

Tajné spisy a exemplá z Obvodov Logických

1. Konverzie numerických systémov:

a) $10 \rightarrow x$

$$(8,8)_{10} = (8 + 0,8)_{10} = (13 + 0,4)_5 = (13,4)_5$$

$8 \div 5 = 1, \text{zv. } 3$ $1 \div 5 = 0, \text{zv. } 1$	$0,8 \times 5 = 4,0$ $0 \times 5 = 0$
--	--

$$(8,8)_{10} = (8 + 0,8)_{10} = (1000 + 0,1100\dots)_2 = (1000,1100)_2$$

$8 \div 2 = 4, \text{zv. } 0$ $4 \div 2 = 2, \text{zv. } 0$ $2 \div 2 = 1, \text{zv. } 0$ $1 \div 2 = 0, \text{zv. } 1$	$0,8 \times 2 = 1,6$ $0,6 \times 2 = 1,2$ $0,2 \times 2 = 0,4$ $0,4 \times 2 = 0,8$ $0,8 \times 2 = 1,6$...
--	---

b) $2 \rightarrow 8 \rightarrow 16 \rightarrow 2\dots$

$$(70\ 167)_8$$

$$(111\ 000\ 001\ 110\ 111)_2$$

$$(0111\ 0000\ 0111\ 0111)_2$$

$$(7\ 077)_{16}$$

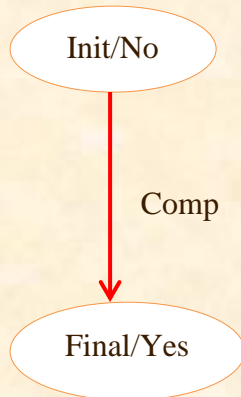
c) $x \rightarrow 10$

$$(7\ 077)_{16} = 7 \times 16^3 + 0 \times 16^2 + 7 \times 16^1 + 7 \times 16^0 = 7 \times 4096 + 7 \times 16 + 7 \times 1 = 28\ 672 + 112 + 7 = (28\ 791)_{10}$$

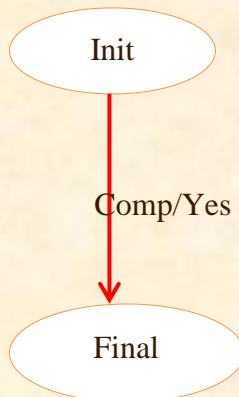
2. Rýchle tabuľky:

A = 10 = 1010	$2^0 = 1$	$2^{10} = \text{kibi}$
B = 11 = 1011	$2^1 = 2$	$2^{20} = \text{Mibi}$
C = 12 = 1100	$2^2 = 4$	$2^{30} = \text{Gibi}$
D = 13 = 1101	$2^3 = 8$	$2^{40} = \text{Tibi}$
E = 14 = 1110	$2^4 = 16$	$2^{50} = \text{Pibi}$
F = 15 = 1111	$2^5 = 32$	$2^{60} = \text{Ebi}$
	$2^6 = 64$	
	$2^7 = 128$	
	$2^8 = 256$	
	$2^9 = 512$	

3. Ako sa číta graf?



FSM je v **súčasnom** stave **Init**, generuje **súčasný** výstup **No**, a pri **súčasnom** vstupe **Comp** prejde do **budúceho** stavu **Final**.



FSM je v **súčasnom** stave **Init**, a pri **súčasnom** vstupe **Comp** generuje **súčasný** výstup **Yes**, a prejde do **budúceho** stavu **Final**.

4. Registre:

V nasledujúcich príkladoch pracujeme s predpokladom, že register je indexovaný v tomto poradí: 7-6-5-4-3-2-1-0. Index sa nazýva aj váhou bitu.

- a) nulovať bit s váhou 3: znak **&**, maska $(1111\ 0111)_2 = \mathbf{0xF7}$
- b) nastaviť bit s váhou 3: znak **|**, maska $(0000\ 1000)_2 = \mathbf{0x08}$
- c) negovať bit s váhou 3: znak **^**, maska $(0000\ 1000)_2 = \mathbf{0x08}$

5. BCD kód (pre zamestnancov IBM):

Zhustený formát:

-5836 = 0x05 0x83 0x6D, D = debet

+2593 = 0x02 0x59 0x3C, C = credit

Rozvinutý zápis:

- funguje rovnako ako zhustený, ale čísla sú písané v každom byte osobitne:

-5836 = 0x00 0x05 0x08 0x03 0x06 0x0D

!POZOR! Pri aritmetických operáciách s BCD kódom je nutné vždy urobiť korekciu výsledku:

```

    1001 (9)
  +0101 (5)
  -----
    1100 (C)
  +0110 (korekcia je vždy číslo 6)
  -----
0001 0100 (1 a 2, teda 12)

```

6. Floating point:

1 0000 0110 1 0110 000...

- prvý bit definuje znamienko (1 = -, 0 = +)

- ďalších 8 bitov definuje exponent. Preveďte toto číslo do decimálnej sústavy a pripočítajte -127

- zvyšné bity definujú significand. Toto číslo ponechajte v binárnej sústave a z 1. znak dajte desatinnú bodku (alebo čiarku)

Výsledok: -1,011 × 2⁻¹²¹ (lebo 6 - 127 = -121)

7. Hradlá:

	&		^
00	0	0	0
01	0	1	1
10	0	1	1
11	1	1	0

8. UTF:

Bits	Last Code Point	Byte 1	Byte 2	Byte 3	Byte 4
7	U+007F	0xxx xxxx			
11	U+07FF	110x xxxx	10xx xxxx		
16	U+FFFF	1110 xxxx	10xx xxxx	10xx xxxx	
21	U+1FFFF	1111 0xxx	10xx xxxx	10xx xxxx	10xx xxxx

a) **0x7C** – obsahuje iba jeden byte, a prvý bit v binárnom formáte (0111 1100) je 0. Kód sa teda nemení a odpoveďou je U+007C.

b) **0xDE 0x8C** – sa skladá z dvoch bytov. Kód najskôr prevedieme do binárnej sústavy.

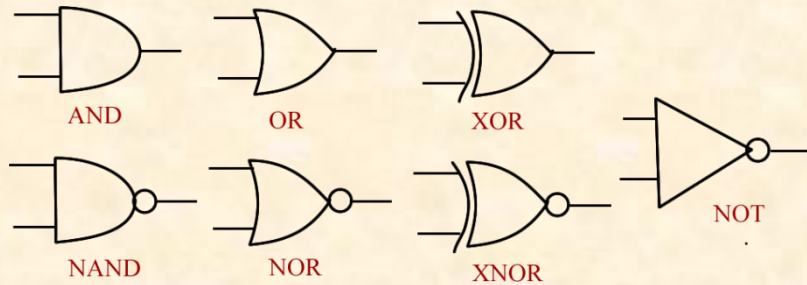
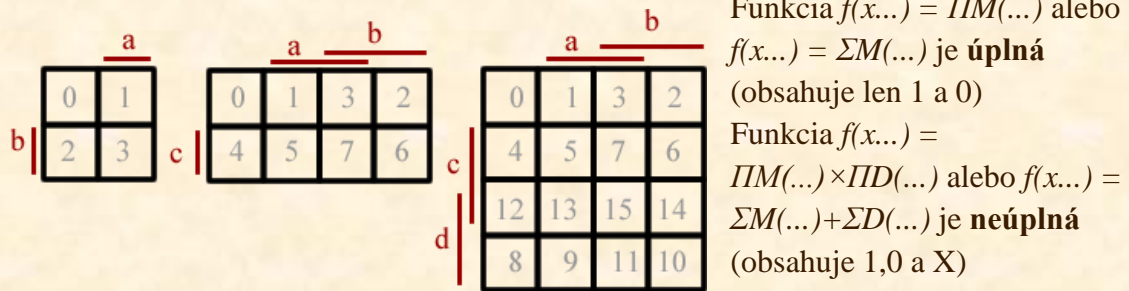
1101 1110 1000 1100 – 110 v prvom byte a 10 v druhom byte sú masky, ktoré môžeme odstrániť. Zvyšok kódu opíšeme. Oстане nám:

0000 0111 1000 1100 – a toto číslo už iba prevedieme do hexa sústavy

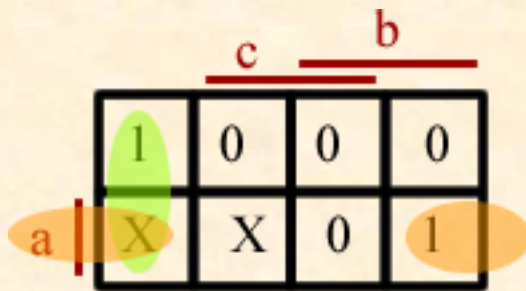
U + 078C

c) podobne, ako v prípade b) sa riešia aj príklady s 3 a 4 bytmi.

9. Karnaughove mapy a zapojenia hradiel:



V prípade funkcie Π sa vypisujú 0 (a X), v prípade Σ sa vypisujú 1 (a X). Následne zakrúžkujte všetky 1, ktoré sú v skupine po 1, 2, 4, 8 alebo 16. V prípade potreby si pomôžte aj X-kami (ktoré zastupujú ľubovoľnú hodnotu 0, alebo 1).



Na obrázku sú Karnaughove mapy s indexmi políček pre $f(b, a)$, $f(c, b, a)$ a $f(d, c, b, a)$:

Následne preved'te minimalizáciu pomocou Booleanovskej aritmetiky a nakreslite zapojenie pomocou hradiel.

Priklad: $f(a, b, c) = \Pi M(1, 2, 3, 7) \times \Pi D(4, 6)$

!NEZABUDNITE! že pri 3-och premenných môžete spárovať aj 1-ky a X ležiace po bočných okrajoch Karnaughovej mapy. V prípade 4-och premenných to platí aj pre vrchný a spodný okraj.

Výsledok: $a \times c' + b' \times c' = c'(a + b')$

Zapojíme teda a a b' pomocou OR hradla a k ich výsledku pripojíme c' pomocou AND.

10. Flags:

N – negative, nastavuje sa, keď je číslo záporné

Z – zero, nastavuje sa, keď je číslo rovné 0

V – overflow, nastavuje sa, keď dôjde k pretečeniu

C – carry, nastavuje sa, ak sa prenáša bit do vyššieho rádu

Príklady na flags:

0111
+1000
0000, N = 0, Z = 1, V = 0, C = 0

0110
+0001
0111, N = 0, Z = 0, V = 0, C = 0

0100
+0101

1001, N = 1, Z = 0, V = 1, C = 0 (prvý bit je vyhradený pre znamienko a 1 znamená -). Avšak súčet dvoch kladných čísel nemôže byť záporný. Došlo teda k pretečeniu a nastavuje sa flag V).

0100
+1101
10001, N = 0, Z = 0, V = 0, C = 1

0011
+1100
1111, N = 1, Z = 0, V = 0, C = 0

1100
+1011

10111, N = 0, Z = 0, V = 1, C = 1 (Okrem prenosu bytu do vyššieho rádu – carry – dochádza aj k pretečeniu; znamienka oboch sčítancov sú – (1), avšak výsledok je kladný (0)).

11. Typické posunutie:

Typické posunutie je jeden zo šialených spôsobov, ako zapisovať a čítať záporné čísla. Využíva sa napr. aj pri floatingu, kde sa od exponentu odpočítava 127.

Typické posunutie pre 4 bity je -7.

100 + 1 šialených spôsobov, ako zapisovať a čítať záporné číslo:

Príklad: 1100

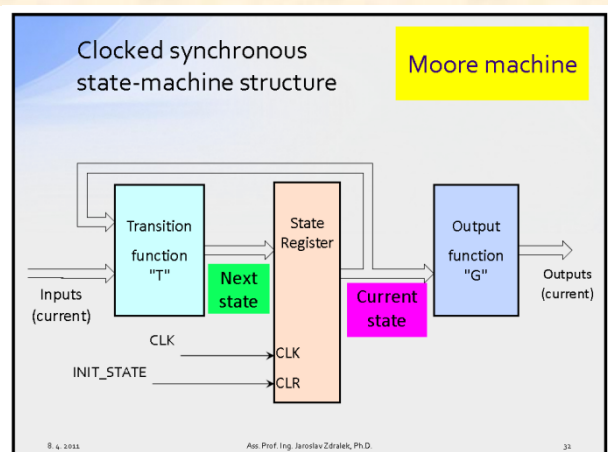
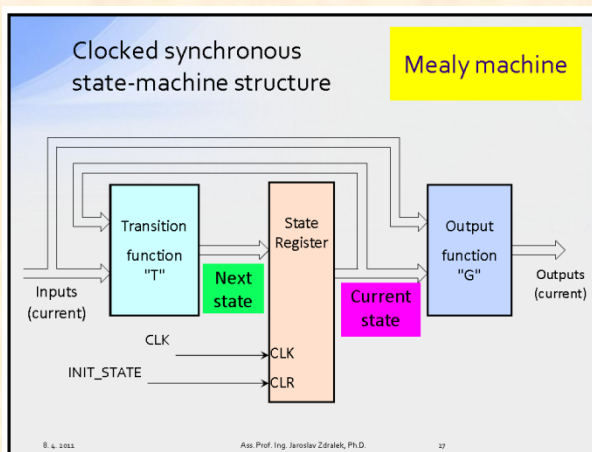
normálne: +12

jednotkový doplnok: not(1100) = 0011, čo je 3, takže výsledok je -3

dvojkový doplnok: not(1100) – 0001 = 0011 – 0001 = 0010, takže výsledok je -2

typické posunutie: 12-7 = 5

12. Mealy a Moore:



13. Veľký a malý endian:

Veľký a malý endian definujú, v akom poradí sa elementy ukladajú do pamäte. Vo veľkom endiane je MSB ukladaný na nižších adresách, v malom endiane je MSB ukladaný na vyšších adresách. Elementy môžu byť ukladané v nibbloch, bytoch alebo slovách (16 bitov, 32 bitov...).

Príklad: Zapište nasledujúci sled bytov do pamäte v 16-bitových slovách podľa princípu malý endian:

34 87 AD 1C

Výsledok:

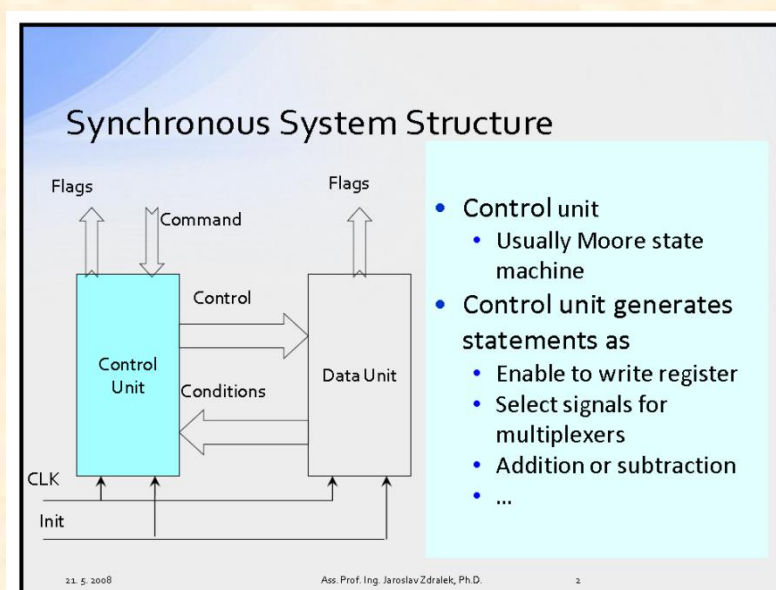
$i + 3$: 00 00

$i + 2$: 00 00

$i + 1$: 34 87

i : AD 1C

14. Synchronný číslicový systém:



Pravdy Obvodov Logických

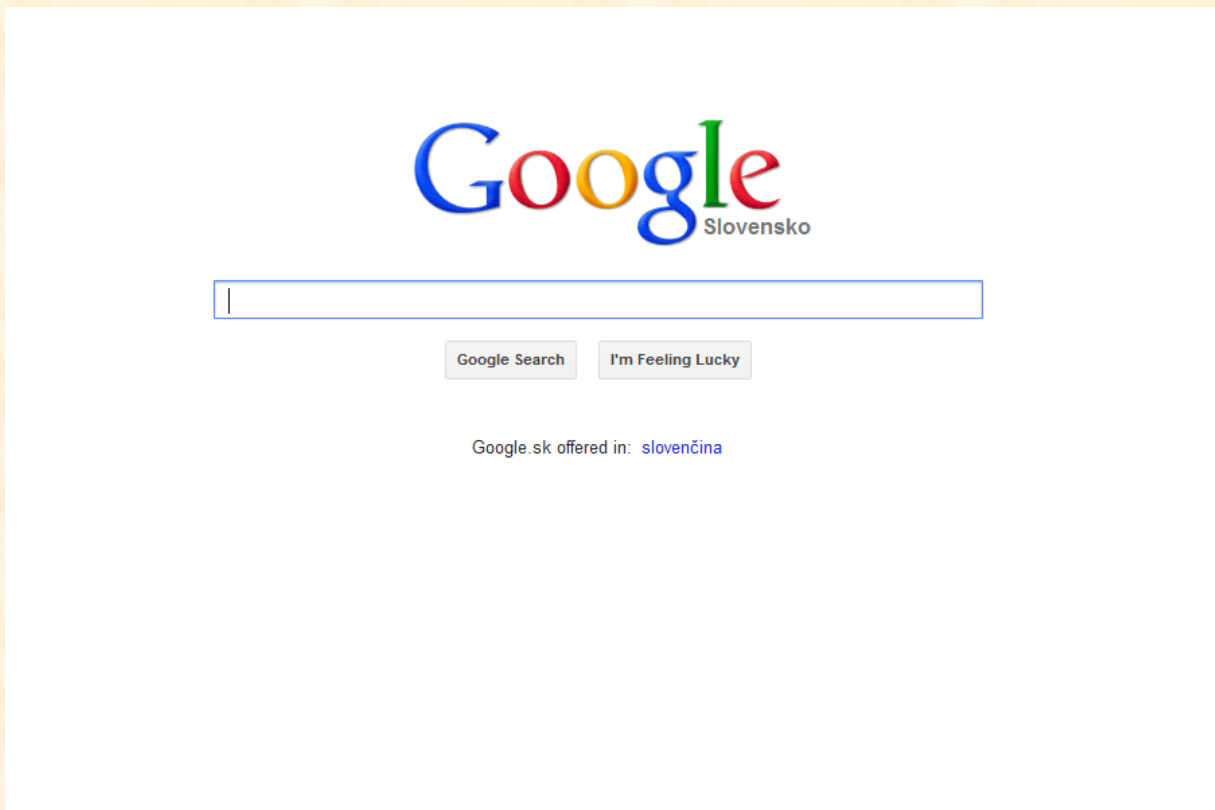
Pravdy Obvodov Logických sú dogmatické. Aj najmenšia pochybnosť o nich je považovaná za kacírstvo. Všetci kacíri budú upálení prvú stredu v mesiaci, presne o 12:00 za zvuku sirény.

Pravdy sú usporiadané abecedne, podľa kľúčového slova:

• Binárna sčítačka s kanálom rýchleho prenosu zmenší oneskorenie súčtu.
• Binárna sčítačka využíva rovnaké zapojenia pre každý bit.
• Booleanovská funkcia (ďalej len Bool f), úplná aj neúplná, je zobrazenie definičného oboru do oboru hodnôt.
• Bool f (úplnú aj neúplnú) jednoznačne definuje Karnaughova mapa.
• Bool f (úplnú aj neúplnú) jednoznačne definuje pravdivostná tabuľka.
• Demultiplexer prevádza jeden vstup na viac výstupov podľa výberového signálu.
• G – daný bit generuje výstup.
• Kóder je zapojenie, ktoré mení vstupný kód na iný výstupný kód.
• Λ – definuje výstupnú abecedu
• Multiplexer vyberá jeden zo vstupov na svoj výstup na základe hodnoty výberových signálov.
• Neúplná Bool f je zobrazenie $\{0,1\}^n$ do $\{0,1,X\}$
• P – bit na vstupe je rovnaký ako bit na výstupe
• S – definuje množstvo stavov
• S_0 – počiatočný stav, patrí do množiny S
• Σ – definuje vstupnú abecedu
• T – to isté, čo P , iba po anglicky (transfer = prenos)
• Úplná Bool f je zobrazenie $\{0,1\}^n$ do $\{0,1\}$

Google

V prípade, že sa Vaša otázka nenachádza v tomto dokumente, skúste použiť Google:



~ ,	! 1	@ 2	# 3	\$ 4	% 5	^ 6	& 7	* 8	(9) 0	- _	+ =	← Backspace
Tab ⇄	Q	W	E	R	T	Y	U	I	O	P	{ [}]	 \ /
Caps Lock ⇧	A	S	D	F	G	H	J	K	L	:	" '	↵ Enter	
Shift ⇧	Z	X	C	V	B	N	M	< ,	> .	? /	⇧ Shift		
Ctrl	Win Key	Alt								Alt	Win Key	Menu	Ctrl

Obsah

Strana 1:

1. Konverzie numerických systémov
2. Rýchle tabuľky

Strana 2:

3. Ako čítať graf?
4. Registre
5. BCD kód

Strana 3:

6. Floating point
7. Hradlá
8. UTF

Strana 4:

9. Karnaughove mapy a zapojenia hradiel
10. Flags

Strana 5:

11. Typické posunutie
12. Moore a Mealey

Strana 6:

- Veľký a malý endian
- Synchronný číslicový systém

Strana 7:

- Pravdy Obvodov Logických

Strana 8:

- Google

Strana 9:

- Obsah

Strana 10:

- Tento dokument...

Tento dokument...

Tento dokument nikdy neexistoval.

Kňaz z Warcraft III varuje: „Možné vedľejšie účinky čtení tohto dokumentu jsou: sucho v ústech, nevoľnosť, zvracenie, zadržovanie vody, bolestivé svēdění konečnku, halucinace, slabomyslnost, psychóza, bezvēdomí, smrt a zápach z úst.“

Tento dokument dokáže deliť nulou.

Po zničení tohto dokumentu vzniká čierna diera.

Plť po česky je vor.

Tento dokument porušuje všetky ľudské práva.

Tento dokument nedodržiava Maastrichtské kritériá a jeho štátny deficit je väčší ako 3%.

Tento dokument porušuje všetky ustanovenia z dohôd SOPA, PIPA, CISPA atď.

Světelná lžýce!

Podľa Róberta Fica, tento dokument neobsahuje dostatok červenej farby.

David Rath by za tento dokument dal aj dve krabice vína.

Chuch Norris tento dokument nikdy nedočítal dokonca.

Dokážem žiť aj bez pozerania Red Dwarf, či Game of Thrones.

Tento dokument NIE JE 8.-mi horcrux lorda Voldemorta.

Tento dokument JE 9.-ty horcrux lorda Voldemorta.

Človek, ktorý vám zlomí nos, nie je váš priateľ.

Univerzita v Plzni a Univerzita v Trenčine by tento dokument pokladala za najlepšiu diplomovú prácu za posledných 100 rokov.

Tento dokument zachráni kapitána Shepharda.

Tento dokument dokáže prestať hrať Diablo III.

42 je 101010.

Tento dokument sa pokladá za siedmu farbu v MTG.

Tomuto dokumentu sa Ridiculously Photogenic Guy nepáčil.

Tento dokument predpovedal: „Cake will be a lie!“

Stavím sa, že z tohto testu budeš mať 50 bodov.

Tento dokument vie, ako skončí The Dark Knight Rises.

VSP tohto dokumentu je 101,1. A stúpa.

Tento dokument má dievča. Skutočné.

Tento dokument pozná skutočný vek Heleny Vondráčkovej

Tento dokument je venovaný všetkým obetiam, ktoré sa stratili v Lese.