Licenciatura em Engenharia Informática

Sistemas Multimédia

Amostragem de Sinais

Telmo Reis Cunha

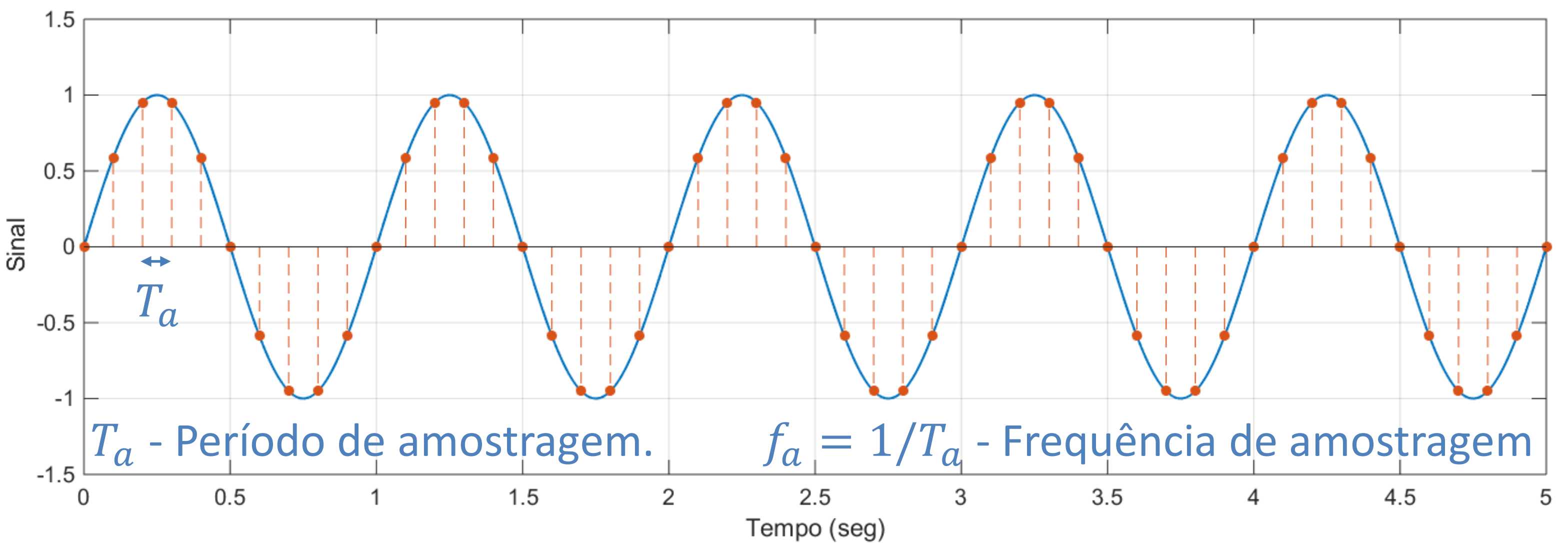
Departamento de Eletrónica, Telecomunicações e Informática
Universidade de Aveiro – 2018/2019

1. Fundamentos da Amostragem de Sinais

- O tratamento e análise de sinais com recurso a processadores digitais é efetuado no domínio de tempo discreto.

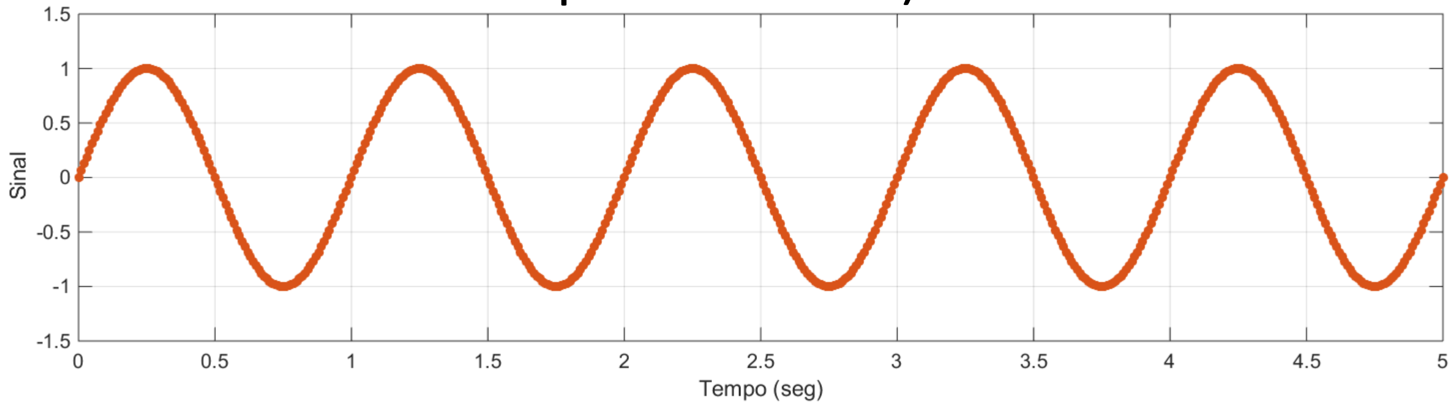
- Tal requer, naturalmente, que os sinais que evoluem continuamente ao longo do tempo sejam **amostrados**.

- O processo de amostragem consiste em adquirir amostras do sinal (tipicamente a um ritmo periódico).

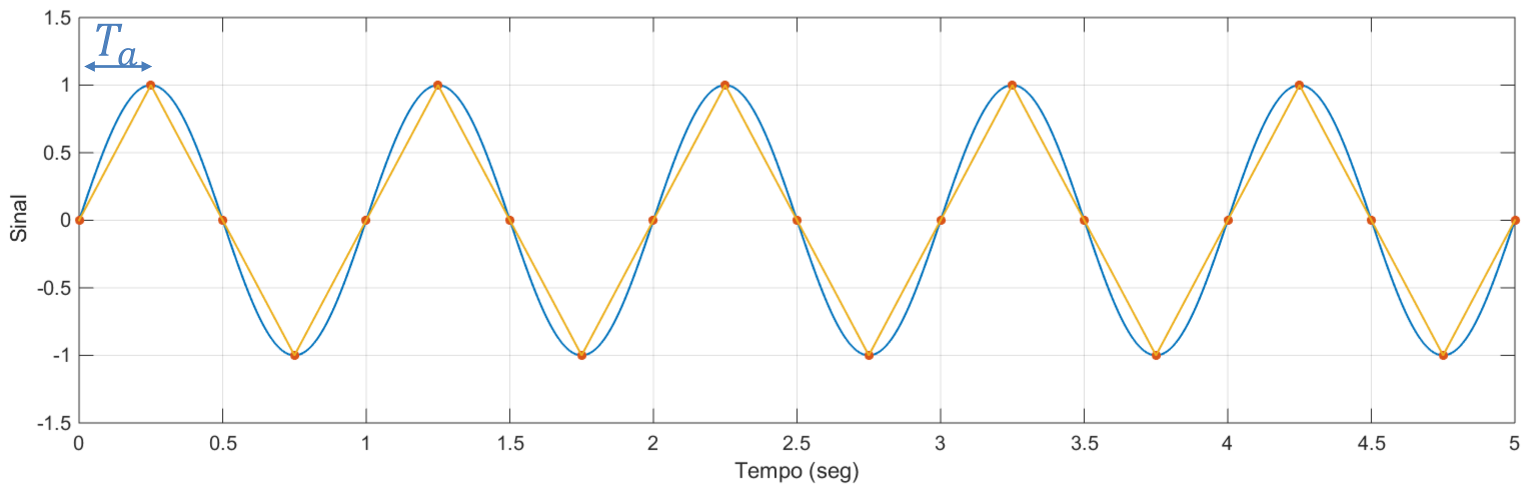


1. Fundamentos da Amostragem de Sinais

- Muitas amostras requerem muita memória (e há pouca "novidade" de amostra para amostra).

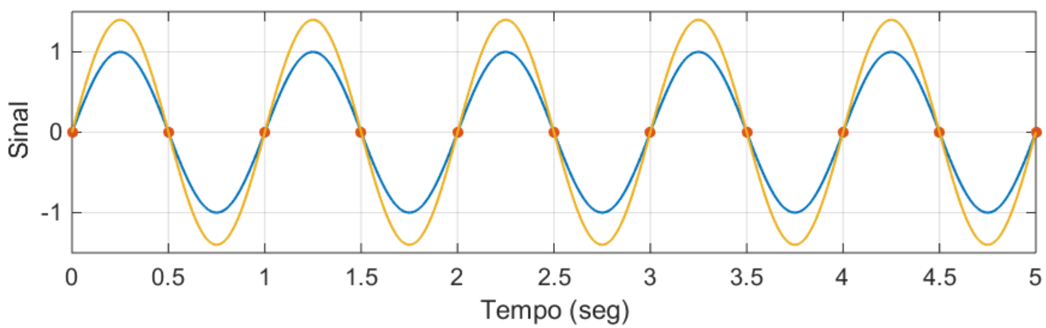


- Poucas amostras podem não ser suficientemente representativas.

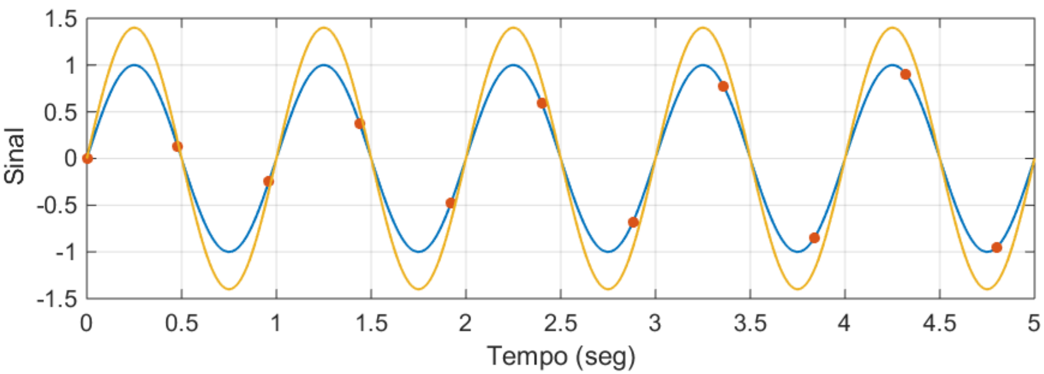


1. Fundamentos da Amostragem de Sinais

- A teoria da amostragem refere que para que um sinal seja amostrado sem perda de informação (i.e., podendo ser integralmente reconstruído a partir das suas amostras), terá que considerar uma frequência de amostragem superior ao dobro da frequência mais elevada do sinal – **Critério de Nyquist**.



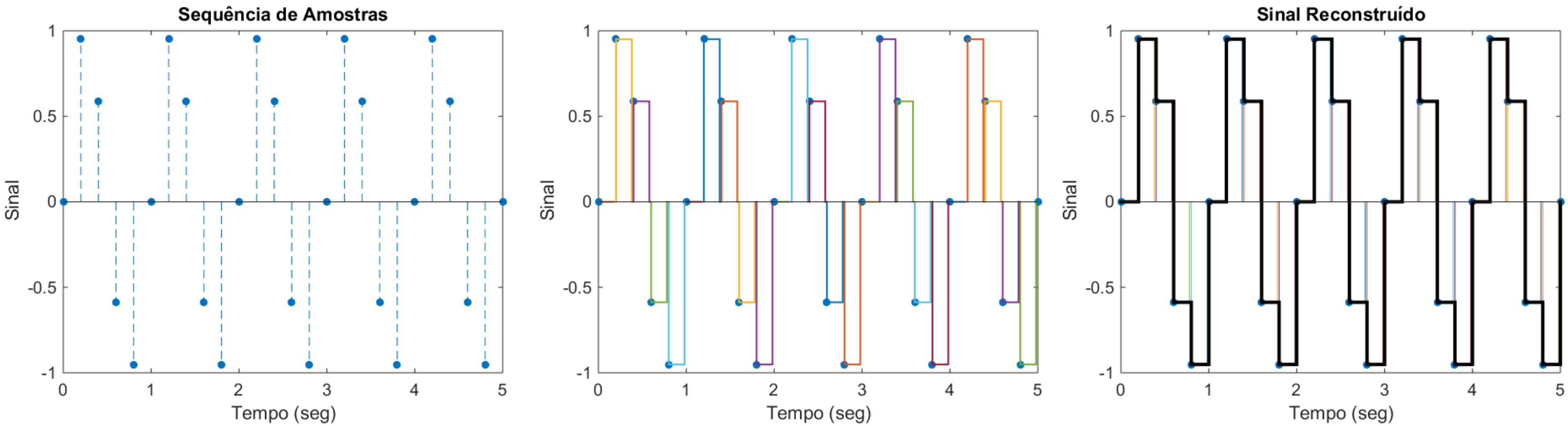
$f_a = 2f_{max}$ - As amostras não distinguem as duas sinusoides (nem outra da mesma frequência; nem o sinal constante igual a 0).



f_a ligeiramente superior a $2f_{max}$ - Ao fim de algum tempo (algumas amostras), a informação do sinal representado está completamente identificada.

2. Reconstrução de Sinais Amostrados

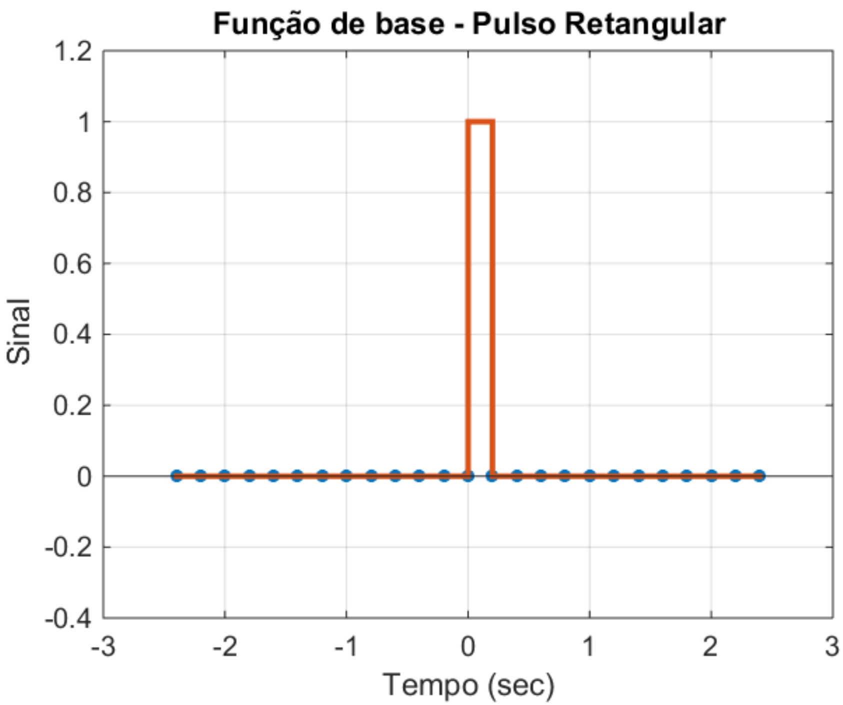
- Seja $x(n), n = 1, \dots, N$, a sequência de amostras de um sinal, com período de amostragem T_a. Como se pode reconstruir de novo o sinal de tempo contínuo, que fora alvo da amostragem?
- É fácil verificar que tal não pode ser feito usando um pulso (pedestal) para cada amostra:

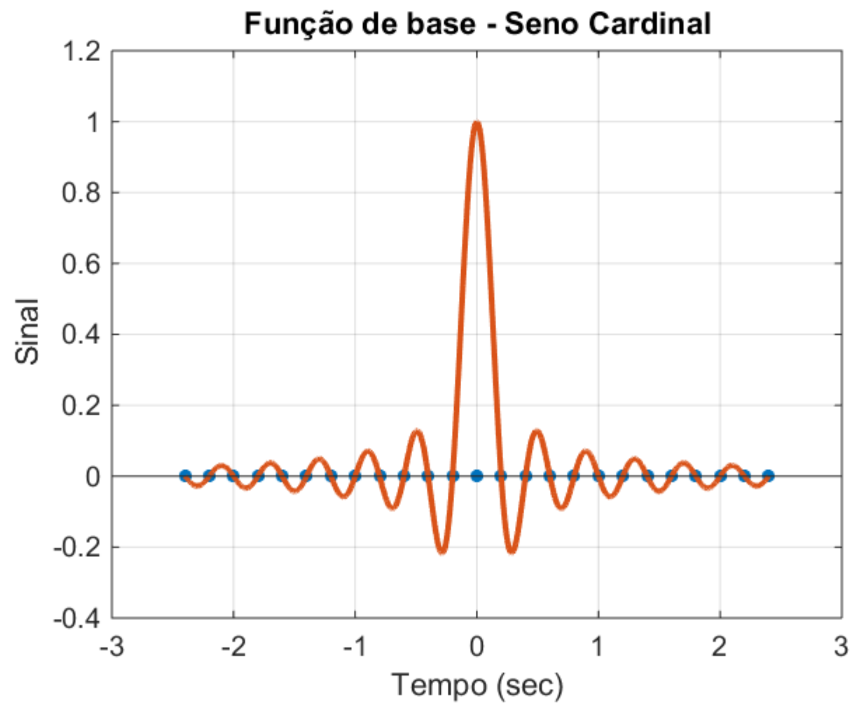


- O sinal resultante tem claramente uma frequência máxima superior a metade da frequência de amostragem.

2. Reconstrução de Sinais Amostrados

- A reconstrução ideal deve considerar a função **seno cardinal** como função de base, em vez de pulsos retangulares.

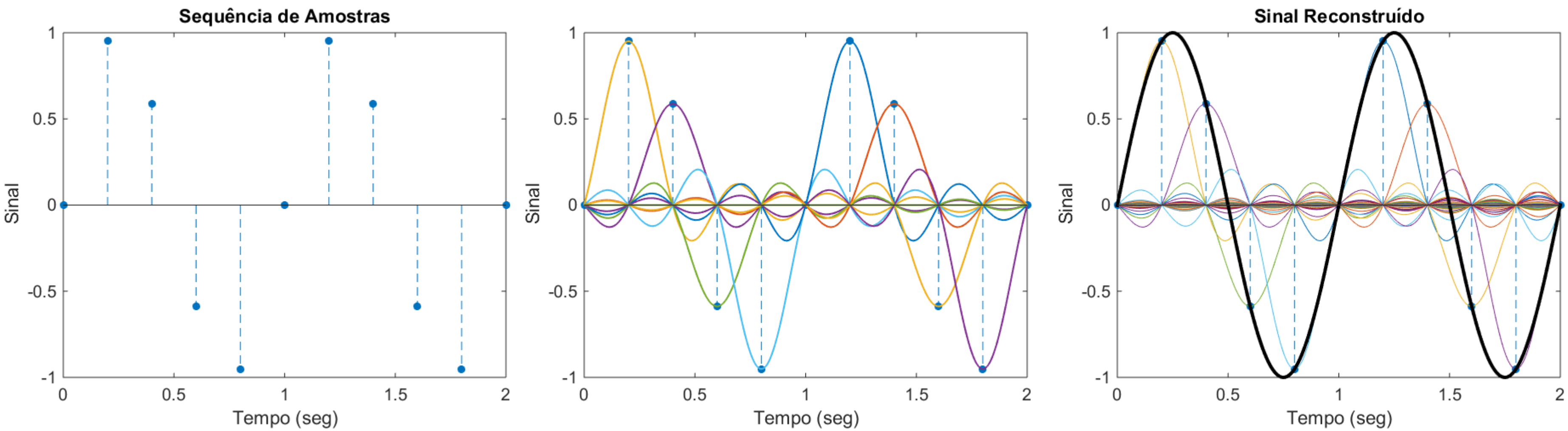




$$h(t) = \frac{\sin\left(2\pi\left(\frac{f_a}{2}\right)t\right)}{2\pi\left(\frac{f_a}{2}\right)t} = \frac{\sin(\pi f_a t)}{\pi f_a t}$$

2. Reconstrução de Sinais Amostrados

- A reconstrução ideal deve considerar a função **seno cardinal** como função de base, em vez de pulsos retangulares.



- O sinal original considerado neste teste foi: $x(t) = \sin(2\pi t)$.

3. Aliasing

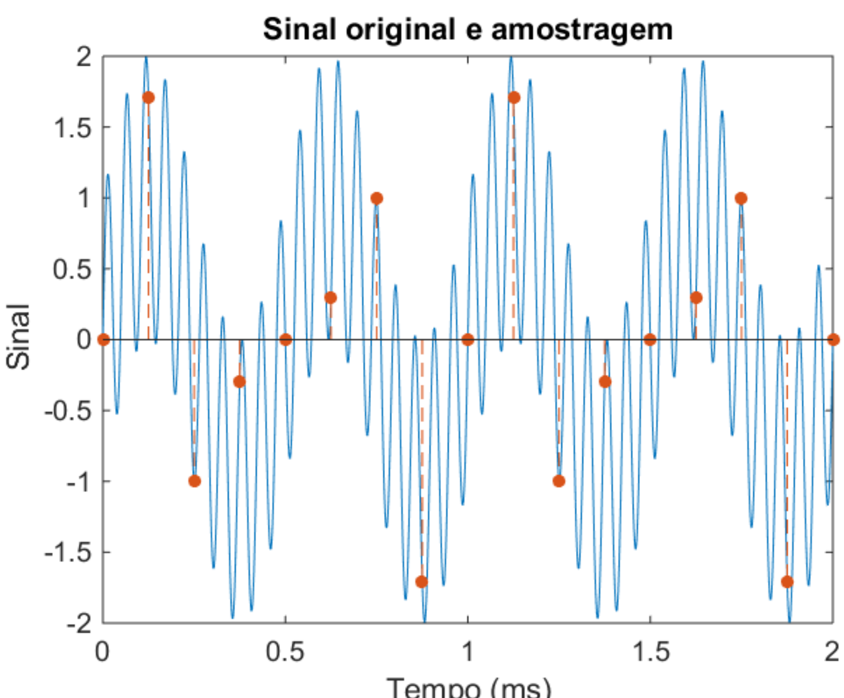
- O que sucede, então, quanto a amostragem não obedece ao Critério de Nyquist? Qual o resultado da reconstrução do sinal neste caso?
- Exemplo de uma imagem submetida a sub-amostragem:

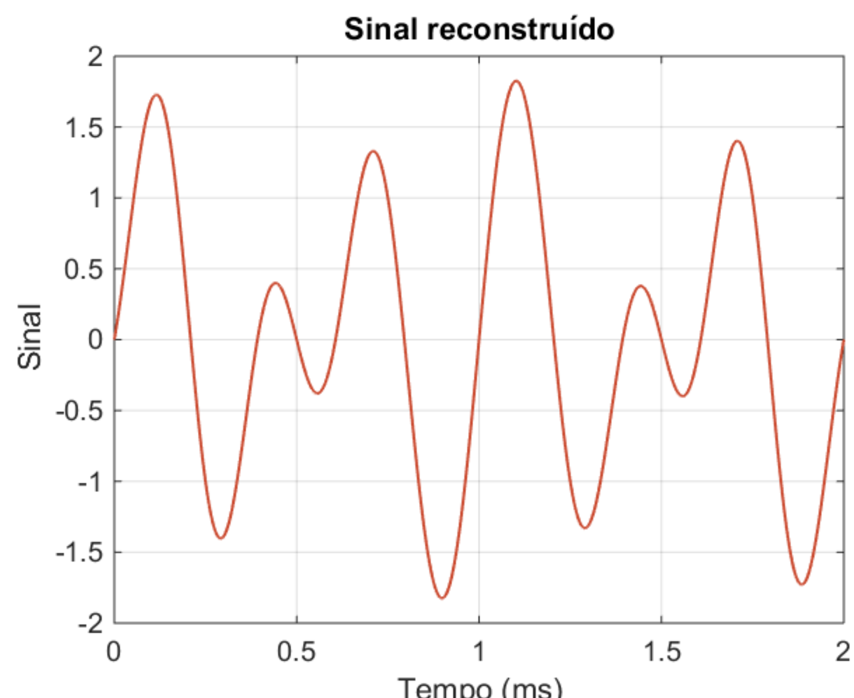
Imagem original

Imagem sub-amostrada (8x8 vezes)

3. Aliasing

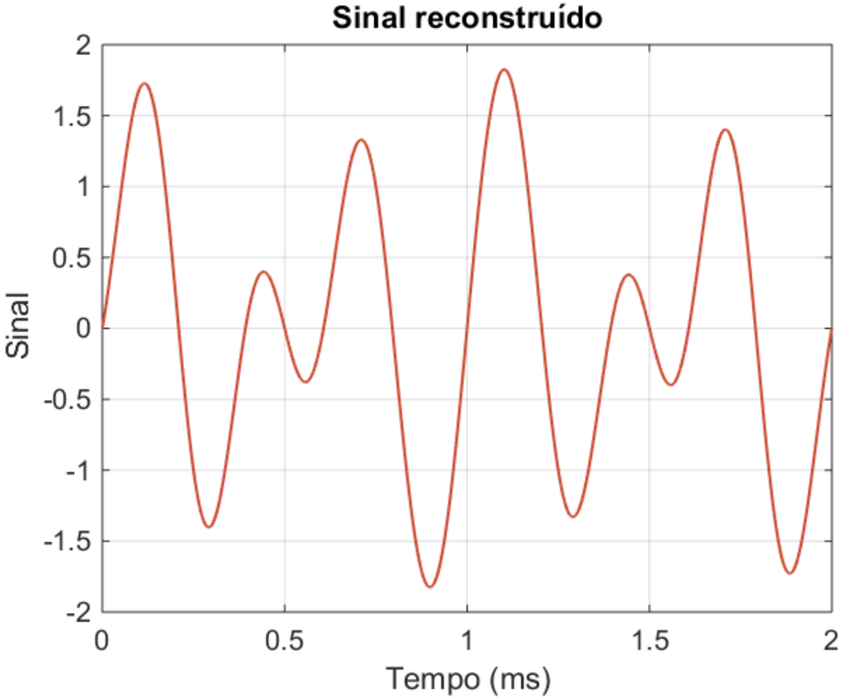
- Como exemplo, considere-se o caso de um sinal composto por dois tons, um correspondente a um som audível ($f_1 = 2kHz$) e outro a um som inaudível ($f_2 = 19kHz$), tendo ambos a mesma potência.
- A frequência de amostragem foi considerada para o sinal audível, tendo sido escolhida $f_a = 8kHz$.

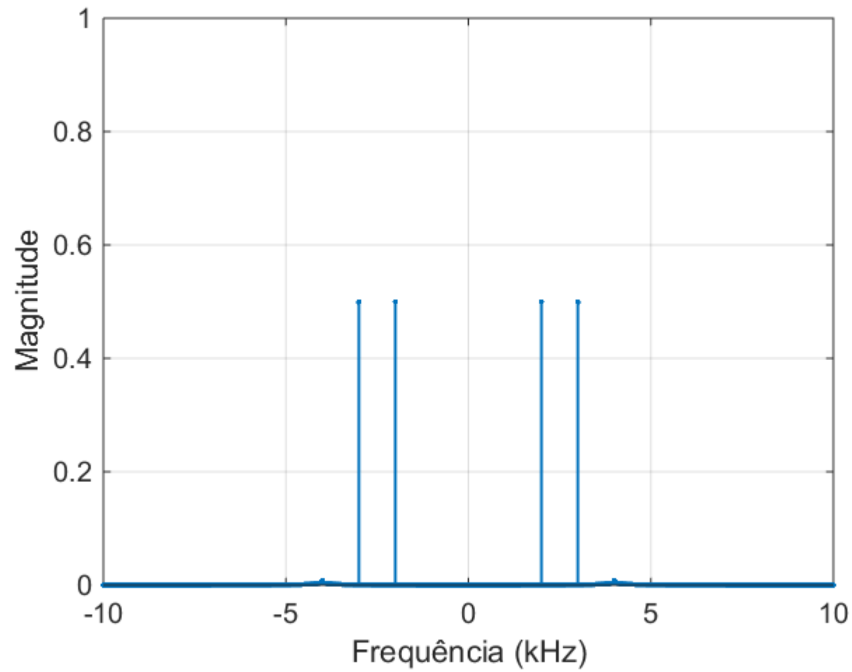




3. Aliasing

- Neste exemplo, o sinal amostrado e reconstruído tem dois tons audíveis:



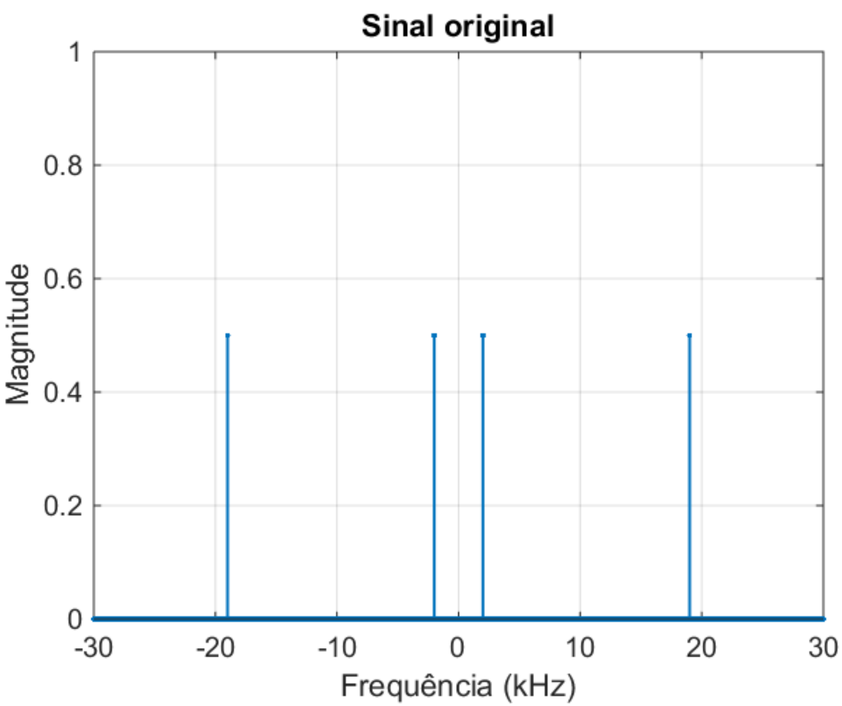


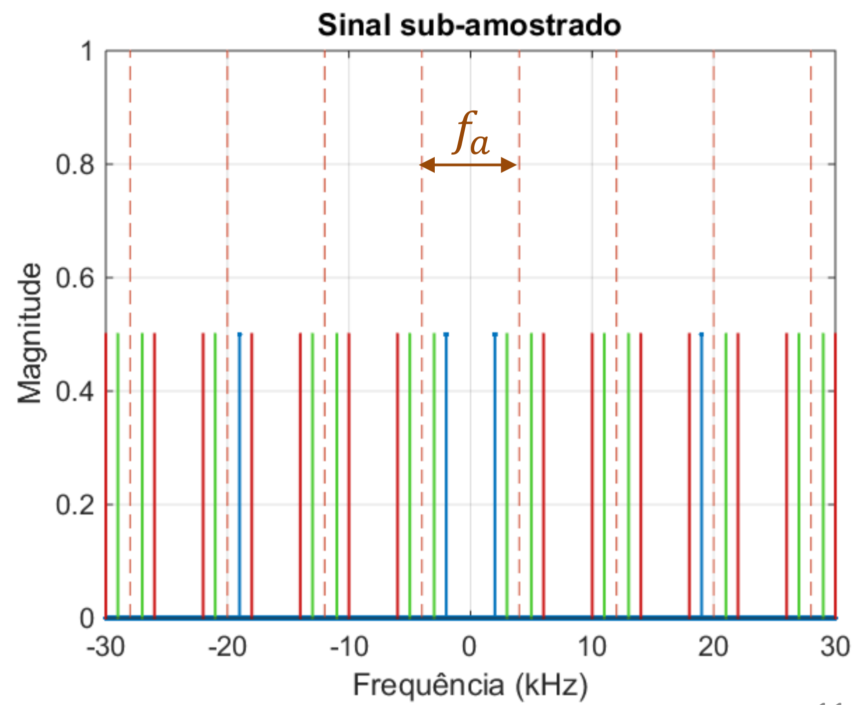
- Para se evitar este fenómeno, o processo de amostragem deve ser precedido de um **filtro *anti-aliasing*** (elimina as componentes de frequência superiores a $f_a/2$).

3. Aliasing

- O *aliasing* deve-se ao facto de o espetro do sinal se repetir em cada intervalo de largura f_a.

- No exemplo anterior do sinal de dois tons.





3. Aliasing

- Exemplos de *aliasing* associados a vídeo:
- https://www.youtube.com/watch?v=ByTsISFXUoY
- https://youtu.be/Ce_jRfM9b4k

4. Quantização da Amplitude dos Sinais

- Nos sistemas de aquisição de sinais, não é só o tempo que é discretizado – também a amplitude dos sinais é discreta.

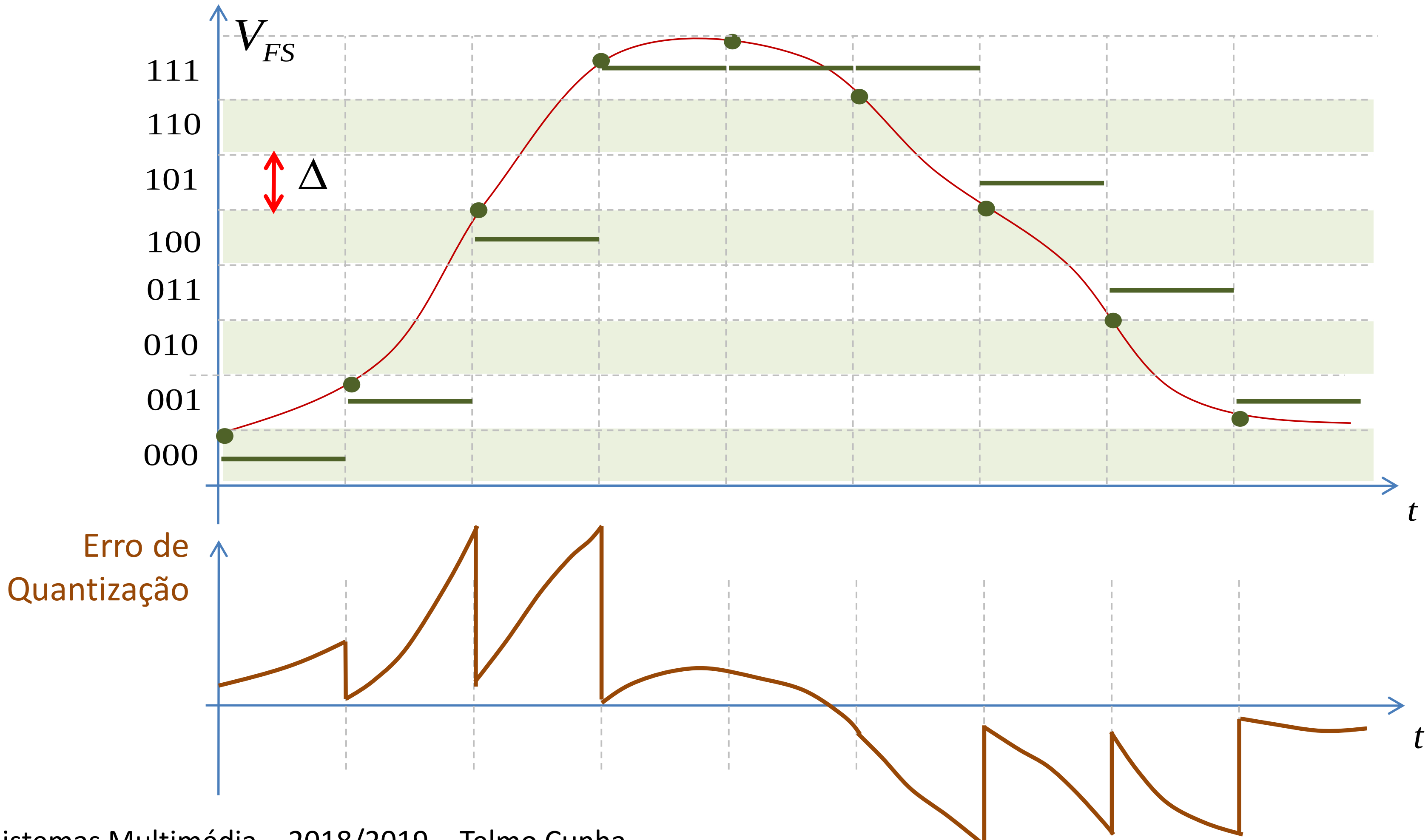
- Cada amostra do sinal é adquirida, e armazenada, numa palavra digital que apresenta um limite de número de bits.

- Logo, cada amostra adquire um valor de um conjunto (finito) de códigos possíveis.

Número de Bits	Número de Códigos
3	8
4	16
5	32
…	…
N	2^N

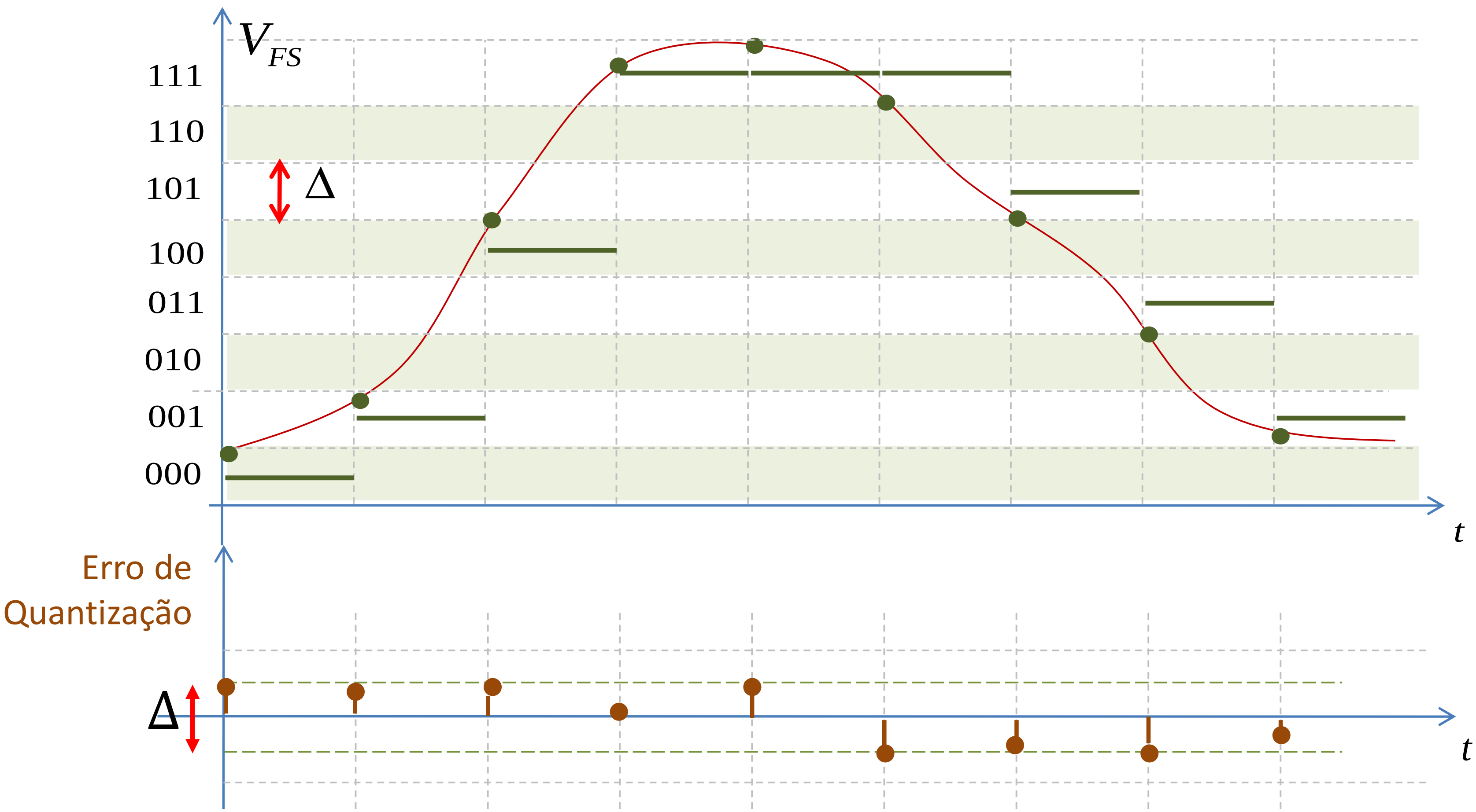
4. Quantização da Amplitude dos Sinais

- Para ilustrar o efeito da quantização da amplitude dos sinais, considere-se o caso do armazenamento em código de 3 bits:



4. Quantização da Amplitude dos Sinais

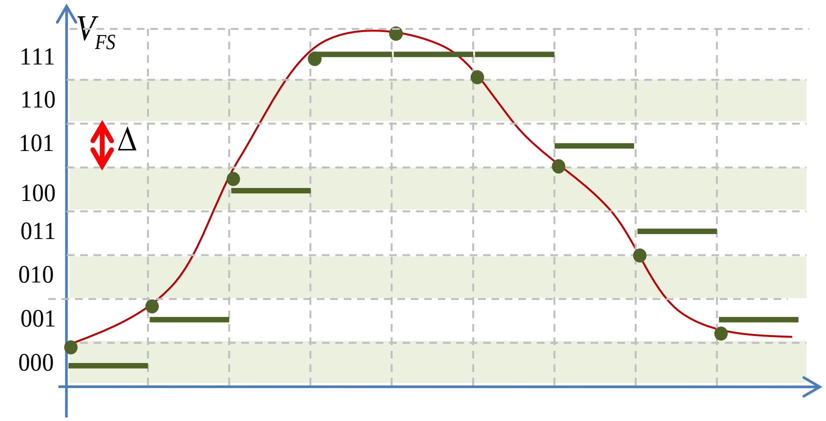
- Para ilustrar o efeito da quantização da amplitude dos sinais, considere-se o caso do armazenamento em código de 3 bits:



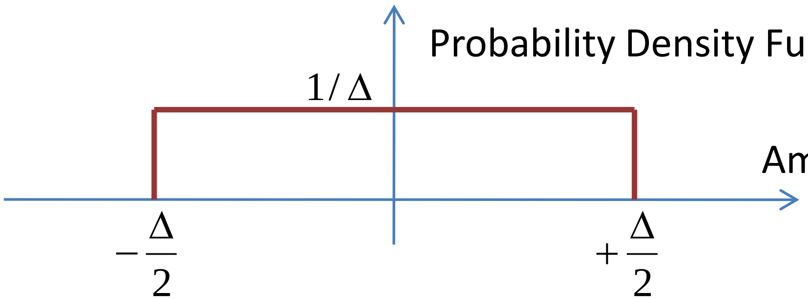
4. Quantização da Amplitude dos Sinais

- Cada código corresponde a um intervalo da amplitude igual a:

$$\Delta = \frac{V_{FS}}{2^N}$$



- Admitindo que o valor do erro de quantização pode adquirir, com igual probabilidade, qualquer valor nesse intervalo:



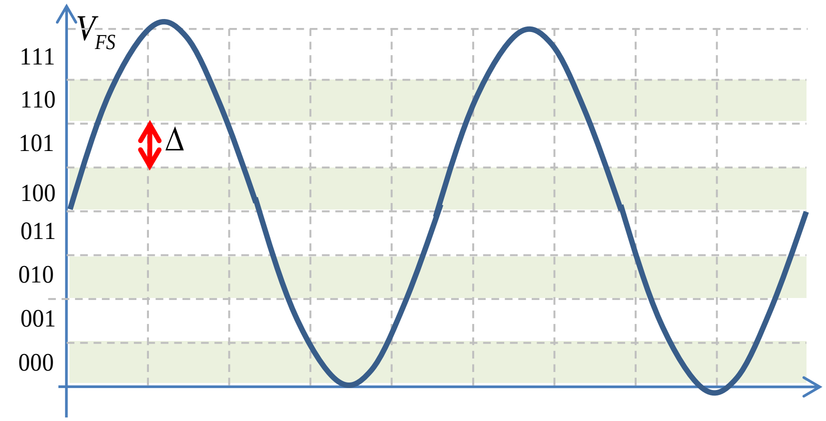
- Então, a potência (normalizada) do erro de quantização é:

$$P(\varepsilon) = \lim_{T\to\infty} \frac{1}{T} \int_{-T/2}^{T/2} \varepsilon^2(\tau)\, d\tau = \int_{-\infty}^{+\infty} \varepsilon^2\, pdf(\varepsilon)\, d\varepsilon = \frac{1}{\Delta} \int_{-\Delta/2}^{+\Delta/2} \varepsilon^2\, d\varepsilon \rightarrow \boxed{P(\varepsilon) = \frac{\Delta^2}{12}}$$

4. Quantização da Amplitude dos Sinais

- Considerando um sinal sinusoidal que percorre toda a escala de amplitude:

$$x(t) = \frac{V_{FS}}{2}\sin(\omega t)$$



- A potência associada a essa onda sinusoidal é, então:

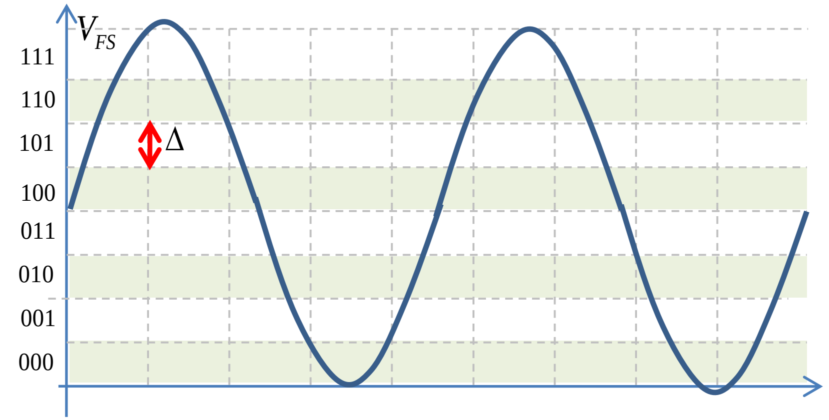
$$P_x = \frac{1}{T}\int_0^T x(t)^2 dt = \frac{1}{2}\cdot\left(\frac{V_{FS}}{2}\right)^2 = \frac{V_{FS}^2}{8}$$

- E a relação sinal-ruído (**SNR** – *Signal to Noise Ratio*) é:

$$\boxed{SNR = \frac{P_x}{P_\epsilon} = \frac{\frac{V_{FS}^2}{8}}{\frac{\Delta^2}{12}} = \frac{3}{2}\cdot\frac{V_{FS}^2}{\Delta^2}}$$

4. Quantização da Amplitude dos Sinais

- A relação sinal-ruído é uma métrica muito importante, pois indica o esforço necessário para se distinguir a informação útil contida num sinal, quando na presença de ruído.



- A relação sinal-ruído é usualmente apresentada em decibéis (dB):

$$SNR_{dB} = 10\log_{10}(SNR)$$

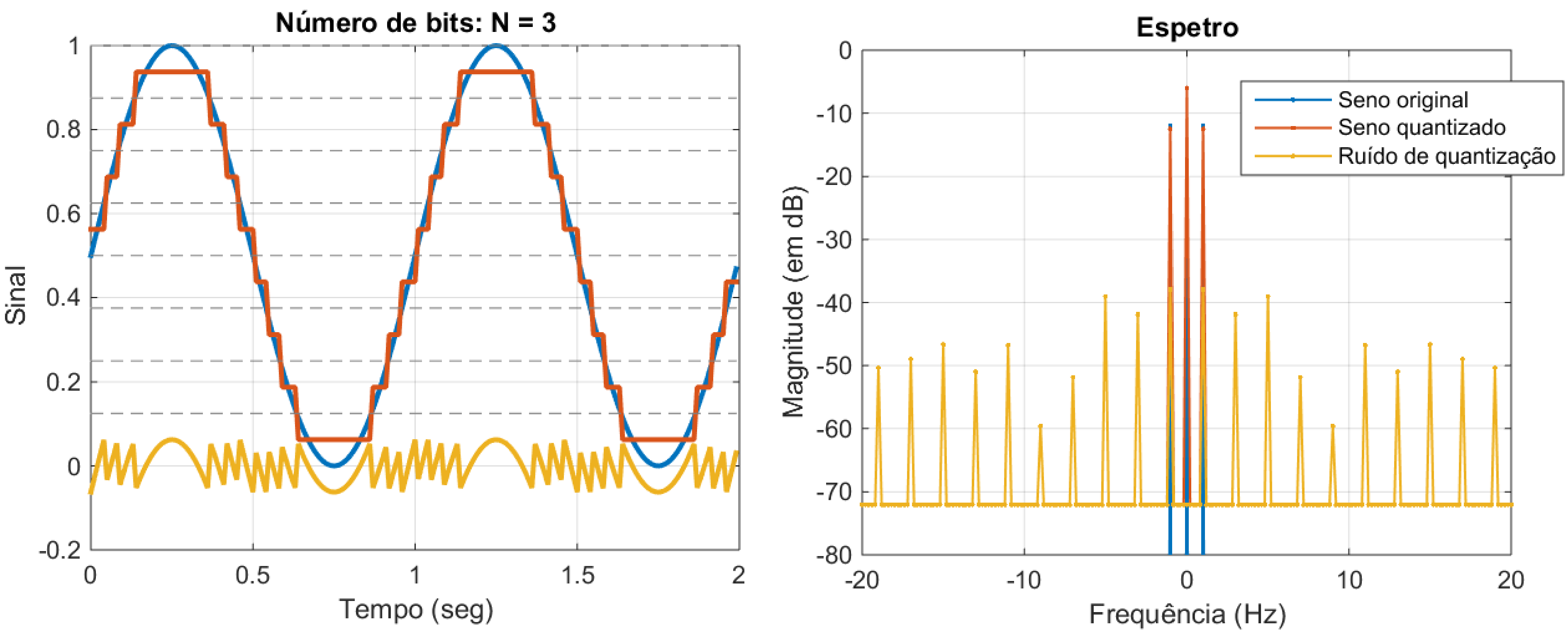
- Notando que $\Delta = V_{FS}/2^N$, verifica-se que a SNR entre a sinusoide de referência e o ruído de quantização é:

$$SNR_{dB} = 10\log_{10}\left(\frac{3}{2}\cdot 2^{2N}\right) = N\cdot 20\log_{10}(2) + 10\log_{10}\left(\frac{3}{2}\right) \rightarrow$$

$$\boxed{SNR_{dB} \approx 6.02N + 1.76}$$

4. Quantização da Amplitude dos Sinais

- Exemplo: Quantização de um seno com resolução de $N = 3$ bits:



4. Quantização da Amplitude dos Sinais

- Exemplo: Quantização de uma imagem:

Imagem original

RGB codificados com 8 bits

256^3=16.8 milhões de cores

RGB codificados com 2 bits

4^3=64 cores

RGB codificados com 1 bit

2^3=8 cores

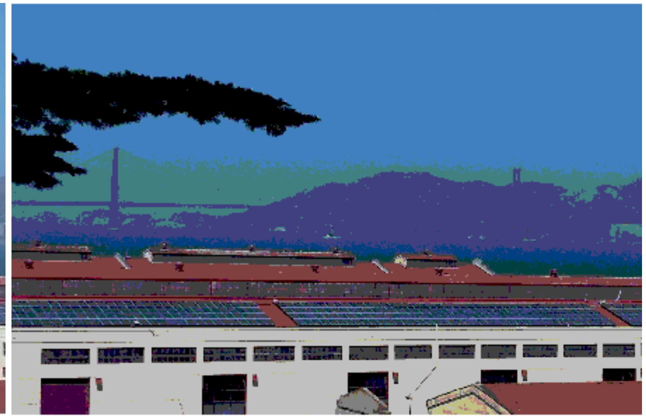
4. Quantização da Amplitude dos Sinais

- Exemplo: Quantização de uma imagem (maior diversidade de cores):

Imagem original
RGB codificados com 8 bits

256^3=16.8 milhões de cores

RGB codificados com 2 bits



4^3=64 cores

RGB codificados com 1 bit

2^3=8 cores

Tons de cinza (8 bits)

256 tons

Tons de cinza (2 bits)

4 tons

Tons de cinza (1 bit)

2 tons

5. Número de Bits Efetivos (ENOB)

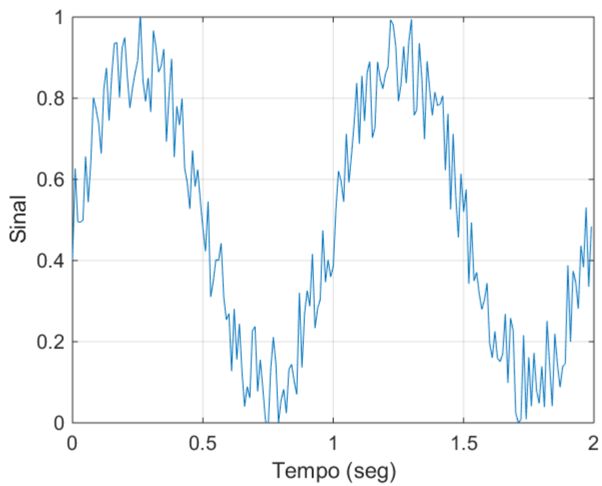
- A análise apresentada anteriormente pode ser colocada de forma inversa:
- Sendo verificada uma determinada relação sinal-ruído, qual é o número de bits que se obtém na representação da informação útil?
- Este é o conceito de Número de Bits Efetivos (**ENOB** – *Effective Number of Bits*):

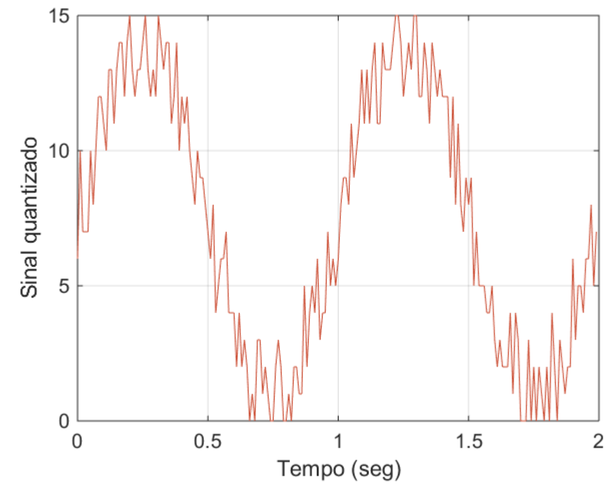
$$ENOB \approx \frac{SNR_{dB} - 1.76}{6.02}$$

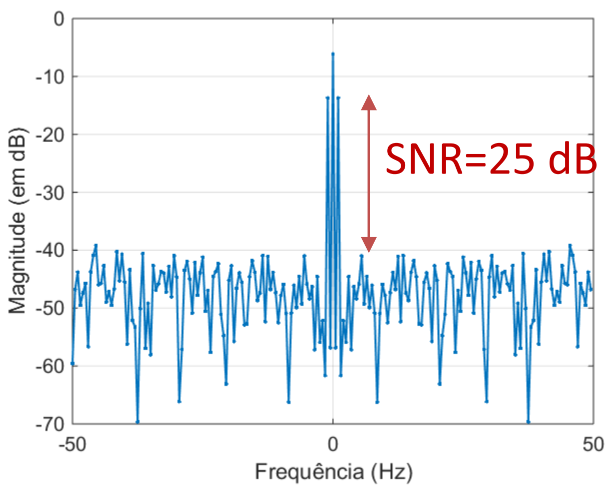
- Neste conceito, o ruído pode incluir diferentes componentes (quantização, ruído térmico, ruído de medida, de processo, ...).

5. Número de Bits Efetivos (ENOB)

- Exemplo da quantização (N=4 bits) de um sinal com ruído:







SNR=25 dB → ENOB=3.86 bit