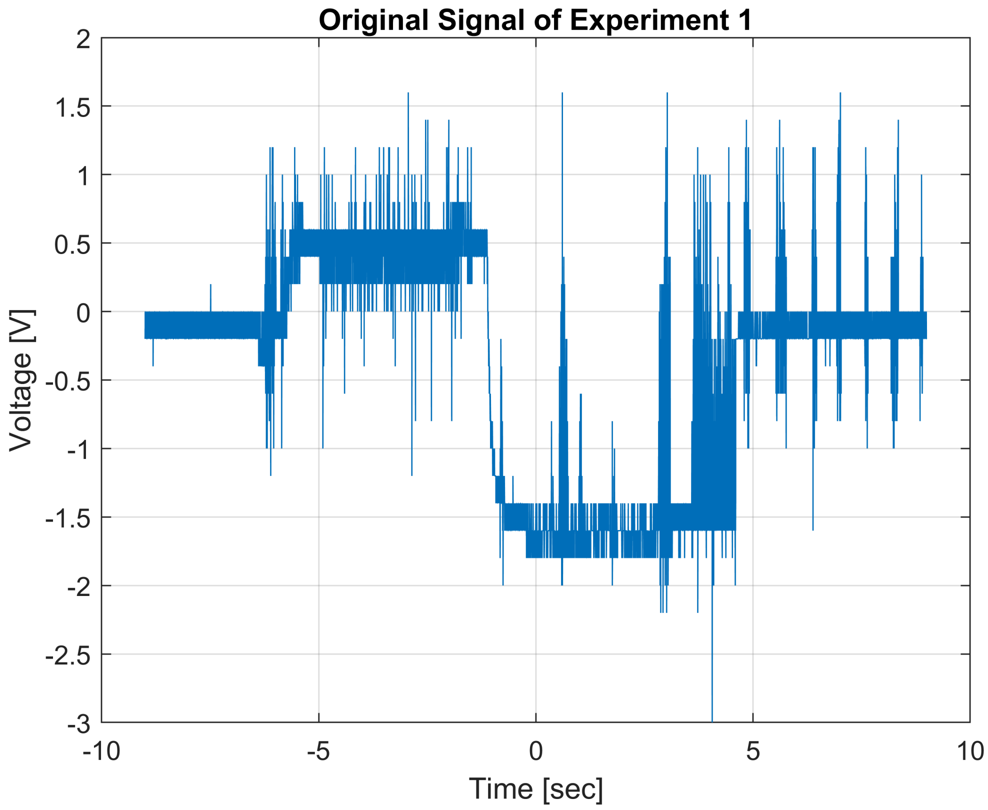
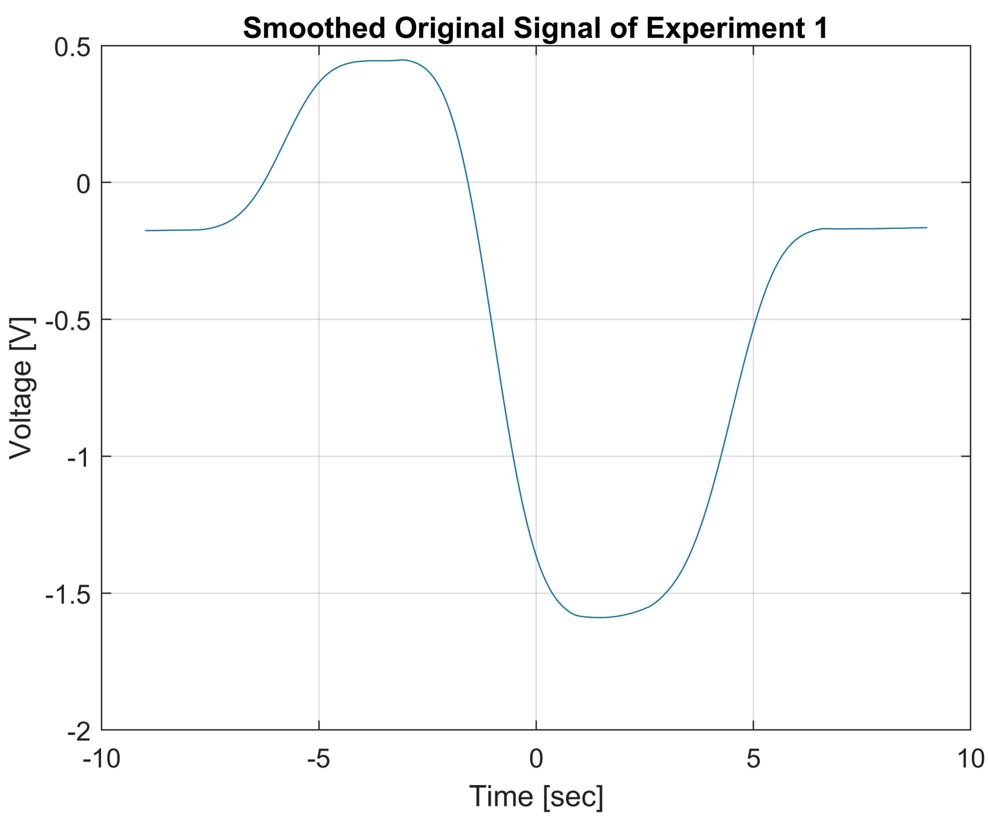
```
% Lab 2 Data analysis

%% BK2
clear;clc;close all

%Initializing
A=readmatrix('BK000002.csv');
Time=A(:,1);
Voltage=A(:,2);
%% Original signal
plot(Time,Voltage),title('Original Signal of Experiment 1')
xlabel("Time [sec]")
ylabel("Voltage [V]")
grid on
```



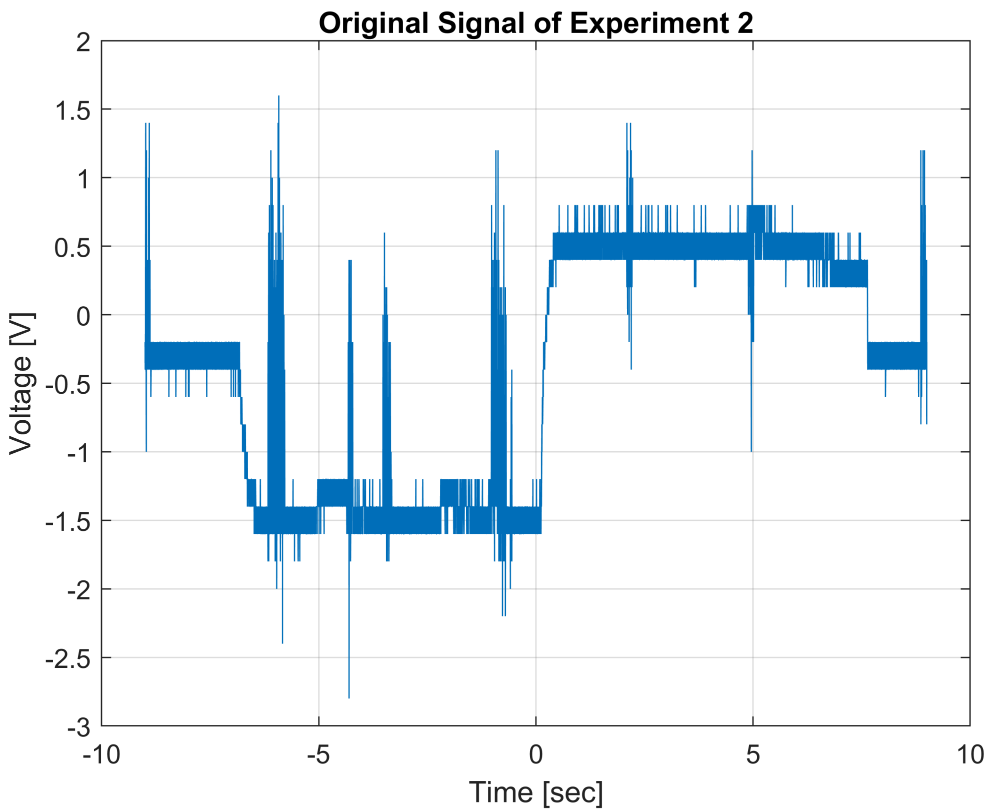

```
%Generate frequency magnitude spectrum using FFT
fs = 50000; %sampling frequency (Hz)
L = length(A); %length of signal (# of samples)
y = fft(A);
ds = abs(y/L); %double-sided amplitude spectrum (w/ normalized amplitudes to match power of tim
ss = ds(1:(L/2)+1); %single-sided amplitude spectrum
%fft divides total input power in 1/2 b/w +/- frequency values, hence . . .
ss(2:end-1) = 2*ss(2:end-1); %doubles amplitudes to compensate for missing (-) values and get t
f = (0:L/2)*fs/L; %converts freq axis from freq bins to Hz
figure
% plot(f,ss)
% xlabel('Frequency (Hz)')
```

```
% ylabel('Amplitude')
% title('Single-Sided Amplitude Spectrum of ECG')

%Signal Processing
B=lowpass(Voltage,1,fs);
figure
% plot(Time,B),title('Signal After Lowpass Filtering')

%% Smoothed curve
figure
C=smoothdata(B,'gaussian');
% plot(Time,C),title('Filtered Signal with Curve Smoothing')
figure
D=smoothdata(Voltage,'gaussian');
plot(Time,D),title('Smoothed Signal of Experiment 1')
xlabel("Time [sec]")
ylabel("Voltage [V]")
grid on
```



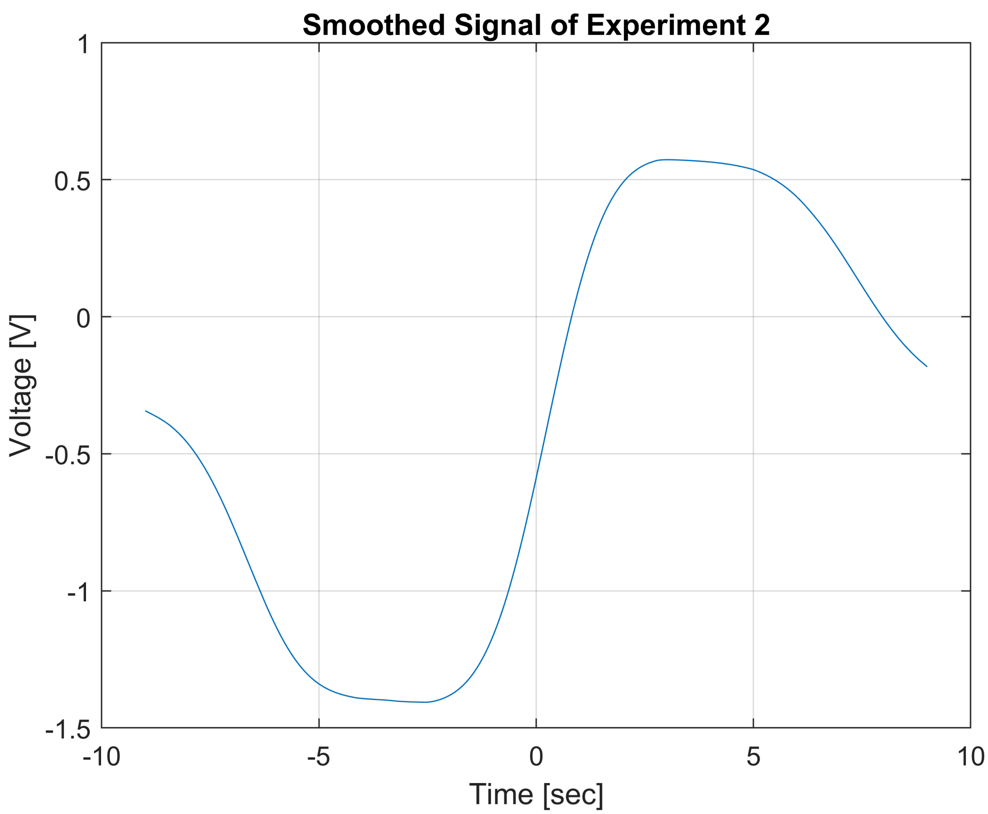

```
% %FIR lowpass filter
% fn = fs/2; %Nyquist frequency
% fc = 1; %cut-off frequency
% n = 30; %filter order
% b = fir1(n,fc/fn);
% z = filtfilt(b,1,Voltage);
% figure
% plot(Time,z),title('FIR Lowpass filter')
```

```
%% BK3
clear;clc;close all

%Initializing
A=readmatrix('BK000003.csv');
Time=A(:,1);
Voltage=A(:,2);
%% Original signal
plot(Time,Voltage),title('Original Signal of Experiment 2')
xlabel("Time [sec]")
ylabel("Voltage [V]")
grid on
```

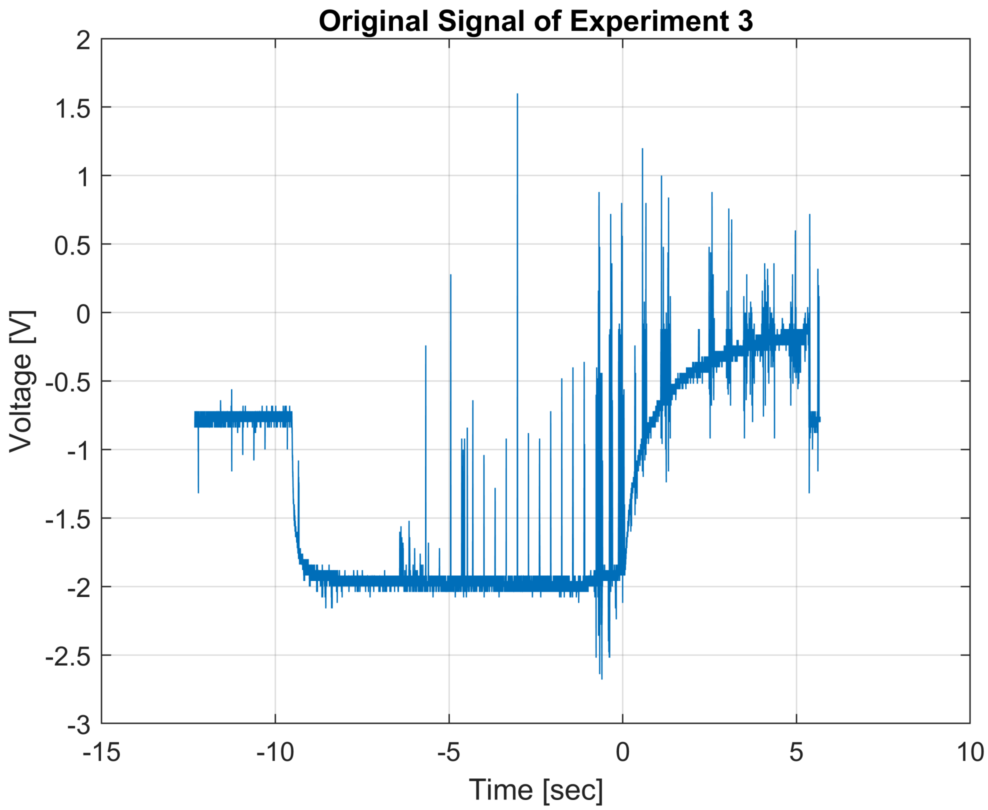



```
%Generate frequency magnitude spectrum using FFT
fs = 50000; %sampling frequency (Hz)
L = length(A); %length of signal (# of samples)
y = fft(A);
ds = abs(y/L); %double-sided amplitude spectrum (w/ normalized amplitudes to match power of tim
ss = ds(1:(L/2)+1); %single-sided amplitude spectrum
%fft divides total input power in 1/2 b/w +/- frequency values, hence . . .
ss(2:end-1) = 2*ss(2:end-1); %doubles amplitudes to compensate for missing (-) values and get t
f = (0:L/2)*fs/L; %converts freq axis from freq bins to Hz
figure

%Signal Processing
B=lowpass(Voltage,1,fs);
```

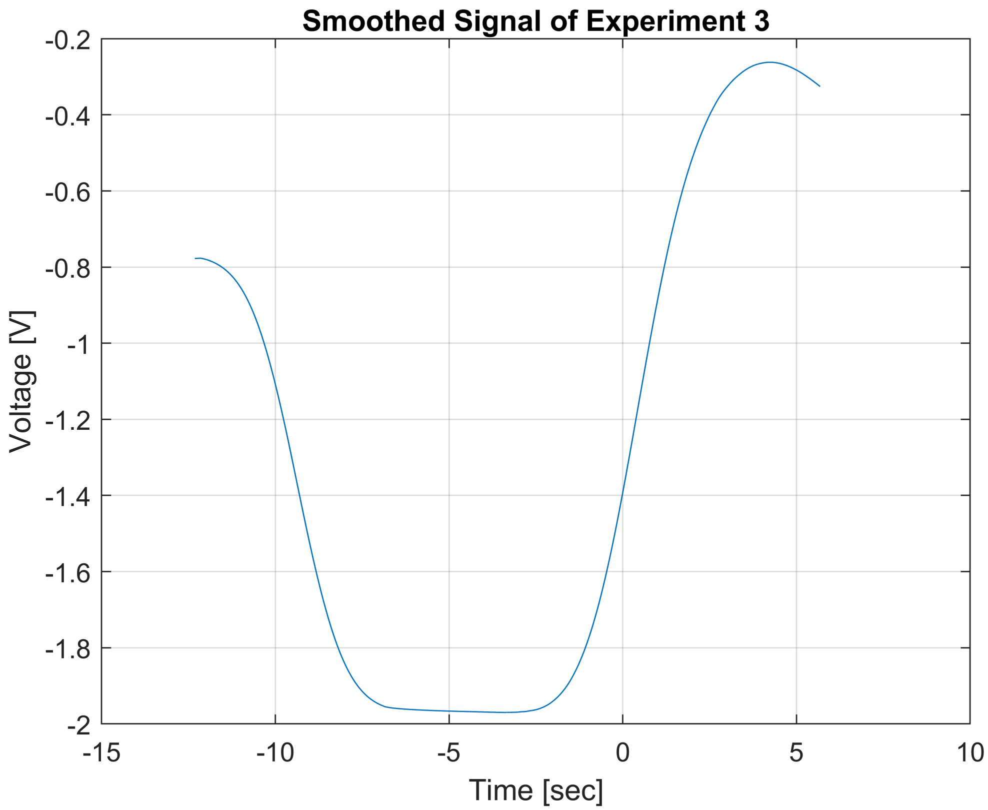
```
figure

%% Smoothed curve
figure
C=smoothdata(B,'gaussian');
figure
D=smoothdata(Voltage,'gaussian');
plot(Time,D),title('Smoothed Signal of Experiment 2')
xlabel("Time [sec]")
ylabel("Voltage [V]")
grid on
```



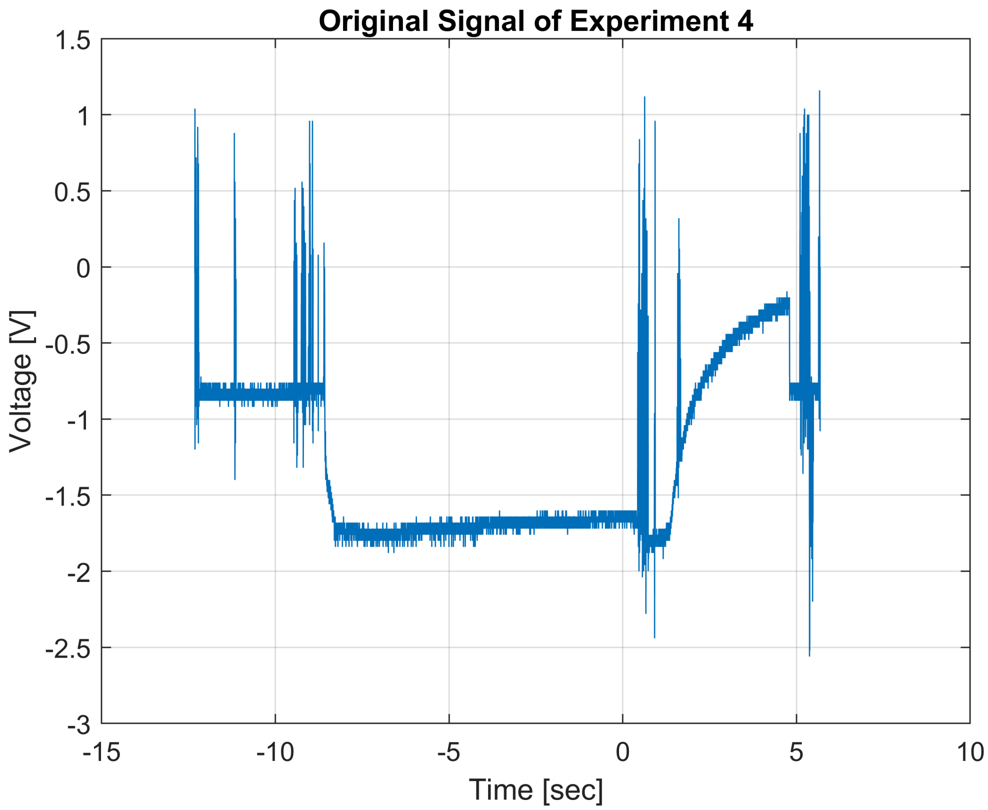

```
%% BK4
clear;clc;close all

%Initializing
A=readmatrix('BK000004.csv');
Time=A(:,1);
Voltage=A(:,2);
%% Original signal
plot(Time,Voltage),title('Original Signal of Experiment 3')
xlabel("Time [sec]")
ylabel("Voltage [V]")
grid on
```



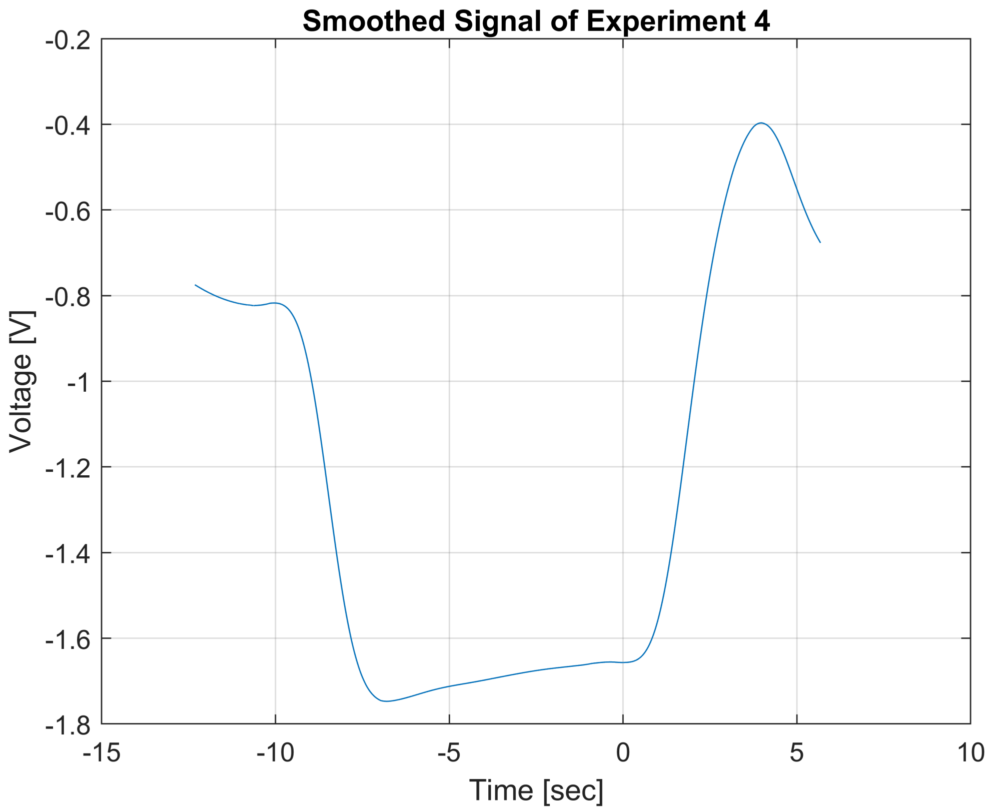

```
%Generate frequency magnitude spectrum using FFT
fs = 50000; %sampling frequency (Hz)
L = length(A); %length of signal (# of samples)
y = fft(A);
ds = abs(y/L); %double-sided amplitude spectrum (w/ normalized amplitudes to match power of tim
ss = ds(1:(L/2)+1); %single-sided amplitude spectrum
%fft divides total input power in 1/2 b/w +/- frequency values, hence . . .
ss(2:end-1) = 2*ss(2:end-1); %doubles amplitudes to compensate for missing (-) values and get t
f = (0:L/2)*fs/L; %converts freq axis from freq bins to Hz
figure

%Signal Processing
B=lowpass(Voltage,1,fs);
figure

%% Smoothed curve
figure
C=smoothdata(B,'gaussian');
figure
D=smoothdata(Voltage,'gaussian');
plot(Time,D),title('Smoothed Signal of Experiment 3')
xlabel("Time [sec]")
ylabel("Voltage [V]")
grid on
```



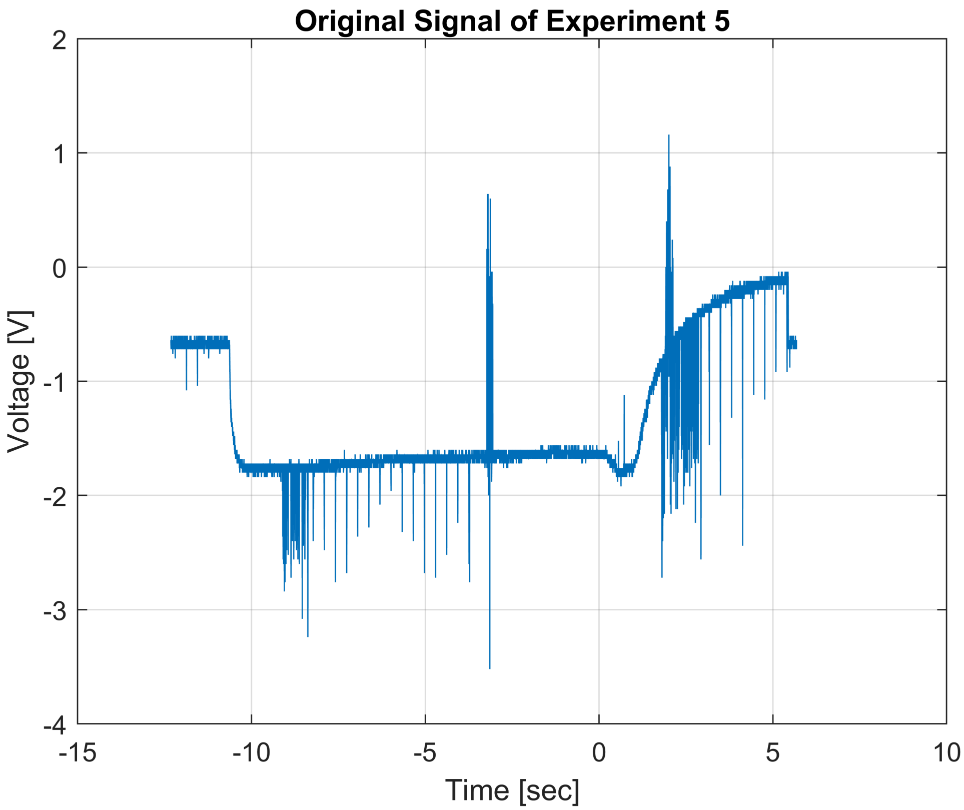

```
%% BK5
clear;clc;close all

%Initializing
A=readmatrix('BK000005.csv');
Time=A(:,1);
Voltage=A(:,2);
%% Original signal
plot(Time,Voltage),title('Original Signal of Experiment 4')
xlabel("Time [sec]")
ylabel("Voltage [V]")
grid on
```



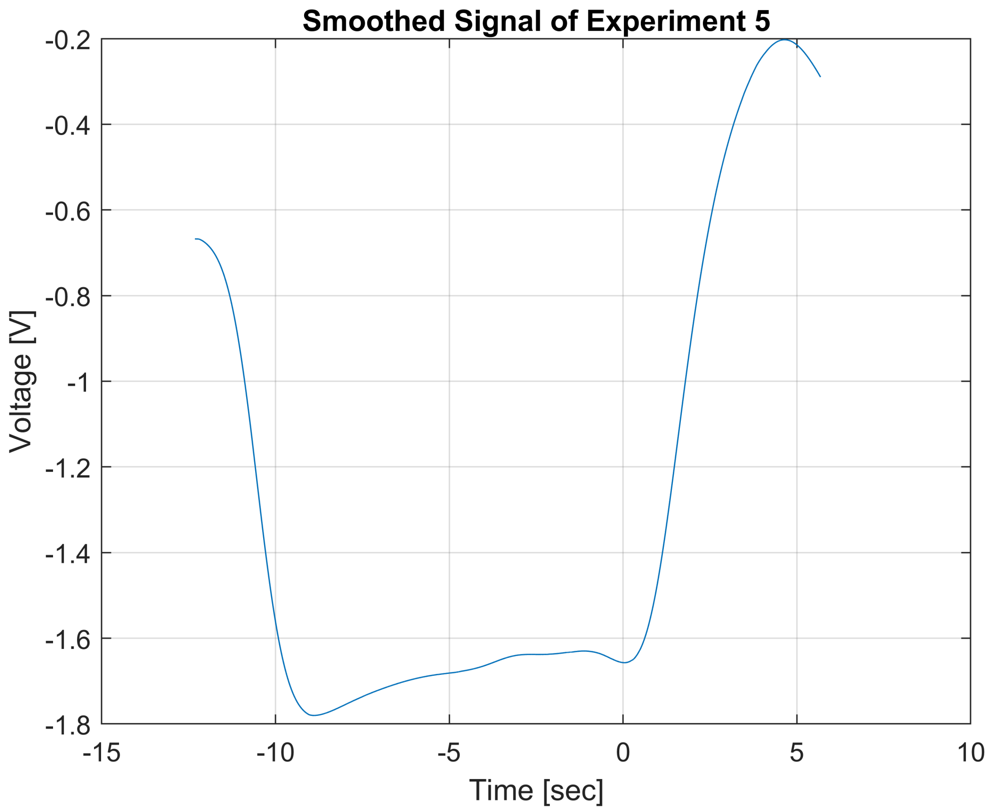

```
%Generate frequency magnitude spectrum using FFT
fs = 50000; %sampling frequency (Hz)
L = length(A); %length of signal (# of samples)
y = fft(A);
ds = abs(y/L); %double-sided amplitude spectrum (w/ normalized amplitudes to match power of tim
ss = ds(1:(L/2)+1); %single-sided amplitude spectrum
%fft divides total input power in 1/2 b/w +/- frequency values, hence . . .
ss(2:end-1) = 2*ss(2:end-1); %doubles amplitudes to compensate for missing (-) values and get t
f = (0:L/2)*fs/L; %converts freq axis from freq bins to Hz
figure

%Signal Processing
B=lowpass(Voltage,1,fs);
figure

%% Smoothed curve
figure
C=smoothdata(B,'gaussian');
figure
D=smoothdata(Voltage,'gaussian');
plot(Time,D),title('Smoothed Signal of Experiment 4')
xlabel("Time [sec]")
ylabel("Voltage [V]")
grid on
```




```
%% BK5
clear;clc;close all

%Initializing
A=readmatrix('BK000006.csv');
Time=A(:,1);
Voltage=A(:,2);
%% Original signal
plot(Time,Voltage),title('Original Signal of Experiment 5')
xlabel("Time [sec]")
ylabel("Voltage [V]")
grid on
```




```
%Generate frequency magnitude spectrum using FFT
fs = 50000; %sampling frequency (Hz)
L = length(A); %length of signal (# of samples)
y = fft(A);
ds = abs(y/L); %double-sided amplitude spectrum (w/ normalized amplitudes to match power of tim
ss = ds(1:(L/2)+1); %single-sided amplitude spectrum
%fft divides total input power in 1/2 b/w +/- frequency values, hence . . .
ss(2:end-1) = 2*ss(2:end-1); %doubles amplitudes to compensate for missing (-) values and get t
f = (0:L/2)*fs/L; %converts freq axis from freq bins to Hz
figure

%Signal Processing
B=lowpass(Voltage,1,fs);
figure

%% Smoothed curve
figure
C=smoothdata(B,'gaussian');
figure
D=smoothdata(Voltage,'gaussian');
plot(Time,D),title('Smoothed Signal of Experiment 5')
xlabel("Time [sec]")
ylabel("Voltage [V]")
grid on
```


Smoothed Signal of Experiment 5
Voltage [V]
-0.2
-0.4
-0.6
-0.8
-1
-1.2
-1.4
-1.6
-1.8
-15
-10
-5
0
5
10
Time [sec]