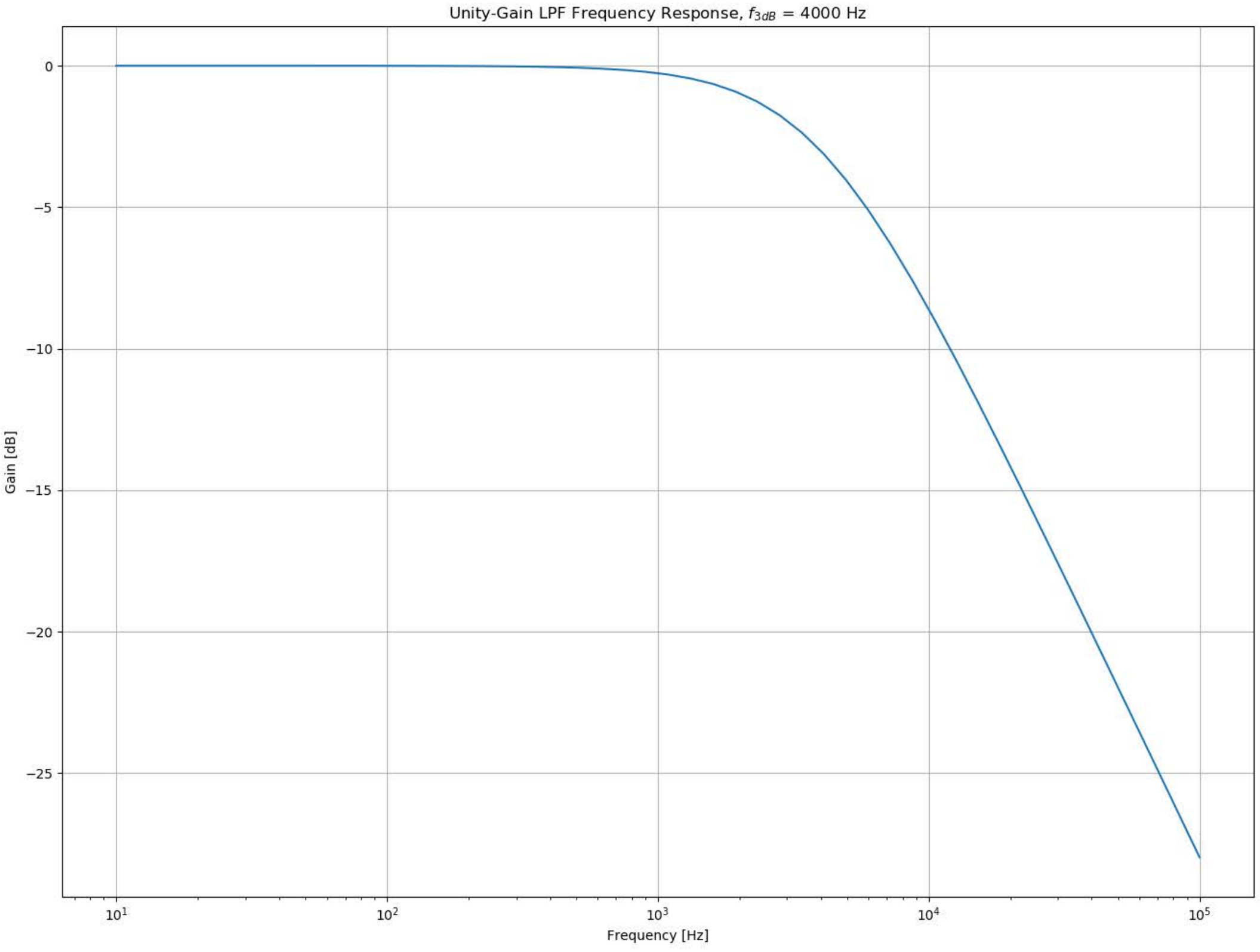
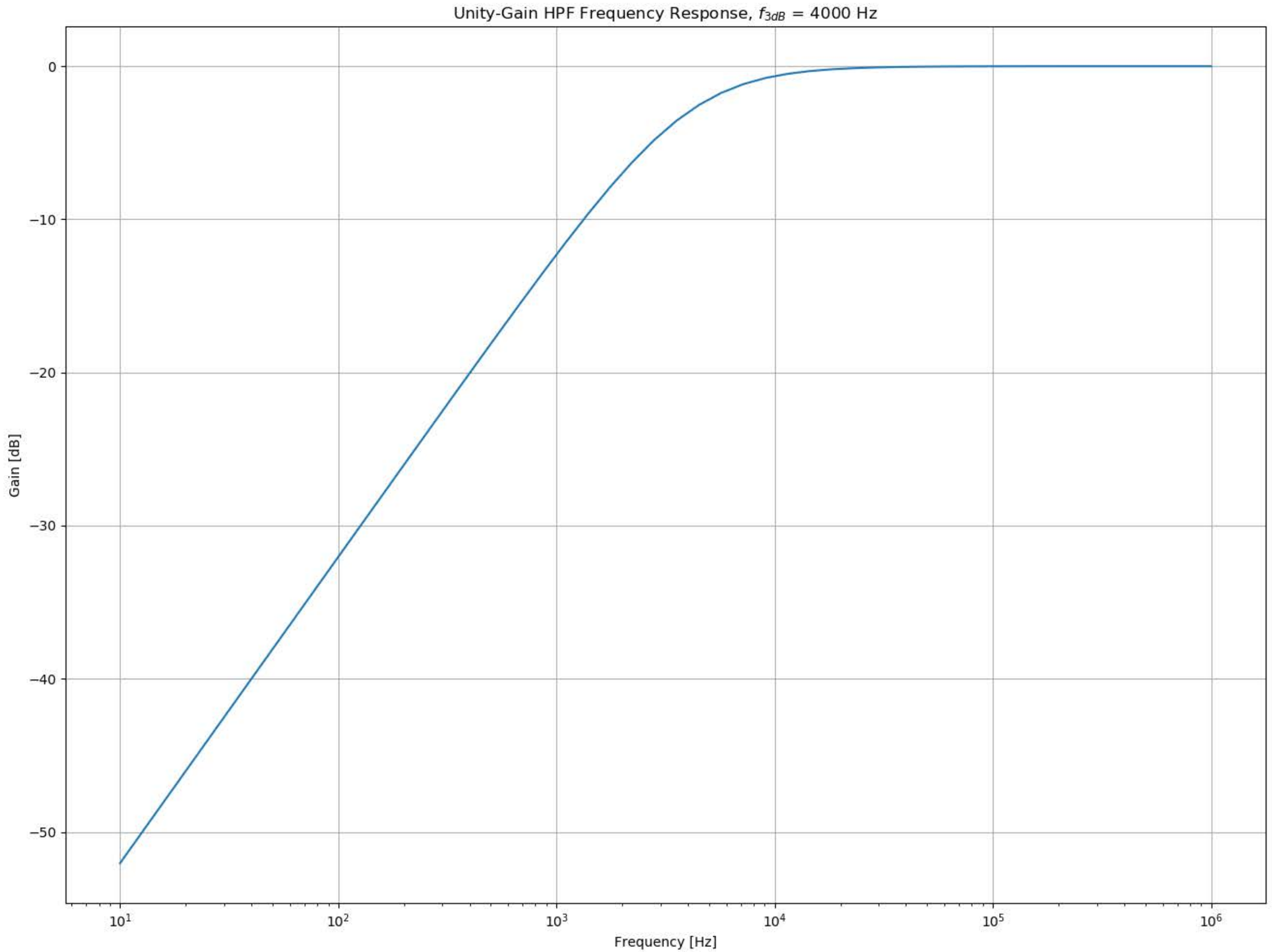
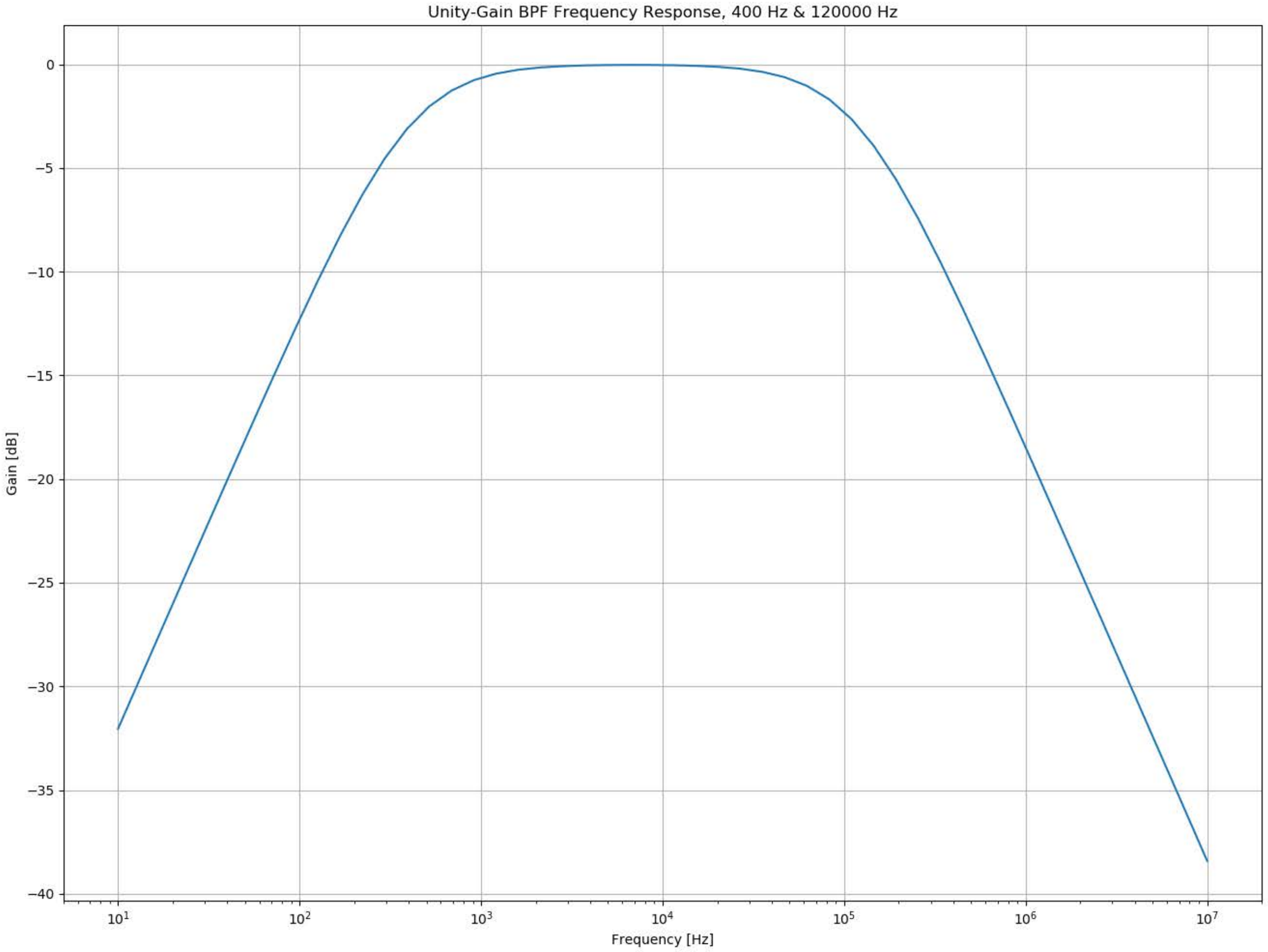
```
# -*- coding: utf-8 -*-
"""
Created on Mon Jun 25 21:30:32 2018

@author: Zengweijie (Fred) Chen

Frequency Response Plot
Low-Pass Filter
"""

from pylab import *

fc = 4000
K = 1
f = logspace(1,5) # frequencies from 10^1 to 10^5

def H(w, K, wc):

   H = K / (1.0 + 1j * w / wc)
   return H

H_log = 20*log10(abs(H(2*pi*f, K, 2*pi*fc)))

# Plotting Configuration
figure(num = 3, figsize = (16, 12), dpi = 100, facecolor='w', edgecolor='k')
plot(f, H_log)
xscale('log')
title(r'Unity-Gain LPF Frequency Response, $f_{3dB}$ = %s Hz' %fc)
xlabel('Frequency [Hz]')
ylabel('Gain [dB]')
grid(True)

savefig('LPF.png')
```


Unity-Gain LPF Frequency Response, f_{3dB} = 4000 Hz
0
−5
−10
−15
−20
−25
Gain [dB]
10^1
10^2
10^3
10^4
10^5
Frequency [Hz]

```
# -*- coding: utf-8 -*-
"""
Created on Mon Jun 25 22:02:29 2018

@author: Zengweijie (Fred) Chen

Frequency Response Plot
High-Pass Filter
"""

from pylab import *

fc = 4000
K = 1
f = logspace(1,6) # frequencies from 10^1 to 10^5

def H(w, K, wc):

   H = K* 1j *w / (wc + 1j*w)
   return H

H_log = 20*log10(abs(H(2*pi*f, K, 2*pi*fc)))

# Plotting Configuration
figure(num = 2, figsize = (16, 12), dpi = 100, facecolor='w', edgecolor='k')
plot(f, H_log)
xscale('log')
title(r'Unity-Gain HPF Frequency Response, $f_{3dB}$ = %s Hz' %fc)
xlabel('Frequency [Hz]')
ylabel('Gain [dB]')
grid(True)

savefig('HPF.png')
```


Unity-Gain HPF Frequency Response, f_{3dB} = 4000 Hz
0
−10
−20
−30
−40
−50
Gain [dB]
10^1
10^2
10^3
10^4
10^5
10^6
Frequency [Hz]

```
# -*- coding: utf-8 -*-
"""
Created on Mon Jun 25 22:02:29 2018

@author: Zengweijie (Fred) Chen

Frequency Response Plot
Band-Pass Filter

"""

from pylab import *

f_l = 400
f_h = 120000

K = 1
f = logspace(1,7) # frequencies from 10^1 to 10^5

def FreqResp(w, K, wl, wh):
    H_LPF = 1 / (1.0 + 1j * w / wh)
    H_HPF = K*1j*w / (wl + 1j*w)
    H = H_LPF * H_HPF
    return H

H_log = 20*log10(abs(FreqResp(2*pi*f, K, 2*pi*f_l, 2*pi*f_h)))


# Plotting Configuration
figure(num = 1, figsize = (16, 12), dpi = 100, facecolor='w', edgecolor='k')
plot(f, H_log)
xscale('log')
title(r'Unity-Gain BPF Frequency Response, %s Hz & %s Hz' %(f_l, f_h))
xlabel('Frequency [Hz]')
ylabel('Gain [dB]')
grid(True)

savefig('BPF.png')
```


Unity-Gain BPF Frequency Response, 400 Hz & 120000 Hz
0
−5
−10
−15
−20
−25
−30
−35
−40
Gain [dB]
10^1
10^2
10^3
10^4
10^5
10^6
10^7
Frequency [Hz]