



**INSTITUTO
FEDERAL**

Santa Catarina

Câmpus
São José

Modulação e Demodulação em Frequência (FM)

Sistemas de Comunicação I

Arthur Cadore Matuella Barcella

09 de Abril de 2024

Sumário

| | |
|--|-----------|
| 1. Introdução | 3 |
| 2. Fundamentação teórica | 3 |
| 3. Análise dos resultados | 3 |
| 3.1. Sinal de áudio Aleatório: | 3 |
| 3.2. Sinal Senoidal Modulante | 8 |
| 4. Scripts e Códigos Utilizados: | 10 |
| 4.1. Definições Iniciais | 10 |
| 4.2. Sinal modulante e Modulado FM | 11 |
| 4.3. FFT dos sinais modulantes | 12 |
| 4.4. Demodulação do sinal e Filtro | 12 |
| 4.5. Filtragem e plotagem dos sinais resultantes | 13 |
| 5. Conclusão | 14 |
| 6. Referências | 14 |

1. Introdução

O objetivo deste relatório é apresentar o desenvolvimento de um sistema de modulação e demodulação em frequência (FM) para sinais de áudio. O sistema foi desenvolvido através de linguagem MATLAB (Octave), e tem como objetivo principal a compreensão do processo de modulação e demodulação em frequência, bem como a análise dos sinais modulados e demodulados.

Neste relatório será apresentado a fundamentação teórica do processo de modulação e demodulação em frequência, bem como a análise dos sinais modulados e demodulados, os scripts MATLAB utilizados e os resultados obtidos.

Desta forma, poderemos compreender o processo de modulação e demodulação FM, bem como a análise dos sinais modulados e demodulados, e a importância deste processo para a transmissão de sinais de áudio em sistemas de comunicação.

2. Fundamentação teórica

Seção II - Conceitos teóricos utilizados no relatório

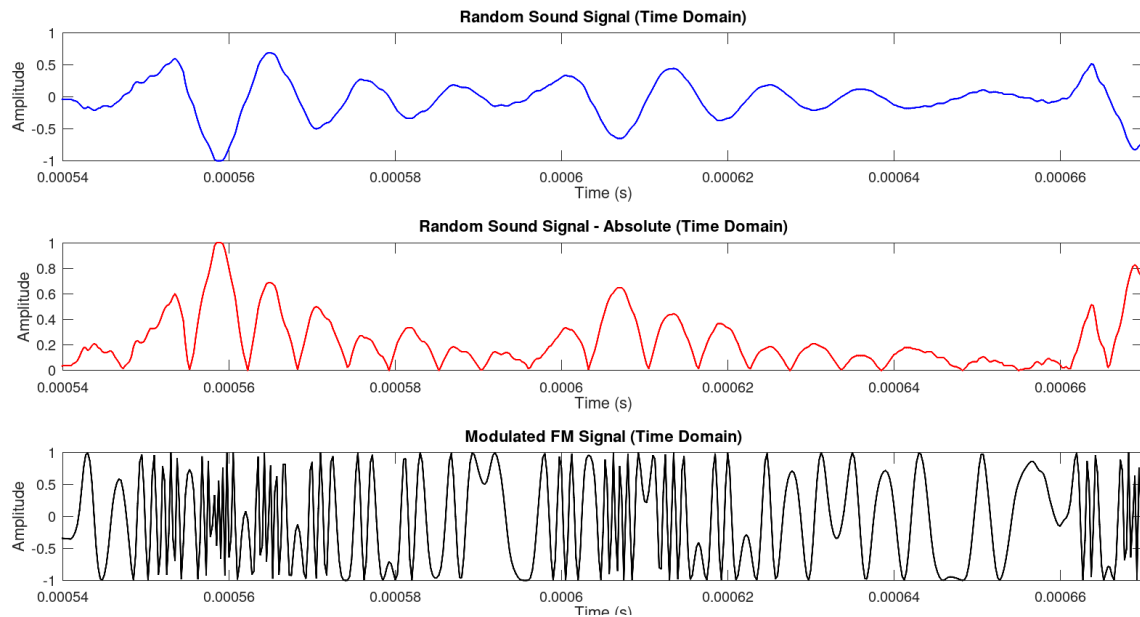
3. Análise dos resultados

3.1. Sinal de áudio Aleatório:

Inicialmente, foi feita a importação de um sinal de áudio para ser utilizado como modulante da portadora em frequência, para transmissão em FM.

A figura abaixo mostra o plot do sinal no domínio do tempo, bem como seu respectivo plot do sinal no domínio da frequência.

Figure 1: Elaborada pelo Autor



Sinal de entrada no domínio do tempo

Uma vez com o sinal de entrada definido, a modulação em frequência foi realizada através da integração do argumento de fase da portadora a partir do sinal da modulante, conforme o script abaixo:

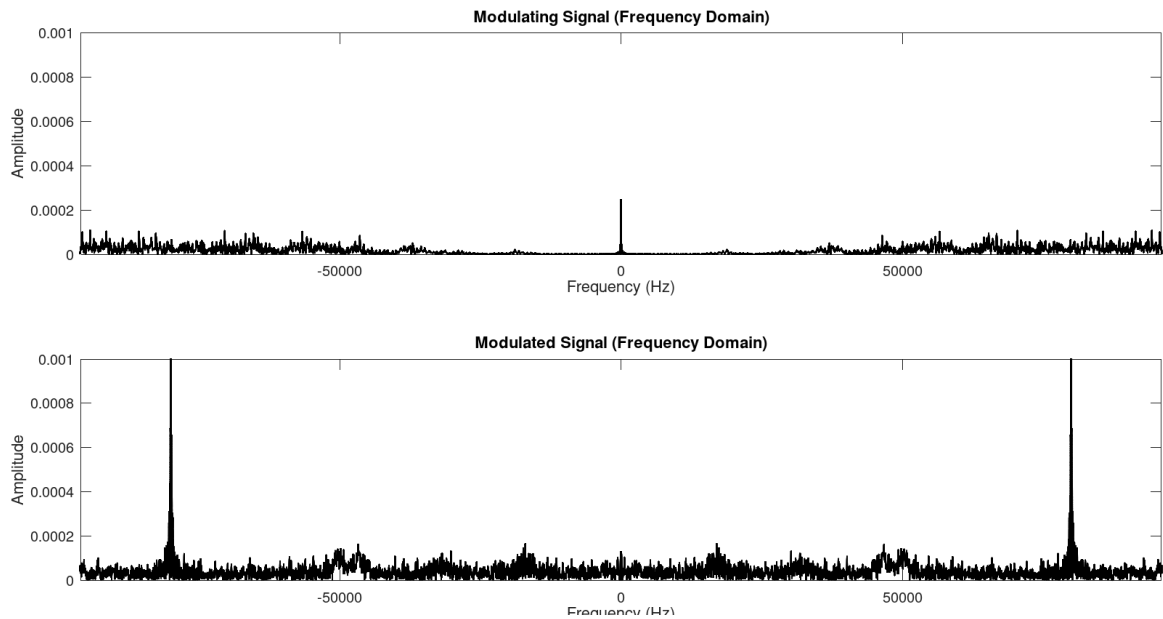
```
1 % Creating the FM modulated signal:
2 phase_argument = 2*pi*k_f*cumsum(modulating_signal)*(Ts);
3 modulated_signal = A_carrier * cos(2*pi*f_carrier*t + phase_argument);
```

Onde na figura acima os parâmetros são:

- `modulating_signal` é o sinal de áudio importado.
- `k_f` é a sensibilidade do modulador para variação de frequência.
- `Ts` é o período de amostragem do sinal.
- `A_carrier` é a amplitude da portadora
- `f_carrier` é a frequência da portadora.
- `t` é o vetor de tempo do sinal modulado (utilizado para realizar a modulação em FM).
- `phase_argument` é o argumento de fase da portadora do sinal, gerado a partir da integração do sinal modulante.
- `modulated_signal` é o sinal modulado em FM.

Uma vez com o sinal modulado em FM, podemos compreender o formato do sinal modulado no domínio do tempo e da frequência, conforme a figura abaixo:

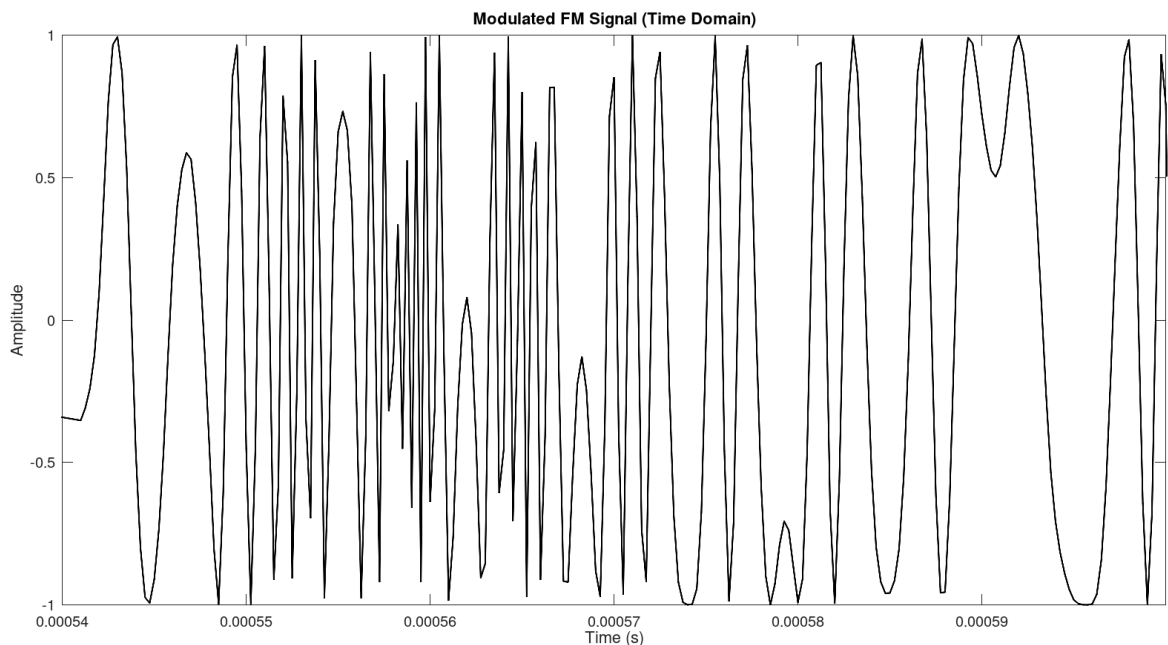
Figure 2: Elaborada pelo Autor



Sinal modulado em FM no domínio do tempo e da frequência

Uma vez com o sinal modulado, e multiplexado, podemos transmiti-lo pelo meio físico sem que haja interferência entre cada portadora (idealmente). O sinal no meio físico é ilustrado abaixo em azul.

Figure 3: Elaborada pelo Autor



Sinal modulado e “transmitido” no meio físico

Na recepção do sinal, precisamos realizar sua demodulação para ter novamente o sinal de áudio original. Para isso, utilizamos um demodulador FM, que é basicamente um circuito que realiza a derivação do sinal modulado, conforme o script abaixo:

```

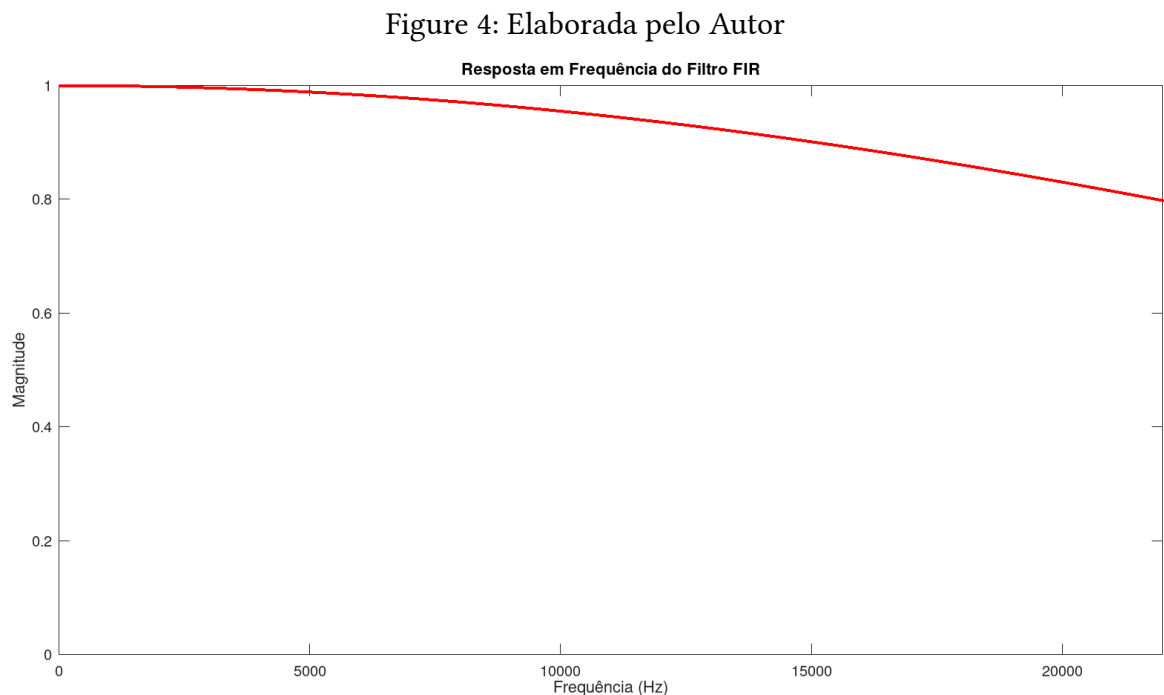
1 % Calculating the FM demodulation for the modulated signal
2 demodulated_signal = diff(modulated_signal) * fs / k0;
3 demodulated_signal = [demodulated_signal, 0]; % Sinal demodulado
4
5 % calculating the FFT of the random signal;
6 demodulated_f = fft(demodulated_signal)/length(demodulated_signal);
7 demodulated_f = fftshift(demodulated_f);
8
9 % Calculating the signal wrap.
10 demodulated_wrap = abs(hilbert(demodulated_signal));

```

Com o sinal demodulado, utilizamos um filtro passa-baixas para eliminar as frequências indesejadas, e obter o sinal de áudio original.

Para isso, foi utilizado um filtro FIR de ordem relativamente alta (neste caso 100), com frequência de corte de 20kHz. A frequência neste script foi fixada em 20kHz, pois trata-se de um sinal de áudio, e portanto, não há informação relevante acima desta frequência para ser capturada.

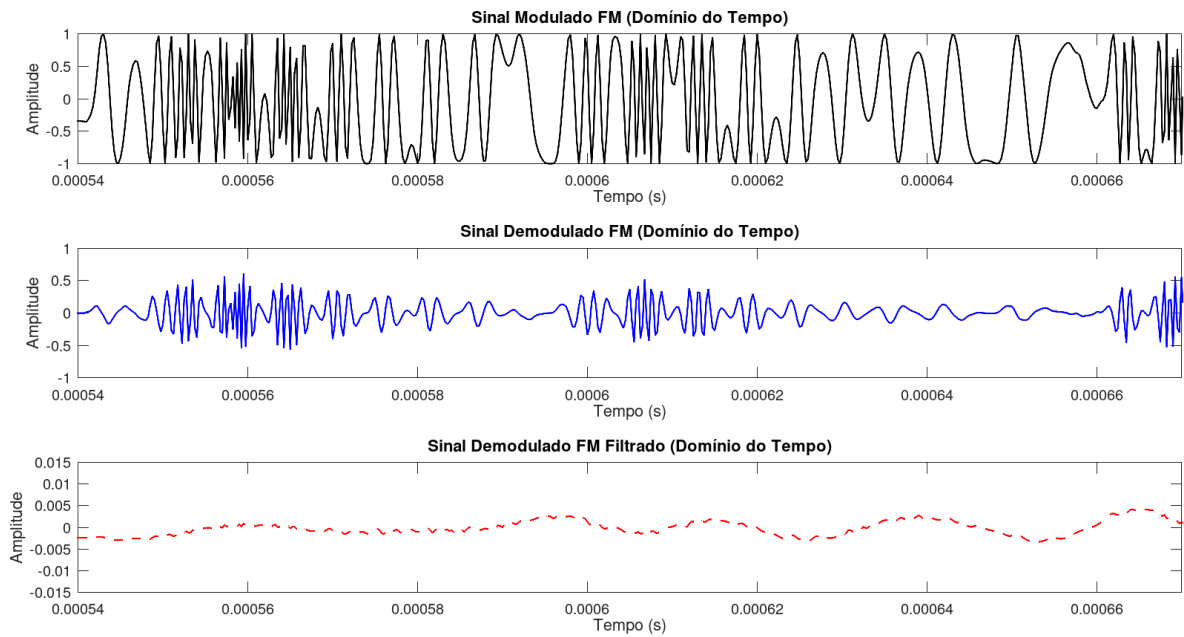
Para verificar se de fato o filtro está atuando corretamente, abaixo está um plot da resposta em frequência do filtro FIR:



Resposta em frequência do filtro FIR

Com o sinal demodulado e filtrado, podemos realizar seu plot no domínio do tempo e também realizar a FFT do sinal para observar as componentes de frequência do sinal demodulado.

Figure 5: Elaborada pelo Autor

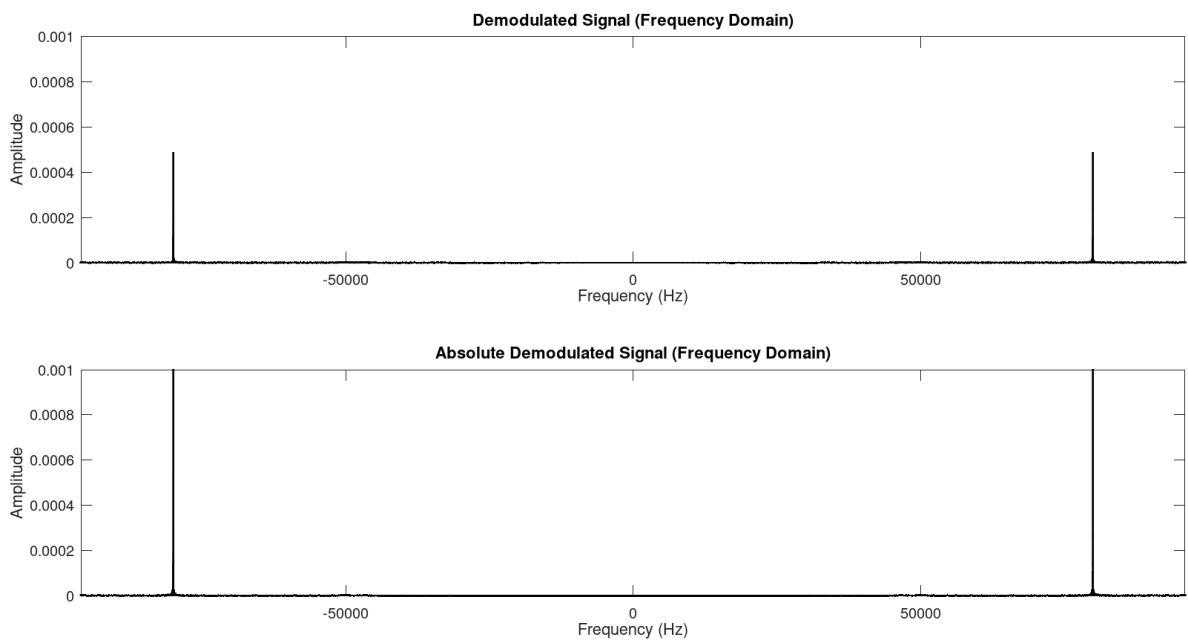


Sinal demodulado no domínio do tempo

Como podemos observar, o sinal demodulado é muito semelhante ao sinal de áudio original, com pequenas distorções devido ao processo de modulação e demodulação em frequência.

O sinal também foi plotado no domínio do tempo e da frequência, para verificar se o sinal demodulado está correto.

Figure 6: Elaborada pelo Autor



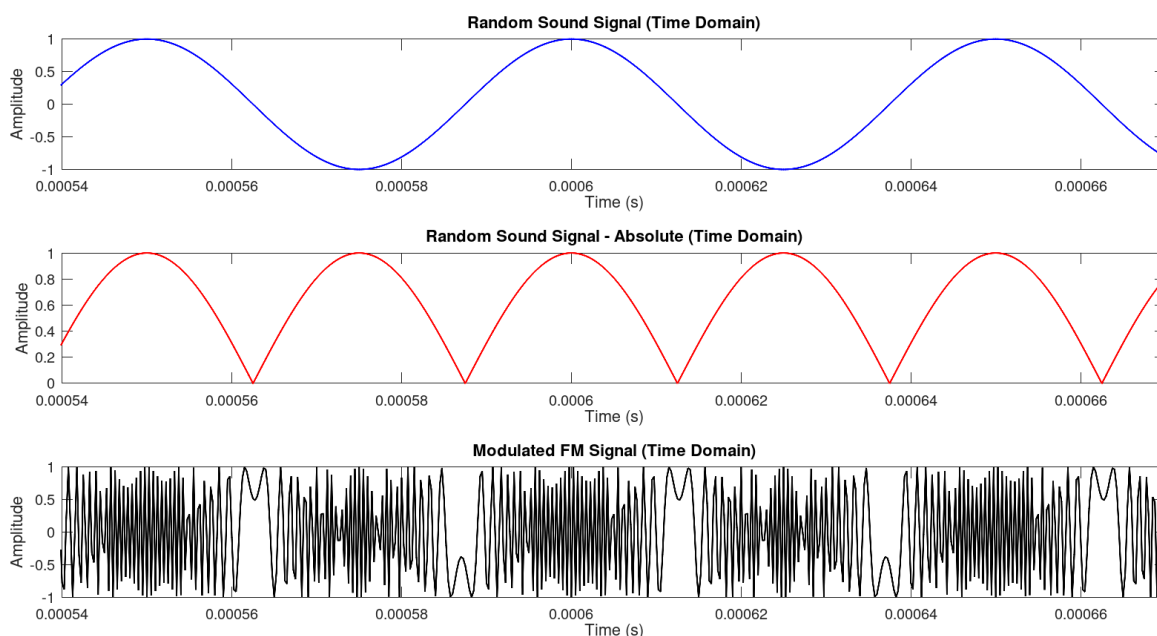
Sinal demodulado no domínio da frequência

3.2. Sinal Senoidal Modulante

Devido as variações no processo de modulação e demodulação apresentadas anteriormente, foi feita a análise de um sinal puramente senoidal como modulante em FM, para verificar se o processo de modulação e demodulação em frequência está correto.

Inicialmente, foi feita a definição dos parâmetros do sinal modulante e em seguida o plot do mesmo no domínio do tempo e também da frequência:

Figure 7: Elaborada pelo Autor

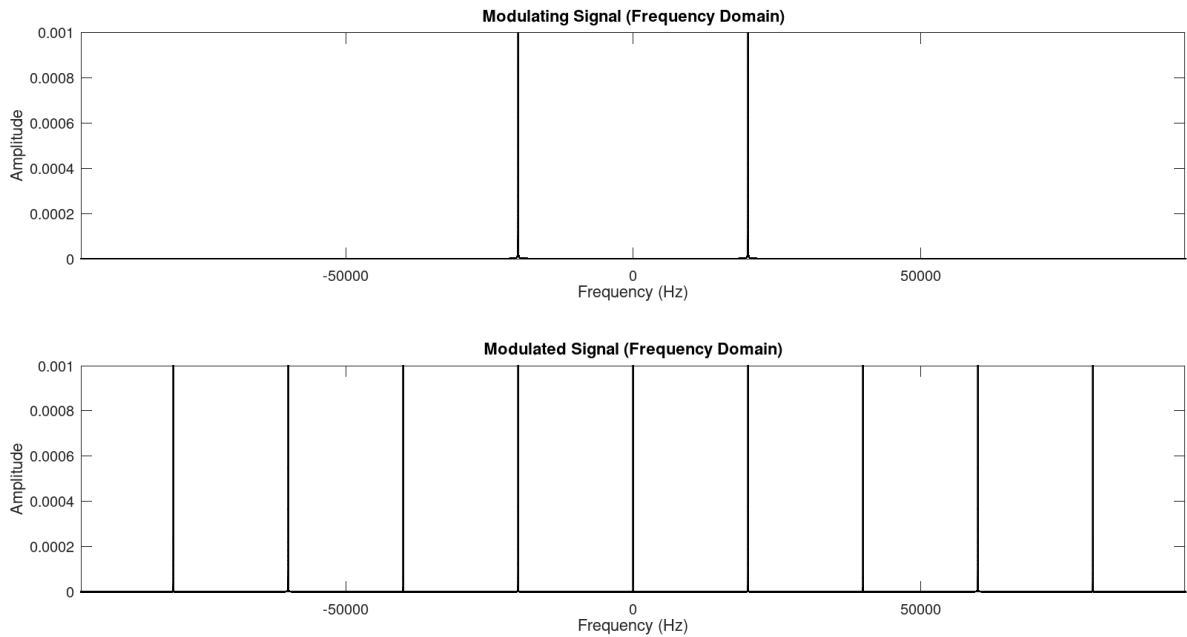


Sinal senoidal modulante no domínio do tempo e da frequência

Em seguida, com o sinal modulante definido, foi feita a modulação em frequência do sinal senoidal, note que para esse processo de modulação, o sinal modulante é uma senoide pura, e portanto, o sinal modulado em FM possui uma variação suave e periódica de frequência ao longo do tempo.

Sendo assim possível analisar o sinal modulado no domínio do tempo e da frequência, conforme a figura abaixo:

Figure 8: Elaborada pelo Autor

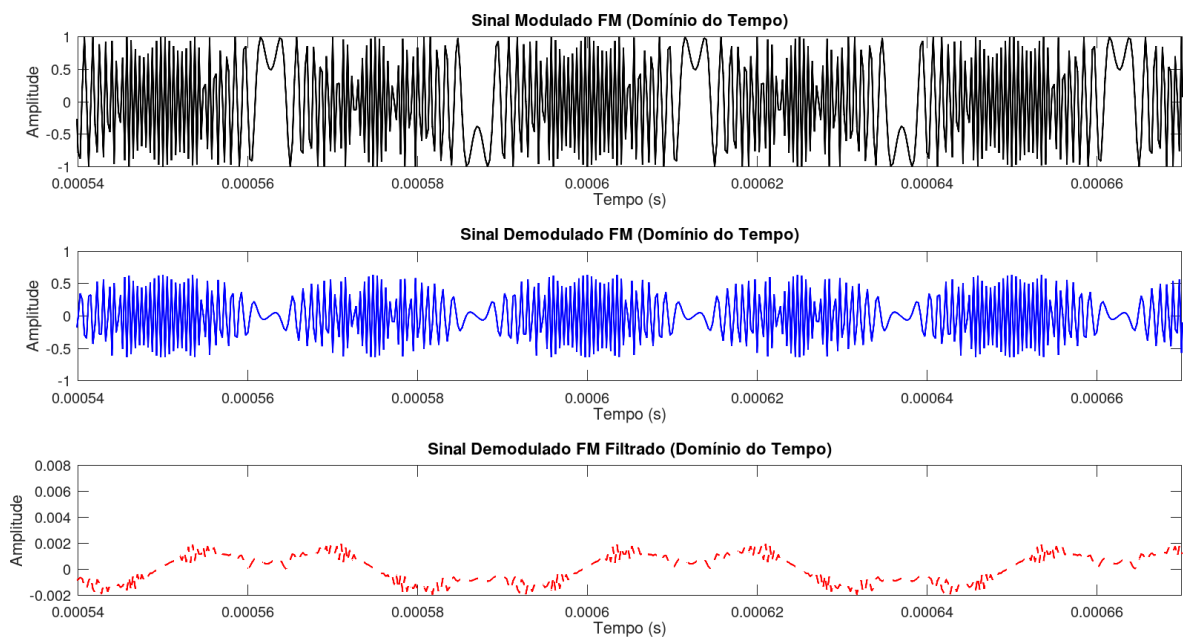


Sinal modulado em FM no domínio do tempo e da frequência

Com o sinal modulado em FM definido, podemos transmiti-lo pelo meio físico, e realizar a demodulação do sinal para obter o sinal senoidal original.

Na recepção, foi feita a demodulação do sinal modulado, e em seguida a filtragem do sinal demodulado para obter o sinal senoidal original, a figura abaixo mostra o sinal demodulado no domínio do tempo:

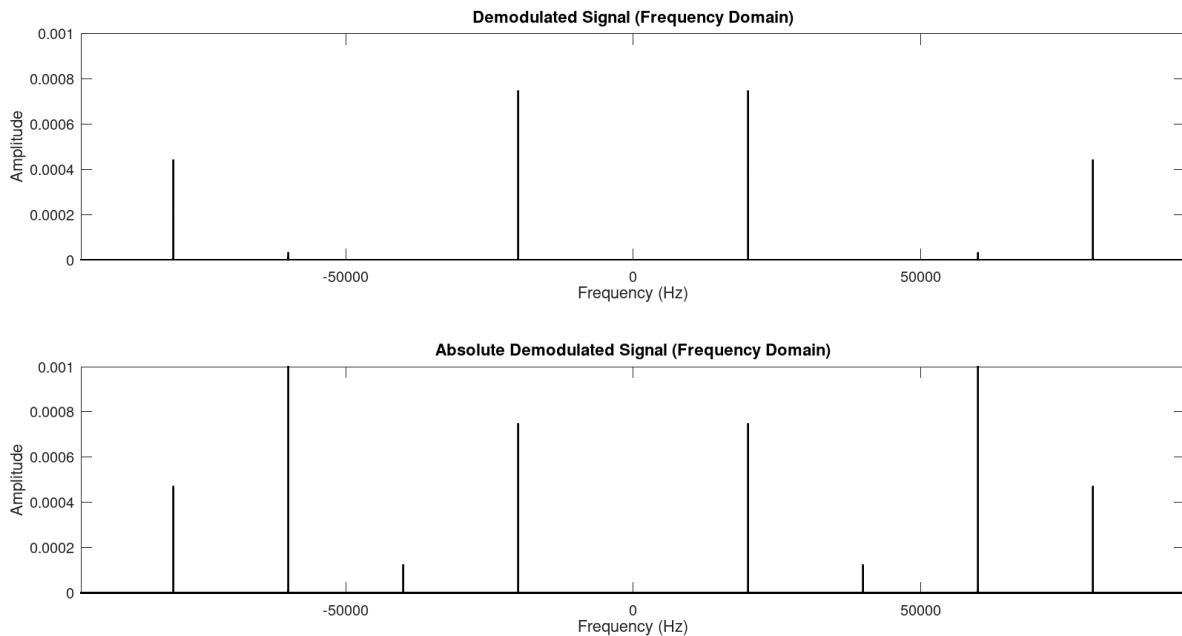
Figure 9: Elaborada pelo Autor



Sinal demodulado no domínio do tempo

Novamente, para verificar se o sinal demodulado está correto, foi feita a análise do sinal demodulado no domínio da frequência, conforme a figura abaixo:

Figure 10: Elaborada pelo Autor



Sinal demodulado no domínio da frequência

4. Scripts e Códigos Utilizados:

4.1. Definições Iniciais

O script abaixo define as variáveis iniciais do sistema, como a amplitude dos sinais, a frequência do sinal modulante, a frequência da portadora, a sensibilidade do modulador para variação de frequência, e o período de amostragem do sinal.

```

1  close all; clear all; clc;
2  pkg load signal;
3
4  % Altera o tamanho da fonte nos plots para 15
5  set(0, 'DefaultAxesFontSize', 20);
6
7  % Defining the signals amplitude.
8  A_modulating = 1;
9  A_carrier = 1;
10
11 % Defining the signals frequency
12 f_modulating_max = 20000;
13 f_carrier = 80000;
14
15 % modulator sensibility for frequency variation (Hz/volts)
16 k_f = 2000000;
17 k0 = 2*pi*k_f;
18

```

```

19 % Delta variable, corresponding to max frequency variation.
20 d_f = k_f*A_modulating;
21
22 % Beta variable, corresponding to percentage of frequency variation about
23 the frequency of the modulating.
24 b = d_f/f_modulating_max;
25
26 % Defining the period and frequency of sampling:
27 fs = 50*f_carrier;
28 Ts = 1/fs;
29 T = 1/f_modulating_max;
30
31 % Defining the sinal period.
32 t_inicial = 0;
33 t_final = 2;
34
35 % "t" vector, corresponding to the time period of analysis, on time domain.
36 t = [t_inicial:Ts:t_final];

```

4.2. Sinal modulante e Modulado FM

O script abaixo importa um sinal de áudio para ser utilizado como modulante da portadora em frequência, e em seguida realiza a modulação em frequência do sinal modulante.

```

1 % Import the audioSignal to use as modulating FM signal:
2 [modulating_signal, Hs] = audioread('randomSignal.wav');
3 modulating_signal = transpose(modulating_signal);
4
5 % Calculate the number of zeros to be added
6 num_zeros = length(t) - length(modulating_signal);
7
8 % Add the zeros to the end of the modulating_signal vector
9 modulating_signal = [modulating_signal, zeros(1, num_zeros)];
10
11 % Transpose the modulated signal if necessary
12 modulated_signal = transpose(modulating_signal);
13
14 % Creating the FM modulated signal:
15 phase_argument = 2*pi*k_f*cumsum(modulating_signal)*(Ts);
16 modulated_signal = A_carrier * cos(2*pi*f_carrier*t + phase_argument);
17
18 % Plot signals on time domain:
19 figure(1)
20 subplot(311)
21 plot(t, (modulating_signal), 'b', 'LineWidth', 2)
22 xlim([0.00054 0.00067])
23 xlabel('Time (s)')
24 ylabel('Amplitude')
25 title('Random Sound Signal (Time Domain)')
26
27 subplot(312)
28 plot(t, abs(modulating_signal), 'r', 'LineWidth', 2)
29 xlim([0.00054 0.00067])
30 xlabel('Time (s)')
31 ylabel('Amplitude')
32 title('Random Sound Signal - Absolute (Time Domain)')

```

```

33
34 subplot(313)
35 plot(t, modulated_signal, 'k', 'LineWidth', 2)
36 xlim([0.00054 0.00067])
37 xlabel('Time (s)')
38 ylabel('Amplitude')
39 title('Modulated FM Signal (Time Domain)')

```

4.3. FFT dos sinais modulantes

O script abaixo calcula a FFT dos sinais modulantes e modulados, e em seguida realiza o plot dos sinais no domínio da frequência.

```

1 % calculating the step of the frequency vector "f" (frequency domain);
2 f_step = 1/t_final;
3
4 % creating the frequency vector "f" (frequency domain);
5 f = [-fs/2:f_step:fs/2];
6
7 % calculating the FFT of the random signal;
8 modulating_f = fft(modulating_signal)/length(modulating_signal);
9 modulating_f = fftshift(modulating_f);
10
11 % calculating the FFT of the modulated signal;
12 modulated_f = fft(modulated_signal)/length(modulated_signal);
13 modulated_f = fftshift(modulated_f);
14
15 % Plotting the modulated signal on frequency domain;
16 figure(2)
17 subplot(211)
18 plot(f, abs(modulating_f), 'k', 'LineWidth', 2)
19 xlabel('Frequency (Hz)')
20 ylabel('Amplitude')
21 title('Modulating Signal (Frequency Domain)')
22 xlim([-f_carrier*1.2 f_carrier*1.2])
23 ylim([0 A_carrier/1000])
24
25 subplot(212)
26 plot(f, abs(modulated_f), 'k', 'LineWidth', 2)
27 xlabel('Frequency (Hz)')
28 ylabel('Amplitude')
29 title('Modulated Signal (Frequency Domain)')
30 xlim([-f_carrier*1.2 f_carrier*1.2])
31 ylim([0 A_carrier/1000])

```

4.4. Demodulação do sinal e Filtro

O script abaixo realiza a demodulação do sinal modulado, e em seguida realiza a filtragem do sinal demodulado para obter o sinal de áudio original.

```

1 % Calculating the FM demodulation for the modulated signal
2 demodulated_signal = diff(modulated_signal) * fs / k0;
3 demodulated_signal = [demodulated_signal, 0]; % Sinal demodulado

```

```

4
5 % Ordem do filtro FIR
6 filtro_ordem = 100;
7
8 % Frequência de corte do filtro FIR
9 % Como trata-se de um sinal de áudio, a frequência de corte pode ser fixada
  em 20kHz
10 frequencia_corte = 20000;
11
12 % Coeficientes do filtro FIR para cada sinal demodulado
13 coeficientes_filtro = fir1(filtro_ordem, frequencia_corte/(fs/2));
14
15 % Resposta em frequência do filtro FIR para cada sinal demodulado
16 [H_fir, f_fir] = freqz(coeficientes_filtro, 1, length(t), fs);
17
18 % Plot da resposta em frequência do filtro:
19 figure(6)
20 plot(f_fir, abs(H_fir), 'r', 'LineWidth', 3)
21 xlim([0 frequencia_corte*1.1])
22 title('Resposta em Frequência do Filtro FIR')
23 xlabel('Frequência (Hz)')
24 ylabel('Magnitude')

```

4.5. Filtragem e plotagem dos sinais resultantes

O script abaixo realiza a filtragem do sinal demodulado, e em seguida realiza o plot dos sinais modulados e demodulados no domínio do tempo e da frequência.

```

1 % Filtragem dos sinais demodulados
2 demodulated_filtered = filter(coeficientes_filtro, 1, demodulated_signal);
3
4 % calculating the FFT of the random signal;
5 demodulated_filtered_f = fft(demodulated_filtered)/
  length(demodulated_filtered);
6 demodulated_filtered_f = fftshift(demodulated_filtered_f);
7
8 % Calculating the signal wrap.
9 demodulated_wrap = abs(hilbert(demodulated_filtered));
10
11 % Plotting the modulated and demodulated signals on time domain:
12 figure(3)
13 subplot(311)
14 plot(t, modulated_signal, 'k', 'LineWidth', 2)
15 xlim([0.00054 0.00067])
16 xlabel('Tempo (s)')
17 ylabel('Amplitude')
18 title('Sinal Modulado FM (Domínio do Tempo)')
19
20 subplot(312)
21 plot(t, demodulated_signal, 'b', 'LineWidth', 2)
22 xlim([0.00054 0.00067])
23 xlabel('Tempo (s)')
24 ylabel('Amplitude')
25 title('Sinal Demodulado FM (Domínio do Tempo)')
26
27 subplot(313)

```

```

28 plot(t, demodulated_filtered, 'r--', 'LineWidth', 2)
29 xlim([0.00054 0.00067])
30 xlabel('Tempo (s)')
31 ylabel('Amplitude')
32 title('Sinal Demodulado FM Filtrado (Domínio do Tempo)')
33
34 figure(4)
35 subplot(211)
36 plot(f, demodulated_filtered_f, 'k', 'LineWidth', 2)
37 xlabel('Frequency (Hz)')
38 ylabel('Amplitude')
39 title('Demodulated Signal (Frequency Domain)')
40 xlim([-f_carrier*1.2 f_carrier*1.2])
41 ylim([0 A_carrier/1000])
42
43 subplot(212)
44 plot(f, abs(demodulated_filtered_f), 'k', 'LineWidth', 2)
45 xlabel('Frequency (Hz)')
46 ylabel('Amplitude')
47 title('Absolute Demodulated Signal (Frequency Domain)')
48 xlim([-f_carrier*1.2 f_carrier*1.2])
49 ylim([0 A_carrier/1000])

```

5. Conclusão

A partir dos conceitos vistos e dos resultados obtidos, podemos concluir que o processo de modulação e demodulação em frequência é uma técnica eficiente para a transmissão de sinais de áudio em sistemas de comunicação, pois permite a transmissão de sinais de áudio com qualidade e fidelidade com baixa interferência devido a informação estar sendo carregada na variação de frequência e não na amplitude do sinal.

Desta forma, podemos compreender seu uso em sistemas de telecomunicação utilizado atualmente pelas rádios analógicas regionais para transmissão de sinais de áudio em broadcast para toda a região, visto que esse tipo de transmissão possui um baixo índice de ruído e distorção.

6. Referências

Para o desenvolvimento deste relatório, foi utilizado o seguinte material de referência:

- Software Defined Radio Using MATLAB & Simulink and the RTL-SDR, de Robert W. Stewart