# MAC0344 Arquitetura de Computadores
# Lista de Exercícios No. 4

Mateus Agostinho dos Anjos
NUSP 9298191

6 de Outubro de 2019

**1 -**

Começamos o código de Hamming definindo os valores de $x_1$ até $x_{11}$.

$$
\begin{array}{lclcl}
x_1 & = & \text{a determinar} & = & ? \\
x_2 & = & \text{a determinar} & = & ? \\
x_3 & = & m_1 & = & 1 \\
x_4 & = & \text{a determinar} & = & ? \\
x_5 & = & m_2 & = & 1 \\
x_6 & = & m_3 & = & 0 \\
x_7 & = & m_4 & = & 0 \\
x_8 & = & \text{a determinar} & = & ? \\
x_9 & = & m_5 & = & 1 \\
x_{10} & = & m_6 & = & 0 \\
x_{11} & = & m_7 & = & 1
\end{array}
$$

Agora calculamos $x_1,\ x_2,\ x_3,\ x_4$ da seguinte forma:
($\oplus$ representa a operação "ou exclusivo"(XOR))

$$
\begin{array}{lcl}
x_1 & = & x_3 \oplus x_5 \oplus x_7 \oplus x_9 \oplus x_{11} \\
x_2 & = & x_3 \oplus x_6 \oplus x_7 \oplus x_{10} \oplus x_{11} \\
x_4 & = & x_5 \oplus x_6 \oplus x_7 \\
x_8 & = & x_9 \oplus x_{10} \oplus x_{11}
\end{array}
$$

Existe uma forma simples para chegar às fórmulas, basta seguir os passos:
1. escrever os números de 1 a 11 em binário.
2. $x_1$ é calculado utilizando os números que possuem o bit $2^0$ igual a 1.
3. $x_2$ é calculado utilizando os números que possuem o bit $2^1$ igual a 1.
4. $x_3$ é calculado utilizando os números que possuem o bit $2^2$ igual a 1.
5. $x_4$ é calculado utilizando os números que possuem o bit $2^3$ igual a 1.

Depois do cálculo da fórmula acima, chegamos em:

$$\begin{aligned} x_1 &= 0 \\ x_2 &= 1 \\ x_4 &= 1 \\ x_8 &= 0 \end{aligned}$$

Portanto o código de Hamming $x_1x_2x_3x_4x_5x_6x_7x_8x_9x_{10}x_{11}$ para o dado $m_1m_2m_3m_4m_5m_6m_7 = 1100101$ será:

$$\begin{aligned} x_1 &= 0 \\ x_2 &= 1 \\ x_3 &= 1 \\ x_4 &= 1 \\ x_5 &= 1 \\ x_6 &= 0 \\ x_7 &= 0 \\ x_8 &= 0 \\ x_9 &= 1 \\ x_{10} &= 0 \\ x_{11} &= 1 \end{aligned}$$

**2 -**

**Identificando se há erro:**
Para detectar erros primeiro devemos comparar cada $x_\alpha$ com seu respectivo $y_\alpha$, veja a tabela de comparação abaixo:

$$\begin{array}{ccc|ccc}
y_1 & = & 0 & 0 & = & x_1 \\
\color{red}{y_2} & \color{red}{=} & \color{red}{0} & \color{red}{1} & \color{red}{=} & \color{red}{x_2} \\
y_3 & = & 1 & 1 & = & x_3 \\
y_4 & = & 1 & 1 & = & x_4 \\
\color{red}{y_5} & \color{red}{=} & \color{red}{0} & \color{red}{1} & \color{red}{=} & \color{red}{x_5} \\
y_6 & = & 0 & 0 & = & x_6 \\
y_7 & = & 0 & 0 & = & x_7 \\
y_8 & = & 0 & 0 & = & x_8 \\
y_9 & = & 1 & 1 & = & x_9 \\
y_{10} & = & 0 & 0 & = & x_{10} \\
y_{11} & = & 1 & 1 & = & x_{11}
\end{array}$$

Em vermelho vemos as linhas em que $y_\alpha$ é diferente de $x_\alpha$.
Como existe $\alpha$ tal que $x_\alpha \neq y_\alpha$, então **há um erro**.

**Corrigindo erro:**
Para corrigir o erro devemos calcular 4 bits de paridade, denominados $k_1, k_2, k_3, k_4$.
O cálculo destes bits de paridade é semelhante ao cálculo de $x_1, x_2, x_3, x_4$ visto anteriormente:

$$\begin{aligned}
k_1 &= y_1 \oplus y_3 \oplus y_5 \oplus y_7 \oplus y_9 \oplus y_{11} \\
k_2 &= y_2 \oplus y_3 \oplus y_6 \oplus y_7 \oplus y_{10} \oplus y_{11} \\
k_3 &= y_4 \oplus y_5 \oplus y_6 \oplus y_7 \\
k_4 &= y_8 \oplus y_9 \oplus y_{10} \oplus y_{11}
\end{aligned}$$

O cálculo de $k_1, k_2, k_3, k_4$ gerará o número binário codificado pelos 4 bits, $k_4k_3k_2k_1$, que determina a posição do bit lido que está errado.
Calculando $k_1, k_2, k_3, k_4$ utilizando a fórmula acima, chegamos em:

$$
\begin{aligned}
k_1 &= 1 \\
k_2 &= 0 \\
k_3 &= 1 \\
k_4 &= 0
\end{aligned}
$$

Portanto, **o bit $y_5$ (0101) está errado**, pois tem o valor 0 e deveria ser 1.