BOOLOVÁ ALGEBRA

Slúži na matematický opis zákonov a pravidiel výrokovej logiky, ktorá rieši vzťahy medzi pravdivými a nepravdivými výrokmi. Pravdivému výroku prideľujeme logickú hodnotu 1 a nepravidelnému výroku logickú hodnotu 0. Nositeľom elementárnej informácie o pravdivosti alebo nepravdivosti výroku je logická premenná, ktorá môže nadobúdať dve hodnoty 0 a 1.

ZÁKLADNÉ OPERÁCIE

1, logický súčin
2, logický súčet
3, logická negácia

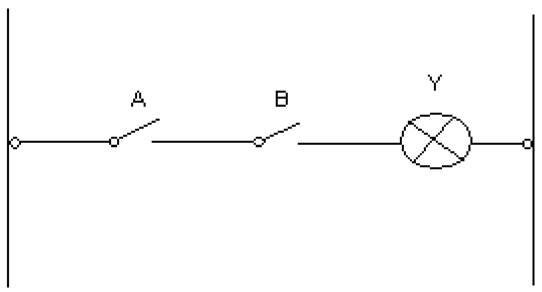
LOGICKÝ SÚČIN

Majme jednoduché boolovské premenné A,B,Y
AND: Y=A.B
Logický súčin **AND** je charakterizovaný tím, že funkčná hodnota Y nadobúda jedničky len vtedy, ak obidve premenné A,B sú jedničky.

PRAVDIVOSTNÁ TABUĽKA :

A	B	Y=A . B
0	0	0
0	1	0
1	0	0
1	1	1

KONTAKTOVÁ REALIZÁCIA :



Logická hodnota 1 predstavuje zopnutý spínač a logická hodnota 0 predstavuje rozopnutý spínač. Obvodom môže tiecť prúd iba ak sú zopnuté spínače.

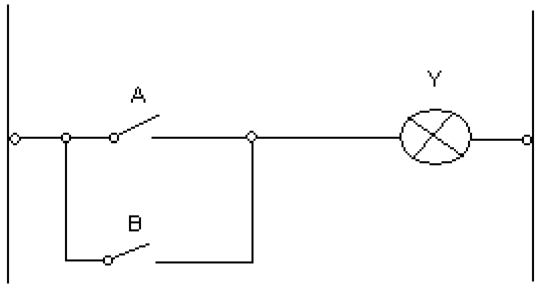
LOGICKÝ SÚČET

Majme jednoduché boolovské premenné A,B,Y
OR : Y = A+B
Logický súčet **OR** je charakterizovaný tím, že funkčná hodnota Y nadobúda hodnotu jedničky práve vtedy ak, aspoň jedna z premenných A, B je jednička.

PRAVDIVOSTNÁ TABUĽKA :

A	B	Y=A+B
0	0	0
0	1	1
1	0	1
1	1	1

KONTAKTOVÁ REALIZÁCIA :



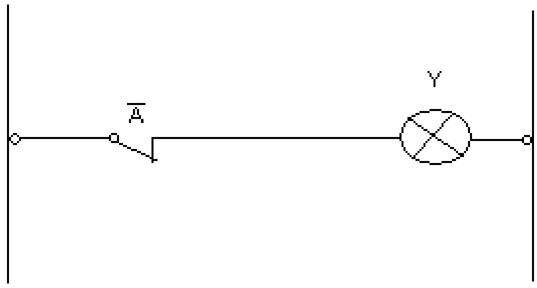
LOGICKÁ NEGÁCIA

Majme jednoduché boolovské premenné A,Y
NOT : Y = $\overline{A}$
Logická negácia **NOT** je charakterizovaná tím, že funkčná hodnota Y nadobúda hodnotu jedničky práve vtedy, ak premenná A je nula.

PRAVDIVOSTNÁ TABUĽKA :

A	Y=$\overline{A}$
0	1
1	0

KONTAKTOVÁ REALIZÁCIA :



Príklad :
Zostrojte pravdivostnú tabuľku pre funkciu $\mathbf{Y = A.(\overline{B}+C)}$

Poznámka : počet riadkov v PT zodpovedá vzorcu : $\mathbf{2^n}$, kde **n** je počet vstupných premenných.

A	**B**	**C**	$\overline{\mathbf{B}}$	**Y**
0	0	0	1	0
0	0	1	1	0
0	1	0	0	0
0	1	1	0	0
1	0	0	1	1
1	0	1	1	1
1	1	0	0	0
1	1	1	0	1

de Morganov zákon :

$\mathbf{\overline{A + B + C + ...} = \overline{A} . \overline{B} . \overline{C} ......}$

$\mathbf{\overline{A . B . C .....} = \overline{A} + \overline{B} + \overline{C} + ...}$

Príklad:

Dokážte platnosť de Morganových pravidiel pomocou PT.

$Y_1 = \overline{A + B}$

$Y_2 = \overline{A} . \overline{B}$
$Y_1 = Y_2$

FORMY ZÁPISU LOGICKÝCH FUNKCIÍ

1) ALGEBRAICKÝ VÝRAZ
2) PRAVDIVOSTNÁ TABUĽKA
3) KARNAUGHOVÁ MAPA

KARNAUGHOVÁ MAPA

Skladá sa z určitého počtu štvorcov (políčok), pričom každej kombinácií nezávislých vstupných premenných v KM zodpovedá jeden štvorec do ktorého vypisujeme zodpovedajúcu hodnotu výstupnej funkcie. Počet štvorcov v KM sa musí rovnať počtu riadkov v PT.
Budeme sa držať najčastejšieho spôsobu značenia máp, podľa ktorého riadky alebo stĺpce v ktorých je príslušná premenná rovná **jednotke** , označíme vedľa mapy zvislou alebo vodorovnou čiarou, ku ktorej pripíšeme meno príslušnej logickej premennej. V riadkoch alebo v stĺpcoch, ktoré nie sú označené, je príslušná premenná rovná **nule** .

a) KM pre funkciu jednej vstupnej premennej Y = f(A)

$Y = \overline{A}$

PT:

A	Y
0	1
1	0

KM:

	A
1	0

b) KM pre funkciu dvoch vstupných premenných Y = f(A, B)

$Y = A \,.\, B + \overline{A} \,.\, \overline{B}$

PT:

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

KM : A

	1	**1**
B	**0**	**0**

Príklad :

Zostrojte KM pre funkciu troch vstupných premenných Y = f(A,B,C)
Y = (A+B).C

VÝPIS FUNKCIE Z PRAVDIVOSTNEJ TABUĽKY

2 Formy :

1) **UNDF** – úplná normálna disjuktívna forma
Súčet základných súčinov.
2) **UNKF** – úplná normálna konjuktívna forma
Súčin základných súčtov.

1) UNDF : $Y = \sum_{i=0}^{2^n-1} y_i m_i(x) = y_0 m_0(x) + y_1 m_1(x) + ...y_{2^n-1} m_{2^n-1}(x)$

Minterm : mi(x) = $x_1 . x_2 . x_3 ...... x_n$

y_i - hodnota funkcie v í – tom riadku PT
m_i - je logický súčin všetkých vstupných premenných v priamom tvare, ak premenná má hodnotu 1 a v inverznom tvare ak premenná má hodnotu 0.

Príklad : Zostavte PT a jej výpis pomocou formy UNDF...
Y = A . B

S	A	B	Y	mi(A,B)
0	0	0	0	A´. B´
1	0	1	0	A´. B
2	1	0	0	A . B´
3	1	1	1	A. B

S – poradie riadkov...S = 2^{n-1}
´ - negácia namiesto čiary ─ pre lepšiu prehľadnosť v tabuľke

$$Y = y_0 m_0(A,B) + y_1 m_1(A,B) + y_2 m_2(A,B) + y_3 m_3(A,B)$$

$$Y = 0.A'.B' + 0.A'.B + 0.A.B' + 1.A.B = A.B$$

2) UNKF :

$$Y = \prod_{i=0}^{2^n-1} (y_i + M_i(x)) = (y_0 + M_0(x)).(y_1 + M_1(x)) ......(y_{2^n-1} + M_{2^n-1}(x))$$

Maxterm : $Mi(x) = x_1 + x_2 + x_3 + ...x_n$

Mi(x) – logický súčet všetkých vstupných premenných v priamom tvare, ak premenná má hodnotu 0 a v inverznom tvare ak premenná má hodnotu 1.

Príklad : Zostavte PT a jej výpis pomocou formy UNKF...
Y = A . B

S	A	B	Y	Mi(A,B)
0	0	0	0	A+B
1	0	1	0	A+B′
2	1	0	0	A′+B
3	1	1	1	A′+B′

$$Y = (y_0 + M_0(A,B)).(y_1 + M_1(A,B)).(y_2 + M_2(A,B)).(y_3 + M_3(A,B))$$
$$Y = (0 + A + B).(0 + A + B').(0 + A' + B).(1 + A' + B')$$
$$Y = (A + B).(A + B').(A' + B).1$$

Príklad:

Zostrojte výpis UNDF a UNKF funkcie z PT a ak je daná PT:

S	A	B	C	Y
0	0	0	0	0
1	0	0	1	0
2	0	1	0	0
3	0	1	1	1
4	1	0	0	1
5	1	0	1	0
6	1	1	0	0
7	1	1	1	1

MINIMALIZÁCIA LOGICKEJ FUNKCIE

Danú LF môžeme zapísať v rôznych tvaroch. Všetky tvary sú matematický rovnocenné, pretože predstavujú rovnakú závislosť aj keď sú tvarovo dosť odlišné. Nie sú však rovnocenné z hľadiska technického a ekonomického. Pre technickú realizáciu je nutné vždy funkciu upraviť do najjednoduchšieho tvaru – **minimalizovať ju**. Minimalizáciou funkcií dosiahneme, že pri jej realizácií budeme potrebovať najmenší počet logických prvkov. Tím sa logický obvod stane jednoduchším a samozrejme aj lacnejším z ekonomického hľadiska.

Pre minimalizáciu existujú rôzne metódy :
Napr.: 1) Algebraická – logickú funkciu zjednodušíme pomocou zákonov a pravidiel Boolovej algebry :

A+B = B+A	A.B=B.A
A+(B+C)=(A+B)+C	A.(B.C)=(A.B).C
A+B.C=(A+B).(A+C)	A.(B+C)=A.B+A.C
A´´=A	A.A=A
A+A=A	A.A´=0
A+A´=1	A.1=A
A+1=1	A.0=0
A+0=A	A.(A´+B)=A.B
A.A´.B=A+B	(A.B)´=A´+B´
(A+B)´=A´.B´	(A+B).(A+B´)=A
A.B+B´.A=A	

2) **Grafická – v KM**

Systém v KM je založený na skutočnosti že dva členy výrazu ktoré obsahujú tie isté premenné a líšia sa iba v jednej premennej - sú redukovateľné ,tj. Možno ich zjednodušiť tak, že vynecháme premennú v ktorej sa líšia. Dva členy, ktoré sa líšia iba v jednej premennej nazývajú sa susediacimi členmi v KM a nachádzajú sa vedľa seba.
Podstata minimalizácie v KM spočíva v zoskupovaní susediacich jedničiek do konfigurácií pričom platí :
1) jedničky musia byť susedné
2) počet susediacich jedničiek v danej konfigurácií je daný vzorcom 2^n .
3) každá jednička sa môže vyskytovať aj vo viacerých konfiguráciach.
4) v mape musíme pokryť konfiguráciami všetky jedničky.

Príklady : Minimalizujte v KM :

1.)

C

B

	0	1	1	0
A	0	0	0	0

UNDF: $Y = A'.B.C' + A'.B.C$

$Y = A'B.\underbrace{(C'+C)}_{1}$

$Y = A'.B$

2.)

B C

	0	0	0	1
A	0	0	0	1

UNDF: $Y = A'.B'.C + A.B'.C$

$Y = B'.C.(A'+A)$

$Y = B'.C$

3.)

C

B

	0	1	1	0
A	0	0	1	0

UNDF : $Y_1 = A'.B$

$Y_2 = B.C$

$Y = A'.B + B.C$

4.)

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

Susediace políčka sú aj políčka na okrajoch mapy. Predstavme si, že mapu „zrolujeme“ tak, že ľavý okraj bude susediaci s pravým a súčasne dolný s horným.

UNDF: Y = B´.C´

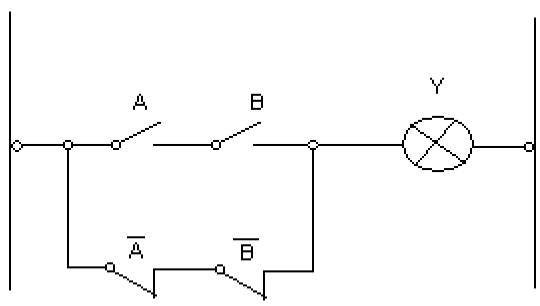
SCHÉMA LOGICKÉHO OBVODU

REALIZÁCIA :

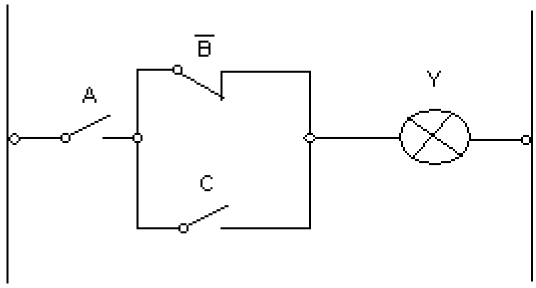
1.) kontaktovými členmi (kontakty, relé, stykače)
2.) logickými členmi (AND, OR, NOT, NAND, NOR)
3.) programovými automatmi (Simatic, LOGO! 230RC)

KONTAKTOVÁ REALIZÁCIA LOGICKEJ FUNCIE

Príklad 1.) : Kontaktovo zrealizujte logickú funkciu : Y = (A.B) + (A´.B´)
Riešenie :

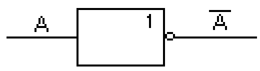
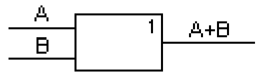
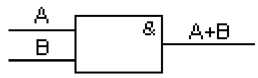


Príklad 2.) : Kontaktovo zrealizujte logickú funkciu Y = A.(B´+C)
Riešenie :



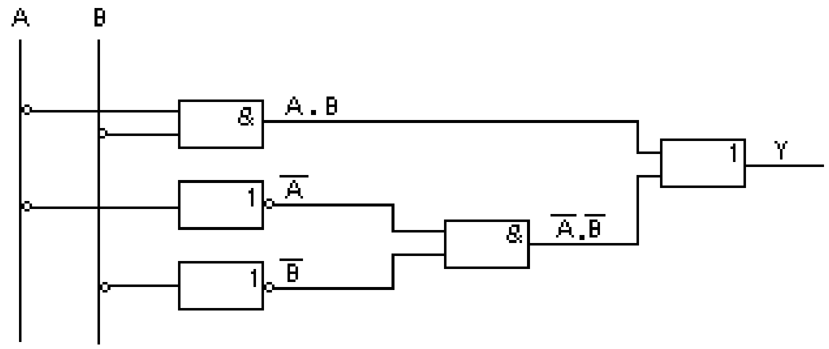
REALIZÁCIA LOGICKEJ FUNKCIE ČLENMI AND, OR, NOT

Logický člen je elementárny číslicový systém, ktorý realizuje niektorú boolovskú funkciu (operaciu) nad vstupnými premennými a jej výsledok poskytuje na svojom výstupe.

Názov logickej funkcie	**Algebraické vyjadrenie**	**Schematická značka**
Negácia **(NOT)**	**Y = A´**	
Súčet **(OR)**	**Y = A+B**	
Súčin **(AND)**	**Y= A.B**	

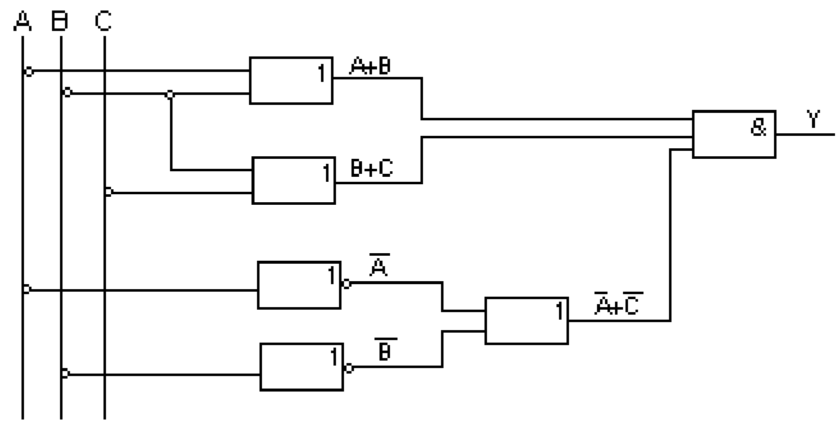
Príklad 1.) Zrealizujte pomocou logických členov funkciu : Y=A.B+A´.B´

Riešenie:



Príklad 2.) Zrealizujte pomocou logických členov funkciu : Y=(A+B).(B+C).(A´+C´)

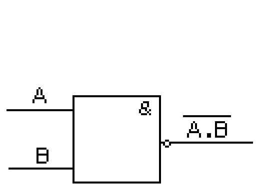
Riešenie:

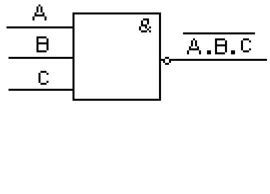


REALIZÁCIA LOGICKEJ FUNKCIE ČLENMI NAND

Logická funkcia : Y = (A.B)´
(Schefferová funkcia) Y = (A.B.C)´
Y = (A.B.C.D)´

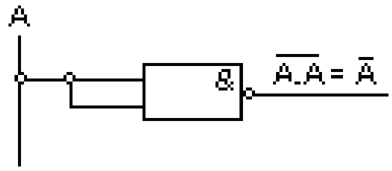
Schematická značka :





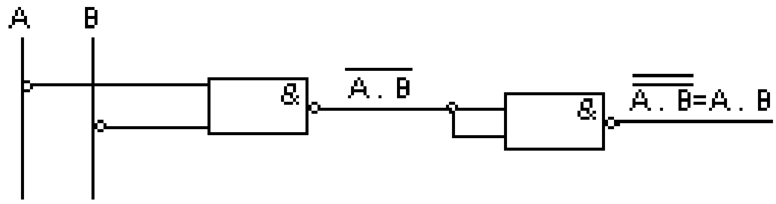
Príklad 1.) Zrealizujte pomocou hradla NAND funkciu Y = A´

Riešenie :



Príklad 2.) Zrealizujte pomocou hradla NAND funkciu Y = A.B

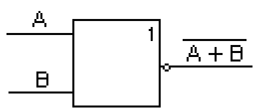
Riešenie :

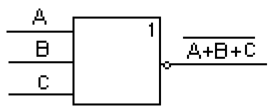


REALIZÁCIA LOGICKEJ FUNKCIE ČLENMI NOR

Logická funkcia : Y = (A+B)´
(Pierceová funkcia) Y = (A+B+C)´

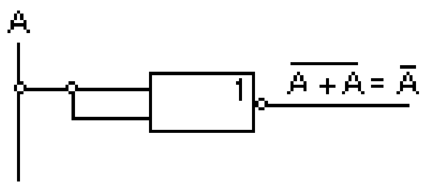
Schematická značka :





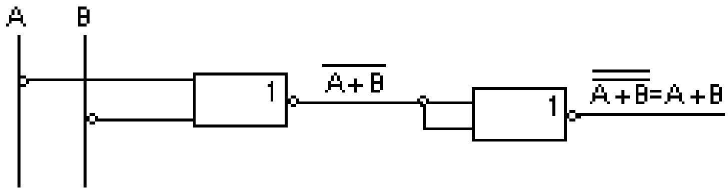
Príklad 1.) Zrealizujte pomocou hradla NOR funkciu Y = A´

Riešenie:



Príklad 2.) Zrealizujte pomocou hradla NOR funkciu Y = A + B

Riešenie:



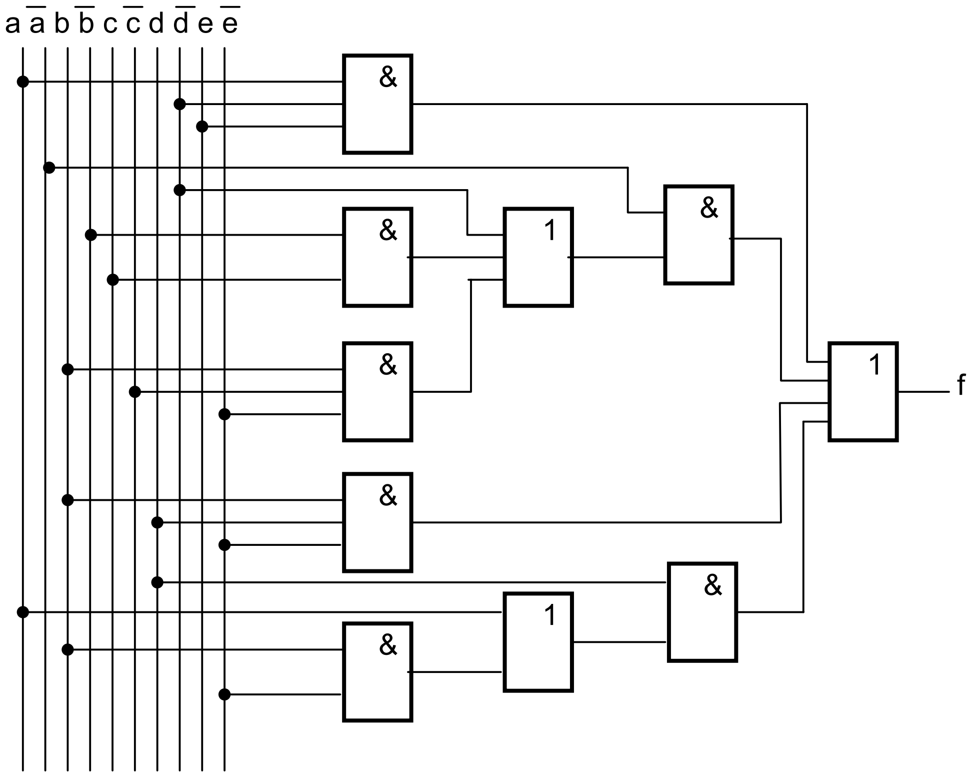
xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx

$$Y = A \cdot \overline{B} + \overline{A} \cdot B$$

Príklad: Upravte funkciu:
Riešenie:

$$Y = \overline{\overline{A \cdot \overline{B} + \overline{A} \cdot B}} = \overline{\overline{A \cdot \overline{B}} \cdot \overline{\overline{A} \cdot B}} \; \overline{\left(\overline{A} + B\right) \cdot \left(A + \overline{B}\right)} = \overline{A \cdot B + \overline{A} \cdot \overline{B}}$$

$a\;\overline{a}\;b\;\overline{b}\;c\;\overline{c}\;d\;\overline{d}\;e\;\overline{e}$



Existuje symbol $\bar{a}$

Tabuľka pre B-funkciu s dvoma nezávisle premennými

algebraické vyjadrenie	$y = a \vee b$	$y = a \,.\, b$	$y = \bar{a}$
slovné vyjadrenie	logický súčet	logický súčin	a non (negácia)
pravdivostná tabuľka	a b y 0 0 0 0 1 1 1 0 1 1 1 1	a b y 0 0 0 0 1 0 1 0 0 1 1 1	a y 0 1 1 0

1. $a \vee \bar{a} = 1$ zákon vylúčenia tretieho
   $a \,.\, \bar{a} = 0$
2. $a \vee b = \bar{a}.b$ de Morganove pravidlá
   $a \,.\, b = \bar{a} \vee b$
   $\bar{a} = a$ pravidlo dvojnásobnej negácie