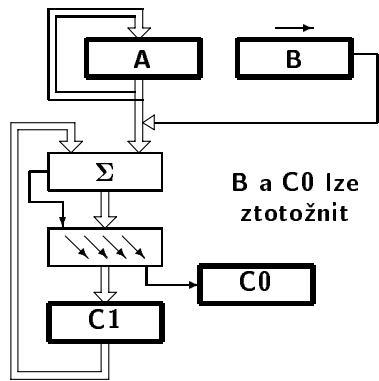


Násobení

Př.:
$$\begin{array}{r} 111 \\ \times 101 \\ \hline 000 \\ 111 \\ 0111 \\ \hline 000 \\ \hline 0011 \\ \hline 111 \\ \hline 1000 \end{array}$$



Dělení

$$\begin{array}{r} 0,101 : 0,110 \\ \downarrow \\ 101 : 110 = 0,110 \quad \text{podíl} \\ - \underline{110} \\ -1 \quad \rightarrow 0, \quad \text{návrat} \\ + \underline{110} \\ 1010 \\ - \underline{110} \\ 1000 \quad \rightarrow 1 \\ - \underline{110} \\ 100 \quad \rightarrow 1 \\ - \underline{110} \\ -10 \quad \rightarrow 0 \\ + \underline{110} \\ 100 \quad \text{návrat} \\ \text{zbytek} \end{array}$$

návrat (přes nulu) — restaurace nezáporného zbytku: pomocná operace obnovující původní dílčí zbytek pro následující odčítání

Dělení bez návratu přes nulu (bez restaurace nezáporného zbytku)

návrat = přičtení jistého čísla $X \rightarrow + X$
následuje odečtení čísla $X \gg 1 \rightarrow -X/2$
 $+X/2$

$$\begin{array}{r} 101 : 110 = 0,110 \quad \text{podíl} \\ - \underline{110} \\ -10 \quad \rightarrow 0, \\ + \underline{110} \\ 1000 \quad \rightarrow 1 \\ - \underline{110} \\ 100 \quad \rightarrow 1 \\ - \underline{110} \\ -10 \quad \rightarrow 0 \\ + \underline{110} \\ 100 \quad \text{návrat} \\ \text{zbytek} \end{array}$$

návrat — pouze v posledním kroku, je-li zbytek záporný, a to jenom v případě, že třeba získat také správný zbytek.

Desítkové kódy

k bitů/číslíce ... k bitový kód

$$2^k \geq 10 \implies k \geq 4$$

10 bitový kód: 1 z 10

5 bitové kódy: bezpečnostní kódy (2 z 5, 3 z 5 a zejm. Brownův kód $3N+2$)

4 bitové kódy: $\binom{16}{10} \cdot 10! = 29\,059\,430\,400$ kódů

	BCD	+3	2 4 2 1	8,4,-2,-1
0	0000	0011	0000	0000
1	0001	0100	0001	0111
2	0010	0101	0010	0110
3	0011	0110	0011	0101
4	0100	0111	0100	0100
5	0101	1000	1011	1011
6	0110	1001	1100	1010
7	0111	1010	1101	1001
8	1000	1011	1110	1000
9	1001	1100	1111	1111

- váhové kódy, např. BCD (8,4,2,1); 2,4,2,1; aj.
- Stibitzův (+3, též XS3), Aikenův (2,4,2,1) a Rubinoffův kód (8,4,-2,-1) — doplněk do 9 se získá inverzí všech bitů (sčítačka \rightarrow odčítačka)

Kód BCD — sčítání

Jednomístná desítková sčítačka

a, b číslice sčítanců p přenos z nižšího řádu

s číslice součtu q přenos do vyššího řádu

Označme $y = a + b + p$; pak $y \geq 10 \iff q = 1$

? $y \rightarrow s$?

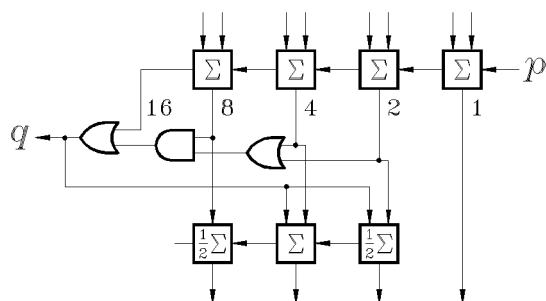
$$\text{korekce} = \begin{cases} -10 & \text{pro } q = 1 \\ 0 & \text{pro } q = 0 \end{cases}$$

$$-10 = -16 + 6$$

Př.: $3+2+0 \rightarrow 05 \dots 00101$

$5+6+1 \rightarrow 12 \dots 01100 + 0110 \dots 10010$

$8+9+0 \rightarrow 17 \dots 10001 + 0110 \dots 10111$



Kód BCD — sčítání — 80x86

ADD/ADC: AF [Auxiliary Flag] přenos z řádu 3

CF [Carry Flag] přenos z řádu 7

Po ADD AL, ... nebo po ADC AL, ... lze použít

DAA: korekce +0 nebo +6 (obě číslice)

vytvoření CF (desítkový přenos)

Př.: $\begin{matrix} 8\text{b.} & 8\text{b.} & & 8\text{b.} & 8\text{b.} & & 8\text{b.} & 8\text{b.} & 8\text{b.} \\ \boxed{x1} & \boxed{x0} & + & \boxed{y1} & \boxed{y0} & \rightarrow & \boxed{s2} & \boxed{s1} & \boxed{s0} \\ 49 & 22 & + & 67 & 18 & = & 01 & 16 & 40 \end{matrix}$

```
mov AL,x0 ; 22h → AL
ADD AL,y0 ; 22h+18h = 3Ah → AL
DAA ; korekce AL: +06h (040)
mov s0,AL ; 40h → s0

mov AL,x1 ; 49h → AL
ADC AL,y1 ; 49h+67h+0 = B0h → AL
DAA ; korekce AL: +66h (116h)
mov s1,AL ; 16h → s1

mov AL,0 ; 0 → AL
ADC AL,0 ; 0+0+1 = 01h → AL
mov s2,AL ; 01h → s2
```

Pohyblivá řádová čárka

Dosud uvažovaná reprezentace čísel:

Pevná řádová čárka [fix point]

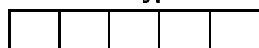
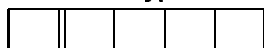
Nejobvyklejší typy řádových mřížek:

1. typ: modul $M = 2 \dots$ abs. hodnota ≤ 1

2. typ: celá čísla

1. typ

2. typ



Př.: 1. typ podle obrázku
přímý kód $\Rightarrow -1 < A < 1$
 $C = A + B = 1 + 2 = ?$
 A ani B není zobrazitelné !?!

měřítka $S = 1:16 = 1:10000_2$ [Scale]

$A^* = A \cdot S = 0,0001_2$

$B^* = B \cdot S = 0,0010_2$

$C^* = C \cdot S = A^* + B^* = 0,0011_2$

$C = C^*/S = 11_2 = 3$

Pohyblivá řádová čárka

Měřítka:

$$\begin{aligned} A + B &= (A^* + B^*) / S \\ A - B &= (A^* - B^*) / S \\ A * B &= (A^* * B^*) / S^2 \\ A / B &= (A^* / B^*) \end{aligned}$$

Vhodné měřítko (!?!)

Lze bez větších problémů předem stanovit pro výpočty z oblasti tzv. zpracování hromadných dat (účetnictví, sklady apod.) — variabilita potřebných měřítek je „minimální“. Vhodné měřítko však prakticky nelze předem stanovit pro tzv. vědecko-technické výpočty — opačná situace.

Řešení ???



Ať se snaží počítač — má na to HW i SW !!!



reprezentace čísel v pohyblivé řádové čárce

Pohyblivá řádová čárka [floating point]

Obrazem čísla A je taková uspořádaná dvojice

$$(m, e)$$

že platí $A = m \cdot z^e$, kde z je základ soustavy (dále budeme uvažovat $z = 2$); tedy:

$$A = m \cdot 2^e$$

m mantisa ... 1. typ řádové mřížky (velmi často)

e exponent ... 2. typ řádové mřížky (vždy)
(též charakteristika)

Příklad: mantisa i exponent v doplňkovém kódu

$$A = -13 = -1101_2 = (-0,1101 \cdot 10^{100})_2$$

$$D(m) = D(-0,1101) = 1,00110 \dots 0$$

$$D'(e) = D'(100) = 0100$$

0	1	0	0	1	0	0	1	1	0	0	0	0	0	0	0	4980 ₁₆
exponent				mantisa												

Pohyblivá řádová čárka

Operace

zjednodušeno

sčítání/odčítání

1. upravit operandy tak, aby měly stejný exponent

2. sečíst/odečíst mantisy

$$\begin{aligned} \text{srov.: } 500 - 3 &= 0,5 \cdot 10^3 - 0,3 \cdot 10^1 = \\ &= 0,5 \cdot 10^3 - 0,003 \cdot 10^3 = \\ &= 0,497 \cdot 10^3 = \\ &= 497 \end{aligned}$$

násobení/dělení

• vynásobit/podělit mantisy

• sečíst/odečíst exponenty

$$\begin{aligned} \text{srov.: } 500 \times 3 &= 0,5 \cdot 10^3 \times 0,3 \cdot 10^1 = \\ &= 0,15 \cdot 10^4 = 1500 \end{aligned}$$

výsledný exponent:

> největší možný \Rightarrow přeplnění [overflow]

< nejmenší možný \Rightarrow nenaplnění [underflow]

$$\text{srov.: } 0,8 \cdot 10^{-99} = 0,000 \dots 008 \rightarrow 0$$

Pohyblivá řádová čárka

Normalizace

Témuž číslu může příslušet několik reprezentací.

$$\text{Srov.: } 8 = 0,8 \cdot 10^1 = 0,08 \cdot 10^2 = \dots$$

Číslo má normalizovaný tvar, není-li možné posunout mantisu doleva.

Příklad: mantisa i exponent v doplňkovém kódu

$$\begin{aligned} A = -13 = -1101_2 &= (-0,1101 \cdot 10^{100})_2 = \\ &= (-0,01101 \cdot 10^{101})_2 \end{aligned}$$

$D(m)$	$D'(e)$	
1,001 100...0	100	normalizovaný tvar
1,100 110...0	101	nenormalizovaný tvar

normalizovaný tvar \Rightarrow jednodušší algoritmy

Na konci každé operace je třeba provést tzv. normalizaci výsledku !!!

Pohyblivá řádová čárka

Skrytá jednička

Nenulová mantisa: • přímý kód
• normalizovaný tvar



Bit v nejvyšším řádu je vždy roven 1.

Tento bit lze ze zápisu nenulových čísel vypustit.



skrytá jednička [hidden one]

Je použita např. v reprezentaci podle
ANSI/IEEE Std 754 - 1985

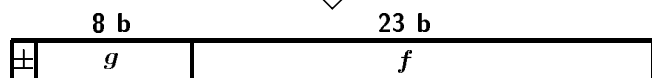
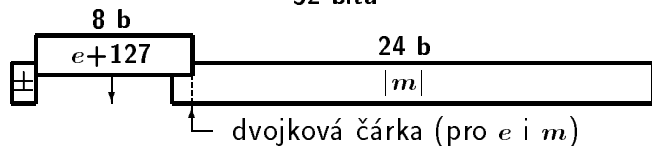
základní formáty:

	znaménko	exponent	mantisa
32 b	1 b	8 b	23 (24) b
64 b	1 b	11 b	52 (53) b

V obou formátech je pro mantisu (včetně znaménka) použit modul $M = 4$, tzn. $|m| < 2$.

ANSI/IEEE Std 754 – 1985

32 bitů



Př.: $-58_{10} = -11\,1010_2 = (-1,11010 \times 10^{101})_2$

$|m| = 1,11010 \Rightarrow f = 110100 \dots 0$

$e = 101 \Rightarrow g = 1000\,0100$

1 1000 0100 1101 0000 0 ... 00

| C | 2 | 6 | 8 | 0 0 0 0 |

little endian: 00 00 68 C2

konvence: 00000000 je obrazem nuly !!!

(neúplně) $g = 255$ a $f = 0$: ∞

$g = 255$ a $f \neq 0$: NaN

NaN [Not a Number] — např. výsledek dělení 0:0

Alfanumerické kódy (Abecedně-číslíkové kódy)

kódy pro zobrazení textů

tj. písmen, číslic (cifer) a dalších znaků

26 písmen anglické (latinské) abecedy

× 2 (velká/malá)

10 číslic

další znaky (mezera, čárka, tečka, ..., plus, ...)

↓

minimálně 6 bitů/znak (raději aspoň 7 bitů/znak)

5bitový kód: CCITT 2 (ITA 2, MTA 2)

dvojí interpretace znaků „přepínaná“ speciálními znaky

8bitový kód: EBCDIC (DKOI)

[Extended Binary-Coded-Decimal Interchange Code
(dvojičnýj kod dlja obměny i obrabotky informacij)]

mezera 40

'a' ... 'i' 81 ... 89 'A' ... 'I' C1 ... C9

'j' ... 'r' 91 ... 99 'J' ... 'R' D1 ... D9

's' ... 'z' A2 ... A9 'S' ... 'Z' E2 ... E9

'0' ... '9' F0 ... F9

Alfanumerické kódy - II.

**7bitový kód: ASCII, popř. USASCII
(CCITT 5, ISO-7, KOI-7)**

[American Standard Code for Information Interchange,
popř. USA Standard Code ...

(Comité consultatif international télégraphique et télé-
phonique, International Standard Organization, kod dlja
obměny i obrabotky informacij)]

rozšíření na 8 bitů:

8bitový kód: ASCII-8 (ISO-8, KOI-8)

mezera 20

'0' ... '9' 30 ... 39

'A' ... 'Z' 41 ... 5A

'a' ... 'z' 61 ... 7A

??? 80 ... FF — „národní abecedy“

kódy: KOI-8 čs

IBM – page 852 (Latin II.)

kód bratří Kamenických

ISO 8859

⋮