

Universidade Federal de Campina Grande  
Departamento de Engenharia Elétrica  
Princípios de Comunicações

Edmar Candeia Gurjão  
ecandeia@dee.ufcg.edu.br

15 de dezembro de 2008

# Teorema da Amostragem

Seja um sinal  $g(t)$  limitado em banda em  $B$  Hz. Para amostrar esse sinal a uma taxa  $f_s$  Hz pode-se multiplicar  $g(t)$  por um trem de impulsos  $\delta_{T_s}(t)$ , ou seja

$$g_s(t) = g(t)\delta_{T_s}(t - nT_s).$$

A série de Fourier trigonométrica do trem de impulsos é dada por

$$\delta_{T_s}(t) = \frac{1}{T_s} [1 + 2 \cos \omega_s t + 2 \cos 2\omega_s t + 2 \cos 3\omega_s t + \dots]$$

$$\omega_s = \frac{2\pi}{T_s} = 2\pi f_s$$

dessa forma

$$\begin{aligned} g_s(t) &= g(t)\delta_{T_s}(t) \\ &= \frac{1}{T_s} [g(t) + 2g(t) \cos 2\omega_s t + 2g(t) \cos 3\omega_s t + \dots] \end{aligned}$$

Aplicando a transformada de Fourier em cada termo, tem-se que o primeiro termo é  $G(\omega)$ , o segundo termo é  $G(\omega - \omega_s) + G(\omega + \omega_s)$ , o terceiro termo é  $G(\omega - 2\omega_s) + G(\omega + 2\omega_s)$  e assim por diante, e que esses termos são o deslocamento do espectro de  $G(\omega)$

para  $\pm n\omega_s$ , o mesmo que é feito na modulação AM. O espectro de  $G_s(\omega)$  consiste de  $G(\omega)$  repetido periodicamente com período  $\omega_s = 2\pi/T_s$ , ou

$$G_s(\omega) = \frac{1}{T_s} \sum_{n=-\infty}^{\infty} G(\omega - n\omega_s). \quad (1)$$

como está representado na Figura 1

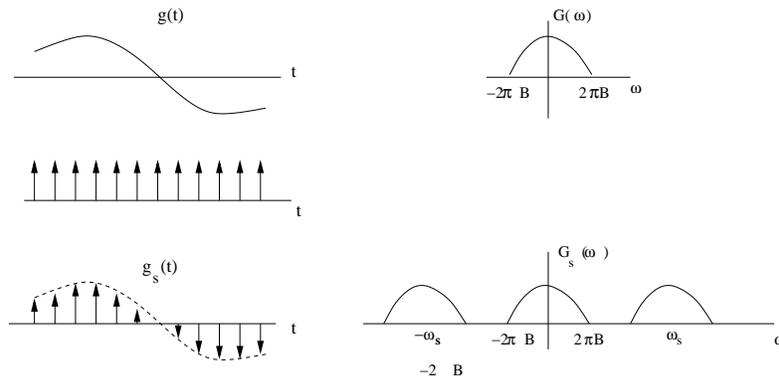


Figura 1:

Para reconstruir  $g(t)$  a partir de  $g_s(t)$  devemos conseguir recuperar  $G(\omega)$  de  $G_s(\omega)$ , isso só será possível se não houver sobreposição dos espectros de  $G_s(\omega)$ , o que implica em

$$f_s > 2B \text{ ou } T_s < \frac{1}{2B}$$

o valor  $f_s = 2B$  é conhecido como **taxa de Nyquist**.

A reconstrução do sinal pode ser feita passando  $g_s(t)$  por um filtro passa baixas de largura de banda  $B$  Hz, o que eliminaria as componentes deslocadas de  $n\omega_s$ ,  $n \neq 0$ , passando somente  $G(\omega)$ . O filtro com essa característica tem função de transferência

$$H(\omega) = T_s \text{rect} \left( \frac{\omega}{4\pi B} \right). \quad (2)$$

A resposta ao impulso  $h(t)$  é

$$h(t) = 2BT_s \text{sinc}(2\pi Bt)$$

e assumindo a taxa de amostragem de Nyquist, isto é,  $2BT_s = 1$  temos

$$h(t) = \text{sinc}(2\pi Bt).$$

Observe que  $h(t)$  é zero para todos os instantes de amostragem ( $t = \pm n/2B$ ) exceto em  $t = 0$ . De uma forma geral

$$\begin{aligned} g(t) &= \sum_k g(kT_s)h(t - kT_s) \\ &= \sum_k g(kT_s)\text{sinc}[2\pi B(t - kT_s)] \\ &= \sum_k g(kT_s)\text{sinc}(2\pi Bt - k\pi) \end{aligned} \quad (3)$$

que é conhecida como fórmula de interpolação que aproxima os valores de  $g(t)$  a partir das amostras.

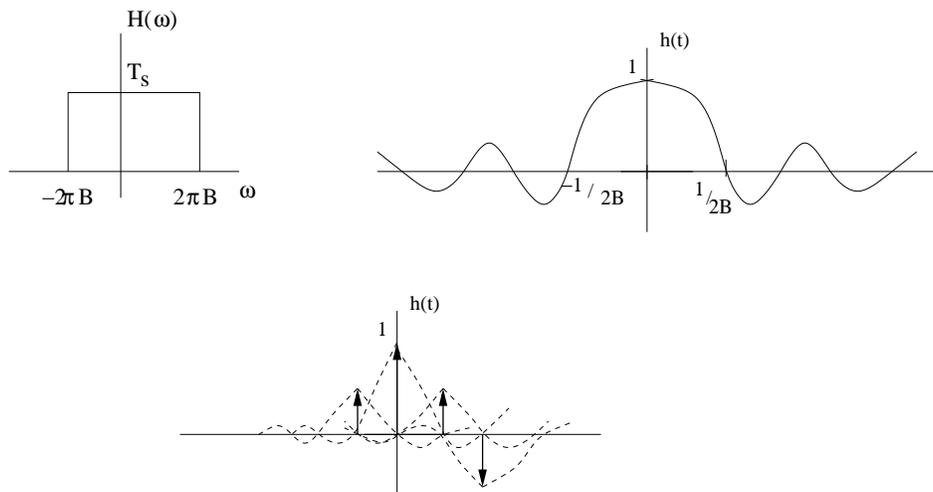


Figura 2:

Como a realização de um filtro ideal é impossível, ao utilizar um filtro realizável na saída teremos  $G(\omega)$  mais uma parte de outras componentes espectrais, fazendo com que o sinal recuperado não seja exatamente  $g(t)$ . Uma solução para isso é aumentar a frequência de amostragem pois com isso as cópias do espectro de  $G(\omega)$  ficarão mais espaçadas e pode-se utilizar filtros mais suaves. Mesmo assim, nem sempre pode-se obter um filtro que elimine somente  $G(\omega)$  ou realizar uma amostragem em taxas mais altas, logo conclui-se que a recuperação exata de  $g(t)$  a partir de suas amostras é muito difícil.

O teorema da amostragem foi demonstrado considerando que o sinal  $g(t)$  é limitado em banda, porém os sinais práticos são naturalmente ilimitados em banda, dessa forma, o espectro  $G_s(\omega)$  do sinal amostrado consistirá de ciclos de  $G(\omega)$  que se sobrepõem a cada  $f_s$  Hz. Devido a essa sobreposição  $G_s(\omega)$  não tem a informação completa sobre  $G(\omega)$ , logo teoricamente não seria possível recuperar  $g(t)$  de  $g_s(t)$ . Este problema é conhecido como *aliasing*.

Uma possibilidade para solucionar o problema do *aliasing* é filtrar o sinal  $g(t)$  antes que ele seja amostrado. Essa filtragem conhecida como *anti-aliasing* deve eliminar as componentes de frequências mais altas do sinal.

Outra dificuldade é que o teorema foi demonstrado utilizando uma sequência de impulsos para amostrar o sinal  $g(t)$ , e como é sabido os impulsos também são fisicamente irrealizáveis. Entretanto, pode-se utilizar um trem de pulsos para obter as amostras de  $g(t)$ , desde que a frequência de Nyquist seja preservada. Para tanto, observe que um trem de pulsos sendo um

sinal periódico pode ser expresso como

$$p_{T_s}(t) = C_0 + \sum_{n=1}^{\infty} C_n \cos(n\omega_s t + \theta_n), \quad \omega_s = \frac{2\pi}{T_s}$$

e

$$\begin{aligned} g_s(t) &= g(t)p_{T_s}(t) = g(t) \left[ C_0 + \sum_{n=1}^{\infty} C_n \cos(n\omega_s t + \theta_n) \right] \\ &= C_0 g(t) + \sum_{n=1}^{\infty} C_n g(t) \cos(n\omega_s t + \theta_n) \end{aligned} \quad (4)$$

e o sinal amostrado  $g_s(t)$  consiste de  $C_0 g(t)$ ,  $C_1 g(t) \cos(\omega_s t + \theta_1)$ ,  $C_2 g(t) \cos(2\omega_s t + \theta_1)$  e assim por diante. O termo desejado é  $c_0 g(t)$  e os demais estão deslocados de  $n\omega_s$  logo podem ser eliminados pelo filtro.

# Modulação por Codificação de Pulsos

Conhecida como PCM, do inglês *Pulse-Code Modulation* é a modulação em pulso mais utilizada. Basicamente, PCM é um método de converter um sinal analógico em um sinal digital (conversão A/D). Considerando que as amplitudes do sinal analógico  $m(t)$  variam no intervalo  $(-m_p, m_p)$ , esse espaço é particionado em  $L$  sub-intervalos, cada um de tamanho  $\Delta v = 2m_p/L$ , em seguida cada amostra é aproximada para um dos  $L$  pontos que esteja mais próximo. O sinal obtido é chamado de digital.

Como exemplo observe a Figura 3, nela uma senóide de frequência  $F_c$  é amostrada com uma frequência  $f_s \geq 2f_c$  em seguida são realizadas duas quantizações distintas, um com  $L = 2$  e outra  $L = 4$  níveis. Observe que para  $L = 2$  a senóide é aproximada por um sinal quadrado, e com  $L = 4$  mesmo não sendo uma aproximação exata já tem-se um formato de senóide.

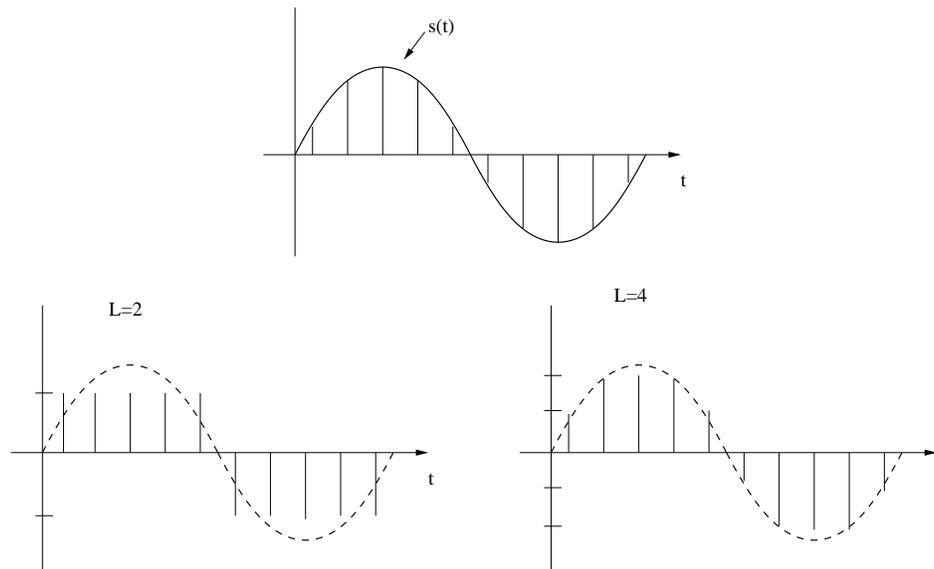


Figura 3:

O processo seguinte é associar a cada um dos  $L$  níveis um conjunto de bits. Se para cada nível for associada a sua representação em binário tem-se o código natural binário (NBC, do inglês *natural binary code*).

Como exemplo um sinal de áudio tem largura de banda de 15KHz, mas mostra-se que a inteligibilidade não é afetada se esse sinal for filtrado em 3400Hz e o sinal resultante é amostrado a 8000 amostras por segundo (8 KHz), maior que a taxa de Nyquist que seria de 6,8KHz, e cada amostra é quantizada em 256 níveis ( $L = 256$ ) que necessitam de um grupo de oito bits para cada amostra, assim um sistema telefônico

necessita de  $8 \times 8000 = 64000$  pulsos binários por segundo.

O CD (*compact disk*) o sinal de som não é filtrado e é utilizada uma taxa de amostragem de 44.1KHz, maior que o 30KHz necessários, e o sinal é quantizado em  $L = 65.536$  níveis para reduzir o erro de quantização. No DVD a taxa de amostragem pode chegar a 192KHz e cada amostra pode usar 24 bits podendo o sinal ser quantizado a  $L = 16.777.216$  níveis.

## Quantização

Consiste em aproximar os valores das amostras para níveis pré-definidos. Para isso vamos limitar a amplitude do sinal mensagem  $m(t)$  para o intervalo  $(-m_p, m_p)$ . Assim  $m_p$  não é um parâmetro do sinal mais do quantizador, em seguida divide-se o intervalo  $(-m_p, m_p)$  em  $L$  subintervalos, cada um com  $\Delta v = 2m_p/L$  e os valores amostrados são aproximados para o valor do intervalo que ele pertence. As amostras quantizadas são codificadas em seguida transmitidas por pulsos binários. No receptor tenta-se recuperar o sinal transmitido a partir das amostras.

# Quantização não Uniforme

Os intervalos de quantização sendo uniformes  $\Delta v = 2m_p/L$  trazem um problema, como o ruído de quantização  $N_q = (\Delta v)^2/12$  é proporcional a  $\Delta v$  se pudermos usar intervalos de quantização menor diminuiremos o ruído. Isso é feito usando intervalos menores para amplitudes menores do sinal de entrada e é conhecido como quantização não uniforme.

Pode-se realizar a quantização não uniforme de diversas maneiras, uma delas é conhecida como lei  $\mu$  utilizada no sistema de telefonia norte americano e japonês e o outro é conhecido como lei A usado no sistema europeu, brasileiro e no resto do mundo. A lei  $\mu$  para amplitudes positivas é dada por

$$y = \frac{1}{\ln(1 + \mu)} \ln \left( 1 + \frac{\mu m}{m_p} \right), \quad 0 \leq \frac{m}{m_p} \leq 1$$

A lei A para amplitudes positivas é dada por

$$y = \begin{cases} \frac{A}{1 + \ln A} \left( \frac{m}{m_p} \right) & 0 \leq \frac{m}{m_p} \leq \frac{1}{A} \\ \frac{A}{1 + \ln A} \left( \frac{Am}{m_p} \right) & \frac{1}{A} \leq \frac{m}{m_p} \leq 1 \end{cases}$$

Os parâmetro  $\mu$  ou  $A$  determinam o grau de compressão.

No receptor as amostras devem ser restauradas para o seu valor original e para isso usa-se um expensor com uma característica complementar a do compressor.

## Largura de banda

Em PCM associa-se um grupo de  $n$  bits a cada um dos  $L$  níveis de quantização, ou seja,

$$L = 2^n \quad \text{ou} \quad n = \log_2 L.$$

Como o sinal  $m(t)$  limitado em  $B$  Hz requer no mínimo  $2B$  amostras por segundo, necessita-se de  $2nB$  bits por segundo. Sabendo que uma largura de banda de 1Hz pode transmitir no máximo duas peças de informação por segundo, necessita-se de

$$B_T = nBHz$$

de largura de banda para transmitir um sinal PCM.

# Multiplexação por Divisão no Tempo

Um sinal amostrado terá valor zero entre as amostras, os intervalos de tempo entre as amostras podem ser usados para transmitir outros sinais, esse é o princípio da multiplexação por divisão no tempo (TDM, do inglês *Time Division Multiplexing*).

Um esquema TDM simples pode ser visto na Figura 4. Várias entradas são pré-filtradas pelos filtros passa baixas e em seguida são amostradas. O amostrador rotatório ou comutador extrai uma amostra de cada entrada por volta e essa amostras podem ser enviadas por um PAM. No receptor, um equipamento chamado de distribuidor irá entrega cada amostra a uma saída para que em seguida os sinais sejam reconstruídos.

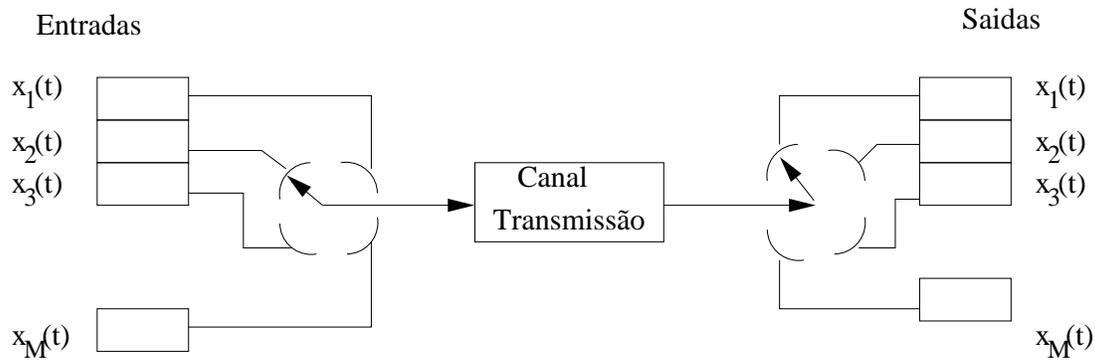


Figura 4: Representação de um sistema TDM.

Se todas as entrada tem a mesma largura de banda  $B$ , o comutador deve girar a uma taxa  $f_s \geq 2B$  de tal sorte que as amostras sucessivas de qualquer uma das entradas é espaçada de  $T_s = 1/f_s$ . O intervalo  $T_s$  contém uma amostra de cada entrada e é chamado de quadro. Se existem  $M$  canais de entrada, o espaço de um pulso deve ser  $T_s/M$ , então o número total de pulsos por segundos é

$$r = M f_s \geq 2MB$$

que representa a taxa de sinalização de um sinal TDM.

# Sistemas de Modulação Digital

Vimos em capítulos anteriores que um sinal pode ser transmitido usando a modulação em amplitude ou em ângulo. Naquele momento, o sinal mensagem  $m(t)$  era analógico. Agora que já sabemos como digitalizar um sinal vamos analisar como são realizadas as modulações com sinais mensagens digitais.

Inicialmente vamos considerar um conjunto de amplitudes discretas  $A_m = (2m - 1 - M)d$ ,  $m = 1, 2, \dots, M$ , sendo  $d$  a distância euclidiana entre dois pontos,  $M$  a quantidade de amplitudes consideradas, e o sinal

$$g_m(t) = A_m p_T(t)$$

sendo  $p_T(t)$  um pulso cujo formato define as características do sinal  $g_m(t)$ . Considera-se que o espectro de  $g_m(t)$  é limitado em  $BHz$ , dado pela densidade espectral de  $p_T(t)$ .

Para transmitir os sinais  $s_m(t)$  vamos multiplicá-los por uma portadora  $\cos \omega_c t$ , obtendo

$$s_m(t) = A_m p_T(t) \cos \omega_c t$$

Quando o pulso  $p_T(t)$  é retangular temos

$$p_T(t) = \begin{cases} \sqrt{\frac{2}{T}}, & 0 \leq t \leq T \\ 0, & \text{c.c.} \end{cases} \quad (5)$$

temos sinal chamado de *Amplitude Shift Keying* (ASK). Na frequência temos

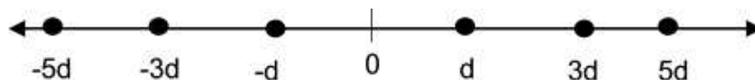
$$S_m(\omega) = \frac{A_m}{2} [P_T(\omega - \omega_c) + P_T(\omega + \omega_c)]$$

que é basicamente o espectro de um sinal AM-DSB.

Observe ainda que podemos representar o sinal ASK como

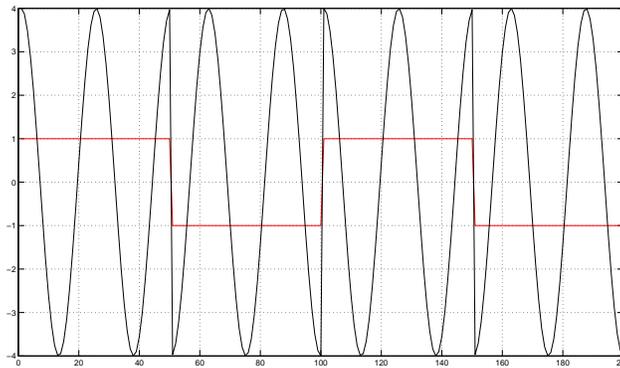
$$s_m(t) = \hat{s}_m \psi(t)$$

sendo  $\psi(t) = g_T(t) \cos \omega_c t$ , e  $\hat{s}_m = A_m$ . Dessa forma podemos representar os sinais por um digrama onde são marcados os valores  $\hat{s}_m$  em uma reta, conhecido como constelação. Como está representado na Figura abaixo.



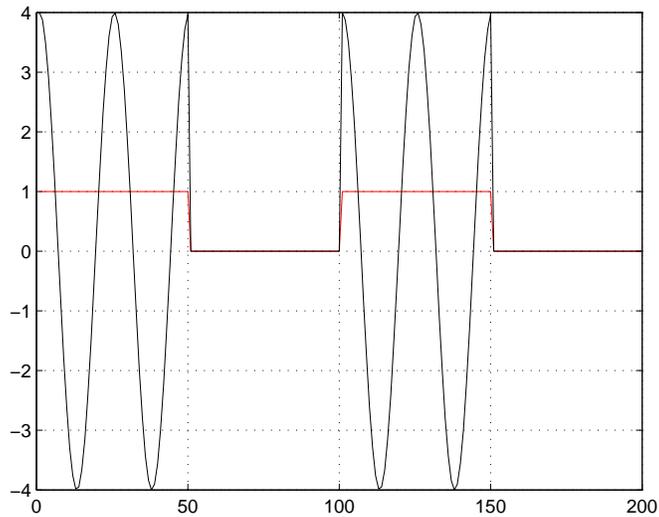
Como exemplo seja  $M = 2$ , nesse caso temos  $A_1 = -d$  e  $A_2 = d$ , e associamos o bit 0 a  $A_1$  e o bit 1 a  $A_2$ .

Esse sinal, no tempo é mostrado na Figura abaixo. Observe que nas mudanças de nível ocorre uma mudança de fase.



De uma forma geral, para  $A_m = (2m - 1 - M)d$ , teremos um chaveamento de fase.

Quando  $M = 2$  normalmente prefere-se fazer  $A_1 = 0$  e  $A_2 = 1$ , e tem-se um caso específico do ASK conhecido como chaveamento liga-desliga (OOK, do inglês *on-off keying*).



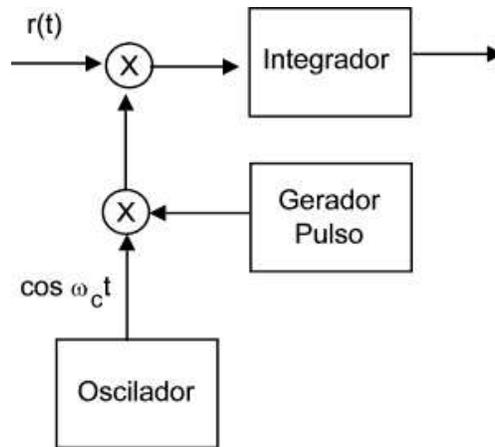
No receptor, chegará o sinal

$$\begin{aligned}
 r(t) &= s(t) + n(t) \\
 &= A_m p_T(t) \cos \omega_c t + n(t)
 \end{aligned} \tag{6}$$

sendo  $n(t)$  o ruído aditivo Gaussiano branco, que pode ser decomposto em

$$n(t) = n_c(t) \cos \omega_c t - n_s(t) \sin \omega_c t$$

Utilizando um receptor baseado em correlator, mostrado abaixo



que produzirá na saída

$$\int_{-\infty}^{\infty} r(t)\psi(t)dt = A_m + n = \hat{s}_m + n$$

O ruído tem média zero e variância dada por

$$\sigma_n^2 = \int_{-\infty}^{\infty} |\Psi(\omega)|^2 \delta_n(\omega) d\omega$$

sendo  $\Psi(\omega)$  a transformada de Fourier de  $\psi(t)$  e  $\delta_n(\omega)$  a densidade espectral de potência do ruído aditivo, dadas por

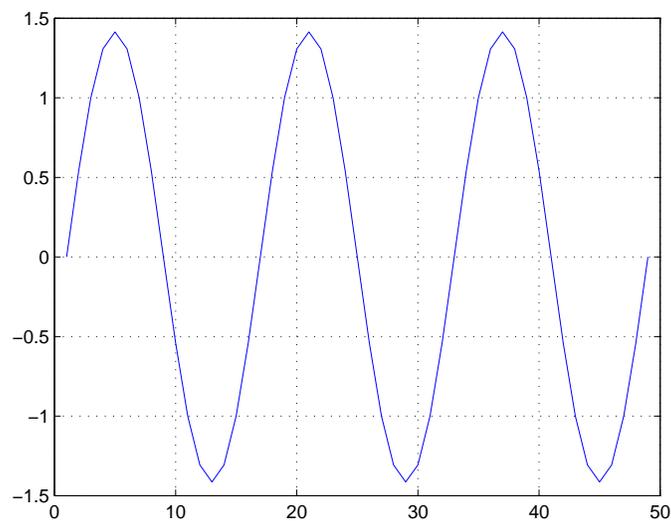
$$\Psi(\omega) = \frac{1}{2} [G_T(\omega - \omega_c) + G_T(\omega + \omega_c)]$$

e a densidade espectral de potência do ruído dada por

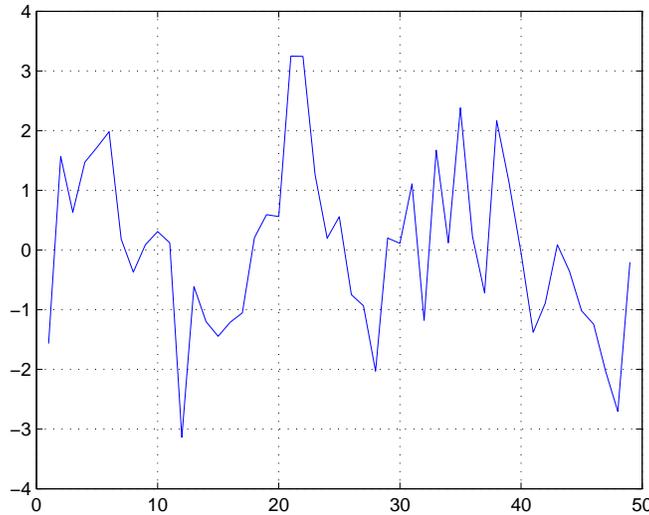
$$\delta_n(\omega) = \begin{cases} \frac{N_0}{2} & |\omega - \omega_c| < 2\pi B \\ 0, & c.c. \end{cases} \quad (7)$$

e dessa forma a variância do ruído será dada por  $\sigma_n^2 = N_0/2$ .

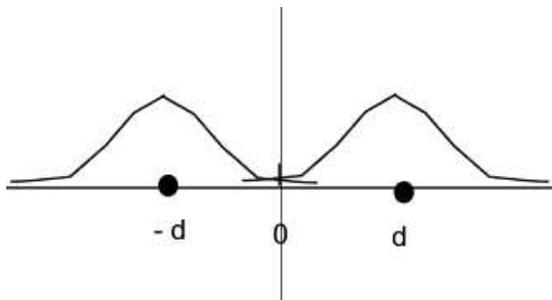
Como exemplo, considere um seno  $\sqrt{2} \sin(6\pi)$  com potência 1 mostrado na figura abaixo.



Adicionando um ruído com variância  $\sigma_n^2 = 1$  (sua potência), ou seja,  $N_0 = 2$ , obtém o sinal mostrado abaixo.



Observe que para os sinais ASK, a adição do ruído implica em deslocar o ponto da contelação  $\hat{s}_m$  por um valor aleatório gerado com distribuição Gaussiana com variância  $\sigma_n^2$ , como está representado na figura abaixo para  $M = 2$ .



Dessa forma, tendo o sinal uma potência  $P_s$  e o ruído um potência  $P_n = \sigma_n^2$  define-se a relação sinal ruído

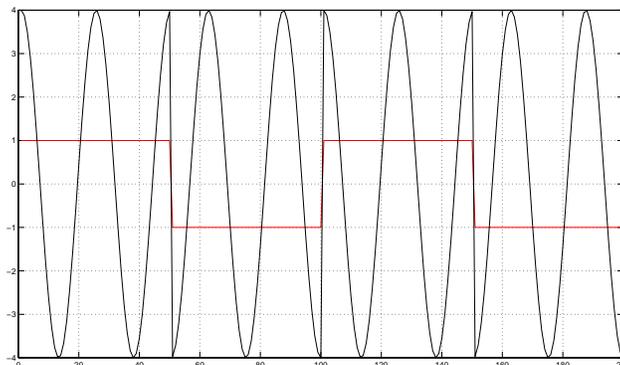
(SNR, do inglês *signal to noise ratio*) como

$$SNR = \frac{P_s}{P_n}$$

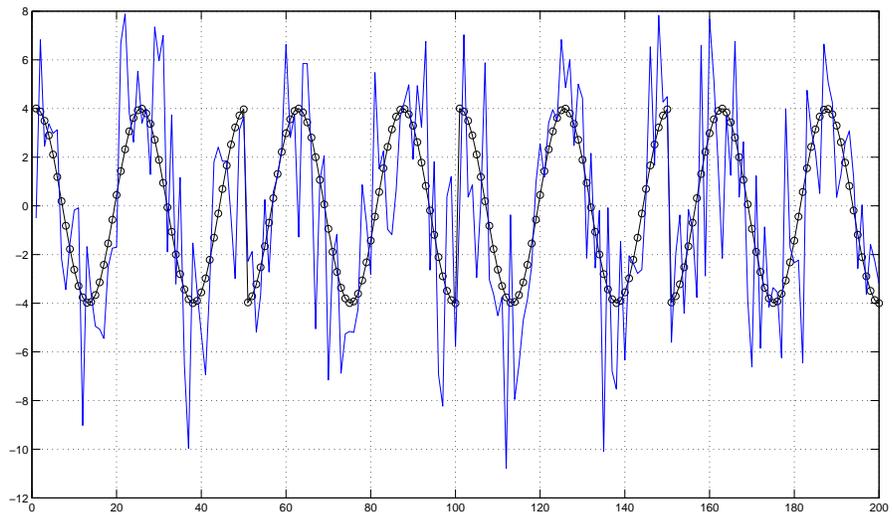
que normalmente é dada em dB,

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

Como exemplo, vamos considerar o sinal ASK, no tempo, com  $M = 2$  mostrado na Figura abaixo



considerando uma variância de  $0dB$ , o sinal recebido está representado na figura abaixo junto com o sinal enviado (marcado com círculos)



# Modulação em Fase

Como o próprio nome indica, a informação é escrita na fase da portadora, e como a fase varia entre  $0 \leq \theta \leq 2\pi$ , as fases usadas para transportar  $M$  mensagens são  $\theta_m = 2\pi m/M$  com  $m = 0, 1, \dots, M - 1$ . Como exemplo, para  $M = 2$  (binário) a portadora tem fases  $\theta = 0$  e  $\theta = \pi rad$ . Para  $k$  bits usa-se  $M = 2^k$  fases distintas.

O sinal enviado será dado por

$$s_m(t) = A p_T(t) \cos \left( \omega_c t + \frac{2\pi m}{M} \right), \quad m = 0, 1, \dots, M - 1$$

sendo  $p_T(t)$  o pulso que determina as características espectrais do sinal transmitido e  $A$  a amplitude do sinal. Este tipo de modulação é chamado de PSK - *Phase Shift Modulation*.

Quando  $p_T(t)$  é um pulso retangular

$$p_T(t) = \sqrt{\frac{2}{T}}, \quad 0 \leq t \leq T$$

e fazendo  $A = \sqrt{E_s}$ ,  $E_s$  a energia transmitida por

símbolo, tem-se

$$s_m(t) = \sqrt{\frac{2E_s}{T}} \cos\left(\omega_c t + \frac{2\pi m}{M}\right), \quad m = 0, 1, \dots, M-1$$

que é um sinal com amplitude constante e que pode mudar de fase de forma brusca no início de cada intervalo.

Podemos escrever

$$\begin{aligned} s_m(t) &= \sqrt{E_s} p_T(t) \cos\left(\frac{2\pi m}{M}\right) \cos(\omega_c t) \\ &+ \sqrt{E_s} p_T(t) \sin\left(\frac{2\pi m}{M}\right) \sin(\omega_c t), \\ m &= 0, 1, \dots, M-1 \end{aligned} \quad (8)$$

$$s_m(t) = \hat{s}_{mc} \psi_1(t) + \hat{s}_{ms} \psi_2(t) \quad (9)$$

sendo

$$\hat{s}_{mc} = \sqrt{E_s} \cos \frac{2\pi m}{M} \quad (10)$$

$$\hat{s}_{ms} = \sqrt{E_s} \sin \frac{2\pi m}{M} \quad (11)$$

e  $\psi_1(t)$  e  $\psi_2(t)$  funções ortogonais definidas por

$$\psi_1(t) = g_T(t) \cos(\omega_c t) \quad (12)$$

$$\psi_2(t) = g_T(t) \sin(\omega_c t) \quad (13)$$

Observe que podemos fazer uma representação dos pontos  $(\hat{s}_{mc}, \hat{s}_{ms})$  já que estão associados à funções ortogonais. E nesse caso temos uma constelação bi-dimensional. Como exemplo, seja  $M = 4$ , as fases são dadas por

$$s_{0c} = \sqrt{E_s} \cos \frac{2\pi 0}{4} = \sqrt{E_s}, \quad s_{0s} = \sqrt{E_s} \sin \frac{2\pi 0}{4} = 0$$

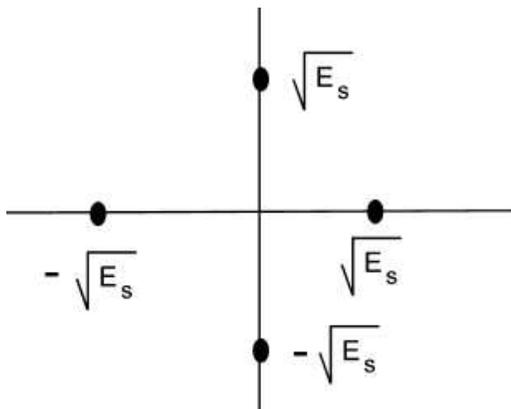
$$s_{1c} = \sqrt{E_s} \cos \frac{2\pi 1}{4} = 0, \quad s_{1s} = \sqrt{E_s} \sin \frac{2\pi 1}{4} = \sqrt{E_s}$$

$$s_{2c} = \sqrt{E_s} \cos \frac{2\pi 2}{4} = -\sqrt{E_s}, \quad s_{2s} = \sqrt{E_s} \sin \frac{2\pi 2}{4} = 0$$

$$s_{3c} = \sqrt{E_s} \cos \frac{2\pi 3}{4} = 0, \quad s_{3s} = \sqrt{E_s} \sin \frac{2\pi 3}{4} = -\sqrt{E_s}$$

e a constelação está representada na Figura abaixo.

Observe que devemos associar 2 bits a ponto da constelação.



# Modulação em Amplitude e Quadratura - QAM

Pode ser interpretada como duas modulações em amplitude independentes: portadora em fase e em quadratura. O sinal modulado em quadratura é dado por

$$s_m(t) = A_{mc}p_T(t) \cos(2\pi f_c t) - A_{ms}p_T(t) \sin(2\pi f_c t)$$

para  $0 \leq t < T$  sendo  $p_T(t)$  o pulso de transmissão e  $T$  a duração do símbolo.

$A_{mc}$  assume uma dentre  $M_c = 2^{k_c}$  amplitudes, cada uma das quais representando a informação de  $k_c$  dígitos binários. Da mesma forma,  $A_{ms}$  assume uma dentre  $M_s = 2^{k_s}$  amplitudes, cada uma das quais representando a informação de  $k_s$  dígitos binários e  $M = 2^k = M_c \times M_s$  e  $k = k_c + k_s$ .

As amplitudes  $A_{mc}$  e  $A_{ms}$  são dadas respectivamente por  $A_{mc} = (2m_c - 1 - M_c)d_c$  e

$A_{ms} = (2m_s - 1 - M_s)d_s$ , sendo que  $d_c$  e  $d_s$  controlam as potências dos sinais modulados em fase e em quadratura (em geral  $d_c = d_s$ ). Além disso,  $m_c = 1, 2, \dots, M_c$  e  $m_s = 1, 2, \dots, M_s$ .

Definindo-se:  $\psi_1(t) = \sqrt{\frac{2}{E_p}} \cos(\omega_c t)$  e

$\psi_2(t) = \sqrt{\frac{2}{E_p}} \sin(\omega_c t)$ , pode-se escrever o sinal modulado em quadratura da seguinte maneira

$$s_m(t) = \mathbf{s}_m^T \boldsymbol{\psi}(t)$$

sendo  $\mathbf{s}_m = [s_{mc} s_{ms}]^T$  e  $\boldsymbol{\psi}(t) = [\psi_1(t) \psi_2(t)]^T$ , sendo  $s_{mc} = \sqrt{\frac{E_p}{2}} A_{mc}$  e  $s_{ms} = \sqrt{\frac{E_p}{2}} A_{ms}$  a partir dos quais podemos construir a constelação dos sinais QAM.

Como exemplo, seja  $M_c = 2$  e  $M_s = 2$ , nesse caso temos  $M = 4$  ou um 4-QAM, que é a menor constelação QAM, tem-se:

$$A_{1c} = (2 \times 1 - 1 - 2)d_c = -d_c, \text{ e } A_{2c} = (2 \times 2 - 1 - 2)d_c = d_c \quad (14)$$

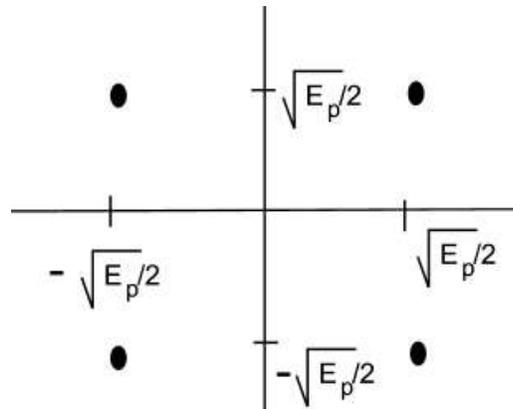
$$A_{1s} = (2 \times 1 - 1 - 2)d_s = -d_s, \text{ e } A_{2s} = (2 \times 2 - 1 - 2)d_s = d_s \quad (15)$$

e o conjunto de pontos  $\mathbf{s}_m$

$$\mathbf{s}_1 = \left(-d_c \sqrt{\frac{E_p}{2}}, -d_s \sqrt{\frac{E_p}{2}}\right), \quad \mathbf{s}_2 = \left(-d_c \sqrt{\frac{E_p}{2}}, d_s \sqrt{\frac{E_p}{2}}\right)$$

$$\mathbf{s}_3 = \left(d_c \sqrt{\frac{E_p}{2}}, -d_s \sqrt{\frac{E_p}{2}}\right), \quad \mathbf{s}_4 = \left(d_c \sqrt{\frac{E_p}{2}}, d_s \sqrt{\frac{E_p}{2}}\right)$$

representados na Figura abaixo para  $d_c = d_s$ .



A partir da constelação deve-se associar a cada ponto  $k$  bits. Caso  $M > 4$  deve escolher algum tipo de codificação para a atribuição dos bits. Como exercício, construa a constelação  $M = 16$  e atribua os respectivos bits.

# Modulação por Chaveamento de Frequência

Conhecida como FSK - *Frequency Shift Keying* nessa modulação transmite-se em uma frequência para cada símbolo. Para transmitir  $k = \log_2 M$  bits deve ser usados  $M$  sinais dados por

$$s_m(t) = \sqrt{\frac{2E_b}{T}} \cos(\omega_c t + 2\pi m \Delta f t),$$
$$m = 0, 1, \dots, M - 1, \quad 0 \leq t \leq T \quad (16)$$

sendo  $E_b$  a energia do bit,  $T$  o seu período e  $\Delta f$  a separação entre as sucessivas frequências.

As  $M$  formas de onda FSK podem ser representadas por

$$s_0 = (\sqrt{E_s}, 0, \dots, 0), \quad s_1 = (0, \sqrt{E_s}, \dots, 0), \dots,$$
$$s_{M-1} = (0, 0, \dots, \sqrt{E_s}) \quad (17)$$

o que nos permite construir uma constelação, entretanto, essa será  $M$ -dimensional.

# Introdução à Codificação

Na transmissão digital os erros implicam em troca ou perda de bits. Uma forma de evitar esses erros é codificar a informação para que seja possível detectar e corrigir erros na sequência recebida. Para isso usa-se os códigos controladores de erro que consistem, basicamente, na adição sistemática de bits redundantes que não transportam informação mais tornam possível detectar e até corrigir erros ocorridos durante a transmissão.

Os códigos controladores de erro podem ser divididos em

- Códigos de Bloco
  - Solicitação de repetição automática - ARQ (*Automatic Repeat Request*)
  - Correção de erro em avanço - FEC (*Forward Error Correction*)
- Código convolucionais

Os métodos ARQ normalmente exigem um canal de retorno entre o receptor e o transmissor, pois ao detectar erro de recepção um sinal é enviado ao transmissor para que ele repita os dados recebidos

com erro. Esse método normalmente só detecta erros. Já o método FEC utiliza a redundância inserida pelo transmissor para verificar se os dados recebidos estão corretos. Dependendo da quantidade de redundância pode-se além de detectar o erro descobrir onde ele ocorreu e assim corrigi-lo. Entretanto, vale salientar que detectar erros é mais fácil que corrigir, portanto os códigos com o primeiro objetivo somente são mais simples do que os códigos que se propõem a corrigir erros.

Neste texto nos determos aos códigos FEC, deixaremos a descrição e análise dos ARQs para outra oportunidade.

Como exemplo considere o ISBN (*International Standard Book Number*) que consiste num conjunto de 10 caracteres que identifica univocamente um livro. Nesse conjunto de 10 número o último é usado como dígito verificador, ou seja, os números  $x_1, x_2, \dots, x_{10}$  são tais que

$$\sum_{i=1}^{10} ix_i = 0(\text{mod } 11).$$

Como exemplo, o número 0-19-511009-9 (ISBN do Lathi) é válido pois

$$\begin{aligned}
& 1 \times 0 + 2 \times 1 + 3 \times 9 + 4 \times 5 + 5 \times 1 + 6 \times 1 \\
+ & 7 \times 0 + 8 \times 0 + 9 \times 9 + 10 \times 9 \\
= & 0 + 2 + 27 + 20 + 5 + 6 + 0 + 0 + 81 + 90 \\
= & 231 = 0 \pmod{11} \tag{18}
\end{aligned}$$

Outro local onde se utiliza um código para verificação de erros no Brasil é no nosso cadastro de pessoas físicas (CPF). O CPF é composto por onze algarismos, onde os dois últimos são chamados de dígitos verificadores, ou seja, os dois últimos dígitos são criados a partir dos nove primeiros. O cálculo é feito em duas etapas utilizando o módulo de divisão 11.

Para exemplificar melhor, iremos calcular os dígitos verificadores de um CPF imaginário, por exemplo, 222.333.666-XX. Para gerar o primeiro dígito verificador vamos fazer a soma

$$\sum_{i=10}^2 ix_{i-9} \pmod{11}$$

que no exemplo fica  $2 \times 10 + 2 \times 9 + 2 \times 8 + 3 \times 7 + 3 \times 6 + 3 \times 5 + 6 \times 4 + 6 \times 3 + 6 \times 2 = 162$

mod 11 = 8 e o resultado é subtraído de 11 (se for menor que 2 vira zero), logo no exemplo acima o dígito verificador é  $11 - 8 = 3$ . Para gerar o segundo dígito verificador vamos fazer a soma

$$\sum_{i=11}^2 ix_{i-10} \pmod{11}$$

e agora inclui-se o primeiro dígito verificador, que no exemplo fica  $2 \times 11 + 2 \times 10 + 2 \times 9 + 3 \times 8 + 3 \times 7 + 3 \times 6 + 6 \times 5 + 6 \times 4 + 6 \times 3 + 3 \times 2 = 201 \pmod{11} = 3$  e o resultado é subtraído de 11 (se for menor que 2 vira zero), logo no exemplo acima o dígito verificador é  $11 - 3 = 8$ . E o CPF final será 222.333.666-38

## Códigos de Bloco

De uma forma geral, como está representado na Figura 5 os sistemas de comunicações que utilizam codificação podem ser representado por uma fonte, que gera a informação e que a partir desse ponto consideraremos que a saída da fonte já foi amostrada e quantizada, portanto a informação já está em um formato digital. Os bits da fonte serão codificados e

em seguida enviados através de um canal de comunicações, que devido ao ruído pode inserir erros que consistem inicialmente em troca de bits. O receptor utiliza um decodificador que recebe os bits da saída do canal e tenta corrigir os erros inseridos pelo canal.

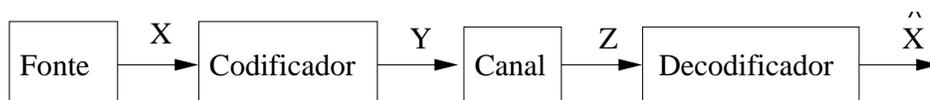


Figura 5:

Nesta seção considerarmos que a sequência de símbolos da fonte são agrupadas em blocos de tamanho  $k$  e que o codificador transforma cada um desses blocos em blocos de tamanho  $n$ , acrescentando  $n - k$  bits de redundância, como está ilustrado na Figura 6. Esse tipo de codificação é conhecido como **código de bloco**. Um parâmetro importante é a taxa do código definida como

$$R = \frac{k}{n}$$

pois nos dá uma indicação do quanto de redundância será inserido pelo código.

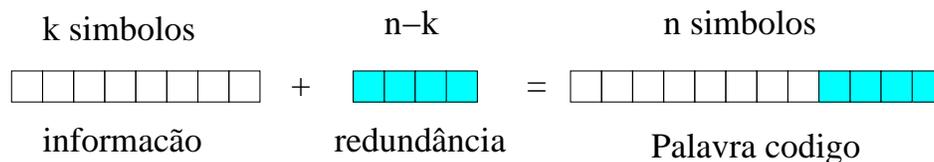


Figura 6: Adição de redundância em um código de bloco.

Observe que as palavras código pertencem a um conjunto de tamanho  $2^n$  e que desse conjunto são escolhidas  $2^k$  palavras. Essa escolha é feita de modo a minimizar a probabilidade de erro, para isso as palavras escolhidas tem o maior número de bits distintos possível. A menor quantidade em que duas palavras código quaisquer é conhecida como **distância do código**.

Duas palavras de um código com distância mínima  $d$  diferem dessa quantidade de bits, logo, se tivermos  $t < d/2 - 1$  erros podemos corrigir a palavra recebida, logo para corrigir  $t$  erros deve-se ter

$$d_{min} = 2t + 1.$$

Já para detectar  $t$  error necessitamos de

$$d_{min} = t + 1.$$

Portanto, Um código que detecte  $t$  erros é mais simples do que um que corrige  $t$  erros.

## Códigos de Bloco Lineares

Sejam  $\mathbf{C} = (c_1, c_2, \dots, c_n)$  uma palavra código e  $\mathbf{d} = (d_1, d_2, \dots, d_k)$  uma palavra de informação, pode-se escrever

$$\mathbf{c} = \mathbf{d}G$$

sendo  $G_{k \times n} = [I_{k \times k} : P_{k \times m}]$  sendo  $I_{k \times k}$  uma matriz identidade e  $P_{k \times m}$  com  $m = n - k$  a matriz de paridade.

**Exemplo 1** *Seja*

$$G = \begin{bmatrix} 1 & 0 & 0 & 1 & 0 & 1 \\ 0 & 1 & 0 & 0 & 1 & 1 \\ 0 & 0 & 1 & 1 & 1 & 1 \end{bmatrix}$$

*então tem-se*

<i>Palavra de informação</i>	<i>Palavra código</i>
000	000000
001	001110
010	010011
011	011101
100	100101
101	101011
110	110110
111	111000

Observe que em um código linear a soma de duas palavras código ainda é uma palavra código, e que a palavra código composta de zeros sempre está presente no código.

Para os códigos lineares pode-se construir a matriz

$$H = [P^T : I_m]$$

tal que  $\mathbf{c}H^T = 0$ . Essa matriz é conhecida com matriz de teste de paridade.

**Exemplo 2** *Para o código do exemplo anterior*

*tem-se*

$$H = \begin{bmatrix} 1 & 0 & 1 & 1 & 0 & 0 \\ 0 & 1 & 1 & 0 & 1 & 0 \\ 1 & 1 & 0 & 0 & 0 & 1 \end{bmatrix}$$

Observe que qualquer palavra que não seja do código, por exemplo  $\mathbf{r} = \mathbf{c} + \mathbf{e}$ , gera um vetor diferente de zero, ou seja

$$\mathbf{s} = \mathbf{r}H^T \quad (19)$$

$$= (\mathbf{c} + \mathbf{e})H^T \quad (20)$$

$$= \mathbf{c}H^T + \mathbf{e}H^T \quad (21)$$

$$= \mathbf{e}H^T \quad (22)$$

o vetor  $\mathbf{s}$  é conhecido como **síndrome**.

## Correção de erros usando síndromes

Considere todos os vetores de erro de peso mínimo e calcule  $\mathbf{s}$  para cada um deles.

### Exemplo 3

<i>Palavra de informação</i>	<i>Palavra código</i>
<i>000001</i>	<i>001</i>
<i>000010</i>	<i>010</i>
<i>...</i>	<i>...</i>
<i>100000</i>	<i>101</i>

Ao receber um vetor  $\mathbf{r}_i$  calcula-se a síndrome  $\mathbf{s}_i = \mathbf{r}_i H^T$  e verifica-se na tabela a qual linha essa síndrome corresponde e encontra-se o erro.

	<i>Erro</i>	<i>Síndrome</i>
	<i>000001</i>	<i>101</i>
	<i>000010</i>	<i>110</i>
<b>Exemplo 4</b>	<i>000100</i>	<i>011</i>
	<i>001000</i>	<i>001</i>
	<i>010000</i>	<i>010</i>
	<i>100000</i>	<i>100</i>

Na decodificação, ao receber um  $\mathbf{r}_i$  calcula-se a síndrome  $\mathbf{s}_i = \mathbf{r}_i \mathbf{H}^T$  e verifica-se na tabela a qual linha essa síndrome está associada e encontra-se qual o erro correspondente. Para um código  $(n, k)$  linear

tem-se  $d_{min} = 1 + n - k$  e um código de Hamming tem como característica  $(n, k) = (2^m - 1, 2^m - 1 - m)$  sendo  $m = n - k$ .