

Министерство образования и науки Российской Федерации
Федеральное государственное бюджетное образовательное
учреждение высшего образования
«ТОМСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ СИСТЕМ
УПРАВЛЕНИЯ И РАДИОЭЛЕКТРОНИКИ»

Кафедра телекоммуникаций и основ радиотехники (ТОР)

Е.В. Рогожников, К. Савенко

Телекоммуникационные технологии

Методические указания для выполнения практических работ

Томск – 2018

Рогожников Е.В. Савенко К. Телекоммуникационные технологии: методическое пособие – Томск: Издательство ТУСУР, 2018. – 52 с. Методическое пособие для студентов ВУЗов посвящено расчету и реализации телекоммуникационных систем в программной среде GNU Radio.

Оглавление

Введение	4
1. Практическая работа «Начало работы с GNU Radio»	5
Задачи практической работы:	5
Ход выполнения работы	5
Контрольные вопросы к практической работе	13
2. Практическая работа «Цифровой FM-передатчик и приемник»	14
Задачи практической работы:	14
Краткие теоретические сведения	14
Ход выполнения работы	16
Задания для самостоятельного выполнения	22
Контрольные вопросы к практической работе	22
3. Практическая работа «Цифровая модуляция QAM-M»	23
Задачи практической работы:	23
Краткие теоретические сведения	23
Ход выполнения работы	25
Задания для самостоятельного выполнения	33
4. Практическая работа «QAM/PSK модуляция и измерение BER»	34
Задачи практической работы:	34
Ход выполнения работы	34
Задания для самостоятельного выполнения	41
Контрольные вопросы к практической работе	41
5. Практическая работа «Технология OFDM»	42
Задачи практической работы:	42
Краткие теоретические сведения	42
Ход выполнения работы	43
Задания для самостоятельного выполнения	51
Контрольные вопросы к практической работе	52

Введение

В современной системе связи, кодирование источника, шифрование и мультиплексирование реализованы цифровым способом. Основываясь на FPGA и DSP-процессорах, формирование импульсов и модуляция выполняются в цифровом виде с высокой скоростью обработки. OFDM, например, как предпочтительная технология модуляции для мобильной системы новых поколений, не может обойтись без цифровой реализации.

Software Defined Radio (SDR) - перспективная разработка, позволяющая производить управление приемником и передатчиком сигналов программными алгоритмами. Вместе с высокоскоростными цифро-аналоговыми преобразователями и универсальными интерфейсами, различные стандарты (например, Bluetooth, WLAN, DECT, ZigBee) могут быть реализованы на одном и том же оборудовании. Единственное, что нужно изменить, это программное обеспечение, которое реализует стек протокола и функции физического уровня.

В пяти практических работах данного методического пособия, представлены эксперименты, позволяющие с легкостью понять принцип работы программно-определяемой радиосистемы (SDR). А на основе полученных знаний можно развиваться в области передачи и приема информации, с помощью передовой технологии SDR. Сборник начинается с аспектов настройки GNU Radio, ознакомлением с его графическим интерфейсом GNU Radio Companion. Это дает некоторые базовые знания о том, как программное обеспечение GNU Radio работает в сочетании с USRP. В следующей практической работе представлен проект GNU Radio, в котором реализован приемник-передатчик для FM сигнала. Еще одна практическая работа посвящена изучению принципов квадратурной модуляции с реализацией передачи через реальный канал. Так же представлена модель для QAM/PSK модуляции, где производится измерения вероятности битовой ошибки от отношения сигнал/шум. В пятой практической работе производится передача аудиоданных при помощи перспективного на сегодняшний день метода OFDM модуляции.

1. Практическая работа «Начало работы с GNU Radio»

Цель работы: Изучить основные функции и блоки GNU Radio и составить тестовую программу для генерации сигнала.

Задачи практической работы:

- 1) Изучить основные функции и блоки GNU Radio Companion.
- 2) Произвести генерацию синусоидального сигнала в GNURadio. Проанализировать характеристики частоты сгенерированных колебаний.
- 3) Произвести генерацию синусоидального сигнала с переменной частотой. Воспроизвести звук на динамических головках ПК. Проанализировать характеристики частоты сгенерированных колебаний.

Ход выполнения работы

GNU Radio Companion (GRC) представляет собой графический пользовательский интерфейс, который позволяет создавать модели при помощи блоков.

Для запуска GRC откройте терминал (Ctrl+Alt+T) и введите:
[gnuradio-companion](#)

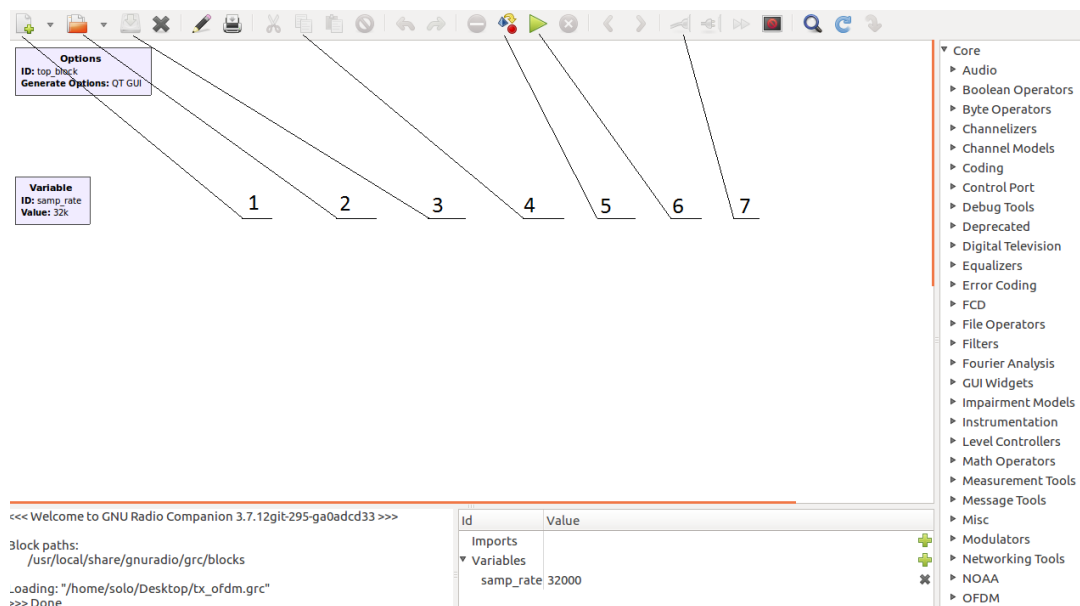


Рисунок 1.1 - Окно программы gnuradio-companion

Панель меню, которая изображена на рисунке 1.1 состоит из следующих команд 1 - создать новую модель, 2 - открыть модель, 3 - сохранить модель, 4 - команды копирования, 5 - компиляция, 6 - запуск и остановка моделирования, 7 - отключение блока от функционирования модели.

Вы увидите окно программы gnuradio-companion, на котором уже расположены два блока. Блок «Options» в верхнем левом углу используется для установки некоторых общих параметров графа, такие как графический пользовательский интерфейс (GUI) для виджетов и вывода результатов моделирования на дисплей, или размер холста, на котором размещены блоки. В нижней части экрана расположено окно терминала, в котором отображается процесс моделирования и описание ошибок, если таковые имеются. Также в нижней части экрана можно увидеть окно, в котором отображаются значения переменных, используемых в модели. Щелкните правой кнопкой мыши на блоке и выберите «Options» (или дважды щелкните на блоке), чтобы увидеть все параметры, которые могут быть настроены.

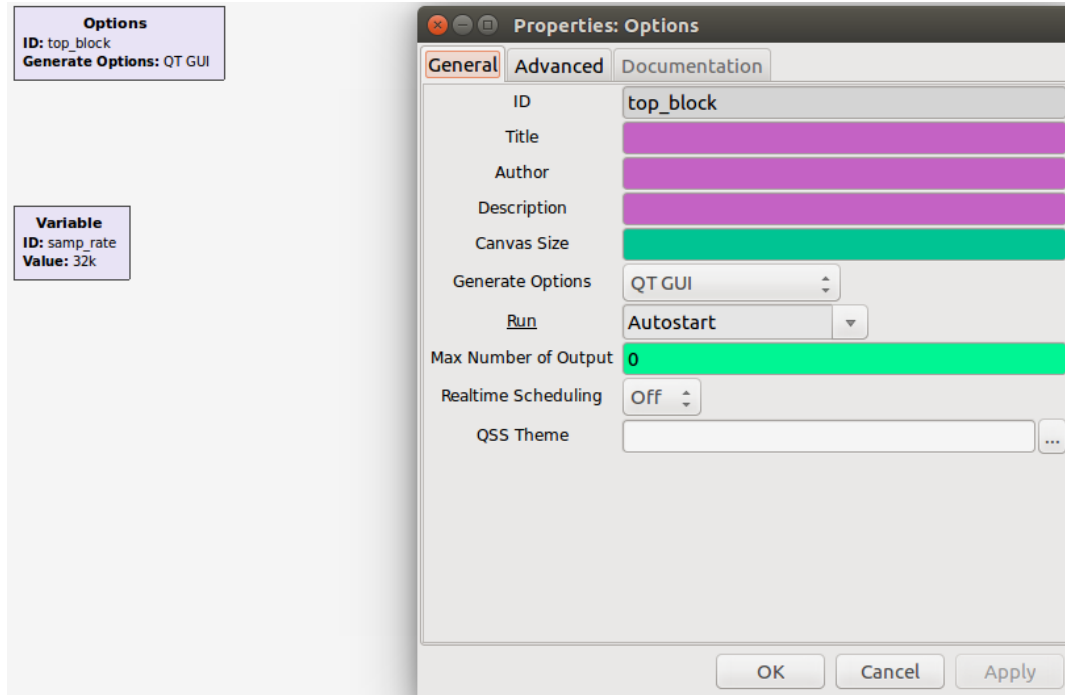


Рисунок 1.2 - Окно параметров блока «Options»

Пока мы оставим настройки по умолчанию без изменений. Ниже расположен блок «Variable», который используется для установки частоты дискретизации, например, в $F_s = 32000$ Гц в окне GRC выше.

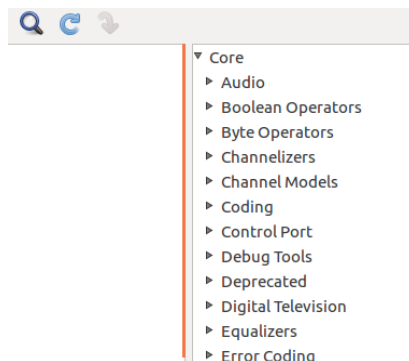


Рисунок 1.3 - Список блоков для моделирования

В правой части рабочего окна находится список доступных для моделирования блоков, которые разбиты на соответствующие категории. Нажмите на треугольник рядом с категорией, например, Генераторы сигналов (Waveform Generators), чтобы увидеть, какие блоки доступны в этой категории. Здесь расположены такие часто используемые блоки, как генератор периодических сигналов (Signal Source), источник шума (Noise Source), Генератор случайной последовательности символов (Random Source) и другие. Поиск необходимых блоков можно выполнить нажав на значок лупы, расположенный на панели меню в верхней части экрана.

Используйте функцию поиска (нажмите на значок лупы в правом верхнем углу), чтобы ввести ключевое слово, например, Source, чтобы увидеть все блоки с «Source» в названии.

Щелкните на «Signal source» и перетащите его в рабочую область, как показано ниже.

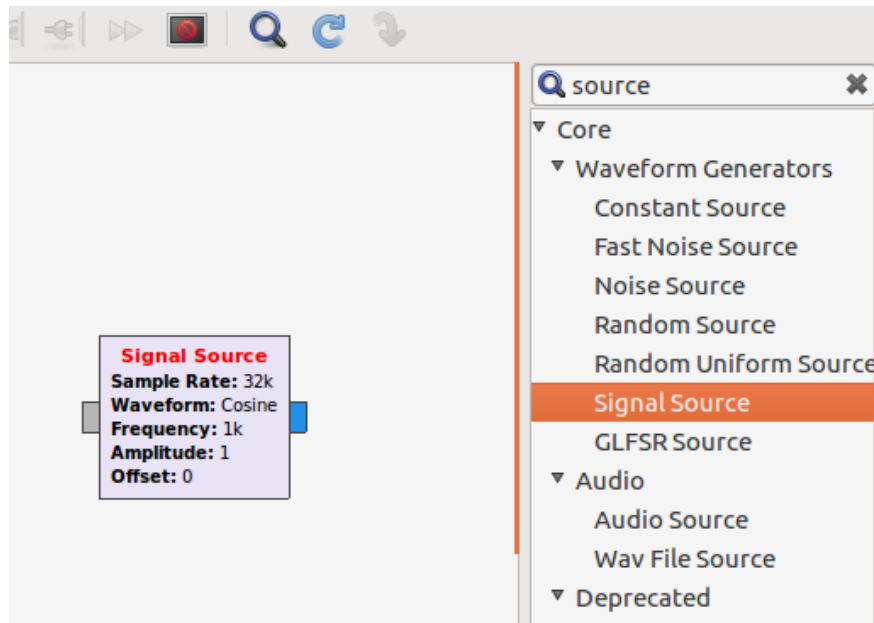


Рисунок 1.4 - Перенос блока в рабочую область программы

Вы также можете вращать блоки, щелкнув правой кнопкой мыши на них а затем нажав либо «Rotate Counterclockwise» или «Rotate Clockwise». Блоки также могут быть временно отключены, нажав на «Disable», который полезен для отладки.

Обратите внимание на то, что блок «Signal source» имеет два порта, один серый слева и синий справа. Цвет порта указывает на тип данных, генерируемых для выходного порта или типа данных, принятых для входа.

Для того чтобы увидеть используемые в GRC типы данных откройте пункт «types» во вкладке меню «help». (Для того чтобы открыть вкладку «help» в Ubuntu 16.04 необходимо навести курсор на верхнюю часть окна программы).

GRC использует потоковую модель обработки для работы с большими объемами данных в режиме реального времени, в отличие от традиционной среды обработки массивов (например Matlab). На практике это означает, что каждый блок обработки сигналов имеет независимый планировщик и работает в своем собственном потоке. То есть если вы работаете с приемно-передающим оборудованием который работает с определенной частотой дискретизации данных (например, 44100 выборки / сек для звукового сигнала, или 10 миллионов выборки / сек для интерфейса SDR), то и частоту дискретизации моделирования необходимо выставить соответствующую.

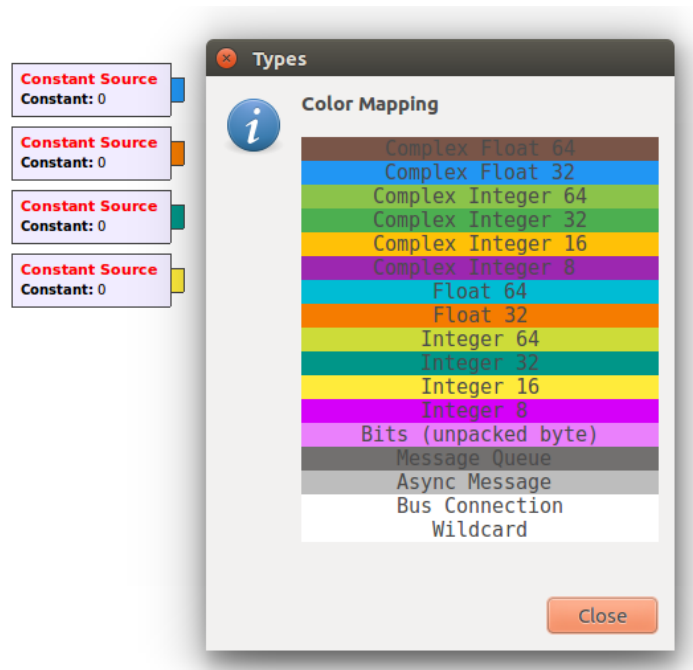


Рисунок 1.5 - Типы данных

Но если и источник и приемник реализуются исключительно в программном обеспечении (например, генератор сигналов или индикация частоты), то будет полезным ограничить скорость обработки данных при конкретной частоте дискретизации вашей модели. Для этого используется блок «Throttle». Его использование значительно снизит нагрузку на процессор и увеличит скорость моделирования.

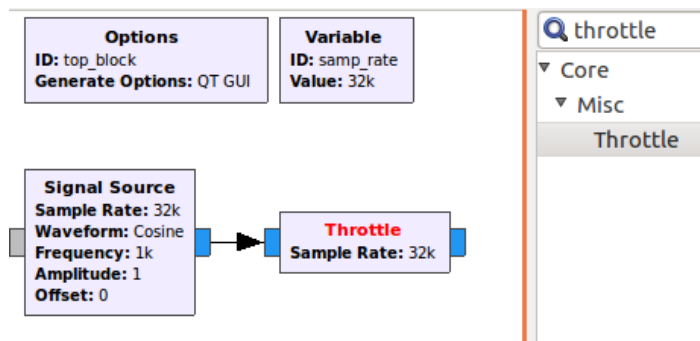


Рисунок 1.6 - Подключение блока «Throttle»

Подключите блок «Throttle» к выходу блока «Signal source» последовательно кликнув левой кнопкой мыши на порты каждого из блоков.

Важно! Чтобы блоки работали, оба порта должны использовать данные одного и того же типа (то есть, оба порта должны быть одного и того же цвета). Если типы данных различны, то стрелка соединения будет красного цвета вместо черного. Обратите также внимание, что надпись на блоке становится красной, указывая, что с этим блоком что-то не так. Так, же моделирование не может быть запущено, если некоторые порты не подключены.

Далее произведем генерацию синусоидального сигнала в GNURadio. Чтобы сгенерировать синусоидальный сигнал с частотой 1000 Гц и отобразить его во временной и частотной областях. Будем исходить из модели, которая была создана в предыдущем пункте. Для того, чтобы изменить тип выходных данных на float, дважды щелкните на блоке и в окне «Properties», нажмите на «Complex» в разделе «Output type» и выберите «Float», как показано ниже.

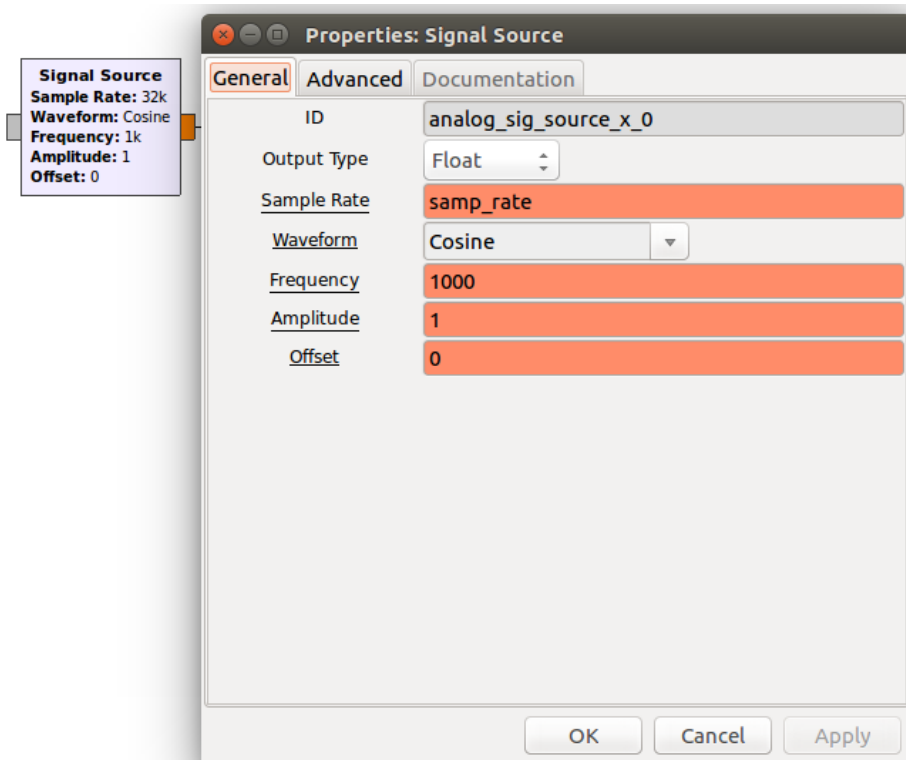


Рисунок 1.7 - Параметры блока «Signal source»

Повторите эти действия для блока «Throttle». Затем выберите «QT» в разделе «Instrumentation» (или просто введите в поиск «QT GUI Sink») и дважды щелкните на «QT GUI Sink». Этот блок позволит вам увидеть форму волны в частотной и во временной области. (Существует несколько видов графических блоков, например «QT GUI» и «WX GUI». Для использования каждого из видов данных блоков необходимо изменить соответствующий пункт в меню настроек блока «Options».) Измените тип данных с «Complex» на «Real» и подключите вход к выходу блока «Throttle». Сохраните модель, например, как LAB_1.grc.

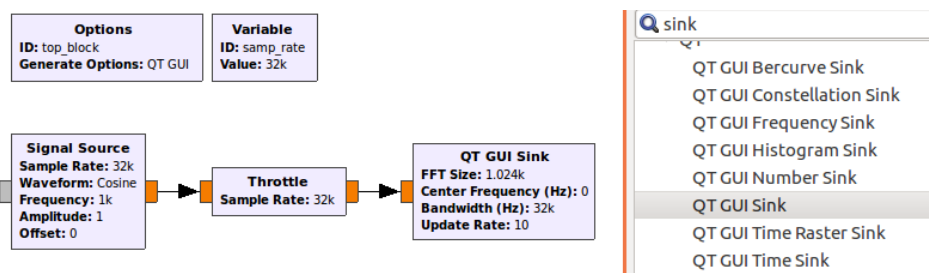


Рисунок 1.8 - Подключение блока «QT GUI Sink»

Теперь вы можете запустить моделирование, нажав на зеленый треугольник в строке меню. Вы можете выбирать между вкладками «Frequency Display» и «Display Time Domain», как показано ниже.

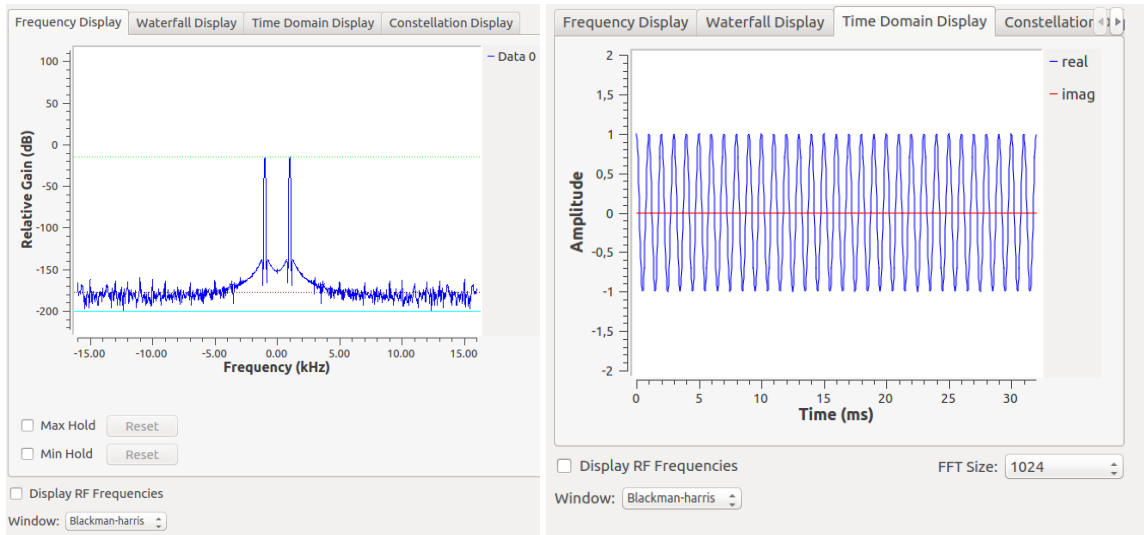


Рисунок 1.9 - «Frequency Display» и «Display Time Domain» блока «QT GUI Sink»

Конечно, вы можете попробовать и другие вкладки и определить, что они показывают. Используйте курсор, чтобы увеличить размер FFT до 4096 или 8192, выбрать типы фильтрующих окон, например, «rectangular» или «Kaiser» и наблюдайте изменения в отображении информации на экране. Следует отметить, что Frequency Display показывает спектральную плотность мощности (PSD), которая по существу пропорциональна квадрату величины преобразования Фурье.

Далее произведем генерацию синусоидального сигнала с переменной частотой и воспроизведем звука.

Откройте ранее сохраненный файл LAB_1.grc, который состоит из «Signal source», в «Throttle», и «QT GUI Sink».

Добавьте на рабочую область блок «QT GUI Range». Дважды щелкнув на блоке, вы можете увидеть его свойства. Измените «ID» из variable_qtgui_range_0 в range_1. Для «Default value» введите 1000. Для «Start» и значения «Stop» введите – 0 и 5000 соответственно. Новые значения параметров приведены ниже.

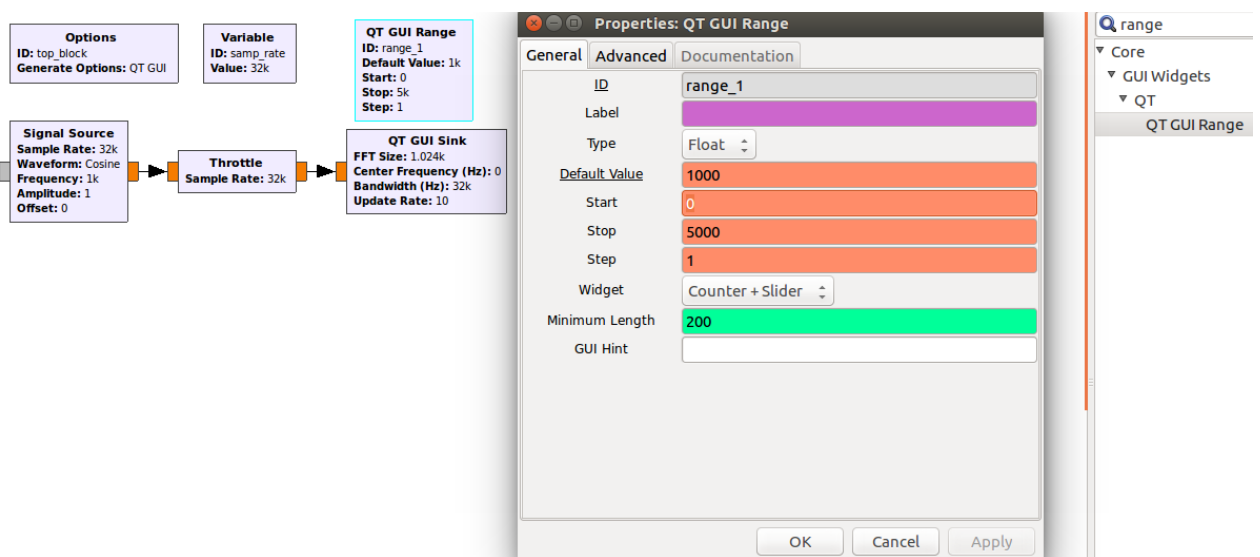


Рисунок 1.10 - Параметры блока «QT GUI Range»

Затем дважды щелкните на блоке «Signal source» и измените запись «Frequency» с 1000 на range_1. Запустите моделирование. При моделировании вы сможете изменять частоту синусоидального сигнала изменяя положение ползунка range_1.

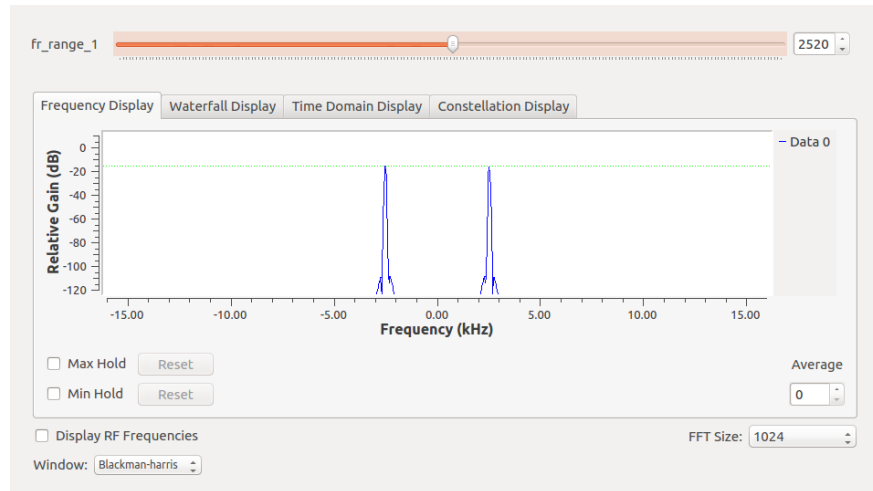


Рисунок 1.11 - Спектр смоделированного сигнала

Чтобы добавить звук, выберите «Audio», а затем дважды щелкните на «Audio Sink».

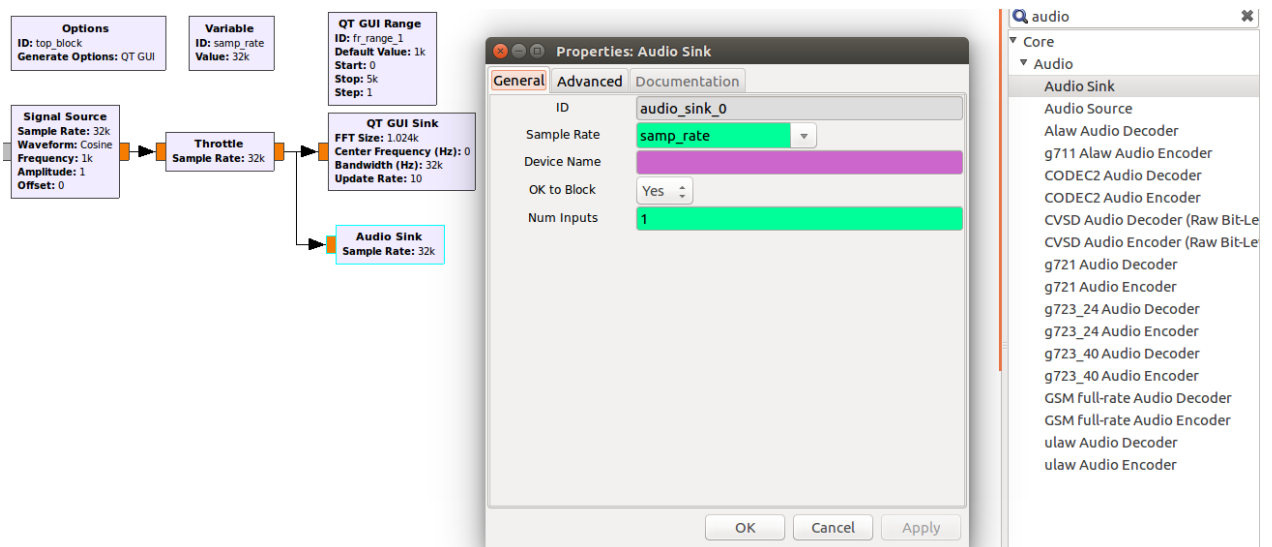


Рисунок 1.12 - Параметры блока «Audio Sink»

Соедините вход «Audio Sink» с выходом «Throttle». Так как «Audio sink» представляет собой аппаратное устройство, то возникает конфликт между скоростями обработки данных в блоке «Audio sink» и остальными блоками модели. Для устранения данного конфликта необходимо преобразовать частоту дискретизации перед блоком «Audio Sink» в 44100 Гц. Для этого установите еще один блок «Throttle» и в его настройках укажите значение «Sample Rate» - 44100.

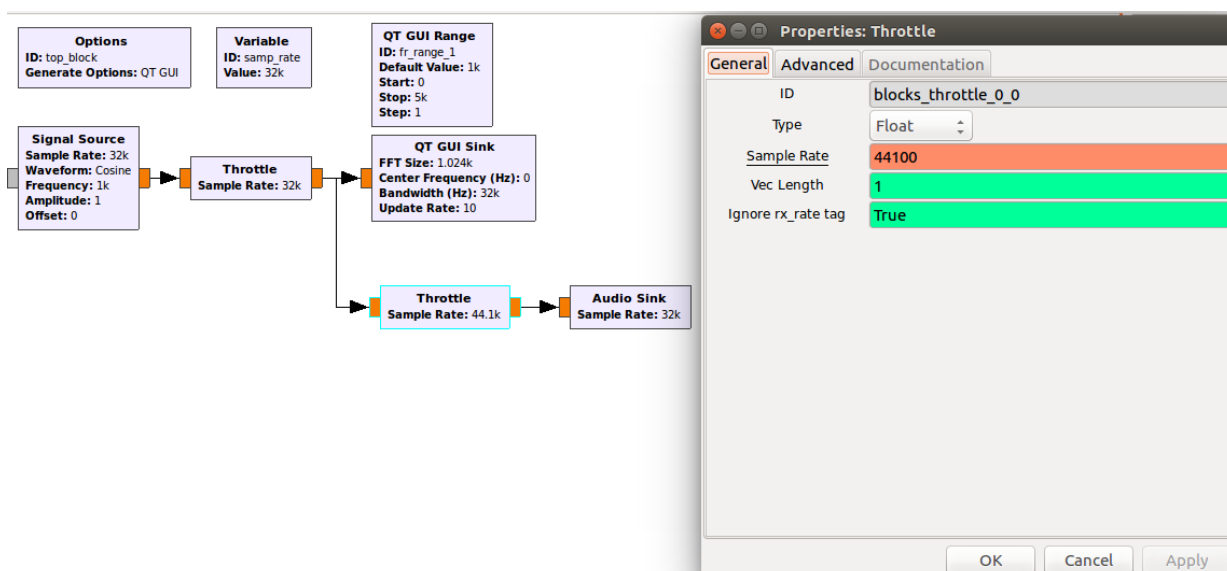


Рисунок 1.13 - Изменение частоты дискретизации для различных блоков GRC

Если запустить моделирование будет происходить изменение частоты «Signal source» от 0 до 5000 Гц при изменении значения range_1. При этом вы будете слышать соответствующий звук. Вы можете увидеть изменение частоты во временной и частотной области при помощи «QT GUI Sink».

Добавьте к модели второй источник сигнала. Для этого поместите в рабочую область блоки «Signal source» и «Add» для того чтобы смешать два сигнала в один поток. Измените в параметрах блока «Signal source» тип генерируемого сигнала с синусоидального на какой-либо другой. Соедините блоки, как показано на рисунке ниже, и запустите моделирование. Вы увидите как изменилась форма сигнала и воспроизводимый моделью звук в соответствии с внесенными в модель изменениями.

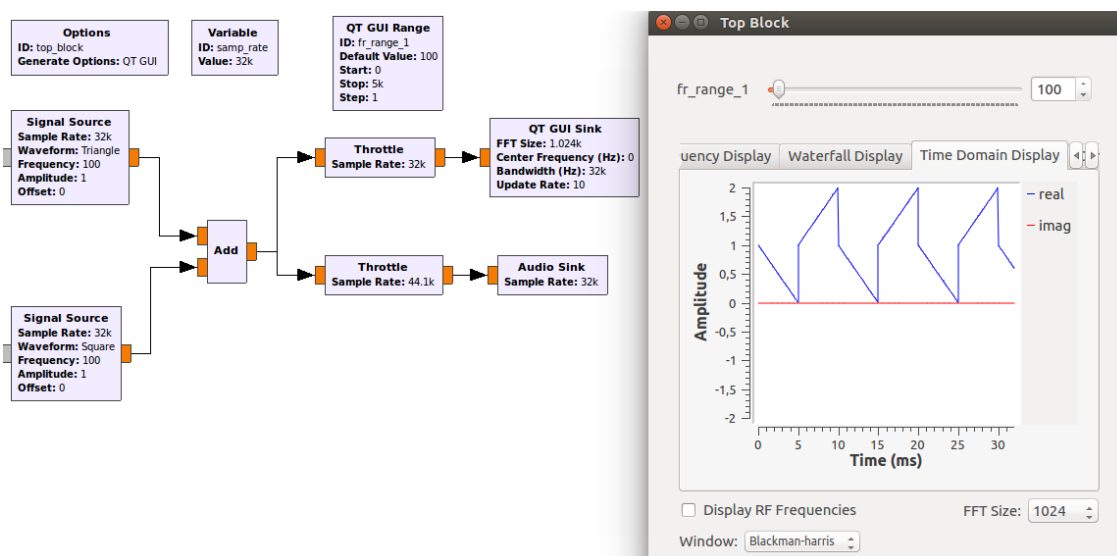


Рисунок 1.14 - Блок-схема модели и представление смоделированного сигнала во временной области

Контрольные вопросы к практической работе

- 1) Какие типы данных используются в программной среде GNU Radio?
- 2) Зачем нужен блок Throttle?
- 3) Что такое частота дискретизации модели? Как она влияет на работу?
- 4) Могут ли блоки из интерфейса QT GUI работать совместно с WX GUI?
- 5) Как зависит скорость обработки блока QT GUI Sink от ширины окна преобразования Фурье?

2. Практическая работа «Цифровой FM-передатчик и приемник»

Цель работы: Реализовать систему передачи FM-сигнала используя программную среду GNU Radio, рассчитать ее основные характеристики. Осуществить прием сигнала с помощью мобильного телефона и программно-определяемой радиосистемы.

Задачи практической работы:

- 1) Составить программу для передачи FM-сигнала среде GNU Radio.
- 2) Произвести передачу FM-сигнала при помощи программно определяемой радиосистемы.
- 3) Составить программу для приема FM-сигнала среде GNU Radio.
- 4) Произвести прием FM-сигнала при помощи программно-определяемой радиосистемы и мобильного телефона (последнее необязательно). Зафиксировать и проанализировать сигнал в различных точках модели в среде GNU Radio.

Краткие теоретические сведения

До 1961 года производилась трансляция монофонических звуковых сигналов по стандартам для AM, FM и TV. FM-вещание в то время также включало вспомогательные коммуникационные услуги, которые были мультиплексированы с основной информацией монофонического канала для обеспечения фоновой музыки и других услуг в офисах и магазинах. В 1961 году была одобрена передача стереофонического звука, который расширяет идею мультиплексирования сигналов для генерации стереофонического аудио сигнала. Одно из основных требований к стереофоническому сигналу заключается в том, чтобы иметь обратную совместимость с большой существующей базой FM моноканальных приемников. Для достижения этой цели полоса от 0 до 15 кГц части мультиплексного сигнала (MPX) должна была содержать левый (L) и правый (R) каналы информации (L + R) для монофонического приема. Стереофонический эффект звука достигается за счет амплитудной модуляции разностного сигнала (L - R) на поднесущей 38 кГц в области от 23 до 53 кГц спектра основной полосы частот. На 19 кГц передается тон пилот-сигнала, который добавляется к мультиплексному сигналу, чтобы дать возможность FM стерео приемникам обнаружить и декодировать стерео сигнал. Формат композитного сигнала основной полосы частот соответствует обратной совместимости, необходимой для FM моно приемников, одновременно обеспечивая достаточную информацию FM стерео ресиверам для декодирования левого и правого каналов. Сегодня сигнал MPX включает в себя поднесущую на 57 кГц, которая несет RDS и RBDS сигналы. На рисунке 2.1 показан спектр сигнала основной полосы частот современных FM радиостанций (важно отметить, что после частотной модуляции спектр будет выглядеть иначе).

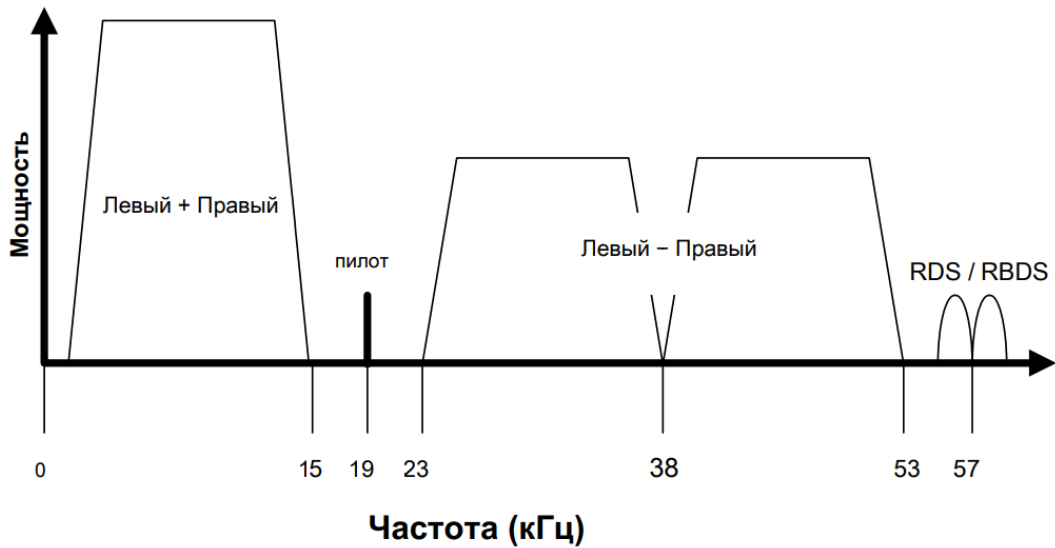


Рисунок 2.1 - Спектр современных FM радиостанций

В соответствии с этим структурная схема FM передатчика имеет следующий вид. $L(t)$ и $R(t)$ обозначают временные формы сигналов левого и правого каналов, а $RDS(t)$ обозначает сигнал RDS / RBDS. Суммарный сигнал $m(t)$ далее поступает на FM модуляцию и передается в эфир. В GNURadio данная схема представлена блоком WBFM transceiver.

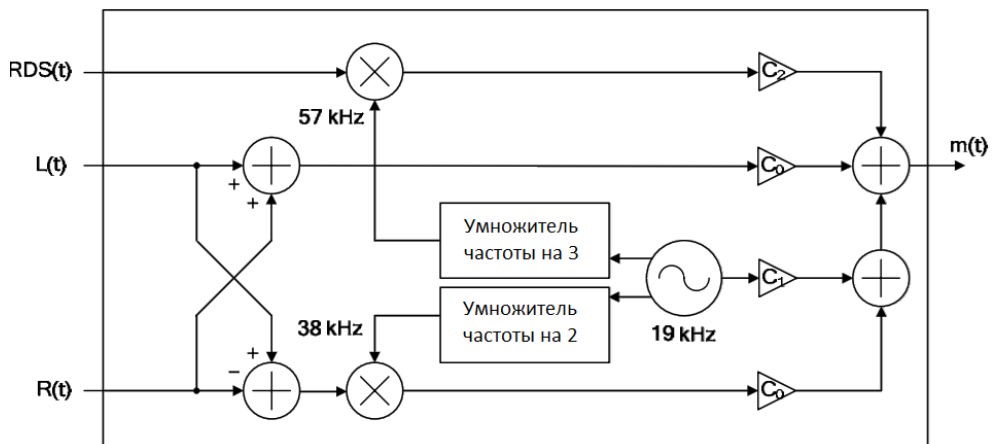


Рисунок 2.2 - Структурная схема FM передатчика

Прием и декодирование сигнала FM радиостанции производится в соответствии с структурной схемой, представленной на рисунке ниже.

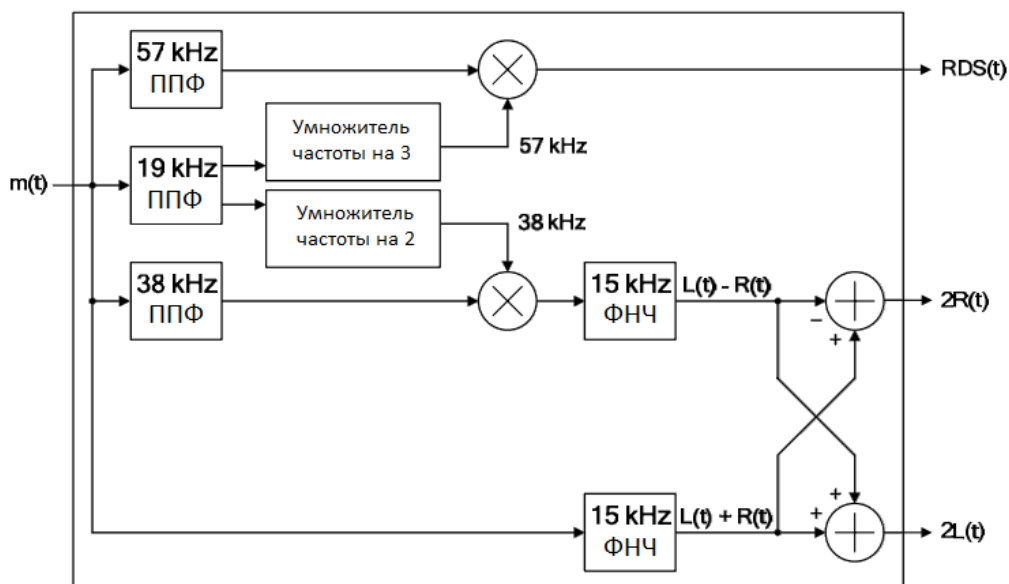


Рисунок 2.3 - Структурная схема FM приемника

На рисунке 2.3 показана структурная схема декодера, используемого для восстановления левого, правого и RDS сигналов из MPX сигнала, $m(t)$. К сигналу применяется три полосовых фильтра с центральными частотами в 19, 38 и 57 кГц, а также фильтр нижних частот с частотой среза 3 дБ на 15 кГц. Полосовой фильтр 19 кГц является узкополосным фильтром, используемым для извлечения 19 кГц-тона пилот-сигнала из сигнала MPX. Затем пилот-сигнал умножается по частоте в два раза и три раза для получения поднесущих 38 и 57 кГц, необходимых для демодуляции (L - R) и RDS сигналов, соответственно. Путем сложения и вычитания сигналов (L + R) и (L - R), можно получить удвоенные значения левого и правого каналов, необходимых для стереофонического звука.

Ход выполнения работы

На данном этапе будет осуществлена реализация FM-передатчика с использованием частотной модуляции. Частотная модуляция использует спектр частот, значительно превосходящий максимальную частоту модулирующего сигнала. Данный тип модуляции применяется при высококачественном радиовещании и для звукового сопровождения телевизионных передач.

Для реализации FM-передатчика запустите gnuradio-companion и создайте в программе блок-схему, которая представлен на рисунке 2.4.

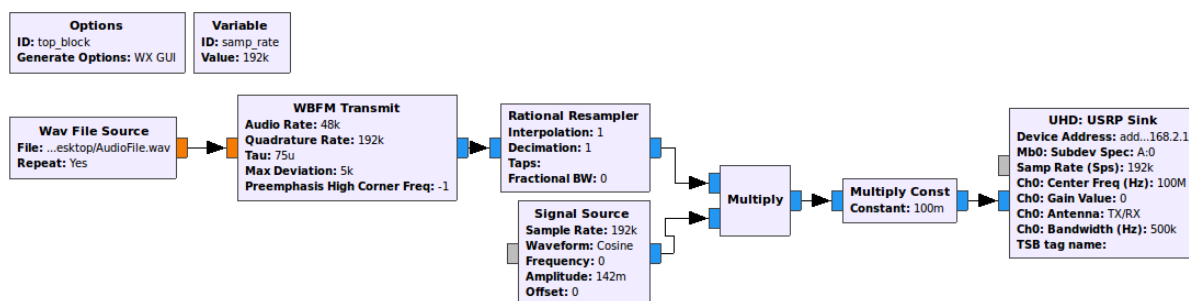


Рисунок 2.4 - Блок-схема FM передатчика

При помощи данной схемы можно передать в эфир звуковой файл через передатчик USRP и получить его на мобильный телефон или на USRP приемник. Таким образом, мы будем использовать источник звука, который является файлом в формате WAV. В блоке WAV

File Source необходимо выбрать источник входного файла, который хранится на нашем компьютере. Соответствующие настройки блока WAV File Source представлены на рисунке 2.5.

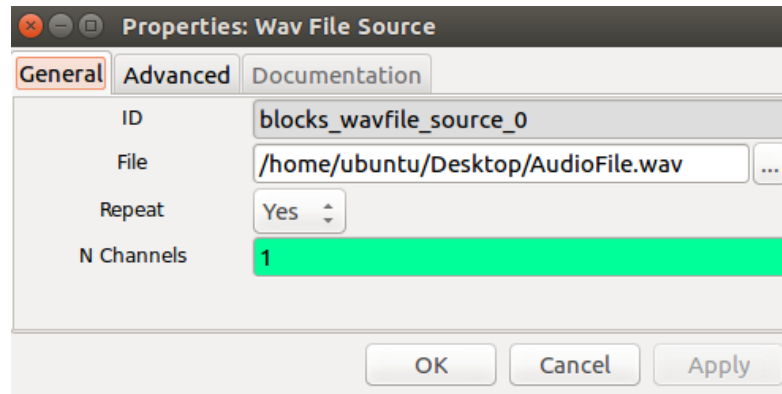


Рисунок 2.5 - Настройки блока WAV File Source

Далее сигнал поступает в блок WBFM Transmit. Этот блок представляет собой модулятор WBFM, структурная схема которого была представлена в теоретической части данной практической работы (рисунок 2.3). Для корректной работы блоков необходимо соблюдать частоту дискретизации входных и выходных данных. Так как исходный WAV файл имеет частоту дискретизации 48kHz, то необходимо установить следующие параметры блока WBFM Transmit (рисунок 2.6).

- 1) Audio rate: 48 kHz
- 2) Quadrature Rate: $48 \text{ kHz} * 4 = 192 \text{ kHz}$
- 3) Tau: 75u

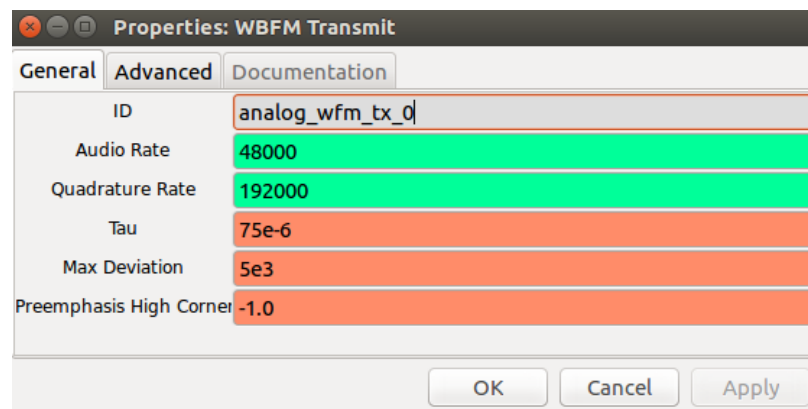


Рисунок 2.6 - Настройки блока WBFM Transmit

Теперь модулированный сигнал из блока WBFM Transmit отправляется в блок Rational Resampler, где мы делаем некоторые преобразования частоты.

Например, WAV File Source требует более низкой частоты дискретизации, а блок USRP Source требует более высокой частоты дискретизации так как этого требует аппаратная составляющая оборудования. Блок Rational Resampler преобразует частоту дискретизации, делая интерполяцию или децимацию сигнала в соответствии с необходимыми требованиями. Настройки блока Rational Resampler представлены на рисунке 2.7.

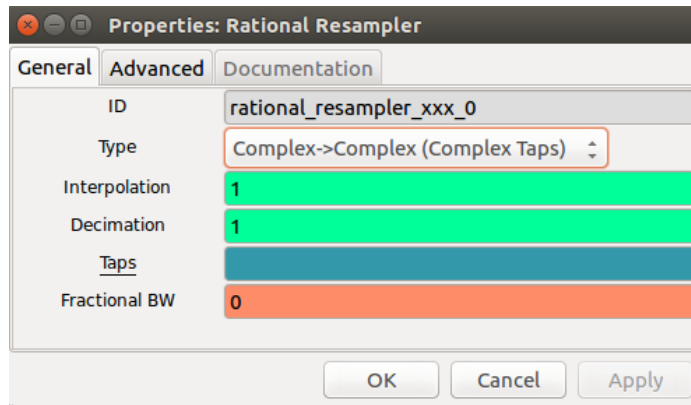


Рисунок 2.7 - Настройки блока Rational Resampler

Сигнал с блока Rational Resampler подается на один из входов умножителя (Multiply), а на второй вход умножителя подается косинусоида с частотой 192 кГц.

Теперь этот умноженный сигнал подается на блок USRP Sink, при помощи которого сигнал поступает на передающую антенну USRP. В данной практической работе мы используем 100 МГц в качестве центральной частоты, поэтому значение center frequency задается равным 100e6. Усиление сигнала не рекомендуется задавать более 10дБ, так как это приводит к искажениям передаваемого сигнала. Если сигнал имеет малую амплитуду его можно усилить в блоке Multiply, умножая на необходимую константу. Антенна соответствующего диапазона подключается к гнезду TX / RX платы USRP B210. Параметры блока USRP Sink представлены на рисунке 2.8. Для правильного приема сигнала из эфира, приемник необходимо настроить в точности с теми же характеристиками, как и передатчик. В пункте Device Arguments необходимо вписать "b210".

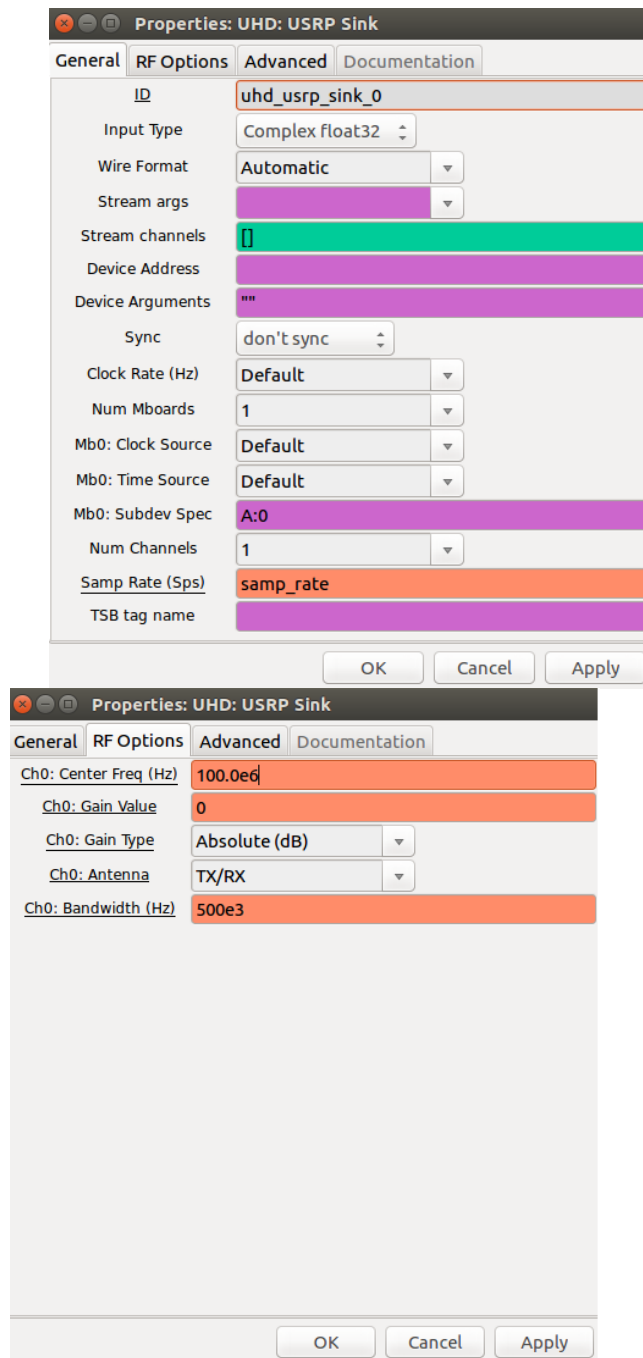


Рисунок 2.8 - Настройки блока USRP Sink

Запустите полученную блок-схему и зафиксируйте спектр сигнала на входе USRP Sink, проанализируйте полученные результаты.

Для прослушивания передаваемого звукового сигнала на любом мобильном телефоне, необходимо подключить гарнитуру и настроить FM-приемник мобильного телефона на частоту передачи, которая равна 100 МГц. На Рисунке 9 показана настройка мобильного телефона.

Для приема FM-сигнала в GNU Radio, в качестве источника входного сигнала используется блок USRP Source с центральной частотой 100 МГц и другими параметрами, которые показаны на рисунке 12. При этом частота дискретизации равна 192 кГц, а коэффициент усиления может быть установлен от 0 до 50 в зависимости от ослабления принимаемого сигнала. Антенна соответствующего диапазона подключается к гнезду RX-2 платы USRP B210.

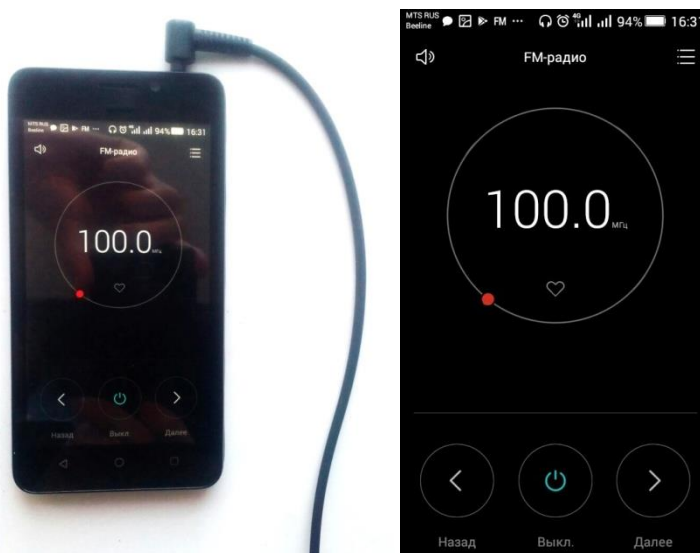


Рисунок 2.9 - Настройка мобильного телефона

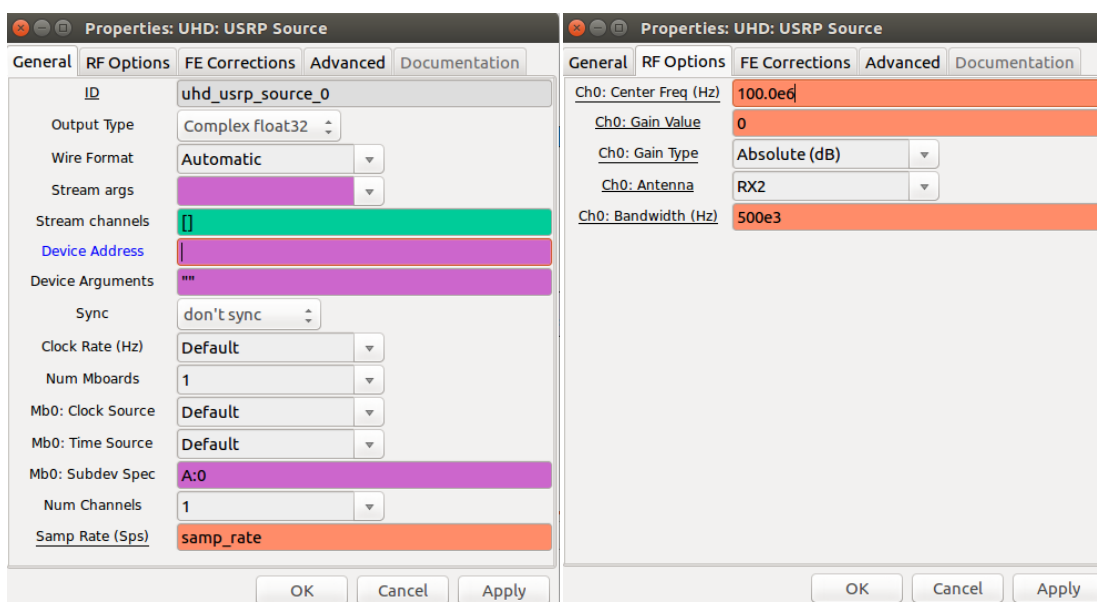


Рисунок 2.10 - Настройки блока USRP Source

Для правильного приема сигнала из эфира, приемник необходимо настроить в точности с теми же характеристиками, как и передатчик. В пункте Device Arguments необходимо вписать "b210".

Далее сигнал поступает на фильтр низких частот. При этом фильтр пропускает сигнал ниже частоты среза и удаляет более высокие частоты, на которых передается сигнал других FM-радиостанций.

Сигнал после фильтрации поступает в блок WBFM Receive. Этот блок является демодулятором FM-сигнала, структурная схема которого показана на рисунке 34. После прохождения WBFM Receive мы получаем исходный поток данных. Настройки данного блока представлены на рисунке 2.

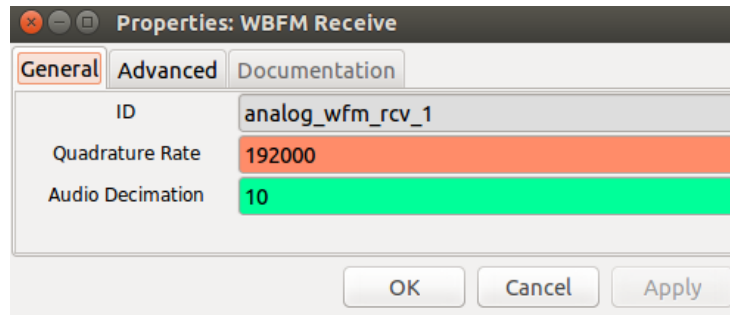


Рисунок 2.11 - Настройки блока WBFM Receive

С выхода блока WBFM Receive сигнал поступает в блок на Rational Resampler, назначение которого описывалось ранее.

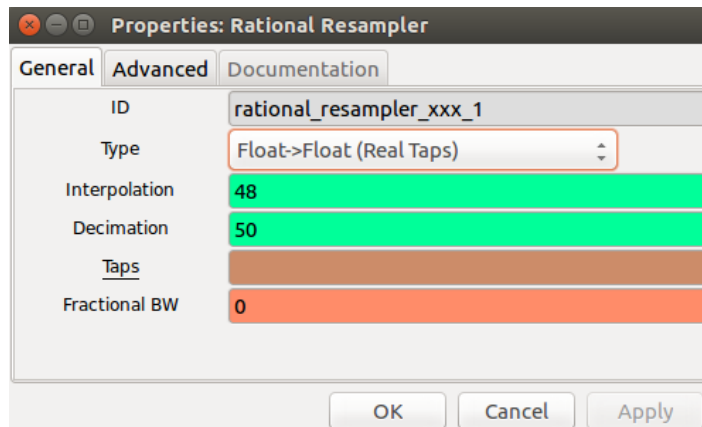


Рисунок 2.12 - Rational Resampler

После преобразования частоты дискретизации, до необходимого блоку Audio Sink значения, амплитуда сигнала изменяется путем умножения на 0,3. Данное преобразование необходимо для устранения нелинейных искажений аудиосигнала. Если же при приеме громкость аудиосигнала мала, то необходимо изменить параметры блока Multiply constant в соответствии с желаемой громкостью. Затем этот сигнал поступает в блок Audio Sink для прослушивания аудиосигнала. Настройки блока Audio Sink представлены на рисунке 2.13. После всех проделанных манипуляций блок-схема выглядит следующим образом (рисунок 2.14). Запустите моделирование. При правильной настройке блок-схемы вы должны услышать передаваемый аудиосигнал.

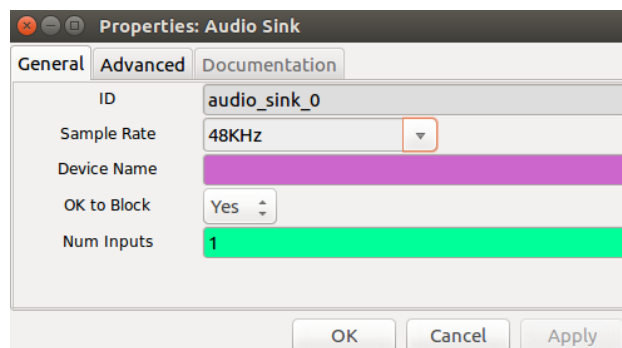


Рисунок 2.13 - Audio Sink

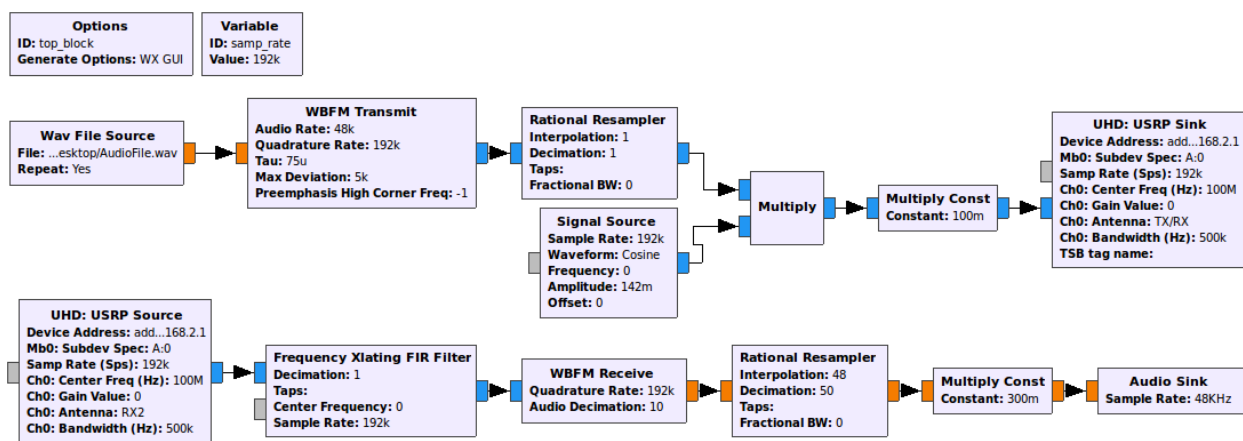


Рисунок 2.14 - Блок-схема FM передатчика и приемника

Задания для самостоятельного выполнения

1) В качестве результатов моделирования зафиксируйте спектры сигналов в следующих точках и проанализируйте результаты:

- на входе блока USRP Sink;
- на выходе блока USRP Source;
- на выходе блока Filter;
- на выходе блока WAV File Source;
- на выходе блока WBFM Transmit;
- на выходе блока WBFM Receive.

2) Найдите частоту вещания некоторых FM-радиостанций в Вашем городе. Настройте блок-схему таким образом, чтобы осуществить прием на частоте выбранной радиостанции. При правильной настройке вы услышите аудиосигнал, передаваемый данной радиостанцией. Постройте соответствующие спектры сигналов в точках, аналогичных пункту 1. Проанализируйте полученные результаты и сравните их с полученными ранее в пункте 1.

Контрольные вопросы к практической работе

- 1) Чем стереофонический сигнал отличается от монофонического?
- 2) Почему на частотах 88-108 МГц применяется в основном частотная модуляция сигналов?
- 3) Что такое избирательность по соседнему и зеркальному каналу?
- 4) Какие виды фильтров вы знаете? Чем полосовой отличается от режекторного фильтра?
- 5) Можно ли произвести прием сигнала без подключения каких либо усилительных элементов к антенне (в цепях приемника только пассивные компоненты)?

3. Практическая работа «Цифровая модуляция QAM-M»

Цель работы: изучить основные типы квадратурной модуляции, применяемые в современных системах передачи информации.

Задачи практической работы:

- 1) Составить программу с применением QAM-модуляции в программной среде GNU Radio.
- 2) Произвести передачу и прием потока бит по реальному каналу. Зафиксировать и проанализировать изменение сигнала при различных настройках в модели GNU Radio.
- 3) Составить программу с применением QAM-модуляции для оценки отношения сигнал/шум.

Краткие теоретические сведения

Квадратурная амплитудная модуляция (с англ. Quadrature amplitude modulation) широко используется во многих видах цифровой радиосвязи. Доступны различные формы QAM, а некоторые из наиболее распространенных форм включают: 16 QAM, 32 QAM, 64 QAM, 128 QAM и 256 QAM. Здесь цифры относятся к числу позиций на созвездии, т. е. количеству различных состояний, которые могут существовать. Различные варианты QAM могут использоваться, когда ресурс PSK-модуляции не позволяет добиться требуемой скорости для системы передачи информации. Это связано с тем, что QAM достигает большего расстояния между соседними точками в плоскости I-Q, распределяя точки более равномерно. Таким образом точки на созвездии более четкие, а ошибки данных уменьшаются. Хотя можно передавать больше бит на символ, например QAM-256, но если энергия созвездия остается неизменной, точки на созвездии должны быть ближе друг к другу, и передача становится более восприимчивой к шуму. Это приводит к более высокой вероятности битовой ошибки, чем для вариантов QAM более низкого порядка. Таким образом, существует баланс между получением более высоких скоростей передачи данных и поддержанием приемлемой частоты ошибок в бит для любой системы радиосвязи.

На данный момент QAM-модуляция используется практически во всех системах радиосвязи. Например, в цифровом телевизионном стандарте DVB-T/T2 (с англ. Digital Video Broadcasting) стандарте используется QAM-64 и QAM-256, а в США QAM-64 и 256 QAM-256 используются в сетях кабельного телевидения, стандартизированные SCTE в стандартном ANSI / SCTE 07 2000. В дополнение к этому, варианты QAM также используются для многих беспроводных и сотовых технологий. Диаграммы созвездий показывают распределение точек для состояний в разных формах QAM, квадратурной амплитудной модуляции. По мере увеличения порядка модуляции увеличивается количество точек на диаграмме созвездий QAM.

На приведенных ниже рисунках можно наблюдать диаграммы созвездий для QAM-16 и QAM-64.

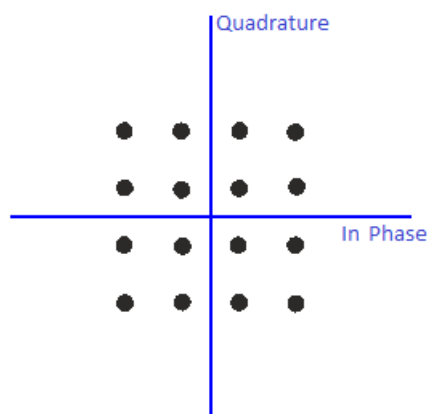


Рисунок 3.1 - Диаграмма созвездий для QAM-16

Особенность использования QAM заключается в том, что это форма модуляции способна переносить несколько бит информации на символ, а выбирая формат QAM более высокого порядка, скорость передачи данных может быть увеличена.

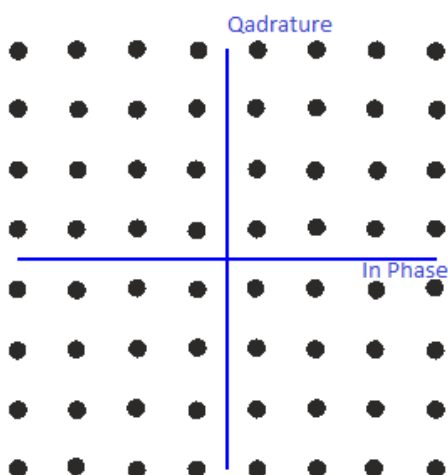


Рисунок 3.2 - Диаграмма созвездий для QAM-64.

В приведенной ниже таблице показано, сколько бит на символ модуляции может быть передано для различных форм QAM и PSK.

Таблица 3.1 - Зависимость количества бит на символ от порядка QAM

Тип цифровой модуляции	Кол-во бит на символ модуляции
QPSK или QAM-4	2
8-PSK	3
16-QAM	4
64-QAM	6

QAM более высокого порядка способна обеспечить гораздо более быструю скорость передачи данных и более высокий уровень спектральной эффективности для системы радиосвязи, но не стоит забывать, что такие типы модуляции значительно менее устойчивы к шуму и помехам.

В результате этого многие системы радиосвязи теперь используют технологии динамической адаптивной модуляции. Они ощущают условия канала и адаптируют схему модуляции для получения максимальной скорости передачи данных для данных условий. По мере уменьшения отношения сигнал / шум ошибки будут увеличиваться вместе с повторными передачами данных, тем самым замедляя пропускную способность. Возвращаясь к схеме моду-

ляции более низкого порядка, связь может быть более надежной с меньшим количеством ошибок данных и повторной отправкой.

Ход выполнения работы

В программной среде GNU Radio составить программу, показанную на рисунке 3.3 и передать поток бит по реальному каналу. Далее приводится описание работы программы.

При помощи генератора случайных чисел формируются биты и подаются на QAM-модулятор, параметры которого задаются переменными. Для правильной работы передатчика B210 необходимо уменьшить значение амплитуды на выходе модулятора (если этого не сделать могут возникнуть ошибки). Уменьшение амплитуды производится умножением на константу. Далее сигнал поступает на вход передатчика B210, производится настройка несущей частоты, параметров усиления и др. Проходя через реальный канал, сигнал поступает на приемник B210, который настроен по тем же основным характеристикам, что и передатчик. После приема, осуществляется первичная обработка блоком синхронизации. Стоит заметить, что после блока синхронизации созвездие QAM может быть произвольно повернуто, что связано с влиянием среды при прохождении реального канала. Для поворота созвездия, можно применить умножение на комплексное число и наблюдать его в блоке виртуального монитора диаграммы созвездий. Поворот созвездия никак не влияет на правильность демодуляции, поскольку алгоритмы Python, применяемые в QAM-демодуляторе учитывают частотную девиацию и отклонения сигнала по фазе. Итак, после умножения на комплексное число сигнал демодулируется QAM-демодулятором и выводится на виртуальный осциллограф через согласующий блок. Параметры каждого блока указаны на рисунках 3.4 - 3.16.

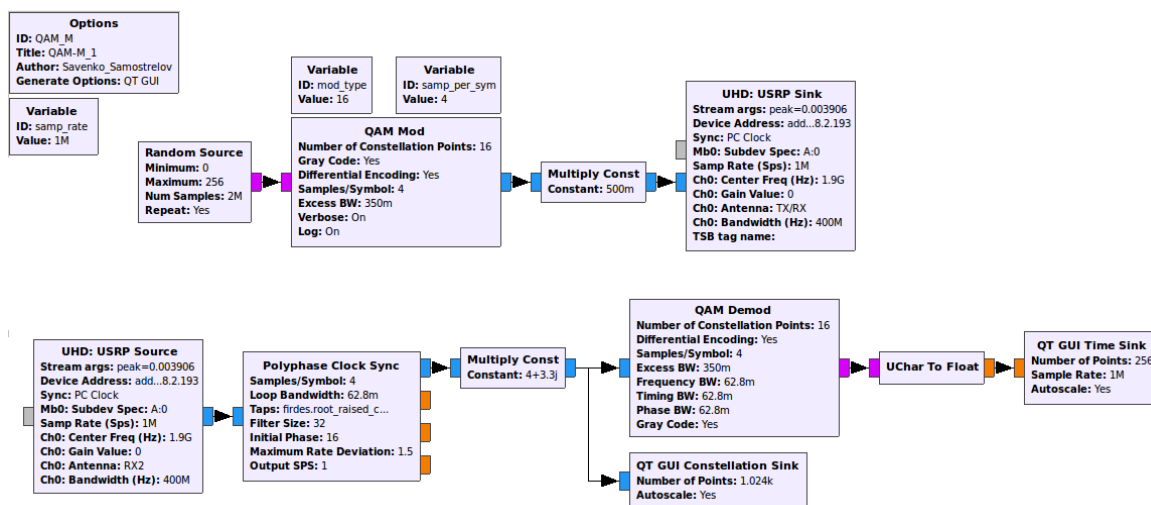


Рисунок 3.3 - Общая схема приемника и передатчика с применением QAM-модуляции в GNU Radio

В начале работы производится настройка частоты дискретизации. Для этого необходимо создать переменную как показано на рисунке ниже.

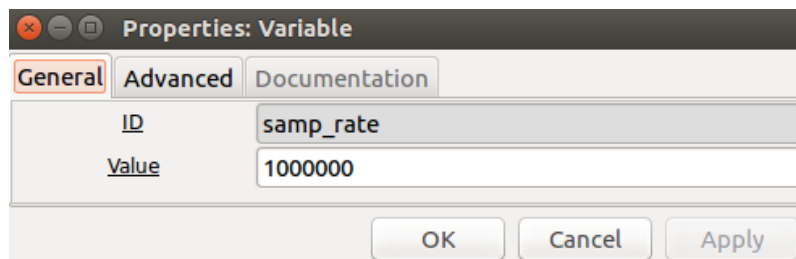


Рисунок 3.4 - Настройка частоты дискретизации системы.

Чтобы подвергнуть биты модулированию, создаем QAM-модулятор и производим его настройку согласно рисункам, представленным ниже.

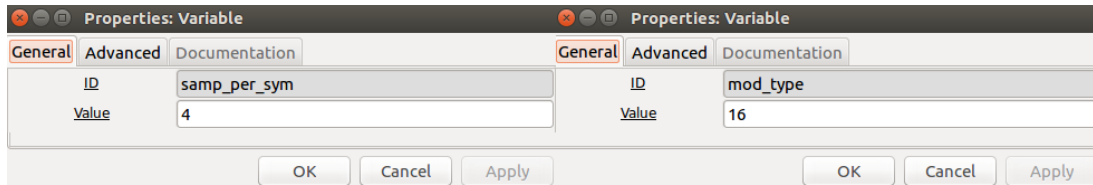


Рисунок 3.5 - Настройка переменных для QAM-модулятора

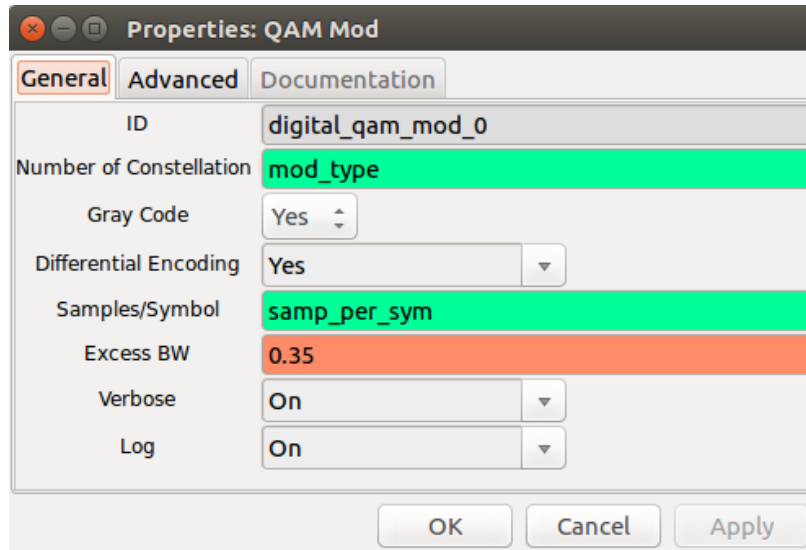


Рисунок 3.6 - Настройки для QAM-модулятора.

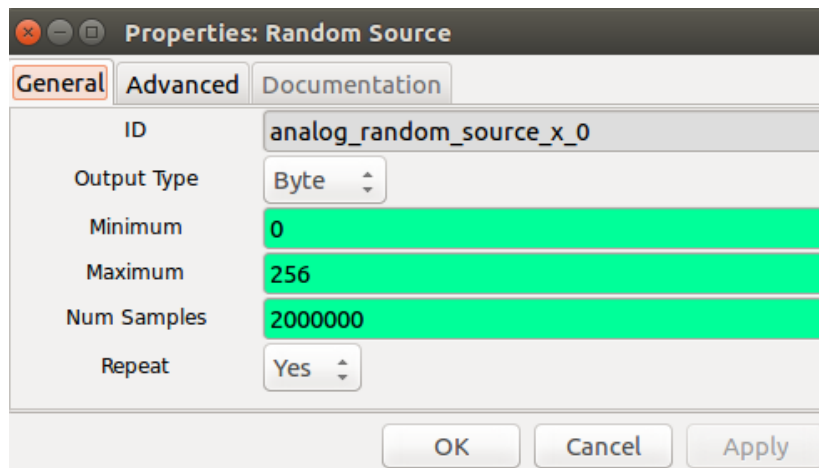


Рисунок 3.7 - Настройки для генератора случайных чисел.

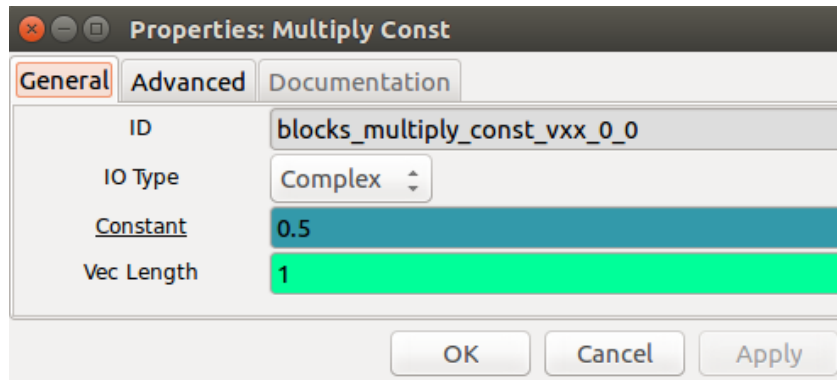


Рисунок 3.8 - Добавление константы для регулировки амплитуды сигнала на выходе QAM-модулятора.

Важным пунктом является точная настройка передатчика, характеристики которого приведены далее на рисунках. В пункте Device Arguments необходимо вписать "b210".

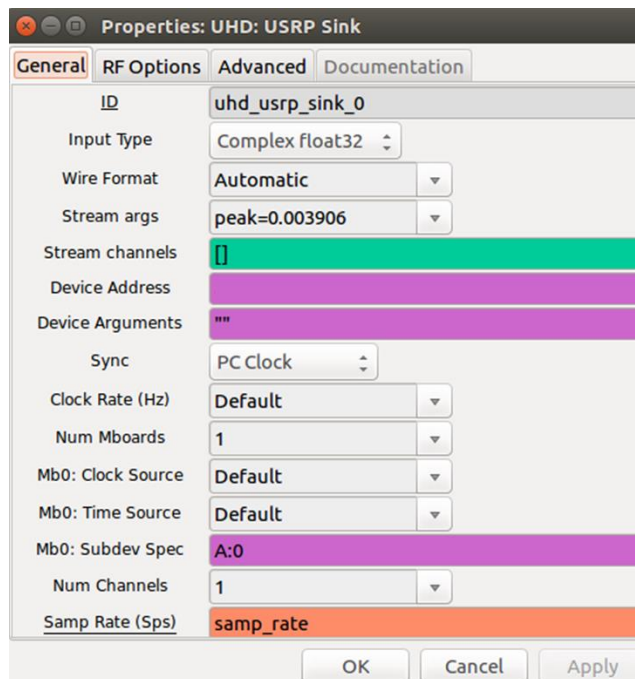


Рисунок 3.9 - Настройка общих параметров для передающей части NI-USRP (Часть 1)

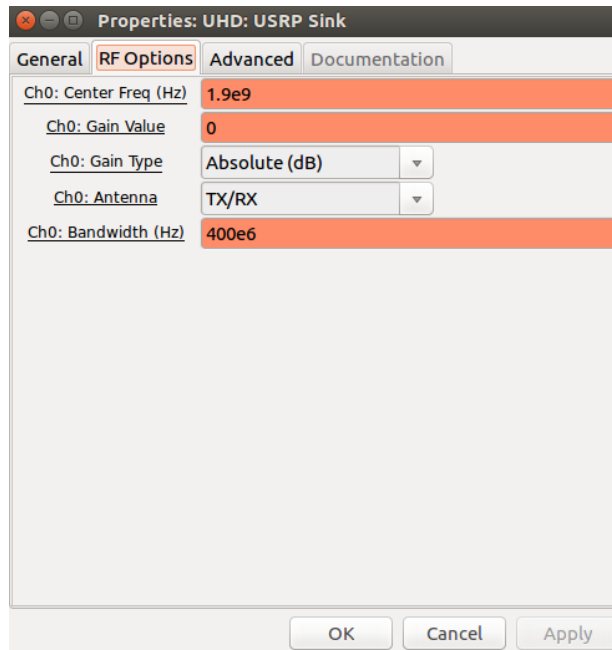


Рисунок 3.10 - Настройка частоты, выбор антенн и других параметров для передающей части NI-USRP (Часть 2)

Для правильного приема сигнала из эфира, приемник необходимо настроить в точности с теми же характеристиками, как и передатчик. В пункте Device Arguments необходимо вписать "b210".

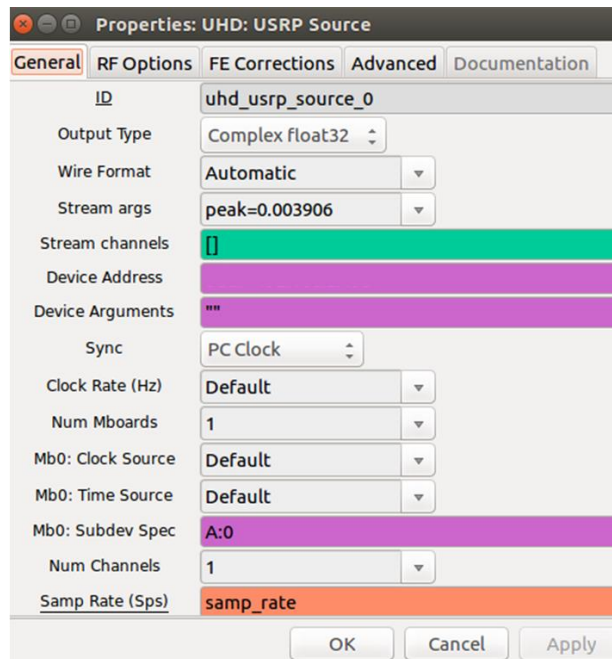


Рисунок 3.11 - Настройка общих параметров для приёмной части NI-USRP (Часть 1)

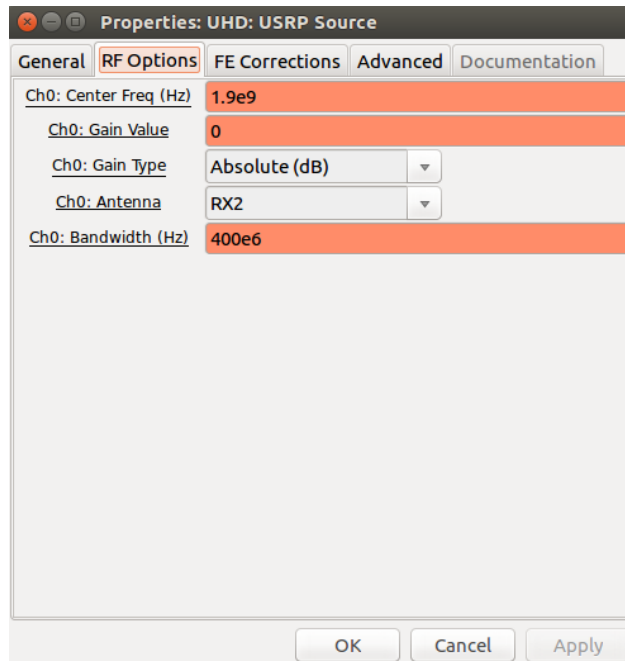


Рисунок 3.12 - Настройка частоты, выбор антенн и других параметров для приёмной части NI-USRP (Часть 2)

Блок синхронизации настраивается в соответствии с алгоритмами Python, по которым работает блок демодуляции QAM. Будьте предельно внимательны при настройке.

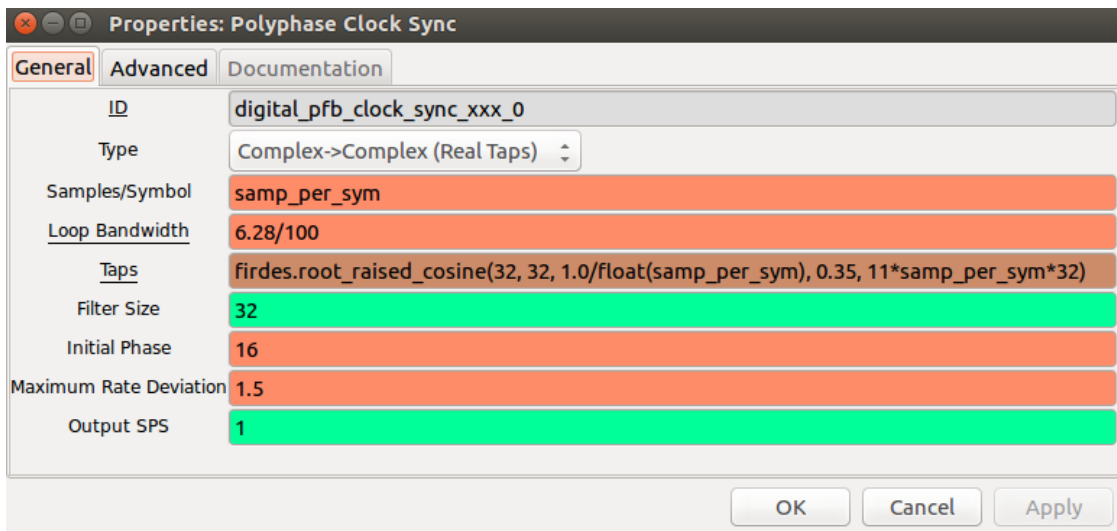


Рисунок 3.13 - Настройка для блока синхронизации для QAM-сигнала.

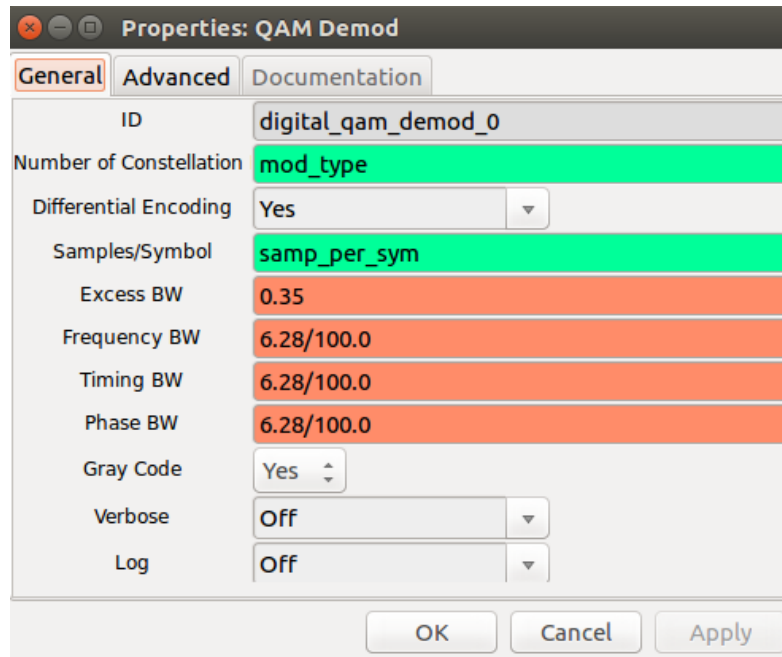


Рисунок 3.14 - Настройка для блока QAM-демодулятора.

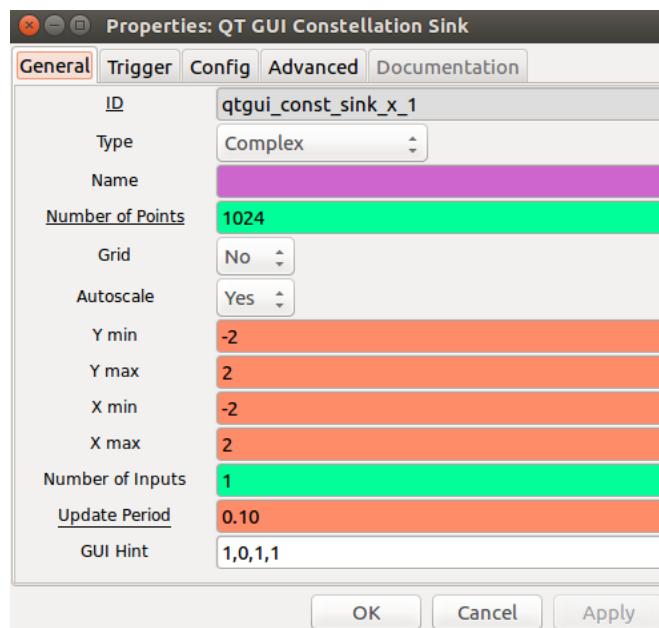


Рисунок 3.15 - Настройки для блока просмотра диаграммы созвездий

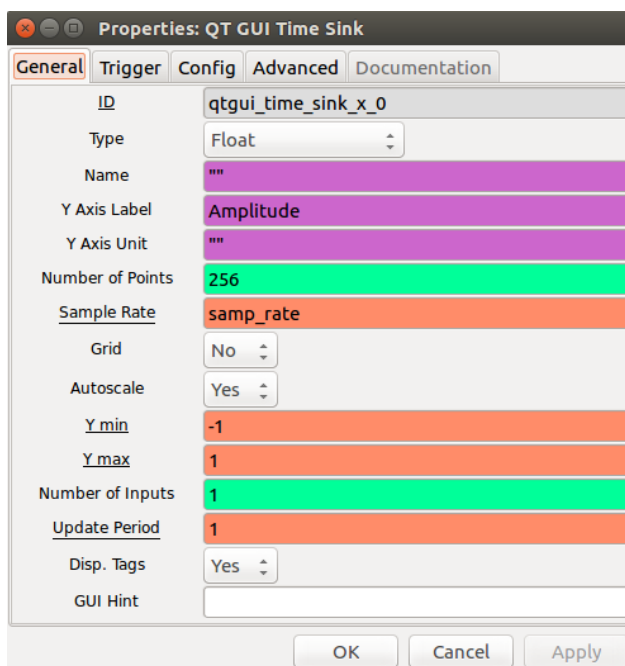


Рисунок 3.16 - Настройки для блока для просмотра сигнала во временной области

После правильного составления программы в GNU Radio, необходимо произвести запуск. Ниже на рисунках приведены графики, полученные в результате прохождения сигнала через реальный канал.

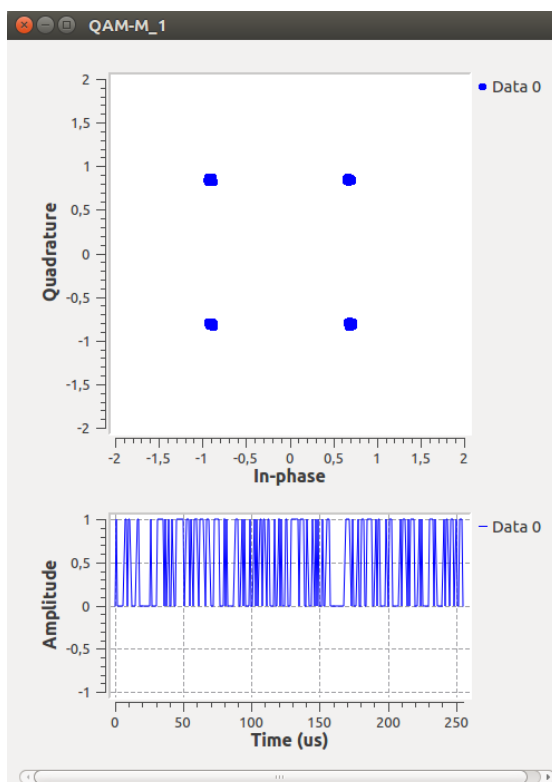


Рисунок 3.17 - Диаграмма созвездий и биты на выходе во временной области для QAM-4

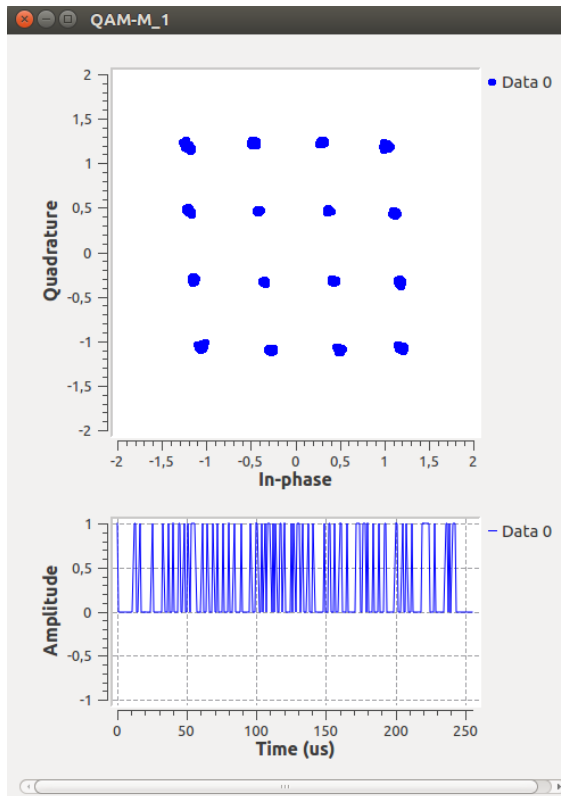


Рисунок 3.18 - Диаграмма созвездий и биты на выходе во временной области для QAM-16

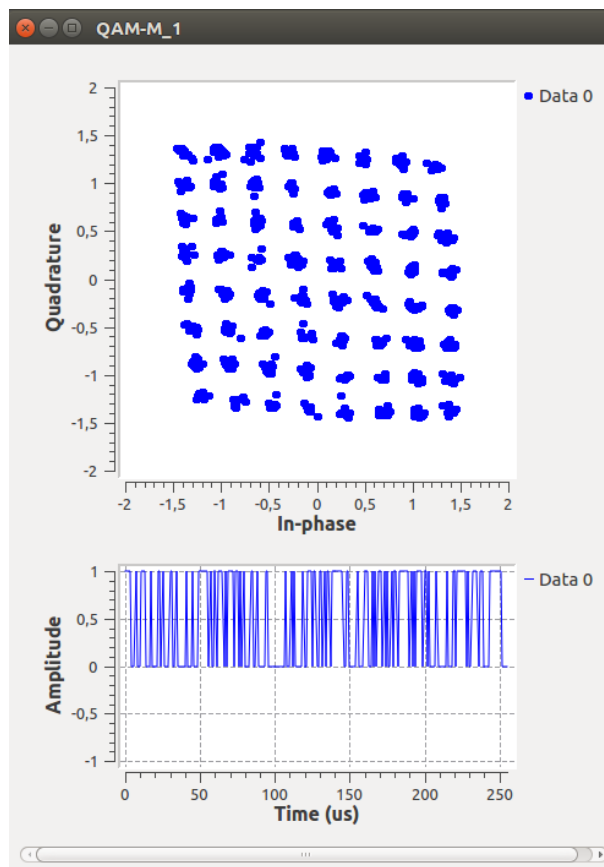


Рисунок 3.19 - Диаграмма созвездий и биты на выходе во временной области для QAM-64

Задания для самостоятельного выполнения

1) Произвести поочередно переключение типов модуляции QAM-4, QAM-16, QAM-64, подключив вместо антенны к передатчику В210 спектрометр и зафиксировать изменения спектра сигнала.

2) Произвести поочередно переключение типов модуляции QAM-4, QAM-16, QAM-64, подключив вместо антенны к передатчику В210 осциллограф и зафиксировать изменения уровней амплитуды сигнала.

4. Практическая работа «QAM/PSK модуляция и измерение BER»

Цель работы: Реализовать систему передачи информации с QAM/PSK модуляцией, используя программную среду GNU Radio, рассчитать значение битовой ошибки (BER) для данной системы.

Задачи практической работы:

- 1) Составить программу с применением QAM-модуляции в программной среде GNU Radio.
- 2) Составить программу с применением PSK-модуляции в программной среде GNU Radio.
- 3) Произвести передачу и прием потока бит по реальному каналу.
- 4) Произвести измерение отношения вероятности битовой ошибки к отношению сигнал/шум для различных типов QAM и PSK модуляции.

Ход выполнения работы

Составить программу с применением QAM-модуляции для оценки отношения сигнал/шум. Произвести измерение отношения вероятности битовой ошибки к отношению сигнал/шум для различных типов QAM.

Схема приведенная ниже позволяет произвести моделирование зависимости вероятности битовой ошибки от отношения сигнал/шум для трех типов модуляции BPSK, QAM-4 и 8-PSK. Принцип работы схемы. Генератор случайных чисел, настройки которого напрямую зависят от выбранного типа модуляции, формирует биты, которые подвергаются преобразованию в комплексные числа в блоке-аналоге QAM-модулятора, который был представлен в предыдущем эксперименте. Далее эти комплексные числа суммируются с Гауссовским белым шумом (тоже комплексные числа), величина которого так же зависит от типа выбранной модуляции. После суммирования с шумом, комплексные числа снова преобразуются в биты и поступают на блок измерения вероятности битовой ошибки. Для наглядности в схеме предусмотрен монитор диаграммы созвездий и монитор просмотра чисел. Изменяя значения переменных, можно менять типы модуляции, и наблюдать, при каком отношении сигнал/шум произойдет появление ошибок. Параметры каждого блока указаны на рисунках 4.2 - 4.17.

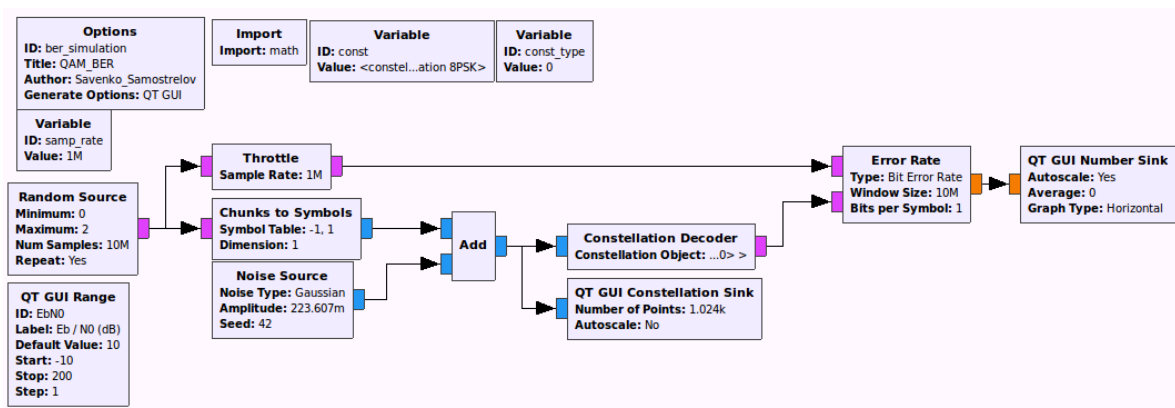


Рисунок 4.1 - Схема для измерения вероятности битовой ошибки в зависимости от отношения сигнал/шум для BPSK, QAM-4 и 8-PSK

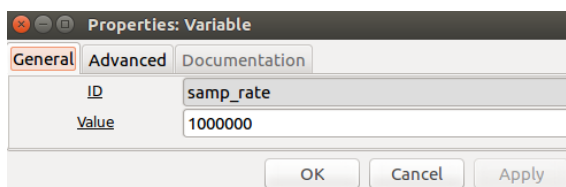


Рисунок 4.2 - Настройка частоты дискретизации

Для правильной работы формул, необходимо произвести импорт библиотек, отвечающих за математические операции.

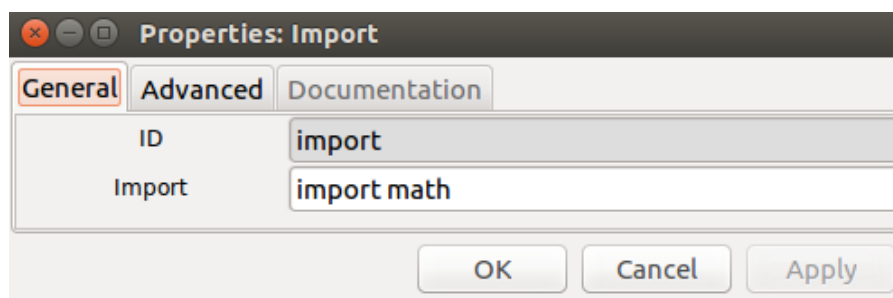


Рисунок 4.3 - Импорт библиотеки "Math"

Чтобы быстро изменять типы модуляции, создаем переменную, в которой прописываем все, как показано на рисунке ниже.

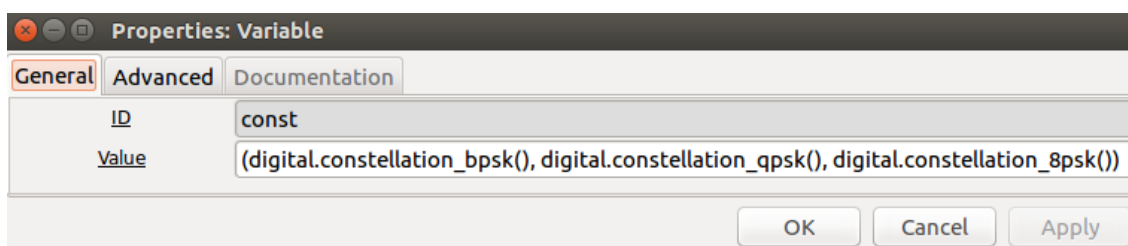


Рисунок 4.4 - Настройка параметров для разных типов модуляции.

Для выбора типа модуляции создаем еще одну переменную. Параметры переменной приведены на рисунке ниже.

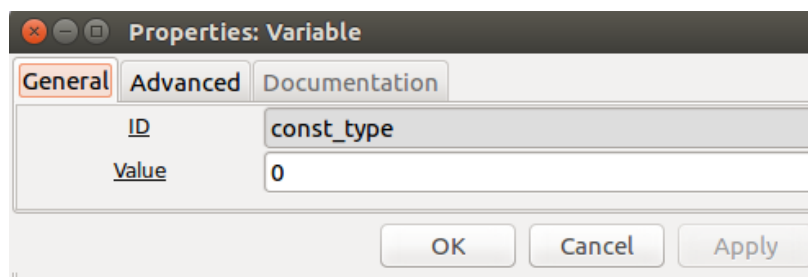


Рисунок 4.5 - Настройка переменной для выбора типа модуляции

Для правильной работы программы, необходимо, чтобы на выходе генератора случайных чисел максимальное значение не превышало лимит, установленный алгоритмами Python. Настройка производится в точности, как показано на рисунке.

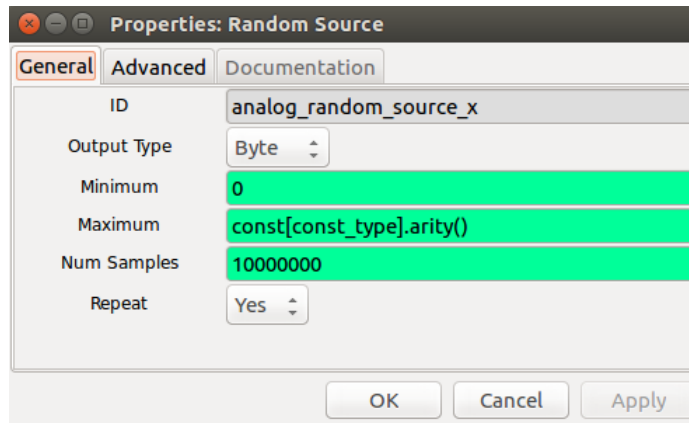


Рисунок 4.6 - Настройки для генератора случайных чисел

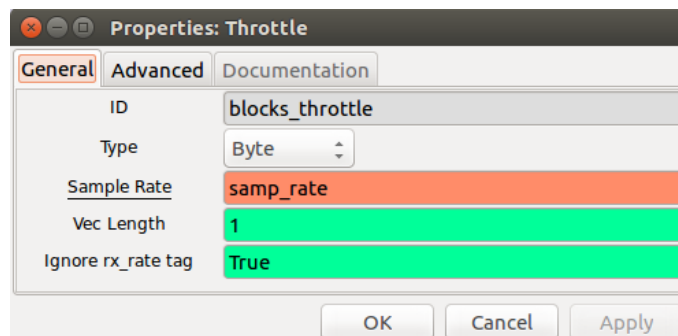


Рисунок 4.7 - Настройки для блока Throttle

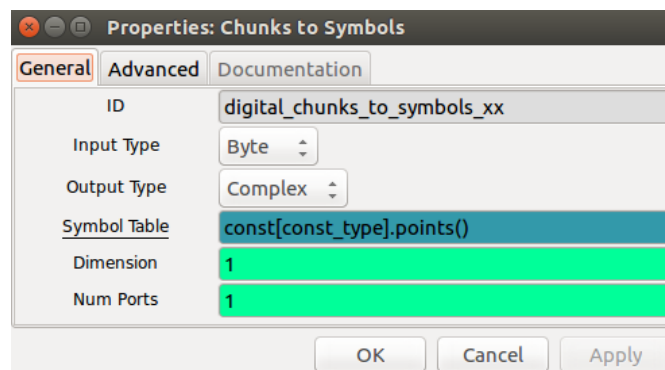


Рисунок 4.8 - Настройки для модулятора QAM/PSK

Преобразование в комплексные числа осуществляется в зависимости от выбранного типа модуляции, заданного переменными.

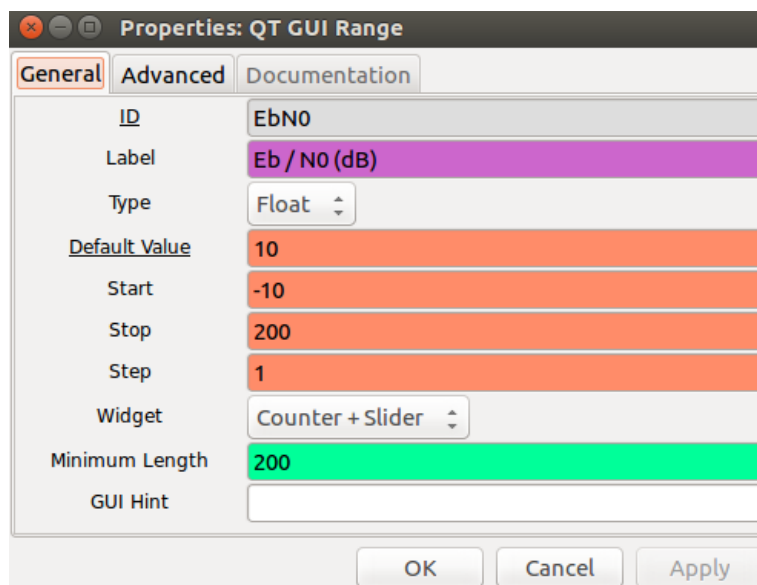


Рисунок 4.9 - Настройки для регулировки отношения сигнал/шум в модели

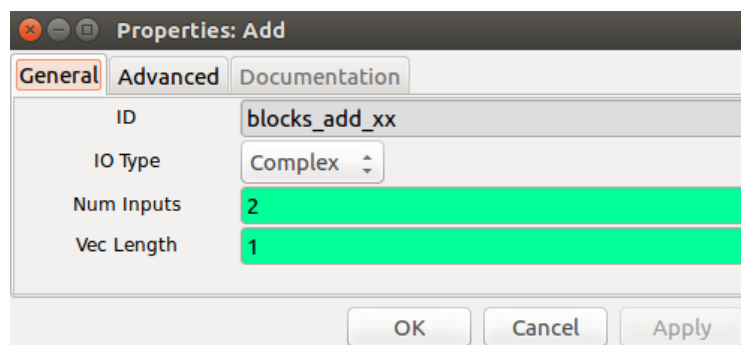


Рисунок 4.10 - Настройки для сумматора

Настройка генератора Гауссовского белого шума, осуществляется в соответствии с выбранным типом модуляции.

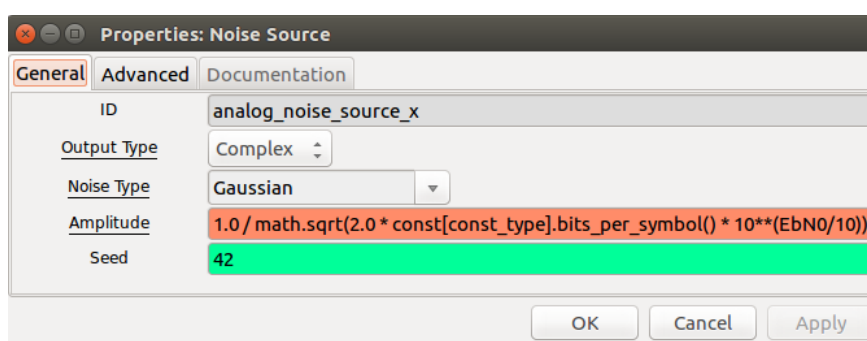


Рисунок 4.11 - Настройки для блока добавления Гауссовского белого шума

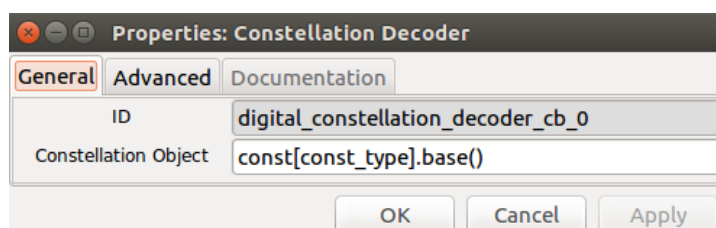


Рисунок 4.12 - Настройки для демодулятора QAM/PSK

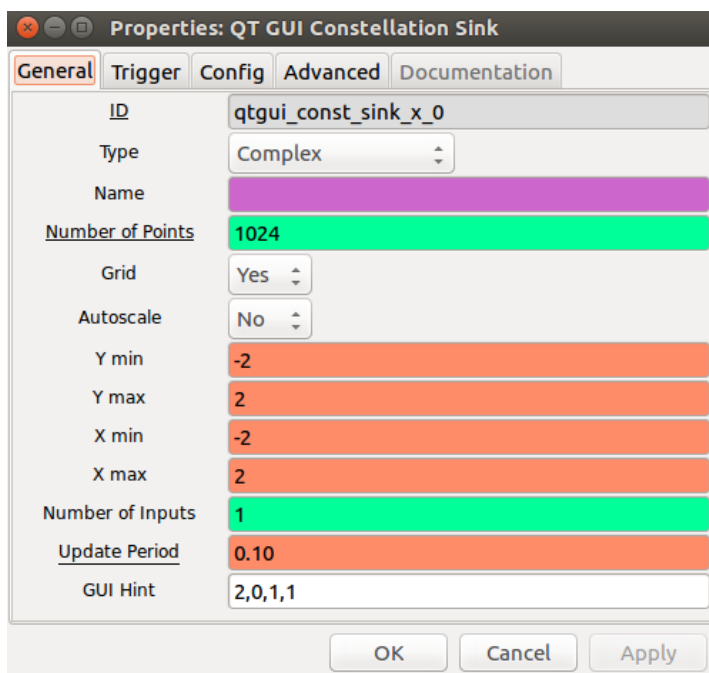


Рисунок 4.13 - Основные настройки для блока просмотра диаграммы созвездий (Часть 1)

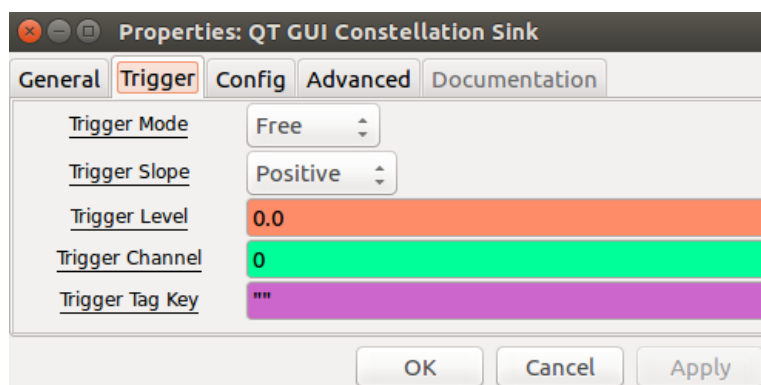


Рисунок 4.14 - Настройки синхронизации для блока просмотра диаграммы созвездий (Часть 2)

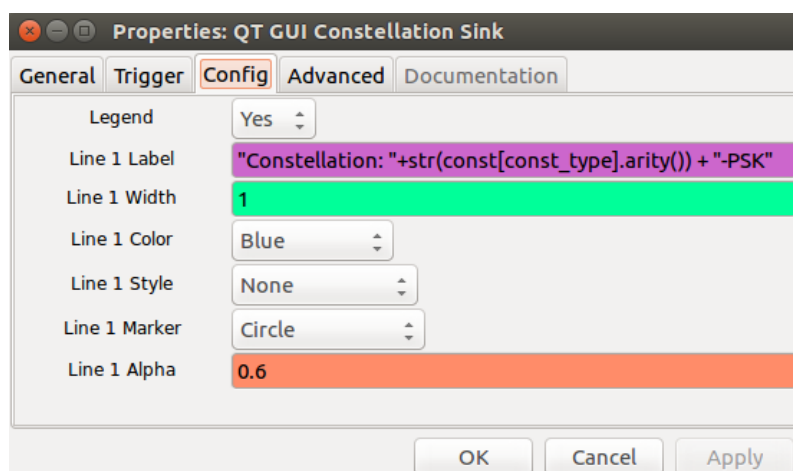


Рисунок 4.15 - Настройки геометрии для блока просмотра диаграммы созвездий (Часть 3)

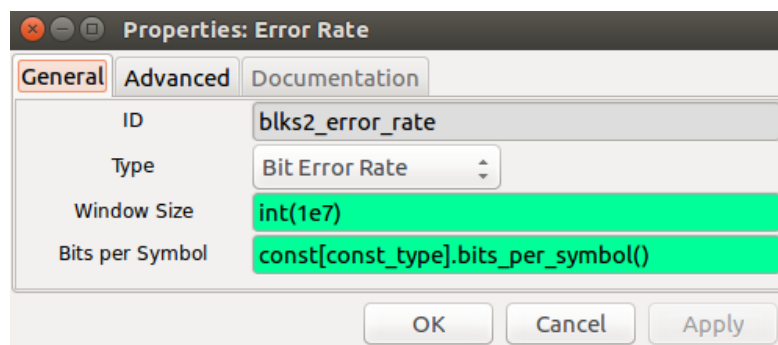


Рисунок 4.16 - Настройки для блока измерения вероятности битовой ошибки

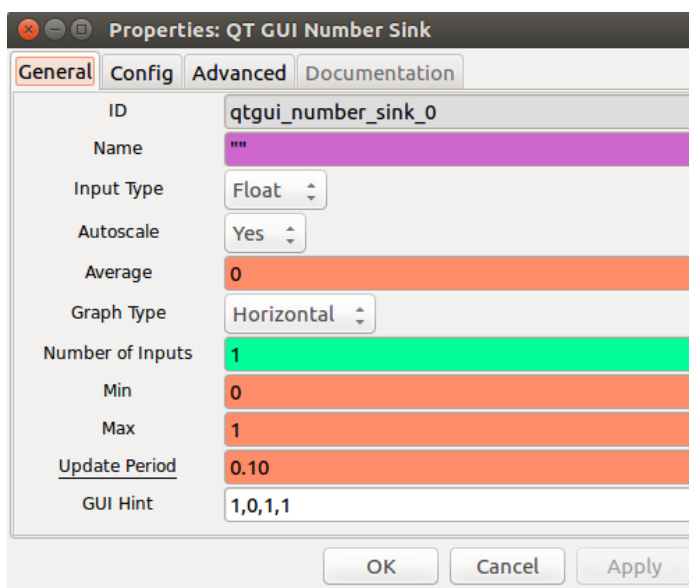


Рисунок 4.17 - Настройки для блока записи ошибок

Таблица 4.1 - Теоретические данные пороговых значений ОСШ для различных типов QAM/PSK

Тип цифровой модуляции	Пороговое значение отношения с/ш при АБГШ
BPSK	8 дБ
QAM-4/QPSK	10 дБ
8-PSK	14 дБ

После составления программы должны получиться графики звездных диаграмм, представленные на рисунках ниже.

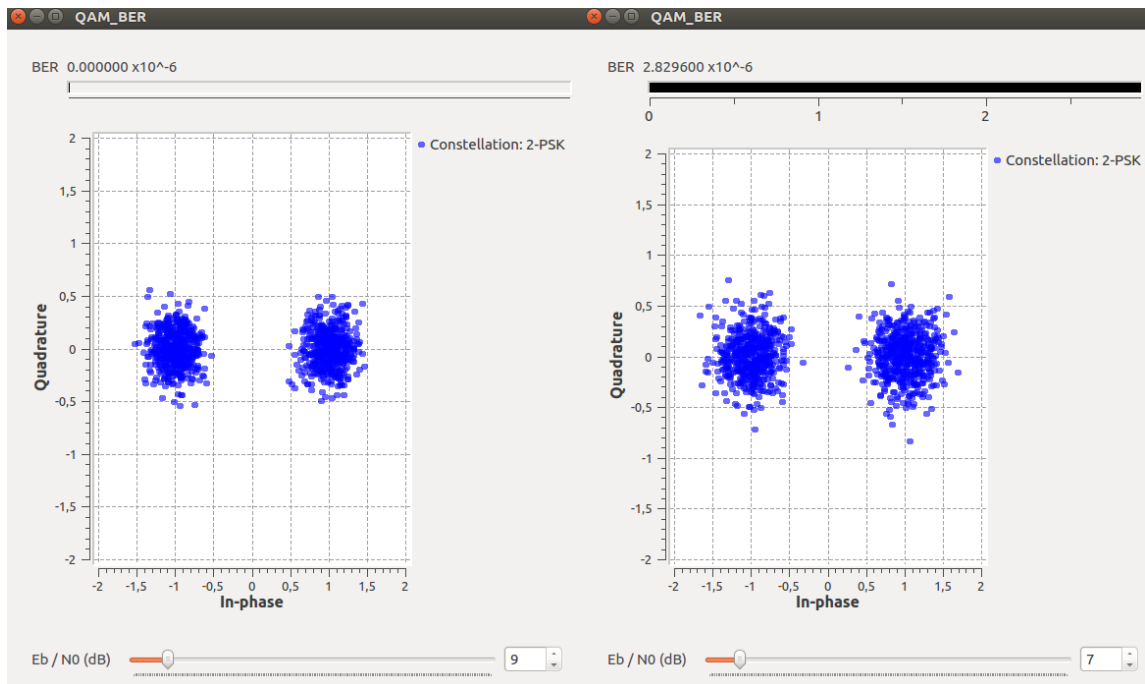


Рисунок 4.18 - Диаграммы созвездий для BPSK с разными параметрами отношения сигнал/шум

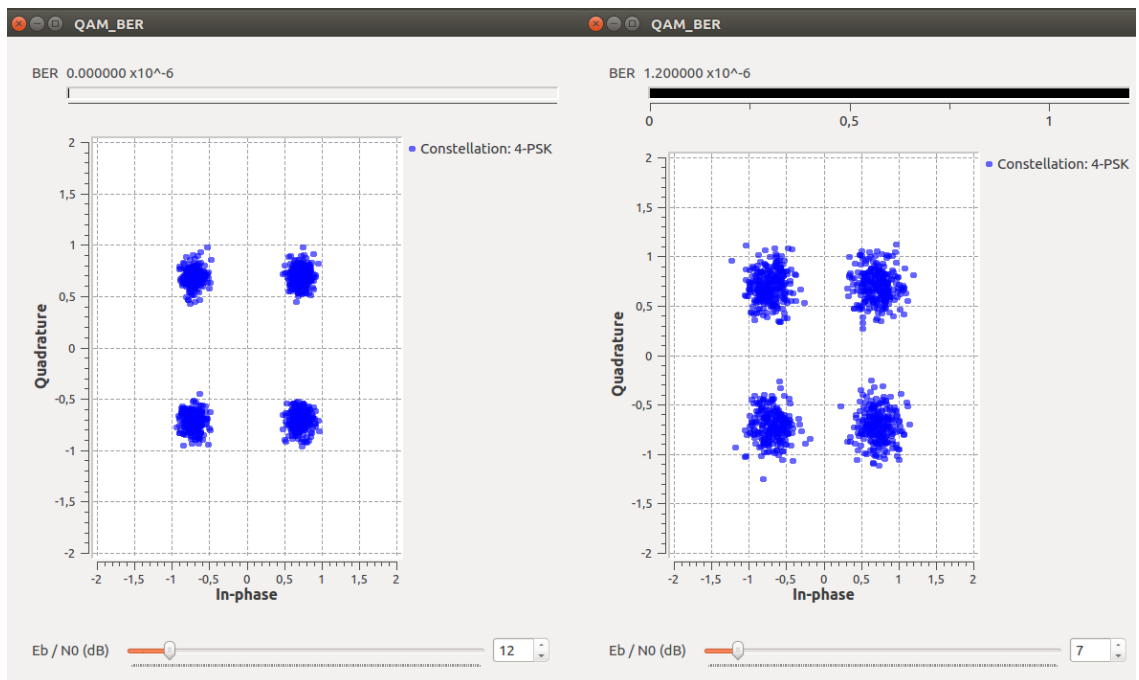


Рисунок 4.19 - Диаграммы созвездий для QAM-4 с разными параметрами отношения сигнал/шум

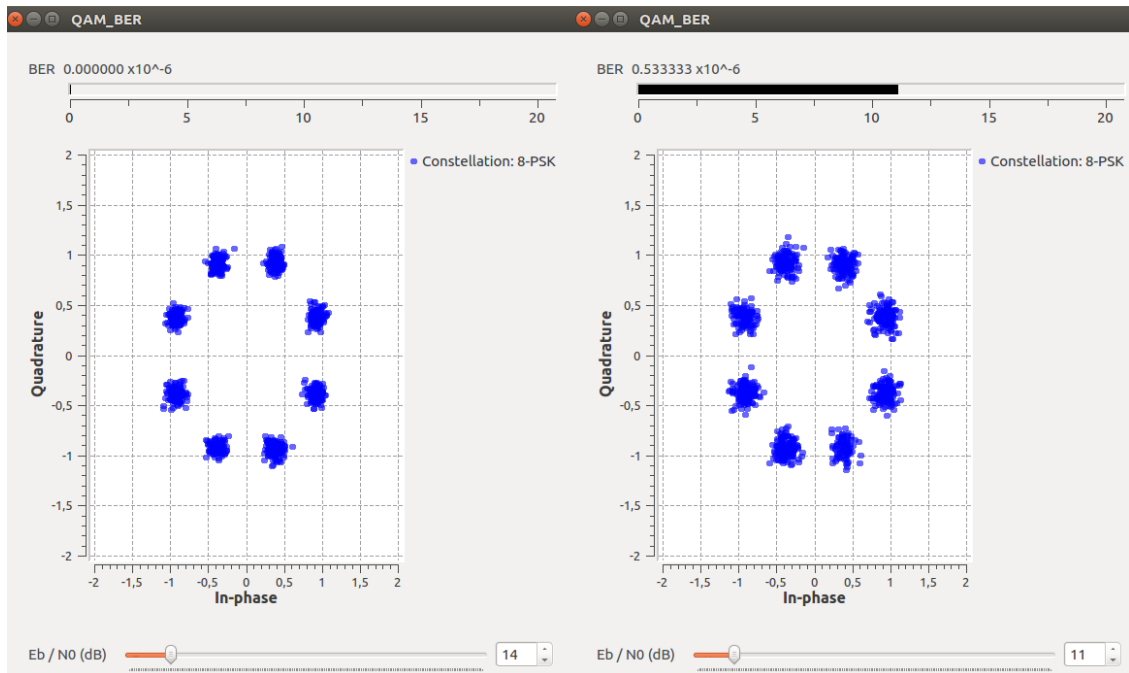


Рисунок 4.20 - Диаграммы созвездий для 8-PSK с разными параметрами отношения сигнал/шум

Задания для самостоятельного выполнения

- 1) Произвести изменение типов модуляции, и зафиксировать те значения отношения сигнал/шум, при которых вероятность битовой ошибки будет равна нулю.
- 2) Уменьшить количество символов в пакете (Num Samples в блоке Random Source) и зафиксировать изменение вероятности битовой ошибки от отношения сигнал/шум.

Контрольные вопросы к практической работе

- 1) Какие типы цифровой модуляции были представлены в данной работе? В чем их отличие.
- 2) Почему при прохождении QAM-модулированного сигнала через реальный канал, происходит произвольный поворот созвездия?
- 3) Что такое вероятность битовой ошибки?
- 4) Что такое отношение сигнал/шум?
- 5) Почему при повышении порядка QAM, необходимо обеспечить более высокий показатель отношения сигнал/шум?

5. Практическая работа «Технология OFDM»

Цель работы: Ознакомиться с технологией OFDM, сформировать канал передачи аудиоданных в программной среде GNURadio.

Задачи практической работы:

- 1) Составить программу с применением технологии OFDM в программной среде GNU Radio.
- 2) Произвести передачу и прием аудиоданных по реальному каналу. Зафиксировать и проанализировать изменение сигнала при прохождении через реальный канал.

Краткие теоретические сведения

Технология OFDM разбивает входные данные (N_s) на параллельные потоки, и каждый символ модулируется с использованием отдельной несущей частоты. Несмотря на наличие спектральных перекрытий между поднесущими, они не мешают друг другу. Другими словами, сохраняется спектральная ортогональность между поднесущими. Рисунок 5.1 иллюстрирует спектр OFDM в частотной области, где T_p - интервал между поднесущими, G - нормированная к единице амплитуда сигнала, F_1-F_7 - поднесущие, f - частота.

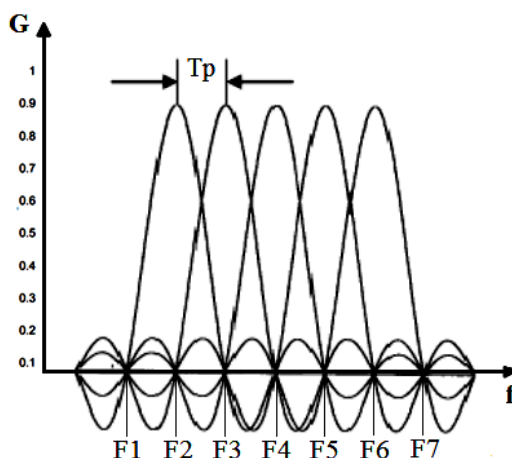


Рисунок 5.1 - Спектр OFDM

Следует сказать, что для большого количества поднесущих OFDM требуется очень точная синхронизация частоты между приемником и передатчиком. Это связано с тем, что с отклонениями по частоте (смещениями) поднесущие теряют свою ортогональность, и это вызывает интерференционные помехи. Частотные смещения обычно вызваны несовершенством аппаратуры или доплеровского сдвига частоты из-за быстрого перемещения передатчика или приемника. Рисунок 5.2 изображает структурную приемопередатчика для системы OFDM. В передатчике информационные биты модулируются в P символов квадратурным модулятором. Символы распределяются на N_s ортогональных поднесущих и преобразуются во временную область с помощью обратного быстрого преобразования Фурье (ОБПФ). Затем к данным добавляются защитные интервалы по бокам спектра и происходит добавление циклического префикса.

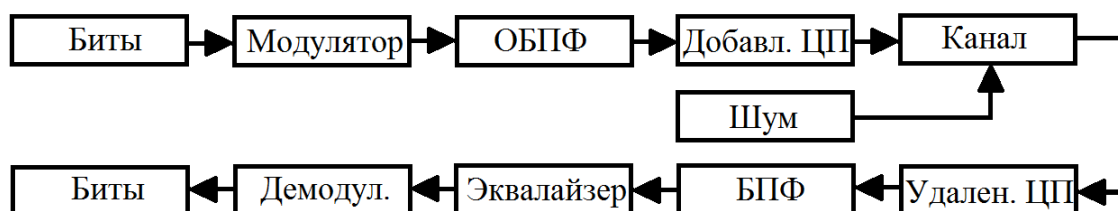


Рисунок 5.2 - Структурная схема формирования OFDM

Циклический префикс вводится в передаваемый сигнал для борьбы с межсимвольной интерференцией, которая возникает между символами OFDM из больших разбросов с задержкой при многолучевом распространении. CP является циклически расширенным защитным интервалом, где каждому символу OFDM добавляется периодическое расширение. Общая длительность символа $T_{total} = T_d + T_g$, где T_g - это защитный интервал, а T_d длительность самого символа. Когда защитный интервал длиннее задержки многолучевого распространения, межсимвольной интерференции можно избежать. CP преобразует линейную свертку передаваемого сигнала с импульсной характеристикой канала в циклическую свертку. Это означает, что эффект многолучевого замирания в переданных символах упрощаются до элементарного умножения между созвездиями переданных данных (d) с частотной характеристикой канала (H). Частотная характеристика канала определяется с помощью преобразования Фурье (FT) от импульсной характеристики канала (h).

После приема сигнала, циклический префикс удаляется и выполняется быстрое преобразование Фурье БПФ размера N_s . Модель принятого сигнала после БПФ на i -й поднесущей может быть охарактеризована как:

$$r_i = H_i d_i + n_i. \quad (4.1)$$

В формуле 4.1 H_i и n_i обозначают характеристики канала и сигнал АБГШ на i -й поднесущей. Частотный отклик подканала на i -й поднесущей ($i = 1, 2, \dots, N_s$) вычисляется по формуле:

$$H_i = \sum_{l=0}^{L-1} h(l) \exp(-j2\pi i \tau_l / N_s). \quad (4.2)$$

В формуле 4.2, τ - временная задержка, которую должен нивелировать циклический префикс. Значение τ , может быть любым положительным действительным числом. Это можно использовать для представления и расчетов реалистичных сред каналов.

Ход выполнения работы

В программной среде GNU Radio составить программу, показанную на рисунке 5.3 и передать аудиоданные по реальному каналу с использованием технологии OFDM. Далее приводится описание работы программы.

При помощи загрузчика аудиофайлов формируются аудиоданные, которые поступают на блок Rational resampler для децимации с шагом 6. После децимации аудиоданные поступают на блок Float to short для изменения типа данных. Далее происходит кодирование специальных телефонным аудиокодеком, с которого выходят данные в виде векторных значений, и для перевода их в строки применяется блок Vector to stream. После этого кодированные данные упаковываются с помощью блока Unpacked to packed и подвергаются модуляции при помощи технологии OFDM блоком OFDM mod. Добавляется константа для правильной работы системы, а так же подключается виртуальный спектроанализатор для отслеживания сигнала. После формирования OFDM символа, данные поступают в передатчик B210, и транслируются в реальный канал распространения сигналов. Приемная часть B210 осуществляет обработку сигнала и отправляет поток бит на OFDM demod. Данный блок производит демодуляцию принятого OFDM символа и отправляет демодулированные данные на блок Packed to unpacked. После этого происходят обратные процедуры конвертации строк в

векторные значения и декодирования телефонным аудиокодеком. Для правильного восстановления аудиоданных происходит процедура интерполяции с шагом 6 и данные воспроизводятся на динамических головках ПК после обработки блоком Audio sink.

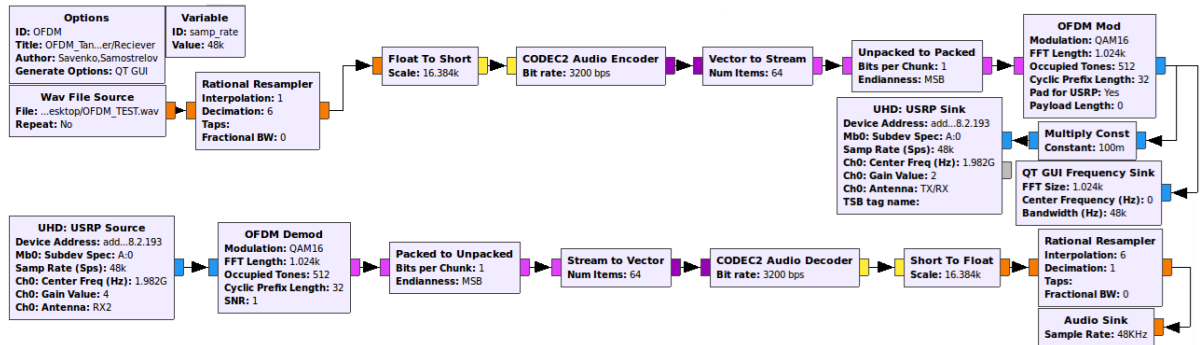


Рисунок 5.3 - Общая схема приемника и передатчика с применением технологии OFDM в GNU Radio

Параметры каждого блока указаны на рисунках 5.4 - 5.24. В начале работы производится настройка частоты дискретизации. Для этого необходимо создать переменную как показано на рисунке ниже.

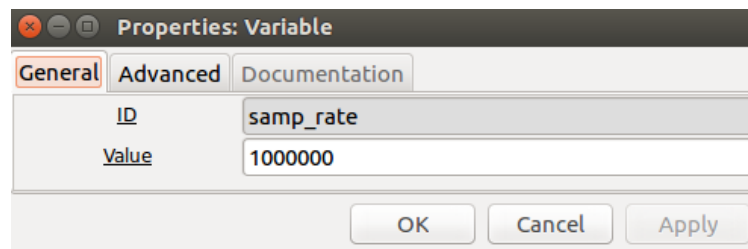


Рисунок 5.4 - Настройка частоты дискретизации системы

Далее создаем генератор загрузчик аудиофайлов, как показано на рисунке ниже. Путь к файлу необходимо указать с учетом его местонахождения на вашем ПК.

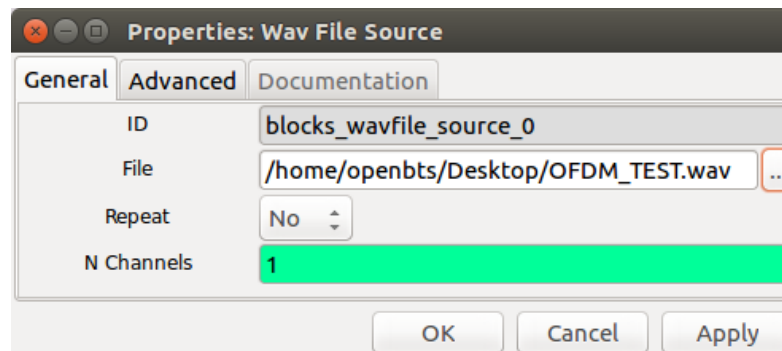


Рисунок 5.5 - Настройки для загрузчика аудиофайлов

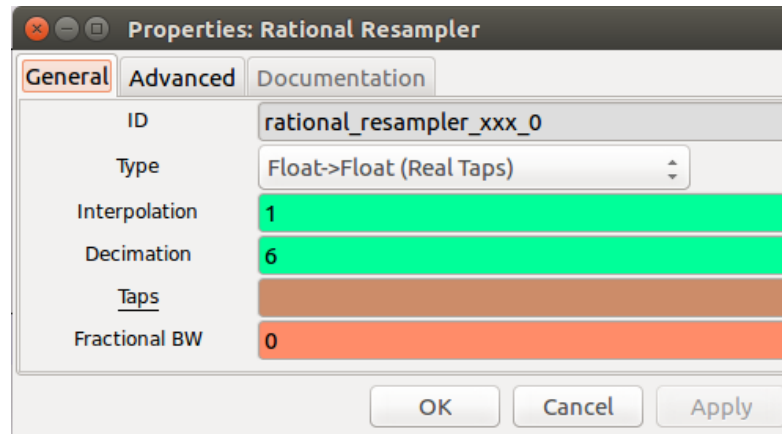


Рисунок 5.6 - Настройки блока Rational resampler

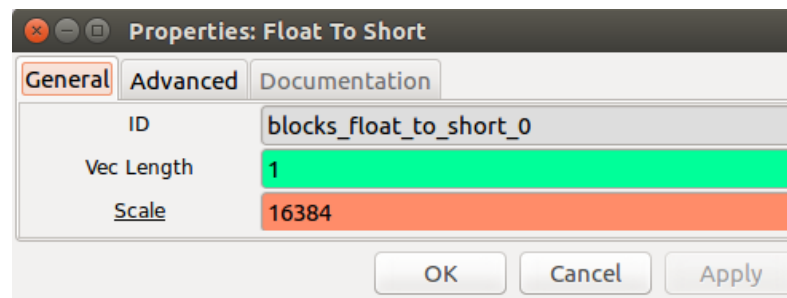


Рисунок 5.7 - Настройки блока Float to short

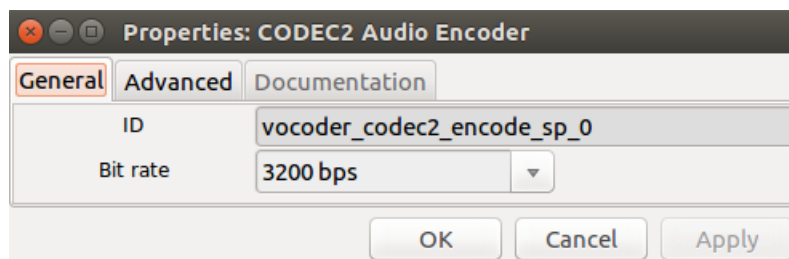


Рисунок 5.8 - Настройки блока аудиокодека

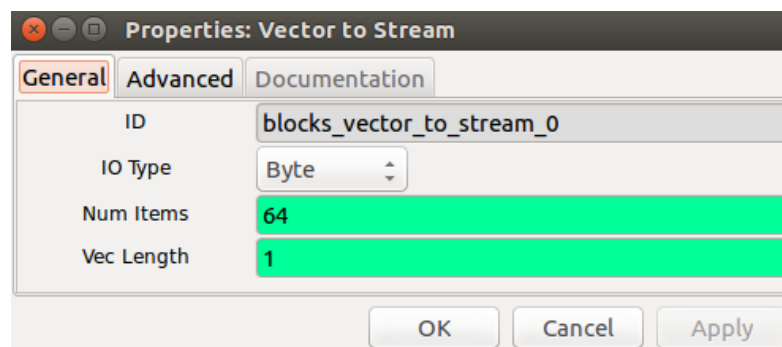


Рисунок 5.9 - Настройки блока Vector to stream

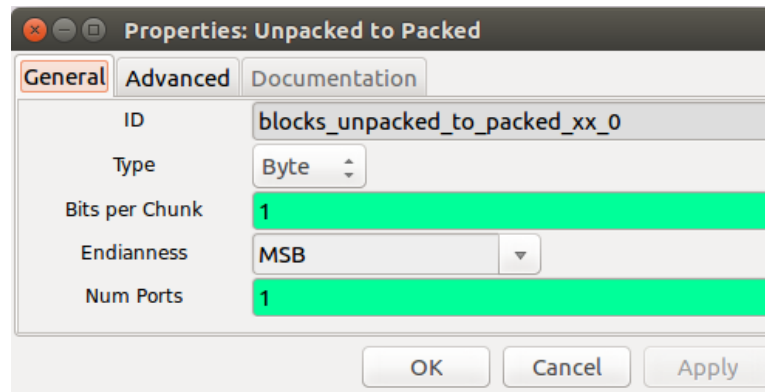


Рисунок 5.10 - Настройки блока Unpacked to packed

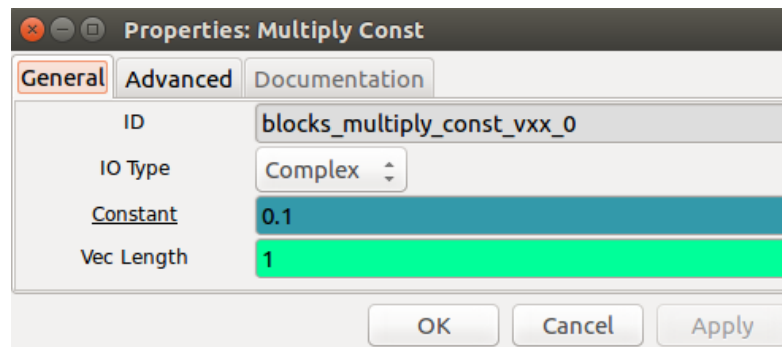


Рисунок 5.11 - Настройки блока Multiply Const

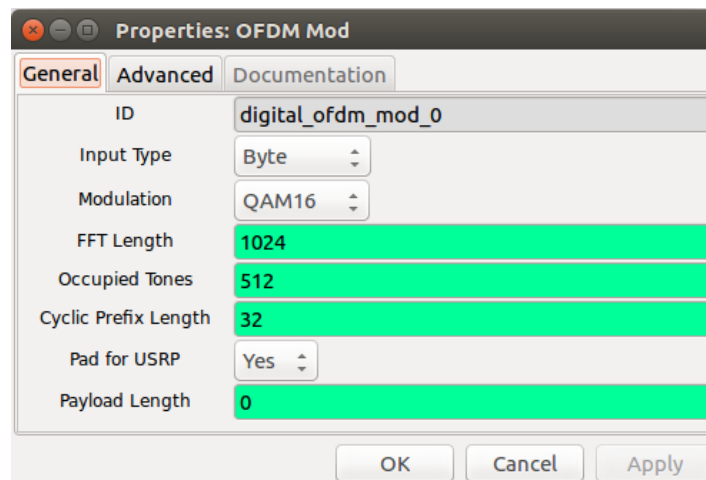


Рисунок 5.12 - Настройки блока OFDM mod

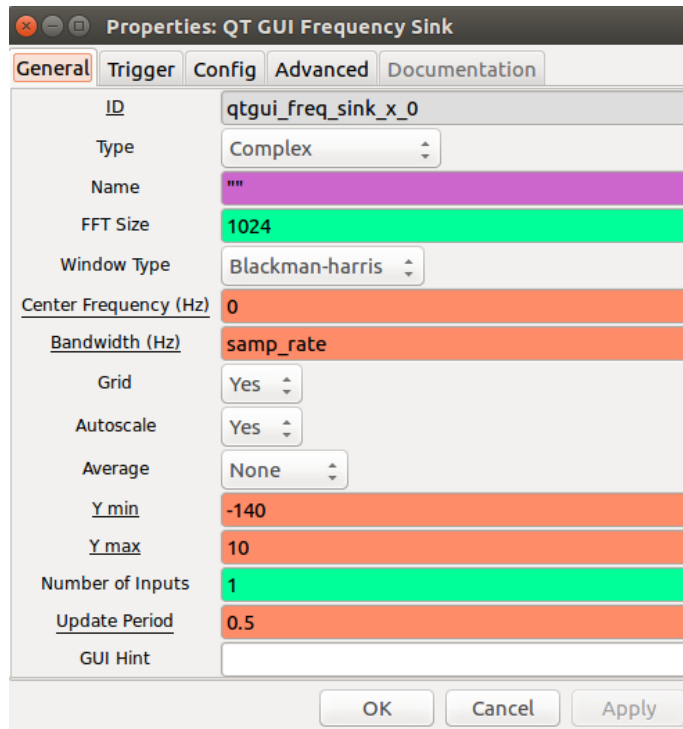


Рисунок 5.13 - Настройки блока спектроанализатора (Часть 1)

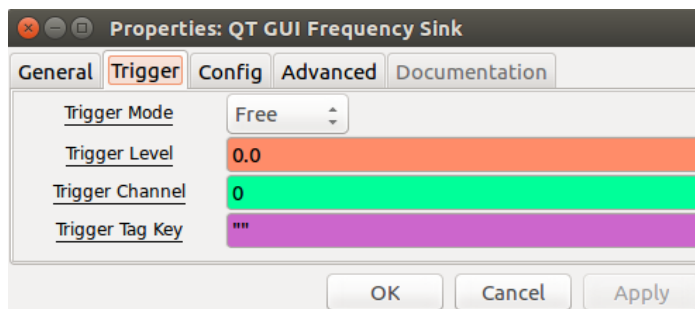


Рисунок 5.14 - Настройки блока спектроанализатора (Часть 2)

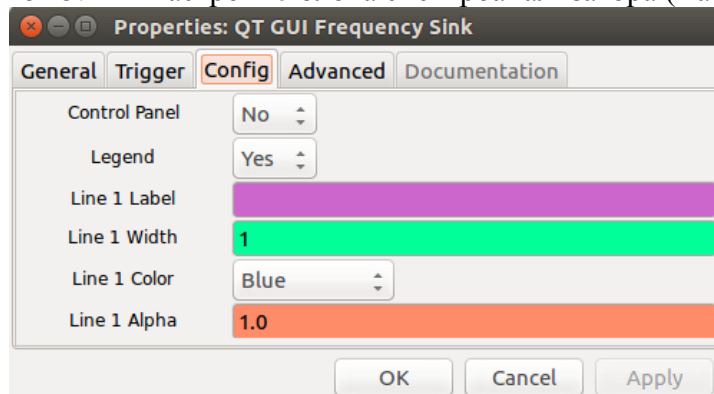


Рисунок 5.15 - Настройки блока спектроанализатора (Часть 3)

Важным пунктом является точная настройка передатчика, характеристики которого приведены далее на рисунках. В пункте Device Arguments необходимо вписать "b210".

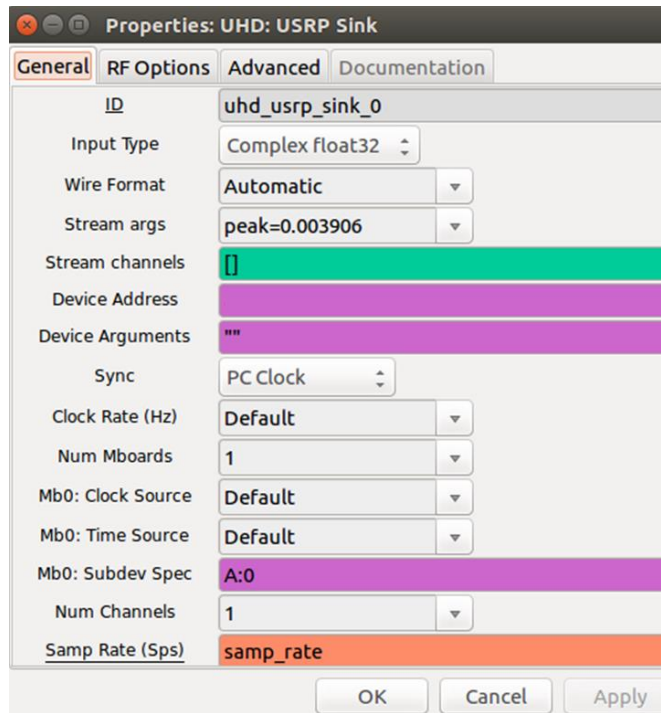


Рисунок 5.16 - Настройки общих параметров для передающей части NI-USRP(Часть 1)

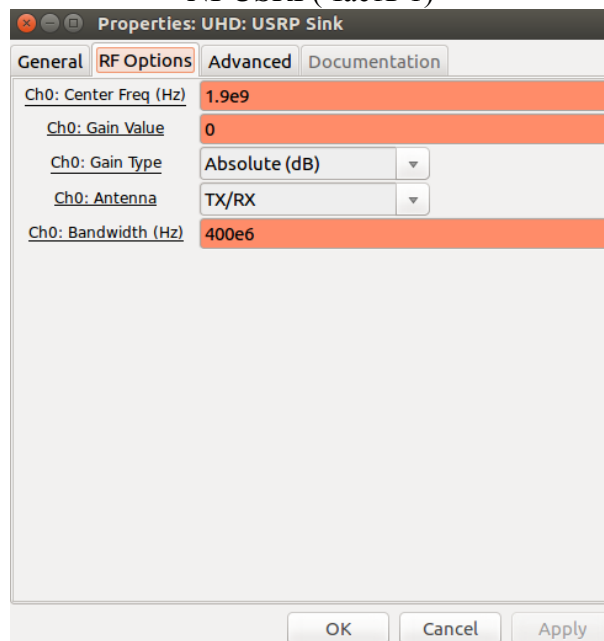


Рисунок 5.17 - Настройки общих параметров для передающей части NI-USRP(Часть 2)

Для правильного приема сигнала из эфира, приемник необходимо настроить в точности с теми же характеристиками, как и передатчик. В пункте Device Arguments необходимо вписать "b210".

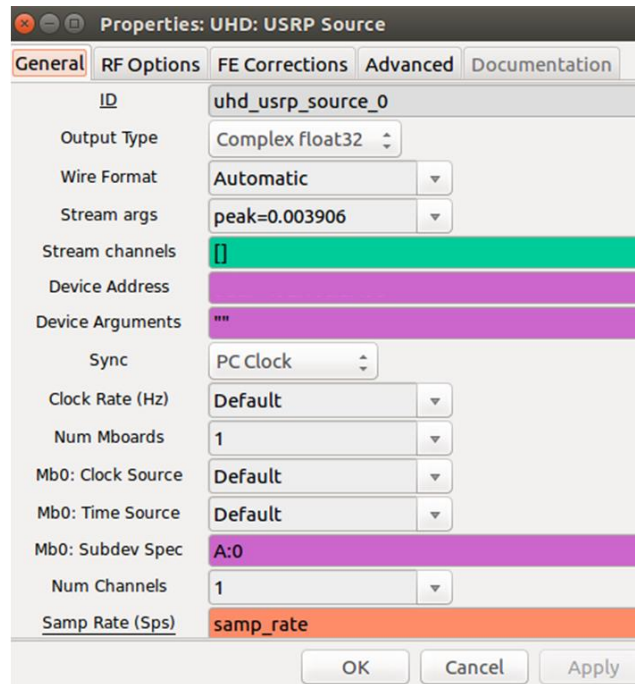


Рисунок 5.18 - Настройки общих параметров для приёмной части NI-USRP (Часть 1)

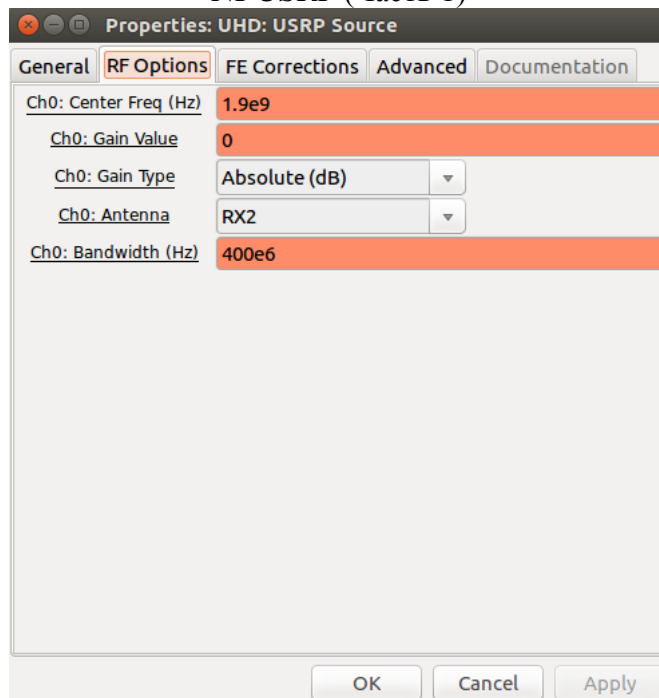


Рисунок 5.19 - Настройка общих параметров для приёмной части NI-USRP (Часть 2)

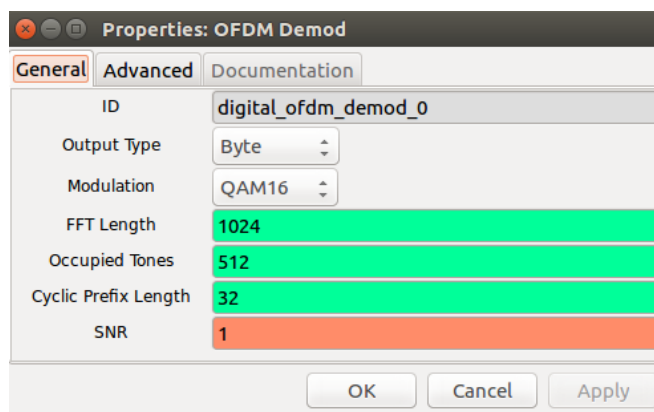


Рисунок 5.20 - Настройки блока OFDM demod

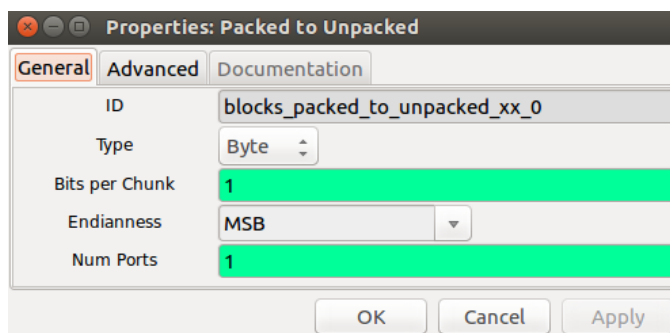


Рисунок 5.21 - Настройки блока Packed to unpacked

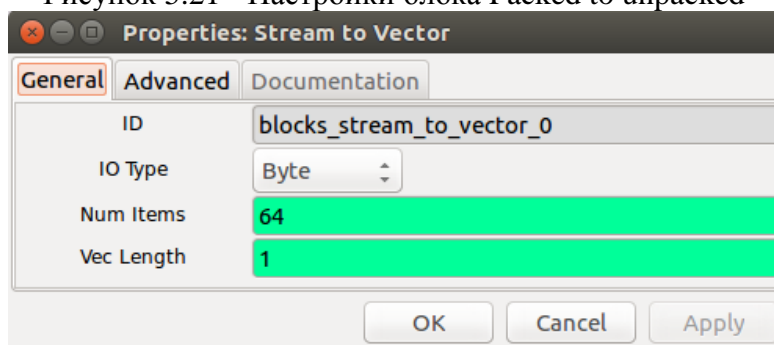


Рисунок 5.22 - Настройки блока Stream to vector

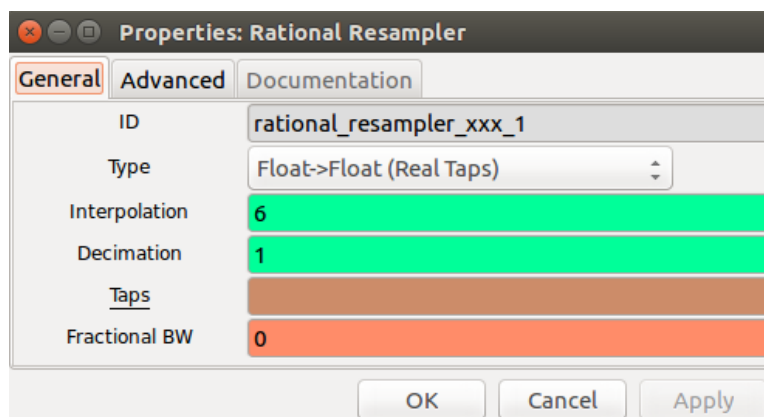


Рисунок 5.23 - Настройки блока Rational resampler для интерполяции

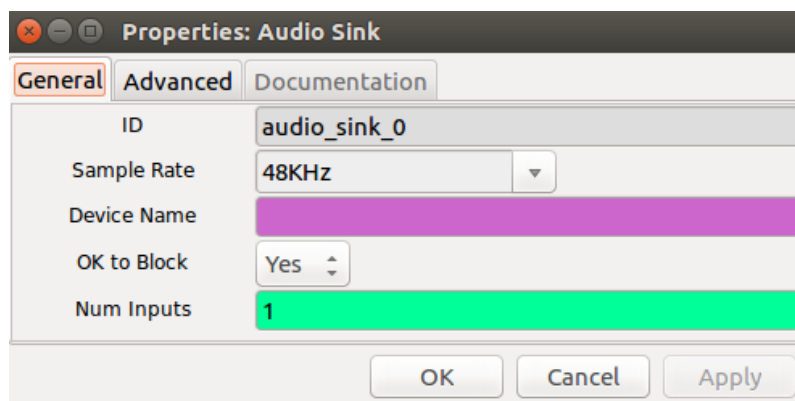


Рисунок 5.24 - Настройки блока Audio sink

После правильного составления программы в GNU Radio, необходимо произвести запуск. Ниже на рисунках приведены графики, полученные после блока формирования OFDM. После правильно составленной программы, в динамических головках звукового оборудования компьютера будет воспроизводиться тестовый звуковой файл.

На выходе виртуального спектроанализатора должен быть показан сформированный OFDM символ, до переноса на несущую частоту.

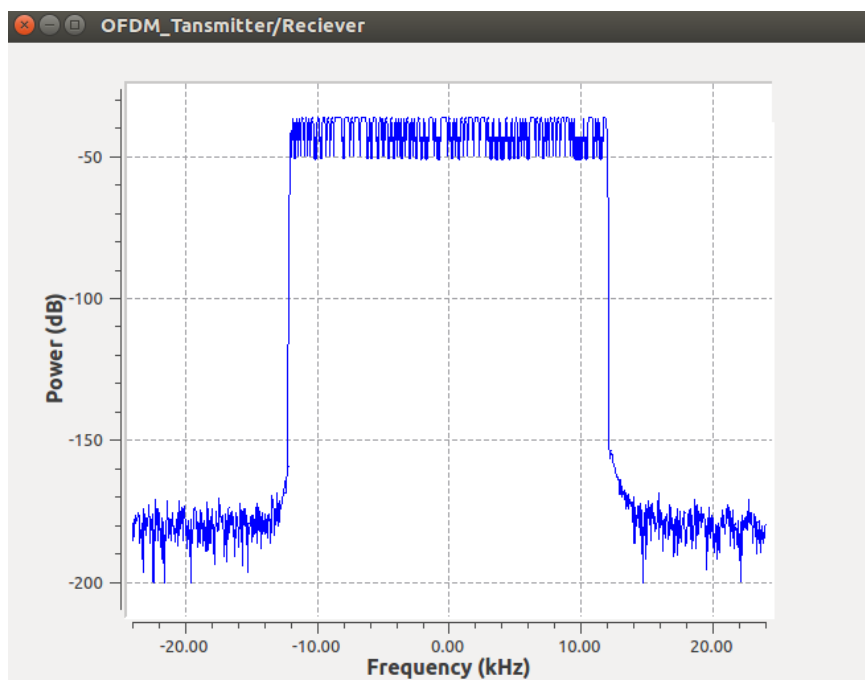


Рисунок 5.25- Сформированный OFDM символ до переноса на несущую частоту

На поднесущих данного OFDM символа производится передача аудиоданных по реальному каналу связи. Для более точного понимания процесса необходимо выполнить задания, которые приведены далее.

Задания для самостоятельного выполнения

- 1) Произвести подключение спектроанализатора к выходу передатчика. Обнаружить сформированный OFDM символ и зафиксировать его параметры.
- 2) Произвести изменения типов модуляции, несущей частоты, размера циклического префикса и других параметров. Произвести анализ полученных данных OFDM символа, после снятия качественных характеристик с помощью цифрового спектроанализатора.

Контрольные вопросы к практической работе

- 1) Что такое OFDM-модуляция?
- 2) Что такое межсимвольная интерференция?
- 3) Что такое циклический префикс? Почему он имеет разные длительности?
- 4) Что такое децимация и интерполяция?
- 5) Каким выражением определяется частотный отклик подканала на i -й поднесущей?