

Лабораторная работа №12. GNU Radio

Сергиенко Кирилл

Июнь 2024

1 Передача сигнала QPSK

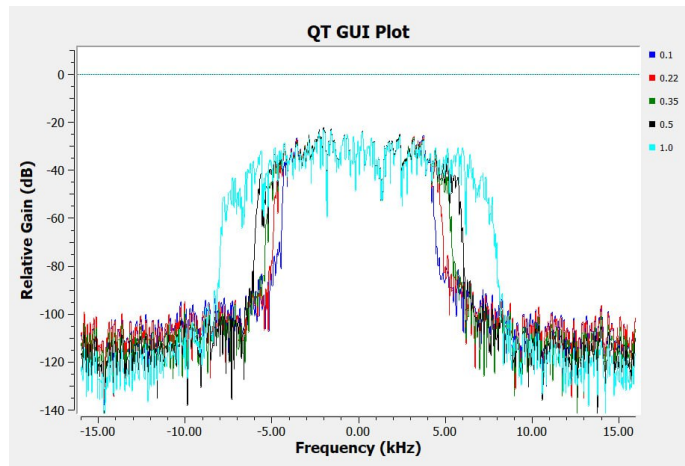
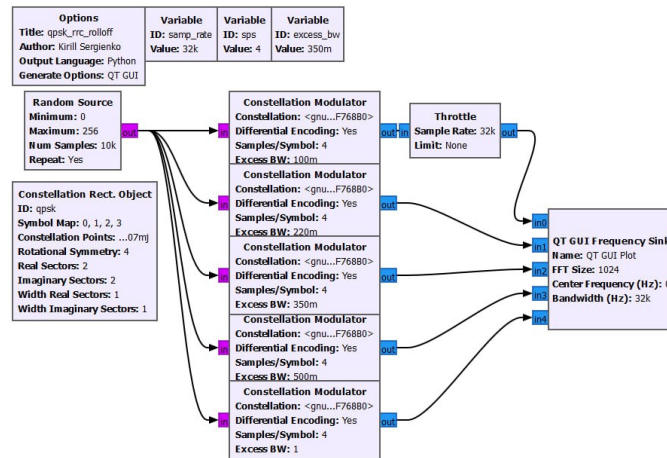
Для передачи сигнала QPSK (квадратурно-фазовой манипуляции) случайным образом генерируется поток битов и модулируется в сложное созвездие. Для этого используем блок Constellation Modulator, который применяет объект созвездия и другие настройки, чтобы контролировать передаваемый сигнал.

В случае моделирования сигнала число экземпляров на символ важно для того, чтобы быть уверенными, что есть соответствие этому числу во время всего процесса передачи сигнала. Зададим значение равное 4. Это должно помочь лучше визуализировать сигнал в разных областях. Также необходимо установить значение избыточной пропускной способности.

1.1 Избыточная пропускная способность

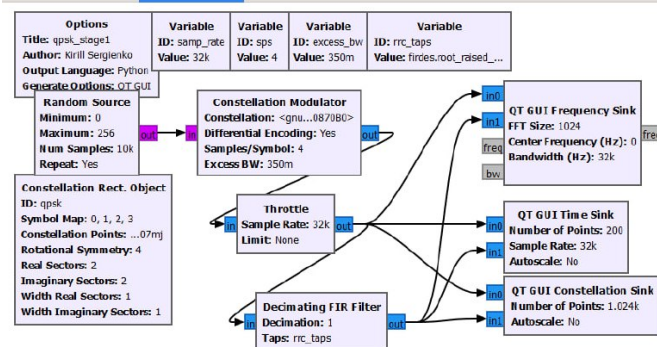
Модулятор созвездия использует фильтр формирования импульсов RRC (корень из приподнятого косинуса), который позволяет нам изменять один параметр для регулирования roll-off коэффициента фильтра (коэффициент сглаживания), математически он известен как «альфа». Получившиеся частотная характеристика фильтра при различных значениях roll-off коэффициента.

Приведенный ниже блок-график генерирует рисунок, показывающий различные значения избыточной полосы пропускания. Обычно используются значения от 0,2 (красная трассировка) до 0,35 (зеленая трассировка). В нашем случае мы будем использоваться 0,35.

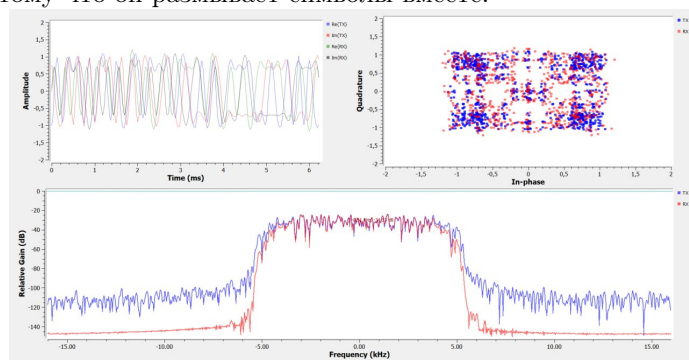


1.2 Согласованные фильтры и ISI

Посмотрим на граф передачи сигнала:



На графике созвездия видно эффекты восходящей выборки (генерация 4 образцов на символ) и процесса фильтрации. В данном случае фильтр RRC добавляет преднамеренную само-интерференцию, известную как межсимвольная интерференция (ISI). ISI плохо подходит для принятого сигнала, потому что он размывает символы вместе.



В результате объединения двух фильтров RRC вместе, получаем повышенный косинусный фильтр, который является формой фильтра Найквиста.

2 Нарушения в канале

Рассмотрим эффекты канала и то, как искажается сигнал в промежутке между моментом его передачи и моментом, когда мы видим сигнал в приемнике. Первым шагом является добавление модели канала. Используем базовый блок модели канала GNU Radio. Он позволяет смоделировать несколько основных проблем, с которыми нам приходится иметь дело.

Первая проблема с приемниками - это шум. Тепловой шум в нашем приемнике вызывает шум, который мы знаем как аддитивный белый гауссовский. Устанавливаем мощность шума путем регулировки значения напряжения. Делаем именно таким образом потому, что нам нужно знать полосу пропускания сигнала, чтобы правильно рассчитать мощность. Можно рассчитать напряжение шума исходя из желаемого уровня мощности, зная другие параметры моделирования.

Еще одна существенная проблема между двумя радиостанциями - это разные часы, которые определяют частоту радиостанций. Часы, во-первых, несовершенны, и поэтому разные радиостанции различаются. Номинально, две различные станции могут иметь одинаковую частоту, однако ввиду несовершенностей они имеют некоторое отклонение от нее. Поэтому, принятый сигнал по итогу будет отличаться от ожидаемого на величину, равную сумме этих отклонений.

К вышеуказанной проблеме относится и вопрос идеальной точки выборки. После увеличения выборки сигнала в передатчике и его формирования при приеме требуется выполнить выборку сигнала в исходной точке, чтобы максимизировать мощность сигнала и минимизировать межсимвольные

помехи. Как и на первом этапе моделирования, после добавления второго RRC-фильтра, можно наблюдать, что среди 4 выборок для каждого символа одна из них находится в идеальной точке $+1$, -1 или 0 . Но опять же, два радиоприемника работают на разных скоростях, поэтому идеальная точка выборки неизвестна.

Данный этап моделирования позволяет нам поэкспериментировать с этими эффектами аддитивного шума, смещения частоты и временного интервала. Во время запуска моделирование установлены следующие параметры: шум $-0,2$; смещение частоты $-0,025$; смещение по времени $-1,0005$.

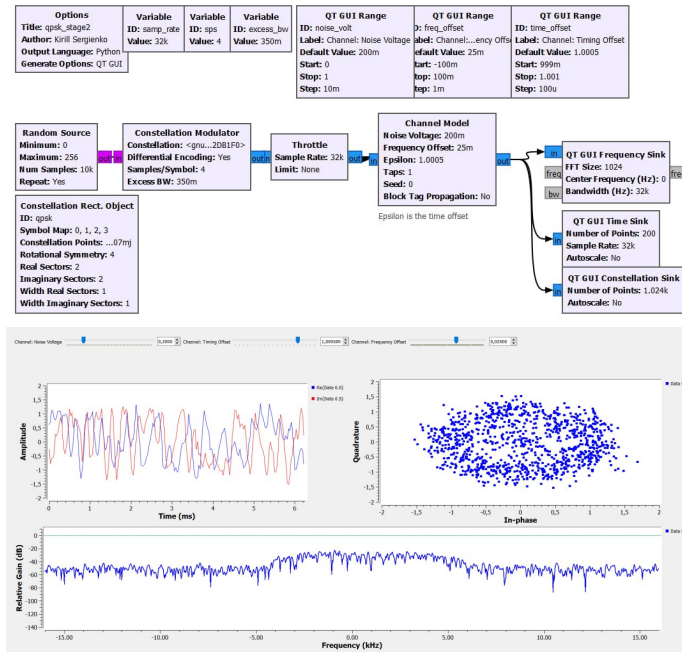


График показывает нам облако сэмплов, гораздо худшее, чем то, с чего мы начинали на последнем этапе. Теперь нам нужно отменить все эти эффекты из этого принятого сигнала.

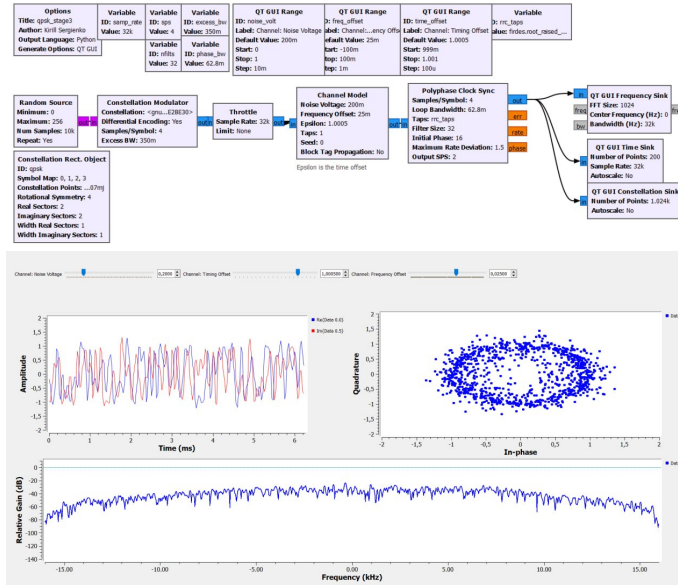
3 Прием сигнала QPSK

3.1 Многофазная синхронизация часов

Многофазная синхронизация часов выполняет три функции. Во-первых, она выполняет восстановление синхронизации часов. Во-вторых, она представляет фильтр согласования с приемником для удаления ISI. В-третьих, он снижает дискретизацию сигнала (уменьшает количество выборок на символ).

Указанная ниже, блок-схема принимает выходные данные модели канала и передает их через блок многофазной синхронизации. Он настроен

с 32 фильтрами и пропускной способностью контура $2\pi/100$. Блок также принимает значение ожидаемых выборок для каждого символа.



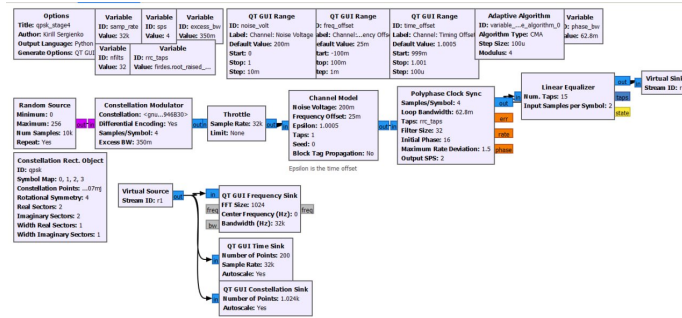
При запуске этого скрипта видно, что в созвездии все еще немного шумно из-за ISI после 32 фильтров, но оно быстро поглощается шумом, как только значение напряжения шума канала больше 0.

3.2 Многолучевость

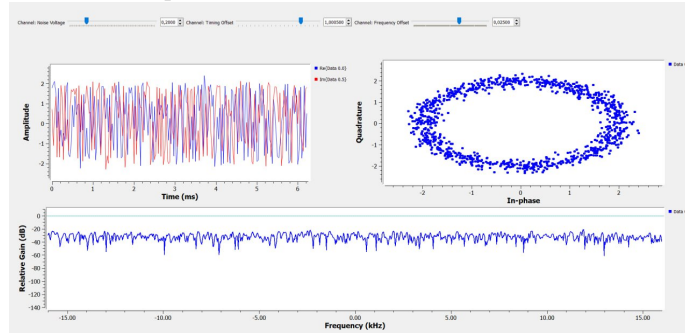
Многолучевость является результатом того, что в большинстве коммуникационных сред нет единого пути прохождения сигнала от передатчика к приемнику. Каждый раз, когда появляется объект, отражающий сигнал, между двумя узлами может быть установлен новый путь. Такие поверхности, как здания, вывески, деревья, люди и так далее, могут создавать отражения сигнала. Каждый из этих отражающих путей будет отображаться на приемнике в разное время в зависимости от длины пути. Их суммирование на приемнике приводит к искажениям, как конструктивным, так и деструктивным.

3.3 Эквалайзер

Адаптивный алгоритм имеет тип алгоритма СМА, или алгоритм постоянного модуля. Это слепой эквалайзер, но он работает только с сигналами, которые имеют постоянную амплитуду или модуль. Это означает, что цифровые сигналы, такие как QPSK, являются хорошими кандидатами, поскольку они имеют точки только на единичной окружности.



Можно наблюдать, как сходится алгоритм СМА. Стоит обратить внимание на то, что, поскольку у нас есть и блок синхронизации часов, и блок эквалайзера, они сходятся независимо, но один этап повлияет на следующий. Таким образом, здесь происходит некоторое взаимодействие, пока оба фиксируют сигнал. Однако, мы можем видеть эффект многолучевого сигнала с временной блокировкой до и после эквалайзера. Перед эквалайзером достаточно неприятный сигнал даже без шума. Эквалайзер прекрасно понимает, как инвертировать и гасить этот канал, чтобы снова получился чистый сигнал. Также можно видеть сам канал и то, как он выравнивается после эквалайзера.



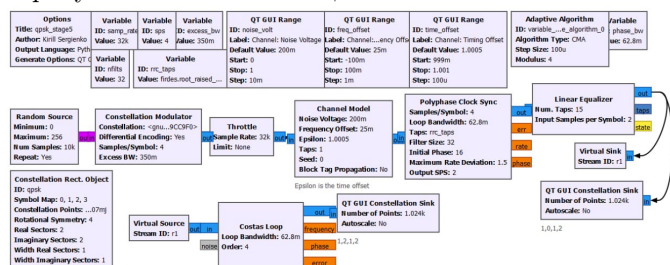
3.4 Фазовая и частотная коррекция

Несмотря на выравнивание канала, проблемы со смещением фазы и частоты все еще не решены. Эквалайзеры, как правило, не адаптируются быстро, и поэтому простое смещение частоты может оказаться за пределами его возможностей. Кроме того, если просто запустить эквалайзер СМА то, все, о чем он заботится — это о приближении к единичному кругу. Он не знает о созвездии, поэтому, когда он блокируется, то это происходит на любой заданной фазе. Теперь нужно скорректировать любое смещение фазы, а также любое смещение частоты.

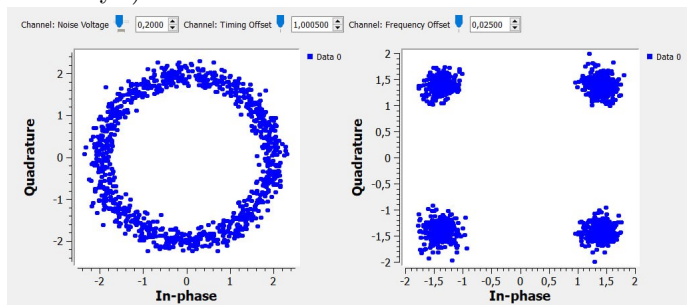
Две особенности этого этапа. Во-первых, используем цикл второго порядка, чтобы присутствовала возможность отслеживать как фазу, так и частоту (которая является производной от фазы) с течением времени. Во-вторых, тип восстановления, с которым мы здесь будем иметь дело, пред-

полагает, что мы выполняем точную коррекцию частоты. Поэтому должна быть уверенность, что мы уже находимся в приличном диапазоне идеальной частоты. Если будем слишком далеко, то цикл здесь не сойдется, и мы продолжим вращаться.

Для этой задачи используем цикл Костаса в примере носителя. Он может синхронизировать BPSK, QPSK и 8PSK. Как и все другие модули, он использует цикл второго порядка и, следовательно, определяется параметром пропускной способности цикла.



После эквалайзера все символы находятся на единичном круге, но вращаются из-за смещения частоты. На выходе блока контуров Костаса можно видеть заблокированное созвездие, с которого мы начали (плюс дополнительный шум).



4 Декодирование

На рисунке ниже находится окончательная блок-схема для декодирования сигнала. Для начала вставим декодер Constellation после цикла Костаса. На этом этапе получаем символы от 0 до 3, потому что это размер алфавита в схеме QPSK. Но как можно быть уверенным в том, что из этих символов 0–3 у нас есть то же самое сопоставление символов с точками созвездия, что и при передаче? Ничто из того, что мы делали, не имело никакого знания о преобразовании передаваемого символа в созвездие, а это означает, что может быть неоднозначность в 90 градусов в созвездии. Однако, мы передавали не само созвездие, а разницу между символами созвездия, установив для параметра Differential в блоке Constellation Modulator значение True.

