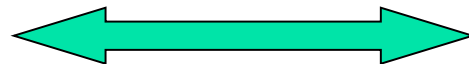


Aritmetika pohyblivej rádovej čiarky (FPU – Float/Floating Point Unit) (FPA -Floating Point Arithmetics)

Presnosť



Rozsah

Opakovanie:



Dnešný (mikro)počítač je stroj na spracovanie čísiel – číslic
Je podmnožinou číslicových počítačov- iné napr. analógové počítače



Poznáme:

- Prirodzené čísla: 1,2,3,4, ...
- Celé čísla: -3,-2-1,0,1,2,3,4, ...
- Racionálne čísla (cč/cč, okrem cč/0) môžeme zapísať v tvare:
Konečných, resp. nekonečných periodický desatinných zlomkov
- Iracionálne čísla: zapísané v tvare nekonečných neperiodických desatinných zlomkov



Čo môžeme zobrazit' do N bitov?

- Celé číslo bez znamienka:

0 až $2^N - 1$

- Celé číslo so znamienkom (Two's Complement)

$-2^{(N-1)}$ až $2^{(N-1)} - 1$

A čo iné čísla?

- Veľmi veľké čísla (sekundy v storočí)
 $3,155,760,000_{10} \quad (3.15576_{10} * 10^9)$
- Veľmi malé čísla (priemer atómu)
 $0.0000000110 \quad (1.010 * 10^{-8})$
- Racionálne (napr. periodické) $2/3$ (0.666666666. . .)

Racionálne=Rozumné- vyjadriteľné zlomkom - predstavujúce pomer

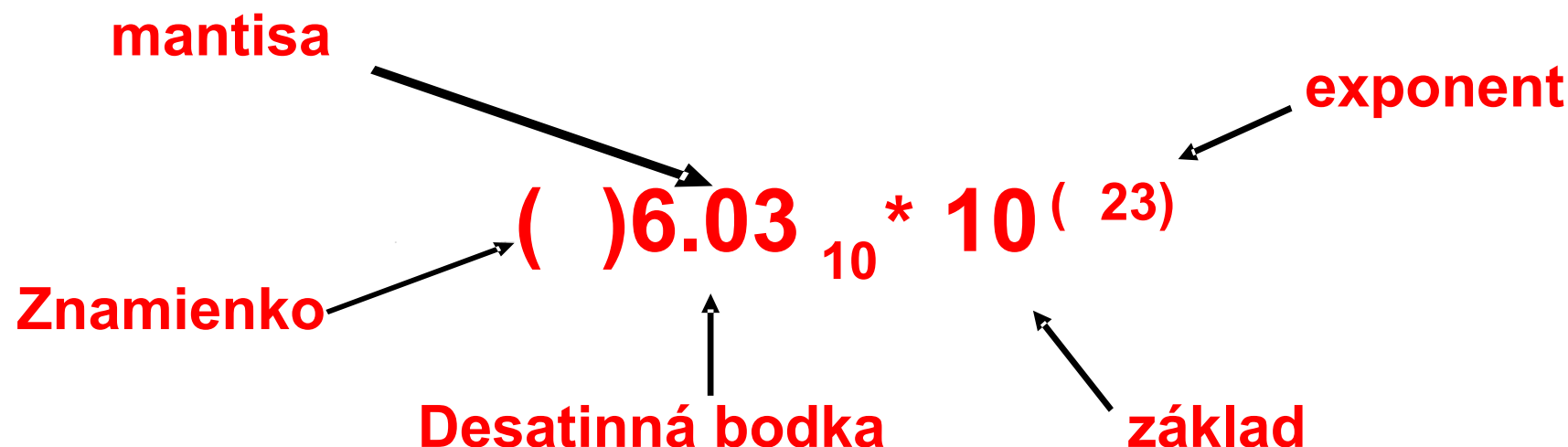
- Iracionálne čísla $2^{1/2}$ (1.414213562373. . .)
 e (2.718...), π 3.141...

Iracionálne=NERozumné- nevyjadriteľné zlomkom -

π predstavuje pomer obvodu kruhu k jeho priemeru

ale kružnica nemá ani začiatok ani koniec a tak je „nerozumná“

Vedecké zobrazenie čísiel (dekadické)



- **Normalizovaný tvar zápisu:** bez vodiacich núl

(naľavo od desatinnej bodky je len jedna nenulová platná číslica)

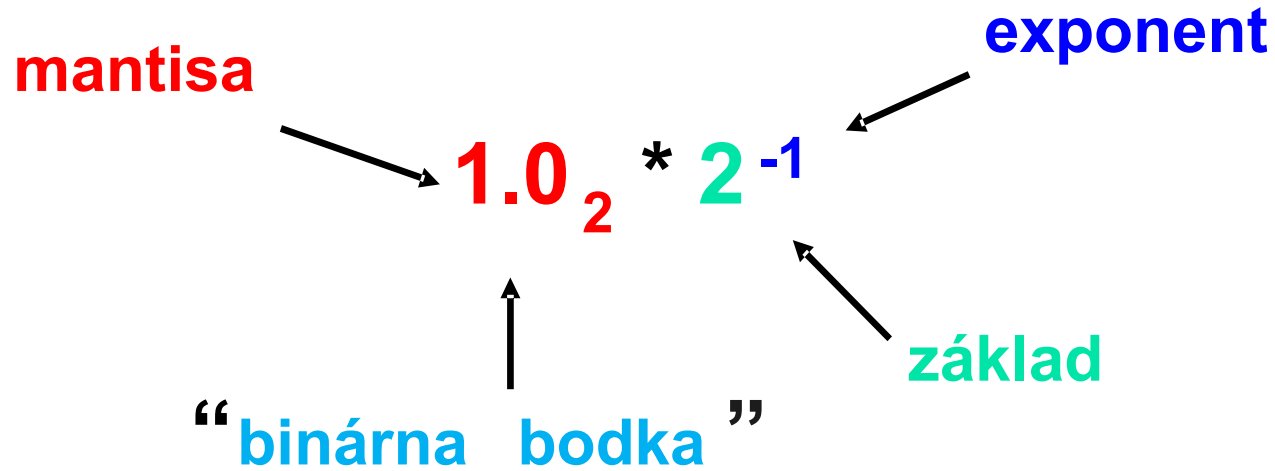
- Iný spôsob zápisu: 1/1 000 000 000

- Normalizovaný: $1.0 * 10^{-9}$

- Nenormalizovaný: $0.1 * 10^{-8}$

$10.0 * 10^{-10}$

Vedecké zobrazenie čísel (binárne)



Potrebuje zapísať:

- „znamienko“ mantisy
- „znamienko“ exponentu

Počítače podporujú prácu s číslami typu float (floating point = plávajúca bodka- desatinná bodky v čísle je na iných rádoch podľa exponentu= teda „pláva“ medzi rádmi):

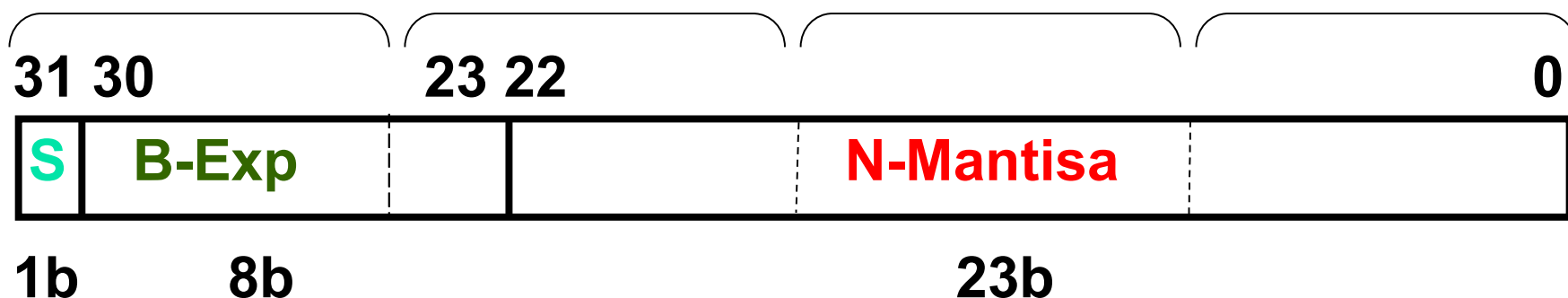
Forma zápisu znamienka ???

- priamy kód
- jednotkový doplnok
- dvojkový doplnok

Jednoduchá presnosť čísiel FP

(Single Precision – SP, C: float)

- Formát zápisu: $(+/-)1.\text{xxxxxxxxxx}_2 * 2^{\text{yyyy}}_2$
- Počet bitov: 32 bits



Mantisa: (priamy kód)

S - Sign znamienko mantisy

$|Mantisa| = 1.\text{xxxxxxxxxx}$, $\text{xxxxxxxxxx} = \text{N-Mantisa}$

- Exponent = B-Exp – Bias, Bias = 127, B-Exp = $\langle 1, 254 \rangle$
- Čísla z rozsahu: $2^{-126}(1.0) \sim 2^{+127}(2 - 2^{-23})$
t.j. $1.18 * 10^{-38} \sim 3.40 * 10^{38}$

Zobrazenie FP čísiel

- Čo sa stane ak je výsledok veľmi veľký?

($> 3.403 \cdot 10^{38}$) Overflow!

- Overflow \Rightarrow Exponent väčší ako sa dá zobrazit' do 8 bitov

- Čo sa stane ak je výsledok veľmi malý?

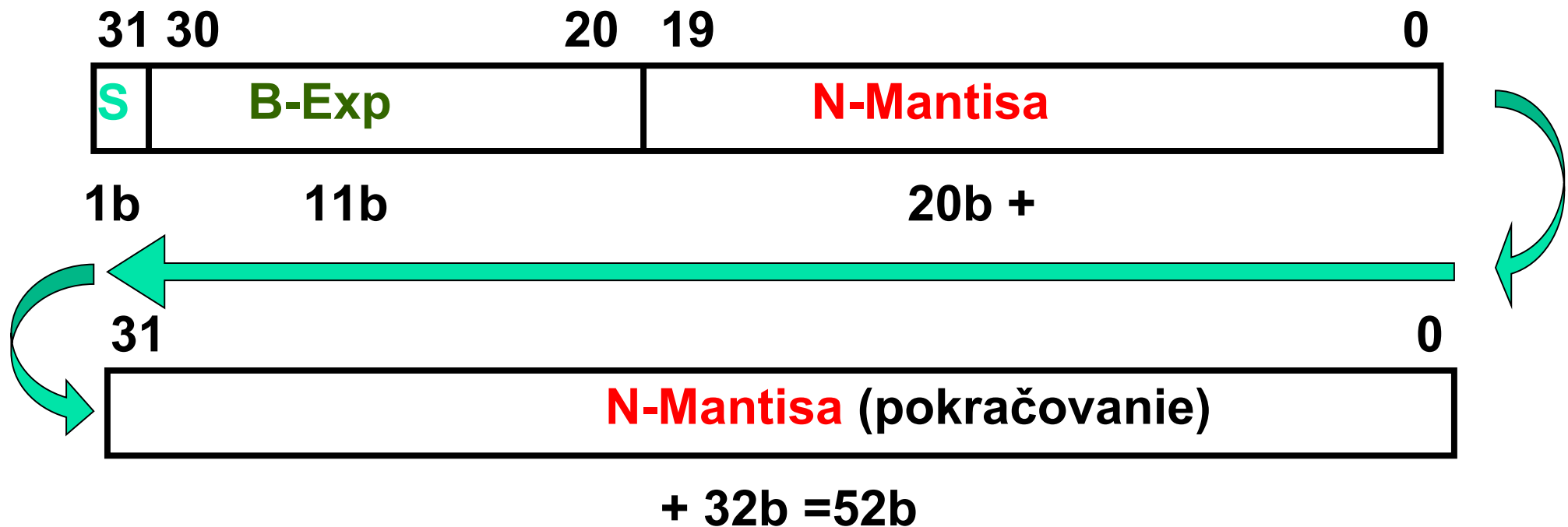
($> 0, < 1.17 \cdot 10^{-38}$) Underflow!

- Underflow \Rightarrow Záporný exponent “väčší” ako sa dá zobrazit' do 8-bitov

- Ako zabránime: pretečeniu – overflow,
podtečeniu – underflow?

Dvojnásobná presnosť čísla FP (Double Precision – DP)

C-ko: double $2 * 32 = 64$ bitov)



Jednoduchá presnosť

Bias: 127

Dvojnásobná presnosť

1023

Čísla z rozsahu:

$$2^{-126} (1.0) \sim 2^{+127} (2 \cdot 2^{-23})$$

$$2.0 \cdot 10^{-308} \sim 2.0 \cdot 10^{308}$$

Väčšou výhodou je vyššia presnosť

Norma IEEE 754, Zdôvodnenie (1/4)

- Jednoduchá presnosť, Dvojnásobná presnosť
- Znamienkový bit - S: 1 - záporné číslo
 0 - kladné číslo
- Mantisa:
 - Vodiaca jednotka sa nepíše v normovanom čísle
 - 1 + 23 bitov pre SP,
 - 1 + 52 bitov pre DP
 - Interval:
 - <1, 2) alebo identicky
 - <0, 1) bez „vodiacej jednotky“
- Poznámka: 0 - číslo nula. Nemá vodiacu – skrytú jednotku,
⇒ Špeciálny zápis pre vyjadrenie čísla nula,
rezervovaný špeciálny exponent

Norma IEEE 754 (2/4)

- Niekedy by sme chceli použiť „float“ aj v takom prípade, keď nemáme FP hardware (staré ARM, i386SX); napr., triediť pomocou celočíselného porovnávania záznamov
 - V takomto prípade „rozbijeme“ FP číslo na tri časti
 - Porovnáme znamienka,
 - Porovnáme exponenty,
 - Potom porovnáme normované mantisy
 - Dá sa predpokladať, že porovnávanie po skupinách bude rýchlejšie, a zvlášť vtedy, keď porovnávané čísla budú len celé kladné
 - Porovnávanie vykonáme v poradí:
 1. Znamienkový bit: záporné < kladné
 2. Exponent: väčšie číslo má väčší exponent
 3. Normovaná mantisa: väčšie číslo má väčšiu mantisu
- Porovnávanie „zastavíme“ pri prvej nezhode

Norma IEEE 754 (3/4)

■ Záporný Exponent ?!?!

- 2's comp? $1.0 * 2^{-1}$?<= >? $1.0 * 2^{+1}$ ($1/2$?<= >? 2)

1/2

0	1111	1111	000	0000	0000	0000	0000	0000
---	------	------	-----	------	------	------	------	------

2

0	0000	0001	000	0000	0000	0000	0000	0000
---	------	------	-----	------	------	------	------	------

- „Celočíselné porovnávanie“ týchto čísiel

- Porovnanie: $1/2$?<= >? 2 dá $1/2 > 2$!

■ Celé kladné číslo 0000 0001 je zápornejšie ako, celé

kladné číslo 1111 1111 \Rightarrow ľahko sa porovnávajú

- $1.0 * 2^{-1}$?<= >? $1.0 * 2^{+1}$ ($1/2$?<= >? 2)

1/2

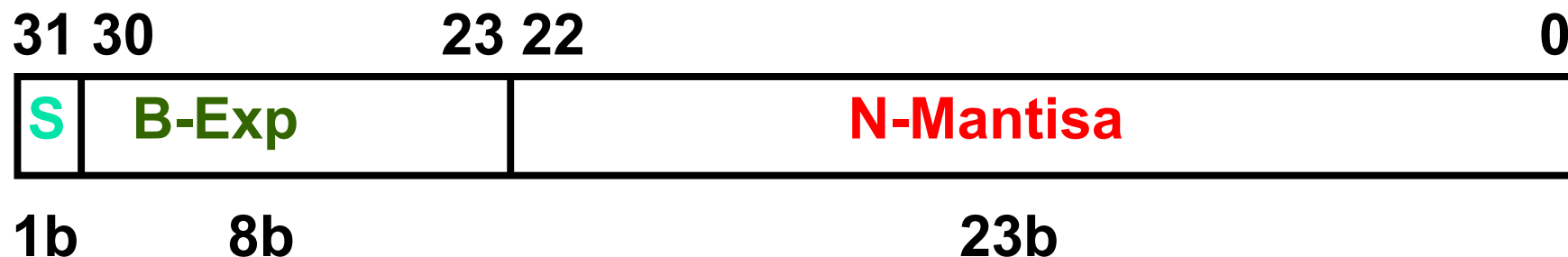
0	0111	1110	000	0000	0000	0000	0000	0000
---	------	------	-----	------	------	------	------	------

2

0	1000	0000	000	0000	0000	0000	0000	0000
---	------	------	-----	------	------	------	------	------

Norma IEEE 754 (4/4)

- Takéto riešenie sa volá: zápis exponentu s posunutou nulou,
- Ak odpočítame od posunutého exponentu posunutie, dostaneme skutočný exponent
 - IEEE 754: posunutie pre: SP: $B = 127$
DP: $B = 1023$
- Jednoduchá presnosť:



- $(-1)^S * (1 + \text{N-Mantisa}) * 2^{(\text{B-Exp} - 127)}$

Zápis pre DP je rovnaký, len posunutie je 1023 a počet bitov je dvojnásobný

N-Mantisa (1/2)

- Spôsob 1. (Zlomky):

- Dekadické číslo: $0.340_{10} \Rightarrow 340_{10}/1000_{10}$
 $\Rightarrow 34_{10}/100_{10}$

- Binárne číslo: $0.110_2 \Rightarrow 110_2/1000_2 = 6_{10}/8_{10}$
 $\Rightarrow 11_2/100_2 = 3_{10}/4_{10}$

N-Mantisa (2/2)

- Spôsob 2. (Hodnota pozície):

- Dekadicky: $1.6732 = (1 \cdot 10^0) + (6 \cdot 10^{-1}) + (7 \cdot 10^{-2}) + (3 \cdot 10^{-3}) + (2 \cdot 10^{-4})$

- Binárne: $1.1001 = (1 \cdot 2^0) + (1 \cdot 2^{-1}) + (0 \cdot 2^{-2}) + (0 \cdot 2^{-3}) + (1 \cdot 2^{-4})$

$$M = 1.\text{xxx...x}_2$$

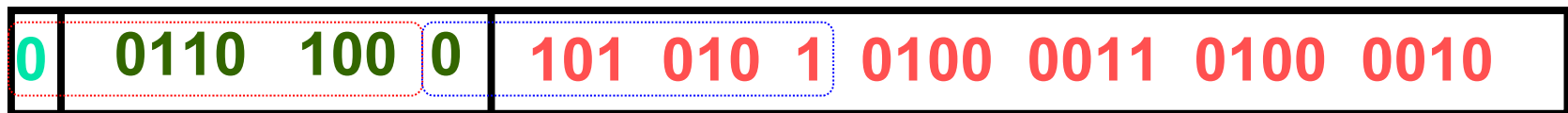
xxx...x: bity normovanej mantisy za “binárnou bodkou”

Minimum: 000...0 ($M = 1.0$)

Maximum: 111...1 ($M = 2.0 - \varepsilon$)

ε - strojová nula

Pr.: Prevod binárneho FP čísla na dekadické



■ Znamienko: 0 \Rightarrow kladné

■ Exponent:

■ $0110\ 1000_2 = 104_{10}$ B-Exp

■ „Vypočítanie“ exponentu: $104 - 127 = -23$

■ Mantisa:

■ $1 + 1 \cdot 2^{-1} + 0 \cdot 2^{-2} + 1 \cdot 2^{-3} + 0 \cdot 2^{-4} + 1 \cdot 2^{-5} + \dots =$
 $= 1 + 2^{-1} + 2^{-3} + 2^{-5} + 2^{-7} + 2^{-9} + 2^{-14} + 2^{-15} + 2^{-17} + 2^{-22} =$
 $= 1.0_{10} + 0.666115_{10}$

■ Predstavuje číslo: $1.666115_{10} \cdot 2^{-23} \sim 1.986 \cdot 10^{-7}$
(približne 2/10 000 000)

Prepočet desatinného čísla na FP číslo (1/3)

- Jednoduché: Ak je menovateľ mocninou 2, t.j. ak (2, 4, 8, 16, atď.), potom je to ľahké.
- Napr.: -0.75
 - $-0.75 = -3/4$
 - $-11_2 / 100_2 = -0.11_2$
 - **Normovanie:** $-1.1_2 * 2^{-1}$
 - $(-1)^S * (1 + \text{N-Mantisa}) * 2^{(\text{B-Exp} - 127)}$
 - $(-1)^{\text{1}} * (1 + .100\ 0000 \dots 0000) * 2^{(126 - 127)}$



Prepočet desatinného čísla na FP číslo(2/3)

- Zložitejší prípad: Ak menovateľ nie je mocninou 2.
 - Potom dané číslo nezobrazíme presne.
 - Aby bolo zobrazenie čo najpresnejšie, použijeme „veľa“ bitov mantisy.
 - Keď máme mantisu, správne číslo pre exp. už získame ľahko.
 - ????? Mantisa ???????

V knižniciach sa často používa jednotková mantisa 1,000...0 a určí sa exponent, ktorý dá najbližšie nižšie číslo a najbližšie vyššie číslo.

Potom sa niekedy urobí spresnenie:

Pridávajú jednotkové bity do mantisy a určí sa najbližšia nižšia mantisa pre nižší exponent a porovnajú sa vzdialenosti

Prepočet desatinného čísla na FP číslo (3/3)

- Je zřejmé, že ... Racionálne čísla (x_{10}) majú veľa platných číslíc.
- Podobne to platí aj pre ich binárny ekvivalent
- Prepočet racionálneho čísla:

Ak nevieme zobrazit' číslo v tvare $x/2^k$

výsledok prevodu vyzerá nasledovne:

Des. hodnota	Dvojkové číslo
	(niekoľko bitov sa zopakuje)
1/3	0.0101 0101 01[01]... ₂
1/5	0.0011 0011 0011 [0011]... ₂
1/10	0.0001 1001 1001 1[0011]... ₂

Príklad:

Čo je dekadický ekvivalent FP čísla ?

1	1000 0001	111 0000 0000 0000 0000 0000
---	-----------	------------------------------

1: -1.75

2: -3.5

3: -3.75

4: -7

5: -7.5

6: -15

7: $-7 * 2^{129}$

8: $-129 * 2^7$

Odpoved':

Dekadický ekvivalent FP čísla:

1	1000 0001	111 0000 0000 0000 0000 0000
---	-----------	------------------------------

S B-Exp

N-Mantisa

$$(-1)^S * (1 + \text{N-Mantisa}) * 2^{(\text{B-Exp}-127)}$$

$$(-1)^1 * (1 + .111) * 2^{(129-127)}$$

$$-1 * (1.111) * 2^2$$

-111.1

-7.5

Alternatívne posledné dva riadky

$$-1 * (1 + 1/2 + 1/4 + 1/8 + 0/16 + 0/32 + \dots) * 2^2 =$$

$$-1 * (8 + 4 + 2 + 1) / 8 * 4 = -1 * 15 / 8 * 4 = -15 / 2 = -7,5$$

1: -1.75

2: -3.5

3: -3.75

4: -7

5: -7.5

6: -15

7: $-7 * 2^{129}$

8: $-129 * 2^7$

“Na záver ”

- Floating Point čísla sú len náhradou tých čísiel, ktoré sme chceli použiť
- IEEE 754 Floating Point Standard je v praxi najrozšírenejší spôsob zápisu takýchto čísiel (**premenovaný na ISO/IEC/IEEE 60559:2011** od r- 2011)
- Od roku ~ 1997 túto normu používa prakticky každý počítač

Viac o FP číslach :

Doteraz sme uvažovali **B-Exp** v rozsahu:
<1 až 254>

Na čo je použitá „0“ a „255“?

Znázornenie ∞

■ V FP aritmetike, delenie 0 dá ∞ , nie pretečenie.

■ Prečo?

- Ak existuje v FP aritmetike ∞ potom výraz $X/0 > Y$ je platné porovnanie
- IEEE 754 (aj ISO/IEC/IEEE 60559:2011) vie zobrazit' ∞
- Najkladnejší exponent **B-Exp = 255** je rezervovaný pre ∞
- N-Mantisa je nulová
- Kladné ∞

$$+ \infty = +1.0 * 2^{128}$$



- Záporné ∞

$$- \infty = -1.0 * 2^{128}$$



Zobrazenie „0“

- Posunutý exponent, samé nuly: $B-Exp = 0$
- rovnako normovaná mantisa samé nuly
- A čo znamienko?

■ +0:	0	0 0 0 0 0 0 0 0	0 0 0 0 0 0 0
■ -0:	1	0 0 0 0 0 0 0 0	0 0 0 0 0 0 0

- Prečo dve nuly?
 - Výhodné pri limitných porovnávaniach

Špeciálne čísla

- Čo ešte môžeme dodefinovať v (Single Precision) ?

B-Exp

N-Mantisa

Výsledok (číslo)

0

0

0

0

nenulová ($\neq 0$)

???

1-254

nenulová

+/- normované FP čísla

255

0

+/- ∞

255

nenulová ($\neq 0$)

???

- Zostalo nám: “Skúsime využiť”

- B-Exp = 0, 255 & N-Mantisa $\neq 0$...

Čísla typu: Not a Number (NaN)

■ Čo je $\sqrt{-4}$ alebo 0/0?

■ Ak ∞ nie je „chyba“, potom by nemuselo byť ani napr. 0/0 .

■ Zaužíval sa názov: Not a Number (NaN)

■ B-Exp = 255, N-Mantisa nenulová

■ Načo je to dobré?

■ Dá sa predpokladať, že NaN sa využijú pri debugovaní? Voš=bug velektrónke v počítači spôsobila prvú chybu vo výpočte počítača- preto odvšivovanie

■ Napr.: $\text{op}(\text{NaN}, X) = \text{NaN}$

Zápis nenormovaných čísiel: (1/2)

- Problém: FP čísla zapísané v normalizovanom tvare generujú okolo nuly „diery“

- Najmenšie zobraziteľné kladné číslo:

$$a = 1.0..._2 * 2^{-126} = 2^{-126}$$

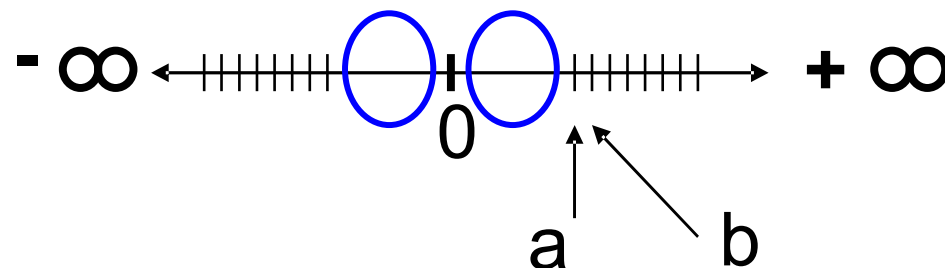
- Druhé najmenšie zobraziteľné kladné číslo:

$$b = 1.000.....1_2 * 2^{-126} = 2^{-126} + 2^{-149}$$

$$a - 0 = 2^{-126}$$

$$b - a = 2^{-149}$$

Diery (zbytočne veľká oblasť)!



Zápis nenormovaných čísiel: s (2/2)

■ Riešenie:

- Zatiaľ sme nepoužili **B-Exp = 0**, N-Mantisa nenulová
- Nenormované čísla: bez vodiacej jednotky,

Najmenší normovaný exponent = -126. (posúvame)

- Najmenšie zobraziteľné kl. číslo:

$$a = 2^{-149} \quad (126+23=149) \quad \text{bez nenormovaných len } a=2^{-126}$$

- Druhé najmenšie zobraziteľné kl. číslo:

$$b = 2^{-148}$$

⇒ Diery okolo nuly sú menšie



Zhrnutie

B-Exp	N-Mantisa	Výsledok (číslo)
0	0	0
0	nenulová ($\neq 0$)	Nenormované FP čísla
1-254	nenulová	Normované FP čísla
255	nulová	$\pm \infty$
255	nenulová ($\neq 0$)	NaN

Zaokrúhľovanie

- Výpočty s reálnymi číslami \Rightarrow problém ako číslo umiestniť do odpovedajúceho priestoru.
- FP hardware obsahuje **2 špeciálne bity** pre presnosť
(zníženie presnosti \Rightarrow zvýšenie rýchlosti)
 - 00** – 24 bitov (SP)
 - 10** – 53 bitov (DP)
 - 11** – 64 bitov (Extended P (vnútorne FPU 80bitov))-default na x86
- Zaokrúhľuje sa vždy pri konvertovaní...
 - DP \Rightarrow SP
 - Číslo FP na integer

IEEE pozná 4 módy zaokrúhľovania

Zaokrúhľovanie smerom $+\infty$

Vždy nahor : $2.1 \Rightarrow 3$, $-2.1 \Rightarrow -2$

Zaokrúhľovanie smerom $-\infty$

Vždy nadol : $1.9 \Rightarrow 1$, $-1.9 \Rightarrow -2$

Odrežanie

Jednoducho zahod' posledné bity (zaokrúhlenie smerom k 0)

Zaokrúhlenie na najbližšie číslo (default =prednastavené),

- vykonáme pripočítaním čísla 1 s váhou o 1 menej, ako je posledný platný rád.
- resp. párne, ak sú dve najbližšie čísla rovnako vzdialené.

Pr.: $2,5 \Rightarrow 2$; $3,5 \Rightarrow 4$

Vlastnosti - problémy FP aritmetiky (1/2)

- Presnosť \longleftrightarrow Rozsah
- Vedecké výpočty vyžadujú chybový manažment
- Nespomenuli sme: Napr.: nie je garantované:
 - $(1/r)*r \neq 1$ ale $r/r=1$, $a*b/c \neq a/c*b$, ak nepretečenie $a*b/c > a/c*b$
 - FPA aritmetika nie je asociatívna !!!!!
 $(A+B) + C \neq A + (B+C), (A*B) * C \neq A * (B*C)$
 - nie je ani vždy distributívna !!!!!
 $(A+B) * C \neq A*B + B*C$
- Implementovanie normy IEEE 754 je ťažké konverzia jednotkou mantisou(1,00...0) bez spresnenia je bežná v knižniciach a tam:
 $1/64 > 0.01 = 1/100 > 1/128 \Rightarrow$ chyba v každej operácii 0.2875 - 0.5626 centu
Blbec programuje účtovníctvo v FPA teda v EUR, SKK
Normálny a génus počíta v centoch resp. halieroch

Vlastnosti - problémy FP aritmetiky (2/2)

Zlé implementovanie FPP spôsobuje chyby:

- Vid' „Pentium bug“!
- Raketa Patriot – „vojna v zálive“: Systém protivzdušnej obrany bol „zapnutý“ cca 100 hodín. Čas generovali ako načítavanie 0,1sek zapísané binárne do 24 b. Binárny ekvivalent 0.1sek je „nekonečné“ číslo => chyba pri **zaokrúhlení** cca 10^{-7} sek. Za 100hodín chyba narástla na $100 * 3600 * 10^{-7} = 0,36$ sek pri rýchlosti 1676m/sek je to cca 600metrov. “**Raketa minula prilietavajúci iracký Scud**”, ktorý zasiahol tábor americkej armády.
- Ariane 5 (v roku 1996) havarovala kvôli chybe pri **konverzii** čísla FP (64b) na signed integer (16b) (“nezmestilo sa”)
Predpokladaná cena nehody: \$500 million



Iné chyby v SW a tým aritmetike a „pokusy“

Chevrolet Volt svolávan k updatu softwaru

Jaromír Šetek , 24. říjen 2012 12:02

<http://www.itbiz.cz/zpravicky/chevrolet-volt-svolavan-k-updatu-softwaru>

Chevrolet Volt má ve svém **řídícím softwaru chybu, která způsobuje vypnutí motoru při jízdě**. Výrobce proto svolává vozy k aktualizaci softwaru, která má trvat přibližně hodinu.

Vozy Nissan budou mít plně elektronické řízení

Jaromír Šetek , 18. říjen 2012 07:56

<http://www.itbiz.cz/zpravicky/vozy-nissan-budou-mit-plne-elektronicke-rizeni>

Nissan začne do svých vozů během příštích 12 měsíců montovat robotické řízení, které odstraní mechanické spojení mezi volantem a koly. Technologie Nissanu je prvním řešením, které odstraňuje mechanické spojení řízení a kol. **Náhradou je elektronický systém se senzory, které snímají otáčení volantu a data jsou pak odeslána ke zpracování**. Servomotor následně natočí kola podle vyhodnocení dat. **Lidská činnost se tak stane jedním z mnoha vstupů ovlivňujících řízení.**

Iné chyby v SW a tým aritmetike a „pokusy“

Chevrolet Volt svolávan k updatu softwaru

Jaromír Šetek , 24. říjen 2012 12:02

<http://www.itbiz.cz/zpravicky/chevrolet-volt-svolavan-k-updatu-softwaru>

Chevrolet Volt má ve svém **řídícím softwaru chybu, která způsobuje vypnutí motoru při jízdě**. Výrobce proto svolává vozy k aktualizaci softwaru, která má trvat přibližně hodinu.

Software, not wetware, now the cause of lousy Volvo drivers

Auto-maker recalls cars that might take a nano-nap on the Motorväg

22 Feb 2016 at 05:02, Simon Sharwood

Volvo has issued a recall for cars that have buggy software.

The code flaw means that engines in about 59,000 2016-model series 60 and 70 cars sometimes stop, for no apparent reason.

The engines then restart.

http://www.theregister.co.uk/2016/02/22/software_not_wetware_now_the_cause_of_lousy_volvo_drivers/

Iné chyby v SW a tým aritmetike a „pokusy“

Úvaha: Jak chyba v algoritmu vyhladila planetu Zemi

10.8.2012 <http://pctuning.tyden.cz/hardware/multimedia-zvuk-tv/24750-uvaha-jak-chyba-v-algoritmu-vyhladila-planetu-zemi>

Prvního srpna se o hrůze rychlých, ale chybně vykonávaných příkazů přesvědčila firma Knight Capital. Na burze NYSE v době mezi

9:30 až 10:15 místního času spustili nový program pro rychloobrátkové

obchodování (HFT, High Frequency Trading), Při rychloobrátkovém obchodování se snažíte levně kupovat a draž prodávat, což se neděje s moc velkým ziskem, pokud to ale děláte vážně hodně rychle, můžete na tom vydělávat. Algoritmus místo toho začal kupovat za vyšší cenu a prodávat za nižší a to tak, že například na nákupu a prodeji akcií společnosti Exelon v každém jednom obchodu prodělal 15 centů.

Takových obchodů udělal 40 za sekundu, tedy 2400 za minutu.

A samozřejmě nebyly to jediné akcie, se kterými obchodoval

- podobných "obchodů" rozjel celou řadu. Protože šlo o „vytrvalé krvácení“

a nikoliv jeden vyšinutý obchod, jsou obchody platné a Knight Capital to přišlo na 440 milionů dolarů,

způsobilo propad jejich akcií a zcela vážně se dostali na hranici bankrotu. V únoru 2011 automatické HFT systémy vyrobily propad ceny cukru o 6% za jednu sekundu, o měsíc později postihlo to samé kakao (13% pokles)

Iné chyby v SW a tým aritmetike a „pokusy“

Úvaha: Jak chyba v algoritmu vyhladila planetu Zemi

10.8.2012 <http://pctuning.tyden.cz/hardware/multimedia-zvuk-tv/24750-uvaha-jak-chyba-v-algoritmu-vyhladila-planetu-zemi>

Dne 25. ledna 1995 ruský radar včasného varování zachytil raketu odpálenou z oblasti Špicberk, vyložil si, že by mohlo jít o americkou ponorkovou raketu Trident, a odhadl, že dopadne na Moskvu za 5 minut. Systém se převedl do bojového módu a o možném útoku byl vyrozuměn i prezident Jelcin. Během pěti minut naštěstí sledovací systém zjistil, že raketa nemíří na Rusko – šlo o norskou vědeckou výzkumnou raketu, o jejímž odpálení Norsko už 16. ledna informovalo s předstihem 35 zemí včetně Ruska.

Při pádu burzy v roce 1929 velcí bankéři měli den na to, aby se sešli, promysleli postup a pokusili se něco udělat. Flash crash z roku 2010 se odehrál během pěti minut !!!! <http://pctuning.tyden.cz/component/content/27678/27678?task=view&start=7>

Stíhací bombardér F-22 Raptor má softvér o velikosti 1,7mil. řádkov kódu

- Dopravné lietadlo Boeing B787 Dreamliner používá asi 6,5 milióna řádkov kódu.
- Moderné luxusné autá obsahovali, v roku 2009 , približne 100 miliónov řádkov kódu, (dnes je to softvér na kolesách s 300 mil. řádkami)
- Mercedes triedy S, má až 20 miliónov řádkov kódu v navigačnom systéme spojenom s rozhlasovým príjimačom

Sarkazmus: Prečo je letecká doprava najbezpečnejšia?

Lebo má najmenej SW a SW nevyvíjajú len IT absolventi

Iné chyby v SW a tým aritmetike,online aktualizácie aut

BMW nechalo bezpečnostnú dieru v iDrive, milióny aut mohli odemknúť kde kedy 2.2.2015 <http://www.autoforum.cz/zajimavosti/bmw-nechalo-bezpecnostni-diru-v-idrive-miliony-aut-mohl-odemknout-kde-kdo/>

Problém, ktorý sa týkal 2,2 miliónu vozů BMW a ďalších aut Rolls Royce a Mini, vyrobených medzi březnem 2010 a 8. prosincem 2014 a vybavených ConnectedDrive. Po nálezoch chyby nemeckým autoklubom nasledovalo rýchle odstránenie nedostatku zo strany BMW; podľa slov automobilky nyní už všetky dotčené systémy boli aktualizované, a to automaticky po pripojení ke službám ConnectedDrive.

Maslo na hlavu v tomto má aj STU ale aj riešenie je z STU..

„Vyvíjam softvér pre najnovšie modely BMW“ (ešteže nie je Nagy z FEI) 16.10.2014 <http://hn.hnonline.sk/ekonomika-a-firmy-117/vyvijam-softver-pre-najnovsie-modely-bmw-633080>

Smolenický rodák Roman Nagy vedie v Mníchove vývojársky tím, ktorý pripravuje počítačové systémy špičkových vozidiel bavorskej automobilky.

- Absolvoval štúdium aplikovanej informatiky na MtF STU v Trnave
- Od roku 2009 pracuje v Mníchove vo firme BMW, kde nastúpil ako vývojový pracovník.
- Začiatkom roku 2013 ho vymenovali do pozície vedúceho vývojového tímu, v ktorej dodnes riadi tím 20 softvérových vývojárov a testerov.38 / 42

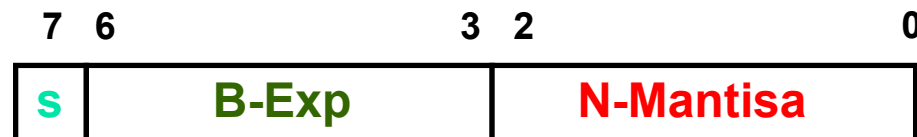
Pr.: Zobrazenie FP čísiel - “malých” (1/3)

■ 8-bitov FP číslo

- Znamienko bit č.7.
- 4 bity exponent, s posunutím 7.
- 3 bity normov. mantisa

■ Niečo čo sa podobá na IEEE Formát

- Normované a nenormované čísla
- zobrazenie 0, NaN, nekonečna)



Pr.: Zobrazenie FP čísiel - “malých” (2/3)

Posunutie =7

B-Exp	B-Exp ₂	E ₁₀	2 ^E	
0	0000	-6	1/64	(nenormované)
1	0001	-6	1/64	(normované)
2	0010	-5	1/32	
3	0011	-4	1/16	
4	0100	-3	1/8	
5	0101	-2	1/4	
6	0110	-1	1/2	
7	0111	0	1	
8	1000	+1	2	
9	1001	+2	4	
10	1010	+3	8	
11	1011	+4	16	
12	1100	+5	32	
13	1101	+6	64	
14	1110	+7	128	
15	1111	+8	(∞, NaN).	

Pr.: Zobrazenie FP čísiel - “malých” (3/3)

	S	B-Exp	N-Man	E ₁₀	číslo	
Nenormaliz. čísla	0	0000	000	-6	0	
	0	0000	001	-6	$1/8 * 1/64 = 1/512$	← skoro nula
	0	0000	010	-6	$2/8 * 1/64 = 2/512$	
	...					
	0	0000	110	-6	$6/8 * 1/64 = 6/512$	
	0	0000	111	-6	$7/8 * 1/64 = 7/512$	← najväčš. nenorm.
Normaliz. čísla	0	0001	000	-6	$8/8 * 1/64 = 8/512$	← najmen. norm.
	0	0001	001	-6	$9/8 * 1/64 = 9/512$	
	...					
	0	0110	110	-1	$14/8 * 1/2 = 14/16$	
	0	0110	111	-1	$15/8 * 1/2 = 15/16$	← skoro 1 (<1)
	0	0111	000	0	$8/8 * 1 = 1$	
	0	0111	001	0	$9/8 * 1 = 9/8$	← skoro 1 (>1)
	0	0111	010	0	$10/8 * 1 = 10/8$	
	...					
	0	1110	110	7	$14/8 * 128 = 224$	
	0	1110	111	7	$15/8 * 128 = 240$	← najväčš. norm
	0	1111	000	8	∞	

Literatúra:

- [1] Clements,A: The Principles of Computer Hardware,
Oxford
- [2] Stalling, W.: Computer Organization and Architecture,
principles ...,
- [3] Jelšina, M.: Architektúry počítačových systémov,