
Laboratório 02

Modulação, Demodulação, Multiplexação e Demultiplexação

Curso: Engenharia de Telecomunicações
Disciplina: COM29007 – Sistemas de Comunicação I
Professor: Mário de Noronha Neto

Aluno
Rafael Teles Espindola

1 Introdução

Laboratório realizado com o objetivo de fixar os conteúdos da disciplina: Modulação e Demodulação por Amplitude (AM) e Multiplexação e Demultiplexação no domínio da frequência.

"A modulação em amplitude é definida como o processo pelo qual a amplitude da onda portadora $c(t)$ é variada em torno de um valor médio, linearmente com o sinal da banda base"[Sistemas de Comunicação - S. Haykin e M. Moher]

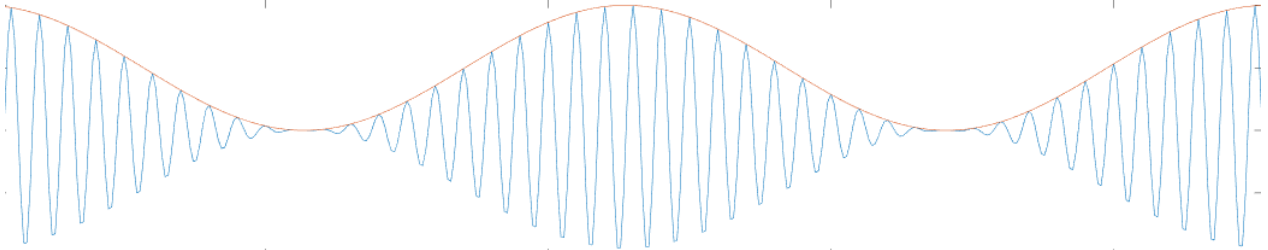


Figura 1: Exemplo de uma modulação AM DSB

2 Exercício 1:

Realizar um processo de modulação AM DSB e AM DSB-SC

Para o caso da modulação AM DSB-SC, realizar o processo de demodulação utilizando a função 'fir1'

Para o caso da modulação AM DSB, variar o 'fator de modulação' (0.25; 0.5; 0.75 e 1 e 1.5) e observar os efeitos no sinal modulado

2.1 Realização do processo de modulação AM DSB e AM DSB-SC

```
1 fs = 90e3;
2 t = 0:1/fs:1-(1/fs);
3
4 Am1 = 1; f1 = 440; Ao1 = 1;           % Parametros AM DSB
5 Am2 = 1; f2 = 770; Ao2 = 0;           % Parametros AM DSB-SC
6 Am3 = 1; f3 = 10e3; Ao3 = 0;          % Parametros Portadora1
7 Am4 = 1; f4 = 15e3; Ao4 = 0;          % Parametros Portadora2
8
9 som1 = Ao1 + Am1*cos(2*pi*f1*t);      % AM DSB
10 som2 = Ao2 + Am2*cos(2*pi*f2*t);      % AM DSB-SC
11 por1 = Ao3 + Am3*cos(2*pi*f3*t);      % Portadora1
12 por2 = Ao4 + Am4*cos(2*pi*f4*t);      % Portadora2
13
14 y1 = por1.*som1;                       % Sinal modulado1 AM DSB
15 y2 = por2.*som2;                       % Sinal modulado2 AM DSB-SC
```

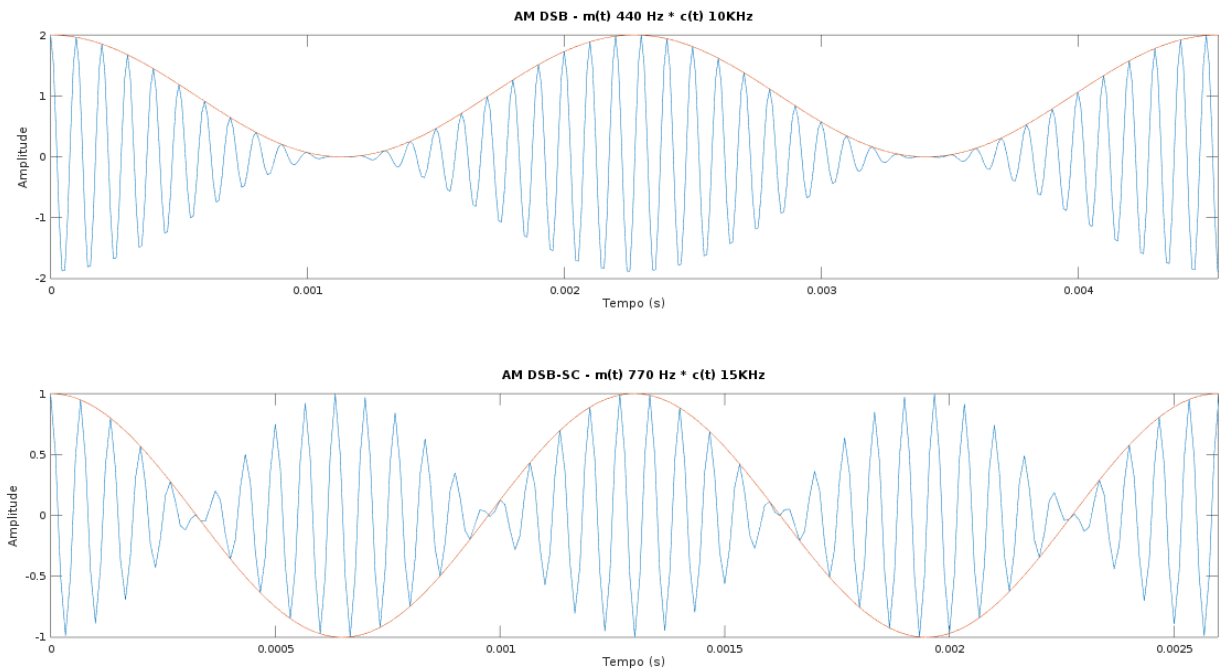


Figura 2: Modulação AM DSB e AM DSB-SC respectivamente

2.2 Realização do processo de demodulação utilizando a função 'fir1'

```

16 y2_demod = por2.*y2; % Sinal demodulado1 AM DSB
17 filtro_pb = fir1(500, (1000*2)/fs); % filtro fir1
18 filtrado = filter(filtro_pb, 1, y2_demod); % Sinal filtrado AM DSB

```

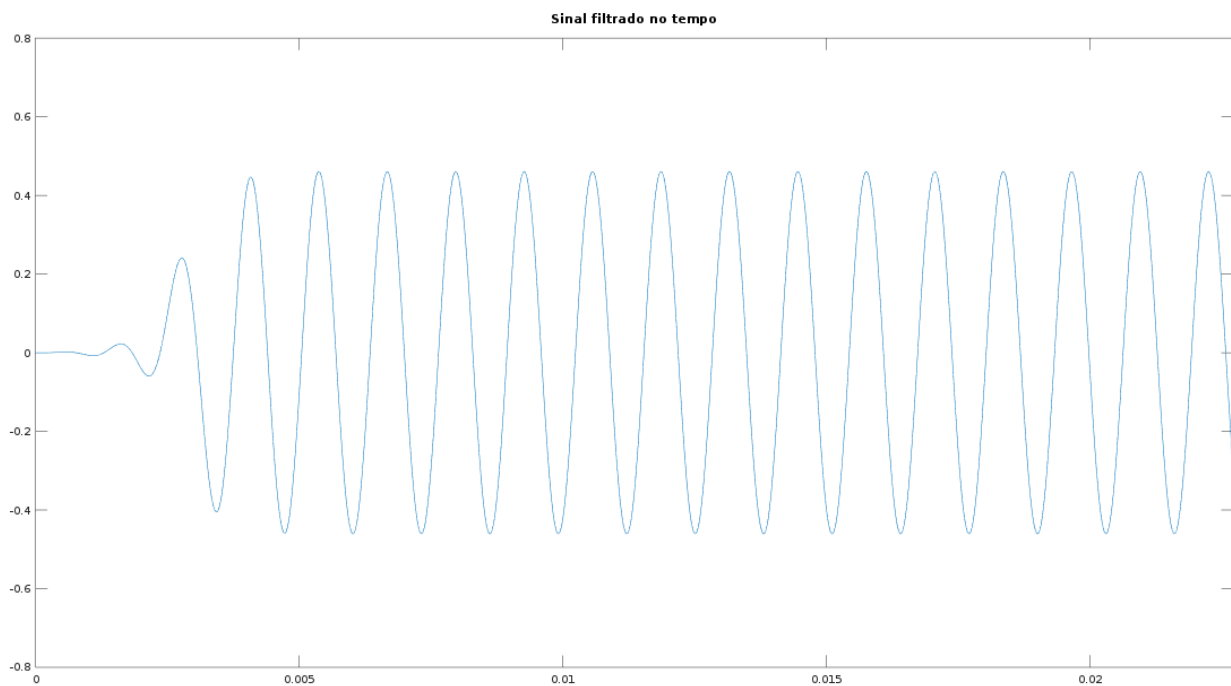


Figura 3: Sinal filtrado com um filtro analógico

2.3 Variação do fator de modulação

```

19 s_t025 = Am3*(1 + 0.25*cos(2*pi*f1*t)).*cos(2*pi*f3*t); % Fator 0.25
20 s_t050 = Am3*(1 + 0.50*cos(2*pi*f1*t)).*cos(2*pi*f3*t); % Fator 0.50
21 s_t075 = Am3*(1 + 0.75*cos(2*pi*f1*t)).*cos(2*pi*f3*t); % Fator 0.75
22 s_t100 = Am3*(1 + 1.00*cos(2*pi*f1*t)).*cos(2*pi*f3*t); % Fator 1.00
23 s_t125 = Am3*(1 + 1.25*cos(2*pi*f1*t)).*cos(2*pi*f3*t); % Fator 1.25
24 s_t150 = Am3*(1 + 1.50*cos(2*pi*f1*t)).*cos(2*pi*f3*t); % Fator 1.50

```

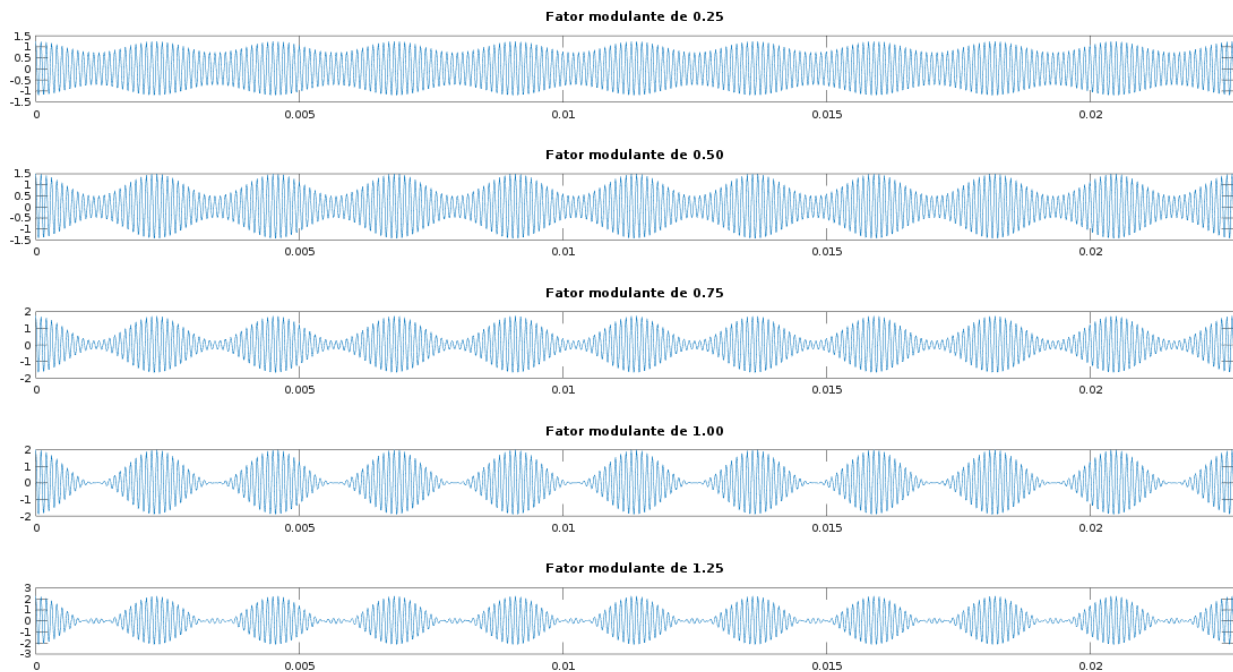


Figura 4: Sinais com diferentes fatores de modulação

3 Exercício 2:

Gerar 3 sinais (cossenos) nas frequências 1k, 2k e 3k

Realizar a multiplexação dos sinais para as frequências 10k, 12k e 14k para a transmissão em um canal de comunicação

Recuperar os sinais originais

3.1 Geração dos sinais (cossenos) nas frequências de 1KHz, 2KHz e 3KHz

```

25 fs = 90e3; % Frequencia de amostragem
26 t = 0:1/fs:1-(1/fs); % Tempo
27
28 Am1 = 1; f1 = 1e3; Ao1 = 0; % Amplitude, frequencia e componente DC do Modulante
29 Am2 = 1; f2 = 2e3; Ao2 = 0; % Amplitude, frequencia e componente DC do Modulante
30 Am3 = 1; f3 = 3e3; Ao3 = 0; % Amplitude, frequencia e componente DC do Modulante
31 Ac1 = 1; f4 = 10e3; Ao4 = 0; % Amplitude, frequencia e componente DC da Portadora
32 Ac2 = 1; f5 = 12e3; Ao5 = 0; % Amplitude, frequencia e componente DC da Portadora
33 Ac3 = 1; f6 = 14e3; Ao6 = 0; % Amplitude, frequencia e componente DC da Portadora
34
35 m_t1 = Ao1 + Am1*cos(2*pi*f1*t); % Sinal Modulante no tempo
36 m_t2 = Ao2 + Am2*cos(2*pi*f2*t); % Sinal Modulante no tempo
37 m_t3 = Ao3 + Am3*cos(2*pi*f3*t); % Sinal Modulante no tempo
38 c_t1 = Ao4 + Ac1*cos(2*pi*f4*t); % Sinal Portadora no tempo

```

```

39 c_t2 = Ao4 + Ac2*cos(2*pi*f5*t); % Sinal Portadora no tempo
40 c_t3 = Ao4 + Ac3*cos(2*pi*f6*t); % Sinal Portadora no tempo
41 s_t1 = m_t1.*c_t1; % Sinal Modulado no tempo
42 s_t2 = m_t2.*c_t2; % Sinal Modulado no tempo
43 s_t3 = m_t3.*c_t3; % Sinal Modulado no tempo
44
45 m_f1 = fft(m_t1); % Transformada de fourier
46 m_f1 = fftshift(m_f1)/length(m_f1); % Sinal c_t1 no domínio da frequencia
47 m_f2 = fft(m_t2); % Transformada de fourier
48 m_f2 = fftshift(m_f2)/length(m_f2); % Sinal c_t2 no domínio da frequencia
49 m_f3 = fft(m_t3); % Transformada de fourier
50 m_f3 = fftshift(m_f3)/length(m_f3); % Sinal c_t1 no domínio da frequencia
51 c_f1 = fft(c_t1); % Transformada de fourier
52 c_f1 = fftshift(c_f1)/length(c_f1); % Sinal c_t1 no domínio da frequencia
53 c_f2 = fft(c_t2); % Transformada de fourier
54 c_f2 = fftshift(c_f2)/length(c_f2); % Sinal c_t2 no domínio da frequencia
55 c_f3 = fft(c_t3); % Transformada de fourier
56 c_f3 = fftshift(c_f3)/length(c_f3); % Sinal c_t1 no domínio da frequencia
57 s_f1 = fft(s_t1); % Transformada de fourier
58 s_f1 = fftshift(s_f1)/length(s_f1); % Sinal s_t1 no domínio da frequencia
59 s_f2 = fft(s_t2); % Transformada de fourier
60 s_f2 = fftshift(s_f2)/length(s_f2); % Sinal s_t2 no domínio da frequencia
61 s_f3 = fft(s_t3); % Transformada de fourier
62 s_f3 = fftshift(s_f3)/length(s_f3); % Sinal s_t1 no domínio da frequencia

```

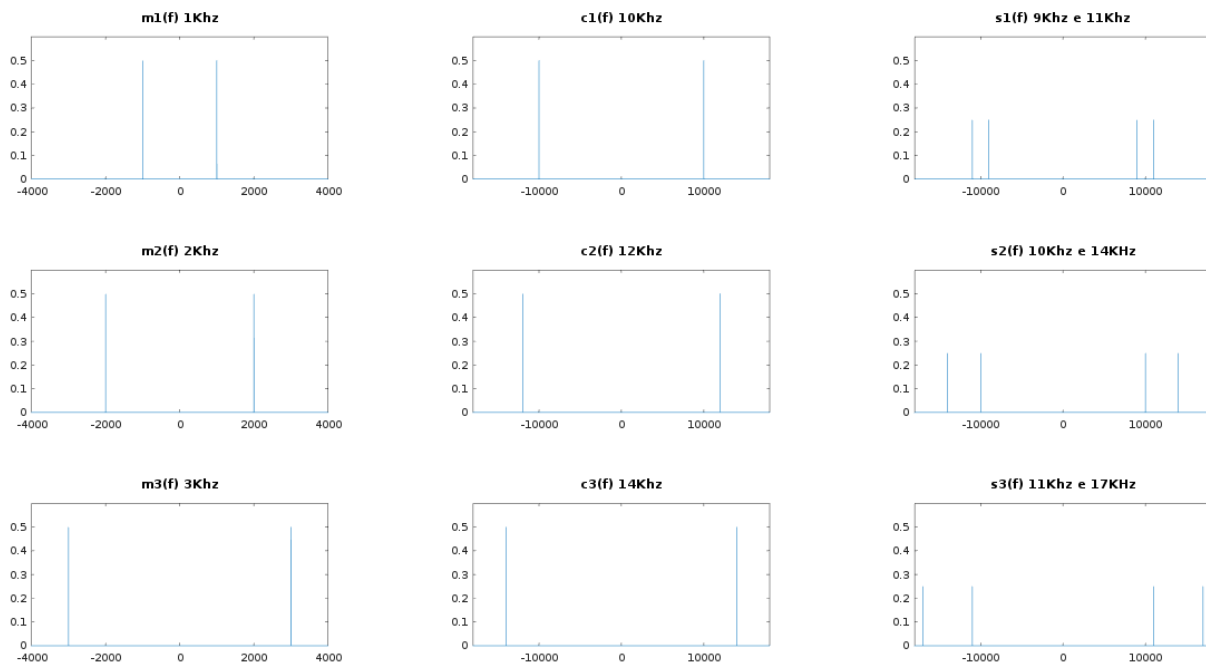


Figura 5

3.2 Realização dos filtros ideais e filtragem

```

63 % Filtros ideais para usar na filtragem no domínio da frequencia
64 filtro_11k = [zeros(1, 33000) ones(1, 2000) zeros(1, 20000) ones(1, 2000) zeros(1, 33000)];
65 filtro_14k = [zeros(1, 30000) ones(1, 2000) zeros(1, 26000) ones(1, 2000) zeros(1, 30000)];
66 filtro_17k = [zeros(1, 27000) ones(1, 2000) zeros(1, 32000) ones(1, 2000) zeros(1, 27000)];
67
68 % Sinal filtrado na frequencia com filtro ideal
69 % Filtrado apenas banda lateral superior
70 s_f1_filtrado = filtro_11k.*s_f1; % Excluindo componente de frequencia de 9KHz
71 s_f2_filtrado = filtro_14k.*s_f2; % Excluindo componente de frequencia de 10KHz

```

```
72 s_f3_filtrado = filtro_17k.*s_f3; % Excluindo componente de frequencia de 11KHz
```

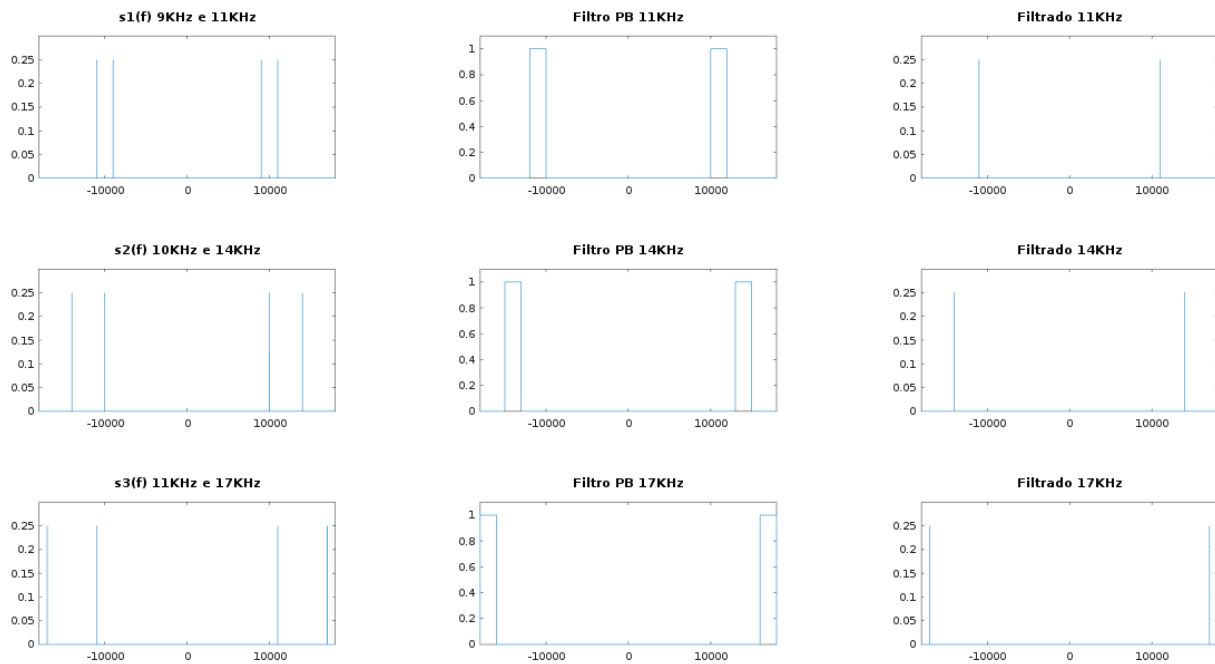


Figura 6: Sinais iniciais

3.3 Sinal a ser transmitido

```
74 % Sinal filtrado na frequencia multiplexado em 3 canais
75 % Este eh o sinal a ser transmidito pela antena
76 y_f = s_f1_filtrado + s_f2_filtrado + s_f3_filtrado;
```

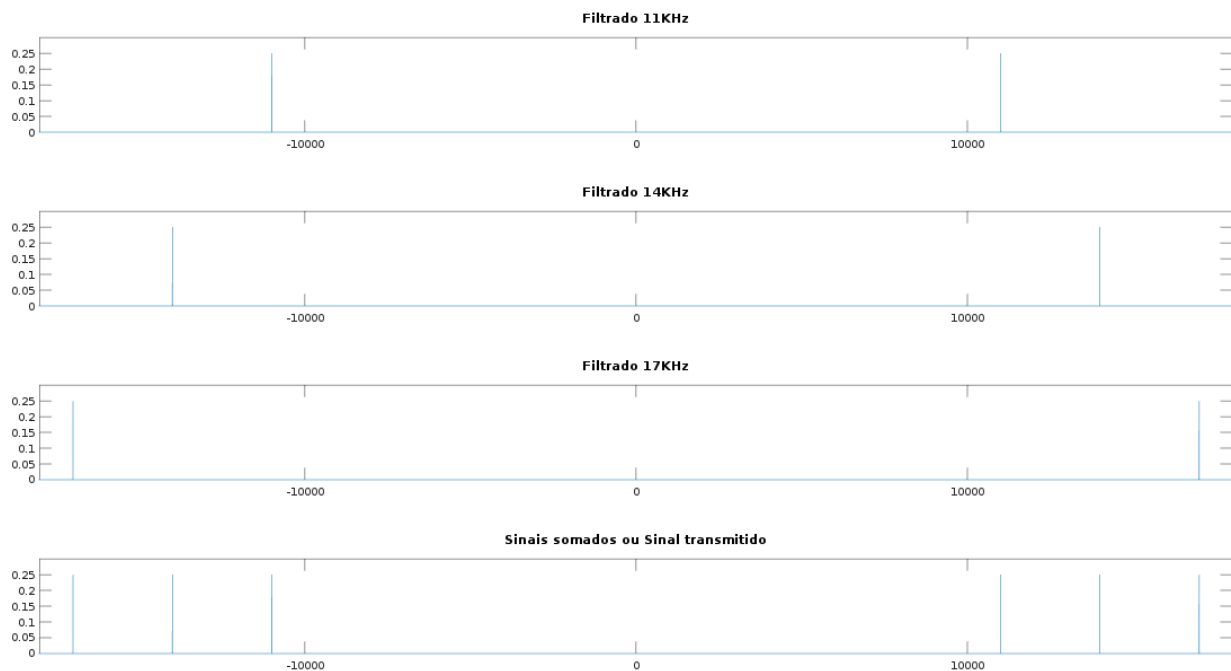


Figura 7: Sinal transmitido

3.4 Sinal recebido e filtrado

```
77 % Recebendo sinal e filtrando em 11k 14k 17k no dominio da frequencia
78 y_11k = y_f.*filtro_11k;
79 y_14k = y_f.*filtro_14k;
80 y_17k = y_f.*filtro_17k;
```

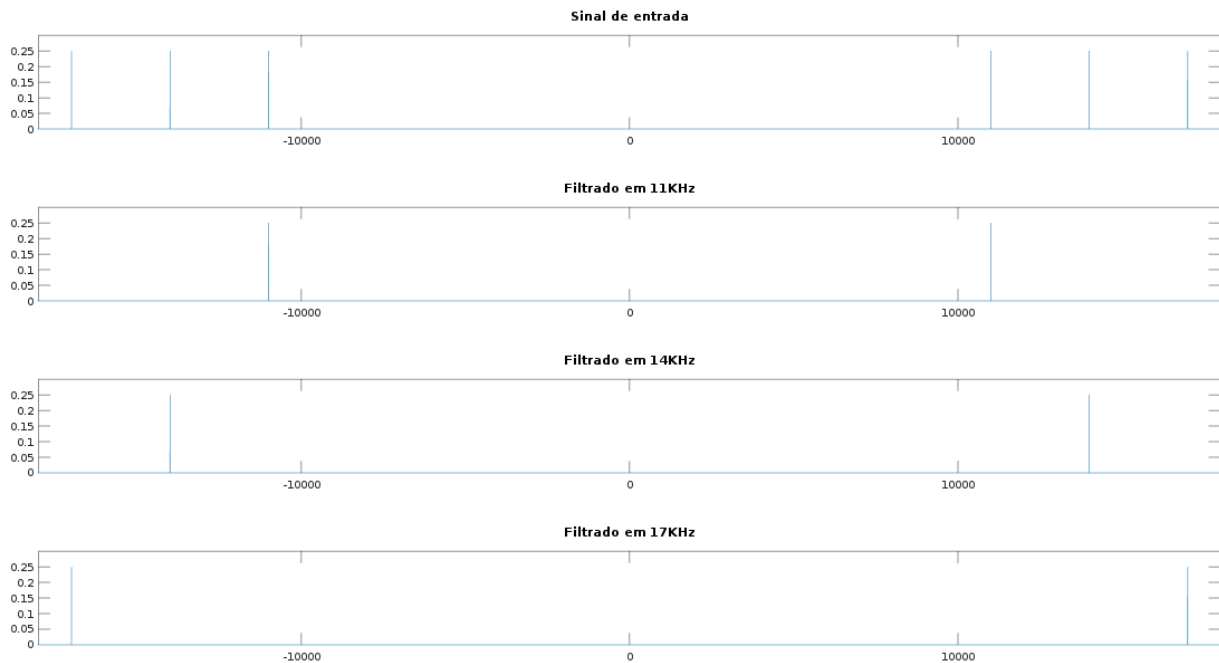


Figura 8: Sinal recebido e filtrado

3.5 Obtendo sinais primitivos, mas ainda não filtrados e puros

```
81 % Convoluindo sinal filtrado e portadora
82 % para obter o sinal na frequencia original
83 y_21k_1k = conv(y_11k, c_f1); % Convoluindo 11KHz com 10KHz
84 y_26k_2k = conv(y_14k, c_f2); % Convoluindo 14KHz com 12KHz
85 y_31k_3k = conv(y_17k, c_f3); % Convoluindo 17KHz com 14KHz
```

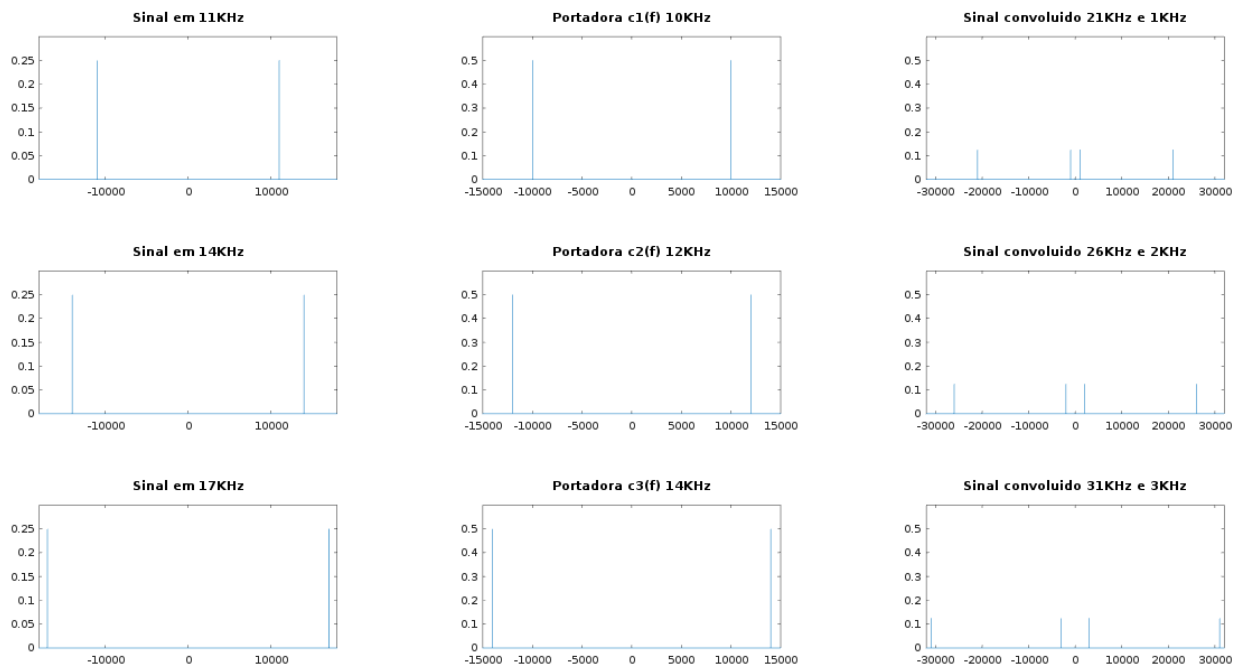


Figura 9: Sinal primitivo conjunto com outra frequência

3.6 Obtendo sinais/modulantes originais

```

86                                     % Filtrando apenas a frequencia original
87 y_1k = y_21k_1k.*filtro_pb_4k; % Obtendo apenas a frequencia de 1kHz
88 y_2k = y_26k_2k.*filtro_pb_4k; % Obtendo apenas a frequencia de 2kHz
89 y_3k = y_31k_3k.*filtro_pb_4k; % Obtendo apenas a frequencia de 3kHz

```

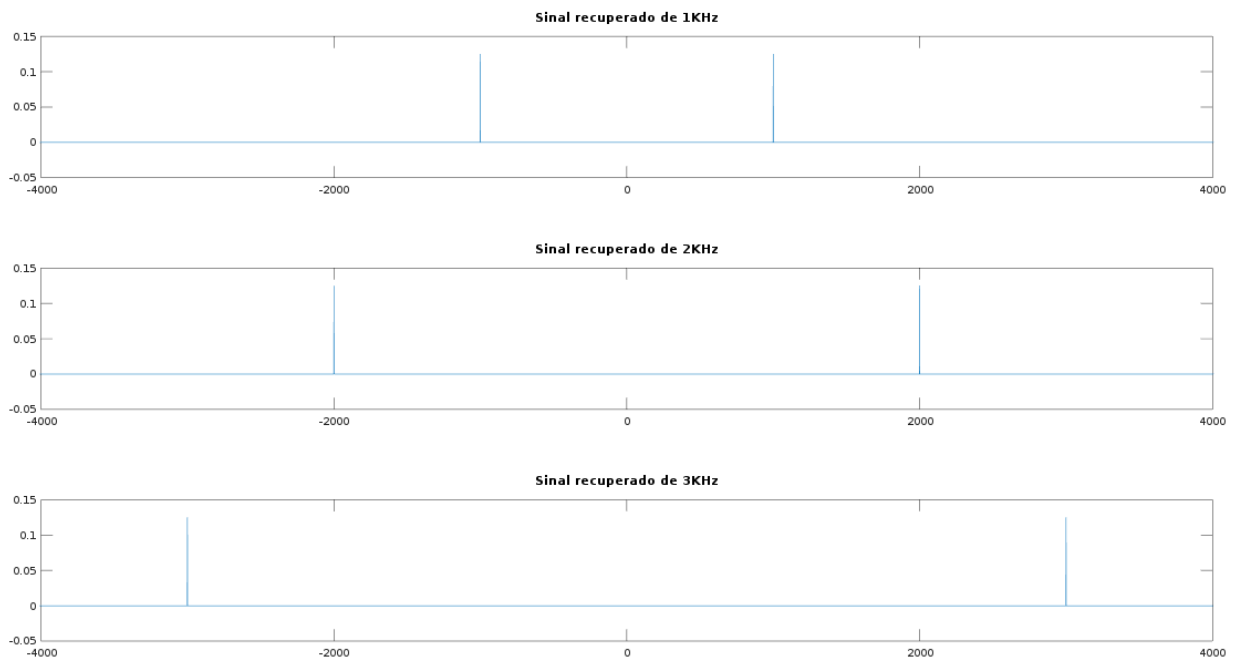


Figura 10: Sinal recuperado

4 Código completo

Veja o código completo [aqui](#)