

2201: Architektúra počítačov

Prednášky: Chamraz, Ravas

Cvičenia: Balogh a kol.

Letný semester

Počet kreditov 6.00

Prednáška/Cvičenie: 3.00/2.00 [hod./tyždeň]

<http://www.kar.elf.stuba.sk/predmety/ap/>

Literatúra

1. Jelšina, M.: Architektúry počítačových systémov, Princípy. ELFA, Košice 2002 .
2. Clements, A: The Principles of Computer Hardware, Oxford.
3. Frištacký, N., Jelšina, M.: Číslicové počítače. ALFA Bratislava 1993.
4. Stalling, W.: Computer Organization and Architecture, principles.
5. Hlavička, J.: Architektura počítačů. ČVUT Praha 1998.
6. Dvořák, V., Drábek V.: Architektura procesoru. Vutium, Brno, 1999
7. Krajčovič, T.: Počítače. STU FEI Bratislava 2000.

Zobrazenie informácie v počítači

- **Bajtová organizácia informácií**
- **Číselné sústavy**
- **Reprezentácia údajov**
- **Kódovanie údajov**

Bajtová organizácia informácií

Diskrétna informácia (nespojitá, zakódovaná informácia,...),
uložená v pamäti počítača

Diskrétny objekt

- inštrukcie
- adresy
- údaje (dáta)

Kódovanie v dvojkovom (binárnom) kóde

bit – (**binary digit**) - najmenšia údajová jednotka, ktorá môže byť spracovávaná ČP (hodnoty 0 alebo 1)

K rôznych objektov $\Rightarrow$ **K** rôznych kódov vyjadriteľné pomocou usporiadanej m -tice binárnych symbolov

$$m = \log_2 K$$

Dvojkový **kód objektu**

$$b_1b_2\cdots b_m$$

Formát objektu definuje interpretáciu významu jednotlivých bitov b_i a veľkosť objektu udávanú počtom bitov

[1] str. 120, dole

Program obsahuje P - inštrukcií s dĺžkami $d_i \leq D$, kde D je maximálna dĺžka inštrukcie, pričom jej dĺžka závisí aj od veľkosti adresy, ktorá býva súčasťou inštrukcie

Dĺžka adresy $\Rightarrow$ dĺžka inštrukcie $\Rightarrow$ potrebná veľkosť pamäte pre program

Optimalizácia vedie k používaniu objektov konštantnej veľkosti strojového slova

Potrebná kapacita na uloženie programu závisí

na adresovaní objektov

od veľkosti (dĺžky) objektov

Efektívna kapacita pamäte

$$C_{ef} = C_{adr} + C_{data}$$

C_{data} – kapacita pam. priestoru pre m_d n-bitových operandov inštrukcií programu

C_{adr} – kapacita pam. priestoru pre m_a inštrukcií programu, ktoré obsahujú adresy operandov

[1] str. 122, obr.3.2 Vplyv dĺžky ...

krivka 1 – spotreba pamäte na uloženie adres (dĺžka adresy klesá s veľkosťou adresovateľného objektu)

krivka 2 – spotreba na uloženie objektov rastie s veľkosťou strojového slova – neobsadené všetky bity

bajt (byte) – skupina 8 bitov, súčasné mikroprocesory a minipočítače sú bajtovo orientované – dĺžka strojového slova je celočíselný násobok bajtu (8, 16, 32, 64 ...t.j. 1-,2-,4-,8-bajtové slovo)

Objekty nemajú rovnakú veľkosť, dva prístupy :

- maximálne kódovanie
- minimálne kódovanie

[1] str. 123, obr.3.3 Maximalne a ...

Maximálne kódovanie – (RISC)

- zvýšená spotreba pamäte
- dôraz na konštantný formát objektov (veľkosť sa prispôsobuje max. veľkosti) $\Rightarrow$ prispôsobenie šírky zberníc, minimalizácia čítacích a zapisovacích cyklov

Minimálne kódovanie – (CISC)

- zložitosť realizácie procesora (zložité dekodovanie inštrukcií)
- konštantný formát objektov rôznych veľkostí (šetrí pamäť)
- priemerne väčší počet čítacích/ zapisovacích cyklov (čítanie inštrukcie z pamäte, zápis/čítanie údajov)

Veľkosť strojového slova závisí od architektúry procesora, ktorá je ovplyvnená

- dĺžkou inštrukcie
- dĺžkou adresy
- orientáciou úloh (teda aj charakterom údajov), pre ktorý je určený

Cray-1 supercomputer	64 bits
ICL 1900 series mainframe	24 bits
UNIVAC 1100 mainframe	36 bits
PDP-11 minicomputer	16 bits
VAX minicomputer	32 bits
8085, Z80, 6802, 6809 microprocessors	8 bits
8086 microprocessor	16 bits
Third-generation Intel and Motorola microprocessors	32 bits
Fourth-generation microprocessors	64 bits
Special-purpose graphics processors	128 bits
Some microcontrollers	4 bits

Číselné sústavy

Zobrazenie čísla pomocou definovanej množiny symbolov-
číslic nazývame číselnou sústavou

Delíme ich na :

- pozičné číselné sústavy PČS
- nepozičné ČS

Pozičné číselné sústavy

- poloha číslice i (symbolu) v reťazci symbolov určuje jej význam
- poloha číslice v reťazci určuje rád číslice, ktorý je definovaný váhou v_i
- pri váhach $v_i = z^i$ nazývame z – základ číselnej sústavy, potom hovoríme o pozičných sústavách s prirodzeným sledovaním váh
- ak z je prirodzené číslo, potom číslice sú reprezentované prirodzenými číslami z intervalu $0, 1, \dots, z-1$

Abeceda symbolov (číslic) pozičnej sústavy zo základom z je definovaná množinou

$$A = \{ a_1, a_2, \dots, a_m \}$$

a označíme (a_j)

číselný ekvivalent (hodnotu) symbolu, pričom platí

$$(a_j) < z \quad \text{pre} \quad j = 1, 2, \dots, m \quad m = z \geq 2$$

potom zápis **racionálneho čísla** (píše sa aj reálneho) čísla v PČS zo základom z

$$N_z = \pm a_{n-1} a_{n-2} \cdots a_1 a_0, a_{-1} a_{-2} \cdots a_{-k} \quad \text{kde} \quad a_i \in A$$

Hodnotu čísla N_z vyjadríme v tvare polynómu

$$(N)_z = \sum_{i=-k}^{n-1} (a_i)z^i$$

Ak označíme **hodnotu číslice**

$$[a_i]_i = (a_i)z^i \quad \text{potom} \quad (N)_z = \sum_{i=-k}^{n-1} [a_i]_i$$

Napr. číslo $(2345,37)_{10}$ je číslo zobrazené v **pozičnej dekadickej sústave**

$$(2345,37)_{10} = 2 \cdot 10^3 + 3 \cdot 10^2 + 4 \cdot 10^1 + 5 \cdot 10^0 + 3 \cdot 10^{-1} + 7 \cdot 10^{-2}$$

v tomto prípade hodnoty číslic sú

$$[2]_3 = 2000 \quad [3]_2 = 300 \quad [4]_1 = 40 \quad [5]_0 = 5 \quad [3]_{-1} = 0,3 \quad [7]_{-2} = 0,07$$

Vlastnosti pozičnej číselnej sústavy

$$N_z = \pm a_{n-1} a_{n-2} \cdots a_1 a_0, a_{-1} a_{-2} \cdots a_{-k}$$

1. Maximálne zobraziteľné číslo

$$\begin{array}{ll} N_{\max} = z^n - z^{-k} & \text{pre celé čísla} \\ N_{\max} = z^n - 1 & \\ & \text{pre desatinné} \\ N_{\max} = 1 - z^{-k} & \end{array}$$

2. Minimálne číslo v absolútnej hodnote rôzne od nuly

$$N_{\min} = z^{-k}$$

3. Krok diskretnosti $h = z^{-k}$

4. Kapacita číselnej sústavy pre m-rádové čísla

$$K = z^m = z^{(n+k)}$$

5. Počet zobrazujúcich rádov $m = \log_z(K + 1)$

Posunuté číselné sústavy

Číselné ekvivalenty číselnej sústavy so základom $z > 1$

- použitím kladných číslíc

$$(a_j) = j \quad \text{pre} \quad j = 0, 1, 2, \dots, z - 1$$

- použitím záporných číslíc

$$(a_j) = -j \quad \text{pre} \quad j = 0, 1, 2, \dots, z - 1$$

Ak číselná sústava obsahuje 0, potom hovoríme o **posunutých číselných sústavách**. V nich môžeme zobrazit' ľubovoľné kladné, alebo záporné číslo.

Symetrické číselné systavy

Symetrická číselná sústava, ak základ $z = 2c + 1$

a množina cifier $A = \{-c, -c + 1, \dots, -1, 0, 1, \dots, c - 1, c\}$

Ku všetkým číslam posunutej číselnej sústavy existuje ekvivalent kombinácie čísl symetrickej číselnej sústavy napr.

$$c + 1 = 1.z^1 + (-c).z^0 = 2c + 1 - c$$

$$c + 2 = 1.z^1 + (-c + 1).z^0 = 2c + 1 - c + 1$$

Príklady :

V posunutej číselnej sústave pri $z = 5$ s ciframi $\{0, 1, 2, 3, 4\}$, je

$$(3) = 3$$

V symetrickej číselnej sústave pri $z = 5$ s ciframi $\{-2, -1, 0, 1, 2\}$, je

$$(1 \bar{2}) = 3 \quad \text{ak } \bar{2} = -2$$

$$1.z^1 + (-2).z^0 = 5 - 2 = 3$$

Ak základ číselnej sústavy je párne číslo $z = 2c$, nie je jednoznačné priradenie číselným ekvivalentom priradiť číslice. Samotné číslo c nie je jednoznačne definované

$$c = c.z^0 \quad \text{alebo} \quad c = 1.z^1 + (-c).z^0 = 2c - c$$

t.j.

$$c = (c) \quad \text{alebo}$$

$$c = (1 \bar{c})$$

Príklad :

V desiatkovej symetrickej sústave ($z=10$ a s množinou znakov $\{-5,-4,-3,-2,-1, 0, 1, 2,3,4,5\}$ je číslo 5 zobrazené

$$5 = (5)_{10S} = 5 \cdot 10^0$$

$$5 = (1 \bar{5})_{10S} = 1 \cdot 10^1 - 5 \cdot 10^0$$

Symetrická číselná sústava pri párnom základe je redundantná (nadbytočná), redundancia sa dá odstrániť.

Pozičné číselné sústavy – prevody medzi číselnými sústavami

x_{10}	x_2	x_3	x_8	$x_{16} \equiv (x)_H$
0	0000	000	00	0
1	0001	001	01	1
2	0010	002	02	2
3	0011	010	03	3
4	0100	011	04	4
5	0101	012	05	5
6	0110	020	06	6
7	0111	021	07	7
8	1000	022	10	8
9	1001	100	11	9
10	1010	101	12	A
11	1011	102	13	B
12	1100	110	14	C
13	1101	111	15	D
14	1110	112	16	E
15	1111	120	17	F

osmičková a hexadecimálna (šestnásťková) sústava sa používa na sprehl'adnenie zápisu binárneho obsahu pamäťového miesta, konštanty, výsledku operácie, napr. pri ladení progrogramov. Podporujú ju aj nižšie a vyššie programovacie jazyky :

Príklady :

Jazyk C	Assembler	Význam
0xAA	0AAH	(AA)₁₆
077	77Q	(77)₈

Prevod z desiatovej sústavy do číselnej sústavy so základom z

Prevod sa vykonáva zvlášť pre

- celočíselnú časť čísla a
- desatinnú časť čísla

Prevod celočíselného dekadického čísla do sústavy so základom z

Metóda je založená na postupnom celočíselnom delení dekadického N číslom z. Celočíselné delenie

$$\frac{N}{z} = M + R$$

kde M- delenec, z- deliteľ, M- podiel, R- zvyšok sú celé čísla

$$(N)_Z = a_{n-1}z^{n-1} + a_{n-2}z^{n-2} + \dots + a_1z + a_0$$

$$(N_1)_Z = \frac{(N)_Z}{z} = a_{n-1}z^{n-2} + a_{n-2}z^{n-3} + \dots + a_1 \quad ((N)_Z \% z) = a_0 \quad (N)_Z \pmod{z} = a_0$$

$$(N_2)_Z = \frac{(N_1)_Z}{z} = a_{n-1}z^{n-3} + a_{n-2}z^{n-4} + \dots + a_2 \quad ((N_1)_Z \% z) = a_1$$

Príklady :

$$4629 \div 8 = 578 \quad R = 5 \quad 53241 \div 16 = 3327 \quad R = 9$$

$$578 \div 8 = 72 \quad R = 2 \quad 3328 \div 16 = 207 \quad R = 15_{10} = F_{16}$$

$$72 \div 8 = 9 \quad R = 0 \quad 207 \div 16 = 12 \quad R = 15_{10} = F_{16}$$

$$9 \div 8 = 1 \quad R = 1 \quad 12 \div 16 = 0 \quad R = 12_{10} = C_{16}$$

$$1 \div 8 = 0 \quad R = 1$$

$$4629_{10} = 11025_8$$

$$53241_{10} = CFF9_{16}$$

Prevod do binárnej sústavy

$$123 \div 2 = 61 \quad R = 1 \quad \textit{LSB(Least - significant bit)}$$

$$61 \div 2 = 30 \quad R = 1$$

$$30 \div 2 = 15 \quad R = 0$$

$$15 \div 2 = 7 \quad R = 1$$

$$7 \div 2 = 3 \quad R = 1$$

$$3 \div 2 = 1 \quad R = 1 \quad \uparrow$$

$$1 \div 2 = 0 \quad R = 1 \quad \textit{MSB(Most - significant bit)}$$

$$123_{10} = 1111011_2$$

$$123 \quad 1 \quad \textit{LSB}$$

$$61 \quad 1$$

$$30 \quad 0$$

$$15 \quad 1$$

$$7 \quad 1$$

$$3 \quad 1 \quad \uparrow$$

$$1 \quad 1 \quad \textit{MSB}$$

$$0$$

Prevod z číselnej sústavy so základom z do desiatkovej sústavy

Vychádza zo vzťahu pre hodnotu čísla vyjadreného v danom základe číselnej sústavy (zápis hodnoty je formálne zhodný so zápisom čísla v dekadickéj sústave)

$$(N)_z = a_{n-1}z^{n-1} + a_{n-2}z^{n-2} + \dots + a_1z + a_0$$

alebo

$$(N)_z = (\dots(((a_{n-1}z + a_{n-2})z + a_{n-3})z + \dots)z + a_1)z + a_0$$

Príklad : Preved'me binárne číslo 1010111 do dekadickéj sústavy

Prvý spôsob :

$$\begin{aligned}(1010111)_2 &= 1*2^6 + 0*2^5 + 1*2^4 + 0*2^3 + 1*2^2 + 1*2^1 + 1*2^0 = \\ &= 1*64 + 0*32 + 1*16 + 0*8 + 1*4 + 1*2 + 1*1 = 87 = 87_{10}\end{aligned}$$

Druhý spôsob :

Diagram illustrating a sequence of fractions and their relationships:

- Top row of bits: 1, 0, 1, 0, 1, 1, 1
- Fractions shown: $\frac{2}{2}$, $\frac{4}{5}$, $\frac{10}{10}$, $\frac{20}{21}$, $\frac{42}{43}$, $\frac{86}{87}$
- Red arrows indicate a path from the first '1' to $\frac{2}{2}$, and from $\frac{2}{2}$ to $\frac{4}{5}$.
- A green arrow points from $\frac{10}{10}$ to $\frac{20}{21}$.
- A cyan '+' sign is next to $\frac{20}{21}$.

$$1010111_2 = 87_{10}$$

Prevod $(6437)_8$
do dekadické sústavy :

$$\begin{array}{r}
 6 \qquad 4 \qquad 3 \qquad 7 \\
 48 \\
 \hline
 52 \\
 \\
 416 \\
 \hline
 419 \\
 \\
 3352 \\
 \hline
 3359 \\
 \\
 6437_8 = 3359_{10}
 \end{array}$$

Prevod z číselnej sústavy so základom z do číselnej sústavy so základom w

Pri prevode zo sústavy so základom z do číselnej sústavy so základom w sa všeobecne používa schéma

$$(N)_z = (N_1)_{10} = (N_2)_w$$

Výnimkou sú prevody medzi sústavami pri základe $z = 2^n$

Prevod z binárnej sústavy do októlovej alebo hexadecimálnej sa vykoná tak, že sa binárne znaky rozdelia „odzadu“ na trojíc alebo štvoríc, a skupiny sa kódujú osobitne

Majme binárne číslo 01101000111_2

rozdelíme na trojice

skupiny binárnych číslíc $01 \mid 101 \mid 000 \mid 111$

kódy oktálových číslíc $1 \quad 5 \quad 0 \quad 7$

$$(01101000111)_2 = (1507)_8$$

rozdelíme na štvorice $011 \mid 0100 \mid 0111$

kódy hexadecimálnych číslíc $3 \quad 4 \quad 7$

$$(01101000111)_2 = (347)_{16}$$

Majme $(775)_8$

kódy oktálových číslíc $7 \quad 7 \quad 5$

skupiny binárnych číslíc $111 \quad 111 \quad 101$

$$(775)_8 = (111111101)_2$$

majme $(775)_{16}$

kódy hexadecimálnych čísl	7	7	5
skupiny binárnych čísl	0111	0111	0101

$(775)_{16} = (011101110101)_2$

V tomto prípade je mechanizmus prevodu založený na prevode cez binárnu sústavu.

Prevod desatinnej časti dekadického čísla do sústavy so základom z

Metóda je založená na postupnom násobení desatinnej časti dekadického N číslom z

$$D.z = M + D_1$$

$$kde \quad |D| < 1 \quad |D_1| < 1$$

a M je celé číslo

$$(N)_z = a_{-1}z^{-1} + a_{-2}z^{-2} + \dots + a_{-k}z^{-k} \quad |(N)_z| < 1$$

$$(N)_z.z = a_{-1} + (N_1)_z \quad a_{-1} - \text{cele cislo} \quad |(N_1)_z| < 1$$

$$(N_1)_z.z = a_{-2} + (N_2)_z \quad a_{-2} - \text{cele cislo} \quad |(N_2)_z| < 1$$

Príklady : Prevedieme číslo $0,12_{10}$ do osmičkovej sústavy,
číslo $0,6875_{10}$ a $0,1_{10}$ do dvojkovej sústavy

$$0,12 * 8 = \mathbf{0}, 96$$

$$0,96 * 8 = \mathbf{7}, 68$$

$$0,68 * 8 = \mathbf{5}, 44$$

$$0,44 * 8 = \mathbf{3}, 52$$

$$0,52 * 8 = \mathbf{4}, 16$$

$$0,16 * 8 = \mathbf{1}, 28$$

.....

$$0,12_{10} \approx 0,075341_8$$

$$0,6875 * 2 = \mathbf{1}, 3750$$

$$0,3750 * 2 = \mathbf{0}, 7500$$

$$0,7500 * 2 = \mathbf{1}, 5000$$

$$0,5000 * 2 = \mathbf{1}, 0000$$

$$0,0000 * 2 = \mathbf{0}, 0000 \quad \textit{koniec}$$

$$0,6875_{10} = 0,1011_2$$

$0,1000 * 2 = \mathbf{0},$ 2000

$0,2000 * 2 = \mathbf{0},$ 4000

$0,4000 * 2 = \mathbf{0},$ 8000

$0,8000 * 2 = \mathbf{1},$ 6000

$0,6000 * 2 = \mathbf{1},$ 2000

$0,2000 * 2 = \mathbf{0},$ 4000

$0,4000 * 2 = \mathbf{0},$ 8000

$0,8000 * 2 = \mathbf{1},$ 6000

$0,1_{10} = 0,000110011\dots_2 = 0,0001\overline{1}_2$

Dôsledky :

Napr. kód v jazyku C

```
double x;  
for (x=0; x!=1; x+=0.1) {  
    }
```

je nekonečným cyklom !!

Preto správne by malo byť :

```
double x;  
for (x=0; x<=1; x+=0.1) {  
    }
```

Číslo $0,1_{10}$ sa nedá vyjadriť konečným počtom binárnych čísl !!

Nepozičné číselné sústavy

V nepozičných číselných sústavách neplatí vždy

$$[a_i]_i = (a_i)z^i$$

Napr.: Číslo $(N)_{10} = CCXV$ zobrazené v rímskej číselnej sústave vyjadruje nepozičný zápis čísla 215. V tomto prípade

$$[(C)]_3 = 100 \quad [(C)]_2 = 100 \quad [(X)]_1 = 10 \quad [(V)]_0 = 5$$

Číselná sústava zvyškových tried

Čísla **A**, **B** patria do tej istej zvyškovej triedy ak platí pre základ **z** platí

$$A(\text{mod } z) = B(\text{mod } z)$$

Číselná sústava zvyškových tried je charakterizovaná viacerými základmi

$$\{p_0, p_1, \dots, p_n\}$$

$$P = p_0 \cdot p_1 \cdot \dots \cdot p_n$$

kde p_i sú nesúdeliteľné kladné celé čísla, P je perióda zvyškových tried.

Zobrazenie dekadického čísla v tejto číselnej sústave

$$(N)_{10} = (v_0 v_1 v_2 \dots v_n)_P \quad \text{kde} \quad v_i = N \pmod{p_i}$$

Príklad :

$$P = p_0 \cdot p_1 \cdot p_2 = 30 \quad (p_0 = 2, p_1 = 3, p_2 = 5)$$

$$(23)_{10} = (23 \pmod{2} \ 23 \pmod{3} \ 23 \pmod{5})_P = (123)_P$$

Výhodou je jednoduchá realizácia aritmetických operácií :

$$\begin{array}{l} (17)_{10} \rightarrow (123)_P; \quad P = 2.5.7 \\ + (19)_{10} \rightarrow (145)_P \end{array}$$

$$(36)_{10} \rightarrow 268$$

$8 : 7 = 1 ; \ v_2 = 1$
 $6 : 5 = 1 ; \ v_1 = 1$
 $2 : 2 = 1 ; \ v_0 = 0$

$(0 \ 1 \ 1)_P = (36)_{10}$

$$(5)_{10} \rightarrow (1\ 0\ 5)_P; \quad P = 2.5.7$$

$$\times (4)_{10} \rightarrow (0\ 4\ 4)_P$$

$$(20)_{10} \rightarrow 0\ 0\ 20$$

$20 : 7 = 2; \quad v_2 = 6$
 $0 : 5 = 0; \quad v_1 = 0$
 $0 : 2 = 0; \quad v_0 = 0$

$(0\ 0\ 6)_P = (20)_{10}$

Vlastnosti číselných sústav zvyškových tried :

2. Počet rádiv čísla v danej sústave je rovnaký, nezávisí od hodnoty čísla
3. Operácie súčtu/rozdielu a násobenia/delenia majú formálne rovnaký algoritmus (vykonávanie operácií trvá približne rovnaký čas)
4. Čísla sa nedajú porovnávať (sústava nemá metriku)
5. Zložitý algoritmus spätného prevodu do dekadickéj sústavy.