	12	11	1									
256-QAM ( $N_{\text{ldpc}} = 16\ 200$ )																									
Вход	0	1	2	3	4	5	6	7																	
Выход	7	3	1	5	2	6	4	0																	
1024-QAM ( $N_{\text{ldpc}} = 64\ 800$ )																									
Вход	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19					
Выход	8	16	7	19	4	15	3	12	0	11	10	9	13	2	14	5	17	6	18	1					
1024-QAM ( $N_{\text{ldpc}} = 16\ 200$ )																									
Вход	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19					
Выход	8	3	7	10	19	4	9	5	17	6	14	11	2	18	16	15	0	1	13	12					
4096-QAM ( $N_{\text{ldpc}} = 64\ 800$ )																									
Вход	0	1	2	3	4	5	6	7	8	9	10	11													
Выход	8	0	6	1	4	5	2	3	7	10	11	9													
4096-QAM ( $N_{\text{ldpc}} = 64\ 800$ )																									
Вход	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	
Выход	8	0	6	1	4	5	2	3	7	10	11	9	6	17	13	20	1	3	9	2	7	8	12	0	

Таблица 5.29. Нормирующие множители для ячеек данных.

Модуляция	Нормирующие множители
QPSK	$f_q = z_q/\sqrt{2}$
16-QAM	$f_q = z_q/\sqrt{10}$
64-QAM	$f_q = z_q/\sqrt{42}$
256-QAM	$f_q = z_q/\sqrt{170}$
1024-QAM	$f_q = z_q/\sqrt{682}$
4096-QAM	$f_q = z_q/\sqrt{2730}$

На рис. 5.197в приведена структура датчиков сигнала синхронизации кадров DVB - C2, содержащего два канал — формирователей заголовка и данных сигнализации L1 .

Кодирование заголовка сигнализации L1 осуществляется первоначально с помощью кодера Рида – Маллера (32, 16). При этом каждый бит 32-битового кодового слова Рида–Маллера разбивается на две ветви. Затем данные отображаются на созвездие QPSK для

устойчивого к ошибкам заголовка кадров FECFrame, или на созвездие 16-QAM для заголовка кадров FECFrame с повышенной эффективностью.

Данные сигнализации L1 подвергаются сокращенному LDPC/VCH – кодированию с последующим побитовым перемежением, демультиплексированием и 16-QAM модуляцией. На рис. 5.197г изображена структура планировщика слайсов, формирователя кадров и OFDM-сигналов системы DVB-C2.

Один или множество логических каналов PLP организуются в группу PLP, и одна или множество таких групп PLP образуют слайс данных. Система C2 может состоять из одного или множества слайсов данных. Предполагается, что приемник всегда должен иметь возможность принимать одну PLP-данных и связанную с ней общую PLP при ее наличии.

Для канала с шириной полосы 8 МГц максимальное число OFDM-несущих при передаче каждого слайса должно быть не более  $nKDC_{max} - KDC_{min} = 3408$  при  $f_{max} - f_{min} = 7,61$  МГц и длительности символа  $TU = 448$  мкс.

Данные слайса от каждого планировщика подвергаются временному и частотному перемежению. Временной перемежитель обычно содержит два банка памяти, в первый из которых осуществляется запись, а из второго производится считывание данных слайса, затем производится переключение режимов работы этих банков. Для реализации частотного перемежения производится изменение порядка считывания символов звездной диаграммы. Данные от планировщиков слайсов поступают на планировщик кадров, на который подаются также специальным образом обработанные данные синхронизации L1.

Структура кадра системы DVB-C2 поясняется рис. 5.198а.

Преамбула кадра включает LP символов (LP1), за ней следует LData символов данных. Преамбула несет информацию о символах блока синхронизации L1 (3408 поднесущих частот в полосе 7,71 МГц). Данные слайсов могут передаваться в произвольной полосе частот, не превышающей полосу частот передачи символов блока L1. Неиспользуемые частоты могут занимать часть всего кадра DVB-C2.

Данные сигнализации L1 циклически повторяются, что обеспечивает возможность восстановить полный L1 блок из частей двух блоков, как показано на рис. 5.199б. На входе формирователя OFDM-сигналов установлена схема ввода пилотных несущих, в состав которых входят пилотные несущие преамбулы, постоянные и рассеянные пилотные несущие в составе передаваемых символов данных, а также граничные пилотные несущие, несущие информацию о границах передачи символов данных.

Номера пилотных несущих преамбулы связаны соотношением:  $k \bmod DP = 0$ , (2) где  $DP = 6$ , т. е. эти пилотные несущие соответствуют значениям  $k = 0, 6, 12, \dots, 3402$ . Параметры пилотных несущих определены следующим образом:

$$\operatorname{Re}\{c_{m,l_P,k}^P\} = A_{PP} \cdot 2(1/2 - r_k), \quad \operatorname{Im}\{c_{m,l_P,k}^P\} = 0,$$

где  $A_{PP} = 1$ ,  $m$  — номер кадра,  $l_P$  — номер символа преамбулы,  $k$  — индекс несущей, а  $r_k$  определено только для значений  $k$ , кратных 6, и вычисляется по формуле:

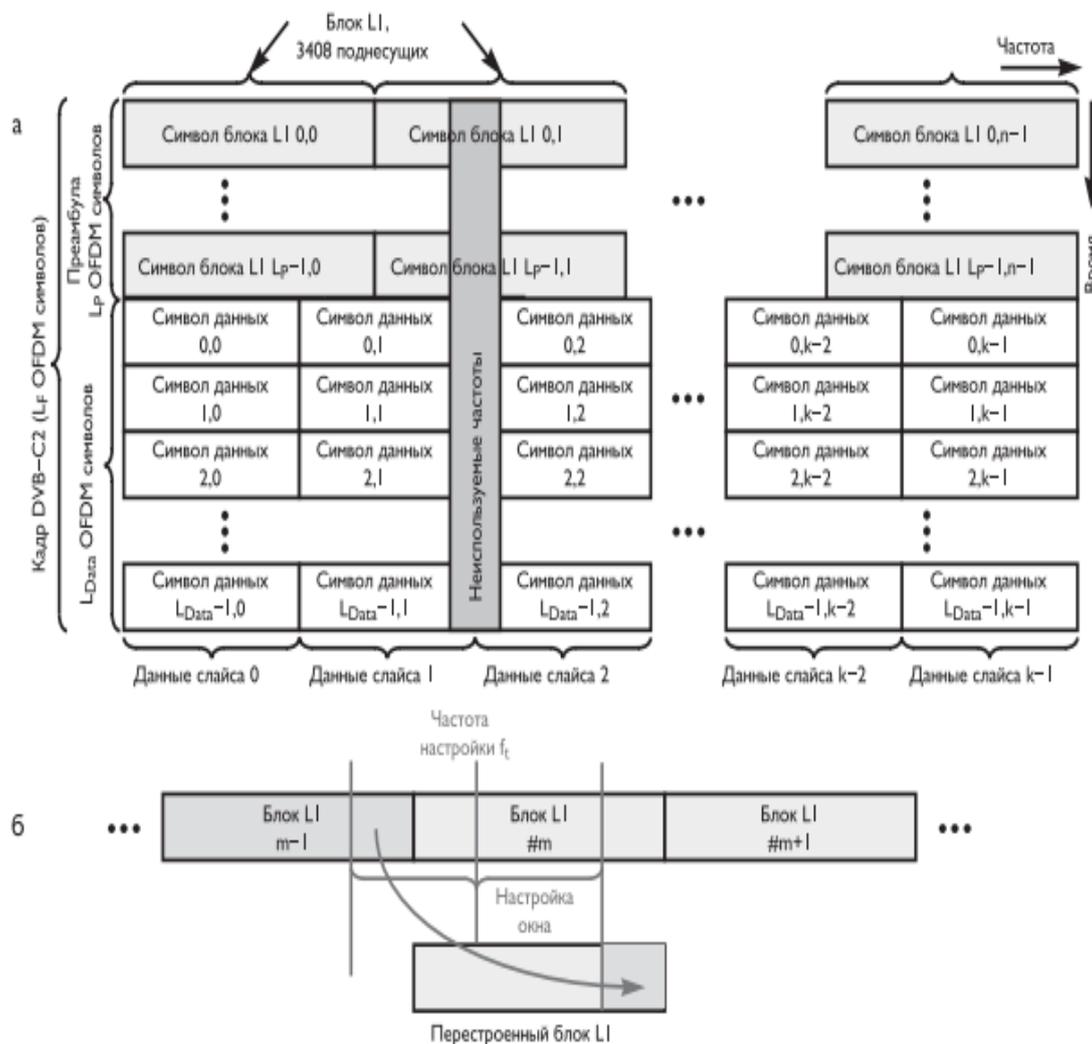


Рис. 5.198. Структура кадра системы DVB-C2.

$$r_k = \begin{cases} w_k^P, & k \bmod K_{L1} = 0; \\ r_{k-6} \oplus w_k^P, & \text{иначе;} \end{cases} \quad w_k^P = w_k \oplus w'_i, i = (k \bmod K_{L1})/D_P;$$

$w_i$  — PRBS регистра сдвига, определяемого соотношением  $x^4 + x^3 + 1$ , иницируемого последовательностью единиц, т. е.  $w_0, w_1, w_2, \dots = 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 0, 0, \dots$ ;  $w_k$  — PRBS регистра сдвига, определяемого соотношением  $x^4 + x^2 + 1$ , иницируемого последовательностью единиц, т. е.  $w_0, w_1, w_2, \dots = 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 0, 0, \dots$ . Номера локальных рассеянных пилотных несущих определяются следующим соотношением:

$$k \bmod (D_X \cdot D_Y) = D_X \cdot (l \bmod D_Y),$$

где  $k$  — индекс несущей,  $l$  — индекс символа данных,  $D_X$ ,  $D_Y$  определены в табл. 5.24.

Таблица 5.30. Коэффициенты, определяющие положение рассеянных пилотных несущих

Коэффициент защитного интервала	Выделение пилотных несущих ( $D_X$ )	Номер символов формирования рассеянных несущих ( $D_Y$ )
1/64	12	4
1/128	24	4

Кроме рассеянных пилотных несущих в каждый символ кадра, за исключением преамбулы, вводятся постоянные несущие. Индексы этих несущих локально в пределах блока из  $KL1=3408$  приведены в табл. 5.33.

Таблица 5.31. Индексы постоянных пилотных несущих.

96	216	306	390	450	486	780	804
924	1026	1224	1422	1554	1620	1680	1902
1956	2016	2142	2220	2310	2424	2466	2736
3048	3126	3156	3228	3294	3366		

В дополнение к рассеянным и постоянным пилотным несущим вводятся также граничные пилотные несущие, определяющие «края» в каждом символе. Такие пилотные несущие вводятся также на границах областей неиспользуемых частот. Некоторые OFDM-ячейки могут быть зарезервированы для подавления пиковых значений формируемого радиосигнала (PAPR). Резервируемые ячейки OFDM-сигнала  $S_0$  определяются соотношением:  $[k \bmod (8 \cdot KL1)] - D_X (l \bmod D_Y) \in S_0, 0 \leq l \leq L_{Data}$ , где  $k$  — абсолютный индекс несущей,  $L_{Data}$  — количество символов данных в кадре DVB-C2. Формируемый на выходе ОБПФ-сигнал может быть записан в следующем виде:

$$s(t) = \operatorname{Re} \left\{ \sum_{m=0}^{\infty} \left[ \frac{1}{\sqrt{K_{\text{total}}}} \sum_{l=0}^{L_F-1} \sum_{k=K_{\text{min}}}^{K_{\text{max}}} c_{m,l,k} \cdot \psi_{m,l,k}(t) \right] \right\},$$

где

$$\psi_{m,l,k}(t) = \begin{cases} \exp \left[ 2\pi j \frac{k}{T_U} (t - T_G - lT_S - mT_F) \right] & \text{при } mT_F + lT_S \leq t \leq \\ & \leq mT_F + (l+1)T_S; \\ 0 & \text{иначе,} \end{cases}$$

$k$  — номер поднесущей;  $l$  — номер символа OFDM от «0» до первого символа преамбулы кадра;  $m$  — номер кадра DVB-C2;  $K_{\text{total}}$  — количество используемых несущих,  $K_{\text{max}}$  —  $K_{\text{min}}$ ;  $L_F$  — количество OFDM символов в кадре (исключая преамбулу);  $T_U$  — длительность активной части символа;  $T_G$  — длительность защитного интервала;  $T_{STU} +$

$T_G$  — полная длительность символа;  $cm,l,k$  — комплексное значение  $k$ -й несущей в  $l$ -м OFDM-символе  $m$ -го кадра DVB-C2;  $T_{FLFTS}$  — длительность кадра;  $K_{min}$  и  $K_{max}$  — индексы первой нижней и последней верхней активных несущих, соответственно.

Поскольку в системе DVB-C2 применены более эффективные методы помехоустойчивого кодирования по сравнению с системой DVB-C, возможно использование звездных диаграмм OFDM-сигналов более высокой размерности. Допустимые комбинации модуляций и параметров LDPC-кодирования приведены в табл. 5.26, в которой также указаны величины отношения сигнал/шум, требуемые для приема, квазисвободного от ошибок.

Таблица 5.32. Параметры OFDM-сигналов для каналов с полосой пропускания 6 МГц и 8 МГц.

Параметр	6 МГц 1/64	6 МГц 1/128	8 МГц 1/64	8 МГц 1/128
Количество OFDM-несущих в блоке сигнализации $L1 - K_{L1}$	3408	3408	3408	3408
Ширина полосы блока сигнализации $L1$	5,61 МГц	5,61 МГц	7,61 МГц	7,61 МГц
Длительность $T_U$ в элементарных периодах $T$	$4096T$	$4096T$	$4096T$	$4096T$
Длительность $T_U$ в мкс	597,3	597,3	448	448
Частотный интервал между несущими $1/T_U$ в Гц	1674	1674	2232	2232
Длительность $T_G$ в элементарных периодах $T$	$64T$	$32T$	$64T$	$32T$
Длительность $T_G$ в мкс	9,33	4,66	7	3,5

Таблица 5.35. Отношение сигнал/шум при различных параметрах системы DVB-C2.

Параметры LDPC кода	16-QAM	64-QAM	256-QAM	1024-QAM	4096-QAM
2/3	–	13,5 дБ	–	–	–
3/4	–	–	20,0 дБ	24,8 дБ	–
4/5	10,7 дБ	16,1 дБ	–	–	–
5/6	–	–	22,0 дБ	27,2 дБ	32,4 дБ
9/10	12,8 дБ	18,5 дБ	24,0 дБ	29,5 дБ	35,0 дБ

Для сравнения эффективностей использования систем кабельного цифрового телевизионного вещания DVB-C2 и DVB-C в табл.14 приведены допустимые скорости передачи информации при эквивалентной ширине канала 8 МГц.

Таблица 5.33. Максимальные скорости передачи информации в системах DVB-C и DVB-C2 при эквивалентной ширине канала 8 МГц.

Система	16-QAM	64-QAM	256-QAM	1024-QAM	4096-QAM	
DVB-C	25 Мбит/с	38,4 Мбит/с	51,2 Мбит/с	–	–	
DVB-C2	2/3	–	31,4 Мбит/с	–	–	
	3/4	–	–	47,1 Мбит/с	58,9 Мбит/с	
	4/5	25,1 Мбит/с	37,7 Мбит/с	–	–	
	5/6	–	–	52,4 Мбит/с	65,4 Мбит/с	78,6 Мбит/с
	9/10	28,3 Мбит/с	41,4 Мбит/с	56,6 Мбит/с	70,7 Мбит/с	84,8 Мбит/с

В отличие от стандартов эфирного вещания, стандарт DVB-C2 может не подчиняться жесткой частотной сетке, поскольку кабельная сеть является закрытой экранированной средой и нет необходимости координировать использование ее спектра с эфирными присвоениями. Напротив, можно гибко адаптировать полосу канала под свои конкретные потребности, что позволяет расширить полосу передаваемого сигнала для размещения в нем большего количества услуг. Чтобы не усложнять и не удорожать абонентское оборудование, реализуется сегментированный прием таких каналов, аналогичный используемому в японской системе эфирного телевидения ISDB-T [8.40]. Приемник со стандартной полосой пропускания извлекает из широкого пакета только необходимую часть спектра, не превышающую, например, 8 МГц.

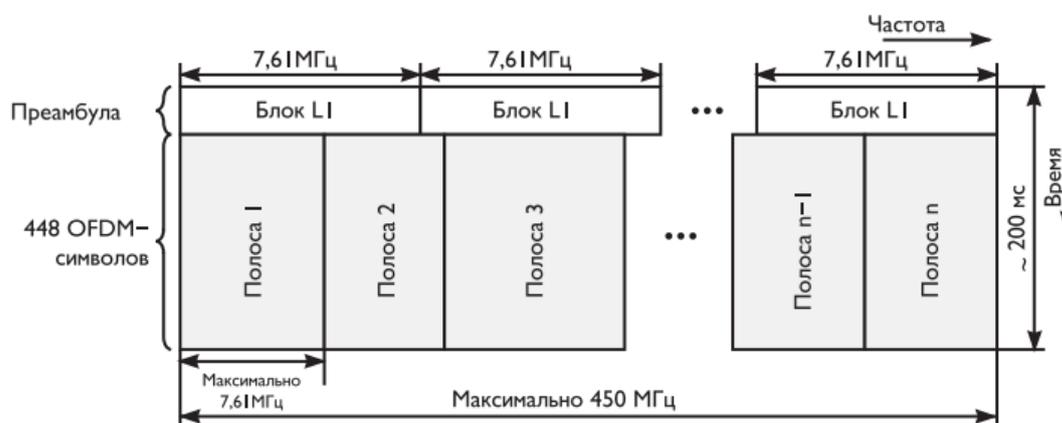


Рис. 5.199. Структура кадра DVB-C2 в частотно-временной области.

Структура кадра DVB-C2 показана на рис.5. Каждый кадр C2 начинается с преамбулы, состоящей из одного или более OFDM-символов и выполняющей две основные функции. С одной стороны, она обеспечивает надежную временную и частотную синхронизацию OFDM- сигнала и самой структуры. С другой стороны, преамбула содержит сигнализацию уровня L1, необходимую для декодирования потоков данных и содержащейся в них полезной информации. Преамбула состоит из циклически передаваемых блоков сигнализации L1, повторяющихся в каждой полосе 7,61 МГц широкого

канала. Фиксированное расположение блоков L1 и их повторение с шагом 7,61 МГц обеспечивают их прием при настройке тюнера на любые 8 МГц из занимаемого кадром диапазона.

Использование многопозиционной модуляции QAM (Quadrature Amplitude Modulation – квадратурная амплитудная модуляция), а также хорошего отношения S/N, который существенно снижает вероятность ошибок BER (Bit Error Rate – частота ошибочных бит) позволило внедрить цифровое телевидение в системы кабельного телевидения.

### Практическая часть [21].

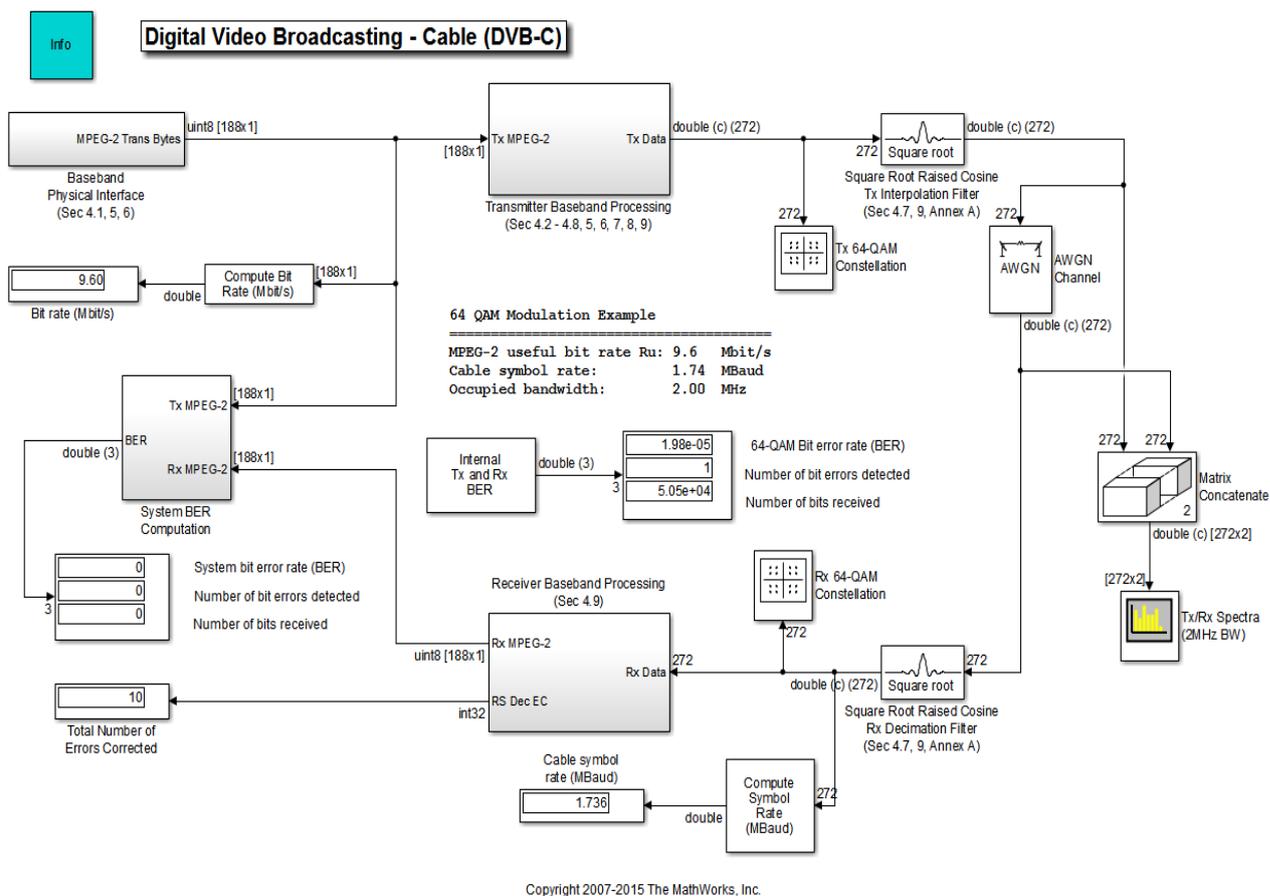


Рис. 5.200. Структурная схема системы цифрового кабельного телевидения DVB-C в Simulink MATLAB 2015

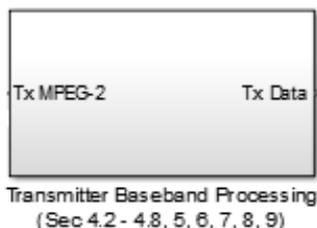


Рис. 5.201 Блок имитация потока данных MPEG-2.

Внутри данного блока производится имитация потока данных MPEG- 2.

Представленный на рис. блок включает в себя:

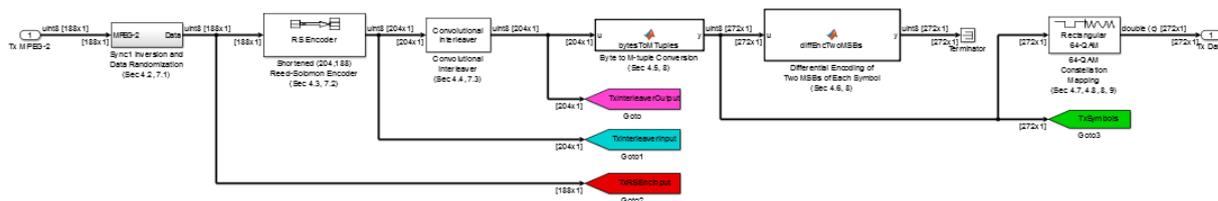


Рис. 5.202. Блок передатчика.

### 1. Sync1 Inversion and Randomization

Эта подсистема инвертирует байт, далее производится рандомизация с целью формирования спектра.

### 2. Кодировщик Рида-Соломона (204, 188).

Добавляет 16 паритетных байтов к MPEG-2.

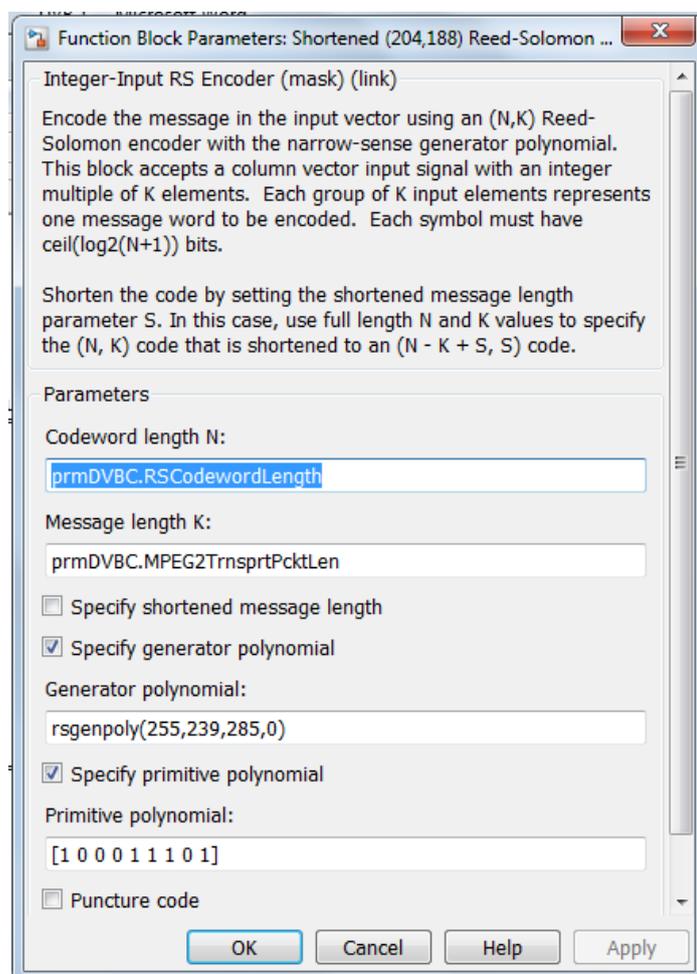


Рис. 5.203. Кодировщик Рида-Соломона (204, 188)

### 3. Сверточный перемежитель.

Процесс перемежения основан на подходе Форни.

### 4. Байт (8 бит) с M-кортежами (6-разрядная версия)

Используется, чтобы преобразовать 8-битные байты данных в 6-битные.

**5. Дифференциальное кодирование.**

**6. Отображение 64-QAM созвездия.**

Отображает в основной полосе частот значения ( I и Q ) при передаче.

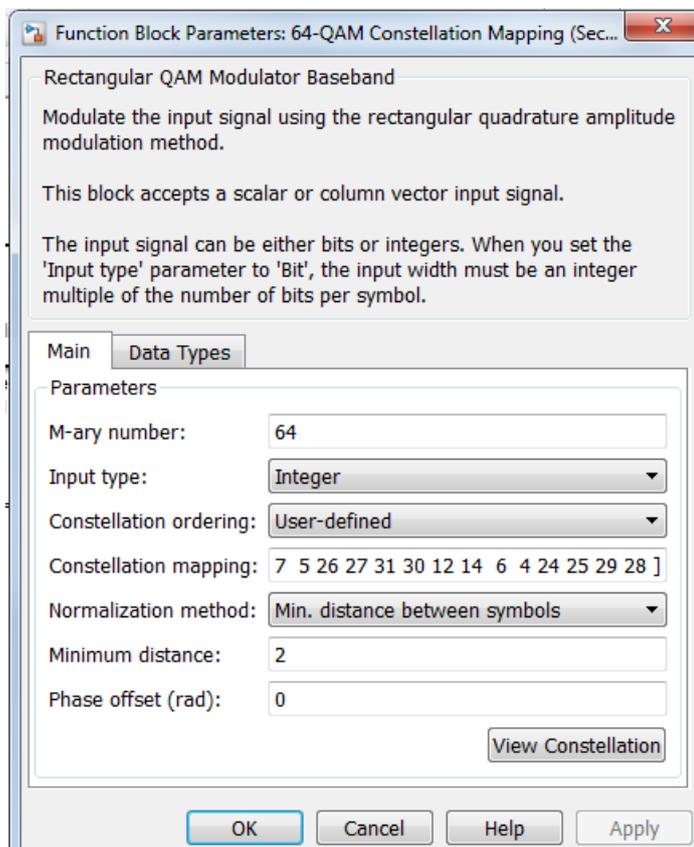
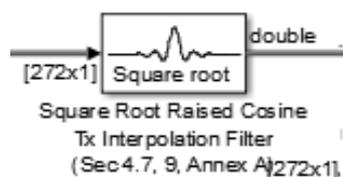
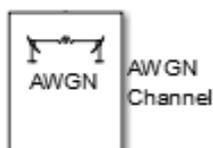


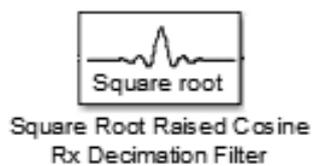
Рис. 5.204 - Отображение 64-QAM созвездия.



Этот блок выполняет низкочастотное формирование значений символов совокупности для передачи комплекса ( I и Q ).



Изменение белого гауссовского шума в пределах от  $10^{-4}$  до  $10^{-10}, 10^{-11}$ .



Прореживает (фильтрует) значения символа созвездия принимаемого комплекса ( I и Q ).

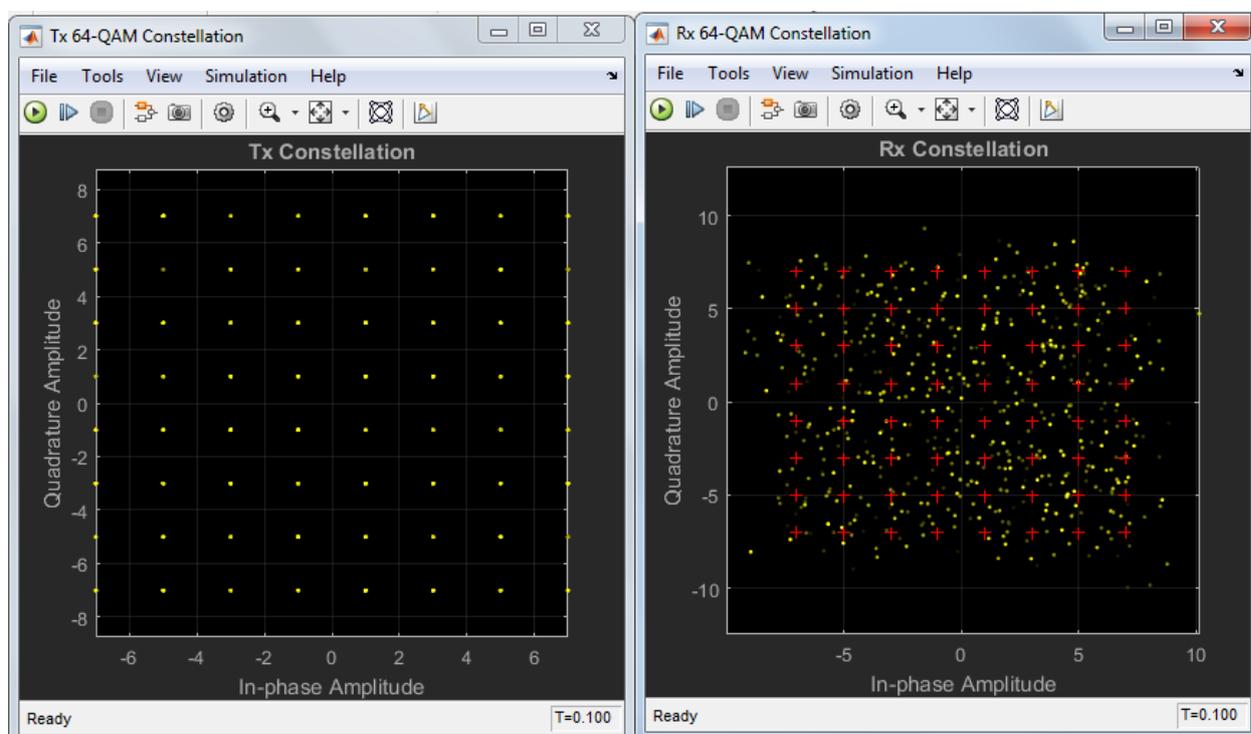
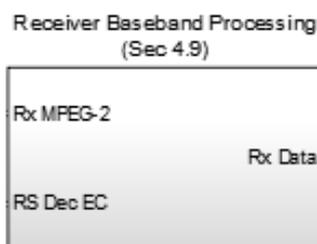


Рис. 5.205. Созвездия передатчика и приёмника при  $E_b/N_0 = 5$  дБ.

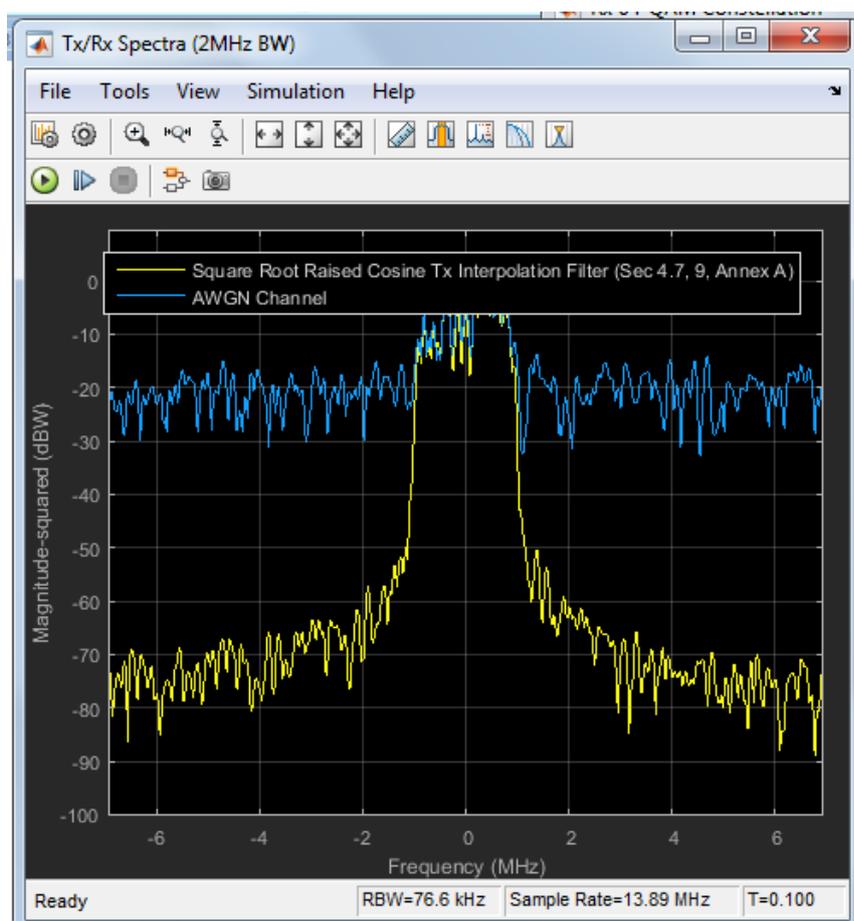


Рис. 5.206. Спектр передатчика и приёмника при  $E_b/N_0 = 5$  дБ.

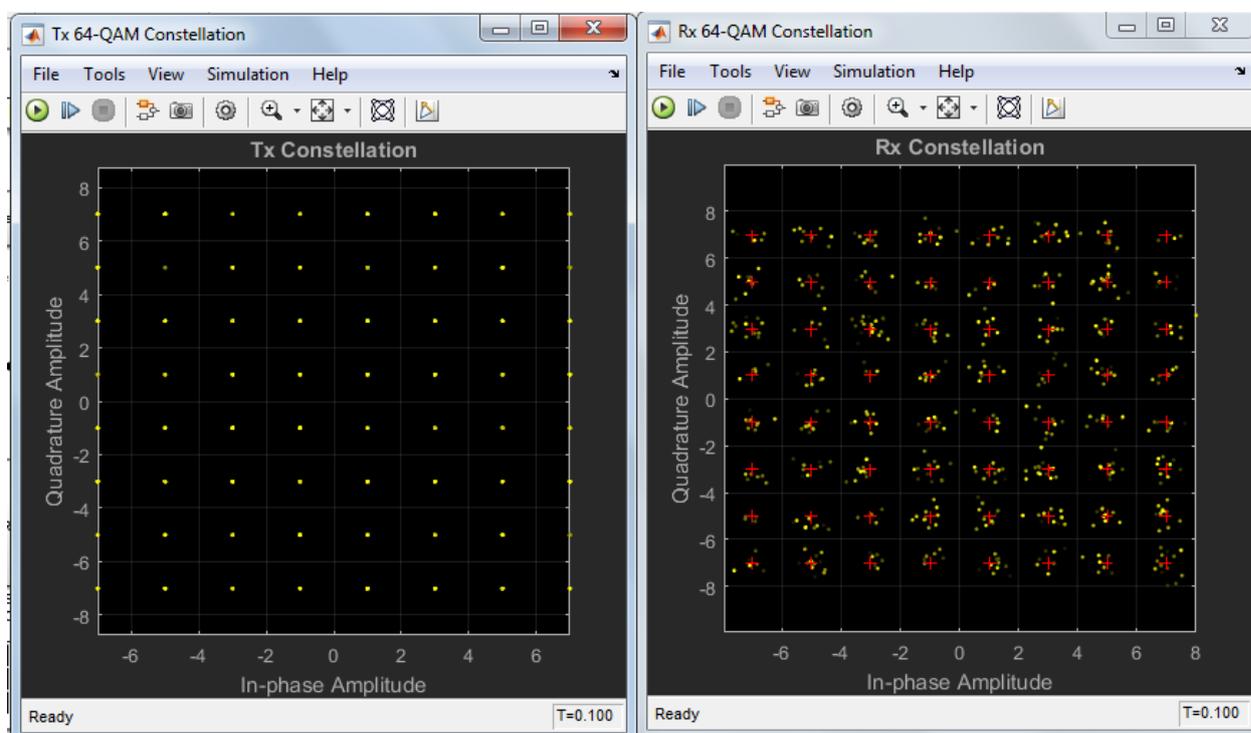


Рис. 5.207. Созвездия передатчика и приёмника при  $E_b/N_0 = 15$  дБ.

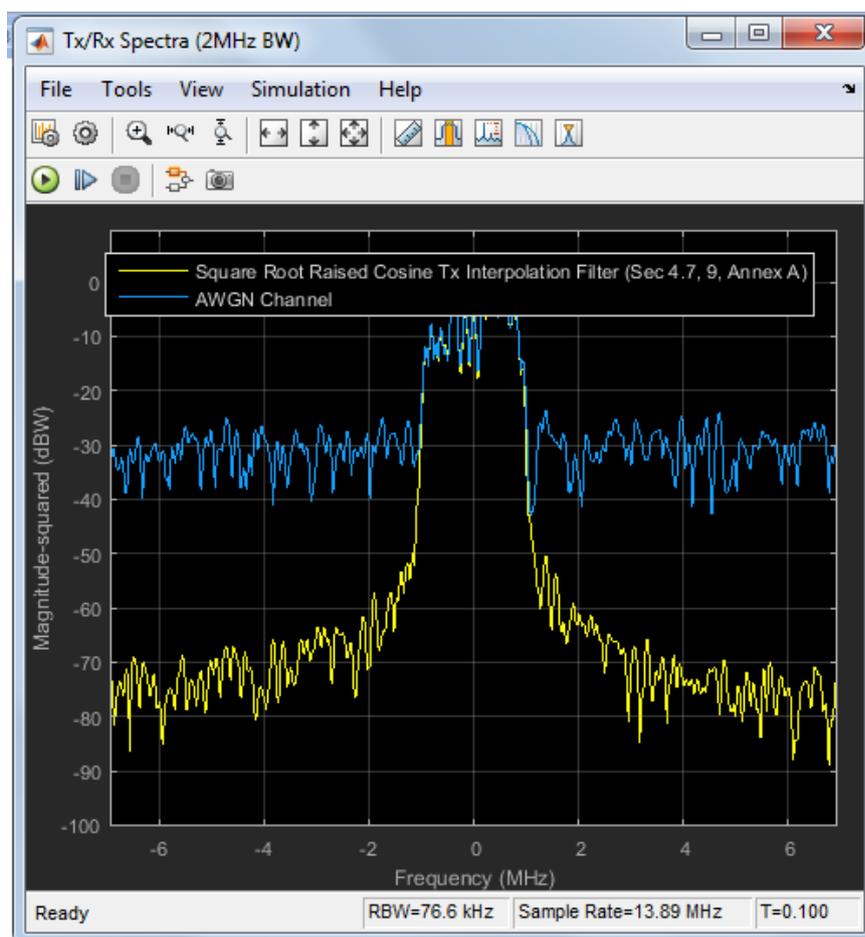


Рис. 5.208. Спектр передатчика и приёмника при  $E_b/N_0 = 15$  дБ.

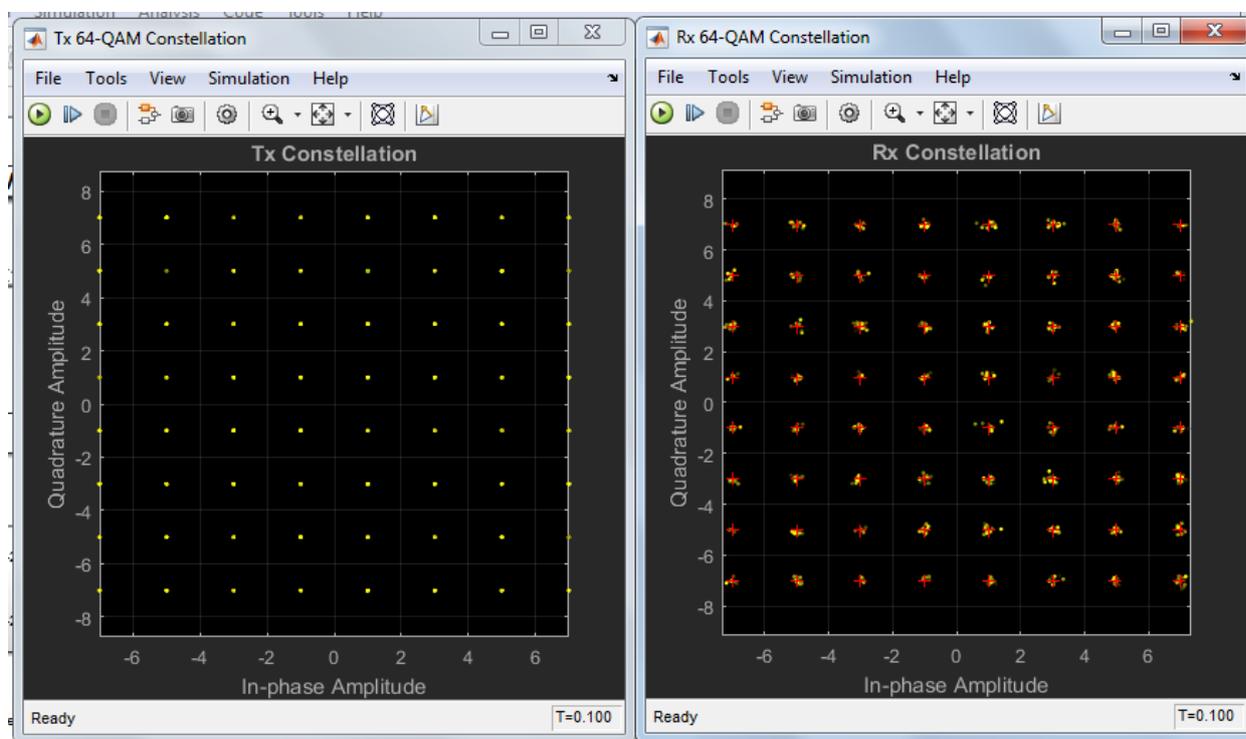


Рис. 5.209. Созвездия передатчика и приёмника при  $E_b/N_0 = 19$  дБ.

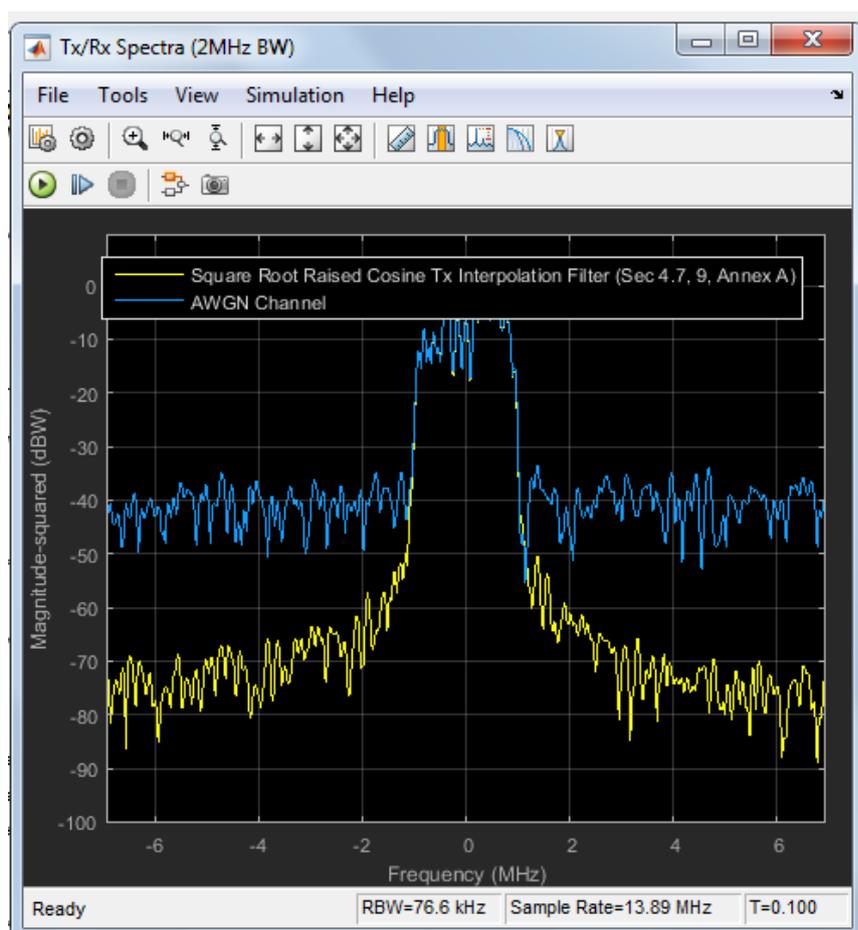


Рис. 5.210. Спектр передатчика и приёмника при  $E_b/N_0 = 19$  дБ.

Таблица 5.34. График зависимость BER от  $E_b/N_0$ .

$E_b/N_0$ , дБ	-10	-5	-2,5	1	0	1	1	1	1	1	20
BER	0.4058	0.3359	0.2885	0.2563	0.03436	0.01264	0.001024	0.000956	0.0009	0.0000	0

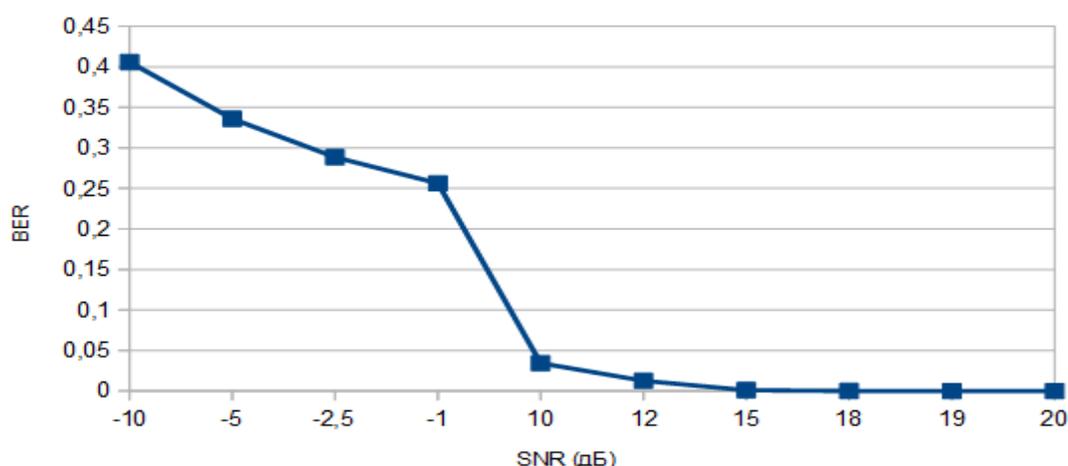


Рис. 5.211. Зависимость BER(SNR) для QAM-64.

На основе проведенного исследования можно сделать следующие выводы:

Использование многопозиционной модуляции QAM (Quadrature Amplitude Modulation – квадратурная амплитудная модуляция), а также хорошего отношения S/N, который существенно снижает вероятность ошибок BER (Bit Error Rate – частота ошибочных бит) позволило внедрить цифровое телевидение в системы кабельного телевидения.

### 5.11. Имитационное моделирование системы цифрового мобильного телевизионного вещания DVB-H и системы высокоскоростного цифрового мобильного ТВ-вещания DVB-H2 [16, 19, 20]

DVB-H (*DigitalVideoBroadcasting – Handheld*) — европейский стандарт мобильного телевидения, один из семейства стандартов DVB. Стандарт DVB-H позволяет передавать цифровой видеосигнал на мобильные устройства, такие как КПК, мобильный телефон или портативный телевизор. Формально, этот стандарт был принят в ноябре 2004 года.

DVB-H является логическим продолжением стандарта DVB-T с поддержкой дополнительных возможностей, отвечающих требованиям для переносных мобильных устройств с автономным питанием.

#### Технологии мобильного вещания телевизионного вещания DVB-H

На сегодняшний день существует 8 форматов вещания, ориентированных на прием мобильными терминалами. Во-первых, это форматы DVB-T и DVB-H. Во-вторых, MediaFLO, закрытая система разработки компании Qualcomm. В-третьих, группа форматов, базирующихся на системе радиовещания DAB. К ней относятся MovioSystem, разработанная BritishTelecom, корейские форматы T-DMB и S-DMB, а также европейский профиль формата

T-DMB. И, наконец, существует японский стандарт эфирного вещания ISDB-T, по своей гибкости пригодный для любых видов эфирного вещания на любые терминалы.

Система DVB-H разработана на базе DVB-T, что обеспечивает их частичную совместимость. Она заключается в том, что трансляции DVB-H за исключением одного режима модуляции могут приниматься приемниками DVB-T, и в одном мультиплексированном потоке возможно совмещение трансляций DVB-H и DVB-T.

В то же время в DVB-H введен ряд добавлений на физическом уровне и заметно изменен канальный уровень.

### **DVB-T и DVB-H**

Характеристики системы DVB-T неоднократно изложены в литературе<sup>1</sup>, поэтому напомним только его основные особенности. Главным отличием DVB-T от кабельной и спутниковой версий стандарта DVB является использование COFDM (CodedOrthogonalDivisionMultiplexing) модуляции. При таком способе модуляции применяется частотное мультиплексирование ортогональных несущих в сочетании с помехоустойчивым кодированием. Использование большого числа несущих позволяет удлинить время передачи каждого символа и выделить период защитного интервала для отстройки от помех многолучевого приема. В зависимости от количества ортогональных несущих в стандарте выделяется два режима 8К (8192 несущих) и 2К (2048 несущих). DVB-T предусматривает возможность использования трех видов модуляции — QPSK, 16 QAM и 64 QAM, четырех вариантов относительной длительности защитного интервала, а также пяти вариантов относительной скорости при наложении сверточного помехозащитного кодирования. Сочетания этих параметров позволяют гибко выбирать режим в зависимости от радиуса охвата соты, ландшафта и РЧ обстановки. Наличие защитного интервала дает возможность использовать DVB-T и для передачи на мобильные терминалы, в том числе движущиеся с большой скоростью. Но для передачи ТВ на мобильные телефоны и другие миниатюрные приемники эта система оказалась малоприменимой. DVB-T оптимизирован для передачи стандартных ТВ потоков, в то время как карманные приемники имеют небольшие экраны, позволяющие воспроизвести картинку формата не более чем 1/4 CIF или 1/8 CIF. Кроме того, эти терминалы питаются от слабых аккумуляторных батарей, которые желательно эксплуатировать в максимально экономичном режиме. И, наконец, они имеют слабые приемные антенны и часто должны принимать сигнал в неблагоприятных условиях, в то время как размещение стационарных эфирных антенн может быть оптимизировано. С учетом всех этих обстоятельств для эфирной передачи на карманные мобильные терминалы была разработана специальная система DVB-H (DigitalVideoBroadcasting-Handheld), по возможности совместимая с DVB-T, но одновременно учитывающая перечисленные

особенности приема. Рассмотрим компоненты DVB-H, относящиеся к физическому и каналному уровням системы ISO/OSI.

Обобщенная архитектура системы DVB-H изображена на рисунке 1. Зеленым цветом помечены элементы, добавленные в DVB-H.

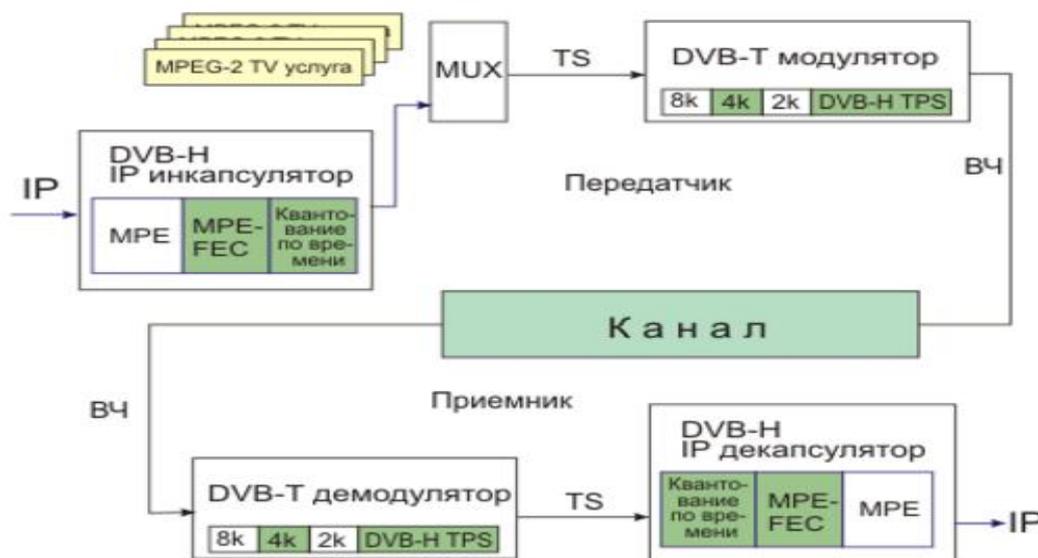


Рис. 5.212. Обобщенная архитектура системы DVB-H

### Физический уровень

На физическом уровне система DVB-H максимально приближена к DVB-T. Поэтому укажем только дополнительные возможности, появившиеся в DVB-H.

Во-первых, к режимам модуляции 2К и 8К был добавлен еще один — 4К. Это дало дополнительную степень свободы в плане обмена максимальной скорости передвижения приемника на радиус охвата одной соты. Чем меньшее количество ортогональных несущих используется при COFDM модуляции, тем больший частотный интервал образуется между соседними несущими, и, соответственно, тем выше скорость движения терминала, при котором прием срывается из-за Доплеровского смещения частот. С другой стороны, чем меньше несущих, тем короче период, выделенный для передачи каждого COFDM символа и, соответственно, короче защитный интервал. А сокращение защитного интервала снижает возможности отстройки от многолучевого приема, то есть уменьшает допустимый радиус соты. Для сетей DVB-T, рассчитанных в основном на стационарный прием, значительно более важным фактором является зона охвата. Что же касается сетей DVB-H, то там большую значимость приобретает возможность приема на скорости, а зона охвата в сильной мере ограничивается уровнем сигнала на входе тюнера.

Для возможности выбора компромиссного варианта был добавлен режим модуляции 4К, заполняющий нишу между 2К и 8К. Трансляции в режиме 4К могут приниматься только

приемниками

DVB-H.

Вторым дополнением на физическом уровне стала возможность более глубокого перемежения данных в режимах 4К и 2К. Канальное кодирование DVB-T предусматривает перемежение данных внутри одного COFDM символа. Оно в основном предназначено для компенсации селективных замираний, несущих при многолучевом приеме. В то же время мобильные терминалы с большей вероятностью могут оказаться в зоне действия широкополосных импульсных помех. И, как уже отмечалось, при приеме на скорости появляется доплеровское смещение частотного спектра, также приводящее к искажениям сигнала. Поэтому в стандартах мобильного вещания на базе COFDM (DAB, ISDB -T) для борьбы с последствиями длительных помех в цикл канального кодирования введено перемежение длинных серий данных, охватывающее десятки, а то и сотни OFDM символов.

Чем длиннее последовательность данных, участвующих в перемежении, тем эффективнее оказывается борьба с последствиями затуханий. Но для DVB-H такой путь невозможен. Во-первых, восстановление длинных последовательностей потребовало бы непрерывного приема, в то время как для целей энергосбережения в DVB-H реализован описанный ниже импульсный режим передачи. Во-вторых, для его осуществления необходимы большие объемы памяти, удорожающие приемник. И, наконец, это противоречит требованию совместимости с DVB-T. Поэтому было выбрано компромиссное решение. Для режима модуляции 8К, наиболее актуального для DVB-T, в DVB-H сохранено перемежение битов в рамках одного символа. А в режимах 4К и 2К, где каждый COFDM символ переносит меньшее количество информации, в качестве опции введена возможность временного перемежения, допускаемого объемами выделенной для этих целей памяти. Для 4К перемежение выполняется с глубиной в 2 COFDM символа, а для режима 2К — с глубиной в 4 COFDM символа. При активизации этой опции совместная передача трансляций DVB-H и DVB-T невозможна. Одновременно предусматривается опция дополнительной помехозащиты, реализованная на базе IP дейтаграмм и позволяющая в сильной мере компенсировать отсутствие глубокого перемежения. Принцип ее действия изложен

позже.

Остальные механизмы внешнего и внутреннего канального кодирования, используемые в DVB-T, без изменения перенесены в DVB-H. Третье дополнение касается транспортной сигнализации (TPS — TransmissionParameterSignalling)<sup>2</sup>, в которую добавлены два бита, индицирующие наличие в потоке услуг, передаваемых в формате DVB-H, а также наличие дополнительной кодозащиты, реализуемой на базе IP дейтаграмм. Четвертым дополнением стала возможность использования полосы 5 МГц при условии, что эта она выделяется не в вещательном диапазоне. Она добавлена к полосам 6, 7 и 8 МГц,

допускаемых к использованию в DVB-T. Ее планируется применять при развертывании сетей DVB-H в США в L-диапазоне (1,670-1,675 ГГц).

Таблица 5.35. Параметры режимов 2К, 4К, 8К

Параметр	2К	4К	8К
Число активных несущих	1705	3409	6817
Число информационных несущих	1512	3024	6048
Длительность периода T, мс	0,109	0,109	0,109
Полезная символьная часть T <sub>u</sub> , мс	224	448	896
Разнос между несущими 1/T <sub>u</sub> , Гц	4464	2232	1116
Разнос между несущими K <sub>min</sub> и K <sub>max</sub> , МГц	7,61	7,61	7,61

### Канальный уровень

Одно из основных отличий DVB-H от DVB-T заключается в том, что в новой системе вся информация должна передаваться в форме IP дейтаграмм, инкапсулируемых в транспортные пакеты MPEG-2 TS с использованием метода многопротокольной инкапсуляции (MPE MultiProtocolEncapsulation). Это один из четырех методов инкапсуляции пакетов данных в транспортные пакеты MPEG-2 TS, определенных DVB, единственно пригодный для передачи потоковых услуг. Схема инкапсуляции показана на рисунке 5.213.

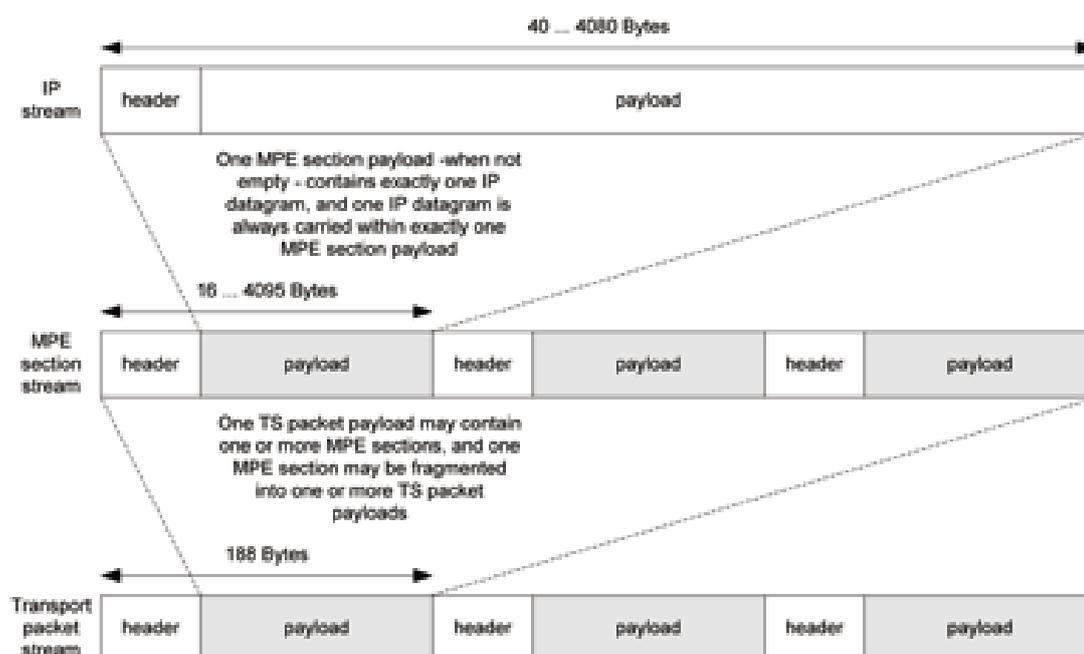


Рис. 5.213. Формат IP дейтаграммы

IP пакеты инкапсулируются в MPE секции, а те, в свою очередь, — в транспортные пакеты MPEG-2 TS, переносящие элементарные потоки. Каждый IP пакет занимает одну MPE секцию, длина которой практически не коррелирована с емкостью пакетов MPEG2-TS. В одном пакете может передаваться множество MPE секций, и, наоборот, одна секция может занимать несколько транспортных пакетов.

Данные, относящиеся к одной услуге, инкапсулируются в транспортные пакеты MPEG-2 с постоянным идентификационным номером PID. Использование такого стека обусловлено тремя причинами.

Во-первых, в системах DVB-H предполагается передавать ТВ потоки, компрессированные не в MPEG-2, а в более эффективных форматах, в первую очередь, в H.264 /AVC3, для которых процесс инкапсуляции компрессированных аудио и видео в транспортные пакеты MPEG-2 TS жестко не специфицирован и обычно реализуется как раз через IP/MPE инкапсуляцию. Более того, DVB-H потенциально рассматривается как составная часть гибридной системы доставки мультимедийных услуг (IPDC).

В связи с этим понятие элементарного потока в DVB-H определяется иначе, чем в стандарте MPEG-2. В DVB-H это просто поток, передаваемый в пакетах с одним PID-ом. Снята жесткая корреляция элементарного потока с данными определенного типа. В одном элементарном потоке могут передаваться все данные, относящиеся к определенной ТВ программе или даже к нескольким программам. В последнем случае потоки разных ТВ программ будут передаваться в дейтаграммах с разным мультикастовым IP адресом и заключаться в MPE секции с разными MAC адресами. Аналогичным образом могут передаваться и не телевизионные услуги.

### **MPE-FEC**

В DVB-H канальное кодирование накладывается на всю последовательность IP дейтаграмм, передаваемых в одном слоте, то есть на максимально возможный объем данных. Это кодирование введено в качестве опции и выполняется кодом Рида-Соломона. Принцип кодирования показан на рисунке 5.214.

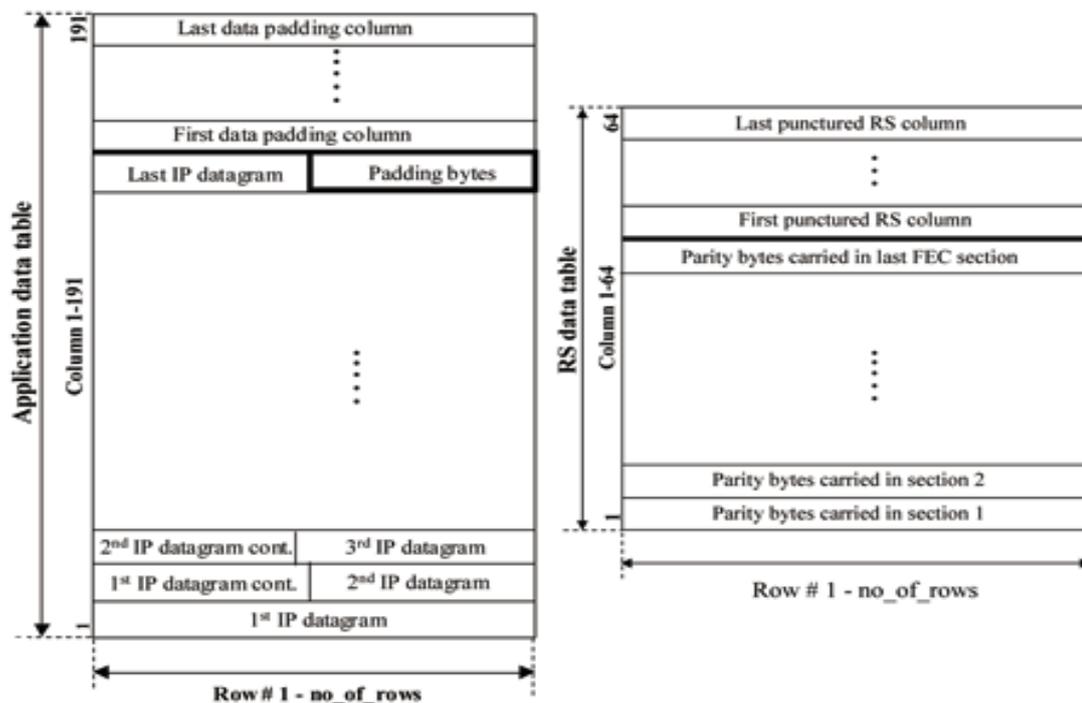


Рис. 5.214. Кодирование пакетов кодом Рида-Соломона

IP дейтаграммы помещаются в таблицу `ApplicationDataTable`, в которой последовательно заполняются столбцы начиная с левого верхнего угла. Высота столбцов может составлять от 1 до 1024 байт в зависимости длины дейтограмм, а их количество всегда одинаково — 191. Если в таблице остается пустое место, то оно заполняется холостыми байтами. Затем каждый ряд таблицы кодируется кодом Рида-Соломона (255, 191), в результате чего формируется 64 контрольных байта, заносимых в соответствующий ряд FEC таблицы. Затем обе таблицы инкапсулируются в пакеты MPEG-2 TS и транслируются в одном слоте. Вначале — информационная часть, а затем — контрольная. Причем байты контрольной таблицы при инкапсуляции считываются не рядами, как формируются, а колонками. Так создается виртуальное перемежение контрольных байт, требующее минимальных ресурсов приемника для восстановления их последовательности. Этот метод помехозащитного кодирования получил название MPE-FEC. MPE-FEC декодирование рекомендуется проводить на базе модели канала со стиранием<sup>4</sup>. Такое декодирование используется в каналах с пакетной передачей и включает два этапа.

На первом этапе с помощью циклического (CRC-х) кода выявляются искаженные пакеты и локализуются пораженные части потока. На втором этапе выполняются восстановление пораженных пакетов, что при предварительной локализации искажений происходит более эффективно. CRC-х кодирование — стандартный способ помехозащиты пакетов информации. В частности, каждая MPE секция защищается кодом CRC-32. При использовании MPE-FEC незащищенными остаются служебные таблицы. Но с учетом того, что их содержание в большинстве случаев довольно статично, то после нескольких циклов

передачи ресиверу удастся получить нужную информацию даже в сложных условиях. Посылка, защищенная MPE-FEC, может быть принята и ресиверами DVB-T. Но они будут игнорировать контрольные байты и не смогут воспользоваться защитой MPE-FEC.

Таким образом, особенности канального уровня формата DVB-H не препятствуют приему трансляций ресиверами DVB-T. Они просто будут принимать их неоптимальным образом. различными будут и условия приема трансляций DVB-T и DVB-H. Экспериментальные измерения показали, что для достижения передатчиками DVB-H и DVB-T одинаковой зоны охвата мощность первого должна быть на 20 дБ больше. В то же время требуемый для устойчивого приема уровень несущая/шум в DVB-H в среднем на 30% ниже, а максимально возможная скорость движения приемника – на 40% выше. В таблицах 5.16 и 5.17 представлены расчетные значения цифровых потоков для разных форматов модуляции и длительностей используемых интервалов.

Таблица 5.36. MPE-FEC кодированием в 3/4

Модуляция	Скорость кодирования	Защитный интервал			
		1/4	1/8	1/16	1/32
QPSK	1/2	3,74	4,15	4,39	4,52
	2/3	4,98	5,53	5,86	6,03
	3/4	5,6	6,22	6,59	6,79
	5/6	6,22	6,92	7,32	7,54
	7/8	6,53	7,26	7,69	7,92
16QAM	1/2	7,46	8,3	8,78	9,05
	2/3	9,95	11,06	11,71	12,07
	3/4	11,2	12,44	13,17	13,58
	5/6	12,44	13,82	14,64	15,08
	7/8	13,07	14,51	15,37	15,83
64QAM	1/2	11,2	12,44	13,17	13,58
	2/3	14,93	16,59	17,57	18,1
	3/4	16,79	18,66	19,76	20,36
	5/6	18,66	20,74	21,95	22,62
	7/8	19,6	21,77	23,06	23,75

Таблица 5.40. Длительность интервалов MPE-FEC кодированием в 3/4

Параметр	Режим											
	2k	4k	8k	2k	4k	8k	2k	4k	8k	2k	4k	8k
Полезная символьная часть $T_U$	2048 T 224 мкс											
Защитный интервал $\Delta/T_U$	1/4			1/8			1/16			1/32		
Длительность защитного интервала $T_g$	51 2 T 56 ms	10 24 T 112 ms	20 48 T 224 ms	25 6 T 28 ms	51 2 T 56 ms	10 24 T 112 ms	12 8 T 14 ms	25 6 T 28 ms	51 2 T 56 ms	64 8 T 7 ms	12 6 T 14 ms	25 6 T 28 ms
Полная символьная продолжительность $T_S = \Delta + T_U$	25 60 T 280 ms	51 20 T 560 ms	10 24 T 1120 ms	23 8 T 252 ms	46 16 T 504 ms	92 76 T 1008 ms	21 52 T 238 ms	43 104 T 476 ms	87 12 T 952 ms	21 24 T 231 ms	42 24 T 462 ms	84 48 T 924 ms

В стандарте DVB-T в качестве базовой используется OFDM модуляция, благодаря которой и достигаются уникальные свойства в части возможности построения одночастотных сетей (SFN – SingleFrequencyNetwork), возможности обеспечения низкого требуемого отношения несущая/шум (C/N), высокой защиты от переотраженных объектов и низкой чувствительности к эффекту Доплера (при приеме в движении). Помимо основных видов модуляции (QPSK, 16 QAM и 64 QAM) в стандарте DVB-T используется также и иерархическая модуляция, позволяющая в потоке с высоким приоритетом передавать меньшее число программ и даже с более худшим качеством, но со значительным увеличением зоны покрытия, представляя тем самым вести прием на мобильные устройства.

## DVB-H2

DVB-H2 (NewGenerationHandheld) - на основе стандарта DVB-T2, DVB-NGH открывает путь для улучшения возможностей приема сигнала на мобильные и портативные устройства.

Они включают в себя MIMO (MultipleInputMultipleOutput, мультивход и мультивыход), частотно-временное разнесение (TFS) с одним тюнером, повернутые созвездия, улучшена и расширена проверка на четность с низкой плотностью, более эффективное чередование по времени и ультра-надежный уровень сжатия Layer-1. Спецификация DVB-NGH также включает гибридные профили, где наземные и спутниковые методы передачи данных могут быть объединены.

DVB-NGH охватывает последние модуляции и технологии кодирования и может рассматриваться как наиболее сложный радиointерфейс наземного вещания. Кроме того, он также предлагает дополнительную гибкость в эксплуатации, например, различные виды защиты для аудио- и видеопотоков в одном сервисе.

### Разработка структурной схемы программного комплекса [21]

Система DVB-H была реализована в программной среде Matlab 2015a. Для того, чтобы запустить систему необходимо вести команду: `open_system('commdvbt_alt');`

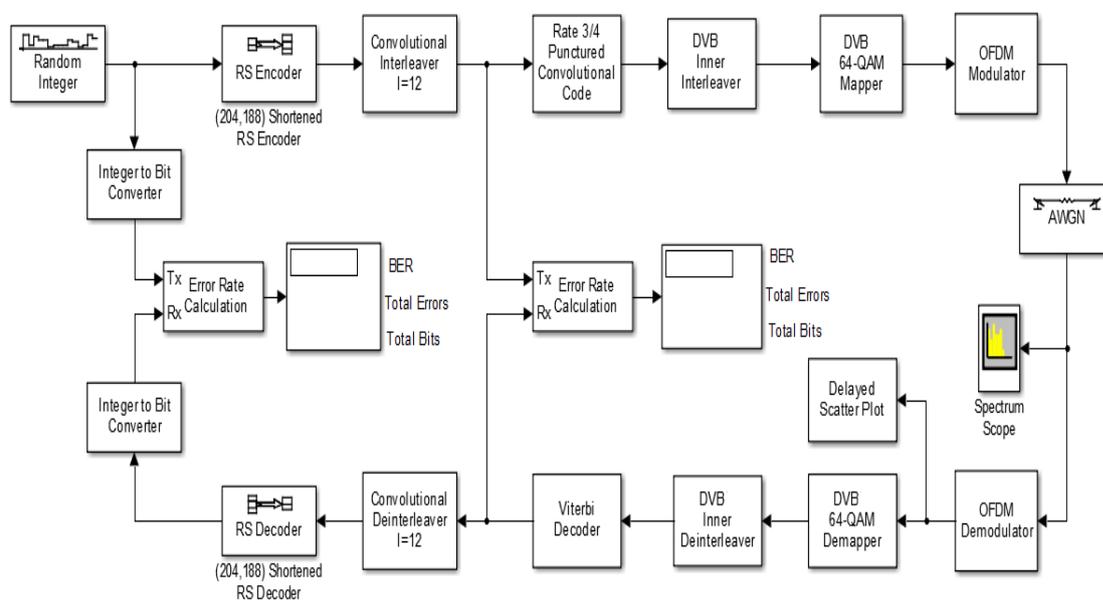


Рис. 5.215. Структурная схема DVB-H в режиме 2k Mode [21]

Передающая часть структурной схемы системы состоит из следующих блоков:

RandomInteger—Генератор псевдослучайное последовательности.

RSEncoder – Код Рида-Соломона (255,191).

ConvolutionalInterleaver – Сверточный перемежитель.

PuncturedConvolutionalCode – Сверточный кодер, с порождающими полиномами  $G1=171$  и  $G2=133$ .

DVBInnerInterleaver – Внутренний перемежитель, состоящий из бит перемежителя и символьного перемежителя. В битовом перемежителе данные демультиплексируются на  $v$  подпотоков, где  $v = 2, 4$  и  $6$  для QPSK, 16-QAM и 64-QAM, соответственно.

DVBM-QAMMapper – Все данные поднесущих объединяются в одном символе OFDM, которые модулируются с использованием QPSK, 16-QAM, 64-QAM.

OFDMModulator – Каждый символ состоит из 6817 и 1705 несущих для 8k и 2k режимов соответственно. Длительность символа состоит из двух частей: полезная часть и защитный интервал (1/4, 1/8, 1/16, 1/32).

AWGN – Канал с шумами.

### Практическая часть

Запустить модель системы DVB-H в программе Matlab следующим образом: Matlab R2015 – Simulink Model – Open – dvbh.slx [21].

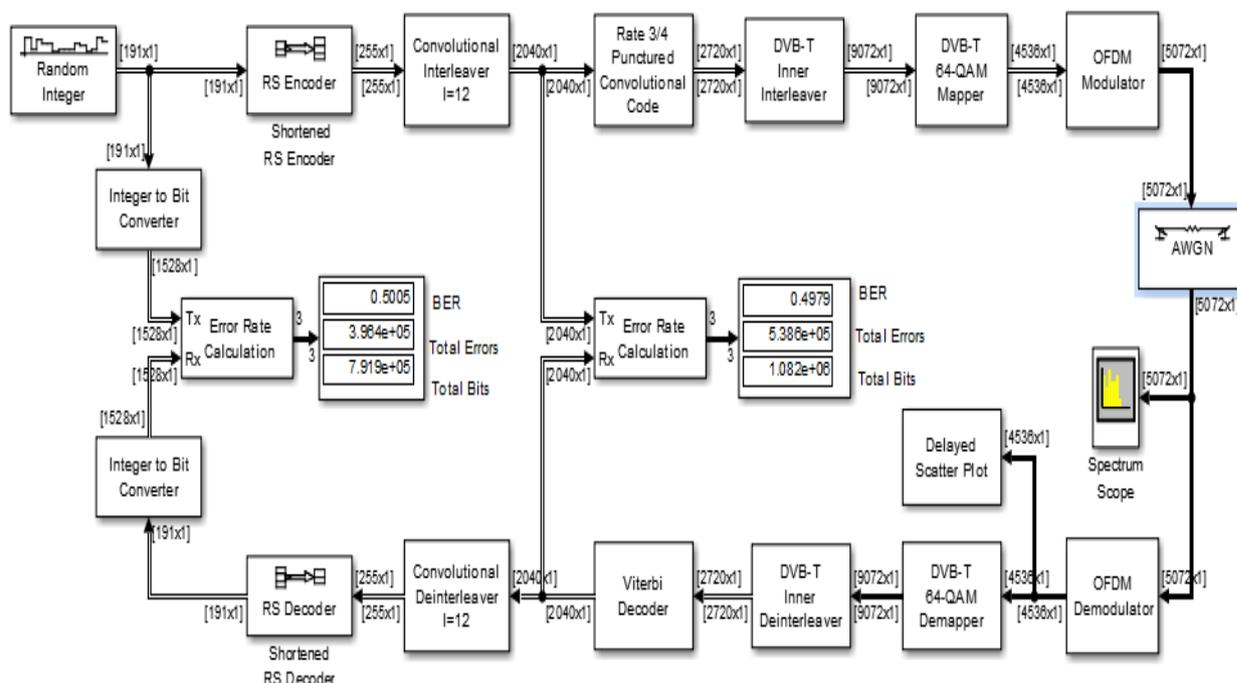


Рис. 5.216. Функциональная схема системы DVB-H реализованная в Matlab R2015b – Simulink

Выставить необходимые параметры для следующих блоков: Random-Integer Generator, RS Encoder-Decoder, Параметры DVB Inner Deinterleaver (Buffer3), OFDM modulator-demodulator (для QPSK длина FFT: 5072, для 16-QAM длина FFT: 2804).

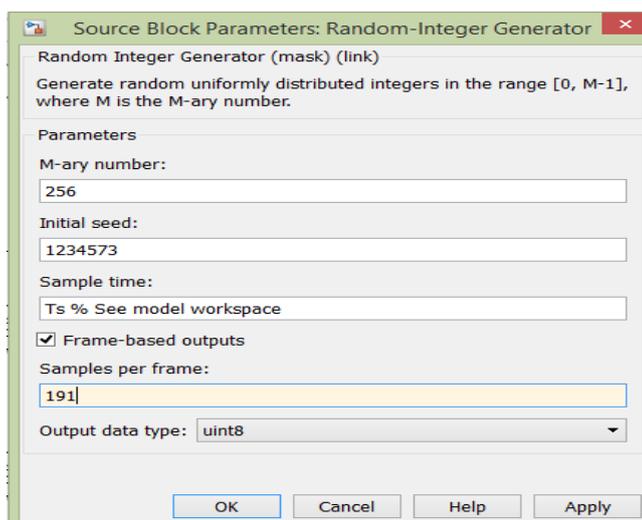


Рис. 5.217. Параметры Random-Integer Generator

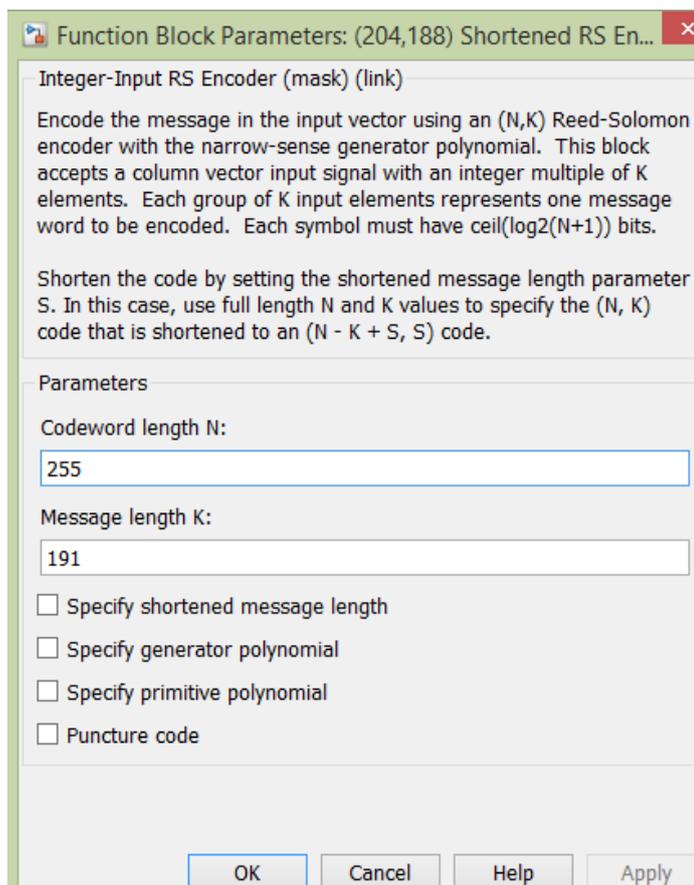


Рис. 5.218. Параметры RS Encoder-Decoder

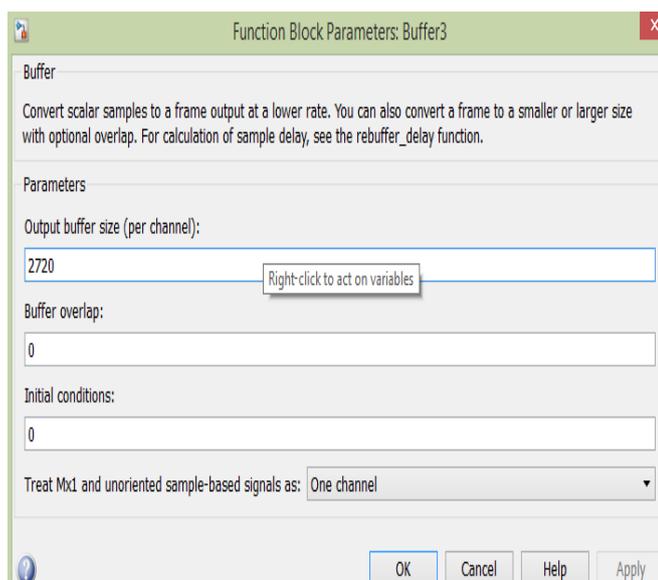


Рис. 5.219. Параметры DVB Inner Deinterleaver (Buffer3)

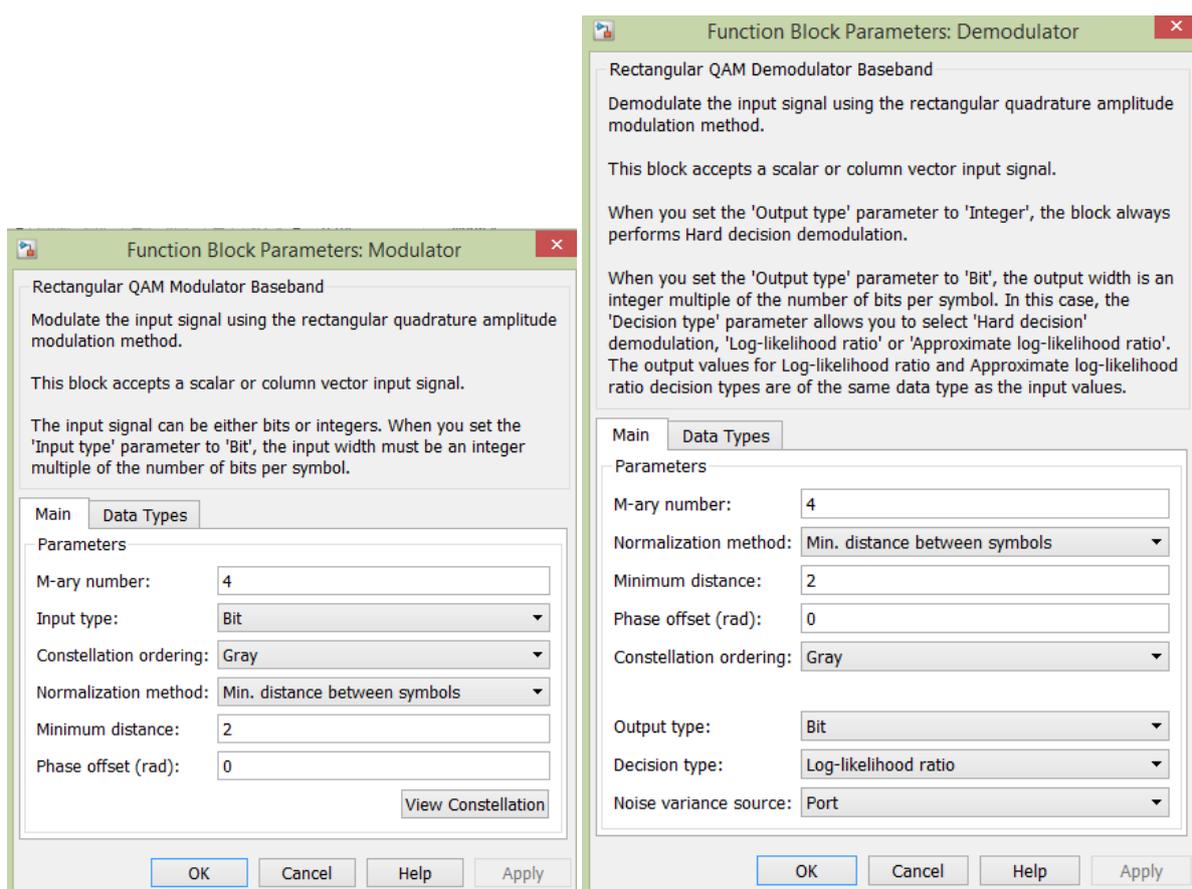


Рис. 5.220. Параметры QPSK модулятора

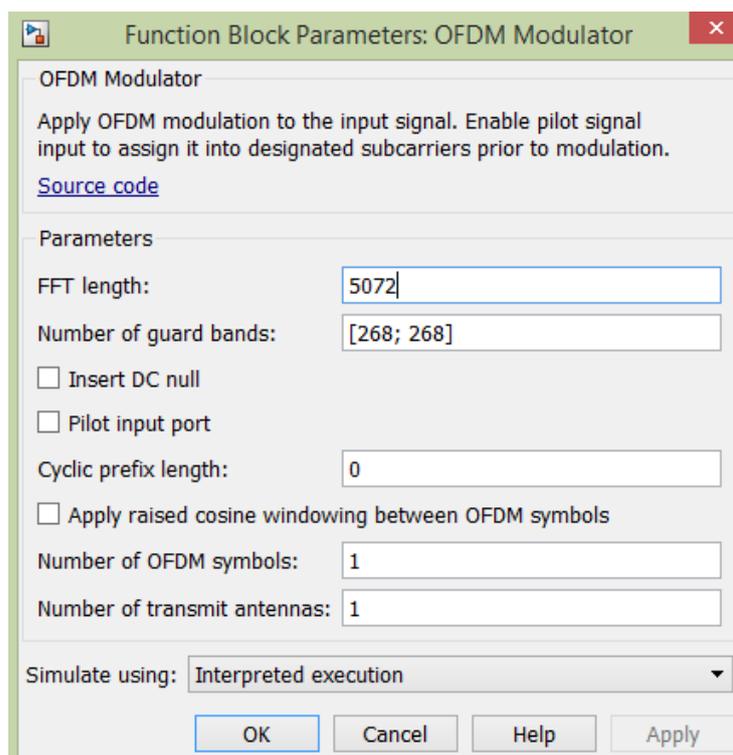
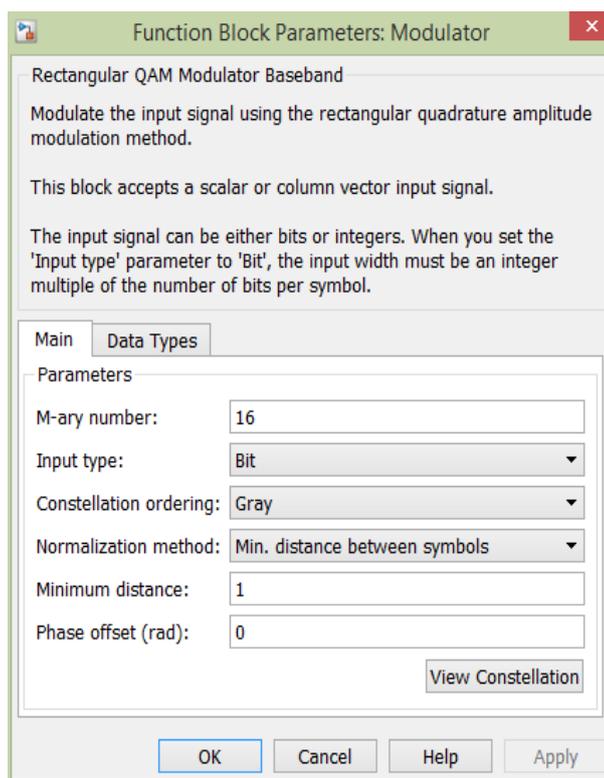


Рис. 5.221. Параметры OFDM modulator-demodulator



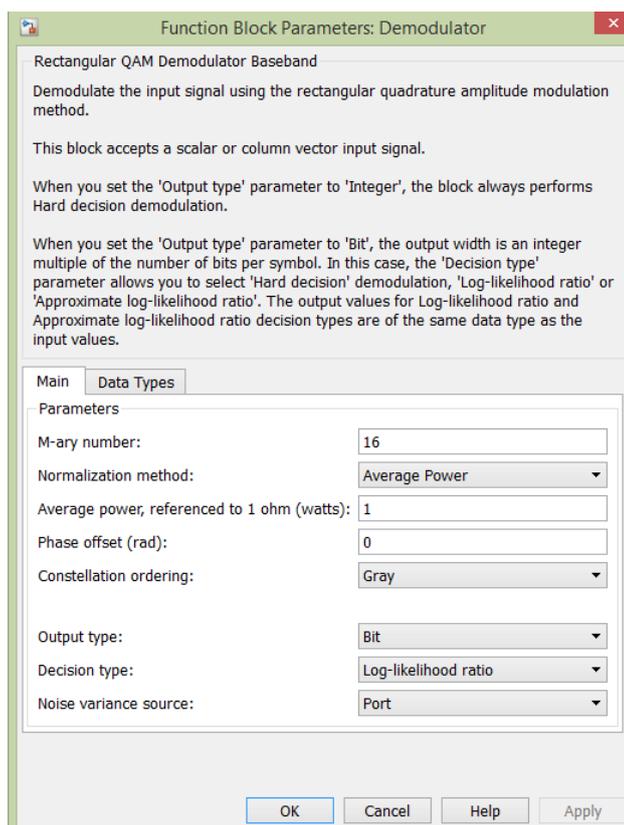


Рис. 5.222. Параметры 16-QAM модулятора

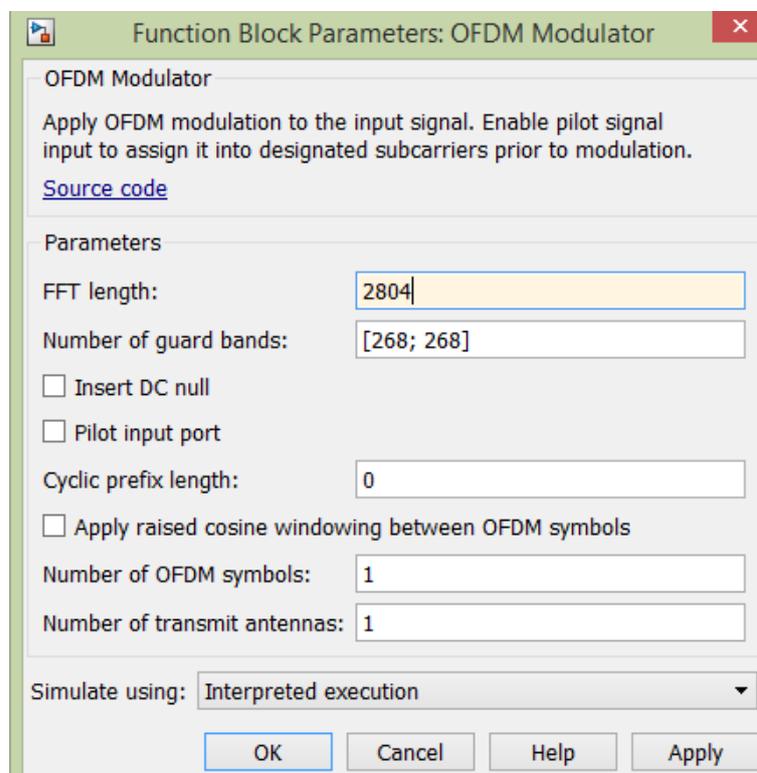


Рисунок 5.223. Параметры OFDM modulator-demodulator

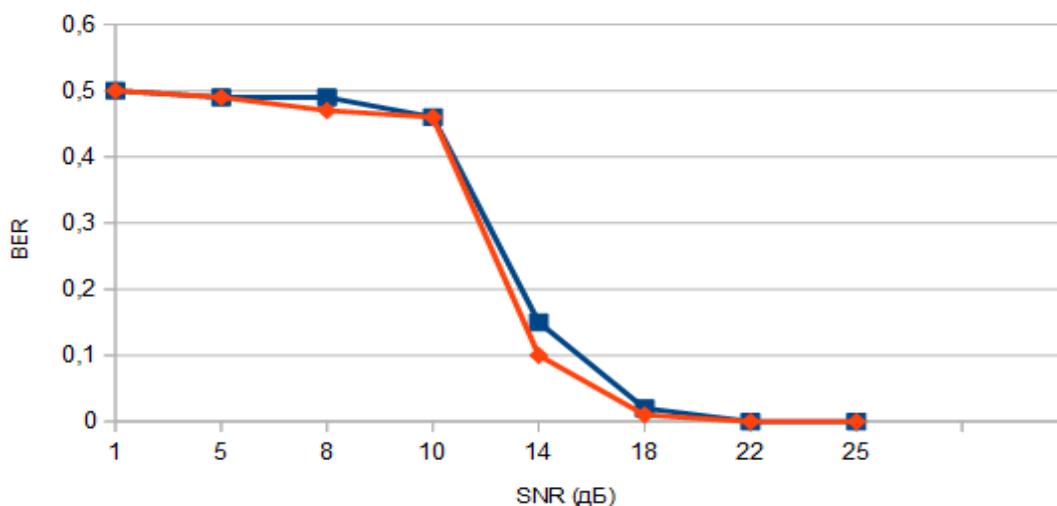


Рис. 5.224. Зависимость BER от SNR для системы DVB-H при использовании QPSK (синий) и 16-QAM (красный).

При исследовании зависимости битовой вероятности битовой ошибки от отношения сигнал/шум рассматриваемой системы мобильного вещания были сняты изображения спектра передаваемого сигнала и диаграммы созвездий QPSK и 16-QAM исследуемой системы при SNR равном 1 дБ, 18 дБ.

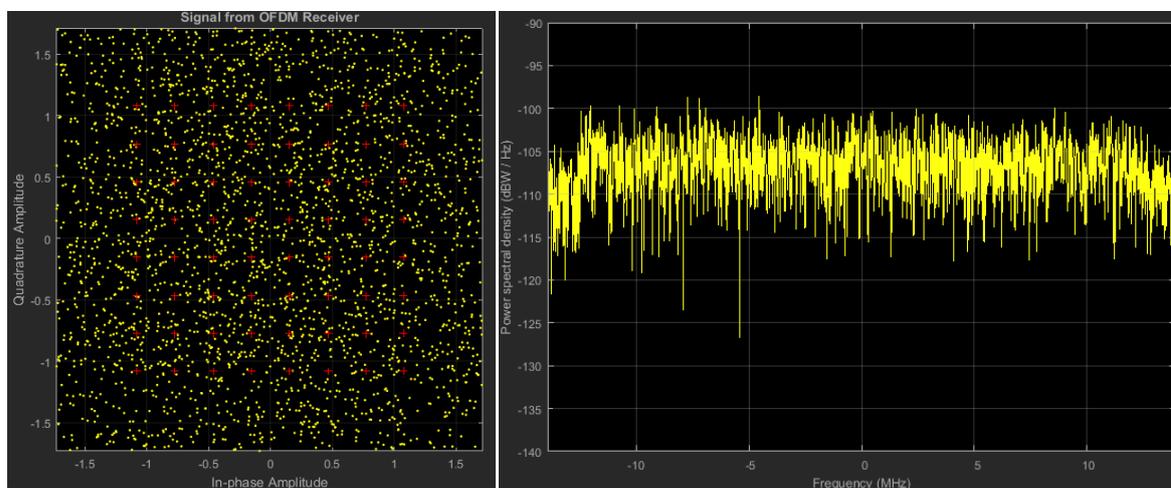


Рис. 5.225. Спектр OFDM-сигнала и диаграмма созвездий QPSK при SNR=1 дБ

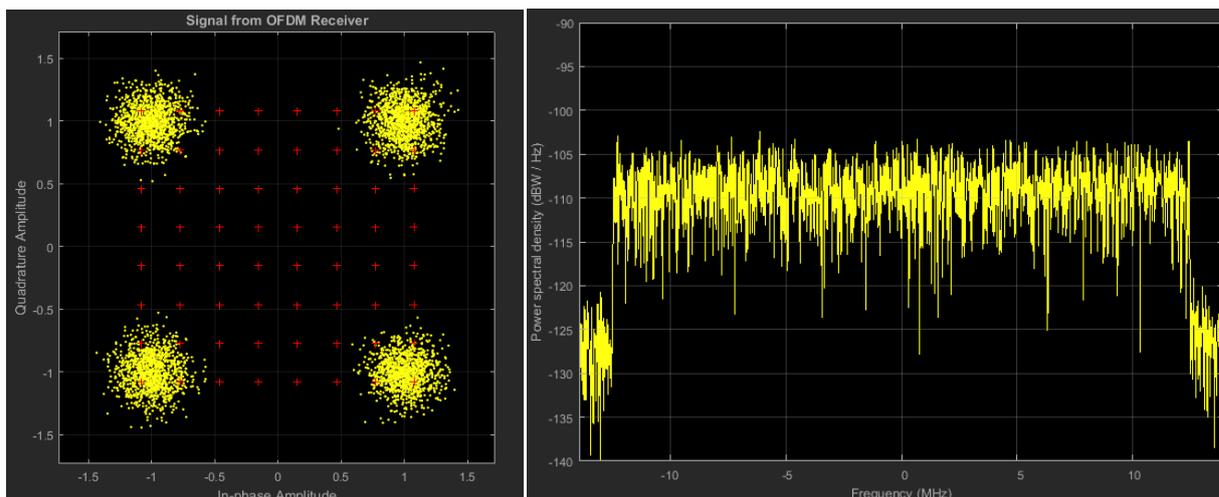


Рис. 5.226. Спектр OFDM-сигнала и диаграмма созвездий QPSK при SNR=18Б

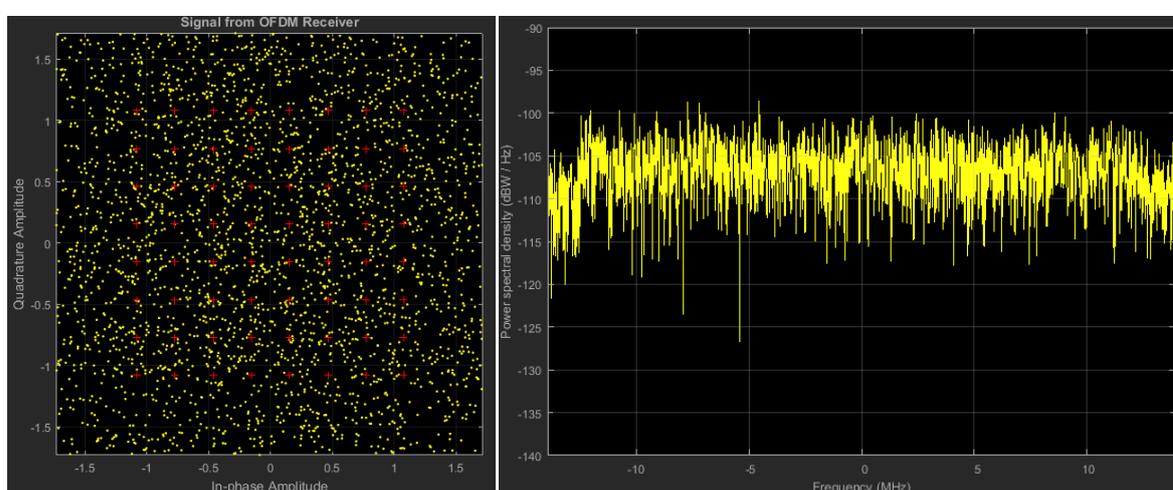


Рис. 5.227. Спектр OFDM-сигнала и диаграмма созвездий 16-QAM при SNR=1 дБ

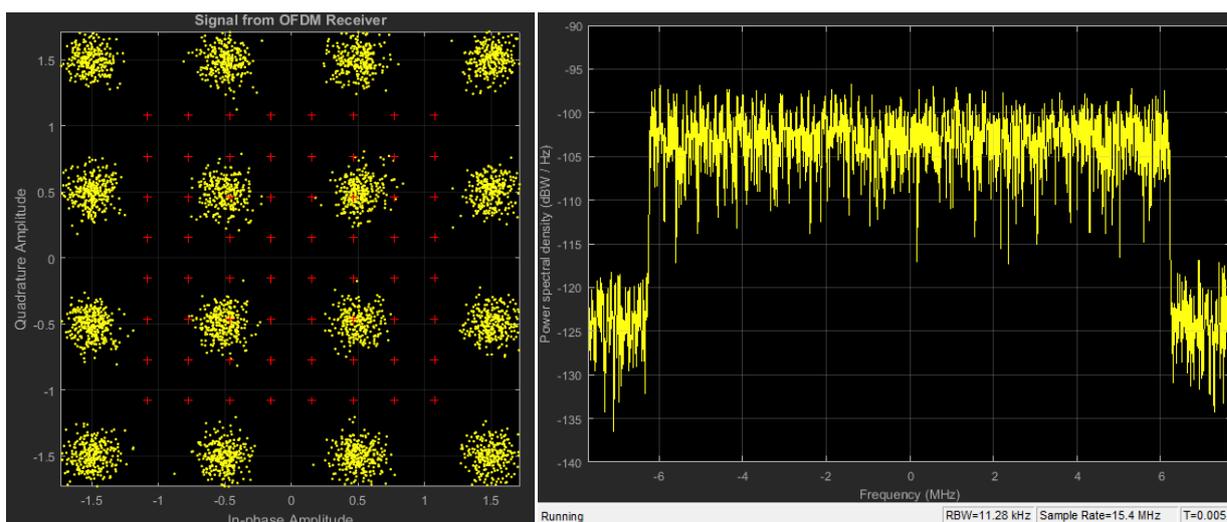


Рис. 5.228. Спектр OFDM-сигнала и диаграмма созвездий 16-QAM при SNR=18Б

В процессе выполнения данной лабораторной работы были изучены основные теоретические аспекты системы цифрового мобильного телевизионного стандарта DVB-H

При выполнении практической части работы была построена зависимость битовой вероятности ошибки от отношения сигнал/шум для QPSK и 16-QAM модуляции. Результат представлен в виде графика (рисунок 3.9).

Были сняты изображения спектра OFDM-символа и диаграммы созвездий QPSK и 16-QAM при прохождении сигнала в канале с аддитивным белым гауссовским шумом.

Полученные в результате моделирования данные позволяют сделать вывод о том, что безошибочная передача данных по каналу связи в системе DVB-T возможна при отношении сигнал/шум не менее 18 дБ.

DVB-H является обновлением для основного стандарта, которое решает проблемы мобильного приема. Главное нововведение - timeslicing. Передатчик циклически выдает в эфир пакеты, принадлежащие всем транслируемым каналам по очереди. Передача осуществляется короткими импульсами с использованием максимальной пропускной способности канала. Приемник включается только в определенные моменты, когда необходимо загрузить очередную порцию видеопотока. Это позволило в 10 раз увеличить продолжительность автономной работы портативных телевизоров. Дело в том, что для приема DVB-T применяются довольно сложные чипы, производятся интенсивные математические вычисления. И когда система работает постоянно, без перерыва, то автономность лучших образцов мобильных устройств достигает 20-40 минут. В свою очередь техника, основанная на DVB-H, способна функционировать до 10 часов от одного заряда батареи.

Другая особенность стандарта - высокая помехоустойчивость за счет введения механизма коррекции ошибок. В обычном DVB-T используется разнесенный прием на несколько антенн, что позволяет системе выбирать наименее поврежденный сигнал. В портативном устройстве такое решение реализовать труднее.

Третья важная особенность - DVB-H основывается на IP-протоколе, а это значительно упрощает и удешевляет построение вспомогательной инфраструктуры. Возможным становится использование готовых, недорогих программных решений.

## **ЗАКЛЮЧЕНИЕ**

В представленном учебном пособии впервые рассмотрены основные виды модуляции, кодирования и сигнально-кодовые конструкции современных модемов и кодеков перспективных стандартов телекоммуникационных систем - CDMA, IEEE 802.11 (WiFi), IEEE 802.15.4 (ZigBee), IEEE 802.15.1 (Bluetooth), IEEE 802.16

(WiMAX), IEEE 802. 20 (LTE), DVB-T, DVB-S и DVB-S2, DVB-C и DVB-C2, DVB-H и DVB-H2.

В учебном пособии рассмотрена теория и практика моделирования модемов сотовой связи FSK, MSK GMSK и модемы спутниковых систем связи M-QAM, M-PSK. Для анализа использовано программное обеспечение NI LabVIEW. Показана возможность создания созвездий различных видов модуляции, спектры сигналов с использованием различных фильтров, получены глазковые диаграммы и фазовые зависимости от отношения отношения сигнал/шум. Для каждого вида модуляции проведен численный анализ вероятности символьной ошибки от отношения сигнал/шум.

Рассмотрены вопросы пропускной способности канала связи, рассмотрены методы кодирования источника (Шеннона-Фано, алгоритмы сжатия Лемпеля - Зива, вейвлет-фрактальное сжатие). Проведены численные оценки коэффициентов сжатия, полученные с использованием современного программного обеспечения.

Проведено имитационное моделирование и исследование помехоустойчивых кодов Хемминга, БЧХ (Боуза-Чоудхури-Хоквенгема), Рида-Соломона на базе MATLAB 2015 Simulink, исследовано сверточные кодирование и декодирование. Проведено моделирование декодирование сверточных кодов по методу Витерби с использованием ПО MATLAB. Исследование турбокодирования, рассмотрена обобщенная схема турбокодера с параллельным каскадированием, сверточные турбокоды, декодирование турбокодов. Характеристики помехоустойчивости сверточных турбокодов исследованы с использованием ПО MATLAB 2015 Simulink. Впервые использовано ПО NI LabVIEW для исследования низкоплотностных кодов - разработан программный комплекс для визуализации и исследования LDPC-кодов. Проведена оценка помехоустойчивости LDPC-кодов в зависимости от отношения сигнал/шум и параметров кода.

Рассмотрены сигнально-кодовые конструкции на основе Треллис кодовой модуляции (TCM) и их анализ с использованием MATLAB. Исследование сигнально-кодовой конструкции на базе системы с ортогональным частотным мультиплексированием и пространственно-временным кодированием OFDM - MIMO с использованием NI LabVIEW.

Проведено имитационное моделирование на базе MATLAB 2015b Simulink модемов и кодеков современных телекоммуникационных систем стандарта CDMA, системы мобильной связи стандарта IEEE 802.11 (WiFi), мобильной связи стандарта IEEE 802.15.4 ZigBee, системы мобильной связи стандарта IEEE 802.15.1 (Bluetooth), системы мобильной связи стандарта IEEE 802.16 (WiMAX), системы мобильной связи стандарта IEEE 802. 20 LTE, системы цифрового наземного телевизионного вещания

DVB-T, системы цифрового спутникового телевизионного вещания DVB-S и системы высокоскоростного цифрового спутникового ТВ-вещания DVB-S2, системы цифрового кабельного телевизионного вещания DVB-C и системы высокоскоростного цифрового кабельного ТВ-вещания DVB-C2, системы цифрового мобильного телевизионного вещания DVB-H и системы высокоскоростного цифрового мобильного ТВ-вещания DVB-H2.

Получены основные характеристики ТКС в зависимости от параметров систем, характеристик сигналов и влияния шумов и многолучевости (для CDMA). Представлены созвездия для модуляторов, спектры сигналов на входе и выходе канов связи, а также зависимости вероятности битовой ошибки от отношения сигнал/шум и многолучевости.

Материалы учебного пособия могут быть использованы как для учебных целей, так и как справочный материал при проектировании ТКС.

К учебному пособию прилагается CD-диск с программным обеспечением для

всех комплексов, включенных в пособие, а также пакетами MATLAB 2015b и NI LabVIEW со всем установочным ПО. Программные комплексы позволят читателю самостоятельно провести моделирование для ТКС со своими характеристиками.

## ЛИТЕРАТУРА

1. Банкет В.Л. Помехоустойчивое кодирование в телекоммуникационных системах: учебн. пособие. - Одесса: ОНАС им А.С. Попова, 2011. - 104 с.
2. Зюко А.Г., Фалько А.И., Панфилов И.П., Банкет В.Л., Иващенко П.В. Помехоустойчивость и эффективность систем передачи информации. М.: Радио и связь. 1985.
3. Методы повышения энергетической и спектральной эффективности цифровой радиосвязи: учеб. пособие / В. А. Варгаузин, И. А. Цикин. — СПб.: БХВ-Петербург, 2013. — 352 с.
4. Банкет В.Л. Сигнально-кодовые конструкции в телекоммуникационных системах. - Одесса: Фешкс, 2009. - 180 с.
5. Мелихов С.В. Аналоговое и цифровое радиовещание: Учебное пособие. Издание второе, исправленное. - Томск: Томск. гос. ун-т систем управления и радиоэлектроники, 2012. – 233 с.
6. Голиков А.М., Уваровский В.Д. Исследование многоуровневых методов модуляции сигналов, используемых в космических системах связи, на базе аппаратуры и ПО labVIEW 2010. Методические указания по лабораторным работам – Томск: Том. гос. ун-т систем управления и радиоэлектроники, 2011. – 50 с.

7. Галкин В.А. Цифровая мобильная радиосвязь. Учебное пособие для вузов. – М.: Горячая линия-Телеком, 2007. – 432 с..
8. Федосов В. П., Нестеренко А. К. Цифровая обработка сигналов в LabVIEW: учеб. пособие / под ред. В. П. Федосова. – М.: ДМК Пресс, 2007. – 456 с.
9. Теория и техника передачи информации : учебное пособие /Ю. П. Акулиничев, А. С. Бернгардт. — Томск: Эль Контент, 2012. — 210 с.
10. Скляр Б. Цифровая связь. — М.: Издательский дом Вильямс. 2003 — 1104с
11. Феер К.: Беспроводная цифровая связь. М.: Радио и связь, 2000. - 520 с.
12. Крейнделин В.Б., Колесников А.В. Оценивание параметров канала в системах связи с ортогональным частотным мультиплексированием. Учебное пособие / МТУСИ.-М., 2010. -29 с.
13. Д. Ватолин, М. Смирнов «Методы сжатия данных: Сжатие изображений»  
// [http://www.compression.ru/book/part2/part2\\_\\_3.htm](http://www.compression.ru/book/part2/part2__3.htm)
14. С. Уэлстид. “Фракталы и вейвлеты для сжатия изображений в действии”. Москва. “Издательство ТРИУМФ” 2003. 360 .
15. <https://sites.google.com/site/szatieinformacii/lekcii/tema13>
16. Дворкович В.П., Дворкович А.В. Цифровые видеоинформационные системы (теория и практика) Москва: техносфера, 2012. – 1008 с.
17. LabVIEW. Справочник по функциям. [Электронный ресурс]. – Режим доступа:[http://chaos.sgu.ru/library/programms/progr/labVIEW/LabVIEW\\_suranov.pdf](http://chaos.sgu.ru/library/programms/progr/labVIEW/LabVIEW_suranov.pdf)
18. Майков, Д.Ю. Оценка сдвига частоты для процедуры Initial Ranging в системе «мобильный WiMax» / Д.Ю. Майков, А.Я. Демидов, Н.А. Каратаева, Е.П. Ворошилин // Доклады ТУСУРа. – 2011. – №2 (24). – 59-63 с.
19. Серов А. В. Эфирное цифровое телевидение DVB-T/H. - БХВ-Петербург, 2010 – 465 с.
- 20 . Стандарт DVB-H. Система мобильного ТВ вещания. [Электронный ресурс] – Режим доступа: <http://www.konturm.ru/tech.php?id=dvvh>
- 21.[http://www.mathworks.com/examples/simulink-communications/mw/comm\\_product-LTEDownlinkExample-lte-phy-downlink-with-spatial-multiplexing](http://www.mathworks.com/examples/simulink-communications/mw/comm_product-LTEDownlinkExample-lte-phy-downlink-with-spatial-multiplexing)
22. J. H. Yuen, et. al. Modulation and Coding for Satellite and Space Communications. Proc. IEEE, vol. 78., n. 7, July, 1990, pp. 1250-1265.
23. Forney G. Concatenated Codes. Cambridge, Massachusetts: M. I. T. Press, 1966.