

МИНИСТЕРСТВО ОБРАЗОВАНИЯ И НАУКИ РФ
Федеральное государственное бюджетное образовательное учреждение
высшего профессионального образования
ТОМСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ СИСТЕМ
УПРАВЛЕНИЯ И РАДИОЭЛЕКТРОНИКИ (ТУСУР)

Кафедра радиотехнических систем (РТС)

Утверждаю:
Зав. каф. РТС, проф., д.т.н.
_____ Мелихов С.В.
_____ 2016 г.



Кологривов В.А., Чаплыгина А.А.

ИССЛЕДОВАНИЕ ПОМЕХОУСТОЙЧИВОСТИ
МНОГОКАНАЛЬНЫХ СИСТЕМ НА ОСНОВЕ PSK МОДУЛЯЦИИ
ПРИ НЕОРТОГОНАЛЬНОМ РАЗНЕСЕНИИ

Учебно-методическое пособие по лабораторной работе
для студентов направления
«Инфокоммуникационные технологии и системы связи»
по дисциплине
«Сети и системы мобильной связи»

Разработчики:
Доц. каф РТС Кологривов В.А. _____
Студент гр. 1В2 Чаплыгина А.А. _____

Томск 2016

Кологривов В. А., Чаплыгина А.А.

«Исследование помехоустойчивости многоканальных систем на основе PSK-модуляции при неортогональном разнесении»: Учебно-методическое пособие по лабораторной работе для студентов направления «Инфокоммуникационные технологии и системы связи» по дисциплине «Сети и системы мобильной связи». – Томск: ТУСУР. Научно-образовательный портал, 2016.– 38 с.

Учебно-методическое пособие содержит описание функциональной модели для исследования цифровых систем при неортогональном и ортогональном частотном разнесении, выполненной в среде функционального моделирования *Simulink*, системы для инженерных и научных расчетов *MatLab*.

В пособии приведены краткие теоретические сведения о способах частотного уплотнения каналов связи, краткая характеристика пакета *Simulink* системы *MatLab*, описание виртуального лабораторного макета и используемых блоков библиотеки *Simulink*.

Аннотация

Лабораторная работа **«Исследование помехоустойчивости многоканальных систем на основе PSK-модуляции при неортогональном разнесении»** посвящена экспериментальному исследованию частотного уплотнения каналов связи в системах передачи с использованием пакета функционального моделирования *Simulink*, системы для инженерных и научных расчетов *MatLab*.

Работа **«Исследование помехоустойчивости многоканальных систем на основе PSK-модуляции при неортогональном разнесении»** относится к циклу лабораторных работ, входящему в дисциплины по направлению **«Инфокоммуникационные технологии и системы связи»**.

В описании сформулирована цель лабораторной работы, приведены краткие теоретические сведения о частотном разнесении каналов в цифровых системах передачи и способах модуляции, краткая характеристика пакета *Simulink* системы *MatLab*, описание виртуального лабораторного макета и используемых блоков библиотеки *Simulink*, а также требования к экспериментальному исследованию и контрольные вопросы, ответы на которые необходимы для успешной защиты лабораторной работы.

Лабораторная работа рассчитана на два задания по 4 часа.

Содержание

1. Цель работы. Краткие сведения из теории	5
2. Краткое описание пакета Simulink	9
2.1. Общая характеристика пакета Simulink	9
2.2. Запуски работа с пакетом Simulink	9
3. Описание лабораторного макета	11
4. Описание используемых блоков библиотеки Simulink	20
5. Экспериментальное задание	31
6. Контрольные вопросы	36
Список использованных источников	38

1Цель работы. Краткие сведения из теории

Цель работы: Изучить принцип работы и особенности цифровых систем при неортогональном разнесении каналов связи.

1.1Теоретическая часть

Одним из методов повышения пропускной способности каналов связи является расширением полосы пропускания для каналов связи, при этом возникает проблема с электромагнитной совместимостью разнородных систем и ограниченность частотного диапазона. Решением данных проблем стало использование разнесения каналов связи при дискретном многочастотном кодировании передаваемых данных.

Разнесение каналов – это метод построения системы связи, обеспечивающий одновременную и параллельную передачу данных от множества отправителей к такому же числу получателей. Существуют такие виды разнесения каналов связи:

1. частотное, когда каждому каналу выделяется определённая частота в общей широкой полосе пропускания линии связи;
2. временное, когда каналы связи включаются в определённый интервал времени в линию связи;
3. кодовое, когда каналы связи закодированы определённой последовательностью.

При этом частотное разнесение каналов может осуществляется при помощи ортогонального или неортогонального набора несущих частот.

Условием ортогональности несущих частот является:

Данные выражение путем преобразование принимает вид:

Если несущие частоты удовлетворяют данному условию, то они являются ортогональными частотами. Тогда разнос несущих частот составляет Δf – для некогерентных систем и $\Delta f/2$ — для когерентных систем. Напомним, что когерентными называются системы, у которых на приемной стороне известна и используется информация о фазе несущего колебания. В последнее время разнос равный $\Delta f/2$ — стали соотносить с неортогональным разносом поднесущих или несущих [1].

Одна из стратегий использования систем со многими несущими следующая - исходная последовательность данных распределяется между всеми несущими частотами каналов, и их передача осуществляется параллельно и одновременно. Это позволяет увеличить скорость передачи или осуществить передачу высокоскоростного потока через набор параллельных ортогональных низкоскоростных каналов передачи, что позволяет увеличить помехоустойчивость приёма [1]. При этом ширина полосы канала должна быть достаточно широкой для обеспечения заданной скорости, но и достаточно узкой для уменьшения уровня помех в канале.

Для экономии общей занимаемой полосы пропускания линии связи используется неортогональное разнесение каналов. При этом несущие частоты каналов не удовлетворяют условиям ортогональности и имеют меньший разнос, чем ортогональные. Это обеспечивает наиболее экономное использование полосы пропускания системы, но при этом возникает наложение спектров каналов, при котором однако еще возможно разделение данных определенного канала на приемном конце.

Как при ортогональном, так и при неортогональном частотном разнесении при мультиплексировании разных несущих частот возникают «биения» или «пик-фактор». «Пик-фактор» это превышение мгновенной мощности сигнала над средней мощностью. Наличие «пик-фактора» ставит серьезные проблемы снижению нелинейных искажений в радиоканалах из-за усложнения схемотехнических решений усилителей мощности с высоким КПД.

Так же при реализации системы с разнесением каналов связи по частоте важно выбрать канальную модуляцию. Выбор вида канальной модуляции выбирается из компромисса между пропускной способностью системы передачи и достоверностью полученных данных. Существуют разные методы цифровой модуляции, но большое распространение получили такие виды цифровой модуляции как: **BPSK**, **M-PSK**, где **M** может принимать значения от 4 до 256, большее значением не используется на практике. С увеличением позиционности модуляции **M** возрастает спектральная эффективность, то есть увеличивается способность передачи информации с большей скоростью в той же полосе частот. При переходе от **BPSK** сигналов к **QPSK** скорость передачи возрастает в 2 раза при неизменной занимаемой полосе частот. Но при увеличении позиционности модуляции требуется увеличение отношения сигнал/шум (**SNR**) для сохранения достоверного приема данных. Отношение сигнал/шум **SNR** является основной характеристикой определяющей качество приема данных, и оно влияет на энергетическую эффективность модуляции. Отношение сигнал/шум на входе системы определяется из выражения:

$$S/N = \dots$$

где S - мощность сигнала;

N -мощность шумов;

- энергия бита;
- спектральная мощность шума;
- занимаемая полоса пропускания;
- длительность бита;
- битовая скорость передачи.

Выражение для определения энергетической эффективности модуляции примет вид:

— — —

В нашем исследовании будем использовать **BPSK** модуляцию, у которой и **QPSK**, у которой , при этом водопадоподобная характеристика энергетической эффективности модуляции одинаковая. На рисунке 1.1 представлены характеристики энергетической эффективности **PSK**-модуляций, в том числе для **BPSK** и **QPSK**.

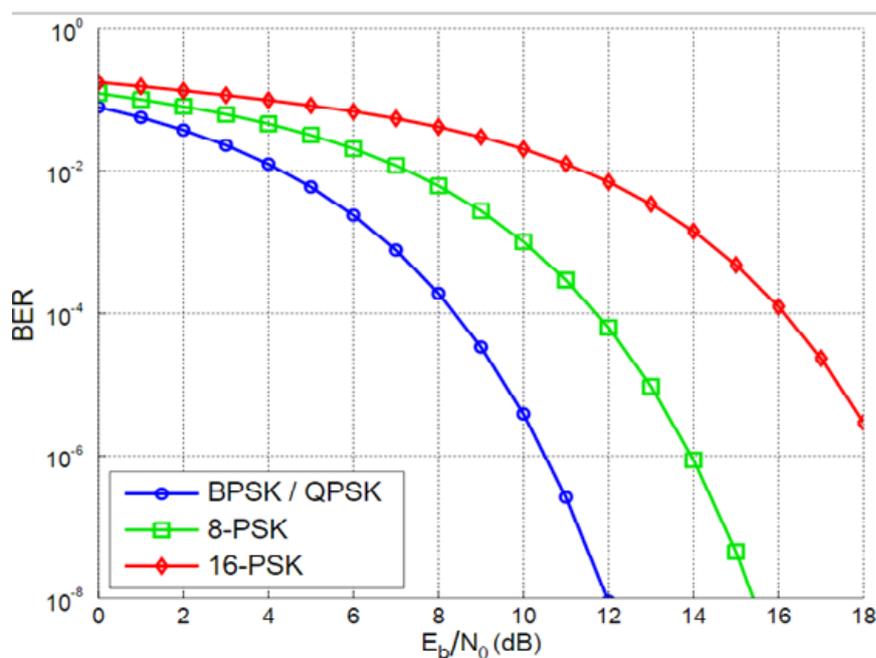


Рисунок 1.1 - Характеристика энергетической эффективности некоторых видов **PSK**-модуляций

У **QPSK**-модуляции требуемая полоса пропускания в 2 раза меньше чем у **BPSK**-модуляции из-за того что символ состоит из двух бит (дибит).

2 Краткое описание пакета **Simulink**

2.1 Общая характеристика пакета **Simulink**

Пакет **Simulink** распространяется в составе математического пакета **MatLab**. Пакет основан на графическом интерфейсе и является типичным средством визуально-ориентированного программирования. Пакет **Simulink** обладает обширной библиотекой готовых блоков с модифицируемыми параметрами для построения моделей рассматриваемых систем и наглядными средствами визуализации результатов моделирования [2, 3, 4].

2.2 Запуск и работа с пакетом **Simulink**

Для запуска системы **Simulink** необходимо выполнить запуск системы **MatLab**. После открытия командного окна системы **MatLab** нужно запустить систему **Simulink**. Существует три способа запуска системы **Simulink**:

- нажать кнопку  (**Simulink**) на панели инструментов системы **MatLab**;
- в строке командного окна **MatLab** напечатать **Simulink** и нажать клавишу **Enter**;
- выполнить опцию **Open** в меню **File** и открыть файл модели (**mdl**-файл).

При применении двух первых способов открывается окно обозревателя библиотеки блоков (**SimulinkLibraryBrowser**). Если нам не требуется добавление новых блоков, а нужно лишь открыть уже готовую модель и провести моделирование, то следует воспользоваться третьим способом.

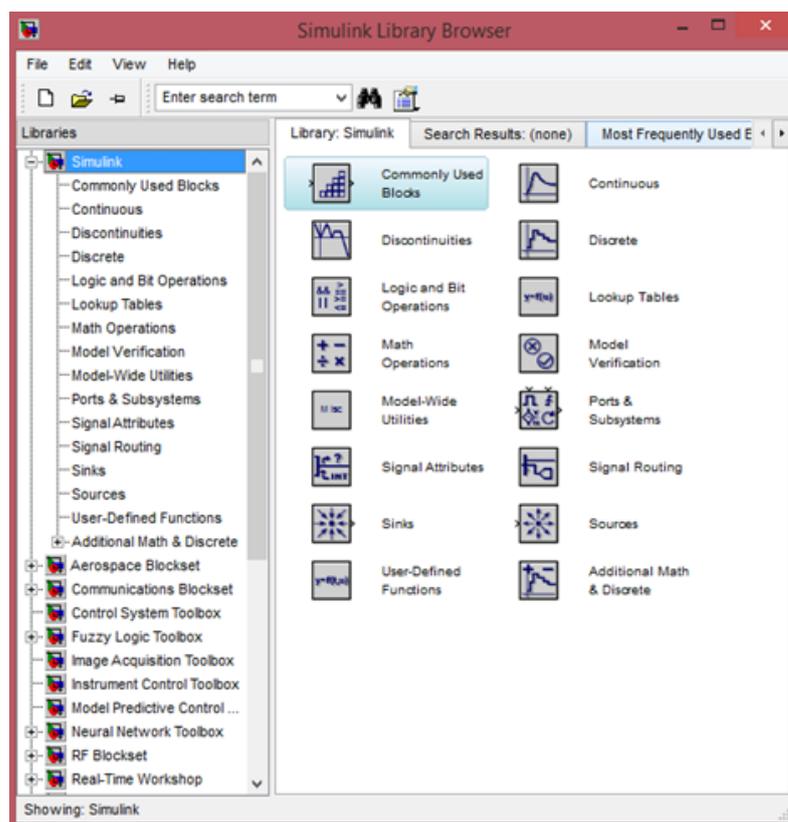


Рисунок 2.1 – Библиотека блоков **Simulink Library Browser**

На рисунке 2.1 выведена библиотека системы **Simulink** и показаны ее разделы. Основная библиотека системы содержит следующие разделы:

- **Continuous** – блоки аналоговых элементов;
- **Discontinuous** – блоки нелинейных элементов;
- **Discrete** – блоки дискретных элементов;
- **Look-Up Tables** – блок таблиц;
- **Math Operations** – блоки элементов, определяющие математические операции;
- **Model Verification** – блоки проверки свойств сигнала;
- **Model-Wide Utilities** – раздел дополнительных утилит;
- **Port & Subsystems** – порты и подсистемы;
- **Signal Attributes** – блоки задания свойств сигналов;
- **Signal Routing** – блоки маршрутизации сигналов;
- **Sinks** – блоки приема и отображения сигналов;

- **Sources** – блоки источников сигнала;
- **User-DefinedFunction** – функции, определяемые пользователем.

Правила работы со списком разделов библиотеки: пиктограмма свернутого узла содержит символ «+», а пиктограмма развернутого – символ «-».

Для того чтобы развернуть или свернуть узел, достаточно щелкнуть на его пиктограмме левой клавишей мыши.

При работе элементы разделов библиотек "**перетаскивают**" в рабочую область удержанием *ЛКМ* на соответствующих изображениях. Для соединения элементов достаточно указать курсором мыши на начало соединения и затем при нажатии левой кнопки мыши протянуть соединение в его конец.

При двойном щелчке *ЛКМ* на выделенном блоке всплывает меню, в котором задаются параметры блоков.

Работа **Simulink** происходит на фоне открытого окна системы **MatLab**, закрытие которого приведёт к выходу из **Simulink**.

3 Описание лабораторного макета

3.1 BPSK-модем

Принцип работы функциональной модели двухканального модема при **BPSK**-модуляции:

Передающая часть. Randomnumber- формирует случайный процесс с нормальным распределением уровня, который поступая на блок **Sign**, преобразуется в биполярную псевдослучайную информационную последовательность с нормальным распределением. Далее, сформированный таким образом цифровой сигнал попадает в блок **Product**,

где происходит перемножение текущих значений сигнала с синусоидальным колебанием несущей частоты, который формируется блоком **SineWave**. В процессе модельного исследования разнос частот несущих колебаний устанавливался равным либо. После модуляции несущих псевдослучайными информационными последовательностями каналов происходит суммирование сигналов каналов (блок **Sum**). Здесь же добавляются широкополосные шумы канала распространения радиосигнала.

Приемная часть. На приемной стороне сигнал поступает на блоки канальных полосовых фильтров (**AnalogFilterDesign**), настроенных на несущие частоты с полосами пропускания порядка . Канальные фильтры производят предварительное выделение радиосигнала своего канала, включая шумы канала распространения и частично подавленный сигнал второго канала. Далее предварительно отфильтрованный сигнал и помехи подаются на блок **Product**, где с целью демодуляции происходит перемножение полученного сигнала с колебаниями опорного генератора несущей частоты канала. После этого сигнал фильтруется в блоке **ФНЧ** (**AnalogFilterDesign**) с целью подавления высокочастотных составляющих на выходе умножителя-преобразователя. В процессе модельного исследования частоты среза ФНЧ принимались равными либо. После **ФНЧ** отфильтрованный сигнал подается на блок **Zero-OrderHold**. В этом блоке происходит принятие решения, поскольку блок фиксирует значение входного сигнала в начале интервала дискретизации и удерживает его до конца интервала. Далее блок **Sign** завершает процесс регенерации формы принятого сигнала – псевдослучайную биполярную последовательность прямоугольных импульсов с нормальным распределением. Для визуализации принятой и переданной последовательностей используется блок **Scope**, выполняющий функцию многолучевого осциллографа.

Детектор ошибок. Для отслеживания ошибок при приеме используем детектор ошибок - блок **Subsystem**, который подсчитывает и отображает на

блоке **Display** количество битовых ошибок. В детектор ошибок поступает задержанная на один такт исходная псевдослучайная информационная последовательность и принятый демодулированный сигнал с выхода приемного тракта, где вычисляется их разность во времени и интегрируется. Функциональная схема детектора ошибок представлена на рисунке 3.1.

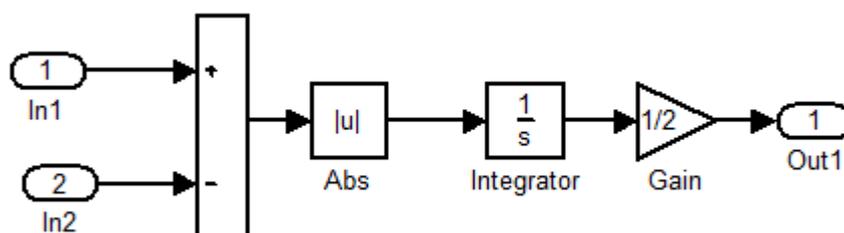


Рисунок 3.1 - Функциональная схема детектора ошибок

На вход 1 поступает полученный сигнал, на вход 2 поступает исходный сигнал, задержанный сигнала на один такт. В сумматоре (**Sum**) вычисляется разница текущих значений входных сигналов и модуль разности (**Abs**) исходной и принятой последовательности. Модуль разности подается на интегратор (**Integrator**) для накопления информации об ошибках. Блок **Gain** используется для коррекции значения ошибок при использовании последовательностей с разной длиной и/или полярностью импульсов. Для отображения числа ошибок с выхода детектора информация поступает на блок **Display**.

Измеритель мощности. Для проведения оценки помехоустойчивости системы используем измеритель мощности – блок **Subsystem**, функциональная схема, которого приведена ниже на рисунке 3.2.

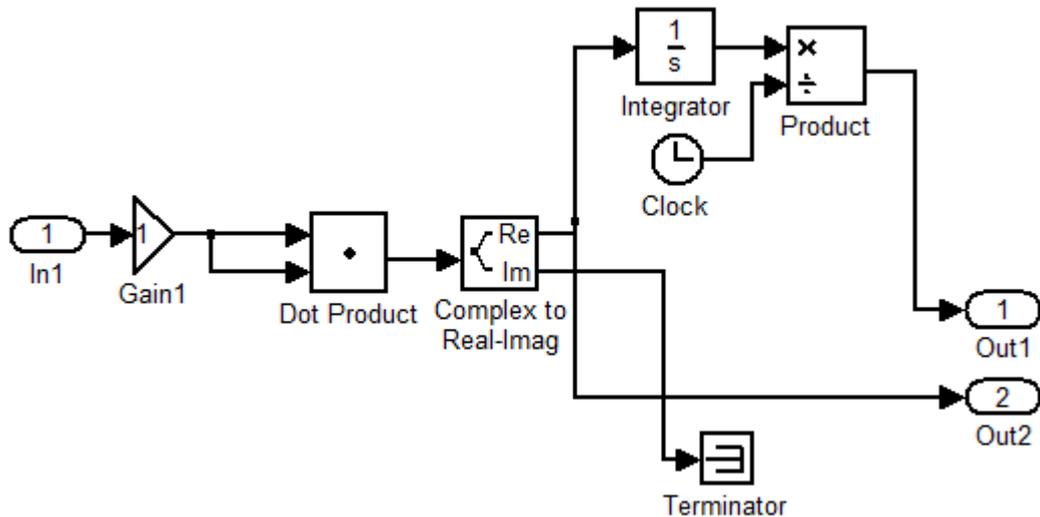


Рисунок 3.2 - Функциональная схема измерителя мощности

На вход 1 поступает полученный сигнал после фильтрации, поступает на **Gain** используемый как масштабный множитель. Далее измеряемый сигнал поступает на блок **DotProduct**, где вычисляется скалярное произведение комплексно сопряженных последовательностей. Если входная последовательность является вещественной, то выходной сигнал будет действительным. Если же входная последовательностей будет иметь комплексный вид, то на выходе будет комплексный сигнал. После сигнал поступает на блок **ComplextoReal-Image**, который вычисляет действительную и мнимую части комплексного числа. Дальше мнимую часть глушим в блоке **Terminator**, а действительная часть числа поступает на выход 2 и на вход осциллографа, а также поступает на интегратор, где накапливается энергия входного процесса, как интеграл от квадрата входной функции. После интегрирования сигнал поступает на блок **Product**, где происходит деление время, т.е. вычисление значения текущей мощности входного процесса. Далее информация о мощности поступает на выход 1 и на блок **Scope** для отображения.

Общая функциональная схема двухчастотного модема с **BPSK**-модуляцией представлена на рисунке 3.3.

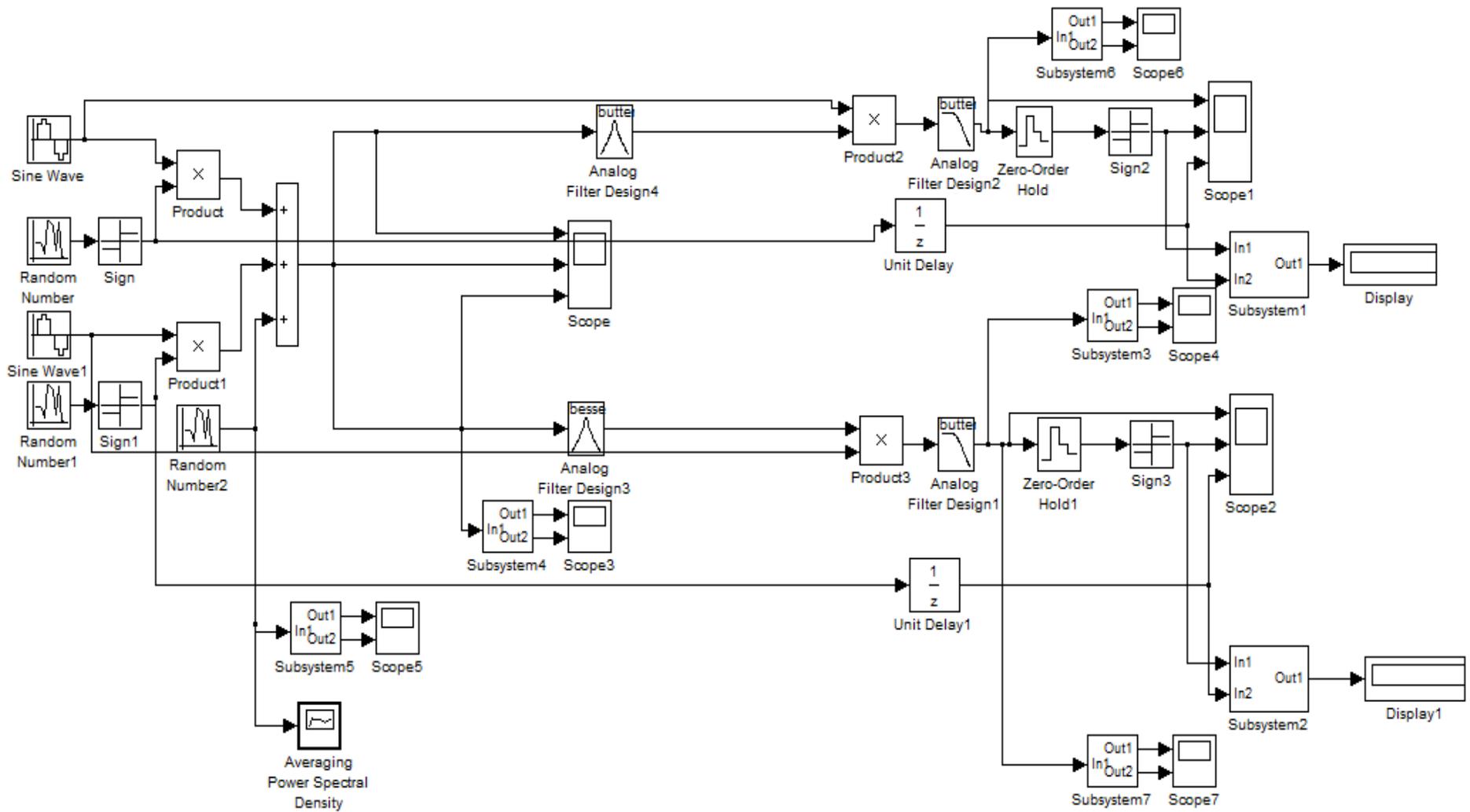


Рисунок 3.3 - Функциональная схема двухчастотного модема с **BPSK**-модуляцией

3.2 QPSK-модем

Рассмотрим принцип работы и структуру двухчастотного модема с **QPSK**-модуляцией.

Передатчик. Randomnumber формирует случайный процесс с нормальным распределением уровня сигнала, который поступая на блок **Sign** преобразуется в биполярную псевдослучайную информационную последовательность с нормальным распределением уровня. Далее последовательность поступает на блок **ComparetoZero**, который производит преобразование биполярной псевдослучайной последовательности в однополярную псевдослучайную последовательность. Для преобразования типа данных **boolean** в тип **double** используется блок **DataTypeconvert**.

Далее следуют блоки фазового кодера, устанавливающего соответствие текущего дибита и фазового состояния несущей (рисунок 3.4).

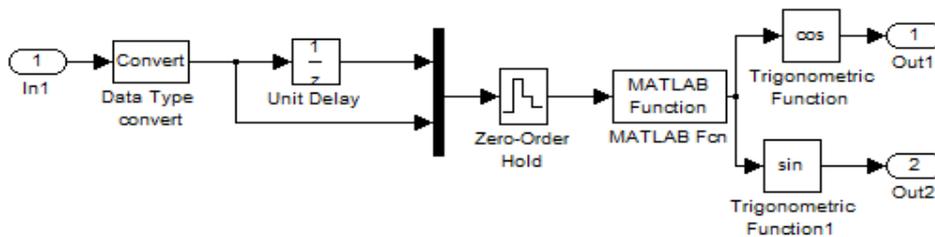


Рисунок 3.4 – Функциональная схема фазового кодера **QPSK**-модулятора

Здесь, на вход мультиплексора **Mux** поступает исходная последовательность и последовательность задержанная во времени на один такт. Мультиплексор объединяет предыдущий и текущий биты (дибит) в вектор. Для расширения во времени компонент вектора до длительности дибита и исключения влияния промежуточных состояний компонент вектора используется блок **Zero-OrderHold**, который на протяжении времени дибита фиксирует его значения и подает на блок фазового кодирования (**MatLabFcn**- код функции представлен на рисунке 3.5). Блок **MatLabFcn** содержит **MatLab**-функцию, которая каждому

состоянию дибита ставит в соответствие значение фазы . Далее блоки **TrigonometricFunction** вычисляют значения $\cos(\varphi)$ и $\sin(\varphi)$, которые являются входными модулирующими сигналами квадратурного модулятора (блоки **Product** и **Sum** см. рисунок 3.6).

```

1 function fi=dibit_phase(x);
2 % преобразование биполярного дибита в фазу
3 % X- вектор (дибит)
4 % fi - фаза
5
6
7 if (x==[0;0]); fi=pi/4;end;
8 if (x==[0;1]); fi=3*pi/4;end;
9 if (x==[1;1]); fi=-3*pi/4;end;
10 if (x==[1;0]); fi=-pi/4;end;

```

Рисунок 3.5 – Код функции dibit_phase

Квадратурный модулятор включает в себя два преобразователя (блоки **Product**) на первые входы которых поступают квадратурные колебания опорной частоты с генераторов гармонических сигналов (блоки **SineWave**). На вторые входы преобразователей подаются сигналы с выхода фазового кодера (см. рисунок 3.4). На выходе преобразователей сигналы суммируются блоком **Sum** и подаются в канал распространения радиосигнала.

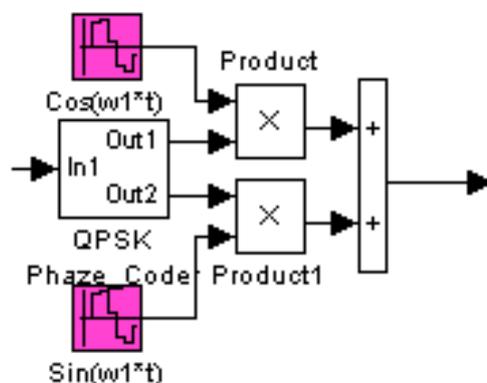


Рисунок 3.6 – Функциональная схема **QPSK**-модулятора

Канал распространения представлен сумматором **Sum** с тремя входами. На первые два входа подаются модулированные сигналы на разных несущих, а на

третий вход подается широкополосный псевдослучайный процесс с блока **Randomnumber**.

Приемник. На приемном конце принятые радиосигналы и шум канала распространения разветвляются на два канала и если необходимо фильтруются канальными фильтрами (блоки **AnalogFilterDesign**), настроенными на несущие частоты с полосами пропускания. Канальные фильтры производят предварительное выделение радиосигнала своего канала, включая шумы канала распространения и частично подавленный сигнал второго канала. Далее предварительно отфильтрованные радиосигналы и помехи подаются на преобразователи демодулятора (блоки **Product**). На вторые входы преобразователей поступают колебания с соответствующих опорных колебаний несущих частот (в функциональной модели для простоты опорные генераторы передатчика и приемника совмещены). С выходов умножителей сигналы подаются на **ФНЧ** для удаления высокочастотных продуктов преобразования. Далее выполняется процедура взятия отсчетов принятых квадратурных компонент фазовых состояний (блоки **Zero-OrderHold**) и объединения их в вектор принятых состояний блоком **Mux**. Функциональная схема демодулятора представлена на рисунке 3.7.

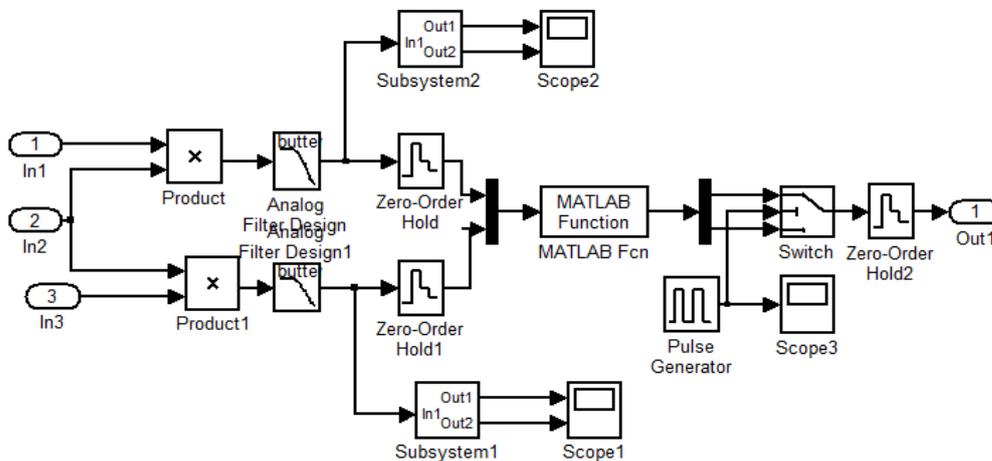
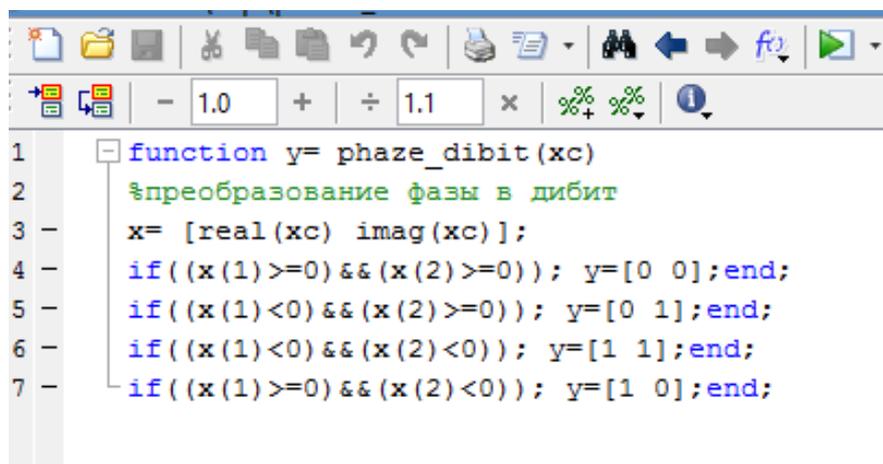


Рисунок 3.7 – Функциональная схема **QPSK**-демодулятора со схемой регенерации формы битов

Значения вектора принятых квадратурных проекций поступают на вход блока **MatLabFcn**, содержащего **MatLab**-функцию преобразования фазовых проекций в вектор переданных битов (дигит), код функции представлен на рисунке 3.8. Далее компоненты векторов битов разделяются блоком **DeMux** и подаются на преобразователь параллельного потока в последовательный на основе двухпортового ключа **Switch**, управляемого импульсами блока

PulseGenerator. Принятый поток битов для контроля временного положения пропускается через блок **Zero-OrderHold** и отображается на осциллограмме (блок **Scope**).

A screenshot of a MATLAB code editor window. The window title is partially visible as '...'. The toolbar at the top includes icons for file operations (save, open, copy, paste), editing (undo, redo), and execution (run, stop). Below the toolbar is a numeric keypad with values 1.0, 1.1, and mathematical symbols like minus, plus, multiply, divide, and percent. The main area contains the following MATLAB code:

```
1 function y= phase_dibit(xc)
2     %преобразование фазы в дибит
3     x= [real(xc) imag(xc)];
4     if((x(1)>=0)&&(x(2)>=0)); y=[0 0];end;
5     if((x(1)<0)&&(x(2)>=0)); y=[0 1];end;
6     if((x(1)<0)&&(x(2)<0)); y=[1 1];end;
7     if((x(1)>=0)&&(x(2)<0)); y=[1 0];end;
```

Рисунок 3.8 – Код функции phase_dibit

В процессе модельных исследований, как и в модеме с **BPSK**-модуляцией используется детектор ошибок и измеритель мощности для отслеживания количества битовых ошибок при приеме и контроля соотношения сигнал/шум. Структурная схема модема с **QPSK**-модуляцией представлена на рисунке 3.9

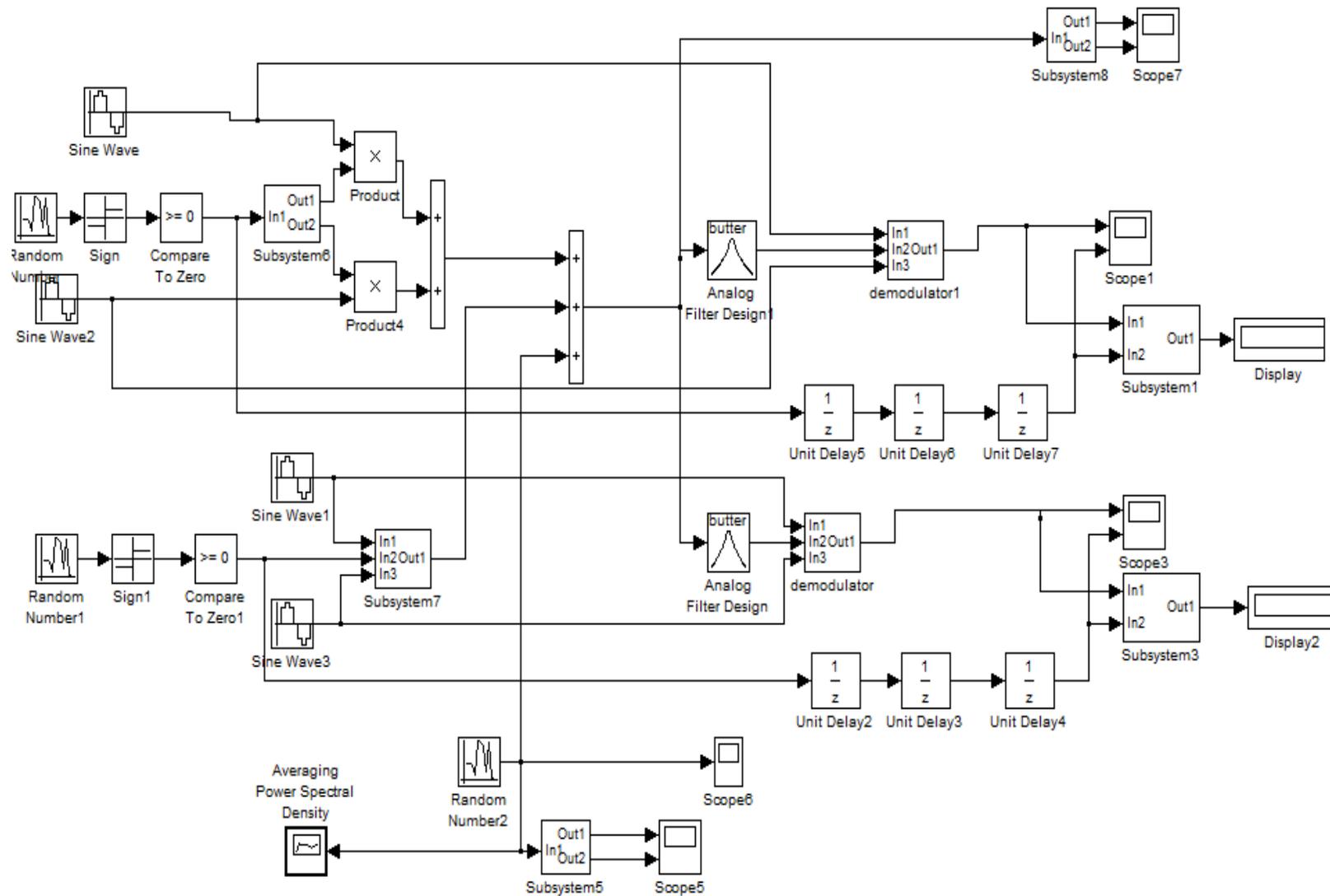


Рисунок 3.7– Функциональная схема двухчастотного модема с QPSK-модуляцией

4 Описание используемых блоков библиотеки Simulink

Ниже описаны основные блоки базовых разделов библиотеки Simulink[2], используемые в функциональной схеме двухканального модема при BFSK модуляции.



RandomNumber – источник случайного сигнала с нормальным распределением. *Назначение:* формирование случайного сигнала с равномерным распределением уровня сигнала. *Параметры блока:* **Mean** – среднее значение сигнала; **Variance** – дисперсия; **Initialseed** – начальное значение генератора случайного сигнала; **Sampletime** – такт дискретности.



Sign – блок определения знака сигнала. *Назначение:* определяет знак входного сигнала, при этом, если X - входной сигнал, то сигнал на выходе определяется выражением

Параметры блока: флажок - **Enablezerocrossingdetection** позволяет фиксировать прохождение сигнала через нулевой уровень.



Switch – блок переключателя. *Назначение:* переключение входных сигналов по сигналу управления. *Параметры блока:* **Criteriaforpassingfirstinput** – условие прохождения сигнала с первого входа, значение выбирается из списка: **u2>=Threshold** – сигнал управления

больше или равен пороговому значению; **u2>Threshold** – сигнал управления больше порогового значения; **u2~=Threshold** – сигнал управления не равен пороговому значению. **Threshold** – порог; флажок **Showadditionalparameters** – показать дополнительные параметры.



Scope

Scope – блок осциллографа. *Назначение:* построение графиков исследуемых сигналов как функций времени. Открытие окна осциллографа производится двойным щелчком ЛКМ на пиктограмме блока. Настройка окна осциллографа выполняется с помощью панелей инструментов, позволяющих: осуществить печать содержимого окна осциллографа; установить *параметры*, в частности, **Numberofaxes** - число входов осциллографа, **Timerange** – отображаемый временной интервал и другие; изменить масштабы графиков; установить и сохранить настройки; перевести в плавающий режим и так далее.



Sine Wave

SineWave – блок источника синусоидального сигнала. *Назначение:* формирование синусоидального сигнала с заданной частотой, амплитудой, фазой и смещением. Для формирования выходного сигнала блоком могут использоваться два алгоритма. Вид алгоритма определяется параметром **SineType** – способ формирования сигнала реализуется двумя алгоритмами: **Time-based** – по текущему времени (для аналоговых систем) или по значению сигнала на предыдущем шаге и величине такта дискретности (для дискретных систем); **Sample-based** – по величине такта дискретности и количеству расчетных шагов на один период синусоидального сигнала. Вид окна задания параметров меняется в

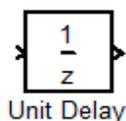
зависимости от выбранного способа формирования синусоидального сигнала.

*Параметры блока в режиме **Time-based:Amplitude***– амплитуда; **Bias**– постоянная составляющего сигнала (смещение); **Frequency (rads/sec)** – частота (рад/с); **Phase (rads)** – начальная фаза (рад); **Sampletime** – такт дискретности. Параметр может принимать следующие значения: **0** (по умолчанию) – используется при моделировании непрерывных систем; **> 0** (положительное значение) – задается при моделировании дискретных систем; **-1** (минус один) – такт дискретности устанавливается таким же, как и в предшествующем блоке. Флажок **Interpretedvectorparametersas 1 - D** – интерпретировать вектор как массив скаляров. Для очень больших значений времени точность вычисления значений сигнала падает.

*Параметры блока в режиме **Sample-based:Amplitude***– амплитуда; **Bias**– постоянная составляющего сигнала (смещение); **Samplesperperiod** – количество тактов на один период синусоидального сигнала:

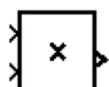
—

Numberoffsetsamples – начальная фаза сигнала, задается количеством тактов дискретности $l = \varphi \cdot p / (2 \cdot \pi)$. **Sampletime** – такт дискретности. Флажок **Interpretedvectorparametersas 1 - D** – интерпретировать вектор как одномерный. В данном режиме ошибка округления не накапливается, поскольку **Simulink**начинает отсчет номера текущего шага с нуля для каждого периода.



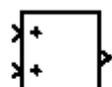
Unit Delay

Unitdelay – блок единичной дискретной задержки. *Назначение:* выполняет задержку дискретного сигнала на заданный такт дискретности. *Параметры блока:* **Initialconditions** – начальное значение выходного сигнала; **Sampletime** – такт дискретности (при задании значения параметра равного **-1** такт дискретности наследуется от предшествующего блока).



Product

Product – блок умножения и деления. *Назначение:* вычисление произведения текущих значений сигналов. *Параметры блока:* **Numberofinputs** – количество входов, может задаваться как число или как список знаков. В списке знаков можно использовать знаки: * - умножить и / - разделить. **Multiplication**– способ выполнения операции, может принимать значения из списка: **Element-wise** – поэлементный; **Matrix**– матричный. Флажок **Showadditionalparameters** – показать дополнительные параметры. При выставленном флажке отображается окно списка **Outputdatatypemode**, в нашем случае флажок не используется.



Sum of Elements

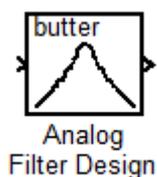
Sum– блок сумматора. *Назначение:* вычисление алгебраической суммы текущих значений входных сигналов. *Параметры блока:* **Iconshape** – форма блока, выбирается из списка: **round**– круг; **rectangular**– прямоугольник. **Listofsign** – список знаков из набора: + - плюс; - - минус, | - делитель. Флажок **Showadditionalparameters** – показать дополнительные параметры, при выставленном флажке отображаются окна списка **Outputdatatypemode**, в нашем случае не используется. Количество входов и соответствующие им операции определяются списком знаков

Listofsign. При этом метки входов обозначаются соответствующими знаками. В списке **Listofsign** можно также указать число входов, при этом все входы будут суммирующими.



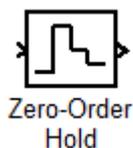
Gain – блок усилителя. *Назначение:* выполняет умножение входного сигнала на постоянный коэффициент; *Параметры блока:* **Gain** – коэффициент усиления. **Multiplication** – способ выполнения операции, значение параметра выбирается из списка: **Element-wise** $K*u$ – поэлементный; **Matrix** $K*u$ – матричный, коэффициент усиления является левосторонним оператором; **Matrix** $u*K$ – матричный, коэффициент усиления является правосторонним оператором; **Matrix** $K*u$ (**u-вектор**) – векторный, коэффициент усиления является левосторонним оператором. Флажок **Showadditionalparameters** – показать дополнительные параметры, при выставленном флажке отображаются окна списков **Parameterdatatypemode**, **Outputdatatypemode**. **Saturateoninteger** – подавлять переполнение целого. При установленном флажке ограничение сигналов целого типа выполняется корректно.

Блоки **Gain** и **MatrixGain** по сути есть один и тот же блок, но с разными начальными установками параметра **Multiplication**.

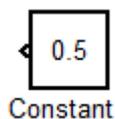


AnalogFilterDesign – блок аналогового фильтра заданного метода проектирования и типа из подраздела **FilterDesign**. *Назначение:* аналоговая фильтрация низкочастотных составляющих спектра входного сигнала. *Параметры блока:* **Designmethod** – метод проектирования, выбирается из списка: **Butterworth** – фильтр Баттерворта; **ChebyshevI** –

фильтр Чебышева 1-го рода; **ChebyshevII** – фильтр Чебышева 2-го рода; **Elliptic**– фильтр эллиптический; **Bessel**– фильтр Бесселя. **Filtertype** – тип фильтра, выбирается из списка: **Lowpass**– нижних частот; **Highpass**– верхних частот; **Bandpass**– полосно-пропускающий; **Bandstop**– полосно-заграждающий. Далее для каждого метода проектирования и типа фильтра выдается свой список параметров. Так для фильтра Баттерворта типа нижних частот параметрами являются: **Filterorder** – порядок фильтра; **Passbandedgefrequency (rads/sec)** – нижняя граничная частота (радиан в секунду). Для других методик проектирования и типов фильтров определяемые параметры очевидны.

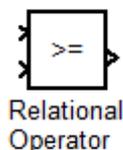


Zero-OrderHold – экстраполятор нулевого порядка. *Назначение:* экстраполяция входного сигнала на интервале дискретизации. Блок фиксирует значение входного сигнала в начале интервала дискретизации и поддерживает на выходе это значение до окончания интервала дискретизации. Затем выходной сигнал изменяется скачком до величины входного сигнала на следующем шаге дискретизации. *Параметры блока:* **Sampletime** – такт дискретности. Блок экстраполятора нулевого порядка может использоваться также для согласования работы дискретных блоков, имеющих разные такты дискретности.

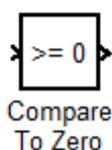


Constant – блок источника постоянного сигнала. *Назначение:* задает постоянный по уровню сигнал. *Параметры блока:* **Constantvalue**– постоянная величина. **Interpretvectorparametersas 1-D** – интерпретировать вектор как массив скаляров. **Showadditionalparameters** – показать дополнительные параметры. При выставленном флажке появится окно списка **Outputdatatypemode**. **Outputdatatypemode**– выбор типы выходных

данных. **Outputdatatype** – тип выходных данных. **OutputScalingMode** – способ масштабирования выходного сигнала. **Outputscalingvalue** – величина масштаба.



RelationalOperator – блок выполнения операций отношения. *Назначение:* блок сравнивает текущие значения входных сигналов. *Параметры блока:* **RelationalOperator** – тип операции отношения, выбирается из списка. **Showadditionalparameters** – показать дополнительные параметры. **Requireallinputstohavesamedatatype** – все входы должны иметь одинаковый тип данных. **Outputdatatypemode** – выбор типа выходных данных. **Outputdatatype** – тип выходных данных. **Enablezerocrossingdetection** – фиксировать прохождение сигнала через нулевой уровень.

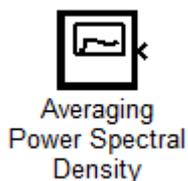


CompareToZero – блок сравнения с нулем. *Назначение:* сравнить с нулем заданный сигнал. *Параметры блока:* **Operation** – операции сравнения. Выбираются из списка. **Outputdatatypemode** – выбор типа выходных данных.

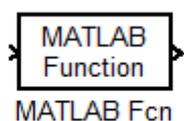


Demux – демультиплексор. *Назначение:* разделяет входной векторный сигнал на отдельные составляющие. *Параметры блока:* **NumberOfOutputs** – количество выходов. **Displayoption** – способ отображения, выбирается из списка: **bar** – вертикальный узкий прямоугольник черного цвета; **none** – прямоугольник с белым фоном без отображения меток

ВХОДНЫХ СИГНАЛОВ. **BusSelectionMode**-режим разделения векторных сигналов по шине.



AveragingPowerSpectralDensity-анализатор усредненной спектральной плотности мощности. *Назначение:* отобразить частотную зависимость спектральной плотности мощности. *Параметры блока:* **Lengthofbuffer**—длина буфера— **Numberofpointsforfft**—количество точек. **Plotafterhowmanypoints**—количество точек по которому выводится график. **Sampletime**—такт дискретности.

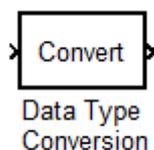


MATLABFunction — блок задания М-функции. *Назначение:* задает выражение в стиле языка программирования MATLAB. *Параметры блока:* **MATLABFunction**—выражение на языке MATLAB. **Outputdimensions** — размерность выходного сигнала. **Outputsignaltypе**— тип выходного сигнала. Выбирается из списка: **real**-действительный сигнал; **complex**- комплексный сигнал; **auto**-автоматическое определение типа сигнала. **Collapse 2-Dresultsto 1-D**-преобразование двумерного выходного сигнала в одномерный.



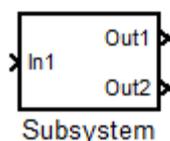
Display — цифровой дисплей. *Назначение:* отображает значение сигнала в виде числа. *Параметры блока:* **Format** — формат отображения данных, может принимать следующие значения: **short**— 5 цифр, включая десятичную точку; **long**- 15 цифр с фиксированной точкой; **short _e**-

5цифр с плавающей точкой; **long_e**-15цифр с плавающей точкой; **bank**– банковский формат.**Decimation** – прореживание. **Sampletime** – такт дискретности.**Floatingdisplay**– изменяющийся режим.



Data Type Conversion –преобразователь типа сигнала.

*Назначение:*преобразует тип входного сигнала. *Параметры блока:***Data Type**-тип данных выходного сигнала.**Saturateonintegeroverflow**-подавлять переполнение целого.



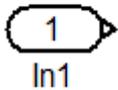
Subsystem – создание подсистем. *Назначение:*Подсистема-

это фрагмент Simulink-модели, оформленный в виде отдельного блока. Использование подсистем при составлении модели преследует следующие цели:

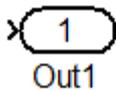
- Уменьшает количество одновременно отображаемых на экране блоков, что облегчает восприятие модели
- Позволяет создавать и отлаживать отдельные фрагменты модели, что повышает технологичность создания модели
- Позволяет создавать собственные библиотеки
- Позволяет синхронизировать параллельно работающие подсистема
- Позволяет включить в модель собственные справочные средства
- Позволяет связывать подсистему с М-файлом, обеспечивая запуск этого файла при открытии подсистемы

*Параметры блока:***Showportlabels** – показать метки портов.**Treatasatomicunit** – считать подсистему неделимой.**Read/WritePermissions** – разрешить чтение и запись. Допустимы

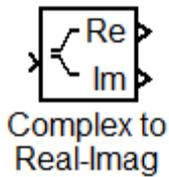
три опции: **ReadWrite**-чтение и запись; **ReadOnly**-только чтение; **NoReadOrWrite** ни чтение, ни записи. **Nameoferrorcallbackfunction** – имя функции ответного вызова.



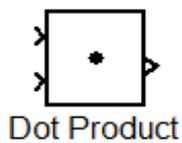
Import – входной порт. *Назначение:* Создает входной порт для подсистемы или выполняет считывание сигнала с рабочей области MATLAB в модель. *Параметры блока:* **Portnumber**-номер порта. **Portdimensions**-размерность входного сигнала. **Sampletime** – такт дискретности. **Showadditionalparameters**-показать дополнительные параметры. **Datatype** – выбор типа выходных данных. **Outputdatatype**-тип выходных данных. **OutputScalingMode** – способ масштабирования выходного сигнала. **Outputscalingvalue** – величина масштаба. **Samplingmode**-режим.



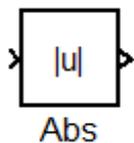
Outputport – выходной порт. *Назначение:* Создает входной порт для подсистемы или для модели верхнего уровня иерархии. *Параметры блока:* **Portnumber**-номер порта. **Outputwhendisabled**-вид сигнала на выходе подсистемы, в случае если подсистема выключена. Используется для подсистем, управляемых внешним сигналом. Может принимать следующие значения: **held**- выходной сигнал подсистемы равен последнему рассчитанному значению; **reset**- выходной сигнал подсистемы равен значению, задаваемому параметром **initialoutput**. **Initialoutput**-начальное значение.



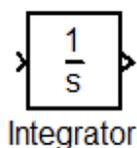
Complex to Real-Imag – блок вычисления действительной и (или) мнимой части комплексного числа. *Назначение:* вычисляет действительную и (или) мнимую часть комплексного числа. *Параметры блока:* **Output** – выходной сигнал. Тип сигнала выбирается из списка: **Real** – действительная часть; **Image** – мнимая часть; **Real&Image** – действительная и мнимая часть.



Dot Product – блок скалярного произведения. *Назначение:* Выполняет вычисление скалярного произведения двух векторов. *Параметры блока:* нет.



Abs – блок вычисления модуля. *Назначение:* Выполняет вычисление абсолютного значения величины сигнала. *Параметры блока:* **Saturate on integer overflow** – подавлять переполнение целого. **Enable zero crossing detection** – фиксировать прохождение сигнала через нулевой уровень.



Integrator – интегратор. *Назначение:* Интегрирование входного сигнала. *Параметры блока:* **External reset** – внешний сброс. Выбирается из списка: **none** – нет (сброс не выполняется); **rising** – нарастающий сигнал (передний фронт сигнала); **falling** – спадающий сигнал (задний фронт сигнала); **either** – нарастающий либо спадающий сигнал; **level** – ненулевой сигнал (сброс выполняется, если сигнал на управляющем входе

становится не равным нулю). **Initialconditionsource-** источник начального значения выходного сигнала.

Выбирается из списка: **internal**-внутренний; **external**-внешний. **Initialcondition-** начальное условие. **Limitoutput-**

ограничение выходного сигнала. **Uppersaturationlimit-**

верхний предел выходного сигнала. **Lowersaturationlimit-**

нижний предел выходного сигнала. **Showsaturationport-** показать на

пиктограмме порт насыщения. Выходной сигнал данного порта может принимать следующие значения: нуль, если интегратор не находится на ограничении; +1, если выходной сигнал интегратора достиг верхнего предела; -1, если выходной сигнал интегратора достиг нижнего предела.

Showstateport - отобразить/скрыть порт состояния блока. **Absolutetolerance-**

абсолютная погрешность. **Enablezerocrossingdetection-** определять

прохождение сигнала через нулевой уровень.



Terminator

Terminator – концевой приемник. *Назначение:* Блок применяется как заглушка для сигнала, поступающего с выхода другого блока. В том случае, когда выход блока оказывается не подключенным ко входу другого блока Simulink выдает предупреждение в командном окне системы MATLAB. Для исключения таких ситуаций следует использовать блок Terminator. *Параметры блока:* Нет



Clock

Clock – источник времени. *Назначение:* Формирует сигнал, величина которого на каждом шаге равна текущему времени моделирования. *Параметры блока:* **Displaytime-** отображение значения времени на пиктограмме блока. **Decimation** – прореживание.

5 Экспериментальная часть

Прежде всего, отметим, что моделирование в среде **Simulink** ведется во временной области с использованием относительных масштабов по времени и частоте. Длительность битов принята равной , длина исследуемой импульсной последовательности в зависимости от ситуации составляет . Несущие частоты при ортогональном и неортогональном разносе несущих составляли и , ортогональный разнос несущих составлял , а неортогональный разнос - . В процессе исследования отслеживалось соотношение сигнал/шум (**SNR**).

Задание 1:

1. Собрать функциональную модель для исследования цифровых систем при частотном уплотнении, реализованную на основе **BPSK**-модема в соответствии с рисунком 3.3. Время исследования для системы установить **1000**.
2. Произвести исследования системы при отсутствии полосового канального фильтра:
 - 2.1. Найти оптимальную частоту среза **ФНЧ** в промежутки от до при ортогональном (и) и неортогональном(и) разносах несущих частот, которые устанавливаются в блоках **SineWave** и заполнить таблицу 1 для двух случаев.

Таблица 1

	Частота среза ФНЧ				
	π	1.25π	1.5π	1.75π	2π

- 2.2. При частотах среза **ФНЧ** и , изменяя мощность шума в блоке **RandomNumber**, параметр **Variance** от 2 до 9 при этом измерить

отношение **сигнал/шум** и полученные данные занести в таблицу 2. Для того, что бы измерить отношение **сигнал/шум** нужно:

1. Измерить сигнал, отключив второй канал и шум канала распространения, измерителем мощности с блоком **Scope**.
2. Подключить оба канала и шум канала распространения снять измерителем мощности с блоком **Scope** суммарную мощность сигнала и шума.
3. Рассчитать отношение сигнал шум с помощью выражения:

где D - это смесь сигнала и шума.

Таблица 2.

			SNR,дБ				SNR,дБ	
10л				10л				
11л						12л		
10л				10л				
11л				12л				

2.3. По результатам данных таблицы 2 построить водопадоподобные энергетические характеристики (зависимости вероятности битовой ошибки от соотношения сигнал/шум) **BPSK**-модема при ортогональном и неортогональном уплотнении, сравнить их с теоретическими данными.

3. Добавить в систему полосовой канальный фильтр и настроить частоты среза фильтра при помощи выражений:

И произвести такие же исследования как 2.1 и 2.2.

Задание 2:

1. Собрать функциональную модель для исследования цифровых систем при частотном уплотнении, реализованную на основе **QPSK**-модема в соответствии с рисунком 3.7. Время исследования для системы установить 1000.

2. Произвести исследование системы при отсутствии полосового канального фильтра перед демодулятором:

2.1. Найти оптимальную частоту среза **ФНЧ**, который находится в демодуляторе, изменяя ее от до , полученные данные занести в таблицу 3. Произвести данные измерения, как для ортогонального, так и для неортогонального разнеса несущих частот.

Таблица 3.

	Частота среза ФНЧ				
	0.5π	0.75 π	π	1.5 π	2 π

2.2. При полученной оптимальной частоте среза **ФНЧ** в пункте 2.1, изменяя мощность шума в блоке **RandomNumber** параметр **Variance** от 2 до 9, измерить отношение **сигнал/шум** и полученные данные занести в таблицу 2. Для измерения **сигнал/шум** использовать такую же методику как для **BPSK**-модема.

2.3. По результатам данных таблицы 2 построить водопадоподобные энергетические характеристики (зависимости вероятности битовой ошибки от соотношения сигнал/шум) **QPSK**-модема при ортогональном и неортогональном уплотнении, сравнить их с теоретическими данными.

3. Добавить в систему полосовой канальный фильтр и настроить частоты среза фильтра при помощи выражений:

—

И произвести такие же исследования как в пункте 2.1 и 2.2.

4. Сравнить две системы между собой. Написать отчет.

6 Контрольные вопросы

1. Причины использования уплотнения каналов? Виды уплотнение каналов связи?

2. Какие проблемы возникают при частотном уплотнении каналов связи и способы их устранения?

3. Какое частотное уплотнение выгоднее ортогональное или неортогональное? Почему?

4. Как выбирается канальная модуляция для систем с уплотнением каналов связи?

5. Нарисовать упрощенную структурную схему **BPSK**-модема и объяснить принцип работы?

6. Нарисовать упрощенную структурную схему **QPSK**-модема и объяснить принцип работы?

7.Какая система передачи окажется более помехоустойчивой? Какую канальную модуляцию лучше использовать?

8.Почему частота среза **ФНЧ** фильтра меняется при наличии полосового канального фильтра?

Список использованных источников

1. Завьялов, С.В. Повышение спектральной эффективности многочастотных неортогональных сигналов. Диссертация на соискание ученой степени кандидата технических наук. – СПб.: ФГАОУ ВО «Санкт-Петербургский государственный политехнический университет», 2015.- 161с.
2. Гультияев, А.К. **MatLab 5.3**. Имитационное моделирование в среде **Windows**: Практическое пособие / А.К.Гультияев. – СПб.: КОРОНА принт, 2001.- 400 с.
3. Черных, И.В. **Simulink**: среда создания инженерных приложений. / Под общ.ред. В.Г. Потемкина. – М.: ДИАЛОГ-МИФИ, 2003.- 496 с.
4. Дьяконов, В.П. **MatLab 6.5 SP1/7 + Simulink 5/6** в математике и моделировании. Сер. Библиотека профессионала / В.П. Дьяконов. - М.: СОЛОН-Пресс, 2005.- 576 с.