

### *Coduri alfanumerice*

Codurile alfanumerice realizează reprezentarea tuturor caracterelor unui alfabet cu ajutorul secvențelor de cifre binare. Alegerea numărului  $n$  de poziții binare necesare este o problemă destul de dificilă, acesta trebuind să îndeplinească, pe lângă condiția (C.15) și o serie de condiții legate de asigurarea protecției la transmiterea informației. Există la ora actuală în uz coduri de 6,7 sau 8 biți.

Un exemplu de cod alfanumeric de 8 biți este codul *EBCDIC* (*Extended Binary Coded Decimal Information Interchange Code*) care în traducere înseamnă „Cod binar-zecimal extins, pentru schimburi de informații”, prezentat în tabelul C.4.

Deși codul EBCDIC cunoaște o largă răspândire, acesta nu este standardizat, spre deosebire de codurile *ASCII* (*American Standard Code for Information Interchange*) în traducere „Cod standard U.S.A. pentru schimburi de informații”. Se cunosc două variante de coduri ASCII, ASCII-7 și ASCII-8, având șapte și respectiv opt biți pe caracter. Codul ASCII-7, reprezentat în tabelul C.3, este recomandat și de Organizația Internațională a Standardelor (*International Standards Organisation*), motiv pentru care poartă și denumirea de cod *ISO*.

Codurile EBCDIC și ASCII prezintă un număr suficient de secvențe pentru codificarea literelor, în ambele variante. În plus, mai rămân destul de multe secvențe neasignate

unor caractere alfanumerice. O parte dintre acestea reprezintă codurile unor semnale de comandă și control în cadrul protocoalelor de transmitere a datelor (început și sfârșit de mesaj, de transmisie, etc.) sau pentru transmiterea unor comenzi specifice la periferice.

Tab.C.3 Codul ASCII-7.

<div>765</div> <div>4321</div>	000	001	010	011	100	101	110	111	
<div>0</div>	<div>1</div>	<div>2</div>	<div>3</div>	<div>4</div>	<div>5</div>	<div>6</div>	<div>7</div>		
0000	0	NUL	DLE	SP	0	@	P	`	p
0001	1	SOH	DC1	!	1	A	Q	a	q
0010	2	STX	DC2	“	2	B	R	b	r
0011	3	ETX	DC3	#	3	C	S	c	s
0100	4	EOT	DC4	\$	4	D	T	d	t
0101	5	ENQ	NAK	%	5	E	U	e	u
0110	6	ACK	SYN	&	6	F	V	f	v
0111	7	BEL	ETB	,	7	G	W	g	w
1000	8	BS	CAN	(	8	H	X	h	x
1001	9	HT	EM	)	9	I	Y	i	y
1010	A	LF	SUB	*	:	J	Z	j	z
1011	B	VT	ESC	+	;	K	[	k	{
1100	C	FF	FS	,	<	L	\	l	
1101	D	CR	GS	-	=	M	]	m	}
1110	E	SO	RS	.	>	N	^	n	~
1111	F	SI	US	/	?	O	—	o	DEL

### Coduri de comandă

NUL	Nul (Null)	DC4	Comanda 4 a dispozitivului (Device control 4)
SOH	Începutul referinței (Start of heading)	NAK	Confirmare de eroare (Negative acknowledgment)
STX	Începutul textului (Start of text)	SYN	Sincronizare (Synchronization)
ETX	Sfârșitul textului (End of text)	ETB	Sfârșitul blocului transmis (End of text block)
EOT	Sfârșitul transmisiei (End of transmission)	CAN	Anulare (Cancel)
ENQ	Căutare (Enquiry)	EM	Terminarea mijlocului de comunicare (End of medium)
ACK	Confirmare (Acknowledgment)	SUB	Înlocuitor (Substitute)
BEL	Avertizare sonoră (Bell)	ESC	Ieșire (Escape)
BS	Spațiu spre stânga (Backspace)	FS	Separator de fișiere (File separator)
HT	Tabulare orizontală (Horizontal tab)	SO	Deplasare către exterior (Shift out)
LF	Salt la linie nouă (Line feed)	SI	Deplasare către interior (Shift in)
VT	Tabulare verticală (Vertical tab)	SP	Spațiu (Space)
FF	Salt la pagină nouă (Form feed)	GS	Separator de grupe (Group separator)
DLE	Ieșire din legătură (Data link escape)	RS	Separator de înregistrări (Record separator)
CR	Început de rând (Carriage return)	US	Separator de unități (Unit separator)
DC1	Comanda 1 a dispozitivului (Device control 1)	DEL	Ștergere (Delete)
DC2	Comanda 2 a dispozitivului (Device control 2)		
DC3	Comanda 3 a dispozitivului (Device control 3)		

Tab.C.4 Codul EBCDIC.

8765 4321	← 00 →				← 01 →				← 10 →				← 11 →			
	00	01	10	11	00	01	10	11	00	01	10	11	00	01	10	11
0000	NUL	DLE	DS		SP	&	—						{	}	\	0
0001	SOH	DC1	SOS				/		a	j	~		A	J		1
0010	STX	DC2	FS	SYN					b	k	s		B	K	S	2
0011	ETX	TM							c	l	t		C	L	T	3
0100	PF	RES	BYP	PN					d	m	u		D	M	U	4
0101	HT	NL	LF	RS					e	n	v		E	N	V	5
0110	LC	BS	ETB	UC					f	o	w		F	O	W	6
0111	DEL	IL	ESC	EOT					g	p	x		G	P	X	7
1000		CAN							h	q	y		H	Q	Y	8
1001		EM							i	r	z		L	R	Z	9
1010	SMM	CC	SM		¢	!		:								
1011	VT	CU1	CU2	CU3	.	\$	,	#								
1100	FF	IFS		DC4	<	*	%	@								
1101	CR	IGS	ENQ	NAK	(	)	—	'								
1110	SO	IRS	ACK		+	;	>	=								
1111	SI	IUS	BEL	SUB		--	?	”								

### Coduri detectoare și corectoare de erori

În timpul prelucrării sau transmisiei datelor în sistemul de calcul, pot apărea erori; în scopul detectării și eventual corectării acestor erori se utilizează coduri cu proprietăți speciale. În general, aceste coduri, conțin pe lângă biții care reprezintă informația respectivă și o serie de *biți de control*, cu ajutorul lor identificându-se și corectându-se eventualele erori apărute.

Cel mai simplu mod de detectare a erorilor de transmisie este utilizarea *bitului de paritate*. Prin aceasta se înțelege atașarea la secvența de cod a fiecărui caracter (de exemplu, la codul 8421 sau EBCDIC) a unui bit suplimentar, a cărui valoare se calculează conform convenției: suma tuturor biților din secvență să fie un număr par (respectiv impar).

Dacă un caracter se exprimă printr-o secvență de cod de patru biți în convenție de paritate impară, atunci calculul bitului de paritate se face conform relației:

$$a_1 \oplus a_2 \oplus a_3 \oplus a_4 \oplus p = 1, \quad (\text{C.20})$$

unde,  $a_i$  reprezintă biții informaționali, iar  $p$  reprezintă bitul de paritate.

La punctul de emisie al informației se calculează bitul de paritate, iar la punctul de recepție se calculează suma biților care va trebui să fie un număr par sau impar conform convenției. În caz de nesatisfacere a acestei condiții, secvența respectivă este eronată.

Trebuie remarcat că prin această metodă nu se poate identifica poziția binară unde a avut loc eroarea și deci nu poate fi corectată. De asemenea, dacă numărul de biți afectat de erori este par, metoda nu detectează secvența de cod ca eronată.

*Codul Hamming.* Hamming a dezvoltat un cod capabil să detecteze și să corecteze o

singură eroare. În acest cod un caracter se exprimă prin șapte biți din care patru sunt *biți informaționali* iar trei *biți de control*:

$$c = p_1 p_2 a_3 p_4 a_5 a_6 a_7, \quad (\text{C.21})$$

unde  $p_i$  cu  $i \in \{1,2,4\}$  sunt biți de control, iar  $a_i$  cu  $i \in \{3,5,6,7\}$  – biți de informație. Biții de control  $p_i$  se calculează astfel:

$$\begin{aligned} s_1 &= p_1 \oplus a_3 \oplus a_5 \oplus a_7 = 0, \\ s_2 &= p_2 \oplus a_3 \oplus a_6 \oplus a_7 = 0, \\ s_3 &= p_4 \oplus a_5 \oplus a_6 \oplus a_7 = 0. \end{aligned} \quad (\text{C.22})$$

La punctul de emisie se calculează valorile biților  $p_i$ , iar la punctul de recepție se verifică sumele  $s_i$  ( $i=1,2,3$ ). Dacă sumele  $s_i$  sunt toate 0, atunci secvența de cod s-a recepționat corect. În caz contrar numărul zecimal codificat binar prin biții  $s_i$  de ponderi  $2^2, 2^1, 2^0$ , indică poziția eronată din secvența de cod transmisă.

Există o mare varietate de coduri cu proprietăți de detecție a erorilor; alegerea unui anumit cod se face în funcție de numărul erorilor care trebuie detectate și numărul biților de control care sunt necesari în acest scop, întrucât costul transmisiei crește odată cu lungimea secvenței de cod.