

1. Maturitná Otázka

A) Základy číslicovej techniky (Hardvérová časť)

Boolová Algebra

- Zahŕňa logické funkcie ,ktoré sa využívajú v informatike a elektrotechnike.
- Na základe týchto funkcií sa vytvárajú logické hradlá ,z ktorých sú zostavené tranzistory, integrované obvody a procesory.
- Boolová algebra využíva binárnu sústavu.
- Premenné v boolovej algebre teda môžu nadobúdať hodnoty 0 alebo 1.
- Vstupné hodnoty premenných a výslednú hodnotu funkcie zapisujeme do **pravdivostnej tabuľky**.
- Vstupné hodnoty označujeme ako nezávislé premenné, výstupnú hodnotu označujeme ako závislo-premennú.

Negácia

– Je najjednoduchšou boolovskou operáciou, pri negácii nadobúda premenná opačnú hodnotu.

– Pravdivostná tabuľka:

x	f(x)
0	1
1	0

– Negácia sa označuje rovnou vodorovnou čiarou nad premennou.

$$\overline{X}$$

Logický súčet

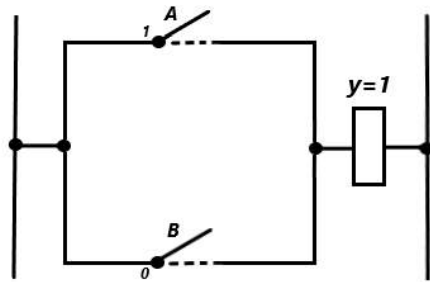
– Je operácia disjunkcie (zjednotenie).

– Býva tiež označovaná

$$y = \overbrace{A + B}^{\text{Disjunkcia}}$$

↑
↙
 Disjunkcia Premenné

– Schematicky ju môžeme znázorniť takto:



A	B	y
0	0	0
1	0	1
0	1	1
1	1	1

– Pre operáciu logického súčtu je charakteristická spojka alebo.

A OR B

A + B

A alebo B

A ∨ B

Logický súčet

– Je operácia konjunkcie alebo prieniku, označuje sa aj takýmto znakom \wedge .

– Pre logický súčin sú typické spojky a, aj, i.

A ∧ B

A & B

A · B

A AND B

– Výsledná hodnota funkcie je jedna len vtedy, ak všetky nezávislé premenné majú hodnotu 1.



A	B	Y
0	0	0

1	0	0
0	1	0
1	1	1

Ďalšie Boolovské funkcie

NAND (Negovaný súčin)

– Funkcia NAND má hodnotu 1, ak aspoň 1 nezávislá premenná má hodnotu 0.

A	B	Y
0	0	1
1	0	1
0	1	1
1	1	0

NOR (Negovaný súčet)

– Funkcia NOR má hodnotu 1, ak všetky vstupy majú hodnotu 0.

A	B	Y
0	0	1
1	0	0
0	1	0
1	1	0

XOR (Exkluzívny súčet)

– Funkcia XOR má hodnotu 1, ak vstupy majú rozdielne hodnoty.

A	B	Y
0	0	0
1	0	1
0	1	1
1	1	0

Karnaughové mapy

- Slúži na zjednodušený zápis funkcie.
- Aby bola mapa karnaughovou mapou musí spĺňať tieto podmienky:
 1. Karnaughová mapa má toľko políčok koľko má pravdivostná tabuľka riadkov (2^n). n =počet vstupov
 2. V Karnaughovej mape musí existovať 1 bunka, v ktorej majú všetky nezávislé vstupy hodnotu 0.
 3. V Karnaughovej mape musí existovať 1 bunka, v ktorej majú všetky nezávislé vstupy hodnotu 1.
 4. Pre každý nezávislý vstup musí platiť, že pre polovicu buniek Karnaughovej mapy má vstup hodnotu 0 a pre polovicu buniek Karnaughovej mapy má vstup hodnotu 1.

2 spôsoby zápisu Karnaughovej mapy

Aritmetický
zápis

	A	B	0	1
0			0	0
1			0	1

Zápis pomocou
svoriek

	<u>B</u>	
	0	0
A	0	1

- Do Karnaughovej mapy zapisujeme hodnotu funkcie, nie hodnotu vstupov.

Preklápacie obvody

- Je to obvod, ktorý sa dokáže dostať zo stavu 0 do stavu 1 a naopak.

1. Monostabilný

– Má stabilný iba jeden stav, ak mu dáme povel (impulz), preklopí sa do opačného stavu a po určitom čase sa vráti do svojho stabilného stavu.

– Napríklad: svetlá vchodoch bytovke sú v stabilnej 0, po stlačení vypínača sa dajú do stabilnej 1 a po určitom čase sa vrátia do stabilnej 0 (zhasnú) a infrasnímač.

2. Bistabilný

– Má stabilné oba stavy, ak je v nule a dostane povel, preklopí sa do jednotky a na ďalší povel sa preklopí do nuly.

– Napríklad: mikrospínač na televízory.

3. Astabilný

– Tento klopný obvod nemá stabilný ani jeden stav a preklápa sa z 0 do 1 alebo z 1 do 0.

– Napríklad: blikač na bicykli.

V niektorých aplikáciách je potreba zariadiť, aby klopný obvod po určitý čas nereagoval na vstup. Na tento účel boli navrhnuté tzv. Synchronne klopné obvody vybavené špeciálnym vstupom, označovaným ako hodinový vstup. Na tento vstup je privedený tzv. Hodinový signál, ktorý určuje, či bude klopný obvod na ostatné vstupy reagovať, alebo ich ignorovať.

Synchronne - s hodinovým vstupom

Asynchronne - bez hodinového vstupu

B) Algoritmizácia (Softvérová časť)

Algoritmizácia

Algoritmus

- *Algoritmus je postupnosť príkazov (postup), kde zo zadaných vstupných údajov po konečnom počte krokov a v konečnom čase dostaneme správy (očakávaný výsledok).*

Vlastnosti algoritmu:

- 1. Elementárnosť** – Algoritmus musí byť zrozumiteľný pre realizátora.
- 2. Determinovanosť** – V každom kroku vykonávania algoritmu musí byť jasné, aký krok bude nasledovať alebo či už algoritmus skončil.
- 3. Rezultatívnosť** – Algoritmus dáva pre rovnaké vstupné údaje vždy rovnaký výsledok.
- 4. Hromadnosť** – Algoritmus je aplikovateľný na celú skupinu prípustných vstupných údajov.
- 5. Konečnosť** – Algoritmus musí skončiť vždy v konečnom čase po vykonaní konečného počtu krokov.
- 6. Efektívnosť** – Postup je vytvorený v čo najkratšom čase s využitím čo najmenšieho počtu prostriedkov.