

Universitatea Tehnica a Moldovei

Facultatea Calculatoare, Informatica si Microelectronica

Departamentul Informatica si Ingineria Sistemelor

Disciplina:

Interfete si Retele Industriale

Tema Nr. 5.1 Codificarea Informatiei in RC.

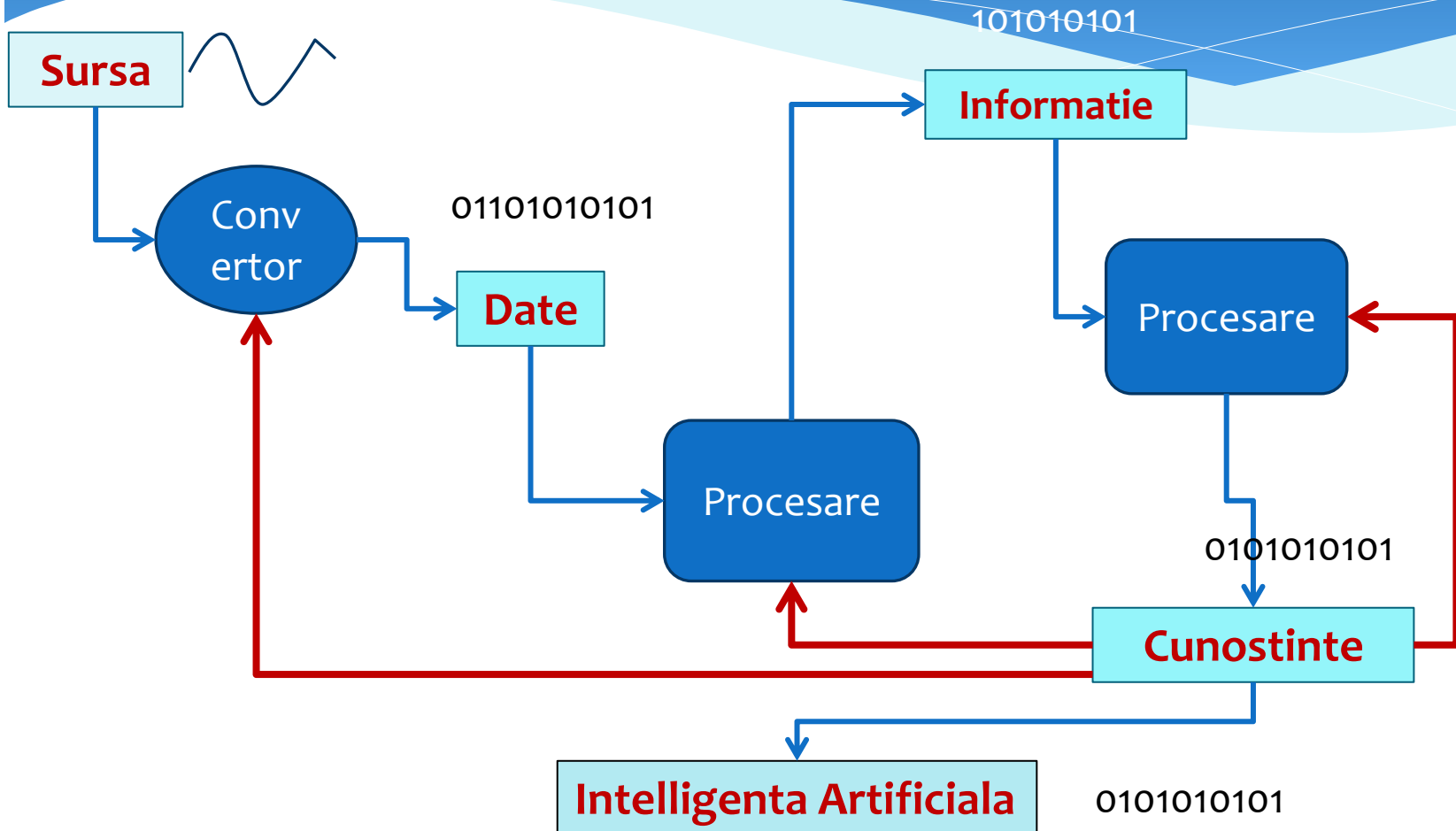
Titular de curs:

Conf.univ.,dr. V. Ababii

Subiecte abordate:

- * Surse discrete de informație.
- * Canale discrete de transmisiuni.
- * Codarea surselor discrete de informație pe canale neperturbate.
- * Codarea surselor discrete de informație pe canale perturbate.
- * Codurile binar, NRZ, unipolar, bipolar, Manchester II, AMI.
- * Codificatorul Manchester II.
- * Decodificatorul Manchester II.
- * Rețele în baza codului Manchester II.
- * Codificarea datelor în baza codului Manchester II.
- * Codificatorul și decodificatorul NRZ, unipolar și bipolar.
- * Coduri detectoare și corectoare de erori: Cod Hamming, Codul Gray, Coduri grup, Coduri ciclice, Coduri Reed – Muller, Coduri convenționale.
- * Compresia datelor (text, imagini, audio, vorbire). Protecția surselor de date.

Raportul: Date – Informatie - Cunostinte



Raportul: Date – Informație - Cunoștințe

Datele reprezintă fapte, situații de producție, deci un material brut care poate fi prelucrat în diferite scopuri și care constituie, prin natura și dimensiunea lor, suportul final al informațiilor. Din punct de vedere funcțional, datele se caracterizează prin : identitate (ceea ce semnifică obiectul acesteia), caracteristici (care îi descriu identitatea) și masuri (reprezentând mărimile valorice care sînt asociate caracteristicilor).

Informația¹ este o comunicare, un mesaj, care conține elemente noi — în raport cu ce cunoștea pînă atunci utilizatorul ei — privind caracterizarea unei anumite situații, fenomen, fapt, proces economic etc., în scopul declanșării acțiunii. Ca atare, informația constituie un instrument de cercetare a proceselor care au loc în diferite sisteme, reflectînd legăturile obiective cauză-efect din mediul înconjurător.

Informatica este știința ce studiază structura informației și metodele de prelucrare a ei cu ajutorul calculatorului (în engleză Informatics).

Raportul: Date – Informație - Cunostinte

Un **sistem informațional** se poate defini ca fiind ansamblul de elemente implicate în procesul de colectare, de transmisie, prelucrare de informație, informației revenindu-i rolul central din sistem.

În cadrul sistemului informațional se regăsesc : informația vehiculată, documentele purtătoare de informații, personalul, mijloace de comunicare, sistemele de prelucrare (de regulă, automată) a informației, etc. Printre posibilele activități desfășurate în cadrul acestui sistem, pot fi enumerate : achiziția de informații din sistemul de bază, completarea documentelor și transferul acestora între diferite compartimente, centralizarea datelor, etc.

În cadrul unui sistem informațional, majoritatea activităților se pot desfășura cu ajutorul tehnicii de calcul. Se pot prelucra datele primare și apoi rezultatul poate fi transferat mai departe, către alt compartiment spre prelucrare. Transferul se poate face și el pe cale electronică prin intermediul unei rețele de calculatoare.

Ansamblul de elemente implicate în tot acest proces de prelucrare și transmitere a datelor pe cale electronică alcătuiesc un sistem informatic.

Într-un sistem informatic pot intra: calculatoare, sisteme de transmisie a datelor, componente hardware și software, datele prelucrate, personalul ce exploatează tehnica de calcul, teoriile ce stau la baza algoritmilor de prelucrare, etc.

Raportul sistem informațional-sistem informatic: sistemul informațional include în cadrul său sistemul informatic, acesta din urmă fiind o componentă esențială a primului.

Coduri binare si zecimal - binare

0101010101010101010101011101010100010101010101010110101010110110

Bistabil:

RS

JK

D

Memorie:

Statica

Dinamica

Operativa

Permamenta

Numere în zecimal	CODURI ZECIMAL-BINARE							
	Coduri ponderate				Coduri neponderate			
	8421	2421	4221	7421	Exces3	Gray	2 din5	8421 cu bit de paritate impară
0	0000	0000	0000	0000	0011	0000	00011	10000
1	0001	0001	0001	0001	0100	0001	00101	00001
2	0010	0010	0010	0010	0101	0011	00110	00010
3	0011	0011	0011	0011	0110	0010	01001	10011
4	0100	0100	0100	0100	0111	0110	01010	00100
5	0101	1011	1001	0101	1000	0111	01100	10101
6	0110	1100	1100	0110	1001	0101	10001	10110
7	0111	1101	1101	0111	1010	0100	10010	00111
8	1000	1110	1110	1001	1011	1100	10100	01000
9	1001	1111	1111	1010	1100	1101	11000	11001

Codul ASCII pe 7 biti

- C** – 100 0011 (coloana 100 linia 0011)
- &** – 010 0110 (coloana 010 linia 0110)
- 9** - 011 1001 (coloana 011 linia 1001).



b ₃ b ₂ b ₁ b ₀	b ₆ b ₄ b ₅							
	000	001	010	011	100	101	110	111
0000	NULL	DLE		0	@	P	`	p
0001	SOH	DC1	!	1	A	Q	a	q
0010	STX	DC2	"	2	B	R	b	r
0011	ETX	DC3	#	3	C	S	c	s
0100	EOT	DC4	\$	4	D	T	d	t
0101	ENQ	NAK	%	5	E	U	e	u
0110	ACK	SYN	&	6	F	V	f	v
0111	BEL	ETB	'	7	G	W	g	w
1000	BS	CAN	(8	H	X	h	x
1001	HT	EM)	9	I	Y	i	y
1010	LF	SUB	*	:	J	Z	j	z
1011	VT	ESC	+	;	K	[k	{
1100	FF	FS	,	<	L	\	l	
1101	CR	GS	-	=	M]	m	}
1110	SO	RS	.	>	N	^	n	~
1111	SI	US	/	?	O	_	o	DEL

Prezemntarea fizica a codurilor bineare

Codul binar

Bitii dintr-un semnal binar pot avea numai doua valori discrete notate simbolic cu „0” si „1”.

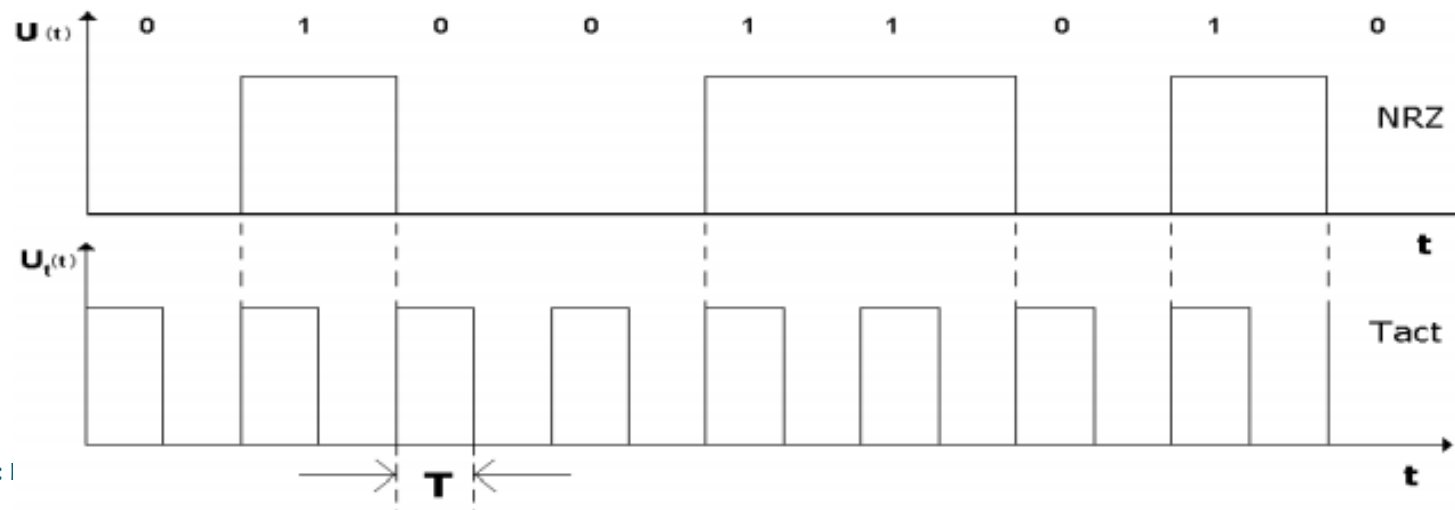
Codul binar poate aparea in doua forme si anume:

- cod NRZ (Non Return – to – Zero code)
- cod RZ (Return – to – Zero code)

In cazul codului NRZ bitii individuali se succed fara nici o pauza, cu alte cuvinte, elementul binar isi mentine valoarea pe toata durata T a unei perioade de tact

In cadrul codului RZ, elementul binar isi mentine valoarea (0 sau 1) numai pe o semiperioada a semnalului de tact, luând in mod obligatoriu valoarea 0 pe durata celeilalte semiperioade

Codul NRZ



Prezemntarea fizica a codurilor bineare

Codul binar

Bitii dintr-un semnal binar pot avea numai doua valori discrete notate simbolic cu „0” si „1”.

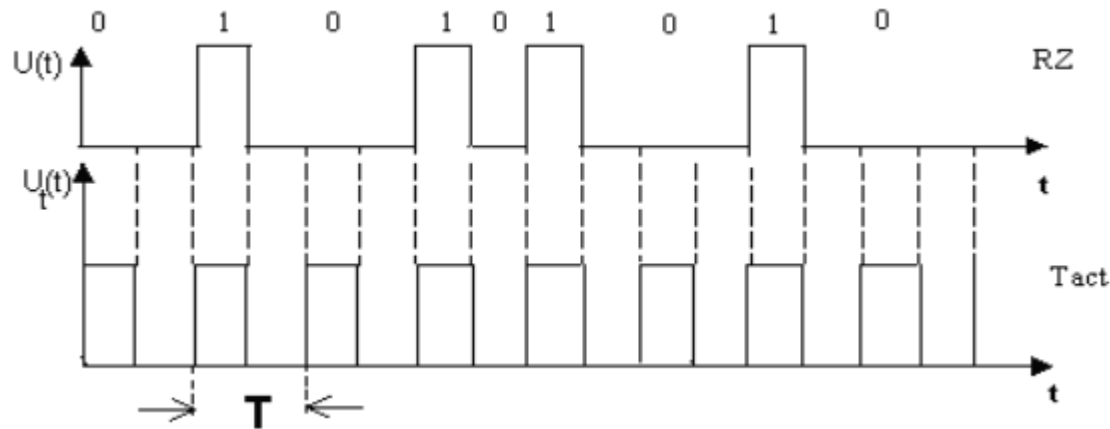
Codul binar poate aparea in doua forme si anume:

- cod NRZ (Non Return – to – Zero code)
- cod RZ (Return – to – Zero code)

In cazul codului NRZ bitii individuali se succed fara nici o pauza, cu alte cuvinte, elementul binar isi mentine valoarea pe toata durata T a unei perioade de tact

In cadrul codului RZ, elementul binar isi mentine valoarea (0 sau 1) numai pe o semiperioada a semnalului de tact, luând in mod obligatoriu valoarea 0 pe durata celeilalte semiperioade

Codul RZ

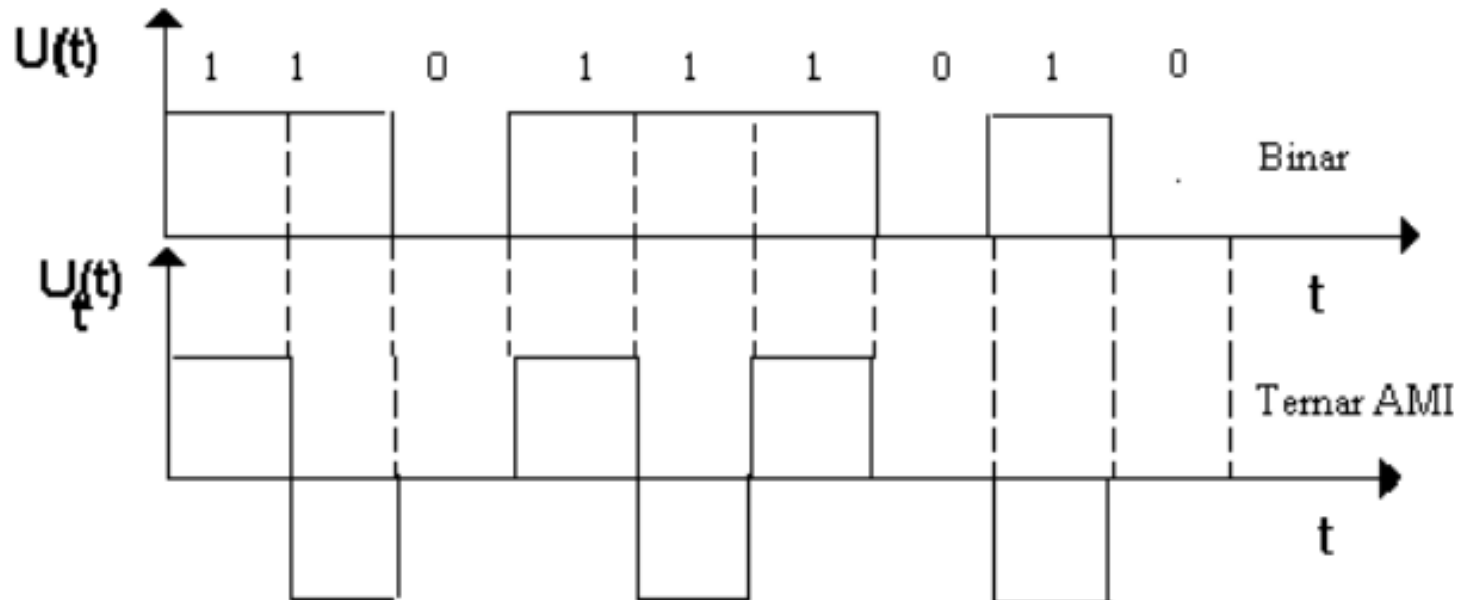


Prezemntarea fizica a codurilor bineare

Codul AMI (Alternate Mark Inversion).

Conversia unui semnal AMI se realizeaza astfel

- bitii de valoare 1 din semnal sunt redati in AMI prin impulsuri de tensiune care au alternativ polaritate pozitiva si negativa;
- bitii de valoare 0 din semnal sunt redati in AMI prin tensiune nula.



Prezemntarea fizica a codurilor bineare

Codul HDB3

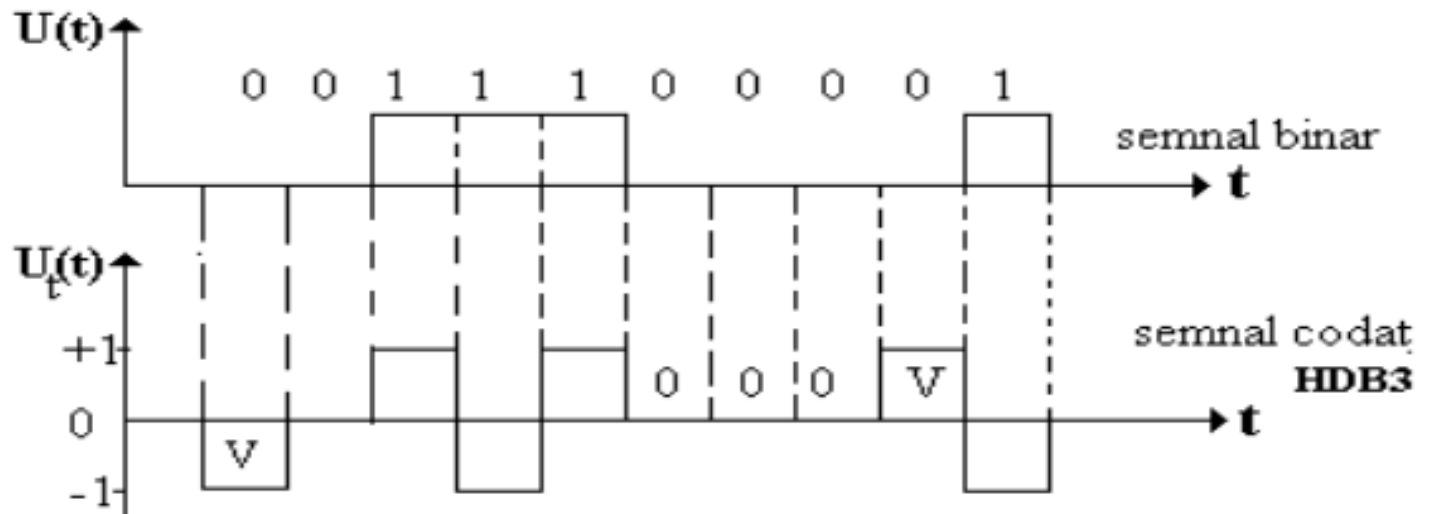
Denumirea acestui cod (High Density Bipolar) indica faptul ca el asigura o densitate (frecventa) ridicata a impulsurilor cu polaritate alternativa. Cifra 3 indica faptul ca intr-un semnal prelucrat dupa acest cod nu apar niciodata mai mult de trei zerouri consecutive.

Codarea unui semnal binar se realizeaza dupa urmatoarele reguli

a) se asociaza elementelor binare 1 impulsuri in linie de polaritati alternând ca semn cu o durata egala cu a elementului binar; elementului binar 0 ii corespunde absenta semnalului;

b) daca exista o secventa de 4 zerouri succesive atunci cel de-al patrulea zero se inlocuieste cu un element redundant V care poate fi recunoscut si poate fi eliminat la receptie deoarece are aceeasi polaritate cu elementul de semnal 1 anterior.

Elementele redundante V trebuie sa alterneze ca semn.



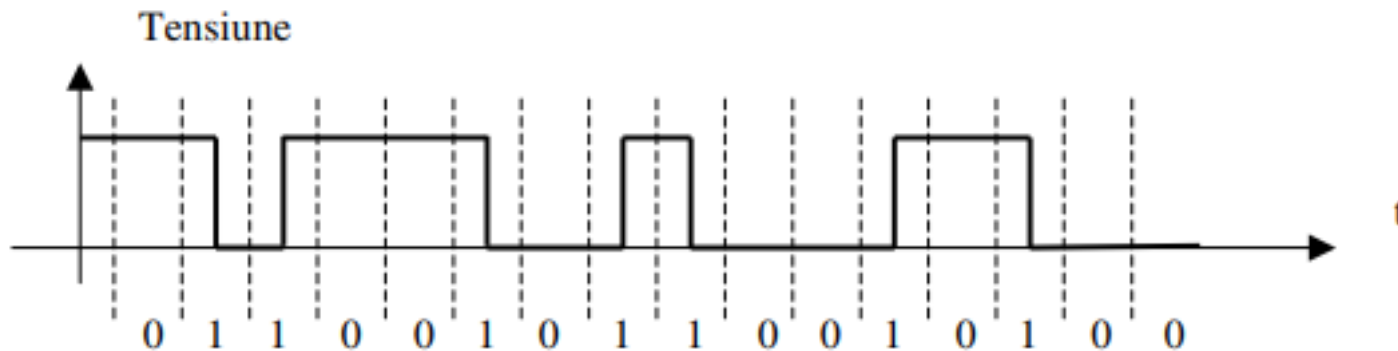
Prezemntarea fizica a codurilor bineare

Codarea NRZI (Non Return to Zero Inverted)

Este o codare asemănătoare cu NRZ, cu regula de codare:

1. Zero se codifică cu lipsa unui front.
2. Unu se codifică cu un front.

În figura 16 biți într-o succesiune aleasă la întâmplare au fost codificați cu 7 tranziții (2,28 biți/tranziție). Dacă apare un șir de 0-uri sau 1-uri semnalul electric nu are variații.



Prezemntarea fizica a codurilor bineare

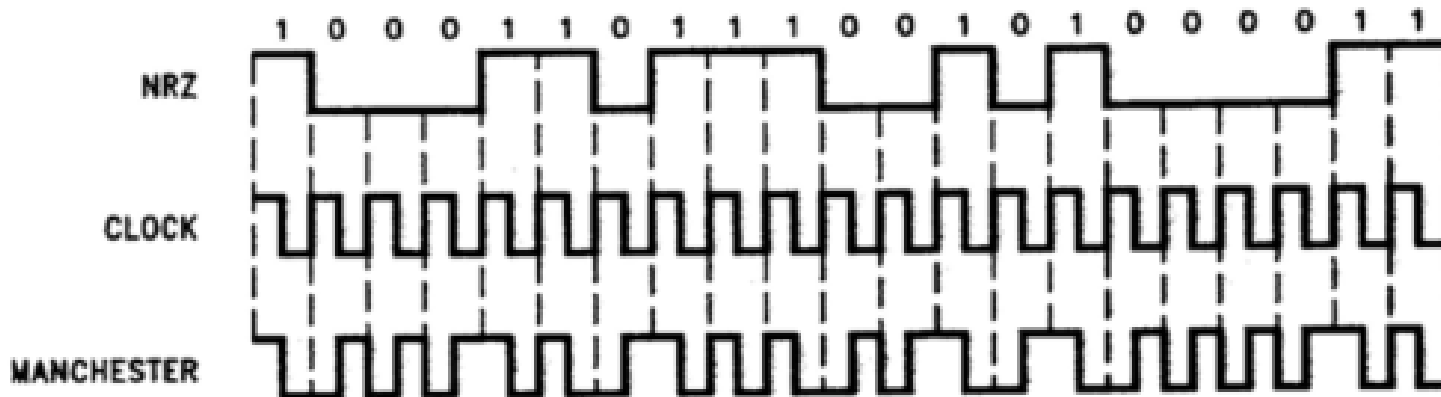
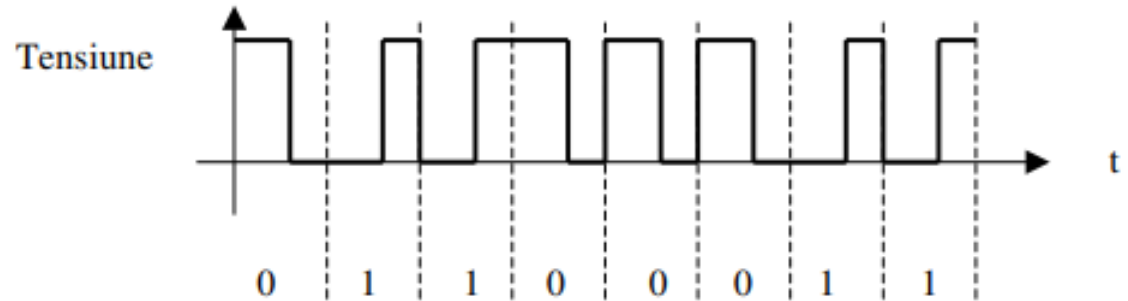
Codarea Manchester

Codarea Manchester a fost realizată prima oară la Universitatea din Manchester cu ocazia construirii calculatorului Mark 1 (numit și Baby) în 1948.

Regula de codare este:

1. Zero se codifică cu un front descrescător;
2. Unu se codifică cu un front crescător.

În figura 8 biți într-o succesiune aleasă la întâmplare au fost codificați 8 biți cu 12 tranziții (1,5 tranziții pe bit). Dacă apare un șir de 0-uri sau 1-uri semnalul electric are variații. Codarea Manchester se folosește la rețeaua Ethernet.



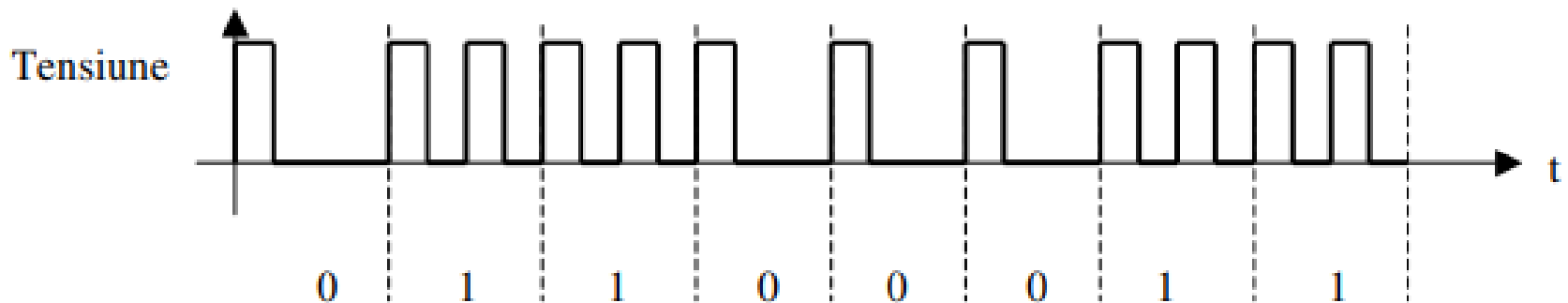
Prezemntarea fizica a codurilor bineare

Codarea FM (Frequency Modulation)

Codarea FM și MFM (Modified FM) au fost introduse de IBM în anii 1970 pentru codificarea datelor la scrierea pe hard discuri. La aceste codări fiecare celulă bit începe cu un impuls de tact, iar regula de codare este:

1. Zero se codifică cu lipsa unui impuls.
2. Unu se codifică cu un impuls.

În figura 8 biți într-o succesiune aleasă la întâmplare au fost codificați cu 24 tranziții (3 tranziții pe bit). Codarea FM s-a folosit la înscrierea datelor pe suporturi magnetice, în prezent fiind înlocuită cu codări mai eficiente.



Eficiența codurilor binare

Un tabel centralizator care cuprinde exemplele prezentate este dat în continuare:

Codare	Eficiență (biți/tranziție)
NRZ	2.28
NRZI	2.28
Manchester	0.66
FM	0.33

Codare	Eficiență (biți/tranziție)	Autosincronizabilitate
NRZ	2.28	NU
NRZI	2.28	NU
Manchester	0.66	DA
FM	0.33	DA

Coduri detectoare de erori

Distanța Hamming

Pe lângă distanțele prezentate într-un material anterior, o distanță de un interes deosebit în problema noastră este distanța Hamming, numită astfel după Richard Hamming, cel care a introdus-o în 1950. **Distanța Hamming** între doi vectori de dimensiuni egale este dată de **numarul de poziții în care aceștia diferă**. Ea măsoară astfel numărul de schimbări care trebuie făcute într-un vector pentru a îl obține pe celălalt, sau reformulat numărul de *erori* care transformă un vector în celălalt.

Exemple:

vector 1	codare	126359	01101011
vector 2	notate	226389	01001110
distanța Hamming	3	2	3

Coduri detectoare de erori cu multiplicare

Daca la receptie obtinem coduri 00 respectiv 11 am receptionat corect 0 respectiv 1 iar daca obtinem 01 sau 10 am detectat o eroare singulara (fara a putea decide însa nimic în sensul corectiei ei).

Exemplu:

mesaj initial:	0.1.0.0.1.0.1.1.0.1
mesaj codat:	00.11.00.00.11.00.11.11.00.11
mesaj receptionat cu eroare:	00.11.00.00. 10 .00.11.11.00.11 => detectie de eroare

Daca la receptie obtinem coduri 000 respectiv 111 am receptionat corect 0 respectiv 1. Daca obtinem 001 sau 010 sau 100 am detectat o eroare singulara si putem **corecta** spre 000 (deci am receptionat 0) iar daca obtinem 011 sau 101 sau 110 am detectat o eroare singulara si putem **corecta** spre 111 (deci am receptionat 1).

Exemplu:

mesaj initial:	0.1.0.0.1.0.1.1.0.1
mesaj codat:	000.111.000.000.111.000.111.111.000.111
mesaj receptionat cu eroare:	000. 101 .000.000.111.000.111.111. 001 .111
mesaj corectat:	000. 111 .000.000.111.000.111.111. 000 .111

=> **detectie** si corectie de erori singulare

Coduri detectoare de erori (Hamming 7,4)

Codul Hamming (7,4)

Unul din cele mai cunoscute coduri detectoare si corectare de erori singulare este codul numit Hamming (7,4). Acesta notatie indica faptul ca avem un cod de 7 biti din care 4 sunt biti de date independenti (restul fiind biti redundanti, reprezentând paritatea a diferite combinatii a bitilor de date). Codul contine deci 4 biti de date d_1, d_2, d_3, d_4 si 3 biti de paritate p_1, p_2, p_3 . Bitii de paritate sunt calculati astfel:

p_1 sumeaza d_1, d_2, d_4

p_2 sumeaza d_1, d_3, d_4

p_3 sumeaza d_2, d_3, d_4

iar organizarea bitilor în cuvântul de cod este urmatoarea:

p_1	p_2	d_1	p_3	d_2	d_3	d_4
-------	-------	-------	-------	-------	-------	-------

De obicei se definesc doua matrici în legatura cu acest cod: matricea generatoare de cod G si matricea de detectie a paritatii H

$$G = \begin{pmatrix} 1 & 1 & 0 & 1 \\ 1 & 0 & 1 & 1 \\ 1 & 0 & 0 & 0 \\ 0 & 1 & 1 & 1 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix} \quad H = \begin{pmatrix} 1 & 0 & 1 & 0 & 1 & 0 & 1 \\ 0 & 1 & 1 & 0 & 0 & 1 & 1 \\ 0 & 0 & 0 & 1 & 1 & 1 & 1 \end{pmatrix}$$

Coduri detectoare de erori (Hamming 7,4)

Se constata în G ca liniile 1,2 si 4 calculeaza sumele aferente paritatilor respective iar liniile 3,5,6,7 simplu copiaza bitii de date. În H cele 3 linii calculeaza paritatile corespunzatoare.

La codare: se determina cuvântul de cod calculând paritatile corespunzatoare (se poate utiliza atât paritatea para cât si paritatea impara - în exemplele urmatoare vom folosi paritatea impara).

La decodare: se calculeaza paritatile corespunzatoare si se verifica cu cele corecte (în fapt se sumeaza si cu paritatile corecte si se verifica sa rezulte 0).

Exemplu de codare-decodare pentru codul Hamming (7,4)

La codare: fie cuvântul de cod 1001. Calculam:

$$p1 = 1+0+1 = 0$$

$$p2 = 1+0+1 = 0$$

$$p3 = 0+0+1 = 1$$

rezultând codul Hamming 0011001 (am reprezentat subliniat bitii de paritate).

Coduri detectoare de erori (Hamming 7,4)

La decodare (cazul 1): presupunem ca am receptionat 0011001 (deci fara erori)

Reconstituim bitii de date: 1 0 0 1

Verificam paritatile:

$$p1 + d1 + d2 + d4 = 0 + 1 + 0 + 1 = 0$$

$$p2 + d1 + d3 + d4 = 0 + 1 + 0 + 1 = 0$$

$$p3 + d2 + d3 + d4 = 1 + 0 + 0 + 1 = 0$$

Cum toate aceste valori sunt 0 rezulta ca nu am avut eroare.

La decodare (cazul 2): presupunem ca am receptionat 0011011 (deci cu o eroare în zona de date)

Reconstituim bitii de date: 1 0 1 1

Verificam paritatile:

$$p1 + d1 + d2 + d4 = 0 + 1 + 0 + 1 = 0$$

$$p2 + d1 + d3 + d4 = 0 + 1 + 1 + 1 = 1$$

$$p3 + d2 + d3 + d4 = 1 + 0 + 1 + 1 = 1$$

Cum aceste valori nu sunt toate 0 rezulta ca am avut eroare (deci am realizat **detectie** de eroare). În plus codul format din pozitiile eronate (aferent p3p2p1) având valoarea 110 indica bitul 6 ca fiind eronat deci al treilea bit de date trebuie corectat (deci am realizat **corectie** de eroare). Deci, bitii de date corectati sunt 1 0 0 1.

Coduri detectoare de erori (Hamming 7,4)

La decodare (cazul 3): presupunem ca am receptionat 0111001 (deci cu o eroare în zona de paritate)

Reconstituim bitii de date: 1 0 0 1

Verificam paritatile:

$$p1 + d1 + d2 + d4 = 0 + 1 + 0 + 1 = 0$$

$$p2 + d1 + d3 + d4 = 1 + 1 + 0 + 1 = 1$$

$$p3 + d2 + d3 + d4 = 1 + 0 + 0 + 1 = 0$$

Cum aceste valori nu sunt toate 0 rezulta ca am avut eroare (deci am realizat **detectie** de eroare). În plus codul format din pozitiile eronate (aferent $p_3p_2p_1$) având valoarea 010 indica bitul 2 ca fiind eronat deci al doilea bit de paritate era eronat iar bitii de date erau corecti (deci am realizat **corectie** de eroare).

Codul Hamming (7,4) descris anterior are distanta Hamming între oricare cuvintele de cod valide minim 3 si deci poate **detecta si corecta erori singulare** sau poate doar detecta erori duble (fara a putea corecta nici un fel de erori).

Calcululele anterioare se pot face si în forma matriciala considerând la codare înmultirea matricii G cu vectorul coloana al bitilor de date iar la verificarea paritatii înmultirea matricii H cu vectorul coloana al codului Hamming.