

Aula prática – modelo de sistema de modulação/demodulação em MATLAB

Objetivo: desenvolver em MATLAB um modelo de sistema de modulação e demodulação utilizando o esquema 8-PSK (*Quadrature Amplitude Modulation*). O código deve funcionar na versão R2012b do MATLAB.

O código desenvolvido deve partir de um vetor de 12 bits gerado aleatoriamente. Os bits devem então ser convertidos para sinais I e Q e modulados, utilizando 8-PSK, em uma portadora de frequência f_c . A Figura 1 mostra a associação entre símbolos e bits a ser utilizada. O receptor deverá então executar a demodulação do sinal recebido, separando as componentes I e Q e filtrando as componentes de alta frequência. Assuma que o receptor tem conhecimento ideal da portadora.

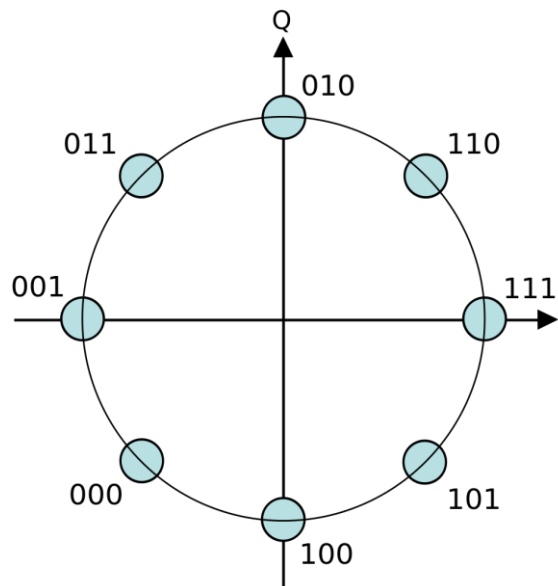


Figura 1: Constelação 8-PSK

Plote (no mínimo) os seguintes sinais:

- 1) as componentes I e Q no transmissor;
- 2) o sinal modulado resultante;
- 3) as componentes I e Q recuperadas no receptor antes da filtragem;
- 4) as componentes I e Q recuperadas no receptor após a filtragem.

O código deve incluir comentários discutindo brevemente os principais passos efetuados pelo sistema.

Entrega: entregue via *moodle* o código (.m) resultante até o dia 19/11.

Dica 1: Embora na prática a filtragem seja feita no domínio tempo para esse tipo de aplicação, o projeto de filtros está fora do escopo dessa disciplina. Assim, recomenda-se o uso das funções `fft`, `fftshift`, `ifftshift` e `ifft` para filtragem do sinal. Dessa forma, as frequências indesejadas podem ser eliminadas uma a uma no domínio frequência. Lembre-se: o domínio frequência é simétrico para sinais reais. Ao manipular a transformada de um sinal, devemos manter essa propriedade. Ainda assim, ruídos numéricos podem introduzir componentes imaginárias no resultado da transformada inversa. Ao retornar para o domínio tempo (`ifft`), utilize somente a parte real do sinal.

Dica 2: Para iniciantes em MATLAB: comece criando um arquivo de script (.m). Nesse arquivo, um bom ponto de partida é o código abaixo:

```
clear;
close all;

%Constantes definidas pelo usuário
Fs=100; %frequência de amostragem
Tb=1; %tempo de um bit
Nb=12; %número de bits
Fc = 10; %frequência da portadora

%Constantes derivadas
Ts=1/Fs; %período de amostragem
t=0:Ts:(Nb*Tb-Ts); %vetor do tempo
L = length(t); %comprimento do vetor tempo
NFFT = 2^nextpow2(L); %número de pontos a serem utilizados na FFT
f = Fs/2*linspace(-1,1,NFFT); %vetor frequência (para sinais no domínio frequência)
```