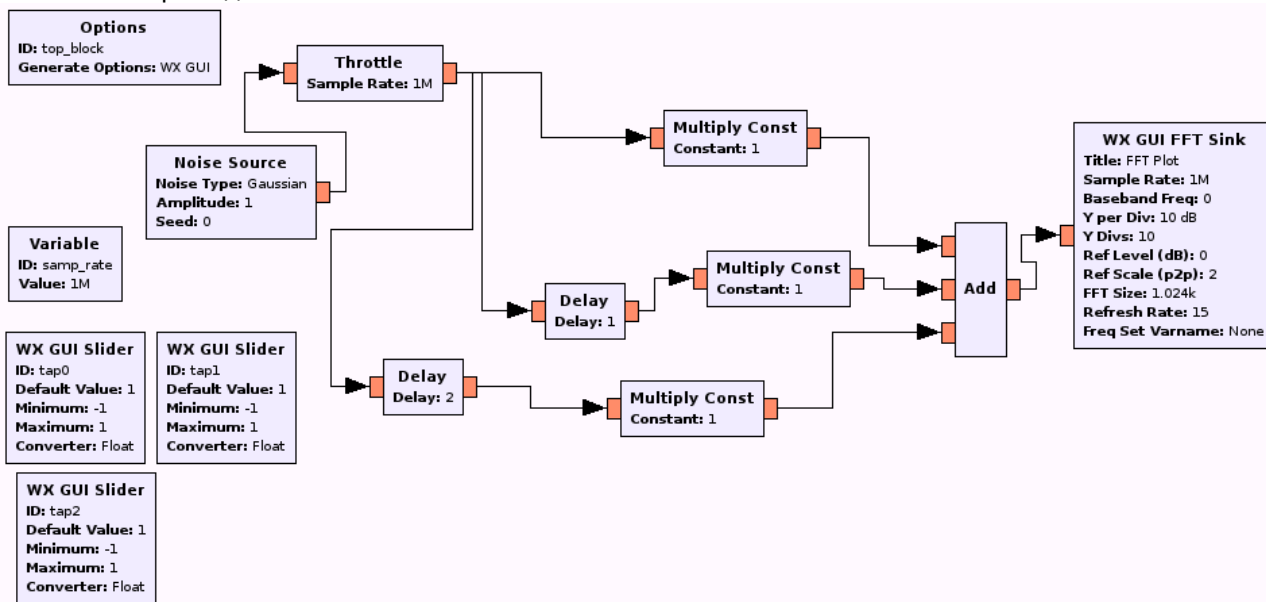


Занятие № 4. «Изучение преобразований цифрового сигнала в среде GNU Radio»

Порядок выполнения работы.

1. Установить и запустить среду визуальной разработки GnuRadio. Собрать блок схему виртуального анализатора спектра: В среде Gnuradio создайте блок-схему, изображенную на скриншоте ниже. Проверьте работоспособность собранной схемы, запустив ее и построив спектр входного сигнала



2. Запустите flowgraph и настройте ползунки, чтобы попытаться создать фильтр нижних частот с полосой останова на 20 дБ ниже полосы пропускания и с полосой пропускания фильтра (полосы пропускания), близкой к половине видимой полосы пропускания. Оцените край полосы пропускания как точку, где частотная характеристика проходит на 3 дБ ниже самого высокого отклика в полосе пропускания.
3. Создайте блок-схему, содержащую блок фильтра нижних частот. (Подсказка: можете использовать блок-схему из предыдущего задания.)
4. Выполните **FlowGraph** или используйте функцию Generate в GNU Radio Companion для генерации вывода программы Python.
5. Найдите сохраненный код программы Python. (Подсказка: имя файла указано без расширения .py в блоке «Параметры» в верхнем левом углу блок-графика).
6. Используйте текстовый редактор или просмотрщик файлов для просмотра программы Python и найдите функцию firdes (FIR design) внутри программы.
7. При каких условиях выполняется функция firdes?
8. Сохраните полученные файлы .grc и подготовьте отчет.

Описание используемых в работе блоков

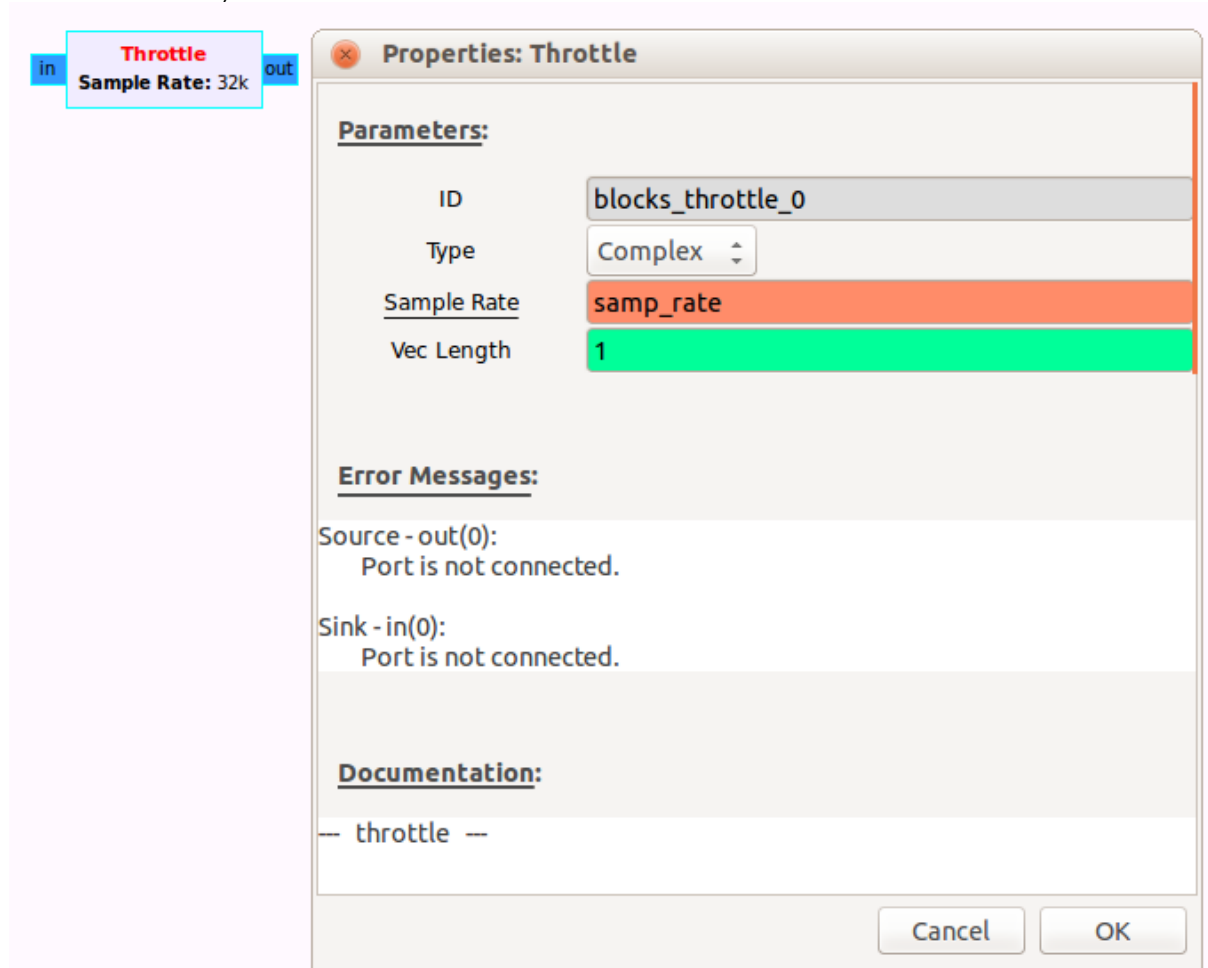
Throttle

[Type](#)

[Sample Rate](#)

[Vec Length](#)

Limits the data throughput to the specified sampling rate. This prevents GNURadio from consuming all CPU resources when the flowgraph is not being regulated by external hardware (ie: audio source/sink or USRP source/sink).



Type

Specifies the data type of the input and output.

Complex	Input and output are complex values.
Float	Input and output are floating point (real) values.
Int	Input and output are integer values.
Short	Input and output are short integer values.
Byte	Inputs and output are byte values.

Sample Rate

Type: real

Specifies the sample rate to limit the flowgraph to.

Vec Length

Type: int

Specifies the vector length for vector processing. Typical applications will use the default value of 1.

Signal Source

[Output Type](#)

[Sample Rate](#)

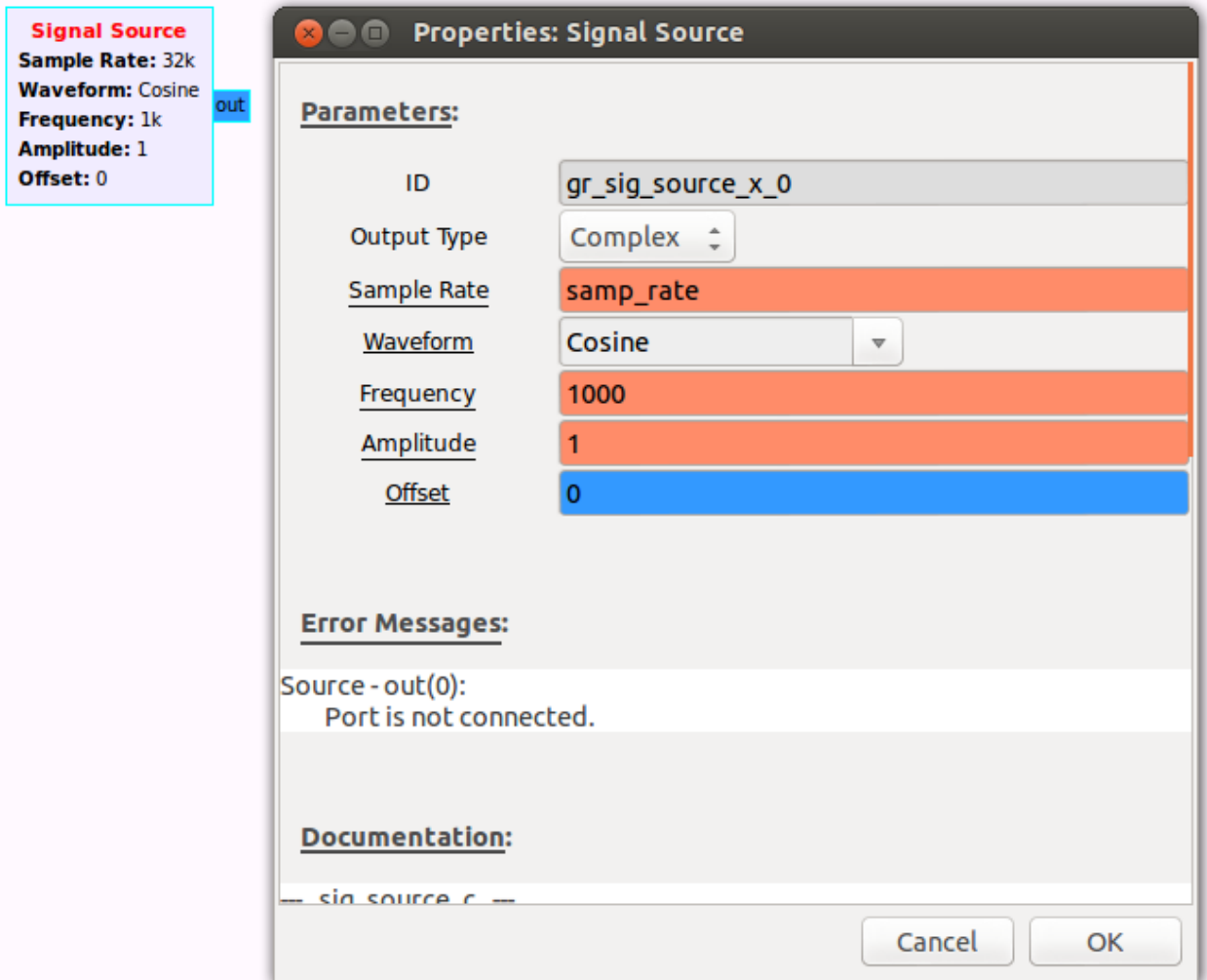
[Waveform](#)

[Frequency](#)

[Amplitude](#)

[Offset](#)

Used to generate a variety of signal types: Sine, Cosine, Square, Triangle and Sawtooth.



Output Type

Specifies the data type of the output.

Complex	Output is complex-valued.
Float	Output is real-valued.
Int	Output is a 32-bit integer.
Short	Output is a 16-bit integer.

Sample Rate

Type: real

Specifies the output sample rate.

Waveform

Specifies the output waveform.

Constant	Output is a constant value equal to the Amplitude parameter plus the Offset parameter. Note that Amplitude is only real while Offset can be complex. The Constant Source block provides the same functionality.
Sine	Output is a sine wave with peak amplitude configured by the Amplitude parameter and average value set by the Offset parameter.
Cosine	Output is a cosine wave with peak amplitude configured by the Amplitude parameter and average value set by the Offset parameter.
Square	Output is a square wave with peak-to-peak amplitude configured by the Amplitude parameter and the average value set by Offset + Amplitude/2. Note that in the Complex case, the imaginary signal is simply another square wave that has been shifted by ninety degrees.
Triangle	Output is a triangle wave with peak-to-peak amplitude configured by the Amplitude parameter and the average value set by Offset + Amplitude/2. Note that in the Complex case, the imaginary signal is simply another triangle wave that has been shifted by ninety degrees.
Sawtooth	Output is a positive-going sawtooth wave with peak-to-peak amplitude configured by the Amplitude parameter and the average value set by Offset + Amplitude/2. Note that in the Complex case, the imaginary signal is simply another sawtooth wave that has been shifted by ninety degrees.

Frequency

Type: real

Specifies the output frequency of the **Signal** Source. Note that aliasing will occur if the frequency is set higher than half of the sample rate.

Amplitude

Type: real

Specifies the peak amplitude (Sine and Cosine) or the peak-to-peak amplitude (Square, Triangle and Sawtooth). When using the Constant output, typically this is set to 0 and the Offset parameter is used.

Offset

Type: Complex

Specifies the offset that is added to the generated waveform.

Delay

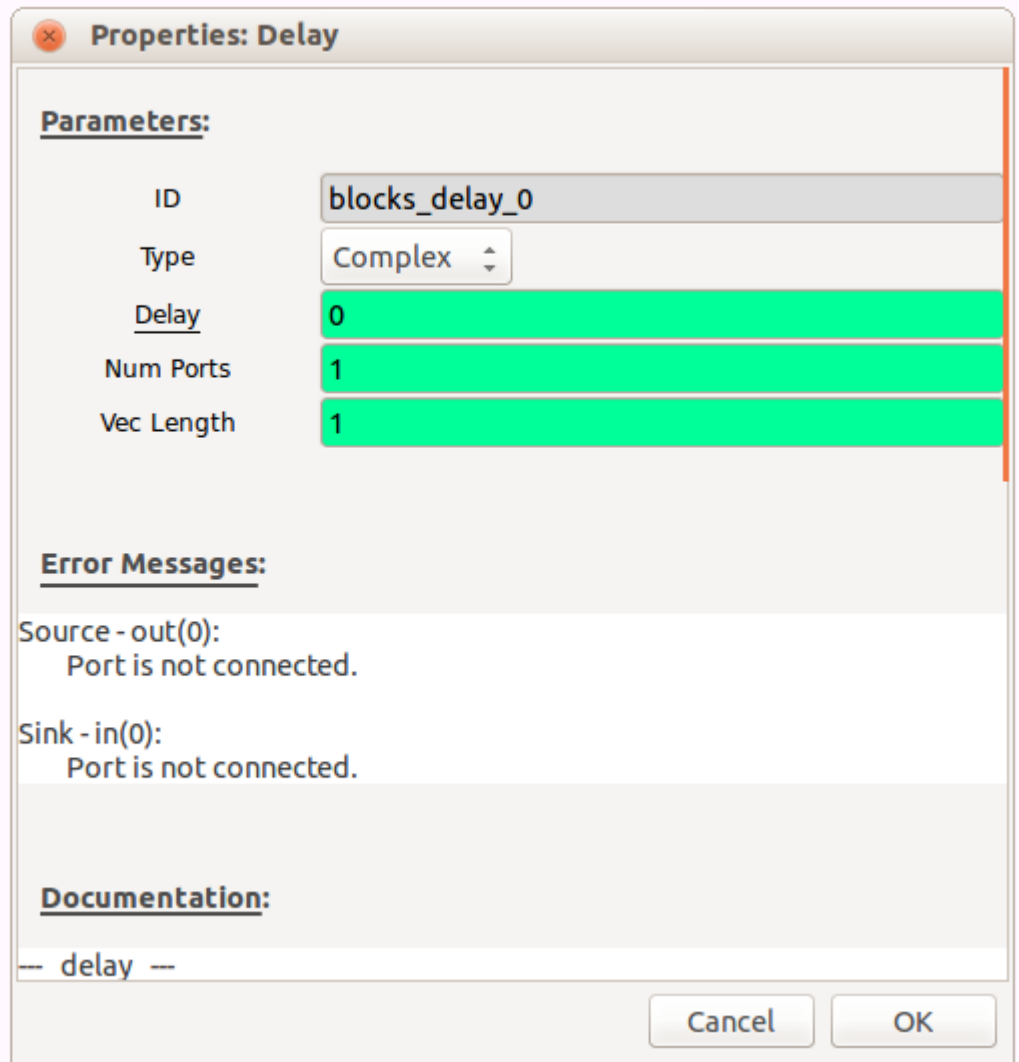
[Type](#)

[Delay](#)

[Num Ports](#)

[Vec Length](#)

Implements a **delay** equal to the specified number of samples.



Type

Specifies the data type of the input and output.

Complex	Input and output are complex values.
Float	Input and output are floating point (real) values.
Int	Input and output are integer values.
Short	Input and output are short integer values.
Byte	Input and output are byte values.

Delay

Type: int

Specifies the delay in samples. Note that the actual delay time depends on the sample rate in this branch of the flowgraph.

Num Ports

Type: int

Specifies the number of inputs/outputs. Essentially creates parallel delay lines.

Vec Length

Type: int

Specifies the vector length for vector processing. Typical applications will use the default value of 1.

