

## Roteiro do trabalho sobre FSK

O código fornecido é para um FSK síncrono, onde o receptor precisa regenerar a portadora. Outro aspecto do código é que o sincronismo, que no caso corresponde a identificar o início do sinal de interesse (ou seja, da transmissão FSK), é feito manualmente. No código do demodulador é definido um limiar ("threshold") e o início é marcado como o momento onde a amplitude do sinal (recebido) supera o limiar. E o código ainda pergunta pela confirmação por parte do usuário. O objetivo do trabalho é atingir um novo FSK que:

a) seja assíncrono, ou seja, não precise da geração da portadora no receptor. Esse FSK deve usar dois filtros e identificar o símbolo recebido a partir das energias de saída desses filtros.

b) o sincronismo deve ser feito de forma automática, com a transmissão de uma sequência pré-definida de símbolos e uso da correlação cruzada para fazer o sincronismo. Sobre isso, talvez seja útil ler a seção **1.10 Correlation: a measure of dependence / similarity** do livro texto.

c) obter a maior taxa (em bps) possível com uma BER baixa (idealmente 0)

Para ajudar, tem algum "provided code":

fskmodul.m - modulation and playback on D/A (sound generation). The user should record sound and save to a WAV file using some utility such as Audacity or Window's Sound Recorder.

fskmodul2.m - similar to previous code but without sound playback

fskdemod.m - demodulation. Reads the WAV file recorded previously.

Some background:

1) You can read about orthogonality in FSK searching the textbook for the script: MatlabOctaveBookExamples/ex\_transforms\_check\_orthogonality.m

2) The script fsk\_synchronous.m in the current directory provides an example of synchronous FSK modulation and demodulation, without transmitting sound.

## Exemplo

Rodando o script fskmodul com as seguintes opções:

```
showGraficos = 1  %mostra graficos (1) ou nao (0)
```

```
f0 = 500  %freq. do bit 0
```

```
f1 = 1000  %freq. do bit 1
```

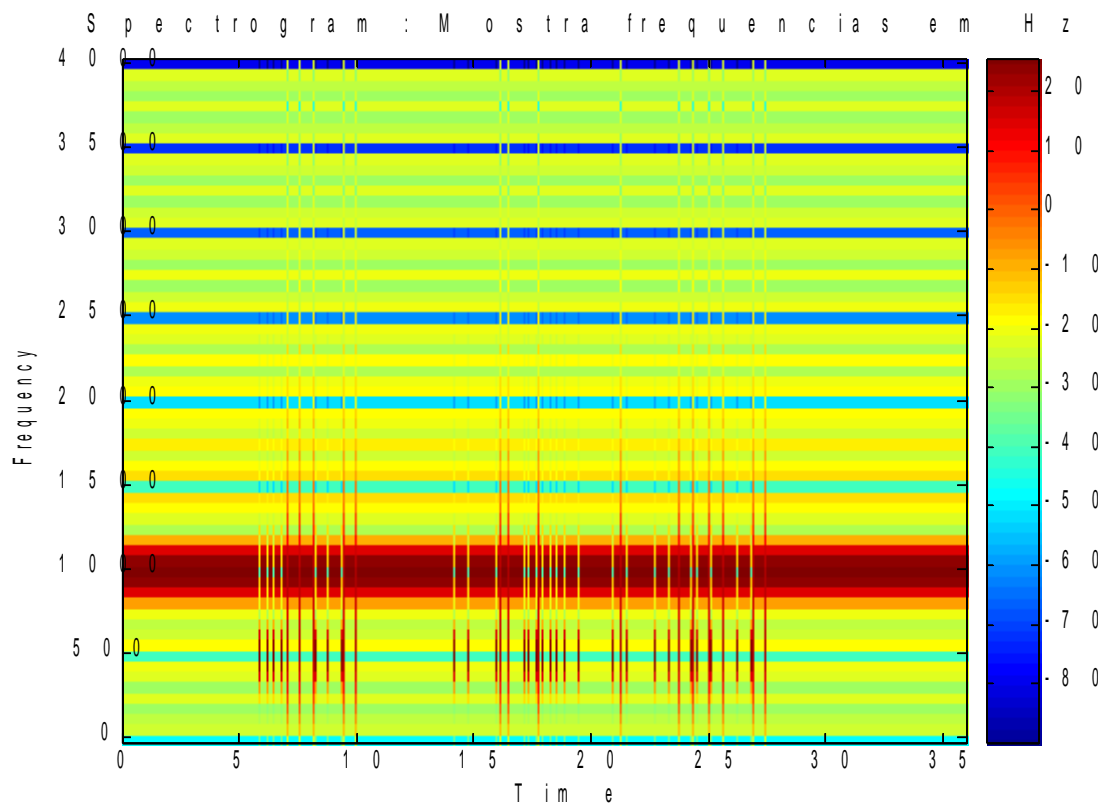
```
fsymbol = 100  %symbol frequency, R = 1/fsymbol in bauds
```

```
Fa = 8000;  %freq. de amostragem em Hz
```

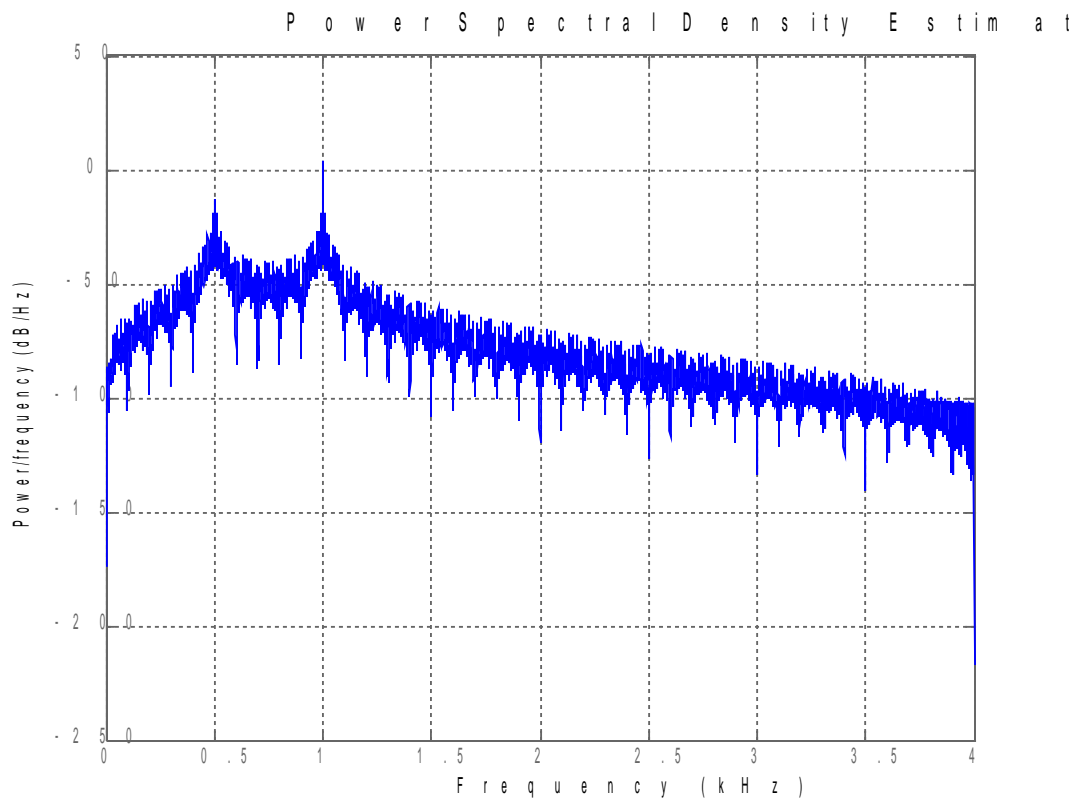
Obtem-se inicialmente a figura binária que se deseja transmitir



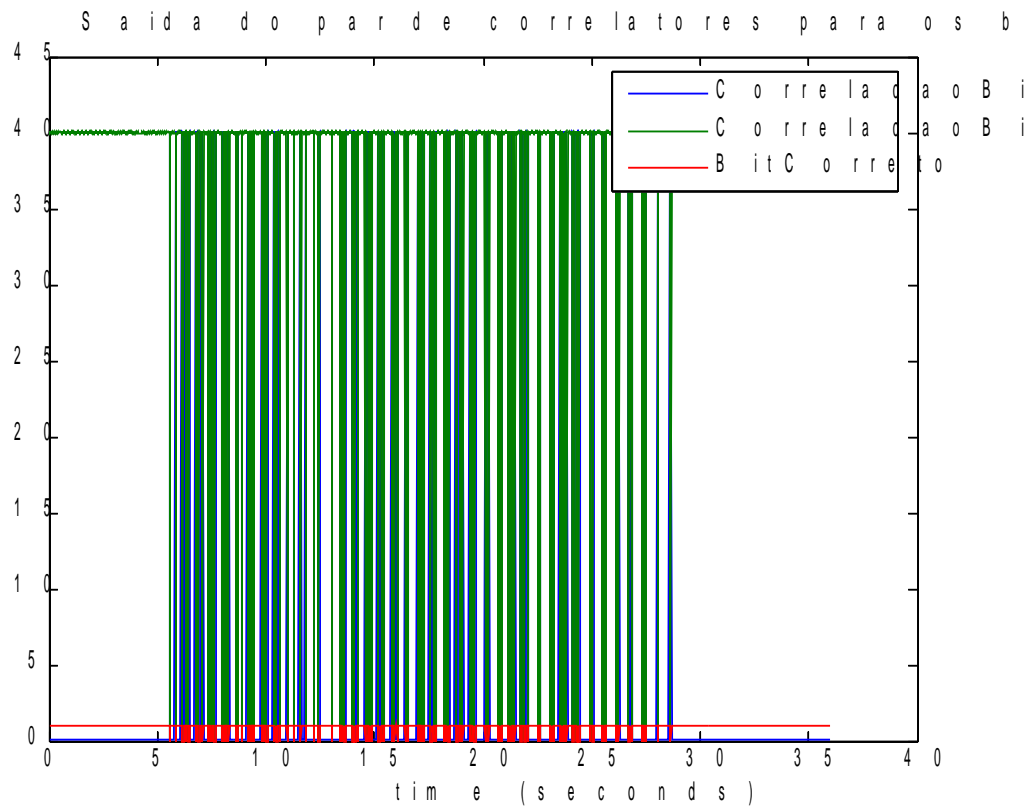
O espectrograma abaixo mostra que a maior parte da imagem eh composta por bits 1, de frequência 1000 Hz, e tem apenas poucos bits 0 pelo meio deles. O plot também mostra os nulls do sync correspondente a taxa de sinalização em Bauds. Como a FFT não eh alinhada, em cada transição de bit tem uma distorção de altas frequências.



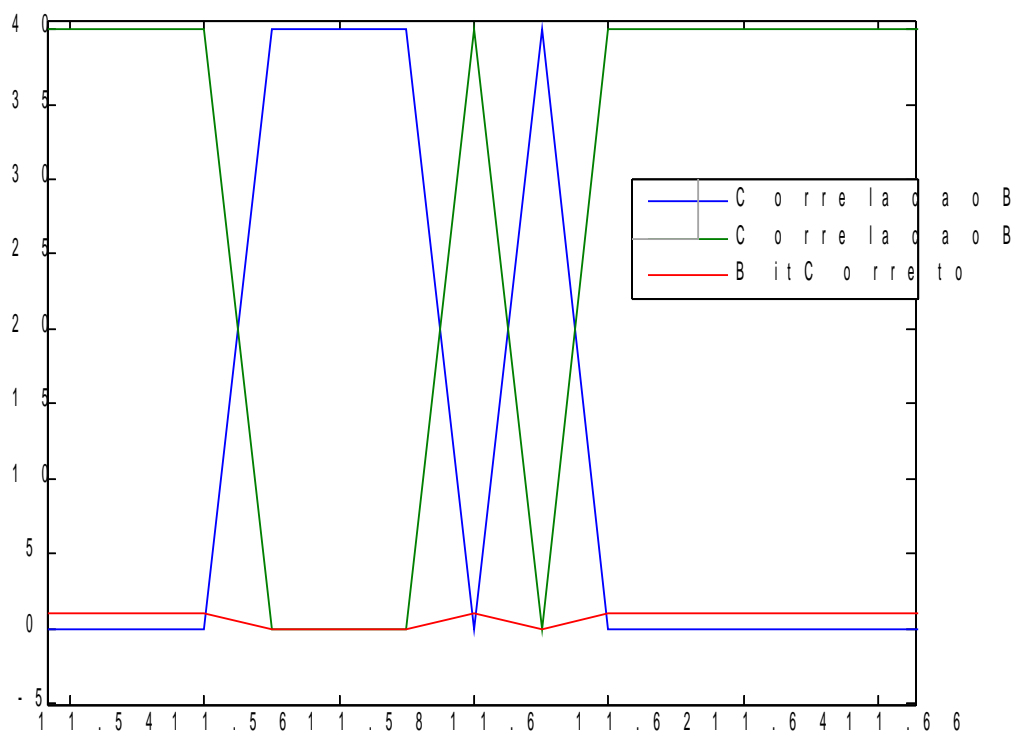
O espectro (PSD) do sinal todo eh dado abaixo, onde aparecem as duas portadoras e os nulls do sync:



No caso de recepção sem passar pelo canal, o sinal é recuperado com perfeição (erro zero). O plot abaixo mostra a saída dos correladores.



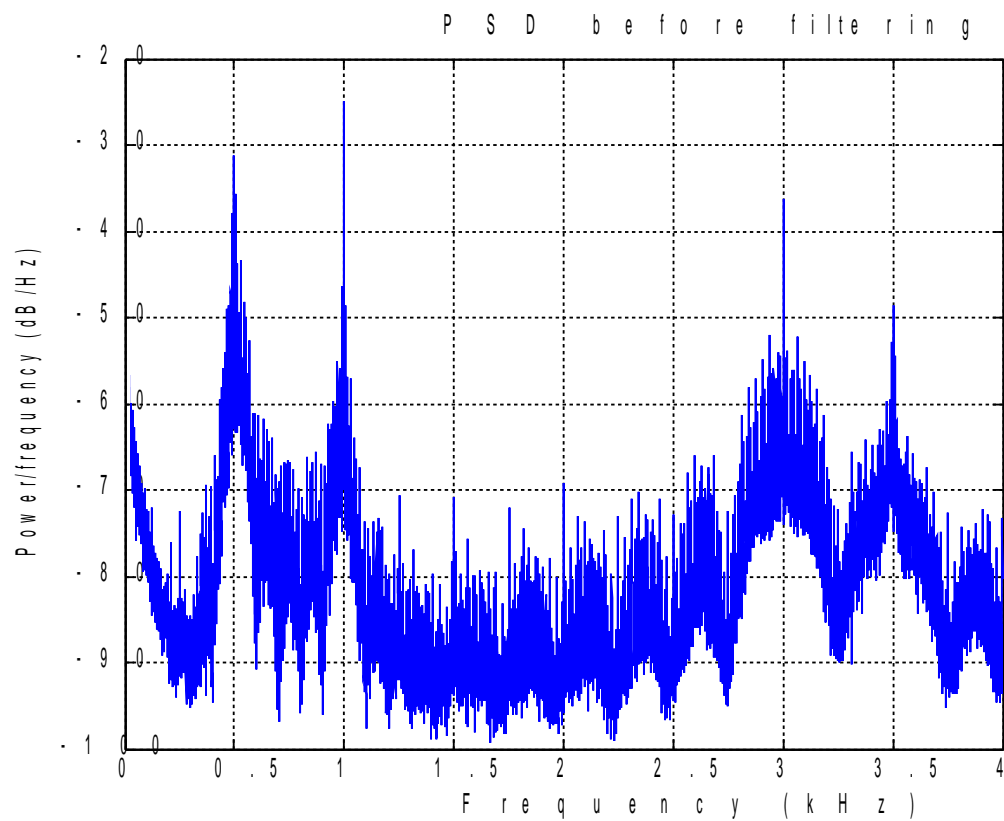
Dando-se um zoom pode-se ver que quando o bit correto é 1, a correlação do bit 0 vai para 0 e vice-versa. Esse é o caso ideal.



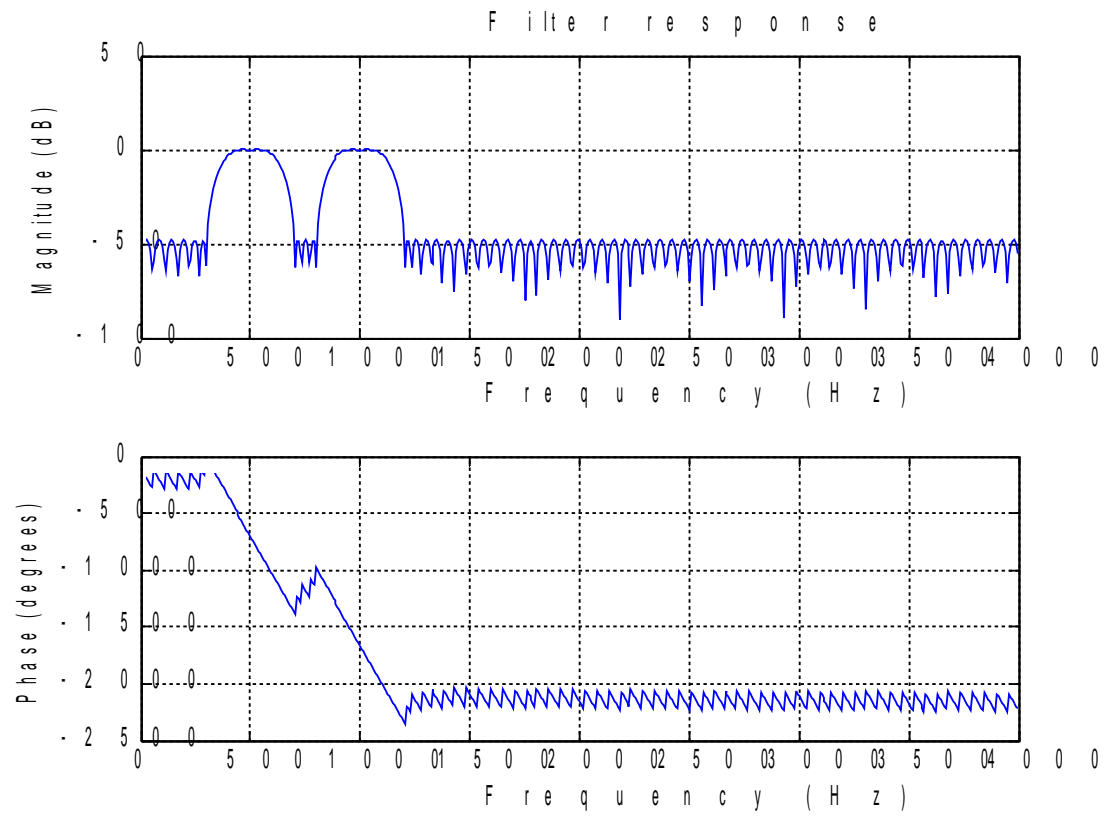
Depois disso o Matlab

Transmitindo 36 segundos de sinal  
Taxa 100 bits por segundo

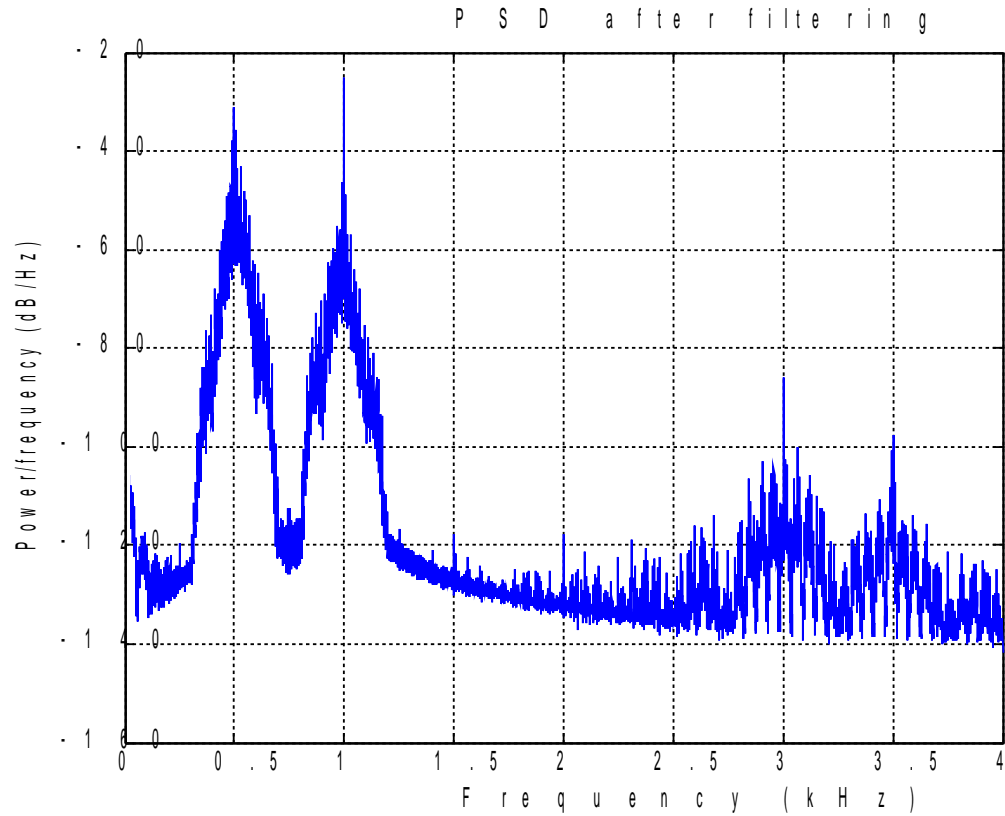
Após gravar o arquivo WAV, pode-se executar o código fskdemod.m. Ele irá mostrar que a PSD do sinal recebido é:



Ou seja, o sistema pode ter criado imagens das frequências de 500 Hz e 1 kHz em 3 e 3.5 kHz que precisam ser eliminadas. Um filtro FIR com as características abaixo foi então criado.



Daí a PSD na saída do filtro é:

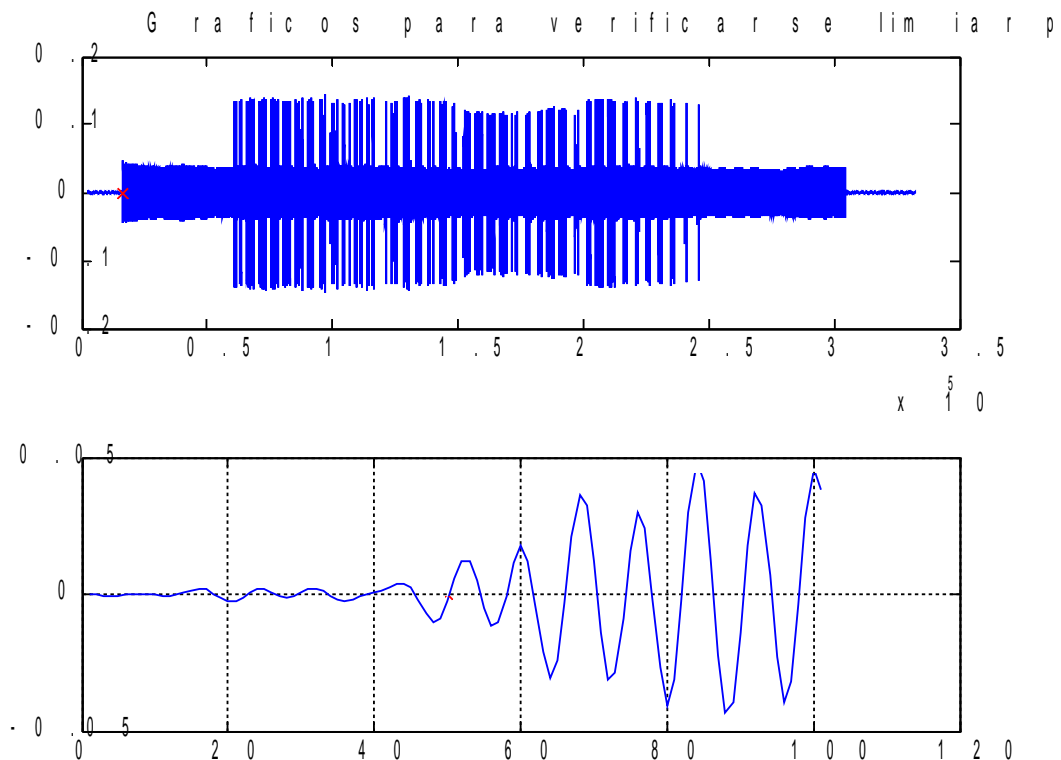


O software mostra então o valor do início de acordo com o limiar estimado e pede a eventual correção por parte do usuário:

início =

16513

Digite o valor para o início da mensagem (em numero de amostras). O sugerido estah acima. O gráfico abaixo ajuda a decisão, onde o símbolo x em vermelho indica a amostra de início.



No caso em questão, houve erros na transmissão e a BER foi de 17.75%. Esses erros aparecem na imagem recebida (abaixo). Imaginando a varredura raster-scan, da esquerda pra direita e de cima para baixo, nota-se que os erros ficaram concentrados em parte de transição rápida da imagem (próximo ao Le da palavra Leão).



i m a g e m r e c e b i d a

