

Министерство науки и высшего образования Российской Федерации

Томский государственный университет
систем управления и радиоэлектроники

Кологривов В. А.
Суздальцева Е.В.

ИССЛЕДОВАНИЕ КВАДРАТУРНОЙ РЕАЛИЗАЦИИ BPSK МОДЕМА

Методические указания по лабораторной работе в среде функционального моделирования
Simulink системы MatLab для студентов радиотехнических специальностей

Томск 2022

УДК 621.396
ББК 32.844
К 61

Рецензент:

Мещеряков А.А., доцент кафедры радиотехнических систем ТУСУР, канд. техн. наук

Кологривов В. А., Суздальцева Е. В.

К 61 Исследование квадратурной реализации **BPSK** модема: методические указания по лабораторной работе для студентов радиотехнических специальностей / Кологривов В. А., Суздальцева Е. В. – Томск: Томск. гос. ун-т систем упр. и радиоэлектроники, 2022 – 30 с.

Настоящие методические указания по лабораторной работе составлены с учетом требований федерального государственного образовательного стандарта высшего образования (ФГОС ВО).

Лабораторная работа «Исследование квадратурной реализации **BPSK** модема» с использованием пакета функционального моделирования **Simulink** системы для инженерных и научных расчетов **MatLab**.

Работа «Исследование квадратурной реализации **BPSK** модема» относится к циклу лабораторных работ по разделу «Модуляция/Демодуляция», входящему в дисциплины радиотехнических специальностей.

В описании сформулирована цель лабораторной работы, приведены краткие теоретические сведения о **BPSK** и **QBPSK** модуляции и особенностях её квадратурной реализации, краткая характеристика пакета **Simulink** системы **MatLab**, описание виртуального лабораторного макета и используемых блоков библиотеки **Simulink**, а также требования к модельному исследованию и контрольные вопросы, ответы на которые необходимы для успешной защиты лабораторной работы.

Одобрено на заседании каф. РТС протокол № 4 от 11.11.22

УДК 621.396
ББК 32.844

© Кологривов В. А., Суздальцева Е. В., 2022
© Томск. гос. ун-т систем упр. и радиоэлектроники, 2022

ОГЛАВЛЕНИЕ

1 ЦЕЛЬ РАБОТЫ. КРАТКИЕ ТЕОРЕТИЧЕСКИЕ СВЕДЕНИЯ О QPSK-МОДУЛЯЦИИ.....	4
1.1 Краткие теоретические сведения	4
1.2 Математическое описание <i>QPSK</i> модуляции.....	7
2 ОПИСАНИЕ ФУНКЦИОНАЛЬНЫХ SIM-МОДЕЛЕЙ QPSK-МОДЕМА	9
3 ПРИНЦИП РАБОТЫ ФУНКЦИОНАЛЬНОЙ SIM-МОДЕЛИ QPSK-МОДЕМА.....	16
4 КРАТКОЕ ОПИСАНИЕ ПАКЕТА SIMULINK И ИСПОЛЬЗУЕМЫХ БЛОКОВ	21
4.1 Запуск и работа с пакетом <i>Simulink</i>	21
4.2 Описание используемых блоков библиотеки <i>Simulink</i>	23
5 ЭКСПЕРИМЕНТАЛЬНОЕ ЗАДАНИЕ.....	28
6 КОНТРОЛЬНЫЕ ВОПРОСЫ.....	29
СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ	30

1 ЦЕЛЬ РАБОТЫ. КРАТКИЕ ТЕОРЕТИЧЕСКИЕ СВЕДЕНИЯ О QBPSK-МОДУЛЯЦИИ

Цель работы: исследование квадратурной реализации **BPSK** модуляции (**QBPSK**) – помехоустойчивости, энергетической и спектральной эффективности, сравнение способов приёма и установление основного достоинства - лучшей среди **MPSK** помехоустойчивостью с использованием пакета функционального моделирования *Simulink*.

1.1 Краткие теоретические сведения

PSK модуляция относится к узкополосной технологии формирования и использования радиосигналов.

Наиболее простой является **BPSK** модуляция. Информационный битовый поток одновременно является модулирующим, то есть бит является управляющим символом. Модулятор **BPSK** представляет собой перемножитель биполярного информационного потока и гармонического колебания несущей частоты, то есть **BPSK** модуляция это манипуляция фазы несущего колебания.

Демодуляция **BPSK** представляет собой приёмник прямого преобразования, при котором принятый модулированный сигнал умножается на опорное колебание несущей частоты. В результате образуется разностный продукт нулевой частоты, содержащий модулирующий сигнал и суммарный продукт второй гармоники несущего колебания. Суммарный продукт преобразования отфильтровывается с помощью ФНЧ, а разностный продукт регенерируется по форме до передаваемого битового потока. Иногда прямое преобразование спектра принимаемого сигнала называют синхронным детектированием.

Вторым возможным способом приёма и обработки **BPSK** модулированного сигнала является корреляционный приём. При этом в буферах плеч коррелятора накапливаются отсчеты опорного гармонического колебания и отсчеты принятого в шумах **BPSK** модулированного сигнала. Далее накопленные в течении бита отсчеты скалярно перемножаются (то есть скалярно перемножаются и суммируются) и нормируются к ± 1 . Это соответствует вычислению автокорреляционной функции при нулевом смещении $\tau = 0$.

Биполярная версия информационного битового потока соответствует состояниям $\pm E$ и межсигнальному расстоянию $2E$, где E - энергия сигнала. Это максимальное межсигнальное расстояние соответствует наибольшей помехоустойчивости **BPSK** или энергетической эффективности.

Спектральная эффективность **BPSK** определяется длительностью бита или скоростью передачи битового потока. В первом приближении спектр псевдослучайного информационного потока приравнивается к спектру биполярного меандра (последовательность прямоугольных импульсов со скважностью 2). Это означает, что при длительности бита τ_b ширина основного лепестка спектра составляет $2\pi/\tau_b$ радиан/сек. Величины фазовых переходов при **BPSK** модуляции составляют 0 и π радиан, что приводит к заметным переходным процессам в полосовом фильтре (провале огибающей радиосигнала) и предъявляет повышенные требования к линейности используемых усилителей мощности.

Квадратурная реализация BPSK. Кроме рассмотренной классической реализации возможна и квадратурная реализация **BPSK** модулятора (**QBPSK**). При этом, вместо назначения $+1$ и -1 фазовых состояний 0 и π в фазовом коде задаются, например, состояния $\pi/4$ и $-3\pi/4$ либо $-\pi/4$ и $3\pi/4$. От этих состояний φ_k вычисляются функции $\cos(\varphi_k)$ и $\sin(\varphi_k)$ и подаются на первые входы умножителей квадратурного модулятора. На вторые входы умножителей поступают квадратурные колебания несущей частоты $\cos(\omega_0 t)$ и $\sin(\omega_0 t)$. На выходе квадратурного модулятора квадратурные составляющие суммируются либо вычитаются в зависимости от выбранных состояний.

Демодулятор **QPSK** также имеет квадратурную структуру. Принятый модулированный сигнал разветвляется на первые входы умножителей (преобразователей), а на вторые входы подаются синхронизированные квадратурные опорные колебания несущих $\cos(\omega_0 t)$ и $\sin(\omega_0 t)$. На выходах умножителей ФНЧ пропускают только низкочастотные разностные продукты преобразования и подавляют суммарные высокочастотные продукты преобразования. Низкочастотные разностные продукты преобразователей, в виде квадратурных проекций фазовых состояний поступают на вход сумматора **Sum** либо вычитателя выполняющего роль фазового декодера в зависимости от выбранных состояний и в результате получаем поток принятых битов.

Вторым возможным способом приёма и обработки **QPSK** модулированного сигнала является корреляционный приём. При этом для каждого квадратурного канала реализуется свой коррелятор. На первом плече коррелятора в пределах принимаемого бита накапливаются отсчеты соответствующего синхронизированного опорного генератора, а на втором плече накапливаются отсчеты принятого в помехах **QPSK** модулированного сигнала. Скалярно перемножая накопленные отсчеты как вектора, восстанавливаем соответствующую квадратурную проекцию. Далее восстановленные квадратурные проекции фазового состояния подаём на сумматор **Sum** либо вычитатель выполняющего роль фазового декодера в зависимости от выбранных состояний и получаем поток принятых битов.

Так как при **QPSK** модулирующие символы совпадают по длительности с символами **BPSK**, ширина основного лепестка спектра **QPSK** сигнала совпадает со спектром **BPSK** сигнала. Пара фазовых состояний, равномерно расположенные на окружности радиусом E дают межсигнальное расстояние $2E$, значит, помехоустойчивости **QPSK** и **BPSK** модуляций совпадают.

При **QPSK** и **BPSK** модуляциях возможные фазовые переходы составляют 0 и π радиан. Это приводит к заметным переходным процессам (провале огибающей радиосигнала) в полосовом фильтре и предъявляет повышенные требования к линейности используемых усилителей мощности.

Следующей версией **PSK** модуляции является квадратурная фазовая манипуляция **QPSK** модуляция. В этом случае текущие биты группируются в пары (дибиты) $\tau_s = 2 \cdot \tau_b$. Скорость передачи в два раза выше, чем у **BPSK**, передача одного символа эквивалентна передаче двух битов. Дибиты могут принимать значения **00**, **01**, **10** и **11**. Каждому дибиту приписывается фазовое состояние, например, $\varphi_k = \pm \pi/4$, при этом используется так называемое кодирование по Грью (соседние состояния отличаются одним битом), что способствует равномерному распределению ошибок. Фазовое кодирование обычно выполняется программно. Далее от фазовых состояний вычисляются значения $\cos(\varphi_k)$ и $\sin(\varphi_k)$ и тем самым образуются квадратурные модулирующие импульсы длиной τ_s и амплитудой $\pm 1/\sqrt{2}$, которые подаются на квадратурный модулятор.

Квадратурный модулятор состоит из двух умножителей (преобразователей) на первые входы которых поступают модулирующие импульсы, а на вторые входы опорные квадратурные колебания несущей $\cos(\omega_0 t)$ и $\sin(\omega_0 t)$. На выходе квадратурного модулятора стоит сумматор квадратурных составляющих модулированных несущих.

Демодулятор **QPSK** также имеет квадратурную структуру. Принятый модулированный сигнал разветвляется на первые входы умножителей (преобразователей), а на вторые входы подаются синхронизированные квадратурные опорные колебания несущих $\cos(\omega_0 \cdot t)$ и $\sin(\omega_0 \cdot t)$. На выходах умножителей ФНЧ пропускают только низкочастотные разностные продукты преобразования и подавляют суммарные высокочастотные продукты преобразования. Низкочастотные разностные продукты преобразователей, представляющие собой квадратурные проекции фазовых состояний, поступают на вход программного блока фазового декодера, в котором определяется квадрант комплексной плоскости, определяемый проекциями. По квадранту восстанавливается принадлежащее ему фазовое состояние и

соответствующий дибит как вектор. Далее преобразователь параллельного представления в последовательное представление на основе двух портового переключателя *Switch* последовательно выдаёт текущие биты принятого дибита.

Вторым возможным способом приёма и обработки *QPSK* модулированного сигнала является корреляционный приём. При этом для каждого квадратурного канала реализуется свой коррелятор. На первом плече коррелятора в пределах принимаемого дибита накапливаются отсчеты синхронизированного опорного генератора, а на втором плече накапливаются отсчеты принятого в помехах *QPSK* модулированного сигнала. Скалярно перемножая накопленные отсчеты как вектора, восстанавливаем соответствующую квадратурную проекцию. Далее восстановленные квадратурные проекции фазового состояния подаём на программный блок фазового декодера и преобразователь параллельного представления в последовательное представление и на выходе получаем последовательный поток принятых битов.

Так как при *QPSK* модулирующие символы в два раза длиннее, чем у *BPSK* ширина основного лепестка спектра сокращается вдвое. Четыре фазовых состояния, равномерно расположенные на окружности радиусом E дают межсигнальное расстояние $E\sqrt{2}$, а значит и меньшую помехоустойчивость.

При *QPSK* модуляции возможные фазовые переходы составляют 0 , $\pi/2$ и π радиан. Смена фазовых состояний вызывает переходные процессы в полосовом фильтре, что выражается в кратковременном изменении уровня огибающей модулированного сигнала и чревато появлением нелинейных искажений в усилителях мощности работающих в нелинейном режиме. Чем больше перепад фаз, тем больше провал огибающей и больше уровень нелинейных продуктов. В связи с этим желательно ограничить перепад фаз.

OQPSK – квадратурная модуляция со смещением является разновидностью *QPSK* модуляции, в которой исключаются фазовые переходы на π радиан, за счет смещения квадратурных модулирующих потоков на половину символа, что исключает одновременную смену управляющих символов. Формально смещение одного из модулирующих потоков на половину символа не должно влиять на энергетическую и спектральную эффективность, однако присутствие в структуре модема полосового фильтра *RF* способно заметно повлиять на ситуацию. Так возможность дальнейшего отслеживания смещения на 1 бит в демодуляторе заставляет при выборе полосы пропускания ориентироваться на τ_b , а не на τ_s . В свою очередь увеличение полосы пропускания полосового фильтра увеличивает уровень проходящих на демодулятор шумов, а значит, снижает помехоустойчивость канала передачи.

8PSK квадратурная модуляция – модулирующие символы по длительности равны трибитам $\tau_s = 3\tau_b$, что эквивалентно снижению требуемой полосы в три раза. Трём битам соответствует 8 возможных фазовых состояний. Равномерное размещение 8 -ми фазовых состояний на окружности радиусом E приводит к сокращению межсигнального расстояния до $E \cdot 0.7654$, а значит и к снижению помехоустойчивости. Скорость передачи в три раза выше, чем у *BPSK*, передача одного символа эквивалентна передаче трёх битов. В отличие от *BPSK* и *QPSK* огибающая *8PSK* модулированного сигнала не постоянна.

16PSK квадратурная модуляция – модулирующие символы по длительности равны квадербитам $\tau_s = 4\tau_b$, что эквивалентно снижению требуемой полосы в четыре раза. четырём битам соответствует 16 возможных фазовых состояний. Равномерное размещение 16 -ми фазовых состояний на окружности радиусом E приводит к сокращению межсигнального расстояния до $E \cdot 0.3902$, а значит и к снижению помехоустойчивости. Скорость передачи в четыре раза выше, чем у *BPSK*, передача одного символа эквивалентна передаче четырёх битов. В отличие от *BPSK* и *QPSK* огибающая *16PSK* модулированного сигнала не постоянна.

1.2 Математическое описание *QBPSK* модуляции

Дадим простое математическое описание *QBPSK* модуляции, используя структуру модема [1-3]. Положим в основу наших рассуждений известную тригонометрическую формулу

$$\cos(\varphi_k) \pm \sin(\varphi_k) = \sqrt{2} \sin(\pi/4 \pm \varphi_k) = \cos(\pi/4 \mp \varphi_k).$$

Обозначим текущее значение бита ± 1 соответствующим фазовым состоянием, например, $\varphi_k = \frac{\pi}{4}$ и $\varphi_k = -\frac{3\pi}{4}$, тогда квадратурные модулирующие импульсы запишутся $d_k = \cos(\varphi_k)$ и $g_k = \sin(\varphi_k)$. На выходе квадратурного модулятора имеем $x_k = \langle d_k \cos(\omega t) + g_k \sin(\omega t) \rangle$.

В синфазном плече демодулятора имеем

$$y_k = \langle d_k \cos(\omega t) + g_k \sin(\omega t) \rangle \cos(\omega t) = \frac{d_k}{2} \langle 1 + \cos(2\omega t) \rangle + \frac{g_k}{2} \sin(2\omega t).$$

После ФНЧ и усиления на 2 имеем $y_k \sim d_k$.

В квадратурном плече демодулятора имеем

$$z_k = \langle d_k \cos(\omega t) + g_k \sin(\omega t) \rangle \sin(\omega t) = \frac{d_k}{2} \sin(2\omega t) + \frac{g_k}{2} \langle 1 - \cos(2\omega t) \rangle.$$

После ФНЧ и усиления на 2 имеем $z_k \sim g_k$.

Сложение сигналов плеч квадратурного демодулятора даёт

$$s_k = d_k + g_k = \cos(\varphi_k) + \sin(\varphi_k) = \sqrt{2} \sin(\pi/4 + \varphi_k) = \cos(\pi/4 - \varphi_k),$$

тогда при $\varphi_k = \frac{\pi}{4}$ на выходе квадратурного демодулятора имеем значение

$$s_k = \sqrt{2} \sin(\pi/4 + \varphi_k) = \cos(\pi/4 - \varphi_k) = \sqrt{2},$$

при $\varphi_k = -\frac{3\pi}{4}$ на выходе квадратурного демодулятора имеем значение

$$s_k = \sqrt{2} \sin(\pi/4 + \varphi_k) = \cos(\pi/4 - \varphi_k) = -\sqrt{2}.$$

Итого после нормировки амплитуд на выходе сумматора демодулятора имеем $s_k = \pm 1$.

Таким образом, при выборе фазовых состояний, $\varphi_k = \frac{\pi}{4}$ и $\varphi_k = -\frac{3\pi}{4}$, для значений битов ± 1 обычный сумматор выполняет роль фазового декодера *QBPS* демодулятора.

Если выбрать фазовые состояния, например, $\varphi_k = -\frac{\pi}{4}$ и $\varphi_k = \frac{3\pi}{4}$, то в квадратурном демодуляторе необходимо использовать вычитание плеч

$$s_k = d_k - g_k = \cos(\varphi_k) - \sin(\varphi_k) = \sqrt{2} \sin(\pi/4 - \varphi_k) = \cos(\pi/4 + \varphi_k),$$

тогда при $\varphi_k = -\frac{\pi}{4}$ на выходе квадратурного демодулятора имеем значение

$$s_k = \sqrt{2} \sin(\pi/4 - \varphi_k) = \cos(\pi/4 + \varphi_k) = \sqrt{2},$$

при $\varphi_k = \frac{3\pi}{4}$ на выходе квадратурного демодулятора имеем значение

$$s_k = \sqrt{2} \sin(\pi/4 - \varphi_k) = \cos(\pi/4 + \varphi_k) = -\sqrt{2}.$$

Итого после нормировки амплитуд на выходе сумматора демодулятора имеем $s_k = \pm 1$.

В данном случае, при выборе фазовых состояний, $\varphi_k = -\frac{\pi}{4}$ и $\varphi_k = \frac{3\pi}{4}$, для значений битов ± 1 обычный вычитатель выполняет роль фазового декодера *QPS* демодулятора.

2 ОПИСАНИЕ ФУНКЦИОНАЛЬНЫХ SIM-МОДЕЛЕЙ QBPSK-МОДЕМА

Функциональная модель **QBPSK** модема с приёмником прямого преобразования (синхронного детектирования) представлена на рис. 2.1.

Источник биполярной информационной последовательности реализован на основе генератора псевдослучайной последовательности с гауссовским распределением **Random Number** и блока двухстороннего ограничителя на основе функции **sign(x)**.

В фазовом кодере **Phase Code** (см. рис. 2.2) задаются фазовые состояния для ± 1 , например, $\pi/4$ и $-3\pi/4$ либо $-\pi/4$ и $3\pi/4$ с помощью **MatLab**-функции **bit_phase_1** (см. рис. 2.3). От этих состояний φ_k вычисляются функции **cos**(φ_k) и **sin**(φ_k), которые являются модулирующими символами длиной τ_b .

Модулятор **QBPSK** сигнала реализован на основе блоков **Product**, на первые входы которых поступают модулирующие символы **cos**(φ_k) и **sin**(φ_k), а на вторые входы – опорные гармонические колебания **cos**($\omega_0 t$) и **sin**($\omega_0 t$) с частотой $\omega_0 = 15 \cdot \pi$ радиан/сек. На выходе квадратурного модулятора стоит блок **Sum**.

Простейшая модель канала распространения собрана на основе блока **Sum**, на второй вход которого поступает шумовая псевдослучайная последовательность с гауссовским распределением **Random Number**. Параметр генератора **Sample Time** позволяет реализовать необходимую широкополосность шумов канала распространения, а параметр **Variance** регулировать мощность шумов в процессе измерения помехоустойчивости (зависимости вероятности битовой ошибки от соотношения сигнал/шум **SNR**).

Полосовой фильтр с полосой пропускания $\Delta\omega = 4 \cdot \pi$ на основе блока **Analog Filter Design** призван отобразить формирующий фильтр на выходе передающей части и входную фильтрацию на входе приемной части.

QBPSK демодулятор представлен умножителями квадратурных каналов обработки. На первые входы умножителей **Product** разветвляется принятый зашумлённый радиосигнал, на вторые входы подаются синхронизированные квадратурные опорные колебания несущих **cos**($\omega_0 t$) и **sin**($\omega_0 t$) с генераторов **Sine Wave** с частотой $\omega_0 = 15 \cdot \pi$ радиан/сек. Система выделения и автоматической фазовой подстройки частоты опорного колебания в модели не представлена. На выходе умножителей установлен **ФНЧ (LF)** с полосой пропускания порядка $\Delta\omega = 2 \cdot \pi$ радиан/сек, для фильтрации высокочастотных продуктов преобразования. Блок **Gain = 2** призван компенсировать коэффициент тригонометрических преобразований равный $1/2$.

Как было показано в подразделе математического описания **QBPSK** модуляции роль фазового декодера при **QBPSK** модуляции, в зависимости от выбора фазовых состояний выполняет либо обычный сумматор, либо вычитатель.

Далее сигнал поступает на регенератор формы битов, состоящий из блока экстраполятора нулевого порядка **Zero Order Hold** и блока двухстороннего ограничителя на основе функции **sign(x)**.

Подсистема **Subsystem Calc Err**, блоки **Display** и **Scope** призваны отображать число ошибок передачи и фиксировать моменты их возникновения.

Подсистема **Subsystem Measuring Power** и блок **Scope** призваны вычислять и отображать изменение уровня мощности сигнала или смеси сигнал+шум в процессе измерения помехоустойчивости. Точка подключения измерителя мощности обусловлена блоком принятия решений, который соответствует блоку **Zero Order Hold**.

Блоки **Spectrum Scope** и **Averaging Power Spectral Density** призваны отображать спектры сигнальных и шумовых потоков.

QBPSK-Modem

$SNR=0.865/(0.995-0.865)=6.6538$ (8.2307 dB) $N_c=3.1$ $d_f=2^*\pi$ $d_F=2.0^*\pi$ $tb=1$ $n=1000$ $n_err=1$ classic BPSK
 $SNR=0.44/(0.485-0.44)=9.7778$ (9.9024 dB) $N_c=1.75$ $d_f=2^*\pi$ $d_F=2.0^*\pi$ $tb=1$ $n=1000$ $n_err=1$ QBPSK_1 chanal Cos
 $SNR=0.424/(0.488-0.424)=6.6250$ (8.2119 dB) $N_c=1.5$ $d_f=2^*\pi$ $d_F=2.0^*\pi$ $tb=1$ $n=1000$ $n_err=1$ QBPSK_1 chanal Sin
 $SNR=0.385/(0.425-0.385)=9.6250$ (9.8340 dB) $N_c=1.3$ $d_f=2^*\pi$ $d_F=1.2^*\pi$ $tb=1$ $n=1000$ $n_err=1$ QBPSK_1 chanal Sin
 $SNR=1.73/(1.9-1.73)=10.1765$ (10.0760 dB) $N_c=2.675$ $d_f=2^*\pi$ $d_F=2.0^*\pi$ $tb=1$ $n=1000$ $n_err=1$ QBPSK_1 duble ortog chanal sum
 $SNR=1.57/(1.75-1.57)=8.7222$ (9.4063 dB) $N_c=3.95$ $d_f=2^*\pi$ $d_F=1.2^*\pi$ $tb=1$ $n=1000$ $n_err=1$ QBPSK_1 duble ortog chanal sum

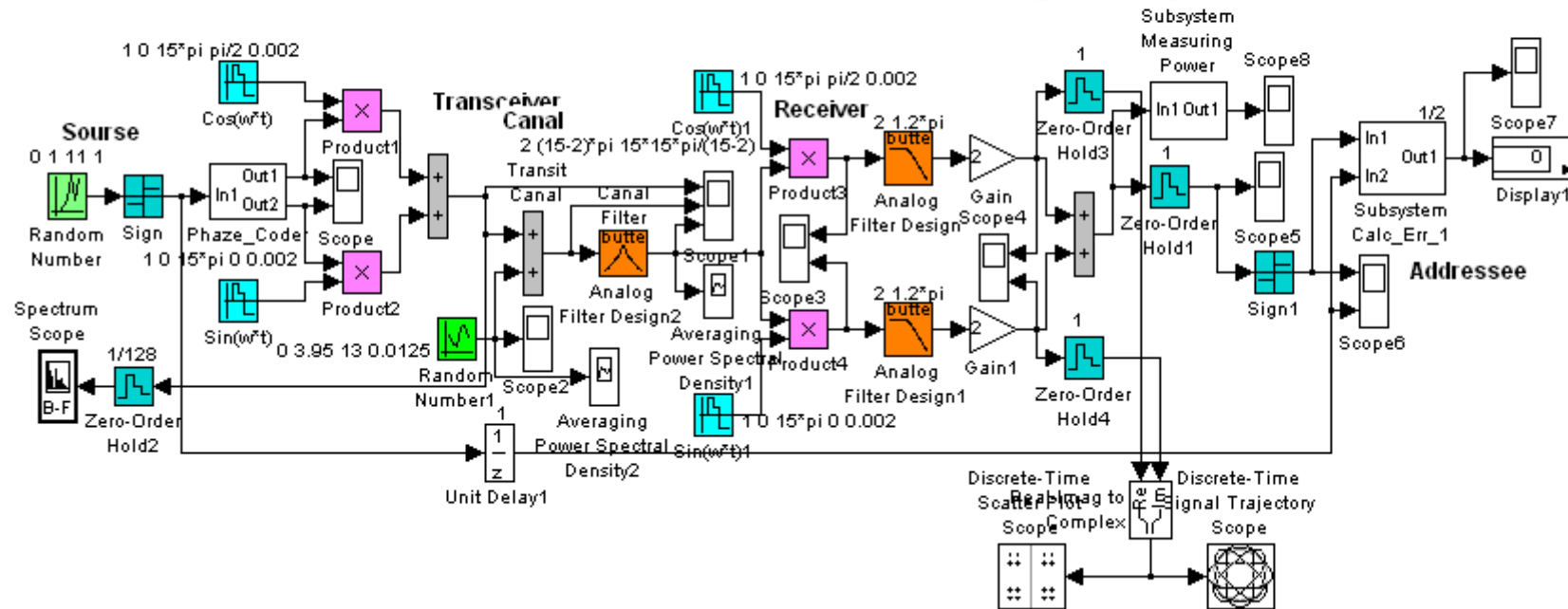


Рисунок 2.1 – Функциональная схема модели *QBPSK*-модема с приёмником прямого преобразования

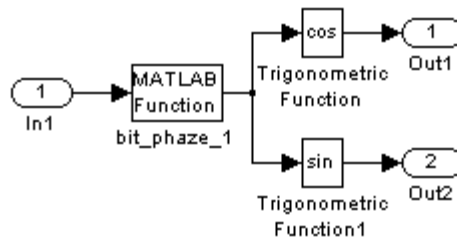


Рисунок 2.2 – Модель подсистемы фазового кодера *Phase Code*

```

function fi=bit_phase_1(x);
% Преобразование бита в фазу
% x- бит
% fi- фаза

if x==1; fi=pi/4; end;
if x==-1; fi=-3*pi/4; end;

```

Рисунок 2.3 – Скрин-файл *MatLab*-функции *bit_phase_1*

Функциональная модель *QBPSK* модема с корреляционным приёмником представлена на рис. 2.4.

При корреляционном приеме определяется значение функции корреляции между принимаемым и опорным сигналом. Функция корреляции является функцией параметра смещения τ и обычно функция автокорреляции симметрична относительно $\tau = 0$ и может содержать боковые пики. Функция корреляции непрерывного сигнала определяется соотношением:

$$R(\tau) = \int x(\tau) * y(t - \tau) d\tau,$$

где $y(t) = x(t) + n(t)$;
 $x(t)$ – полезный сигнал;
 $n(t)$ – АБГШ.

При полном совпадении сигналов будет получен положительный максимум значения функции корреляции. Если сигналы находились в противофазе, то максимум станет отрицательным. При отсутствии схожести между сигналами значение функции корреляции будет равно нулю. При частичном совпадении и при наличии помех значение функции корреляции находится в диапазоне от нуля до значения ее максимума. Для дискретного и цифрового сигнала функция корреляции определяется через сумму произведения функций отсчетов

$$R(m) = \lim_{N \rightarrow \infty} \frac{1}{N} \sum_{n=0}^{\infty} x(n) \cdot y(n - m).$$

При описании *QBPSK* модема с корреляционными приёмниками целесообразно остановиться на реализации корреляционного приема и обработке сигнала, поскольку передающие части модемов совпадают.

Для каждой квадратурной составляющей реализуется свой корреляционный приёмник. Плечи корреляционного приёмника состоят из экстраполяторов нулевого порядка

Zero Order Hold и блока **Buffer**, которые накапливают в течение бита отсчеты одного из синхронизированных опорных колебаний и принятого в шумах **QBPSK** модулированного радиосигнала. Накопленные отсчеты, как вектора скалярно перемножаются и суммируются блоками **Product** и **Sum**. Блок **Frame Status Conversion** призван преобразовать фреймовый тип данных в тип **double**. В данном случае реализуется накопление **64** отсчетов в течение бита. Как было показано в подразделе математического описания **QBPSK** модуляции роль фазового декодера при **QBPSK** модуляции, в зависимости от выбора фазовых состояний выполняет либо обычный сумматор, либо вычитатель. Так при выборе фазовых состояний, $\varphi_k = \frac{\pi}{4}$ и $\varphi_k = -\frac{3\pi}{4}$, для значений битов ± 1 обычный сумматор **Sum** выполняет роль фазового декодера. Блок **Sign** нормирует отсчеты к ± 1 и тем самым восстанавливает форму принятых битов.

Такая упрощенная реализация корреляционной обработки позволяет найти отсчет автокорреляционной функции опорного колебания и зашумленного радиосигнала при смещении $\tau = 0$ и в зависимости от знака корреляции принять решение о приеме **1** или **-1**.

QPSK-Modem

$$10 \cdot \log_{10}(64) = 18.0618 \text{ dB}$$

$SNR = 0.865 / (0.995 - 0.865) = 6.6538$ (8.2307 dB) $N_c = 3.1$ $df = 2 \cdot \pi$ $dF = 2.0 \cdot \pi$ $tb = 1$ $n = 1000$ $n_err = 1$ classic BPSK
 $SNR = 0.44 / (0.485 - 0.44) = 9.7778$ (9.9024 dB) $N_c = 1.75$ $df = 2 \cdot \pi$ $dF = 2.0 \cdot \pi$ $tb = 1$ $n = 1000$ $n_err = 1$ QPSK_1 chanal Cos
 $SNR = 0.424 / (0.488 - 0.424) = 6.6250$ (8.2119 dB) $N_c = 1.5$ $df = 2 \cdot \pi$ $dF = 2.0 \cdot \pi$ $tb = 1$ $n = 1000$ $n_err = 1$ QPSK_1 chanal Sin
 $SNR = 0.385 / (0.425 - 0.385) = 9.6250$ (9.8340 dB) $N_c = 1.3$ $df = 2 \cdot \pi$ $dF = 1.2 \cdot \pi$ $tb = 1$ $n = 1000$ $n_err = 1$ QPSK_1 chanal Sin
 $SNR = 1.73 / (1.9 - 1.73) = 10.1765$ (10.0760 dB) $N_c = 2.675$ $df = 2 \cdot \pi$ $dF = 2.0 \cdot \pi$ $tb = 1$ $n = 1000$ $n_err = 1$ QPSK_1 duble ortog chanal sum
 $SNR = 1.57 / (1.75 - 1.57) = 8.7222$ (9.4063 dB) $N_c = 3.95$ $df = 2 \cdot \pi$ $dF = 1.2 \cdot \pi$ $tb = 1$ $n = 1000$ $n_err = 1$ QPSK_1 duble ortog chanal sum
 $SNR_in = 3.81 / (3.81 - 0.5) = 1.1511$ (0.6110 dB) $N_c = 3.3$ $tb = 1$ $n = 1000$ $n_err = 1$ QPSK+Corr
 $SNR_out = 2045 / (2250 - 2045) = 9.9756$ (9.9894 dB) $N_c = 3.3$ $tb = 1$ $n = 1000$ $n_err = 1$ QPSK+Corr

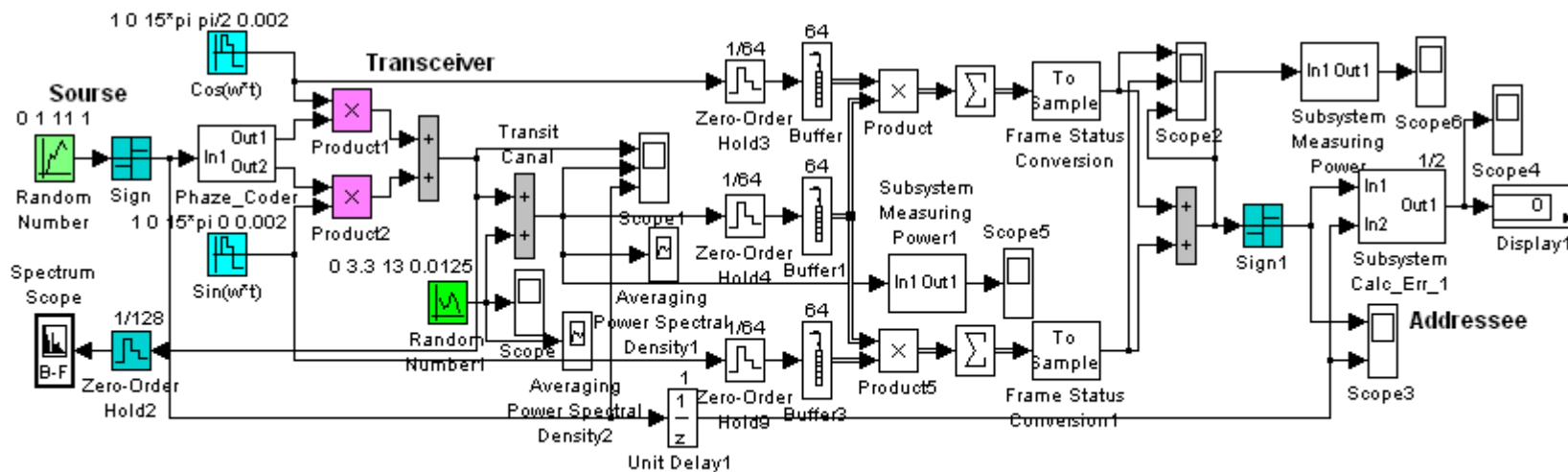


Рисунок 2.4 – Функциональная схема модели QPSK-модема с корреляционными приемниками

Модель подсистемы детектора ошибок *Subsystem Calc Err* приведена на рис. 2.5.

В детекторе ошибок вычисляется интеграл от модуля разности входных последовательностей, т.е. площадь разностного процесса. Затем, в зависимости от длительности битов и одно- или биполярности подбирается множитель *Gain*, переводящий интеграл разности в эквивалентное количество битов (ошибок).

На первый вход подается сигнал, прошедший через всю схему, на второй вход поступает сигнал с генератора входной информационной последовательности с необходимой задержкой. К выходу подсистемы присоединены дисплей и осциллограф, на дисплее в случае наличия ошибок, будет указано их количество, а на осциллографе можно фиксировать время их возникновения.

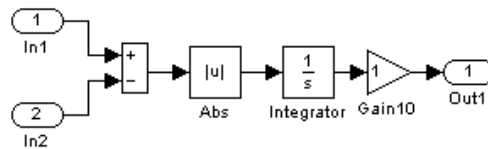


Рисунок 2.5 - Модель подсистемы детектора ошибок

Модель подсистемы измерителя мощности *Subsystem Measuring Power* приведена на рис. 2.6.

Подсистема позволяет измерить мощность как регулярных, так и случайных процессов, как вещественных, так и комплексных. С помощью блока *Dot Product* процесс умножается на сопряженный. Блоком *Complex to Real-Imag* выделяется вещественная часть. Блок *Integrator* вычисляет энергию процесса, а деление блоком *Product* энергии на время (блок *Clock*) вычисляет мощность как скорость поступления энергии.

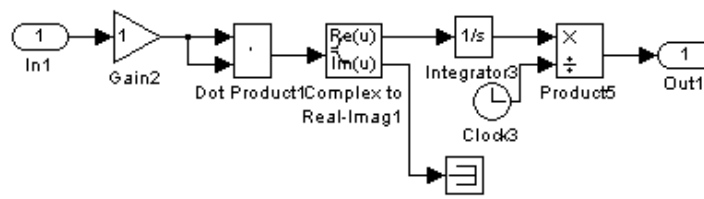


Рисунок 2.6 - Модель подсистемы измерителя мощности

Модель коррелятора. Корреляционный приемник реализован следующим образом (см. рис. 2.7):

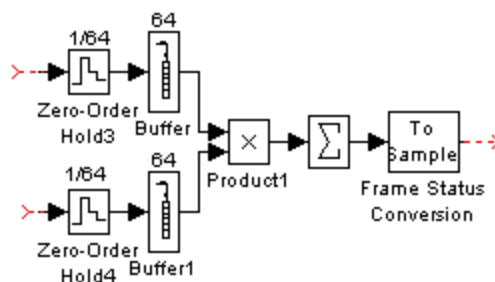


Рисунок 2.7 – Реализация корреляционного приемника

При реализации функциональной модели корреляционного приема решено ограничиться значением функции корреляции при $\tau = 0$.

Т.к. используются дискретные (цифровые) сигналы, то значение функции корреляции вычисляется через сумму произведений отсчетов принятого сигнала и колебаний опорного генератора соответствующей несущей. Модель коррелятора содержит два плеча накопления отсчетов. В каждом плече с помощью блоков *Zero-Order Hold* сигнал дискретизируется на отсчеты (в данном случае на **64**) и накапливается в блоках *Buffer*. Далее накопленные отсчеты попарно перемножаются в блоке *Product*. Затем вычисляется сумма этих произведений блоком *Sum*, т.е. реализуется скалярное произведение векторов накопленных отсчетов.

Измерение SNR. Отношение сигнал/шум (*SNR*) измеряется по следующей методике: для того чтобы определить мощность сигнала без шумовой составляющей **S** необходимо отсоединить генератор шума от модели канала распространения. Для того чтобы определить мощность смеси полезного сигнала с шумом **SN** генератор шумов должен быть подключен к каналу распространения. Изменение отношения сигнал/шум **SNR** достигается вариацией параметра дисперсии генератора псевдослучайной гауссовской последовательности модели канала распространения.

SNR рассчитывается по формуле, представленной ниже:

$$SNR = S / (SN - S)$$

Результаты измерений фиксируются в дБ.

При большом числе испытаний, частота появлений битовых ошибок, т.е. отношение числа ошибок к общему числу битов, стремится к вероятности битовых ошибок.

Рекомендации для проведения исследования помехоустойчивости. При отключенном генераторе шумов канала распространения на выходе преобразователя измеряется уровень мощности сигнала **S**. Подключив генератор шумов канала распространения, меняя параметр *Variance* добиваемся скажем **1**-ой ошибки на **1000** битов, получаем одну точку водопадоподобной кривой. Далее, постепенно увеличивая мощность шумов с помощью параметра *Variance*, добиваемся по очереди **3**-х, **5**-ти и **8**-ми ошибок, получаем **4**-е точки водопадоподобной кривой, которую строим в полулогарифмическом масштабе с помощью *MatLab*- функции *semilogy*.

Таким образом, измерение *SNR* при соответствующем числе ошибок позволяет определить точки водопадоподобной характеристики, позволяющей оценить помехоустойчивость модема.

3 ПРИНЦИП РАБОТЫ ФУНКЦИОНАЛЬНОЙ SIM-МОДЕЛИ QPSK-МОДЕМА

Опишем принцип работы **QPSK** модема с приёмником прямого преобразования по функциональной модели представленной на рис. 2.1. Источник на основе генератора **Random Number** и блока **Sign** формирует псевдослучайный информационный биполярный битовый поток.

В фазовом кодере **Phase Code** задаются фазовые состояния для ± 1 . От этих состояний φ_k вычисляются функции $\cos(\varphi_k)$ и $\sin(\varphi_k)$, которые являются модулирующими символами длиной τ_b .

Модулятор **QPSK** сигнала реализован на основе блоков **Product**, на первые входы которых поступают модулирующие символы, а на вторые входы – опорные гармонические колебания $\cos(\omega_0 t)$ и $\sin(\omega_0 t)$ с частотой $\omega_0 = 15 \cdot \pi$ радиан/сек. На выходе квадратурного модулятора стоит сумматор.

Модель канала распространения на основе блока **Sum**, на второй вход которого поступает шумовая псевдослучайная последовательность с гауссовским распределением **Random Number** с параметром **Sample Time** и **Variance** регулирующем мощность шумов в процессе измерения помехоустойчивости.

Полосовой фильтр на основе блока **Analog Filter Design** с полосой пропускания $\Delta\omega = 4 \cdot \pi$ соответствует формирующему фильтру на выходе передающей части и входному фильтру на входе приемной части.

Демодулятор **QPSK** представлен умножителями квадратурных каналов обработки. На первые входы умножителей **Product** разветвляется принятый зашумлённый радиосигнал, на вторые входы подаются синхронизированные квадратурные опорные колебания несущих с частотой $\omega_0 = 15 \cdot \pi$ радиан/сек. На выходе умножителей установлен **ФНЧ** с полосой пропускания порядка $\Delta\omega = 2 \cdot \pi$ радиан/сек, для фильтрации высокочастотных продуктов преобразования. Блок **Gain = 2** призван компенсировать коэффициент тригонометрических преобразований равный $1/2$.

Как было показано в подразделе математического описания **QPSK** модуляции роль фазового декодера при **QPSK** модуляции, в зависимости от выбора фазовых состояний выполняет либо обычный сумматор, либо вычитатель.

Далее сигнал поступает на регенератор формы битов, состоящий из блока экстраполятора нулевого порядка **Zero Order Hold** и блока двухстороннего ограничителя на основе функции $\text{sign}(x)$.

Блок **Scope**, установленный на выходе, позволяет наблюдать и сравнивать принятый информационный поток и подаваемый на второй вход передаваемый информационный поток.

Опишем принцип работы **QPSK** модема с корреляционными приёмниками по функциональной модели представленной на рис. 2.4. Поскольку передающая часть модемов совпадает, рассмотрим подробнее корреляционные приёмники. Корреляционные приёмники работают с отсчетами в пределах каждого бита.

Для каждой квадратурной составляющей реализуется свой корреляционный приёмник. Плечи корреляционного приёмника накапливают в течение бита отсчеты одного из синхронизированных опорных колебаний и принятого в шумах **QPSK** модулированного радиосигнала. Накопленные отсчеты, как вектора скалярно перемножаются и суммируются блоками **Product** и **Sum**. В данном случае реализуется накопление **64** отсчетов в течение бита. Как было показано в подразделе математического описания **QPSK** модуляции роль фазового декодера при **QPSK** модуляции, в зависимости от выбора фазовых состояний выполняет либо обычный сумматор, либо вычитатель. Так при выборе фазовых состояний, $\varphi_k = \frac{\pi}{4}$ и $\varphi_k = -\frac{3\pi}{4}$, для значений битов ± 1 обычный сумматор **Sum** выполняет роль фазового декодера. Блок **Sign** нормирует отсчеты к ± 1 и тем самым восстанавливает форму принятых

битов.

Подразумевается, что генератор опорного колебания приёмника синхронизирован с генератором опорного колебания передатчика системой фазовой автоподстройки частоты ФАПЧ с точностью до фазы. Результат этого скалярного произведения накопленных векторов представляет собой отсчет корреляционной функции при смещении $\tau = 0$. В зависимости от знака принимаемого бита и шумов канала распространения значение отсчета корреляционной функции может быть как положительным так и отрицательным. Применение блока двухстороннего ограничителя на основе функции $sign(x)$ нормирует принятые значения к ± 1 , что эквивалентно восстановлению биполярной формы принятого информационного потока.

Измерение отношения сигнал/шум SNR может быть осуществлено как на входе так и на выходе коррелятора, что позволяет оценить влияние коррелятора на SNR в зависимости от конкретных условий.

Для иллюстрации работы $QBPSK$ модема на рис. 3.1 приведены фрагменты осциллограмм $QBPSK$ модулированного сигнала, зашумленного передаваемого сигнала и сигнала прошедшего полосовой фильтр.

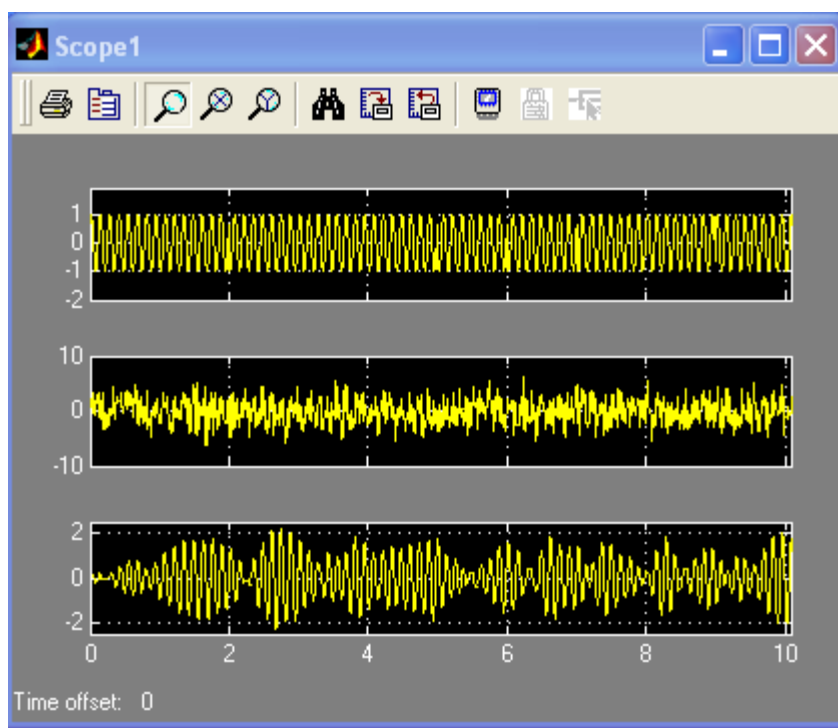


Рисунок 3.1 - Фрагменты осциллограмм $QBPSK$ модулированного сигнала, зашумленного передаваемого сигнала и сигнала прошедшего полосовой фильтр

На рисунке 3.2 приведен фрагмент осциллограммы квадратурных составляющих сигнала принятых в шумах после ФНЧ демодулятора.

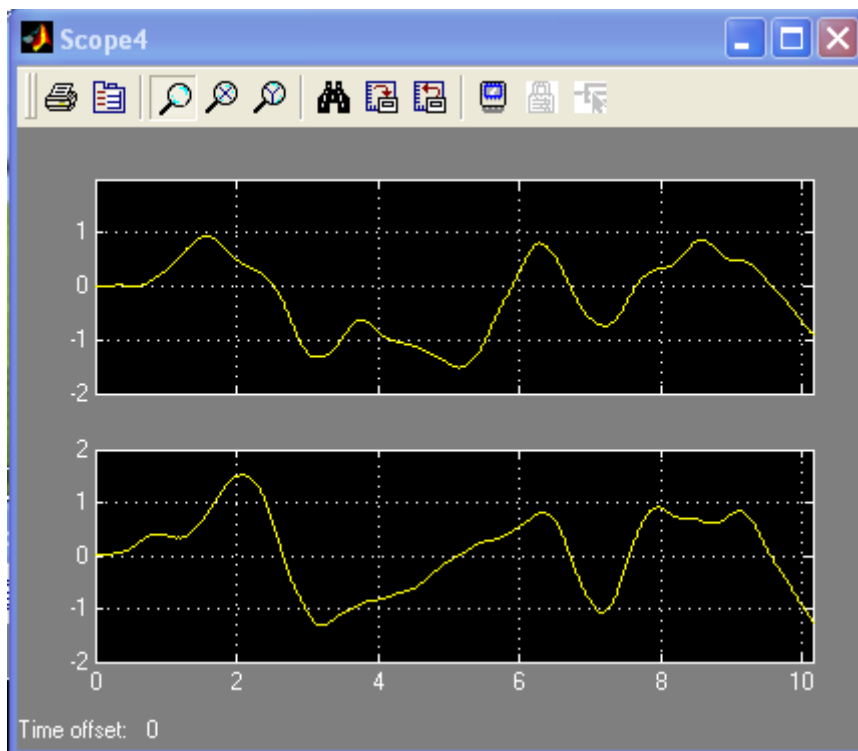


Рисунок 3.2 - Фрагмент осциллограммы квадратурных составляющих сигнала принятых в шумах после ФНЧ демодулятора

На рисунке 3.3 приведены фрагменты осциллограмм принятого и передаваемого информационных потоков.

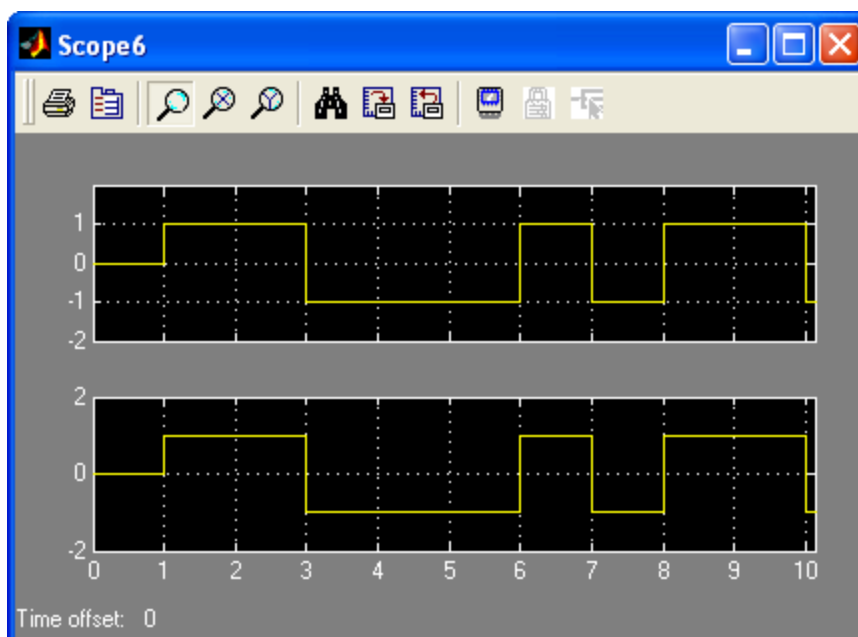


Рисунок 3.3 - Фрагменты осциллограмм принятого и передаваемого информационных потоков в отсутствии шумов

На рисунке 3.4 приведена диаграмма фазовых переходов в приёмнике *QBPSK* модема в отсутствии шумов канала распространения.

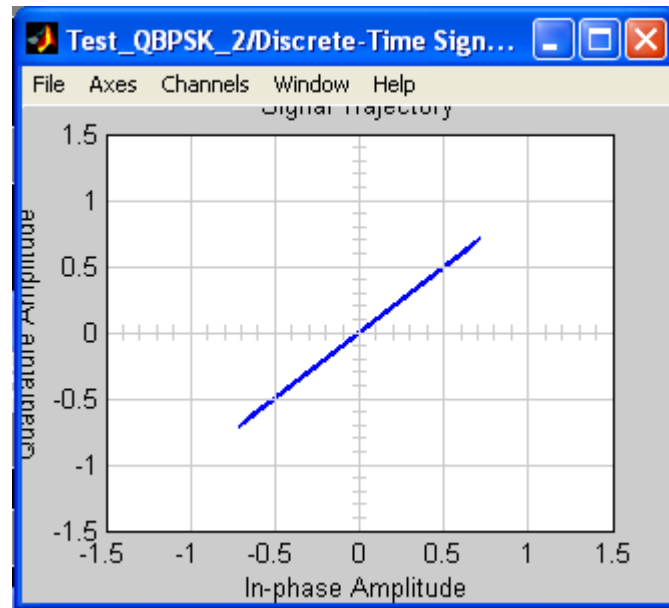


Рисунок 3.4 - Диаграмма фазовых переходов в приёмнике *QBPSK* модема в отсутствии шумов канала распространения

На рисунке 3.5 приведена спектрограмма *QBPSK* модулированного сигнала в отсутствие шумов.

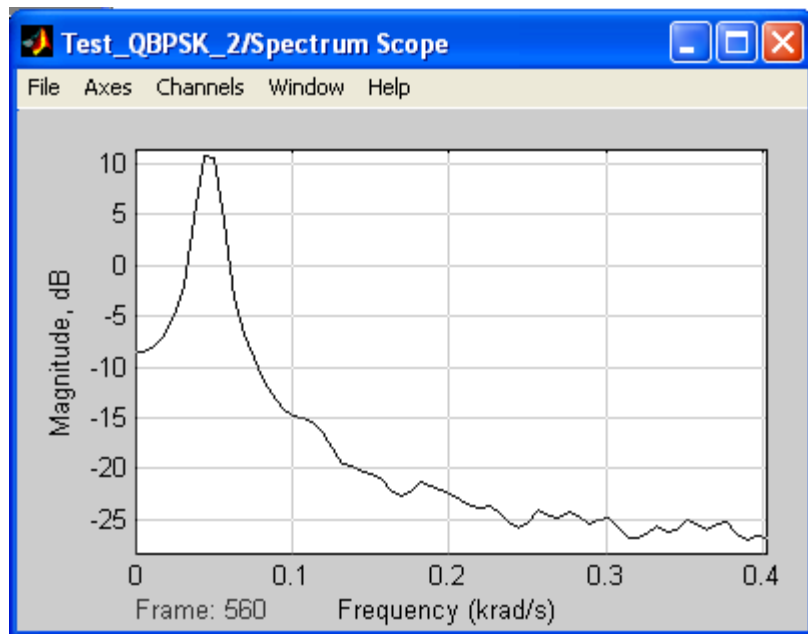


Рисунок 3.5 - Спектрограмма *QBPSK* модулированного сигнала в отсутствие шумов

На рисунке 3.6 приведен фрагмент осциллограмм на выходе корреляторов перед и после суммирования в присутствии шумов.

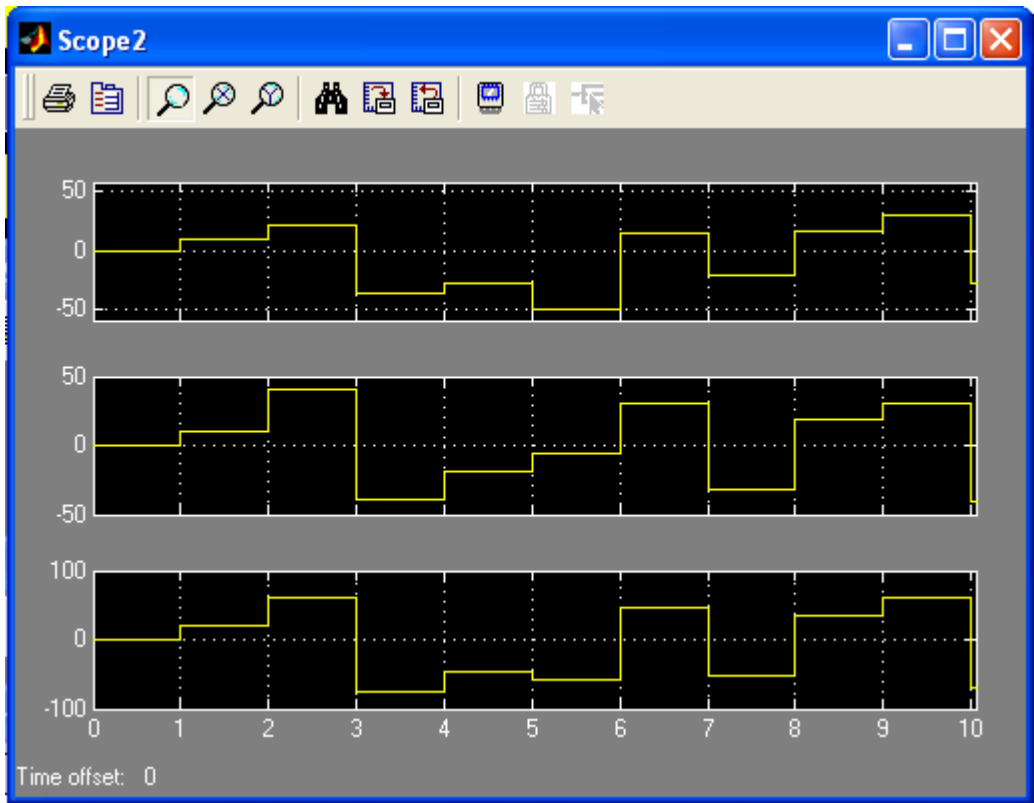


Рисунок 3.6 - Фрагмент осциллограмм на выходе корреляторов перед и после суммирования в присутствии шумов


Приведённые осциллограммы модельного исследования *QBPSK* модема призваны ориентировать студента в процессе выполнения лабораторной работы.

4 КРАТКОЕ ОПИСАНИЕ ПАКЕТА SIMULINK И ИСПОЛЬЗУЕМЫХ БЛОКОВ

Пакет *Simulink* разработан компанией *Mathworks* и распространяется в составе математического пакета *MatLab*. Пакет основан на графическом интерфейсе и является типичным средством визуально-ориентированного программирования. Он обладает обширной библиотекой готовых блоков с модифицируемыми параметрами для построения моделей рассматриваемых систем и наглядными средствами визуализации результатов моделирования [4 – 7].

4.1 Запуск и работа с пакетом *Simulink*

Для запуска пакета *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*) (рисунок 4.1).

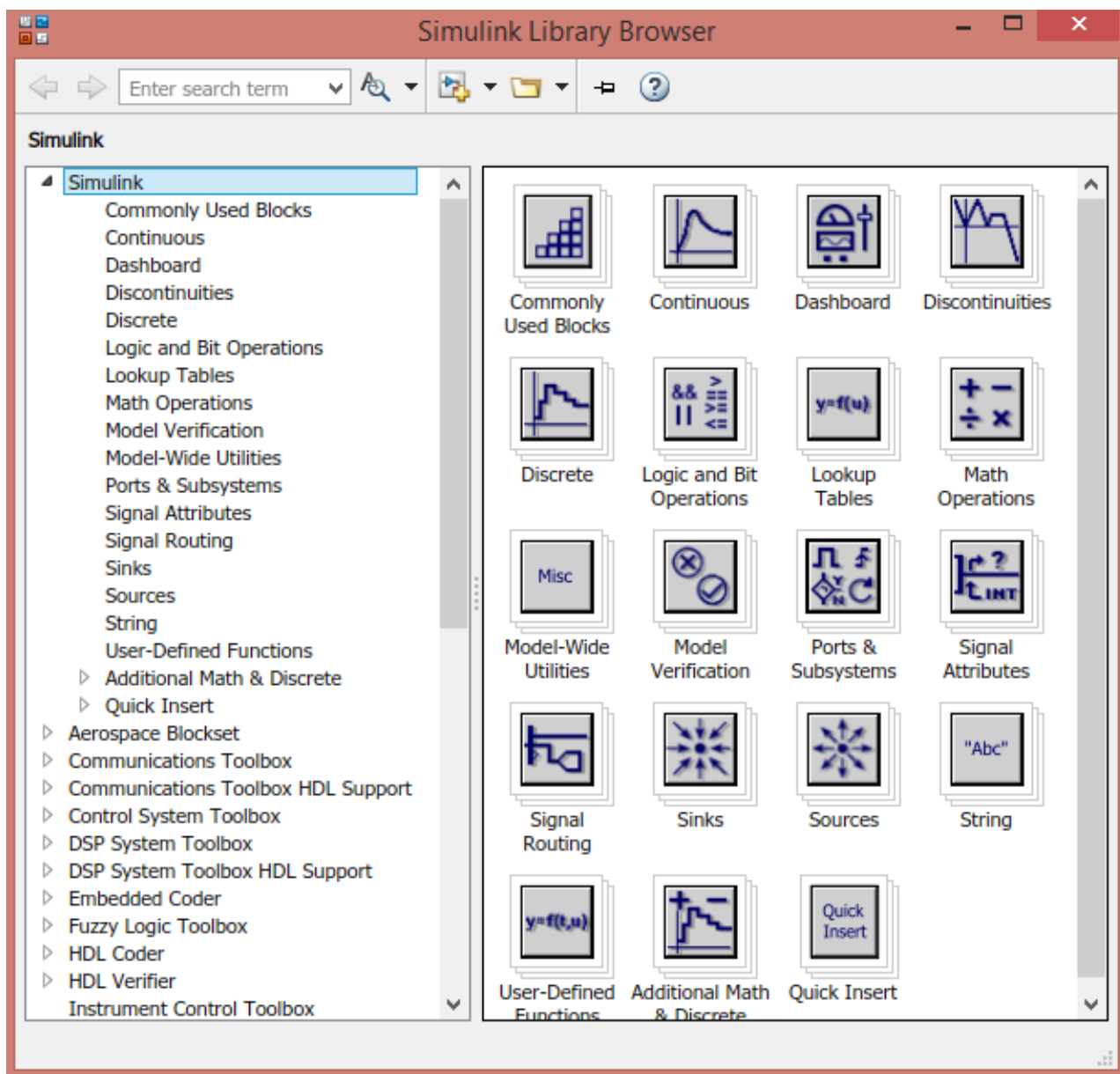


Рисунок 4.1 – Библиотека блоков *Simulink Library Browser*

На рисунке 4.1 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 маршрутизации сигналов;
- **Signal Routing** – блоки маршрутизации сигналов;
- **Sinks** – блоки приема и отображения сигналов;
- **Sources** – блоки источников сигнала;
- **User-Defined Function** – функции, определяемые пользователем.

4.2 Описание используемых блоков библиотеки *Simulink*

Ниже описаны основные блоки базовых разделов библиотеки *Simulink*, используемые в функциональной схеме **OQPSK**-модема:



Random Number

Random Number – блок источника случайного дискретного сигнала с нормальным распределением. Назначение: формирование случайного сигнала с нормальным распределением уровня сигнала. Параметры блока: **Mean** - среднее значение сигнала, **Variance** - дисперсия (среднеквадратическое отклонение), **Initial seed** – начальное значение.

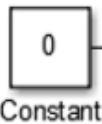


Sign

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

$$\text{sign} = \begin{cases} -1, & \text{где } x < 0; \\ 0, & \text{где } x = 0; \\ 1, & \text{где } x > 0. \end{cases}$$

Параметры блока: флажок **Enable zero crossing detection** позволяет фиксировать прохождение сигнала через нулевой уровень.



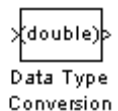
Constant

Constant – блок источника постоянного сигнала. Назначение: задает постоянный по уровню сигнал. Параметры: **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.



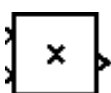
Relational Operator

Relational Operator – блок вычисления операции отношения. Назначение: блок сравнивает текущие значения входных сигналов. Параметры: **Relational Operator** – тип операции отношения (выбирается из списка): «= =» – тождественно равно, «~ =» – не равно, «<» – меньше, «< =» – меньше или равно, «> =» – больше или равно, «>» – больше.



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 u boolean. Saturate on integer overflow** (флажок) – подавляет переполнение целого. При установленном флажке ограничение сигналов целого типа выполняется корректно.



Product

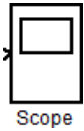
Product – блок умножения и деления. Назначение: вычисление произведения текущих значений сигналов. Параметры блока: **Number of inputs** – количество входов, может задаваться как число или как список знаков. В списке знаков можно использовать знаки: * – умножить и / – разделить. **Multiplication** – 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

Sine Wave – блок источника синусоидального сигнала. Назначение:

формирование синусоидального сигнала с заданной частотой, амплитудой, фазой и смещением. Параметры блока: **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.



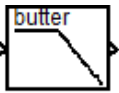
Scope

Scope – блок осциллографа. Назначение: построение графиков исследуемых сигналов как функций времени. Открытие окна осциллографа производится двойным щелчком ЛКМ на пиктограмме блока. В случае векторного сигнала каждая компонента вектора отображается отдельным цветом. 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.



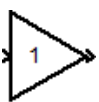
Add

Add – блок сумматора. Назначение: вычисление алгебраической суммы текущих значений входных сигналов. Параметры блока: **Icon shape** – форма блока, выбирается из списка: **round** – круг; **rectangular** – прямоугольник. **List of sign** – 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**; подраздела **Filtering**, раздела **DSP Blockset**. 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: **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: **Filter order** – порядок фильтра; **Passband edge frequency (rads/sec)** – нижняя граничная частота (радиан в секунду). Для других методов проектирования и типов фильтров определяемые параметры очевидны.



Gain

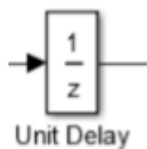
Gain – блок усилителя. Назначение: блок **Gain** умножает входной сигнал на постоянный коэффициент; Параметры блока: **Multiplication** – способ выполнения операции, значение параметра выбирается из списка: **Element-wise $K*u$** – поэлементный; **Matrix $K*u$** – матричный, коэффициент усиления является 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.



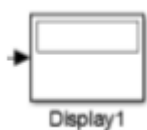
Zero-Order Hold

Zero-Order Hold – экстраполятор нулевого порядка. Назначение:

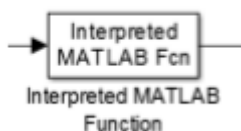
экстраполяция входного сигнала на интервале дискретизации. Блок фиксирует значение входного сигнала в начале интервала дискретизации и поддерживает на выходе это значение до окончания интервала дискретизации. Затем выходной сигнал изменяется скачком до величины входного сигнала на следующем шаге дискретизации.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



Unit delay – блок единичной дискретной задержки. Назначение: выполняет задержку дискретного сигнала на заданный шаг модельного времени. Параметры блока: **Initial conditions** – начальное значение выходного сигнала; **Sample time** – шаг модельного времени.

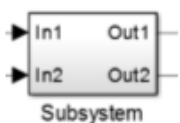


Display – блок цифрового дисплея. Назначение: отображает значение сигнала в виде числа. Параметры: **Format** – формат отображения данных. Параметр **Format** может принимать следующие значения: **short** – 5 значащих десятичных цифр, **long** – 15 значащих десятичных цифр, **short_e** – 5 значащих десятичных цифр и 3 символа степени десяти, **long_e** – 15 значащих десятичных цифр и 3 символа степени десяти, **bank** – "денежный" формат. Формат с фиксированной точкой и двумя десятичными цифрами в дробной части числа; **Decimation** – кратность отображения входного сигнала, при **Decimation = 1** отображается каждое значение входного сигнала, при **Decimation = 2** отображается каждое второе значение, при **Decimation = 3** – каждое третье значение и т.д.; **Sample time** – шаг модельного времени. Определяет дискретность отображения данных; **Floating display** (флажок) – перевод блока в "свободный" режим. В данном режиме входной порт блока отсутствует, а выбор сигнала для отображения выполняется щелчком ЛВМ на соответствующей линии связи. В этом режиме для параметра расчета **Signal storage reuse** должно быть установлено значение **off** (вкладка **Advanced** в окне диалога **Simulation parameters...**).



MatLab Fcn – блок задания функции. Назначение: задает выражение в стиле языка программирования **MatLab**. Параметры: **MatLab function** – Выражение на языке **MatLab**. **Output dimensions** – размерность выходного сигнала. Значение параметра минус 1 предписывает блоку определять размерность автоматически. **Output signal type** – тип выходного сигнала.

Выбирается из списка: **real** – действительный сигнал, **complex** – комплексный сигнал, **auto** – автоматическое определение типа сигнала; **Collapse 2-D results to 1-D** – преобразование двумерного выходного сигнала к одномерному.



Subsystem – виртуальная и монолитная подсистемы. Доступ к окну параметров подсистемы осуществляется через меню **Edit** командой **Block Parameters**. Параметры: **Show port labels** – показать метки портов, **Treat as atomic unit** (флажок) – считать подсистему монолитной. Таким образом, блоки виртуальной и монолитной подсистем – это один и тот же блок,

отличающийся значением данного параметра. **Access** – доступность подсистемы для изменений. Выбирается из списка: **ReadWrite** – пользователь может открывать и изменять подсистему, **ReadOnly** – пользователь может открывать подсистему только для просмотра, **NoReadOrWrite** – пользователь не может открывать и изменять подсистему; **Name of error callback function** – имя функции используемой для обработки ошибок возникающих в данной подсистеме.



Переключатель. Назначение: Выполняет переключение входных сигналов по сигналу управления.

Параметры блока: **Criteria for passing first input:** Значение параметра выбирается из списка:

U2 >= Threshold – входной сигнал больше или равен пороговому значению;

$U2 > Threshold$ – входной сигнал больше порогового значения;
 $U2 \sim Threshold$ – входной сигнал не равен пороговому значению.
 $Threshold$ – порог.



Pulse generator – блок источника импульсного сигнала. *Назначение:* формирование сигнала в форме прямоугольных импульсов. *Параметры блока:* **Pulse Type** – способ формирования сигнала, может принимать два значения: **Time-based** – по текущему времени; **Sample-based** – по величине такта дискретности и количеству шагов моделирования. Вид окна параметров зависит от выбранного способа формирования сигнала. **Amplitude** – амплитуда; **Period** – период, задается в секундах при способе **Time-based** или количеством тактов при способе **Sample-based**; **Pulse width** – ширина импульса, задается в процентах от периода при способе **Time-based** или количеством тактов при способе **Sample-based**; **Phase delay** – фазовая задержка, задается в секундах при способе **Time-based** или количеством тактов при способе **Sample-based**; **Sample time** – такт дискретности; флажок **Interpret vector parameters as 1 - D** – интерпретировать вектор как массив скаляров.



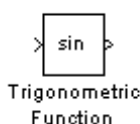
Источник времени. *Назначение:* Формирует сигнал, величина которого на каждом шаге равна текущему времени моделирования. *Параметры блока:* **Display time:** Включение отображения времени на пиктограмме; **Decimation:** Шаг обновления времени.



Блок скалярного произведения. *Назначение:* Выполняет вычисление скалярного произведения двух векторов. *Параметров блока нет.* Блок выполняет вычисление выходного сигнала в соответствии с выражением

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

где $u1$ и $u2$ – входные векторы; conj – операция комплексного сопряжения; sum – операция суммирования.



Trigonometric Function- тригонометрическая функция. *Назначение:* вычляет выбранную тригонометрическую функцию. *Параметры блока:* **Function-** вид вычисляемой функции. Вид функции выбирается из списка: **sin, cos, tan, asin, acos, atan, atan2, sinh, cosh, tanh.** **Output signal type-** тип выходного сигнала. Тип выходного сигнала выбирается из списка: **auto-** автоматическое определение типа, **real-** действительный сигнал, **complex-** комплексный сигнал. При векторном или матричном входном сигнале блок выполняет поэлементное вычисление заданной функции.



Блок вычисления комплексного числа по его действительной и мнимой части. *Параметры блока:*

Input: Значение параметра выбирается из списка:

Real – действительная часть;

Image – мнимая часть;

Real&Imag – действительная и мнимая часть.



Блок вычисления действительной и/или мнимой части комплексного числа. *Параметры блока:*

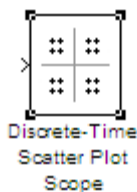
Output: Тип сигнала выбирается из списка:

Real – действительная часть;

Image – мнимая часть;

Real&Imag – действительная и мнимая часть.

Входной сигнал блока может быть скалярным, векторным или матричным.



Discrete-Time Scatter Plot Scope – блок отображения диаграммы рассеяния фазовых состояний сигнала из подраздела **Comm Sinks** раздела **Communication Blockset** библиотеки **Simulink** [6]. *Назначение:* отображение диаграммы рассеяния фазовых состояний за счет влияния полосы пропускания и помех тракта передачи. *Параметры блока:* Флажок **Show Plotting Properties** – показать графические установки; **Samples per Symbol** – шаг периода символа; **Offset (samples)** –

смещение шагов; **Points displayed** – число отсчетов сигнала, начиная с которого отображается диаграмма; **New points per display** – число отсчетов при обновлении отображения; флажки **Show Rendering Properties**, **Show Axes Properties**, **Show Figure Properties** – показать свойства отображения, осей и фигуры в нашем случае не используются.



Discrete-Time Signal Trajectory Scope – блок отображения диаграммы переходов фазовых состояний сигнала из подраздела **Comm Sinks** раздела **Communication Blockset** библиотеки **Simulink** [6]. *Назначение:* отображение диаграммы рассеяния фазовых состояний за счет влияния полосы пропускания и помех тракта передачи. *Параметры блока:* Флажок **Show Plotting Properties** – показать графические установки; **Samples per Symbol** – шаг периода символа; **Offset (samples)** –

смещение шагов; **Points displayed** – число отсчетов сигнала, начиная с которого отображается диаграмма; **New points per display** – число отсчетов при обновлении отображения; флажки **Show Rendering Properties**, **Show Axes Properties**, **Show Figure Properties** – показать свойства отображения, осей и фигуры в нашем случае не используются.

В нашем случае диаграммы блоков **Discrete-Time Scatter Plot Scope** и **Discrete-Time Signal Trajectory Scope** позволяют оценить влияние метода модуляции, полосы пропускания и шумов канала передачи на фазовую диаграмму состояний принятого сигнала.

5 ЭКСПЕРИМЕНТАЛЬНОЕ ЗАДАНИЕ

Исходные данные.

Для исследования *QBPSK* модема устанавливаем в генераторе *Random Number* с помощью параметра *Sample Time* длительность бита информационного потока $\tau_b = 1$, для обеспечения широкополосности шума в генераторе шума канала распространения параметр *Sample Time* задать равным *0.0125*; частоты опорных гармонических генераторов выставить равными $\omega_0 = 15 \cdot \pi$ радиан/сек; для корректной работы блока *Averaging Power Spectral Density* следующие параметры: *Length of buffer* – 128, *Number of points for fft* – 512, *Plot after how many points* – 64, *Sample time* – 0.05. Все основные параметры блоков, в качестве подсказок указаны на рис. 2.1 и 2.4 над блоками.

Экспериментальное задание.

1. Собрать *Sim*-модели *QBPSK* модемов с приёмником прямого преобразования и корреляционным приёмником, показанные на рисунках 2.1 и 2.4. Набрать текст *MatLab*-функции *bit_phase_1* (рис. 2.3). Параметры моделей задать, как показано на рис. 2.1 и 2.4. Отладку моделей вести при отключенных шумах канала распространения.

2. Определить при каком уровне шума появляется вероятность появления битовой ошибки равная *0.001* для исследуемых моделей. По полученным данным рассчитать *SNR* (в *dB*). Длину информационного потока установить порядка *1000* битов. Данные исследования занести в отчет.

3. Сравнить результаты измерения помехоустойчивостей (*SNR*) *QBPSK* и *BPSK* модемов и сделать выводы. (Справка: помехоустойчивость *BPSK* модема при вероятности битовых ошибок $P_b = 10^{-3}$ составляет *SNR=8.23 dB*).

4. Составить отчёт по проделанной работе (цели, задачи, функциональные схемы, принцип работы, наиболее важные осциллограммы и спектрограммы, выводы).

6 КОНТРОЛЬНЫЕ ВОПРОСЫ

1. Принцип *BPSK QPSK* манипуляции (модуляции).
2. Принцип приёмника прямого преобразования (синхронного детектирования).
3. Принцип работы корреляционного приёмника.
4. Устройство *QPSK* модулятора.
5. Назначение ФНЧ (**LF**) *QPSK* демодулятора.
6. Назначение полосового фильтра в *QPSK* модеме с приёмником прямого преобразования.
7. Как реализуется фазовый кодер *QPSK* модулятора?
8. Как реализуется фазовый декодер *QPSK* демодулятора?
9. От чего зависит вероятность появления битовых ошибок?
10. Какова помехоустойчивость *QPSK* модуляции среди *MPSK* модуляций?
11. Какова спектральная эффективность *QPSK* модуляции среди *MPSK* модуляций?
12. Чем определяется полоса пропускания псевдослучайного битового потока?

СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ

1. Муравьёв В.В. Полосовая модуляция в системах телекоммуникаций: учеб.-метод. пособие / В.В. Муравьёв, С.А. Корневский, Т.М. Печень. – Минск: БГУИР, 2019.- 79 с.
2. Леонидов В.В. Конспект лекций «Модуляция и демодуляция цифровых сигналов». Учебно-методический комплект по дисциплине «Цифровая обработка». МГТУ имени Н.Э. Баумана. [Электронный ресурс]. – Режим доступа: <https://leonidov.su/wp-content/uploads/2020/04/Modulation-and-Demodulation-of-Digital-Signals-Lecture-V.V.-Leonidov.pdf> (дата обращения 09.09.2022).
3. Сигналы с двоичной фазовой манипуляцией (BPSK). [Электронный ресурс]. Режим доступа: https://ru.dsplib.org/content/signal_bpsk/signal_bpsk.html (дата обращения 14.09.2022).
4. Гультяев А.К. MatLab 5.3. Имитационное моделирование в среде Windows: Практическое пособие / А.К. Гультяев – СПб.: КОРОНА принт, 2001.– 400 с.
5. Черных И.В. Simulink: среда создания инженерных приложений. / Под общ. ред. В.Г. Потемкина – М.: ДИАЛОГ-МИФИ, 2003.– 496 с.
6. Дьяконов В.П. MatLab 6.5 SP1/7 + Simulink 5/6. Основы применения. Сер. Библиотека профессионала / В. П. Дьяконов - М.: СОЛОН-Пресс, 2005.– 800 с.
7. 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.