

Министерство науки и высшего образования РФ
Федеральное государственное бюджетное образовательное учреждение
высшего образования
ТОМСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ СИСТЕМ УПРАВЛЕНИЯ
И РАДИОЭЛЕКТРОНИКИ (ТУСУР)

Кафедра радиотехнических систем (РТС)

Кологривов В.А., Васильева Т.В.

**ИССЛЕДОВАНИЕ РАДИОМОДЕМОВ С ИСПОЛЬЗОВАНИЕМ
ОРТОГОНАЛИЗАЦИИ СЛУЧАЙНЫХ ПРОЦЕССОВ**

Учебно-методическое пособие по лабораторной и самостоятельной работе и
практическим занятиям

Министерство науки и высшего образования РФ
Федеральное государственное бюджетное образовательное учреждение
высшего образования
ТОМСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ СИСТЕМ
УПРАВЛЕНИЯ И РАДИОЭЛЕКТРОНИКИ (ТУСУР)
Кафедра радиотехнических систем (РТС)

Утверждаю:

Зав. кафедрой РТС, проф., д.т.н.

_____ Мелихов С.В.

«__» _____ 2019 г.

**ИССЛЕДОВАНИЕ РАДИОМОДЕМОВ С ИСПОЛЬЗОВАНИЕМ
ОРТОГОНАЛИЗАЦИИ СЛУЧАЙНЫХ ПРОЦЕССОВ**

**Учебно-методическое пособие по лабораторной и самостоятельной
работе и практическим занятиям для студентов направления
«Инфокоммуникационные технологии и системы связи»**

Разработчики:

Доц. каф. РТС Кологривов В.А. _____

Студентка гр. 1В5 Васильева Т.В. _____

Кологривов В.А., Васильева Т.В.

«Исследование радиомодемов с использованием ортогонализации случайных процессов»: Учебно-методическое пособие по лабораторной и самостоятельной работе и практическим занятиям для студентов направления «Инфокоммуникационные технологии и системы связи» Томск: ТУСУР. Образовательный портал, 2019. – 32 с.

Учебно-методическое пособие содержит описание модема на основе ортогонализации случайных процессов, выполненного в среде функционального моделирования *Simulink* системы для инженерных и научных расчетов *MatLab*.

В пособии приведены краткие теоретические сведения по передаче на основе шумоподобных сигналов, алгоритм ортогонализации Грама-Шмидта, описание виртуального макета и используемых блоков библиотеки *Simulink*, а также требования к экспериментальному исследованию и контрольные вопросы, ответы на которые необходимы для успешной защиты лабораторной работы.

АННОТАЦИЯ

Лабораторная работа «**Исследование радиомодемов с использованием ортогонализации случайных процессов**» посвящена экспериментальному исследованию модели канала передачи данных на основе ортогонализации случайных процессов с использованием пакета функционального моделирования *Simulink* системы для инженерных и научных расчетов *MatLab*.

Работа «Исследование радиомодемов с использованием ортогонализации случайных процессов» относится к циклу лабораторных работ по разделу «Системы связи с шумоподобными сигналами», входящему дисциплины по направлению «Инфокоммуникационные технологии и системы связи».

В описании сформулирована цель лабораторной работы, приведены краткие теоретические сведения по передаче на основе шумоподобных сигналов, алгоритм ортогонализации Грама-Шмидта, 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.

СОДЕРЖАНИЕ

1 ЦЕЛЬ РАБОТЫ. КРАТКИЕ СВЕДЕНИЯ	6
2 ЗАПУСК И РАБОТА С ПАКЕТОМ SIMULINK	12
3 ОПИСАНИЕ ЛАБОРАТОРНОГО МАКЕТА	14
4 ОПИСАНИЕ ИСПОЛЬЗУЕМЫХ БЛОКОВ БИБЛИОТЕКИ SIMULINK	20
5 ЭКСПЕРИМЕНТАЛЬНОЕ ЗАДАНИЕ	30
6 КОНТРОЛЬНЫЕ ВОПРОСЫ	31
СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ	32

1 ЦЕЛЬ РАБОТЫ. КРАТКИЕ СВЕДЕНИЯ

Цель работы: изучение принципа передачи данных на основе ортогональных случайных процессов с использованием пакета функционального моделирования Simulink.

Теоретические сведения

Шумоподобный сигнал (ШПС) называют такой сигнал, у которого произведение ширины спектра ΔF на длительность T много больше единицы. Это произведение называется базой сигнала [1]:

$$B = T * \Delta F \gg 1.$$

В результате выполнения операции расширения спектра происходит уменьшение уровня спектральной плотности передаваемого сигнала, что затрудняет обнаружение факта работы системы связи и перехват передаваемых сообщений (рис.1.1).

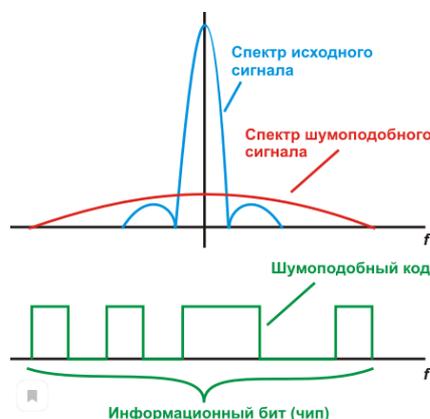


Рисунок 1.1 – Сравнение спектров узкополосного и шумоподобного сигналов

Псевдослучайная последовательность (ПСП) – это последовательность чисел (импульсов), которая была вычислена по некоторому определённом арифметическому правилу, но имеет все свойства случайной последовательности чисел в рамках решаемой задачи [2].

Для расширения спектра передаваемого сигнала исходному сигналу ставится в соответствие более сложный сигнал. Псевдослучайность делает

сигналы похожими на случайный шум и трудными для демодуляции «чужими» приемниками. На практике ПСП чаще всего является бинарной с элементами 0 и 1 и обладает свойствами, схожими со свойствами случайной бинарной последовательности. В данной работе используется ПСП с заданным шагом и со случайной амплитудой.

ПСП, модулированная информационной битовой последовательностью, представляет собой сигнал с расширенным спектром, передаваемый непосредственно либо с помощью несущей.

В настоящее время существует несколько методов формирования сигналов с расширенным спектром. В работе используется генератор ПСП с нормальным распределением амплитуд.

Рассматриваемая технология использования ПСП основана на *ортогонализации* ее фрагментов в зависимости от передачи нуля либо единицы.

Шумовой сигнал разделяется на части (шумовые импульсы) заданной длительности. При передаче текущего шумового импульса следующий шумовой импульс подвергается *ортогонализации* или *коллинеаризации* в зависимости от передаваемого бита данных.

Два соседних импульса \bar{a} и \bar{b} можно назвать *коллинеарными*, если существует такое число n , что [3]:

$$\bar{a} = n * \bar{b}.$$

Напомним основные особенности ортогональных сигналов. Любой набор равно энергетических сигналов $\{S_i(t)\}$, $i = \overline{1, M}$, которые заданы на интервале времени T , будет *ортогональным* и *нормированным* к единице тогда и только тогда, когда [4]:

$$Z_{ij} = \frac{1}{E} \int_0^T S_i(t) * S_j(t) dt = \begin{cases} 1, & \text{при } i = j, \\ 0, & \text{при } i \neq j, \end{cases}$$

где Z_{ij} – коэффициент взаимной корреляции;

E – энергия сигнала:

$$E = \int_0^T S_i^2(t) dt.$$

Одним из наиболее известных алгоритмов разложения матрицы на произведение ортогональной и треугольной матриц является *алгоритм Грама-Шмидта*.

Суть алгоритма основана на следующем утверждении: всякую неособенную матрицу A можно представить в виде произведения матрицы Q с ортогональными столбцами и верхнетреугольной матрицы R с единичной диагональю либо в виде произведения ортонормальной по столбцам матрицы Q и верхнетреугольной матрицы R :

$$A = Q * R.$$

Алгоритм для вычисления двух ортогональных друг другу векторов представлен выражениями (1.2-1.3):

$$q^{(1)} = a^{(1)}, \quad (1.1)$$

$$q^{(2)} = a^{(2)} - \frac{(a^{(2)}, q^{(1)})}{(q^{(1)}, q^{(1)})} * q^{(1)}, \quad (1.2)$$

где $a^{(1)}, a^{(2)}$ – заданные вектора,

$q^{(1)}, q^{(2)}$ – искомые ортогональные вектора.

Для получения *ортонормированных* векторов вместо ортогональных найдем *норму* каждого из векторов $q^{(k)}$, где $k = \{1, 2\}$, и пронормируем по данной величине:

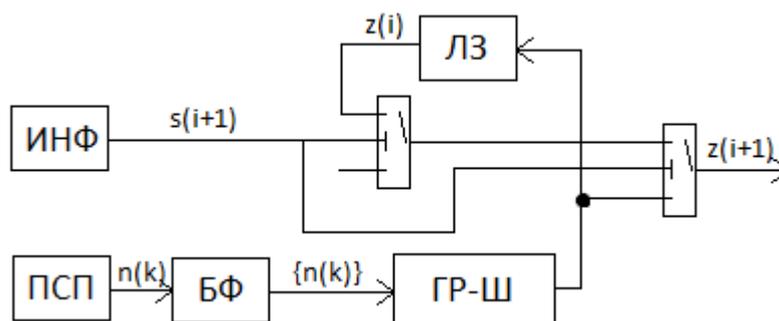
$$\widehat{q^{(k)}} = \frac{q^{(k)}}{\|q^{(k)}\|},$$

где $\|q^{(k)}\| = \sqrt{\sum (q_{ik})^2} = \sqrt{(q^{(k)}, q^{(k)})}$.

Учтем, что $(q^{(k)}, q^{(k)}) = \|q^{(k)}\|^2$. Тогда из (1.1-1.2) получим следующую систему:

$$Q = \begin{cases} q^{(1)} = \frac{a^{(1)}}{\|a^{(1)}\|}, \\ q^{(2)} = a^{(2)} - (a^{(2)}, q^{(1)}) * q^{(1)}, \\ q^{(2)} = \frac{q^{(2)}}{\|q^{(2)}\|}. \end{cases} \quad (1.3)$$

Разберем подробнее работу передатчика и приемника, основанного на ортогонализации случайных процессов. На рисунке 1.2 изображена структурная схема передатчика.



ИНФ – источник информационных битов; ЛЗ – линия задержки; ПСП – генератор псевдослучайной последовательности; БФ – буфер; ГР-Ш – блок ортогонализации по методу Грама-Шмидта

Рисунок 1.2– Структурная схема ортошумового передатчика

От генератора псевдослучайной последовательности поступает случайный сигнал $n(k)$ в буфер. В буфере происходит накопление отсчетов в течение длительности одного информационного бита. Далее над сформированными векторами $\{n(k)\}_{i+1}$, если необходимо, выполняется процедура ортогонализации по методу Грама-Шмидта (1.3). На выходе этого блока образуются ортонормированные шумовые символы с единичной энергией $z(i + 1)$.

Информационные сигналы $s(i + 1)$ являются сигналами управления. При сигнале $s(i + 1) = 1$ сигнал на выходе передатчика повторяет предыдущий символ $z(i + 1) = z(i)$, т.е. является коллинеарным. А при сигнале $s(i + 1) = -1$ передается символ ортогональный предыдущему. Другими словами,

взаимная энергия двух соседних шумовых символов при передаче «-1» будет обнуляться, а при передаче «1» будет равна 1.

В передатчике будут производиться следующие действия:

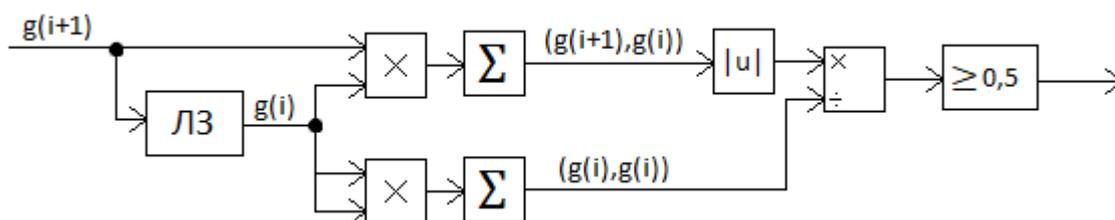
$$z_{i+1} = \begin{cases} \frac{\{n(k)\}_{i+1} - (\{n(k)\}_{i+1}, z_i) * z_i}{\|z_{i+1}\|}, & \text{если } s(i+1) = -1 \\ z_i, & \text{если } s(i+1) = 1. \end{cases}$$

Процесс передачи сопровождается воздействием шумов канала распространения.

Прием передаваемой ПСП основан на использовании корреляционного метода. Разберем более детально работу приемника.

На приемной стороне вычисляется взаимная энергия текущего и предыдущего шумового импульса и нормируется на энергию предыдущего импульса. Если эта энергия меньше порога, принимается решение о передаче нуля, в противном случае – о передаче единицы. Величину порога примем как среднее между энергией ортогональных и коллинеарных сигналов, т.е. равным 0,5.

На рисунке 1.3 представлена структурная схема ортошумового приемника.



ЛЗ – линия задержки; \times – перемножитель; Σ – сумматор; $|u|$ - модуль

Рисунок 1.3– Структурная схема ортошумового приемника

На вход приемника поступает сигнал $g(i+1)$. Далее происходит корреляционная обработка предыдущего и текущего символов. Вычисляется их

взаимная энергия $(g(i + 1), g(i))$ и находится модуль из полученного значения.

Определение переданного бита происходит следующим образом:

$$\begin{cases} s = 1, & \text{если } \frac{|g(i + 1), g(i)|}{(g(i), g(i))} \geq 0,5; \\ s = -1, & \text{если } \frac{|g(i + 1), g(i)|}{(g(i), g(i))} < 0,5. \end{cases}$$

Можно сказать, что при приеме «-1» корреляция с предыдущим символом минимальна, а при приеме «1» - максимальна.

Для исследования помехоустойчивости важно значение отношения сигнал/шум SNR , которое определяется следующим образом:

$$SNR = \frac{N_c}{N_{c+ш} - N_c}, \quad (1.4)$$

где N_c - мощность сигнала без шумов;

$N_{c+ш}$ – мощность сигнала с шумами канала.

Мощность сигнала измеряется на выходе ФНЧ приемника, т.е. на входе блока Zero-Order Hold, принимающего решение о принятом значении.

Отношение сигнал/шум является основной характеристикой, определяющей качество приема данных, и оно влияет на энергетическую эффективность способа передачи.

2 ЗАПУСК И РАБОТА С ПАКЕТОМ SIMULINK

Для запуска системы Simulink необходимо предварительно выполнить запуск системы MatLab. После открытия командного окна системы MatLab нужно запустить систему Simulink. Это можно сделать одним из трех способов:

- 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).

Последний способ предпочтителен при запуске уже готовой и отлаженной модели, когда требуется лишь провести моделирование и не нужно добавлять новые блоки в модель. При применении двух первых способов открывается окно обозревателя библиотеки блоков (Simulink Library Browser) [5].

На рисунке 2.1 выведена библиотека системы Simulink (в левой части окна) и показаны ее разделы (в правой части окна). Основная библиотека системы содержит следующие разделы:

- Continuous – блоки аналоговых элементов;
- Discontinuous – блоки нелинейных элементов;
- Discrete – блоки дискретных элементов;
- Look-Up Tables – блоки 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-Defined Function – функции, определяемые пользователем.

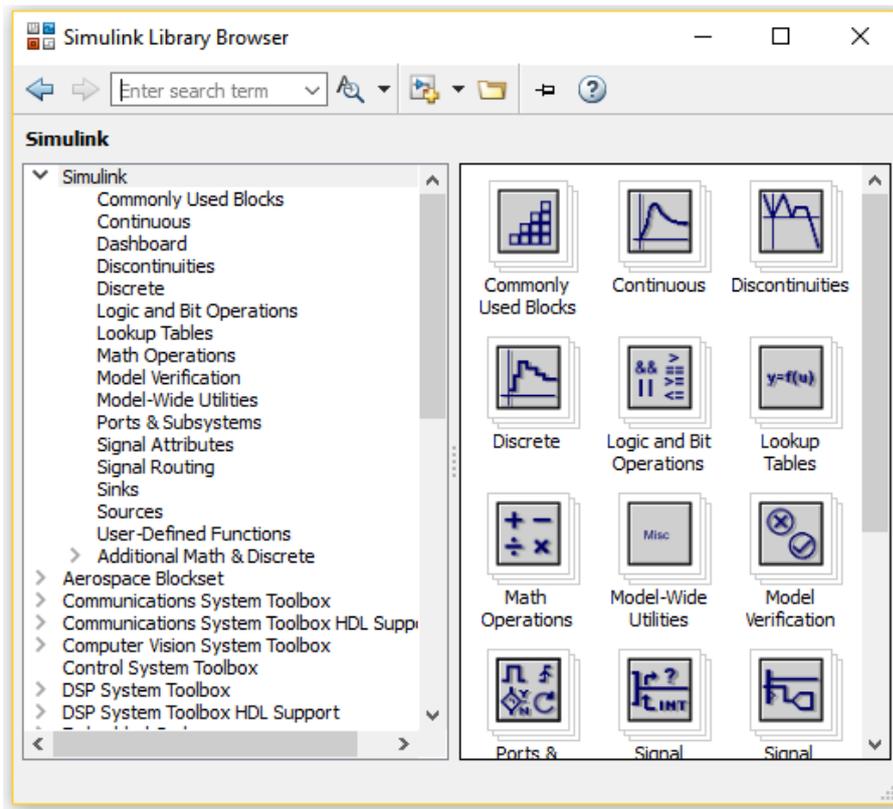


Рисунок 2.1 – Библиотека блоков Simulink Library Browser

Список разделов библиотеки представлен в виде дерева, и правила работы с ним являются общими для списков такого вида: пиктограмма свернутого узла дерева содержит символ «+», а пиктограмма развернутого – символ «-».

Для того чтобы развернуть или свернуть узел дерева, достаточно щелкнуть на его пиктограмме левой клавишей мыши (ЛКМ). При выборе соответствующего раздела библиотеки его содержимое отображается в правой части окна.

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

Приведем описание работы модема на основе ортогонализации случайных процессов в среде функционального моделирования Simulink системы MatLab. Функциональная модель представлена на рисунке 3.1.

В качестве генераторов псевдослучайной последовательности (ПСП) и информационных данных выступают генераторы случайного сигнала с нормальным распределением **Random Number**.

Длительность информационного бита составляет 1, в то время как длительность сигнала ПСП – 1/16.

Модем с ортогонализацией случайных процессов

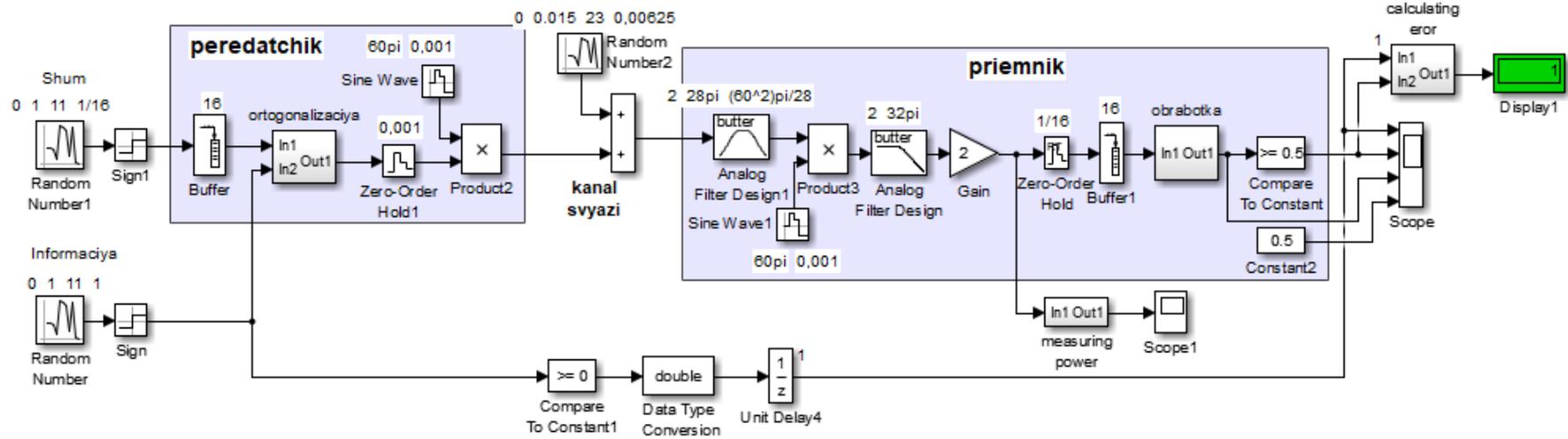


Рисунок 3.1 – Функциональная схема модема с ортогонализацией случайных процессов

Передача: Для ограничения амплитуды шумового сигнала в схеме присутствует блок **Sign**. Далее сигнал накапливается в блоке **Buffer** до 16 отсчетов и поступает на вход подсистемы **ortogonalizaciya**. Схема, находящаяся внутри данной подсистемы, представлена на рисунке 3.2.

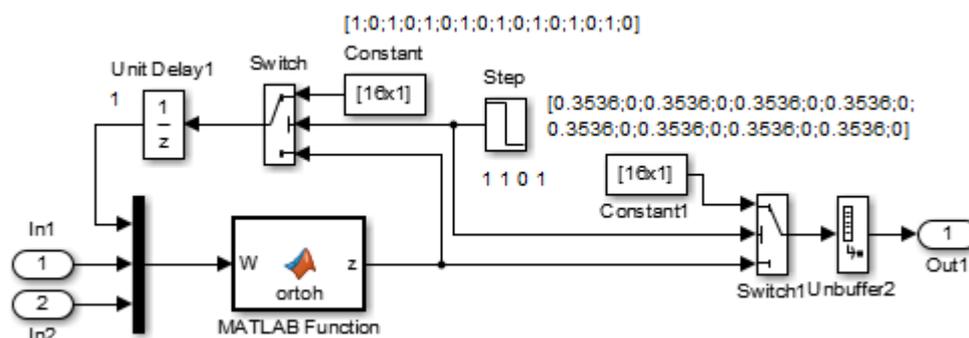


Рисунок 3.2 – Схема в подсистеме ortogonalizaciya

Описание работы подсистемы ortogonalizaciya. Входными сигналами здесь являются вектор из 16 отсчетов шумовой последовательности и текущий информационный бит. Данная подсистема выполняет ортогонализацию или повтор вектора.

В блоке **Constant** находится вектор с размерностью (16,1), состоящий из 1 и 0. Эта последовательность поступает в первый такт работы блока MATLAB Function в качестве предыдущей последовательности. Она нужна для запуска функции ортогонализации.

В блоке **Constant1** формируется вектор с размерностью (16,1), представляющий собой нормированный вектор, который создавался блоком **Constant**. Эта последовательность выходит первой из передатчика. Она необходима для сравнения на приемной стороне с первым обработанным символом.

Далее приведен код, прописанный в блоке **MATLAB Function**. Он представляет собой функцию ортогонализации входных векторов по методу Грама-Шмидта, а также повтора предыдущего шумового символа.

```

function z = ortoh(W)
m=length(W);
n=(m-1)/2;
V=zeros(n,2);
V(:,1)=W(1:n);           %предыдущий обработанный символ
V(:,2)=W(n+1:m-1);      %текущий символ
s=W(m);                 %информационный бит

if V(1,1)==0            %для создания нулевых элементов в начале симуляции
    z=zeros(n,1);
else
    if s==-1            %ортогонализация
        Q=zeros(n,2);
        Q(:,1)=V(:,1)/norm(V(:,1));
        Q(:,2)=V(:,2)-Q(:,1)*Q(:,1)'+V(:,2);
        Q(:,2)=Q(:,2)/norm(Q(:,2));
        z=Q(:,2);
    else
        z=V(:,1)/norm(V(:,1)); %повтор символа (коллинеарность)
    end;
end;
end;

```

Здесь входной вектор W имеет размерность, равную $(2n + 1, 1)$. Его длина $m = 2n + 1$. Вектор W содержит входную шумовую последовательность, предыдущий обработанный символ и текущий информационный бит.

Матрица V составляется из элементов вектора W . Ее размерность = $(n, 2)$. 1-ый столбец матрицы V состоит из элементов $W(1:n)$. Данный столбец представляет собой входную шумовую последовательность. 2-ой столбец матрицы V состоит из элементов $W(n + 1:m - 1)$. Он представляет собой предыдущий обработанный символ.

Последним элементом входного вектора $s=W(m)$ является информационный бит. Он исполняет роль управляющего сигнала, т.е. запускает процесс ортогонализации текущего вектора методом Грама-Шмидта.

Выходной вектор z имеет размерность $(n, 1)$. При $s = 1$ вектор z повторяет предыдущую обработанную последовательность; при $s = -1$ вектор z становится ортогональным вектором.

После формирования последовательности векторов полученный сигнал проходит через блок **Unbuffer**, где вектора преобразуются в последовательность.

Путём умножения полученного сигнала на несущую ($f = 60\pi$), происходит перенос сигнала на высокую частоту.

Модель канала распространения представлена в виде сумматора, на один из входов которого подаётся модель шума, реализованная с помощью блока **Random Number**.

Полосовой фильтр отображает фильтрацию на выходе передатчика и на входе приёмного тракта.

Прием: Демодулятор выполнен на основе умножителя **Product**, на входы которого поступают принятый сигнал и сигнал несущей. Происходит «сброс» сигнала в область низких частот.

Для фильтрации высших гармоник после умножителя установлен **ФНЧ**.

Далее сигнал поступает на блоки **Zero-Order Hold** и **Buffer**, преобразовываясь в последовательность векторов. Она поступает в подсистему **obrabotka**. Схема данной подсистемы изображена на рисунке 3.3.

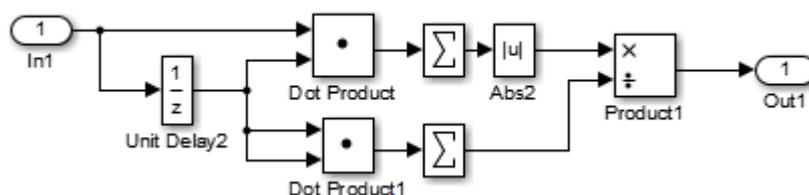


Рисунок 3.3- Схема в подсистеме obrabotka

Описание работы подсистемы obrabotka. В подсистеме происходит вычисление модуля скалярного произведения текущего и задержанного на длительность одного бита символ с помощью блоков **Dot Product**, **Sum of Elements** и **Abs**. Также выполняется нормировка полученного значения по энергии предыдущего символа.

Дальше идет сравнение с порогом, равным 0,5, в блоке **Compare to Constant**. При значении меньше этого порога формируется «0», в противном случае – «1».

Измеритель мощности. Для обеспечения возможности измерения помехоустойчивости, в частности соотношения сигнал/шум, собрана схема измерителя мощности. Ее схема представлена на рисунке 3.4.

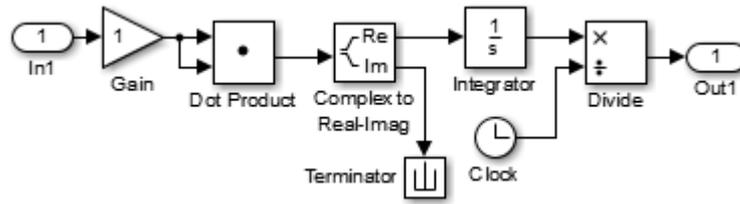


Рисунок 3.4– Измеритель мощности

Детектор ошибок. Для контроля ошибок был поставлен детектор ошибок, схема которого представлена на рисунке 3.5.

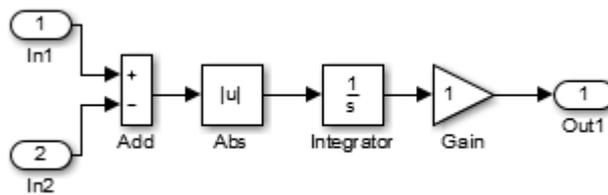


Рисунок 3.5– Детектор ошибок

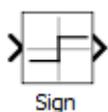
Количество битовых ошибок отображается в блоке **Display**.

4 ОПИСАНИЕ ИСПОЛЬЗУЕМЫХ БЛОКОВ БИБЛИОТЕКИ SIMULINK

Ниже описаны блоки базовых разделов библиотеки Simulink, используемые в функциональной схеме модема с ортогонализацией случайного сигнала [6].

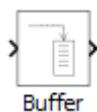


Random number – источник случайного сигнала с нормальным распределением. *Назначение:* формирование случайного сигнала с равномерным распределением уровня сигнала. *Параметры блока:* **Mean** - среднее значение сигнала; **Variance** - дисперсия; **Seed** - начальное значение генератора случайного сигнала; **Sample time** - такт дискретности.

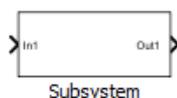


Sign – блок определения знака сигнала. *Назначение:* определяет знак входного сигнала, при этом, если x - входной сигнал, то сигнал на выходе определяется выражением:

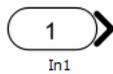
$$\text{sign}(x) = \begin{cases} -1 & \text{при } x < 0, \\ 0 & \text{при } x = 0, \\ 1 & \text{при } x > 0. \end{cases}$$



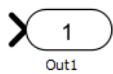
Buffer – блок буферизации. *Назначение:* служит для буферизации сигналов. Поток данных сигнала дробится на части (фреймы) заданного размера. Размер буфера, выделяемого под задержанный сигнал, в байтах (число, кратное 8, по умолчанию 1024 байта).



Subsystem – создание подсистем. *Назначение:* Подсистема - это фрагмент Simulink-модели, оформленный в виде отдельного блока.



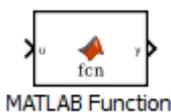
Import – входной порт. *Назначение:* Создает входной порт для подсистемы или выполняет считывание сигнала с рабочей области MATLAB в модель. *Параметры блока:* **Portnumber** - номер порта; **Portdimensions** - размерность входного сигнала; **Sampletime** - такт дискретности; **Show Additional Parameters** - показать дополнительные параметры. **Data Type** - выбор типа выходных данных; **Output Data Type** - тип выходных данных; **Output Scaling Mode** - способ масштабирования выходного сигнала; **Output Scaling Value** - величина масштаба; **Sampling mode** - режим.



Outputport – выходной порт. *Назначение:* Создает входной порт для подсистемы или для модели верхнего уровня иерархии. *Параметры блока:* **Portnumber** – номер порта; **Output When Disabled** – 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 **Initial Output** - начальное значение.



Mux – блок мультиплексора. *Назначение:* объединяет входные сигналы в вектор. Параметры блока: **Number of Inputs** - количество входов; **Display option** - способ отображения, выбирается из списка: **bar** - вертикальный узкий прямоугольник черного цвета; **signals** - прямоугольник с белым фоном и отображением меток входных сигналов; **none** - прямоугольник с белым фоном без отображения меток входных сигналов.



MATLAB Function – блок задания М - функции. *Назначение:* задает выражение в стиле языка программирования MATLAB. *Параметры блока:* MATLAB Function - выражение на языке MATLAB. **Output Dimensions** – размерность выходного сигнала; **Output Signal Type** - 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.



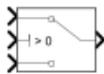
Constant

Constant – источник постоянного сигнала. *Назначение:* задает постоянный по уровню сигнал. *Параметры блока:* **Constant value** - постоянная величина; **Interpret vector parameters as 1-D** - интерпретировать вектор параметров как одномерный (при установленном флажке). Значение константы может быть действительным или комплексным числом, вычисляемым выражением, вектором или матрицей.



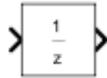
Step

Step – генератор ступенчатого сигнала. *Назначение:* формирует ступенчатый сигнал. *Параметры:* **Step time** - время наступления перепада сигнала (с); **Initial value** - начальное значение сигнала; **Final value** - Конечное значение сигнала. Перепад может быть как в большую сторону (конечное значение больше чем начальное), так и в меньшую (конечное значение меньше чем начальное). Значения начального и конечного уровней могут быть не только положительными, но и отрицательными (например, изменение сигнала с уровня -5 до уровня -3).



Switch

Switch – блок двухпортового переключателя. *Назначение:* переключение входных сигналов по сигналу управления. *Параметры блока:* **Criteria for passing first input** - условие прохождения сигнала с первого входа, значение выбирается из списка: $u_2 \geq \text{Threshold}$ - сигнал управления больше или равен пороговому значению; $u_2 > \text{Threshold}$ - сигнал управления больше порогового значения; $u_2 \neq \text{Threshold}$ - сигнал управления не равен пороговому значению. **Threshold** - порог; флажок **Show additional parameters** - показать дополнительные параметры.



Unit Delay **Unit delay** – блок единичной дискретной задержки. *Назначение:*

выполняет задержку дискретного сигнала на заданный такт дискретности.

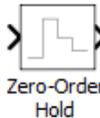
Параметры блока: **Initial conditions** – начальное значение выходного сигнала;

Sample time – такт дискретности (при задании значения параметра равного -1 такт дискретности наследуется от предшествующего блока).



Unbuffer **Unbuffer** – блок, выполняющий обратную операции буферизации.

Назначение: преобразовывает кадр определенной длины в поток данных.



Zero-Order Hold **Zero-Order Hold** – 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:* **Sample time** – такт дискретности. Блок экстраполятора нулевого порядка может использоваться также для согласования работы дискретных блоков, имеющих разные такты дискретности.



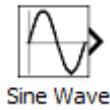
Product **Product** – блок умножения и деления. *Назначение:* вычисление

произведения текущих значений сигналов. *Параметры блока:* **Number of inputs** – 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 **Show additional parameters** – показать дополнительные параметры. При

выставленном флажке отображается окно списка **Output data type mode**, в нашем случае флажок не используется.



Sine Wave – блок источника синусоидального сигнала.

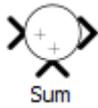
Назначение: формирование синусоидального сигнала с заданной частотой, амплитудой, фазой и смещением. Для формирования выходного сигнала блоком могут использоваться два алгоритма. Вид алгоритма определяется параметром **Sine Type** – способ формирования сигнала реализуется двумя алгоритмами: **Time-based** – 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 блоков в режиме *Time-based*: **Amplitude** – амплитуда; **Bias** – постоянная составляющего сигнала (смещение); **Frequency (rads/sec)** – частота (рад/с); **Phase (rads)** – начальная фаза (рад); **Sample time** – 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 **Interpret Vector Parameters As 1 – D** – интерпретировать вектор как массив скаляров. Для очень больших значений времени точность вычисления значений сигнала падает.

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

$$N = \text{Samples per second} = \frac{1}{f \cdot T}$$

$$p = \text{Samples per period} = 2 \cdot \pi \cdot N$$



Sum

Sum of Elements – 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**, | - **разделитель**. Флажок **Show Additional Parameters** – показать дополнительные параметры, при выставленном флажке отображаются окна списка **Output Data Type Mode**, в нашем случае не используется. Количество входов и соответствующие им операции определяются списком знаков **List Of Sign**. При этом метки входов обозначаются соответствующими знаками. В списке **List Of Sign** можно также указать число входов, при этом все входы будут суммирующими.

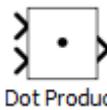
Analog
Filter Design

Analog Filter Design – блок аналогового фильтра заданного

метода проектирования и типа из подраздела **Filter Design**. *Назначение:* аналоговая фильтрация низкочастотных составляющих спектра входного сигнала. *Параметры блока*: **Design Method** – метод проектирования, выбирается из списка: **Butterworth** – фильтр Баттерворта; **Chebyshev I** – фильтр Чебышева 1-го рода; **Chebyshev II** – фильтр Чебышева 2-го рода; **Elliptic** – фильтр эллиптический; **Bessel** – фильтр Бесселя. **Filter Type** – тип фильтра, выбирается из списка: **Low Pass** – нижних частот; **High pass** – верхних частот; **Band Pass** – полосно-пропускающий; **Band Stop** – полосно-заграждающий. Далее для каждого метода проектирования и типа фильтра выдается свой список параметров. Так для фильтра Баттерворта типа нижних частот параметрами являются: **Filter Order** – порядок фильтра; **Passband Edge Frequency (rads/sec)** – нижняя граничная частота (радиан в секунду). Для других методик проектирования и типов фильтров определяемые параметры очевидны.



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 **Show additional parameters** – показать дополнительные параметры, при выставленном флажке отображаются окна списков **Parameter data type mode**, **Output data type mode**. **Saturate on integer** – подавлять переполнение целого. При установленном флажке ограничение сигналов целого типа выполняется корректно.



Dot Product – блок скалярного произведения. *Назначение:* выполняет вычисление скалярного произведения (свертку) двух векторов. Блок выполняет вычисление выходного сигнала в соответствии с выражением:

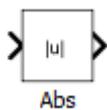
$$y = \text{sum}(\text{conj}(u1).* u2),$$

где **$u1$** и **$u2$** – входные векторы;

conj – операция вычисления комплексно-сопряженного числа;

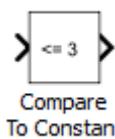
sum – операция вычисления суммы.

Если оба входных вектора являются действительными, то выходной сигнал также будет действительным. Если хотя бы один из входных векторов содержит комплексный сигнал, то выходной сигнал будет комплексным.



Abs – блок вычисления модуля. *Назначение:* выполняет вычисление абсолютного значения величины сигнала. *Параметры:* **Saturate on integer overflow** (флажок) – подавлять переполнение целого. При

установленном флажке ограничение сигналов целого типа выполняется корректно.

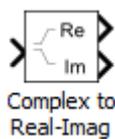


Compare to Constant – блок сравнения. *Назначение:* сравнивает сигнал с указанной константой. *Параметры:* **Constant value** – постоянная величина; **Operator** – тип сравнения с указанной величиной выбирается из списка: **==** – определяет, является ли входной сигнал равным константе; **~=** – определяет, является ли входной сигнал не равным константе; **<** – определяет, является ли входной сигнал меньше константы; **<=** – определяет, является ли входной сигнал равным константе или меньше нее; **>** – определяет, является ли входной сигнал больше константы; **>=** – определяет, является ли входной сигнал равным константе или больше нее.

На выходе блока формируется «0», если сравнение не выполняется, и «1», если сравнение выполняется.



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, **Number of axes** - число входов осциллографа, **Time range** – отображаемый временной интервал и другие; изменить масштабы графиков; установить и сохранить настройки; перевести в плавающий режим и так далее.



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&Imag** – действительная и мнимая часть.



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 к входу другого блока **Simulink** выдает предупреждение в командном окне системы MATLAB. Для исключения таких ситуаций следует использовать блок **Terminator**.



Integrator

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). **Initial Condition Source** – источник начального значения выходного сигнала.

Выбирается из списка: **internal** – внутренний; **external**-внешний. **Initial Condition** – начальное условие. **Limit output** –ограничение выходного сигнала. **Upper Saturation Limit**-верхний предел выходного сигнала. **Lower Saturation Limit** – нижний предел выходного сигнала. **Show Saturation Port** -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. **Show State Port** - отобразить/скрыть порт состояния блока. **Absolute Tolerance** – абсолютная погрешность. **Enable Zero Crossing Detection** -определять прохождение сигнала через нулевой уровень.



Clock

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



Data Type Conversion

Data Type Conversion – блок преобразования типа сигнала. *Назначение:* блок преобразует тип входного сигнала. *Параметры:* **Data type** – тип данных выходного сигнала. Может принимать значения (выбираются из списка): **auto, double, single, int8, int16, int32, uint8, uint16, uint32** и **boolean**. **Saturate on integer overflow** (флажок) – подавлять переполнение целого. При установленном флажке ограничение сигналов целого типа выполняется корректно. Значение **auto** параметра **Data type** используется в том случае, если необходимо установить тип данных такой же, как у входного порта блока получающего сигнал от данного блока.

Входной сигнал блока может быть действительным или комплексным. В случае комплексного входного сигнала выходной сигнал также будет комплексным. Блок работает со скалярными, векторными и матричными сигналами.

5 ЭКСПЕРИМЕНТАЛЬНОЕ ЗАДАНИЕ

1. Собрать Sim-модель для исследования метода передачи на основе ортогонализации случайного сигнала в соответствии с рисунком 3.1.

2. Выставить следующие параметры в блоках Sim-модели, при которых за время передачи одного бита формируется 16 отсчетов:

блок Random Number1: Sample time = 1/16;

блоки Buffer и Buffer1: Output buffer size = 16;

блок Zero-Order Hold: Sample time = 1/16;

блок Const подсистемы ortogonalizaciya: Constant value = [1;0;1;0;1;0;1;0;1;0;1;0;1;0;1;0].

3. Пронаблюдать и зафиксировать основные осциллограммы, иллюстрирующие работу модема на основе ортогонализации случайного сигнала (**Scope**).

4. Снять значения мощности сигнала на выходе **ФНЧ** при наличии и отсутствии шумов и рассчитать значение **SNR** по формуле (1.4) при числе ошибок $n_{err} = 1$ на 10^3 шагов.

5. Увеличить количество случайных сигналов до 32 за время передачи одного информационного бита, выставив в следующих блоках значения параметров:

блок Random Number1: Sample time = 1/32;

блоки Buffer и Buffer1: Output buffer size = 32;

блок Zero-Order Hold: Sample time = 1/32;

блок Const подсистемы ortogonalizaciya: Constant value = [1;0;1;0;1;0;1;0;1;0;1;0;1;0;1;0;1;0;1;0;1;0;1;0;1;0;1;0;1;0;1;0].

6. Повторить пункт 4.

7. Сравнить полученные результаты и сделать выводы.

8. Написать отчет с кратким описанием работы исследуемого модема.

9. Защитить отчет.

6 КОНТРОЛЬНЫЕ ВОПРОСЫ

1. Какой сигнал называется широкополосным?
2. Что такое ПСП?
3. Какие векторы являются коллинеарными?
4. Как определяется норма вектора?
5. С помощью какого метода можно получить ортонормированные вектора?
6. Как в исследуемой модели достигается коллинеарность соседних векторов?
7. Какой сигнал называется широкополосным?
8. Назовите условие, при котором в исследуемом передатчике происходит ортогонализация символа.
9. Что такое корреляция?
10. Поясните, почему в приемнике при обработке принятого сигнала выбран порог, равный 0,5.
11. Как влияет на помехоустойчивость увеличение количества случайных отсчетов за время передачи одного информационного бита?

СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ

- 1 Метод прямого расширения спектра [Электронный ресурс]. – режим доступа https://bstudy.net/641297/tehnika/metod_pryamogo_rasshireniya_spektra_dsss (дата обращения: 03.06.19)
- 2 Псевдослучайная последовательность [Электронный ресурс]. – режим доступа https://ru.wikipedia.org/wiki/Псевдослучайная_последовательность (дата обращения: 03.06.19)
- 3 Коллинеарность [Электронный ресурс]. – режим доступа <https://ru.onlinemschool.com/math/library/vector/colinearity/> (дата обращения: 04.06.19)
- 4 Ортогональность [Электронный ресурс]. – режим доступа https://studref.com/397045/tehnika/ortogonalnye_nizkochastotnye_signaly (дата обращения: 04.06.19)
- 5 Гультяев, А.К. MatLab 5.3. Имитационное моделирование в среде Windows: Практическое пособие / А.К.Гультяев. – СПб.: КОРОНА принт, 2001.- 400 с.
- 6 Черных, И.В. Simulink: среда создания инженерных приложений. / Под общ.ред. В.Г. Потемкина. – М.: ДИАЛОГ-МИФИ, 2003.- 496 с.