Topics to be covered

- Introduction to digital filters.
- Steps for designing FIR filters.
- Zero shift FIR filter.
- Low Pass FIR filter.
- High Pass FIR filter.
- Band Pass filter.
- Band Stop filter.

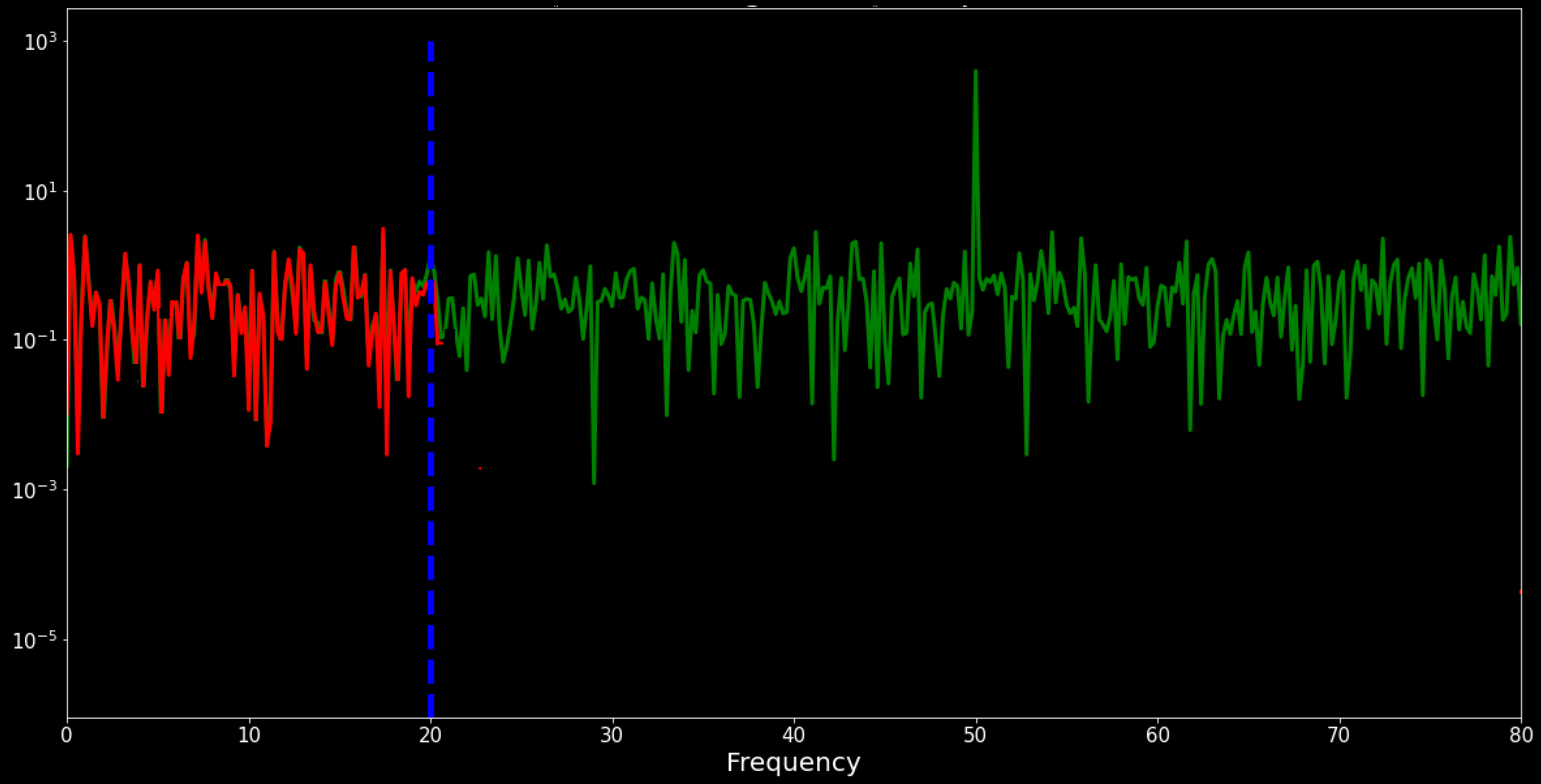
Digital Filters

We design digital filters to serve two purposes.

- Signal Separation.
- Signal Restoration.

Signal Separation

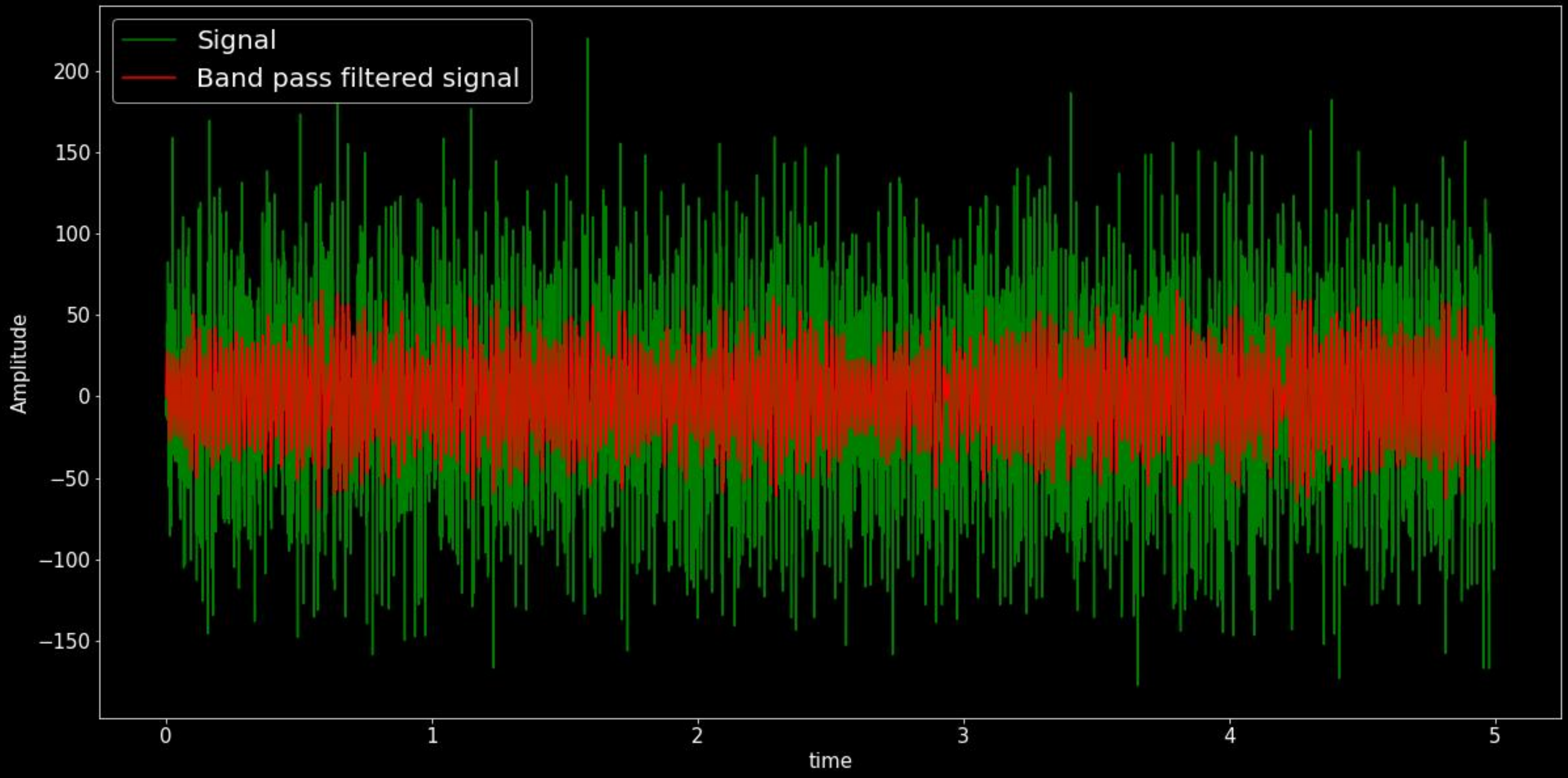
If two signals are spectrally separable then we can design a digital filter to separate them. Spectrally separable means that they have different frequency contents.



10^3
10^1
10^{-1}
10^{-3}
10^{-5}
0
10
20
30
40
50
60
70
80
Frequency

Signal Restoration

If a signal is corrupted by noise then we can restore original signal by designing digital filter with proper characteristics depending upon the type of noise corrupted the signal.


Signal
Band pass filtered signal
200
150
100
50
0
−50
−100
−150
Amplitude
0
1
2
3
4
5
time

Filter types based on Impulse Response

Based on Impulse Response, we have following two types of filters.

1. Finite Impulse Response (FIR) Filter
2. Infinite Impulse Response (IIR) Filter

Finite Impulse Response Filter

A finite impulse response (FIR) filter is a signal processing filter whose impulse response or response to any finite length input signal is of finite duration. This is because it converges to zero in a short time and thus it is a stable filter.

Infinite Impulse Response Filter

In contrast to FIR filter, Infinite Impulse Response (IIR) filters have impulse response which does not converge to zero over infinite length of time. Thus they are unstable filters.

Steps for designing FIR Filter

1. Describe the frequency domain shape of the filter with cut-off(s).
2. Define the frequency domain shape with cut-off(s) and transition width.
3. Generate filter with Least Square or Window method.
4. Evaluate the filter in the frequency domain using FFT.
5. Filter the given signal with the evaluated filter

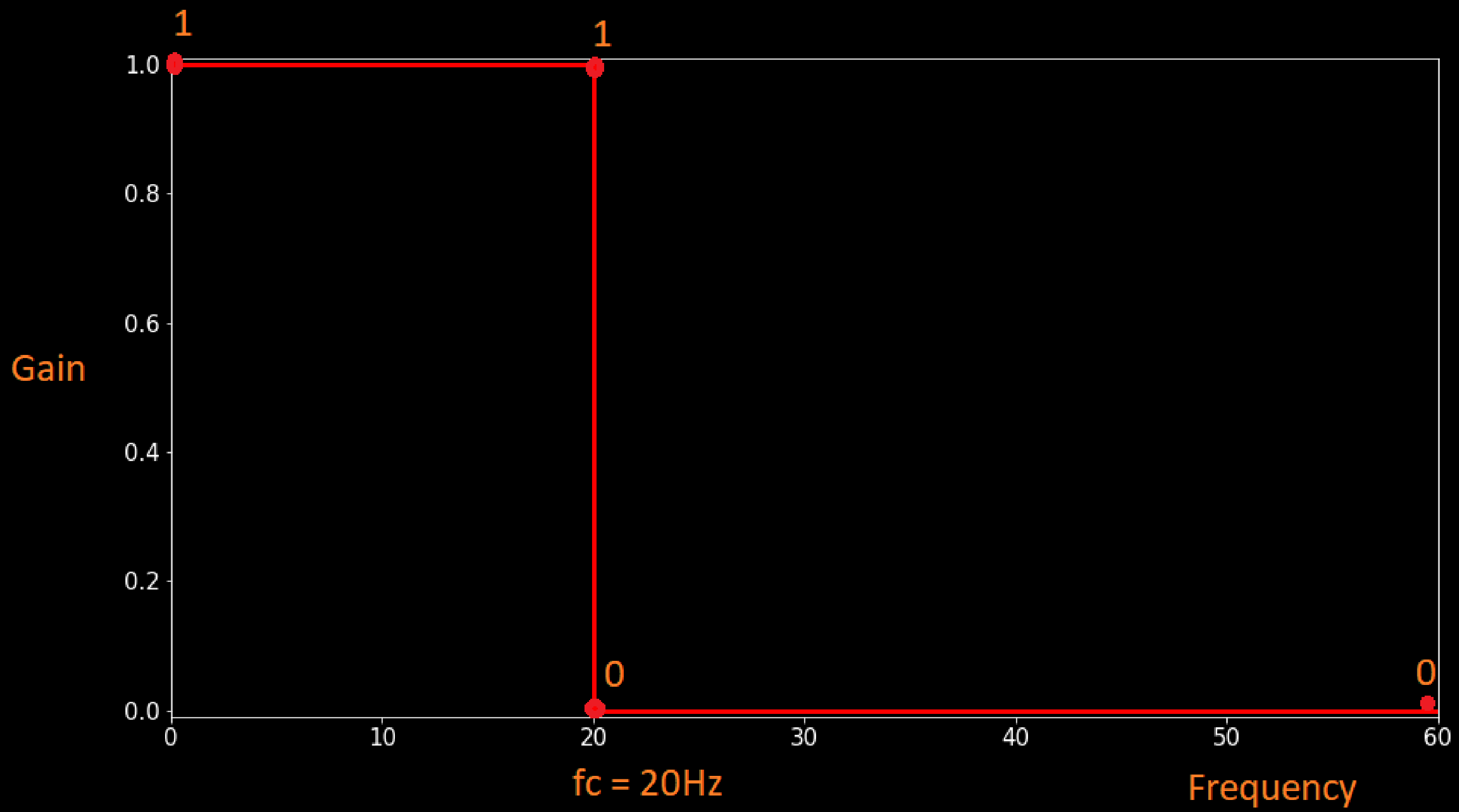
   OR

   Apply the filter to the given signal.

Step 1 : Define Frequency domain shapes and cutt-off(s)

1. Low pass filter
2. High pass filter
3. Band pass filter
4. Band stop filter

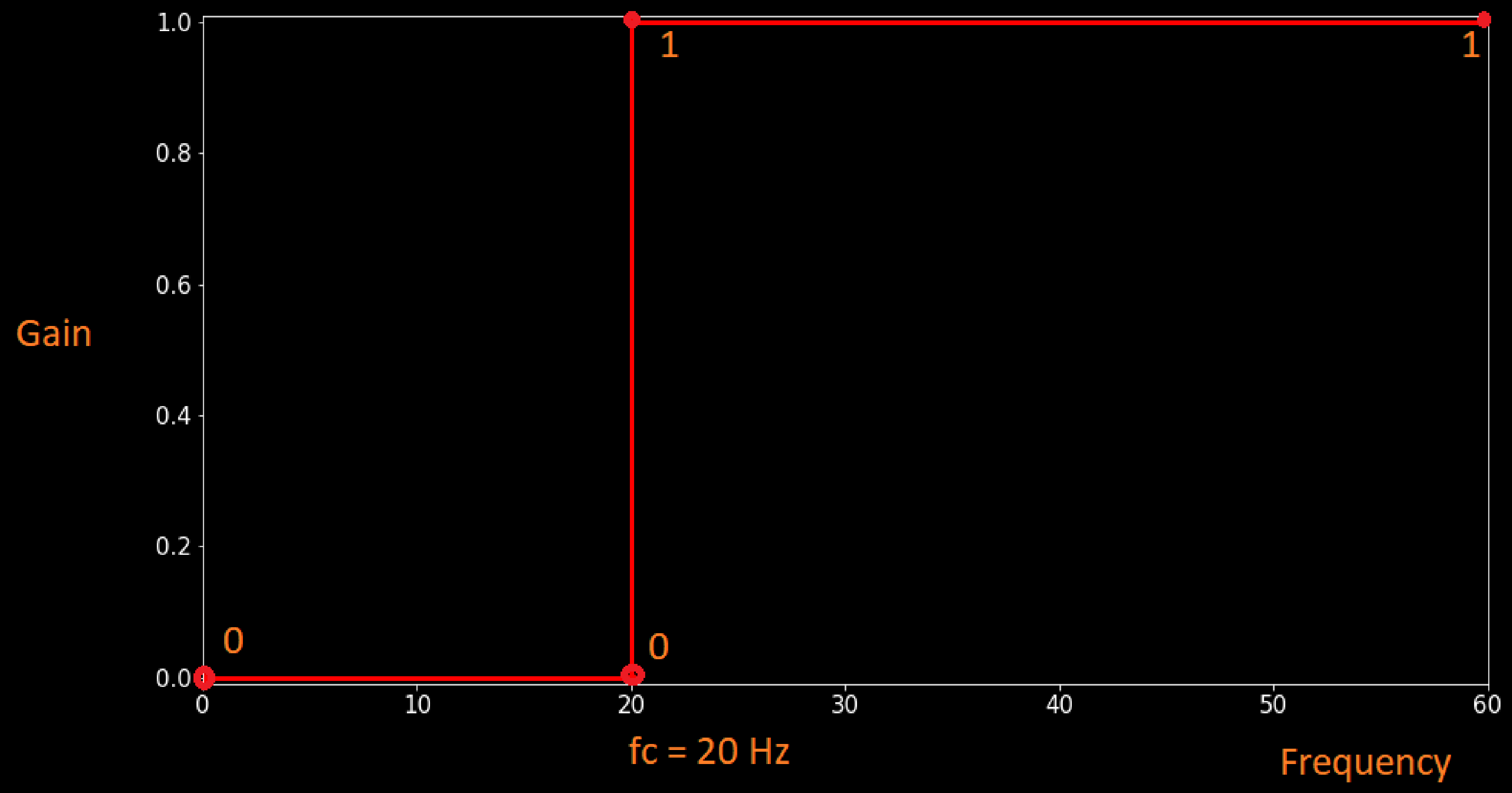
Low pass filter



Low pass filter requires four points to define the shape

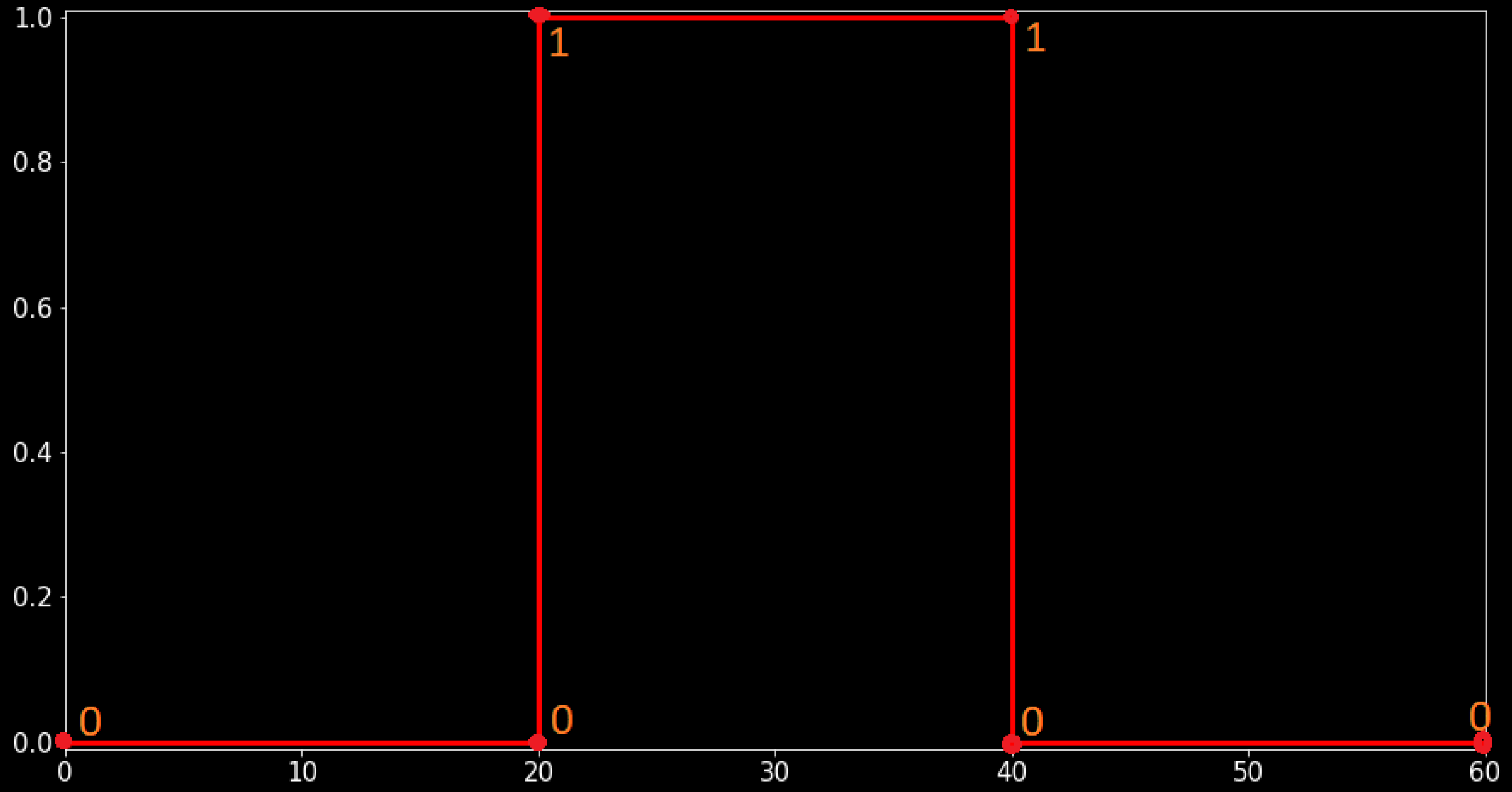
Points = [1, 1, 0, 0]

High pass filter



Points to define the shape = [0, 0, 1, 1]

Band pass filter



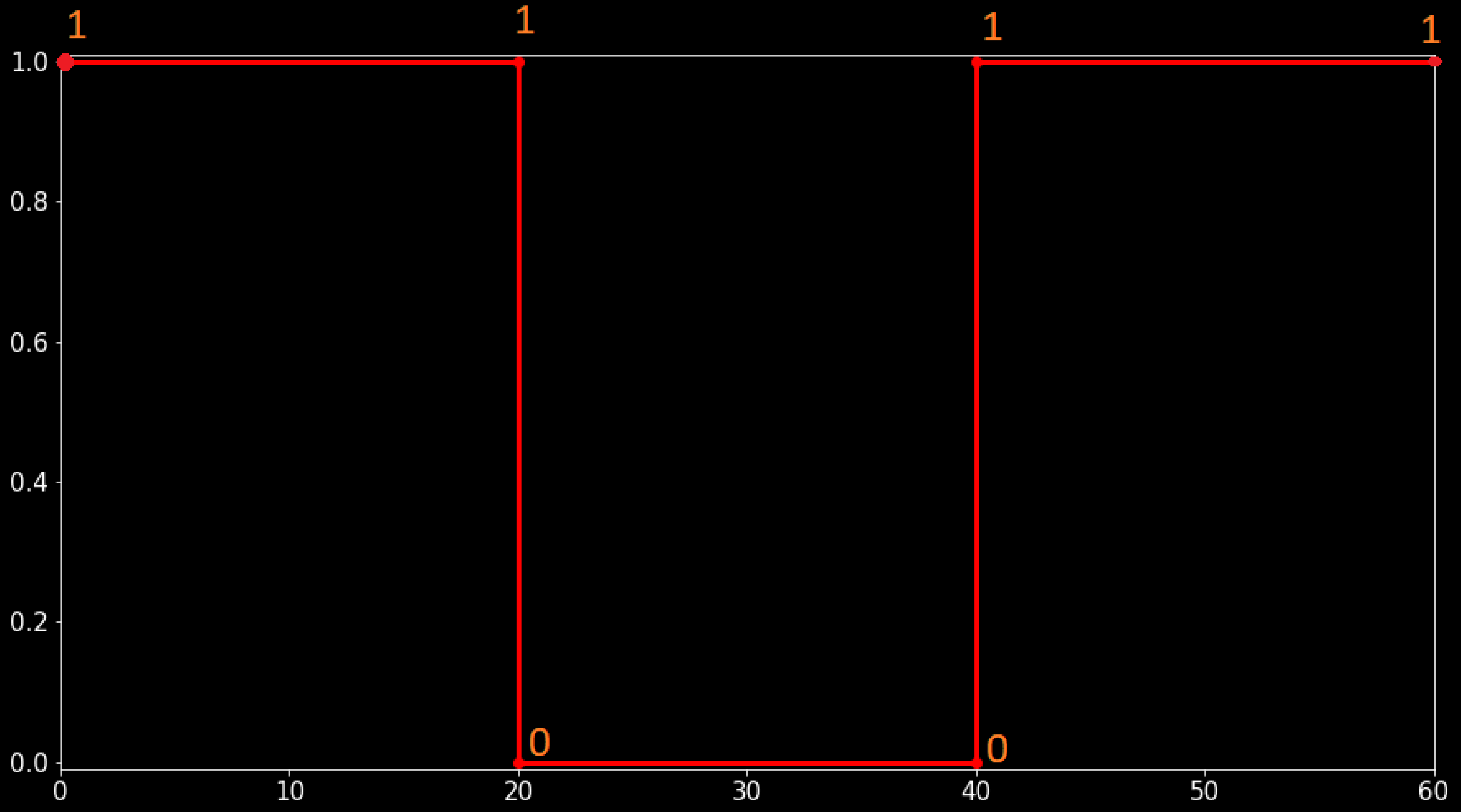
Low cut-off = 20 Hz

High cut-off = 40Hz

Band pass filter requires six points to define the shape.

Points = [0, 0, 1, 1, 0, 0]

Band stop filter

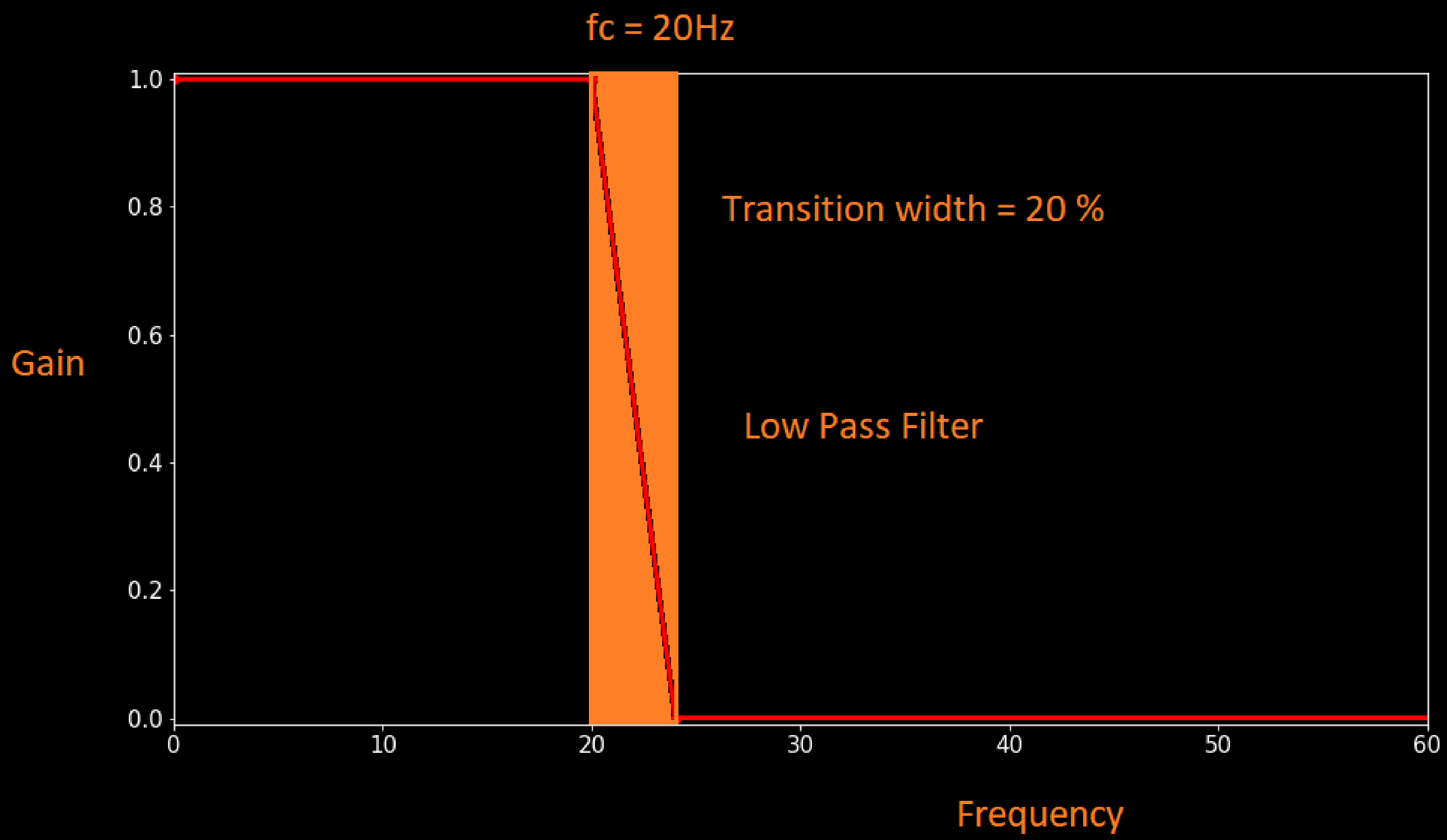


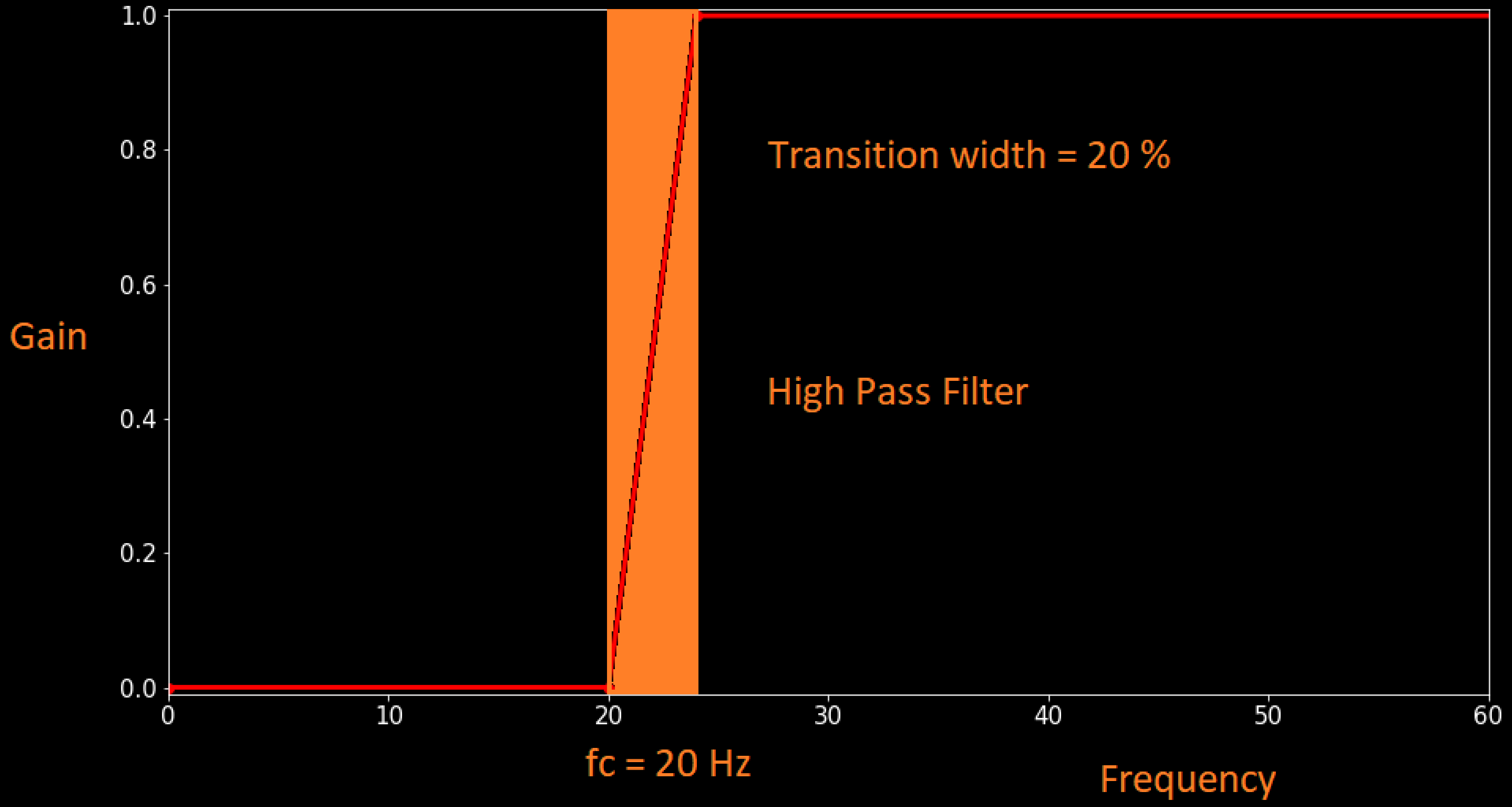
Low cut-off = 20 Hz

High cut-off = 40 Hz

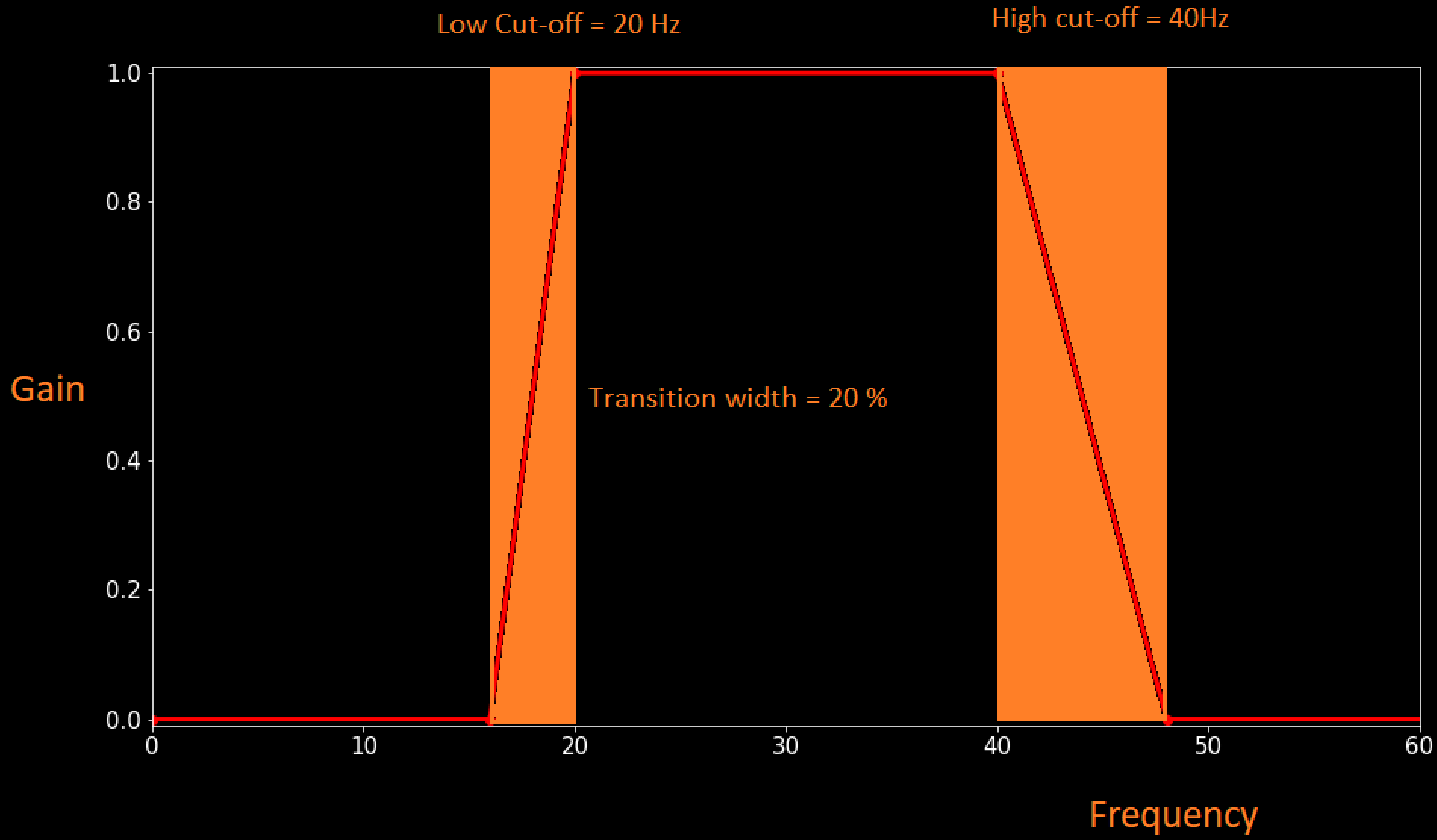
Points = [1, 1, 0, 0, 1, 1]

Step 1 and 2: Define Frequency domain shapes With cutt-off(s) and Transition Widths





Gain
1.0
0.8
0.6
0.4
0.2
0.0
0
10
20
30
40
50
60
Transition width = 20 %
High Pass Filter
fc = 20 Hz
Frequency



Band Pass Filter

Example for BandPass Filter

srate = 1024

Nyquist = srate / 2

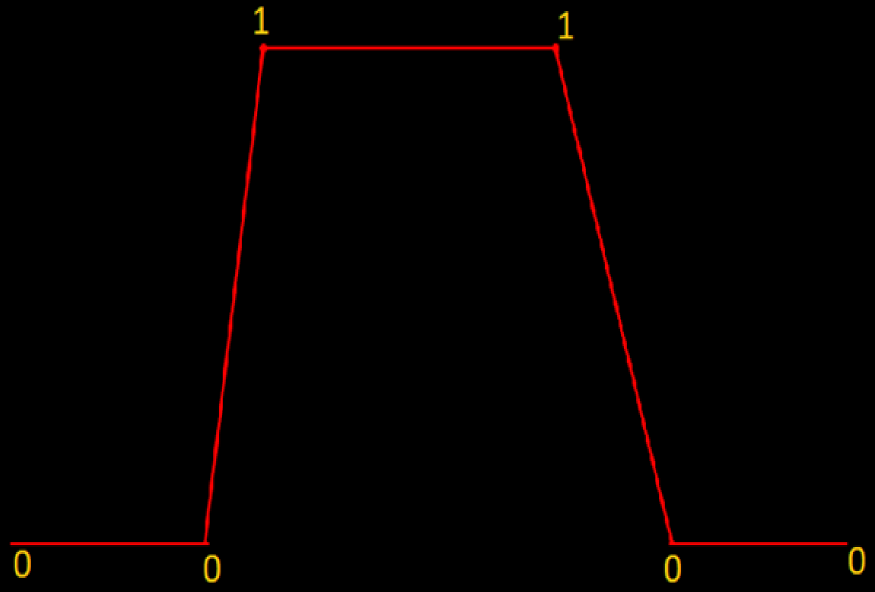
cuttoffs = [30 50]

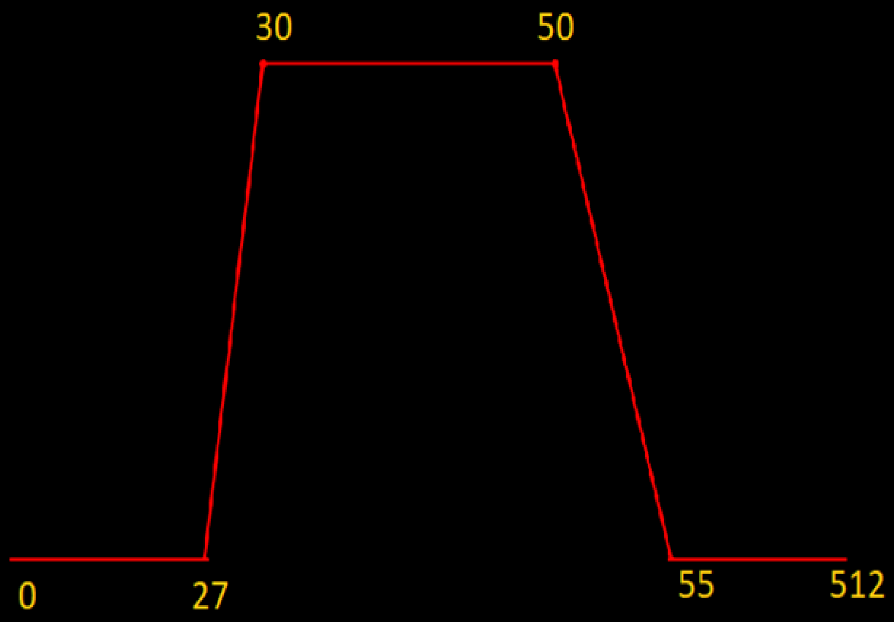
transw = 0.1

desired shape = [0 0 1 1 0 0]

frequencies = [0   cuttoff [0] - cuttoff [0] * transw   cuttoff[0]   cuttoff [1]   cuttoff [1] + cuttoff [1] * transw   Nyquist]

frequencies = [0   27   30   50   55   512]





FIR Low Pass Filter

Desired shape = [1   1   0   0]

Cuttoff = 20 hZ

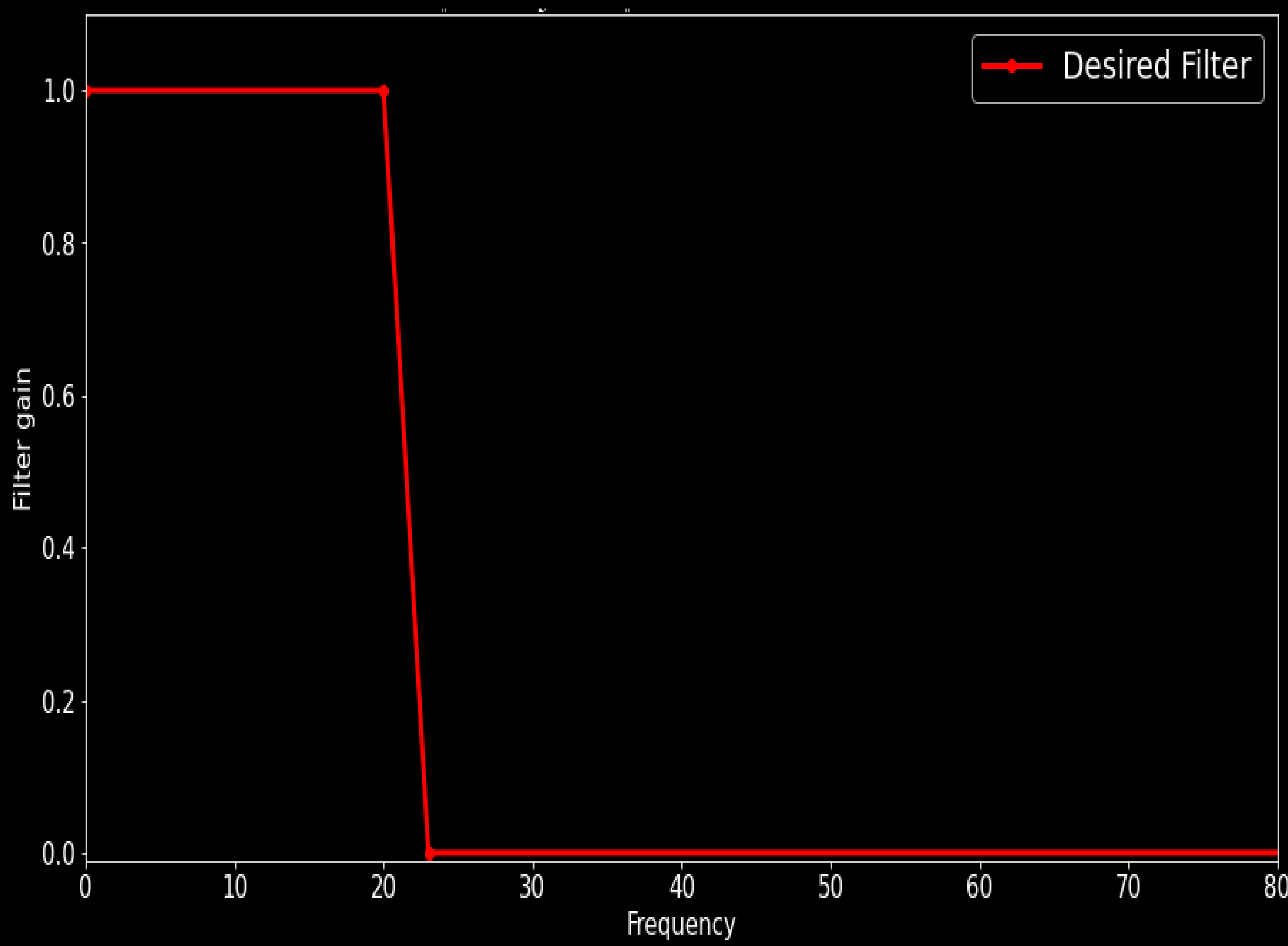
srate = 1024

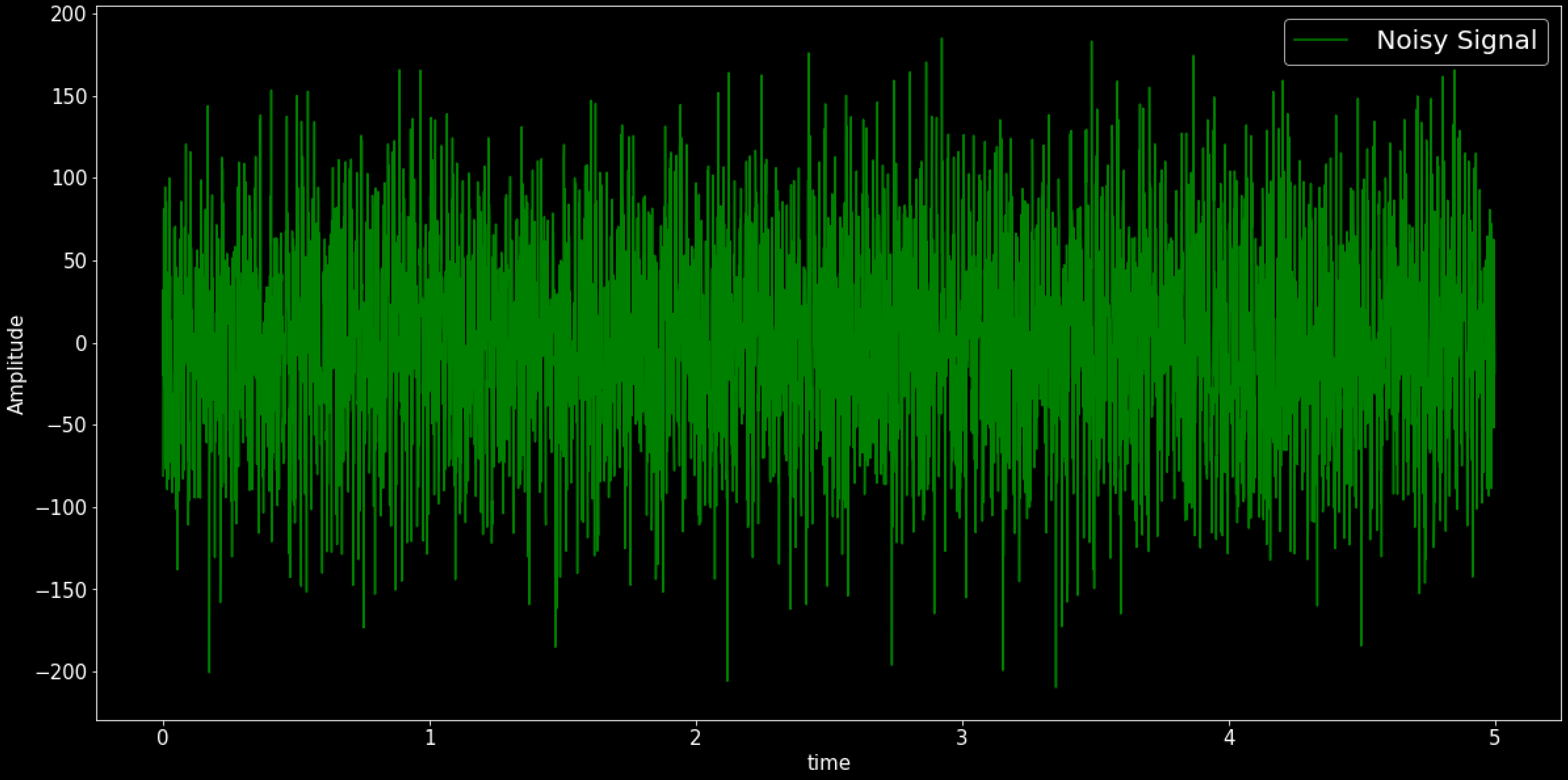
transw = 0.15

Nyquist = srate / 2

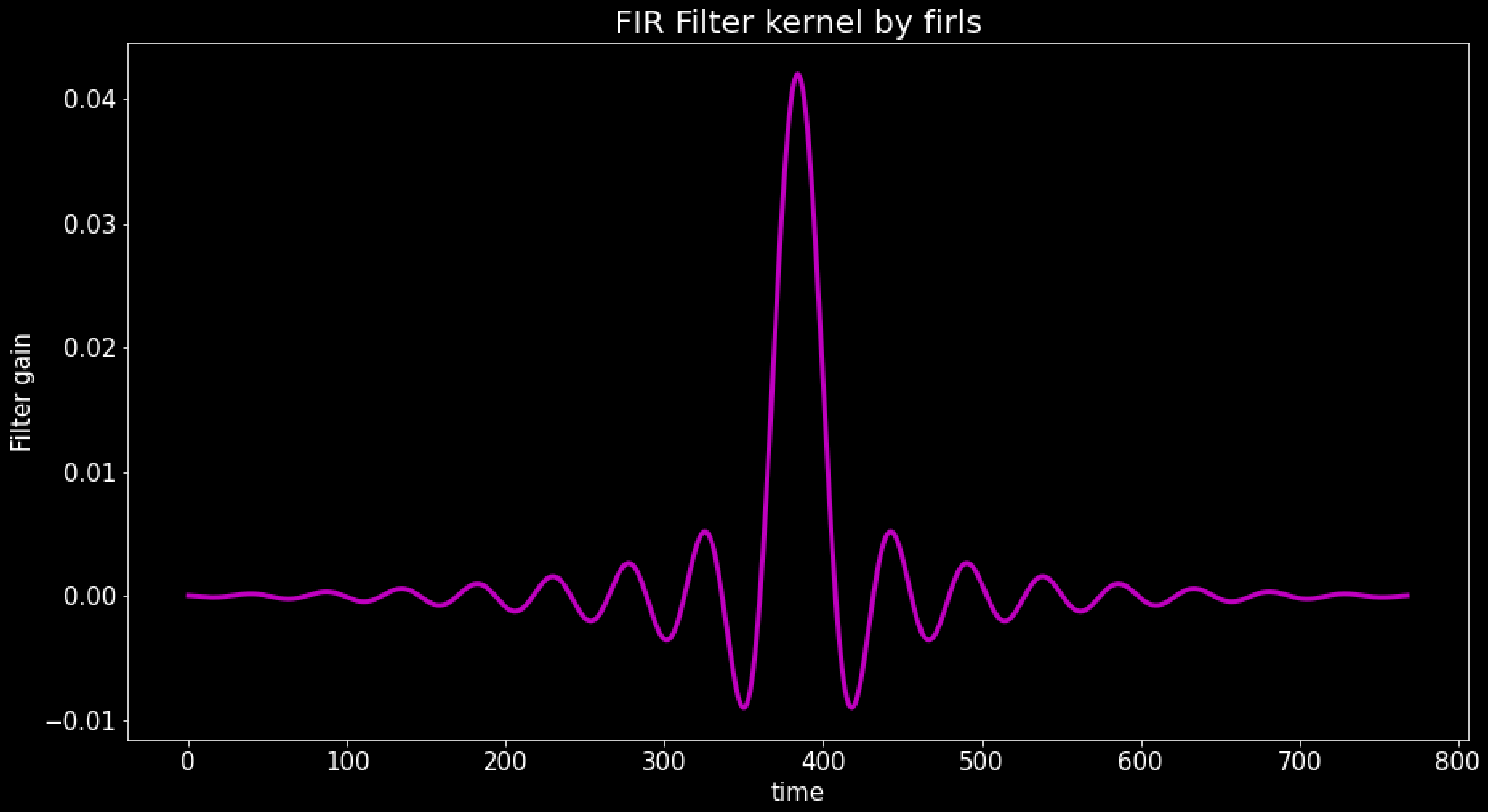
Frequencies = [0 cuttoff   cuttoff + cuttoff * transw + Nyquist]

Frequencies = [0   20   23   512]

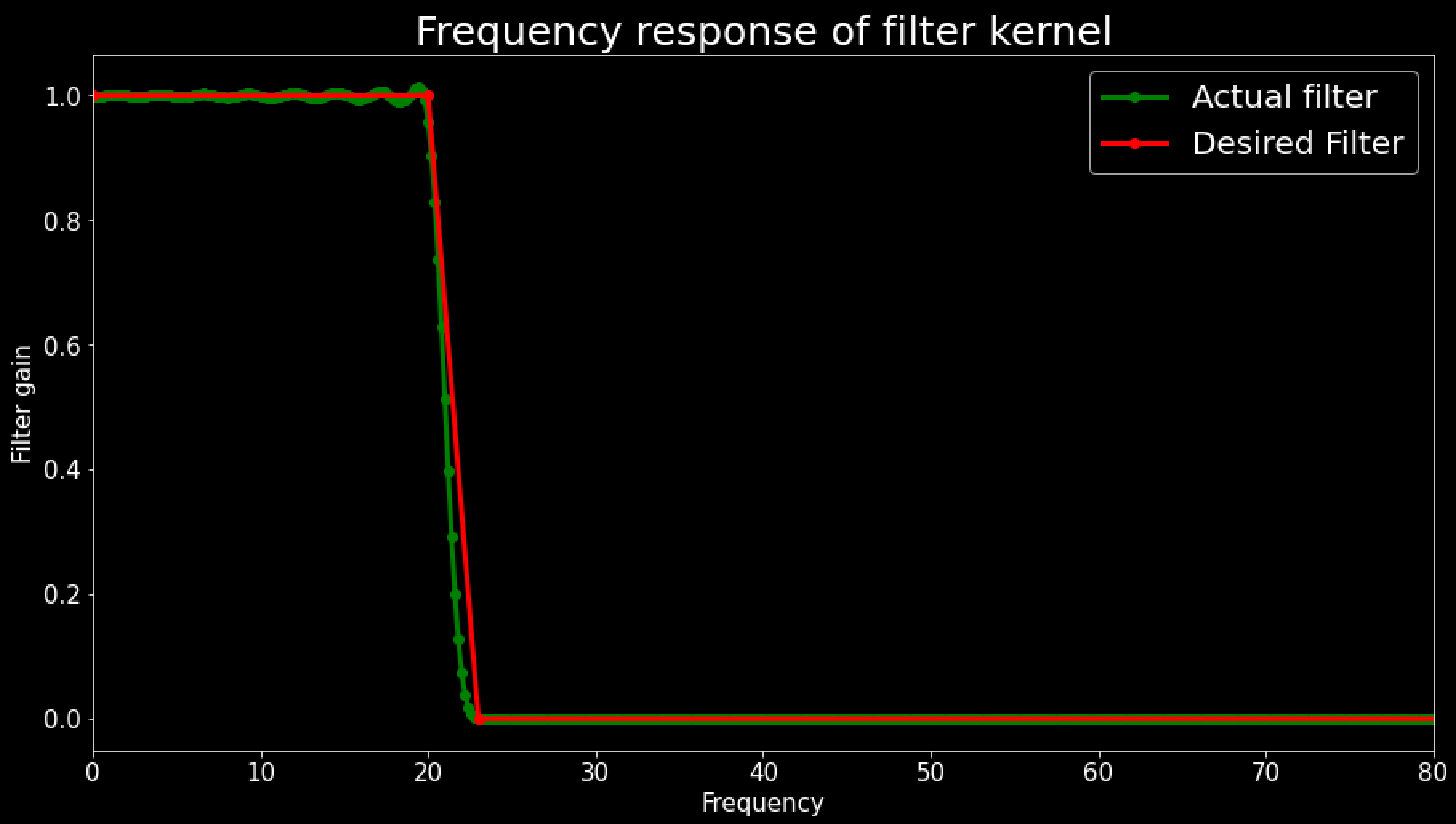




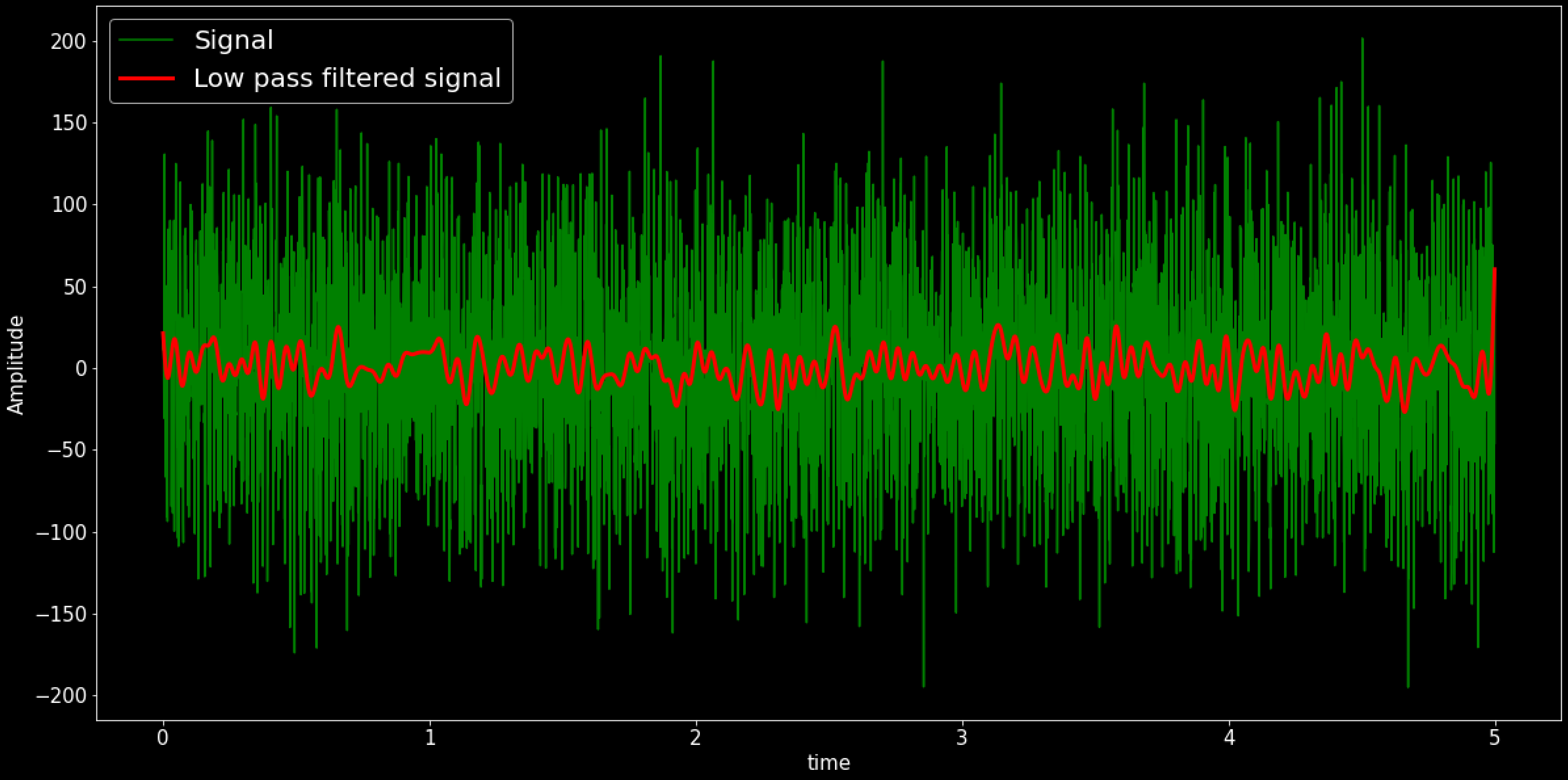
Noisy Signal
200
150
100
50
0
−50
−100
−150
−200
Amplitude
0
1
2
3
4
5
time



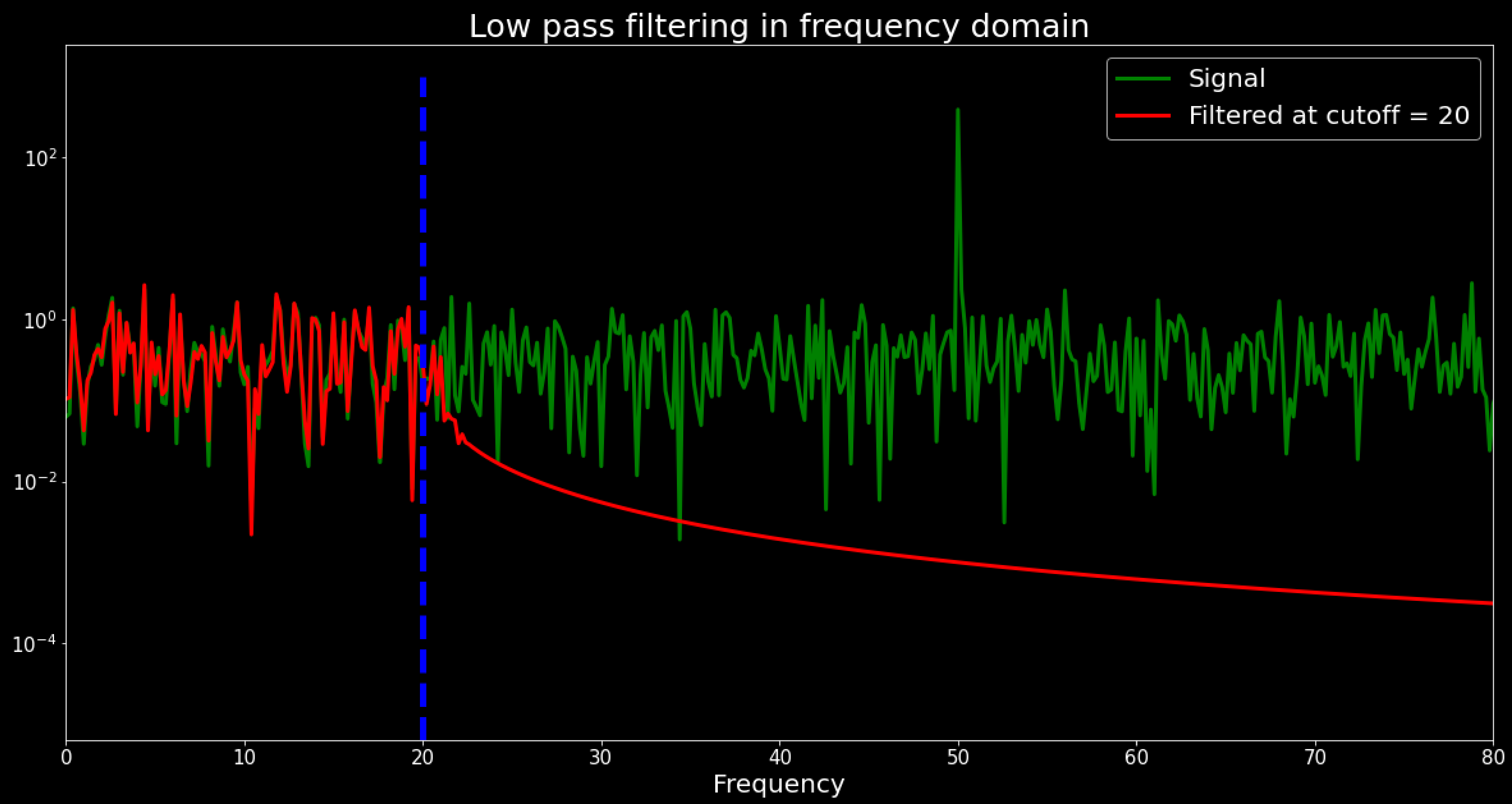
FIR Filter kernel by firls
Filter gain
0.04
0.03
0.02
0.01
0.00
−0.01
0
100
200
300
400
500
600
700
800
time



Frequency response of filter kernel
Actual filter
Desired Filter
Filter gain
Frequency
0.0
0.2
0.4
0.6
0.8
1.0
0
10
20
30
40
50
60
70
80



Signal
Low pass filtered signal
Amplitude
time
200
150
100
50
0
−50
−100
−150
−200
0
1
2
3
4
5



Low pass filtering in frequency domain
Signal
Filtered at cutoff = 20
10^2
10^0
10^{-2}
10^{-4}
0
10
20
30
40
50
60
70
80
Frequency

FIR High Pass Filter

Desired shape = [0   0   1   1]

Cuttoff = 20 hZ

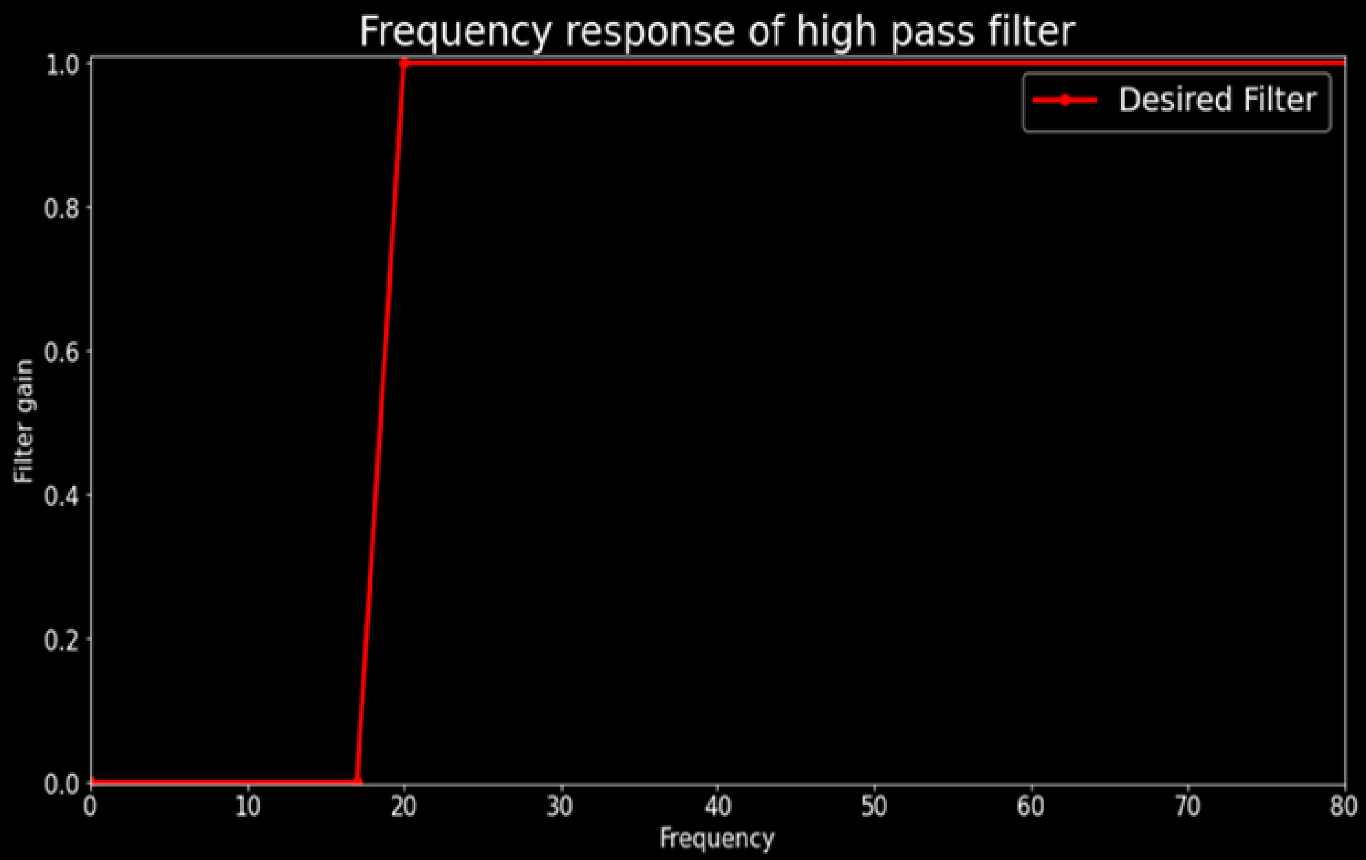
srate = 1024

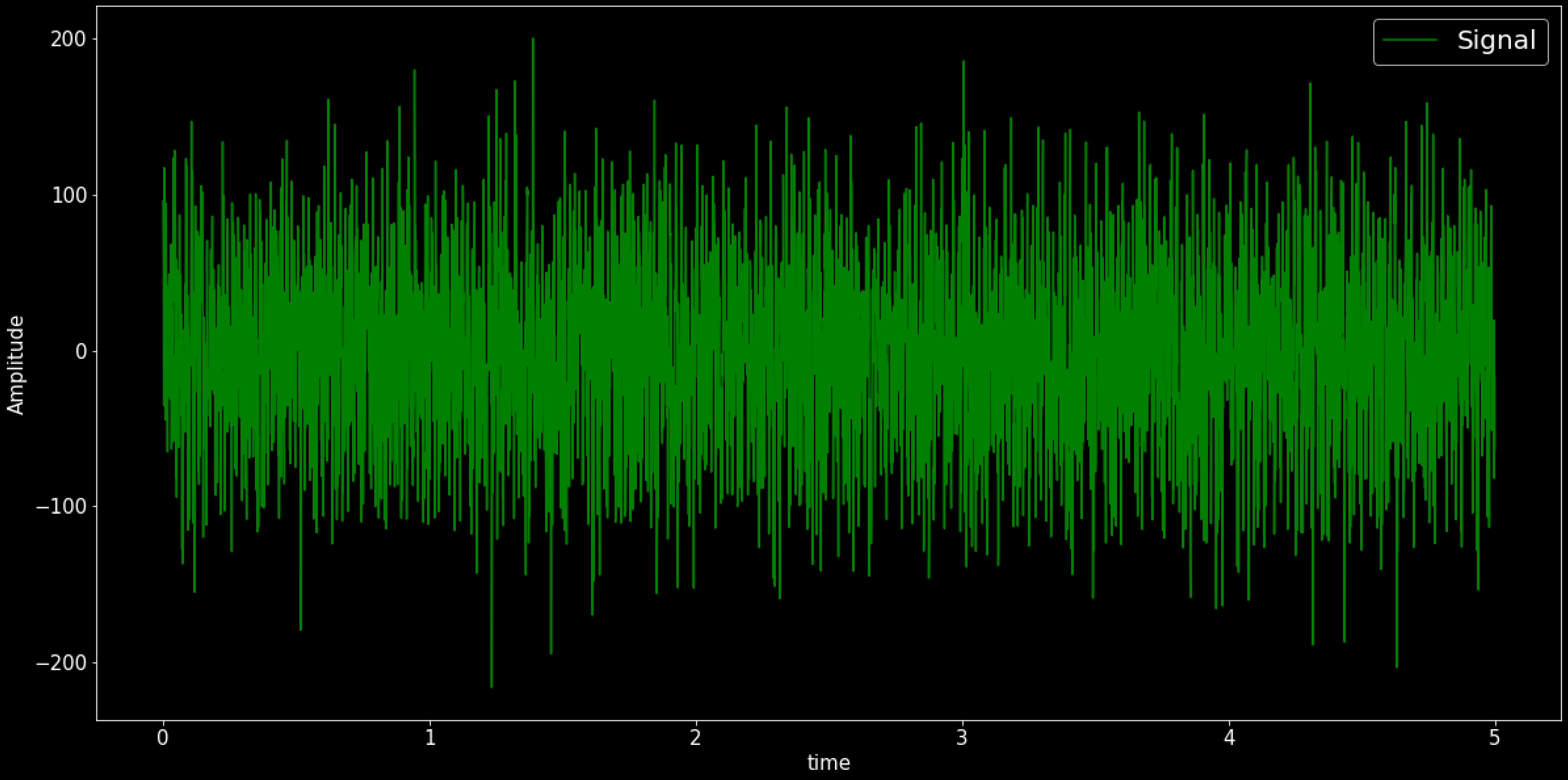
transw = 0.15

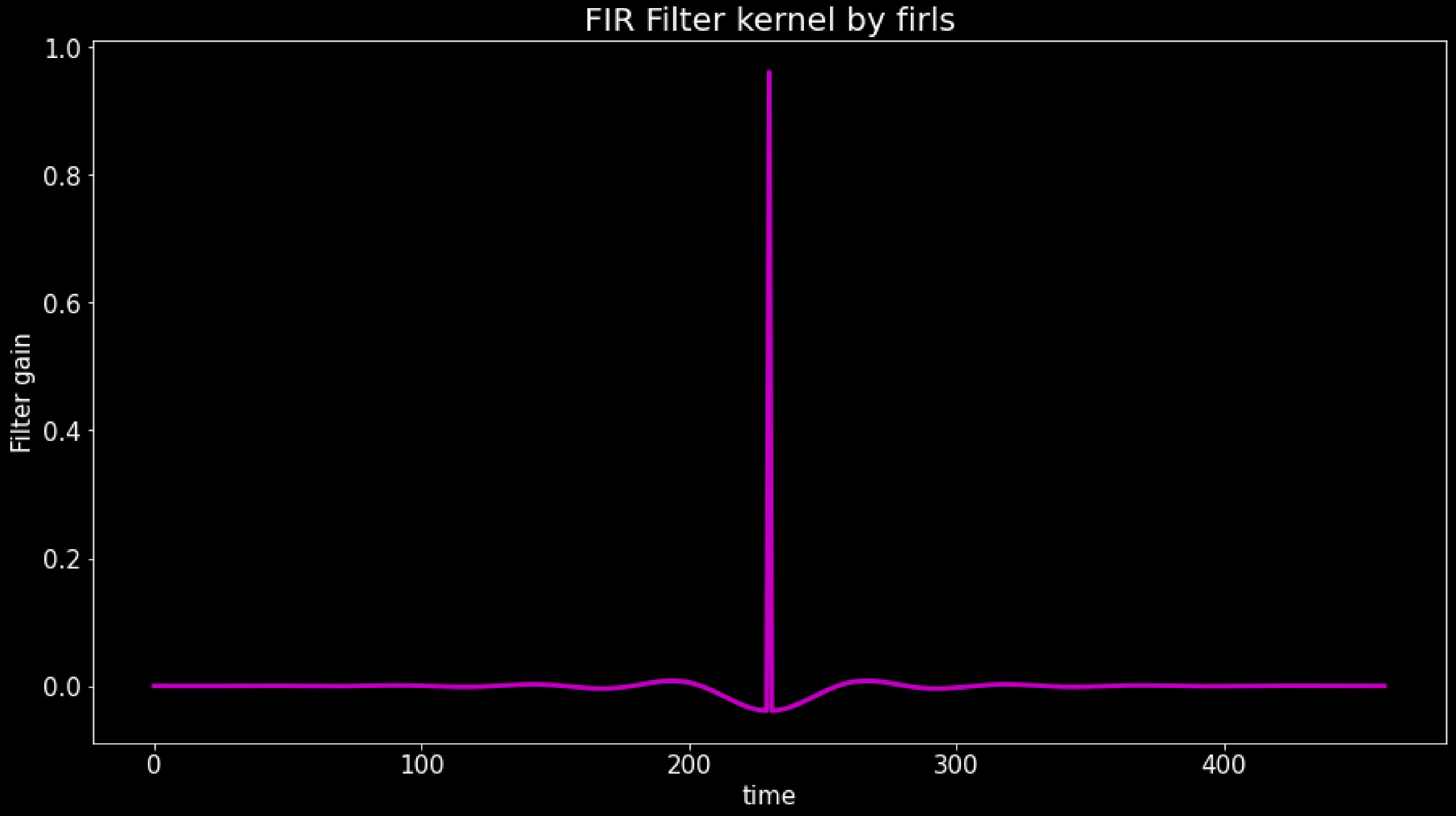
Nyquist = srate / 2

Frequencies = [0 cuttoff - cuttoff * transw   cuttoff   Nyquist]

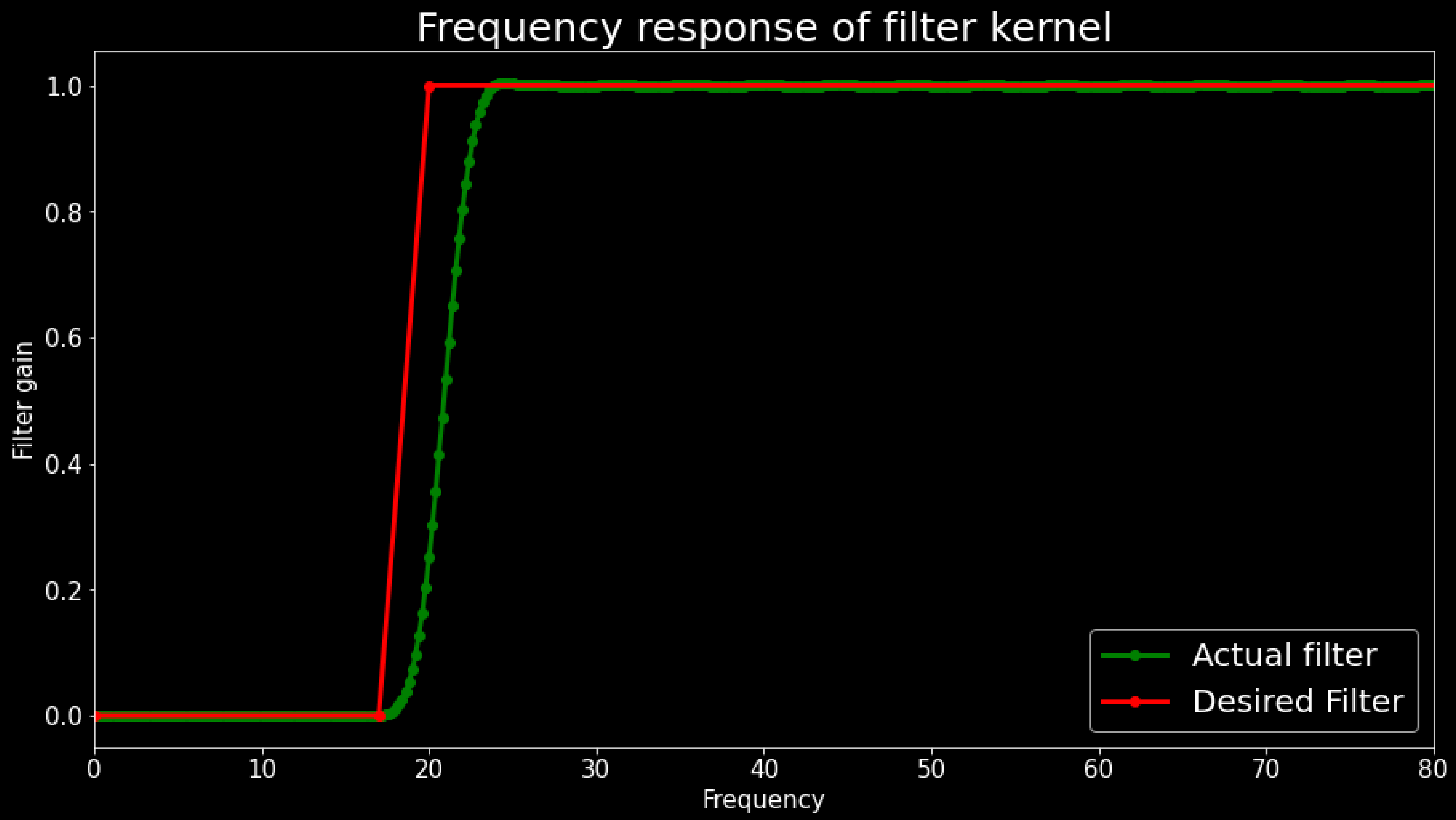
Frequencies = [0   17   20   512]

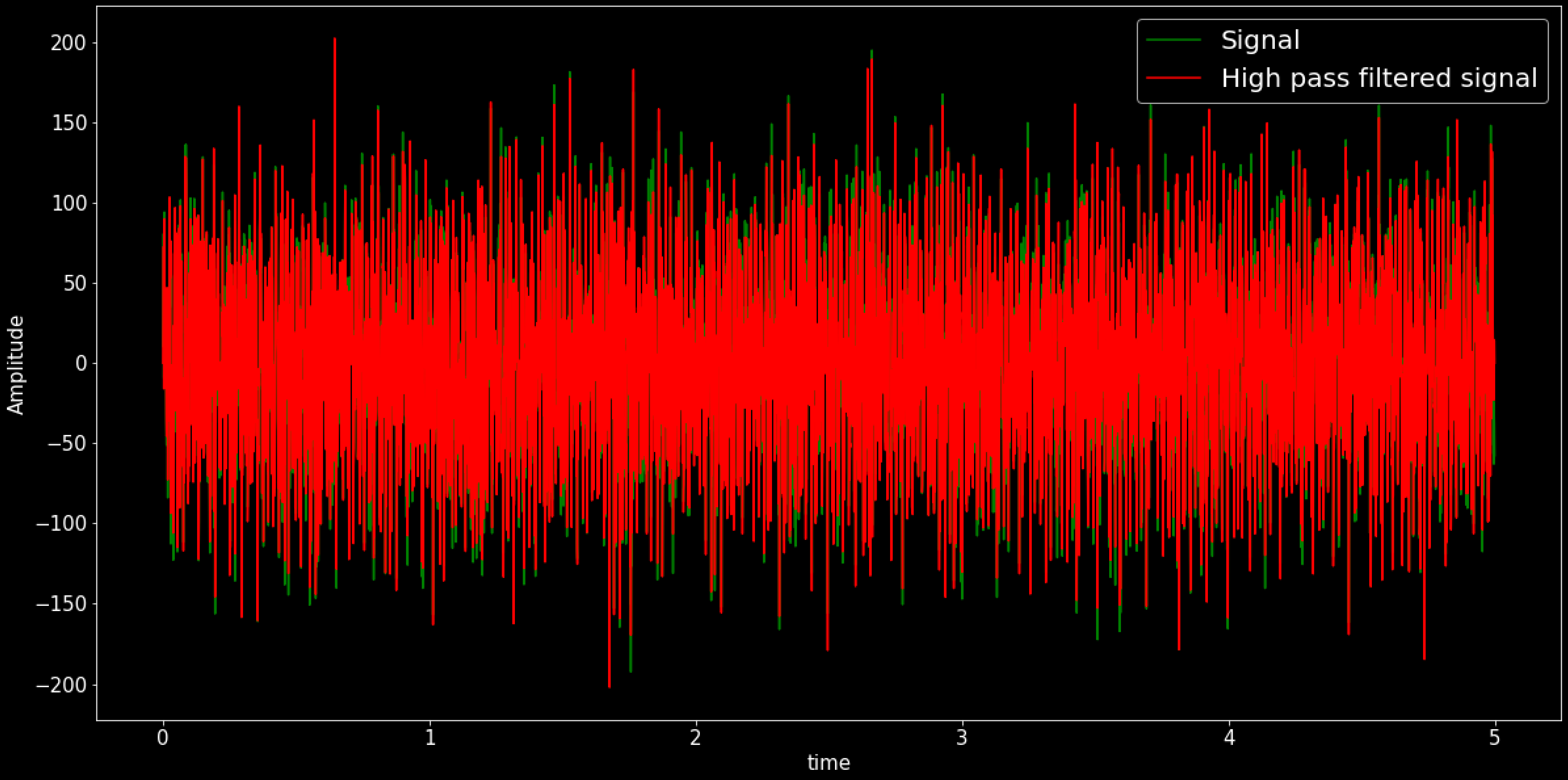



Signal
Amplitude
time
200
100
0
−100
−200
0
1
2
3
4
5

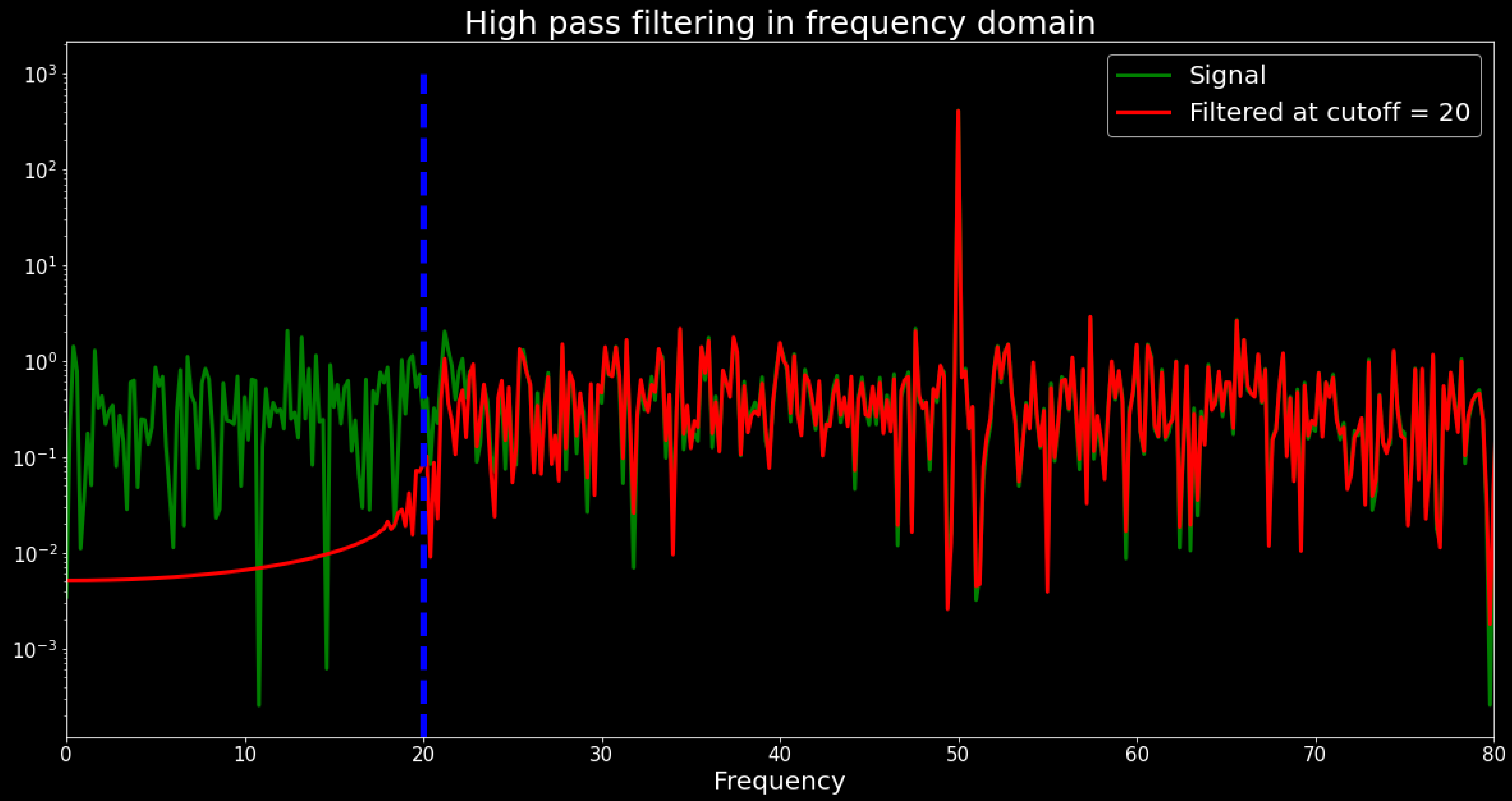


FIR Filter kernel by firls
Filter gain
time
1.0
0.8
0.6
0.4
0.2
0.0
0
100
200
300
400


Frequency response of filter kernel
1.0
0.8
0.6
0.4
0.2
0.0
Filter gain
0
10
20
30
40
50
60
70
80
Frequency
Actual filter
Desired Filter



Signal
High pass filtered signal
Amplitude
time
200
150
100
50
0
−50
−100
−150
−200
0
1
2
3
4
5


High pass filtering in frequency domain
Signal
Filtered at cutoff = 20
Frequency
10^3
10^2
10^1
10^0
10^{-1}
10^{-2}
10^{-3}
0
10
20
30
40
50
60
70
80

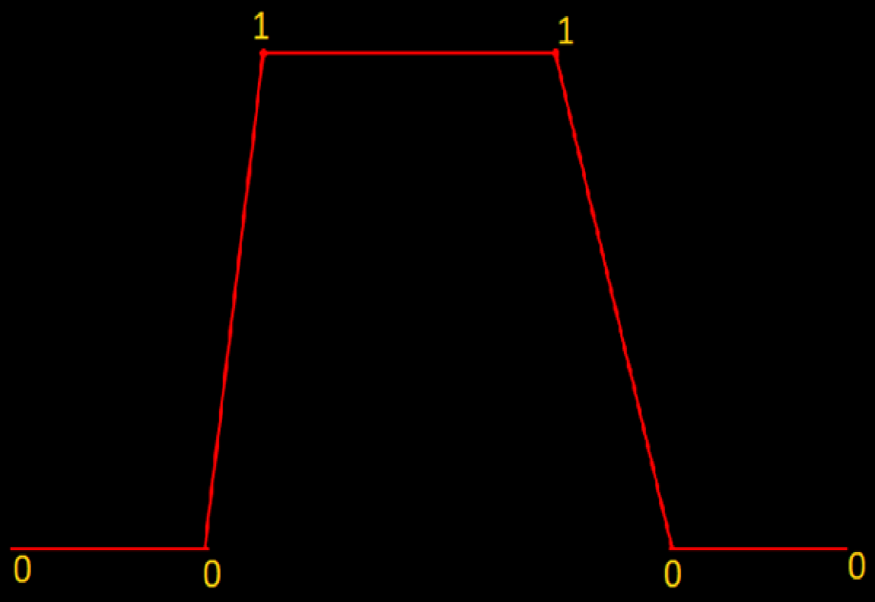
Example for BandPass Filter

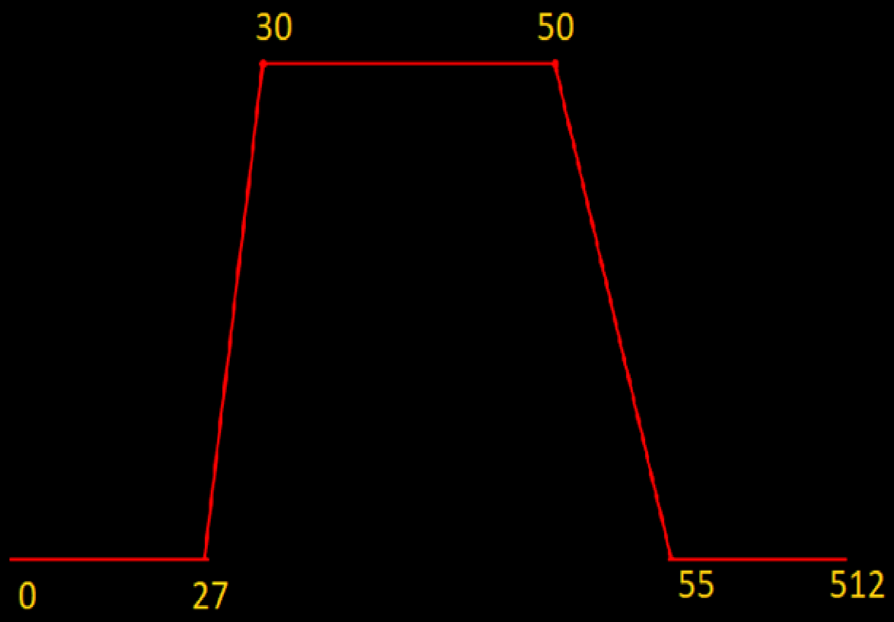
srate = 1024
Nyquist = srate / 2
cuttoffs = [30 50]
transw = 0.1

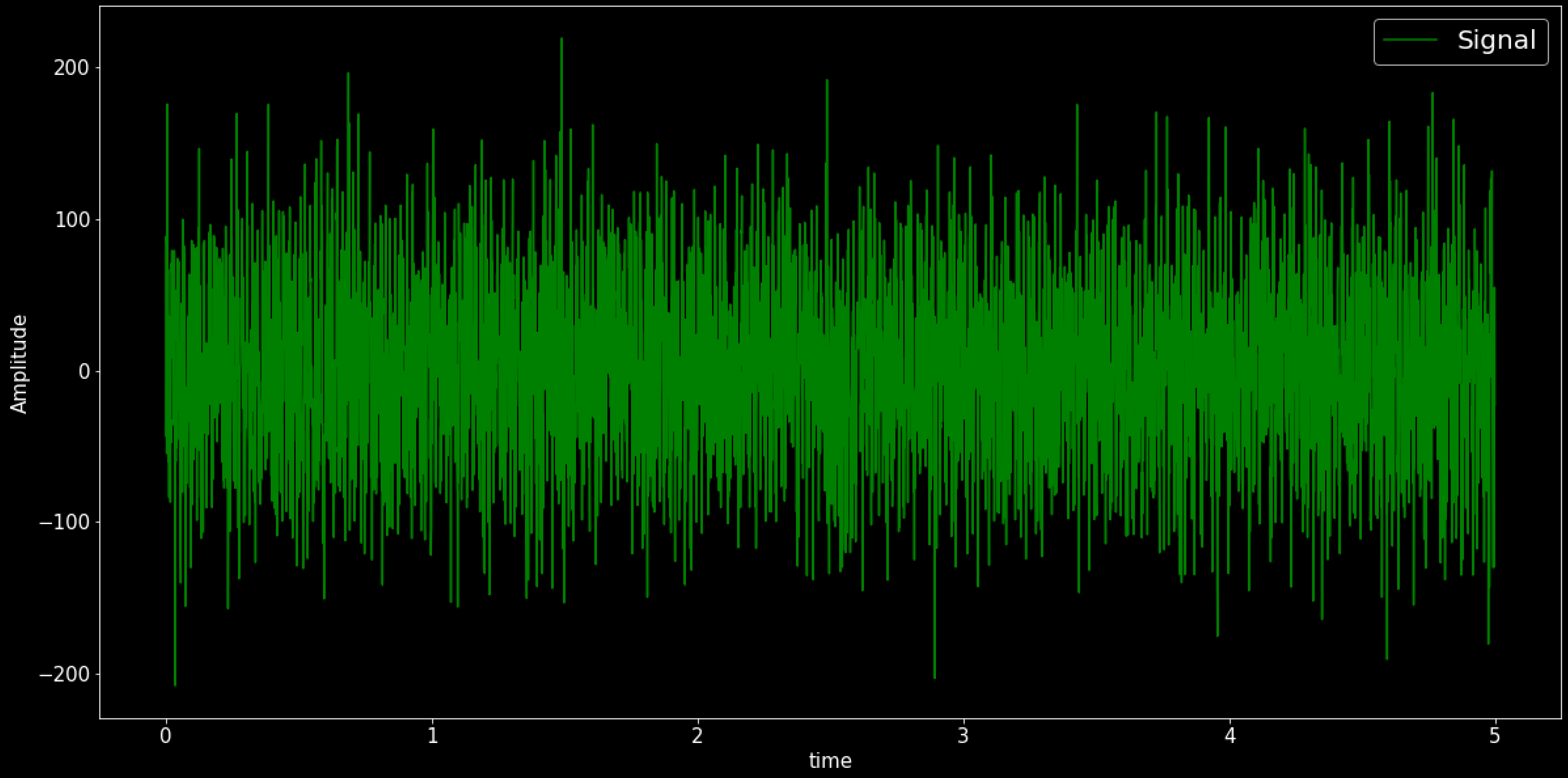
desired shape = [0 0 1 1 0 0]

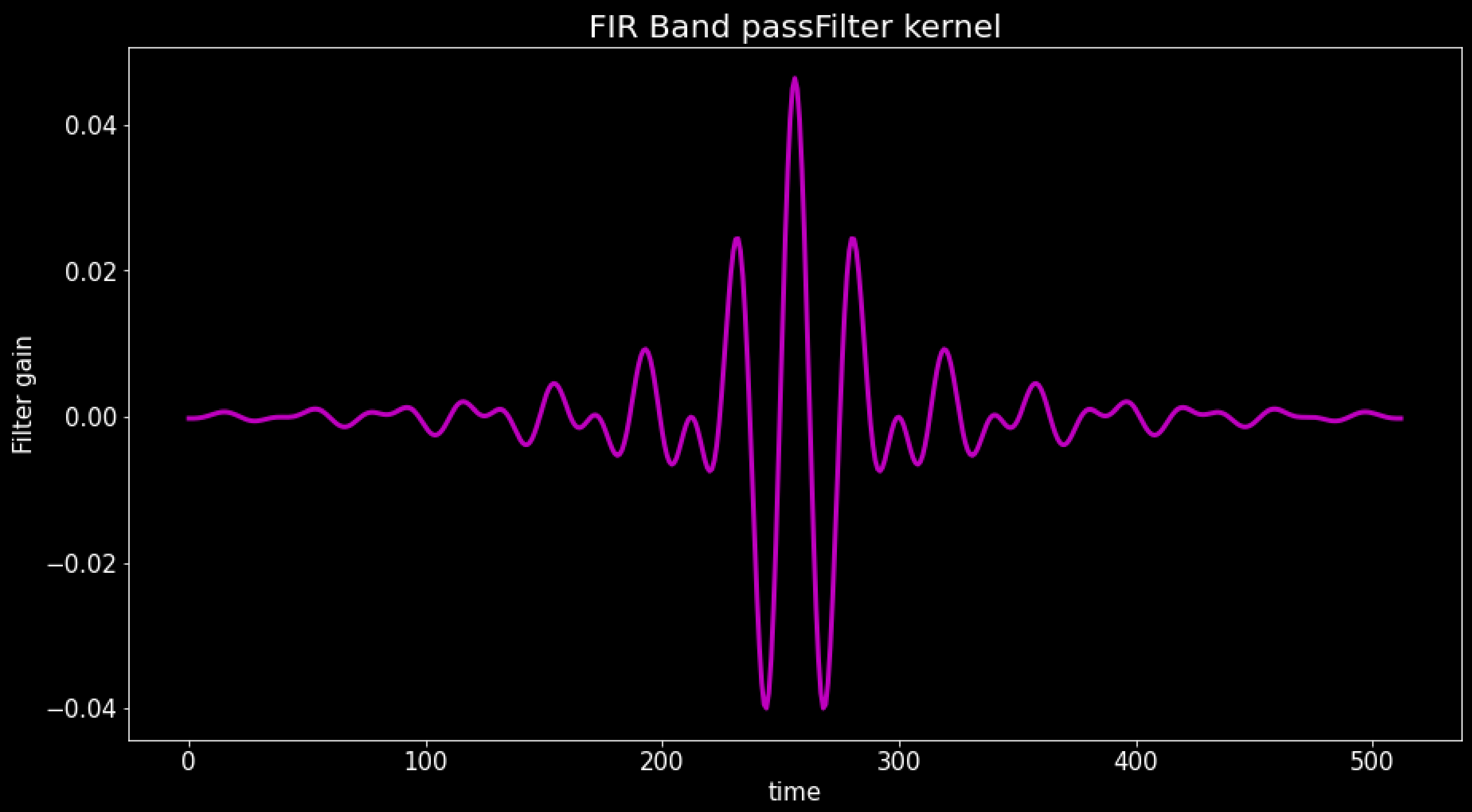
frequencies = [0   cuttoff [0] - cuttoff [0] * transw   cuttoff[0]   cuttoff [1]   cuttoff [1] + cuttoff [1] * transw   Nyquist]

frequencies = [0   27   30   50   55   512]

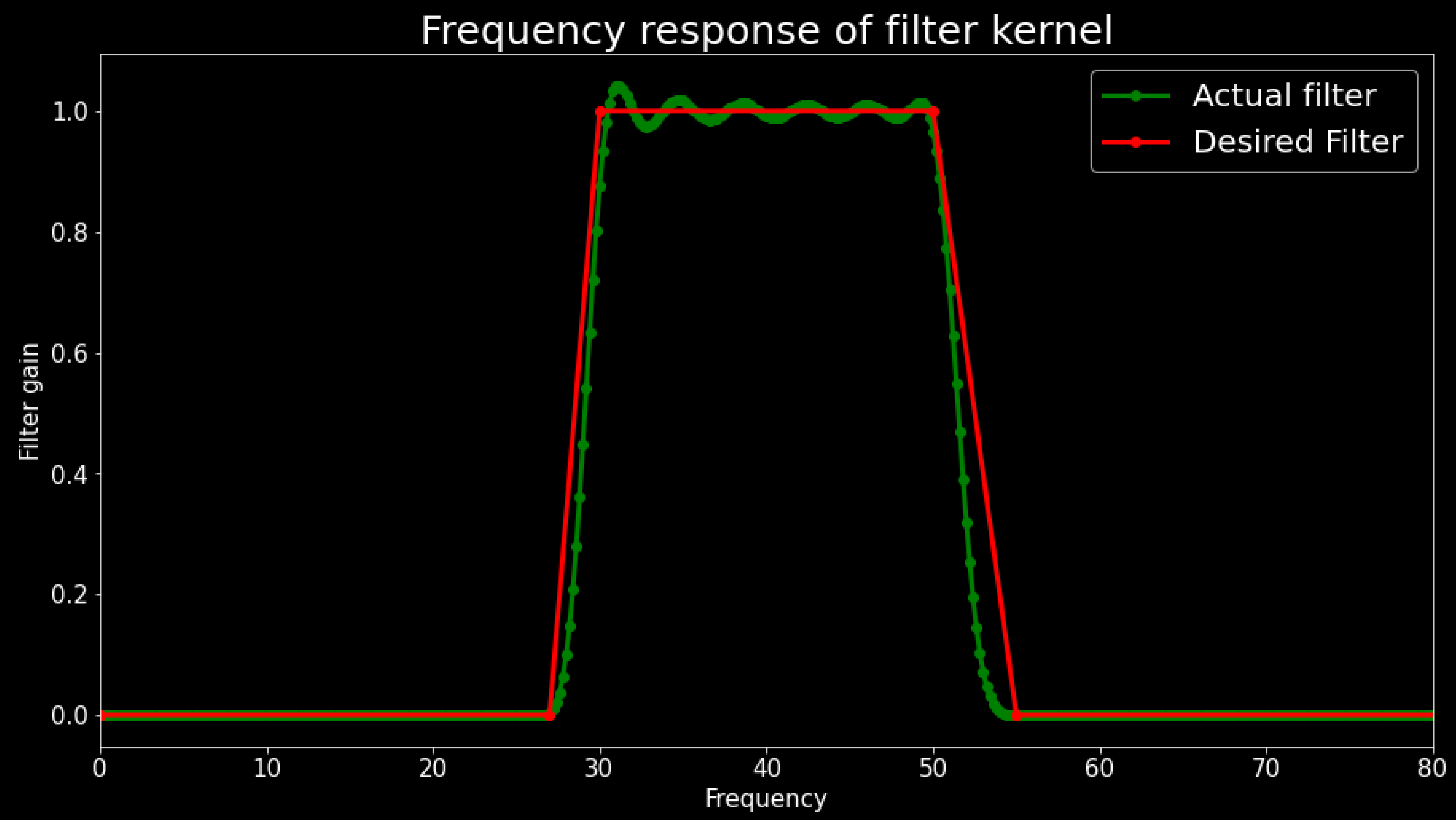


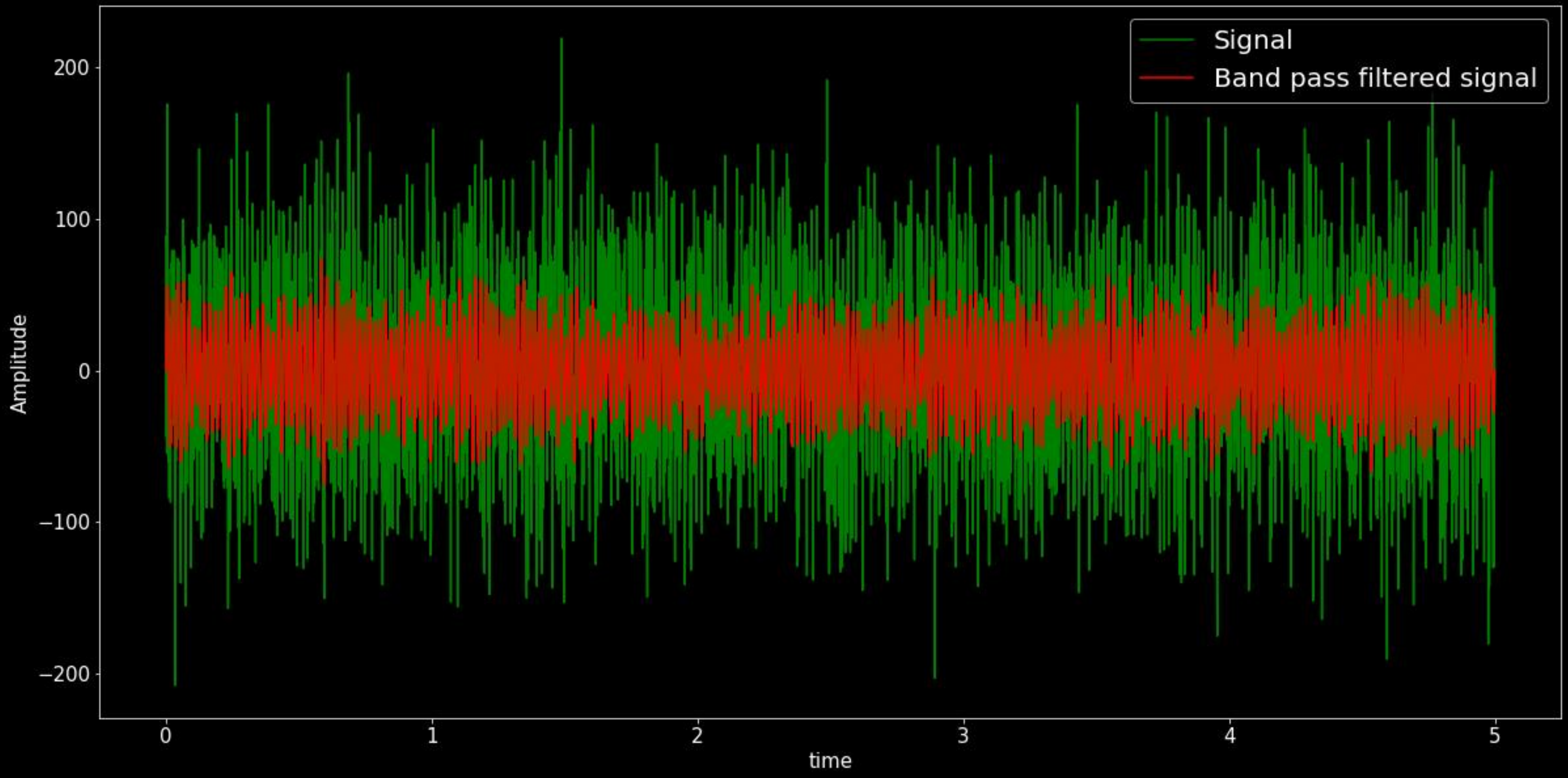


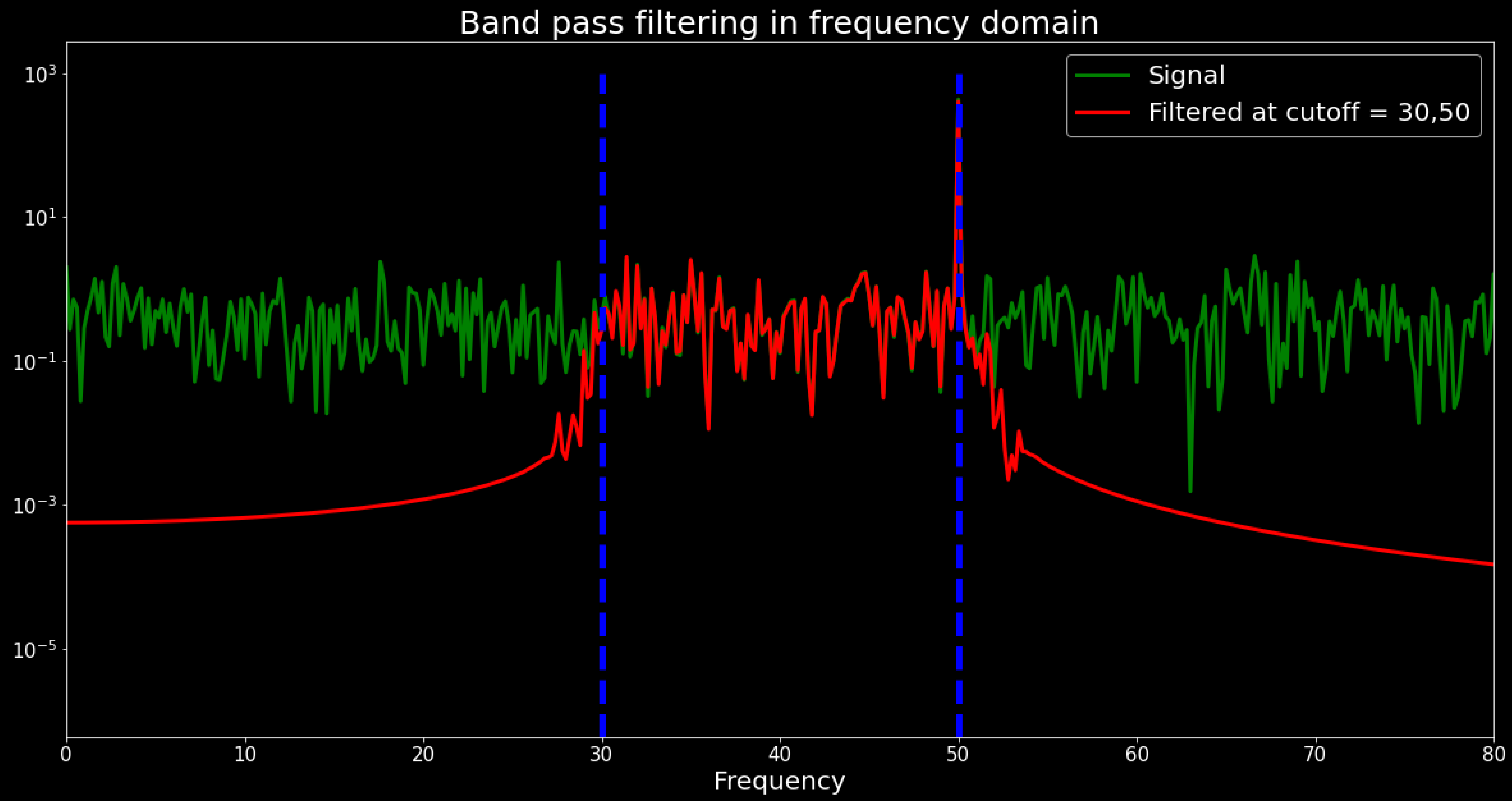

Signal
Amplitude
time
200
100
0
−100
−200
0
1
2
3
4
5



FIR Band passFilter kernel
0.04
0.02
0.00
−0.02
−0.04
Filter gain
0
100
200
300
400
500
time


Frequency response of filter kernel
Actual filter
Desired Filter
Filter gain
Frequency
1.0
0.8
0.6
0.4
0.2
0.0
0
10
20
30
40
50
60
70
80


Signal
Band pass filtered signal
200
100
0
−100
−200
Amplitude
0
1
2
3
4
5
time



Band pass filtering in frequency domain
Signal
Filtered at cutoff = 30,50
10^3
10^1
10^{-1}
10^{-3}
10^{-5}
0
10
20
30
40
50
60
70
80
Frequency

Tasks for Students

- Design Low pass, High pass and Band pass filters using Window method.
- Design Band stop filter using both Window and Least Square method.