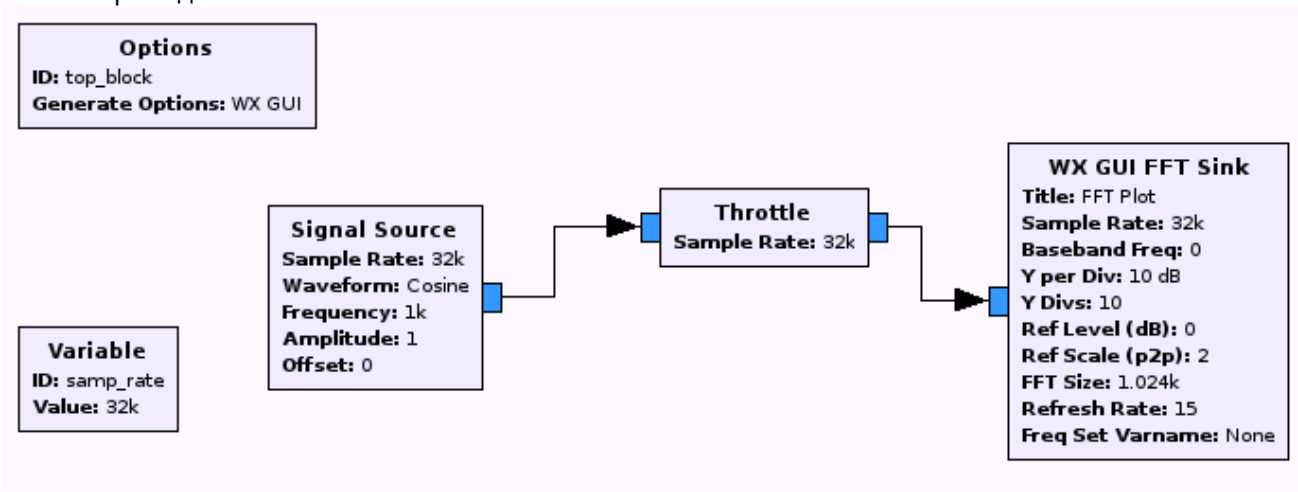
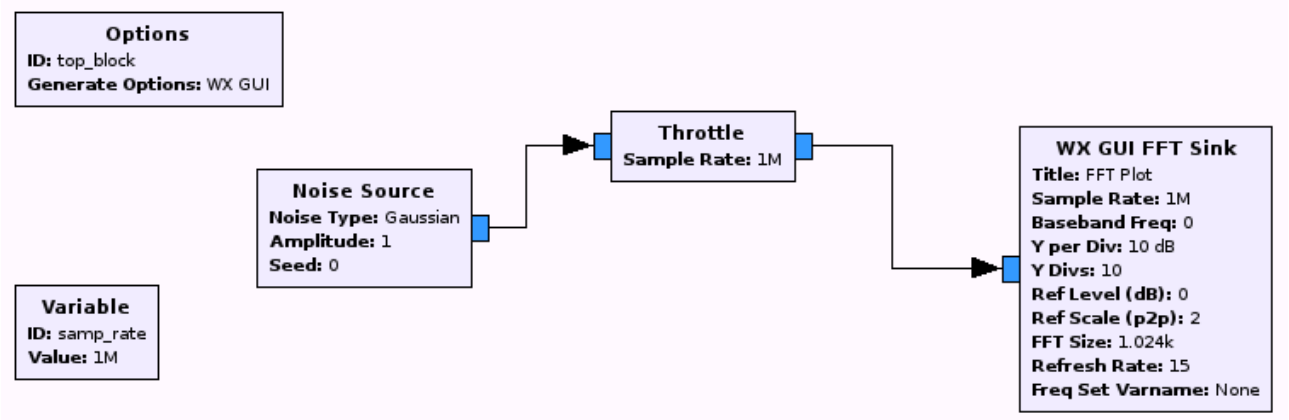
Занятие № 3. Исследование шумов и искажений цифрового сигнала

Порядок выполнения работы.

1. Установить и запустить среду визуальной разработки.
2. Знакомство с программно-определяемой радиосистемой (Software-defined radio - SDR) HackRF и со средой визуального программирования GNU Radio Companion. Изучение принципов ЦОС в среде GNU Radio
3. Блок-схема анализатора спектра: В среде Gnuradio создайте блок-схему, изображенную на скриншоте ниже. Проверьте работоспособность собранной схемы, запустив ее и построив спектр входного сигнала



4. Начните с блок схемы из задания 2. Добавьте виртуальный осциллограф (**Scope Sink**)
5. Как изменятся показания анализатора спектра и осциллографа при изменении типа данных с **complex** на **float**?
6. Что произойдет при задании отрицательных частот и типа данных **float**? Что произойдет при задании отрицательных частот и типа данных **float**?
7. Соберите блок схему, изображенную ниже.



8. Запустите блок-схему на выполнение и сделайте скриншоты при включенной и выключенной опции **Average** (усреднение). Сделайте вывод о спектральном составе шума.
9. Добавьте в схему блок **Low Pass Filter**, а также два блока **Slider** для изменения параметров фильтра частота среза (**filter's cutoff frequency)** и ширины переходной полосы (**transition width**). Установите в блоке **Slider** значения по умолчанию равными samp_rate/8, минимальное значение - samp_rate/1000, а максимальное - samp_rate/2, и 1000 отсчетов (Steps).
10. Изменяя параметра фильтра, сделайте несколько скриншотов.
11. Откройте диспетчер задач Windows и оцените загрузку процессора при изменении параметров фильтра. (**Tranisiton width**).

12. Замените блок Low pass filter, на high pass а затем band pass filter. После замены блока выполните пункт 9.

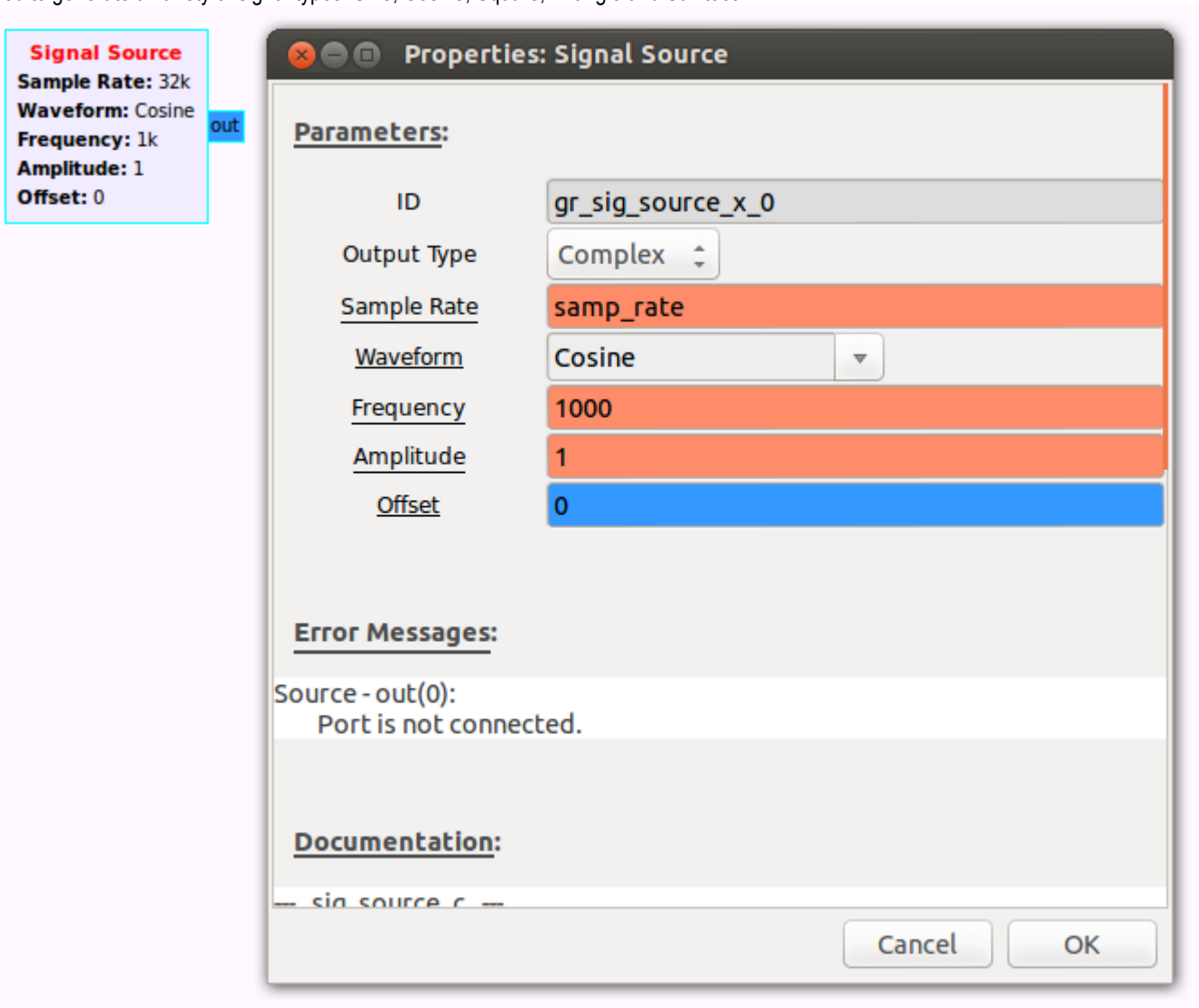
Описание используемых в работе блоков

С документацией к используемым блокам можно самостоятельно ознакомиться, воспользовавшись справкой к программе в меню *Help* > *Help* (или нажав клавишу F1 в открытой программе).

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 paramater.
Cosine	Output is a cosine wave with peak amplitude configured by the Amplitude parameter and average value set by the Offset paramater.
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.

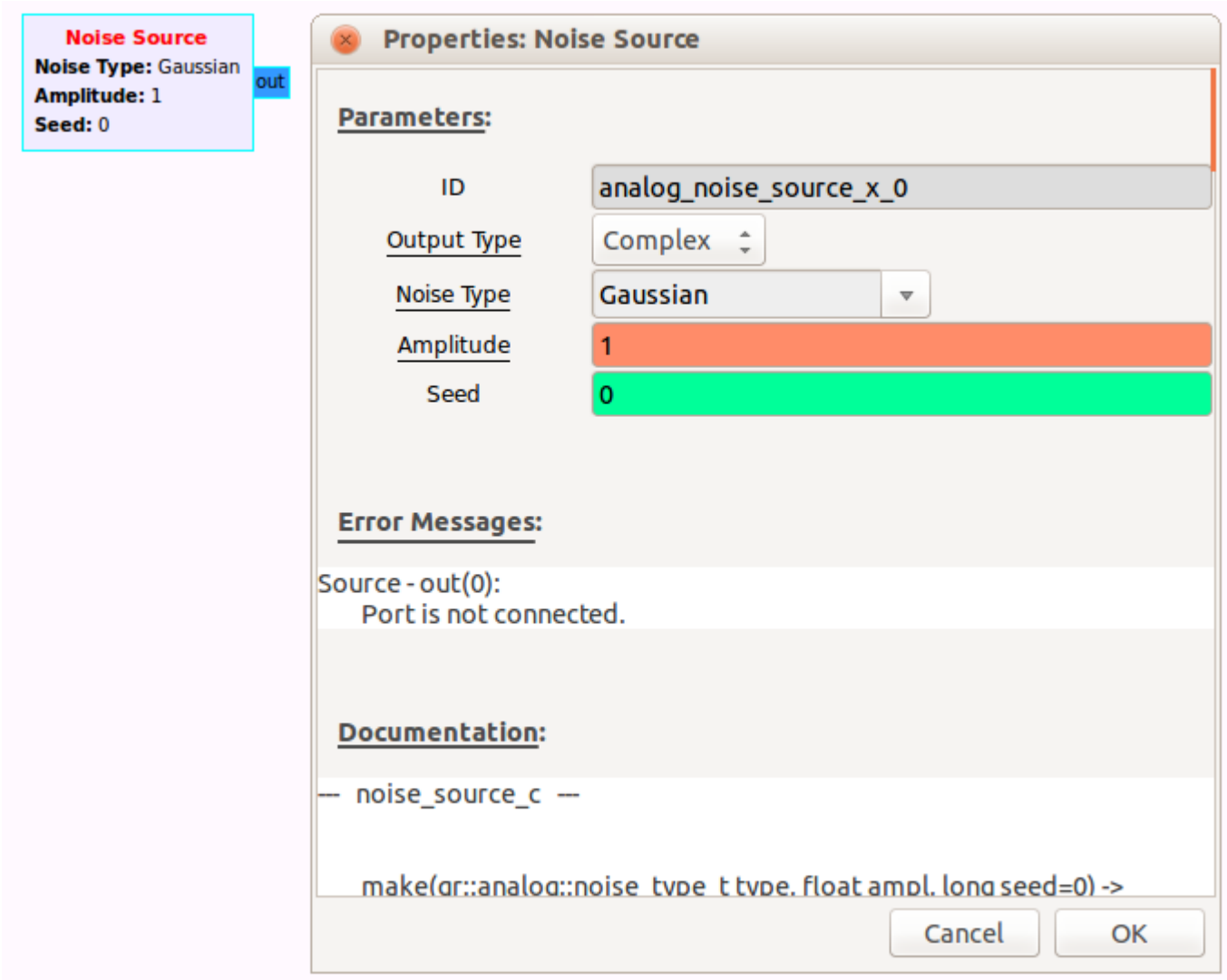
Frequency
Type: real
Specifies the output frequency of the Signal Source. Note that aliasing will occur if the frequency is set higher than half of the sample rate.

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.

Noise Source

Output Type
Noise Type
Amplitude
Seed
Implements a noise source. The distribution of the noise can be selected from various standard distributions.



Output Type

Sets the data type of the output stream.

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

Noise Type

Selects the statistical distribution of the noise.

Uniform	Selects a uniform distribution.
Gaussian	Selects a Gaussian distribution.
Laplacian	Selects a Laplacian distribution.
Impulse	Selects an Impulse distribution.

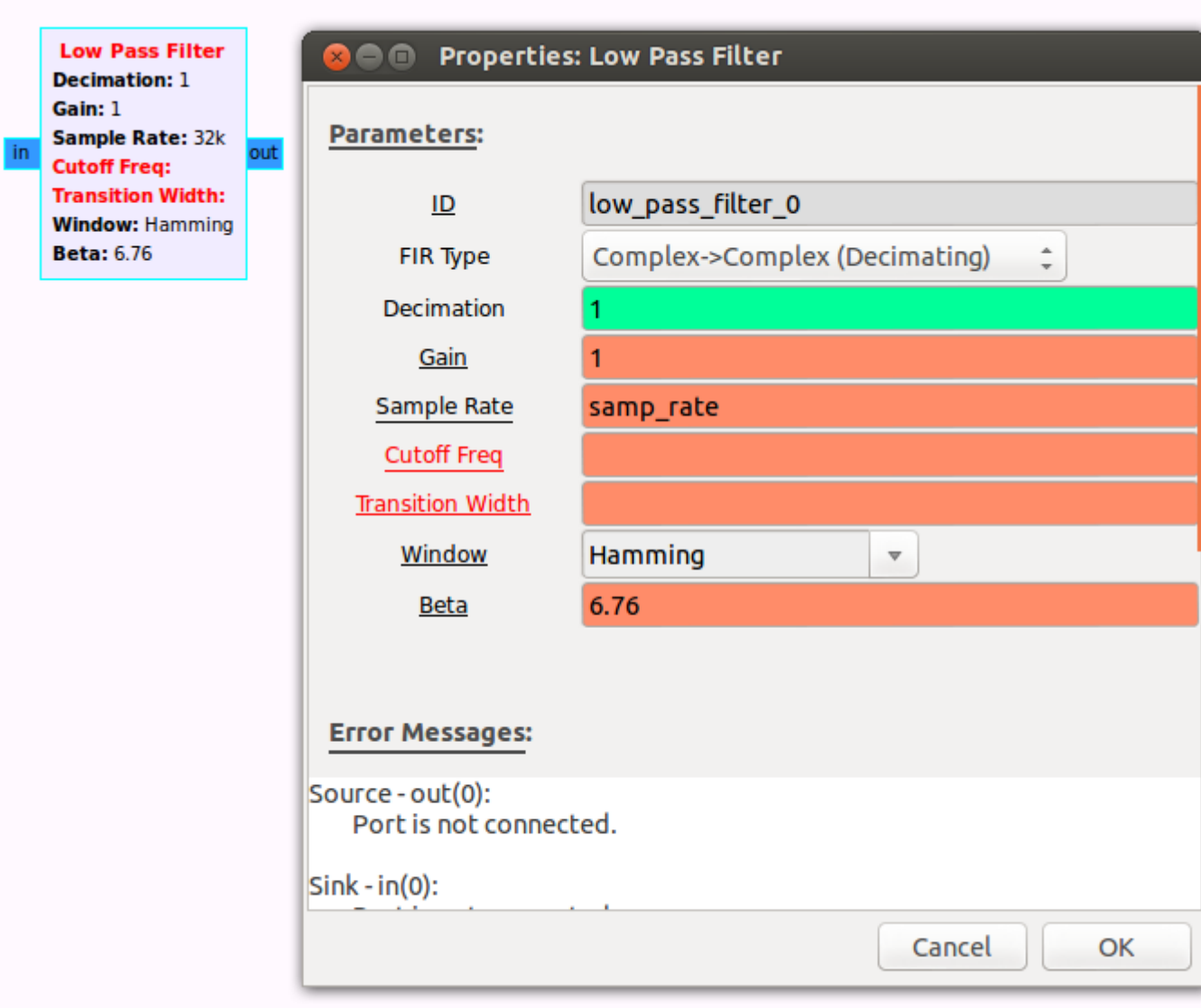
Amplitude

Type: real

Sets the amplitude of the output. For this will affect the statistical characteristics of the noise source. For example, it will affect the standard deviation of the Gaussian source.

Seed
Type: int
Sets the seed for the random number generator used to create the noise source. For most purposes, the default value is sufficient.

Low Pass Filter
FIR Type
Decimation/Interpolation
Gain
Sample Rate
Cutoff Freq
Transition Width
Window
Beta
This filter is a convenience wrapper for an FIR filter and a firdes taps generating function. Sample rate, cutoff frequency, and transition width are in Hertz. The beta paramater only applies to the Kaiser window.



FIR Type
Specifies the data type of the input and output streams, with the option of either decimation or interpolation.

Complex->Complex (Decimating)	Input and output streams are complex, with the option to decimate the output.
Complex->Complex	Input and output streams are complex, with the option to interpolate the

(Interpolating)	output.
Float->Float (Decimating)	Input and output streams are real, with the option to decimate the output.
Float->Float (Interpolating)	Input and output streams are real, with the option to interpolate the output.

Decimation/Interpolation
Type: int
Decimation or interpolation can be selected through the FIR Type parameter. If resampling is not required set this parameter to 1.

Gain
Type: real
Sets the gain of the filter.

Sample Rate
Type: real
Sets the sample rate of the filter, in Hz.

Cutoff Freq
Type: real
Sets the cutoff frequency of the filter, in Hz.

Transition Width
Type: real
Sets the transition width between the pass band and stop band. A small transition width will increase the length of the FIR filter.

Window
Specifies the window function that will be applied to the FIR filter.

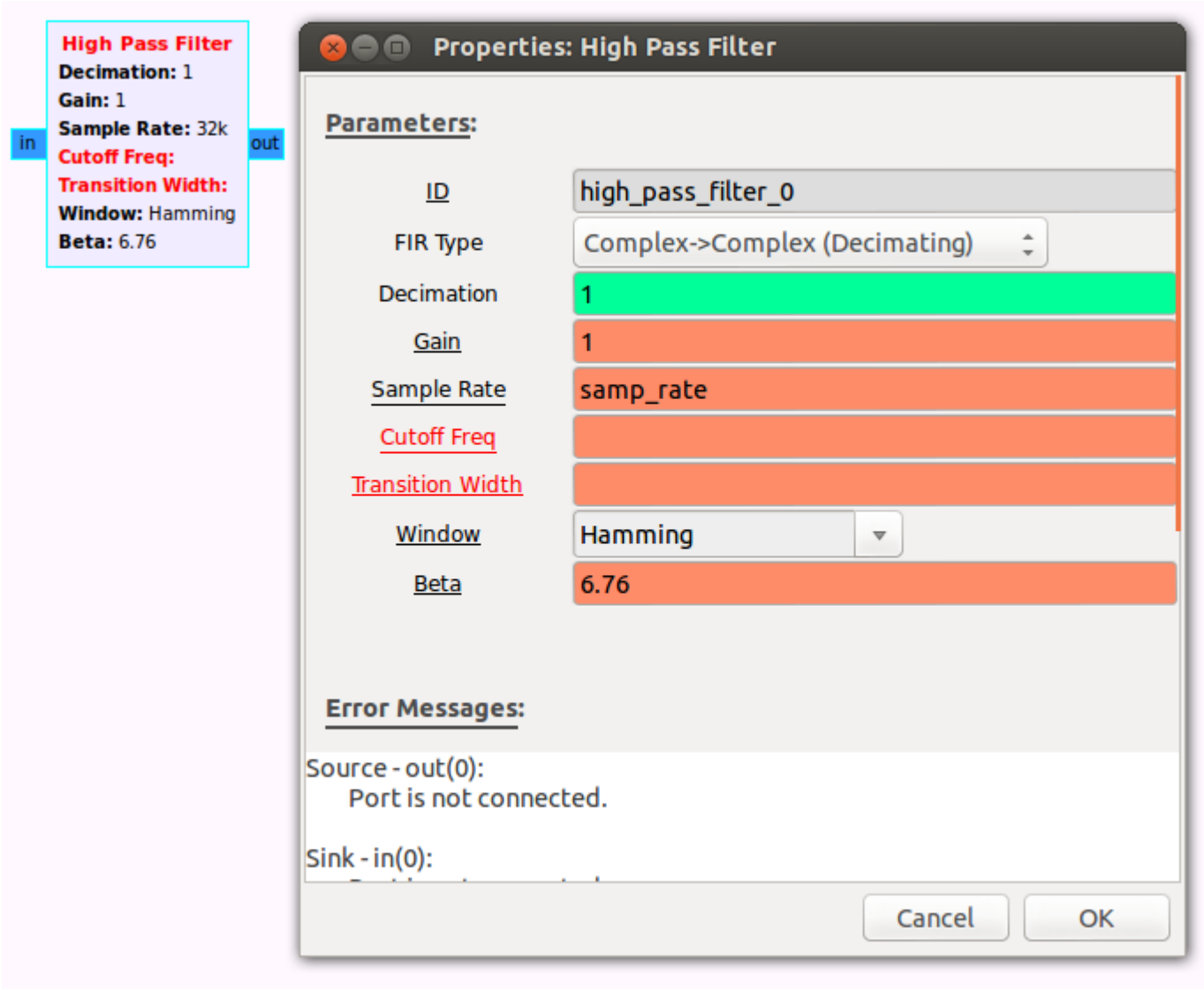
Hamming	Apply a Hamming window.
Hann	Apply a Hann window.
Blackman	Apply a Blackman window.
Rectangular	Apply a rectangular window.
Kaiser	Apply a Kaiser window. The Beta parameter can be controlled.

Beta
Type: real
Beta parameter for the Kaiser window.

High Pass Filter

FIR Type
Decimation/Interpolation
Gain
Sample Rate
Cutoff Freq
Transition Width
Window
Beta

This filter is a convenience wrapper for an FIR filter and a firdes taps generating function. Sample rate, cutoff frequency, and transition width are in Hertz. The beta paramater only applies to the Kaiser window.



FIR Type

Specifies the data type of the input and output streams, with the option of either decimation or interpolation.

Complex->Complex (Decimating)	Input and output streams are complex, with the option to decimate the output.
Complex->Complex (Interpolating)	Input and output streams are complex, with the option to interpolate the output.
Float->Float (Decimating)	Input and output streams are real, with the option to decimate the output.
Float->Float (Interpolating)	Input and output streams are real, with the option to interpolate the output.

Decimation/Interpolation

Type: int

Decimation or interpolation can be selected through the FIR Type parameter. If resampling is not required set this parameter to 1.

Gain

Type: real

Sets the gain of the filter.

Sample Rate

Type: real

Sets the sample rate of the filter, in Hz.

Cutoff Freq

Type: real

Sets the cutoff frequency of the filter, in Hz.

Transition Width

Type: real

Sets the transition width between the pass band and stop band. A small transition width will increase the length of the FIR filter.

Window

Specifies the window function that will be applied to the FIR filter.

Hamming	Apply a Hamming window.
Hann	Apply a Hann window.
Blackman	Apply a Blackman window.
Rectangular	Apply a rectangular window.
Kaiser	Apply a Kaiser window. The Beta parameter can be controlled.

Beta

Type: real

Beta parameter for the Kaiser window