

E-Fólio A de Sistemas em Rede

Efólio-A – 28/11/2013
David Pintassilgo, aluno nº 1100896

Questão 1: Um fluxo de bits 1001 1101 é transmitido com a utilização do método de CRC padrão descrito no capítulo 3 do livro de apoio. O polinómio gerador é x^3+1 . Assim sendo, indique:

a) Qual é a string de bits realmente transmitida.

Para começar, temos o polinómio gerador x^3+1 , $x^3+1 \rightarrow 1.x^3+0.x^2+0.x+1 \rightarrow 1001$

O código CRC que vai ser adicionado ao fluxo de bits original, é resto da divisão polinomial entre os dados (neste caso 1001 1101), e o polinómio gerador (neste caso 1001).

$$\begin{array}{r} 1001 \ 1101 \\ \underline{1001} \\ \underline{1001} \\ 0000 \ 0100 \end{array}$$

O resto é 100.

Assim, a string de bits que realmente será transmitida são os dados iniciais + o resto, ou seja:

1001 1101 100

b) Suponha que o terceiro bit a partir da esquerda seja invertido durante a transmissão. Mostre que esse erro é detectado na extremidade receptora.

Caso o terceiro bit a partir da esquerda seja invertido, quando chega ao receptor a string é:

1011 1101 100

Na receção, este fluxo de bits recebido é dividido pelo polinómio gerador (1001) e o resto vai dar diferente de zero. Assim é detectado o erro e a frame será novamente enviada.

$$\begin{array}{r} 1 \ 0 \ 1 \ 1 \ 1 \ 1 \ 0 \ 1 \ 1 \ 0 \ 0 \\ \underline{1 \ 0 \ 0 \ 1} \\ 0 \ 0 \ 1 \ 0 \ 1 \ 1 \\ \underline{1 \ 0 \ 0 \ 1} \\ 0 \ 0 \ 1 \ 0 \ 0 \ 1 \\ \underline{1 \ 0 \ 0 \ 1} \\ \underline{0 \ 0 \ 0 \ 0} \ 1 \ 0 \ 0 \end{array}$$

Para confirmação, vejamos o que acontece se não houver qualquer diferença entre o envio e a receção:

$$\begin{array}{r} 1 \ 0 \ 0 \ 1 \ 1 \ 1 \ 0 \ 1 \ 1 \ 0 \ 0 \\ \underline{1 \ 0 \ 0 \ 1} \\ 0 \ 0 \ 0 \ 0 \ 1 \ 1 \ 0 \ 1 \\ \underline{1 \ 0 \ 0 \ 1} \\ 0 \ 1 \ 0 \ 0 \ 1 \\ \underline{1 \ 0 \ 0 \ 1} \\ 0 \ 0 \ 0 \ 0 \end{array}$$

Questão 2: Determine o padrão de bits transmitido no caso da mensagem 110 100 110 011 0101, supondo que é utilizada a paridade par no código de Hamming.

A paridade adiciona bits que indicam se determinado conjunto de bits tem um número de bits com o valor 1 par ou ímpar, neste caso, par.

Para começar, temos a mensagem a transmitir:

1 1 0 1 0 0 1 1 0 0 1 1 0 1 0 1

Os Bit's de paridade estão em posições que são potencias de 2 (a vermelho), a mensagem vai ficar nas restantes posições.

Posição:	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21
String final			1		1	0	1		0	0	1	1	0	0	1		1	0	1	0	1

Vamos então calcular os bit's de paridade (par) um a um, pois cada um deles, vai ser afectado por um conjunto diferente de bit's.

Para o bit de paridade na posição 1:

Posição:	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21
String final	0		1		1	0	1		0	0	1	1	0	0	1		1	0	1	0	1

Para o bit de paridade na posição 2:

Posição:	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21
String final	0	1	1		1	0	1		0	0	1	1	0	0	1		1	0	1	0	1

Para o bit de paridade na posição 4:

Posição:	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21
String final	0	1	1	1	1	0	1		0	0	1	1	0	0	1		1	0	1	0	1

Para o bit de paridade na posição 8:

Posição:	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21
String final	0	1	1	1	1	0	1	1	0	0	1	1	0	0	1		1	0	1	0	1

Para o bit de paridade na posição 16:

Posição:	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21
String final	0	1	1	1	1	0	1	1	0	0	1	1	0	0	1	1	1	0	1	0	1

O que será transmitido é:

0 1 1 1 1 0 1 1 0 0 1 1 0 0 1 1 1 0 1 0 1