

S1. Recapitulare

Sisteme numerice. Conversii de bază

(Sistemul zecimal, octal, binar și hexazecimal)

1. Obiective

- O1. Deprinderea modului de codificare a informației
- O2. Operarea cu sistemul zecimal, octal, hexa și binar
- O3. Transformarea valorilor în diferite baze numerice

2. Considerații teoretice

2.1 Notăția zecimală

Notăția zecimală este scrierea numerelor în sistemul de numerație baza 10. Pentru reprezentarea oricăror numere, indiferent cât de mari, se folosesc exact 10 cifre, având 10 valori diferite: 0, 1, 2, 3, 4, 5, 6, 7, 8 și 9.

Sistemul zecimal este un sistem de numerație pozițional, în care poziția fiecărei cifre indică înmulțirea valorii cifrei respective cu o putere a lui 10. Fiecare poziție indică o valoare de 10 ori mai mare decât poziția din dreapta sa. Dacă există parte fracționară, ea urmează după număr, după un separator zecimal, ex. virgulă. Semnul numărului este plasat înaintea lui și este unul dintre simbolurile + (pentru numere pozitive) sau – (pentru numere negative).

Thousands	Hundreds	Tens	Units
10^3	10^2	10^1	10^0
1000	100	10	1
5	9	7	3

2.2 Notăția octală

Există 8 cifre unice în acest sistem: 0, 1, 2, 3, 4, 5, 6, 7. deci orice număr reprezentat în baza 8 este format dintr-o combinație a acestora.

2.3 Notăția binară

În orice sistem informatic, conform definiției lui *Turing*, este nevoie de o memorie fiabilă (sigură în funcționare). Cea mai fiabilă metodă de prelucrare și respectiv de stocare a datelor de calculator se bazează pe sistemul binar: "celula este magnetizată sau nu este magnetizată", "trece curent sau nu trece curent", "cartela este perforată sau nu este perforată" etc.

Cea mai apropiată alternativă ar fi fost utilizarea numerației în baza 3 (sistemul ternar), ceea ce ar fi necesitat componente electronice cu trei stări stabile, ducând la o logică trivalentă de genul "senzorul nu este excitat, senzorul este puțin excitat, senzorul este excitat", ceea ce, deși induce un nivel mai bun de discernere a informației, reduce, cel puțin în principiu, fiabilitatea potențială a sistemului.


În sistemul (de numerație) binar există doar două cifre posibile, 0 și 1. Conform definiției lui *Claude Shannon*, o cifră binară conține cantitatea de informație de 1 bit. Sistemul binar este în același timp și cel mai natural mod de stocare a informației în domeniul calculatoarelor, deoarece acolo 1 bit (celula de memorie cu capacitate minimă) găzduiește unitatea elementară de informație: valoarea bitului poate fi ori un 0, ori un 1.




Datorită ușurinței implementării sistemului binar în circuitele electronice, el se folosește practic la toate calculatoarele moderne.

One-hundred and twenty-eights	Sixty-fours	Thirty-twos	Sixteens	Eights	Fours	Twos	Units
2^7	2^6	2^5	2^4	2^3	2^2	2^1	2^0
128	64	32	16	8	4	2	1
0	1	1	0	1	0	1	0

2.4 Notăția hexazecimală

Hexadecimal	Binary	Denary
0	0000	0
1	0001	1
2	0010	2
3	0011	3
4	0100	4
5	0101	5
6	0110	6
7	0111	7
8	1000	8
9	1001	9
A	1010	10
B	1011	11
C	1100	12
D	1101	13
E	1110	14
F	1111	15



What we see/hear		Inside computers
Text	a,b,c	01100001,01100010,01100011
Number	1,2,3	00000001,00000010,00000011
Sound		01001100010101000110100...
Image		10001001010100000100111...
Video		00110000001001101011001...

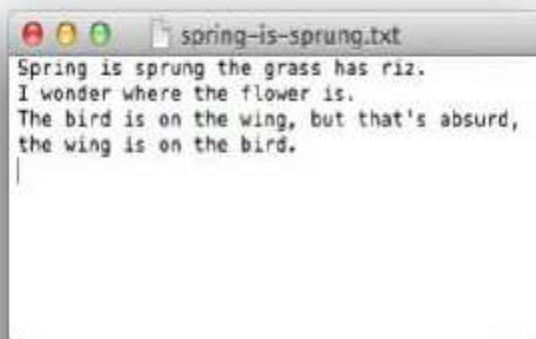
Discrete/digital and binary!

Bit: o valoare binară (0 sau 1), denumirea provenind din **B**inary digiT

- 1 bit poate codifica 2 tipare (patterns)
- 2 biți pot codifica 4 tipare
- 8 biți pot codifica 256
- n biți pot codifica 2^n tipare

Byte: grup de 8 biți – unitate de stocare de bază

- numerele se vor stoca începând de la 0, deci vor rezulta 255 tipare unice
- 1 byte conține un număr între 0...255
- caracterele ASCII (ex: 32-space, 37-%, 65-A, 97-a, 98-b, 99-c etc)
- valori culori RGB



Underlying bytes in RAM

s	p	r	i	...
83	112	114	105	

2.5 Conversii

2.5.1 Din zecimal în binar

Observație:

Sistemul binar permite codarea primelor 8 cifre zecimale cu ajutorul a trei biți. Bitul cel mai nesemnificativ se află pe ultima poziție iar cel mai semnificativ se situează pe prima poziție. Semnificația biților crește astfel de la dreapta la stânga.

Zecimal		Binar		Justificare
0	→	0 0 0	→	$0 * 2^0 + 0 * 2^1 + 0 * 2^2$
1	→	0 0 1	→	$1 * 2^0 + 0 * 2^1 + 0 * 2^2$
2	→	0 1 0	→	$0 * 2^0 + 1 * 2^1 + 0 * 2^2$
3	→	0 1 1	→	$1 * 2^0 + 1 * 2^1 + 0 * 2^2$
4	→	1 0 0	→	$0 * 2^0 + 0 * 2^1 + 1 * 2^2$
5	→	1 0 1	→	$1 * 2^0 + 0 * 2^1 + 1 * 2^2$
6	→	1 1 0	→	$0 * 2^0 + 1 * 2^1 + 1 * 2^2$
7	→	1 1 1	→	$1 * 2^0 + 1 * 2^1 + 1 * 2^2$

Pentru a trece din codul binar în forma zecimală se înmulțește fiecare valoare (bit) 0 sau 1 de la dreapta la stânga cu 2 la puterea k , unde $k \in \mathbb{N}$, $k \in \{0,1,2,3,\dots,n\}$, puterea k crescând tot de la dreapta la stânga.

La apariția cifrei 8 însă, codarea pe trei biți devine insuficientă, ceea ce atrage după sine adăugarea unui bit pe poziția cea mai semnificativă din șir, după exemplul dat:

$$8 \rightarrow \mathbf{1\ 0\ 0\ 0} \rightarrow 0 * 2^0 + 0 * 2^1 + 0 * 2^2 + \mathbf{1} * 2^3$$

Procedura prin care se realizează conversia din sistemul zecimal în cel binar seste cunoscută sub denumirea de **împărțire repetată la 2**. La fiecare divizare vom reține restul împărțirii la 2 iar scrierea șirului de biți se face de la ultimul rest obținut spre primul rest din coloană.

Exemplul 1:

Să se convertească cifra 9 din bază zecimală în bază binară.

$$\begin{array}{rcl} 9/2 = & 4 & r = \mathbf{1} \\ 4/2 = & 2 & r = \mathbf{0} \\ 2/2 = & 1 & r = \mathbf{0} \\ 1/2 = & 0 & r = \mathbf{1} \end{array} \quad \uparrow$$

Astfel, vom scrie:

$$9_{10} = 1001_2$$

Exemplul 2:

Să se convertească numărul 0,625 din bază zecimală în sistemul binar.

În cazul unui număr unde avem parte fracționară (valorile după virgulă) vom recurge la o metodă de rezolvare bazată pe 2 pași:

Pasul 1:

Se multiplică cifra zecimală cu 2. Partea întreagă ce rezultă este *prima cifră binară* în partea dreaptă a punctului (virgulei):

Pentru că $0.625 * 2 = 1.25$, prima cifră va fi 1.

Pasul 2:

Ignorând partea întreagă din rezultatul precedent (în cazul nostru cifra 1), înmulțim din nou cu 2, iar partea întreagă din rezultat va fi *a doua cifră binară* din dreptul punctului.

Se va continua până când:

- partea fracționară va fi 0 sau
- observăm o repetiție

Pentru că $0,25 * 2 = 0.50$, a doua cifră va fi 0.

Deoarece de data aceasta nu avem parte întreagă (rezultatul a fost 0.50), nu avem nicio cifră de ignorat. Înmulțim cu 2, iar partea întreagă va fi *a treia cifră*.

Pentru că $0.50 * 2 = 1.00$, a treia cifră va fi 1.

Dat fiind faptul că partea fracționară este 0, algoritmul se va opri aici:

$$0.625_{10} = 0.101_2$$

2.5.2 Din binar în zecimal

Exemplul 3:

Să se convertească șirul de biți 11101.01 din baza 2 în baza 10.

Pentru a simplifica sarcina de lucru vom apela la întocmirea următorului tabel de valori:

Position	2^4	2^3	2^2	2^1	2^0	2^{-1}	2^{-2}
Values	16	8	4	2	1	0.5	0.25
Binary Digits	1	1	1	0	1	.	0 1

Conform tabelului de mai sus vom redacta expresia:

$$\begin{aligned}
 11101.1 &= 1 * 2^4 + 1 * 2^3 + 1 * 2^2 + 0 * 2^1 + 1 * 2^0 + 0 * 2^{-1} + 1 * 2^{-2} \\
 &= (1 * 16) + (1 * 8) + (1 * 4) + (0 * 2) + (1 * 1) + (0 * 0.5) + (1 * 0.25) \\
 &= 16 + 8 + 4 + 0 + 1 + 0 + 0.25 \\
 &= 29.25
 \end{aligned}$$

Deci: $11101.1_2 = 29.25_{10}$

Observație:

În general:

$$3^2 = 3 * 3 = 9$$

$$3^{-2} = 1 * 3^{-2} = \frac{1}{3^2} = \frac{1}{3 * 3} = \frac{1}{9} = 0.111111$$

2.5.3 Din zecimal în octal**Exemplul 4:**

Reprezentați numărul 31 din baza 10 în sistemul octal.

Metoda de rezolvare pentru acest exercițiu este similară cu algoritmul prezentat în 2.5.2 *Conversia din zecimal în binar* (vezi pagina 6 – **Exemplul 1**). Diferența aici constă în faptul că divizarea nu se va mai face la valoarea 2 ci, fiind vorba de sistemul octal, vom împărți repetat la cifra 8.

$$\begin{array}{rcl} 31/8 & = & 3 \quad r = 7 \\ 3/8 & = & 0 \quad r = 3 \end{array} \quad \uparrow$$

$$\text{Deci: } 31_{10} = 37_8$$

2.5.4 Din zecimal în hexazecimal**Exemplul 5:**

Reprezentați același număr de la *Exemplul 4* în sistemul hexazecimal.

$$\begin{array}{rcl} 31/16 & = & 1 \quad r = 15 \\ 1/16 & = & 0 \quad r = 1 \end{array} \quad \uparrow$$

$$\text{Deci: } 31_{10} = 1F_{16} \text{ (vezi Tabelul pagina 3)}$$

2.5.5 Din binar în octal

Exemplul 6:

Să se convertească următorul șir de biți 11010011 din baza 2 în sistemul octal.

Soluție:

Grupăm biții câte 3 (valoarea maximă în sistemul octal este 7, reprezentat de 111_2), începând din dreapta spre stânga, iar dacă îi nevoie, completăm cu 0-uri biții neacoperiți:

011 | 010 | 011 – add extra bit

3 2 3

$$11010011_2 = 323_8$$

2.5.6 Din binar în hexazecimal

Exemplul 7:

Să se convertească șirul de biți 10011101 din sistemul binar în bază hexazecimală.

Soluție:

Se vor grupa biții câte 4 (valoarea maximă în hexa este 15, F), iar pentru 10011101 vom avea:

1001 | 1101

9 13

9 D

Vom scrie:

$$10011101_2 = 9D_{16}$$

3. Referințe bibliografice

Suport didactic laborator *”Baze de numerație. Conversii de bază”* – Ing. Drd. Gabriel Barina, 2016