evropský sociální fond v ČR
EVROPSKÁ UNIE
MINISTERSTVO ŠKOLSTVÍ, MLÁDEŽE A TĚLOVÝCHOVY
OP Vzdělávání pro konkurenceschopnost
INVESTICE DO ROZVOJE VZDĚLÁVÁNÍ

Aplikace kybernetiky ve strojírenství

9. přednáška

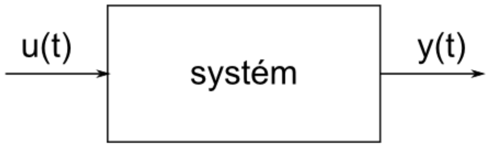
Doc. Ing. Eduard Janeček, CSc.

Ing. Jan Jakl

Podpořeno v rámci projektu CZ.1.07/2.2.00/15.0383
Inovace studijního oboru Dopravní a manipulační technika
s ohledem na potřeby trhu práce

Kybernetické systémy s dvoustavovými veličinami

Systémy logického typu



U systémů logického typu nabývají vstup i výstup systému pouze dvou hodnot.

Možné hodnoty dvoustavových veličin:

- pravda, 1, ano, přítomen, topí, spínač sepnut
- nepravda, 0, ne, nepřítomen, netopí, spínač rozepnut

Logická proměnná – proměnná, která může nabývat pouze jedné ze dvou hodnot.

Ve vztahu ke stavu systému vyjadřuje jeho mezní stavy, nebo hodnoty veličin vůči dané mezi.

Např. logická proměnná A vyjadřuje stav čerpadla a logická proměnná B vyjadřuje hodnoty otáček čerpadla vůči mezi 3000 ot/min.

čerpadlo	A
Vypnuto	0
Zapnuto	1

otáčky	B
≤ 3000 ot/min	0
> 3000 ot/min	1

Logická funkce - je funkcí logické proměnné/proměnných.

Funkce logického typu lze rozdělit na

- kombinační - výstup systému je funkcí pouze vstupu systému y(t) = f(u(t))
    - synchronní: $t \in \langle t_0, t_0 + \Delta t, t_0 + 2\Delta t, \ldots \rangle$
    - asynchronní: $t \in \langle t_0, t_1, t_2, \ldots \rangle$
- sekvenční - výstup systému je závislý na vstupu systému a na současném stavu systému y(t) = f(u(t), x(t))

Způsoby vyjádření logických funkcí

- slovní
- analytické
- pravdivostní tabulkou
- mapou (např. Karnaughovou)
- blokovým schématem

Získám zápočet, jestliže vypracuji semestrální práci nebo úspěšně napíšu zápočtový test.

y u_1 u_2

$\boxed{y = u_1 \vee u_2}$

i	**u_1**	**u_2**	**y**
0	0	0	0
1	0	1	1
2	1	0	1
3	1	1	1

Logické funkce jedné proměnné y = f(u)

i	u	y_1	y_2	y_3	y_4
0	0	0	1	0	1
1	1	0	0	1	1

Pro jednu logickou proměnnou lze definovat 4 logické funkce (2^{2^n}):

y_1 – falsum, nabývá hodnoty nula pro libovolnou hodnotu proměnné u,

y_2 – negace, nabývá opačné hodnoty než proměnná u, označuje se $y = \overline{u}$,

y_3 – aserce (opakování), nabývá stejné hodnoty jako proměnná u,

y_4 – verum, nabývá hodnoty jedna pro libovolnou hodnotu proměnné u.

S přibývajícím počtem vstupních proměnných roste počet možných kombinací jejich hodnot a také počet možných logických funkcí.

Logické funkce dvou proměnných $y = f(u_1, u_2)$

i	u_1	u_2	y_1	y_2	y_3	y_4	y_5	y_6	y_7	y_8	y_9	y_{10}	y_{11}	y_{12}	y_{13}	y_{14}	y_{15}	y_{16}
0	0	0	0	0	0	0	0	0	0	0	1	1	1	1	1	1	1	1
1	0	1	0	0	0	0	1	1	1	1	0	0	0	0	1	1	1	1
2	1	0	0	0	1	1	0	0	1	1	0	0	1	1	0	0	1	1
3	1	1	0	1	0	1	0	1	0	1	0	1	0	1	0	1	0	1

U logických funkcí dvou proměnných mohou tyto proměnné nabýt jedné ze 4 (2^2) kombinací a celkem lze definovat 16 (2^4) funkcí.

y_1 – 0 pro libovolnou kombinaci u_1 a u_2

y_2 – logický součin (konjunkce), $y = u_1 \cdot u_2$

y_3 – inhibice, $y = u_1 \cdot \bar{u}_2$

y_4 – opakování, $y = u_1$

y_5 – inhibice, $y = \bar{u}_1 \cdot u_2$

y_6 – opakování, $y = u_2$

y_7 – dilema, $y = \bar{u}_1 \cdot u_2 \vee u_1 \cdot \bar{u}_2$

y_8 – logický součet (disjunkce), $y = u_1 \vee u_2$

y_9 – NOR (Piercova funkce), $y = \overline{u_1 \vee u_2} = \bar{u}_1 \cdot \bar{u}_2$

y_{10} – ekvivalence, $y = \bar{u}_1 \cdot \bar{u}_2 \vee u_1 \cdot u_2$

y_{11} – negace, $y = \bar{u}_2$

y_{12} – implikace, $y = u_1 \vee \bar{u}_2$ (u_1 implikuje u_2)

y_{13} – negace, $y = \bar{u}_1$

y_{14} – implikace, $y = \bar{u}_1 \vee u_2$ (u_2 implikuje u_1)

y_{15} – NAND (Shefferova funkce), $y = \overline{u_1 \cdot u_2} = \bar{u}_1 \vee \bar{u}_2$

y_{16} – jednička pro libovolnou kombinaci u_1 a u_2

Libovolnou logickou funkci jedné nebo dvou logických proměnných lze vyjádřit pomocí funkcí negace, logického součinu (konjunkce) a logického součtu (disjunkce).
Pro praktickou realizaci pomocí logických obvodů se využívají funkce negace a NAND, případně také funkce NOR.
Logickou funkci je vhodné „minimalizovat“, tzn. popsat tuto funkcí pomocí minimálního počtu logických proměnných a logických výrazů, čímž získáme efektivní vyjádření, které je vhodné pro praktickou realizaci.
Jedním z matematických prostředků, pomocí něhož lze pracovat s logickými funkcemi a proměnnými je Booleova algebra.

Základní pravidla Booleovy algebry

1. Zákon vyloučení třetího: $u \vee \overline{u} = 1$
2. Logický rozpor: $u \cdot \overline{u} = 0$
3. Zákon dvojité negace: $\overline{\overline{u}} = u$
4. Zákony opakování: $u \cdot u = u, \quad u \vee u = u$
5. Komutativní zákony: $u_1 \vee u_2 = u_2 \vee u_1, \quad u_1 \cdot u_2 = u_2 \cdot u_1$
6. Asociativní zákony: $u_1 \vee (u_2 \vee u_3) = u_1 \vee u_2 \vee u_3, \quad u_1 \cdot (u_2 \cdot u_3) = u_1 \cdot u_2 \cdot u_3$
7. Distributivní zákony: $u_1 \cdot (u_2 \vee u_3) = u_1 \cdot u_2 \vee u_1 \cdot u_3, \quad u_1 \vee u_2 \cdot u_3 = (u_1 \vee u_2) \cdot (u_1 \vee u_3)$
8. Absorpční zákony: $u_1 \vee (u_1 \cdot u_2) = u_1, \quad u_1 \cdot (u_1 \vee u_2) = u_1$

   $u_1 \vee (\overline{u}_1 \cdot u_2) = u_1 \vee u_2, \quad u_1 \cdot (\overline{u}_1 \vee u_2) = u_1 \cdot u_2$

9. Neutralita: $u \vee 0 = u, \quad u \cdot 1 = u$
10. Agresivita: $u \vee 1 = 1, \quad u \cdot 0 = 0$
11. De Morganovy zákony: $\overline{u_1 \cdot u_2} = \overline{u}_1 \vee \overline{u}_2, \quad \overline{u_1 \vee u_2} = \overline{u}_1 \cdot \overline{u}_2$

Použití těchto pravidel nabízí základní způsob pro minimalizaci logické funkce. Záleží však na našich zkušenostech (umět se rozhodnout jaká pravidla použít) a ne vždy se tak podaří získat minimální tvar funkce.

Příklad: Pomocí pravidel Booleovy algebry dokažte vztah pro disjunkci $y_8 = u_1 \vee u_2$.

Z pravdivostní tabulky uvedené na str. 5 napíšeme tuto funkci ve tvaru

$$y_8 = \bar{u}_1 u_2 \vee u_1 \bar{u}_2 \vee u_1 u_2$$

Návod: Pro pozice v tabulce, kde logická funkce y_8 nabývá hodnoty 1, zapíšeme v logickém součinu kombinaci logických proměnných na daných pozicích. Jestliže některá logická proměnná nabývá v dané kombinaci hodnoty 0, zapíšeme její negaci. Výrazy pro jednotlivé pozice spojíme funkcí logický součet.

Řešení:

aplikace pravidla 7

$$y_8 = \bar{u}_1 u_2 \vee u_1 \bar{u}_2 \vee u_1 u_2 = u_1 \underbrace{\left(u_2 \vee \bar{u}_2\right)}_{1} \vee u_2 \underbrace{\left(u_1 \vee \bar{u}_1\right)}_{1} = u_1 \vee u_2$$

aplikace pravidla 1

Příklad: Pomocí pravidel Booleovy algebry dokažte vztah pro implikaci $y_{12} = u_1 \vee \bar{u}_2$.

Podle pravdivostní tabulky na str. 5 zapíšeme tuto funkci ve tvaru $y_{12} = \bar{u}_1 \bar{u}_2 \vee u_1 \bar{u}_2 \vee u_1 u_2$.

Řešení:

aplikace pravidla 7

$$y_{12} = \bar{u}_1 \bar{u}_2 \vee u_1 \bar{u}_2 \vee u_1 u_2 = u_1 \underbrace{\left(u_2 \vee \bar{u}_2\right)}_{1} \vee \bar{u}_2 \underbrace{\left(u_1 \vee \bar{u}_1\right)}_{1} = u_1 \vee \bar{u}_2$$

aplikace pravidla 1

Příklad: Pomocí pravidel Booleovy algebry minimalizuje logickou funkci $y = f(u_1,u_2,u_3)$.

$$y = \bar{u}_1 u_2 u_3 \vee u_1 \bar{u}_2 \bar{u}_3 \vee u_1 \bar{u}_2 u_3 \vee u_1 u_2 \bar{u}_3 \vee u_1 u_2 u_3$$

aplikace pravidla 7

Řešení:

$$y = \bar{u}_1 u_2 u_3 \vee u_1 \bar{u}_2 \bar{u}_3 \vee u_1 \bar{u}_2 u_3 \vee u_1 u_2 \bar{u}_3 \vee u_1 u_2 u_3 = \bar{u}_1 u_2 u_3 \vee u_1 \bar{u}_2 \underbrace{\left(\bar{u}_3 \vee u_3\right)}_{1} \vee u_1 u_2 \underbrace{\left(\bar{u}_3 \vee u_3\right)}_{1} =$$

aplikace pravidla 1

aplikace pravidla 1

$$= \bar{u}_1 u_2 u_3 \vee u_1 \left(\bar{u}_2 \vee u_2\right) = \bar{u}_1 u_2 u_3 \vee u_1 = \bar{u}_1 u_2 u_3 \vee u_1 \left(\overline{u_2 u_3} \vee u_2 u_3 \vee u_2 u_3\right) =$$

aplikace pravidla 4

aplikace pravidel 7 a 1

$$= \bar{u}_1 u_2 u_3 \vee u_1 \overline{u_2 u_3} \vee u_1 u_2 u_3 \vee u_1 u_2 u_3 = u_1 \left(\overline{u_2 u_3} \vee u_2 u_3\right) \vee u_2 u_3 \left(\bar{u}_1 \vee u_1\right) = u_1 \vee u_2 u_3$$

Příklad: Pomocí pravidel Booleovy algebry minimalizuje logickou funkci y = f(u_1,u_2,u_3, u_4)

$$y=(u_1 \vee u_2)(\bar{u}_3 \vee \bar{u}_4) \vee \overline{\bar{u}_1\bar{u}_2 \vee \bar{u}_3\bar{u}_4}$$

aplikace pravidla 11, 2X

Řešení:

$$y=(u_1 \vee u_2)(\bar{u}_3 \vee \bar{u}_4) \vee \overline{\bar{u}_1\bar{u}_2 \vee \bar{u}_3\bar{u}_4} = (u_1 \vee u_2)(\bar{u}_3 \vee \bar{u}_4) \vee \overline{\bar{u}_1\bar{u}_2} \cdot \overline{\bar{u}_3\bar{u}_4} =$$

aplikace pravidla 3

$$=(u_1 \vee u_2)(\bar{u}_3 \vee \bar{u}_4) \vee (\bar{\bar{u}}_1 \vee \bar{\bar{u}}_2)(\bar{\bar{u}}_3 \vee \bar{\bar{u}}_4) = (u_1 \vee u_2)(\bar{u}_3 \vee \bar{u}_4) \vee (u_1 \vee u_2)(u_3 \vee u_4) =$$

$$=(u_1 \vee u_2)\left((\bar{u}_3 \vee \bar{u}_4) \vee (u_3 \vee u_4)\right) = u_1 \vee u_2$$

Vyjádření logických funkcí

Verbální

Daný problém je vyjádřen pomocí větného spojení. Úkolem je nalézt v dané větě výroky a určit jejich vzájemný vztah a tím logickou funkční závislost.

Příklad

Petr jde nakoupit, pokud má hlad nebo doma nemá žádné rohlíky.

V tomto případě je logickou funkcí y výrok „Petr jde nakoupit". Funkce y je závislá na proměnných u_1 a u_2, což jsou výroky „má hlad" a „nemá doma žádné rohlíky". Funkční závislost je v tomto případě $y = u_1 \vee u_2$, jelikož je pravda, že Petr jde nakoupit (logická funkce nabývá hodnoty 1) jestliže je pravdivý alespoň jeden z výroků u_1 a u_2 (tj. proměnné u_1 a u_2 nabývají hodnoty 1).

Příklad

$$\underbrace{\text{Pavel přijede do práce včas}}_{y}\text{, pokud }\underbrace{\text{nastartuje auto}}_{u_1}\text{ a }\underbrace{\text{centrum nebude ucpané}}_{u_2}.$$

V tomto případě je pro pravdivost výroku y nutné současné splnění obou výroků u_1 a u_2. Funkční závislost tedy je $y = u_1 \cdot u_2$.

Pravdivostní tabulka

Pravdivostní tabulka vyjadřuje hodnoty logické funkce pro všechny kombinace logických proměnných, které se ve výrazu vyskytují.

i – stavový index, určuje pozici v tabulce, zároveň odpovídá dekadické hodnotě kombinace proměnných u (binární číslo). Index nabývá hodnot 0 – 2^n-1, kde n je počet proměnných.

Velikost tabulky se rychle zvyšuje s narůstajícím počtem proměnných (2, 4, 8, 16, 32,…).

$$y = u_1 \cdot u_2 \vee \bar{u}_1 \cdot u_2 \cdot u_3$$

kombinace vstupních proměnných

i	u_1	u_2	u_3	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	0
5	1	0	1	0
6	1	1	0	1
7	1	1	1	1

stavový index

hodnoty funkce y

Sestavení pravdivostní tabulky

Pro danou logickou funkci lze pravdivostní tabulku sestavit dvěma způsoby. Názorně si je ukážeme na příkladu funkce $y = f(u_1, u_2, u_3)$.

$$y = u_1 \vee \bar{u}_1 \bar{u}_2 u_3$$

1. řešení:

Logická funkce je tvořena dvěma logickými výrazy spojenými operací disjunkce (nebo, logický součet). Pro oba výrazy tedy můžeme stanovit kombinace proměnných u_1, u_2 a u_3, pro které nabývá daný výraz logické hodnoty 1.

Výraz u_1: nabývá hodnoty 1, jestliže $u_1 = 1$ a pro libovolná u_2 a u_3.

Výraz $\bar{u}_1\bar{u}_2u_3$: nabývá hodnoty 1, jestliže $u_1=0$, $u_2=0$ a $u_3 = 1$.

Vzhledem k tomu, že logické výrazy jsou spojeny disjunkcí, je kombinace logických proměnných u_1, u_2 a u_3 , pro které nabývá funkce y hodnotu 1, dána „sjednocením“ kombinací proměnných pro výrazy u_1 a $\bar{u}_1\bar{u}_2u_3$.

kombinace pro výraz $\bar{u}_1\bar{u}_2u_3$

kombinace pro výraz u_1

i	**u_1**	**u_2**	**u_3**	**y**
0	0	0	0	0
1	0	0	1	1
2	0	1	0	0
3	0	1	1	0
4	1	0	0	1
5	1	0	1	1
6	1	1	0	1
7	1	1	1	1

2. řešení:

Všechny kombinace vstupních proměnných dosazujeme do předpisu funkce y a výsledky zapisujeme do tabulky.

Zápis logické funkce ze zadané pravdivostní tabulky

Nyní bude ukázáno, jak lze ze zadané pravdivostní tabulky napsat předpis pro logickou funkci. Prvně byl tento přístup zmíněn na straně 8. Mějme zadanou následující pravdivostní tabulku:

i	$\mathbf{u_1}$	$\mathbf{u_2}$	$\mathbf{u_3}$	**y**
0	0	0	0	0
1	0	0	1	0
2	0	1	0	1
3	0	1	1	1
4	1	0	0	0
5	1	0	1	1
6	1	1	0	1
7	1	1	1	0

Postup:

1. V pravdivostní tabulce najdeme indexy, pro které nabývá funkce y hodnotu 1. Jedná se o indexy 2, 3, 5 a 6.

2. Pro jednotlivé indexy zapíšeme logické proměnné u_1, u_2 a u_3 nebo jejich negace ve tvaru logického součinu, tak aby pro danou kombinaci byla výsledkem hodnota 1. Výraz $u_1u_2u_3$, je roven hodnotě 1, jestliže všechny proměnné nabývají hodnoty 1. Jestliže některá proměnná nabývá hodnoty 0, nahradíme ji její negací. Např. pro index 2 je hodnota funkce y rovna 1 pro kombinaci proměnných $u_1 = 0$, $u_2 = 1$, $u_3 = 0$. Této kombinaci proměnných *u* odpovídá logický výraz:

$$y = \bar{u}_1 u_2 \bar{u}_3$$

$$y = \bar{0} 1 \bar{0} = 1$$

Stejným způsobem pokračujeme také u ostatních indexů a výsledek zapíšeme ve tvaru logického součtu jednotlivých výrazů.

$$y = \bar{u}_1 u_2 \bar{u}_3 \vee \bar{u}_1 u_2 u_3 \vee u_1 \bar{u}_2 u_3 \vee u_1 u_2 \bar{u}_3$$

Pozor: Výsledek obvykle není v minimálním tvaru.

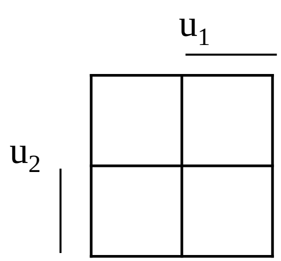
Karnaughova mapa

Karnaughova mapa je dalším způsobem pro vyjádření logické funkce. Jedná se o tabulku, která se skládá z 2^n polí, kde n značí počet logických proměnných dané funkce. Při vyjádření logické funkce ve formě Karnaughovy mapy lze snadným způsobem danou funkci minimalizovat.
Konstrukce Karnaughovy mapy pro logické funkce dvou, tří a čtyř logických proměnných bude vysvětlena na příkladech.

Konstrukce Karnaughovy mapy pro logickou funkci dvou proměnných
Uvažujme logickou funkci vyjádřenou pomocí pravdivostní tabulky.

i	**u_1**	**u_2**	**y**
0	0	0	1
1	0	1	0
2	1	0	1
3	1	1	0

Pro funkci dvou proměnných se Karnaughova mapa skládá ze 4 polí.



Jednotlivá pole odpovídají řádkům (kombinacím log. proměnných) pravdivostní tabulky.

Čárka u daného sloupce nebo řádku značí, že na těchto polích nabývá daná logická proměnná hodnotu 1.
Na zbylých polích nabývá hodnotu 0.

Při vyplňování Karnaughovy mapy postupujeme tak, že do jednotlivých polí dosazujeme hodnoty logické funkce.

i	**u_1**	**u_2**	**y**
0	0	0	1
1	0	1	0
2	1	0	1
3	1	1	0

		u_1
	1	1
u_2	0	0

$i = 0, u_1 = 0, u_2 = 0, y = 1$

$i = 1, u_1 = 0, u_2 = 1, y = 0$

$i = 2, u_1 = 1, u_2 = 0, y = 1$

$i = 3, u_1 = 1, u_2 = 1, y = 0$

Příklad: Vyjádřete logickou funkci $y = \bar{u}_1 u_2 \vee u_1 \bar{u}_2$ pomocí Karnaughovy mapy.

Řešení:

1. Logickou funkci zapíšeme pomocí pravdivostní tabulky:

i	**u_1**	**u_2**	**y**
0	0	0	0
1	0	1	1
2	1	0	1
3	1	1	0

2. Sestavíme Karnaughovu mapu

		u_1
	0	1
u_2	1	0

Konstrukce Karnaughovy mapy pro logickou funkci tří proměnných

i	u_1	u_2	u_3	y
0	0	0	0	1
1	0	0	1	1
2	0	1	0	0
3	0	1	1	0
4	1	0	0	0
5	1	0	1	1
6	1	1	0	1
7	1	1	1	1

Karnaughova mapa pro logickou funkci tří proměnných se skládá z osmi polí. Postup při jejím sestavování je stejný jako u mapy logické funkce dvou proměnných. Jestliže chceme Karnaughovu mapu využívat pro minimalizaci logických funkcí, musíme se při sestavování map funkcí tří a více proměnných držet pravidla, že v sousedních polích se může měnit hodnota jen jedné proměnné. V ilustrativních mapách nebudou jednotlivá pole rozlišena barevně, ale v dolním rohu každého pole bude index odpovídající řádce v pravdivostní tabulce.

Možné Karnaughovy mapy:

1

		u_2	u_2 u_3	u_3
	1_0	0_2	0_3	1_1
u_1	0_4	1_6	1_7	1_5

2

		u_2	u_2 u_1	u_1
	1_0	0_2	1_6	0_4
u_3	1_1	0_3	1_7	1_5

3

		u_3	u_3 u_2	u_2
	1_0	1_1	0_3	0_2
u_1	0_4	1_5	1_7	1_6

Příklad: Vyjádřete logickou funkci $y = \bar{u}_1 u_2 u_3 \vee \bar{u}_1 u_2 \bar{u}_3 \vee u_1 \left(u_2 \bar{u}_3 \vee \bar{u}_2 u_3 \right)$ pomocí Karnaughovy mapy.

1. Logickou funkci zapíšeme pomocí pravdivostní tabulky:

i	**u_1**	**u_2**	**u_3**	**y**
0	0	0	0	0
1	0	0	1	0
2	0	1	0	1
3	0	1	1	1
4	1	0	0	0
5	1	0	1	1
6	1	1	0	1
7	1	1	1	0

2. Sestavíme Karnaughovu mapu: zvolíme typ č. 2

u_2 u_1

	$0\,_{0}$	$1\,_{2}$	$1\,_{6}$	$0\,_{4}$
u_3	$0\,_{1}$	$1\,_{3}$	$0\,_{7}$	$1\,_{5}$

Konstrukce Karnaughovy mapy pro logickou funkci čtyř proměnných

i	u_1	u_2	u_3	u_4	y
0	0	0	0	0	0
1	0	0	0	1	1
2	0	0	1	0	1
3	0	0	1	1	0
4	0	1	0	0	1
5	0	1	0	1	0
⋮	⋮	⋮	⋮	⋮	⋮
14	1	1	1	0	0
15	1	1	1	1	1

Při sestavování Karnaughovy mapy se opět musíme držet pravidla, že v sousedních polích se může měnit hodnota jen jedné proměnné. Stejně jako u Karnaughovy mapy logické funkce tří proměnných, i v tomto případě jsou možné různá rozložení pozic v mapě.

Možné Karnaughovy mapy:

1

u_2 u_1

	0_{0}	1_{4}	1_{12}	1_{8}
u_4	1_{1}	0_{5}	1_{13}	0_{9}
	0_{3}	0_{7}	1_{15}	1_{11}
u_3	1_{2}	1_{6}	0_{14}	0_{10}

2

u_1 u_2

	0_{0}	1_{8}	1_{12}	1_{4}
u_3	1_{2}	0_{10}	0_{14}	1_{6}
	0_{3}	1_{11}	1_{15}	0_{7}
u_4	1_{1}	0_{9}	1_{13}	0_{5}

Příklad: Vyjádřete logickou funkci $y=\left(u_1 \vee u_2 \vee u_3\right)\left(u_1\bar{u}_3 \vee \overline{u_2u_3}u_4\right)$ pomocí Karnaughovy mapy.

Řešení:

1. Upravíme logickou funkci podle pravidel Booleovy algebry:

$$y=\left(u_1 \vee u_2 \vee u_3\right)\left(u_1\bar{u}_3 \vee \overline{u_2u_3}u_4\right)=u_1u_1\bar{u}_3 \vee u_1u_2\bar{u}_3 \vee u_1u_3\bar{u}_3 \vee u_1\overline{u_2u_3}u_4 \vee u_2\overline{u_2u_3}u_4 \vee u_3\overline{u_2u_3}u_4=$$

$$=u_1\bar{u}_3 \vee u_1u_2\bar{u}_3 \vee u_1u_4\left(\bar{u}_2 \vee \bar{u}_3\right) \vee u_2u_4\left(\bar{u}_2 \vee \bar{u}_3\right) \vee u_3u_4\left(\bar{u}_2 \vee \bar{u}_3\right)=u_1\bar{u}_3 \vee u_1u_2\bar{u}_3 \vee u_1\bar{u}_2u_4 \vee u_1\bar{u}_3u_4 \vee$$

$$u_2\bar{u}_2u_4 \vee u_2\bar{u}_3u_4 \vee \bar{u}_2u_3u_4 \vee u_3\bar{u}_3u_4=\left(1 \vee u_2 \vee u_4\right) \vee u_1\bar{u}_2u_4 \vee u_2\bar{u}_3u_4 \vee \bar{u}_2u_3u_4=u_1\bar{u}_3 \vee u_1\bar{u}_2u_4 \vee u_2\bar{u}_3u_4 \vee \bar{u}_2u_3u_4$$

2. Logickou funkci zapíšeme do pravdivostní tabulky:

i	**u_1**	**u_2**	**u_3**	**u_4**	**y**	**i**	**u_1**	**u_2**	**u_3**	**u_4**	**y**
0	0	0	0	0	0	8	1	0	0	0	1
1	0	0	0	1	0	9	1	0	0	1	1
2	0	0	1	0	0	10	1	0	1	0	0
3	0	0	1	1	1	11	1	0	1	1	1
4	0	1	0	0	0	12	1	1	0	0	1
5	0	1	0	1	1	13	1	1	0	1	1
6	0	1	1	0	0	14	1	1	1	0	0
7	0	1	1	1	0	15	1	1	1	1	0

3. Sestavíme Karnaughovu mapu, zvolíme typ č. 1:

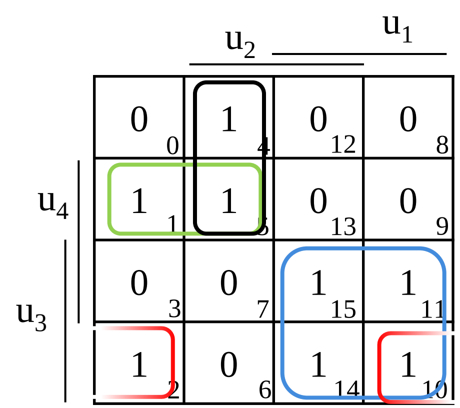
		u_2	u_2, u_1	u_1
	0_{0}	0_{4}	1_{12}	1_{8}
u_4	0_{1}	1_{5}	1_{13}	1_{9}
u_4, u_3	1_{3}	0_{7}	0_{15}	1_{11}
u_3	0_{2}	0_{6}	0_{14}	0_{10}

Minimalizace logických výrazů s využitím Karnaughovy mapy

Uvažujme logickou funkci vyjádřenou pomocí Karnaughovy mapy

		u_2	u_2, u_1	u_1
	0_{0}	0_{4}	0_{12}	1_{8}
u_4	1_{1}	1_{5}	0_{13}	0_{9}
u_4, u_3	0_{3}	0_{7}	1_{15}	1_{11}
u_3	1_{2}	0_{6}	1_{14}	1_{10}

1. Nalezneme v Karnaughově mapě všechna pole s hodnotou 1.
2. Pole, která spolu sousedí označíme smyčkami. Smyčky se mohou skládat z 1, 2, 4, 8,… až 2^n polí. Smyčky se mohou navzájem překrývat.
3. Jedno pole může být součástí více smyček.
4. Protilehlé rohy spolu sousedí.
5. Pro minimalizace logické funkce používáme co nejmenší počet smyček (tj. volíme smyčky co největší).



5. Určíme logický výraz pro každou smyčku. Přitom se držíme následujících pravidel:

- jestliže v rámci dané smyčky mění proměnná svoji hodnotu (nabývá hodnoty 0 i 1), je tato proměnná z logického výrazu vyloučena,
- jestliže v rámci dané smyčky nabývá proměnná hodnoty 0, pak v logickém výrazu vystupuje její negace.

Smyčka $\square$: logická proměnná u_1 nabývá ve smyčce pouze hodnotu 0, proměnná u_2 nabývá ve smyčce hodnoty 0 i 1 (je eliminována), proměnná u_3 nabývá ve smyčce pouze hodnotu 0, proměnná u_4 nabývá ve smyčce pouze hodnotu 1. Logický výraz pro tuto smyčku tedy je :

$$\boxed{y_1 = \bar{u}_1 \bar{u}_3 u_4}$$

Smyčka $\square$: logická proměnná u_1 nabývá ve smyčce pouze hodnotu 0, proměnná u_2 nabývá ve smyčce pouze hodnotu 1, logická proměnná u_3 nabývá ve smyčce pouze hodnotu 0, logická proměnná u_4 mění svou hodnotu. Logický výraz pro tuto smyčku tedy je:

$$\boxed{y_2 = \bar{u}_1 u_2 \bar{u}_3}$$

Smyčka $\square$: logická proměnná u_1 nabývá ve smyčce pouze hodnotu 1, proměnná u_2 mění svou hodnotu, logická proměnná u_3 nabývá ve smyčce pouze hodnotu 1, logická proměnná u_4 mění svou hodnotu. Logický výraz pro tuto smyčku tedy je:

$$\boxed{y_3 = u_1 u_3}$$

Smyčka $\square$: logická proměnná u_1 mění svou hodnotu, proměnná u_2 nabývá hodnotu 0, proměnná u_3 nabývá hodnotu 1, proměnná u_4 nabývá hodnotu 0. Logický výraz tedy je:

$$\boxed{y_4 = \bar{u}_2 u_3 \bar{u}_4}$$

Logickou funkci se zapíše ve tvaru disjunkce výrazů pro jednotlivé smyčky

$$\boxed{y = \bar{u}_1 \bar{u}_3 u_4 \vee \bar{u}_1 u_2 \bar{u}_3 \vee u_1 u_3 \vee \bar{u}_2 u_3 \bar{u}_4}$$

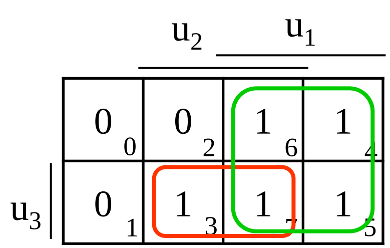
Příklad:

Pomocí pravidel Booleovy algebry jsme minimalizovali logickou funkci

$$y = \bar{u}_1 u_2 u_3 \vee u_1 \bar{u}_2 \bar{u}_3 \vee u_1 \bar{u}_2 u_3 \vee u_1 u_2 \bar{u}_3 \vee u_1 u_2 u_3$$

Nyní provedeme minimalizaci pomocí Karnaughovy mapy.

i	u_1	u_2	u_3	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	1
6	1	1	0	1
7	1	1	1	1



Smyčka : proměnná u_1 nabývá hodnotu 1, proměnné u_2 a u_3 mění svou hodnotu:

$$y_1 = u_1$$

Smyčka : proměnná u_1 mění svou hodnotu, proměnné u_2 a u_3 nabývají hodnotu 1:

$$y_2 = u_2 u_3$$

Funkce v minimálním tvaru:

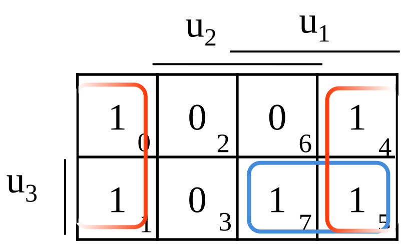
$$y = u_1 \vee u_2 u_3$$

Minimalizace pomocí Karnaughových map je rychlejší a jednodušší, než používání pravidel Booleovy algebry. Je však vhodná pro funkce s nižším počtem proměnných.

Příklad:

Zapište logickou funkci do Karnaughovy mapy a proveďte její minimalizaci

$$y = u_1 u_2 u_3 \vee u_1 \bar{u}_2 u_3 \vee \bar{u}_1 \bar{u}_2 u_3 \vee u_1 \bar{u}_2 \bar{u}_3 \vee \bar{u}_1 \bar{u}_2 \bar{u}_3$$



Smyčka : $y_1 = u_1 u_3$

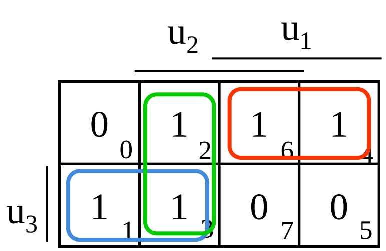
Smyčka : $y_2 = \bar{u}_2$

$$y = u_1 u_3 \vee \bar{u}_2$$

Příklad:

Zapište logickou funkci do Karnaughovy mapy a proveďte její minimalizaci

$$y = u_1 u_2 \bar{u}_3 \vee u_1 \bar{u}_2 \bar{u}_3 \vee \bar{u}_1 u_2 u_3 \vee \bar{u}_1 u_2 \bar{u}_3 \vee \bar{u}_1 \bar{u}_2 u_3$$



Smyčka : $y_1 = \bar{u}_1 u_3$

Smyčka : $y_2 = \bar{u}_1 u_2$

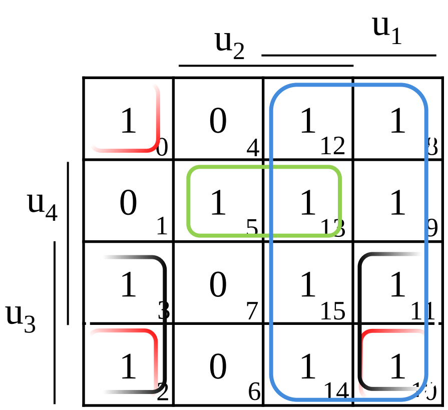
Smyčka : $y_3 = u_1 \bar{u}_3$

$$y = \bar{u}_1 u_3 \vee \bar{u}_1 u_2 \vee u_1 \bar{u}_3$$

Příklad:

Zapište logickou funkci do Karnaughovy mapy a proveďte její minimalizaci

$$y = u_1\bar{u}_2\bar{u}_3\bar{u}_4 \vee u_1\bar{u}_2\bar{u}_3u_4 \vee u_1\bar{u}_2u_3u_4 \vee u_1\bar{u}_2u_3\bar{u}_4 \vee u_1u_2\bar{u}_3\bar{u}_4 \vee u_1u_2\bar{u}_3u_4 \vee u_1u_2u_3u_4 \vee u_1u_2u_3\bar{u}_4 \vee$$
$$\bar{u}_1u_2\bar{u}_3u_4 \vee \bar{u}_1\bar{u}_2\bar{u}_3\bar{u}_4 \vee \bar{u}_1\bar{u}_2u_3u_4 \vee \bar{u}_1\bar{u}_2u_3\bar{u}_4$$



Smyčka : $y_1 = u_1$

Smyčka : $y_2 = u_2\bar{u}_3u_4$

Smyčka : $y_3 = \bar{u}_2u_3$

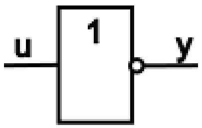
Smyčka : $y_4 = \bar{u}_2\bar{u}_4$

$$y = u_1 \vee u_2\bar{u}_3u_4 \vee \bar{u}_2u_3 \vee \bar{u}_2\bar{u}_4$$

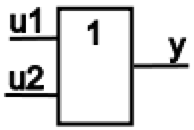
Blokové schéma

Pro realizaci logický funkcí (logických obvodů) je vhodný jejich zápis ve formě blokových schémat.

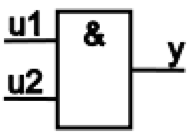
Základní prvky blokového schématu



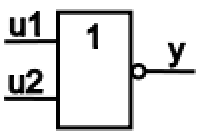
negace, invertor (NOT), $y = \overline{u}$



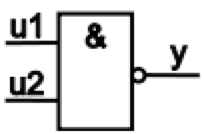
logický součet, disjunkce (OR), $y = u_1 \vee u_2$



logický součin, konjunkce (AND), $y = u_1 \cdot u_2$



negovaný logický součet (NOR), $y = \overline{u_1 \vee u_2}$

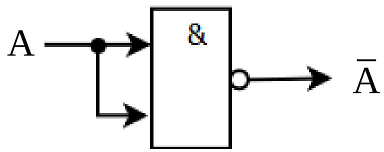


negovaný logický součin (NAND), $y = \overline{u_1 \cdot u_2}$

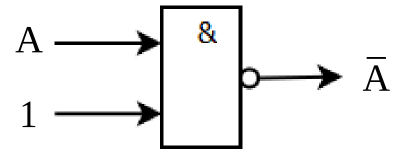
Počty vstupů jednotlivých bloků se mohou lišit v závislosti na daném bloku (hradle).

V číslicové technice se pro realizaci logických funkcí nejčastěji používají invertory, hradla NAND a hradla NOR. Vhodným zapojením hradel NAND a NOR je možné realizovat jakoukoliv logickou funkci.

Realizace invertoru pomocí hradla NAND:

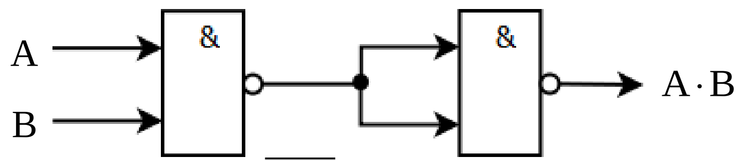


$\overline{A\cdot A}=\overline{A}$



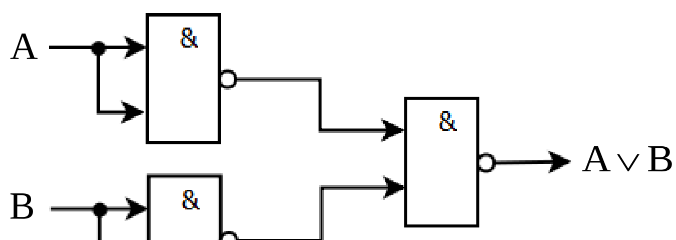
$\overline{A\cdot 1}=\overline{A}$

Realizace funkce AND pomocí hradel NAND:



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

Realizace funkce OR pomocí hradel NAND:



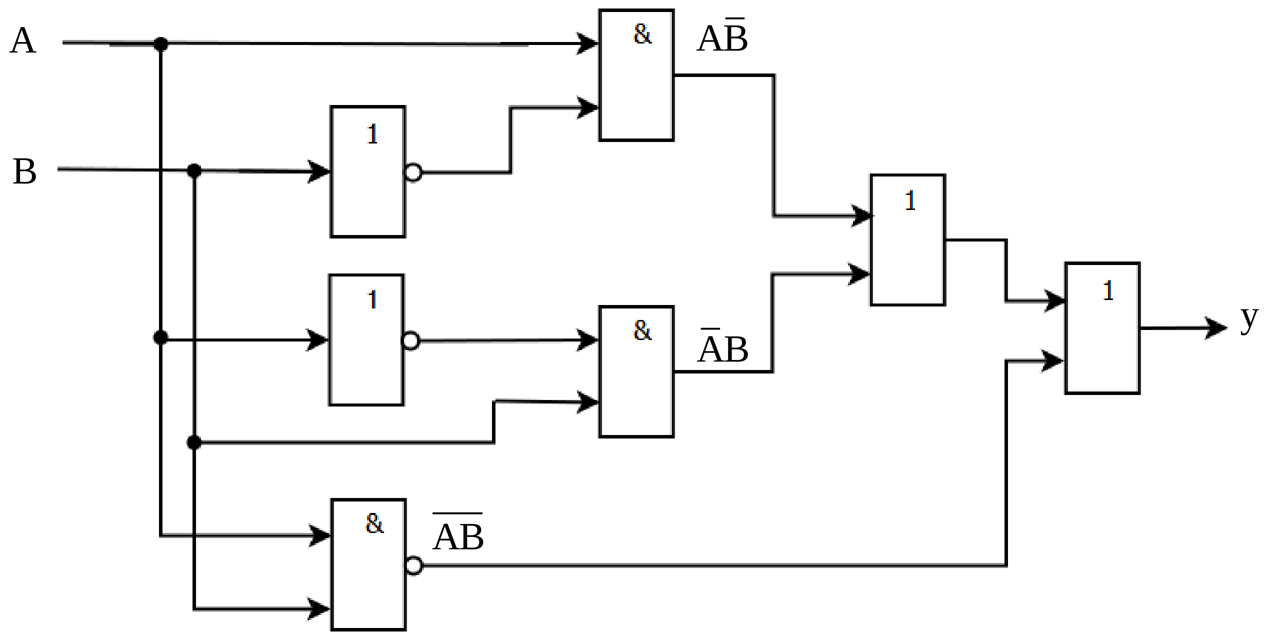
$\overline{\overline{A\vee B}}=\overline{\overline{A}\cdot\overline{B}}$ De Morganovy zákony

Příklad

Nakreslete blokové schéma pro logický výraz

$$y = A\bar{B} \vee \bar{A}B \vee \overline{AB}$$

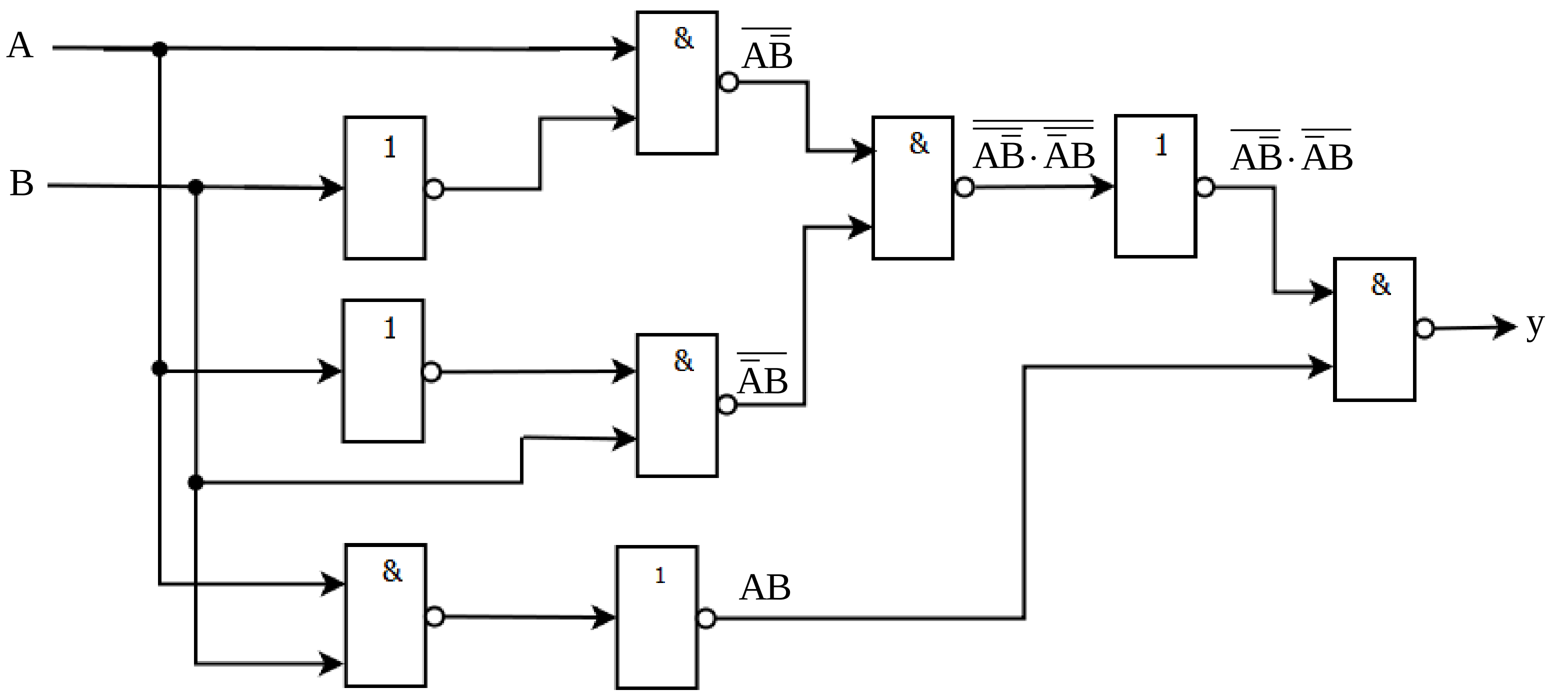
Uvažujme pouze hradla s 1 nebo 2 vstupy.



Blokové schéma s použitím hradel NOT a NAND:

S využitím De Morganových zákonů upravíme logickou funkci

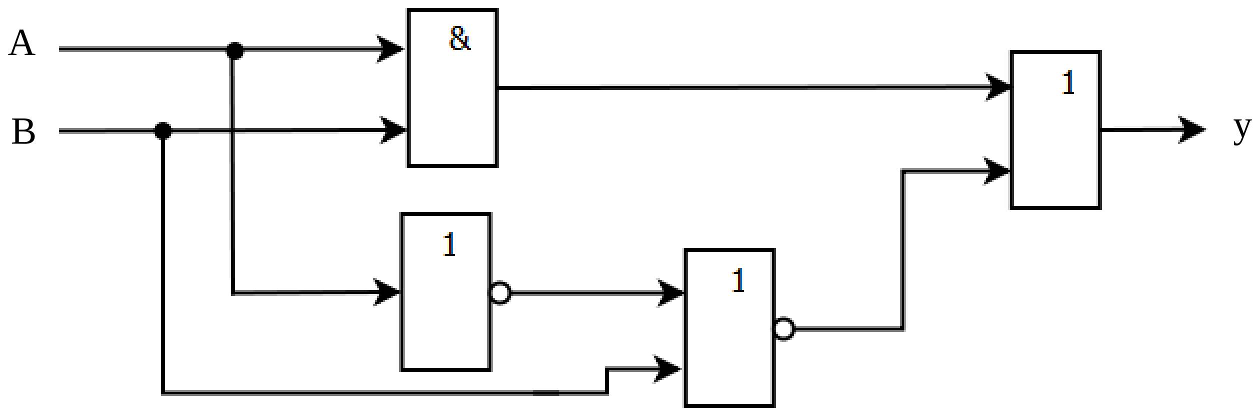
$$y=\overline{\overline{A\bar{B}\vee\bar{A}B\vee\overline{AB}}}=\overline{\overline{A\bar{B}}\cdot\overline{\bar{A}B}\cdot\overline{\overline{AB}}}$$



Příklad

Nakreslete blokové schéma pro logický výraz

$$y = A \cdot B \vee \overline{\overline{A} \vee B}$$



Příklad

Systém vytápění místnosti pracuje podle pravidla:

Zapni topení v případě, že je nastavena manuální regulace teploty a stisknuto tlačítko plus nebo v případě, že je nastavena automatická regulace a skutečná teplota v místnosti je nižší než požadovaná teplota.

Navrhněte blokové schéma pro logický obvod zajišťující vytápění v místnosti:

Označme si tyto logické výrazy

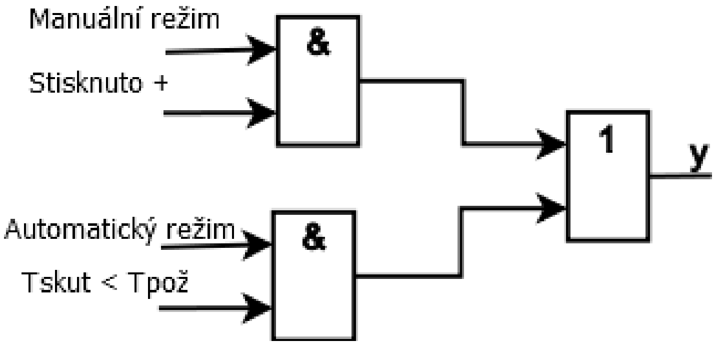
y – topení (zapnuto 1/vypnuto 0)

A – manuální režim

B – stisknuto +

C – automatický režim

D – Tskut < Tpož



Pro funkci y tedy platí

$$y = A \cdot B \vee C \cdot D$$

Pokud budeme uvažovat, že ovládací panel má pouze jedno tlačítko pro přepínání mezi manuálním a automatickým režimem, pak by byl výraz u_3 přebytečný ($u_3 = \overline{u}_1$). Upravte blokové schéma podle nového zadání.

Příklad

Automatický stroj na řezání dřeva se skládá z pily a posuvníku, který vždy po odříznutí posune zbývající materiál o stanovenou vzdálenost. Pila se může pohybovat pouze ve dvou směrech, doprava (y_1) a doleva (y_2). V levé krajní poloze je umístěn snímač A, který signalizuje, že pila dojela doleva, v pravé krajní poloze je snímač B, který signalizuje, že pila dojela doprava. Pokud se pila pohybuje zprava doleva, pak při dojezdu do levé krajní polohy vyšle snímač A signál, který změní směr pohybu pily doprava. Naopak signál ze snímače B mění směr pohybu pily zprava doleva.

Sestavíme pravdivostní tabulku

pila je v levé krajní poloze a pohybuje se doprava

pila je v pravé krajní poloze a pohybuje se doleva

A	**B**	$\mathbf{y_1}$	$\mathbf{y_2}$
1	0	1	0
0	0	1	0
0	1	0	1
0	0	0	1

stejné kombinaci A a B odpovídá jiná kombinace y_1 a y_2

Z pravdivostní tabulky vidíme, že signál A spouští y_1 a maže y_2, signál B spouští y_2 a maže y_1.

Jedná se o sekvenční logický obvod – výstup logického systému závisí i předchozím stavu (obvody s pamětí).

Z uvedené pravdivostní tabulky plyne

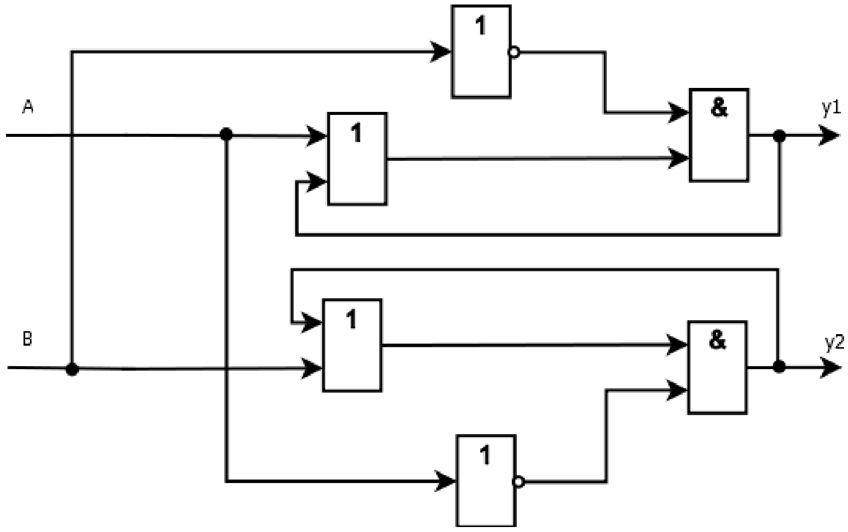
$$y_1 = \left(y_1 \vee A\right) \cdot \overline{B}$$

pila jede doprava, pokud v předchozím časovém okamžiku jela doprava (paměť) nebo snímač A vyslal signál a zároveň se nedostal do pravé krajní polohy

$$y_2 = \left(y_2 \vee B\right) \cdot \overline{A}$$

pila jede doleva, pokud v předchozím časovém okamžiku jela doleva (paměť) nebo snímač B vyslal signál a zároveň se nedostal do levé krajní polohy

Blokové schéma



Zdroje a doporučená literatura

- F. Tůma: Kybernetika, Západočeská univerzita v Plzni
- I. Švarc: učební texty k předmětu Automatizace a regulace, VUT v Brně, dostupné na http://autnt.fme.vutbr.cz/svarc/ZakladyAutomatizace.pdf
- J. Koziorek, L. Chromčák: Logické systémy řízení, učební text, VŠB – TU Ostrava, 2007, dostupné na http://www.elearn.vsb.cz/archivcd/FEI/LSA/Logicke%20systemy%20rizeni.pdf

evropský sociální fond v ČR

EVROPSKÁ UNIE

MINISTERSTVO ŠKOLSTVÍ, MLÁDEŽE A TĚLOVÝCHOVY

OP Vzdělávání pro konkurenceschopnost

INVESTICE DO ROZVOJE VZDĚLÁVÁNÍ

Poděkování

Tento projekt je spolufinancován
Evropským sociálním fondem a státním rozpočtem České republiky

Projekt CZ.1.07/2.2.00/15.0383
Inovace studijního oboru Dopravní a manipulační technika
s ohledem na potřeby trhu práce