UPF / **Sinais & Sistemas I** **Trabalho II** [13/12/2023]
Prof. Fernando Passold

Trabalho II

1) Trace esboços (assíntotas) para os diagramas de Bode associados com as funções transferência abaixo:

   a) $G_a(s) = \dfrac{5059{,}6}{(s+40)(s+400)}$

   b) $G_b(s) = \dfrac{20(s+20)}{(s+2)(s+200)}$

   c) $G_c(s) = \dfrac{100}{s^2+2s+100}$

   d) $G_d(s) = \dfrac{s^2+8s+1600}{(s+4)^2(s+100)}$

   Obs.: Cada item deve mostrar o cálculo da linha de base (ou do ganho DC), e nos esboços devem ser destacados as assíntotas associadas com cada polo e zero da função transferência, além da curva final. Para alguns casos, iniciar o diagrama de Bode usando como linha de base = 0 dB.

2) Reproduza a forma de onda (no domínio tempo) resultante na saída das funções transferências ressaltadas abaixo quando submetidas a um sinal senoidal de entrada oscilando à frequência de ω_{in} (rad/s), defasado de 0^o (zero graus):

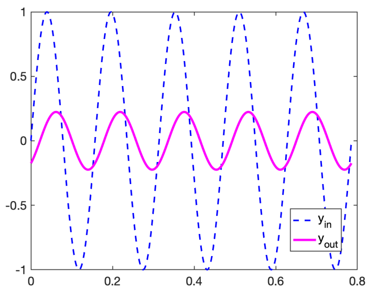
   a) $G_a(s)$ com: $\omega_{in} = 10$ rad/s.

   b) $G_a(s)$ com: $\omega_{in} = 100$ rad/s

   (c) $G_b(s)$ com: $\omega_{in} =$ 7,071 rad/s.

   Acompanhando cada resposta, os cálculos de ganho e defasagem devem ser apresentados; você pode ressaltar nos Diagramas de Bode anteriores, os pontos nos diagramas de magnitude e de fase que correspondem às frequências de entrada citadas para cada função transferência, informando o ganho e defasagem que o sinal de entrada sofrerá em cada caso.

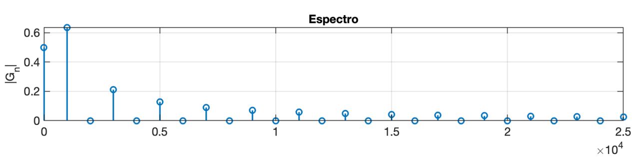
   Por exemplo, se no caso da função $G_a(s)$, fosse inserida uma senoide oscilando à 40 rad/s, isto é, $y_{in}(t) = 1{,}0 \cdot \sin(40\,t)$, o seguinte sinal de saída seria obtido: $y_{out}(t) = 0{,}22387 \cdot \sin(40\,t - 0.88837)$, resultando nas formas de onda mostradas na figura ao lado.



3) Sintetize uma onda quadrada de 1 KHz, usando série de Fourier (mostre seu gráfico):
   a) Até sua 5a harmônica;
   b) Até sua 25a-harmônica.

   Compare lado à lado as figuras geradas nos itens (a) e (b).
   Obs.: Os itens (a) e (b) devem mostrar um diagrama no tempo da forma de onda resultante e o espectro da onda (somente parte de magnitudes absolutas), além de uma tabela mostrando as amplitudes calculadas de cada harmônica. Um gráfico semelhante ao mostrado abaixo deverá ser gerado

Figura exemplo questão 3: Resultados esperados para uma onda quadrada de 1 KHz, 25 harmônicas:



Espectro.

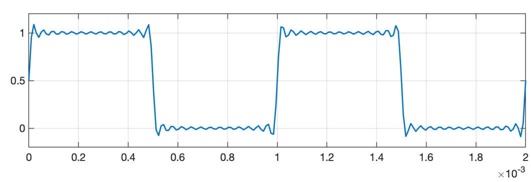
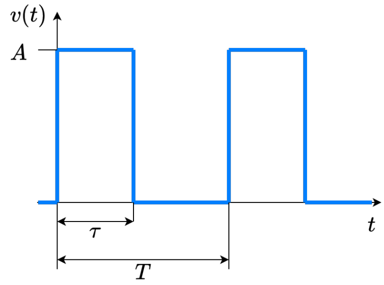


Diagrama no tempo.

Obs.: para o gráfico do espectro do sinal foi usada a função `stem()` do Matlab, algo como: `stem(f, mag)`. Ela funciona tal qual a função `plot()` com a diferença de que gera barras verticais com marcadores nos pontos repassados para a mesma.

Continua ➡

4) Calcule os termos a_0 (valor médio ou "ganho DC"), a_n (termos dos senos) e b_n (termos dos cossenos) da Série de Fourier para uma onda retangular do tipo mostrada na figura ao lado. Esta onda é periódica, de amplitude A, de período T, com largura de pulso τ (*tau*). O parâmetro τ pode ser considerado como um valor percentual de T, ou seja, $\%\,\tau$ (representando o *duty-cycle* de onda quadrada por exemplo).



Para esta onda usar a equação:

$$v(t) = \begin{cases} A & \text{, para: } 0 \le t \le \tau \\ 0 & \text{, para: } \tau \le t \le T \end{cases}$$

Lembrando que os termos da série podem ser calculados como:

$a_0 = \dfrac{\text{Área dentro do periodo, } T}{\text{periodo, } T}$, o que neste caso rende: $a_0 = \dfrac{A \cdot d}{T}$;

$a_n = a_n = \dfrac{2}{T}\displaystyle\int_{t_0}^{t_0+T} f(t)\,\cos(n\,\omega_0\,t)$; que neste caso inicia como: $a_n = = \dfrac{2}{T}\displaystyle\int_0^d A\,\cos(n\cdot\omega_0\cdot t)$

$b_n = \dfrac{2}{T}\displaystyle\int_{t_0}^{t_0+T} f(t)\,\sin(n\,\omega_0\,t)$; que neste caso inicia com: $b_n = \dfrac{2\,A}{T}\displaystyle\int_0^d \sin\left[\left(n\,\dfrac{2\pi}{T}\right)t\right]$

5) Usando as equações para a_0, a_n e $b_n$obtidas no item anterior para a série de Fourier, mostrar um gráfico espectral de magnitude e de fase para esta onda onda e um diagrama no tempo para até 2 ciclos do sinal, quando sua frequência for de 1 KHz, o *duty-cycle* de 25%, amplitude de 1,0 Volt e considerando até 10 harmônicas ($h = 10$).

Lembrando da definição da Série de Fourier:

$$f(t) = a_0 + \sum_{n=1}^{h} a_n\cos(n\,\omega_0 t) + \sum_{n=1}^{h} b_n\sin(\omega_0 t),$$

Onde: ω_0 = frequência fundamental da onda (em rad/s), ou: $\omega_0 = 2\pi f$; f = frequência (fundamental) da onda em Hz; n = número da harmônica.

Neste caso a Magnitude c_n pode ser calculada como:

$$c_n = \sqrt{a_n^2 + b_n^2}$$

A fase pode ser determinada como:

$\theta_n = -\tan^{-1}\left(\dfrac{b_n}{a_n}\right)$ (lembrar que a resposta será em rad/s no Matlab!)

Deverá ser criada uma figura como a mostrada abaixo

Espectro:

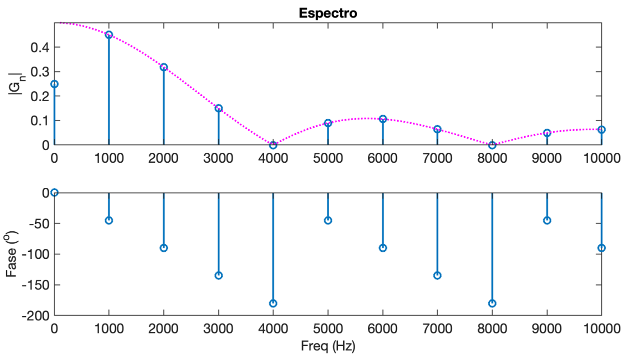
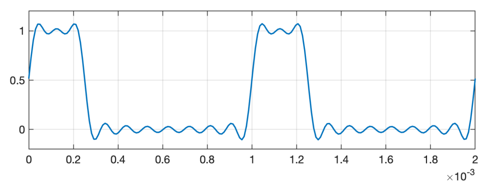


Diagrama no tempo:



Alguns resultados esperados:

Nível DC = $a_0 = \dfrac{A\cdot d}{T} = \dfrac{1\cdot 0.00025}{0.001} = 0.25$

n	Freq	a_n	b_n	$\lvert G_n\rvert$	$\angle_n$
1	1000	0.31831	0.31831	$\cong 0$	-45°
2	2000	$\cong 0$	0.31831	$\cong 0$	-90°
3	3000	-0.106103	0.106103	$\cong 0$	-135°
4	4000	$\cong 0$	$\cong 0$	$\cong 0$	-180°
5	5000	0.063662	0.063662	$\cong 0$	-45°
6	6000	$\cong 0$	0.106103	$\cong 0$	-90°
7	7000	-0.0454728	0.0454728	$\cong 0$	-135°
8	8000	$\cong 0$	$\cong 0$	$\cong 0$	-180°
9	9000	0.0353678	0.0353678	$\cong 0$	-45°
10	10000	$\cong 0$	0.063662	$\cong 0$	-90°

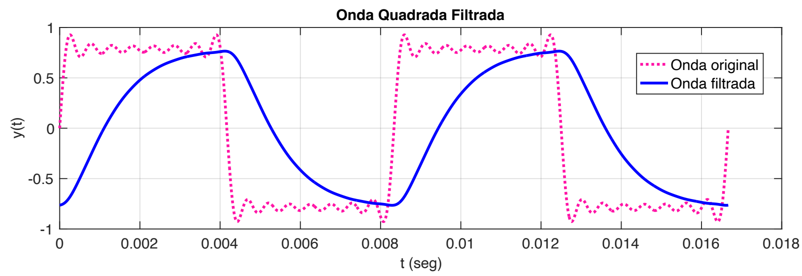
Obs.: Sugere-se fortemente uma consulta ao Cap. 15 (Série de Fourier e Transformada de Fourier, pág. 674 – 682, de DORF, Richard C,; SVOBODA, James A.; **Introdução aos Circuitos Elétricos**, 8a-ed, LTC, 2012)

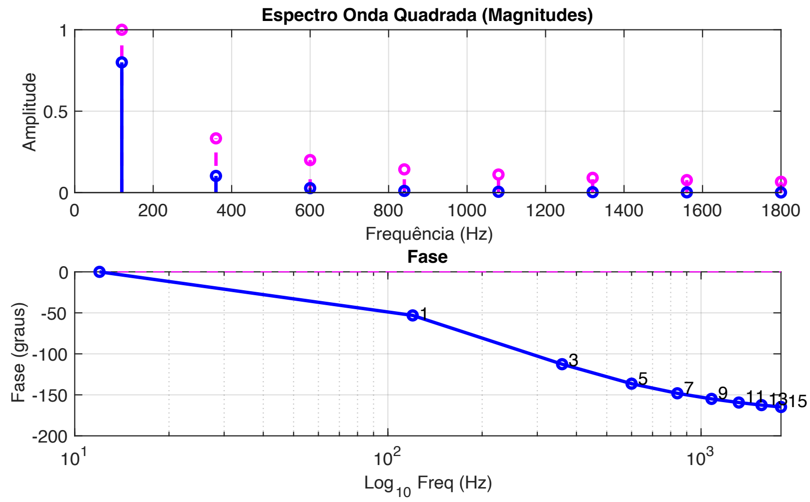
Continua ➠

6) Simule uma onda quadrada oscilando em 120 Hz passado por um filtro passa-baixas Butterworth de 4a-ordem com frequência de corte em $f_c = 240$ Hz. Apresente um diagrama no tempo mostrado na mesma figura a onda quadrada original e o sinal filtrado. Mostre também o espectro original e filtrado da onda quadrada.
Observações: sintetize a onda quadrada (*duty-cycle* = 50%) usando a série de Fourier já calculada no item (3) e aplique a função transferência do filtro (equações para magnitude e fase) nas frequências não nulas das harmônicas já calculadas. Como são esperados um gráfico espectral e um diagrama no tempo, limite o número de harmônicas em até 15 (ou seja, a última harmônica (ímpar) ocorre na frequência $f = 1{,}8$ KHz). Apresente uma tabela mostrando a amplitude original da onda quadrada em cada uma de suas harmônicas (em Volts) e depois outras colunas mostrando a atenuação (em dB) e defasagem (em graus) geradas pelo filtro em cada frequência. Em seguida, outras colunas podem ser acrescentadas mostrando a amplitude final de cada harmônica (em Volts). A seguir, gere um diagrama espectral mostrando como o filtro atua e um diagrama temporal mostrando o resultado da passagem deste sinal pelo filtro.

Espera-se que o estudante obtenha algo como mostrado nas próximas figuras:

Atenção: as figuras ao lado mostram o caso de uma onda quadrada oscilando à 120 Hz (sintetizada até sua 15ª-harmônica), sobre a qual foi aplicado um filtro passa-baixas de 2a-ordem com frequência de corte em 240 Hz.





Obs.: O filtro Butterworth de 4a-ordem com frequência de corte em $f_c = 240$ Hz, pode ser determinado/calculando usando-se Matlab:

```
>> fc=240;
>> wn=2*pi*fc
wn =
        1508
>> [b,a] = butter(4,wn,'low','s');
>> G=tf(b,a);
>> zpk(G)

                       5.1709e+12
  -----------------------------------------------
  (s^2 + 2786s + 2.274e06) (s^2 + 1154s + 2.274e06)

>>
```

A Magnitude em determinada frequência pode ser calculada usando a função `abs()` e a fase, usando-se a função `angle()`.

Fim.Obs.: Seguem Anexos que podem auxiliar a resolver os problemas anteriores.

ANEXO A:
Série de Fourier e Resposta em Frequência

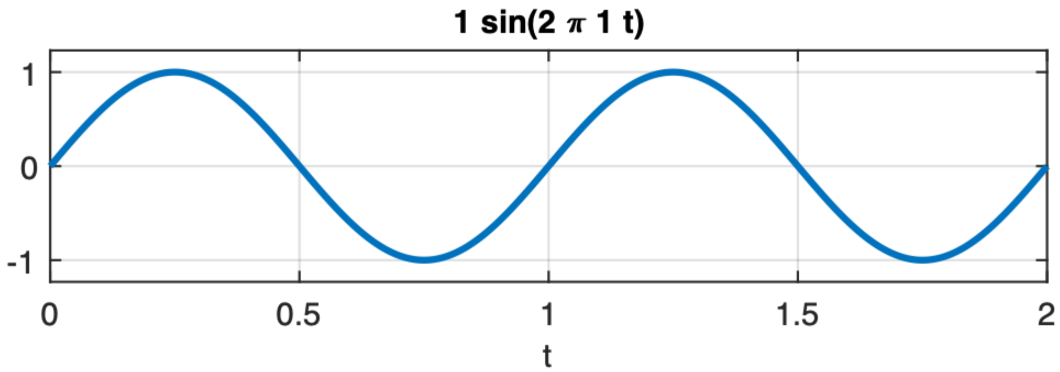
Note que uma onda quadrada pode ser sintetizada através da série de Fourier. Neste caso, esta forma de onda só possui harmônicas ímpares com amplitudes decrescentes conforme aumenta o número da harmônica, ou seja:

$$f(t) = \underbrace{sin(2\pi f \cdot t)}_{1^a\ harmônica} + \underbrace{\frac{1}{3} sin(3 \cdot 2\pi f \cdot t)}_{3^a\ harmônica} + \underbrace{\frac{1}{5} sin(5 \cdot 2\pi f \cdot t)}_{5^a\ harmônica} + \underbrace{\frac{1}{7} sin(7 \cdot 2\pi f \cdot t)}_{7^a\ harmônica} + \cdots$$

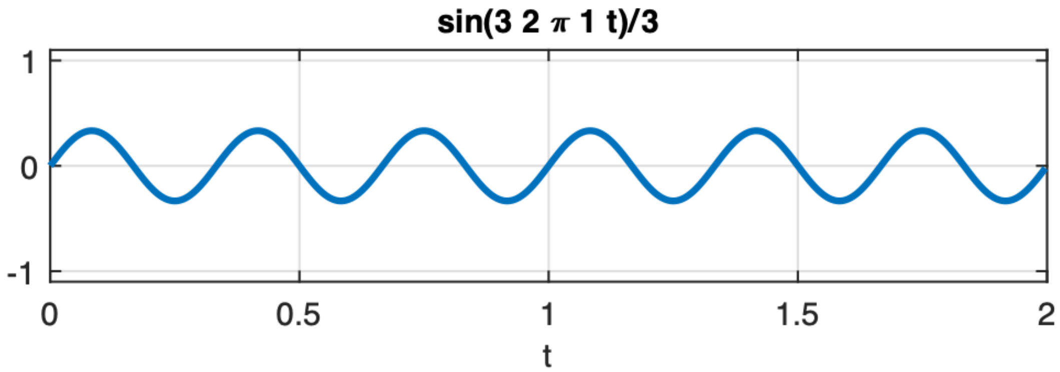
Este tipo de sinal é considerada uma função ímpar, por isto só apresenta componentes harmônicas ímpares.

Note como este sinal é "formado":

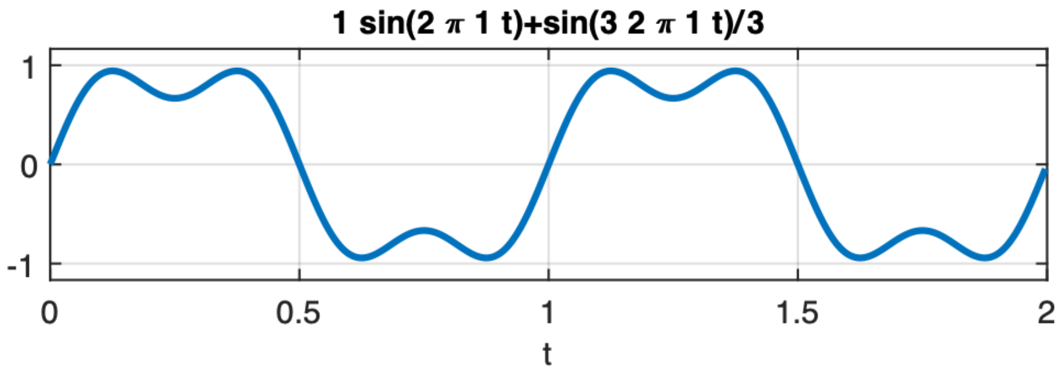
a) Segue forma de onda para: $f(t) = 1 sin(2\pi \cdot 1 \cdot t)$:



b) Segue forma de onda para $f(t) = sin(3 \cdot 2\pi \cdot 1 \cdot t)/3$:

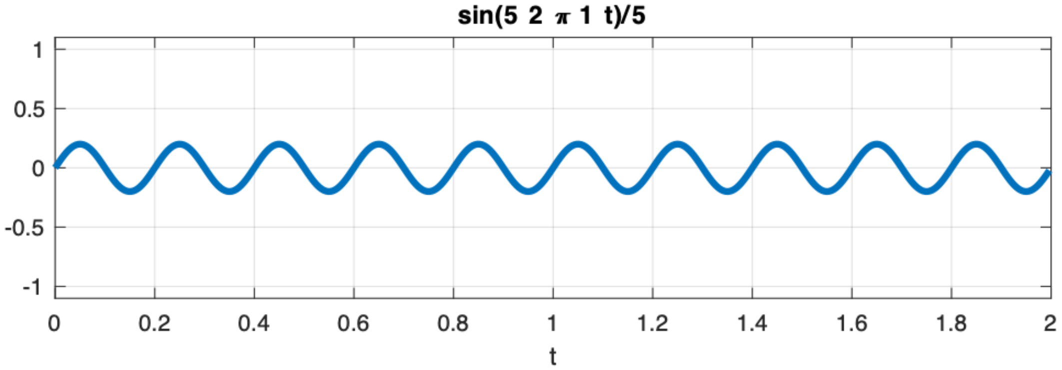


c) Segue forma de onda para: $f(t) = sin(2\pi \cdot 1 \cdot t) + sin(3 \cdot 2\pi \cdot 1 \cdot t)/3$:

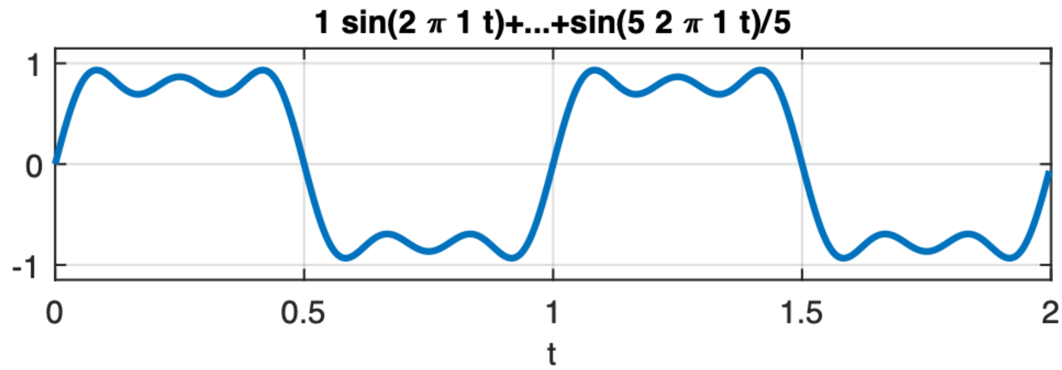


Continua:

d) Segue forma de onda para $f(t) = sin(5 \cdot 2\pi \cdot 1 \cdot t)/5$:

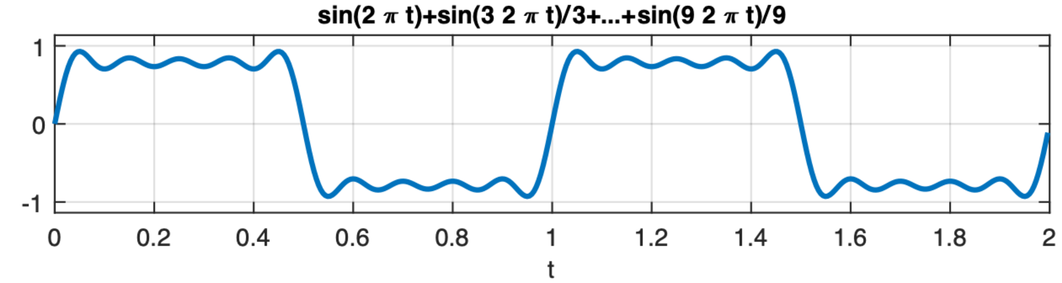


e) Segue forma de onda para $f(t) = sin(2\pi \cdot 1 \cdot t) + sin(3 \cdot 2\pi \cdot 1 \cdot t)/3 + sin(5 \cdot 2\pi \cdot 1 \cdot t)/5$:



Note que estamos nos aproximando do formato esperado para uma onda quadrada. No último caso, sintetizamos apenas 3 termos da série, ou apenas até a 5a-harmônica de onda quadrada.

Se sintetizarmos uma onda quadrada usando até 9 harmônica (apenas 7 componentes da série) teremos:



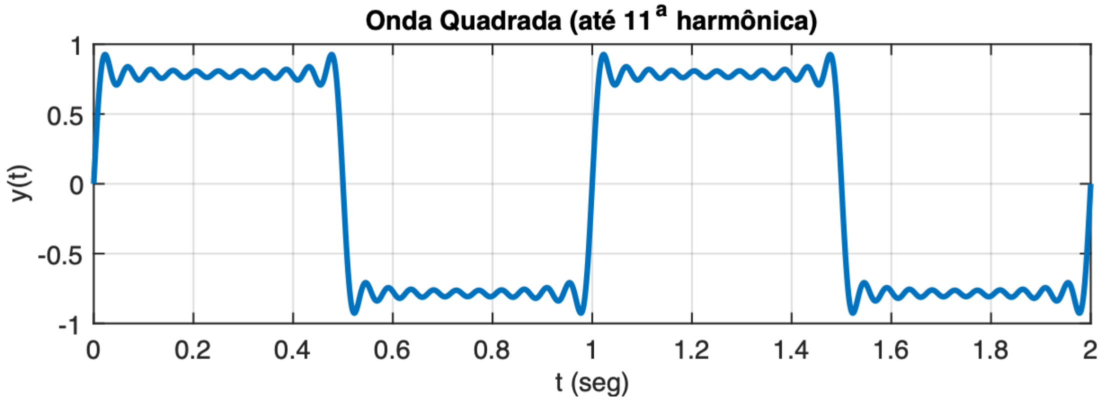
Note que a onda quadrada pode ser sintetizada à partir de uma equação como:

$$f(t) = \sum_{i=1}^{\infty} \frac{1}{(2i-1)} \cdot sin[(2i-1) \cdot 2\pi \cdot f \cdot t]$$

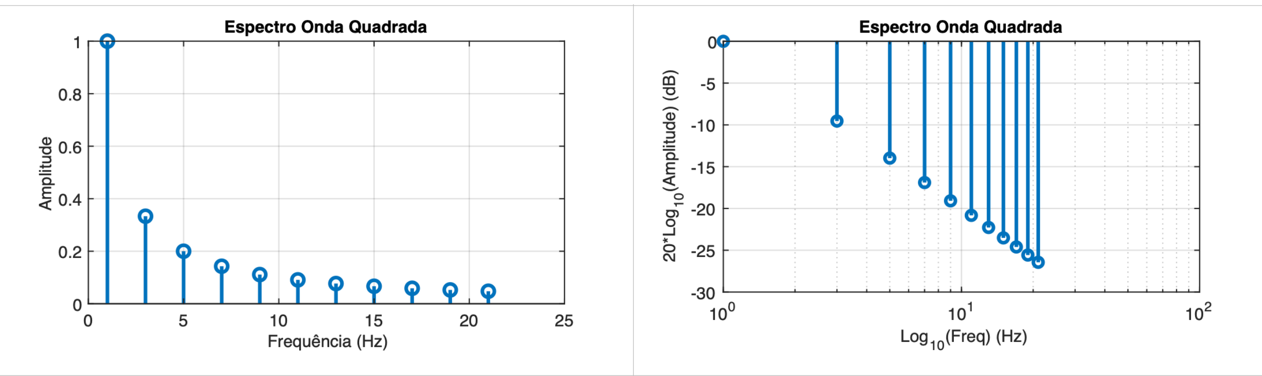
Onde $i = 1,2,3,\dots,\infty$ (componente da série).

[Ref.: https://www.mathsisfun.com/calculus/fourier-series.html (Acessado em 17/06/2022)]

Sintetizando até a 11-harmônica (apenas 6 termos da série), obteremos um sinal como mostrado na próxima figura:

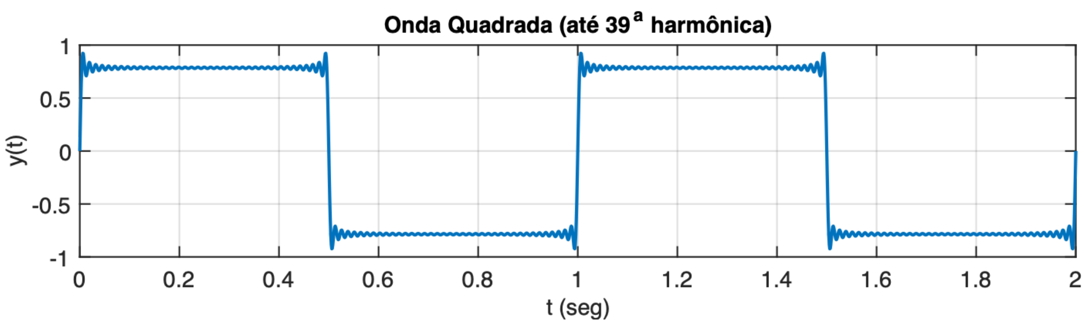


Note o espectro do sinal referente à figura anterior (onda quadrada até 11ª harmônica):

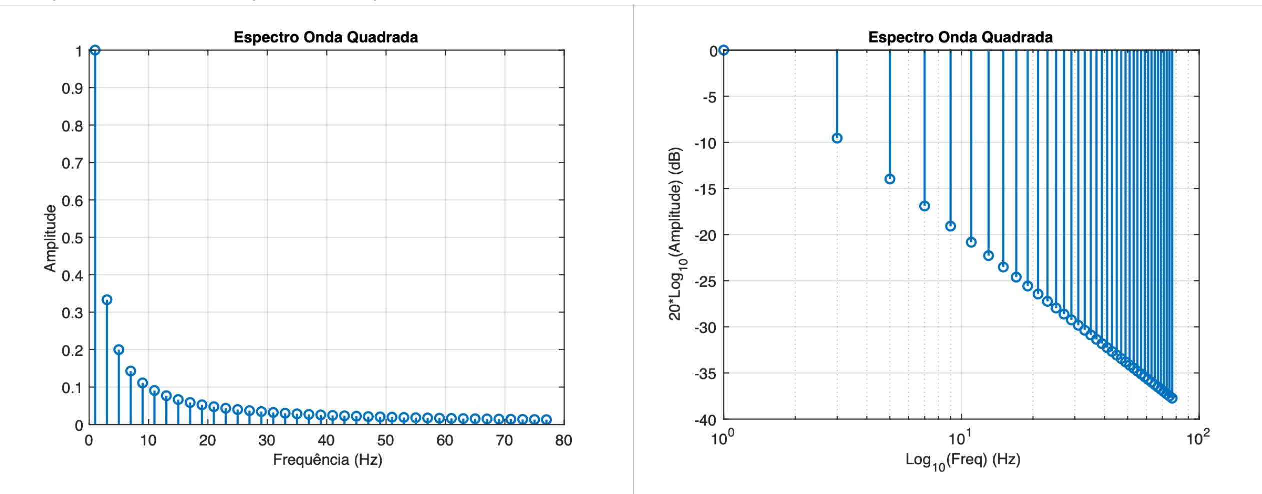


(a) usando escalas lineares.	(b) usando escalas logarítmicas.

Sintetizando-se os primeiros 20 termos da série (até a 39ª harmônica), obtemos o seguinte sinal:



E os espectros deste sinal aparecem na próxima figura:



(a) Usando escalas lineares.	(b) Usando escalas logarítmicas.

Suponha agora que se este sinal será amostrado, numa frequência duas vezes maior que a frequência fundamental da onda quadrada, isto é, $f_s =$2 Hz. Suponha ainda que antes do conversor A/D que amostrará este sinal, existe um filtro passa baixas com frequência de corte igual à frequência de amostragem adotada: $f_c = f_s =$2 Hz.

Suponha que este filtro passa-baixas (necessário em todos circuitos analógicos de digitalização de sinais) seja de 2a-ordem, ou de -40 db/déc.

Com base nos espectros da onda quadrada levantados anteriormente, esboce um diagrama temporal que mostre o resultado da digitalização da onda quadrada sob estas condições.

Solução:
A função transferência de um filtro passa-baixas passivo de 1a-ordem, segue a função transferência:

$$H(s) = \frac{\omega_c}{s + \omega_c} = \frac{1}{1 + s/\omega_c}$$

No nosso caso, onde $f_c =$2 Hz, levaria à ($\omega_c = 2\pi f_c = 2\pi \cdot 2 = 12{,}566$ rad/s):

$$H_1(s) = \frac{12{,}566}{s + 12{,}566}$$

Note que o ganho DC (ganho de "base") resulta em:

$$y(\infty) = \lim_{s \to 0} s \cdot \underbrace{\left(\frac{1}{s}\right)}_{Degrau} \cdot H_1(s) = \lim_{s \to 0} s \cdot \left(\frac{1}{s}\right) \cdot \left(\frac{12{,}566}{s + 12{,}566}\right) = 1$$

Este tipo de filtro pode ser obtido na prática usando um simples circuito (passivo) RC, com o capacitor conectado ao terra.

Um filtro passa-baixas simples de 2a-ordem pode ser obtido simplesmente, cascateando-se um filtro passa baixas de 1a-ordem em seguida de outro, ou na prática, um circuito RC após outro.

Do ponto de vista de diagramas de blocos, o cascateamento de filtros desta forma leva à:

$H_f(s) = H_1(s) \cdot H_1(s)$, ou seja:

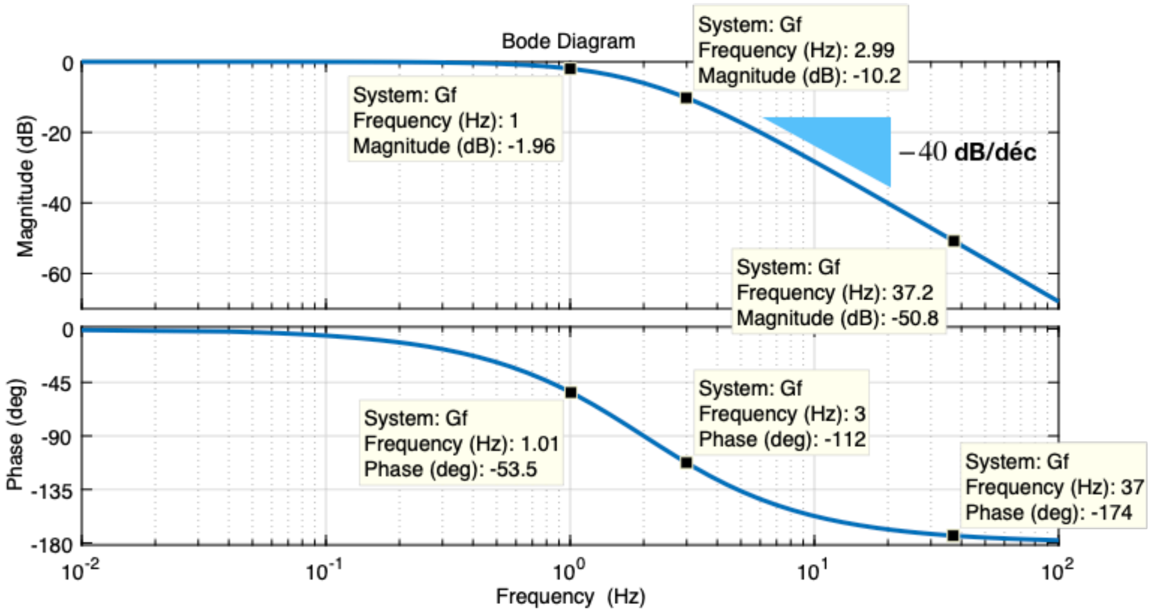
$$H_f(s) = \frac{\omega_c^2}{(s + \omega_c)^2}$$

No nosso caso, levando à:

$$H_f(s) = \frac{157{,}91}{(s + 12{,}57)^2}$$

Note o diagrama de Bode deste filtro:

Diagrama de Bode do filtro passa-baixas passivo de 2a-ordem:



Note que o sistema de aquisição (digitalização) de um sinal analógico pode ser visto como:

O que acontece então quando introduzimos na entrada deste sistema, nossa onda quadrada de 1 Hz?

Solução:
Basta multiplicar o resultado da função transferência (ou espectro do sinal) da onda quadrada, pela função transferência de um filtro passa-baixa de 2a-ordem. Neste caso, no domínio frequência, basta multiplicar a amplitude que a onda quadrada deveria manter nas suas harmônicas vezes o "ganho" (atenuação) causada pelo filtro naquela frequência. Adicionalmente, deve-se levar em conta a defasagem (atraso no tempo) causada pelo filtro nos componentes do sinal original. Neste caso, cada harmônica do sinal original (seu seno) vai ser deslocado pela defasagem causada pelo filtro naquela frequência. Estes cálculos podem ser facilitados, se montarmos uma tabela comparando o sinal original x sinal filtrado (amplitudes e defasagens de cada harmônica), conforme mostrado na próxima tabela.

Cálculos:

$$H(s) = \frac{\omega_c^2}{(s+\omega_c)^2} = \frac{\omega_c^2}{s^2 + 2\omega_c s + \omega_c^2}$$

$$H(j\omega) = \frac{\omega_c^2}{(j\omega+\omega_c)^2} = \frac{\omega_c^2}{(\omega_c^2 - \omega^2) + j2\omega_c\omega}$$

$$|H(j\omega)| = \omega_c^2 / \sqrt{(\omega_c^2 - \omega^2)^2 + (2\omega_c\omega)^2}$$

$$|H(j\omega)| = \omega_c^2 / \sqrt{4\omega^2\omega_c^2 + (\omega^2 - \omega_c^2)^2}$$

$$\angle G(j\omega) = -tan^{-1}\left(\frac{2\omega_c\omega}{\omega_c^2 - \omega^2}\right)$$

Aplicando as equações anteriores para descobrir de que forma o filtro "deforma" o sinal de entrada nas harmônicas componentes da onda quadrada, podemos levantar a seguinte tabela:

Segue tabela relacionando frequências x ganho x defasagens geradas pelo FPB:

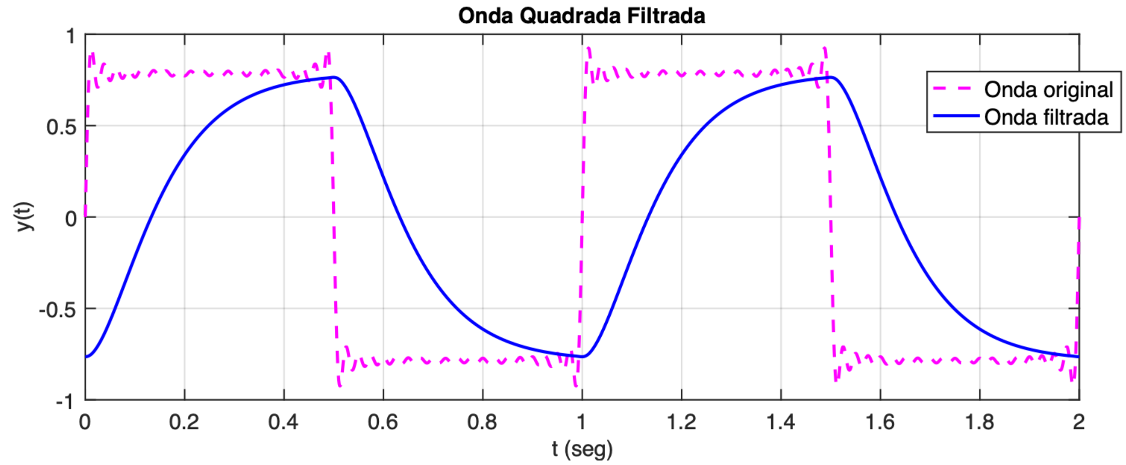
Hz	Ganho	Atenuação (dB)	Defasagem
1	0.8	-1.9382 dB	-53.1301^o
3	0.307692	-10.2377 dB	-112.62^o
5	0.137931	-17.2068 dB	-136.397^o
7	0.0754717	-22.4443 dB	-148.109^o
9	0.0470588	-26.5472 dB	-154.942^o
11	0,032	-29.897 dB	-159.39^o
⋮	⋮	⋮	⋮
35	0.00325468	-49.7498 dB	-173.459^o
37	0.00291333	-50.7122 dB	-173.812^o
⋮	⋮	⋮	⋮
75	0.000710606	-62.9674 dB	-176.945^o
77	0.000674195	-63.4243 dB	-177.024^o

Com os dados desta tabela (gerados usando rotina 'FPB2a_ordem.m' do Matlab), pode-se montar outra tabela que reflete o impacto do filtro atuando sobre os componentes da onda quadrada na sua entrada.

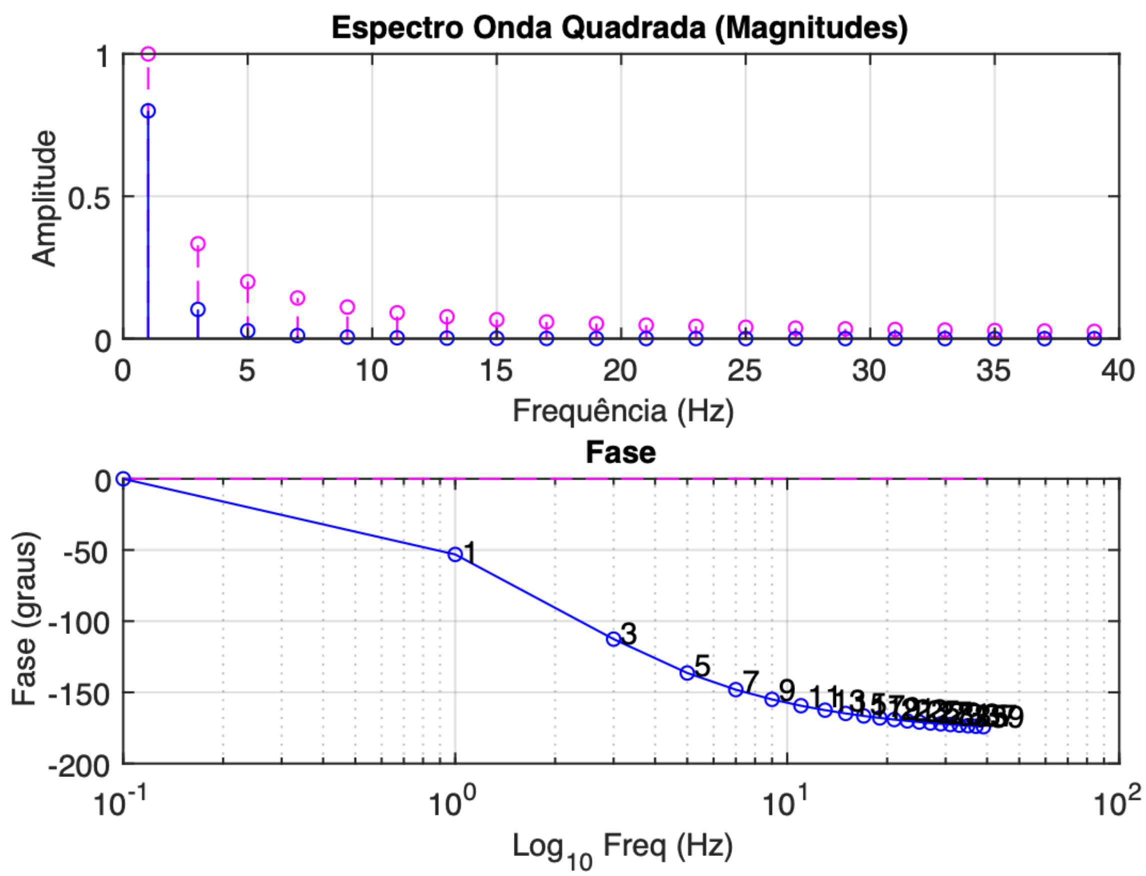
Freq (Hz)	Amplitude Original	Ganho Filtro	Atenuação Filtro	Amplitude Final	Defasagem
1	1	0.8	-1.9382 dB	0.8	-53.1301°
3	0.333333	0.307692	-10.2377 dB	0.102564	-112.62°
5	0.2	0.137931	-17.2068 dB	0.0275862	-136.397°
7	0.142857	0.0754717	-22.4443 dB	0.0107817	-148.109°
9	0.111111	0.0470588	-26.5472 dB	0.00522876	-154.942°
11	0.0909091	0,032	-29.897 dB	0.00290909	-159.39°
⋮	⋮	⋮	⋮	⋮	⋮
35	0.0285714	0.00325468	-49.7498 dB	9.29908e-05	-173.459°
37	0.027027	0.00291333	-50.7122 dB	7.87386e-05	-173.812°
⋮	⋮	⋮	⋮	⋮	⋮
75	0.0133333	0.000710606	-62.9674 dB	9.47474e-06	-176.945°
77	0.012987	0.000674195	-63.4243 dB	8.75578e-06	-177.024°

Transformando os dados anteriores num gráfico temporal, obtemos a próxima figura:

Resultado da onda quadrada de 1 Hz após passar pelo filtro passa-baixas com f_c =2 Hz:



Modificações geradas no espectro frequência:



Os diagramas espectrais ajudam a entender de que forma as componentes originais da onda quadrada foram afetadas pelo filtro passa-baixas.

ANEXO B:
Rotinas Matlab:

Forma de onda fig (a):
```
>> ezplot(@(t) 1*sin(2*pi*1*t), [0 2])
>> grid
```

Forma de onda para fig (c):
```
>> ezplot(@(t) sin(3*2*pi*1*t)/3, [0 2])
>> axis([0 2 -1.1 1.1])
>> grid
```

Forma de onda fig (c):
```
>> ezplot(@(t) 1*sin(2*pi*1*t) + sin(3*2*pi*1*t)/3, [0 2])
>> grid
```

Forma de onda fig (e):
```
>> ezplot(@(t) 1*sin(2*pi*1*t) + sin(3*2*pi*1*t)/3 + sin(5*2*pi*1*t)/5, [0 2])
>> grid
```

Cálculos relacionados com filtro passivo passa-baixa de 1a-ordem:
```
>> fc=2;
>> wc=2*pi*fc % freq. de corte em rad/s
wc =       12.566
>> G1=tf(wc,[1 wc])

G1 =
    12.57
  ---------
  s + 12.57

>> Gf=G1*G1 % simples blocos em série. FPB de 2a-ordem
Gf =
         157.9
  --------------------
  s^2 + 25.13 s + 157.9

>> zpk(Gf)

    157.91
  -----------
  (s+12.57)^2

>> figure; bode(Gf)
>> grid
>> % Bode com frequência em Hz:
>> options = bodeoptions;
>> options.FreqUnits = 'Hz'; % or 'rad/second', 'rpm', etc.
>> figure
>> bode(Gf,options);
>> grid
```

Desenvolvendo expressões com auxílio do Matlab:
```
>> syms w wc
>> aux=(wc^2-w^2)^2+(2*wc*w)^2
aux =
4*w^2*wc^2 + (w^2 - wc^2)^2
>> sqrt(aux)
ans =
(4*w^2*wc^2 + (w^2 - wc^2)^2)^(1/2)
>> aux2=(wc^2)/sqrt(aux)
aux2 =
wc^2/(4*w^2*wc^2 + (w^2 - wc^2)^2)^(1/2)
>> pretty(aux2)
               2
             wc
----------------------------
        2   2     2     2 2
sqrt(4 w  wc  + (w  - wc ) )
```

Script 'onda_quadrada_fourier.m' usada para gerar dados da onda quadrada sintetizada via série de Fourier:

```
% onda_quadrada_fourier.m
% Sintetizando onda quadrada
% Fernando Passold, em 17/06/2022

clear freq G G_dB t y % evitar problemas se re-executar script

disp('Sintese de Onda Quadrada usando Série de Fourier');
f=input('Frequencia da onda quadrada (Hz)? ');
hh=input('Quantos harmonicas para a onda quadrada? ');
n=floor(hh/2)+1; % quantidade de termos da série a serem gerados

T=1/f; % periodo da onda quadrada
fs=f*50*20; % freq. "amostragem" onda quadrada (p/efeitos síntese gráficos)
TT=1/fs;
ciclos=2*T; % número de ciclos plotados da onda quadrada

t=0:TT:ciclos; % forma vetor tempo
u=length(t); % numero de pontos
for k=1:u   % varrendo vetor tempo
    sum = 0;
    for h=1:n
        termo=2*h-1;
        freq(h)=f*termo; % frequencia da harmonica h
        G(h)=1/termo; % amplitude da harmonica h
        G_dB(h)=20*log10(G(h));
        sum = sum + G(h)*sin(2*pi*freq(h)*t(k));
    end
    y(k)=sum;
end
figure;
plot(t,y);
grid
texto=num2str(termo);
title(['Onda Quadrada (até ' texto '^a harmônica)']);
xlabel('t (seg)');
ylabel('y(t)');

% Espectro da onda quadrada gerada
figure;
stem(freq, G); % note que são "impulsos" em freq's ímpares
title('Espectro Onda Quadrada');
xlabel('Frequência (Hz)');
ylabel('Amplitude');
grid

figure;
stem(freq, G_dB); % note que são "impulsos" em freq's ímpares
title('Espectro Onda Quadrada');
xlabel('Log_{10}(Freq) (Hz)');
ylabel('20*Log_{10}(Amplitude) (dB)');
set(gca, 'XScale', 'log')
grid
```

```
% FPB2a_ordem.m
% Sintetizando Filtro Passa Baixas (passivo) de 2a-ordem
% Este filtro é resultado do cascateamento em série de 2 filtros de
% 1a-ordem do tipo:
%   H(s)=wc/(s+wc)
% o que resulta, num filtro de 2a-ordem com:
% Magnitude:
%   |H(jw)|= wc^2/sqrt[(wc^2-w^2)^2+(2*wc*w)^2]
%   /angle G(jw) = atan2(2*wc*w, wc^2-w^2)
% Entrada:
%   freq = vetor com frequencias que devem ter calculadas |G(jw)| /angle
% Obs: dado anterior gerado usando 'onda_quadrada_fourier.m' (executar
% antes).
% Saída gerada:
%   wc = freq de corte do filtro em rad/s
%   Tabela relacionando freq x ganho x defasagem
%
% Fernando Passold, em 18/06/2022

clear Mag Phase % evitar problemas se re-executar script
```

```
fc=input('Freq. de corte do Filtro PB (Hz)? ');

u=length(freq); % elementos presentes dentro vetor freq
wc=2*pi*fc;

% gerando tabela na tela em formato compatível MarkDown (table)
disp('Comportamento do Filtro com:')
fprintf('freq de corte, fc = %g Hz', fc)
fprintf(' (%g rad/s)\n\n', wc)
%disp('\begin{tabular}{rrrr} \hline')
disp('| Freq | Amplitude | Amplitude     | Defasagem |')
disp('| (Hz) | Original  | Original (dB) |  (graus)  |')
disp('| ---: |    ---:   |   ---:        |  ---:     |')

for k=1:u  % varre freq's desejadas
    w = 2*pi*freq(k); % freq em rad/s
    Re = wc*wc - w*w;
    Im = 2*wc*w;
    Mag(k) = (wc*wc)/sqrt( Re*Re + Im*Im ); % ganho absoluto
    Phase(k) = -atan2( Im, Re ); % em rad/s
    fprintf('| %g |', freq(k));
    fprintf(' %g |', Mag(k));
    fprintf(' %g dB |', 20*log10(Mag(k)));
    fprintf(' %g° |\n', Phase(k)*180/pi); % valor em graus
end
% disp('  \hline')
% disp('\end{tabular}')
disp(' ');
```

Script ‘FPB2a_ordem.m' criada para computar atuação do Filtro Passa Baixas:

```
% impacto_FPB.m
% Gera tabela mostrando impacto causado pelo FPB na onda quadrada
% Executar antes: 'FPB2a_ordem.m'
% Fernando Passold, em 18/06/2022

clear amp_orig amp_final y_orig y_filt % evigar problemas se re-executar script

disp('Impacto causado pelo FPB sobre a onda Quadrada');
disp(' ');
u=length(freq); % elementos presentes dentro vetor freq
disp('| Freq (Hz) | Amplitude Original | Ganho Filtro | Atenuação Filtro | Amplitude Final | Defasagem
|');
disp('|   ---:    |       ---:         |   ---:       |    ---:          |    ---:         |   ---:
|');
for k=1:u  % varre freq's desejadas
    fprintf('| %g |', freq(k));
    harmonica(k) = 2*k-1; % determina numero da harmonica
    amp_orig(k) = 1/harmonica(k);
    fprintf(' %g |', amp_orig(k));
    fprintf(' %g |', Mag(k)); % "ganho" do filtro
    fprintf(' %g dB |', 20*log10(Mag(k))); % impacto do filtro em dB
    amp_final(k) = amp_orig(k)*Mag(k);
    fprintf(' %g |', amp_final(k));
    fprintf(' %g° |\n', Phase(k)*180/pi); % valor em graus
end
disp(' ');

% Gera gráfico temporal comparando onda quadrada original x filtrada
% Parte do código baseado em 'onda_quadrada_fourier.m'

fs=f*50*20; % freq. "amostragem" onda quadrada (p/efeitos síntese gráficos)
TT=1/fs;
ciclos=2*T; % número de ciclos plotados da onda quadrada

t=0:TT:ciclos; % forma vetor tempo
t_leng=length(t); % numero de pontos
```

```
for tt=1:t_leng   % varrendo vetor tempo
   sum_orig = 0;
   sum_filt = 0;
   for k=1:u  % varre as freq's das harmonicas
      % freq(h) = frequencia da harmonica k
      sum_orig = sum_orig + amp_orig(k)*sin(2*pi*freq(k)*t(tt));
        sum_filt = sum_filt + amp_final(k)*sin(2*pi*freq(k)*t(tt)+Phase(k));
    end
    y_orig(tt)=sum_orig;
    y_filt(tt)=sum_filt;
end
figure;
plot(t,y_orig,'m--', t,y_filt,'b-');
grid
texto=num2str(n);
title(['Onda Quadrada Filtrada']);
xlabel('t (seg)');
ylabel('y(t)');
legend('Onda original','Onda filtrada')

% Espectro linear da onda quadrada original x filtrada
figure;
subplot(211);
stem(freq, G, 'm--'); % note que são "impulsos" em freq's ímpares
hold on;
stem(freq, amp_final, 'b-');
title('Espectro Onda Quadrada (Magnitudes)');
xlabel('Frequência (Hz)');
ylabel('Amplitude');
grid

subplot(212); % gráfico da Fase
% Acrescentando pontos à esquerda do gráfico numa década inferior
% a da 1a-harmônica da onda quadrada.
clear freq2 Phase2 Ph_original
freq2=[f/10 freq];
Phase2=[0 Phase];
uu=length(freq2);
Ph_original=zeros(1,uu);
plot(freq2, Ph_original*180/pi, 'm--')
hold on
plot(freq2, Phase2*180/pi, 'bo-')
% acrescentando texto com No. da harmônica
for k=1:u
    texto = num2str(harmonica(k));
    text(freq(k)*1.05, 0.95*Phase(k)*180/pi, texto);
end
set(gca, 'XScale', 'log')
title('Fase');
xlabel('Log_{10} Freq (Hz)')
ylabel('Fase (graus)');
```

Script '' usado para observar impacto do filtro sobre a onda quadrada: