Modul 4_1

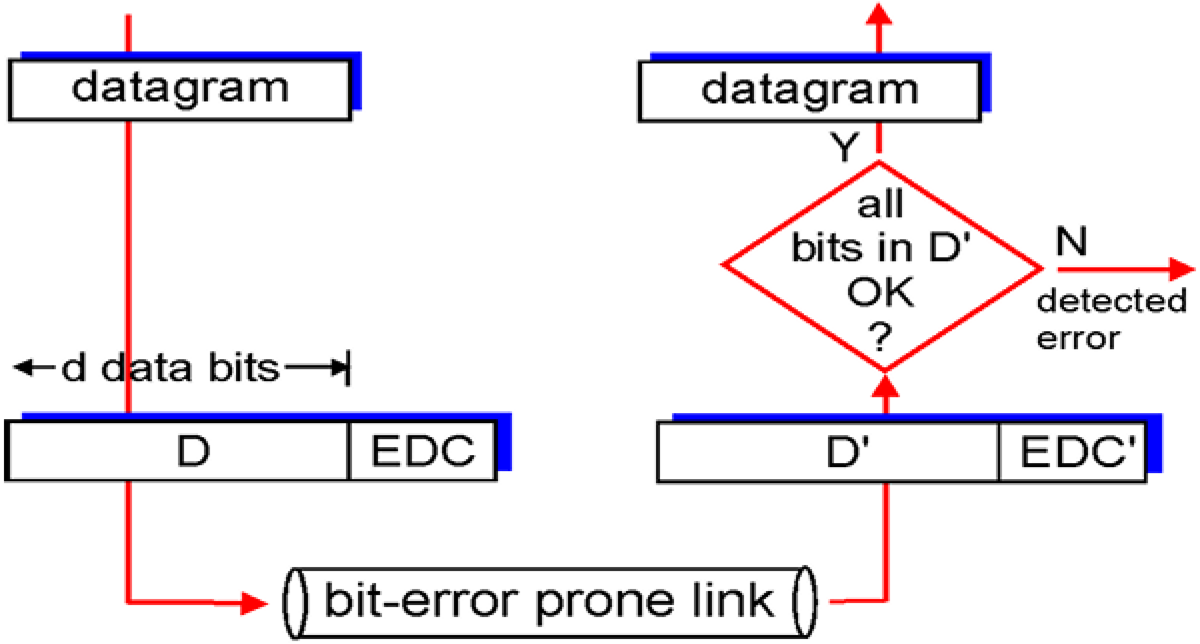
ERROR DETECTION & AVOIDANCE!

Nivelul legatura de date : detectarea erorilor

Detectarea erorilor : principiu

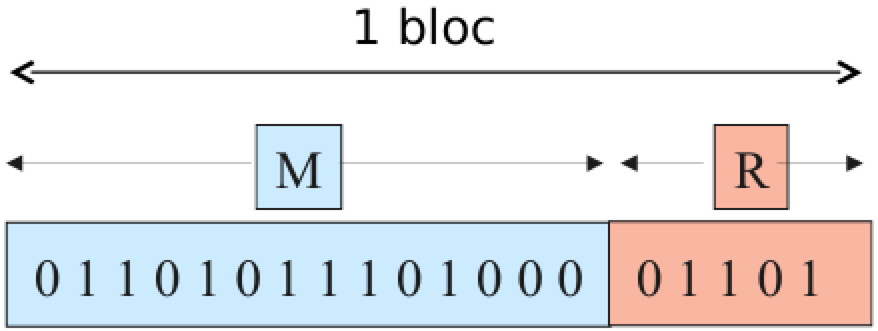
EDC= Biti de Corectare si de Detectare a erorilor
D = Date "protejate" (date utile + antet)

- Nu e fiabil 100%!
  - Protocoalele pot lasa erori nedetectate
  - Un camp EDC mai mare ➔ fiabilitate sporita



Coduri systemice

- Cod sistemic: mesajul este împărțit în blocuri, fiecare bloc este însoțit de o informație de control, calculată din câmpul de date



1 bloc

M

R

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

0 1 1 0 1

M = m biti ale mesajului

R = r biti redundanti

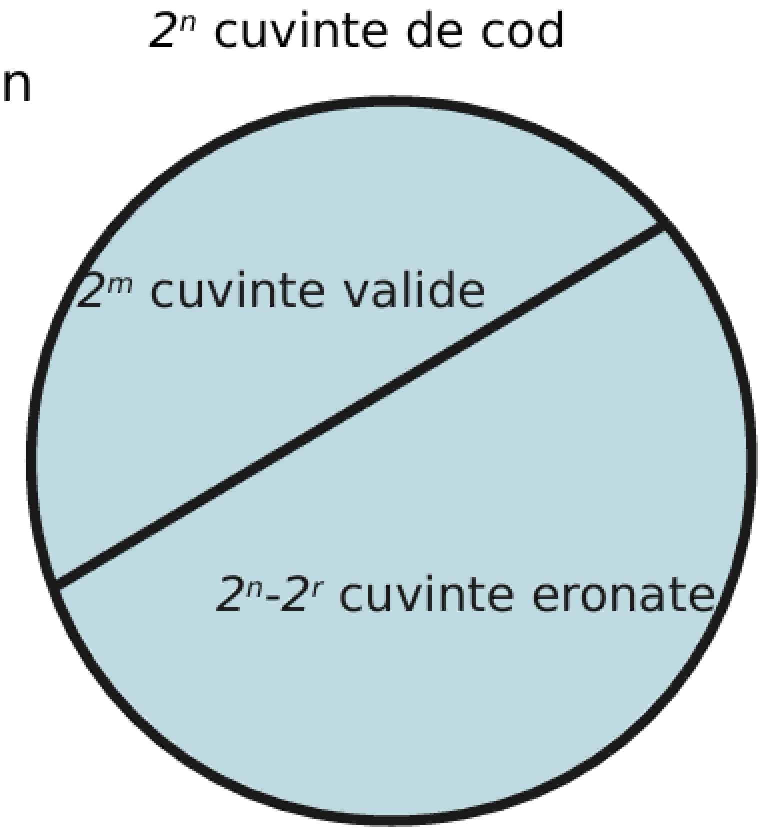
Coduri in bloc

informatie utila: m biti ---> 2^m cuvinte
informatie de control : r biti definiti din m biti utili → 2^m cuvinte valide de n biti
lungime bloc : $n= m+r$ ---> 2^n cuvinte in total (printre care 2^n-2^r eromate)
Randamentul unui cod (m,n) est:

$$R=m/n$$

distanta Hamming dintre doua cuvinte : $d(m1, m2)$ = numarul de biti diferiti de acelasi rang
Exemplu: $m1$ = 10110010 $m2$ = 10000110

$$d(m1,m2) = 3$$

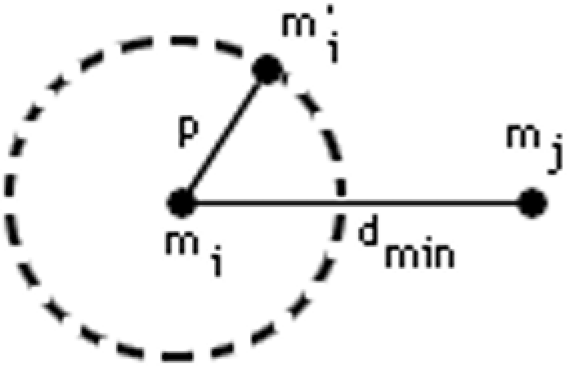


Detectarea erorilor

2 cuvinte de cod vor fi confundate cu atat mai putin cu cat distanta lor Hamming va fi mai mare : definirea unei distance minimum d_{min}

Daca $d(m1, m2) < d_{min}$, atunci *m2* este o copie eronata a *m1*

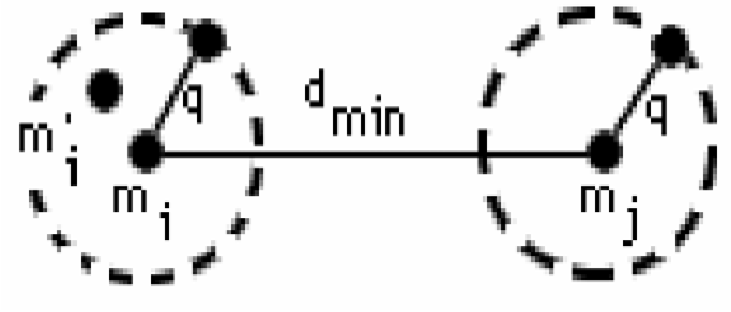
Regula 1 :
Pentru a detecta *p* erori,
tebuie ca $d_{min} > p$



Exemplu : detectarea erorilor simple : $d_{min}>2$

Corectarea erorilor

Corectarea erorilor de ordinul *q : fiecarui cuvant* de cod si copiile sale "admisibile" trebuie sa fie in sfere care nu se intersecteaza



Regula 2 : Pentru a corecta erori de ordinul q, $d_{min} > 2q+1$

Exemplu : corectarea erorilpr simple necesita $d_{min} > 3$

Exercitiu

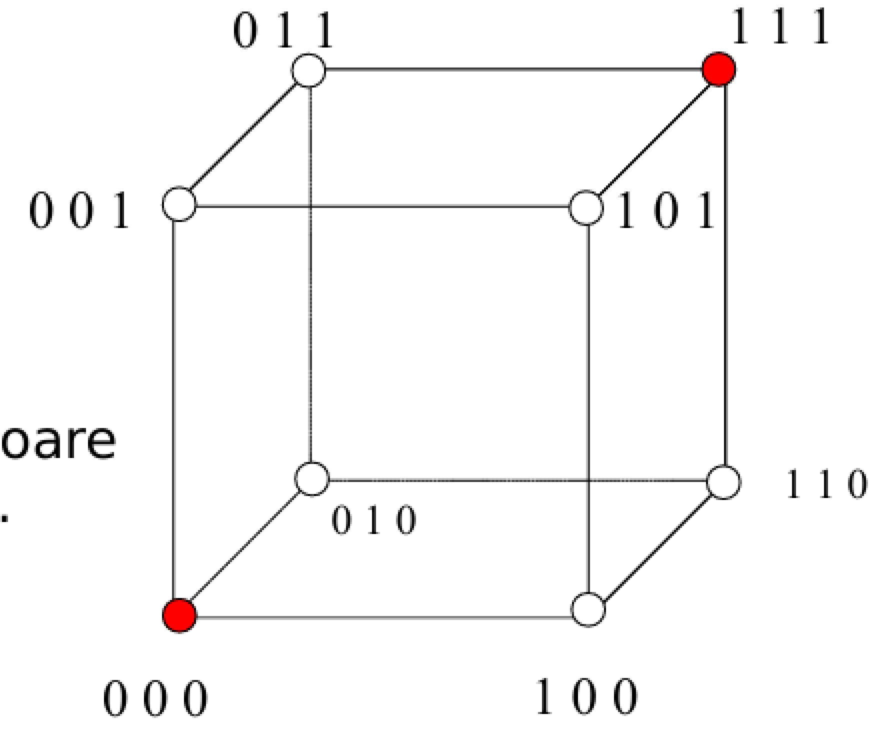
Fie C = { 0000 0000, 0000 1111, 1111 0000, 1111 1111 }

- Care e distanta Hamming ?
- Daca un receptor primeste 0000 0111, ce conluzii ar trebui sa facem?
- Este oare posibil sa recunoastem cuvantul original ?

Exemplu: coduri cu repetitie

M	R
0	0 0
1	1 1

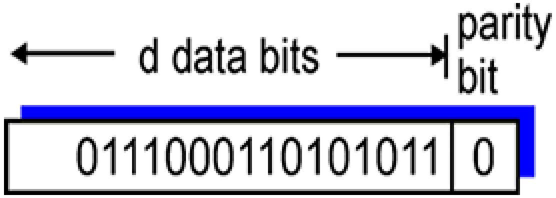
Poate corecta 1 erroare
si detecta 2 erori ...



Paritate

Bit de paritate:

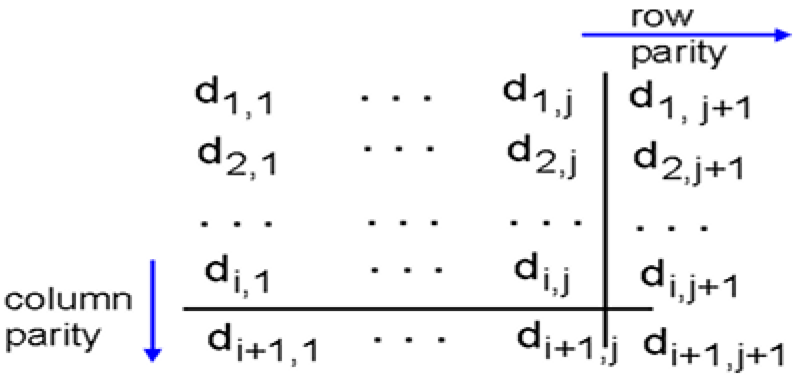
Detecteaza o singura eroare



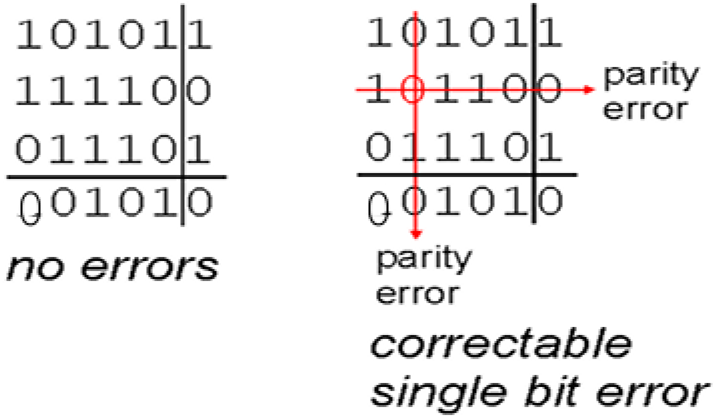
Even = par
Odd = impar

Paritate bi-dimensionala :

Detecteza si corecteaza o singura erroare



$$\begin{array}{cccc|c} d_{1,1} & \cdots & d_{1,j} & & d_{1,j+1} \\ d_{2,1} & \cdots & d_{2,j} & & d_{2,j+1} \\ \cdots & \cdots & \cdots & & \cdots \\ d_{i,1} & \cdots & d_{i,j} & & d_{i,j+1} \\ \hline d_{i+1,1} & \cdots & d_{i+1,j} & & d_{i+1,j+1} \end{array}$$



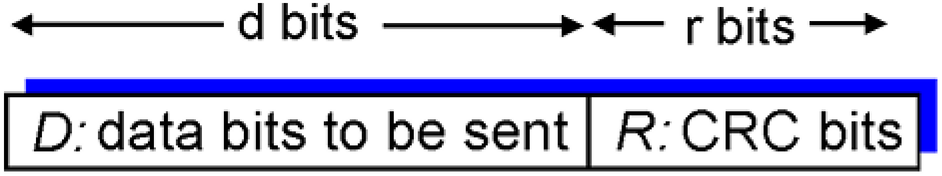
$$\begin{array}{c|c} 10101 & 1 \\ 11110 & 0 \\ 01110 & 1 \\ \hline 00101 & 0 \end{array} \qquad \begin{array}{c|c} 10101 & 1 \\ 10110 & 0 \\ 01110 & 1 \\ \hline 00101 & 0 \end{array}$$

no errors

correctable single bit error

Detectarea erorilor: Cyclic Redundancy Check

- Fie ansamblul de biti de date D ca un numar binar
- Selectam un polinom generator *G*
- scopul: sa adugam r biti CRC, R, pentru ca
  - ansamblul <D,R> sa fie divizibil cu *G* (modulo 2) fara de rest
  - Receptorul cunoaste *G*, divizeaza <D,R> la *G*. Daca restul = 0: nu exista erori, in caz contrar sunt detectate erori
  - Detecteaza suite de errori pana la r+1 biti
- Foarte utilizat in practica



bit pattern

$$D * 2^{r} \quad XOR \quad R$$

mathematical formula

Coduri polinomiale

Code polynomial (n,m) : n bits in total, m bits utili,
$r = n\text{-}m$ bits de control

$G(x) = x^r + x^{r\text{-}i} + \ldots. + 1$ polinom generatorr
$M(x) = u_1 x^{m\text{-}1} + u_2 x^{m\text{-}2} + \ldots + u_{m\text{-}1} x + u_m$ informatie utila
$x^r M(x)$ polynom de ordinul $m+r\text{-}1 = n\text{-}1$
Divizarea $x^r M(x)$ la $G(x)$:

$$x^r M(x) = Q(x)*G(x) + R(x) \qquad (R(x) \text{ e de ordinul } r\text{-}1)$$

$Y(x) = Q(x)*G(x) = x^r M(x)+R(x)$ mesaj transmis

$Y(x)$ contine n termini si este de ordinul $n\text{-}1$.

Caz particular : cod ciclic, code polynomial unde G(x) se imparte la $x^n + 1$

$$x^n + 1 = G(x)Q(x)$$

CRC : “Cyclic Redundancy Code”

Exemplu cod CRC

Mesaj $M = 1\ 1\ 0\ 0\ 1\ 1$

$= 1\,x^5 + 1\,x^4 + 0\,x^3 + 0\,x^2 + 1\,x^1 + 1\,x^0$

$= x^5 + x^4 + x + 1$

Polynôm $G = 1\ 0\ 0\ 1 = x^3 + 1 \rightarrow$ 4 biti

Numarul de biti de control: $4 - 1$ =3

$$x^r M = x^8 + x^7 + x^4 + x^3$$

trebuie de divizat x^rM la G

Calcul CRC

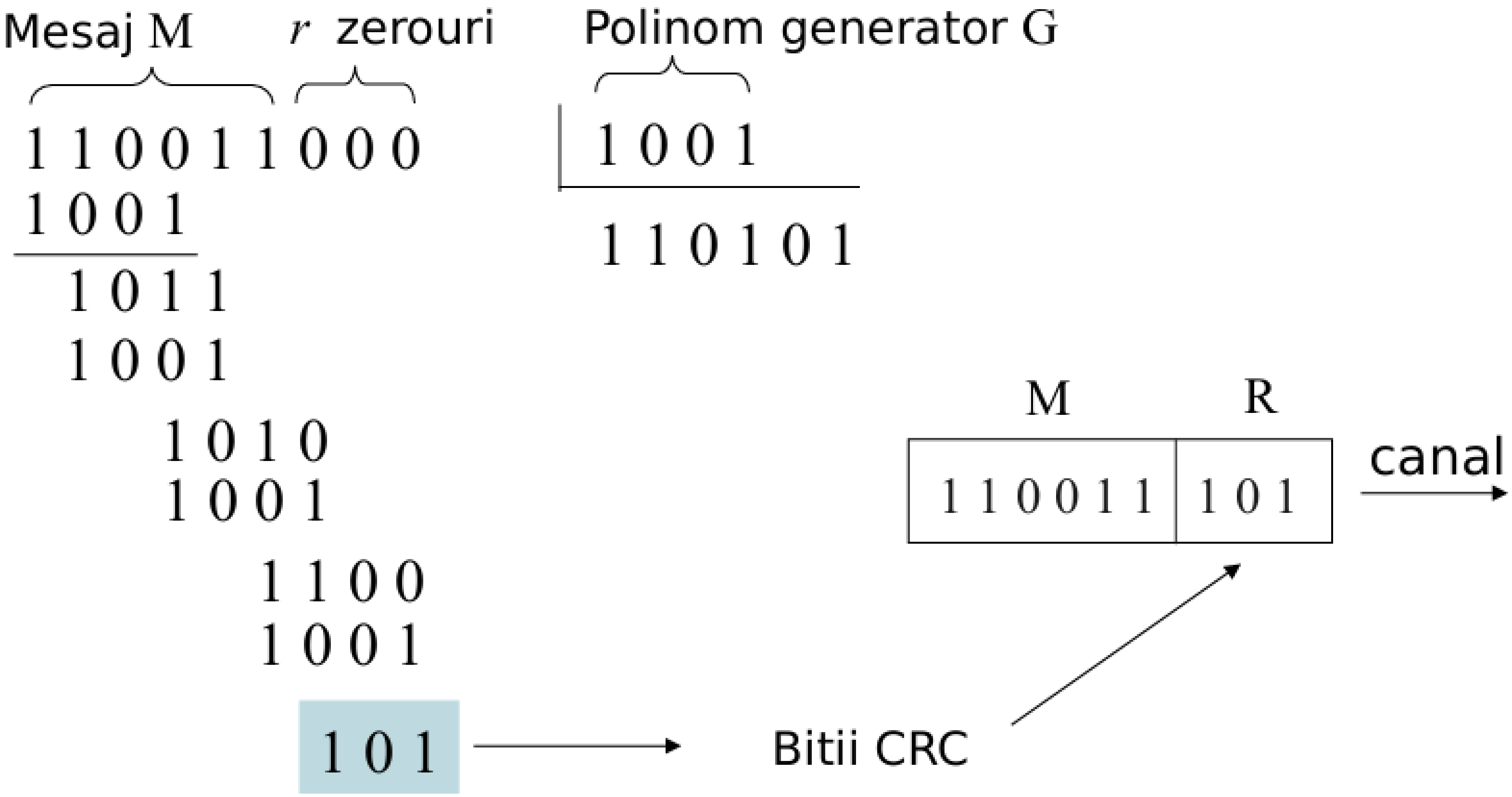
$$\begin{array}{ccccccc}
x^8 & +\,x^7\,+ & & x^4 & +\,x^3 & & \\
x^8 & & +\,x^5 & & & & \\
\cline{1-3}
& x^7 & +\,x^5 & & & & \\
& x^7 & & +\,x^4 & & & \\
\cline{2-4}
& & x^5 & & & & \\
& & x^5\,+ & & & x^2 & \\
\cline{3-6}
& & & & x^3 & +\,x^2 & \\
& & & & x^3\,+ & & 1 \\
\cline{5-7}
& & & & & x^2 & +\,1
\end{array}$$

$$\begin{array}{|l}
x^3 + 1 \\
\hline
x^5 + x^4 + x^2 + 1
\end{array}$$

$$\longrightarrow \text{Rest} = \text{R} = 1\ 0\ 1$$

bitii
de paritate (CRC)

Calcul CRC



Mesaj M

r zerouri

Polinom generator G

1 1 0 0 1 1 0 0 0

1 0 0 1

1 0 1 1

1 0 0 1

1 0 1 0

1 0 0 1

1 1 0 0

1 0 0 1

1 0 1

1 0 0 1

1 1 0 1 0 1

Bitii CRC

M

R

1 1 0 0 1 1

1 0 1

canal

Calcul CRC in serie

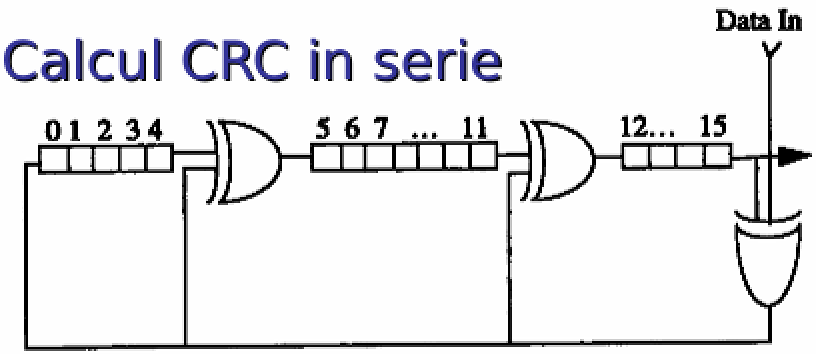


Schéma du circuit de calcul du CRC avec
$G(x) = x^{16} + x^{12} + x^5 + 1$

n°bit	DI	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
2	1	1	0	0	0	0	1	0	0	0	0	0	0	1	0	0	0
3	0	0	1	0	0	0	0	1	0	0	0	0	0	0	1	0	0
4	0	0	0	1	0	0	0	0	1	0	0	0	0	0	0	1	0
5	1	1	0	0	1	0	1	0	0	1	0	0	0	1	0	0	1
6	1	0	1	0	0	1	0	1	0	0	1	0	0	0	1	0	0
7	0	0	0	1	0	0	1	0	1	0	0	1	0	0	0	1	0
8	1	1	0	0	1	0	1	1	0	1	0	0	1	1	0	0	1
9	0	1	1	0	0	1	1	1	1	0	1	0	0	0	1	0	0
10	1	1	1	1	0	0	0	1	1	1	0	1	0	1	0	1	0
11	1	1	1	1	1	0	1	0	1	1	1	0	1	1	1	0	1
12	0	1	1	1	1	1	1	1	0	1	1	1	0	0	1	1	0
13	1	1	1	1	1	1	0	1	1	0	1	1	1	1	0	1	1
14	1	0	1	1	1	1	1	0	1	1	0	1	1	1	1	0	1
15	1	0	0	1	1	1	1	1	0	1	1	0	1	1	1	1	0
16	1	1	0	0	1	1	0	1	1	0	1	1	0	0	1	1	1

Tableau des valeurs du reste (calculé à l'émission)
avec $x^{16} M(x) = x^{30} + x^{27} + x^{26} + x^{24} + x^{22} + x^{21} + x^{19} + x^{18} + x^{17} + x^{16}$

CRC calculat de catre emitator

Après le seizième décalage, le registre contient les bits de contrôle, c'est-à-dire la valeur du CRC calculé par l'émetteur, que celui-ci doit émettre après les bits de données. La suite de bits transmise au récepteur (dans lequel les bits du CRC sont écrits en gras) est donc :

0 1 0 0 1 1 0 1 0 1 1 0 1 1 1 1 **1 1 1 0 0 1 1 0 1 1 0 1 1 0 0 1**

CRC

CRC calculat de catre receptor

n°bit	DI	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
16	1	1	0	0	1	1	0	1	1	0	1	1	0	0	1	1	1
17	1	0	1	0	0	1	1	0	1	1	0	1	1	0	0	1	1
18	1	0	0	1	0	0	1	1	0	1	1	0	1	1	0	0	1
19	1	0	0	0	1	0	0	1	1	0	1	1	0	1	1	0	0
20	0	0	0	0	0	1	0	0	1	1	0	1	1	0	1	1	0
21	0	0	0	0	0	0	1	0	0	1	1	0	1	1	0	1	1
22	1	0	0	0	0	0	0	1	0	0	1	1	0	1	1	0	1
23	0	0	0	0	0	0	0	0	0	1	0	0	1	1	0	1	1
25	1	0	0	0	0	0	0	0	0	0	1	0	0	1	1	0	1
26	1	0	0	0	0	0	0	0	0	0	0	1	0	0	1	1	0
27	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	1	1
28	1	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	1
29	1	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0
30	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0
31	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1
32	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Tableau des valeurs du reste (calculé à la reception) avec
$x^{16} * M(x) = x^{30} + x^{27} + x^{26} + x^{24} + x^{22} + x^{21} + x^{19} + x^{18} + x^{17} + x^{16}$

Proprietati CRC pe 16 bits

Tipul de eroare	Rata de detectare
pe 1 bit	100 %
pe 2 biti	100 %
Numar par de biti	100 %
Serie de pana la 17 biti	100 %
Serie de 17 biti	99,997 %
Serie de 18 biti si mai mult	99,998 %

Exemple coduri polinomiale

Cod CCITT V41, polinom generator

$$G(x) = x^{16} + x^{12} + x^{5} + 1$$

utilizat in HDLC

Cod CRC 16, polinom generatorr

$$G(x) = x^{16} + x^{15} + x^{2} + 1$$

utilizare in procedura BSC, cu un codaj pe EBCDIC

Code CRC 12, polinom generator

$$G(x) = x^{12} + x^{11} + x^{3} + x^{2} + x + 1$$

utilizare in procedura BSC, cu un codaj pe 6 biti

Cod ARPA, polinom generator

$$G(x) = x^{24} + x^{23} + x^{17} + x^{16} + x^{15} + x^{13} + x^{11} + x^{10} + x^{9} + x^{8} + x^{5} + x^{3} + 1$$

Cod Ethernet, polinom generator

$$G(x) = x^{32} + x^{26} + x^{23} + x^{22} + x^{16} + x^{12} + x^{11} + x^{10} + x^{8} + x^{7} + x^{5} + x^{4} + x^{2} + x + 1$$

Coduri CRC – exemplu

ÉMETTEUR		RÉCEPTEUR	
10111010010000	10011	10111010010110	10011
10011	1010010010	10011	1010010010
- 01000		- 01000	
00000		00000	
- 10001		- 10001	
10011		10011	
-00100		-00100	
00000		00000	
-01000		-01000	
00000		00000	
- 10001		- 10001	
10011		10011	
-00100		- 00100	
00000		00000	
-01000		-01001	
00000		00000	
-10000		- 10011	
10011		10011	
-00110		-00000	
00000		00000	
-0110		-0000	→ pas d'erreur !

$M(x) = 1011101001$

$C(x) = x^4 + x + 1$

$2^4\,M(x) = 10111010010000$

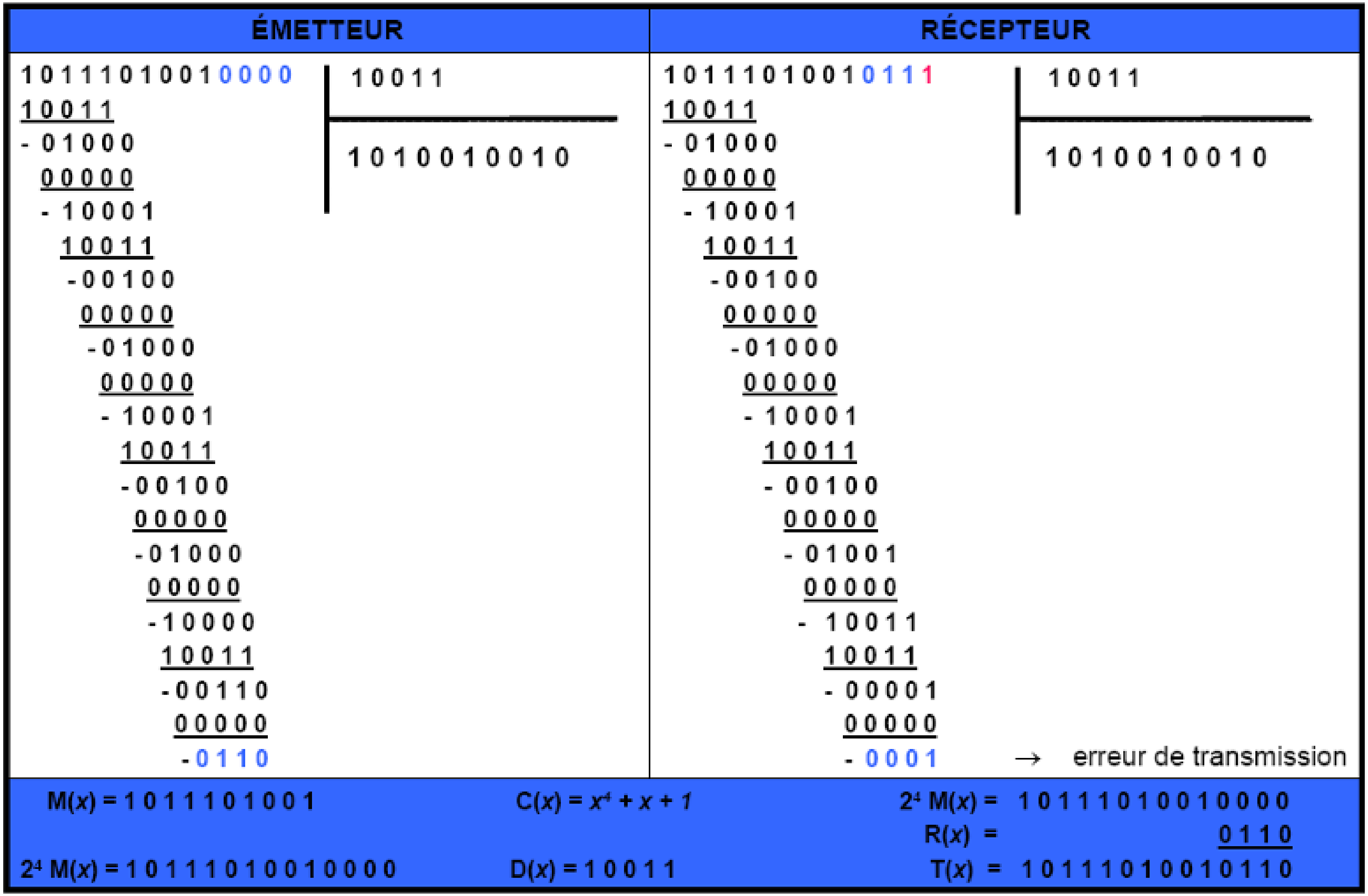
$R(x) = 0110$

$T(x) = 10111010010110$

$2^4\,M(x) = 10111010010000$

$D(x) = 10011$

Coduri CRC - exemplu



ÉMETTEUR

```
1 0 1 1 1 0 1 0 0 1 0 0 0 0  |  1 0 0 1 1
1 0 0 1 1                    |-----------------------
- 0 1 0 0 0                  |  1 0 1 0 0 1 0 0 1 0
  0 0 0 0 0
  - 1 0 0 0 1
    1 0 0 1 1
    - 0 0 1 0 0
      0 0 0 0 0
      - 0 1 0 0 0
        0 0 0 0 0
        - 1 0 0 0 1
          1 0 0 1 1
          - 0 0 1 0 0
            0 0 0 0 0
            - 0 1 0 0 0
              0 0 0 0 0
              - 1 0 0 0 0
                1 0 0 1 1
                - 0 0 1 1 0
                  0 0 0 0 0
                  - 0 1 1 0
```

RÉCEPTEUR

```
1 0 1 1 1 0 1 0 0 1 0 1 1 1  |  1 0 0 1 1
1 0 0 1 1                    |-----------------------
- 0 1 0 0 0                  |  1 0 1 0 0 1 0 0 1 0
  0 0 0 0 0
  - 1 0 0 0 1
    1 0 0 1 1
    - 0 0 1 0 0
      0 0 0 0 0
      - 0 1 0 0 0
        0 0 0 0 0
        - 1 0 0 0 1
          1 0 0 1 1
          - 0 0 1 0 0
            0 0 0 0 0
            - 0 1 0 0 1
              0 0 0 0 0
              - 1 0 0 1 1
                1 0 0 1 1
                - 0 0 0 0 1
                  0 0 0 0 0
                  - 0 0 0 1        →  erreur de transmission
```

M(*x*) = 1 0 1 1 1 0 1 0 0 1

2⁴ M(*x*) = 1 0 1 1 1 0 1 0 0 1 0 0 0 0

C(*x*) = $x^4 + x + 1$

D(*x*) = 1 0 0 1 1

```
2⁴ M(x) =  1 0 1 1 1 0 1 0 0 1 0 0 0 0
   R(x) =                        0 1 1 0
   T(x) =  1 0 1 1 1 0 1 0 0 1 0 1 1 0
```

Exercitii

Vom utiliza polynomul generator $x^4 + x^2 + x$.

1. Transmitem mesajul M1 = 1111011101, care este CRC pentru el ?
2. Acelasi lucru pentru M2 = 1100010101
3. Receptionez mesajele 1111000101010, 1100010101010110, sunt oare ele corecte ?