1. Číselné soustavy, binární aritmetické operace, převody mezi soustavami (10, 2, 16), doplňkový a aditivní kód, zobrazení čísla bez a se znaménkem (8bit), přetečení.

HARDWARE A APLIKAČNÍ SOFTWARE

Číselné soustavy

Poziční číselné soustavy

- Charakterizovány tzv. základem neboli bází (anglicky radix, značí se r).
- R je kladné celé číslo definující maximální počet číslic, které jsou v dané soustavě k dispozici.
- Poziční soustavy (kromě jedničkové) se nazývají také polyadické (= vlastnost, že číslo v nich zapsané lze vyjádřit součtem mocnin základu dané soustavy vynásobených příslušnými platnými číslicemi)

  Dvojková (BIN)
  - r = 2; binární
  - Přímá implementace v digitálních elektronických obvodech (použitím logických členů), čili interně ji používají všechny moderní počítače.
  - Mocniny čísla 2.
  - Používá dva symboly (0 a 1) odpovídají dvěma jednoduše rozdělitelným stavům elektrického obvodu.
    - vypnuto × zapnuto
    - nepravda × pravda

  Osmičková (OCT)
  - r = 8; oktální, oktalová

  Desítková (DEC)
  - r = 10; decimální, dekadická
  - Nejpoužívanější v běžném životě.

  Dvanáctková
  - r = 12
  - Dnes málo používaná, ale dodnes z ní zbyly názvy prvních dvou řádů – tucet a veletucet.

  Šestnáctková (HEX)
  - r = 16; hexadecimální
  - Používá se v oblasti informatiky, pro číslice 10 až 15 se používají písmena A až F.

  Šedesátková
  - r = 60
  - Používá se k měření času pro zlomky hodiny.
  - Číslice se obvykle zapisují desítkovou soustavou jako 00 až 59 a řády se oddělují dvojtečkou.
  - Staré názvy prvních dvou řádů jsou kopa a velekopa.

Nepoziční číselné soustavy

- Způsob reprezentace čísel, ve kterém není hodnota číslice dána jejím umístěním v dané sekvenci číslic.

- Tyto způsoby zápisu čísel se dnes již téměř nepoužívají a jsou považovány za zastaralé.
- A=1, B=10, C=100, D=1000, pak by vyjádřením čísla 3542 mohl být například řetězec „AABBBBCCCCCDDD“, ale stejně dobře i „ACDABBCCCCDDBB“ apod. (z hlediska hodnoty, ale za cenu horší srozumitelnosti).
- Často neobsahovaly symbol pro nulu a záporná čísla.
- Dlouhý zápis čísel, která výrazně převyšují hodnotu největšího symbolu soustavy

Římské číslice

- Římské číslice se píšou velkými písmeny abecedy: I = 1, V = 5, X = 10, L = 50, C = 100, D = 500, M = 1000.
- Při zápisu pomocí římských číslic v podstatě neexistuje žádná směrodatná norma. Obvykle se kombinují nejvýše tři stejné římské číslice (III = 3, XXX = 30)

Binární aritmetické operace

Sčítání

- Binární čísla je možné sčítat stejným způsobem, jakým sčítáme čísla desítková.
- K přenosu jedničky do vyššího řádu dojde tehdy, je-li výsledkem součtu dvou čísel pod sebou hodnota větší nebo rovna 10_2.
- Ne náhodou je číslo ciferně shodné s číslem dekadickým. Musíte si však uvědomit, že $10_2 = 2_{10}$.
- Postup začneme v příkladu tak, že v pravém sloupci: 0 + 1 = 1, v prostředním sloupci je opět 1 + 0 = 1 a ve sloupci levém je 1 + 1 = 102.
- Zde dojde k přenosu jedničky, zapíšeme tedy nulu a do nejvyššího řádu připíšeme jedničku.

Binárně	Dekadicky
1 1 0	6
+ 1 0 1	+ 5
----------	----------
1 0 1 1	1 1

Sečtěte:

	1	1	1	1	0	0	0	1
+	1	0	1	0	1	0	0	0
1	1	0	0	1	1	0	0	1

Odčítání

- Budeme postupovat obdobným způsobem.
- V příkladu nejprve odečteme 1 - 1 = 0 v pravém sloupci tak, jak jsme zvyklí, a v následujícím sloupci 1 - 0 = 1.
- Ve třetím sloupci zprava pak počítáme rozdíl 0 - 1 (odečítáme větší číslo od menšího). U desítkových čísel si v takovém případě vypomáháme přidáním jedničky ve vyšším řádu menšence (číslo, od kterého odčítáme).
- Tu vykompenzujeme tím, že ji odečteme v následujícím sloupci vlevo (dojde tedy k přenosu -1). Např. rozdíl 5 - 8 spočítáme jako 15 - 8 = 7, zapíšeme sedm a v následujícím vyšším řádu odečteme jedničku. Stejným způsobem budeme postupovat i zde.
- Místo 0 - 1 tedy budeme počítat 10 - 1 = 1 a do výsledku zapíšeme 1.

- V dalším řádu (v levém sloupci) odečteme jedničku, kterou jsme si vypůjčili, tedy 1 - 1 = 0, přičemž nulu již do výsledku nezapisujeme.

Binárně	Dekadicky
1 0 1 0	1 1
- 1 0 1	- 5
----------	----------
1 1 0	6

Odečtěte:

	1	1	1	0	1	0	0	0
-	0	0	0	1	0	1	1	1
0	1	1	0	1	0	0	0	1

Násobení

- Způsob násobení binárních čísel se nijak neliší od způsobu, jakým násobíme čísla desítková.
- Při samotném násobení vlastně ani nijak nepocítíme, že se jedná o binární čísla.
- Rozdíl nastane až při sčítání mezivýsledků, kdy budeme postupovat způsobem popsaným výše.
- Všimněte si, že v podstatě neděláme nic jiného, než že horní číslo buď opisujeme v nezměněné podobě, pokud násobíme jedničkou, nebo píšeme samé nuly.
- Bez zajímavosti není také násobení dvěma. Můžete si vyzkoušet vynásobit jakékoliv binární číslo dvěma.
- Dvojka je v binární soustavě reprezentována číslem 10_2.
- Efekt bude stejný jako kdybyste v desítkové soustavě násobili deseti. Budeme v podstatě jen přidávat nuly zprava.
- Násobení (i dělení) binárního čísla mocninami dvojky se tak stává velice snadnou záležitostí.

Binárně	Dekadicky
1 0 1 1	1 1
* 1 0 1	* 5
--------------	-------
1 0 1 1	5 5
0 0 0 0	
1 0 1 1	
--------------	
1 1 0 1 1 1	

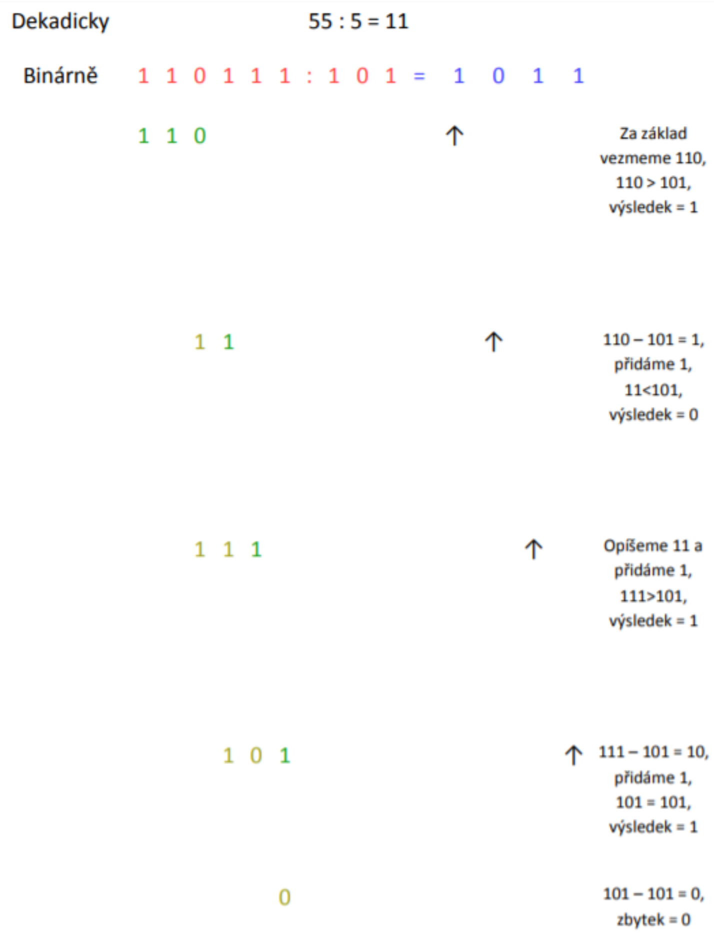
Vynásobte:

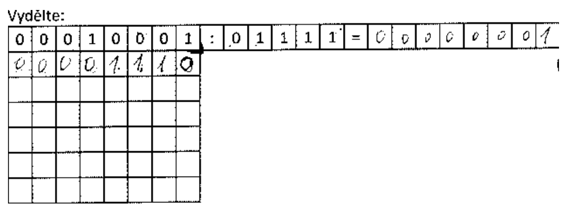
	0	0	0	0	0	1	1	1	0	0
*	0	0	0	0	0	1	0	1	1	1
	0	0	0	0	0	1	1	1	0	0
0	0	0	0	0	1	1	1	0	0	
0	0	0	0	1	1	1	0	0		
0	0	0	0	0	0	0	0			
0	0	1	1	1	0	0				
0	1	0	1	0	0	0	0	1	0	0

Dělení

- Při dělení si vystačíme v podstatě jen s odčítáním.
- Za základ dělení si vezmeme takovou část dělence, která je větší nebo rovna děliteli, ale menší než jeho dvojnásobek, v našem případě tedy číslo 110_2 (viz první řádek s komentáři v tabulce).

- Nyní provedeme podíl 110:101 (zvolené číslo vydělíme dělitelem tak jak jsme tomu zvyklí u dělení desítkových čísel).
- Výsledkem by byla nula v případě, že by bylo 110<101. My jsme však číslo zvolili záměrně tak, aby bylo větší nebo rovno číslu 101 (děliteli) a v takovém případě je výsledkem podílu jednička.
- Vidíme, že se nám tak celé dělení redukuje na porovnávání velikostí.
- Nyní vezmeme náš výsledek (1), vynásobíme jím dělitele (1·101) a odečteme ho od hodnoty 110:110 - 1·101 = 1 (druhý řádek komentářů v tabulce).
- K číslu 1 přidáme příslušnou cifru dělence a pokračujeme postupem uvedeným výše.
- 11:101 = 0, protože je 11<101. Nyní od čísla 11 neodečítáme nic (0·101=0) a můžeme tak rovnou pokračovat připsáním další číslice dělence (třetí řádek komentářů v tabulce).
- Protože 111>101, píšeme 111:101 = 1. Zapíšeme výsledek a provedeme rozdíl 111 - 1·101 = 10 (čtvrtý řádek).
- K číslu 10 přidáme poslední číslici dělence a vzniklé číslo vydělíme dělitelem 101:101 = 1 (dělíme-li dvě stejně velká čísla, výsledkem je jednička).
- Po odečtení 101 - 101 nám vyjde nulový zbytek.
- Pokud by byl zbytek nenulový, mohli bychom pokračovat v dělení standardním způsobem – k výsledku bychom připsali desetinnou čárku a ke zbytku připsali další cifru dělence (jsou to již jenom nuly, kterých si můžeme vpravo za desetinnou čárkou přidat kolik chceme – např. 110111,00000…).





Převody mezi soustavami (10, 2, 16)

Převod z desítkové do dvojkové

- Nechť máme na papíře číslo 70.
- Toto číslo budeme nyní chtít převést do dvojkové, binární soustavy.
- Princip je poměrně jednoduchý, číslo, které chceme převést, dělíme neustále dvojkou, až dojdeme k nule, přičemž si zapisujeme zbytky po celočíselném dělení.
- Pokud chceme převést číslo do jiné soustavy, například do šestnáctkové, budeme dělit šestnáctkou.
- Pokud do šestkové, dělíme šestkou.
- Takže v praxi to bude vypadat takto:

$$\begin{aligned} 70:2&=35 &&\longrightarrow 0 \quad \text{(zbytek po dělení)}\\ 35:2&=17 &&\longrightarrow 1\\ 17:2&=8 &&\longrightarrow 1\\ 8:2&=4 &&\longrightarrow 0\\ 4:2&=2 &&\longrightarrow 0\\ 2:2&=1 &&\longrightarrow 0\\ 1:2&=0 &&\longrightarrow 1 \end{aligned}$$

- Výsledné číslo ve dvojkové soustavě udávají zbytky po dělení.
- Nebereme ale zbytek svrchu, ale od spodu.
- Takže číslo **70** v binární soustavě je **1000110**.

Převod z dvojkové do desítkové soustavy

- Mějme číslo 1100010 a převeďme ho do desítkové soustavy.
- Tento směr je jednodušší, stačí vypočítat tento součet:

$$1100010_{10} = 1\cdot 2^6 + 1\cdot 2^5 + 0\cdot 2^4 + 0\cdot 2^3 + 0\cdot 2^2 + 1\cdot 2^1 + 0\cdot 2^0$$

- Každý sčítanec má tvar $x \cdot 2^i$, kde x je číslice z původního binárního čísla, a i se zprava postupně zvětšuje vždy o jedna.
- Protože převádíme číslo 1100010, vypadá tento součet takto:

$$1100010_{10} = \boxed{1}\cdot 2^6 + \boxed{1}\cdot 2^5 + \boxed{0}\cdot 2^4 + \boxed{0}\cdot 2^3 + \boxed{0}\cdot 2^2 + \boxed{1}\cdot 2^1 + \boxed{0}\cdot 2^0$$

- Číslo 1100010 má sedm číslic, takže mocniny u čísla dva budou postupně 6, 5, …, 1, 0.
- Po umocnění a vynásobení získáme výraz:

$$1100010_{10} = 64 + 32 + 2 = 98.$$

Převod do jiných soustav

- Předchozí postup na převod z desítkové do binární soustavy je natolik univerzální, že lze použít i na jiné soustavy.

- Pokud chceme převést číslo 185 do šestnáctkové soustavy, jen dělíme 16:

$$\begin{aligned} 185 : 16 &= 11 \longrightarrow 9 \quad \text{(zbytek po dělení)} \\ 11 : 16 &= 0 \longrightarrow 11 \end{aligned}$$

- Číslo 185 by v 16 soustavě mělo tvar (11, 9).
- Místo „číslic“ nad 9 se obvykle používají písmena, takže 10 = A, 11 = B, 12 = C, 13 = D, 14 = E, 15 = F.
- Můžeme tak napsat, že číslo 185 má v 16 soustavě tvar B9.
- Podobně můžeme převést číslo B9 z 16 soustavy do desítkové.

$$B9_{10} = 11 \cdot 16^1 + 9 \cdot 16^0 = 11 \cdot 16 + 9 = 185$$

Doplňkový a aditivní kód

Doplňkový kód/Dvojkový doplněk

- Způsob kódování celých, tedy kladných i záporných čísel ve dvojkové soustavě používaný v počítačích.
- Dvojkový doplněk umožňuje zjednodušit konstrukci aritmeticko-logické jednotky uvnitř procesoru díky tomu, že sčítání a odečítání se provádí pro čísla se znaménkem stejně jako pro čísla bez znaménka, odlišná je pouze interpretace přetečení.
- Odečítání lze převést na sečtení prvního operandu s dvojkovým doplňkem druhého.
- Dvojkový doplňkový kód je nejrozšířenějším způsobem reprezentace celých čísel se znaménkem v počítači.

  Způsob kódování

  - Na záporném čísle provedeme negaci všech bitů dvojkového zápisu čísla a k výsledku se přičte jednička.
  - Kladné číslo se nemění

Určete obraz čísel -105_{10} a 38_{10} v doplňkovém kódu.

1	0	0	1	0	1	1	1
0	0	1	0	0	1	1	0

Aditivní kód/Kód s posunutou nulou

- Využívá se zde aditivní konstanta.
- Aditivní konstanta je číslo, které se ke každému číslu přičítá.
- Pro osmibitové číslo se používá nejčastěji číslo 127 nebo 128. Je možné ale použít jakékoli číslo.
- Osmibitová čísla, která mohou reprezentovat 256 různých čísel, je možné 00000000 považovat za −127, nulu vyjádříme jako 01111111 a symbol 11111111 je 128.
- Toto číslo musí být konstantní pro celý systém čísel, který v projektu používáme.
- Používá se pro interpretaci se zápornými čísly v paměti počítače, protože nejsou nutné obvody pro testování čísla lze s číslem normálně pracovat.

Určete obraz čísel -31_{10} a 85_{10} v aditivním kódu (k=127).

0	1	1	0	0	0	0	0
1	1	0	1	0	1	0	0

Zobrazení čísla bez a se znaménkem (8bit)

- Pro zobrazení čísla bez znaménka se využívá doplňkový kód nebo aditivní kód.
- Pro zobrazení čísla se znaménkem se využívá přímý kód.

Přímý kód

- Jedná se o vyčlenění prvního bitu jako znaménkového bitu.
- Pokud například binární číslo 00000001 vyjadřuje jedničku, pak 10000001 označuje - 1.
- Tento způsob ale komplikuje algoritmy pro praktické počítání – pro sčítání a odčítání jsou potřeba odlišné algoritmy a nejprve je vždy třeba testovat znaménkový bit a podle výsledku provést sčítání nebo odčítání.
- Další nevýhodou je, že existují dvě reprezentace čísla nula – kladná nula a záporná nula. Proto byl později pro záznam záporných čísel zaveden doplňkový kód.

Určete obraz čísel -110_{10} a 108_{10} v přímém kódu.

1	1	1	0	1	1	1	0
0	1	1	0	1	1	0	0

Přetečení

- Overflow, Carry
- Přetečení je jev, který nastane, pokud výsledek aritmetické operace nelze vyjádřit v daném číselném formátu.
- přetečení se hovoří zpravidla v souvislosti s mikroprocesory a dvojkovou soustavou.
- Při vykonání aritmetické operace, při které dojde k přenosu na nejvýznamnějším bitu registru, se přenos nemůže provést do vyššího bitu. Proto se místo toho nastaví příslušný příznak přenosu (carry flag; Jedná se o jeden z bitů registru příznaků v procesoru. Díky jemu lze sčítat a odčítat příliš vysoká čísla. Indikuje, kdy byl z nejvýznamnějšího bitu generován aritmetický přenos nebo výpůjčka).
- Příkladem může být sčítání dvou osmibitových kladných celých čísel, kde součet přesáhne 255.

Který z následujících součtů celých čísel bez znaménka způsobí na osmibitové řádové mřížce přetečení (carry):

240+15	240+16	220+10	210+50
0	1	0	1

Který z následujících součtů obrazů čísel v doplňkovém kódu na osmibitové řádové mřížce způsobí přetečení (overflow)?

0xFF-0x01	0x7F+0x01	0x80-0x01	0x80+0x02
0	1	1	0

Který z následujících součtů/rozdílů čísel způsobí přetečení (overflow)? Předpokládáme, že z čísel v desítkové soustavě se nejprve vytvoří binární obrazy v doplňkovém kódu na osmibitové řádové mřížce.

120+10	115+12	-127-1	-120-30
1	0	0	1