

Tarjeta de Audio

Escanear dispositivos de audio

```
audioinfo=audiodevinfo
```

```
audioinfo = struct with fields:  
    input: [1x1 struct]  
    output: [1x1 struct]
```

Crear objeto de audio

```
Id=-1
```

```
Id = -1
```

```
fs=8000; %Frecuencia de muestreo por segundo [Hz] la cantidad  
% es proporcional a la calidad y a la memoria :c
```

```
CH_No=1; %No. de canales de entrada (Stereo[2] o Mono[1])  
BitDepth=8; %Profundidad de Bits por muestra
```

```
%Default: fs=8kHz, BitDepth=8 bit, 1CH.  
recordObjeto=audiorecorder(fs,BitDepth,CH_No,Id)
```

```
recordObjeto =  
    audiorecorder with properties:  
  
        SampleRate: 8000  
        BitsPerSample: 8  
        NumChannels: 1  
        DeviceID: -1  
        CurrentSample: 1  
        TotalSamples: 0  
            Running: 'off'  
            StartFcn: []  
            StopFcn: []  
            TimerFcn: []  
        TimerPeriod: 0.0500  
            Tag: ''  
        UserData: []  
        Type: 'audiorecorder'
```

Grabacion

```
T=4;%Timepo de grabacion  
recordblocking(recordObjeto,T);
```

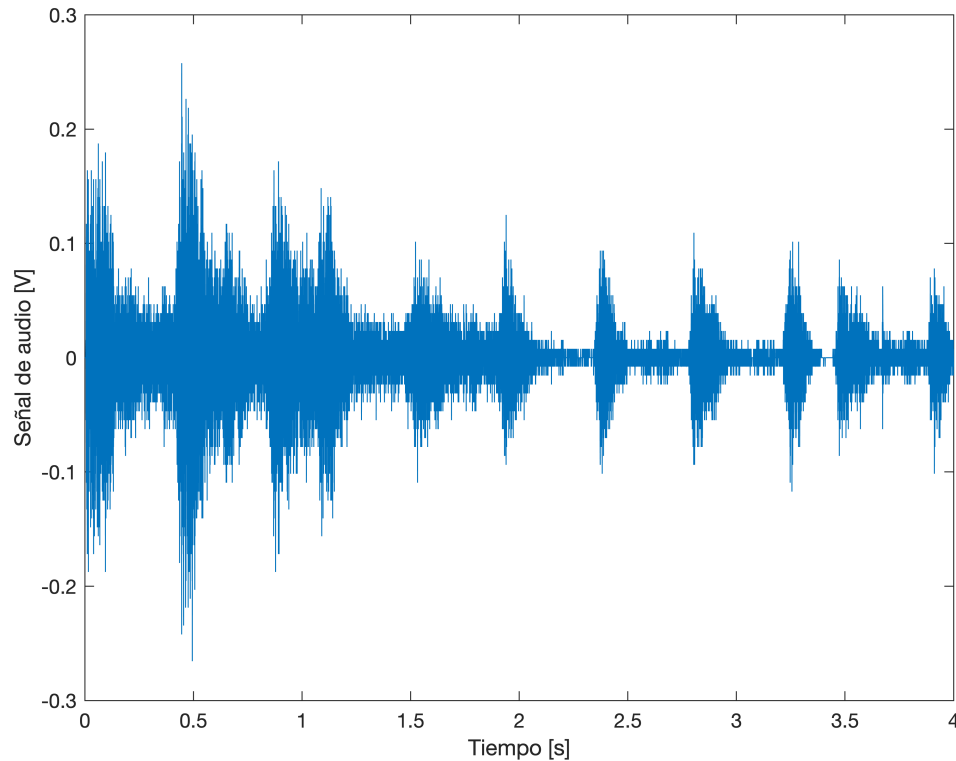
Ploteo

```
%% puse a viva la vida-coldplay, SIUUU  
data_grabacion=getaudiodata(recordObjeto); %Guarda el audio y lo saca como matriz
```

```

%Vector de tiempo
t=1/fs : 1/fs : T;
plot(t,data_grabacion)
xlabel("Tiempo [s]")
ylabel("Señal de audio [V]")

```



Reproduccion de audio

```

% play(recordObjeto); %Reproduce la grabacion
sound(data_grabacion,round(fs*2))%altera el audio, altera la frecuencia xd

```

Espectro

```

%Parametros importantes
N=numel(data_grabacion); %Numero de muestras de la señal
N_spec=N/50; %Numero de muestras de espectrograma. (mas grande
% el denominador mayor es la resolucion)

% Funciones de descomposicion espectral
[Pxx_Per,F_Pxx]=periodogram(data_grabacion,rectwin(N),[],fs);
[~,F_Spec,T_Spec,Pxx_Spec]=spectrogram(data_grabacion,rectwin(N_spec),[],[],fs);

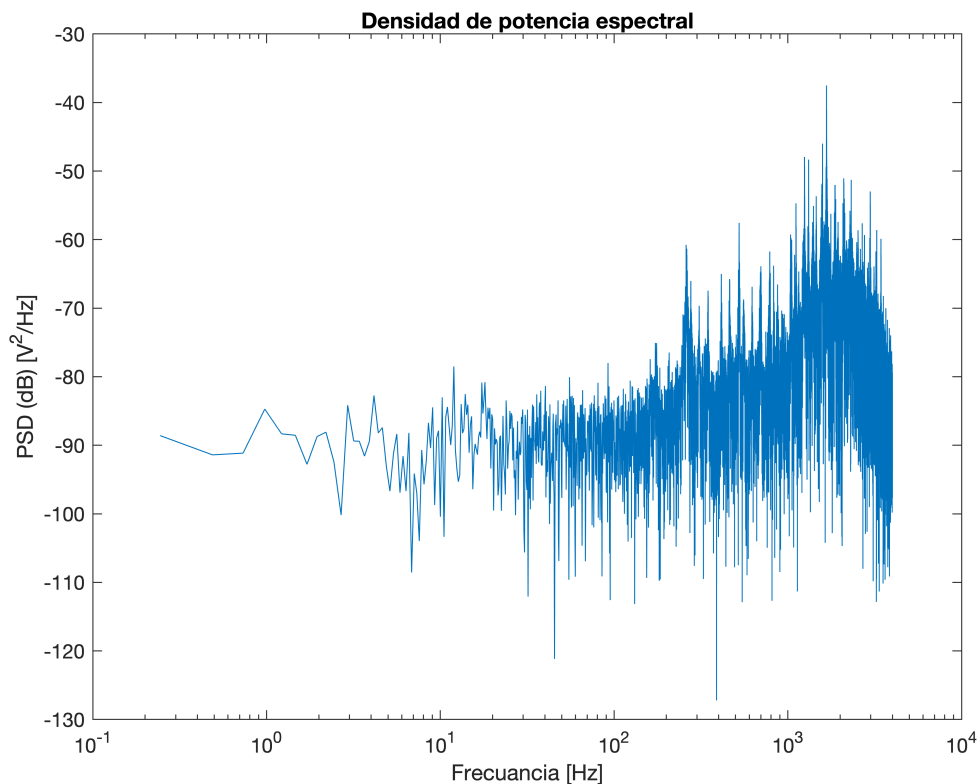
%Periodograma
figure
semilogx(F_Pxx,10*log10(Pxx_Per))

```

```

title("Densidad de potencia espectral")
xlabel("Frecuencia [Hz]")
ylabel("PSD (dB) [V^2/Hz]")

```



%%mas alto el pico mas energia, escala logaritmica

```

%Espectrograma
figure
p=pcolor(T_Spec,F_Spec,10*log10(Pxx_Spec));
set(p,'EdgeColor','none')
xlabel("Tiempo [s]")
ylabel("Frecuencia [Hz]")
title("Espectrograma")
c=colorbar;
c.Label.String="PSD [V^2/Hz]"

```

```

c =
  ColorBar (PSD [V^2/Hz]) with properties:

    Location: 'eastoutside'
    Limits: [-133.0919 -34.3575]
    FontSize: 9
    Position: [0.8482 0.1095 0.0286 0.8155]
    Units: 'normalized'

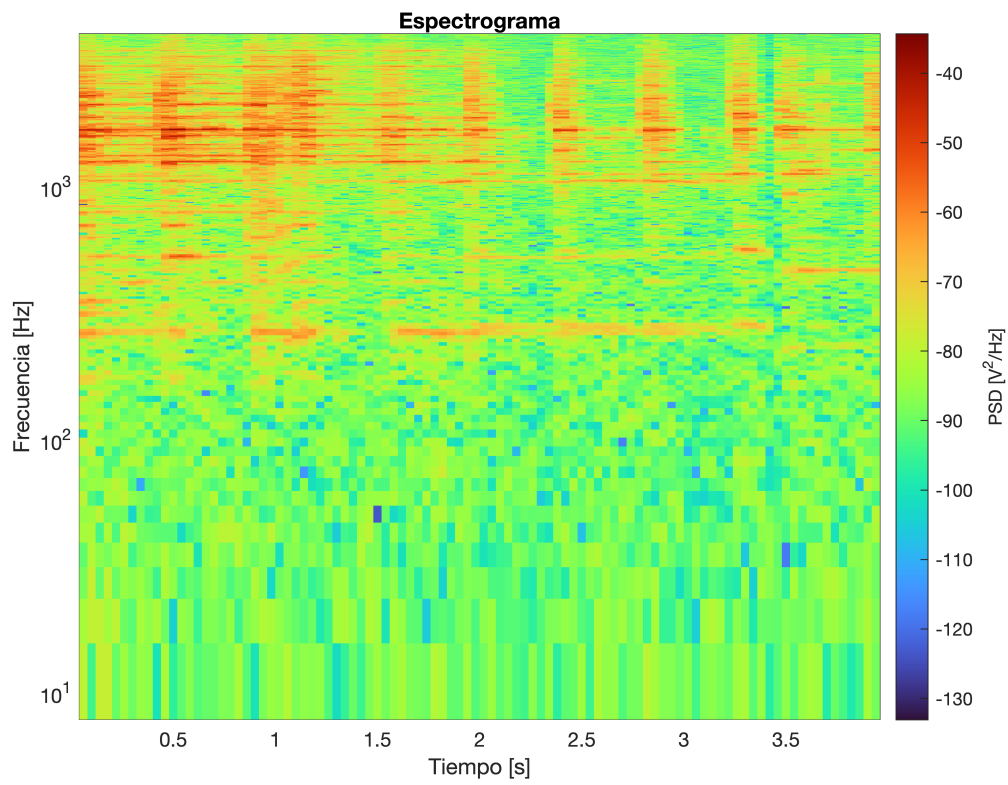
```

Show all properties

```

colormap('turbo')
set(gca,'YScale','log')

```



Guardar Audio

```
%Guardar Audio
audiowrite("Output\voz.wav",data_grabacion,fs)
%Guardar Matrices Writematrix
%guardar tablas Writetable
```