```python
import numpy as np
import sounddevice as sd
import soundfile as sf
import matplotlib.pyplot as plt

from scipy import signal
from scipy.signal import butter, lfilter

# Parametros 7: f1 = 5.7 kHz, f2 = 6 kHz, δ2 = 0.1
# Butterworth filter


# Noise function
def noise_audio(voice, a1, a2):
    n = np.arange(len(voice))  # Time axis
    noise = a1 * np.cos(0.76 * np.pi * n) + a2 * np.cos(0.8 * np.pi * n)
    noisy_voice = noise + voice
    return noisy_voice  # Return audio with noise


# Applying filter
def butterworth_filter(noisy_voice, cutoff, sample_rate, order):
    b, a = butter(N=order, Wn=cutoff, fs=sample_rate, btype='low', analog=False)
    y = lfilter(b, a, noisy_voice)
    return y


# Set-up
recording_time = 5  # seconds
sample_rate = 44100  # sample rate(Hz)
std_audio = "Audio padrao IIR"  # Audio gravado
noisy_audio = "Audio com Ruído IIR"  # Audio com ruido
filtred_audio_butterworth = "Audio filtrado com Butterworth"  # Audio filtrado com butterworth
a1 = 0.01
a2 = 0.01

```

```
37 # IIR filter settings
38 order = 4
39 cutoff = 1000  # Cut-Off Frequency (rad)
40
41 # Audio recording
42 print("Gravando..")
43 voice = sd.rec(int(recording_time * sample_rate),
   samplerate=sample_rate, channels=1)
44 sd.wait()
45
46 # Noisy audio
47 noisy_voice = noise_audio(voice.flatten(), 0.7, 0.7)
48
49 # Applying butteworth filter
50 btw_filtered_sign = butterworth_filter(noisy_voice,
   cutoff, sample_rate, order)
51
52 # Save WAV file
53 file_name_wav1 = noisy_audio + ".wav"
54 sf.write(file_name_wav1, noisy_voice, sample_rate)
55
56 file_name_wav2 = std_audio + ".wav"
57 sf.write(file_name_wav2, voice.flatten(), sample_rate
   )
58
59 file_name_wav3 = filtred_audio_butterworth + ".wav"
60 sf.write(file_name_wav3, btw_filtered_sign,
   sample_rate)
61
62 print("""Gravado com sucesso.
63     Arquivo salvo com sucesso.""")
64
65 # Charts plot
66 t = np.linspace(0, recording_time, num=len(voice))
67 plt.figure(figsize=(10, 8))
68
69 # Original audio file
70 plt.subplot(3, 1, 1)
71 plt.plot(t, voice.flatten(), color="purple")
72 plt.title("Sinal Original")
73 plt.xlabel("Tempo(s)")
```

```
74 plt.ylabel("Amplitude")
75 plt.ylim(-0.10, 0.10)
76
77 # Noisy audio
78 plt.subplot(3, 1, 2)
79 plt.plot(t, noisy_voice, color="lightgreen")
80 plt.title("Sinal com Ruído")
81 plt.xlabel("Tempo (s)")
82 plt.ylabel("Amplitude")
83
84 # Filtered audio
85 plt.subplot(3, 1, 3)
86 plt.plot(t, btw_filtered_sign, color="lightblue")
87 plt.title("Sinal Filtrado")
88 plt.legend("Filtro Butterworth")
89 plt.xlabel("Tempo (s)")
90 plt.ylabel("Amplitude")
91 plt.ylim(-0.10, 0.10)
92
93 plt.tight_layout()
94 plt.savefig("plots_IIR.png")
95 plt.show()
96
```