CAPITOLUL 4

CODAREA SURSELOR DISCRETE DE INFORMAȚIE PE CANALE PERTURBATE

4.1. Detecția şi corecția erorilor

În cazul transmiterii informației la distanțe relativ mari şi/sau cu puteri relativ mici, efectul perturbațiilor nu mai poate fi neglijat. În acest caz succesiunea simbolurilor recepționate nu va mai coincide cu succesiunea simbolurilor transmise.

În cele ce urmează, se va considera că pe canalul de transmisiuni se introduc numai *perturbații aditive*, adică, datorită perturbațiilor, un simbol transmis se poate transforma în oricare din simbolurile alfabetului codului folosit. În cazul cel mai frecvent întâlnit în aplicații, alfabetul codului este definit prin mulțimea X={0,1}. Dacă se transmite unul din cele două simboluri pe un canal perturbat, fie acesta $x \in X$, atunci, în cazul recepționării eronate, va rezulta simbolul $\mathbf{x}' = \mathbf{x} \oplus \mathbf{1}$, unde $\oplus$ reprezintă simbolul sumei modulo 2.

Într-adevăr, dacă s-a transmis x=0, se va recepționa $\mathbf{x}' = \mathbf{0} \oplus \mathbf{1} = \mathbf{1}$, iar dacă s-a transmis x=1, se va recepționa $\mathbf{x}' = \mathbf{1} \oplus \mathbf{1} = \mathbf{0}$.

În general, numărul de erori în secvența recepționată depinde de calitatea canalului şi de viteza de transmitere. Cu cât calitatea canalului va fi mai slabă şi viteza de transmitere mai mare, cu atât numărul de erori în secvența recepționată va fi mai mare.

După modul cum perturbațiile de pe canalele de transmisiuni afectează simbolurile din cuvântul transmis, se disting:

a) canale la care fiecare simbol poate fi perturbat *independent* şi

b) canale în care durata perturbațiilor este mai mare decât durata corespunzătoare unui simbol din alfabetul codului, caz în care erorile apar grupate, determinând aşa numitele *pachete de erori*.

Se va analiza mai întâi cazul în care fiecare simbol poate fi perturbat independent.

În transmisiile pe canale perturbate se urmăresc două probleme principale:

a) detecția erorilor;

b) corecția erorilor.

În cazul detecției erorilor se pune problema realizării unei astfel de transmisii, încât la recepție să existe posibilitatea de a se decide dacă ceea ce s-a recepționat este corect sau nu, fără pretenția localizării eventualelor erori din secvența recepționată.

În cazul corecției erorilor se pune problema realizării unei astfel de transmisii, încât la recepție să existe posibilitatea corecției automate a eventualelor erori din secvența recepționată.

Deoarece operația de detecție a erorilor este mai simplă decât cea de corecție a acestora, echipamentul folosit în cazul detecției este mai puțin sofisticat. În cazul când este posibilă numai detecția erorilor, se va cere retransmiterea secvenței ori de câte ori se detectează erori. În felul acesta, simplitatea echipamentului folosit în cazul detecției erorilor este plătit de necesitatea unui canal cu dublu sens, (într-un sens realizându-se transmisia, iar în sens invers cerându-se eventualele necesități de retransmitere), precum şi un timp mai mare până la realizarea unei recepții corecte. Sunt situații când numai detectarea erorilor este complet nesatisfăcătoare. Astfel de situații apar când informația este stocată pe un suport fizic (banda magnetică, dischetă, disc etc). Dacă pe anumite porțiuni suportul fizic este deteriorat (de exemplu, apar zgârieturi), la o eventuală "citire" a informației, deşi se detectează erori, orice cerere de retransmitere (citire repetată) nu va conduce la reconstituirea informației corecte.

În transmisiile curente se realizează corecția unui număr de e erori sau mai puține care apar cel mai frecvent pe canalul de transmisiuni şi detecția unui număr mai mare de e erori.

Pentru detecția şi/sau corecția erorilor se folosesc numeroase tipuri de coduri. O primă clasificare a lor se poate face funcție de structura cuvintelor de cod. Din acest punct de vedere, codurile se împart în coduri bloc şi nonbloc (recurente sau convoluționale).

În cazul codurilor bloc, toate cuvintele au aceeaşi lungime, de exemplu, n simboluri 0 şi/sau 1 pentru codurile binare.

În cazul codurilor nonbloc, cuvintele nu mai sunt reprezentate prin blocuri formate din n simboluri, transmiterea informației realizându-se sub forma unei succesiuni continue de simboluri.

4.2. Relații deterministe între numărul de erori detectabile sau corectabile şi distanța Hamming

Fie X={0,1} alfabetul codului. În cazul codurilor bloc, considerând că toate cuvintele au lungimea n, înseamnă că se pot întocmi 2^n cuvinte distincte. Mulțimea acestora se va nota cu $\mathbf{M}_{2^n}^{n}$.

Fie x şi y două cuvinte din această mulțime, de forma $x \rightarrow a_1a_2... a_n$ şi $y \rightarrow b_1b_2... b_n$, cu $a_i, b_i \in X$, $\mathbf{i = \overline{1,n}}$.

Prin definiție, se numeşte *distanță Hamming* între cele două cuvinte numărul de necoincidențe dintre acestea.

Dacă se notează cu d(x,y) distanța Hamming dintre cele două cuvinte, conform definiției, se poate scrie:

$$\mathbf{d(x,y)=\sum_{i=1}^{n}(a_i \oplus b_i)} \tag{4.1}$$

Aşa, de exemplu, dacă x→110011 şi y→101010, numărul de necoincidenţe este egal cu 3, iar cu relaţia (4.1) rezultă

$$\mathbf{d(x,y)=(1\oplus 1)+(1\oplus 0)+(0\oplus 1)+(0\oplus 0)+(1\oplus 1)+(1\oplus 0)=3}.$$

Se poate verifica uşor că distanţa Hamming astfel definită, respectă proprietăţile oricărei metrici, adică:

$$\mathbf{d(x,y)\geq 0} \tag{4.2}$$

$$\mathbf{d(x,x)=0} \tag{4.3}$$

$$\mathbf{d(x,y)\leq d(x,z)+d(z,y),\ \ (\forall)\ x,y,z\in M_{2^n}^n} \tag{4.4}$$

Dacă pe un canal de transmisiuni perturbaţiile determină e erori, atunci, dacă se transmite cuvântul $\mathbf{x\in M_{2^n}^n}$, se va recepţiona cuvântul $\mathbf{y\in M_{2^n}^n}$, care se află la distanţa Hamming egală cu e de cuvântul transmis.

Se pune problema cum trebuie întocmite *cuvintele de cod*, astfel încât la recepţie să fie posibilă fie detecţia erorilor, fie corecţia acestora. Pentru aceasta, se presupune o sursă discretă de informaţie ce poate furniza mesajele din mulţimea $S=\{s_0, s_1, ..., s_{N-1}\}$. Dacă mesajelor li se ataşează cuvinte de cod distincte de aceeaşi lungime, atunci numărul k al simbolurilor binare din fiecare cuvânt de cod se determină cu relaţia:

$$\mathbf{2^k > N} \tag{4.5}$$

În relaţia (4.5) se alege numărul întreg pozitiv k, cel mai mic, care o satisface. Cele k simboluri se numesc informaţionale.

Mulţimea cuvintelor astfel formate se va nota cu $\mathbf{M_{2^k}^k}$. Dacă se transmite un cuvânt $\mathbf{x\in M_{2^k}^k}$, se va recepţiona pe un canal perturbat un cuvânt $\mathbf{y\in M_{2^k}^k}$, neputându-se realiza detecţia şi cu atât mai puţin corecţia erorilor.

Dacă perturbaţiile de pe canalul de transmisiuni pot cauza cel mult e erori, pentru a se putea realiza detecţia acestora, fiecare cuvânt de cod se va forma dintr-un număr n>k simboluri binare, astfel încât din mulţimea de 2^n cuvinte posibile distincte, de lungime n, să se aleagă drept cuvinte de cod, un număr de 2^k, astfel încât distanţa Hamming dintre acestea să satisfacă relaţia:

$$\mathbf{d(x,y)\geq e+1} \tag{4.6}$$

Mulţimea cuvintelor de cod care satisface relaţia (4.6) se va nota cu $\mathbf{M_{2^k}^n}$.

Fie $\mathbf{x\in M_{2^k}^n}$ un cuvânt de cod care se transmite pe un canal pe care pot să apară cel mult e erori. Dacă cuvântul recepţionat este $x^{/}$, atunci acesta nu va aparţine mulţimii $\mathbf{M_{2^k}^n}$.

Într-adevăr, dacă cuvântul x' ar aparține mulțimii $\mathbf{M}_{2^k}^n$, atunci, conform relației (4.6), se poate scrie relația:

$$\mathbf{d(x, x') \geq e + 1} \tag{4.7}$$

Pe de altă parte, deoarece pe canalul de transmisiuni pot să apară cel mult e erori, rezultă:

$$\mathbf{d(x, x') \leq e} \tag{4.8}$$

Comparând (4.7) cu (4.8) rezultă o contradiție, deci presupunerea că x' aparține mulțimii cuvintelor de cod $\mathbf{M}_{2^k}^n$ este falsă.

La recepție se cunoaşte mulțimea cuvintelor de cod $\mathbf{M}_{2^k}^n$, dar nu se ştie care cuvânt de cod a fost transmis. Dacă se recepționează un cuvânt ce aparține mulțimii $\mathbf{M}_{2^k}^n$, se decide că acesta nu conține erori, iar dacă nu aparține acestei mulțimi, se decide că în cuvântul recepționat s-au introdus erori, deci se realizează detecția erorilor. Deoarece distațele Hamming dintre cuvintele de cod satisfac relația (4.6), iar pe canal pot să apară cel mult e erori, rezultă că prin eronarea unui cuvânt de cod, acesta nu se poate transforma în alt cuvânt de cod.

Relația (4.6) se numeşte *relație deterministă între numărul de erori detectabile şi distanța Hamming*. Distanța minimă, D_{min}, dintre cuvintele de cod pentru detecția a e erori sau mai puține, conform relației (4.6), este:

$$D_{min} = e + 1 \tag{4.9}$$

Pentru a stabili o relație deterministă între numărul de erori corectabile şi distanța Hamming, se presupune că transmisia se realizează pe un canal binar simetric cu graful reprezentat în figura 4.1.

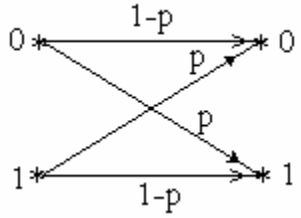

Fig.4.1. Graful unui canal binar simetric

Se reamintește că p reprezintă probabilitatea de recepționare eronată a unui simbol binar, iar 1-p probabilitatea recepționării corecte.

Dacă se transmite pe un astfel de canal un cuvânt de cod $\mathbf{x} \in \mathbf{M}_{2^k}^n$, la recepție, în medie, un număr de np simboluri vor fi eronate, în timp ce, în medie, un număr de n-np simboluri vor fi recepționate corect. Rezultă atunci că probabilitatea de a se recepționa cuvântul x' pe un canal binar simetric, caracterizat de parametrul p, dacă s-a transmis cuvântul de cod x, se poate determina cu relația:

$$\mathbf{p(x'|x) = p^{np} \cdot (1-p)^{n-np}} \tag{4.10}$$

Pe de altă parte, np=d(x,x$^{'}$) şi deci relaţia (4.10) se poate scrie, echivalent, sub forma:

$$p(x^{'}|x) = p^{d(x,x^{'})}.(1-p)^{n-d(x,x^{'})} \tag{4.11}$$

Pentru toate canalele reale, binar simetrice, este îndeplinită condiţia:

$$0 \le p < \frac{1}{2} \tag{4.12}$$

Dacă relaţia (4.12) este satisfăcută, funcţia p(x$^{'}$|x) dată de relaţia (4.11) este monoton descrescătoare. Ţinând cont de acest fapt, rezultă că la recepţionarea cuvântului x$^{'}$ se va decide că s-a transmis acel cuvânt de cod care se află la distanţa Hamming minimă de cuvântul recepţionat. Dacă pe canalul de transmisiuni pot să apară cel mult e erori, pentru a exista un singur cuvânt de cod la distanţa Hamming minimă de cuvântul recepţionat, cuvintele de cod se aleg astfel încât să fie satisfăcută relaţia:

$$d(x,y) \ge 2e+1, (\forall)x,y \in M_{2^k}^n \tag{4.13}$$

Într-adevăr, dacă se transmite cuvântul de cod x pe un canal pe care pot să apară e erori, cuvântul recepţionat x$^{'}$ se va afla la distanţa Hamming faţă de cuvântul de cod transmis ce satisface relaţia:

$$d(x,x^{'}) = e \tag{4.14}$$

Se presupune că ar mai exista un cuvânt de cod $y \in M_{2^k}^n$ la aceeaşi distanţă Hamming faţă de acelaşi cuvânt recepţionat x$^{'}$, adică:

$$d(x^{'},y) = e \tag{4.15}$$

Conform relaţiei (4.4), se poate scrie:

$$d(x,y) \le d(x,x^{'}) + d(x^{'},y), \tag{4.16}$$

sau, ţinând cont de (4.14), (4.15) şi (4.16), rezultă:

$$d(x,y) \le 2e \tag{4.17}$$

Comparând relaţiile (4.13) şi (4.17), rezultă o contradicţie, deci presupunerea existenţei a două cuvinte de cod x şi y la aceeaşi distanţă Hamming de cuvântul recepţionat este falsă.

În concluzie, dacă transmisia se realizează pe un canal binar simetric (cu p<1/2) şi pe acesta apar cel mult e erori, atunci, alegându-se cuvintele de cod astfel încât să fie respectată relaţia (4.13), la recepţionarea unui cuvânt de cod se va decide că s-a transmis acel cuvânt de cod care este la distanţa Hamming minimă de cuvântul recepţionat. Comparând cele două cuvinte, se pot localiza poziţiile erorilor şi, deci, se poate realiza corecţia acestora.

Relaţia (4.13) se numeşte *relaţie determinstă între numărul de erori corectabile şi distanţa Hamming*. Distanţa minimă, D_{min}, dintre cuvintele de cod pentru corecţia a e erori sau mai puţine, conform relaţiei (4.13), este

$$D_{\min} = 2e + 1 \qquad \textbf{(4.18)}$$

Din analiza efectuată se constată că atât în cazul detecției erorilor, cât şi în cazul corecției acestora, la cele k *simboluri informaționale* vor trebui adăugate m simboluri, numite de control, astfel încât, dacă lungimea cuvintelor de cod este n, atunci:

$$\mathbf{n = m + k} \qquad \textbf{(4.19)}$$

Numărul simbolurilor de control ce trebuie adăugate la simbolurile informaționale depinde de numărul de erori care trebuie detectate, respectiv de numărul de erori care urmează a fi corectate.

În cazul detecției a e erori sau mai puține se va adăuga un număr de simboluri de control, astfel încât din cele 2^n cuvinte distincte de lungime n, să se poată alege 2^k cuvinte de cod între care distanța Hamming minimă să respecte relația (4.9).

În cazul corecției a e erori sau mai puține se va adăuga un număr de simboluri de control, astfel încât din cele 2^n cuvinte distincte de lungime n, să se poată alege 2^k cuvinte de cod, între care distanța Hamming minimă să respecte relația (4.18).

4.3. Definirea matricei de control şi generatoare în cazul codurilor bloc, liniare, binare

Pentru o scriere mai compactă, în continuare, cuvintele de cod se vor scrie sub formă matriceală, adică:

$$\mathbf{[v] = [a_1 a_2 \ldots a_n],\ a_i \in \{0,1\},} \qquad \textbf{(4.20)}$$

Notația, [v] nu este întâmplătoare, deoarece, aşa cum se va arăta ulterior, mulțimea cuvintelor de cod formează un subspațiu vectorial.

La emisie, instalația numită *codor* determină cele m simboluri de control, cunoscute fiind cele k simboluri de informație. Pentru simplificarea implementării codorului, simbolurile de control se determină prin combinații liniare ale simbolurilor informaționale, formându-se astfel codurile bloc *liniare*.

Determinarea simbolurilor de control, în cazul codurilor bloc, liniare, binare se poate efectua cel mai simplu din următorul sistem de ecuații liniare:

$$\left.\begin{aligned} &\mathbf{h_{11}a_1 \oplus h_{12}a_2 \oplus \ldots \oplus h_{1n}a_n = 0} \\ &\mathbf{h_{21}a_1 \oplus h_{22}a_2 \oplus \ldots \oplus h_{2n}a_n = 0} \\ &\mathbf{- - - - - - - - - - - - - - - - - -} \\ &\mathbf{h_{m1}a_1 \oplus h_{m2}a_2 \oplus \ldots \oplus h_{mn}a_n = 0} \end{aligned}\right\} \qquad \textbf{(4.21)}$$

unde $h_{ij} \in \{0,1\}$.

Dacă cele m ecuații din sistemul (4.21) sunt liniar independente, atunci se pot determina oricare m simboluri $a_i \in \{0,1\}$, dacă celelalte k=n-m simboluri binare sunt cunoscute.

Sistemul de ecuații (4.21) poate fi scris compact, sub formă matriceală, dacă se face notația:

$$\begin{bmatrix} h_{11}h_{12}...h_{1n} \\ h_{21}h_{22}...h_{2n} \\ - - - - - - \\ h_{m1}h_{m2}...h_{mn} \end{bmatrix} \underline{\underline{\text{not.}}} [H] \qquad (4.22)$$

Cu (4.20) şi (4.22) sistemul de ecuații (4.21) se poate scrie compact, sub forma:

$$[H][v]^T = [0] \qquad (4.23)$$

unde $[v]^T$ este transpusa matricei [v].

În relația (4.23) produsul celor două matrice se efectuează clasic, înlocuindu-se numai suma clasică cu suma modulo 2.

Matricea [H] definită cu relația (4.22) se numeşte *matrice de control*, aceasta având un număr de linii egal cu numărul simbolurilor de control şi un număr de coloane egal cu lungimea cuvintelor de cod.

Ecuația matriceală (4.23) fiind echivalentă cu un sistem de m ecuații liniar independente, rezultă că rangul matricei [H] este egal cu m. În aceste condiții înseamnă că prin transformări elementare asupra acestei matrice, aceasta poate fi

$$[H] = \begin{bmatrix} 10...0 & q_{11}q_{12}\cdots q_{1k} \\ 01...0 & q_{21}q_{22}\cdots q_{2k} \\ - - - - - - - - - - \\ 00...1 & q_{m1}q_{m2}\cdots q_{mk} \end{bmatrix} = [I_m Q] \qquad (4.24)$$

adusă totdeauna la forma:

unde $[I_m]$ este matricea unitate de ordinul m, iar $q_{ij} \in \{0,1\}$

Prin definiție, o matrice de forma:

$$[G] = \begin{bmatrix} g_{11}g_{12}\cdots g_{1n} \\ g_{21}g_{22}\cdots g_{2n} \\ - - - - - - \\ g_{k1}g_{k2}\cdots g_{kn} \end{bmatrix}, \quad g_{ij} \in \{0,1\} \qquad (4.25)$$

se numeşte *matrice generatoare* a codului bloc, liniar, dacă este satisfăcută relația:

$$[v] = [i_1 i_2 \ldots i_k][G], \qquad (4.26)$$

unde [v] este cuvântul de cod ce satisface relația (4.23), iar $i_j \in \{0,1\}$, $\mathbf{j = \overline{1,k}}$, sunt simbolurile de informație, cunoscute.

Produsul matriceal din relația (4.26) se efectuează clasic, înlocuindu-se numai suma clasică cu suma modulo 2. Deoarece din relația (4.26) trebuie să rezulte 2^k cuvinte de cod distincte, este necesar ca liniile matricei generatoare să fie liniar independente, adică rangul acesteia trebuie să fie egal cu k. În aceste condiții înseamnă că prin transformări elementare, aceasta poate fi adusă totdeauna la forma:

$$[G] = \begin{bmatrix} p_{11}p_{12}\dots p_{1m}\ 10\dots0 \\ p_{21}p_{22}\dots p_{2m}\ 01\dots0 \\ - - - - - - - - - - \\ p_{k1}p_{k2}\dots p_{km}\ 00\dots1 \end{bmatrix} = [PI_k] \qquad (4.27)$$

unde $[I_k]$ este matricea unitate de ordinul k, iar $p_{ij} \in \{0,1\}$.

Matricea generatoare are un număr de linii egal cu numărul simbolurilor de informație şi un număr de coloane egal cu lungimea cuvintelor de cod.

Prin definiție, un cod bloc liniar se numeşte *sistematic,* dacă simbolurile informaționale sunt plasate grupat fie la sfârşitul cuvântului de cod, fie la începutul acestuia.

Dacă matricea de control este adusă la forma (4.24) sau matricea generatoare la forma (4.27), codul obținut va fi sistematic, deoarece din relația (4.23), respectiv (4.26), vor rezulta cuvinte de cod de forma:

$$[v] = [c_1c_2\dots c_m\ i_1i_2\dots i_k] \qquad (4.28)$$

unde $c_j \in \{0,1\}$, $\mathbf{j = \overline{1,m}}$, sunt simbolurile de control, iar $i_l \in \{0,1\}$, $\mathbf{l = \overline{1,k}}$, simboluri informaționale.

Înlocuind (4.26) în relația (4.23), rezultă:

$$[H][G]^T[i_1i_2\dots i_k]^T = [0] \qquad (4.29)$$

Deoarece ecuația matriceală (4.29) este adevărată pentru orice matrice $[i_1i_2...i_k]$, rezultă următoarea legătură între matricea de control şi generatoare:

$$[H][G]^T = [0] \qquad (4.30)$$

Dacă matricele [H] şi [G] sunt aduse la formele (4.24), respectiv (4.27), atunci se poate uşor deduce una, dacă cealaltă este cunoscută.

Într-adevăr, înlocuind (4.24) şi (4.27) în (4.30), rezultă:

$$[Q] = [P]^T \qquad (4.31)$$

Transpunând ambii termeni ai relației (4.31), se poate scrie:

$$[P] = [Q]^T \qquad (4.32)$$

Cu relația (4.31) se poate deduce matricea de control, dacă este cunoscută matricea generatoare, iar cu relația (4.32) se poate deduce matricea generatoare, dacă este cunoscută matricea de control.

Liniile matricei de control sau generatoare pot fi considerate vectori cu n componente. Mulțimea tuturor vectorilor cu n componente ce pot lua valoarea 0 sau 1 este egală cu 2^n. Definindu-se pentru aceşti vectori operația de sumare modulo 2, de forma:

$$[v_1] \oplus [v_2] = [a_1 a_2 \dots a_n] \oplus [b_1 b_2 \dots b_n] = \\ = [a_1 \oplus b_1 \; a_2 \oplus b_2 \dots a_n \oplus b_n] \tag{4.33}$$

şi multiplicarea clasică cu scalarii 0, respectiv 1, din câmpul format din aceste două elemente în care suma se efectuează modulo 2, iar produsul în mod clasic, cele 2^n cuvinte posibile formează un spațiu vectorial n dimensional.

Din relația (4.26) rezultă că liniile matricei generatoare sunt cuvinte de cod. Dacă se notează aceste linii cu $[v_1]$, $[v_2]$,...,$[v_k]$, din relația (4.26) rezultă:

$$[v] = i_1[v_1] \oplus i_2[v_2] \oplus \dots \oplus i_k[v_k] \tag{4.34}$$

Conform relației (4.34) rezultă că orice cuvânt de cod este o combinație liniară a liniilor matricei generatoare. Pe de altă parte, mulțimea combinațiilor liniare a liniilor unei matrice determină *spațiul linie* al acesteia.

Cu această observație rezultă că un cuvânt este cuvânt de cod, dacă şi numai dacă se află în spațiu linie a matricei generatoare. În felul acesta, mulțimea cuvintelor de cod formează un subspațiu vectorial k dimensional. Din relația (4.23) rezultă că orice vector corespunzător unui cuvânt de cod este ortogonal pe vectorii corespunzători spațiului linie al matricei de control.

Prin definiție, se numeşte *spațiul nul* al unui subspațiu vectorial, mulțimea vectorilor ortogonali pe vectorii din subspațiul vectorial.

Prin definiție, spațiul nul al spațiului linie al unei matrice se numeşte *spațiul nul al acesteia*.

Cu această observație rezultă că mulțimea cuvintelor de cod aparține spațiului nul al matricei de control.

4.4. Definirea corectorilor cuvintelor recepționate în cazul codurilor bloc, liniare, binare

Fie

$$[v] = [a_1 a_2 \dots a_n], \quad a_i \in \{0,1\}, i = \overline{1,n} \tag{4.35}$$

cuvântul de cod transmis pe un canal pe care apar perturbații aditive şi fie

$$[v'] = [a'_1 a'_2 \dots a'_n], \quad a'_i \in \{0,1\}, i = \overline{1,n} \tag{4.36}$$

cuvântul recepționat.

Prin definiție, se numeşte *cuvânt eroare* suma modulo 2 dintre cuvântul de cod transmis şi cuvântul recepționat.

Notând cu [E] cuvântul eroare, conform definiției, se poate scrie:

$$[E] = [v] \oplus [v'] \tag{4.37}$$

Adunând modulo 2 cuvântul de cod [v] în ambii membri ai relației (4.37), rezultă:

$$[v'] = [v] \oplus [E] \tag{4.38}$$

Conform relației (4.38), perturbațiile P de pe canalul de transmisiuni C sunt sugerate în figura 4.2.

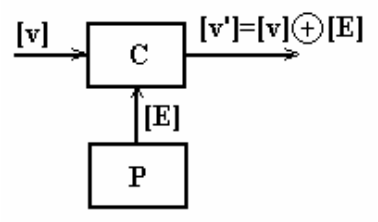

Fig.4.2. Reprezentarea unui canal cu perturbații aditive

Din această reprezentare rezultă că perturbațiile aditive care apar pe canalele de transmisiuni pot fi simulate prin generarea cuvintelor eroare, care se adună modulo 2 la cuvintele de cod.

Conform relației (4.37), rezultă că structura matriceală a cuvântului eroare este de forma:

$$[E] = [e_1 e_2 \dots e_n] \tag{4.39}$$

unde e_i=1, dacă pe poziția i s-a introdus eroare şi e_j=0, dacă pe poziția j nu s-a introdus eroare. Uneori, cuvântul eroare se scrie sub forma:

$$[E] = [\dots \alpha_i \dots \alpha_j \dots \alpha_k \dots], \tag{4.40}$$

unde $\alpha_i=\alpha_j=\alpha_k$=1, specificându-se astfel că pe pozițiile i,j şi k s-au introdus erori, restul elementelor fiind nule, specificându-se astfel că pe celelalte poziții nu s-au introdus erori.

Instalația de la emisie, numită codor, se implementează după relația:

$$[H][v]^T = [0] \tag{4.41}$$

Ecuația matriceală (4.41) este echivalentă cu un sistem de m ecuații liniar independente, din care se deduc cele m simboluri de control, cunoscute fiind cele k simboluri informaționale.

Instalația de la recepție, numită decodor, se implementează după relația:

$$[Z] = [H][v']^T \tag{4.42}$$

unde [Z] se numeşte *corectorul* cuvântului recepționat.

Ținând cont de structura matricei de control [H], definită prin relația (4.22) şi structura matriceală a cuvântului recepționat [v'], definită prin relația (4.36), rezultă din (4.42) că structura matriceală a corectorului este de forma:

$$[Z] = \begin{bmatrix} z_1 \\ z_2 \\ . \\ . \\ . \\ z_m \end{bmatrix}, \quad z_i \in \{0,1\}, \ i = \overline{1,m} \tag{4.43}$$

Dacă în relația (4.42) se înlocuieşte (4.38) şi se ține cont de (4.41), se poate scrie o relație echivalentă pentru calculul corectorului:

$$[Z] = [H][E]^T \tag{4.44}$$

Dacă corectorul calculat cu relația (4.42) rezultă diferit de matricea nulă, adică:

$$[Z] \neq [0], \tag{4.45}$$

se decide că s-a recepționat un cuvânt eronat, deoarece, dacă ar fi fost un cuvânt de cod, conform relației (4.41), ar fi trebuit să rezulte [Z]=[0].

În felul acesta, decodorul poate realiza detecția erorilor, dacă matricea de control se alege astfel încât să rezulte corectori diferiți de matricea nulă ori de câte ori se recepționează un cuvânt eronat.

Pentru corecția erorilor, matricea de control trebuie astfel întocmită, încât să rezulte corectori diferiți de matricea nulă şi, în plus, distincți pentru toate combinațiile posibile ale cuvintelor eroare generate pe canalul de transmisiuni. În felul acesta se poate stabili o bijecție între mulțimea cuvintelor eroare ce pot apărea pe un canal de transmisiuni şi mulțimea corectorilor calculați cu relația (4.44).

Calculând corectorul unui cuvânt recepționat, se identifică cuvântul eroare corespunzător şi, conform relației (4.37), cuvântul de cod transmis se deduce cu relația:

$$[v] = [v'] \oplus [E] \tag{4.46}$$

În cazul în care corectorul unui cuvânt recepționat este egal cu matricea nulă, se decide că în cuvântul recepționat nu sunt erori, deoarece în acest caz toate componentele cuvântului eroare fiind nule, conform relației (4.44), rezultă într-adevăr un corector cu toate elementele nule.

Trebuie subliniat faptul că produsul a două matrice [H] şi $[E]^T$ poate să conducă la un corector cu toate elementele nule, deşi $[E]^T$ nu are toate elementele nule. În acest caz se va decide că s-a recepționat un cuvânt fără erori, deşi cuvântul recepționat este eronat. Erorile care au fost introduse în acest caz pe canalul de transmisiuni au transformat cuvântul de cod transmis în alt cuvânt de cod.

Pentru eliminarea acestui neajuns, matricea de control trebuie astfel întocmită, încât, dacă pe canal pot să apară e sau mai puține erori, să rezulte cuvinte de cod, conform relației (4.41), cu distanța Hamming minimă între ele egală cu 2e+1. În felul acesta, sumarea modulo 2 a unui cuvânt eroare ce poate conține maxim e unități la cuvântul de cod transmis, nu va putea să-l transforme în alt cuvânt de cod şi, în plus,

la recepționarea unui cuvânt eronat, acesta se va afla la distanța Hamming minimă de un singur cuvânt de cod, adică vor rezulta corectori distincți, diferiți de matricea nulă, pentru cuvintele eroare ce pot să apară pe canalul de transmisiuni.

4.5. Relații între coloanele matricei de control pentru detecția erorilor

Aşa cum a rezultat din analiza efectuată în paragraful precedent, pentru detecția erorilor matricea de control trebuie astfel întocmită, încât să rezulte corectori diferiți de matricea nulă ori de câte ori se recepționează un cuvânt eronat.

Pentru a stabili relațiile între coloanele matricei de control în scopul detecției erorilor, se va considera pentru început cazul apariției unei erori pe o poziție oarecare în cuvântul de cod transmis.

Cuvântul eroare corespunzător acestei situații este de forma:

$$[\mathbf{E_i}] = [\ldots \alpha_i \ldots] \tag{4.47}$$

unde $\alpha_i = 1, (\forall) i = \overline{1, n}$, specificându-se că pe poziția i s-a introdus o eroare, restul elementelor din cuvântul eroare fiind nule, specificându-se astfel, că pe celelalte poziții nu s-au introdus erori.

În scopul simplificării scrierii, se notează cu $[h_1]$, $[h_2]$,...,$[h_n]$ coloanele matricei de control, adică, conform relației (4.22), se poate scrie:

$$[\mathbf{H}] = [\mathbf{h_1 h_2 \ldots h_n}] \tag{4.48}$$

Înlocuind (4.47) şi (4.48) în relația (4.44) şi notând cu $[Z_i]$ corectorul rezultant, se obține:

$$[\mathbf{Z_i}] = [\mathbf{h_i}], (\forall) i = \overline{1, n} \tag{4.49}$$

Pentru detecția acestei erori este necesar ca $[Z_i] \neq [0]$, adică:

$$[\mathbf{h_i}] \neq [0]; (\forall) i = \overline{1, n} \tag{4.50}$$

Cu alte cuvinte, rezultă că pentru detecția unei erori, indiferent de poziția pe care aceasta apare în cuvântul de cod transmis, este necesar ca matricea de control să fie astfel întocmită, încât toate coloanele acesteia să fie diferite de matricea nulă.

Se presupune, în continuare, posibilitatea apariției a două erori, caracterizate prin cuvântul eroare:

$$[\mathbf{E_{ij}}] = [\ldots \alpha_i \ldots \alpha_j \ldots], (\forall) i, j = \overline{1, n}, i \neq j \tag{4.51}$$

Înlocuind (4.48) şi (4.51) în relația (4.44) şi notând cu $[Z_{ij}]$ corectorul rezultant, se poate scrie:

$$[\mathbf{Z_{ij}}] = [\mathbf{h_i}] \oplus [\mathbf{h_j}], (\forall) i, j = \overline{1, n}, i \neq j \tag{4.52}$$

Pentru detecția celor două erori este necesar ca $[Z_{ij}] \neq [0]$, adică:

$$[h_i] \oplus [h_j] \neq [0], (\forall) i, j = \overline{1, n}, i \neq j \quad (4.53)$$

Cu alte cuvinte, rezultă că pentru detecția a două erori distincte matricea de control trebuie astfel întocmită, încât suma modulo 2 a oricăror două coloane distincte să fie diferită de matricea nulă. Reunind concluziile de mai sus, rezultă că pentru detecția a două erori sau mai puține matricea de control trebuie astfel întocmită, încât suma modulo 2 a oricăror două coloane sau mai puține să fie diferită de matricea nulă.

Raționând în mod analog, se poate conchide că, în general, pentru detecția a e erori sau mai puține este necesar ca matricea de control să fie astfel întocmită, încât suma modulo 2 a e coloane sau mai puține să fie diferită de matricea nulă.

Pe de altă parte, s-a demonstrat în §4.2 că pentru detecția a e erori sau mai puține, între cuvintele de cod trebuie să existe o distanță Hamming minimă egală cu e+1 (relația 4.9).

Se poate demonstra că, dacă această condiție este satisfăcută, atunci este necesar ca suma modulo 2 a oricăror e coloane distincte din matricea de control să fie diferită de matricea nulă.

Într-adevăr, fie două cuvinte de cod de forma:

$$[v_a] = [a_1 a_2 \dots a_n] \quad (4.54)$$

$$[v_b] = [b_1 b_2 \dots b_n] \quad (4.55)$$

pentru care sunt satisfăcute relațiile:

$$[H][v_a]^T = [0] \quad (4.56)$$

$$[H][v_b]^T = [0] \quad (4.57)$$

Distanța Hamming dintre aceste cuvinte se poate calcula cu relația:

$$d(v_a, v_b) = \sum_{i=1}^{n} (a_i \oplus b_i) \quad (4.58)$$

Conform relației generale (4.34), dacă $[v_a]$ şi $[v_b]$ sunt cuvinte de cod, atunci şi suma modulo 2 a acestora, determină un nou cuvânt de cod, adică, dacă

$$[v_a] \oplus [v_b] \underline{\underline{\text{not}}} [v_c] = [a_1 \oplus b_1, a_2 \oplus b_2, \dots, a_n \oplus b_n], \quad (4.59)$$

rezultă:

$$[H][v_c]^T = [0] \quad (4.60)$$

Comparând relația (4.58) cu (4.59), înseamnă că distanța Hamming dintre două cuvinte de cod este egală cu numărul de unități din alt cuvânt de cod.

Prin definiție, se numeşte *pondere* a unui cuvânt numărul de unități din cuvântul respectiv.

Cu această definiție se poate afirma că distanța Hamming minimă dintre cuvintele de cod este egală cu ponderea minimă a cuvintelor de cod, exceptând

cuvântul de cod de pondere nulă. Înseamnă atunci că, dacă distanța Hamming minimă dintre cuvintele de cod este e+1, ponderea minimă a cuvintelor de cod este e+1, exceptând cuvântul de cod format numai din zerouri (adică de pondere nulă).

Dacă [v] este un cuvânt de cod de pondere e+1, din relația:

$$[H][v]^T = [0] \tag{4.61}$$

rezultă că suma modulo 2 a e+1 coloane din matricea de control este nulă. Dacă ponderea e+1 este minimă, înseamnă că suma modulo 2 a e coloane din matricea de control trebuie să fie diferită de matricea nulă, c.c.t.d.

4.6. Relații între coloanele matricei de control pentru corecția erorilor

Aşa cum s-a arătat în §4.4, pentru corecția erorilor matricea de control trebuie astfel întocmită, încât să rezulte corectori diferiți de matricea nulă şi, în plus, distincți pentru toate combinațiile posibile ale cuvintelor eroare ce pot să apară pe canalul pe care se realizează transmisiunea.

Pentru a stabili relațiile între coloanele matricei de control în scopul corecției erorilor, se va considera pentru început cazul apariției unei erori pe pozițiile i, respectiv j, oarecare, cu i≠j. Când eroarea apare pe poziția i, corectorul $[Z_i]$ este dat de

$$[Z_j] = [h_j],\ (\forall) j = \overline{1,n},\ j \neq i \tag{4.62}$$

relația (4.49). Când eroarea apare pe poziția j≠i, corectorul $[Z_j]$ va fi de forma:

În scopul corecției acestei erori este necesar să fie îndeplinite condițiile:

$$\left.\begin{aligned} &[Z_i] \neq [0] \Leftrightarrow [h_i] \neq [0] \\ &[Z_j] \neq [0] \Leftrightarrow [h_j] \neq [0] \\ &[Z_i] \neq [Z_j] \Leftrightarrow [Z_i] \oplus [Z_j] \neq [0] \Leftrightarrow [h_i] \oplus [h_j] \neq [0] \end{aligned}\right\} \tag{4.63}$$

Condițiile (4.63) conduc la concluzia că, pentru corecția unei erori, matricea de control trebuie astfel întocmită, încât suma modulo doi a două coloane distincte sau mai puține să fie diferită de matricea nulă.

Se presupune în continuare posibilitatea apariției a două erori caracterizate prin cuvintele eroare:

$$[E_{ij}] = [\ldots \alpha_i \ldots \alpha_j \ldots] \tag{4.64}$$

cu $\alpha_i=\alpha_j=1$, i≠j şi

$$[E_{lk}] = [\ldots \alpha_l \ldots \alpha_k \ldots] \tag{4.65}$$

cu $\alpha_l=\alpha_k=1$, l≠k≠i≠j.

Corectorul corespunzător cuvântului eroare $[E_{ij}]$ se va calcula cu relația (4.52), în timp ce corectorul corespunzător cuvântului eroare $[E_{lk}]$ cu relația:

$$[Z_{lk}] = [h_l] \oplus [h_k] \tag{4.66}$$

Pentru corecția a două erori este necesar să fie îndeplinite condițiile:

$$\left.\begin{aligned} &[Z_{ij}] \neq [0] \Leftrightarrow [h_i] \oplus [h_j] \neq [0] \\ &[Z_{lk}] \neq [0] \Leftrightarrow [h_l] \oplus [h_k] \neq [0] \\ &[Z_{ij}] \neq [Z_{lk}] \Leftrightarrow [Z_{ij}] \oplus [Z_{lk}] \neq [0] \Leftrightarrow \\ &[h_i] \oplus [h_j] \oplus [h_l] \oplus [h_k] \neq [0] \end{aligned}\right\} \tag{4.67}$$

oricare ar fi $\mathbf{i, j, l, k = \overline{1, n}}$ şi i≠j≠l≠k.

Pentru corecția a două erori sau mai puține este necesar ca pe lângă relația (4.67) să fie îndeplinite condițiile:

$$\left.\begin{aligned} &[Z_{ij}] \neq [Z_i] \Leftrightarrow [h_j] \neq [0] \\ &[Z_{ij}] \neq [Z_j] \Leftrightarrow [h_i] \neq [0] \\ &[Z_{lk}] \neq [Z_i] \Leftrightarrow [h_l] \oplus [h_k] \oplus [h_i] \neq [0] \\ &[Z_{lk}] \neq [Z_j] \Leftrightarrow [h_l] \oplus [h_k] \oplus [h_j] \neq [0] \end{aligned}\right\}, \tag{4.68}$$

rezultă că pentru corecția a două erori sau mai puține este necesar ca suma modulo 2 a patru coloane sau mai puține trebuie să fie diferită de matricea nulă.

Raționând în mod analog, se poate conchide că, în general, pentru corecția a e erori sau mai puține este necesar ca matricea de control să fie astfel întocmită, încât suma modulo 2 a 2e coloane sau mai puține să fie diferită de matricea nulă. Pe de altă parte, s-a demonstrat în § 4.2 că pentru corecția a e erori sau mai puține între cuvintele de cod trebuie să existe o distanță Hamming minimă egală cu 2e+1 (relația 4.18). Se poate demonstra că, dacă această condiție este satisfăcută, atunci şi suma modulo 2 a oricăror 2e coloane este diferită de matricea nulă.

Într-adevăr, dacă distanța Hamming minimă dintre cuvintele de cod este 2e+1, rezultă că ponderea minimă a cuvintelor de cod este 2e+1 şi deci suma modulo 2 a 2e coloane oarecare este diferită de matricea nulă.

4.7. Margini inferioare asupra numărului simbolurilor de control, în cazul corecției erorilor

Dacă pe canalul de transmisiuni pot să apară cel mult e erori, se pune problema determinării numărului de simboluri de control ce trebuie adăugate simbolurilor informaționale pentru corecția erorilor.

Numărul minim de simboluri de control necesar corecției a e erori sau mai puține se numeşte *margine inferioară* a numărului simbolurilor de control.

Dintre marginile cele mai utilizate se amintește *marginea Hamming* şi *marginea Varşarmov-Gilbert*.

Pentru stabilirea marginii Hamming se ține cont de faptul că matricea de control trebuie astfel întocmită, încât să rezulte corectori diferiți de matricea nulă şi, în plus, distincți pentru toate posibilitățile de apariție a cuvintelor eroare care conțin e sau mai puține erori.

Dacă în cuvântul de cod transmis, de lungime n, apare o eroare, indiferent pe ce poziție, se pot întocmi cuvinte eroare distincte, în număr de:

$$\mathbf{N_1 = C_n^1} \tag{4.69}$$

Dacă în cuvântul de cod de lungime n apar două erori distincte, se pot întocmi cuvinte eroare distincte, în număr de:

$$\mathbf{N_2 = C_n^2} \tag{4.70}$$

În general, dacă în cuvântul de cod de lungime n apar "e" erori distincte, se pot întocmi cuvinte eroare distincte, în număr de:

$$\mathbf{N_e = C_n^e} \tag{4.71}$$

Luând în considerare şi cuvântul eroare format numai din zerouri, specificând cazul transmisiei fără erori, rezultă că în cazul apariției a e erori sau mai puține numărul cuvintelor eroare corespunzătoare se poate determina cu relația:

$$\mathbf{N = 1 + N_1 + N_2 + \ldots + N_e = \sum_{i=0}^{e} C_n^i} \tag{4.72}$$

Pe de altă parte, din relația (4.43) rezultă că se pot forma 2^m corectori distincți. Pentru a se realiza o bijecție între mulțimea cuvintelor eroare posibile şi mulțimea

$$\mathbf{2^m \geq \sum_{i=0}^{e} C_n^i} \tag{4.73}$$

corectorilor este necesar să fie îndeplinită condiția:

Numărul întreg pozitiv m, cel mai mic, care satisface relația (4.73), defineşte *marginea Hamming*. Marginea Hamming astfel obținută reprezintă numai o condiție necesară şi nu totdeauna şi suficientă pentru corecția a e erori sau mai puține. Acest lucru se datorează faptului că există cuvinte eroare distincte, de aceeaşi pondere, cărora le corespund corectori identici.

În cazul particular al apariției unei singure erori, fiecărui cuvânt eroare de pondere unu îi va corespunde un corector distinct, deoarece cele n coloane ale matricei de control sunt distincte.

Cu alte cuvinte, relația (4.73) devine necesară şi suficientă pentru e=1, de forma:

$$\mathbf{2^m \geq 1 + n} \tag{4.74}$$

Deoarece un cuvânt de cod este format din k simboluri informaționale şi m simboluri de control, rezultă:

$$\mathbf{n = k + m} \tag{4.75}$$

Cu (4.75) relația (4.74) devine:

$$\mathbf{2^m \geq 1 + k + m} \tag{4.76}$$

În relația (4.76) se va alege numărul întreg pozitiv m, cel mai mic, care o satisface.

Pentru stabilirea marginii Varşarmov-Gilbert se ține cont de faptul că pentru corecție a e erori sau mai puține matricea de control trebuie astfel întocmită, încât suma modulo 2 a oricăror 2e coloane sau mai puține să fie diferită de matricea nulă. În stabilirea acestei margini se presupune că primele n-1 coloane ale matricei de control satisfac această condiție, punându-se problema, în continuare, de a se determina condițiile pe care trebuie să le satisfacă ultima coloană $[h_n]$. Pentru corecția a e erori sau mai puține coloana $[h_n]$ trebuie să satisfacă condițiile:

$$\mathbf{[h_n] \neq [0]} \tag{4.77}$$

$$\mathbf{[h_n] \neq [h_{i_1}], (\forall) i_1 = 1,2,\ldots,n-1} \tag{4.78}$$

$$\mathbf{[h_n] \neq [h_{i_1}] \oplus [h_{i_2}] \underline{\underline{not}}\, [l_{i_1 i_2}], (\forall) i_1, i_2 = \overline{1,(n-1)}, i_1 \neq i_2} \tag{4.79}$$

- -

$$\mathbf{[h_n] \neq [h_{i_1}] \oplus [h_{i_2}] \oplus \ldots \oplus [h_{i_{2e-1}}] \underline{\underline{not}}\, [l_{i_1 i_2 \ldots i_{2e-1}}],} \tag{4.80}$$

oricare ar fi $\mathbf{i_1, i_2, \ldots, i_{2e-1} = \overline{1,(n-1)}}$ şi $i_1 \neq i_2 \neq \ldots \neq i_{2e-1}$.

Din relațiile (4.77) ÷ (4.80) rezultă că $[h_n]$ trebuie să fie diferit de $\mathbf{[0]}$, $\mathbf{[h_{i_1}]}$, $\mathbf{[l_{i_1 i_2}]}$, …, $\mathbf{[l_{i_1 i_2 \ldots i_{2e-1}}]}$. Pentru a determina o astfel de coloană trebuie ca numărul simbolurilor de control m, să fie suficient de mare, astfel încât din cele 2^m coloane distincte ce pot fi formate să se poată elimina coloana [0] (în număr de $\mathbf{C_{n-1}^0 = 1}$), coloanele $\mathbf{[h_{i_1}]}$ (în număr de $\mathbf{C_{n-1}^1}$), coloanele $\mathbf{[l_{i_1 i_2}]}$ (în număr de $\mathbf{C_{n-1}^2}$) ş.a.m.d. până la coloanele $\mathbf{[l_{i_1 i_2 \ldots i_{2e-1}}]}$ (în număr de $\mathbf{C_{n-1}^{2e-1}}$) şi să mai rămână cel puțin o coloană care să fie $[h_n]$.

$$\mathbf{2^m > \sum_{i=0}^{2e-1} C_{n-1}^i} \tag{4.81}$$

Rezultă atunci condiția:

Numărul întreg pozitiv m, cel mai mic, care satisface această relație stabileşte marginea Varşarmov-Gilbert.

Condiția (4.81) este suficientă şi nu totdeauna necesară, deoarece în stabilirea acesteia s-a considerat că toate combinațiile coloanelor de forma $\mathbf{l_{i_1 i_2}, \ldots, l_{i_1 i_2 \ldots i_{2e-1}}}$ sunt distincte.

Cu alte cuvinte, pot fi imaginate coduri corectoare de e erori sau mai puține care necesită un număr de simboluri de control, m, mai mic decât cel care rezultă din relația (4.81).

4.8. Tabele de decodare

Aşa cum s-a arătat în § 4.4, instalația de la recepție, adică decodorul, se implementează după relația:

$$[Z] = [H][v']^T \tag{4.82}$$

Relația (4.82) este echivalentă cu:

$$[Z] = [H][E]^T \tag{4.83}$$

Se pune problema dacă odată determinat corectorul cuvântului recepționat cu relația (4.82), nu se poate deduce din (4.83) cuvântul eroare care, adunat apoi modulo 2 la cuvântul recepționat să determine cuvântul de cod transmis. Acest lucru nu se poate realiza, deoarece matricea de control fiind dreptunghiulară, având un număr de linii m şi un număr de coloane n>m, nu are inversă.

Depăşirea acestui obstacol se poate realiza dacă se ține cont că la recepționarea unui cuvânt se va decide că s-a transmis acel cuvânt de cod care se află la distanța Hamming minimă față de cuvântul recepționat (v.§4.2). Aceasta este echivalent cu identificarea cuvântului eroare de pondere minimă care realizează acelaşi corector ca al cuvântului recepționat. În scopul deducerii unui astfel de cuvânt eroare, se folosesc tabelele de decodare.

Un tabel de decodare conține două coloane. În prima coloană se scriu cuvintele eroare posibile în ordinea crescătoare a ponderilor, iar în coloana a doua se înscriu corectorii corespunzători, calculați cu relația (4.83). Dacă în procesul calculului corectorilor rezultă un corector care a mai fost obținut anterior dintr-un cuvânt eroare de pondere mai mică, atunci cuvântul eroare de pondere mai mare este eliminat din tabel. Tabelul de decodare se consideră complet când prin acest procedeu se obțin toți cei 2^m corectori distincți.

Odată întocmit tabelul de decodare, pentru corecția erorilor dintr-un cuvânt recepționat se procedează astfel:

- se calculează cu relația (4.82) corectorul cuvântului recepționat;
- se identifică în tabelul de decodare cuvântul eroare (evident de pondere minimă) care determină acelaşi corector;
- se adună modulo 2 cuvântul eroare identificat la cuvântul recepționat, rezultând astfel cuvântul de cod transmis.

Pentru fixarea ideilor se consideră un cod bloc, liniar, binar, caracterizat de matricea de control:

$[H]=\begin{bmatrix}100011\\010101\\001110\end{bmatrix}$. Dacă cuvântul recepționat este $[v']=[101111]$, să se determine cuvântul de cod transmis.

Tabelul de decodare este de forma:

	Cuvinte eroare	$[Z]^T$
1	000000	000
2	100000	100
3	010000	010
4	001000	001
5	000100	011
6	000010 ←	101
7	000001	110
8	~~110000~~	~~110~~
9	~~101000~~	~~101~~
10	100100	111

Linia întâi a tabelului corespunde cuvântului eroare de pondere zero. Liniile 2÷7 corespund cuvintelor eroare de pondere unu. Deoarece în liniile 8 şi 9 celor două cuvinte eroare de pondere doi le corespund corectorii din linia a şaptea, respectiv a şasea, obținuți din cuvinte eroare de pondere unu, aceste cuvinte eroare se elimină din tabel.

Numărul corectorilor distincți este egal cu $2^m=2^3=8$. Corectorul cuvântului recepționat se calculează cu relația:

$$[Z]=[H][v']^T=\begin{bmatrix}100011\\010101\\001110\end{bmatrix}\begin{bmatrix}1\\0\\1\\1\\1\\1\end{bmatrix}=\begin{bmatrix}1\\0\\1\end{bmatrix}\Rightarrow [Z]^T=[101]$$

Cuvântul eroare identificat din tabelul de decodare corespunde liniei a şasea. Înseamnă că s-a transmis cuvântul de cod

$$[v]=[v']\oplus[E]=[101111]\oplus[000010]=[101101].$$

În general, structura cuvintelor eroare care compun tabelul de decodare determină structura erorilor ce pot fi corectate. În exemplul de mai sus se pot corecta toate erorile singulare. Deoarece corectorul $[Z]^T=[111]$ poate rezulta şi din alte cuvinte eroare de pondere doi înseamnă că, dacă corectorul unui cuvânt recepționat coincide cu cel din linia a zecea, se va decide că în cuvântul recepționat au apărut erori pe pozițiile 1 şi 4, deşi cele două erori pot apărea şi pe alte poziții.

Prin definiție, se numeşte *cod perfect* codul care poate corecta toate combinațiile de e erori sau mai puține, dar nici o combinație de e+1 erori.

Conform definiției, înseamnă că, în cazul unui cod perfect, în tabelul de decodare vor exista $C_n^0=1$ cuvinte eroare de pondere zero, un număr de $C_n^1=n$ cuvinte eroare de pondere unu, un număr de C_n^2 cuvinte eroare de pondere doi ş.a.m.d., un număr de C_n^e cuvinte eroare de pondere e, toate posedând corectori distincți.

Datorită acestei particularități, marginea Hamming (relația 4.73) devine o condiție necesară şi suficientă pentru codurile perfecte.

Până în prezent se cunosc coduri perfecte corectoare de o eroare şi codul Golay, corector de 3 erori sau mai puține, în care lungimea cuvintelor de cod este n=23, iar numărul simbolurilor de control este m=11, deoarece:

$$2^{11} = C_{23}^{0} + C_{23}^{1} + C_{23}^{2} + C_{23}^{3} \qquad (4.84)$$

4.9. Codor şi decodor Hamming corector de o eroare

Fie k numărul simbolurilor informaționale necesare transmiterii unei informații.

Conform relației (4.76), numărul simbolurilor de control, m, ce trebuie adăugate în scopul corecției unei erori, se determină din relația:

$$2^{m} \geq m + k + 1 \qquad (4.85)$$

S-a demonstrat în § 4.6 că pentru corecția unei erori matricea de control trebuie astfel întocmită, încât toate coloanele acesteia să fie diferite de matricea nulă, $[h_i] \neq [0],\ i = \overline{1,n}$ şi, în plus, suma modulo 2 a oricăror două coloane distincte să fie, de asemenea, diferită de matricea nulă, adică $[h_i] \oplus [h_j] \neq [0]$, oricare ar fi $i, j = \overline{1,n}, i \neq j$..

Pentru a satisface aceste cerințe, Hamming a propus întocmirea matricei de control după următoarea regulă: coloana $[h_i]$ să fie reprezentarea binară a numărului i pe m biți, cu bitul cel mai semnificativ în prima linie. Conform acestei reguli matricea de control va avea forma:

$$[H] = \begin{bmatrix} 0\,0\,0 \ldots 1 \\ \cdots \\ 0\,1\,1 \ldots 1 \\ 1\,0\,1 \ldots 1 \end{bmatrix}_{(m \times n)} \qquad (4.86)$$

În scopul simplificării implementării codorului, tot Hamming a propus ca în structura matriceală a cuvintelor de cod simbolurile de control să fie plasate pe pozițiile 2^0, 2^1, 2^2, ..., 2^{m-1}, numărătoarea efectuându-se de la stânga la dreapta. Cele k simboluri de informație vor ocupa locurile rămase libere, adică:

$$[v] = [c_1 c_2 i_3 c_4 i_5 i_6 i_7 c_8 i_9 \ldots i_n], \qquad (4.87)$$

unde prin $c_j \in \{0,1\}$ s-au notat simbolurile de control, iar pin $i_k \in \{0,1\}$ simbolurile informaționale.

Determinarea simbolurilor de control, cunoscute fiind simbolurile informaționale, se efectuează cu relația (4.41), în care matricea de control [H] are forma (4.86), iar cuvântul de cod structura (4.87), adică:

$$\begin{bmatrix} 0\,0\,0\,\ldots\,1 \\ \cdot\ \cdot\ \cdot \\ \cdot\ \cdot\ \cdot \\ \cdot\ \cdot\ \cdot \\ 0\,1\,1\,\ldots\,1 \\ 1\,0\,1\,\ldots\,1 \end{bmatrix} \begin{bmatrix} c_1 \\ c_2 \\ i_3 \\ c_4 \\ i_5 \\ i_6 \\ i_7 \\ c_8 \\ . \\ . \\ . \\ i_n \end{bmatrix} = [0] \qquad (4.88)$$

$$\left.\begin{aligned} c_1 &= i_3 \oplus i_5 \oplus \ldots \oplus i_n \\ c_2 &= i_3 \oplus i_6 \oplus \ldots \oplus i_n \\ c_4 &= i_5 \oplus i_6 \oplus \ldots \oplus i_n \\ &- - - - - - - - - - - - \end{aligned}\right\} \qquad (4.89)$$

Ecuația matriceală (4.88) este echivalentă cu următorul sistem de ecuații:

Instalația de la emisie, adică codorul, care are misiunea de a calcula simbolurile de control, cunoscute fiind simbolurile informaționale, va fi formată dintr-un registru care conține n circuite basculante bistabile în care se stochează simbolurile informaționale în pozițiile corespunzătoare din cuvântul de cod şi o serie de sumatoare modulo 2 care, conform sistemului de ecuații (4.89), calculează simbolurile de control. Simbolurile de control astfel calculate se stochează apoi în pozițiile în care acestea intervin în cuvântul de cod.

Pentru fixarea ideilor, se presupune că pentru transmiterea unei anumite informații sunt necesare k=4 simboluri. Pentru a întocmi instalația de codare, se calculează mai întâi numărul necesar de simboluri de control cu relația (4.85), adică:

$$2^m \geq m+5 \Rightarrow m=3 \Rightarrow n=m+k = 7$$

Cele trei simboluri de control, conform sistemului de ecuații (4.89), se calculează cu relațiile:

$$c_1 = i_3 \oplus i_5 \oplus i_7$$
$$c_2 = i_3 \oplus i_6 \oplus i_7$$
$$c_4 = i_5 \oplus i_6 \oplus i_7$$

Codorul va avea atunci structura din figura 4.3.

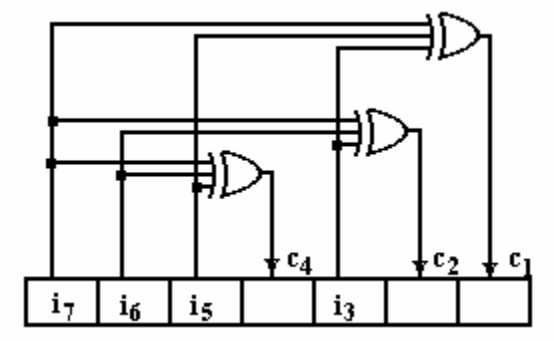

Fig.4.3 Codor Hamming (7,4).

Instalația de la recepție, adică decodorul, se implementează conform relației (4.42). Dacă cuvântul recepționat este de forma:

$$[v'] = [c_1' c_2' i_3' c_4' i_5' i_6' i_7' c_8' i_9' \ldots i_n'], \tag{4.90}$$

atunci, înlocuind (4.86) şi (4.90) în (4.42) şi ținând cont de (4.43), rezultă:

$$\left.\begin{aligned} z_m &= c_1' \oplus i_3' \oplus i_5' \oplus \ldots \oplus i_n' \\ z_{m-1} &= c_2' \oplus i_3' \oplus i_6' \oplus \ldots \oplus i_n' \\ z_{m-2} &= c_4' \oplus i_5' \oplus i_6' \oplus \ldots \oplus i_n' \\ &- - - - - - - - - - - - - - - - - \end{aligned}\right\} \tag{4.91}$$

Pe de altă parte, se ştie că în cuvântul recepționat există o singură eroare, caracterizată de cuvântul eroare:

$$[E] = [\ldots \alpha_i \ldots], \tag{4.92}$$

unde $\alpha_i = 1, (\forall) i = \overline{1, n}$, restul elementelor fiind nule.

Notând cu $[h_1]$, $[h_2]$, ..., $[h_n]$ coloanele matricei definite cu relația (4.86), rezultă din (4.44) şi (4.92) structura matriceală a corectorului:

$$[Z] = \begin{bmatrix} z_1 \\ z_2 \\ . \\ . \\ . \\ z_m \end{bmatrix} = \alpha_i [h_i] = [h_i] \tag{4.93}$$

Deoarece coloana $[h_i]$ este reprezentarea binară a numărului i pe m biți, rezultă că poziția erorii în cuvântul recepționat se deduce astfel:

- se calculează componentele corectorului cu relația (4.42), sau, echivalent, cu sistemul de ecuații (4.91);
- se transcrie în zecimal numărul binar corespunzător componentelor corectorului, adică:

$$(z_1 z_2 \ldots z_m)_2 = (i)_{10} \tag{4.94}$$

În felul acesta se precizează poziția i a erorii. Decodorul Hamming, corector de o eroare va fi format, deci, dintr-un registru cu n circuite basculante bistabile în care va fi memorat cuvântul recepționat, o serie de sumatoare modulo 2 care, conform sistemului de ecuații (4.91), calculează componentele corectorului şi un descifrator care, conform relației (4.94), este un decodificator din binar în zecimal cu m intrări şi n ieşiri.

Pentru fixarea ideilor, se presupune că s-a recepționat cuvântul $[v'] = [c_1' c_2' i_3' c_4' i_5' i_6' i_7']$. Conform relației (4.91), se poate scrie, în acest caz particular, sistemul de ecuații:

$$z_1 = c_4' \oplus i_5' \oplus i_6' \oplus i_7'$$
$$z_2 = c_2' \oplus i_3' \oplus i_6' \oplus i_7'$$
$$z_3 = c_1' \oplus i_3' \oplus i_5' \oplus i_7'$$

Schema bloc a decodorului este reprezentată în figura 4.4.

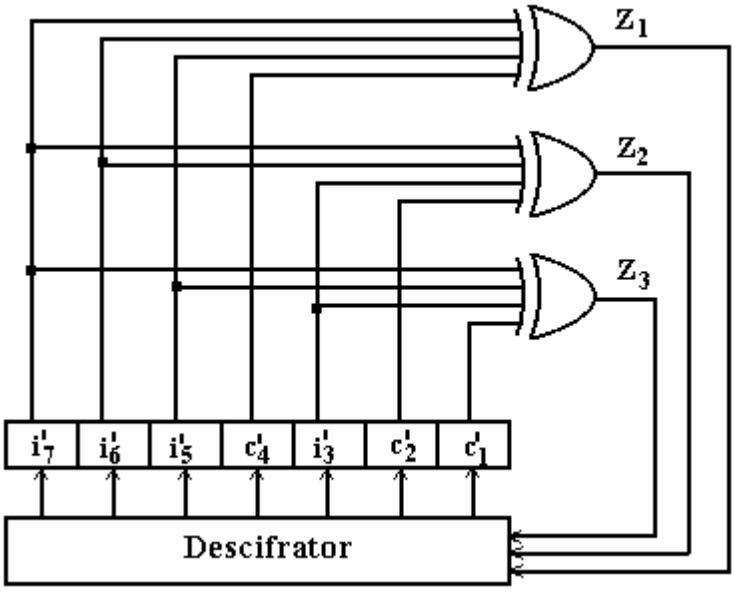

Fig.4.4. Decodor Hamming (7,4).

4.10. Codor şi decodor Hamming corector de o eroare, detector de erori duble

În scopul corecției unei erori şi a detectării erorilor duble trebuie adăugat la cele k simboluri informaționale, pe lângă simbolurile de control necesare corecției unei erori, un simbol suplimentar, numit de *control al parității*, prin care se decide dacă în cuvântul recepționat sunt două erori sau o eroare.

Structura matriceală a cuvântului de cod în acest caz este de forma:

$$[v] = [c_0 c_1 c_2 i_3 c_4 i_5 i_6 i_7 c_8 i_9 \ldots i_n], \qquad (4.95)$$

unde c_0 este simbolul suplimentar de control al parității, ales astfel încât să respecte relația:

$$\mathbf{c_0 \oplus c_1 \oplus c_2 \oplus i_3 \oplus c_4 \oplus i_5 \oplus \ldots \oplus i_n = 0} \qquad \textbf{(4.96)}$$

Pentru ca simbolurile de control c_1, c_2, c_4, ... să fie calculate cu aceleaşi relații ca în cazul codului Hamming corector de o eroare (rel.4.89) şi să aibă loc suplimentar relația (4.96), matricea de control trebuie să aibă structura:

$$\mathbf{[H]} = \begin{bmatrix} 0 & h_1 & h_2 & \ldots & h_n \\ 1 & 1 & 1 & \ldots & 1 \end{bmatrix}, \qquad \textbf{(4.97)}$$

unde $\mathbf{[h_i]}$, $\mathbf{i = \overline{1,n}}$, reprezintă coloanele matricei de control din cazul codului Hamming corector de o eroare.

Fie

$$\mathbf{[v'] = [c'_0 c'_1 c'_2 i'_3 c'_4 i'_5 i'_6 i'_7 c'_8 i'_9 \ldots i'_n]} \qquad \textbf{(4.98)}$$

cuvântul recepționat.

Calculând corectorul acestui cuvânt cu relația (4.42), în care [H] este dată de relația (4.97), structura matriceală a corectorului va fi de forma:

$$\mathbf{[Z]^T = [z_1 z_2 \ldots z_m z_0],} \qquad \textbf{(4.99)}$$

unde componentele corectorului z_1, z_2, ..., z_m vor fi calculate cu aceleaşi relații ca în cazul codului Hamming corector de o eroare (relația 4.91), în timp ce

$$\mathbf{z_0 = c'_0 \oplus c'_1 \oplus c'_2 \oplus i'_3 \oplus c'_4 \oplus i'_5 \oplus \ldots \oplus i'_n} \qquad \textbf{(4.100)}$$

Comparând relația (4.96) cu (4.100), rezultă că, dacă în cuvântul recepționat a apărut un număr impar de erori, atunci z_0=1, iar dacă a apărut un număr par de erori, atunci z_0=0.

Dacă pe canalul de transmisiuni pot să apară cel mult două erori, rezultă următoare trei situații:

a) Toate componentele corectorului sunt nule, adică:

$$\mathbf{z_0 = z_1 = z_2 = \ldots = z_m = 0}, \qquad \textbf{(4.101)}$$

În acest caz se decide că ceea ce s-a recepționat este corect.

b) Componenta z_0=1. În acest caz, în cuvântul recepționat a apărut o eroare a cărei poziție se determină transcriind în zecimal numărul binar format din componentele corectorului z_1, z_2, ..., z_m, adică:

$$\mathbf{(z_1 z_2 \ldots z_m)_2 = (i)_{10}} \qquad \textbf{(4.102)}$$

c) Componenta z_0=0, iar celelalte componente ale corectorului nu sunt toate nule. În acest caz se decide că în cuvântul recepționat sunt două erori necorectabile, caz în care se va cere retransmisia cuvântului respectiv.

4.11. Definirea cuvintelor de cod în cazul codurilor ciclice nesistematice şi sistematice

În cazul *codurilor ciclice binare*, cuvintele de cod sunt exprimate prin polinoame de grad n-1 sau mai mic, de forma:

$$\mathbf{v(x) = a_0 \oplus a_1 x \oplus \ldots \oplus a_{n-1} x^{n-1},\ \ a_i \in \{0,1\},\ i = \overline{0,(n-1)}} \qquad \textbf{(4.103)}$$

Din mulțimea polinoamelor de grad n-1 sau mai mic, se aleg drept cuvinte de cod numai acelea care posedă următoarele trei proprietăți:

1) Polinomul v(x) este divizibil la un polinom numit *polinom generator al codului ciclic*, g(x), definit cu relația:

$$\begin{aligned} &\mathbf{g(x) = g_0 \oplus g_1 x \oplus \ldots \oplus g_m x^m,\ \ g_0 = g_m = 1;} \\ &\mathbf{g_i \in \{0,1\},\ i = \overline{1,(m-1)},\ m \le n-1} \end{aligned} \qquad \textbf{(4.104)}$$

2) Lungimea cuvintelor de cod, n, trebuie să satisfacă relația:

$$\mathbf{n = 2^m - 1,} \qquad \textbf{(4.105)}$$

unde m este gradul polinomului generator al codului.

3) Polinomul generator al codului este un divizor al polinomului $\mathbf{x^n \oplus 1}$.

Se poate demonstra că orice permutare ciclică a unui cuvânt de cod determină un nou cuvânt de cod, de unde şi denumirea acestor coduri.

Într-adevăr, fie v(x), dat de relația (4.103), un cuvânt de cod, adică un polinom divizibil cu g(x), ceea ce se va scrie sub forma:

$$\mathbf{g(x) \,|\, (a_0 \oplus a_1 x \oplus \ldots \oplus a_{n-1} x^{n-1})} \qquad \textbf{(4.106)}$$

Dacă relația (4.106) este adevărată, atunci, evident, g(x) va divide şi polinomul x·v(x), adică:

$$\mathbf{g(x) \,|\, (a_0 x \oplus a_1 x^2 \oplus \ldots \oplus a_{n-1} x^n)} \qquad \textbf{(4.107)}$$

Relația (4.107) poate fi scrisă, echivalent, sub forma:

$$\mathbf{g(x) \,|\, (a_0 x \oplus a_1 x^2 \oplus \ldots \oplus a_{n-1} x^n \oplus a_{n-1} \oplus a_{n-1}),}$$

sau

$$\mathbf{g(x) \,|\, [(a_{n-1} \oplus a_0 x \oplus a_1 x^2 \oplus \ldots \oplus a_{n-2} x^{n-1}) \oplus a_{n-1}(x^n \oplus 1)]} \qquad \textbf{(4.108)}$$

Deoarece

$$\mathbf{g(x) \,|\, (x^n \oplus 1),} \qquad \textbf{(4.109)}$$

rezultă:

$$\mathbf{g(x) \,|\, (a_{n-1} \oplus a_0 x \oplus a_1 x^2 \oplus \ldots \oplus a_{n-2} x^{n-1})} \qquad \textbf{(4.110)}$$

Din relația (4.110) rezultă că polinomul $\mathbf{a_{n-1} \oplus a_0 x \oplus a_1 x^2 \oplus \ldots \oplus a_{n-2} x^{n-1}}$, care este prima permutare ciclică a polinomului v(x), este, de asemenea, cuvânt de

cod. În mod similar se poate demonstra că orice permutare ciclică a unui cuvânt de cod determină un alt cuvânt de cod.

Pentru întocmirea cuvintelor de cod în cazul codurilor ciclice se construieşte mai întâi *polinomul informațional*, ai cărui coeficienți sunt simbolurile informaționale $i_0, i_1, ..., i_{k-1} \in \{0,1\}$. Astfel, pentru transmiterea numărului zecimal N, simbolurile informaționale rezultă prin transcrierea binară a numărului N, adică:

$$(N)_{10} = (i_0 i_1 \ldots i_{k-1})_2 \tag{4.111}$$

Polinomul informațional va fi de forma:

$$i(x) = i_0 \oplus i_1 x \oplus \ldots \oplus i_{k-1} x^{k-1} \tag{4.112}$$

În cazul codurilor *ciclice nesistematice*, cuvintele de cod se întocmesc după relația:

$$v(x) = g(x) \cdot i(x) \tag{4.113}$$

care, evident, va fi un polinom de grad n-1 sau mai mic, divizibil cu polinomul generator al codului. Se reaminteşte că lungimea cuvintelor de cod se determină cu relația:

$$n = m + k \tag{4.114}$$

În cazul codurilor *ciclice sistematice* cu simbolurile informaționale plasate grupat la sfârşitul cuvintelor de cod se procedează astfel:

a) se reține restul divizării polinomului $x^m \cdot i(x)$ la g(x), adică, dacă se notează acest rest cu r(x), se poate scrie:

$$x^m \cdot i(x) = q(x)g(x) \oplus r(x) \tag{4.115}$$

unde q(x) este câtul divizării.

b) cuvântul de cod se întocmeşte după relația:

$$v(x) = r(x) \oplus x^m \cdot i(x) \tag{4.116}$$

Se poate arăta uşor că polinomul v(x) dat de relația (4.116) este divizibil la polinomul generator al codului, g(x), adică satisface definiția cuvintelor de cod din cazul codurilor ciclice.

Într-adevăr, înlocuind (4.115) în relația (4.116), rezultă:

$$\left.\begin{array}{l} v(x) = r(x) \oplus q(x)g(x) \oplus r(x) \\ r(x) \oplus r(x) = 0 \end{array}\right\} \Rightarrow v(x) = q(x) \cdot g(x) \tag{4.117}$$

ceea ce trebuia demonstrat.

Mai mult, deoarece restul r(x) este un polinom de grad maxim m-1, iar polinomul $x^m \cdot i(x)$ are gradul minim egal cu m, rezultă că restul r(x) nu va afecta simbolurile informaționale, caracterizate de polinomul i(x).

Pentru fixarea ideilor, fie numărul N=9. Dacă polinomul generator al codului este $g(x) = 1 \oplus x \oplus x^3$, atunci cuvântul de cod ciclic nesistematic se determină după

cum urmează: se transcrie în binar numărul zecimal 9, rezultând astfel polinomul informațional, adică: $(9)_{10} = (1\ 0\ 0\ 1) \Rightarrow i(x) = 1 \oplus x^3$. Conform relației (4.113),

$$\begin{matrix} & \downarrow & \downarrow & \downarrow & \downarrow \\ & i_0 & i_1 & i_2 & i_3 \end{matrix}$$

cuvântul de cod va fi de forma: $v(x) = i(x)\cdot g(x) = (1\oplus x^3)(1\oplus x\oplus x^3) = 1\oplus x^3\oplus x\oplus x^4\oplus \oplus x^3\oplus x^6 = 1\oplus x\oplus x^4\oplus x^6$. Grupând coeficienții acestui polinom într-o matrice linie, rezultă:

$$[v] = \begin{matrix}[& 1 & 1 & 0 & 0 & 1 & 0 & 1 &] \\ & \downarrow & \downarrow & \downarrow & \downarrow & \downarrow & \downarrow & \downarrow & \\ & x^0 & x^1 & x^2 & x^3 & x^4 & x^5 & x^6 & \end{matrix}$$

Cuvântul de cod ciclic sistematic se formează conform relației (4.116), adică: $x^m\cdot i(x) = x^3(1\oplus x^3)=x^3\oplus x^6$. $x^6\oplus x^3 = (x^3+x)(x^3\oplus x\oplus 1) \oplus (x^2\oplus x)$, unde $x^3\oplus x$ este câtul divizării, iar restul $r(x) = x^2\oplus x$. Rezultă atunci $v(x) = r(x) \oplus x^m i(x) = x\oplus x^2\oplus x^3\oplus x^6$. Sub formă matriceală, cuvântul de cod este [v] = [0111001]. Primele trei simboluri sunt simbolurile de control, în timp ce ultimele 4 sunt informaționale, obținându-se într-adevăr un cod sistematic, cu simbolurile informaționale plasate grupat la sfârşitul cuvântului de cod.

Fie

$$\mathbf{v'(x) = a'_0 \oplus a'_1 x \oplus \ldots \oplus a'_{n-1} x^{n-1}}, \qquad \textbf{(4.118)}$$

cuvântul recepționat.

La recepție, decodorul realizează divizarea polinomului $v'(x)$ la acelaşi polinom generator al codului, g(x). Dacă în urma divizării rezultă un rest nul, se decide că s-a recepționat un cuvânt corect. Dacă în urma divizării rezultă un rest diferit de zero, se realizează operația de detecție a erorilor, adică se decide că în cuvântul recepționat sunt erori. Mai mult, dacă din structura restului se pot localiza pozițiile erorilor, se realizează operația de corecție a acestora.

În cazul codurilor ciclice atât la emisie, cât şi la recepție, trebuie să se realizeze operații de multiplicare şi/sau divizare a polinoamelor cu coeficienți în mulțimea {0,1}.

4.12. Definirea funcției de transfer a circuitelor de multiplicare, sau divizare a polinoamelor cu coeficienți în mulțimea {0,1}

Circuitele de multiplicare, sau divizare a polinoamelor cu coeficienți în mulțimea {0,1}, sunt circuite electronice secvențiale liniare cu o intrare şi o ieşire.

În cazul circuitelor secvențiale cu o intrare şi o ieşire, starea ieşirii la un moment dat depinde atât de starea intrării, cât şi de starea circuitului din acel moment.

Circuitele secvențiale folosite în calitate de circuite de multiplicare sau divizare a polinoamelor cu coeficienți în mulțimea {0,1} funcționează de obicei sincron, adică

aplicarea intrării, schimbarea stării circuitului, precum şi furnizarea ieşirii se realizează la momente discrete de timp, de obicei echidistante, cu ajutorul impulsurilor de tact. Circuitele secvențiale utilizate ca circuite de multiplicare sau divizare în cazul codurilor ciclice binare sunt liniare.

Circuitele secvențiale se numesc *liniare*, dacă starea ieşirii la un moment dat este o combinație liniară a stărilor intrării. Deoarece suma modulo 2 este o operație liniară, în cazul circuitelor secvențiale liniare folosite în calitate de circuite de multiplicare sau divizare, prin combinații liniare, se va înțelege realizarea de sume modulo 2.

Pentru întocmirea circuitelor de multiplicare sau divizare a polinoamelor cu coeficienți în mulțimea {0,1} se folosesc trei elemente de bază:

1) Circuite basculante bistabile, numite şi celule binare;

2) Sumatoare modulo 2;

3) Multiplicatoare cu constantele 0 sau 1.

Cele trei elemente de bază vor fi reprezentate în circuitele de multiplicare sau divizare, aşa cum sunt indicate în figura 4.5.

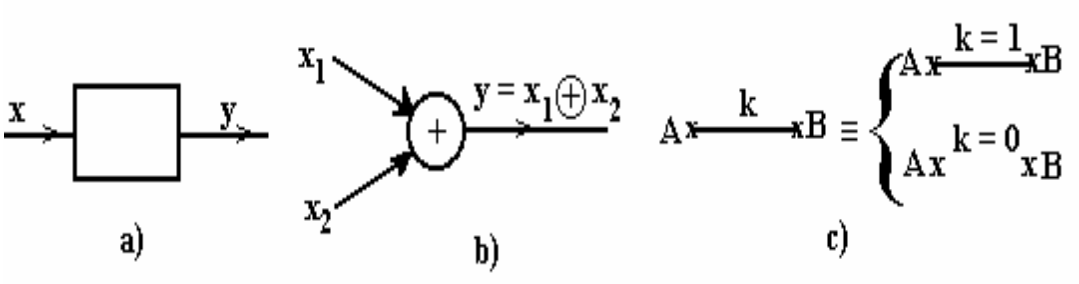

Fig. 4.5. Reprezentarea elementelor de bază din circuitele de multiplicare sau divizare

În figura 4.5,a, s-a reprezentat un circuit basculant bistabil, fără a se mai indica intrarea de tact. La aplicarea tactului, celula binară va trece în starea corespunzătoare stării intrării, în timp ce la ieşire se va furniza starea precedentă. În scopul evidențierii întârzierii de un tact realizate de o celulă binară, se consideră un circuit basculant bistabil de tip D (de exemplu, CDB - 474E), însoțit de formele de undă ale tactului, intrării şi ieşirii, aşa cum este reprezentat în figura 4.6.

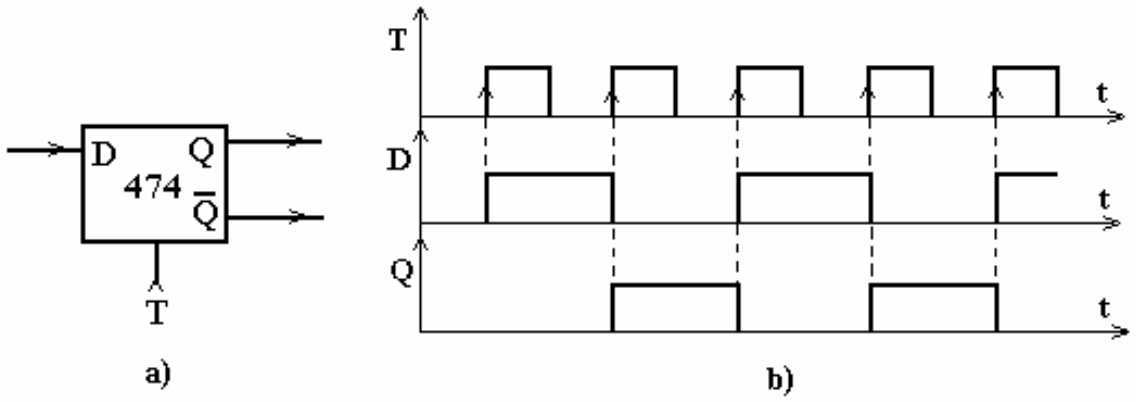

Fig.4.6. C.B.B. de tip D, a), şi formele de undă b).

În figura 4.5,b s-a reprezentat un sumator modulo 2, iar în fig.4.5,c un multiplicator cu constanta k, ce poate lua numai două valori. În cazul în care k=1, există legătură galvanică între punctele A şi B, în timp ce, atunci când k=0, legătura dintre aceste puncte este întreruptă.

Se face convenţia ca aplicarea polinomului la intrarea circuitelor de multiplicare sau divizare să se efectueze în ordinea descrescătoare a puterilor lui x. Aceasta înseamnă că dacă la intrarea unui astfel de circuit se aplică polinomul:

$$\mathbf{P(x) = a_0 \oplus a_1 x \oplus \ldots \oplus a_{n-1} x^{n-1} \oplus a_n x^n}, \qquad \textbf{(4.119)}$$

atunci, la primul tact, la intrarea circuitului se aplică simbolul $a_n \in \{0,1\}$, la al doilea tact simbolul $a_{n-1} \in \{0,1\}$ ş.a.m.d., la tactul n+1 aplicându-se simbolul $a_0 \in \{0,1\}$.

Analiza circuitelor de multiplicare sau divizare a polinoamelor cu coeficienţi în mulţimea {0,1} se face comod, dacă se introduce *operatorul de întârziere D*, care specifică întârzierea de un tact.

Produsul $D^i a_k$ semnifică întârzierea simbolului binar a_k cu i tacte.

Utilizând operatorul de întârziere D, secvenţa de intrare corespunzătoare polinomului P(x) se poate scrie sub forma:

$$\mathbf{P(D) = a_n \oplus Da_{n-1} \oplus \ldots \oplus D^{n-1} a_1 \oplus D^n a_0}, \qquad \textbf{(4.120)}$$

sau, echivalent:

$$\mathbf{P(D) = D^n (a_0 \oplus a_1 D^{-1} \oplus \ldots \oplus a_{n-1} D^{-(n-1)} \oplus a_n D^{-n})} \qquad \textbf{(4.121)}$$

Comparând relaţia (4.119) cu (4.121), se poate scrie echivalenţa:

$$\mathbf{D^{-i} = x^i} \qquad \textbf{(4.122)}$$

Datorită acestei echivalenţe se poate înlocui un polinom în variabila D^{-1} printr-un polinom în variabila x.

În general, funcţia oricărui circuit liniar, invariant în timp este definită prin funcţia sa de transfer. În cazul particular al circuitelor de multiplicare sau divizare a polinoamelor cu coeficienţi în mulţimea {0,1}, realizate prin circuite secvenţiale liniare, "*funcţia de transfer*" a acestora se defineşte ca raportul dintre secvenţa de ieşire, Y, şi secvenţa de la intrare, X, ambele exprimate prin operatorul de întârziere D.

Funcţia de transfer a circuitelor de multiplicare sau divizare poate fi determinată uşor, dacă acestor circuite li se ataşează un graf şi, apoi, cu ajutorul formulei lui Mason, se determină transmitanţa grafului, T, care va reprezenta funcţia de transfer a circuitului.

Graful ataşat unui circuit de multiplicare sau divizare a polinoamelor cu coeficienţi în mulţimea {0,1} se construieşte după următoarele trei reguli:

1) Fiecărei celule binare din circuit îi va corespunde în graf o ramură orientată, înzestrată cu transmitanţa egală cu D, punându-se astfel în evidenţă întârzierea de un tact, realizată de celula binară;

2) Fiecărui sumator modulo 2 din circuit îi va corespunde în graf un nod;

3) Multiplicatorului cu constanta $k \in \{0,1\}$ îi va corespunde în graf o ramură de transmitanță zero, dacă k=0, şi o ramură de transmitanță 1, dacă k=1.

Notând cu C_1, C_2, ..., C_p transmitanțele căilor din graful ataşat şi cu L_1, L_2, ..., L_r transmitanțele buclelor, transmitanța grafului se deduce cu formula lui Mason:

$$\mathbf{T = \frac{Y}{X} = \frac{[(C_1 \oplus C_2 \oplus \ldots \oplus C_p)(1 \oplus L_1)(1 \oplus L_2)\ldots(1 \oplus L_r)]^*}{[(1 \oplus L_1)(1 \oplus L_2)\ldots(1 \oplus L_r)]^*}} \qquad \textbf{(4.123)}$$

Asteriscurile de la numărătorul şi numitorul relației (4.123) specifică faptul că trebuie eliminați termenii care conțin bucle adiacente, precum şi termenii corespunzători adiacențelor dintre căi şi bucle.

4.13. Circuite de multiplicare a polinoamelor cu coeficienți în mulțimea {0,1}

Fie g(x) dat de relația (4.104) polinomul după care se întocmesc circuitele de multiplicare. Aceste circuite se pot întocmi în două variante:

a) cu sumatoarele modulo 2 intercalate între celulele binare;

b) cu sumatoarele modulo 2 plasate în afara celulelor binare.

Se va arăta că circuitul reprezentat în figura 4.7 este un circuit de multiplicare, întocmit după polinomul g(x), cu sumatoarele modulo 2 intercalate între celulele binare.

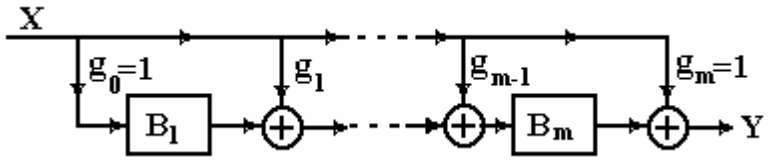

Fig.4.7. Circuit de multiplicare cu sumatoarele modulo 2 intercalate între celulele binare.

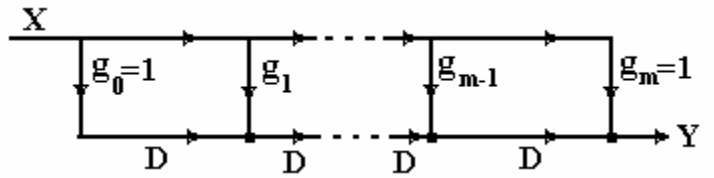

Fig.4.8. Graful ataşat circuitului de multiplicare din fig.4.7.

În figura 4.7, prin B_i, $i=\overline{1,m}$, s-au notat celulele binare. Graful ataşat acestui circuit este reprezentat în figura 4.8. Din graf se constată inexistența buclelor, deci:

$$\mathbf{L_1 = L_2 = ... = L_r = 0} \tag{4.124}$$

şi existența căilor de transmitanțe:

$$\left.\begin{aligned} &\mathbf{C_1 = g_m = 1} \\ &\mathbf{C_2 = g_{m-1} D} \\ &\mathbf{- - - - - - -} \\ &\mathbf{C_m = g_1 D^{m-1}} \\ &\mathbf{C_{m+1} = g_0 D^m = D^m} \end{aligned}\right\} \tag{4.125}$$

Ținând cont de relațiile (4.124) şi (4.125), relația generală (4.123) devine:

$$\begin{aligned} \mathbf{T = \frac{Y}{X} = C_1 \oplus C_2 \oplus ... \oplus C_{m+1} =} \\ \mathbf{= g_m \oplus g_{m-1} D \oplus ... \oplus g_1 D^{m-1} \oplus g_0 D^m =} \\ \mathbf{= D^m (g_0 \oplus g_1 D^{-1} \oplus ... \oplus g_{m-1} D^{-(m-1)} \oplus g_m D^{-m})} \end{aligned} \tag{4.126}$$

Având în vedere (4.121) şi (4.122), relația (4.126) se poate scrie, echivalent, sub forma:

$$T = \frac{Y}{X} = D^m \cdot g(x) \Leftrightarrow Y = D^m g(x) X \qquad (4.127)$$

Relația (4.127) demonstrează că circuitul propus în figura 4.7 realizează multiplicarea polinomului de la intrare, reprezentat prin secvența X cu polinomul g(x), după care s-a întocmit circuitul. Factorul D^m din această relație specifică faptul că multiplicarea este realizată cu o întârziere de m tacte.

Se va arăta în continuare că circuitul reprezentat în figura 4.9 este un circuit de multiplicare, cu sumatoarele modulo 2 plasate înafara celulelor binare.

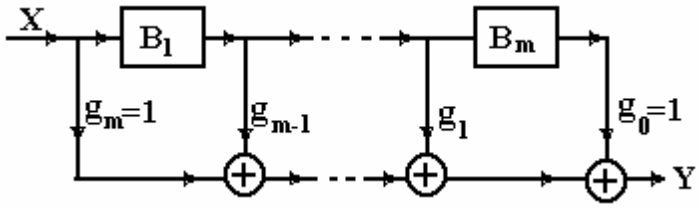

Fig.4.9. Circuit de multiplicare cu sumatoarele modulo 2 plasate înafara celulelor binare.

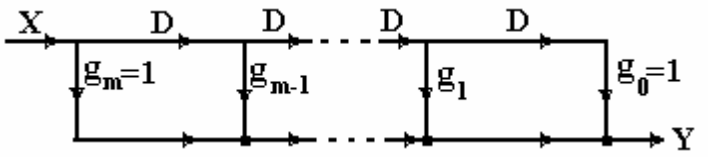

Fig.4.10. Graful ataşat circuitului de multiplicare din figura 4.9.

Graful ataşat acestui circuit este reprezentat în figura 4.10. Din acest graf se constată inexistența buclelor şi existența aceloraşi căi ca în cazul circuitului din figura 4.7, fiind deci adevărată şi în acest caz relația (4.127).

În ambele circuite de multiplicare analizate se observă necesitatea utilizării unui număr de celule binare egal cu gradul polinomului g(x). Deoarece multiplicarea se realizează cu o întârziere de m tacte, rezultă că, dacă la intrare se aplică un polinom de grad n, atunci multiplicarea se va efectua după n+m+1 tacte, deoarece sunt necesare n+1 tacte pentru aplicarea la intrare a celor n+1 coeficienți binari ai polinomului de la intrare şi m tacte datorate întârzierii realizate de circuit.

În ultimele m tacte, la intrarea circuitelor se va aplica "0" logic. Înaintea primului tact, intrarea şi toate celulele binare se resetează.

Pentru fixarea ideilor, fie polinomul de la intrare $P(x)=1\oplus x^3\oplus x^4$ şi polinomul generator $g(x)=1\oplus x\oplus x^3$. Circuitul de multiplicare întocmit după acest polinom, cu sumatoarele modulo 2 plasate înafara celulelor binare, este reprezentat în fig.4.11.

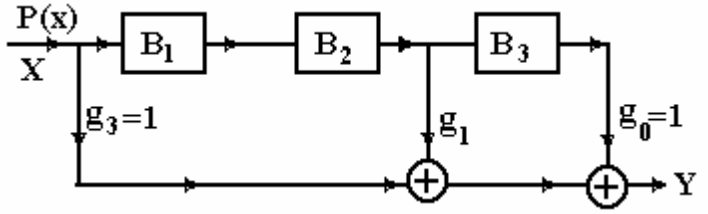

Fig.4.11. Circuitul de multiplicare întocmit după polinomul $g(x)= 1\oplus x\oplus x^3$, cu sumatoarele modulo 2 plasate în afara celulelor binare.

Tact	X	B_1	B_2	B_3	Y
	0.	0.	0.	0.	
1	$x^4 \rightarrow 1.$	0.	0.	0.	1 x^7
2	$x^3 \rightarrow 1.$	1.	0.	0.	1 x^6
3	$x^2 \rightarrow 0.$	1.	1.	0.	1. x^5
4	$x \rightarrow 0.$	0.	1.	1.	0 x^4
5	$x^0 \rightarrow 1.$	0.	0.	1.	0 x^3
6	0.	1.	0.	0.	0 x^2
7	0.	0.	1.	0.	1 x^1
8	0.	0.	0.	1	1 x^0

Funcționarea circuitului este sintetizată în tabelul alăturat. Conform covenției, polinomul se aplică la intrarea circuitului în ordinea descrescătoare a puterilor lui x. Deoarece polinomul de la intrare are gradul n=4, iar g(x) are gradul m=3, rezultă că multiplicarea se va efectua după n+m+1 = 4+3+1 = 8 tacte. În ultimele m=3 tacte, la intrare se aplică zerouri.

Din circuitul de multiplicare reprezentat în figura 4.11 rezultă că la aplicarea tactului celula binară B_1 va trece în starea anterioară intrării X, celula binară B_2 va trece în starea anterioară a lui B_1, celula binară B_3 va trece în starea anterioară a lui B_2, iar ieşirea se va obține sumând modulo 2 stările curente ale intrării şi ale celulelor binare B_2 şi B_3.

Inițial, intrarea şi celulele binare sunt resetate, fapt marcat de prima linie a tabelului. Polinomul produs rezultant este $x^7 \oplus x^6 \oplus x^5 \oplus x \oplus 1$, fapt care poate fi verificat prin multiplicarea polinoamelor $P(x) = 1 \oplus x^3 \oplus x^4$ şi $g(x) = 1 \oplus x \oplus x^3$.

Într-adevăr,

$P(x) \cdot g(x) = (1 \oplus x^3 \oplus x^4)(1 \oplus x \oplus x^3) = 1 \oplus x^3 \oplus x^4 \oplus x \oplus x^4 \oplus x^5 \oplus x^3 \oplus x^6 \oplus x^7 =$

$= 1 \oplus x \oplus x^5 \oplus x^6 \oplus x^7$, deoarece prin sumarea modulo 2 a unui număr par de termeni identici, rezultatul este nul, iar prin sumarea modulo 2 a unui număr impar de termeni identici, va rezulta un singur termen.

4.14 Circuite de divizare a polinoamelor cu coeficienți în mulțimea {0,1}

Circuitele de divizare a polinoamelor cu coeficienți în mulțimea {0,1}, se pot realiza, ca şi circuitele de multiplicare, în două variante şi anume: a) cu sumatoarele modulo 2 intercalate între celulele binare; b) cu sumatoarele modulo 2 plasate înafara celulelor binare.

Se va arăta că circuitul reprezentat în figura 4.12 este un circuit de divizare, întocmit după polinomul g(x), cu sumatoarele modulo 2 intercalate între celulele binare.

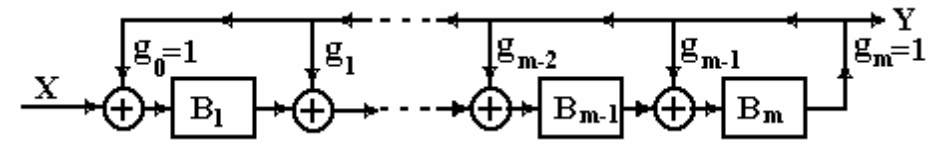

Fig.4.12. Circuit de divizare cu sumatoarele modulo 2 intercalate între celulele binare.

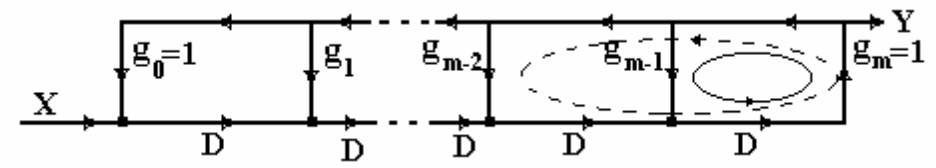

Fig.4.13. Graful ataşat circuitului de divizare din fig.4.12.

Graful ataşat acestui circuit este reprezentat în figura 4.13. Din graf se constată existența unei singure căi, de transmitanță:

$$\mathbf{C_1 = D^m} \tag{4.127}$$

$$\left.\begin{aligned} &\mathbf{L_1 = g_{m-1}\ D\ (bucla\ figurata\ cu\ linie\ continua)}\\ &\mathbf{L_2 = g_{m-2}\ D^2\ (bucla\ figurata\ cu\ linie\ \ int\ rerupta)}\\ &\mathbf{- - - - - - -}\\ &\mathbf{L_{m-1} = g_1 D^{m-1}}\\ &\mathbf{L_m = g_0 D^m = D^m} \end{aligned}\right\} \tag{4.128}$$

şi buclele cu transmitanțele:

Deoarece calea C_1 este adiacentă cu toate buclele şi toate buclele sunt adiacente

$$\mathbf{T = \frac{Y}{X} = \frac{C_1}{1 \oplus L_1 \oplus L_2 \oplus \ldots \oplus L_{m-1} \oplus L_m}} \tag{4.129}$$

între ele, relația generală (4.123) devine:

Înlocuind (4.127) şi (4.128) în relația (4.129), rezultă:

$$\mathbf{T = \frac{Y}{X} = \frac{D^m}{1 \oplus g_{m-1}D \oplus g_{m-2}D^2 \oplus \ldots \oplus g_1 D^{m-1} + g_0 D^m} =}$$
$$\mathbf{= \frac{D^m}{D^m (g_0 \oplus g_1 D^{-1} \oplus \ldots \oplus g_{m-1} D^{-(m-1)} \oplus D^{-m})}} \tag{4.130}$$

Ținând cont de (4.122), relația (4.130) devine:

$$\mathbf{T = \frac{Y}{X} = \frac{1}{g(x)} \Rightarrow Y = \frac{X}{g(x)}} \tag{4.131}$$

Relația (4.131) demonstrează că circuitul propus în figura 4.12 realizează divizarea polinomului de la intrare, reprezentat prin secvența X, prin polinomul g(x), după care s-a întocmit circuitul. Divizarea are loc fără nici o întârziere. Astfel, dacă polinomul de la intrare are gradul n, divizarea va avea loc după un număr de tacte egal cu numărul coeficienților polinomului de la intrare, adică n+1 tacte. Pentru aflarea restului divizării mai este necesar un tact, deci un total de n+2 tacte. Coeficientul termenului x^{m-1} al restului va fi stocat în celula binară B_m, coeficientul termenului x^{m-2} în B_{m-1} ş.a.m.d., coeficientul lui x^0 în celula binară B_1. Evident, dacă în urma divizării rezultă un rest nul, toate celulele binare vor trece în starea "0" logic.

Inițial, intrarea şi toate celulele binare se aduc în starea "0" logic şi, conform convenției, polinomul de la intrare se aplică în ordinea descrescătoare a puterilor lui x.

Pentru fixarea ideilor, se presupune că la intrarea unui circuit de divizare cu sumatoarele modulo 2 plasate între celulele binare, întocmit după polinomul $g(x)=1\oplus x\oplus x^3$, se aplică polinomul $P(x)= 1\oplus x^5\oplus x^6$. Circuitul de divizare este reprezentat în figura 4.14.

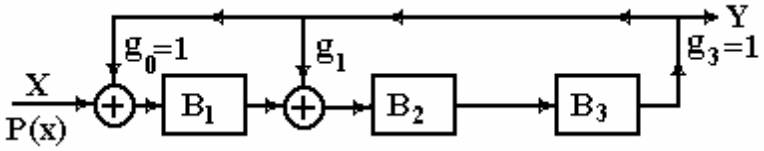

Fig.4.14. Circuit de divizare întocmit după polinomul $g(x)=1\oplus x\oplus x^3$.

Tact	X	B_1	B_2	B_3	Y
	0	0	0	0	
1	$x^6\rightarrow 1.$	0.	0.	0.	0
2	$x^5\rightarrow 1.$	1.	0.	0.	0
3	$x^4\rightarrow 0.$	1.	1.	0.	0
4	$x^3\rightarrow 0.$	0.	1.	1.	$1\ x^3$
5	$x^2\rightarrow 0.$	1.	1.	1.	$1\ x^2$
6	$x\rightarrow 0.$	1.	0.	1.	$1\ x^1$
7	$x^0\rightarrow 1.$	1.	0.	0.	$0\ x^0$
8		1	1	0	

Funcționarea circuitului este sintetizată în tabelul alăturat. Deoarece polinomul de la intrare are gradul n=6, câtul divizării se va obține în primele n+1=6+1=7 tacte, restul fiind obținut la tactul 8.

Din circuitul de divizare reprezentat în figura 4.14 rezultă că, la aplicarea tactului, celula binară B_1 va trece în starea corespunzătoare sumei modulo 2 a stărilor precedente ale intrării şi celulei B_3, celula binară B_2 va trece în starea corespunzătoare sumei modulo 2 a stărilor precedente ale celulelor binare B_1 şi B_3, celula binară B_3 va trece în starea precedentă a celulei binare B_2, iar ieşirea Y va fi identică cu starea lui B_3. Conform tabelului, rezultă câtul $q(x) = x^3\oplus x^2\oplus x$ şi restul $r(x) = x\oplus 1$, fapt ce poate fi verificat prin împărțirea polinomului de la intrare la polinomul g(x), după care s-a întocmit circuitul.

Se va arăta în continuare că circuitul reprezentat în figura 4.15 este un circuit de divizare întocmit după polinomul g(x), dar cu sumatoarele modulo 2 plasate în afara celulelor binare.

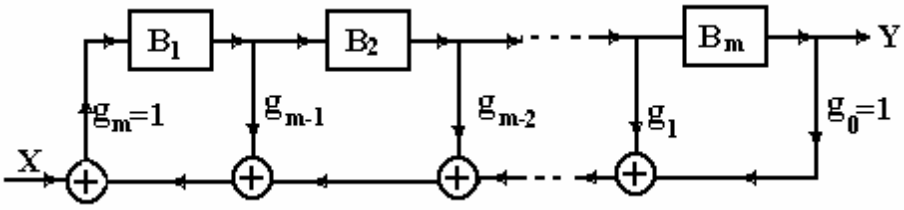

Fig.4.15. Circuit de divizare cu sumatoarele modulo 2 plasate înafara celulelor binare.

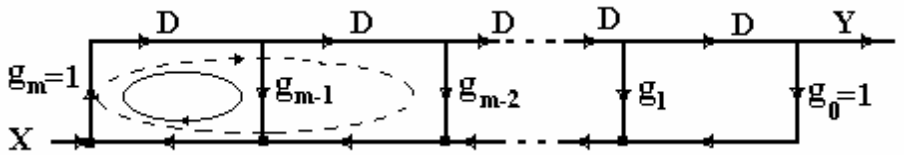

Fig.4.16. Graful ataşat circuitului de divizare din fig.4.15.

Graful ataşat acestui circuit este reprezentat în figura 4.16. Din acest graf se constată o singură cale de transmitanță calculată cu relația (4.127) şi buclele cu transmitanțele calculate cu relația (4.128). Deoarece, şi în acest caz, singura cale C_1 este adiacentă cu toate buclele şi toate buclele sunt adiacente între ele, rezultă că şi pentru acest circuit sunt adevărate relațiile (4.129) şi (4.131), deci şi acest circuit realizează operația de divizare. Spre deosebire de circuitul de divizare cu sumatoarele modulo 2 intercalate între celulele binare, în acest caz, dacă există rest diferit de zero, acesta este stocat în celulele B_1, B_2, ..., B_m într-o formă modificată. Ambele variante de circuite de divizare conțin un număr de celule binare egal cu gradul polinomului după care sunt întocmite.

4.15. Registre de deplasare cu reacție (R.D.R.) utilizate în codarea şi decodarea codurilor ciclice

Un registru de deplasare cu reacție se obține dintr-un circuit de divizare prin suprimarea intrării. Rezultă atunci că un R.D.R. este un circuit secvențial liniar care poate funcționa autonom, adică fără date aplicate din exterior.

Astfel, suprimând intrarea X din circuitul de divizare din figura 4.15, se obține R.D.R-ul reprezentat în figura 4.17.

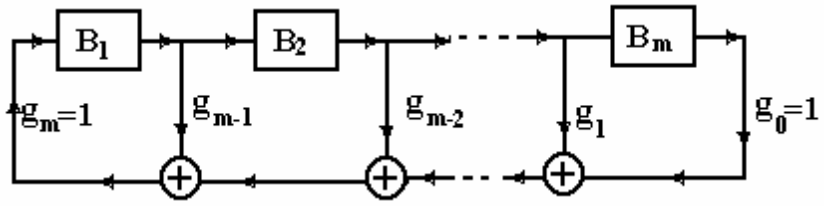

Fig.4.17. Registru de deplasare cu reacție.

Mulțimea stărilor celulelor la un moment dat defineşte *starea registrului* din acel moment.

Fie $s_1 \in \{0,1\}$ starea celulei binare B_1, $s_2 \in \{0,1\}$ starea celulei binare B_2 ş.a.m.d., $s_m \in \{0,1\}$ starea celulei binare B_m, toate la un anumit moment de timp. Această stare a R.D.R-ului se notează, de obicei matriceal, sub forma:

$$[S] = \begin{bmatrix} s_m \\ s_{m-1} \\ \cdots\cdots \\ s_2 \\ s_1 \end{bmatrix} \qquad (4.132)$$

Considerând starea R.D.R-ului dată de relația (4.132), după aplicarea primului tact stările celulelor binare devin $s_1' \in \{0,1\}$, $s_2' \in \{0,1\}, \ldots, s_m' \in \{0,1\}$.

Notând noua stare cu [S$^{'}$], se poate scrie:

$$[S^{'}] = \begin{bmatrix} s^{'}_{m} \\ s^{'}_{m-1} \\ \cdots\cdots \\ s^{'}_{2} \\ s^{'}_{1} \end{bmatrix}, \qquad (4.133)$$

Conform circuitului din figura 4.17, rezultă:

$$\left.\begin{aligned} s^{'}_{m} &= s_{m-1} \\ s^{'}_{m-1} &= s_{m-2} \\ &------ \\ s^{'}_{2} &= s_{1} \\ s^{'}_{1} &= g_0 s_m \oplus g_1 s_{m-1} \oplus \ldots \oplus g_{m-1} s_1 \end{aligned}\right\} \qquad (4.134)$$

Sistemul de ecuații (4.134) poate fi scris compact sub formă matriceală, astfel:

$$[S^{'}] = [T][S], \qquad (4.135)$$

unde:

$$[T] = \begin{bmatrix} 0 & 1 & 0 & \ldots & 0 \\ 0 & 0 & 1 & \ldots & 0 \\ - & - & - & - & - \\ 0 & 0 & 0 & \ldots & 1 \\ g_0 & g_1 & g_2 & \cdots & g_{m-1} \end{bmatrix} \qquad (4.136)$$

Matricea [T], definită cu relația (4.136), este cunoscută sub denumirea de *matricea conexiunilor* R.D.R-ului.

Din relația (4.135) se constată că, pentru ca o anumită stare [S$^{'}$] să provină dintr-o stare unică [S], trebuie ca matricea conexiunilor [T] să fie nesingulară, adică să admită inversă. (numai în acest caz se poate determina starea [S] din relația (4.135)). Pe de altă parte, din (4.136) se deduce uşor că matricea conexiunilor este nesingulară, dacă g_0=1. Pentru existența reacției este necesar, de asemenea, ca g_m=1. Aceasta este raţiunea pentru care în circuitele de divizare, deci implicit şi în cazul registrelor de deplasare cu reacție, s-a impus condiția g_0=g_m=1.

Dacă se notează cu [S(0)] şi [S(1)] starea inițială, respectiv starea după primul tact a R.D.R-ului, conform relației generale (4.135), se poate scrie:

$$[S(1)] = [T][S(0)] \qquad (4.137)$$

Notând cu [S(2)], [S(3)], ..., [S(i)] stările R.D.R-ului la tactele doi, trei, respectiv i, se pot scrie relațiile:

$$[S(2)] = [T][S(1)] = [T]^2[S(0)] \quad (4.138)$$

$$[S(3)] = [T]^3[S(0)] \quad (4.139)$$

- - - - - - - - - - - - - - - - - - -

$$[S(i)] = [T]^i[S(0)] \quad (4.140)$$

Deoarece R.D.R.-ul este format dintr-un număr finit de celule binare, numărul stărilor distincte ale acestuia va fi, de asemenea, finit. Dacă [S(0)] ≠ [0] este starea inițială, fie n numărul de tacte după care se revine în starea inițială.

Ținându-se cont de relația generală (4.140), se poate scrie în acest caz:

$$[S(n)] = [T]^n[S(0)] = [S(0)], \quad (4.141)$$

sau, echivalent:

$$\left[[I_m] \oplus [T]^n\right][S(0)] = [0], \quad (4.142)$$

unde $[I_m]$ este matricea unitate de ordinul m.

Deoarece starea inițială [S(0)] este oarecare şi diferită de starea cu toate celulele binare în "0" logic, rezultă că relația (4.142) va fi satisfăcută, dacă:

$$[T]^n = [I_m] \quad (4.143)$$

Numărul de tacte n pentru care este adevărată relația (4.141) sau, echivalent, (4.143) defineşte *perioada* R.D.R.-ului.

Pentru a stabili *perioada maximă* a unui R.D.R., se ține cont că acesta conține m celule binare. Deoarece fiecare celulă binară se poate afla în două stări ("0" logic sau "1" logic), rezultă un număr de stări distincte posibile, egal cu 2^m. Eliminând starea cu toate celulele binare în "0" logic, rezultă că perioada maximă a unui R.D.R. întocmit după un polinom g(x) de grad m este:

$$n_{max} = 2^m - 1 \quad (4.144)$$

Polinoamele g(x) care conduc la perioada maximă a R.D.R.-ului se numesc *polinoame primitive*. Se poate demonstra că polinoamele primitive sunt ireductibile şi divid polinomul $x^n \oplus 1$ pentru $n \geq 2^m-1$. Se poate, de asemenea, demonstra că pentru fiecare grad există cel puțin un polinom primitiv. În literatura de specialitate sunt listate polinoame primitive până la grade suficient de mari.

Pentru fixarea ideilor, se presupun două polinoame: $g_1(x)=1 \oplus x \oplus x^2 \oplus x^3$ şi $g_2(x)=1 \oplus x \oplus x^3$. Registrele de deplasare întocmite după aceste polinoame sunt reprezentate în fig.4.18, a, b.

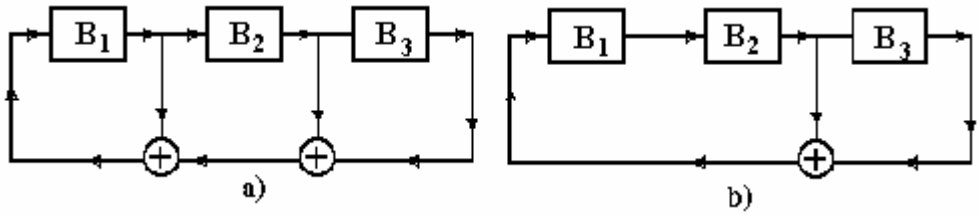

Fig.4.18. Registru de deplasare întocmit după polinomul $g_1(x)=1\oplus x\oplus x^2\oplus x^3$, a) şi după polinomul $g_2(x)=1\oplus x\oplus x^3$, b).

Pentru determinarea perioadei acestor registre de deplasare, se pleacă dintr-o stare inițială diferită de zero şi apoi se întocmesc tabele cu evoluția stărilor celulelor binare, aşa cum este arătat în continuare.

Tabel cu evoluția stărilor registrului întocmit după polinomul $g_1(x)=1\oplus x\oplus x^2\oplus x^3$

Tact	B_1	B_2	B_3
1	1	0	0
2	1	1	0
3	0	1	1
4	0	0	1
5	1	0	0

Tabel cu evoluția stărilor registrului întocmit după polinomul $g_2(x)=1\oplus x\oplus x^3$

Tact	B_1	B_2	B_3
1	1	0	0
2	0	1	0
3	1	0	1
4	1	1	0
5	1	1	1
6	0	1	1
7	0	0	1
8	1	0	0

Se constată că registrul de deplasare cu reacție întocmit după polinomul $g_1(x)$ are perioada n=4, în timp ce cel întocmit după polinomul $g_2(x)$ are perioada $n_{max} = 2^3-1 = 7$, deci polinomul $g_2(x)$ este primitiv.

Se poate demonstra că perioade R.D.R.-ului întocmit după un polinom primitiv are totdeauna valoarea maximă, indiferent de starea inițială, în timp ce, în cazul polinoamelor neprimitive, perioada R.D.R.-ului depinde de starea inițială şi nu este niciodată maximă. Câteva polinoame primitive: pentru gradul doi $x^2\oplus x\oplus 1$, pentru gradul 3 sunt două polinoame primitive $x^3\oplus x\oplus 1$ şi $x^3\oplus x^2\oplus 1$, pentru gradul 4 sunt de asemenea două polinoame primitive, şi anume $x^4\oplus x\oplus 1$ şi $x^4\oplus x^3\oplus 1$ ş.a.m.d.

4.16. Codor ciclic, corector de o eroare, realizat cu registre de deplasare cu reacție

Fie k numărul simbolurilor informaționale necesare transmiterii unei anumite informații. În scopul corecției unei erori, la cele k simboluri informaționale trebuie adăugate m simboluri de control calculate cu relația:

$$2^m \geq m + k + 1 \tag{4.145}$$

Se reaminteşte că în relația (4.145) se alege numărul întreg pozitiv m, cel mai mic, care o satisface.

Se alege un polinom primitiv de grad m. Fie acesta de forma:

$$\begin{aligned} &g(x) = g_0 \oplus g_1 x \oplus \ldots \oplus g_m x^m, g_0 = g_m = 1; \\ &g_i \in \{0,1\}, i = \overline{1,(m-1)} \end{aligned} \tag{4.146}$$

Codorul ciclic, corector de o eroare, realizat cu R.D.R., întocmit după polinomul g(x), este reprezentat în figura 4.19.

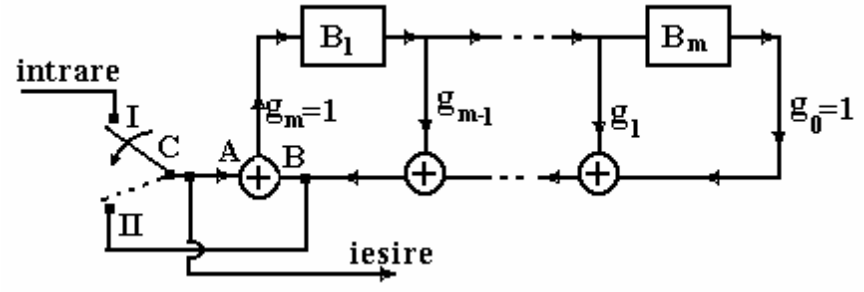

Fig.4.19. Codor ciclic, corector de o eroare, realizat cu R.D.R.

Inițial, comutatorul C este pe poziția I şi toate celulele binare se resetează. În acest caz, evident, circuitul funcționează ca un circuit de divizare cu sumatoarele modulo 2 plasate înafara celulelor binare.

Fie

$$\begin{aligned} v(x) = a_0 \oplus a_1 x \oplus \ldots \oplus a_{n-k-1} x^{n-k-1} \oplus a_{n-k} x^{n-k} \oplus \\ \oplus \ldots \oplus a_{n-1} x^{n-1} \end{aligned} \tag{4.147}$$

un cuvânt de cod ciclic sistematic, cu $(a_0, a_1, ..., a_{n-k-1}) \in \{0,1\}$ simboluri de control şi $(a_{n-k}, ..., a_{n-1}) \in \{0,1\}$ simbolurile informaționale cunoscute.

Deoarece v(x) este cuvânt de cod, el este divizibil la polinomul generator al codului, g(x). Evoluția stărilor R.D.R.-ului la aplicarea la intrare a cuvântului de cod v(x) poate fi descrisă comod, dacă se foloseşte matricea conexiunilor R.D.R.-ului, definită cu relația (4.136) şi se introduce matricea:

$$[U]=\begin{bmatrix} 0 \rightarrow B_m \\ 0 \rightarrow B_{m-1} \\ . \\ . \\ . \\ 0 \rightarrow B_2 \\ 1 \rightarrow B_1 \end{bmatrix} \tag{4.148}$$

care marchează faptul că toate celulele binare sunt în starea "0" logic, cu excepția celulei B_1 care se află în starea "1" logic.

Conform convenției, polinomul v(x) se aplică la intrarea circuitului în ordinea descrescătoare a puterilor lui x. Astfel, la primul tact starea R.D.R.-ului devine:

$$[S(1)]=a_{n-1}[U]=\begin{bmatrix} 0 \rightarrow B_m \\ 0 \rightarrow B_{m-1} \\ . \\ . \\ . \\ 0 \rightarrow B_2 \\ a_{n-1} \rightarrow B_1 \end{bmatrix} \tag{4.149}$$

La tactul al doilea, starea R.D.R.-ului se calculează cu relația:

$$[S(2)]=[T][S(1)]\oplus a_{n-2}[U], \tag{4.150}$$

sau, dacă se ține cont de (4.149), se poate scrie, echivalent, relația:

$$[S(2)]=a_{n-1}[T][U]\oplus a_{n-2}[U], \tag{4.151}$$

La tactul al treilea, starea R.D.R.-ului se deduce cu relația:

$$[S(3)]=[T][S(2)]\oplus a_{n-3}[U], \tag{4.152}$$

sau, dacă se ține cont de (4.151), se poate scrie, echivalent:

$$[S(3)]=a_{n-1}[T]^2[U]\oplus a_{n-2}[T][U]\oplus a_{n-3}[U]$$

Raționând în mod analog, se poate scrie că la tactul n starea R.D.R.-ului devine:

$$\begin{aligned}[S(n)]=a_{n-1}[T]^{n-1}[U]\oplus a_{n-2}[T]^{n-2}[U]\oplus \\ \oplus\ldots\oplus a_1[T][U]\oplus a_0[U]\end{aligned} \tag{4.153}$$

Deoarece polinomul v(x) este divizibil la g(x), starea [S(n)] trebuie să

$$\begin{aligned}a_{n-1}[T]^{n-1}[U]\oplus a_{n-2}[T]^{n-2}[U]\oplus \\ \oplus\ldots\oplus a_1[T][U]\oplus a_0[U]=[0]\end{aligned} \tag{4.154}$$

corespundă situației când toate celulele binare se află în starea "0" logic, adică:

Se fac notațiile:

$$[U, TU, \ldots T^{n-1}U] \underline{\underline{not}} [H] \tag{4.155}$$

$$[a_0\ a_1 \ldots a_{n-1}] \underline{\underline{not}} [v] \tag{4.156}$$

Cu notațiile (4.155) şi (4.156) relația (4.154) se poate scrie sub forma:

$$[H][v]^T = [0] \tag{4.157}$$

Deoarece relația (4.157) este identică cu relația (4.23), matricea [H] definită cu relația (4.155) se va numi *matricea de control a codurilor ciclice sistematice corectoare de o eroare*.

Trebuie subliniat faptul că relația (4.157), după care se implementează de fapt codorul, este adevărată pentru toate codurile bloc liniare. Diferența constă în structura matricei de control, care diferă de la un cod la altul. Aşa, de exemplu, în cazul codului Hamming corector de o eroare, matricea de control este dată de (4.86), în cazul codului Hamming corector de o eroare, detector de erori duble, de relația (4.97), iar în cazul codurilor ciclice corectoare de o eroare, de relația (4.155).

Ținând cont de analiza de mai sus, funcționarea codorului ciclic, corector de o eroare, realizat cu R.D.R., este următoarea:

În primele k tacte se aplică la intrarea circuitului simbolurile informaționale cunoscute care, pe de o parte rezultă direct la ieşire, iar pe de altă parte sunt introduse în R.D.R. prin intrarea notată cu A. Când ultimul simbol informațional este introdus în R.D.R., comutatorul C este trecut pe poziția II. În acest caz, intrările A şi B ale sumatorului modulo 2 devin identice (fie "0" logic, fie "1" logic), astfel încât la tactul k+1 la ieşire va rezulta simbolul de control a_{n-k-1}, iar celula binară B_1 va trece în starea "0" logic. La tactul k+2 la ieşire va rezulta cel de al doilea simbol de control, a_{n-k-2}, iar celulele binare B_1 şi B_2 vor trece în starea "0" logic. Raționând în mod analog, înseamnă că la tactul k+m=n la ieşire va rezulta ultimul simbol de control, a_0, în timp ce toate celulele binare din R.D.R. vor fi în starea "0" logic, respectându-se astfel relația (4.154) sau, echivalent, relația (4.157).

Cuvântul de cod astfel rezultat s-ar putea să aibă o lungime n, care nu satisface relația (4.105). Pentru satisfacerea acestei relații se pot adăuga zerouri la sfârşitul cuvântului de cod până când această relație este satisfăcută. În felul acesta, orice permutare ciclică a unui cuvânt de cod determină un nou cuvânt de cod. Deoarece la emisie această condiție nu este restrictivă, se poate transmite cuvântul de cod rezultat la ieşirea codorului chiar dacă relația (4.105) nu este îndeplinită. La recepție însă, lungimea cuvintelor trebuie să satisfacă această relație, motiv pentru care, la sfârşitul cuvântului vor fi adăugate atâtea zerouri, până când relația este satisfăcută.

Pentru fixarea ideilor, se presupune că se doreşte transmiterea numărului N=5 pe un canal pe care poate să apară o eroare, utilizând un cod ciclic, corector de o eroare, realizat cu R.D.R. Numărul simbolurilor informaționale se determină cu relația

$2^k>5$, rezultând k=3. Cele trei simboluri informaționale rezultă prin transcrierea binară a numărului 5, adică $(5)_{10} = (101)_2$. Numărul simbolurilor de control necesare se determină din relația $2^m \geq m+3+1$, rezultând m=3. Se alege polinomul primitiv $g(x)=1 \oplus x \oplus x^3$. Codorul ciclic este reprezentat în figura 4.20.

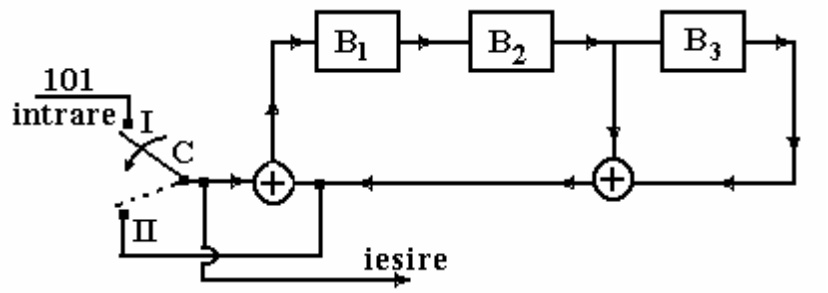

Fig.4.20. Codor ciclic corector de o eroare, întocmit după polinomul $1 \oplus x \oplus x^3$.

Funcționarea codorului din figura 4.20 poate fi sintetizată într-un tabel în care se vor pune în evidență evoluția stărilor celulelor binare şi a ieşirii, aşa cum este arătat mai jos.

Tact	intrare	B_1	B_2	B_3	ieşire
	0.	0	0.	0	
1	1.	0	0.	0.	1
2	0.	1	0.	0.	0
3	1.	0	1.	0.	1
4		0	0	1	1
5		0	0	0	0
6		0	0	0	0

Inițial, toate celulele binare sunt resetate şi comutatorul C este pe poziția I. În această situație, la un anumit tact, celula binară B_1 va trece în starea corespunzătoare sumei modulo 2 dintre stările precedente ale intrării şi ale celulelor binare B_2 şi B_3. Celula binară B_2 va trece în starea precedentă a lui B_1, iar B_3 în starea precedentă a lui B_2. La ieşire vor rezulta direct simbolurile informaționale aplicate la intrare. După introducerea ultimului simbol informațional în R.D.R., (tactul 4), comutatorul C este trecut în poziția II. În această situație, la ieşire va rezulta suma modulo 2 a stărilor curente ale celulelor binare B_2 şi B_3, în timp ce la intrarea celulei binare B_1 se va aplica "0" logic.

Cuvântul de cod este [v] = [0 0 1 1 0 1], în care primele trei

$$\downarrow \downarrow \quad \downarrow$$
$$x^0\ x^1 \ldots\ldots x^5$$

simboluri binare sunt de control, în timp ce ultimele trei, simbolurile informaționale cunoscute. Sub formă polinomială, cuvântul de cod este $v(x) = x^2 \oplus x^3 \oplus x^5$

4.17. Decodor ciclic, corector de o eroare, realizat cu registre de deplasare cu reacție

Fie

$$\mathbf{v'(x) = a'_0 \oplus a'_1 x \oplus \ldots \oplus a'_{n_1-1} x^{n_1-1}} \qquad \textbf{(4.158)}$$

cuvântul recepționat, scris sub formă polinomială.

Cuvântul recepționat, scris sub formă matriceală, este:

$$\mathbf{[v'] = [a'_0\ a'_1 \ldots a'_{n_1-1}]} \qquad \textbf{(4.159)}$$

Dacă relația (4.105) nu este verificată, adică dacă:

$$\mathbf{n_1 < 2^m - 1,} \qquad \textbf{(4.160)}$$

la sfârşitul cuvântului recepționat se va adăuga un număr de zerouri, până când relația (4.105) este verificată. Cuvântul recepționat, cu eventuale zerouri adăugate la sfârşit, va fi de forma:

$$\mathbf{v'(x) = a'_0 \oplus a'_1 x \oplus \ldots \oplus a'_{n_1-1} x^{n_1-1} \oplus a'_{n_1} x^{n_1} \oplus \ldots \oplus a'_{n-1} x^{n-1}} \qquad \textbf{(4.161)}$$

unde $\mathbf{a'_{n_1} = \ldots = a'_{n-1} = 0}$ şi $n = 2^m-1$.

Schema bloc a decodorului ciclic, corector de o eroare, realizat cu R.D.R., este dată în figura 4.21.

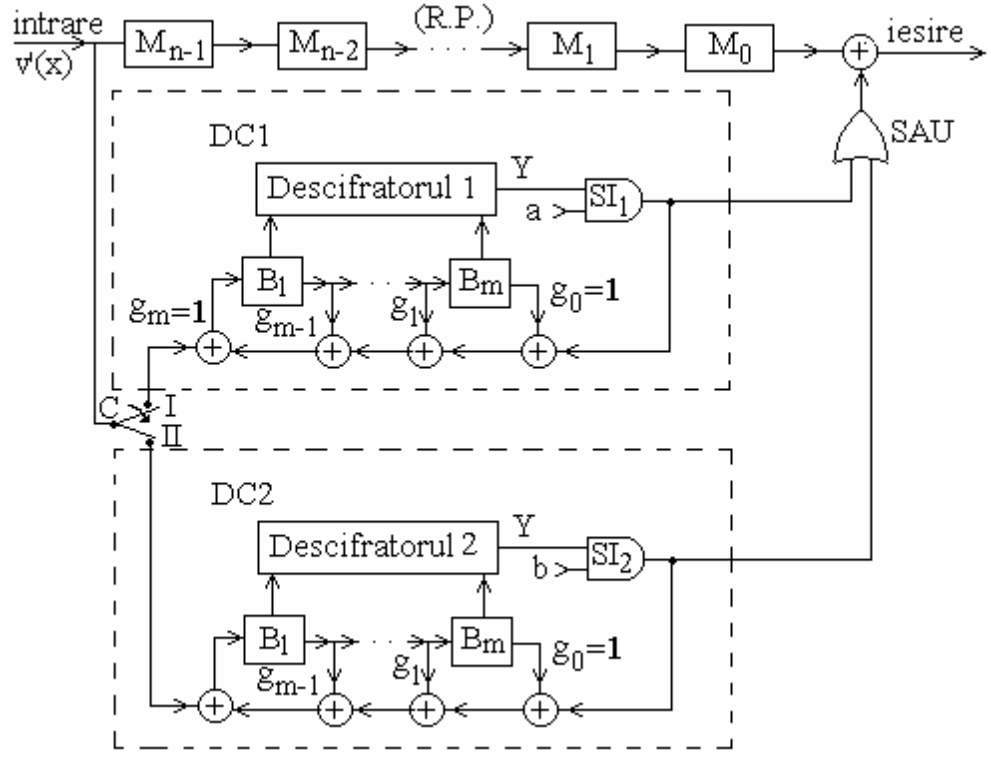

Fig.4.21. Decodor ciclic realizat cu registre de deplasare cu reacție.

Decodorul ciclic conține un registru principal (R.P.) format din n celule binare în care se vor stoca simbolurile cuvântului recepționat, eventual şi zerourile adăugate suplimentar la sfârşitul cuvântului.

Decodorul mai conține două blocuri identice (DC_1 şi DC_2), formate, fiecare, dintr-un R.D.R., întocmit după acelaşi polinom generator al codului g(x), cu care s-a realizat codarea la emisie şi un descifrator astfel sintetizat, încât să furnizeze "1" logic,

la ieşire atunci când simbolul eronat ajunge în ultima celulă binară M_0 a registrului principal.

Unitatea logică furnizată de descifrator se adună modulo 2 la simbolul recepționat eronat, furnizându-l corectat la ieşire.

Inițial, comutatorul C este pe poziția I şi toate celulele binare se resetează.

Conform convenției, cuvântul $v'(x)$, dat de relația (4.161), se introduce în decodor în ordinea descrescătoare a puterilor lui x. La primul tact, simbolul $\mathbf{a'_{n-1}}$ va fi memorat în celula binară M_{n-1} pe de o parte, iar pe de altă parte în R.D.R.-ul din DC1, determinându-i starea:

$$[S'(1)] = a'_{n-1}[U] = \begin{bmatrix} 0 \rightarrow B_m \\ 0 \rightarrow B_{m-1} \\ . \\ . \\ . \\ 0 \rightarrow B_2 \\ a'_{n-1} \rightarrow B_1 \end{bmatrix} \qquad (4.162)$$

La al doilea tact, la intrare se aplică simbolul $\mathbf{a'_{n-2}}$, astfel încât în celulele binare M_{n-2} şi
M_{n-1} vor fi stocate simbolurile a'_{n-1}, respectiv a'_{n-2}, în timp ce starea R.D.R.-ului din DC1 va deveni:

$$[S'(2)] = [T][S'(1)] \oplus a'_{n-2}[U], \qquad (4.163)$$

sau, ținând cont de (4.162), se poate scrie echivalent:

$$[S'(2)] = a'_{n-1}[T][U] \oplus a'_{n-2}[U] \qquad (4.164)$$

La al treilea tact, celulele binare M_{n-3}, M_{n-2} şi M_{n-1} vor memora simbolurile a'_{n-1}, a'_{n-2}, respectiv a'_{n-3}, în timp ce R.D.R.-ul din DC1 va trece în starea:

$$[S'(3)] = [T][S'(2)] \oplus a'_{n-3}[U], \qquad (4.165)$$

sau, ținând cont de (4.164):

$$[S'(3)] = a'_{n-1}[T]^2[U] \oplus a'_{n-2}[T][U] \oplus a'_{n-3}[U] \qquad (4.166)$$

Raționând în mod analog, la tactul n celula binară M_0 va memora simbolul a'_{n-1}, M_1 simbolul a'_{n-2} ş.a.m.d., celula M_{n-1} simbolul a'_0, în timp ce starea R.D.R.-ului din DC1 va deveni:

$$[S'(n)] = a'_{n-1}[T]^{n-1}[U] \oplus a'_{n-2}[T]^{n-2}[U] \oplus \ldots \oplus \\ \oplus a'_1[T][U] \oplus a'_0[U] \qquad (4.167)$$

Dacă se fac notațiile:

$$[a'_0\ a'_1 \ldots a'_{n-1}] \underline{\underline{\text{not.}}} [v'] \qquad (4.168)$$

$$[U, TU, \ldots T^{n-1}U] \underline{\underline{\text{not.}}} [H], \tag{4.169}$$

relația (4.167) poate fi scrisă compact, sub forma matriceală:

$$[S'(n)] = [H][v']^T \tag{4.170}$$

Comparând relația (4.42) cu (4.170), rezultă că starea R.D.R.-ului, la recepționarea întregului cuvânt coincide cu expresia corectorului cuvântului respectiv.

După recepționarea primului cuvânt, comutatorul C este trecut pe poziția II, simultan efectuându-se corecția primului cuvânt şi recepția celui de al doilea. După recepționarea celui de al doilea cuvânt, comutatorul C este trecut din nou pe poziția I, efectuându-se simultan corecția celui de al doilea cuvânt şi recepția celui de al treilea ş.a.m.d. Cu alte cuvinte, comutatorul C va fi pe poziția I la recepționarea cuvintelor numerotate cu numere impare şi va fi pe poziția II la recepționarea cuvintelor numerotate cu numere pare.

Aşa cum s-a precizat, descifratoarele trebuie astfel proiectate, încât să furnizeze o unitate logică la ieşire atunci când simbolul eronat din cuvântul recepționat este memorat în ultima celulă binară a registrului principal, M_0. Pentru a nu rezulta o unitate logică falsă la ieşirea descifratoarelor, s-au conectat două porți logici ŞI, notate cu $ŞI_1$, respectiv $ŞI_2$, validate de intrările a şi b. Când comutatorul C se află pe poziția I, sincron se realizează a = "0" logic şi b = "1" logic, invalidându-se astfel poarta logică $ŞI_1$ şi validându-se poarta logică $ŞI_2$. În felul acesta, la recepționarea unui cuvânt numerotat cu un număr impar, dacă ar rezulta o unitate logică la ieşirea descifratorului din DC1, aceasta nu va putea trece de poarta logică $ŞI_1$, evitându-se astfel sumarea modulo 2 a acestei unități logice la un simbol din cuvântul precedent. În mod similar, când comutatorul C este pe poziția II, rezultă a = 1 şi b = 0, poarta logică $ŞI_2$ fiind invalidată. O eventuală unitate logică rezultată în timpul recepționării unui cuvânt numerotat cu un număr par nu se mai sumează modulo 2 cu un simbol din cuvântul precedent. Cu alte cuvinte, datorită porților $ŞI_1$, respectiv $ŞI_2$, descifratorul 1 va corecta eventualele erori din cuvintele numerotate cu numere impare, iar descifratorul 2 eventualele erori din cuvintele numerotate cu numere pare.

În principiu, decodorul ar putea funcționa cu un singur decodor, dar în acest caz va trebui să se aştepte un timp corespunzător a n tacte, timp în care se corectează cuvântul recepționat şi se asigură pregătirea pentru recepționarea următorului cuvânt.

4.18. Sinteza descifratoarelor din decodorul ciclic, corector de o eroare realizat cu registre de deplasare

Fără a micşora generalitatea, se presupune că simbolul eronat este $\mathbf{a}'_{n-r}$, oricare ar fi r cuprins între 1 şi n.

Cuvântul recepționat va fi atunci de forma:

$$v'(x) = a_0 \oplus a_1 x \oplus \ldots \oplus (a_{n-r} \oplus 1) x^{n-r} \oplus \ldots \oplus a_{n-1} x^{n-1} \qquad (4.171)$$

Conform relației (4.167), starea R.D.R.-ului se calculează în acest caz cu relația:

$$[S'(n)] = a_{n-1}[T]^{n-1}[U] \oplus a_{n-2}[T]^{n-2}[U] \oplus \ldots \oplus a_{n-r}[T]^{n-r}[U] \oplus \oplus [T]^{n-r}[U] \oplus \ldots \oplus a_0[U] \qquad (4.172)$$

Ținând cont că la emisie codorul realizează condiția stabilită de relația (4.154), rezultă că (4.172) se poate scrie, echivalent, sub forma:

$$[S'(n)] = [T]^{n-r}[U] \qquad (4.173)$$

Simbolul presupus eronat va fi memorat în ultima celulă binară M_0, a registrului principal, după r-1 tacte.

Starea R.D.R.-ului după r-1 tacte va deveni:

$$[S_1] = [T]^{r-1}[S'(n)],$$

sau, ținând cont de (4.173), rezultă:

$$[S_1] = [T]^{n-1}[U] \qquad (4.174)$$

Deoarece relația (4.105) este satisfăcută, rezultă că şi (4.143) este adevărată.

Cu (4.143), relația (4.174) devine:

$$[S_1] = [T]^{-1}[U] \qquad (4.175)$$

Din relația (4.175) se constată că starea R.D.R.-ului atunci când simbolul eronat este memorat în ultima celulă binară a registrului principal nu depinde de poziția erorii. Acest fapt remarcabil permite sinteza descifratorului, independent de poziția în care s-a introdus o eroare în cuvântul recepționat.

Pentru a putea deduce starea $[S_1]$ din relația (4.175), va trebui mai întâi să se determine $[T]^{-1}$. În acest scop, se pleacă de la relația evidentă:

$$[T][T]^{-1} = [I_m], \qquad (4.176)$$

unde $[I_m]$ este matricea unitate de ordinul m.

Notând cu $[l_1]$, $[l_2]$, ..., $[l_m]$ liniile matricei necunoscute $[T]^{-1}$ şi ținând cont de relația (4.136), rezultă:

$$\begin{bmatrix} 0 & 1 & 0 & \ldots & 0 \\ 0 & 0 & 1 & \ldots & 0 \\ - & - & - & - & - \\ 0 & 0 & 0 & \ldots & 1 \\ g_0 & g_1 & g_2 & \cdots & g_{m-1} \end{bmatrix} \begin{bmatrix} l_1 \\ l_2 \\ - \\ l_{m-1} \\ l_m \end{bmatrix} = [I_m] \qquad (4.177)$$

Din (4.177) se pot calcula liniile matricei $[T]^{-1}$, după cum urmează

$$\left.\begin{array}{l}[I_2]=[1\ 0\ldots0\ 0]_{(1xn)}\\ [I_3]=[0\ 1\ldots0\ 0]_{(1xn)}\\ ----------\\ [I_m]=[0\ 0\ldots1\ 0]_{(1xn)}\end{array}\right\} \quad (4.178)$$

şi

$$g_0[I_1]\oplus g_1[I_2]\oplus\ldots\oplus g_{m-1}[I_m]=[0\,0\ldots0\,1]_{(1xn)} \quad (4.179)$$

Ţinând cont de (4.178) şi de faptul că totdeauna g_0=1, din (4.179) rezultă:

$$[I_1]=[g_1\ g_2\ \cdots g_{m-1}\ 1] \quad (4.180)$$

Cu (4.178) şi (4.180) se poate scrie:

$$[T]^{-1}=\begin{bmatrix} g_1\ g_2\cdots g_{m-1}\ 1\\ 1\ \ 0\ \ldots 0\ \ \ 0\\ 0\ \ 1\ \ldots 0\ \ \ 0\\ -\ -\ -\ -\ -\ -\ -\\ 0\ \ 0\ \ldots 1\ \ \ 0\end{bmatrix} \quad (4.181)$$

Înlocuind (4.148) şi (4.181) în (4.175), se deduce starea pe care trebuie să o sesizeze descifratorul:

$$[S_1]=\begin{bmatrix} 1\rightarrow B_m\\ 0\rightarrow B_{m-1}\\ .\\ .\\ .\\ 0\rightarrow B_2\\ 0\rightarrow B_1\end{bmatrix} \quad (4.182)$$

Din relaţia (4.182) rezultă că descifratorul va fi o poartă logică ŞI cu m intrări, aşa cum este reprezentat în figura 4.22.

La tactul următor, unitatea logică furnizată la ieşirea descifratorului (poarta logică ŞI cu m intrări) este, pe de o parte, sumată la simbolul recepţionat eronat, corectându-l, iar pe de altă parte, este introdusă în R.D.R., determinându-i starea:

$$[S_0]=[T][S_1]\oplus[U] \quad (4.183)$$

Înlocuind (4.136) şi (4.182) în relaţia (4.183), rezultă:

$$[S_0]=[U]\oplus[U]=[0] \quad (4.184)$$

adică registru de deplasare cu reacție este adus în starea cu toate celulele binare în zero logic, fiind astfel pregătit pentru recepționarea altui cuvânt.

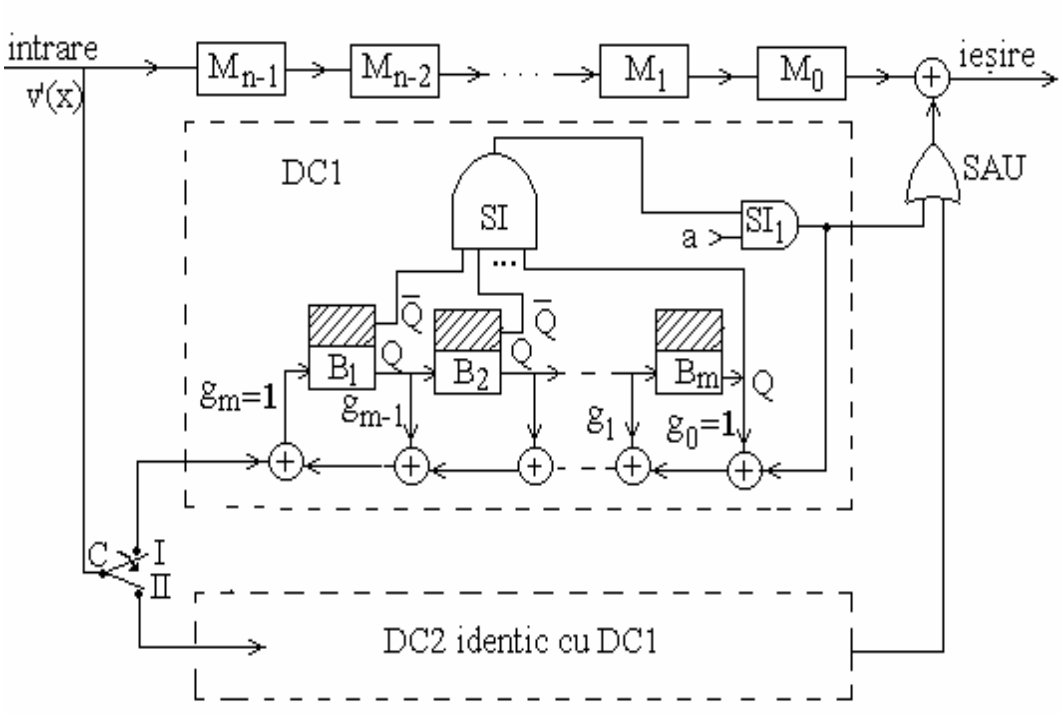

Fig.4.22. Decodor ciclic, corector de o eroare, realizat cu registre de deplasare cu reacție.

Comutatorul C poate fi implementat fizic aşa cum este arătat în figura 4.23.

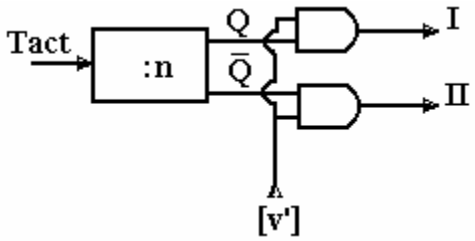

Fig.4.23. Implementarea comutatorului C din decodorul ciclic din fig.4.22.

În figura 4.23, prin blocul : n s-a specificat un divizor cu n. Dacă inițial Q=1 şi $\bar{Q}$=0, cuvântul recepționat va fi furnizat la ieşirea I. După n tacte, când se trece la recepționarea următorului cuvânt, rezultă Q=0 şi $\bar{Q}$=1, acesta fiind furnizat la ieşirea II.

Pentru fixarea ideilor, se presupune că s-a recepționat cuvântul $v'(x)=x^2 \oplus x^5$ pe un canal pe care poate să apară cel mult o eroare, iar polinomul generator al codului este $g(x)=1 \oplus x \oplus x^3$. Structura decodorului se poate deduce în acest caz particular, având în vedere că structura matriceală a cuvântului recepționat este [v'] = [0 0 1 0 0 1], deci n_1=6, iar gradul polinomului genarator al codului fiind m=3, înseamnă că 2^m-1

= 2^3-1=7. Rezultă că pentru satisfacerea relației (4.105), la sfârşitul cuvântului recepționat mai trebuie adăugat un zero, adică se consideră $[v^{'}] = [0\ 0\ 1\ 0\ 0\ 1\ 0]$.

Registrul principal va conține 7 celule binare, iar decodorul va avea structura reprezentată în figura 4.24.

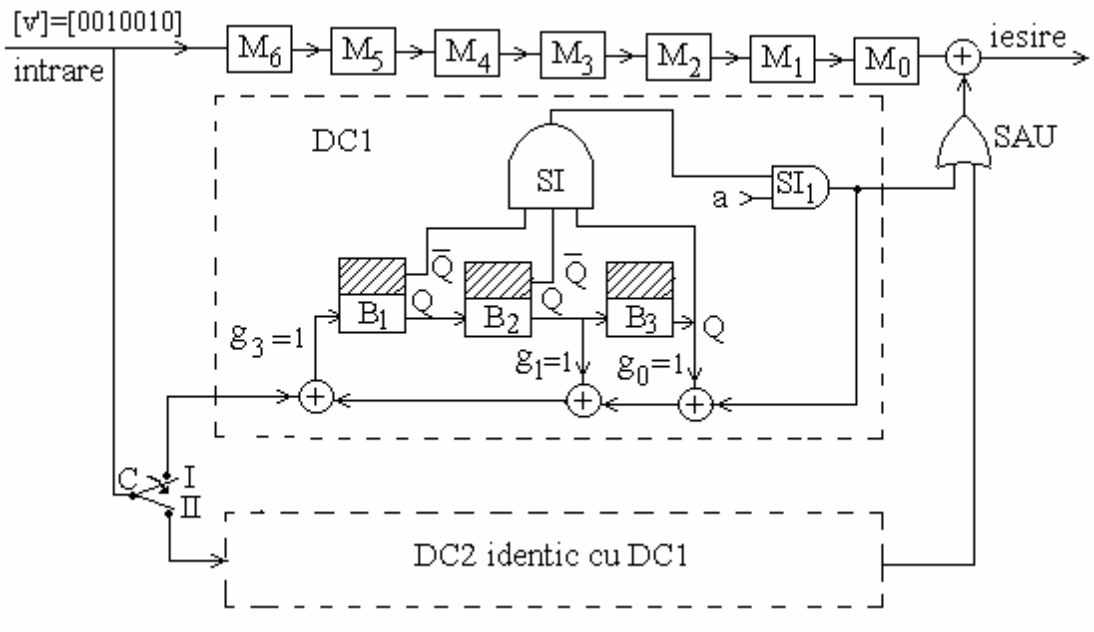

Fig.4.24. Decodor ciclic, corector de o eroare, realizat cu R.D.R., după polinomul $g(x) = 1 \oplus x \oplus x^3$.

Funcționarea decodorului din figura 4.24 este sintetizată în tabelul de mai jos.

Tact	Intr.	M_6	M_5	M_4	M_3	M_2	M_1	M_0	B_1	B_2	B_3	Ieşire
	0	0	0	0	0	0	0	0	0	0	0	
1	0	0	0	0	0	0	0	0	0	0	0	
2	1	0	0	0	0	0	0	0	0	0	0	
3	0	1	0	0	0	0	0	0	1	0	0	
4	0	0	1	0	0	0	0	0	0	1	0	
5	1	0	0	1	0	0	0	0	1	0	1	
6	0	1	0	0	1	0	0	0	0	1	0	
7	0	0	1	0	0	1	0	0	1	0	1	
8	0	0	0	1	0	0	1	0	1	1	0	0
9	0	0	0	0	1	0	0	1	1	1	1	1
10	0	0	0	0	0	1	0	0	0	1	1	0
11	0	0	0	0	0	0	1	0	0	0	1	1
12	0	0	0	0	0	0	0	1	0	0	0	1
13	0	0	0	0	0	0	0	0	0	0	0	0
14	0	0	0	0	0	0	0	0	0	0	0	0

Din tabel rezultă cuvântul corectat [0 0 1 1 0 1 0].

Inițial, toate celulele binare sunt resetate. Pentru simplificarea scrierii, s-a considerat că următorul cuvânt recepționat este format din 7 zerouri (tactele 8÷14). La tactul 8, când ultimul simbol din primul cuvânt recepționat este introdus în R.D.R., comutatorul C este trecut pe poziția II. La tactul 11, la ieşirea porții logice ŞI va rezulta "1" logic care, pe de o parte, sumat modulo 2 cu "0" logic memorat în M_0 va determina corectarea acestui simbol, iar, pe de altă parte, va determina trecerea celulelor binare B_1, B_2 şi B_3 în "0" logic, pregătind astfel R.D.R.-ul pentru recepționarea altui cuvânt. Dacă se înlătură simbolul "0" care a fost adăugat suplimentar, rezultă că s-a transmis cuvântul de cod [v] = [0 0 1 1 0 1]. Deoarece polinomul generator al codului are gradul m=3, rezultă că primele 3 simboluri din cuvântul transmis sunt simboluri de control, iar următoarele 3 simboluri informaționale. Trecând în zecimal numărul binar corespunzător simbolurilor informaționale, rezultă că s-a transmis numărul zecimal $(101)_2 = (5)_{10}$. Dacă nu s-ar fi efectuat corecția erorii, s-ar fi decis că s-a transmis numărul $(001)_2 = (1)_{10}$.

4.19. Definirea matricei generatoare şi de control în cazul codurilor ciclice

Fie

$$\begin{aligned} &\mathbf{g(x) = g_0 \oplus g_1 x \oplus \ldots \oplus g_m x^m, g_0 = g_m = 1;} \\ &\mathbf{g_i \in \{0,1\}, i = \overline{1,(m-1)}} \end{aligned} \qquad \textbf{(4.185)}$$

polinomul generator al codului şi

$$\mathbf{i(x) = i_0 \oplus i_1 x \oplus \ldots \oplus i_{k-1} x^{k-1}, i_j \in \{0,1\}, j = \overline{0,(k-1)}} \qquad \textbf{(4.186)}$$

polinomul informaţional.

Conform definiției cuvintelor de cod ciclice, acestea sunt polinoame v(x), de grad n-1 sau mai mic, divizibile la polinomul generator al codului. Un astfel de polinom ar putea fi, de exemplu:

$$\mathbf{v(x) = i(x) \cdot g(x),} \qquad \textbf{(4.187)}$$

Înlocuind (4.186) în relația (4.187), rezultă:

$$\mathbf{v(x) = i_0 g(x) \oplus i_1 x g(x) \oplus \ldots \oplus i_{k-1} x^{k-1} g(x)} \qquad \textbf{(4.188)}$$

Deoarece g(x) satisface definiția cuvintelor de cod ciclice, rezultă că şi xg(x), x^2g(x), ..., x^{k-1}g(x) sunt, de asemenea, cuvinte de cod şi, conform relației (4.188), şi orice combinație liniară a acestora determină un nou cuvânt de cod ciclic.

Pe de altă parte, conform paragrafului 4.3, mulțimea cuvintelor de cod în cazul codurilor bloc liniare, deci, în particular, şi în cazul codurilor ciclice, formează spațiul linie al matricei generatoare.

Ținând cont de (4.188), rezultă atunci că matricea generatoare a unui cod ciclic se poate scrie sub forma:

$$[G] = \begin{bmatrix} g(x) \\ xg(x) \\ . \\ . \\ . \\ x^{k-1}g(x) \end{bmatrix} = \begin{bmatrix} g_0 & g_1 & \ldots & g_m & 0 & 0 & \ldots & 0 & 0 \\ 0 & g_0 & \ldots & g_{m-1} & g_m & 0 & \ldots & 0 & 0 \\ . & . & . & . & . & . & . & . & . \\ 0 & 0 & 0 & 0 & 0 & \ldots & g_0 & \ldots & g_m \end{bmatrix}$$

$$\begin{matrix} \downarrow & \downarrow & \cdots & \downarrow & & & \cdots & \downarrow & \\ x^0 & x^1 & \cdots & x^m & & & \cdots & x^{m+k-1} = x^{n-1} & \end{matrix} \qquad \textbf{(4.189)}$$

Se reaminteşte că matricea generatoare are un număr de linii egal cu numărul simbolurilor informaționale şi un număr de coloane egal cu lungimea cuvintelor de cod.

Dacă se consideră că prima coloană a matricei generatoare corespunde lui x^0, a doua lui x^1 ş.a.m.d., ultima coloană lui $x^{m+k-1}=x^{n-1}$, atunci prima linie a matricei generatoare este formată din coeficienţii polinomului generator, restul coeficienţilor corespunzători lui x^{m+1} până la x^{n-1} completându-se cu zerouri. A doua linie a matricei generatoare reprezintă coeficienţii polinomului xg(x), adică prima permutare ciclică la dreapta a primei linii, a treia linie coeficienţii polinomului $x^2g(x)$ ş.a.m.d., ultima linie coeficienţii polinomului $x^{k-1}g(x)$, adică prima permutare ciclică la dreapta a penultimei linii, sau, echivalent, permutarea ciclică de ordinul k-1 a primei linii. Se poate verifica că, dacă în relaţia (4.26) se înlocuieşte matricea generatoare cu (4.189), elementele matricei [v] reprezintă coeficienţii puterilor lui x, determinaţi cu relaţia (4.187). Codul ciclic astfel obţinut este nesistematic. Pentru a obţine un cod ciclic sistematic, cu simbolurile informaţionale plasate grupat la sfârşitul cuvintelor de cod, matricea generatoare trebuie adusă la forma:

$$[G]=\begin{bmatrix} p_{11}p_{12}\ldots p_{1m}\ 1\ 0\ldots 0 \\ p_{21}p_{22}\ldots p_{2m}\ 0\ 1\ldots 0 \\ - - - - - - - - - - - - \\ p_{k1}p_{k2}\ldots p_{km}\ 0\ 0\ldots 1 \end{bmatrix}=[PI_k], p_{ij}\in\{0,1\}. \quad \text{(4.190)}$$

$$\downarrow\ \downarrow\quad \downarrow\ \downarrow\ \downarrow$$

$$x^0\ x^1\ \ldots x^{m-1}\ x^m\ \ldots x^{n-1}$$

Deoarece nu totdeauna este uşor de a determina transformările elementare ce trebuie efectuate asupra matricei [G], date de relaţia (4.189), pentru a fi adusă la forma (4.190), se va indica în continuare o metodă sistematică de întocmire a matricei generatoare, direct sub forma (4.190).

Pentru formarea primei linii a unei astfel de matrice generatoare, se împarte x^m la polinomul generator al codului g(x).

Dacă $r_1(x)$ este restul acestei divizări, se poate scrie:

$$\mathbf{x^m = g(x)\oplus r_1(x),} \quad \text{(4.191)}$$

sau, echivalent,

$$\mathbf{r_1(x)\oplus x^m = g(x)} \quad \text{(4.192)}$$

Conform relaţiei (4.192), rezultă că membrul stâng al acestei relaţii satisface definiţia cuvintelor de cod ciclice de a fi divizibile la polinomul generator al codului. Înseamnă atunci că p_{11}, p_{12}, ..., p_{1m} vor fi coeficienţii lui $r_1(x)$, iar unitatea coeficientul lui x^m.

Pentru a obţine elementele celei de a doua linii a matricei generatoare definite cu relaţia (4.190), se împarte x^{m+1} la g(x). Dacă se notează cu $r_2(x)$ restul acestei divizări, se poate scrie:

$$\mathbf{x^{m+1} = (a_0\oplus x)g(x)\oplus r_2(x),\ a_0\in\{0,1\},} \quad \text{(4.193)}$$

sau, echivalent

$$\mathbf{r_2(x) \oplus x^{m+1} = (a_0 \oplus x)g(x)} \quad \textbf{(4.194)}$$

Polinomul din membrul stâng al relației (4.194) fiind un polinom de grad n-1, sau mai mic, divizibil la polinomul generator al codului g(x), satisface definiția cuvintelor de cod şi deci coeficienții acestuia reprezintă a doua linia a matricei generatoare în care p_{21}, p_{22}, ..., p_{2m} sunt coeficienții restului $r_2(x)$. Procedând în mod analog, ultima linie a matricei generatoare se obține divizând x^{m+k-1} la g(x). Dacă se notează cu $r_k(x)$ restul acestei divizări, se poate scrie:

$$\mathbf{x^{m+k-1} = q(x)g(x) \oplus r_k(x),} \quad \textbf{(4.195)}$$

sau, echivalent

$$\mathbf{r_k(x) \oplus x^{m+k-1} = q(x)g(x),} \quad \textbf{(4.196)}$$

unde q(x) este câtul divizării.

Polinomul $r_k(x) \oplus x^{m+k-1}$ fiind cuvânt de cod, înseamnă că p_{k1}, p_{k2}, ..., p_{km} sunt coeficienții restului $r_k(x)$.

Odată adusă matricea generatoare la forma (4.190), cu ajutorul relației (4.31) se poate deduce matricea de control, dată de relația (4.24).

4.20. Decodor ciclic cu logică de prag

În cazul codurilor ciclice, implementarea decodorului poate fi simplificată, dacă acesta se realizează după relația:

$$\mathbf{[A] = [H_R][v']^T,} \quad \textbf{(4.197)}$$

sau, echivalent

$$\mathbf{[A] = [H_R][E]^T,} \quad \textbf{(4.198)}$$

unde *matricea redusă* $[H_R]$ se obține prin transformări elementare asupra matricei de control [H] a codului ciclic şi, eventual, reduceri de linii, astfel încât matricea redusă să se bucure de următoarele proprietăți:

a) ultima coloană să fie formată numai din unități;

b) celelalte coloane să conțină o singură unitate, indiferent de linia în care aceasta apare.

Fie

$$[H_R] = \begin{bmatrix} l_{1,0}\ l_{1,1}\ \ldots l_{1,n-1} \\ l_{2,0}\ l_{2,1}\ \ldots l_{2,n-1} \\ - - - - - - - - \\ l_{j,0}\ l_{j,1}\ \ldots l_{j,n-1} \\ - - - - - - - - \\ l_{J,0}\ l_{J,1} \ldots l_{J,n-1} \end{bmatrix}, l_{j,i} \in \{0,1\};\ j = \overline{1,J};\ i = \overline{0,(n-1)} \qquad (4.199)$$

matricea redusă, obținută prin transformări elementare asupra matricei de control şi, eventual, reduceri de linii, J≤m, cu

$$l_{j,n-1} = 1,\ j = \overline{1,J}, \qquad (4.200)$$

iar în celelalte coloane să existe o singură unitate.

Dacă

$$[v'] = [a'_0 a'_1 \ldots a'_{n-1}] \qquad (4.201)$$

este cuvântul recepționat, respectiv:

$$[E] = [e_0 e_1 \ldots e_{n-1}] \qquad (4.202)$$

este cuvântul eroare corespunzător, din relațiile (4.197) şi (4.198) rezultă:

$$A_j = l_{j,0} a'_0 \oplus l_{j,1} a'_1 \oplus \ldots \oplus l_{j,n-1} a'_{n-1},\ (\forall) j = \overline{1,J}, \qquad (4.203)$$

$$A_j = l_{j,0} e_0 \oplus l_{j,1} e_1 \oplus \ldots \oplus l_{j,n-1} e_{n-1},\ (\forall) j = \overline{1,J}, \qquad (4.204)$$

unde A_1, A_2, ..., A_J sunt elementele matricei [A].

Datorită celor două proprietăți ale matricei reduse $[H_R]$, rezultă că toate simbolurile cuvântului recepționat sunt incluse în cele J sume A_j şi fiecare simbol din cuvântul recepționat intervine o singură dată în sumele A_j, cu excepția simbolului a'_{n-1} care intervine în toate cele J sume. Din acest motiv, sumele A_j se numesc *ortogonale* pe simbolul a'_{n-1}. Deoarece în cazul codurilor ciclice orice permutare ciclică a unui cuvânt de cod determină un nou cuvânt de cod, înseamnă că se pot forma sume ortogonale pe fiecare din cele n simboluri recepționate.

În mod analog, în relația (4.204) în toate sumele A_j intervine e_{n-1}, fiind deci ortogonale pe acest simbol, în timp ce fiecare simbol e_i, $i=\overline{0,(n-2)}$ apare o singură dată în aceste sume.

Se poate demonstra că, dacă numărul de erori e care poate să apară pe canalul de transmisiuni satisface relația:

$$e \le \left[\frac{J}{2}\right], \qquad (4.205)$$

unde [J/2] semnifică partea întreagă a numărului J/2, atunci aceste erori pot fi corectate.

Într-adevăr, se presupune, pentru generalitate, că s-au format sume ortogonale pe simbolul e_k, $(\forall)k = \overline{0, (n-1)}$.

Dacă e_k=0, înseamnă că pe poziția k a cuvântului recepționat (numărătoarea efectuându-se de la stânga la dreapta) nu s-a introdus eroare. În acest caz, numărul maxim de sume A_j egale cu unitatea este egal cu [J/2], lucru care s-ar putea întâmpla numai dacă cele [J/2] erori apar în sume distincte.

Dacă e_k=1, înseamnă că pe poziția k a cuvântului recepționat s-a introdus o eroare. Numărul minim de sume A_j egale cu unitatea în acest caz este egal cu J- [J/2] +1. Într-adevăr, deoarece o eroare a apărut pe poziția k, celelalte rămase, în număr de [J/2]-1, pot anula prin sumare modulo 2 cel mult un număr egal de unități e_k=1, rămânând astfel un număr minim de sume A_j=1 egal cu J - ([J/2]-1) = J - [J/2] +1.

Cu alte cuvinte, dacă:

$$\sum_{j=1}^{J} A_j \leq \left[\frac{J}{2}\right] \tag{4.206}$$

rezultă că simbolul pe care s-au format sume ortogonale s-a recepționat corect, iar dacă:

$$\sum_{j=1}^{J} A_j \geq J - \left[\frac{J}{2}\right] + 1 \tag{4.207}$$

rezultă că simbolul pe care s-au format sume ortogonale s-a recepționat eronat.

Mai mult, se poate demonstra că totdeauna:

$$J - \left[\frac{J}{2}\right] + 1 > \left[\frac{J}{2}\right] \tag{4.208}$$

Într-adevăr, relația (4.208) se poate scrie, echivalent, sub forma:

$$J + 1 > 2\left[\frac{J}{2}\right] \tag{4.209}$$

Dacă J este un număr par, J=2i, $i \in \{N \backslash 0\}$, atunci (4.209) devine:

$$2i + 1 > 2i, \tag{4.210}$$

ceea ce este, evident, adevărat.

Dacă J este un număr impar, J=2i+1, $i \in \{N \backslash 0\}$, atunci (4.209) devine

$$2i + 2 > 2i, \tag{4.211}$$

care, de asemenea, este adevărată.

Schema bloc a decodorului care poate corecta un număr de erori dat de relația (4.205), implementat după relația (4.197) şi care ia decizii conform relațiilor (4.206) şi (4.207) este reprezentată în figura 4.25.

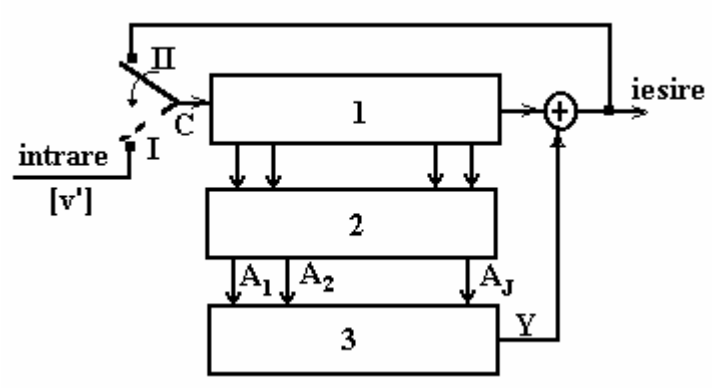

Fig.4.25. Decodor ciclic cu logică de prag.

Blocul notat cu 1 reprezintă un registru de deplasare în care se memorează simbolurile cuvântului recepționat [v']. Blocul 2 este alcătuit dintr-o serie de sumatoare modulo 2 care furnizează sumele A_j, conform relației (4.203). Blocul 3 este un circuit combinațional astfel sintetizat, încât să furnizeze "0" logic la ieşire, dacă este satisfăcută relația (4.206) şi "1" logic, dacă este îndeplinită relația (4.207). Inițial, comutatorul C este pe poziția I şi toate celulele binare din registrul de deplasare se resetează. Dacă polinomul corespunzător cuvântului recepționat este:

$$\mathbf{v'(x) = a'_0 \oplus a'_1 x \oplus \ldots \oplus a'_{n-1} x^{n-1}}, \qquad \textbf{(4.212)}$$

atunci, conform convenției, simbolurile $\mathbf{a'_i}$ sunt introduse prin comutatorul C în registrul de deplasare în ordinea descrescătoare a puterilor lui x. După ce întreg cuvântul recepționat a fost stocat în registrul de deplasare, comutatorul C este trecut pe poziția II. În felul acesta s-au format sume ortogonale pe simbolul $\mathbf{a'_{n-1}}$. Dacă acest simbol a fost recepționat corect, este satisfăcută relația (4.206) şi la ieşirea blocului 3 rezultă "0" logic. În felul acesta, la tactul următor, pe de o parte, simbolul recepționat corect $\mathbf{a'_{n-1}}$ este furnizat la ieşire, iar pe de altă parte, introdus în prima celulă binară a registrului de deplasare, formându-se astfel sume ortogonale pe $\mathbf{a'_{n-2}}$. Dacă simbolul $\mathbf{a'_{n-1}}$ a fost recepționat eronat, este satisfăcută relația (4.207) şi la ieşirea blocului 3 rezultă "1" logic. În felul acesta, la tactul următor, pe de o parte, simbolul recepționat eronat se corectează, fiind furnizat la ieşire, iar pe de altă parte, este stocat corectat în prima celulă binară a registrului de deplasare, formându-se sume ortogonale pe simbolul $\mathbf{a'_{n-2}}$.

Dacă codul ciclic este sistematic, cu cele k simboluri informaționale plasate grupat la sfârşitul cuvântului de cod, comutatorul C va fi menținut pe poziția II k tacte, deoarece interesează numai corecția simbolurilor informaționale. După corecția simbolurilor informaționale, comutatorul C este trecut din nou pe poziția I, efectuându-se simultan furnizarea la ieşire a simbolurilor de control din cuvântul precedent, (eventual eronate) şi recepționarea următorului cuvânt.

Deoarece la ieşirea circuitului combinațional rezultă "1" logic, dacă majoritatea sumelor A_j au valoarea egală cu unitatea, decodarea descrisă este întâlnită în literatura de specialitate şi sub denumirea de decodare cu logică majoritară.

4.21. Definirea pachetelor de erori

În transmisiile pe canale perturbate s-a considerat numai cazul în care perturbațiile afectau independent fiecare simbol binar transmis. Sunt însă şi situaţii când durata perturbațiilor este mai mare decât cea alocată unui simbol binar, dând astfel naştere la erori grupate, cunoscute sub denumirea de *pachete de erori*. Acelaşi efect se petrece în cazul stocării informației discrete pe diverse suporturi fizice (benzi magnetice, discuri, dischete etc.). Perturbația datorată unor zgârieturi sau alte defecte ocupă în acest caz un spațiu mai mare decât cel alocat înregistrării unui simbol binar.

Prin definiție, se numeşte *lungimea l a unui pachet de erori* o succesiune de l simboluri alăturate în care primul şi ultimul simbol sunt eronate.

Fie un canal de transmisiuni pe care pot apărea pachete de erori de lungime l, care conțin e erori ($e \le l$). Dacă pe un astfel de canal se transmit cuvintele de cod de lungime egală cu $n \ge l$, pachetul de erori de lungime l poate ocupa pe lungimea cuvintelor de cod un număr de n-l+1 poziții distincte. Deoarece pachetul de erori de lungime l conține e erori, iar două dintre acestea apar la începutul, respectiv sfârşitul pachetului de erori, restul de e-2 erori ar putea ocupa în cadrul pachetului de erori un număr de poziții distincte, egal cu combinări de l-2 luate câte e-2, (C_{l-2}^{e-2}).

Înseamnă, deci, că pe un canal de transmisiuni pe care pot apărea pachete de erori de lungime l, care conțin e erori, numărul pachetelor de erori distincte se poate determina cu relația:

$$N = (n-l+1)C_{l-2}^{e-2}, \quad 2 \le e \le l \le n \tag{4.213}$$

Cuvântul eroare corespunzător unui pachet de erori de lungime l va fi de forma:

$$[E_i] = [\ldots \alpha_i e_{i+1} \ldots e_{i+k} \ldots e_{i+l-2}\, \alpha_{i+l-1} \ldots] \tag{4.214}$$

unde $\alpha_i=\alpha_{i+l-1}=1$ reprezintă erorile de la capetele pachetului de erori, plasat pe poziția i, oricare ar fi $i = \overline{1,(n-l+1)}$, iar e_{i+k}, $k = \overline{1,l-2}$ va fi "1" logic, dacă pe poziția i+k a apărut o eroare şi "0" logic, dacă pe poziția respectivă nu s-a introdus eroare (numărătoarea efectuându-se de la stânga la dreapta).

Exact ca în cazul apariției erorilor individuale (independente), şi la apariția pachetelor de erori se pun aceleaşi două probleme: detecția pachetelor de erori, respectiv corecția acestora.

4.22. Relații între coloanele matricei de control pentru detecția, respectiv corecția pachetelor de erori

Se consideră că pe un canal perturbat ar putea apărea pachete de erori de lungime l sau mai mică conținând e erori sau mai puține ($2 \le e \le l$).

Fie

$$[\mathbf{H}] = [\mathbf{h}_1\, \mathbf{h}_2 \ldots \mathbf{h}_n] \tag{4.215}$$

matricea de control în care prin $[h_i]$ s-au notat coloanele acesteia.

Corectorul corespunzător se poate calcula cu relația (4.44), adică:

$$[\mathbf{Z}_i] = [\mathbf{H}][\mathbf{E}_i]^T, (\forall)\mathbf{i} = \overline{\mathbf{1,(n-l+1)}} \tag{4.216}$$

Înlocuind (4.214) şi (4.215) în relația (4.216), rezultă:

$$[\mathbf{Z}_i] = \alpha_i[\mathbf{h}_i] \oplus \mathbf{e}_{i+1}[\mathbf{h}_{i+1}] \oplus \ldots \oplus \mathbf{e}_{i+l-2}[\mathbf{h}_{i+l-2}] \oplus \alpha_{i+l-1}[\mathbf{h}_{i+l-1}] \tag{4.217}$$

Se reamintește că pentru detecția erorilor matricea de control trebuie astfel întocmită, încât să rezulte un corector diferit de matricea nulă ori de câte ori s-a recepționat un cuvânt eronat.

Impunând în relația (4.217), $[Z_i] \neq [0]$ şi ținând cont că $\alpha_i = \alpha_{i+l-1} = 1$, rezultă că pentru detecția pachetelor de erori de lungime d sau mai mică este necesar să fie satisfăcută relația:

$$[\mathbf{h}_i] \oplus \mathbf{e}_{i+1}[\mathbf{h}_{i+1}] \oplus \ldots \oplus \mathbf{e}_{i+l-2}[\mathbf{h}_{i+l-2}] \oplus [\mathbf{h}_{i+l-1}] \neq [\mathbf{0}] \tag{4.218}$$

oricare ar fi $i = \overline{1,(n-l+1)}$ şi l = 1, 2, ..., d.

Cu alte cuvinte, rezultă că pentru detecția pachetelor de lungime d sau mai mică matricea de control trebuie astfel întocmită, încât suma modulo 2 a d sau mai puține coloane *alăturate* să fie diferită de matricea nulă.

Cum era de aşteptat, în cazul detecției pachetelor de erori se impun condiții mai puțin restrictive asupra matricei de control, comparativ cu cazul detecției erorilor independente. Aceasta se datorează faptului că în cazul pachetelor de erori se cunoaşte aprioric că cele e erori apar grupate şi nu sunt împrăştiate aleator în cuvântul recepționat.

Se reamintește că în cazul detecției a d sau mai puține erori independente matricea de control a fost astfel întocmită, încât suma modulo 2 a oricăror d sau mai puține coloane *oarecare* şi, deci, în particular, şi a d sau mai puține coloane *alăturate* să fie diferită de matricea nulă.

Pentru corecția erorilor matricea de control trebuie astfel întocmită, încât să rezulte corectori diferiți de matricea nulă, şi, în plus, distincți pentru orice cuvânt recepționat eronat.

Fie cuvântul eroare:

$$[\mathbf{E}_j] = [\ldots \alpha_j \mathbf{e}_{j+1} \ldots \mathbf{e}_{j+k} \ldots \mathbf{e}_{j+l-2} \alpha_{j+l-1} \ldots] \tag{4.219}$$

corespunzător unui pachet de erori de lungime l, plasat pe poziția $j \neq i$, cu $\alpha_j = \alpha_{j+l-1} = 1$, oricare ar fi $j = \overline{1,(n-l+1)}$.

Corectorul corespunzător acestui cuvânt eroare se calculează cu relația:

$$[Z_j]=[h_j]\oplus e_{j+1}[h_{j+1}]\oplus ...\oplus e_{j+l-2}[h_{j+l-2}]\oplus[h_{j+l-1}] \tag{4.220}$$

Pentru corecția pachetelor de erori de lungime d sau mai mică este necesar să fie îndeplinite condițiile:

$$\left.\begin{array}{l}[\mathbf{Z_i}]\neq[\mathbf{0}]\\ [\mathbf{Z_j}]\neq[\mathbf{0}]\\ [\mathbf{Z_i}]\neq[\mathbf{Z_j}]\Leftrightarrow[\mathbf{Z_i}]\oplus[\mathbf{Z_j}]\neq[\mathbf{0}]\end{array}\right\},\quad (\forall)\,\mathbf{i,j}=\overline{\mathbf{1,(n-l+1)}},\ \mathbf{i\neq j,l}=\overline{\mathbf{1,d}}. \tag{4.221}$$

Ținând cont de (4.217) şi (4.220), sistemul de ecuații (4.221) se poate scrie, echivalent, sub forma:

$$\begin{array}{l}[\mathbf{h_i}]\oplus\mathbf{e_{i+1}}[\mathbf{h_{i+1}}]\oplus\ldots\oplus\mathbf{e_{i+l-2}}[\mathbf{h_{i+l-2}}]\oplus[\mathbf{h_{i+l-1}}]\oplus\\ [\mathbf{h_j}]\oplus\mathbf{e_{j+1}}[\mathbf{h_{j+1}}]\oplus\ldots\oplus\mathbf{e_{j+l-2}}[\mathbf{h_{j+l-2}}]\oplus[\mathbf{h_{j+l-1}}]\neq[\mathbf{0}]\\ (\forall)\,\mathbf{i,j}=\overline{\mathbf{1,(n-l+1)}},\ \mathbf{i\neq j,\ l}=\overline{\mathbf{1,d}}.\end{array} \tag{4.222}$$

Din relația (4.222) rezultă că pentru corecția pachetelor de lungime d sau mai mică matricea de control trebuie astfel întocmită, încât suma modulo 2 *a două grupe a câte d sau mai puține coloane alăturate* să fie diferită de matricea nulă.

Condiția rezultată este mai puțin restrictivă decât în cazul corecției unui număr egal de erori independente. Se reaminteşte (v. § 4.6.) că în cazul corecției a d sau mai puține erori independente, matricea de control a fost astfel întocmită, încât suma modulo 2 a 2d sau mai puține coloane *oarecare*, deci, în particular, şi a două grupe a câte d, sau mai puține coloane *alăturate* să fie diferită de matricea nulă. Comparând relația (4.218) cu (4.222), rezultă că detecția pachetelor de lungime 2d sau mai mică este echivalentă cu corecția pachetelor de lungime d sau mai mică.

4.23. Determinarea numărului simbolurilor de control pentru detecția pachetelor de erori

Fie cuvântul de cod

$$[v]=[a_0a_1...a_{n-1}] \tag{4.223}$$

şi ecuațiile de paritate:

$$\left.\begin{array}{l}a_0\oplus a_d\oplus a_{2d}\oplus...=0\\ a_1\oplus a_{d+1}\oplus a_{2d+1}\oplus...=0\\ ----------\\ a_{d-1}\oplus a_{2d-1}\oplus a_{3d-1}\oplus...=0\end{array}\right\} \tag{4.224}$$

unde simbolurile al căror indice este mai mare ca n-1 sunt considerate nule.

Acelaşi sistem de ecuaţii se întocmeşte şi la recepţie cu simbolurile cuvintelor recepţionate. Simbolurile care ar putea fi eronate de pachetele de erori de lungime d sau mai mică se află totdeauna în ecuaţii diferite.

Datorită acestui fapt, dacă se aleg primele d simboluri de verificare a parităţii, atunci nesatisfacerea uneia sau mai multor ecuaţii pentru cuvântul recepţionat este un indiciu că există un pachet de erori de lungime d sau mai mică.

Rezultă, deci, că pentru detecţia pachetelor de erori de lungime d sau mai mică sunt necesare m=d simboluri de control.

4.24. Margini inferioare ale numărului simbolurilor de control în cazul corecţiei pachetelor de erori

În conformitate cu § 4.22, corecţia unui pachet de erori de lungime d sau mai mică este echivalentă cu detecţia unui pachet de erori de lungime 2d sau mai mică. Ţinând cont de acest rezultat şi de § 4.23 rezultă că o primă margine inferioară a numărului simbolurilor de control necesar corecţiei pachetelor de erori de lungime d sau mai mică se determină cu relaţia:

$$\mathbf{m \geq 2d} \qquad \textbf{(4.225)}$$

O altă margine inferioară pentru numărul simbolurilor de control în cazul corecţiei pachetelor de erori se poate obţine ţinând cont de numărul cuvintelor eroare distincte care se pot forma. Considerând în relaţia (4.213) pachetele de erori ce conţin e = 2, 3, ..., l erori, înseamnă că numărul cuvintelor eroare distincte ce pot fi formate se determină cu relaţia:

$$\mathbf{N_1 = \sum_{e=2}^{l} N = (n-l+1) \sum_{e=2}^{l} C_{l-2}^{e-2} = (n-l+1) \cdot 2^{l-2}} \qquad \textbf{(4.226)}$$

Considerând în relaţia (4.226), l = 2, 3, ..., d, va rezulta un număr distinct de cuvinte eroare care poate fi calculat cu relaţia:

$$\mathbf{N_2 = \sum_{l=2}^{d} N_1 = \sum_{l=2}^{d} (n-l+1) \cdot 2^{l-2}} \qquad \textbf{(4.227)}$$

Luând în consideraţie şi cazul apariţiei unei erori care determină n cuvinte eroare distincte şi cazul transmisiei fără erori, căruia îi corespunde cuvântul eroare cu toate componentele nule, rezultă că în cazul apariţiei pachetelor de erori de lungime d sau mai mică numărul cuvintelor eroare distincte posibile este:

$$N_3 = 1 + n + N_2 = 1 + n + \sum_{l=2}^{d} (n - l + 1) \cdot 2^{l-2} = \\ = (1 + n)\left(1 + \sum_{l=2}^{d} 2^{l-2}\right) - \sum_{l=2}^{d} l \cdot 2^{l-2} \tag{4.228}$$

Pentru corecția pachetelor de erori de lungime d sau mai mică, numărul necesar al simbolurilor de control, m, va trebui astfel ales, încât numărul corectorilor distincți, egal cu 2^m ce pot fi formați, să satisfacă relația:

$$2^m \geq N_3 \tag{4.229}$$

În scopul scrierii mai compacte a relației (4.228), se consideră suma progresiei geometrice:

$$\sum_{l=0}^{d} x^l = \frac{x^{d+1} - 1}{x - 1} \tag{4.230}$$

Derivând în raport cu x relația (4.230), rezultă:

$$\sum_{l=0}^{d} l x^{l-1} = \frac{(x - 1)(d + 1)x^d - x^{d+1} + 1}{(x - 1)^2} \tag{4.231}$$

Impunând x=2 în relațiile (4.230) şi (4.231), se poate scrie:

$$\sum_{l=0}^{d} 2^l = 2^{d+1} - 1 \tag{4.232}$$

$$\sum_{l=0}^{d} l \cdot 2^{l-1} = 2^d (d - 1) + 1 \tag{4.233}$$

Dar

$$\sum_{l=0}^{d} 2^l = \sum_{l=0}^{d} 2^2 \cdot 2^{l-2} = 2^2 \cdot 2^{-2} + 2^2 \cdot 2^{-1} + 2^2 \sum_{l=2}^{d} 2^{l-2} = \\ = 2^{d+1} - 1, \tag{4.234}$$

de unde rezultă:

$$\sum_{l=2}^{d} 2^{l-2} = 2^{d-1} - 1 \tag{4.235}$$

În mod analog:

$$\sum_{l=0}^{d} l \cdot 2^{l-1} = \sum_{l=0}^{d} 2l \cdot 2^{l-2} = 2 \cdot 2^{-1} + 2 \sum_{l=2}^{d} l \cdot 2^{l-2} = \\ = 2^d (d - 1) + 1, \tag{4.236}$$

de unde rezultă:

$$\sum_{l=2}^{d} l \cdot 2^{l-2} = 2^{d-1} (d - 1) \tag{4.237}$$

Înlocuind (4.228) în (4.229) şi ţinând cont de (4.235) şi (4.237), se poate scrie:

$$2^m \geq 2^{d-1}(n-d+2) \tag{4.238}$$

Logaritmând în bază 2 relaţia (4.238), rezultă o altă margine inferioară a numărului simbolurilor de control, şi anume:

$$m \geq d-1+\log_2(n-d+2) \tag{4.239}$$

Această margine inferioară constituie o condiţie necesară şi nu totdeauna suficientă, deoarece, ca şi în cazul erorilor individuale, şi în cazul pachetelor de erori este posibil ca unor pachete de erori distincte, de aceeaşi lungime şi pondere, să le corespundă corectori identici.

O altă margine inferioară a numărului simbolurilor de control pentru corecţia pachetelor de erori a fost stabilită de Varşarmov şi Gilbert. Pentru determinarea acestei margini se pleacă de la condiţia pe care trebuie să o satisfacă coloanele matricei de control în cazul corecţiei pachetelor de lungime d sau mai mică. În § 4.22 această condiţie este definită prin relaţia (4.222).

Pentru a determina numărul simbolurilor de control pentru corecţia pachetelor de erori de lungime d sau mai mică, se presupune că primele n-1 coloane ale matricei de control satisfac relaţia (4.222), urmând a se determina condiţiile ce trebuie îndeplinite de ultima coloană $[h_n]$.

În acest scop se înlocuieşte indicele j+l-1 din relaţia (4.222) cu n, astfel încât în suma dată să intervină coloana $[h_n]$, adică se poate scrie:

$$\begin{aligned}&[h_i]\oplus e_{i+1}[h_{i+1}]\oplus\ldots\oplus e_{i+l-2}[h_{i+l-2}]\oplus[h_{i+l-1}]\oplus\\&\oplus[h_{n-l+1}]\oplus e_{n-l+2}[h_{n-l+2}]\oplus\ldots\oplus e_{n-1}[h_{n-1}]\neq[h_n]\end{aligned} \tag{4.240}$$

Cu alte cuvinte, dacă primele n-1 coloane ale matricei de control sunt astfel întocmite încât satisfac relaţia (4.222), ultima coloană $[h_n]$ trebuie să satisfacă relaţia (4.240) pentru toate combinaţiile posibile ale membrului stâng al acestei relaţii. Pe de altă parte, membrul stâng al relaţiei (4.240) conţine un pachet de erori de lungime d sau mai mică cu poziţie fixă, deoarece conţine coloana $[h_n]$ şi un pachet de erori de lungime d sau mai mică, care poate fi plasat în n-d poziţii distincte.

Numărul cuvintelor eroare distincte corespunzătoare pachetului de erori de lungime d sau mai mică ce conţine coloana $[h_n]$ se determină cu relaţia

$$N_1 = 2^{d-1} \tag{4.241}$$

deoarece o eroare este pe poziţia n, iar celelalte d-1 sau mai puţine erori posibile determină un număr de cuvinte eroare distincte determinat de această relaţie.

Celălalt pachet de erori va determina un număr de cuvinte eroare distincte ce poate fi determinat din membrul drept al relaţiei (4.238), înlocuind n cu n-d, deoarece acest pachet poate ocupa n-d poziţii distincte. Notând cu N_2 numărul cuvintelor eroare distincte cauzate de cel de al doilea pachet de erori, se poate scrie:

$$N_2 = 2^{d-1}\cdot(n-2d+2) \tag{4.242}$$

Numărul total al cuvintelor eroare distincte va fi atunci determinat cu relația:

$$N = N_1 \cdot N_2 = 2^{2(d-1)} \cdot (n - 2d + 2) \quad (4.243)$$

Pentru ca $[h_n]$ să satisfacă relația (4.240) trebuie ca numărul de coloane cu m simboluri (0,1) ce poate fi format, adică 2^m, să fie suficient de mare, astfel încât, după formarea unui număr de N coloane distincte, stabilit cu relația (4.243), să mai rămână cel puțin o coloană care să poată reprezenta pe $[h_n]$, adică:

$$2^m > N \Leftrightarrow 2^m > 2^{2(d-1)} \cdot (n - 2d + 2) \quad (4.244)$$

Logaritmând în bază 2 ambii membri ai relației (4.244), rezultă:

$$m > 2(d-1) + \log_2 (n - 2d + 2) \quad (4.245)$$

Valoarea lui m dată de relația (4.245) este o condiție suficientă şi nu totdeauna necesară pentru corecția pachetelor de erori de lungime d sau mai mică, deoarece în calculul numărului N coloanele matricei de control s-au considerat distincte.

4.25. Codor ciclic corector de două erori adiacente (alăturate) sau mai puține

Structura codorului ciclic corector de două erori adiacente sau mai puține este identică cu cea din cazul codorului ciclic corector de o eroare, cu deosebirea privind alegerea polinomului generator al codului, g(x). Aşa cum s-a stabilit anterior, pentru corecția unei erori, este necesar un număr de simboluri de control, determinat ca numărul întreg pozitiv m, cel mai mic, ce satisface relația:

$$2^m \geq n + 1, \quad (4.246)$$

unde n este lungimea cuvintelor de cod.

Ca şi în cazul codurilor Hamming corectoare de o eroare, detectoare de erori duble, şi în cazul corecției a două erori adiacente sau mai puține trebuie adăugat un simbol de control suplimentar, prin care să se poată decide dacă pachetul de erori este format din două simboluri sau dintr-un singur simbol.

Notând cu m_c numărul simbolurilor de control necesare pentru corecția pachetelor de erori de lungime doi sau mai mică, se poate scrie:

$$m_c = m + 1 \quad (4.247)$$

Ținând cont că lungimea cuvintelor de cod este egală cu suma dintre numărul simbolurilor de control, m_c, şi numărul simbolurilor informaționale, k, se poate scrie:

$$n = m_c + k \quad (4.248)$$

Cu (4.247) şi (4.248) relația (4.246) devine:

$$2^{m_c - 1} \geq m_c + k + 1 \quad (4.249)$$

Din acest motiv, polinomul generator al codului va fi de forma:

$$\mathbf{g(x) = (x \oplus 1) \cdot g_1(x)} \tag{4.250}$$

unde $g_1(x)$ este un polinom primitiv de gradul m.

Dacă polinomul primitiv $g_1(x)$ este de forma:

$$\mathbf{g_1(x) = g_{10} \oplus g_{11}x \oplus g_{12}x^2 \oplus \ldots \oplus g_{1m}x^m,} \quad \mathbf{g_{10} = g_{1m} = 1,\ g_{1i} \in \{0,1\}} \tag{4.251}$$

atunci, conform relației (4.250), rezultă:

$$\mathbf{g(x) = g_0 \oplus g_1x \oplus \ldots \oplus g_mx^m \oplus g_{m+1} \cdot x^{m+1},} \quad \mathbf{g_0 = g_{m+1} = 1,\ g_i \in \{0,1\}} \tag{4.252}$$

Structura codorului ciclic corector de două erori alăturate, sau mai puține, este dată în figura 4.26.

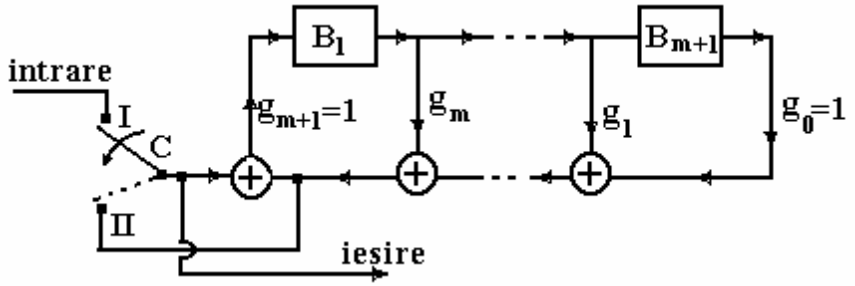

Fig.4.26. Codor ciclic corector de două erori adiacente sau mai puține.

Inițial, comutatorul C este pe poziția I şi toate celulele binare se resetează. În ritmul impulsurilor de tact, la intrare se aplică simbolurile informaționale care, pe de o parte, rezultă direct la ieşire, iar pe de altă parte, sunt introduse în registrul de deplasare cu reacție întocmit după polinomul g(x), dat de relația (4.252). După ce ultimul simbol informațional a fost introdus în registrul de deplasare cu reacție, comutatorul C este trecut pe poziția II şi în ultimele m_c tacte la ieşire vor rezulta simbolurile de control.

În acest caz vor fi adevărate relațiile din cazul codorului ciclic, corector de o eroare, realizat cu R.D.R., (4.153) ÷ (4.157), cu deosebirea că matricea conexiunilor şi matricea [U] sunt definite astfel:

$$\mathbf{[T]} = \begin{bmatrix} 0 & 1 & 0 & \ldots & 0 \\ 0 & 0 & 1 & \ldots & 0 \\ - & - & - & - & - \\ 0 & 0 & 0 & \ldots & 1 \\ g_0 & g_1 & g_2 & \cdots & g_m \end{bmatrix}; \quad \mathbf{[U]} = \begin{bmatrix} 0 \rightarrow B_{m+1} \\ 0 \rightarrow B_m \\ . \\ . \\ . \\ 0 \rightarrow B_2 \\ 1 \rightarrow B_1 \end{bmatrix} \tag{4.253}$$

Pentru fixarea ideilor, se consideră că trebuie întocmit cuvântul de cod pentru transmiterea numărului N=3 pe un canal pe care pot să apară două erori adiacente sau mai puține. Numărul şi structura simbolurilor informaționale rezultă prin transcrierea binară a numărului ce urmează a fi codat, adică $(N)_{10} = (3)_{10} = (11)_2$.

Conform relației (4.249), rezultă $\mathbf{2^{m_c-1} \geq m_c + 3}$, de unde se deduce m_c=4.

Din relația (4.250) rezultă că polinomul primitiv $g_1(x)$ trebuie să aibă gradul m=3. Fie polinomul $g_1(x)=1\oplus x^2\oplus x^3$. Rezultă atunci $g(x) = (x \oplus 1)\ (1 \oplus x^2 \oplus x^3) = 1 \oplus \oplus\ x \oplus x^2 \oplus x^4$. Structura codorului pentru acest caz particular este dată în figura 4.27.

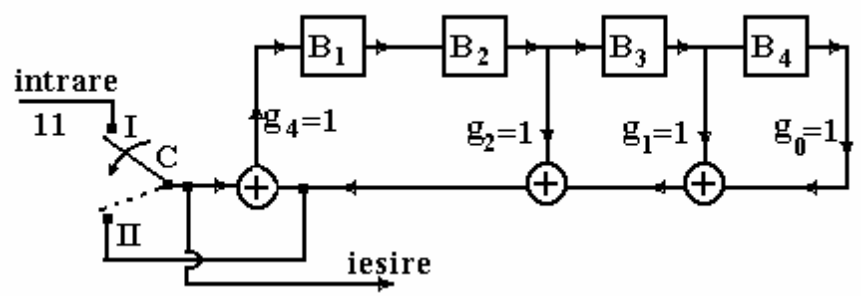

Fig.4.27. Codor ciclic întocmit după polinomul $g(x) =1 \oplus x \oplus x^2 \oplus x^4$.

Tact	I_n	B_1	B_2	B_3	B_4	Ieşire
	0	0	0	0	0	
1	1	0	0	0	0	1
2	1	1	0	0	0	1
3		1	1	0	0	1
4		0	1	1	0	0
5		0	0	1	1	0
6		0	0	0	1	1

Funcționarea codorului este sintetizată în tabelul alăturat. Cuvântul de cod rezultat la ieşire este [v] = [1 0 0 1 1 1], unde primele 4 simboluri sunt de control, iar ultimele două informaționale. Imediat după tactul 3, când ultimul simbol informațional a fost introdus în registru de deplasare cu reacție, comutatorul C este trecut pe poziția II, la ieşire rezultând, în ritmul impulsurilor de tact, simbolurile de control, ca sumă modulo 2 a stărilor celulelor binare B_2, B_3 şi B_4.

4.26. Decodor ciclic corector de două erori adiacente (alăturate), sau mai puține

Fie

$$\mathbf{v'(x) = a'_0 \oplus a'_1 x \oplus \ldots \oplus a'_{n_1-1} x^{n_1-1}} \qquad \textbf{(4.254)}$$

cuvântul recepționat sau, sub formă matriceală:

$$\mathbf{[v'] = [a'_0\ a'_1 \ldots a'_{n_1-1}]} \qquad \textbf{(4.255)}$$

Dacă

$$\mathbf{n_1 < 2^m - 1,} \qquad \textbf{(4.256)}$$

unde m este gradul polinomului primitiv utilizat, atunci la sfârşitul cuvântului recepționat va fi adăugat un număr de zerouri, până când este satisfăcută relația

$$\mathbf{2^m - 1 = n} \qquad \textbf{(4.257)}$$

Cuvântul recepționat, cu eventualele zerouri adăugate la sfârşitul cuvântului, va fi de forma:

$$\mathbf{v'(x) = a'_0 \oplus a'_1 x \oplus \ldots \oplus a'_{n_1-1} x^{n_1-1} \oplus a'_{n_1} x^{n_1} \oplus \ldots \oplus a'_{n-1} x^{n-1}}, \qquad \textbf{(4.258)}$$

unde $\mathbf{a'_{n_1} = \ldots = a'_{n-1} = 0}$.

Adăugarea de zerouri la sfârşitul cuvântului recepționat până la satisfacerea relației (4.257) este necesară pentru a fi satisfăcută relația:

$$\mathbf{[T]^n = [I_{m+1}]}, \qquad \textbf{(4.259)}$$

unde [T] este matricea conexiunilor din relația (4.253), iar $[I_{m+1}]$ matricea unitate de ordinul m+1.

Structura decodorului ciclic, corector de două erori adiacente sau mai puține este identică cu cea din cazul decodorului ciclic corector de o eroare, realizat cu R.D.R. (fig.4.21), cu deosebirea că registrele de deplasare cu reacție se întocmesc după polinomul generator g(x) dat de relația (4.250), iar descifratoarele trebuie astfel sintetizate, încât să furnizeze *câte o unitate logică* când simbolul eronat ajunge în ultima celulă M_0 a registrului principal.

Fără a micşora generalitatea, se presupune că cele două simboluri alăturate eronate sunt $\mathbf{a'_{n-s}}$ şi $\mathbf{a'_{n-s-1}}$, oricare ar fi $s=\overline{1,(n-1)}$, adică structura matriceală a cuvântului recepționat este de forma:

$$\mathbf{[v'] = [a_0\ a_1 \ldots, (a_{n-s-1} \oplus 1), (a_{n-s} \oplus 1), \ldots a_{n-1}]} \qquad \textbf{(4.260)}$$

Matricea de control se va determina cu relația (4.169), unde matricele [T] şi [U] sunt specificate de relația (4.253). Corectorul cuvântului recepționat se va calcula cu relația:

$$\mathbf{[Z_2] = [H][v']^T} \qquad \textbf{(4.261)}$$

Având în vedere că la emisie este satisfăcută relația (4.154), rezultă:

$$\mathbf{[Z_2] = [T]^{n-s-1}[U] \oplus [T]^{n-s}[U]}, \qquad \textbf{(4.262)}$$

sau, ținând cont de (4.259), se poate scrie:

$$\mathbf{[Z_2] = [T]^{-s-1}[U] \oplus [T]^{-s}[U]} \qquad \textbf{(4.263)}$$

Primul simbol eronat din cele două adiacente, adică $\mathbf{a'_{n-s}}$, va ajunge în ultima celulă binară, M_0, a registrului principal după s-1 tacte. Starea R.D.R.-ului după s-1 tacte se determină cu relația:

$$\mathbf{[S_2] = [T]^{s-1}[Z_2]}, \qquad \textbf{(4.264)}$$

sau, ținând cont de (4.263), rezultă:

$$[S_2] = [T]^{-2}[U] \oplus [T]^{-1}[U] \tag{4.265}$$

Când R.D.R.-ul ajunge în starea $[S_2]$, descifratorul trebuie astfel sintetizat, încât să furnizeze la ieşire o unitate logică care, pe de o parte, sumată modulo 2 cu simbolul $\mathbf{a}'_{n-s}$ îl corectează, iar pe de altă parte, va fi introdusă în R.D.R., astfel că la tactul următor, când cel de al doilea simbol eronat, $\mathbf{a}'_{n-s-1}$, ajunge în celula binară M_0, starea R.D.R.-ului devine:

$$[S_1] = [T][S_2] \oplus [U] \tag{4.266}$$

Înlocuind (4.265) în relaţia (4.266), rezultă

$$[S_1] = [T]^{-1}[U] \oplus [U] \oplus [U] = [T]^{-1}[U] \tag{4.267}$$

Când R.D.R.-ul ajunge în starea $[S_1]$, descifratorul va trebui să furnizeze o altă unitate logică la ieşire care, pe de o parte, se va aduna modulo 2 cu simbolul eronat $\mathbf{a}'_{n-s-1}$, corectându-l, iar pe de altă parte, va fi introdusă în R.D.R., astfel că la tactul următor starea acestuia va deveni:

$$[S_0] = [T][S_1] \oplus [U] = [U] \oplus [U] = [0], \tag{4.268}$$

pregătindu-l astfel pentru recepţionarea următorului cuvânt.

Dacă în cuvântul eronat ar fi apărut o singură eroare, de exemplu simbolul $\mathbf{a}'_{n-r}$ ar fi eronat, cuvântul recepţionat va fi de forma:

$$[v'] = [a_0\ a_1 \ldots, (a_{n-r} \oplus 1), \ldots a_{n-1}] \tag{4.269}$$

Când întregul cuvânt recepţionat este memorat în registrul principal, starea R.D.R.-ului, ca şi în cazul decodorului ciclic corector de o eroare, va stabili corectorul cuvântului recepţionat, determinat cu relaţia:

$$[Z_1] = [H][v']^T \tag{4.270}$$

Înlocuind matricea de control din relaţia (4.169), unde [T] şi [U] sunt specificate de (4.253) şi $[v']$ dat de (4.269) şi având în vedere că la emisie codorul satisface relaţia (4.154), rezultă:

$$[Z_1] = [T]^{n-r}[U], \tag{4.271}$$

sau, ţinând cont de (4.259):

$$[Z_1] = [T]^{-r}[U] \tag{4.272}$$

Simbolul $\mathbf{a}'_{n-r}$ ajunge în celula binară M_0 a registrului principal după r-1 tacte. Starea R.D.R.-ului după r-1 tacte devine:

$$[S_1] = [T]^{r-1}[Z_1] = [T]^{-1}[U], \tag{4.273}$$

adică aceeaşi ca aceea dată de relaţia (4.267).

Înseamnă, deci, că în cazul apariţiei a două erori adiacente sau mai puţine, descifratorul trebuie astfel sintetizat, încât la apariţia stărilor $[S_2]$ şi $[S_1]$ să furnizeze

"1" logic la ieşire. Se poate demonstra, exact ca în cazul descifratorului din cadrul decodorului ciclic corector de o eroare (§ 4.18), că, dacă matricea conexiunilor este dată de relația (4.253), atunci:

$$[\mathbf{T}]^{-1} = \begin{bmatrix} \mathbf{g_1\ g_2 \ldots g_m\ 1} \\ \mathbf{1\ 0\ \ldots 0\ 0} \\ \mathbf{0\ 1\ \ldots 0\ 0} \\ \mathbf{- - - - - - -} \\ \mathbf{0\ 0\ \ldots 1\ 0} \end{bmatrix} \qquad \textbf{(4.274)}$$

Pentru fixarea ideilor, se presupune că s-a recepționat cuvântul $[v^{'}] = [100100]$ şi că la codare s-a folosit polinomul generator $g(x) = 1 \oplus x \oplus x^2 \oplus x^4$. Dacă pe canalul de transmisiuni pot să apară două erori adiacente sau mai puține, pentru a deduce numărul zecimal transmis se verifică mai întâi relația (4.257). Deoarece g(x) are gradul $m_c=4$, rezultă că polinomul primitiv are gradul m=3. Înseamnă că $2^m-1=2^3-1=7$, în timp ce lungimea cuvântului recepționat este $n_1=6$. Va trebui adăugat un zero la sfârşitul cuvântului recepționat, adică se consideră $[v^{'}] = [1001000]$. Particularizând relația (4.253) pentru acest caz, rezultă:

$$[\mathbf{T}] = \begin{bmatrix} \mathbf{0\,1\,0\,0} \\ \mathbf{0\,0\,1\,0} \\ \mathbf{0\,0\,0\,1} \\ \mathbf{1\,1\,1\,0} \end{bmatrix} \quad \textbf{si} \quad [\mathbf{U}] = \begin{bmatrix} \mathbf{0} \rightarrow \mathbf{B_4} \\ \mathbf{0} \rightarrow \mathbf{B_3} \\ \mathbf{0} \rightarrow \mathbf{B_2} \\ \mathbf{1} \rightarrow \mathbf{B_1} \end{bmatrix}$$

Conform relației (4.274), se poate scrie:

$$[\mathbf{T}]^{-1} = \begin{bmatrix} \mathbf{1\,1\,0\,1} \\ \mathbf{1\,0\,0\,0} \\ \mathbf{0\,1\,0\,0} \\ \mathbf{0\,0\,1\,0} \end{bmatrix}.$$

Starea $[S_1]$, conform relației (4.273), se poate determina cu relația:

$$[S_1] = \begin{bmatrix} 1101 \\ 1000 \\ 0100 \\ 0010 \end{bmatrix} \begin{bmatrix} 0 \\ 0 \\ 0 \\ 1 \end{bmatrix} = \begin{bmatrix} 1 \rightarrow B_4 \\ 0 \rightarrow B_3 \\ 0 \rightarrow B_2 \\ 0 \rightarrow B_1 \end{bmatrix}$$

Pentru a determina starea $[S_2]$ cu relația (4.265), se calculează mai întâi

$$[\mathbf{T}]^{-2}[\mathbf{U}] = [\mathbf{T}]^{-1}([\mathbf{T}]^{-1}[\mathbf{U}]) = \begin{bmatrix} \mathbf{1\,1\,0\,1} \\ \mathbf{1\,0\,0\,0} \\ \mathbf{0\,1\,0\,0} \\ \mathbf{0\,0\,1\,0} \end{bmatrix} \begin{bmatrix} \mathbf{1} \\ \mathbf{0} \\ \mathbf{0} \\ \mathbf{0} \end{bmatrix} = \begin{bmatrix} \mathbf{1} \\ \mathbf{1} \\ \mathbf{0} \\ \mathbf{0} \end{bmatrix}.$$

Înseamnă atunci că:

$$[\mathbf{S}_2]=\begin{bmatrix}\mathbf{1}\\ \mathbf{1}\\ \mathbf{0}\\ \mathbf{0}\end{bmatrix}\oplus\begin{bmatrix}\mathbf{1}\\ \mathbf{0}\\ \mathbf{0}\\ \mathbf{0}\end{bmatrix}=\begin{bmatrix}\mathbf{0}\rightarrow\mathbf{B}_4\\ \mathbf{1}\rightarrow\mathbf{B}_3\\ \mathbf{0}\rightarrow\mathbf{B}_2\\ \mathbf{0}\rightarrow\mathbf{B}_1\end{bmatrix}.$$

Descifratorul trebuie să sesizeze stările $[S_1]$ şi $[S_2]$, adică trebuie să verifice tabela de adevăr:

	B_1	B_2	B_3	B_4	Y
$S_1\rightarrow$	0	0	0	1	1
$S_2\rightarrow$	0	0	1	0	1

Din această tabelă de adevăr rezultă:

$$Y=\overline{B_1}\,\overline{B_2}\,\overline{B_3}\,B_4\cup\overline{B_1}\,\overline{B_2}\,B_3\,\overline{B_4}=$$
$$=\overline{B_1}\,\overline{B_2}(\overline{B_3}\,B_4\cup B_3\,\overline{B_4})=\overline{B_1}\,\overline{B_2}(B_3\oplus B_4).$$

Schema decodorului ciclic capabil să corecteze două erori alăturate sau mai puține din cuvântul recepționat, la care s-a mai adăugat un zero la sfârşitul acestuia, este dată în figura 4.28.

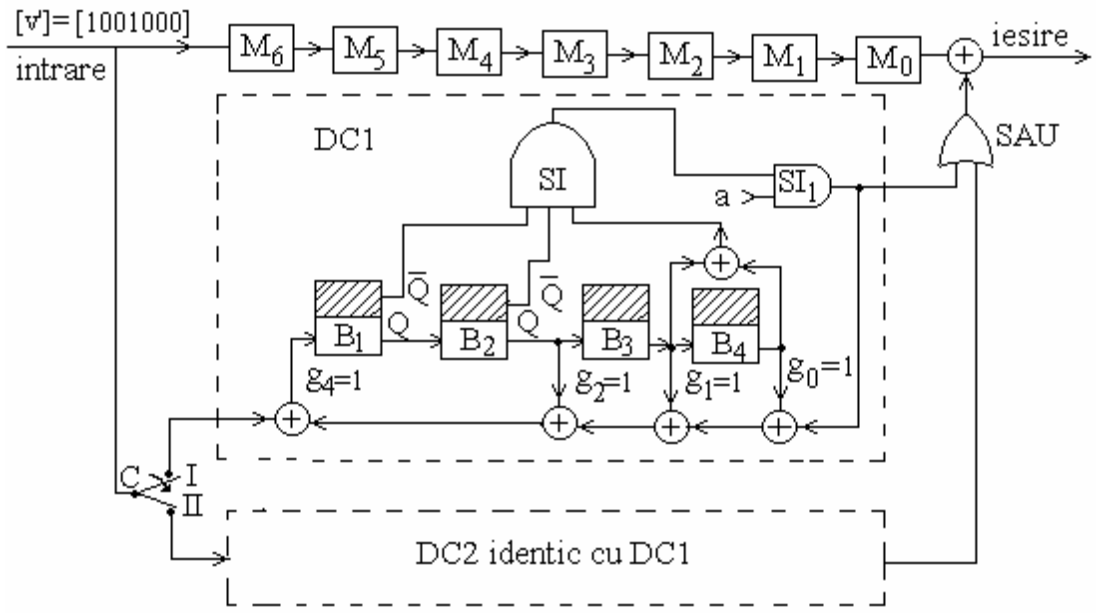

Fig.4.28. Decodor ciclic corector de două erori adiacente sau mai puține, întocmit după polinomul $g(x) = 1 \oplus x \oplus x^2 \oplus x^4$.

Funcționarea decodorului este sintetizată în tabela de mai jos:

Tact	In.	M_6	M_5	M_4	M_3	M_2	M_1	M_0	B_1	B_2	B_3	B_4	Ieş.
	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	
2	0	0	0	0	0	0	0	0	0	0	0	0	
3	0	0	0	0	0	0	0	0	0	0	0	0	
4	1	0	0	0	0	0	0	0	0	0	0	0	
5	0	1	0	0	0	0	0	0	1	0	0	0	
6	0	0	1	0	0	0	0	0	0	1	0	0	
7	1	0	0	1	0	0	0	0	1	0	1	0	
8	0	1	0	0	1	0	0	0	0	1	0	1	0
9	0	0	1	0	0	1	0	0	0	0	1	0	1
10	0	0	0	1	0	0	1	0	0	0	0	1	1
11	0	0	0	0	1	0	0	1	0	0	0	0	1
12	0	0	0	0	0	1	0	0	0	0	0	0	0
13	0	0	0	0	0	0	1	0	0	0	0	0	0
14	0	0	0	0	0	0	0	1	0	0	0	0	1

Inițial, comutatorul C este pe poziția I şi toate celulele binare se resetează. După ce la intrare s-a aplicat cuvântul $[v^{'}] = [1\ 0\ 0\ 1\ 0\ 0\ 0]$, următorul cuvânt s-a considerat format din 7 zerouri. Când comutatorul este pe poziția I starea celulei binare B_1, la un tact oarecare, este egală cu suma modulo 2 a stărilor precedente ale intrării şi celulelor binare B_2, B_3 şi B_4. Când comutatorul C este pe poziția II, starea celulei binare B_1, la un tact oarecare, se obține sumând modulo 2 stările precedente ale celulelor binare B_2, B_3 şi B_4 şi eventual a lui "1" logic rezultat la ieşirea porții $ŞI_1$. (Intrarea a=1 numai când comutatorul C este pe poziția II). Se observă că la tactul 9 starea R.D.R.-ului este egală cu $[S_2]$, determinând la ieşirea descifratorului Y=1 care, validat de poarta logică $ŞI_1$, se adună modulo 2 la simbolul memorat în celula M_0, furnizând la ieşire $0 \oplus 1 = 1$ pe de o parte, iar pe de altă parte, determină la tactul următor ca starea R.D.R.-ului să devină egală cu $[S_1]$, obținându-se din nou Y=1. În felul acesta, la tactul 10, la ieşire rezultă $0 \oplus 1 = 1$, iar starea R.D.R.-ului la tactul următor ajunge cu toate celulele binare în starea de "0" logic, fiind astfel pregătit pentru recepționarea următorului cuvânt. Cuvântul corectat obținut la ieşire este [v] = [1 0 0 1 1 1 0]. Îndepărtând ultimul zero, rezultă că s-a transmis cuvântul de cod [v] = [1 0 0 1 1 1]. Deoarece polinomul generator al codului, g(x), are gradul 4, înseamnă că primele 4 simboluri sunt de control, iar următoarele de informație. Transcriind în zecimal numărul binar format din simbolurile informaționale, înseamnă că s-a transmis numărul $(11)_2=(3)_{10}$. Dacă nu se efectua corecția, se decidea că s-a transmis numărul zero.