

Квадратурная фазовая манипуляция

12.1 Цель

Все системы модуляции, рассматриваемые до сих пор, выполняли кодирование одного бита данных в один символ. Для приложений, требующих высоких скоростей передачи данных при ограниченной полосе пропускания канала, рекомендуется в одном символе кодировать несколько бит данных. В этой лабораторной работе мы рассмотрим квадратурную фазовую манипуляцию (QPSK, quadrature phase-shift keying), вариант двоичной фазовой манипуляции, которая кодирует два бита в каждом символе. Используя QPSK, мы сможем передавать данные с удвоенной скоростью без расширения пропускной способности канала. Однако это приведет к некоторой потере производительности, а также потребует большей мощности передатчика для поддержания заданной вероятности битовой ошибки при наличии шума.

В данной лабораторной работе рассматриваются новые принципы кодирования сигналов, более сложные, чем кодирование, используемое в BPSK. В этой работе будет также рассмотрено сигнальное созвездие, которое является инструментом визуализации, очень полезным для понимания высокоэффективных методов модуляции.

QPSK просто расширяется на методы модуляции, которые позволяют передавать даже более двух бит в одном символе. Одним из таких методов является "квадратурная амплитудная модуляция" (QAM, quadrature amplitude modulation). Например, 16-QAM может передавать 4 бита данных в одном символе. Однако, большее количество бит, передаваемых в одном символе, требует большей мощности передатчика для поддержания заданной вероятности битовой ошибки.

12.2 Вводная информация

В лабораторной работе, в которой рассматривалась BPSK, PSK сигнал представлял собой последовательность импульсов следующей формы

$$x(t) = A g_{TX}(t) \cos(2\pi f_c t + \theta), \quad (1)$$

где A - константа, $g_{TX}(t)$ - импульс определенной формы, предназначенный для ограничения полосы частот сигнала и управления межсимвольными искажениями, f_c - частота несущей.

Информация передается фазой θ . В BPSK достаточно было взять два значения фазы θ , при которых каждый импульс содержал один бит информации. Для QPSK необходимо взять четыре значения фазы θ . Поскольку для различения одного из четырех состояний необходимо два бита, каждый импульс QPSK несет в себе два бита информации. Мы будем использовать четыре возможных фазовых угла $\theta = \pm\pi/4, \pm 3\pi/4$.

Другой способ представления импульсов дает уравнение разложения по косинусу (1). Можно записать

$$x(t) = A \cos(\theta) g_{TX}(t) \cos(2\pi f_c t) - A \sin(\theta) g_{TX}(t) \sin(2\pi f_c t). \quad (2)$$

Подстановка четырех возможных значений θ дает

$$x(t) = \pm (A/\sqrt{2}) g_{TX}(t) \cos(2\pi f_c t) \mp (A/\sqrt{2}) g_{TX}(t) \sin(2\pi f_c t). \quad (3)$$

Уравнение (3) можно интерпретировать, предположив, что один бит данных определяет полярность $(A/\sqrt{2}) g_{TX}(t) \cos(2\pi f_c t)$ (синфазный терм), в то время как второй бит данных определяет полярность $(A/\sqrt{2}) g_{TX}(t) \sin(2\pi f_c t)$ (квадратурный терм).

Чтобы создать передаваемый импульс, представленный уравнениями (1) или (3), мы должны передать в USRP импульс в комплексной форме

$$\tilde{x}(t) = A g_{TX}(t) e^{j\theta}, \quad \theta = -3\pi/4, -\pi/4, \pi/4, 3\pi/4. \quad (4)$$

В этом случае символ является комплексным числом $Ae^{j\theta}$. USRP будет генерировать передаваемый сигнал, выраженный уравнением (1):

$$x(t) = \text{Re}[\tilde{x}(t) e^{j2\pi f_c t}] = A g_{TX}(t) \cos(2\pi f_c t + \theta). \quad (5)$$

Удобный способ визуализации сигнала в комплексной форме, представленной формулой (4), - нарисовать возможные значения символов $Ae^{j\theta}$ на комплексной плоскости. Сигнальное созвездие QPSK показано на Рисунок 1.

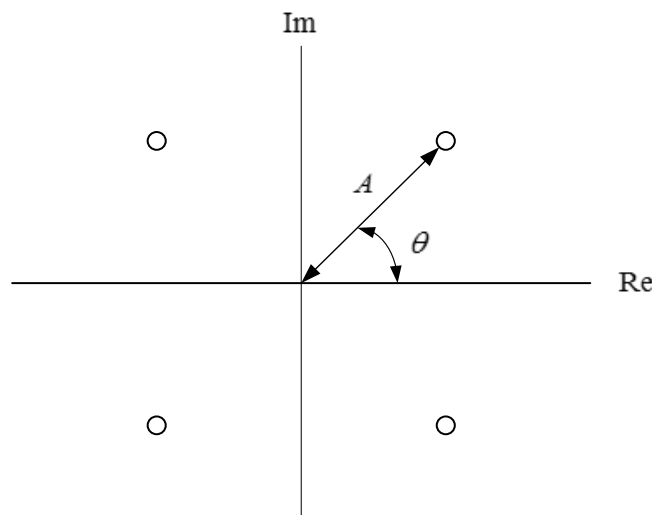


Рисунок 1. Сигнальное созвездие QPSK

Из сигнального созвездия легко увидеть, что существуют четыре возможных передаваемых сигнала (отсюда два бита на символ), каждый из возможных символов имеет одну и ту же амплитуду A , и возможны четыре фазовых угла $\theta = \pm\pi/4, \pm3\pi/4$.

Давайте проверим связь между парой входных бит и точками на сигнальном созвездии (т.е. значениями символа). Отображение пары бит точкой на сигнальном созвездии фактически произвольно, при условии, что передатчик и приемник используют одно и то же кодирование. Если один бит ассоциируется с полярностью на действительной части комплексной плоскости (синфазный) и один бит – с полярностью на мнимой части (квадратурный), мы получим созвездие, изображенное на рисунке 2. Как показано на рисунке, бит справа определяют полярность действительной части, в то время как бит слева определяет полярность мнимой части. Это будет удобно при последующей интерпретации пары бит двоичными числами. При этом левый бит будем интерпретировать, как младший значащий бит, а правый бит – как старший значащий бит. Подобная интерпретация может показаться интуитивно противоречивой, но эти последовательности бит будут храниться в виде массивов, и естественно записывать массивы с увеличением индекса вправо. Символьное кодирование может быть также представлено таблицей. В таблице 1 символы закодированы в соответствии с рисунком 2.

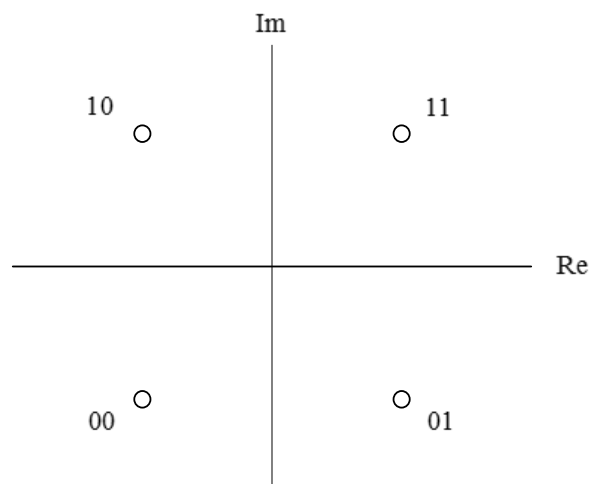


Рисунок 2. Вариант символьного кодирования

Таблица 1. Кодирование символов

Номер индекса	Комбинация бит	Значение символа
0	00	$Ae^{-j3\pi/4} = -A/\sqrt{2} - jA/\sqrt{2}$
1	10	$Ae^{j3\pi/4} = -A/\sqrt{2} + jA/\sqrt{2}$
2	01	$Ae^{-j\pi/4} = A/\sqrt{2} - jA/\sqrt{2}$
3	11	$Ae^{j\pi/4} = A/\sqrt{2} + jA/\sqrt{2}$

Приемник QPSK начинается с DSB-SC демодулятора. Когда передаваемый QPSK сигнал поступает на приемник, он имеет форму последовательности импульсов, заданных формулой

$$r(t) = Dg_{TX}(t)\cos(2\pi f_c t + \theta + \phi), \quad (6)$$

где D константа (обычно много меньше константы A в передаваемом сигнале), угол θ несет в себе информацию, а угол ϕ представляет собой разность фаз между генераторами несущей передатчика и приемника. Если частота несущей приемника задана равной частоте несущей передатчика, то приемник USRP будет выполнять большую часть операций по демодуляции QPSK сигнала. *Fetch Rx Data* приемника обеспечит последовательность выходных импульсов, каждый из которых задается ¹⁰

$$\tilde{r}(t) = \frac{D}{2} g_{TX}(t) e^{j(\theta+\phi)}. \quad (7)$$

Последовательность импульсов, представленных уравнением (7) пропускается через согласованный фильтр с импульсной характеристикой $g_{RX}(t)$. Если записать $g(t) = g_{TX}(t) * g_{RX}(t)$, то выходной сигнал согласованного фильтра может быть представлен

$$\tilde{y}(t) = \frac{D}{2} g(t) e^{j(\theta+\phi)}. \quad (8)$$

Выход согласованного фильтра затем дискретизируется по одному отсчету на символ. В идеальном случае $g(0) = 1$, и на выходе дискретизатора получаем последовательность комплексных чисел вида

¹⁰ Все сигналы представлены в LabVIEW, как сигналы, дискретизированные во времени. Представление их, как сигналы, непрерывные во времени, делает описание более удобным для чтения.

$$\tilde{y}[k] = \frac{D}{2} e^{j(\theta_k + \phi)}, \quad (9)$$

где θ_k значение фазы в момент выборки $t = kT$, а D и ϕ являются константами. На практике отсчеты $\tilde{y}[k]$ будут искажены из-за межсимвольных помех и шума. Как и в предыдущей лабораторной работе, межсимвольные искажения устраняются фильтром эквалайзера. Мы также видели, что использование эквалайзера позволяет устранить фазовую ошибку ϕ , и в результате получить

$$\tilde{y}[k] = \frac{D}{2} e^{j\theta_k} + \text{residual ISI} + \text{noise}. \quad (10)$$

Следующие операции, которые должны быть выполнены приемником, это детектирование и декодирование сигнала. Детектирование сигнала представляет собой процесс проверки каждого отсчета $\tilde{y}[k]$ и определения, какому переданному символу он вероятнее всего соответствует. Оказывается, что алгоритм детектирования, называемый *правилом принятия решения*, на удивление прост и имеет изящную геометрическую формулировку. На Рисунке 3 показано сигнальное созвездие с отсчетом $\tilde{y}[k]$. Вследствие остаточных межсимвольных искажений и шума $\tilde{y}[k]$ не может в точности соответствовать какому-либо из возможных значений символов.

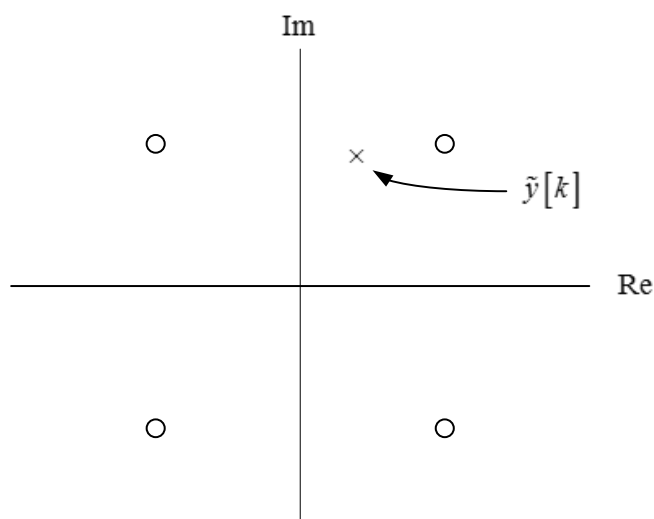


Рисунок 3. Сигнальное созвездие и полученный отсчет

Правило принятия решения для определения наиболее вероятного переданного символа $\tilde{y}[k]$ предполагает следующее: вычисление расстояния между $\tilde{y}[k]$ и каждым возможным символом.

Наименьшее расстояние от $\tilde{y}[k]$ определяет наиболее вероятный символ. На Рисунок 3 один наиболее вероятный символ с наименьшим расстоянием от $\tilde{y}[k]$ расположен в первом квадранте. После того, как символ определен, для преобразования символа в битовую последовательность

Таблица 1 можно воспользоваться таблицей кодирования 1. В примере на Рисунок 3 полученный отсчет соответствует комбинации бит 11.

12.3 Подготовка к лабораторной работе

1. Создайте функцию для кодирования символов в передатчике. (Детектирование и декодирование в приемнике будет осуществляться *MT PSK Feedforward Equalizer*). В таблице 2 приведены характеристики входных и выходных параметров вашей функции.

Таблица 2. Характеристики для кодирования данных

Входы			Выход		
Имя	Тип		Имя	Тип	
Data In (Входные данные)	Одномерный массив 8-ми битных целочисленных значений	Поток битовых данных	Symbols Out (Выходные символы)	Одномерный массив	Передаваемый поток символов
Symbol Map (Таблица кодирования символов)	Одномерный массив комплексных чисел двойной точности	Символы, индексированные парой бит, представлены в виде чисел			

Ваша функция должна соответствовать кодировке символов из

Таблица 1 с $A = 1$. В шаблоне *MapDataTemplate.gvi* входы и выходы уже подключены. Массив, содержащий коды символов, необходим вам в качестве одного из входов и доступен в шаблоне *QPSK Tx.gvi*, который описан ниже.

Сохраните вашу функцию кодирования символов в файл, имя которого включает в себя "MapData" и ваши инициалы (например, *MapData_BAB.gvi*).

2. Создайте программу, реализующую QPSK передатчик. Для начала работы вам предлагается воспользоваться шаблоном *QPSKTxTemplate.gvi*. Этот шаблон включает входы и выходы, массив кодов символов, функции для отображения сигнала и спектра мощности, а также функции, необходимые для интерфейса с USRP.

Вам нужно будет добавить

- функцию кодирования символов по пункту 1, описанному выше,
- функцию *AddFrameHeader(Complex)*
- передискретизацию
- формирование импульса, сглаженных по закону косинуса
- масштабирование амплитуды

Подсказка: Воспользуйтесь в качестве модели вашим BPSK передатчиком. QPSK передатчик отличается только методом кодирования символов.

Сохраните ваш приемник в файл, имя которого включает в себя "QPSKTx" и ваши инициалы (например, *QPSKTx_BAB.gvi*).

5. Создайте функцию эквалайзера для QPSK. Для этого вы можете скопировать функцию BPSK эквалайзера и выполнить следующие изменения:
 - Измените значение входа "M-PSK" *MT Generate System Parameters* с 2 на 4.
 - Замените таблицу кодирования символов путем копирования таблицы кодирования символов QPSK из шаблона *QPSKTxTemplate.gvi*.
 - Замените массив констант обучающей выборки на новую обучающую последовательность, в которой продублирован каждый бит. На Рисунок 4 показан простой способ дублирования с помощью функции *Interleave 1D Arrays*.

Сохраните ваш эквалайзер в файл, имя которого включает в себя слово "QPSK_Equalizer" и ваши инициалы (например, *QPSK_Equalizer_BAB.gvi*).

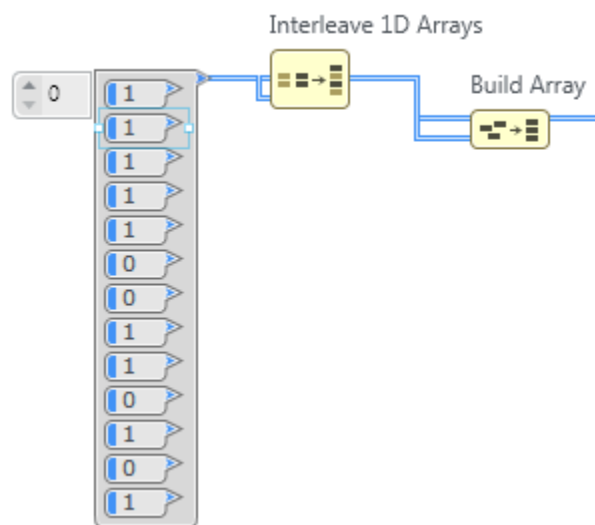


Рисунок 4. Изменение обучающей выборки

4. Создайте программу, реализующую QPSK приемник. Для начала работы вам предлагается воспользоваться шаблоном QPSKRxTemplate.gvi. Шаблон включает в себя входы и выходы, функции, необходимые для взаимодействия с USRP и несколько функций для настройки индикаторов.

Подсказка: Воспользуйтесь в качестве модели вашим BPSK приемником. QPSK приемник очень похож.

Вам необходимо добавить

- *Channel.gvi*.
- Функцию *Cluster Properties* после *Channel.gvi*, чтобы получить доступ к компоненту "Y", массиву отсчетов в виде комплексных чисел двойной точности.
- Согласованный фильтр с характеристикой по закону приподнятого косинуса и функцию *Convolution* для реализации фильтра.
Примечание: Там нет синхронизатора фаз, так как эта функция будет выполняться эквалайзером. Кроме того, нет необходимости использовать действительную часть полученных значений выборок.
- Функцию *Cluster Properties* для создания кластера комплексных чисел двойной точности, получаемых с выхода функции *Convolution*.
- *PulseAlign (Complex)*.
- Функцию *Cluster Properties* после *PulseAlign(Complex)*, чтобы получить доступ к элементу "Y". Добавьте *Decimate (single shot)* для получения выборки выравненных импульсов.
- *FrameSync(Complex)*.
- Ваш эквалайзер, реализованный при выполнении пункта 3, потребуется позже. Выход "Output Signal" *FrameSync(Complex)* является входом эквалайзера. Не используйте выход "Aligned Samples" *FrameSync(Complex)*.
- Подключите выход эквалайзера "Output Bits" к функции *Array Subset*. Установите значение "index" равное 52 для того, чтобы удалить обучающую последовательность, и задайте значение "length" соответствующим количеству бит в сообщении. Выходные значения

функции *Array Subset* могут быть отображены как "Output Bits" приемника, а также могут быть переданы в *MT Calculate BER*.

- В шаблоне есть две функции *MT Format Eye Diagram*. Подключите выход "Output Signal" *PulseAlign(Complex)* к одной из этих функций. Это позволит визуализировать глазковую диаграмму синфазного компонента принимаемого сигнала прежде, чем будет выполнен эквалайзинг. Подключите выход эквалайзера "Output Complex Waveform" ко второй глазковой диаграмме. Это позволит визуализировать глазковую диаграмму синфазного элемента принимаемого сигнала после эквалайзинга.

Подключите также выход эквалайзера "Output Complex Waveform" ко входу "waveform" *MT Format Constellation*.

Сохраните ваш приемник в файл, имя которого включает в себя "QPSKRx" и ваши инициалы (например, *QPSKRx_BAB.gvi*).

Вопросы

1. Скорость передачи символов вашего передатчика и приемника равна 10 000 символов/с. Какова скорость передачи данных в бит/с?
2. Если приемник настроен на частоту IQ 200 кГц, сколько отсчетов на символ будет использоваться в приемнике?
3. *Спектральная эффективность* модулированного сигнала это отношение скорости передачи данных в бит/с к ширине спектра передаваемого сигнала. Вычислите спектральную эффективность BPSK сигнала, использующего импульсы, сглаженные по закону косинуса, которые вы сгенерировали в лабораторной работе по BPSK. Сравните со спектральной эффективностью сигнала QPSK, при условии, что ширина спектра сигнала остается той же самой.

12.4. Порядок выполнения лабораторной работы

1. Подсоедините тестовый кабель и аттенюатор между разъемами TX 1 и RX 2 USRP. Подключите USRP к вашему компьютеру и включите питание. Запустите LabVIEW и откройте передатчик, созданный вами при подготовке к выполнению лабораторной работы.

2. Убедитесь, что для передатчика заданы следующие настройки:

Несущая частота (Carrier Frequency): 915.0 МГц

IQ частота (IQ Rate): 200 кГц Примечание: Это задает значение $1/T_x$.

Коэффициент усиления (Gain): 0 дБ

Активная антенна (Active Antenna): TX1

Скорость передачи символов (Symbol rate): 10 000 символов/с

Длина сообщения (Message Length): 1000 бит

Фильтр, формирующий импульсы (Pulse shaping filter): Root Raised

3. Запустите передатчик. Используйте большую кнопку STOP на лицевой панели для отключения разъемов передатчика.

4. Используя индикатор спектра мощности на лицевой панели передатчика, измерьте ширину спектра комплексного сигнала.

5. Убедитесь, что для приемника заданы следующие настройки:

Несущая частота (Carrier Frequency): 915.0 МГц

IQ частота (IQ Rate): 200 кГц Примечание: Это задает значение $1/T_z$. Обратите внимание, что параметры T_z и dt совпадают.

Коэффициент усиления (Gain): 0 дБ

Активная антенна (Active Antenna): RX2

Скорость передачи символов (Symbol rate): 10 000 символов/с

Длина сообщения (Message Length): 1000 бит

Фильтр, формирующий импульсы (Pulse shaping filter): Root Raised

Использовать канал (Use Channel): off

6. Запустите сначала передатчик, затем приемник. После того, как приемник получит данные, вы можете завершить работу передатчика. Приемник должен показать значение BER, равное 0.0.

7. Сравните глазковые диаграммы до и после эквалайзинга. Запустите приемник несколько раз, пока вы не определите, что изменяется, а что не изменяется от запуска к запуску. Иногда на глазковой диаграмме, полученной до эквалайзинга, будут отображаться не два отсчета, а четыре. Сделайте скриншоты глазковой диаграммы до и после эквалайзинга.

8. Установите в модели канала по умолчанию элемент управления Use Channel в состояние "on", а значение задержки распространения сигнала задайте равную 100 мкс. Запустите передатчик, а затем несколько раз запустите приемник. Сравните глазковые диаграммы до и после эквалайзинга. Измерьте раскрытие глазка до и после эквалайзинга. Попытайтесь получить наилучший результат измерений для случая "до эквалайзинга".

9. Установите элемент управления Use Channel в состояние "off", и снова запустите передатчик и приемник. В этот раз наблюдайте результат по индикатору сигнального созвездия. Сделайте скриншот созвездия.

10. Вместо эквалайзера подключите выход "Output Signal" *FrameSync(Complex)* к *MT Format Constellation*. Запустите передатчик, а затем несколько раз запустите приемник. Как изменилось значение угла на сигнальном созвездии? Как изменился радиус на сигнальном созвездии?

Вопросы

1. Какова измеренная ширина спектра переданного QPSK сигнала? (Примечание: не ширина спектра модулирующего сигнала). Сравните с шириной спектра BPSK сигнала, которую вы измерили в предыдущей лабораторной работе.
2. Зафиксируйте результаты измерения раскрытия глазка, выполненные в пункте 6.
3. Что явилось причиной поворота сигнального созвездия, который вы наблюдали при выполнении пункта 8? Как выглядит сигнальное созвездие, когда на глазковой диаграмме, полученной до эквалайзинга, отображаются четыре отсчета вместо 2? Объясните, почему это происходит с глазковой диаграммой. Подсказка: Запомните, что график глазковой диаграммы отображает только синфазную компоненту сигнала.
4. Как будет выглядеть сигнальное созвездие, если несущие частоты передатчика и приемника немного отличаются?
6. Какое вы ожидаете изменение вероятности ошибок BER при вращении сигнального созвездия? (Предположим, что эквалайзер отсутствует).
7. Понаблюдайте за переходными процессами на графическом индикаторе сигнального созвездия. Обратите внимание, что переходы сигналов между первым и третьим квадрантами проходят через начало координат графика, как и переходы сигналов между вторым и четвертым квадрантами. Объясните, что происходит с амплитудой принятого сигнала при этих переходах.

12.5 Отчет

Подготовка к лабораторной работе

Подготовьте документацию для созданных вами четырех функций: кодирования символов, передатчика, эквалайзера и приемника. Кроме того, включите в отчет документацию для других созданных вами функций. Для получения документации распечатайте разборчивые скриншоты лицевой панели и блок-диаграммы.

Ответьте на все вопросы раздела "Подготовка к лабораторной работе".

Выполнение лабораторной работы

Включите в отчет программы кодирования символов, передатчика, эквалайзера и приемника. Повторно предоставьте документацию для любых функций, которые вы модифицировали при выполнении лабораторной работы.

Включите в отчет графики, которые получены при выполнении пунктов 5 и 7.

Ответьте на вопросы из раздела "Порядок выполнения лабораторной работы".