

Tema 6.2 Coduri Hamming

Un cod Hamming (n, n_0) este format din cuvinte de cod cu lungimea de n simboluri, dintre care n_0 simboluri sunt informaționale și $k=n-n_0$ simboluri sunt de control.

Codul Hamming corector de o eroare are $d_{\min}=3$.

Dacă vom considera că la transmiterea unui cuvânt de n simboluri poate apărea o singură eroare, atunci numărul variantelor posibile va fi egal cu $n+1$, în care o combinație va reprezenta cuvântul de cod transmis fără erori, iar celelalte n combinații - toate variantele posibile de erori singulare. Deci, pentru a corecta presupusa eroare, utilizând k simboluri de control, este necesar a descrie $n+1$ evenimente distincte: $n+1 = n_0 + k + 1 \leq 2^k$ (4.1)

Din relație rezultă:

- 1) Lungimea cuvântului de cod: $n \leq 2^k - 1$
- 2) Numărul simbolurilor informaționale: $n_0 \leq 2^k - k - 1$

Utilizând relația (4.1), vom calcula numărul simbolurilor de control k , fiind cunoscut numărul simbolurilor informaționale n_0 .

n_0	1	2-4	5-11	12-26	27-57
k	2	3	4	5	6
n	3	5-7	9-15	17-31	33-63

Detectarea erorilor în codurile Hamming se bazează pe același principiu ca și în codurile cu paritate. Deosebirea constă în următoarele:

- *pozițiile de control sunt date de puteri ale lui 2;*
- *fiecare simbol de control răspunde pentru paritatea anumitor simboluri informaționale.*

Simbolurile de control se plasează pe pozițiile 1, 2, 4, 8, 16, ... deoarece la reprezentarea binară a acestor numere este prezentă doar o singură unitate și astfel

se simplifică procedura de calcul a acestor simboluri. Fiecare simbol de control reprezintă paritatea unui set de simboluri informaționale. Poziția simbolului de control determină secvența biților, care verifică și sare alternativ. Poziția 1 verifică un bit și sare 1 bit (1, 3, 5, 7, 9 etc.). Poziția 2 verifică 2 biți și sare 2 biți (2, 3, 6, 7, 10, 11 etc.). Poziția 4 verifică 4 biți și sare 4 biți (4, 5, 6, 7, 12, 13, 14, 15 etc.). Poziția 8 verifică 8 biți și sare 8 biți (8-15, 24-31 etc.).

După transmiterea cuvintelor de cod, se calculează încă o dată paritatea seturilor de biți, formându-se *sindromul erorii*. Dacă cuvintele au fost transmise fără erori, sindromul este nul. În cazul apariției unei erori, sindromul va indica poziția ei în cuvântul transmis, iar simbolul eronat va fi corectat prin inversare. În final, din cuvântul de cod vor fi extrase doar simbolurile informaționale.

Vom examina codul Hamming (12,8). Un cuvânt de cod are $n=12$ simboluri, dintre care $n_0=8$ simboluri informaționale și $k=n-n_0=4$ simboluri de control.

Vom nota prin d_i simbolurile informaționale și prin c_i simbolurile de control.

Se va genera un cuvânt de cod V

$$V = \{d_{12}, d_{11}, d_{10}, d_9, c_8, d_7, d_6, d_5, c_4, d_3, c_2, c_1\}$$

Calculul simbolurilor de control:

$$c_1 = d_3 \oplus d_5 \oplus d_7 \oplus d_9 \oplus d_{11}$$

$$c_2 = d_3 \oplus d_6 \oplus d_7 \oplus d_{10} \oplus d_{11}$$

$$c_4 = d_5 \oplus d_6 \oplus d_7 \oplus d_{12}$$

$$c_8 = d_9 \oplus d_{10} \oplus d_{11} \oplus d_{12}$$

La recepționarea cuvântului de cod $V' = \{d'_{12}, d'_{11}, d'_{10}, d'_9, c'_8, d'_7, d'_6, d'_5, c'_4, d'_3, c'_2, c'_1\}$, se va calcula sindromul erorii conform următoarelor relații:

$$s_1 = c'_1 \oplus d'_3 \oplus d'_5 \oplus d'_7 \oplus d'_9 \oplus d'_{11}$$

$$s_2 = c'_2 \oplus d'_3 \oplus d'_6 \oplus d'_7 \oplus d'_{10} \oplus d'_{11}$$

$$s_4 = c'_4 \oplus d'_5 \oplus d'_6 \oplus d'_7 \oplus d'_{12}$$

$$s_8 = c'_8 \oplus d'_9 \oplus d'_{10} \oplus d'_{11} \oplus d'_{12}$$

Sindromul erorii (s_8, s_4, s_2, s_1) va fi nul, dacă cuvântul a fost transmis fără erori.

În cazul apariției unei erori, sindromul va indica poziția simbolului eronat.

Să analizăm un exemplu concret. Fie că este necesar a forma un cuvânt de cod pentru simbolurile informaționale $\{1,1,0,0,1,1,1,0\}$. Le vom plasa în felul următor:

Poz	d_{12}	d_{11}	d_{10}	d_9	c_8	d_7	d_6	d_5	c_4	d_3	c_2	c_1
V	1	1	0	0		1	1	1		0		

Vom calcula simbolurile de control:

$$c_1 = d_3 \oplus d_5 \oplus d_7 \oplus d_9 \oplus d_{11} = 0 \oplus 1 \oplus 1 \oplus 0 \oplus 1 = 1$$

$$c_2 = d_3 \oplus d_6 \oplus d_7 \oplus d_{10} \oplus d_{11} = 0 \oplus 1 \oplus 1 \oplus 0 \oplus 1 = 1$$

$$c_4 = d_5 \oplus d_6 \oplus d_7 \oplus d_{12} = 1 \oplus 1 \oplus 1 \oplus 1 = 0$$

$$c_8 = d_9 \oplus d_{10} \oplus d_{11} \oplus d_{12} = 0 \oplus 0 \oplus 1 \oplus 1 = 0$$

Se va obține următorul cuvânt de cod:

Poz	d_{12}	d_{11}	d_{10}	d_9	c_8	d_7	d_6	d_5	c_4	d_3	c_2	c_1
V	1	1	0	0	0	1	1	1	0	0	1	1

Să presupunem, că la transmitere a fost recepționat cuvântul de cod

Poz	d_{12}'	d_{11}'	d_{10}'	d_9'	c_8'	d_7'	d_6'	d_5'	c_4'	d_3'	c_2'	c_1'
V'	1	1	1	0	0	1	1	1	0	0	1	1

Pentru a determina dacă cuvântul de cod a fost transmis corect, se va calcula sindromul erorii

$$s_1 = c_1' \oplus d_3' \oplus d_5' \oplus d_7' \oplus d_9' \oplus d_{11}' = 1 \oplus 0 \oplus 1 \oplus 1 \oplus 0 \oplus 1 = 0$$

$$s_2 = c_2' \oplus d_3' \oplus d_6' \oplus d_7' \oplus d_{10}' \oplus d_{11}' = 1 \oplus 0 \oplus 1 \oplus 1 \oplus 1 \oplus 1 = 1$$

$$s_4 = c_4' \oplus d_5' \oplus d_6' \oplus d_7' \oplus d_{12}' = 0 \oplus 1 \oplus 1 \oplus 1 \oplus 1 = 0$$

$$s_8 = c_8' \oplus d_9' \oplus d_{10}' \oplus d_{11}' \oplus d_{12}' = 0 \oplus 0 \oplus 1 \oplus 1 \oplus 1 = 1$$

Sindromul $(s_8, s_4, s_2, s_1) = (1, 0, 1, 0)$ indică că eroarea a avut loc în poziția 10 (bitul d_{10}). Acest bit poate fi corectat prin inversare.

Să presupunem, că la transmitere a fost recepționat un cuvânt de cod cu 2 erori:

Poz	d_{12}'	d_{11}'	d_{10}'	d_9'	c_8'	d_7'	d_6'	d_5'	c_4'	d_3'	c_2'	c_1'
V'	1	1	0	0	0	0	1	1	0	0	0	1

$$s_1 = 1 \oplus 0 \oplus 1 \oplus 1 \oplus 0 \oplus 1 = 0$$

$$s_2 = 0 \oplus 0 \oplus 0 \oplus 1 \oplus 0 \oplus 1 = 0$$

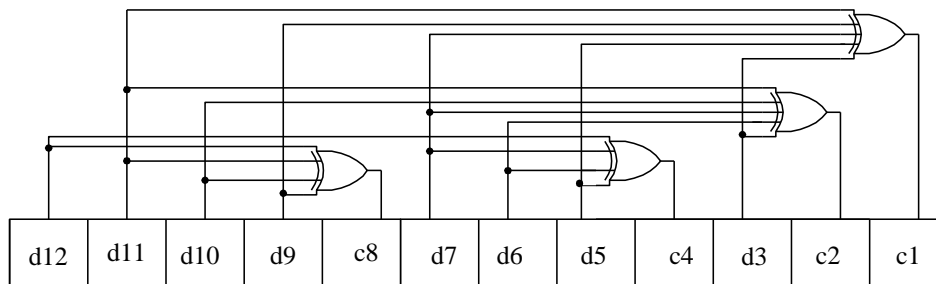
$$s_4 = 0 \oplus 1 \oplus 0 \oplus 1 \oplus 1 = 1$$

$$s_8 = 0 \oplus 0 \oplus 0 \oplus 1 \oplus 1 = 0$$

Sindromul indică eroare în poziția 4, pe când erorile au avut loc în pozițiile 2 și 6.

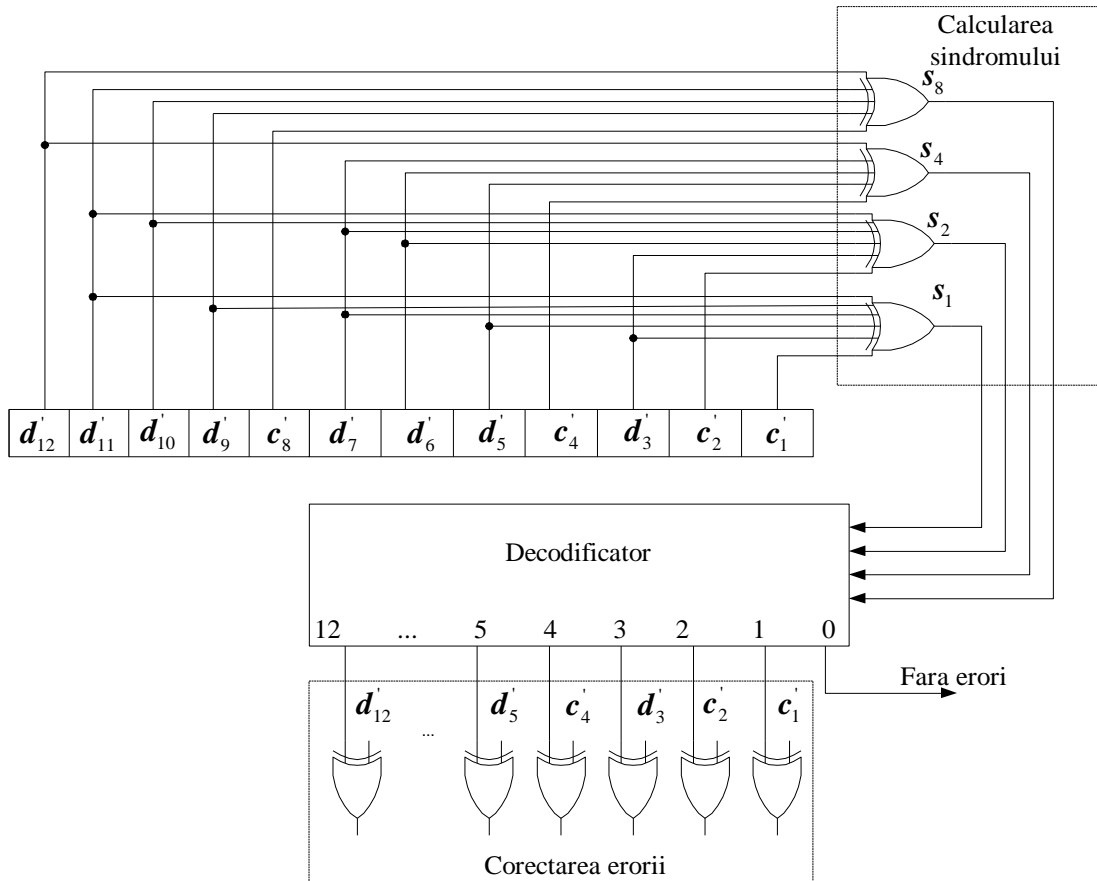
Codorul și decodorul Hamming detector și corector de o eroare

Codorul este format dintr-un registru, care conține n circuite basculante bistabile, în care se stochează simbolurile informaționale în pozițiile corespunzătoare din cuvântul de cod și o serie de sumatoare modulo 2, care calculează simbolurile de control. Simbolurile de control, astfel calculate, se stochează apoi în pozițiile, în care acestea intervin în cuvântul de cod.



Decodorul Hamming este format dintr-un registru cu n circuite basculante bistabile, în care va fi memorat cuvântul recepționat, k sumatoare modulo 2, care calculează componentele sindromului și un decodificator binar cu k intrări și $n+1$ ieșiri. La intrările decodificatorului vor fi aplicați cei k biți ai sindromului. În dependență de semnalul activ de la ieșirea decodificatorului se va stabili dacă cuvântul de cod a fost transmis fără erori sau, în cazul unei erori, se va determina

poziția acesteia. Corectarea prin inversare a simbolului eronat se va efectua prin intermediul a n sumatoare modulo 2.



Codul Hamming detector de două erori și corector de o eroare

În scopul corecției unei erori și a detectării erorilor duble trebuie adăugat un simbol suplimentar, numit de *control al parității*, prin care se decide dacă în cuvântul recepționat sunt două erori sau o eroare.

Structura cuvântului de cod în acest caz este de forma:

$$V = \{d_{12}, d_{11}, d_{10}, d_9, c_8, d_7, d_6, d_5, c_4, d_3, c_2, c_1, c_0\}$$

Simbolul suplimentar de control al parității c_0 determină paritatea tuturor simbolurilor din cuvântul de cod și este calculat conform relației:

$$c_0 = d_{12} \oplus d_{11} \oplus d_{10} \oplus d_9 \oplus c_8 \oplus d_7 \oplus d_6 \oplus d_5 \oplus c_4 \oplus d_3 \oplus c_2 \oplus c_1$$

Sindromul erorii va include un simbol suplimentar s_0 și va avea următoarea formă: $s = \{s_8, s_4, s_2, s_1, s_0\}$

Biții s_8, s_4, s_2 și s_1 vor fi calculați conform relațiilor respective, iar bitul s_0 conform următoarei relații:

$$s_0 = d'_{12} \oplus d'_{11} \oplus d'_{10} \oplus d'_9 \oplus c'_8 \oplus d'_7 \oplus d'_6 \oplus d'_5 \oplus c'_4 \oplus d'_3 \oplus c'_2 \oplus c'_1 \oplus c'_0$$

Criteriile de detectare a erorilor:

$S=(s_8,s_4,s_2,s_1)$	s_0	Erori
0	0	Erori nu sunt
$\neq 0$	1	1 eroare
$\neq 0$	0	2 erori
0	1	Eroare in c'_0

Avantaje:

1. Costul de implementare este relativ mic. Pentru o magistrală de date de 64 biți, algoritmul dat necesită 8 biți de control, la fel cași în cazul utilizării codului cu paritate.
2. Timpul de reținere este proporțional cu $\log_2 N$ și poate necesita încă un ciclu de citire latentă în dependență de structura circuitului de control.

Neajunsuri: Nu este posibil de detectat erori de rang impar, care pot fi confundate cu erori de rangul 1.