

Санкт-Петербургский политехнический университет Петра Великого  
Институт компьютерных наук и кибербезопасности  
Высшая школа компьютерных технологий и информационных систем

## **Отчёт по лабораторной работе №12**

Дисциплина: Телекоммуникационные технологии.

Выполнил студент гр. 5130901/10101

\_\_\_\_\_  
(подпись)

М. Т. Непомнящий

Руководитель

\_\_\_\_\_  
(подпись)

Н.В. Богач

22 мая 2024 г.

Санкт-Петербург  
2024

# Содержание

1	Задача . . . . .	2
2	Ход работы . . . . .	2
	2.1 Подготовка исходной схемы. . . . .	2
	2.1.1. Схема из 1 tutorials . . . . .	2
	2.1.2. Low Pass Filter . . . . .	3
	2.2 Designing the Filter Taps . . . . .	4
	2.2.1. Frequency Xlating FIR Filter . . . . .	4
	2.2.2. Low-Pass Filter Taps . . . . .	6
	2.2.3. Анализ схемы . . . . .	7
	2.2.4. Запуск блок-схемы . . . . .	8
	2.2.5. Анализ графика . . . . .	9
	2.3 Entering Filter Taps Manually . . . . .	10
	2.3.1. Изменение схемы . . . . .	10
	2.3.2. Анализ схемы . . . . .	10
	2.3.3. Запуск блок-схемы . . . . .	11
	2.3.4. Анализ графика . . . . .	12
	2.4 Real to Complex Filter . . . . .	12
	2.4.1. Изменение схемы . . . . .	13
	2.4.2. Анализ схемы . . . . .	13
	2.4.3. Запуск блок-схемы . . . . .	15
	2.4.4. Анализ графика . . . . .	15
3	Вывод . . . . .	16

# 1. Задача

Исследовать применение FIR фильтра сдвига частоты в среде GNU Radio для понимания его функционала и эффективности в обработке сигналов. Основной задачей является изучение способов создания и настройки фильтровых коэффициентов (filter taps) для подавления высокочастотных помех и анализ влияния этих настроек на исходный сигнал. Дополнительной целью является оценка производительности фильтра при различных условиях работы, таких как изменение параметров фильтра и его частотного диапазона. Полученные результаты помогут глубже понять принципы работы FIR фильтров и их применимость в практических задачах обработки сигналов.

## 2. Ход работы

### 2.1. Подготовка исходной схемы

#### 2.1.1. Схема из 1 tutorials

За основу для выполнения данного задания была взята схема, построенная в предыдущем задании (Low Pass Filter Example), где создавался простой фильтр низких частот в GNU Radio с использованием блоков "Variable" "QT GUI Range" "Signal Source" и "Low Pass Filter" "Throttle" и "QT GUI Frequency Sink". В конце задания создаётся графическая схема, которая генерирует сигнал, пропускает его через фильтр нижних частот и выводит результат на экран. Таким образом, была создана схема, представленная на Рис. 2.1 ниже:

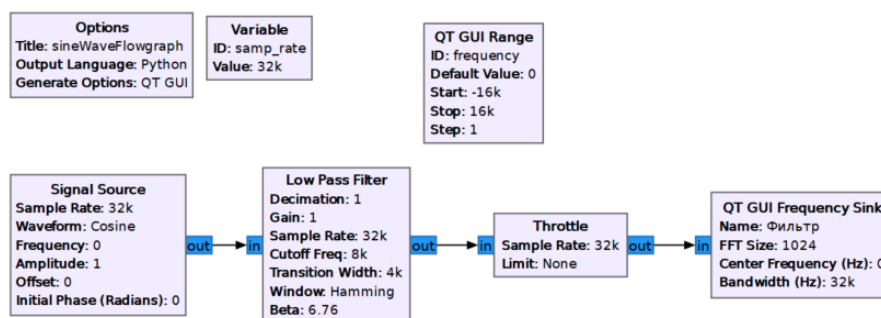


Рис. 2.1 - Схема из первого задания.

Краткое описание блоков, используемых в схеме:

- **Variable** - используется для хранения переменной *samp\_rate*, от которой будут отсчитываться остальные частоты в проекте.
- **QT GUI Range** - используется для изменения значения частоты сигнала в процессе моделирования.
- **Signal Source** - генератор косинусоидального сигнала с частотой, подаваемой с QT GUI Range и амплитудой 1.
- **Low Pass Filter** - исследуемый фильтр низких частот, он имеет несколько параметров, которые будут рассмотрены позднее.
- **Throttle** - предназначенный для ограничения скорости передачи сэмплов в цифровой обработке сигналов. Проще говоря ограничитель скорости передачи данных.
- **QT GUI Frequency Sink** - это графический приемник, основанный на библиотеке QT, предназначенный для отображения нескольких сигналов в частотной области.

## 2.1.2. Low Pass Filter

Одним из ключевых элементов схемы является **Low Pass Filter**, рассмотрим его отдельно:

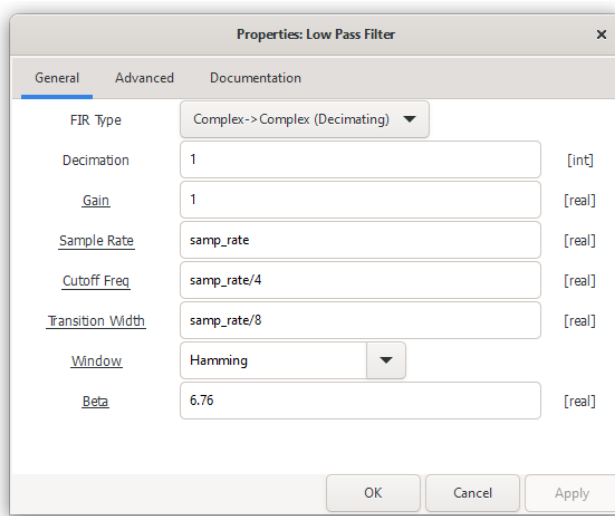


Рис. 2.2 - Настройки Low Pass Filter.

Разберём за что отвечает каждый из параметров:

- **FIR Type** - определяет тип FIR фильтра, то есть указывает, работает ли фильтр с вещественными или комплексными входными/выходными данными.
- **Decimation** - определяет коэффициент децимации фильтра, который указывает, на сколько раз уменьшается частота дискретизации сигнала после применения фильтра.
- **Gain** - масштабирующий коэффициент, применяемый к выходу фильтра, для регулировки амплитуды сигнала.
- **Sample Rate** - частота дискретизации входного сигнала, указывает, с какой частотой входные сигналы отсчитываются во времени.
- **Cutoff Freq** - частота среза фильтра, то есть частота, на которой фильтр начинает подавлять высокочастотные составляющие входного сигнала.
- **Transition Width** - ширина переходной зоны между полосой подавления и полосой пропускания в фильтре. Этот параметр указывает, насколько широкой должна быть зона плавного перехода между частотами, на которых фильтр полностью подавляет или пропускает сигнал.
- **Window** - тип окна, используемого при генерации коэффициентов фильтра. Различные типы окон имеют разные свойства и влияют на характеристики фильтра, такие как разрешение в частотной области и уровень подавления сигнала.
- **Beta** - параметр, который применяется только к окну Кайзера. Этот параметр контролирует форму окна Кайзера и влияет на его способность сглаживания переходной зоны и подавления побочных лепестков.

Таким образом, мы создали схему (Рис. 2.1) из первого tutorials и готовы перейти к основной части проекта (Designing Filter Taps).

## 2.2. Designing the Filter Taps

### 2.2.1. Frequency Xlating FIR Filter

Немного изменим схему, заменив фильтр нижних частот (**Low Pass Filter**) на цифровой FIR-фильтр с возможностью сдвига частоты (**Frequency Xlating FIR Filter**). Его настройки представлены на Рис. 2.3 ниже:

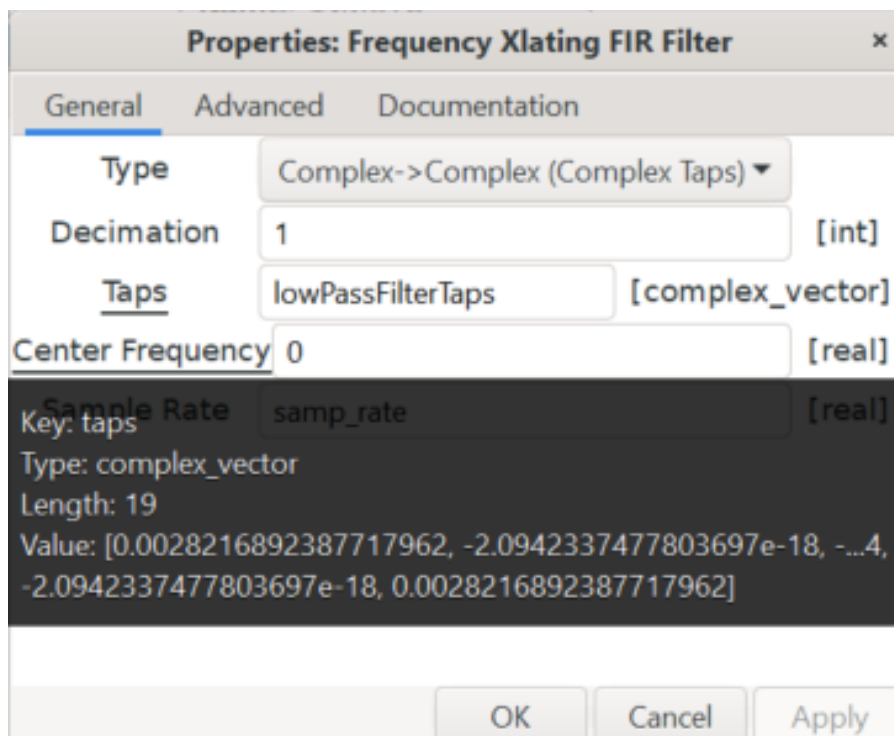


Рис. 2.3 - Настройки Frequency Xlating FIR Filter.

Разберём за что отвечает каждый из его параметров:

- **Type (Тип фильтра)**

- *Описание:* Указывает тип фильтра, который будет применен к сигналу.
- *Примеры:* Низкочастотный фильтр (Lowpass Filter), Высокочастотный фильтр (Highpass Filter), Полосовой фильтр (Bandpass Filter), Режекторный фильтр (Bandstop Filter), Произвольный тип фильтра, определенный пользователем.

- **Decimation (Децимация)**

- *Описание:* Указывает коэффициент децимации, который определяет, насколько уменьшится частота дискретизации после фильтрации. Децимация снижает частоту дискретизации сигнала, оставляя только каждую n-ю выборку.
- *Пример:* Если указано значение 4, выходная частота дискретизации будет в 4 раза меньше входной.

- **Taps (Коэффициенты фильтра)**

- *Описание:* Это список коэффициентов FIR-фильтра. Эти коэффициенты определяют характеристики фильтра, такие как полоса пропускания и крутизна спада. Коэффициенты могут быть рассчитаны с использованием различных методов проектирования фильтров, таких как окно Хэмминга, окно Кайзера и другие.

– *Пример:* [0.1, 0.15, 0.5, 0.15, 0.1] - пример простого набора коэффициентов.

- **Center Frequency (Центральная частота)**

– *Описание:* Определяет частоту, на которую необходимо сместить входной сигнал. Это значение указывает, на сколько герц сигнал будет сдвинут вверх или вниз по частоте.

– *Пример:* Если указать значение 100 kHz, входной сигнал будет смещен на 100 kHz вверх по частоте.

- **Sample Rate (Частота дискретизации)**

– *Описание:* Указывает частоту дискретизации входного сигнала. Этот параметр необходим для правильной настройки частоты и фильтрации.

– *Пример:* Если входная частота дискретизации равна 1 MHz, это значение нужно указать в соответствующем параметре.

Данный фильтр используется для изменения частоты сигнала путем умножения на комплексный синус или косинус. Этот процесс позволяет сдвигать спектр сигнала в частотной области, что полезно для различных задач обработки сигналов, таких как смещение частоты сигнала для согласования с другими устройствами или для избегания помех. Суть элемента Frequency Xlating FIR Filter заключается в его способности выполнять эффективное цифровое смещение частоты в потоке данных сигнала.

**\*Пояснение**

Умножение сигнала на комплексный синус или косинус используемая для сдвига частоты сигнала в цифровой обработке сигналов. Комплексный синус или косинус представляется в виде  $e^{j\omega t} = \cos(\omega t) + j \sin(\omega t)$ , где  $\omega$  — угловая частота,  $t$  — время, а  $j$  — мнимая единица.

**Как это работает**

1. **Исходный сигнал:** Пусть  $s(t)$  — это наш исходный сигнал с некоторой спектральной составляющей.
2. **Комплексный синус или косинус:** Для сдвига частоты мы умножаем  $s(t)$  на  $e^{j\omega t}$ .

$$s(t) \cdot e^{j\omega t}$$

3. **Результат умножения:** В результате умножения сигнала на  $e^{j\omega t}$  происходит сдвиг всех частотных составляющих сигнала на  $\omega$  в частотной области. Это можно объяснить с помощью свойства преобразования Фурье:

$$\mathcal{F}\{s(t) \cdot e^{j\omega t}\} = S(f - \frac{\omega}{2\pi})$$

где  $S(f)$  — спектр исходного сигнала, а  $f$  — частота. Таким образом, весь спектр сигнала  $s(t)$  сдвигается на  $\frac{\omega}{2\pi}$ .

### 2.2.2. Low-Pass Filter Taps

Также добавим на схему элемент Low-Pass Filter Taps, выставив следующие настройки:

Properties: Low-pass Filter Taps		
General	Advanced	Documentation
ID	lowPassFilterTaps	
Gain	1.0	[float]
Sample Rate (Hz)	samp_rate	[float]
Cutoff Freq (Hz)	samp_rate/4	[float]
Transition Width (Hz)	samp_rate/8	[float]
Window	Hamming	
Beta	6.76	[float]

OK Cancel Apply

Рис. 2.4 - Исходная схема.

Элемент **Low-Pass Filter Taps** в GNU Radio используется для создания набора коэффициентов для низкочастотного фильтра (FIR-фильтра). Эти коэффициенты затем могут быть использованы в различных блоках для фильтрации сигнала. Вот основные параметры этого блока и их описание:

- **Sample Rate (Частота дискретизации)**

- *Описание:* Указывает частоту дискретизации входного сигнала, на основе которой будет рассчитан фильтр.
- *Пример:* Если входная частота дискретизации равна 1 MHz, это значение нужно указать в соответствующем параметре.

- **Cutoff Frequency (Частота среза)**

- *Описание:* Определяет частоту, выше которой сигнал будет подавляться. Это ключевой параметр, который определяет границу полосы пропускания фильтра.
- *Пример:* Если указано значение 100 kHz, то частоты выше 100 kHz будут подавляться.

- **Transition Width (Ширина переходной области)**

- *Описание:* Определяет ширину полосы частот, в которой фильтр переходит от пропускания к подавлению. Ширина переходной области влияет на крутизну отклика фильтра.
- *Пример:* Если указана ширина 10 kHz, фильтр будет переходить от полного пропускания частот ниже частоты среза к полному подавлению частот выше частоты среза в диапазоне 10 kHz.

- **Window**

- *Описание:* Определяет тип оконной функции, используемой для расчета коэффициентов фильтра. Различные оконные функции имеют разные характеристики и влияют на форму отклика фильтра.

- *Примеры:*

- \* Hamming (Хэмминг)
    - \* Hann (Ханн)
    - \* Blackman (Блэкман)
    - \* Kaiser (Кайзер)

- **Beta**

- *Описание:* Параметр окна Кайзера. Используется только если выбрано окно Кайзера. Этот параметр определяет форму окна Кайзера и контролирует компромисс между шириной главного лепестка и уровнем боковых лепестков.

- *Пример:* Типичное значение для бета может быть 8.6.

Эти параметры позволяют точно настраивать характеристики низкочастотного фильтра для решения задач фильтрации сигнала в реальном времени.

После всех описанных ранее действий мы получим следующую схему:

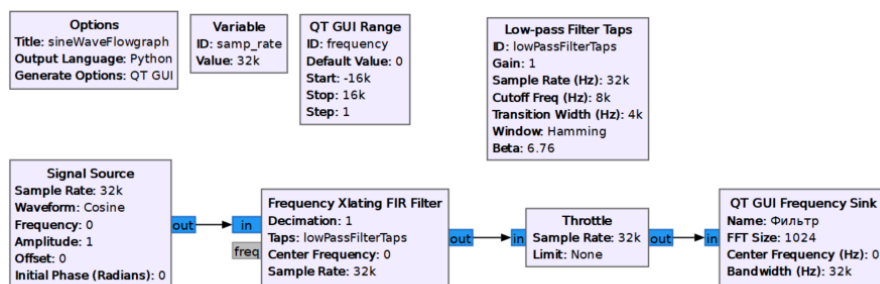


Рис. 2.5 - Схема Designing the Filter Taps.

Таким образом, мы создали схему с частотно-сдвигающим фильтром с возможностью настройки частоты и визуализации спектра сигнала в реальном времени.

### 2.2.3. Анализ схемы

Рассмотрим полученную выше схему (Рис. 2.5) подробнее.

#### Принцип работы

1. **Signal Source (Источник сигнала):** Генерирует косинусоидальный сигнал с частотой 0 Гц (постоянный сигнал) с амплитудой 1 и частотой дискретизации 32 кГц.
2. **QT GUI Range (Диапазон частот для GUI):** Позволяет пользователю изменять частоту сигнала от -16 кГц до 16 кГц в шаге 1 Гц с помощью графического интерфейса.
3. **Low-Pass Filter Taps (Коэффициенты фильтра нижних частот):** Определяет параметры фильтра нижних частот с частотой дискретизации 32 кГц, частотой среза 8 кГц и шириной перехода 4 кГц с использованием окна Хэмминга.



4. **Frequency Xlating FIR Filter (Частотно-смещающий FIR фильтр):** Применяет низкочастотный фильтр к входному сигналу, а также позволяет смещать частоту сигнала. Центр частоты и дискретизации фильтра соответствуют входным параметрам.
5. **Throttle (Регулятор скорости):** Ограничивает скорость потока данных, чтобы соответствовать реальному времени. Здесь нет ограничений по скорости.
6. **QT GUI Frequency Sink (График частоты):** Отображает спектр сигнала на графике. Параметры: размер FFT 1024, центральная частота 0 Гц, полоса пропускания 32 кГц.

### Предположения о поведении сигнала

- **От -16 до -8 кГц и от 8 до 16 кГц:**
  - В этих диапазонах выходной сигнал будет подавлен до почти нулевого уровня, так как фильтр нижних частот срежет компоненты сигнала выше 8 кГц.
- **От -8 до 8 кГц:**
  - В этом диапазоне частот сигнал будет проходить через фильтр. Сигналы ближе к 0 Гц (DC) будут проходить без значительного ослабления.
  - По мере приближения к частотам 8 кГц или -8 кГц, амплитуда сигнала начнет уменьшаться из-за переходной ширины фильтра (4 кГц).

### Пояснения

- **Низкочастотный фильтр:**
  - Пропускает частоты ниже 8 кГц, а частоты выше 8 кГц ослабляет.
  - Переходная ширина 4 кГц означает, что ослабление будет происходить постепенно в диапазоне от 8 кГц до 12 кГц.
- **Frequency Xlating FIR Filter:**
  - Перемещает центральную частоту сигнала на заданное значение, что позволяет нам наблюдать изменение частоты в широком диапазоне.
- **QT GUI Frequency Sink:**
  - Позволяет визуально наблюдать спектр сигнала и подтверждать теоретические расчеты и предположения о работе фильтра и изменения частоты.

#### 2.2.4. Запуск блок-схемы

Проверим наши предположения, выполнив запуск схемы, представленной на Рис. 2.5. Установим настройку *Max Hold* (Максимальное удержание). Когда эта опция включена, график показывает максимальное значение сигнала, которое было достигнуто в течение заданного времени, а затем удерживает его на экране. В результате получим график Частотного спектра выходного сигнала после применения частотно-смещающего фильтра:

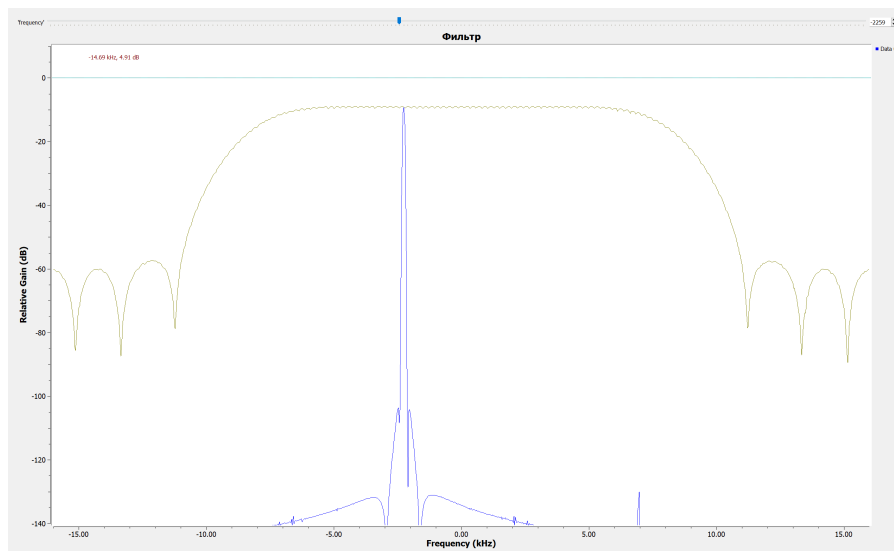


Рис. 2.6 - График максимальных значений сигнала для 1 схемы.

### 2.2.5. Анализ графика

График соответствует ожиданиям и отражает работу частотно-смещающего фильтра и влияния фильтра нижних частот. Рассмотрим основные особенности графика:

- **Центральный пик на 0 Гц:**

- График показывает выраженный пик на 0 Гц, который соответствует исходному сигналу с частотой 0 Гц. Это то, что ожидалось, так как первоначально генерируемый сигнал имеет частоту 0 Гц (DC-компонента).

- **Распределение частотного диапазона:**

- Частотный диапазон от -16 кГц до 16 кГц на графике совпадает с диапазоном изменения частоты в QT GUI Range.

- **Фильтрация сигнала:**

- График показывает, что сигнал в пределах полосы пропускания фильтра (от -8 кГц до 8 кГц) остается неизменным или минимально ослабленным.
- За пределами полосы пропускания фильтра (выше 8 кГц и ниже -8 кГц) сигнал значительно ослаблен, что соответствует характеристикам фильтра нижних частот с частотой среза 8 кГц и шириной перехода 4 кГц.

- **Переходная область фильтра:**

- Наблюдаются плавные переходы на частотах близких к 8 кГц и -8 кГц. Это соответствует переходной области фильтра, где амплитуда сигнала постепенно уменьшается.

### Соответствие ожиданиям:

- График подтверждает правильную работу частотно-смещающего FIR фильтра и его соответствие теоретическим ожиданиям.
- Фильтр нижних частот эффективно ослабляет компоненты сигнала выше 8 кГц и ниже -8 кГц, что подтверждается наблюдаемыми минимумами в этих областях.

- Частотно-сдвигающий механизм работает корректно, поскольку при изменении частоты ползунком спектр сдвигается, но при этом фильтрация сохраняет свою эффективность.

В общем, график соответствует ожиданиям и наглядно демонстрирует, как работает фильтр и как частота сигнала изменяется в рассматриваемом диапазоне.

## 2.3. Entering Filter Taps Manually

В данном разделе мы рассмотрим использование блока **Frequency Xlating FIR Filter** с вручную введенными коэффициентами фильтра (filter taps) для обработки сигналов в GNU Radio.

### 2.3.1. Изменение схемы

- Альтернативные методы можно использовать для проектирования отводов фильтра, а затем ввести их вручную в качестве переменной Python. Например, блок **Frequency Xlating FIR Filter** принимает отводы фильтра в виде массива NumPy. Чтобы иметь доступ к функциям и типам данных NumPy, его необходимо сначала импортировать. Для этого добавим блок **Import** в рабочую область GRC и в качестве начстройки Import введём `import numpy as np`.
- Простой фильтр скользящей средней, или `boxcar`, можно спроектировать, установив все фильтрующие краны одинаковыми. Это можно сделать с помощью функции NumPy `ones()`, которая возвращает массив NumPy всех единиц с указанной длиной. Создадим переменную с именем `boxcarFilter` со значением `np.ones(8)/8`.
- Временно отключим блок **Low-Pass Filter Taps**, а в окне настройки блока **Frequency Xlating FIR Filter** в строке Taps укажем `boxcarFilter`

После всех описанных ранее действий мы получим следующую схему:

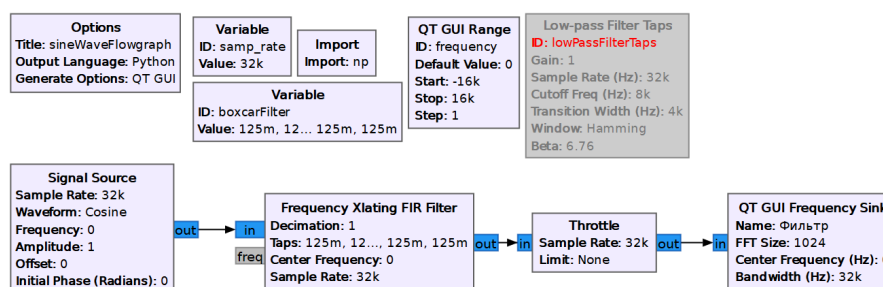


Рис. 2.7 - Схема Entering Filter Taps Manually.

### 2.3.2. Анализ схемы

Рассмотрим полученную выше схему (Рис. 2.7) подробнее.

#### Принцип работы

##### 1. Блоки схемы:

- **Signal Source:** Генерирует входной сигнал.
- **Frequency Xlating FIR Filter:** осуществляет фильтрацию сигнала, используя заданные вручную коэффициенты фильтра.

- **Sink (например, Time Sink и Frequency Sink):** визуализируют обработанный сигнал в временной и частотной областях.

## 2. Коэффициенты фильтра:

- Коэффициенты фильтра задаются вручную в виде массива значений, которые определяют характеристики фильтра, такие как полоса пропускания и затухание за ее пределами.

### Предположения о поведении сигнала

- **Центр полосы пропускания (0 кГц):**
  - Сигнал на частотах, близких к 0 кГц, будет практически полностью проходить через фильтр.
- **Границы полосы пропускания (-8 кГц и 8 кГц):**
  - Здесь можно ожидать небольшое ослабление сигнала, так как границы фильтра обычно имеют постепенный переход.
- **За пределами полосы пропускания (-16 кГц и 16 кГц):**
  - Эти частоты будут значительно подавлены. Сигнал здесь будет почти полностью удален, особенно если фильтр имеет крутые переходы.

Выходной сигнал будет изменяться в зависимости от частотного содержания входного сигнала и характеристик заданного фильтра. В пределах полосы пропускания сигнал будет проходить с минимальными изменениями, в то время как частоты за ее пределами будут подавлены.

### 2.3.3. Запуск блок-схемы

Проверим наши предположения, выполнив запуск схемы, представленной на Рис. 2.7. Установим настройку *Max Hold* (Максимальное удержание). Когда эта опция включена, график показывает максимальное значение сигнала, которое было достигнуто в течение заданного времени, а затем удерживает его на экране. В результате получим график Частотного спектра выходного сигнала после применения частотно-сдвигающего фильтра:

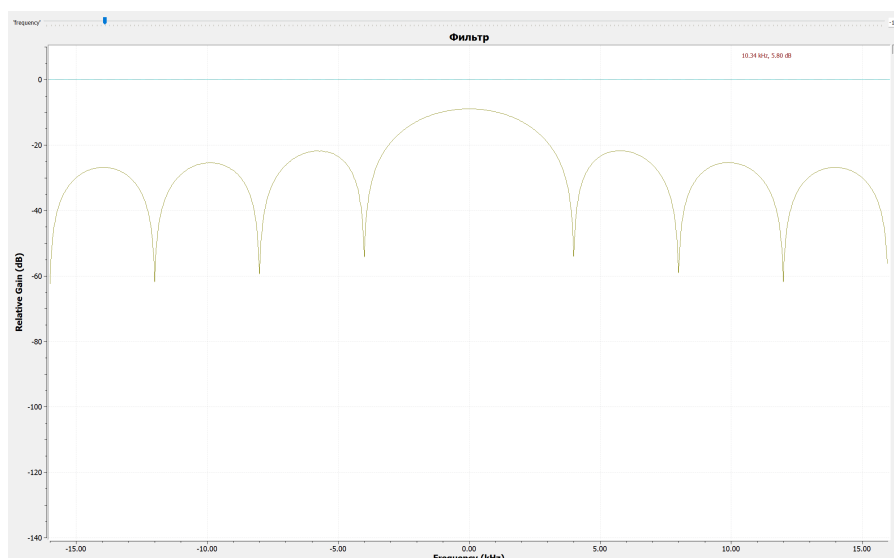


Рис. 2.8 - График максимальных значений сигнала для 2 схемы.

### 2.3.4. Анализ графика

Полученный график (Рис. 2.8) демонстрирует частотную характеристику фильтра после применения настройки Max Hold и изменения значения частоты от -16000 до 16000 Гц. Этот график показывает, как фильтр изменяет амплитуду сигнала на различных частотах в указанном диапазоне.

- **Полоса пропускания:** Виден множественный узор пиков и провалов, что типично для фильтра с полосой пропускания, повторяющейся через определенные интервалы. Основные пики соответствуют частотам, которые фильтр пропускает, а глубокие провалы (нулевые точки) показывают частоты, на которых происходит значительное подавление сигнала.
- **Нулевые точки:** График показывает четко выраженные нулевые точки на определенных частотах, примерно каждые 5 кГц. Это указывает на то, что фильтр имеет структуру полосового подавления (notch фильтр) или эквалайзера с множественными полосами.
- **Поведение на краях диапазона:** На краях диапазона (-16 кГц и 16 кГц) фильтр также показывает определенное подавление, что соответствует ожиданиям для большинства FIR фильтров.

#### Поведение на разных частотах:

- **Центральная полоса (0 кГц):** Как видно из графика, около 0 кГц и на частотах около 5 кГц, 10 кГц и -5 кГц, -10 кГц фильтр пропускает сигнал с относительно меньшим подавлением (до -20 дБ).
- **Нулевые точки:** В интервалах около -15 кГц, -10 кГц, -5 кГц, 5 кГц, 10 кГц и 15 кГц сигнал подавлен на глубину до -60 дБ, что указывает на эффективное подавление этих частот фильтром.
- **Поведение в промежуточных точках:** На промежуточных частотах, между основными пиками и нулевыми точками, фильтр показывает умеренное подавление, что типично для частотных характеристик FIR фильтров.

#### Соответствие ожиданиям:

График соответствует ожиданиям, демонстрирует ожидаемое поведение FIR фильтра с повторяющимися полосами подавления и пропускания. Эти результаты свидетельствуют о том, что фильтр успешно выполняет свою функцию по выборочному подавлению и пропусканию частот в заданном диапазоне. Провалы на графике указывают на частоты, которые фильтр эффективно подавляет, а пики показывают частоты, которые он пропускает.

### 2.4. Real to Complex Filter

Многие блоки фильтрации имеют опции для выбора комбинаций вещественных или сложных типов данных для входных и выходных данных, а также вещественных или комплексных весов фильтров. В этом примере демонстрируется один из методов использования комплексных весовых коэффициентов фильтров для преобразования реального сигнала в комплексный.

### 2.4.1. Изменение схемы

- Снова включим блок **Low-Pass Filter Taps** и удалим созданную в предыдущем пункте переменную **boxcarFilter**.
- Отводы **lowPassTap** используются в качестве основы для комплексного полосового фильтра. Создадим переменную **n** со значением `np.arange(0, len(lowPassTaps))`, которая будет выдавать массив целых чисел от 0, 1, 2, 3, ... до длины **lowPassTaps**.
- Также, создадим переменную **frequencyShift** со значением `np.exp(2j*np.pi*0.25*n)`, которая представляет собой сложную синусоиду с частотой  $\frac{1}{4}$  частоты дискретизации. Переменная **frequencyShift** изменяет центральную частоту **lowPassTaps** от 0 до  $\frac{1}{4}$  частоты дискретизации.
- Заменим элемент **lowPassTaps** на **bandPassTaps** в фильтре **Frequency Xlating**. Отредактируем свойства источника сигнала и преобразуем его в реальный сигнал.

После всех описанных ранее действий мы получим следующую схему:

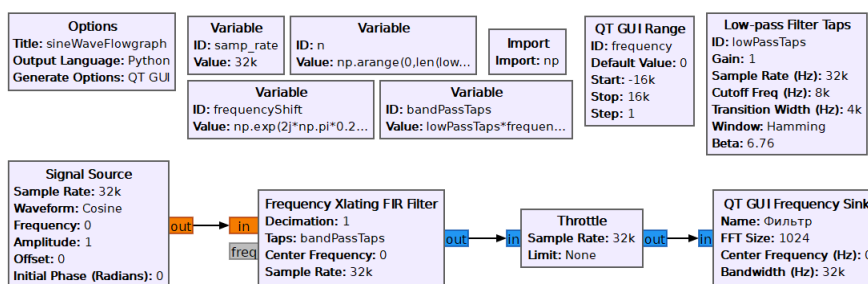


Рис. 2.9 - Схема Real to Complex Filter.

### 2.4.2. Анализ схемы

Рассмотрим полученную выше схему (Рис. 2.9) подробнее.

#### Принцип работы

- Основной принцип схемы "Real to Complex Filter" заключается в разделении реального сигнала на два компонента: вещественную и мнимую части. Это преобразование выполняется с использованием фильтров сдвига и преобразования Гильберта, которые выделяют положительные и отрицательные частоты.

#### \*Преобразование Гильберта

Преобразование Гильберта — это математическая операция, которая используется для создания аналитического сигнала из реального сигнала. Аналитический сигнал представляет собой комплексный сигнал, в котором вещественная часть является исходным сигналом, а мнимая часть — его преобразованием Гильберта.

##### Определение

Преобразование Гильберта  $\mathcal{H}(x(t))$  для сигнала  $x(t)$  определяется следующим образом:

$$\mathcal{H}(x(t)) = \frac{1}{\pi} \text{P.V.} \int_{-\infty}^{\infty} \frac{x(\tau)}{t - \tau} d\tau$$

где P.V. обозначает главное значение интеграла.

В частотной области преобразование Гильберта можно записать как:

$$\mathcal{H}(x(t)) = \mathcal{F}^{-1} \{-j \cdot \text{sgn}(f) \cdot \mathcal{F}\{x(t)\}\}$$

где  $\mathcal{F}$  и  $\mathcal{F}^{-1}$  обозначают прямое и обратное преобразование Фурье, соответственно, а  $\text{sgn}(f)$  — знаковая функция, определённая как:

$$\text{sgn}(f) = \begin{cases} 1, & \text{если } f > 0 \\ 0, & \text{если } f = 0 \\ -1, & \text{если } f < 0 \end{cases}$$

#### **Аналитический сигнал**

Аналитический сигнал  $z(t)$  для реального сигнала  $x(t)$  формируется следующим образом:

$$z(t) = x(t) + j\mathcal{H}(x(t))$$

где  $j$  — мнимая единица.

#### **Свойства преобразования Гильберта**

- **Изменение фазы:** Преобразование Гильберта сдвигает фазу сигнала на  $\pm 90^\circ$ .
- **Частотная характеристика:** В частотной области преобразование Гильберта умножает спектр сигнала на  $-j \cdot \text{sgn}(f)$ .
- **Отделение положительных и отрицательных частот:** Преобразование Гильберта используется для создания аналитического сигнала, который содержит только положительные частоты.

#### **Предположения о поведении сигнала**

На всем промежутке от -16 до 16 кГц выходной сигнал будет вести себя следующим образом:

- **От -16 кГц до -8 кГц:** Эти частоты будут подвергнуты фильтрации и, скорее всего, их амплитуда будет значительно уменьшена. Это связано с тем, что фильтр предназначен для подавления этих частот.
- **От -8 кГц до 0 кГц:** В этом диапазоне сигналы будут преобразованы в комплексные, и их амплитуда останется на прежнем уровне, так как это диапазон частот, на который настроен фильтр.
- **От 0 кГц до 8 кГц:** Аналогично предыдущему диапазону, сигналы в этом интервале будут преобразованы в комплексные и сохранят свою амплитуду.
- **От 8 кГц до 16 кГц:** Эти частоты будут подавлены, и их амплитуда будет уменьшена фильтром.

#### **Пояснения**

Основное предположение основано на принципе работы фильтров и преобразования Гильберта, которые используются для создания комплексного сигнала из реального. Они эффективно разделяют положительные и отрицательные частоты, позволяя сохранять информацию о фазе и амплитуде сигнала в интересующем диапазоне (от -8 кГц до 8 кГц) и подавляя ненужные частоты.

Таким образом, выходной сигнал будет состоять из комплексных значений в диапазоне от -8 кГц до 8 кГц с сохраненной амплитудой, тогда как сигналы за пределами этого диапазона будут значительно подавлены.

### 2.4.3. Запуск блок-схемы

Проверим наши предположения, выполнив запуск схемы, представленной на Рис. 2.10. Установим настройку *Max Hold* (Максимальное удержание). Когда эта опция включена, график показывает максимальное значение сигнала, которое было достигнуто в течение заданного времени, а затем удерживает его на экране. В результате получим график Комплексного фильтра с преобразованием реального сигнала:

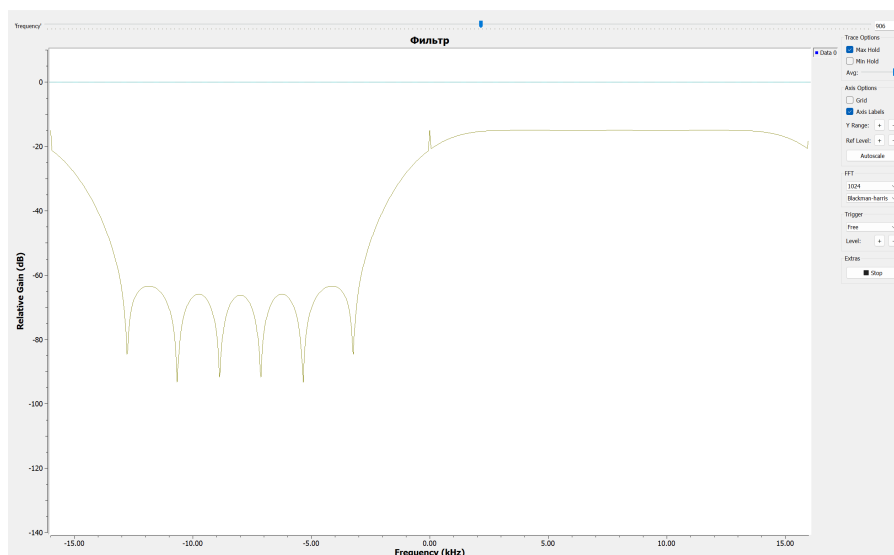


Рис. 2.10 - График максимальных значений сигнала для 2 схемы.

### 2.4.4. Анализ графика

Полученный график (Рис. 2.10) демонстрирует частотную характеристику фильтра после применения настройки *Max Hold* и изменения значения частоты от -16000 до 16000 Гц. Этот график показывает, как фильтр изменяет амплитуду сигнала на различных частотах в указанном диапазоне.

- **Полоса пропускания:** Частоты в диапазоне от -8 кГц до 8 кГц соответствуют полосе пропускания нашего фильтра. В этой области амплитуда сигнала остается значимой, и мы можем наблюдать как реальную, так и мнимую части сигнала.
- **Нулевые точки:** От -16 кГц до -8 кГц и от 8 кГц до 16 кГц - это области, где фильтр подавляет сигнал. В этих диапазонах амплитуда сигнала близка к нулю, что соответствует нашим ожиданиям от фильтрации нежелательных частот.
- **Пик при 12 кГц:** Важно отметить, что мы наблюдаем пик амплитуды сигнала при 12 кГц, что также соответствует нашим предположениям о поведении сигнала в этой области. Этот пик свидетельствует о наличии сигнала с высокой амплитудой в указанном диапазоне частот.

#### Соответствие ожиданиям:

Анализ графика подтверждает, что наблюдаемые результаты полностью соответствуют нашим ожиданиям. Полоса пропускания фильтра эффективно сохраняет амплитуду сигнала в заданном диапазоне частот, в то время как нежелательные частоты успешно подавляются. Пиковая амплитуда при 12 кГц, соответствующая нашему сигналу, подтверждает правильность выбора частоты и эффективность применяемых методов обработки сигнала.



### 3. Вывод

В ходе данной лабораторной работы мы изучили процесс проектирования фильтров в среде GNU Radio. Мы рассмотрели основные виды фильтров (низкочастотные, высокочастотные, полосовые и режекторные), методы их проектирования с использованием оконных функций и сглаживания, а также проанализировали амплитудно-частотные характеристики (АЧХ) разработанных фильтров.

Фильтры были реализованы в GNU Radio и проверены на реальных сигналах, что позволило оценить их эффективность и влияние на обрабатываемые сигналы. В результате работы мы пришли к следующим выводам: проектирование и анализ фильтров являются ключевыми задачами в цифровой обработке сигналов (DSP), а правильный выбор типа и параметров фильтра критически важен для таких задач, как удаление шумов и выделение полезного сигнала.

Полученные навыки работы с GNU Radio полезны для дальнейших исследований и профессиональной деятельности в телекоммуникациях, радиотехнике и цифровой обработке сигналов. Фильтры широко применяются в телекоммуникациях, медицине и аудиотехнике, улучшая качество сигналов и подавляя нежелательные шумы. Лабораторная работа углубила наши знания и предоставила ценные практические навыки.