In [2]:
```
# -*- coding: utf-8 -*-
import numpy as np
import matplotlib.pyplot as plt
from scipy.io import wavfile
from scipy.signal import welch, lfilter
from scipy import fftpack
```

In [3]:
```
def plot_time_domain_channels_stereo(time, lc, rc):
    #Plota as figuras ao longo do tempo
    #Plota os canais esquerdo e direito
    plt.figure(1,figsize=(20, 5))
    plt.plot(time, lc, label="Canal esquerdo")
    plt.legend()
    plt.xlabel("Tempo [s]")
    plt.ylabel("Amplitude")
    plt.show()

    plt.figure(2,figsize=(20, 5))
    plt.plot(time, rc, color="red",label="Canal direito")
    plt.legend()
    plt.xlabel("Tempo [s]")
    plt.ylabel("Amplitude")
    plt.show()
```

In [4]:
```
def plot_spect_welch_channels_stereo(lc, rc, fs):
    #Sample Frequencies, Power Spectral Density
    sf_lc, psd_lc = welch(
        x=lc,
        fs=fs,
        window='flattop',
        nperseg=512,
        scaling='spectrum'
    )
    sf_rc, psd_rc = welch(
        x=rc,
        fs=fs,
        window='flattop',
        nperseg=512,
        scaling='spectrum'
    )

    #Plota o espectro do sinal para frequencias normalizadas entre 0 1 pi
    #(frequencias positivas)
    plt.subplots(figsize=(15,5))
    plt.subplot(1, 2, 1)
    plt.semilogy(sf_lc, psd_lc, label="Canal esquerdo")
    plt.legend()
    plt.xlabel('Frequencia [rad]')
    plt.ylabel('Espectro')

    plt.subplot(1, 2, 2)
    plt.semilogy(sf_rc, psd_rc, color="red", label="Canal direito")
    plt.legend()
    plt.xlabel('Frequencia [rad]')
    plt.ylabel('Espectro')
    plt.show()
```

In [5]:
```
def plot_spect_fft_channels_stereo(lc,rc, sampling_rate, nfft):
    freq_lc = np.linspace(0., sampling_rate, nfft) #Interpola para determina
```

```
    sig_fft_lc = fftpack.rfft(lc,nfft)
    plt.subplots(figsize=(15,5))
    plt.subplot(1, 2, 1)
    plt.title("Canal esquerdo")
    plt.plot(freq_lc, np.abs(sig_fft_lc), label="fft")
    plt.legend()
    plt.xlabel('Frequencia [Hz]')
    plt.ylabel('Espectro de amplitudes')
    #plt.plot(freq_lc, np.abs(fftpack.fftshift(sig_fft_lc)), label="fftshift
    plt.legend()

    freq_rc = np.linspace(0., sampling_rate, nfft) #Interpola para determinar
    sig_fft_rc = fftpack.rfft(rc,nfft)
    plt.subplot(1, 2, 2)
    plt.title("Canal direito")
    plt.plot(freq_rc, np.abs(sig_fft_rc), color="red", label="fft")
    plt.legend()
    plt.xlabel('Frequencia [Hz]')
    plt.ylabel('Espectro de amplitudes')
    #plt.plot(freq_rc, np.abs(fftpack.fftshift(sig_fft_rc)), color="green", l
    plt.legend()
    plt.show()
```

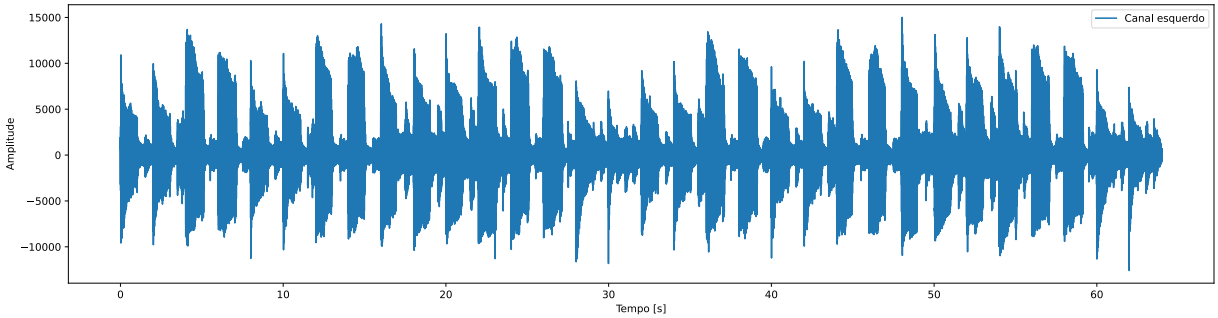
In [6]:

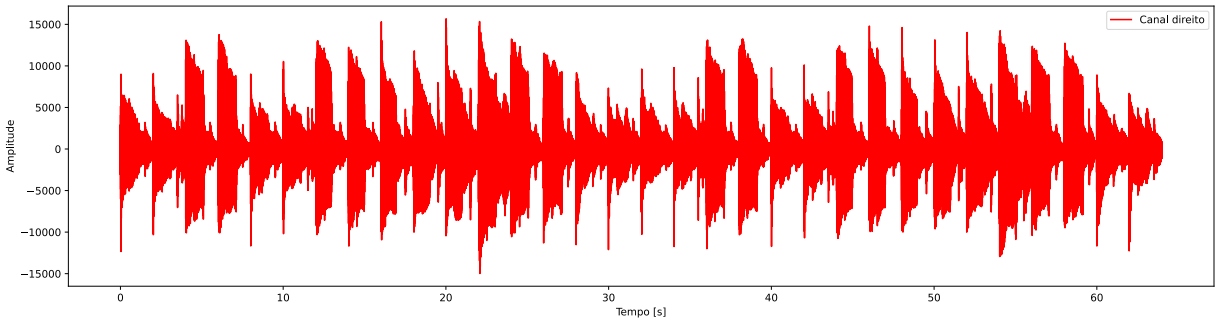
```
#Carrega o arquivo
sampling_rate, data = wavfile.read('569127__josefpres__dark-loops-201-simple-
#sampling_rate, data = wavfile.read('581010__xcreenplay__smoking-in-the-angel

number_of_samples = data.shape[0]
number_of_channels = data.shape[1]

#Tempo total = numero de amostras / fs
duration = number_of_samples / sampling_rate

#Carrega o arquivo em dois canais (audio estereo)
left_channel = data[:, 0]
right_channel  = data[:, 1]

print(f"Numero de canais = {number_of_channels}")
print(f"Duracao = {duration}s")
print(f'Numero de amostras: {number_of_samples}')
print(f"Amostras por segundo: {sampling_rate}Hz")
```

```
Numero de canais = 2
Duracao = 64.0s
Numero de amostras: 2822400
Amostras por segundo: 44100Hz
/tmp/ipykernel_361103/1384073729.py:2: WavFileWarning: Chunk (non-data) not u
nderstood, skipping it.
  sampling_rate, data = wavfile.read('569127__josefpres__dark-loops-201-simpl
e-mix-2-short-loop-60-bpm.wav')
```

In [7]:

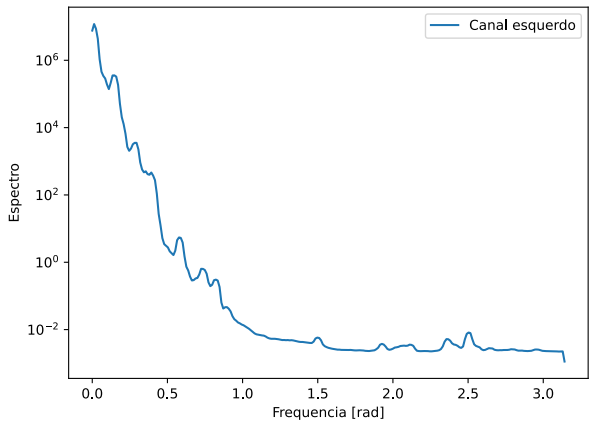
```
#Interpola para determinar eixo do tempo
time = np.linspace(0., duration, number_of_samples)
plot_time_domain_channels_stereo(
    time=time,
    lc=left_channel,
    rc=right_channel
)
```

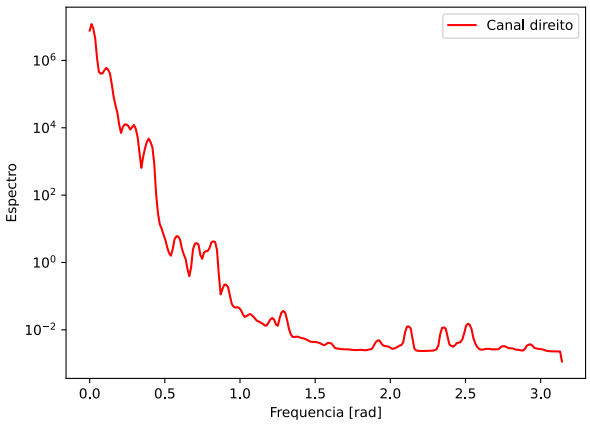


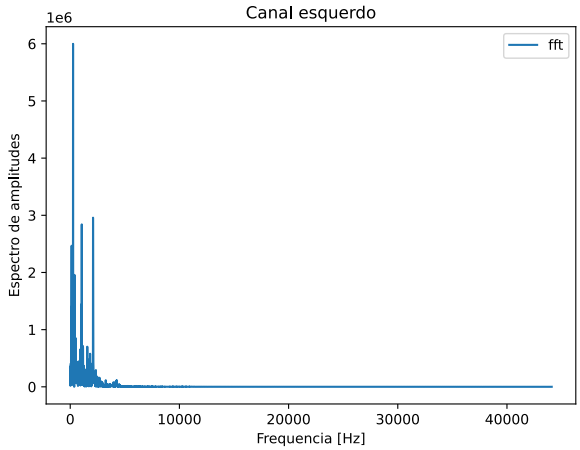


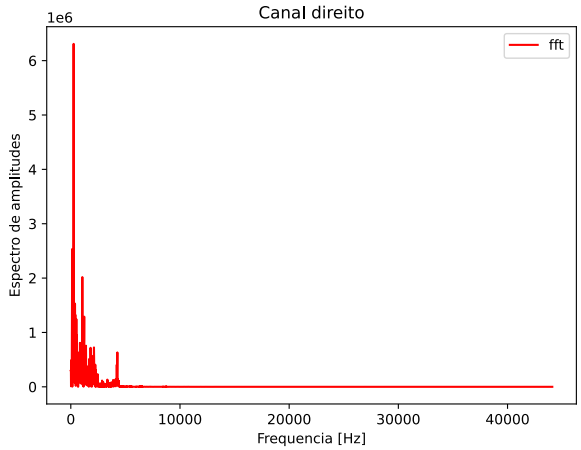
In [8]:

```
plot_spect_welch_channels_stereo(
    lc  = left_channel,
    rc  = right_channel,
    fs = 2*np.pi
)

plot_spect_fft_channels_stereo(
    lc  = left_channel,
    rc  = right_channel,
    sampling_rate=sampling_rate,
    nfft=4096
)
```



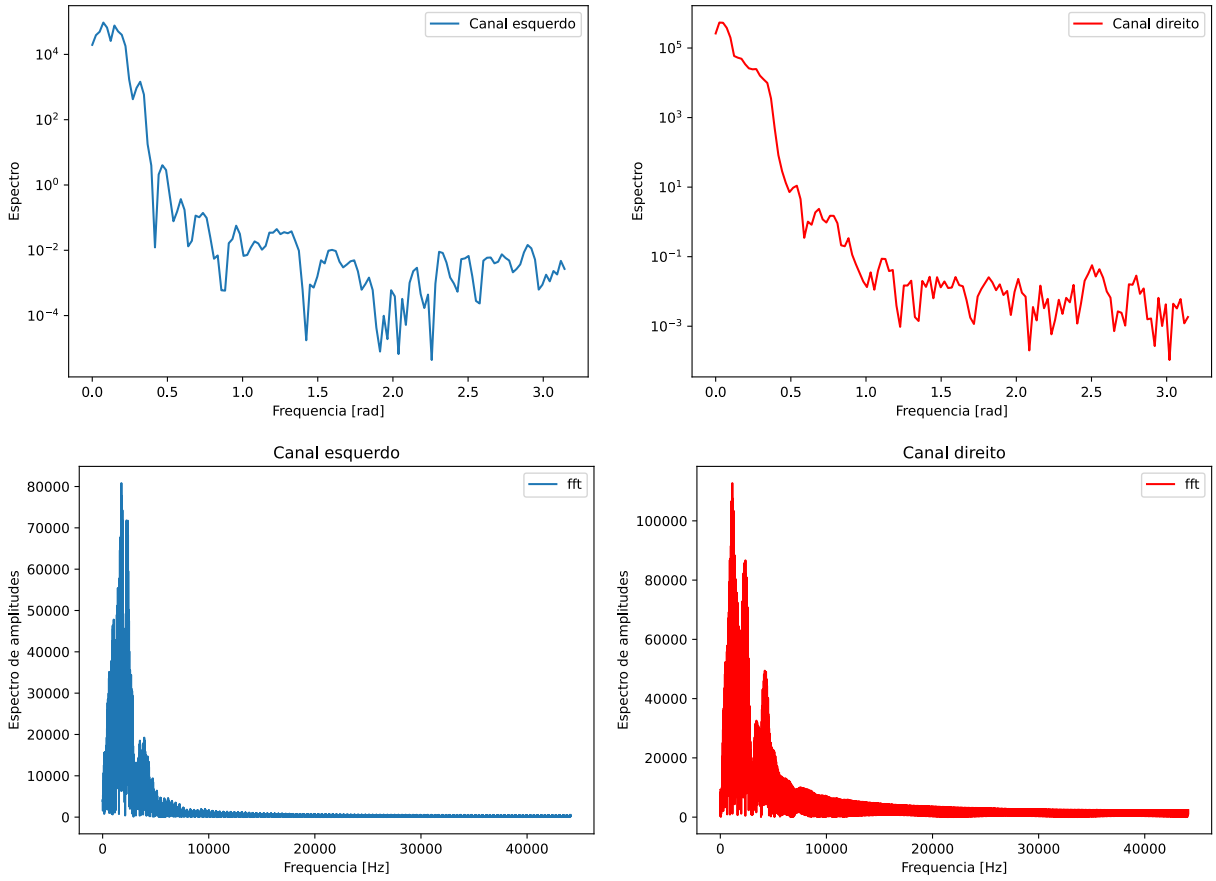






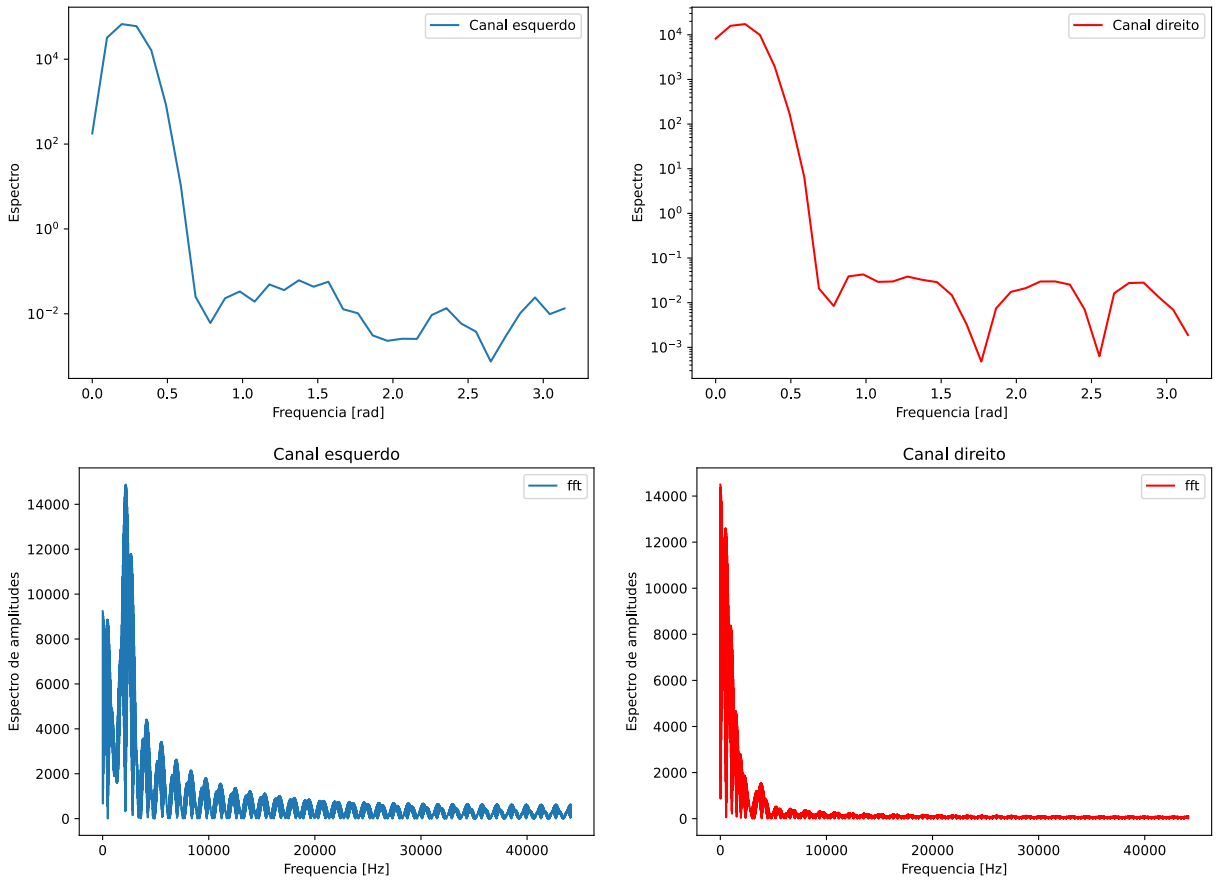
In [9]:
```
plot_spect_welch_channels_stereo(
    lc  = left_channel[:256],
    rc  = right_channel[:256],
    fs = 2*np.pi
)

plot_spect_fft_channels_stereo(
    lc  = left_channel[:256],
    rc  = right_channel[:256],
    sampling_rate=sampling_rate,
    nfft=4096
)
```

```
/home/hemerson/.cache/pypoetry/virtualenvs/pds-vmrzwVap-py3.8/lib/python3.8/site-packages/scipy/signal/spectral.py:1964: UserWarning: nperseg = 512 is greater than input length  = 256, using nperseg = 256
  warnings.warn('nperseg = {0:d} is greater than input length '
```



In [12]:
```
plot_spect_welch_channels_stereo(
    lc  = left_channel[:64],
    rc  = right_channel[:64],
    fs = 2*np.pi
)

plot_spect_fft_channels_stereo(
    lc  = left_channel[:64],
    rc  = right_channel[:64],
    sampling_rate=sampling_rate,
    nfft=4096
)
```

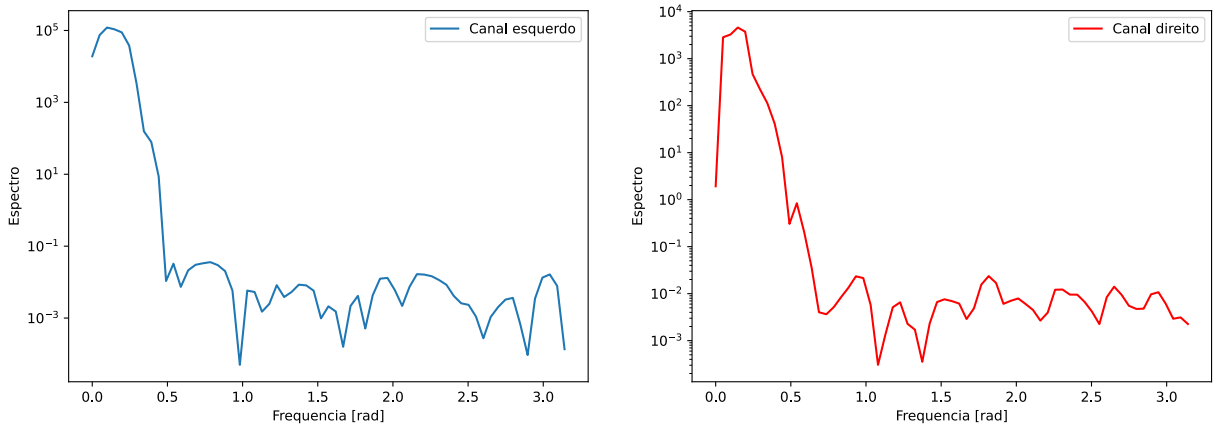
```
/home/hemerson/.cache/pypoetry/virtualenvs/pds-vmrzwVap-py3.8/lib/python3.8/site-packages/scipy/signal/spectral.py:1964: UserWarning: nperseg = 512 is greater than input length  = 64, using nperseg = 64
  warnings.warn('nperseg = {0:d} is greater than input length '
```

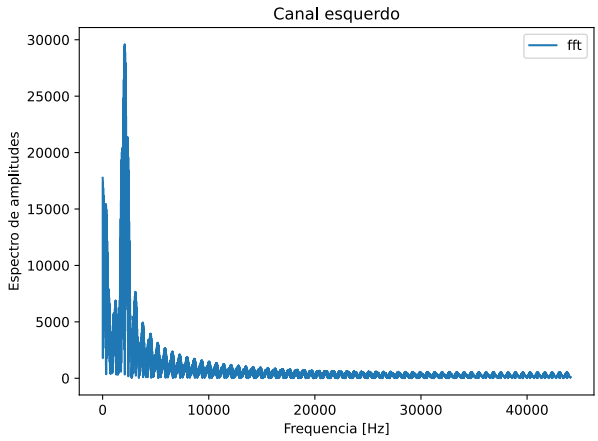


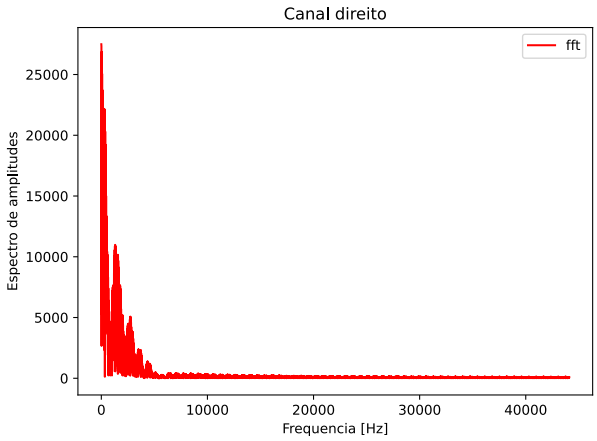
In [10]:

```
plot_spect_welch_channels_stereo(
    lc  = left_channel[:128],
    rc  = right_channel[:128],
    fs = 2*np.pi
)

plot_spect_fft_channels_stereo(
    lc  = left_channel[:128],
    rc  = right_channel[:128],
    sampling_rate=sampling_rate,
    nfft=4096
)
```

```
/home/hemerson/.cache/pypoetry/virtualenvs/pds-vmrzwVap-py3.8/lib/python3.8/site-packages/scipy/signal/spectral.py:1964: UserWarning: nperseg = 512 is greater than input length  = 128, using nperseg = 128
  warnings.warn('nperseg = {0:d} is greater than input length '
```

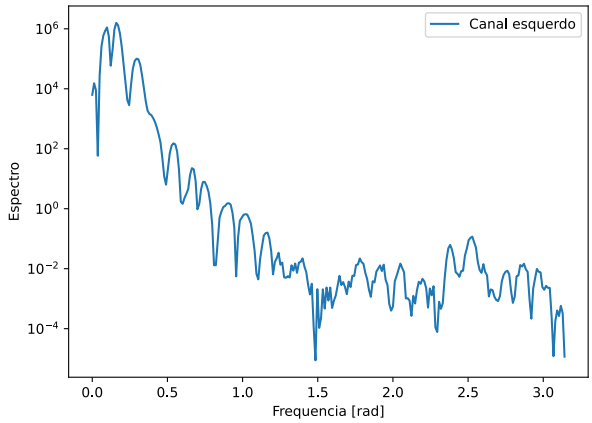


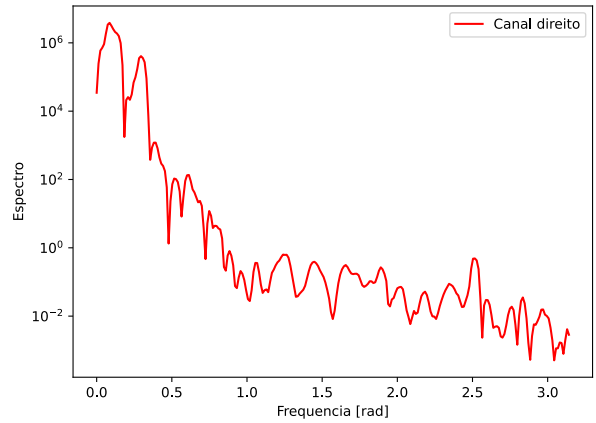


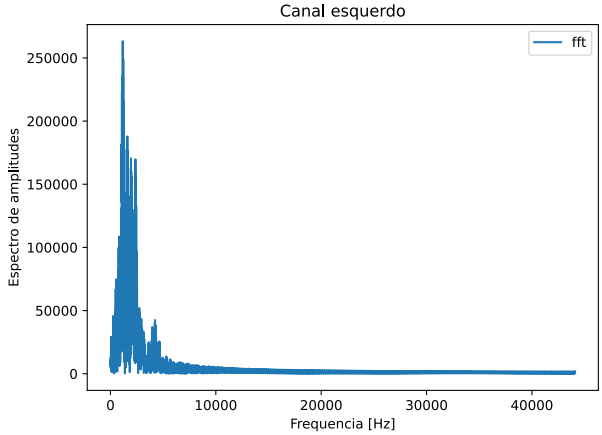


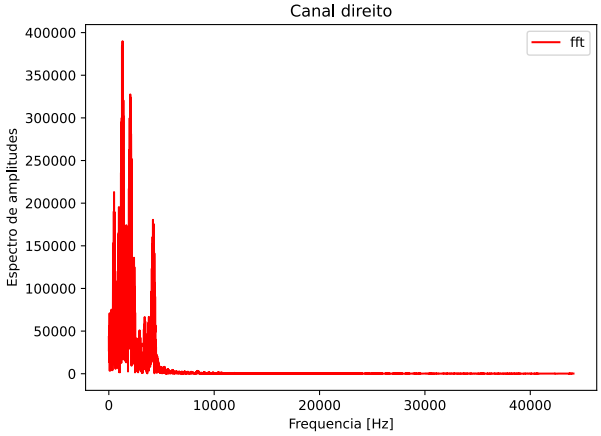
In [11]:

```
plot_spect_welch_channels_stereo(
    lc  = left_channel[:512],
    rc  = right_channel[:512],
    fs = 2*np.pi
)

plot_spect_fft_channels_stereo(
    lc  = left_channel[:512],
    rc  = right_channel[:512],
    sampling_rate=sampling_rate,
    nfft=4096
)
```









In [13]:

```
plot_spect_welch_channels_stereo(
    lc  = left_channel[:1024],
    rc  = right_channel[:1024],
    fs = 2*np.pi
)

plot_spect_fft_channels_stereo(
```

```
    lc  = left_channel[:1024],
    rc  = right_channel[:1024],
    sampling_rate=sampling_rate,
    nfft=4096
)
```