# BROJNI SISTEMI





## POZICIONI BROJNI SISTEMI

~~OSTALI BROJNI SISTEMI KAO ŠTO SU VAVILONSKI, RIMSKI, MAJANSKI, …~~

Hindu-Arabik ili Indo-Arabik
Razvoj od 1. do 10. veka nove ere
Aryabhata - definisao pozicioni sistem
Brahmagupta - uveo nulu
Abu'l-Hasan al-Uqlidisi – uveo decimalnu tačku

*r* – radiks (radix) – osnova brojnog sistema – za nas od interesa kada je to celobrojna vrednost

$c_i$ – cifre brojnog sistema $c_i \in \{0,1,2,\ldots r-2, r-1\}$

*V* – vrednost koja se prikazuje (za sada samo pozitivne)

zapis $c_{n-1}c_{n-2}\,c_{n-3}\ldots\,c_1\,c_0.\,c_{-1}\,c_{-2}c_{-3}\ldots\,c_{-k+1}\,c_{-k}$

*n* celih mesta
*k* razlomljenih mesta
. tačka osnove

$$V = \sum_{i=-k}^{n-1} c_i r^i$$

Težinski – svaka cifra $c_i$ sa svojom pozicijom *i* nosi težinu $r^i$

$$c_{n-1}c_{n-2}\,c_{n-3}\dots\,c_1\,c_0.\,c_{-1}\,c_{-2}c_{-3}\dots\,c_{-k+1}\,c_{-k}$$

$$V=\sum_{i=-k}^{n-1}c_ir^i=\sum_{i=0}^{n-1}c_ir^i+\sum_{i=-k}^{-1}c_ir^i=CV+RV$$

$CV\ celobrojna\ vrednost\ \rightarrow c_{n-1}c_{n-2}\,c_{n-3}\dots\,c_1\,c_0$

$RV\ razlomljena\ vrednost \rightarrow 0.c_{-1}\,c_{-2}c_{-3}\dots\,c_{-k+1}\,c_{-k}$

0. - ima samo razlomljenu vrednost,
celobrojna vrednost jednaka 0

Celobrojna vrednost – dodatak VODEĆIH nula ne menja vrednost

$$c_{n-1}c_{n-2}\,c_{n-3}\dots\,c_1\,c_0 \leftrightarrow 00..00c_{n-1}c_{n-2}\,c_{n-3}\dots\,c_1\,c_0$$

Razlomljena vrednost – dodatak PRATEĆIH nula ne menja vrednost

$$0.\,c_{-1}\,c_{-2}c_{-3}\dots\,c_{-k+1}\,c_{-k} \leftrightarrow 0.\,c_{-1}\,c_{-2}c_{-3}\dots\,c_{-k+1}\,c_{-k}00..00$$



3

Bez obzira na osnovu pozicionog brojnog sistema uvek je $RV < 1$

$$RV=\sum_{i=-k}^{-1}c_ir^i$$

$$RV_{max}=\sum_{i=-k}^{-1}(c_i)_{max}r^i$$

$c_i \in \{0,1,2,\dots r-2, r-1\}$ $\qquad (c_i)_{max}=r-1$

$$RV_{max}=\sum_{i=-k}^{-1}(r-1)r^i=\sum_{i=-k}^{-1}\left(r^{i+1}-r^i\right)=(r^0-\cancel{r^{-1}})+(\cancel{r^{-1}}-\cancel{r^{-2}})+\cdots+\left(\cancel{r^{-k+2}}-\cancel{r^{-k+1}}\right)+\left(\cancel{r^{-k+1}}-r^{-k}\right)$$

$$RV_{max}=\left(1-r^{-k}\right)<1$$

Koliko će biti „blizu 1“ zavisi od broja upotrebljenih razlomljenih mesta za prikaz broja (normalno i od osnove ali razmišljamo o istoj osnovi)
NAPOMENA: Ovo važi za brojne sisteme sa celobrojnim osnovama



4

Prelazak iz jednog u drugi brojni sistem

Prelazak iz brojnog sistema sa osnovom *r* u decimalni brojni sistem

Ako je prikazan broj

$$c_{n-1}c_{n-2}\, c_{n-3}\dots\, c_1\, c_0.\, c_{-1}\, c_{-2}c_{-3}\dots\, c_{-k+1}\, c_{-k}$$

u brojnom sistemu sa osnovom *r*, što najčešće zapisujemo

$$(c_{n-1}c_{n-2}\, c_{n-3}\dots\, c_1\, c_0.\, c_{-1}\, c_{-2}c_{-3}\dots\, c_{-k+1}\, c_{-k})_r$$

Prelazak u decimalni brojni sistem (osnova 10 što je prirodno za nas) je direktno po izrazu

$$V = \sum_{i=-k}^{n-1} c_i r^i$$

Pri čemu se stepenovanje, množenje, sabiranje izvodi u decimalnom brojnom sistemu





PRIMER: $326.23_7 = ?_{10}$

$$326.23_7 = 3\cdot 7^2 + 2\cdot 7^1 + 6\cdot 7^0 + 2\cdot 7^{-1} + 3\cdot 7^{-2} = 167 + \frac{17}{49}$$

I verovatno bi smatrali da je ovo sasvim „korektan“ način prikazivanja. Tako su nas i terali da radimo.
Međutim razlomci „ne postoje“.
Želimo „standardan“ način prikazivanja.

$\frac{17}{49} = 0.346938775510204081632653061224 49\dots$

$326.23_7 = 167.34693877551020408163265306122449\dots_{10}$

Prikaz u jednom brojnom sistemu sa konačnim brojem cifara neće obavezno proizvesti prikaz u drugom brojnom sistemu takođe sa konačnim brojem cifara. Možda hoće, možda neće.

Prelazak iz decimalnog brojnog sistema u brojni sistem sa osnovom *r*

$$V_{10} = \sum_{i=-k}^{n-1} c_i r^i \qquad c_i \in \{0,1,2, \dots r-2, r-1\}$$

$$V_{10} = \sum_{i=0}^{n-1} c_i r^i + \sum_{i=-k}^{-1} c_i r^i = CV_{10} + RV_{10}$$

Celobrojna vrednost

$$CV_{10} = c_{n-1} \cdot r^{n-1} + c_{n-2} \cdot r^{n-2} + \ldots + c_1 \cdot r^1 + c_0 \cdot r^0$$

Razlomljena vrednost

$$RV_{10} = c_{-1} \cdot r^{-1} + c_{-2} \cdot r^{-2} + \ldots + c_{-k+1} \cdot r^{-k+1} + c_{-k} \cdot r^{-k}$$





Uočimo za celobrojnu vrednost

$$CV_{10} \div r = (c_{n-1} \cdot r^{n-1} + c_{n-2} \cdot r^{n-2} + \ldots + c_1 \cdot r^1 + c_0 \cdot r^0) \div r = c_{n-1} \cdot r^{n-2} + c_{n-2} \cdot r^{n-3} + \ldots + c_1 \cdot r^0 + \frac{c_0}{r}$$

$$\frac{c_0}{r} < 1 \qquad c_i \in \{0,1,2, \dots r-2, r-1\}$$

$$CV_{10} \div r = (CV_{10})_1 + \frac{c_0}{r} = (CV_{10})_1 \text{ i ostatak } c_0$$

Izdvojili smo cifru $c_0$ i nastavljamo dalje

$$(CV_{10})_1 \div r = (CV_{10})_2 + \frac{c_1}{r} = (CV_{10})_2 \text{ i ostatak } c_1$$

Izdvojili smo cifru $c_1$ i nastavljamo dalje …

Zaustavljamo se kada bude zadovoljen uslov

$$(CV_{10})_{n-1} < r \Rightarrow (CV_{10})_{n-1} = c_{n-1}$$

$(CV_{10})_n = 0$ Dobijali bi vodeće nule

# DA LI POSTOJI PROBLEM KONAČNOG BROJA CIFARA

Da nećemo stići do kraja po datom uslovu

$$(CV_{10})_{n-1} < r \Rightarrow (CV_{10})_{n-1} = c_{n-1}$$

Kod celobrojnih vrednosti pošto uzastopno delimo sa $r$ pitanje je da li uvek možemo naći prirodan broj $n$ takav da bude zadovoljen uslov

$$\frac{CV_{10}}{r^n} < 1 \Rightarrow CV_{10} < r^n \qquad n \geq \log_r CV_{10}$$



Uočimo razlomljenu vrednost

$$RV_{10} \cdot r = \left(c_{-1} \cdot r^{-1} + c_{-2} \cdot r^{-2} + \ldots + c_{-k+1} \cdot r^{-k+1} + c_{-k} \cdot r^{-k}\right) \cdot r$$
$$= c_{-1} \cdot r^{0} + c_{-2} \cdot r^{-1} + \ldots + c_{-k+1} \cdot r^{-k+2} + c_{-k} \cdot r^{-k+1}$$

$$c_{-1} \geq 0 \qquad c_i \in \{0,1,2, \ldots r-2, r-1\}$$

$$RV_{10} \cdot r = c_{-1} + (RV_{10})_{-2} = \text{celobrojna vrednost } c_{-1} + (RV_{10})_{-2}$$

Izdvojili smo cifru $c_{-1}$ i nastavljamo dalje

$$(RV_{10})_{-2} \cdot r = \text{celobrojna vrednost } c_{-2} + (RV_{10})_{-3}$$

Izdvojili smo cifru $c_{-2}$ i nastavljamo dalje …

Zaustavljamo se kada bude zadovoljen uslov

$$(RV_{10})_{-k-1} = 0 \Rightarrow c_{-k-1} = 0$$

Stalno bi dobijali prateće nule

# DA LI POSTOJI PROBLEM KONAČNOG BROJA CIFARA

Da nećemo stići do kraja po datom uslovu

$$(RV_{10})_{-k-1} = 0 \Rightarrow c_{-k-1} = 0$$





U bilo kojem brojnom sistemu razlomljeni deo može da se napiše kao

$$RV_r = c_{-1} \cdot r^{-1} + c_{-2} \cdot r^{-2} + \ldots + c_{-k+1} \cdot r^{-k+1} + c_{-k} \cdot r^{-k} = \frac{c_{-1} \cdot r^{k-1} + c_{-2} \cdot r^{k-2} + \ldots + c_{-k+1} \cdot r^{1} + c_{-k} \cdot r^{0}}{r^k}$$

Odnosno kao odgovarajuća celobrojna vrednost $RV_r = \dfrac{CV_r}{r^k}$

Ako je razlomljeni deo u decimalnom brojnom sistemu prikazan sa $l$ cifara i smatramo da će u drugom brojnom sistemu biti prikazan sa konačnim brojem cifara $k$ uslov se svodi na

$$\frac{(CV_{10})_l}{10^l} = \frac{(CV_r)_k}{r^k}$$

odnosno

$$(CV_r)_k = \frac{r^k}{10^l}(CV_{10})_l$$

Da li je uvek moguće zadovoljiti ovaj uslov. Odgovor je NE, kao što smo i videli a i videćemo.

Znači za razlomljeni deo uslov $(RV_{10})_{-k-1} = 0 \Rightarrow c_{-k-1} = 0$

A ako ga nije moguće postići: izračunat dovoljan broj cifara (šta god to za sada značilo)

PRIMER: $13.2_{10} = ?_2$

CELOBROJNA VREDNOST

| CV:2 | REZULTAT | OSTATAK |
|---|---|---|
| 13:2 | 6 | 1 |
| 6:2 | 3 | 0 |
| 3:2 | 1 | 1 |
| 1:2 | 0 | 1 |
| 0:2 | 0 | 0 |

1101

RAZLOMLJENA VREDNOST

| RV x 2 | REZULTAT | CEO DEO |
|---|---|---|
| 0.2 x 2 | 0.4 | 0 |
| 0.4 x 2 | 0.8 | 0 |
| 0.8 x 2 | 1.6 | 1 |
| 0.6 x 2 | 1.2 | 1 |
| 0.2 x 2 | 0.4 | 0 |
| 0.4 x 2 | 0.8 | 0 |
| 0.8 x 2 | 1.6 | 1 |
| 0.6 x 2 | 1.2 | 1 |
| 0.2 x 2 | 0.4 | 0 |
| … | … | … |

.001100110….

$$13.2_{10} = 1101.001100110\ldots_2$$

Što veći broj cifara u razlomljenom delu imaćemo tačnije „rezultat“ prelaska.
Apsolutno tačan nikada. Moramo se dogovoriti koji broj cifara uzimamo.
U ovom primeru. Za neke druge primere mogli bi da dobijemo i tačan rezultat prelaska. Na primer 0.5 ili 0.25 ili 0.75 ili ….



Primer prelaska $V_p = V_q$

Najbliže našem načinu razmišljanja $V_p \to V_{10} \to V_q$

Ne mora. Mi smo naviknuti od osnovnih škola na decimalni brojni sistem. Tablica sabiranja napamet. Tablica množenja napamet. I ti algoritmi računanja su nam postali urođeni, pa o njima i ne razmišljamo. Ali sada smo prinuđeni da bi „obučili“ naš digitalni sistem da se vratimo na ta računanja i ustanovimo algoritme.

$$236_7 = ?_2$$

Računanje radimo u brojnom sistemu sa osnovom 7

| | | | |
|---|---|---|---|
| prvi korak | | 2:2 = 1 ostatak 0 | $236_7 \div 2 = 1 \ldots$ |
| drugi korak | $236_7 \div 2 = ?$ | 03:2 = 1 ostatak 1 | $236_7 \div 2 = 11 \ldots$ |
| treći korak | $236_7 \div 2 = ?$ | 16:2 = 6 ostatak 1 | $236_7 \div 2 = 116$ |

$236_7 = ?_2$

| CV:2 | REZULTAT | OSTATAK |
|---|---|---|
| 236:2 | 116 | 1 |
| 116:2 | 43 | 0 |
| 43:2 | 21 | 1 |
| 21:2 | 10 | 1 |
| 10:2 | 3 | 1 |
| 3:2 | 1 | 1 |
| 1:2 | 0 | 1 |

$236_7 = 1111101_2$

1111101

$$236_7 = 2 \cdot 7^2 + 3 \cdot 7^1 + 6 \cdot 7^0 = 125_{10}$$

$$1111101_2 = 1 \cdot 2^6 + 1 \cdot 2^5 + 1 \cdot 2^4 + 1 \cdot 2^3 + 1 \cdot 2^2 + 0 \cdot 2^1 + 1 \cdot 2^0 = 125_{10} = 127_{10} - 2_{10}$$

Sedam cifara – maksimalna vrednost 127 – fali samo $0x2^1$





Kompatibilni brojni sistemi

Kod kompatibilnih brojnih sistema osnove su u odnosu $p = q^k$

To olakšava prelazak iz jednog sistema u drugi.
Transformiše se svaka cifra zasebno.

$$CV_p = c_{n-1} \cdot p^{n-1} + c_{n-2} \cdot p^{n-2} + \ldots + c_1 \cdot p^1 + c_0 \cdot p^0$$

$$CV_p = c_{n-1} \cdot q^{k(n-1)} + c_{n-2} \cdot q^{k(n-2)} + \ldots + c_1 \cdot q^{k \cdot 1} + c_0 \cdot q^{k \cdot 0}$$

$$c_i \in \{0,1,2, \ldots p-2, p-1\} = \{0,1,2, \ldots q^k - 2, q^k - 1\}$$

$$c_i = b_{k-1} \cdot q^{k-1} + b_{k-2} \cdot q^{k-2} + \cdots + b_1 \cdot q^1 + b_0 \cdot q^0$$

$$b_i \in \{0,1,2, \ldots q-2, q-1\}$$

$$CV_q = b_{(n-1)(k-1)} \cdot q^{kn-1} + b_{(n-1)(k-2)} \cdot q^{kn-2} + \ldots + b_{0,1} \cdot q^1 + b_{0,0} \cdot q^0$$

Važi i obrnuto. Kada se sa manje osnove ide na veću.
Grupišu se cifre manje osnove u grupe od *k* i ta grupa se prikaže cifrom veće osnove.

Kompatibilni brojni sistemi

Nama zanimljivi binarni, oktalni i heksadecimalni

binarni $r = 2$ $c_i \in \{0,1\}$

oktalni $r = 8 = 2^3$ $c_i \in \{0,1,2,3,4,5,6,7\}$

heksadecimalni $r = 16 = 2^4$ $c_i \in \{0,1,2,3,4,5,6,7,8,9,A,B,C,D,E,F\}$

Kompatibilni

binarni $\longleftrightarrow$ oktalni

binarni $\longleftrightarrow$ heksadecimalni



17

*k*=3

| Oktalna cifra | U binarnom |
|---|---|
| 0 | 000 |
| 1 | 001 |
| 2 | 010 |
| 3 | 011 |
| 4 | 100 |
| 5 | 101 |
| 6 | 110 |
| 7 | 111 |

*k*=4

| Heksadecimalna cifra | U binarnom |
|---|---|
| 0 | 0000 |
| 1 | 0001 |
| 2 | 0010 |
| 3 | 0011 |
| 4 | 0100 |
| 5 | 0101 |
| 6 | 0110 |
| 7 | 0111 |
| 8 | 1000 |
| 9 | 1001 |
| A | 1010 |
| B | 1011 |
| C | 1100 |
| D | 1101 |
| E | 1110 |
| F | 1111 |



18

PRIMER $1AC3_{16} =?_2$

Svaku cifru za sebe $1AC3_{16} = 0001\ 1010\ 1100\ 0011_2$

Pa možemo da izbrišemo vodeće nule $= 0001\ 1010\ 1100\ 0011_2 = 1101011000011_2$

PRIMER $10100010010101101_2 =?_{16}$

Da bi olakšali posao pravimo grupe od po četiri cifre počevši sa desne strane

$1\ 0100\ 0100\ 1010\ 1101_2 = 0001\ 0100\ 0100\ 1010\ 1101_2$

Dodali smo vodeće nule da bi dobili grupe od četiri cifre

$0001\ 0100\ 0100\ 1010\ 1101_2 = 144AD_{16}$

Potpuno identično binarni u oktalni i obrnuto samo su nam grupe sa po 3 cifre





Ako postoji razlomljeni deo broja

Sve isto osim što se grupisanje radi:
- za celobrojnu vrednost na levo od tačke
- za razlomljenu vrednost na desno od tačke

PRIMER $1A.3C_{16} =?_2$

Svaku cifru za sebe $1A.3C_{16} = 0001\ 1010\ .\ 0011\ 1100_2$

Pa možemo da izbrišemo vodeće i prateće nule $= 0001\ 1010\ .\ 0011\ 1100_2 = 11010.001111_2$

PRIMER $101100.1011011_2 =?_{16}$

Da bi olakšali posao pravimo grupe od po četiri cifre

$10\ 1100\ .\ 1011\ 011_2 = 0010\ 1100\ .\ 1011\ 0110_2$

Dodali smo vodeće i prateće nule da bi dobili grupe od četiri cifre

$0010\ 1100\ .\ 1011\ 0110_2 = 2C.B6_{16}$

Potpuno identično binarni u oktalni i obrnuto samo su nam grupe sa po 3 cifre

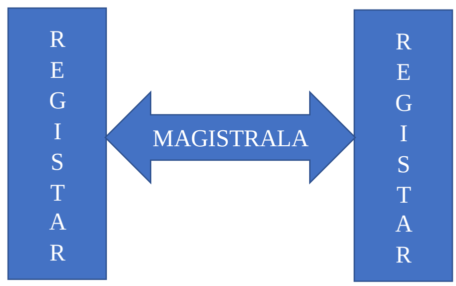
Prikaz brojeva u digitalnom sistemu

Konačna dužina prikaza – konačna dužina reči

Digitalni sistem radi sa apstraktnim signalima logičke nule i logičke jedinice

$$A = \{"0", "1"\} \qquad S \in A$$

Jako pojednostavljen model dela digitalnog sistema koji čuva i obrađuje informacije sadržane u signalima




Informacije se čuvaju u registrima

Informacije se razmenjuju putem magistrale



21

Informacije u registrima su prikazane logičkim nulama i logičkim jedinicama

Preko magistrale se prenose informacije u vidu logičkih nula i logičkih jedinica

RELAN FIZIČKI SVET

Broj informacija u registrima KONAČAN

| $I_{n-1}$ | $I_{n-2}$ | $I_{n-3}$ | . | . | . | . | $I_1$ | $I_0$ |
|---|---|---|---|---|---|---|---|---|
| *„0" ili „1"* | *„0" ili „1"* | *„0" ili „1"* | | | | | *„0" ili „1"* | *„0" ili „1"* |

Broj informacija koje je moguće preneti magistralom KONAČAN

*N* fizičkih linija veze

*N* konačan broj

*„0"*

*ili*

*„1"*



22

Jedna ćelija registra – BIT – najmanja količina informacija *b*

Grupa od četiri bita – NIBBLE – $b_3\ b_2\ b_1\ b_0$

Grupa od osam bita – BYTE – $b_7\ b_6\ b_5\ b_4\ b_3\ b_2\ b_1\ b_0$

Grupa od dva bajta, 16 bita - WORD

Grupa od dve reči, 32 bita – DOUBLE WORD

8 – bitni registar

| $D_7$ | $D_6$ | $D_5$ | $D_4$ | $D_3$ | $D_2$ | $D_1$ | $D_0$ |
|---|---|---|---|---|---|---|---|
| $b_7$ | $b_6$ | $b_5$ | $b_4$ | $b_3$ | $b_2$ | $b_1$ | $b_0$ |

ISTO VAŽI ZA MAGISTRALU

Na poziciji $D_i$ nalazi se informacija sadržana u bitu $b_i$

Bit $b_i$ nosi informaciju …

*$b_7$ – Most Significant Bit – MSB – bit najveće težine*

*$b_0$ – Least Significant Bit – LSB – bit najmanje težine*



Primer: Koje informacije se nalaze u registru

| $D_7$ | $D_6$ | $D_5$ | $D_4$ | $D_3$ | $D_2$ | $D_1$ | $D_0$ |
|---|---|---|---|---|---|---|---|
| „0“ | „1“ | „1“ | „0“ | „1“ | „0“ | „0“ | „0“ |

Odgovor: ???

Možda je logička jedinica na poziciji $D_6$ informacija da je napolju toplo. Ali sto tako možda je i informacija da treba uključiti svetlo. A možda je …

KO daje značenje tim bitima. PROJEKTANT sistema. On je projektovao sistem, odnosno gde se nalazi ta informacija, šta znači, kako je obrađena i smeštena na tu poziciju itd…

A možda je ta informacija, kompletna, i vrednost nekog broja

Ako naš digitalni sistem treba da obrađuje i čuva vrednosti u obliku brojeva onda je logično da „radi“ u binarnom brojnom sistemu.

Logična je veza
logička nula = binarna nula
logička jedinica = binarna jedinica

Iz tog razloga više nećemo stavljati apostrofe

| $D_7$ | $D_6$ | $D_5$ | $D_4$ | $D_3$ | $D_2$ | $D_1$ | $D_0$ |
|---|---|---|---|---|---|---|---|
| 0 | 1 | 1 | 0 | 1 | 0 | 0 | 0 |

Kao što smo rekli u fazi projektovanja smo dali značenje tim bitima – logička informacija ili cifra

Pa ako je to broj onda je njegova vrednost

$$01101000_2 = 68_{16} = 0x68 = x68 = 68h = 104_{10}$$

JESTE LI SIGURNI?





GDE JE TAČKA OSNOVE – U OVOM SLUČAJU BINARNA TAČKA

Možda je u registru celobrojna vrednost, možda samo razlomljena vrednost a možda i broj koji ima i celobrojnu i razlomljenu vrednost

| $D_7$ | $D_6$ | $D_5$ | $D_4$ | $D_3$ | $D_2$ | $D_1$ | $D_0$ |
|---|---|---|---|---|---|---|---|
| 0 | 1 | 1 | 0 | 1 | 0 | 0 | 0 |

Raspolažemo samo sa 0 i 1

Tačku ne možemo označiti sa 0 ili 1, nemoguće ju je razlikovati od cifara

POZICIJU BINARNE TAČKE JE PREDVIDEO PROJEKTANT
I TA INFORMACIJA MORA DA NAM BUDE POZNATA

Znači pored informacije da se u registru nalazi broj potrebna nam je i informacija gde se nalazi binarna tačka

Na primer
1. iza bita $D_5$ pa je binarna vrednost 011.01000
2. ili iza bita $D_0$ što je u stvari značenje da je u registru celobrojna vrednost
3. ili ispred bita $D_7$ što je u stvari značenje da u registru razlomljena vrednost
4. ili …

DIGRESIJA

| $D_7$ | $D_6$ | $D_5$ | $D_4$ | $D_3$ | $D_2$ | $D_1$ | $D_0$ |
|---|---|---|---|---|---|---|---|
| 0 | 1 | 1 | 0 | 1 | 0 | 0 | 0 |

Na primer
1. iza bita $D_5$ pa je binarna vrednost 011.01000
2. ili iza bita $D_0$ što je u stvari značenje da je u registru celobrojna vrednost
3. ili ispred bita $D_7$ što je u stvari značenje da u registru razlomljena vrednost
4. ili …

Uočite da na osnovu naših prethodnih zapažanja kako možemo posmatrati razlomljeni deo

$$RV_r = c_{-1} \cdot r^{-1} + c_{-2} \cdot r^{-2} + \ldots + c_{-k+1} \cdot r^{-k+1} + c_{-k} \cdot r^{-k} = \frac{c_{-1} \cdot r^{k-1} + c_{-2} \cdot r^{k-2} + \ldots + c_{-k+1} \cdot r^{1} + c_{-k} \cdot r^{0}}{r^k}$$

odnosno kao odgovarajuća celobrojna vrednost $RV_r = \dfrac{CV_r}{r^k}$

I u ovom slučaju broj u registru možemo posmatrati kao celobrojnu *CV* vrednost a kada znamo položaj binarne tačke onda je stvarna vrednost

1. iza bita $D_5$ pa je vrednost $\dfrac{CV}{2^5}$

2. ili iza bita $D_0$ pa je vrednost *CV*

3. ili ispred bita $D_7$ pa je vrednost $\dfrac{CV}{2^8}$



Da li smo sa ovim prethodnim jednoznačno odredili brojnu vrednost?

NEGATIVNE VREDNOSTI BROJEVA?

Do sada nam nisu bile od interesa. Naš zapis je *–V.* Prilikom prelaska iz brojnog sistema sa osnovom p u brojni sistem sa osnovom q nekako smo podrazumevalo da ćemo to raditi preko apsolutnih vrednosti i samo menjati znak

$$-V_P = -(V_P) = -(V_q) = -V_q$$

Na raspolaganju nam je bio simbol, znak, - i nismo uočili da bi tu mogao da se pojavi neki problem

Ovo važi i za binarni brojni sistem, ali ne i za onaj koji se koristi u digitalnim sistemima.
Nema posebnog logičkog nivoa za simbol -. Znači moraju da se koriste logičke nule i jedinice i neka definisanja od strane projektanta kao za binarnu tačku. Moramo da KODUJEMO negativne vrednosti brojeva.

Radi jednostavnosti posmatraćemo četvorobitni digitalni sistem. Dužina registara i širina magistrala je 4 bita. Broj mogućih kombinacija, kodova, sa 4 bita je 16. Logično, dobro, je da pola kodova bude rezervisano za pozitivne a pola za negativne brojeve. I za početak ćemo posmatrati samo celobrojne vrednosti.

| Kod $D_3D_2D_1D_0$ | Samo pozitivni | Znak i apsolutna vrednost |
|---|---|---|
| 0000 | 0 | 0 |
| 0001 | 1 | 1 |
| 0010 | 2 | 2 |
| 0011 | 3 | 3 |
| 0100 | 4 | 4 |
| 0101 | 5 | 5 |
| 0110 | 6 | 6 |
| 0111 | 7 | 7 |
| 1000 | 8 | -0 |
| 1001 | 9 | -1 |
| 1010 | 10 | -2 |
| 1011 | 11 | -3 |
| 1100 | 12 | -4 |
| 1101 | 13 | -5 |
| 1110 | 14 | -6 |
| 1111 | 15 | -7 |

Bit $D_3$ je bit znaka

$D_3$=0 – pozitivna vrednost

$D_3$=1 – negativna vrednost

Biti $D_2$ $D_1$ $D_0$ su cifre pozitivnog trobitnog binarnog broja. Apsolutna vrednost.



29

| Kod $D_3D_2D_1D_0$ | Samo pozitivni | Sa ofsetom -8 |
|---|---|---|
| 0000 | 0 | -8 |
| 0001 | 1 | -7 |
| 0010 | 2 | -6 |
| 0011 | 3 | -5 |
| 0100 | 4 | -4 |
| 0101 | 5 | -3 |
| 0110 | 6 | -2 |
| 0111 | 7 | -1 |
| 1000 | 8 | 0 |
| 1001 | 9 | 1 |
| 1010 | 10 | 2 |
| 1011 | 11 | 3 |
| 1100 | 12 | 4 |
| 1101 | 13 | 5 |
| 1110 | 14 | 6 |
| 1111 | 15 | 7 |

Biti $D_3$ $D_2$ $D_1$ $D_0$ su cifre pozitivnog četvorobitnog binarnog broja.

Prava vrednost se dobija kada se na tu vrednost doda ofset -8.



30

| Kod $D_3D_2D_1D_0$ | Samo pozitivni | Prvi komplement |
|---|---|---|
| 0000 | 0 | 0 |
| 0001 | 1 | 1 |
| 0010 | 2 | 2 |
| 0011 | 3 | 3 |
| 0100 | 4 | 4 |
| 0101 | 5 | 5 |
| 0110 | 6 | 6 |
| 0111 | 7 | 7 |
| 1000 | 8 | -7 |
| 1001 | 9 | -6 |
| 1010 | 10 | -5 |
| 1011 | 11 | -4 |
| 1100 | 12 | -3 |
| 1101 | 13 | -2 |
| 1110 | 14 | -1 |
| 1111 | 15 | -0 |

Pozitivni brojevi su sa bitom $D_3$=0.

Negativna vrednost se dobija tako što se od maksimalne vrednosti koda, oduzme odgovarajuća pozitivna vrednost.

$$-3 = 1111 - 0011 = 1100$$

Bit $D_3$ će biti bit znaka znaka
= 0 pozitivna vrednost
= 1 negativna vrednost





| Kod $D_3D_2D_1D_0$ | Samo pozitivni | Drugi komplement |
|---|---|---|
| 0000 | 0 | 0 |
| 0001 | 1 | 1 |
| 0010 | 2 | 2 |
| 0011 | 3 | 3 |
| 0100 | 4 | 4 |
| 0101 | 5 | 5 |
| 0110 | 6 | 6 |
| 0111 | 7 | 7 |
| 1000 | 8 | -8 |
| 1001 | 9 | -7 |
| 1010 | 10 | -6 |
| 1011 | 11 | -5 |
| 1100 | 12 | -4 |
| 1101 | 13 | -3 |
| 1110 | 14 | -2 |
| 1111 | 15 | -1 |

Pozitivni brojevi su sa bitom $D_3$=0.

Negativna vrednost se dobija tako što se od maksimalne vrednosti koda +1, oduzme odgovarajuća pozitivna vrednost.

$$-3 = 1111 + 1 - 0011 = 1100\text{+1=1101}$$

Bit $D_3$ će biti bit znaka znaka
= 0 pozitivna vrednost
= 1 negativna vrednost

## Komplement maksimalne vrednosti i komplement osnove

Ova dva pojma se uvode u nedostatku simbola, znaka, minus kako bi se prikazali i negativni brojevi, a koji će omogućiti da se na „lak“ način izvode aritmetičke operacije i sa pozitivnim i sa negativnim brojevima.

U opštem slučaju za brojni sistem sa osnovom *r* i prikazom sa dužinom reči *n* (broj cifara) komplement maksimalne vrednosti, odnosno prikaz negativnog broja je

$$-V \equiv (r^n - 1) - V$$

U binarnom brojnom sistemu – komplement jedinice ili prvi komplement

Zanimljiv pošto se dobija direktnim invertovanjem vrednosti bita

$$111 \ldots 11 - b_{n-1}b_{n-2}\, b_{n-3} \ldots\, b_1\, b_0 = \overline{b_{n-1}}\,\overline{b_{n-2}}\,\overline{b_{n-3}} \ldots \overline{b_1}\,\overline{b_0}$$

PRIMER: u decimalnom brojnom sistemu sa tri cifre broj različitih vrednosti je 1000. Ako vrednosti od 0 do 499 smatramo pozitivnim onda će negativna vrednost -123 biti prikazana u komplementu 9tke kao

$$-123 \equiv 999 - 123 = 876$$

Odnosno vodeće cifre 0,1,2,3,4 će biti za pozitivne brojeve a 5,6,7,8,9 za negativne brojeve



U opštem slučaju za brojni sistem sa osnovom *r* i prikazom sa dužinom reči *n* (broj cifara) komplement osnove, odnosno prikaz negativnog broja jec

$$-V \equiv r^n - V = (r^n - 1) - V + 1$$

U binarnom brojnom sistemu – komplement dvojke ili drugi komplement

Zanimljiv pošto se dobija direktnim invertovanjem vrednosti bita i dodatkom 1 a i videćemo da je „prirodan“ za aritmetičke operacije

$$111 \ldots 11 + 1\; - b_{n-1}b_{n-2}\, b_{n-3} \ldots\, b_1\, b_0 = \overline{b_{n-1}}\,\overline{b_{n-2}}\,\overline{b_{n-3}} \ldots \overline{b_1}\,\overline{b_0}{+}1$$

PRIMER: u decimalnom brojnom sistemu sa tri cifre broj različitih vrednosti je 1000. Ako vrednosti od 0 do 499 smatramo pozitivnim onda će negativna vrednost -123 biti prikazana u komplementu 9tke kao

$$-123 \equiv 1000 - 123 = 877$$

Odnosno vodeće cifre 0,1,2,3,4 će biti za pozitivne brojeve a 5,6,7,8,9 za negativne brojeve

Komplement maksimalne vrednosti – dva prikaza za nulu

Komplement osnove – nejednak broj pozitivnih i negativnih vrednosti izuzimajući nulu
Maksimalna apsolutna negativna vrednost veća od maksimalnog pozitivnog broja

Ako je broj negativan, njegova apsolutna vrednost se dobija istim komplementiranjem.

Znači moramo znati i na koji način su prikazane negativne vrednosti.

$$111\ldots11-\overline{b_{n-1}}\,\overline{b_{n-2}}\,\overline{b_{n-3}}\ldots\overline{b_1}\,\overline{b_0}=b_{n-1}\,b_{n-2}\,b_{n-3}\ldots b_1\,b_0$$

$$111\ldots11+1\ -\overline{b_{n-1}}\,\overline{b_{n-2}}\,\overline{b_{n-3}}\ldots\overline{b_1}\,\overline{b_0}+1=b_{n-1}b_{n-2}\,b_{n-3}\ldots b_1\,b_0$$



35

UOČITE

Za računanje negativnih vrednosti u drugom komplementu binarnog brojnog sistema

Znamo da je prikaz broja negativan

Pravu vrednost dobijamo komplementiranjem i stavljanjem znaka minus ispred

$$-(2^n-b_{n-1}b_{n-2}\,b_{n-3}\ldots b_1\,b_0)=-(2^n-b_{n-1}2^{n-1}-b_{n-2}\,b_{n-3}\ldots b_1\,b_0)=$$
$$=-\big((2-b_{n-1})2^{n-1}-b_{n-2}\ \,b_{n-3}\ldots b_1\,b_0\big)=-(2-b_{n-1})2^{n-1}+b_{n-2}b_{n-3}\ldots b_1\,b_0=$$
$$=(b_{n-1}-2)2^{n-1}+b_{n-2}b_{n-3}\ldots b_1\,b_0$$

$$b_{n-1}=1$$

$$-b_{n-1}\,2^{n-1}+b_{n-2}b_{n-3}\ldots b_1\,b_0$$

PRELAZAK

$$(-b_{n-1})\ b_{n-2}b_{n-3}\ldots b_1\,b_0$$



36

Uočite razlike
Drugi komplement u odnosu sa ofsetom

| | Drugi komplement | Sa ofsetom |
|---|---|---|
| -8 | 1000 | 0000 |
| -7 | 1001 | 0001 |
| -6 | 1010 | 0010 |
| -5 | 1011 | 0011 |
| -4 | 1100 | 0100 |
| -3 | 1101 | 0101 |
| -2 | 1110 | 0110 |
| -1 | 1111 | 0111 |
| 0 | 0000 | 1000 |
| 1 | 0001 | 1001 |
| 2 | 0010 | 1010 |
| 3 | 0011 | 1011 |
| 4 | 0100 | 1100 |
| 5 | 0101 | 1101 |
| 6 | 0110 | 1110 |
| 7 | 0111 | 1111 |



37

Ako je u sadržaju registra i razlomljena vrednost

Zbog negativnih brojeva i rezervisanja MSB za bit znaka,
maksimalna pozicija tačke je iza MSB

Prikaz negativnih brojeva, komplementiranje itd… radimo kao da su u pitanju celobrojne vrednosti

Zbog načina prikazivanja uvek važi $RV_r = \frac{CV_r}{r^k}$

PRIMER sa 4 bita i fiksnom tačkom iza MSB da prikažemo -0.75 u drugom komplementu

Pozitivan broj 0.75 je prikazan kao 0.110
U registru 0110 (6 podeljeno sa 8)

Negativna vrednost -6 u drugom komplementu je 1010
A ono što smo malopre videli

$$-1 \cdot 2^0 + 0 \cdot 2^{-1} + 1 \cdot 2^{-2} + 0 \cdot 2^{-3} = -0.75$$



38

SVE OVO DO SADA JE PRIKAZ BROJEVA SA FIKSNOM TAČKOM

POSTOJI PRIKAZ BROJEVA I SA POKRETNOM TAČKOM

FLOATING POINT NUMBER

Broj prikazan u formatu sa pokretnom tačkom ima dva registra ili jedna registar u kojem su dva dela
Mantisa i eksponent

$$V = mantisa \times osnova^{eksponent}$$

U binarnom brojnom sistemu
Mantisa - označeni binarni broj znak i apsolutna vrednost sa fiksnom tačkom iza MSB
Eksponent – označeni binarni broj, celobrojna vrednost, sa ofsetom koji je definisan standardom

Dozvoljava veće opsege i tačnost prikazivanja.