CAPITOLUL 7

CODURI CICLICE

7.1. DESCRIEREA CODURILOR CICLICE

Ne-am convins, credem, de avantajele ce se obțin dând unui cod bloc o structură particulară. Astfel, limitându-ne interesul la codurile bloc *liniare*, am putut beneficia de suportul matematic al algebrei liniare, cuvintele de cod fiind văzute drept vectori, iar codul fiind generat de o matrice $k \times n$. Pentru un cod liniar *sistematic*, matricea generatoare este mai simplă, fiind partiționată în două submatrice: matricea identitate $k \times k$ şi o matrice $(n-k) \times n$. În acest capitol, introducem *codurile ciclice*, care sunt o subclasă importantă a codurilor bloc liniare.

Fie un *n*–tuplu $\mathbf{v} = (v_0, v_1, \cdots, v_{n-1})$. În fig. 7.1, se arată un registru de deplasare *recirculant* cu *n* etaje în care sunt înscrise componentele lui **v**. *Recirculant* înseamnă că ieşirea etajului *n* este legată la intrarea primului etaj, astfel încât conținutul registrului de deplasare poate fi deplasat, la dreapta sau la stânga, cu ajutorul semnalului de tact:

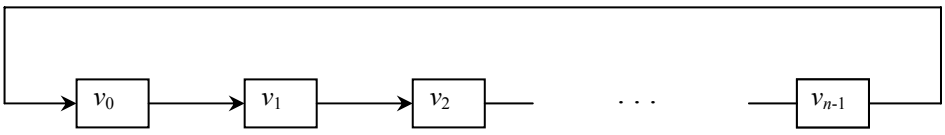


Fig. 7.1. Registru de deplasare recirculant în care sunt înscrise componentele unui *n*–tuplu $\mathbf{v} = (v_0, v_1, \cdots, v_{n-1})$.

Dacă se deplasează ciclic la dreapta cu un loc componentele *n*-tuplului **v**, se obține un alt *n*–tuplu, $\mathbf{v}^{(1)} = (v_{n-1}, v_0, v_1, \cdots, v_{n-2})$, care se numeşte o *deplasare ciclică* a lui **v**. Dacă se deplasează ciclic cu *i* locuri la dreapta componentele lui **v**, se obține *n*-tuplul

$$\mathbf{v}^{(i)} = (v_{n-i}, v_{n-i+1}, \cdots, v_{n-1}, v_0, v_1, \cdots, v_{n-i-1}).$$

Este clar că deplasarea ciclică a lui **v** cu *i* locuri la dreapta are acelaşi efect cu deplasarea ciclică a lui **v** cu *n–i* locuri la stânga.

DEFINIȚIA 7.1: Un cod bloc liniar (*n,k*) notat cu *C* se numeşte *cod ciclic* dacă fiecare deplasare ciclică a oricărui cuvânt de cod din *C* este tot un cuvânt de cod din *C*.

După cum ne vom convinge în curând, este convenabil să considerăm componentele unui cuvânt de cod din *C* $\mathbf{v} = (v_0, v_1, \cdots, v_{n-1})$ drept coeficienții unui polinom:

$$v(x) = v_0 + v_1 X + v_2 X^2 + \cdots + v_{n-1} X^{n-1} \tag{7.1}$$

Fiecare cuvânt de cod corespunde, deci, unui polinom de grad (*n*–1) sau mai mic: dacă $v_{n-1} \neq 0$, gradul lui *v*(*x*) este (*n*–1), iar dacă $v_{n-1} = 0$, gradul lui *v*(*x*) este mai mic decât (*n*–1). Acest *v*(*x*) se numeşte polinomul de cod al lui **v**. Polinomul de cod corespunzător cuvântului de cod $v^{(i)}$ este:

$$\begin{aligned} v^{(i)}(X) = {} & v_{n-i} + v_{n-i+1} X + \cdots + v_{n-1} X^{i-1} + \\ & + v_0 X^i + v_1 X^{i+1} + \cdots + v_{n-i-1} X^{n-1} \end{aligned} \tag{7.2}$$

Înmulțind *v*(X) cu X^i, obținem:

$$X^i v(X) = v_0 X^i + v_1 X^{i+1} + \cdots + v_{n-i-1} X^{n-1} + \cdots + v_{n-1} X^{n+i-1} \tag{7.3}$$

Pentru un cod binar, ale cărui cuvinte sunt vectori binari, având, deci, componente ce iau valori în corpul Galois *CG*(2), $v_i + v_i = 0$ pentru orice *i*. Având în vedere aceasta, putem aduna la (7.3) un termen egal cu zero

$$(v_{n-i} + v_{n-i+1} X + \cdots + v_{n-1} X^{i-1}) + (v_{n-i} + v_{n-i+1} X + \cdots + v_{n-1} X^{i-1})$$

Obținem astfel:

$$\begin{aligned} & X^i v(X) = v_{n-i} + v_{n-i+1} X + \cdots + v_{n-1} X^{i-1} + v_0 X^i + v_0 X^i + v_1 X^{i+1} + \cdots + \\ & + v_{n-i-1} X^{n-1} + v_{n-i}(1 + X^n) + v_{n-i+1} X(1 + X^n) + \cdots + v_{n-1} X^{i-1}(1 + X^n) = \\ & = q_i(X)(1 + X^n) + v^{(i)}(X) \end{aligned} \tag{7.4}$$

În (7.4), am notat

$$q_i(X) = v_{n-i} + v_{n-i+1} X + \cdots + v_{n-1} X^{i-1} \tag{7.5}$$

Din (7.4), se vede că polinomul de cod $v(X)$ este restul ce rezultă prin împărțirea polinomului $X^i v(X)$ la (X^n+1).

Să studiem acum consecințele structurii ciclice a codului bloc.

TEOREMA 7.1: Polinomul de cod diferit de zero de grad minim dintr-un ciclic C este unic.

DEMONSTRAȚIE

Fie $g(x) = g_0 + g_1 X + \cdots + g_{r-1} X^{r-1} + X^r$ un polinom de cod diferit de zero de grad minim din C. În ipoteza că $g(X)$ nu este unic, mai există un polinom de cod de grad r, să spunem $g'(X) = g'_0 + g'_1 X + \cdots + g'_{r-1} X^{r-1} + X^r$. Întrucât C este un cod bloc liniar, suma $g(X) + g'(X) = (g_0 \oplus g'_0) X + (g_1 \oplus g'_1) \cdots + (g_{r-1} \oplus g'_{r-1}) X^{r-1}$ este tot un polinom de cod care are însă gradul mai mic decât r. Dacă $g(X) + g'(X) \neq 0$, el este un polinom de cod diferit de zero cu grad mai mic decât gradul minim r. Dar acest lucru este imposibil. Prin urmare, $g(X) + g'(X) = 0$, de unde rezultă că $g'(X) = g(X)$. Am demonstrat astfel că $g(X)$ este unic.

TEOREMA 7.2: Fie $g(X) = g_0 + g_1 X + \cdots + g_{r-1} X^{r-1} + X^r$ polinomul de cod diferit de zero de grad minim dintr-un cod ciclic C. Termenul constant g_0 trebuie să fie egal cu 1.

DEMONSTRAȚIE

În ipoteza că $g_0 = 0$, avem:

$$\begin{aligned} g(X) &= g_1 X + g_2 X^2 + \cdots + g_{r-1} X^{r-1} + X^r = \\ &= X\left(g_1 + g_2 X + \cdots + g_{r-1} X^{r-2} + X^{r-1}\right) \end{aligned} \tag{7.6}$$

Dacă deplasăm ciclic $g(X)$ cu un loc la stânga sau cu $(n-1)$ locuri la dreapta, obținem un polinom de cod diferit de zero, $g_1 + g_2 X + \cdots + g_{r-1} X^{r-2} + X^{r-1}$, al cărui grad este mai mic decât r. Aceasta, însă, contrazice afirmația din teoremă că $g(X)$ este polinomul de cod diferit de zero de grad minim, încât trebuie să conchidem că $g_0 \neq 0$.

Din cele două teoreme, rezultă că polinomul de cod diferit de zero de grad minim dintr-un cod ciclic (n,k) C este de forma următoare:

$$g(X) = 1 + g_1 X + g_2 X^2 + \cdots + g_{r-1} X^{r-1} + X^r \tag{7.7}$$

Să considerăm polinoamele $Xg(X), X^2 g(X), \cdots, X^{n-r-1} g(X)$ care au grade $r+1, r+2, \cdots, n-1$, respectiv. Toate aceste polinoame având grad mai mic decât *n*, în (7.4), câturile $q_i(X)$ trebuie să fie identic zero astfel încât putem scrie:

$$\begin{aligned} Xg(X) &= g^{(1)}(X) \\ X^2 g(X) &= g^{(2)}(X) \\ &\vdots \\ X^{n-r-1} g(X) &= g^{(n-r-1)}(X) \end{aligned} \tag{7.8}$$

Acestea sunt deplasări ciclice ale polinomului de cod $g(X)$ astfel încât ele sunt polinoame de cod din *C*. Dar fiindcă *C* este liniar, o combinaţie liniară a polinoamelor $g(X), Xg(X), \cdots, X^{n-r-1} g(X)$ este tot un polinom de cod din *C* pentru setul de coeficienţi $\{u_i\}$, unde $0 \le i \le n-r-1$, iar $u_i = 0$ sau 1:

$$\begin{aligned} v(X) &= u_0 g(X) + u_1 Xg(X) + \cdots + u_{n-r-1} X^{n-r-1} g(X) = \\ &= \left(u_0 + u_1 X + \cdots + u_{n-r-1} X^{n-r-1}\right) g(X) \end{aligned} \tag{7.9}$$

TEOREMA 7.3: Fie $g(X) = 1 + g_1 X + \cdots + g_{r-1} X^{r-1} + X^r$ polinomul de cod diferit de zero de grad minim dintr-un cod ciclic (n,k) *C*. Un polinom binar de grad $(n-1)$ sau mai mic este polinom de cod dacă şi numai dacă el este un multiplu de $g(X)$.

DEMONSTRAŢIE

Fie $v(X)$ un polinom binar de grad $(n-1)$ sau mai mic. Să presupunem că $v(X)$ este un multiplu de $g(X)$. În acest caz:

$$\begin{aligned} v(X) &= \left(a_0 + a_1 X + \cdots + a_{n-r-1} X^{n-r-1}\right) g(X) = \\ &= a_0 g(X) + a_1 Xg(X) + \cdots + a_{n-r-1} X^{n-r-1} g(X) \end{aligned} \tag{7.10}$$

Întrucât $v(X)$ este o combinaţie liniară de polinoamele de cod $g(X), Xg(X), \cdots, X^{n-r-1} g(X)$, este şi el un polinom de cod din *C*. Am demonstrat astfel prima parte a teoremei, care spune că, dacă un polinom de grad $(n-1)$ sau mai mic este multiplu de $g(X)$, el este polinom de cod.

Fie un polinom de cod $v(X)$ din *C*. Împărţind $v(X)$ la $g(X)$, obţinem:

$$v(X) = a(X) g(X) + b(X) \tag{7.11}$$

unde fie restul $b(X)$ este identic zero, fie gradul lui $b(X)$ este mai mic decât gradul lui $g(X)$. Ecuația (7.11) se rescrie:

$$b(X) = v(X) + a(X)g(X) \tag{7.12}$$

Din prima parte a teoremei, urmează că $a(X)\ g(X)$ este un polinom de cod. Dar dacă $v(X)$ şi $a(X)\ g(X)$ sunt polinoame de cod, suma lor $b(X)$ trebuie să fie, de asemene, un polinom de cod. Dacă $b(X) \neq 0$, el trebuie să fie un polinom de grad inferior gradului lui $g(X)$. Aceasta, însă, contrazice ipoteza că $g(X)$ este polinomul de cod diferit de zero de grad minim. În concluzie, $b(X)$ trebuie să fie identic cu zero. Am demonstrat astfel şi partea a doua a teoremei, care spune că orice polinom de cod este un multiplu de $g(X)$.

Polinomul $a(X)$ din (7.10) are $(n–r)$ coeficienţi binari, iar numărul seturilor posibile de coeficienţi este egal cu 2^{n-r}. Deci, numărul polinoamelor binare de grad $(n–1)$ sau mai mic care sunt multipli de $g(X)$ este 2^{n-r}. Din teorema 7.3, urmează că aceste polinoame formează toate polinoamele de cod ale codului ciclic (n,k) C. Dar fiindcă în C există 2^k polinoame de cod, 2^{n-r} trebuie să fie egal cu 2^k. Rezultă că:

$$r = n - k\ . \tag{7.13}$$

Prin urmare, polinomul de cod diferit de zero de grad minim dintr-un cod ciclic (n,k) C este de forma următoare:

$$g(X) = 1 + g_1 X + g_2 X^2 + \cdots + g_{n-k-1} X^{n-k-1} + X^{n-k} \tag{7.14}$$

Am demonstrat astfel următoarea teoremă:

TEOREMA 7.4: Într-un cod ciclic (n,k), există un polinom şi numai unul de grad $(n–k)$, dat de (7.14).

Din teorema 7.4, urmează că orice polinom de cod $v(X)$ dintr-un cod ciclic (n,k) se poate exprima în forma următoare:

$$v(X) = u(X)g(X) = \left(u_0 + u_1 X + \cdots + u_{k-1} X^{k-1}\right) g(X) \tag{7.15}$$

Ecuația (7.15) este baza codării. Dacă $u_0, u_1, \cdots, u_{k-1}$, coeficienţii lui $u(X)$, sunt cei k biţi de mesaj, polimonul de cod corespunzător este $v(X)$. Codarea se realizează, deci, înmulţind mesajul $u(X)$ cu $g(X)$. Prin urmare, un cod ciclic (n,k) este specificat complet de polinomul său de cod diferit de zero de grad minim $g(X)$ dat de (7.14). Polinomul $g(X)$ se numeşte *polinomul generator* al codului. Gradul lui $g(X)$ este egal cu numărul biţilor de control.

EXEMPLUL 7.1: În tabelul 7.1, se dă codul ciclic (7,4) generat de $g(X)=1+X+X^3$.

Tabelul 7.1

Cod ciclic (7,4) generat de $g(X)=1+X+X^3$.

Mesaje	Vectori de cod	Polinoame de cod
0 0 0 0	0 0 0 0 0 0 0	$0=0\cdot g(X)$
1 0 0 0	1 1 0 1 0 0 0	$1+X+X^3=1\cdot g(X)$
0 1 0 0	0 1 1 0 1 0 0	$X+X^2+X^4=X\cdot g(X)$
1 1 0 0	1 0 1 1 1 0 0	$1+X^2+X^3+X^4=(1+X)\cdot g(X)$
0 0 1 0	0 0 1 1 0 1 0	$X^2+X^3+X^5=X^2\cdot g(X)$
1 0 1 0	1 1 1 0 0 1 0	$1+X+X^2+X^5=(1+X^2)\cdot g(X)$
0 1 1 0	0 1 0 1 1 1 0	$X+X^3+X^4+X^5=(X+X^2)\cdot g(X)$
1 1 1 0	1 0 0 0 1 1 0	$1+X^4+X^5=(1+X+X^2)\cdot g(X)$
0 0 0 1	0 0 0 1 1 0 1	$X^3+X^4+X^6=X^3\cdot g(X)$
1 0 0 1	1 1 0 0 1 0 1	$1+X+X^4+X^6=(1+X^3)\cdot g(X)$
0 1 0 1	0 1 1 1 0 0 1	$X+X^2+X^3+X^6=(X+X^3)\cdot g(X)$
1 1 0 1	1 0 1 0 0 0 1	$1+X^2+X^6=(1+X+X^3)\cdot g(X)$
0 0 1 1	0 0 1 0 1 1 1	$X^2+X^4+X^5+X^6=(X^2+X^3)\cdot g(X)$
1 0 1 1	1 1 1 1 1 1 1	$1+X+X^2+X^3+X^4+X^5+X^6=(1+X^2+X^3)\cdot g(X)$
0 1 1 1	0 1 0 0 0 1 1	$X+X^5+X^6=(X+X^2+X^3)\cdot g(X)$
1 1 1 1	1 0 0 1 0 1 1	$1+X^3+X^5+X^6=(1+X+X^2+X^3)\cdot g(X)$

TEOREMA 7.5: Polinomul generator $g(X)$ al unui cod ciclic (n,k) este un factor al lui X^n+1.

DEMONSTRAȚIE
Conform ecuației (7.4), dacă se împarte $X^i v(X)$ la X^n+1, restul este $v^{(i)}(X)$. Fie $i=k$. Polinomul $X^k g(X)$ este de grad n. Împărțind $X^k g(X)$ la X^n+1 obținem:

$$X^k g(X)=(X^n+1) + g^{(k)}(X) \tag{7.16}$$

În (7.16), $g^{(k)}(X)$ este restul împărțirii. Dar $g^{(k)}(X)$ este polinomul de cod obținut deplasând $g(X)$ ciclic la dreapta de k ori, astfel încât este un multiplu de $g(X)$, să spunem $g^{(k)}(X) = a(X)g(X)$. Din (7.16) obținem că :

$$X^n+1 = [X^k+a(X)]g(X) \tag{7.17}$$

Prin urmare, $g(X)$ este un factor al lui X^n+1.

Există, însă, un cod ciclic (n,k) pentru orice pereche de numere naturale n şi k? Răspunsul la această legitimă întrebare este dat de teorema următoare.

TEOREMA 7.6: Dacă $g(X)$ este un polinom de grad $(n–k)$ care este un factor al lui (X^n+1), $g(X)$ generează un cod ciclic (n,k).

DEMONSTRAȚIE
Fie $g(X)$ un polinom de grad $(n–k)$ care este un factor al lui (X^n+1). Să considerăm cele k polinoame $g(X), Xg(X), \cdots, X^{k-1}g(X)$, care au toate grad $(n–1)$ sau mai mic. O combinație liniară a acestor k polinoame :

$$\begin{aligned} v(X) &= a_0 g(X) + a_1 Xg(X) + \cdots + a_{k-1}X^{k-1}g(X) = \\ &(a_0 + a_1 X + \cdots + a_{k-1}X^{k-1})g(X) \end{aligned} \tag{7.18}$$

este tot un polinom de grad $(n–1)$ sau mai mic şi este un multiplu de $g(X)$. Există în total 2^k astfel de polinoame şi ele alcătuiesc un cod liniar (n,k). Rămâne să demonstrăm că acest cod liniar este ciclic. Fie $v(X) = v_0 + v_1 X + \cdots + v_{n-1}X^{n-1}$ un polinom de cod din acest cod. Înmulțind $v(X)$ cu X, obținem

$$\begin{aligned} Xv(X) &= v_0 X + v_1 X^2 + \cdots + v_{n-2}X^{n-1} + v_{n-1}X^n \\ &= v_{n-1}(X^n+1) + v_{n-1} + v_0 X + v_1 X^2 + \cdots + v_{n-2}X^{n-1} \\ &= v_{n-1}(X^n+1) + v^{(1)}(X) \end{aligned} \tag{7.19}$$

În (7.19), $v^{(1)}(X)$ este o deplasare ciclică a lui $v(X)$. Întrucât ambele polinoame $Xv(X)$ şi (X^n+1) sunt divizibile cu $g(X)$, rezultă că şi $v^{(1)}(X)$ trebuie să fie divizibil cu $g(X)$. Deci, $v^{(1)}(X)$ este un multiplu de $g(X)$ şi este o combinaţie liniară de $g(X), Xg(X), \cdots, X^{k-1}g(X)$. Prin urmare, $v^{(1)}(X)$ este tot un polinom de cod. Conform definiţiei, codul bloc liniar generat de $g(X), Xg(X), \cdots, X^{k-1}g(X)$ este un cod ciclic (n,k).

Teorema 7.6 spune că orice factor al lui (X^n+1) de grad $(n–k)$ generează un cod ciclic (n,k). Pentru n mare, (X^n+1) poate avea mulţi factori de grad $(n–k)$. Nu toate aceste polinoame, însă, generează coduri performante.

EXEMPLUL 7.2: Polinomul (X^7+1) se poate descompune în factori primi astfel:

$$X^7+1=(1+X)(1+X+X^3)(1+X^2+X^3).$$

Codul dat cu titlu de exemplu în Tabelul 7.1 este generat de $g(X)=1+X+X^3$. Se vede că el are pondere minimă 3, astfel încât poate corecta o eroare de bit pe cuvânt de cod. Codul generat de $(1+X)$ are un singur bit de control prin paritate şi nu posedă dacât o capacitate de detecţie, dar nu şi de corecţie a erorilor.

Codul (7,4) dat în tabelul 7.1 este nesistematic. Dar, fiind dat polinomul generator $g(X)$ al unui cod ciclic (n,k), codul poate fi pus în formă sistematică: cei mai de la dreapta k biţi din fiecare cuvânt de cod sunt biţii de informaţie fără nici o modificare , iar cei mai de la stânga $(n–k)$ biţi sunt biţii de control.

Fie $\mathbf{u}=(u_0,u_1,\cdots,u_{k-1})$ mesajul de codat cu polinomul de mesaj corespunzător:

$$u(X)=u_0+u_1X+\cdots+u_{k-1}X^{k-1} \tag{7.20}$$

Înmulţind $u(X)$ cu X^{n-k}, obţinem un polinom de grad $(n–1)$ sau mai mic

$$X^{n-k}u(X)=u_0X^{n-k}+u_1X^{n-k+1}+\cdots+u_{k-1}X^{n-1} \tag{7.21}$$

Împărţind acum $X^{n-k}u(X)$ la polinomul generator $g(X)$, avem

$$X^{n-k}u(X)=a(X)g(X)+b(X) \tag{7.22}$$

În (7.22), $a(X)$ este câtul, iar $b(X)$ restul împărţirii. Întrucât gradul lui $g(X)$ este $(n–k)$, gradul restului $b(X)$ trebuie să fie $(n–k–1)$ sau mai mic, astfel încât

$$b(X) = b_0 + b_1 X + \cdots + b_{n-k-1} X^{n-k-1} \tag{7.23}$$

Rearanjând (7.22), obţinem următorul polinom de grad $(n–1)$ sau mai mic:

$$b(X) + X^{n-k} u(X) = a(X) g(X) \tag{7.24}$$

Acest polinom, fiind un multiplu al polinomului generator $g(X)$, este un polinom de cod al codului ciclic generat de $g(X)$. Scriind explicit (7.24), avem:

$$\begin{aligned} b(X) + X^{n-k} u(X) = b_0 + b_1 X + \cdots + b_{n-k-1} X^{n-k-1} + \\ + u_0 X^{n-k} + u_1 X^{n-k+1} + \cdots + u_{k-1} X^{n-1} \end{aligned} \tag{7.25}$$

Polinomul din (7.25) corespunde cuvântului de cod $(b_0, b_1, \cdots, b_{n-k-1}, u_0, u_1, \cdots, u_{k-1})$. Se vede că cei $(n–k)$ biţi de control sunt coeficienţii restului împărţirii polinomului de mesaj $X^{n-k} u(X)$ la polinomul generator $g(X)$. Codarea în formă sistematică, deci, constă în trei paşi:

Pasul 1. Se înmulţeşte mesajul $u(X)$ cu X^{n-k}.

Pasul 2. Prin împărţirea lui $X^{n-k} u(X)$ la polinomul generator $g(X)$, se obţine restul $b(X)$, având drept coeficienţi biţii de control.

Pasul 3. Se combină $b(X)$ cu $X^{n-k} u(X)$ pentru a obţine polinomul de cod $b(X) + X^{n-k} u(X)$.

EXEMPLUL 7.3: Fie codul ciclic (7,4) generat de $g(X) = 1 + X + X^3$. Cele 16 cuvinte de cod în formă sistematică sunt listate în tabelul 7.2.

Tabelul 7.2

Cod ciclic (7,4) generat de $g(X) = 1 + X + X^3$ în formă sistematică

Mesaje	Cuvinte de cod	Polinoame de cod
0 0 0 0	0 0 0 | 0 0 0 0	$0 = 0 \cdot g(X)$
1 0 0 0	1 1 0 | 1 0 0 0	$1 + X + X^3 = 1 \cdot g(X)$
0 1 0 0	0 1 1 | 0 1 0 0	$X + X^2 + X^4 = X \cdot g(X)$
1 1 0 0	1 0 1 | 1 1 0 0	$1 + X^2 + X^3 + X^4 = (1 + X) \cdot g(X)$

Tabelul 7.2 (*continuare*)

Mesaje	Cuvinte de cod	Polinoame de cod
0 0 1 0	1 1 1 | 0 0 1 0	$1+X+X^2+X^5=\left(1+X^2\right)\cdot g\left(X\right)$
1 0 1 0	0 0 1 | 1 0 1 0	$X^2+X^3+X^5=X^2\cdot g\left(X\right)$
0 1 1 0	1 0 0 | 0 1 1 0	$1+X^4+X^5=\left(1+X+X^2\right)\cdot g\left(X\right)$
1 1 1 0	0 1 0 | 1 1 1 0	$X+X^3+X^4+X^5=\left(X+X^2\right)\cdot g\left(X\right)$
0 0 0 1	1 0 1 | 0 0 0 1	$1+X^2+X^6=\left(1+X+X^3\right)\cdot g\left(X\right)$
1 0 0 1	0 1 1 | 1 0 0 1	$X+X^2+X^3+X^6=\left(X+X^3\right)\cdot g\left(X\right)$
0 1 0 1	1 1 0 | 0 1 0 1	$1+X+X^4+X^6=\left(1+X^3\right)\cdot g\left(X\right)$
1 1 0 1	0 0 0 | 1 1 0 1	$X^3+X^4+X^6=X^3\cdot g\left(X\right)$
0 0 1 1	0 1 0 |0 0 1 1	$X+X^5+X^6=\left(X+X^2+X^3\right)\cdot g\left(X\right)$
1 0 1 1	1 0 0 | 1 0 1 1	$1+X^3+X^5+X^6=\left(1+X+X^2+X^3\right)\cdot g\left(X\right)$
0 1 1 1	0 0 1 | 0 1 1 1	$X^2+X^4+X^5+X^6=\left(X^2+X^3\right)\cdot g\left(X\right)$
1 1 1 1	1 1 1 | 1 1 1 1	$1+X+X^2+X^3+X^4+X^5+X^6=\left(1+X^2+X^5\right)\cdot g\left(X\right)$

7.2. MATRICELE GENERATOARE ŞI DE CONTROL ALE CODURILOR CICLICE

Am văzut că un cod ciclic este generat cu ajutorul unui polinom generator $g(X)$, în vreme ce, pentru a genera un cod bloc liniar neciclic, este necesară o matrice **G**. Aceasta este o simplificare considerabilă şi justifică interesul acordat codurilor ciclice. În această secțiune, vom arăta că orice cod ciclic are o matrice generatoare **G** şi o matrice de control **H**, ca orice cod bloc liniar.

Să considerăm un cod ciclic (n,k) C cu polinom generator $g\left(X\right)=g_0+g_1X+\cdots+g_{n-k}X^{n-k}$, unde $g_0=g_{n-k}=1$. Am arătat că cele k polinoame de cod $g(X), Xg(X),\cdots, X^{k-1}g(X)$ generează C. Cu alte cuvinte, ele constituie un sistem de vectori liniar independenți şi formează, deci, o bază a spațiului vectorial al celor 2^k polinoame de cod. Prin urmare,

putem utiliza cele *k n*–tupluri binare corespunzătoare acestor *k* polinoame de cod drept linii ale unei matrice $k \times n$, obținând următoare matrice generatoare pentru *C*:

$$\mathbf{G} = \begin{bmatrix} g_0 & g_1 & g_2 & \ldots & g_{n-k} & 0 & 0 & 0 & \ldots & 0 \\ 0 & g_0 & g_1 & g_2 & \ldots & 0 & g_{n-k} & 0 & 0 & \ldots & 0 \\ 0 & 0 & g_0 & g_1 & g_2 & \ldots & 0 & 0 & g_{n-k} & 0 & \ldots & 0 \\ . & & & & & & & & & & & \\ . & & & & & & & & & & & \\ . & & & & & & & & & & & \\ 0 & 0 & 0 & 0 & 0 & \ldots & g_0 & g_1 & g_2 & \ldots & g_{n-k} \end{bmatrix} \tag{7.26}$$

În general, **G** nu este în formă sistematică. Ea se poate însă pune în formă sistematică prin operații cu linii, aşa cum se arată în exemplul următor.

EXEMPLUL 7.4: Codul ciclic (7,4) dat de tabelul 7.1, cu polinom generator $g(X) = 1 + X + X^3$, are matricea generatoare:

$$\mathbf{G} = \begin{pmatrix} 1 & 1 & 0 & 1 & 0 & 0 & 0 \\ 0 & 1 & 1 & 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 1 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 & 1 & 0 & 1 \end{pmatrix}.$$

Se vede că **G** nu este în formă sistematică. Pentru a obține matricea $\mathbf{G}'$ în formă sistematică, adunăm prima linie la linia a treia şi suma primelor două linii la linia a patra:

$$\mathbf{G}' = \begin{pmatrix} 1 & 1 & 0 & 1 & 0 & 0 & 0 \\ 0 & 1 & 1 & 0 & 1 & 0 & 0 \\ 1 & 1 & 1 & 0 & 0 & 1 & 0 \\ 1 & 0 & 1 & 0 & 0 & 0 & 1 \end{pmatrix}..$$

Cele două matrice, **G** şi $\mathbf{G}'$, generează acelaşi cod, în sensul că ele generează mulțimi identice ale cuvintelor de cod, dar atenție: corespondența dintre mesaj şi cuvântul de cod corespunzător nu este aceeaşi.

Vom introduce acum *polinomul de control* al unui cod ciclic. Ştim că polinomul generator *g*(*X*) este un factor al lui $(X^n + 1)$, să spunem

$$X^n + 1 = g(X)h(X). \tag{7.27}$$

Polinomul $h(X)$ din (7.27) are grad k şi este de forma următoare, cu $h_0 = h_k = 1$:

$$h(X) = h_0 + h_1 X + \cdots + h_k X^k. \tag{7.28}$$

Vom arăta că din $h(X)$ se poate obține o matrice de control a lui C. Fie $\mathbf{v} = (v_0, v_1, \cdots, v_{n-1})$ un vector de cod din C. Avem atunci $v(X) = a(X)g(X)$. Înmulțind $v(X)$ cu $h(X)$, obținem

$$\begin{aligned} v(X)h(X) &= a(X)g(X)h(X) \\ &= a(X)(X^n + 1) \\ &= a(X) + X^n a(X). \end{aligned} \tag{7.29}$$

Dar gradul lui $a(X)$ este (k–1) sau mai mic, astfel încât în $a(X) + X^n a(X)$ nu apar puterile $X^k, X^{k+1}, \cdots, X^{n-1}$. Aceasta înseamnă că, dacă dezvoltăm membrul stâng al lui (7.29), în produsul $v(X)$ $h(X)$, coeficienții lui $X^k, X^{k+1}, \cdots, X^{n-1}$ trebuie să fie egali cu zero. Obținem astfel (n–k) egalități:

$$\sum_{i=0}^{k} h_i v_{n-i-j} = 0 \quad \text{pentru} \quad 1 \le j \le n-k \tag{7.30}$$

Cele (n–k) egalități (7.30) ne permit să construim o matrice $(n-k) \times n$, notată cu **H**, astfel încât orice vector de cod **v** din C să fie ortogonal cu orice linie a lui **H**:

$$\mathbf{H} = \begin{bmatrix} h_k & h_{k-1} & h_{k-2} & \ldots & h_0 & 0 & 0 & \ldots & 0 \\ 0 & h_k & h_{k-1} & h_{k-2} & \ldots & h_0 & 0 & \ldots & 0 \\ 0 & 0 & h_k & h_{k-1} & h_{k-2} & \ldots & h_0 & \ldots & 0 \\ . & & & & & & & & \\ . & & & & & & & & \\ . & & & & & & & & \\ 0 & 0 & 0 & 0 & 0 & \ldots & h_k & h_{k-1} & h_{k-2} & \ldots & h_0 \end{bmatrix} \tag{7.31}$$

Matricea **H** nu se construieşte direct din $h(X)$; mai întâi, se formează *reciprocul* lui $h(X)$, definit după cum urmează:

$$X^k h\left(X^{-1}\right) = h_k + h_{k-1}X + h_{k-2}X^2 + \cdots + h_0 X^k \tag{7.32}$$

Dacă în (7.27) înlocuim X cu X^{-1}, obţinem că:

$$X^{-1} + 1 = g\left(X^{-1}\right)h\left(X^{-1}\right)X^k h\left(X^{-1}\right) \tag{7.33}$$

Din (7.33), se vede că şi $X^k h\left(X^{-1}\right)$, reciprocul lui $h(X)$, este un factor al lui $X^n + 1$. Polinomul reciproc $X^k h\left(X^{-1}\right)$ generează un cod ciclic $(n,n–k)$ având drept matrice generatoare **H** din (7.31). Întrucât matricea de control a codului C, **H**, se obţine din polinomul $h(X)$, fie şi indirect , $h(X)$ se numeşte *polinomul de control* al lui C. Un cod ciclic este univoc specificat de polinomul său de control. Am demonstrat astfel o altă proprietate importantă, enunţată în teorema următoare.

TEOREMA 7.7: Fie C un cod ciclic (n,k) cu polinom generator $g(X)$. Codul dual C_d al lui C este şi el ciclic şi este generat de polinomul $X^k h\left(X^{-1}\right)$, unde $h(X) = \left(X^n + 1\right)/\, g(X)$.

EXEMPLUL 7.5: Să considerăm codul ciclic (7,4) dat în tabelul 7.1. Acesta are polinomul generator $g(X) = 1 + X + X^3$. Polinomul de control este

$$h(X) = \frac{X^7 + 1}{X^3 + X + 1} = X^4 + X^2 + X + 1.$$

Reciprocul lui $h(X)$ este

$$X^4 h\left(X^{-1}\right) = X^4\left(1 + X^{-1} + X^{-2} + X^{-4}\right) = 1 + X^2 + X^3 + X^4.$$

Polinomul acesta divide $X^7 + 1$:

$$\left(X^7 + 1\right)/\, X^4 h\left(X^{-1}\right) = 1 + X^2 + X^3.$$

Cuvintele codului (7,3) generat de $X^4 h\left(X^{-1}\right) = 1 + X^2 + X^3 + X^4$ sunt date în tabelul 7.3. Se vede că distanţa minimă a codului este 4. Prin urmare, codul este capabil să corecteze o singură eroare de bit şi simultan să detecteze orice combinaţie de două erori de bit dintr-un cuvânt de şapte biţi.

Tabelul 7.3

Cod ciclic (7,3) generat de polinomul $1 + X^2 + X^3 + X^4$

Mesaje	Cuvinte de cod
0 0 0	0 0 0 0 0 0 0
1 0 0	1 0 1 1 1 0 0
0 1 0	0 1 0 1 1 1 0
1 1 0	1 1 1 0 0 1 0
0 0 1	0 0 1 0 1 1 1
1 0 1	1 0 0 1 0 1 1
0 1 1	0 1 1 1 0 0 1
1 1 1	1 1 0 0 1 0 1

Matricea generatoare **G** a unui cod ciclic se poate pune în formă sistematică. Pentru aceasta, se împarte X^{n-k-i} la polinomul generator $g(X)$, $i = 0,1,\cdots,k-1$, obținând

$$X^{n-k-1} = a_i(X)g(X) + b_i(X) \tag{7.34}$$

Restul $b_i(X)$ are forma următoare:

$$b_i(X) = b_{i0} + b_{i1}X + b_{i2}X^2 + \cdots + b_{i,n-k-1}X^{n-k-1}. \tag{7.35}$$

Din (7.34), rezultă că polinoamele $b_i(X) + X^{n-k-i}$ pentru $i = 0,1,\cdots,k-1$, sunt multipli de $g(X)$, aşa încât sunt polinoame de cod. Aranjând aceste k polinoame de cod ca linii ale unei matrice $k \times n$, obținem:

$$\mathbf{G} = \begin{bmatrix} b_{00} & b_{01} & b_{02} & \cdots & b_{0,n-k-1} & 1 & 0 & 0 & \cdots & 0 \\ b_{10} & b_{11} & b_{12} & \cdots & b_{1,n-k-1} & 0 & 1 & 0 & \cdots & 0 \\ b_{20} & b_{21} & b_{22} & \cdots & b_{2,n-k-1} & 0 & 0 & 1 & \cdots & 0 \\ \cdot & & & & & & & & & \\ \cdot & & & & & & & & & \\ \cdot & & & & & & & & & \\ b_{k-1,0} & b_{k-1,1} & b_{k-1,2} & \cdots & b_{k-1,n-k-1} & 0 & 0 & 0 & \cdots & 1 \end{bmatrix} \tag{7.36}$$

Aceasta este matricea generatoare a lui C în formă sistematică. Matricea de control a lui C este:

$$\mathbf{H} = \begin{bmatrix} 1\ 0\ 0\ \ldots\ 0 & b_{00} & b_{10} & b_{20} & \ldots & b_{k-1,0} \\ 0\ 1\ 0\ \ldots\ 0 & b_{01} & b_{11} & b_{21} & \ldots & b_{k-1,1} \\ 0\ 0\ 1\ \ldots\ 0 & b_{02} & b_{12} & b_{22} & \ldots & b_{k-1,2} \\ . & & & & & \\ . & & & & & \\ . & & & & & \\ 0\ 0\ 0\ \ldots\ 1 & b_{0,n-k-1} & b_{1,n-k-1} & b_{2,n-k-1} & \ldots & b_{k-1,n-k-1} \end{bmatrix} \tag{7.37}$$

EXEMPLUL 7.6: Fie din nou codul ciclic (7,4) generat de $g(X) = 1 + X + X^3$. Împărțind X^3, X^4, X^5 şi X^6 la $g(X)$, obținem:

$$\begin{aligned} X^3 &= g(X) + (1 + X) \\ X^4 &= Xg(X) + (X + X^2) \\ X^5 &= (1 + X^2)g(X) + (1 + X + X^2) \\ X^6 &= (1 + X + X^3)g(X) + (1 + X^2) \end{aligned}$$

Rearanjând ecuațiile de mai sus, obținem următoarele patru polinoame de cod:

$$\begin{aligned} v_0(X) &= 1 + X + X^3 \\ v_1(X) &= X + X^2 + X^4 \\ v_2(X) &= 1 + X + X^2 + X^5 \\ v_3(X) &= 1 + X^2 + X^6 \end{aligned}$$

Luând aceste patru polinoame de cod drept linii ale unei matrice 4×7, deducem următoarea matrice generatoare în formă sistematică pentru codul ciclic (7,4):

$$\mathbf{G} = \begin{pmatrix} 1 & 1 & 0 & 1 & 0 & 0 & 0 \\ 0 & 1 & 1 & 0 & 1 & 0 & 0 \\ 1 & 1 & 1 & 0 & 0 & 1 & 0 \\ 1 & 0 & 1 & 0 & 0 & 0 & 1 \end{pmatrix}.$$

G este identică după cum se vede cu matricea $\mathbf{G}'$ obținută în exemplul 7.4 prin operații cu liniile matricei în formă nesistematică.

7.3. CODAREA CODURILOR CICLICE

Pentru comoditate, repetăm aici cei trei paşi necesari pentru a coda un cod ciclic (n,k) în formă sistematică:

1. se înmulţeşte polinomul de mesaj $u(X)$ cu X^{n-k};
2. se împarte $X^{n-k}u(X)$ la $g(X)$ pentru a obţine restul $b(X)$;
3. se formează cuvântul de cod $b(X)+X^{n-k}u(X)$.

Cei trei paşi se pot realiza cu un circuit de împărţire. Aşa cum se vede în fig. 7.2, acesta este un registru de deplasare cu $(n–k)$ etaje cu legături de reacţie bazate pe polinomul generator

$$g(X)=1+g_1X+g_2X^2+\cdots+g_{n-k-1}X^{n-k-1}+X^{n-k}.$$

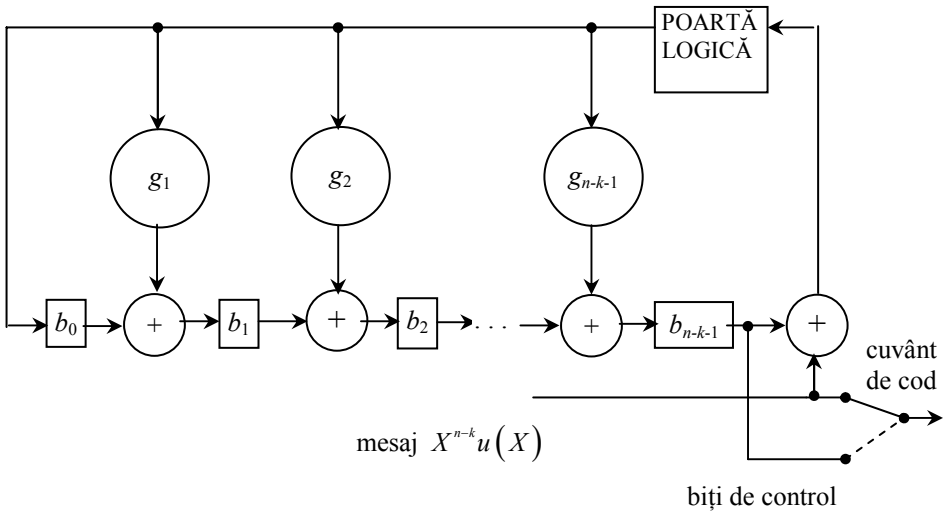


Fig. 7.2. Codor sistematic pentru codul ciclic (n,k) generat de $g(X)$.

Funcţionarea poate fi descrisă după cum urmează:

Pasul 1. Cu poarta logică deschisă, cei k biţi de informaţie $u_0,u_1,\cdots,u_{k-1}$ se deplasează în circuit şi simultan spre modulator. Deplasarea mesajului $u(X)$ în circuit pe la extremitatea din dreapta este echivalentă cu înmulţirea lui $u(X)$ cu X^{n-k}. Îndată ce întregul mesaj va fi intrat în circuit, registrul de deplasare conţine restul împărţirii, iar cele $(n–k)$ simboluri binare din etajele sale constituie biţii de control.

Pasul 2. Poarta logică se închide, întrerupând legătura de reacţie.

Pasul 3. Prin deplasare, biții de control ies din registru, aplicându-se la intrarea modulatorului. Cei (n–k) biți de control $b_0, b_1, \cdots, b_{n-k-1}$, împreună cu cei k biți de informație, constituie un cuvânt de cod complet.

EXEMPLUL 7.7: Considerăm acelaşi cod ciclic (7,4) generat de $g(X) = 1 + X + X^3$. Codorul este arătat în fig. 7.3.

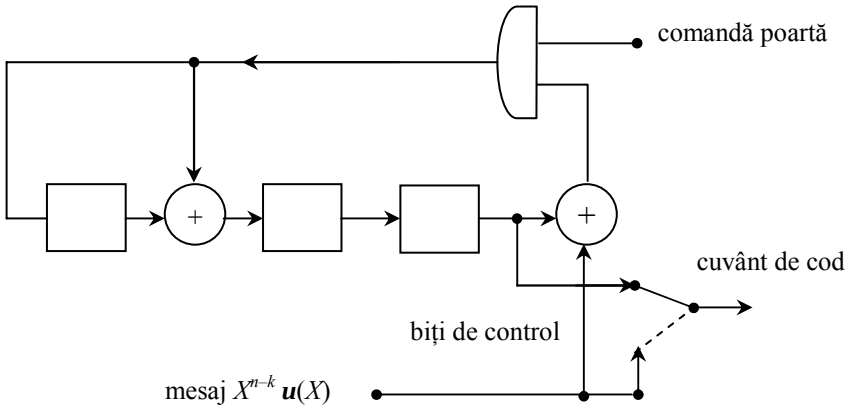


Fig. 7.3. Codor sistemtic pentru codul ciclic (7,4) generat de $1+X+X^3$

Cu titlu de exemplu, să presupunem că mesajul de codat este $\mathbf{u} = (1\ 0\ 1\ 1)$. Bitul cel mai din dreapta este primul care intră în codor. La fiecare puls de tact, conținutul registrului se deplasează la dreapta cu un etaj. Conținutul succesiv al celor trei etaje ale registrului de deplasare este după cum urmează:

Tabelul 7.4

Intrare	Tact	Conținutul registrului
	0	0 0 0
1	1	1 1 0
1	2	1 0 1
0	3	1 0 0
1	4	1 0 0

După patru deplasări, registrul conține (1 0 0). Cuvântul de cod complet este deci (1 0 0 1 0 1 1), iar polinomul de cod corespunzător este $1 + X^3 + X^5 + X^6$.

Codarea unui cod ciclic se mai poate realiza utilizând polinomul său de control $h(X) = h_0 + h_1 X + \cdots + h_k X^k$. Fie $\mathbf{v} = (v_0, v_1, \cdots, v_{n-1})$ un vector de cod. Componentele lui *v* satisfac cele (*n*–*k*) ecuații (7.30). Deoarece $h_k = 1$, aceste ecuații se pot pune în forma următoare:

$$v_{n-k-j} = \sum_{i=0}^{k-1} h_i v_{n-i-j} \quad \text{pentru } 1 \le j \le n-k \tag{7.38}$$

O expresie de forma (7.38) se numeşte *ecuație cu diferențe finite.* Pentru un cod ciclic în formă sistematică, cele mai din dreapta *k* componente ale fiecărui vector de cod, $v_{n-k}, v_{n-k+1}, \cdots, v_{n-1}$, sunt biții de informație. Dându-se aceşti *k* biți, (7.38) este o regulă pentru determinarea celor (*n*–*k*) biți de control, $v_0, v_1, \cdots, v_{n-k-1}$. Un codor bazat pe (7.38) are schema de principiu din fig. 7.4. Legăturile de reacție reflectă coeficienții polinomului de control *h*(*X*), unde $h_0 = h_k = 1$.

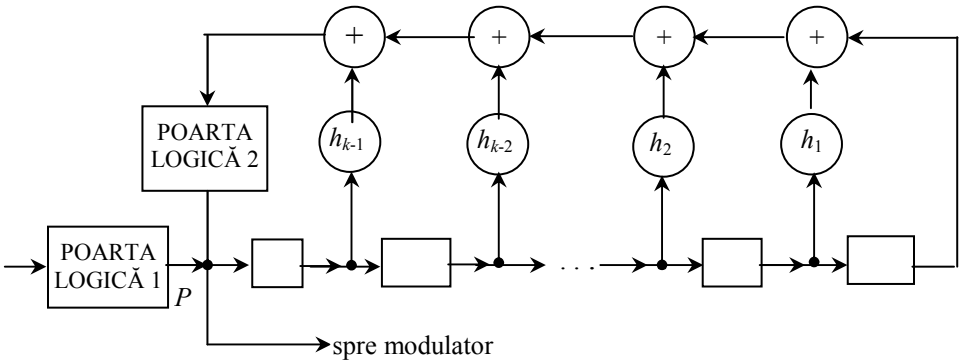


Fig. 7.4. Codor sistematic pentru codul ciclic (*n,k*) având polinom de control *h*(*X*).

Operația de codare se desfăşoară în următorii paşi:

Pasul 1. Inițial, poarta logică 1 este deschisă iar poarta 2 este închisă. Cei *k* biți de informație intră în registrul de deplasare începând cu u_{k-1} şi terminând cu u_0; simultan, în aceeaşi ordine, se prezintă la intrarea modulatorului.

Pasul 2. Îndată ce aceşti *k* biți de informație vor fi intrat în registrul de deplasare, poarta logică 1 se închide şi se deschide poarta 2. Se formează şi apare în punctul *P* primul simbol de control

$$\begin{aligned} v_{n-k-1} &= h_0 v_{n-1} + h_1 v_{n-2} + \cdots + h_{k-1} v_{n-k} \\ &= u_{k-1} + h_1 u_{k-2} + \cdots + h_{k-1} u_0 \end{aligned}$$

Pasul 3. Sub comanda semnalului de tact, registrul se deplasează o dată. Primul bit de control intră în registru şi simultan se

prezintă la intrarea modulatorului. Acum, în punctul *P* se formează al doilea bit de control

$$v_{n-k-2} = h_0 v_{n-2} + h_1 v_{n-3} + \cdots + h_{k-1} v_{n-k-1}$$
$$= u_{k-2} + h_1 u_{k-3} + \cdots + h_{k-2} u_0 + h_{k-1} v_{n-k-1}$$

Pasul 4. Se repetă pasul 3 până când se formează şi se deplasează spre modulator toţi cei (*n–k*) biţi de control. După aceea, poarta 1 se deschide din nou şi se închide poarta 2. Registrul este acum pregătit pentru următorul mesaj.

Acest codor utilizează un registru de deplasare cu *k* etaje. Comparând cele două circuite de codare prezentate în fig. 7.2. şi, respectiv, fig. 7.4, putem face următoarea remarcă:

1. Pentru coduri la care numărul biţilor de control (*n–k*) este mai mare decât numărul biţilor de mesaj *k*, codorul cu *k* etaje este preferabil.
2. În caz contrar, este preferabil circuitul de codare cu (*n–k*) etaje.

EXEMPLUL 7.8: Polinomul de control al codului ciclic (7,4) generat de $g(X) = 1 + X + X^3$ este:

$$h(X) = \frac{X^7 + 1}{X^3 + X + 1} = 1 + X + X^2 + X^4.$$

Codorul bazat pe acest $h(X)$ are schema de principiu arătată în figura 7.5.

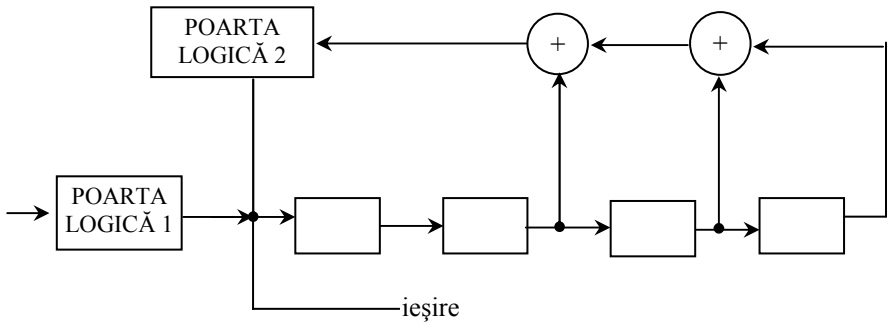


Fig. 7.5. Codor sistematic pentru codul ciclic (7,4) având polinomul de control $1+X+X^2+X^4$.

Fiecare cuvânt de cod este un vector de forma $\mathbf{v} = (v_0, v_1, v_2, v_3, v_4, v_5, v_6)$, unde v_3, v_4, v_5 şi v_6 sunt biţii de mesaj iar v_0, v_1 şi v_2 sunt biţii de control. Ecuaţia cu diferenţe finite care determină biţii de control este

$$\begin{aligned} v_{3-j} &= 1 \cdot v_{7-j} + 1 \cdot v_{6-j} + 1 \cdot v_{5-j} + 0 \cdot v_{4-j} \\ &= v_{7-j} + v_{6-j} + v_{5-j} \end{aligned} \quad \text{pentru } 1 \le j \le 3 .$$

Cu titlu de exemplu, să presupunem că mesajul de codat este (1 0 1 1). În acest caz, $v_3 = 1, v_4 = 0, v_5 = 1, v_6 = 1$. Primul bit de control este:

$$v_2 = v_6 + v_5 + v_4 = 1 + 1 + 0 = 0 .$$

Bitul al doilea de control este:

$$v_1 = v_5 + v_4 + v_3 = 1 + 0 + 1 = 0 .$$

Bitul al treilea de control este:

$$v_0 = v_4 + v_3 + v_2 = 0 + 1 + 0 = 1 .$$

Vectorul de cod corespunzător mesajului (1 0 1 1) este, deci, (1 0 0 1 0 1 1).

7.4. CALCULUL SINDROMULUI ŞI DETECŢIA ERORILOR

Atunci când se transmite un cuvânt de cod **v** pe un canal real, din cauza zgomotului, este posibil ca vectorul recepţionat $\mathbf{r} = (r_0, r_1, \cdots, r_{n-1})$ să fie diferit de **v**. La decodarea unui cod bloc liniar, primul pas este acela de a calcula sindromul $\mathbf{s} = \mathbf{r} \cdot \mathbf{H}^T$, unde **H** este matricea (*n–k, n*) de control. Dacă sindromul este zero, **r** este un cuvânt de cod iar decodorul nu are nici un motiv să nu accepte că acesta este chiar cel emis în realitate. Dacă, însă, sindromul nu este identic cu zero, **r** nu este un cuvânt de cod şi este astfel detectată prezenţa erorilor. Fie un cod ciclic în formă sistematică. Vectorul recepţionat **r** este considerat drept un polinom de grad (*n*–1) sau mai mic:

$$r(X) = r_0 + r_1 X + r_2 X^2 + \cdots + r_{n-1} X^{n-1} .$$

Împărţind $r(X)$ la polinomul generator $g(X)$, obţinem:

$$r(X) = a(X) g(X) + s(X) . \tag{7.39}$$

Restul $s(X)$ este un polinom de grad $(n–k–1)$ sau mai mic. Cei $(n–k)$ coeficienţi ai lui $s(X)$ formează sindromul **s**. Conform cu teorema 7.3, $s(X)$ este identic cu zero dacă şi numai dacă polinomul recepţionat $r(X)$ este un polinom de cod. Din acest motiv, $s(X)$ se numeşte sindromul codului ciclic. Calculul sindromului se poate face cu un circuit ca cel reprezentat în fig. 7.6. El este un circuit de împărţire realizat cu $(n–k)$ etaje de registru de deplasare care, însă, nu sunt interconectate direct, ci prin intermediul unor porţi logice de tip SAU EXCLUSIV.

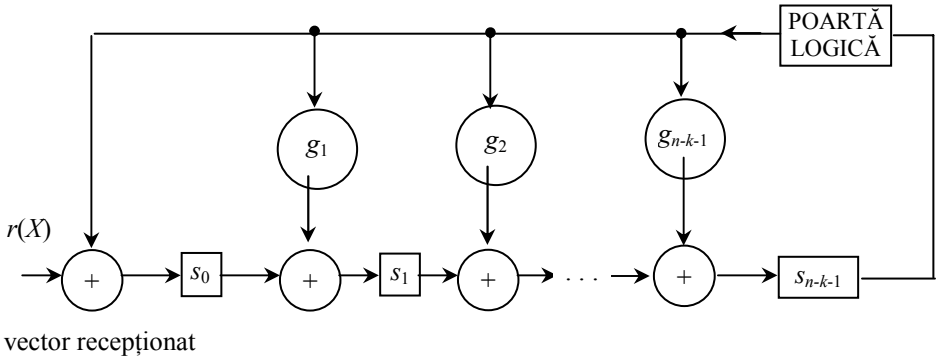


Fig. 7.6. Circuit de calcul al sindromului unui cod ciclic (n,k).

Iniţial, în toate cele $(n–k)$ etaje se înscrie 0. Polinomul recepţionat $r(X)$ este apoi deplasat în registru. Îndată ce şi ultimul bit, r_0, va fi intrat în registru, conţinutul registrului formează sindromul $s(X)$.

Datorită structurii ciclice a codului, sindromul $s(X)$ se bucură de următoarea proprietate:

TEOREMA 7.8: Fie $s(X)$ sindromul unui polinom de recepţie $r(X) = r_0 + r_1 X + \cdots + r_{n-1} X^{n-1}$. Restul $s^{(1)}(X)$ care rezultă prin împărţirea lui $Xs(X)$ la polinomul generator $g(X)$ este atunci sindromul lui $r^{(1)}(X)$, care este o deplasare ciclică a lui $r(X)$.

DEMONTRAŢIE
Prin definiţia codului ciclic, avem că:

$$r^{(1)}(X) = r_{n-1} + r_0 X + r_1 X^2 + \cdots + r_{n-2} X^{n-1}. \tag{7.40}$$

Dar

$$\begin{aligned} Xr(X) &= r_0 X + r_1 X^2 + \cdots + r_{n-2} X^{n-1} + r_{n-1} X^n \\ &= r_{n-1} + r_0 X + r_1 X^2 + \cdots + r_{n-2} X^{n-1} + r_{n-1}\left(1 + X^n\right) \end{aligned} \tag{7.41}$$

Din (7.41), este clar că

$$Xr(X) = r_{n-1}(X^n + 1) + r^{(1)}(X) \tag{7.42}$$

Rearanjând (7.42), avem:

$$r^{(1)}(X) = r_{n-1}(X^n + 1) + Xr(X). \tag{7.43}$$

Fie $p(X)$ restul împărțirii lui $r^{(1)}(X)$ la $g(X)$. Împărțind ambii membri ai ecuației (7.43) la $g(X)$ şi având în vedere că $X^n + 1 = g(X)h(X)$, avem:

$$c(X)g(X) + p(X) = r_{n-1}g(X)h(X) + X[a(X)g(X) + s(X)] \tag{7.44}$$

Rearanjând (7.44), deducem următoarea relație între $p(X)$ şi $X\,s(X)$:

$$Xs(X) = [c(X) + r_{n-1}h(X) + Xa(X)]g(X) + p(X) \tag{7.45}$$

Din (7.45), se vede că $p(X)$ nu este numai restul împărțirii lui $r^{(1)}(X)$ la $g(X)$, dar şi restul împărțirii lui $X\ s(X)$ la acelaşi $g(X)$. Prin urmare, $p(X) = s^{(1)}(X)$.

Din teorema 7.8, urmează că restul $s^{(i)}(X)$ ce se obține prin împărțirea lui $X^i s(X)$ la polinomul generator $g(X)$ este sindromul lui $r^{(i)}(X)$, care este deplasarea ciclică a lui $r(X)$ cu i locuri la dreapta. Această proprietate se arată utilă la decodarea codurilor ciclice.

Sindromul $s^{(1)}(X)$ al lui $r^{(1)}(X)$ se poate obține cu ajutorul schemei din fig. 7.6 deplasând, prin semnalul de tact, registrul sindrom cu o poziție, conținutul inițial fiind $s(X)$ iar poarta de intrare fiind închisă. Deplasarea cu o poziție a registrului sindrom al cărui conținut inițial este $s(X)$ este echivalentă cu împărțirea lui $X\ s(X)$ la polinomul generator $g(X)$. După deplasare, registrul conține $s^{(1)}(X)$.

Pentru a obține sindromul $s^{(i)}(X)$ al lui $r^{(i)}(X)$, se deplasează registrul sindrom de i ori, conținutul inițial fiind $s(X)$.

EXEMPLUL 7.9: În fig. 7.7, se dă schema pentru calculul sindromului pentru codul ciclic (7,4) generat de $g(X) = 1 + X + X^3$.

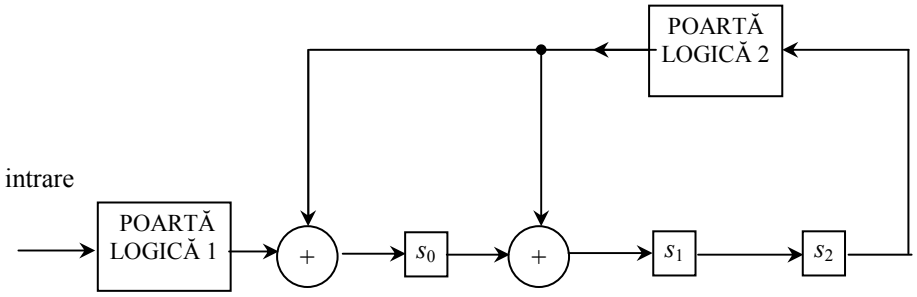


Fig. 7.7. Calculul sindromului pentru codul ciclic (7,4) generat de $1+X+X^3$.

Cu titlu de exemplu, să presupunem că vectorul recepționat este $\mathbf{r} = (0\ 0\ 1\ 0\ 1\ 1\ 0)$. Sindromul lui **r** este $\mathbf{s} = (1\ 0\ 1)$. Conținutul registrului, pe măsură ce biții de recepție se deplasează în circuit, este dat în tabelul de mai jos:

Tabelul 7.5

Deplasare	Intrare	Conținutul registrului			Notă
		s_0	s_1	s_2	
		0	0	0	stare inițială
1	0	0	0	0	
2	1	1	0	0	
3	1	1	1	0	
4	0	0	1	1	
5	1	0	1	1	
6	0	1	1	1	
7	0	1	0	1	sindromul s
8	–	1	0	0	sindromul $s^{(1)}$
9	–	0	1	0	sindromul $s^{(2)}$

După a şaptea deplasare, toți cei şapte biți ai vectorului de recepție vor fi intrat în circuit, astfel încât registrul conține sindromul $\mathbf{s} = (1\ 0\ 1)$. Dacă se mai deplasează o dată registrul cu poarta 1 închisă, noul conținut va fi $\mathbf{s}^{(1)} = (1\ 0\ 0)$, care este sindromul lui $\mathbf{r}^{(1)} = (0\ 0\ 0\ 1\ 0\ 1\ 1)$, o deplasare ciclică a lui **r**.

Fie $v(X)$ polinomul de cod emis şi $e(X) = e_0 + e_1 X + \cdots + e_{n-1} X^{n-1}$ n-tuplul de eroare. Polinomul de recepție este:

$$r(X) = v(X) + e(X) \tag{7.46}$$

Dar $v(X)$ este multiplu de $g(X)$, să spunem $v(X) = b(X)g(X)$. Combinând (7.39) cu (7.46), obținem următoarea relație între n–tuplul de eroare şi sindrom:

$$e(X) = [a(X) + b(X)]g(X) + s(X) \tag{7.47}$$

Ecuația (7.47) arată că sindromul este restul împărțirii polinomului de eroare la polinomul generator. Se vede că $s(X)$ este identic cu zero dacă şi numai dacă n–tuplul de eroare $e(X) = 0$ sau dacă el coincide cu un polinom de cod.

Experiența ne arată că numai rareori erorile apar izolat; cel mai adesea, ele se grupează în pachete. Un pachet de erori de lungime l este o succesiune de l biți, cei mai mulți, dar nu neapărat toți, fiind eronați. În ipoteza că n–tuplul de eroare $e(X)$ include un pachet de erori de lungime (n–k) sau mai mică, putem exprima $e(X)$ în forma următoare:

$$e(X) = X^j B(X) \tag{7.48}$$

unde $0 \le j \le n-1$, iar $B(X)$ este un polinom de grad (n–k–1) sau mai mic. Întrucât gradul lui $B(X)$ este mai mic decât gradul polinomului generator $g(X)$, $B(X)$ nu este divizibil cu $g(X)$. Având în vedere că $g(X)$ este un factor al lui $(X^n + 1)$, iar X nu este un factor al lui $(X^n + 1)$, polinoamele $g(X)$ şi X^j trebuie să fie relativ prime. Iată de ce, $e(X) = X^j B(X)$ nu este divizibil cu $g(X)$. Drept rezultat, sindromul produs de $e(X)$ nu este egal cu zero. Concluzia care se trage din aceasta este că un cod ciclic (n,k) poate detecta orice pachet de erori de lungime (n–k) sau mai mică.

Pentru un cod ciclic, un n–tuplu de eroare în care erorile apar grupate la i locuri superioare şi la (l–i) locuri inferioare se consideră tot un pachet de lungime l sau mai puțin. De exemplu, $\mathbf{e} = (1\ 0\ 1\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 1\ 1\ 0\ 1)$ include un pachet de eroare de lungime 7.

7.5. DECODAREA CODURILOR CICLICE

Structura ciclică a unui cod ne permite să decodăm un vector recepționat $r(X) = r_0 + r_1 X + r_2 X^2 + \cdots + r_{n-1} X^{n-1}$ bit cu bit, serial. Cu alte cuvinte, la un moment dat, se decodează un singur bit dar, cu acelaşi circuit,

se decodează toți biții, succesiv. După calcularea sindromului, circuitul de decodare controlează dacă sindromul *s(X)* corespunde unui *n*–tuplu de eroare corectabil $e(X) = e_0 + e_1X + \cdots + e_{n-1}X^{n-1}$, cu o eroare în locul de ordin maxim X^{n-1}, adică $e_{n-1} = 1$. Dacă *s(X)* nu corespunde unui *n*–tuplu de eroare cu $e_{n-1} = 1$, se deplasează ciclic cu o poziție polinomul recepționat, memorat într-un registru tampon şi, simultan, registrul sindrom. Se obține astfel $r^{(1)}(X) = r_{n-1} + r_0X + \cdots + r_{n-2}X^{n-1}$, iar noul conținut al registrului sindrom formează sindromul $s^{(1)}(X)$ al lui $r^{(1)}(X)$. Bitul al doilea r_{n-2} al lui *r(X)* devine primul bit al lui $r^{(1)}(X)$. Acelaşi circuit de decodare va controla acum dacă $s^{(1)}(X)$ corespunde unui *n*–tuplu de eroare în poziția fruntaşă X^{n-1}.

Dacă sindromul *s*(*X*) al lui *r*(*X*) corespunde unui *n*–tuplu de eroare în poziția fruntaşă X^{n-1}, adică $e_{n-1} = 1$, primul bit recepționat r_{n-1} este eronat şi trebuie, deci, corectat. Corecția se efectuează facând suma $r_{n-1} \oplus e_{n-1}$. Această corecție are drept rezultat un polinom recepționat modificat, notat cu

$$r_1(X) = r_0 + r_1X + \cdots + r_{n-2}X^{n-2} + (r_{n-1} \oplus e_{n-1})X^{n-1}.$$

Se elimină astfel efectul bitului eronat e_{n-1} asupra sindromului *s(X)*. Pentru a obține sindromul polinomului de recepție modificat $r_1(X)$, se adună sindromul lui $e'(X) = X^{n-1}$ la *s(X)*. Se deplasează acum ciclic cu o poziție $r_1(X)$ şi, simultan, registrul sindrom. Rezultatul acestei deplasări este polinomul de recepție

$$r_1^{(1)}(X) = (r_{n-1} \oplus e_{n-1}) + r_0X + \cdots + r_{n-2}X^{n-1}.$$

Sindromul $s_1^{(1)}(X)$ al lui $r_1^{(1)}(X)$ este restul împărțirii lui $X[s(X) + X^{n-1}]$ la polinomul generator *g(x)*. Dar restul împărțirii lui *Xs(x)* la *g(x)* este $s_1^{(1)}(X)$ iar restul împrțirii lui X^n la *g(x)* este 1, astfel încât avem :

$$s_1^{(1)}(X) = s^{(1)}(X) + 1. \tag{7.49}$$

Prin urmare, dacă se adună 1 la extremitatea din stânga a registrului sindrom atunci când el este deplasat, se obține $s_1^{(1)}(X)$. Circuitul de decodare procedează la decodarea bitului de recepție r_{n-2}. Decodarea lui r_{n-2} şi a celorlalți biți de recepție se desfăşoară identic cu decodarea lui r_{n-1}. Ori de câte ori se detectează şi se corectează o eroare, efectul ei asupra

sindromului este înlăturat. Decodarea se opreşte după un total de *n* deplasări. Dacă *e(x)* este un polinom de eroare corectabil, la sfârşitul operaţiei de decodare, conţinutul registrului sindrom va fi zero iar polinomul de recepţie *r(x)* va fi decodat corect. Dacă, însă, la încheierea procesului de decodare, registrul sindrom conţine cel puţin un bit de 1, se detectează astfel o combinaţie necorectabilă de erori.

Pentru a se elimina efectul unui bit eronat asupra sindromului, bitul de eroare se introduce în registrul de deplasare pe la stânga printr-o poartă logică SAU EXCLUSIV. Reamintim că, prin definiţie,

$$r_{n-1} \oplus e_{n-1} = r_{n-1}\overline{e_{n-1}} + \overline{r_{n-1}}e_{n-1} \tag{7.50}$$

Din (7.50), este clar că $e_{n-1}=1$ inversează, sau neagă logic, r_{n-1}.

Schema de principiu a decodorului Meggitt, ce poartă numele celui care l-a proiectat, este arătată în fig. 7.8.

Decodorul funcţionează după cum urmează.

Pasul 1. Polinomul de recepţie $r(X)$ este memorat în registrul tampon şi, simultan, este deplasat în registrul sindrom pentru a forma sindromul.

Pasul 2. Circuitul de detecţie a *n*–tuplului de eroare este o schemă logică combinaţională proiectată astfel încât ieşirea sa este 1 logic dacă şi numai dacă sindromul din registrul sindrom corespunde unui *n*–tuplu de eroare corectabil, cu o eroare în poziţia fruntaşă X^{n-1}. Deci, dacă la ieşirea detectorului apare un 1 logic, se presupune că bitul de recepţie din etajul extrem dreapta al registrului tampon este eronat şi trebuie corectat; dacă, însă, la ieşirea detectorului apare 0 logic, se presupune că bitul de recepţie din etajul extrem dreapta este corect, astfel încât nu este necesară corecţia.

Pasul 3. Primul bit recepţionat este citit din registrul tampon. Simultan, se deplasează o dată registrul sindrom. Dacă primul bit recepţionat este găsit eronat, el este corectat de ieşirea detectorului. Ieşirea detectorului mai este aplicată, ca reacţie, şi la registrul sindrom, pentru a elimina efectul erorii asupra sindromului. Rezultă astfel un nou sindrom, ce corespunde vectorului de recepţie modificat, deplasat cu un loc la dreapta.

Pasul 4. Noul sindrom format la pasul 3 este utilizat pentru a detecta dacă cel de al doilea bit de recepţie (aflat acum în etajul extrem dreapta al registrului tampon) este sau nu eronat. Decodorul repetă paşii 2 şi 3. Cel de al doilea bit de recepţie

este corectat exact la fel cum a fost corectat şi primul bit de recepție.

Pasul 5. Decodorul continuă decodarea bit cu bit a vectorului de recepție până când întregul vector de recepție este citit din registrul tampon.

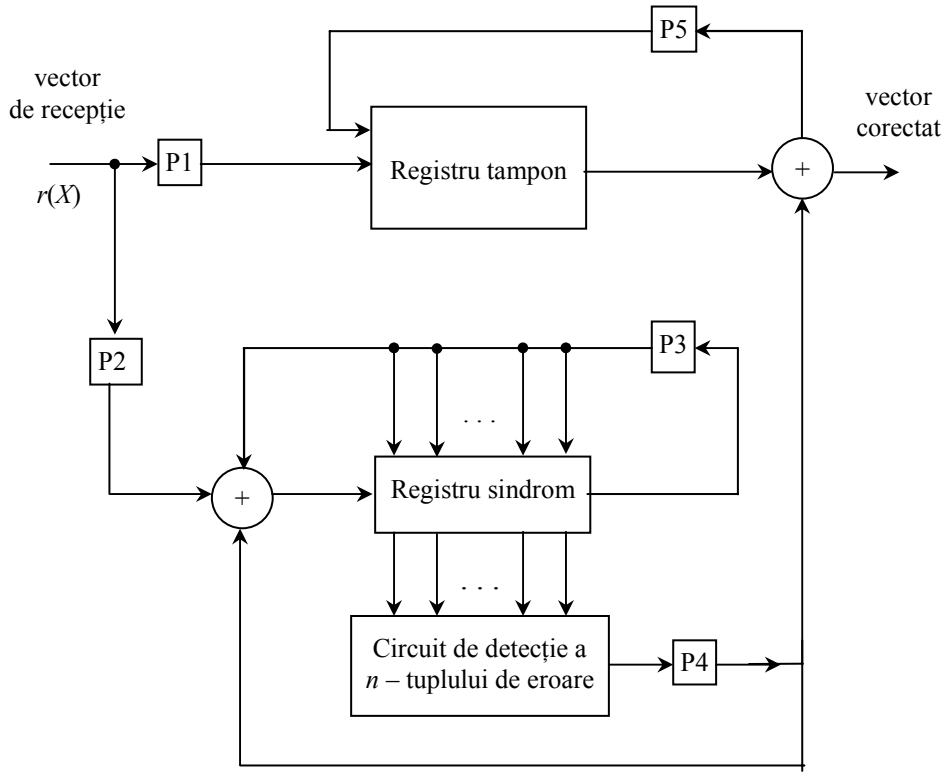


Fig. 7.8. Schema de principiu a decodorului Meggitt.

EXEMPLUL 7.10: Să considerăm decodarea codului ciclic (7.4) generat de $g(x)=1+X+X^3$. Acest cod are distanța minimă 3 şi poate, deci, corecta orice eroare singulară dintr-un cuvânt de cod de şapte biți. Există şapte 7-tupluri conținând o singură eroare. Aceste şapte 7-tupluri de eroare împreună cu vectorul având toate componentele egale cu 0 formează toți liderii de coset din tabloul de decodare standard. Ele formează, deci, toate combinațiile de eroare corectabile. Să presupunem că polinomul de recepție

$$r(X) = r_0 + r_1 X + r_2 X^2 + r_3 X^3 + r_4 X^4 + r_5 X^5 + r_6 X^6$$

este deplasat în registrul sindrom pe la extremitatea din stânga. Cele şapte combinații de o singură eroare şi sindroamele lor corespunzătoare sunt listate în Tabelul 7.6:

Tabelul 7.6

Polinom de eroare $e(X)$	Sindrom	Vector sindrom		
	$s(X)$	s_0	s_1	s_2
$e_0(X)=X^0$	1	1	0	0
$e_1(X)=X^1$	X	0	1	0
$e_2(X)=X^2$	X^2	0	0	1
$e_3(X)=X^3$	$1+X$	1	1	0
$e_4(X)=X^4$	$X+X^2$	0	1	1
$e_5(X)=X^5$	$1+X+X^2$	1	1	1
$e_6(X)=X^6$	$1+X^2$	1	0	1

După cum se vede, $e_6(X)=X^6$ este singurul polinom de eroare cu o eroare în poziția X^6. Dacă apare acest polinom de eroare, după ce întreg polinomul de recepție $r(X)$ va fi intrat în registrul sindrom, sindromul va fi (1 0 1). Detecția acestui sindrom indică faptul că r_6 este un bit eronat şi că trebuie corectat. Să presupunem că apare o singură eroare în poziția X^i, adică $e_i(X)=X^i$. După ce întregul polinom de recepție va fi fost deplasat în registrul sindrom, sindromul din registru nu este (1 0 1). Totuşi, după alte (6–i) deplasări, conținutul registrului sindrom va ajunge să fie (1 0 1), iar următorul bit de recepție ce trebuie să iasă din registrul tampon este bitul eronat. Din acest motiv, singurul sindrom care trebuie detectat este (1 0 1), ceea ce se poate realiza cu o poartă logică ŞI cu trei intrări şi cu un inversor logic, aşa cum se arată în fig. 7.9.

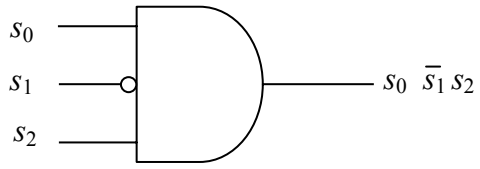


Fig. 7.9. Poartă logică pentru detecția sindromului (1 0 1).

Circuitul complet de decodare este reprezentat în fig. 7.10.

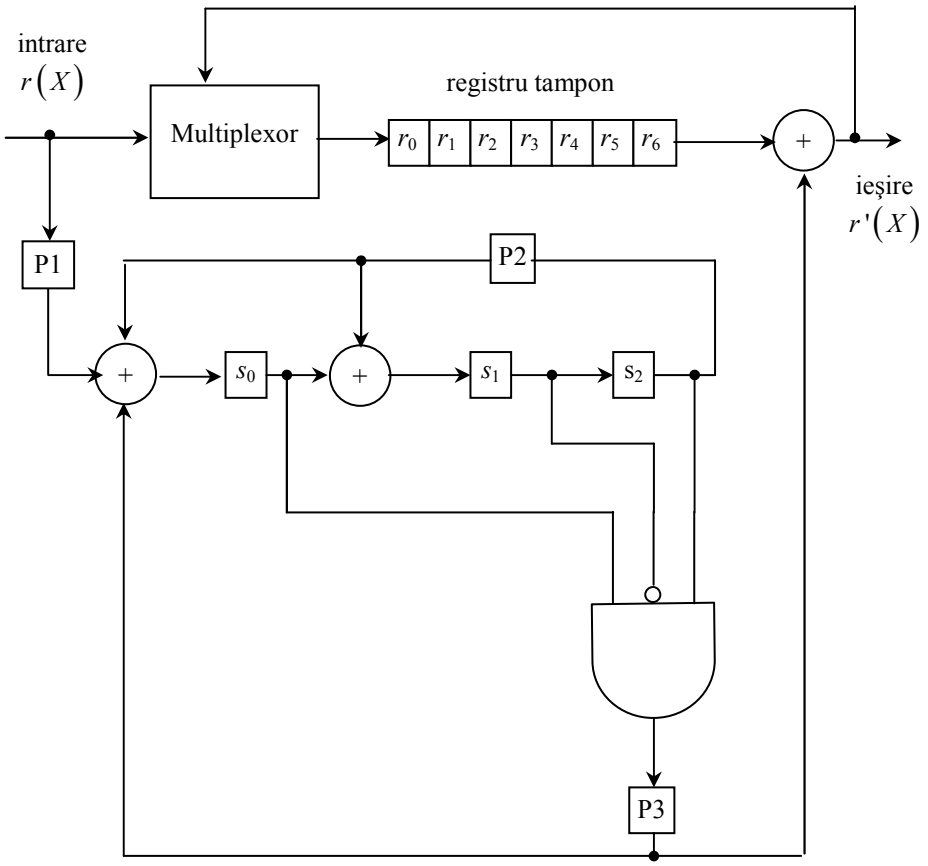


Fig. 7.10. Decodor Maggitt pentru codul ciclic (7,4) generat de $g(X) = 1+X+X^3$.

Procesul de decodare este ilustrat în fig. 7.11. Cu titlu de exemplu, să presupunem că vectorul de cod emis este $\mathbf{v} = (1\ 0\ 0\ 1\ 0\ 1\ 1)$, dar se recepționează $\mathbf{r} = (1\ 0\ 1\ 1\ 0\ 1\ 1)$. În expresie polinomială, $v(X) = 1 + X^3 + X^5 + X^6$ şi $r(X) = 1 + X^2 + X^3 + X^5 + X^6$. Când întregul vector de recepție va fi fost deplasat în registrele sindrom şi tampon, registrul sindrom va conține (0 0 1). În fig. 7.11, se arată conținutul registrului sindrom şi al celui tampon după fiecare deplasare. De asemenea, se indică printr-o săgeată poziția erorii după fiecare deplasare. Vedem că, după încă patru deplasări, conținutul registrului sindrom este (1 0 1) şi că bitul eronat r_2 este cel care urmează să părăsească registrul tampon.

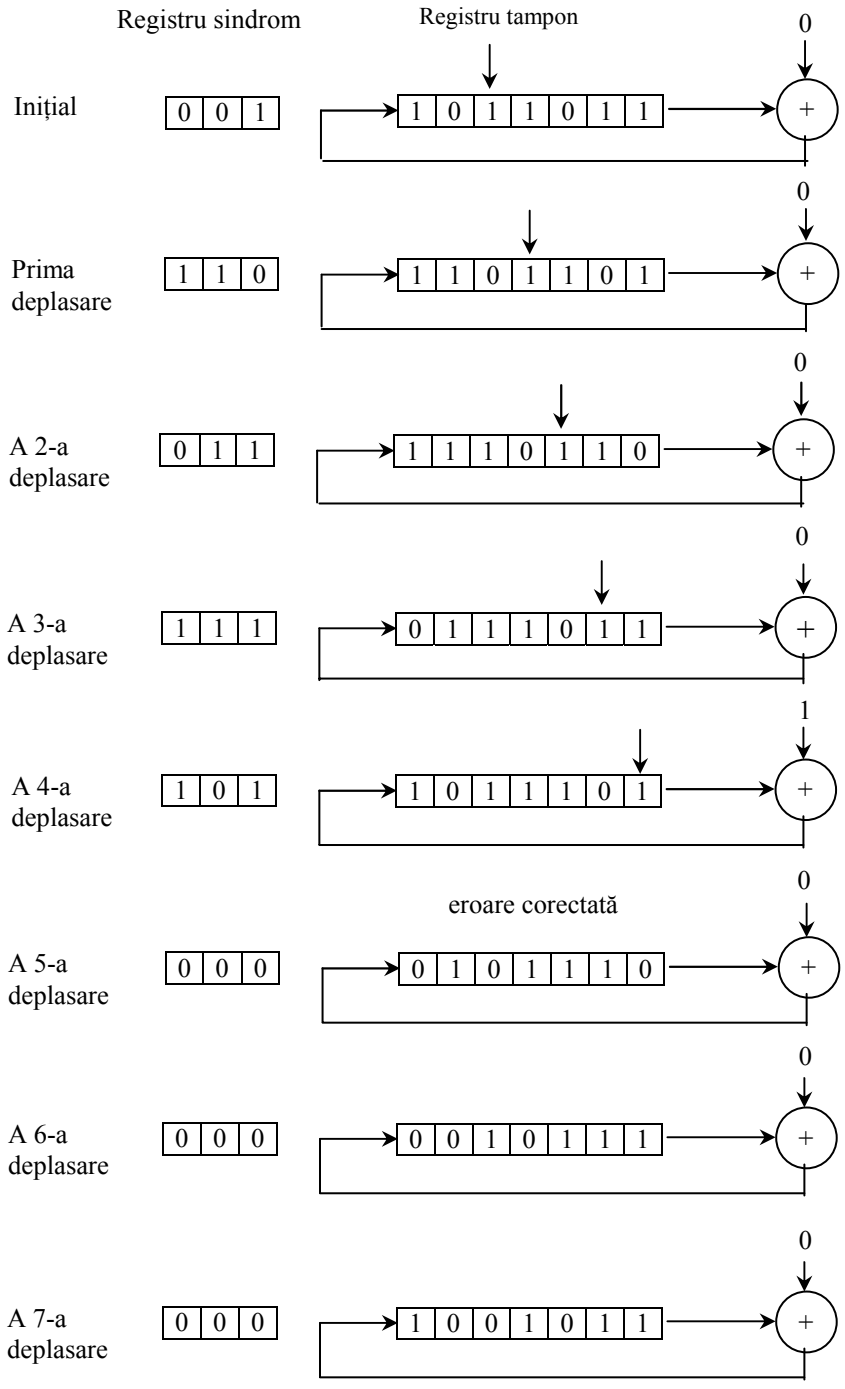


Fig. 7.11. Procesul de corecție a erorii pentru circuitul arătat în fig. 7.10.

7.6. DECODAREA CODURILOR CICLICE PRIN METODA CAPCANEI

Revăzând ecuația (7.47), ne reamintim că sindromul $s(X)$ este restul împărțirii polinomului de eroare $e(X)$ la polinomul generator $g(X)$. Să presupunem că erorile nu apar decât în cele $(n–k)$ locuri superioare, $X^k, X^{k+1}, \cdots, X^{n-1}$, ale lui $r(X)$, adică:

$$e(X) = e_k X^k + e_{k+1} X^{+1} + \cdots + e_{n-1} X^{n-1} \tag{7.51}$$

Dacă se deplasează polinomul de recepție $r(X)$ ciclic de $(n–k)$ ori la dreapta, erorile se vor limita la cele $(n–k)$ locuri inferioare, $X^0, X^1, \cdots, X^{n-r-1}$, ale lui $r^{(n-k)}(X)$. Polinomul de eroare corespunzător va fi:

$$e^{(n-k)}(X) = e_k + e_{k+1} X + \cdots + e_{n-1} X^{n-k-1} \tag{7.52}$$

Întrucât sindromul $s^{(n-k)}(X)$ al lui $r^{(n-k)}(X)$ este egal cu restul împărțirii lui $e^{(n-k)}(X)$ la $g(X)$ şi deoarece $e^{(n-k)}(X)$ este mai mic decât $(n–k)$, obținem următoarea egalitate:

$$s^{(n-k)}(X) = e^{(n-k)}(X) = e_k + e_{k+1} X + \cdots + e_{n-1} X^{n-k-1} \tag{7.53}$$

Înmulțind acum $s^{(n-k)}(X)$ cu X^k, obținem:

$$\begin{aligned} X^k s^{(n-k)}(X) &= e(X) \\ &= e_k X^k + e_{k+1} X^{k+1} + \cdots + e_{n-1} X^{n-1} \end{aligned} \tag{7.54}$$

Ecuația (7.54) ne spune că, dacă erorile se limitează la cele $(n–k)$ locuri superioare ale polinomului de recepție $r(X)$, polinomul de eroare $e(X)$ este identic cu $X^k s^{(n-k)}(X)$, unde $s^{(n-k)}(X)$ este sindromul lui $r^{(n-k)}(X)$, cea de a $(n–k)$-a deplasare ciclică a lui $r(X)$. Dacă apare acest eveniment, calculăm $s^{(n-k)}(X)$ şi adunăm $X^k s^{(n-k)}(X)$ la $r(X)$. Rezultatul acestei operații va fi cuvântul de cod emis.

Să presupunem că erorile, deşi nu se limitează la cele $(n–k)$ locuri superioare, se limitează totuşi la $(n–k)$ locuri consecutive, să spunem $X^i, X^{i+1}, \cdots, X^{n-k+i-1}$ ale lui $r(X)$, inclusiv cazul în care erorile apar la extremitățile vectorului de recepție. Dacă se deplasează ciclic $r(X)$ de $(n–i)$ ori la dreapta, erorile se vor limita la cele $(n–k)$ locuri inferioare ale lui

$r^{(n-i)}(X)$, iar polinomul de eroare va fi identic cu $X^i s^{(n-i)}(X)$, unde $s^{(n-i)}(X)$ este sindromul lui $r^{(n-i)}(X)$.

Să presupunem acum că deplasăm polinomul de recepție $r(X)$ în registrul sindrom pe la extremitatea din dreapta. Deplasarea lui $r(X)$ în registrul sindrom pe la extremitatea din dreapta este echivalentă cu înmulțirea prealabilă a lui $r(X)$ cu X^{n-k}. După ce întreg polinomul de recepție $r(X)$ va fi fost deplasat în registrul sindrom, conținutul acestuia va forma sindromul $s^{(n-k)}(X)$ al lui $r^{(n-k)}(X)$. Dacă erorile se limitează la (n–k) poziții superioare, $X^k, X^{k+1}, \cdots, X^{n-1}$, ale lui $r(X)$, ele sunt identice cu $s^{(n-k)}(X)$. Dacă, însă, erorile se limitează la (n–k) poziții consecutive (inclusiv legate la extremități), altele decât cele (n–k) poziții superioare ale lui $r(X)$, după ce întregul polinom de recepție $r(X)$ va fi fost deplasat în registrul sindrom, registrul sindrom trebuie deplasat de un anumit număr de ori mai înainte ca el să aibă un conținut identic cu biții de eroare. Această deplasare a registrului sindrom până când conținutul său devine identic cu biții de eroare se numeşte „prinderea în capcană a erorilor“. Dacă erorile se limitează la (n–k) poziții consecutive ale lui $r(X)$ şi dacă putem detecta când sunt „prinse în capcană“ erorile în registrul sindrom, corecția erorilor se poate efectua adunând conținutul registrului sindrom la biții de recepție din cele (n–k) poziții corespunzătoare.

Să presupunem că se utilizează pentru transmiterea fiabilă a datelor un cod ciclic corector de t erori. Pentru a detecta evenimentul constând în prinderea erorilor în registrul sindrom, putem testa *ponderea* sindromului după fiecare deplasare a registrului sindrom. Îndată ce ponderea sindromului va fi devenit t sau mai mică, presupunem că erorile au fost prinse în registrul sindrom. În cazul în care numărul erorilor conținute în polinomul de recepție $r(X)$ este t sau mai mic şi ele se limitează la (n–k) poziții consecutive, erorile vor fi fost prinse în registrul sindrom dacă şi numai dacă ponderea sindromului din registru va fi devenit t sau mai mică. Într-adevăr, un polinom de eroare $e(X)$ cu t erori sau mai puține ce se limitează la (n–k) poziții consecutive trebuie să fie de forma $e(X) = X^j B(X)$, unde $B(X)$ are t termeni sau mai puțini şi gradul (n–k–1) sau mai mic. Împărțind $e(X)$ la polinomul generator $g(X)$, avem:

$$X^j B(X) = a(X)g(X) + s(X) \tag{7.55}$$

În (7.55), $s(X)$ este sindromul lui $X^j B(X)$. Întrucât $s(X) + X^j B(X)$ este un multiplu de $g(X)$, el este un polinom de cod. Sindromul $s(X)$ nu poate avea pondere t sau mai mică decât dacă $s(X) = X^j B(X)$. Într-adevăr, în ipoteza

că ponderea lui $s(X)$ este t sau mai mică şi $s(X) \neq X^j B(X)$, $s(X) + X^j B(X)$ este un polinom de cod diferit de zero cu pondere mai mică decât $(2t+1)$. Dar acest lucru este imposibil, căci un cod corector de t erori trebuie să aibă o pondere minimă de cel puţin $(2t+1)$. Desprindem concluzia că erorile vor fi prinse în registrul sindrom numai dacă ponderea sindromului a devenit t sau mai mică.

Schema de principiu a unui decodor bazat pe conceptul de prindere în capcană a erorilor este dată de fig. 7.12.

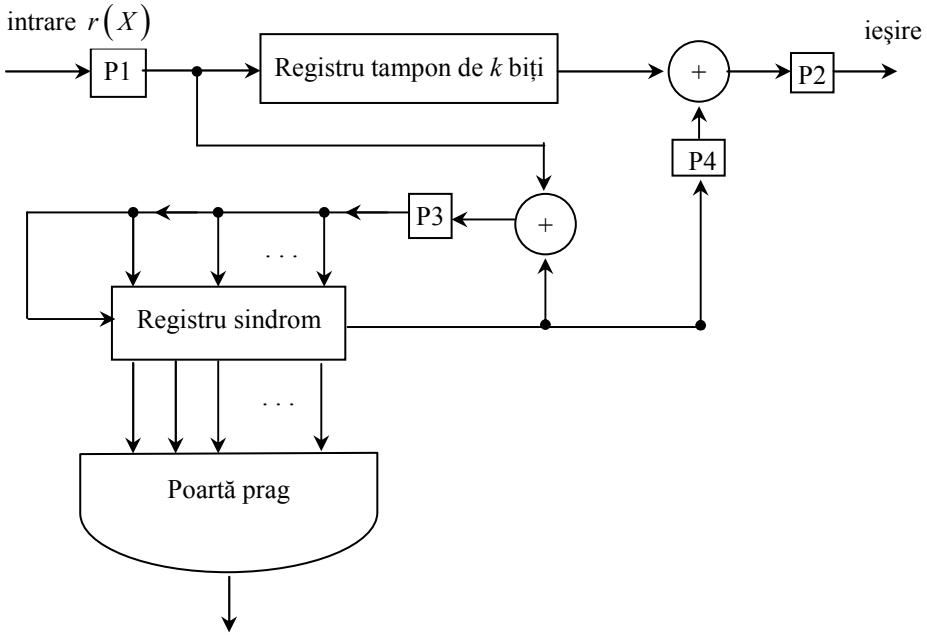


Fig. 7.12. Decodor cu prindere în capcană a erorilor.

Decodarea se face în următorii paşi:

Pasul 1. Cu porţile P1 şi P3 deschise şi cu porţile P2 şi P4 închise, polinomul de recepţie $r(X)$ este făcut să se deplaseze în registrul tampon şi, simultan, în registrul sindrom. Deoarece nu ne interesează decât recuperarea celor k biţi de informaţie, registrul tampon n-are de memorat decât aceşti k biţi de informaţie.

Pasul 2. Îndată ce întregul polinom de recepţie $r(X)$ va fi fost deplasat în registrul sindrom, se testează ponderea sindromului din registru cu o poartă logică de tip *prag* cu $(n-k)$ intrări a cărei ieşire este în 1 logic dacă t din intrările

sale sau mai puține sunt în 1 logic; în caz contrar, ieşirea este în 0 logic. Pot fi două situații:

1. Dacă ponderea sindromului este t sau mai mică, biții din registrul sindrom coincid cu biții de eroare din cele (n–k) poziții superioare $X^k, X^{k+1}, \cdots, X^{n-1}$ ale lui $r(X)$. Acum, se deschid porțile P2 şi P4 şi se închid porțile P1 şi P3. Vectorul de recepție este extras din registrul tampon bit cu bit, fiind corectat de biții de eroare ce se deplasează afară din registrul sindrom.
2. Dacă însă ponderea sindromului este mai mare decât t, erorile <u>nu</u> se limitează la cele (n–k) poziții superioare ale lui $r(X)$ şi nu au fost, deci, prinse în registrul sindrom. Se merge la pasul 3.

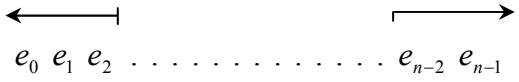
Pasul 3. Se deplasează ciclic o dată registrul sindrom cu poarta P3 deschisă, celelalte porți fiind închise. Se testează ponderea noului sindrom.

1. Dacă ponderea este t sau mai mică, erorile se limitează la pozițiile $X^{k-1}, X^k, \cdots, X^{n-2}$ ale lui $r(X)$ iar conținutul registrului sindrom coincide cu erorile din aceste poziții. Fiindcă primul bit recepționat r_{n-1} nu este afectat de eroare, el este extras din registrul tampon cu poarta P2 deschisă. Îndată ce r_{n-1} va fi fost citit din registrul tampon, poarta P4 se deschide iar poarta P3 se închide. Biții din registrul sindrom sunt extraşi din el şi utilizați pentru a corecta următorii (n–k) biți de recepție care ies din registrul tampon.
2. Dacă ponderea sindromului este mai mare decât t, registrul sindrom mai este făcut să se deplaseze o dată cu poarta P3 deschisă.

Pasul 4. Registrul sindrom este făcut să se deplaseze continuu până când ponderea conținutului său scade la t sau la mai puțin. Dacă după i deplasări, pentru $1 \le i \le k$, ponderea ajunge la t sau la mai puțin, primii i biți de recepție, $r_{n-i}, r_{n-i+1}, \cdots, r_{n-1}$, din registrul tampon sunt neafectați de eroare iar conținutul registrului sindrom coincide cu erorile de la pozițiile lui $X^{k-i}, X^{k-i+1}, \cdots, X^{n-i-1}$. Îndată ce cei i biți de recepție neafectați de eroare vor fi fost extraşi din registrul tampon,

biții din registrul sindrom sunt extraşi şi utilizați pentru a corecta următorii (*n–k*) biți de recepție ce vor ieşi din registrul tampon. Atunci când cei *k* biți de informație recepționați vor fi fost extraşi din registrul tampon şi corectați, poarta P2 se va deschide. Orice biți diferiți de zero rămaşi în registrul sindrom sunt erori din partea de control a lui $r(X)$ şi vor fi ignorați.

Pasul 5. Dacă ponderea sindromului nu scade niciodată la *t* sau la mai puțin în timpul în care registrul sindrom a fost făcut de *k* ori să se deplaseze, atunci fie a apărut un polinom de eroare cu erorile la (*n–k*) poziții consecutive în jurul extremității



fie a apărut o combinație necorectabilă de erori. Continuăm să deplasăm registrul sindrom. Să presupunem că ponderea conținutului său devine *t* sau mai mică după (*k*+*l*) deplasări, cu $1 \le l \le n-k$. În acest caz, erorile se limitează la cele (*n–k*) poziții consecutive în jurul extremității, $X^{n-l}, X^{n-l+1}, \cdots, X^{n-1}, X^0, X^1, \cdots, X^{n-k-l-1}$, ale lui $r(X)$. Cei *l* biți din cele mai de la stânga *l* etaje ale registrului sindrom corespund cu erorile din cele *l* poziții superioare $X^{n-l}, X^{n-l+1}, \cdots, X^{n+1}$ ale lui $r(X)$. Întrucât nu ne interesează erorile din cele (*n–k–l*) poziții de control, deplasăm registrul de (*n–k–l*) ori cu toate porțile închise. Acum, cele *l* erori din pozițiile $X^{n-l}, X^{n-l+1}, \cdots, X^{n-1}$ ale lui $r(X)$ sunt conținute în cele mai de la dreapta *l* etaje ale registrului sindrom. Cu porțile P2 şi P4 deschise şi cu porțile P1 şi P3 închise, biții de recepție din registrul tampon sunt extraşi şi corectați de biții de eroare corespunzători ce se deplasează afară din registrul sindrom.

Dacă ponderea sindromului nu ajunge niciodată egală cu *t* sau mai mică în timpul în care registrul sindrom va fi fost făcut să se deplaseze în total de *n* ori, fie a apărut o combinație de erori nedetectabilă, fie erorile nu se limitează la (*n–k*) poziții consecutive. În ambele cazuri, erorile sunt detectate.