

1. ČÍSELNÉ SOUSTAVY

1.1. Zobrazení informace

Když pracujeme s čísly, používáme symboly, jimž přiřazujeme deset různých hodnot od nuly do devíti. Používáme to, co matematici nazývají **soustava zobrazení čísel se základem deset**.

V běžném životě využíváme i jiné soustavy např. šedesátkové (sekundy), dvanáctkové (hodiny).

První stroje na zpracování informací byly mechanické a nebylo u nich složité zobrazovat čísla se základem deset pomocí mechanické součástky, která mohla snadno mít deset stabilních stavů, z nichž každý odpovídal jednomu znaku se základem deset. Mechanické systémy jsou pomalé, složité a tedy i drahé, zvláště je-li třeba zvětšovat výkonnost. Rovněž stupeň jejich rozvoje v podstatě končí u jednoduchých aritmetických strojů.

Velký pokrok v oblasti číslicových výpočtů nastal během druhé světové války zavedením elektronek. Potom se kolem r. 1958 objevily tranzistorové počítače a poměrně nedávno počítače s integrovanými obvody a nynější mikro počítače. Nové stroje jsou od dřívějších desítkových strojů odlišné strukturou, výkonem, spolehlivostí a zvláště použitím dvojkové soustavy zobrazení čísel a veličin. Realizovat elektrické obvody se dvěma stabilními stavy je totiž jednoduché a umožňuje to další vývoj těchto zařízení.

1.2. Číselné soustavy

Jak již bylo řečeno, běžně používanou soustavou, v níž provádíme písemné výpočty, je desítková soustava. Jednotlivé číslice čísla udávají počty mocnin deseti.

$$(4385)_{10} = 4 \cdot 10^3 + 3 \cdot 10^2 + 8 \cdot 10^1 + 5 \cdot 10^0$$

Desítková soustava patří mezi polyadické soustavy, které se vyznačují tím, že celé číslo **a** je v nich vyjádřeno jako mnohočlen:

$$a = a_n z^n + a_{n-1} z^{n-1} + \dots + a_1 z^1 + a_0 z^0 + a_{-1} z^{-1} + a_{-2} z^{-2} + \dots + a_{-n} z^{-n}$$

kde **z** je základ soustavy (v našem případě 10). Součinitelé **a₀** až **a_n** mohou nabývat hodnot 0, 1, ..., z-1 (v našem případě 0 až 9). Obdobně **a₋₁** až **a_{-n}**.

Pomocí elektronických prvků je výhodnější zobrazovat číslo ve dvojkové (binární) soustavě, protože tyto prvky mají většinou dva stavy. Čísla jsou zde vyjádřena pomocí číslic **0** a **1** jako součet mocnin dvou. Tak jako v desítkové soustavě má i zde každá číslice význam odpovídající jejímu umístění v dvojkovém čísle. Převod dvojkového čísla na desítkové provedeme jednoduše tak, že řády, v nichž je ve dvojkovém čísle **1**, vyjádříme v desítkové soustavě pomocí mocnin dvou.

$$(101000101)_2 = 1 \cdot 2^8 + 0 \cdot 2^7 + 1 \cdot 2^6 + 0 \cdot 2^5 + 0 \cdot 2^4 + 0 \cdot 2^3 + 1 \cdot 2^2 + 0 \cdot 2^1 + 1 \cdot 2^0 = (325)_{10}$$

Lze dokázat, že každé dvojkové číslo má průměrně 3,3 krát více míst než stejné číslo vyjádřené v desítkové soustavě. Proto se dvojková soustava nehodí pro ruční výpočty a je určena výhradně pro použití v počítačích.

Jakým způsobem se převádějí čísla z desítkové do dvojkové soustavy? Jednoduchý způsob převodu **celých desítkových čísel** záleží v tom, že číslo dělíme postupně dvěma a zbytky po dělení (**0** nebo **1**) tvoří obraz čísla ve dvojkové soustavě počínaje nejnižší číslicí.

| | | |
|----------------------|-----------|---|
| $(49)_{10} : 2 = 24$ | zbytek: 1 | ↑ |
| $24 : 2 = 12$ | zbytek: 0 | |
| $12 : 2 = 6$ | zbytek: 0 | |
| $6 : 2 = 3$ | zbytek: 0 | |
| $3 : 2 = 1$ | zbytek: 1 | |
| $1 : 2 = 0$ | zbytek: 1 | |

Výsledné dvojkové číslo je tedy $(110001)_2$

Tento způsob převodu vyžaduje operace s desítkovými čísly, proto se hodí pro ruční převod čísel z jedné z jedné soustavy do druhé. Pro počítače pracující ve dvojkové soustavě se používají převody jiné.

Pro usnadnění práce s čísly v dvojkové soustavě se často pracuje s čísly zobrazenými v osmičkové (oktálové) soustavě nebo v šestnáctkové (hexadecimální) soustavě. Převody mezi soustavou desítkovou, dvojkovou, osmičkovou a šestnáctkovou jsou uvedeny v tabulce 1.

Tab.1.: Převodní tabulka:

| D | B | O | H |
|----|------|----|---|
| 0 | 0000 | 0 | 0 |
| 1 | 0001 | 1 | 1 |
| 2 | 0010 | 2 | 2 |
| 3 | 0011 | 3 | 3 |
| 4 | 0100 | 4 | 4 |
| 5 | 0101 | 5 | 5 |
| 6 | 0110 | 6 | 6 |
| 7 | 0111 | 7 | 7 |
| 8 | 1000 | 10 | 8 |
| 9 | 1001 | 11 | 9 |
| 10 | 1010 | 12 | A |
| 11 | 1011 | 13 | B |
| 12 | 1100 | 14 | C |
| 13 | 1101 | 15 | D |
| 14 | 1110 | 16 | E |
| 15 | 1111 | 17 | F |

Nejsnadnější způsob převodu desítkového čísla do soustavy šestnáctkové je přes soustavu dvojkovou.

Dvojkové číslo rozdělíme na čtyřmístné skupiny a to zleva do prava a každou skupinu vyjádříme šestnáctkovým číslem.

Příklad:

$$(709)_{10} = (1011000101)_2 = (0010\ 1100\ 0101)_2 = (2C5)_{16}$$

Převod do soustavy desítkové se převádí takto:

$$(2C5)_{16} = 2 \cdot 16^2 + 12 \cdot 16^1 + 5 \cdot 16^0$$

2. BOOLEOVA ALGEBRA

Booleova algebra je zvláštní matematický způsob popisu chování i struktury logických obvodů. Tato algebra pracuje s log. proměnnými, které mohou nabývat pouze dvou hodnot, log. Hodnoty '0' nebo log. hodnoty '1'.

2.1. Výroková logika a logické funkce

Logický obvod je tvořen skupinou vzájemně spojených logických členů, o kterých lze říci, že jsou zařízení uskutečňující log. funkci. S log. funkcemi se můžeme setkat ve výrokové logice, která je součástí matematické logiky.

Základním pojmem výrokové logiky je výrok, což je každé tvrzení, o kterém má smysl prohlásit, zda je pravdivé nebo nepravdivé.

Logická funkce je předpis, který kombinaci, popř. i sledu hodnot jedné nebo více (nezávislých) log. proměnných jednoznačně přiřazuje hodnoty jedné (závislé) log. proměnné.

Negace je takovou funkcí jedné proměnné, u které má závisle proměnná vždy opačnou hodnotu, než nezávisle proměnná. Algebraické vyjádření této funkce je: $A = \bar{B}$

Disjunkce nebo-li logický součet je takovou funkcí dvou proměnných A, B, že závisle proměnná Y nabývá hodnoty 1 tehdy, je-li A nebo B nebo A i B současně rovno 1.

Algebraické vyjádření této funkce je: $Y = A + B$

Konjunkce nebo-li logický součin je takovou funkcí dvou proměnných A, B, že závisle proměnná Y nabývá hodnoty 1 pouze tehdy, mají-li současně A i B hodnotu 1. V ostatních případech nabývá proměnná Y hodnoty 0.

Algebraické vyjádření této funkce je: $Y = A \cdot B$

Shefferova funkce (NAND) je takovou funkcí dvou proměnných A, B, že závisle proměnná Y nabývá hodnoty 0 pouze tehdy, mají-li současně A i B hodnotu 1. V ostatních případech nabývá proměnná Y hodnoty 1. Algebraické vyjádření této funkce je: $Y = \overline{A \cdot B}$

Pierceova funkce (NOR) je takovou funkcí dvou proměnných A, B, že závisle proměnná Y nabývá hodnoty 1 pouze tehdy, mají-li současně A i B hodnotu 0. V ostatních případech nabývá proměnná Y hodnoty 0. Algebraické vyjádření této funkce je: $Y = \overline{A + B}$

Přehled základních kombinačních logických funkcí je uveden v tabulce 2.

2.2. Základní pravidla a zákony Booleovy algebry.

Booleova algebra je dvouhodnotová logická algebra, používající log. součtu, součinu a negace jako úplného systému základních logických funkcí. Používá se k úpravě a zjednodušování (minimalizaci) logických funkcí. Obsahuje následující zákony a pravidla:

1. Pravidlo agresivnosti a neutrálnosti hodnot 0 a 1

$$\begin{array}{ll} X + 1 = 1 & X + 0 = X \\ X \cdot 0 = 0 & X \cdot 1 = X \end{array}$$

2. Pravidlo komutativní

$$X + Y = Y + X \qquad X \cdot Y = Y \cdot X$$

3. Pravidlo asociativní

$$\begin{array}{l} X + (Y + Z) = (X + Y) + Z = X + Y + Z \\ X \cdot (Y \cdot Z) = (X \cdot Y) \cdot Z = X \cdot Y \cdot Z \end{array}$$

4. Pravidlo distributivní

$$X + (Y \cdot Z) = (X + Y) \cdot (X + Z) \qquad X \cdot (Y + Z) = X \cdot Y + X \cdot Z$$

5. Pravidlo absorpce

$$\begin{array}{ll} X + X = X & X \cdot X = X \\ X + X \cdot Y = X & X \cdot (X + Y) = X \end{array}$$

6. Pravidlo absorpce negace

$$\begin{array}{ll} X + \bar{X} \cdot Y = X + Y & X \cdot (\bar{X} + Y) = X \cdot Y \\ \bar{X} + X \cdot Y = \bar{X} + Y & \bar{X} \cdot (X + Y) = \bar{X} \cdot Y \end{array}$$

7. Pravidlo o vyloučeném třetím

$$X + \bar{X} = 1 \qquad X \cdot \bar{X} = 0$$

8. Pravidlo dvojité negace

$$\overline{\bar{X}} = X \qquad \overline{\overline{X + Y}} = X + Y \qquad \overline{\overline{X \cdot Y}} = X \cdot Y$$

9. Pravidla o vytvoření negace - De Morganovy zákony

$$\overline{X + Y} = \bar{X} \cdot \bar{Y} \qquad \overline{X \cdot Y} = \bar{X} + \bar{Y}$$

Tab. 2: Přehled kombinačních logických funkcí.

| Logický člen | Logická funkce | Pravdivostní tabulka | Schematická značka | Schematická značka DIN (SW - Multisim) | Schematická značka ANSI (SW - Multisim) | | | | | | | | | | | | | | | |
|------------------------|----------------------------|---|--------------------|--|---|---|---|---|---|---|---|---|---|---|---|---|---|--|--|--|
| Opakovač | $Y = A$ | <table border="1"> <tr><th>A</th><th>Y</th></tr> <tr><td>0</td><td>0</td></tr> <tr><td>1</td><td>1</td></tr> </table> | A | Y | 0 | 0 | 1 | 1 | | | | | | | | | | | | |
| A | Y | | | | | | | | | | | | | | | | | | | |
| 0 | 0 | | | | | | | | | | | | | | | | | | | |
| 1 | 1 | | | | | | | | | | | | | | | | | | | |
| Negace NOT | $Y = \bar{A}$ | <table border="1"> <tr><th>A</th><th>Y</th></tr> <tr><td>0</td><td>1</td></tr> <tr><td>1</td><td>0</td></tr> </table> | A | Y | 0 | 1 | 1 | 0 | | | | | | | | | | | | |
| A | Y | | | | | | | | | | | | | | | | | | | |
| 0 | 1 | | | | | | | | | | | | | | | | | | | |
| 1 | 0 | | | | | | | | | | | | | | | | | | | |
| Konjunkce AND | $Y = A \cdot B$ | <table border="1"> <tr><th>A</th><th>B</th><th>Y</th></tr> <tr><td>0</td><td>0</td><td>0</td></tr> <tr><td>0</td><td>1</td><td>0</td></tr> <tr><td>1</td><td>0</td><td>0</td></tr> <tr><td>1</td><td>1</td><td>1</td></tr> </table> | A | B | Y | 0 | 0 | 0 | 0 | 1 | 0 | 1 | 0 | 0 | 1 | 1 | 1 | | | |
| A | B | Y | | | | | | | | | | | | | | | | | | |
| 0 | 0 | 0 | | | | | | | | | | | | | | | | | | |
| 0 | 1 | 0 | | | | | | | | | | | | | | | | | | |
| 1 | 0 | 0 | | | | | | | | | | | | | | | | | | |
| 1 | 1 | 1 | | | | | | | | | | | | | | | | | | |
| Disjunkce OR | $Y = A + B$ | <table border="1"> <tr><th>A</th><th>B</th><th>Y</th></tr> <tr><td>0</td><td>0</td><td>0</td></tr> <tr><td>0</td><td>1</td><td>1</td></tr> <tr><td>1</td><td>0</td><td>1</td></tr> <tr><td>1</td><td>1</td><td>1</td></tr> </table> | A | B | Y | 0 | 0 | 0 | 0 | 1 | 1 | 1 | 0 | 1 | 1 | 1 | 1 | | | |
| A | B | Y | | | | | | | | | | | | | | | | | | |
| 0 | 0 | 0 | | | | | | | | | | | | | | | | | | |
| 0 | 1 | 1 | | | | | | | | | | | | | | | | | | |
| 1 | 0 | 1 | | | | | | | | | | | | | | | | | | |
| 1 | 1 | 1 | | | | | | | | | | | | | | | | | | |
| Shefferova funkce NAND | $Y = \overline{A \cdot B}$ | <table border="1"> <tr><th>A</th><th>B</th><th>Y</th></tr> <tr><td>0</td><td>0</td><td>1</td></tr> <tr><td>0</td><td>1</td><td>1</td></tr> <tr><td>1</td><td>0</td><td>1</td></tr> <tr><td>1</td><td>1</td><td>0</td></tr> </table> | A | B | Y | 0 | 0 | 1 | 0 | 1 | 1 | 1 | 0 | 1 | 1 | 1 | 0 | | | |
| A | B | Y | | | | | | | | | | | | | | | | | | |
| 0 | 0 | 1 | | | | | | | | | | | | | | | | | | |
| 0 | 1 | 1 | | | | | | | | | | | | | | | | | | |
| 1 | 0 | 1 | | | | | | | | | | | | | | | | | | |
| 1 | 1 | 0 | | | | | | | | | | | | | | | | | | |
| Pierceova funkce NOR | $Y = \overline{A + B}$ | <table border="1"> <tr><th>A</th><th>B</th><th>Y</th></tr> <tr><td>0</td><td>0</td><td>1</td></tr> <tr><td>0</td><td>1</td><td>0</td></tr> <tr><td>1</td><td>0</td><td>0</td></tr> <tr><td>1</td><td>1</td><td>0</td></tr> </table> | A | B | Y | 0 | 0 | 1 | 0 | 1 | 0 | 1 | 0 | 0 | 1 | 1 | 0 | | | |
| A | B | Y | | | | | | | | | | | | | | | | | | |
| 0 | 0 | 1 | | | | | | | | | | | | | | | | | | |
| 0 | 1 | 0 | | | | | | | | | | | | | | | | | | |
| 1 | 0 | 0 | | | | | | | | | | | | | | | | | | |
| 1 | 1 | 0 | | | | | | | | | | | | | | | | | | |

2.3. Pravdivostní tabulka

Tabulka č.3:

Důkaz De-Morganova zákona tabulkovou metodou

| A | B | $A + B$ | $\overline{A + B}$ | \bar{A} | \bar{B} | $\overline{A \cdot B}$ |
|---|---|---------|--------------------|-----------|-----------|------------------------|
| 0 | 0 | 0 | 1 | 1 | 1 | 1 |
| 0 | 1 | 1 | 0 | 1 | 0 | 0 |
| 1 | 0 | 1 | 0 | 0 | 1 | 0 |
| 1 | 1 | 1 | 0 | 0 | 0 | 0 |

Pozn.:

Pro n proměnných má tabulka 2^n kombinací

Tabulka stavů je tabulka, která jednoznačně přiřazuje vstupní proměnné k jedné nebo několika výstupním proměnným. Má dvě části. Levá část zahrnuje všechny možné kombinace hodnot vstupních proměnných, pravá část obsahuje výslednou hodnotu výrazu pro každou kombinaci

hodnot z levé části. Velikost levé části závisí na počtu vstupních proměnných, neboť počet řádků v tabulce je roven 2^n . Kombinace metodou vstupních proměnných je nejlépe vyjádřit pomocí dvojkových čísel. Tabulku stavů používáme nejen pro snadné popsání činnosti log. obvodu, ale můžeme ji použít i pro zjištění, jsou-li dva výrazy shodné. Příklad pro důkaz De Morganova zákona je uveden v tabulce 3. V levé části jsou uvedeny vstupní proměnné A, B a ve výstupní části je názorně ukázán celý postup důkazu.

2.4. Minimalizace

Při návrhu log. obvodu dbáme na to, aby jej bylo možno realizovat pokud možno co nejmenším počtem co nejjednodušších logických členů. To znamená, že algebraický výraz má být složen z co nejmenšího počtu členů, z nichž každý obsahuje nejmenší počet proměnných. Postup pomocí kterého nahradíme složitější algebraický výraz jednodušším výrazem, nazýváme minimalizací.

Způsob úpravy algebraického výrazu, který využívá všech známých zákonů a pravidel Booleovy algebry se nazývá **přímá minimalizace**. Nyní si na několika příkladech ukážeme zjednodušování výrazů.

Příklad 1:

$$\begin{aligned} Z &= \overline{A}\overline{B}\overline{C}D + \overline{A}B\overline{C}D + A\overline{B}\overline{C}D + \overline{A}BCD \\ Z &= BD(\overline{A}\overline{C} + \overline{A}C + AC + \overline{A}C) \\ Z &= BD[\overline{C}(A + \overline{A}) + C(A + \overline{A})] \\ Z &= BD(C + \overline{C}) \\ Z &= BD \end{aligned}$$

Příklad 2:

$$\begin{aligned} Z &= \overline{X} + XY \\ Z &= \overline{X}(Y + \overline{Y}) + XY \\ Z &= \overline{X}Y + \overline{X}\overline{Y} + XY \\ Z &= \overline{X}Y + \overline{X}\overline{Y} + \overline{X}\overline{Y} + XY \\ Z &= \overline{X}(Y + \overline{Y}) + Y(X + \overline{X}) \\ Z &= \overline{X} + Y \end{aligned}$$

Příklad 3:

$$\begin{aligned} Z &= (A + B)(A + C) \\ Z &= A + AB + AC + BC \\ Z &= A(1 + B + C) + BC \\ Z &= A + BC \end{aligned}$$

Příklad 4:

$$\begin{aligned} Z &= \overline{A}BC + \overline{A}\overline{B} + \overline{A} \cdot \overline{B} + ABC \\ Z &= BC(A + \overline{A}) + \overline{B}(A + A) \\ Z &= BC + \overline{B} \quad (\text{pravidlo absorpce negace}) \\ Z &= \overline{B} + C \end{aligned}$$

2.5. Zápis funkce z pravdivostní tabulky

Co je to pravdivostní tabulka, jsme si již popsali dříve. Nyní si ukážeme, jak pomocí této tabulky budeme řešit určitý problém a jak z ní sestavíme log. funkci.

Zadání:

Sestavte log. funkci pro ovládání žárovky ze dvou míst (případ schodišťového přepínače)

Řešení:

Nejprve sestavíme pravdivostní tabulku pro dvě proměnné, kde spínače budou označeny **A**, **B** a žárovka **Z**.

| A | B | Z |
|---|---|---|
| 0 | 0 | 0 |
| 0 | 1 | 1 |
| 1 | 0 | 1 |
| 1 | 1 | 0 |

Z tabulky zapíšeme logickou funkci:

$$\begin{aligned} Z &= \overline{A} \cdot B + A \cdot \overline{B} && (\text{součet součinů}) \\ \text{nebo} \quad Z &= (A + B) \cdot (\overline{A} + \overline{B}) && (\text{součin součtů}) \end{aligned}$$

Algebraický výraz můžeme dostat z tabulky ve dvou tvarech. A to buď jako (1) součet součinů nebo jako (2) součin součtů.

- (1) Výraz ve tvaru **součtu součinů** dostaneme, napíšeme-li pro každé políčko, které obsahuje 1 pro výstupní proměnnou, kombinaci odpovídajících vstupních proměnných ve formě součinu. Tyto jednotlivé součiny pak spojíme v součet.
- (2) Výraz ve tvaru **součinu součtů** dostaneme, napíšeme-li pro každé políčko, které obsahuje 0

pro výstupní proměnnou, kombinaci odpovídajících vstupních proměnných v podobě součtu. Tyto součty pak spojíme v součín.

Příklad 4: Z následující tabulky sestavte log. funkci.

| A | B | C | Z |
|---|---|---|---|
| 0 | 0 | 0 | 0 |
| 0 | 0 | 1 | 1 |
| 0 | 1 | 0 | 0 |
| 0 | 1 | 1 | 1 |
| 1 | 0 | 0 | 1 |
| 1 | 0 | 1 | 0 |
| 1 | 1 | 0 | 0 |
| 1 | 1 | 1 | 0 |

Řešení:

$$Z = \bar{A} \cdot \bar{B} \cdot C + \bar{A} \cdot B \cdot C + A \cdot \bar{B} \cdot \bar{C}$$

tuto funkci můžeme dále zjednodušit:

$$Z = \bar{A} \cdot C \cdot (B + \bar{B}) + A \cdot \bar{B} \cdot \bar{C}$$

$$Z = \bar{A} \cdot C + A \cdot \bar{B} \cdot \bar{C}$$

Často také musíme řešit problém opačný, a to sestavení tabulky z logické funkce.

Příklad 5:

| A | B | C | Z |
|---|---|---|---|
| 0 | 0 | 0 | 0 |
| 0 | 0 | 1 | 1 |
| 0 | 1 | 0 | 0 |
| 0 | 1 | 1 | 0 |
| 1 | 0 | 0 | 1 |
| 1 | 0 | 1 | 0 |
| 1 | 1 | 0 | 0 |
| 1 | 1 | 1 | 1 |

Z dané logické funkce sestavte tabulku stavů:

$$Z = A \cdot B \cdot C + A \cdot \bar{B} \cdot \bar{C} + \bar{A} \cdot \bar{B} \cdot C$$

$$1 \cdot 1 \cdot 1 + 1 \cdot 0 \cdot 0 + 0 \cdot 0 \cdot 1$$

Protože se jedná o součet součinů, musí se každý součin rovnat 1. Najdeme tedy odpovídající kombinace vstupních proměnných a k výstupní proměnné napíšeme 1. K ostatním kombinacím napíšeme nuly.

Příklad 6:

| A | B | C | Z |
|---|---|---|---|
| 0 | 0 | 0 | 1 |
| 0 | 0 | 1 | 0 |
| 0 | 1 | 0 | 0 |
| 0 | 1 | 1 | 1 |
| 1 | 0 | 0 | 1 |
| 1 | 0 | 1 | 0 |
| 1 | 1 | 0 | 0 |
| 1 | 1 | 1 | 0 |

Z dané logické funkce sestavte tabulku stavů:

$$Z = \bar{A} \cdot B \cdot C + \bar{B} \cdot \bar{C}$$

Funkci můžeme také rozepsat takto:

$$Z = \bar{A} \cdot B \cdot C + \bar{B} \cdot \bar{C} \cdot (A + \bar{A})$$

$$Z = \bar{A} \cdot B \cdot C + A \cdot \bar{B} \cdot \bar{C} + \bar{A} \cdot \bar{B} \cdot \bar{C}$$

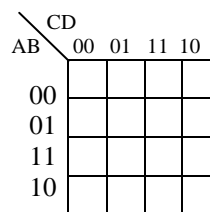
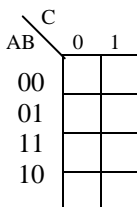
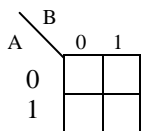
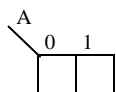
$$0 \cdot 1 \cdot 1 + 1 \cdot 0 \cdot 0 + 0 \cdot 0 \cdot 0$$

2.6. Karnaughova mapa

Pro zobrazení log. funkce se kromě tabulky stavů používá také Karnaughova mapa. Slouží hlavně pro minimalizaci funkce. Její nevýhodou je, že ji můžeme použít nejvýše pro čtyři proměnné. Použití pro více proměnných je již poměrně složité.

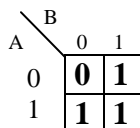
Na rozdíl od tabulky stavů se hodnoty vstupních proměnných zapisují po straně mapy a nad mapou a výstupní proměnná se zapisuje do políček mapy (tab. 4).

Tab.4: Karnaughova mapa pro jednu, dvě, tři a čtyři proměnné



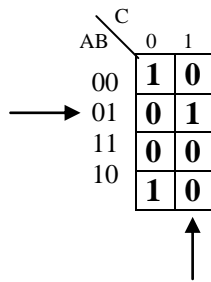
Mapa funkce dvou proměnných má čtyři políčka. Jako příklad si můžeme uvést tabulku stavů pro logickou funkci OR a k této tabulce odpovídající Karnaughovu mapu.

| A | B | Z |
|---|---|---|
| 0 | 0 | 0 |
| 0 | 1 | 1 |
| 1 | 0 | 1 |
| 1 | 1 | 1 |



Mapa funkce tří proměnných má osm políček, stejně jako má osm řádků tabulka pro tři proměnné. Pořadí sloupců v mapě neodpovídá pořadí, v němž píšeme řádky tabulky. To proto, aby se v sousedních sloupcích změnila hodnota pouze jedné proměnné. Důvodem je minimalizace z mapy.

| A | B | C | Z |
|---|---|---|---|
| 0 | 0 | 0 | 1 |
| 0 | 0 | 1 | 0 |
| 0 | 1 | 0 | 0 |
| 0 | 1 | 1 | 1 |
| 1 | 0 | 0 | 1 |
| 1 | 0 | 1 | 0 |
| 1 | 1 | 0 | 0 |
| 1 | 1 | 1 | 0 |

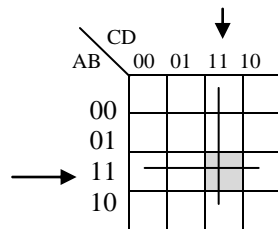


V tabulce je uvedena ukázka jak určit polohu příslušného políčka pro dané hodnoty proměnných:

$$A=0, B=1, C=1$$

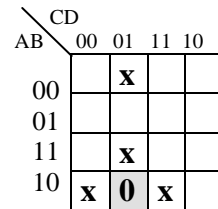
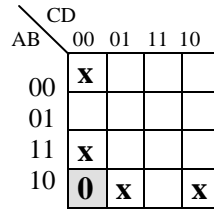
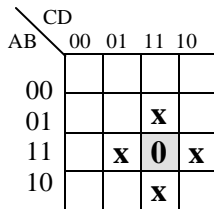
Mapa funkce čtyř proměnných má šestnáct políček rozdělených do čtyř řádků a čtyř sloupců. Pořadí řádků a sloupců je stejné: 00, 01, 11, 10. Je to opět proto, aby se v sousedních řádcích a sousedních sloupcích změnila hodnota pouze jedné proměnné. Polohu políčka na mapě čtyř proměnných určujeme stejným způsobem, jako jsme ji určovali na mapě pro jiný počet proměnných.

| A | B | C | D | Z |
|---|---|---|---|---|
| 0 | 0 | 0 | 0 | 1 |
| 0 | 0 | 0 | 1 | 0 |
| 0 | 0 | 1 | 0 | 1 |
| 0 | 0 | 1 | 1 | 1 |
| 0 | 1 | 0 | 0 | 0 |
| 0 | 1 | 0 | 1 | 0 |
| 0 | 1 | 1 | 0 | 1 |
| 0 | 1 | 1 | 1 | 1 |
| 1 | 0 | 0 | 0 | 1 |
| 1 | 0 | 0 | 1 | 1 |
| 1 | 0 | 1 | 0 | 0 |
| 1 | 0 | 1 | 1 | 1 |
| 1 | 1 | 0 | 0 | 1 |
| 1 | 1 | 0 | 1 | 1 |
| 1 | 1 | 1 | 0 | 1 |
| 1 | 1 | 1 | 1 | 1 |



Na mapě je určena poloha políčka pro hodnoty proměnných: $A=1, B=1, C=1, D=1$

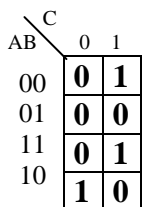
Příklady odvození **sousedních** políček pro políčko zadané:



2.7. Zapis algebraického výrazu funkce z Karnughovy mapy

Z mapy lze přímo určit odpovídající výraz ve formě součtu součinů nebo součinu součtů, přičemž se budeme snažit o co nejjednodušší a nejhodnější zápis.

Příklad 7: Z dané mapy určete algebraický výraz ve tvaru součtu součinů.



Zápis je obdobný jako z tabulky stavů:

$$Z = \bar{A} \cdot \bar{B} \cdot C + A \cdot B \cdot C + A \cdot \bar{B} \cdot \bar{C}$$

$$0 \cdot 0 \cdot 1 + 1 \cdot 1 \cdot 1 + 1 \cdot 0 \cdot 0$$

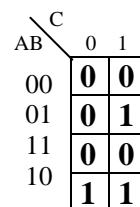
2.8. Sestavení Karnughovy mapy z algebraického výrazu

Příklad 8: Nakreslete mapu a zapište do ní hodnoty fce Z, která je dána vztahem: $Z = \bar{A}BC + A\bar{B}$

Řešení:

$$Z = \bar{A}BC + A\bar{B}(C + \bar{C}) = \bar{A}BC + A\bar{B}C + A\bar{B}\bar{C}$$

$$0 \cdot 1 \cdot 1 + 1 \cdot 0 \cdot 1 + 1 \cdot 0 \cdot 0$$



2.9. Minimalizace pomocí Karnaughovy mapy

| | | |
|----|---|---|
| | C | |
| AB | 0 | 1 |
| 00 | | |
| 01 | | |
| 11 | | 1 |
| 10 | | 1 |

Na uvedené mapě jsou zapsány dvě jedničky v sousedních políčkách, která mají tu vlastnost, že různé hodnoty má pouze proměnná **B**. Napíšeme nyní výraz za obě jedničky:

$$Z = \overline{A}BC + ABC$$

Vidíme, že výraz se skládá ze dvou členů a lze jej dále zjednodušit.

$$Z = AC(B + \overline{B})$$

$$Z = AC$$

Pokusme se nyní sestavit obecné pravidlo, které by určilo, jakým způsobem můžeme za sousední políčka v mapě napsat přímo jeden člen. Jak již bylo uvedeno, mění se při přechodu mezi sousedními políčky jedna proměnná - v našem případě proměnná **B**. Proto tento člen nezávisí na této proměnné a je určen jen ostatními proměnnými, v našem případě výrazem **AC**.

Toto pravidlo platí i pro větší počet políček.

Při minimalizaci pomocí Karnaughovy mapy volíme následující postup:

1. **Jedničky v mapě uzavíráme pomocí smyček, které obsahují dvě, čtyři nebo osm sousedních políček. Přitom je třeba mít na paměti, že čím větší smyčku vytvoříme, tím je odpovídající výraz jednodušší.**
2. **Smyčky se mohou navzájem protínat.**
3. **Za každou smyčku zapíšeme pouze jeden výraz, který neobsahuje ty proměnné, jejichž hranice smyčku protínají.**
4. **Algebraické výrazy za jednotlivé smyčky sečteme.**

Závěrem lze říci, že smyčky s jedničkami odpovídají výrazu ve tvaru součtu součinů. Obdobným způsobem můžeme vytvářet i smyčky s nulami. V tom případě však získáme vztah ve tvaru součinu součtů.

Nyní si celý postup minimalizace pomocí Karnaughovy mapy objasníme na několika příkladech, kde bude naším úkolem napsat algebraické výrazy v obou tvarech pro uvedené mapy.

Příklad 9:

a) součin součtů:

| | | | |
|----|---|---|------------------------|
| | C | | |
| AB | 0 | 1 | $Z = A + \overline{C}$ |
| 00 | 1 | 0 | |
| 01 | 1 | 0 | |
| 11 | 1 | 1 | |
| 10 | 1 | 1 | |

b) součet součinů:

| | | | |
|----|---|---|---|
| | C | | |
| AB | 0 | 1 | |
| 00 | 1 | 0 | → V této části mapy se mění stavy proměnných A a B . Tyto proměnné vyloučíme a zůstane proměnná C (ovšem negovaná) |
| 01 | 1 | 0 | |
| 11 | 1 | 1 | → V této části se mění stavy proměnných B a C . Proto se tyto proměnné vyloučí a zůstane proměnná A . |
| 10 | 1 | 1 | |

V této části se mění stavy proměnných **B** a **C**. Proto se tyto proměnné vyloučí a zůstane proměnná **A**.

Výsledný zápis je opět: $Z = A + \overline{C}$

Příklad 10: sestavte minimalizovaný výraz za pomoci Karnaughovy mapy.

| | | | | |
|----|----|----|----|----|
| | CD | | | |
| AB | 00 | 01 | 11 | 10 |
| 00 | 0 | 0 | 1 | 0 |
| 01 | 0 | 1 | 1 | 0 |
| 11 | 0 | 1 | 1 | 0 |
| 10 | 0 | 1 | 1 | 0 |

Řešení:

| | | | | | |
|----|----|----|----|----|----|
| | | CD | | | |
| | AB | 00 | 01 | 11 | 10 |
| 00 | | 0 | 0 | 1 | 0 |
| 01 | | 0 | 1 | 1 | 0 |
| 11 | | 0 | 1 | 1 | 0 |
| 10 | | 0 | 1 | 1 | 0 |

V této části tabulky se mění stavy proměnných A a C. Proto se vyloučí a zůstane proměnná B a D.

| | | | | | |
|----|----|----|----|----|----|
| | | CD | | | |
| | AB | 00 | 01 | 11 | 10 |
| 00 | | 0 | 0 | 1 | 0 |
| 01 | | 0 | 1 | 1 | 0 |
| 11 | | 0 | 1 | 1 | 0 |
| 10 | | 0 | 1 | 1 | 0 |

Zde se mění stavy proměnné B a C. Po vyloučení těchto proměnných zůstanou proměnné A a D.

| | | | | | |
|----|----|----|----|----|----|
| | | CD | | | |
| | AB | 00 | 01 | 11 | 10 |
| 00 | | 0 | 0 | 1 | 0 |
| 01 | | 0 | 1 | 1 | 0 |
| 11 | | 0 | 1 | 1 | 0 |
| 10 | | 0 | 1 | 1 | 0 |

Zde se mění stavy proměnných A a B. Po jejich vyloučení zůstanou proměnné C a D.

Výsledný výraz je dán součtem jednotlivých částí tabulky:

$$Z = BD + AD + CD$$

Sestavení výrazu pomocí součinu součtů:

| | | | | | |
|----|----|----|----|----|----|
| | | CD | | | |
| | AB | 00 | 01 | 11 | 10 |
| 00 | | 0 | 0 | 1 | 0 |
| 01 | | 0 | 1 | 1 | 0 |
| 11 | | 0 | 1 | 1 | 0 |
| 10 | | 0 | 1 | 1 | 0 |

| | | | | | |
|----|----|----|----|----|----|
| | | CD | | | |
| | AB | 00 | 01 | 11 | 10 |
| 00 | | 0 | 0 | 1 | 0 |
| 01 | | 0 | 1 | 1 | 0 |
| 11 | | 0 | 1 | 1 | 0 |
| 10 | | 0 | 1 | 1 | 0 |

$$Z = D(A + B + C)$$

Příklad 11: sestavte minimalizovaný výraz za pomoci Karnaughovy mapy metodou součtu součinů:

| | | | | | |
|----|----|----|----|----|----|
| | | CD | | | |
| | AB | 00 | 01 | 11 | 10 |
| 00 | | 1 | 1 | 1 | 1 |
| 01 | | 1 | 1 | 1 | 0 |
| 11 | | 0 | 0 | 1 | 0 |
| 10 | | 1 | 0 | 1 | 1 |

| | | | | | |
|----|----|----|----|----|----|
| | | CD | | | |
| | AB | 00 | 01 | 11 | 10 |
| 00 | | 1 | 1 | 1 | 1 |
| 01 | | 1 | 1 | 1 | 0 |
| 11 | | 0 | 0 | 1 | 0 |
| 10 | | 1 | 0 | 1 | 1 |

| | | | | | |
|----|----|----|----|----|----|
| | | CD | | | |
| | AB | 00 | 01 | 11 | 10 |
| 00 | | 1 | 1 | 1 | 1 |
| 01 | | 1 | 1 | 1 | 0 |
| 11 | | 0 | 0 | 1 | 0 |
| 10 | | 1 | 0 | 1 | 1 |

Výsledný výraz je dán součtem jednotlivých částí tabulky:

$$Z = \bar{A} \cdot \bar{C} + CD + \bar{B} \cdot \bar{D}$$

metodou součinu součtů:

| | | | | | |
|----|----|----|----|----|----|
| | | CD | | | |
| | AB | 00 | 01 | 11 | 10 |
| 00 | | 1 | 1 | 1 | 1 |
| 01 | | 1 | 1 | 1 | 0 |
| 11 | | 0 | 0 | 1 | 0 |
| 10 | | 1 | 0 | 1 | 1 |

| | | | | | |
|----|----|----|----|----|----|
| | | CD | | | |
| | AB | 00 | 01 | 11 | 10 |
| 00 | | 1 | 1 | 1 | 1 |
| 01 | | 1 | 1 | 1 | 0 |
| 11 | | 0 | 0 | 1 | 0 |
| 10 | | 1 | 0 | 1 | 1 |

| | | | | | |
|----|----|----|----|----|----|
| | | CD | | | |
| | AB | 00 | 01 | 11 | 10 |
| 00 | | 1 | 1 | 1 | 1 |
| 01 | | 1 | 1 | 1 | 0 |
| 11 | | 0 | 0 | 1 | 0 |
| 10 | | 1 | 0 | 1 | 1 |

Výsledný výraz je dán součinem jednotlivých částí tabulky:

$$Z = (\bar{A} + \bar{B} + C) \cdot (\bar{B} + \bar{C} + D) \cdot (\bar{A} + C + \bar{D})$$

3. SCHÉMA LOGICKÉHO OBVODU

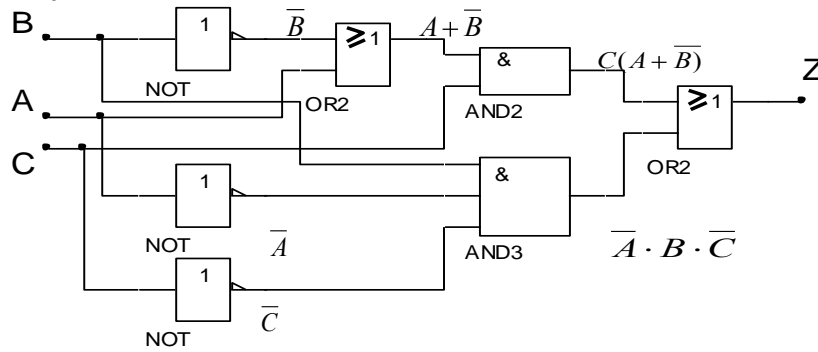
Na základě znalostí algebraického výrazu, který popisuje činnost určitého log. obvodu, můžeme nakreslit jeho schéma.

Způsob kreslení si ukážeme na příkladě č. 12.

Příklad 12: Nakreslete blokové schéma obvodu, jehož činnost je vyjádřena vztahem:

$$Z = (A + \bar{B}) \cdot C + \bar{A} \cdot B \cdot \bar{C}$$

Řešení je následující:



3.1. Realizace obvodu Shefferovou funkcí

V praxi se velmi často setkáváme s log. členy uskutečňujícími Shefferovu a Pierceovu funkci především proto, že jejich realizace je technickými prostředky snadná. Tyto funkce tvoří základ Shefferovy a Pierceovy algebry. Dále si ukážeme, jak je třeba postupovat, abychom získali schéma obvodu, který by byl složen jen ze členů uskutečňujících Shefferovu funkci. Celý postup bude spočívat ve vhodných úpravách následujícího výrazu.

Příklad 13: Upravte následující výraz na výraz vhodný pro realizaci pomocí Shefferových členů.

$$Z = CD + \bar{C} \cdot \bar{A} + \bar{A}BC$$

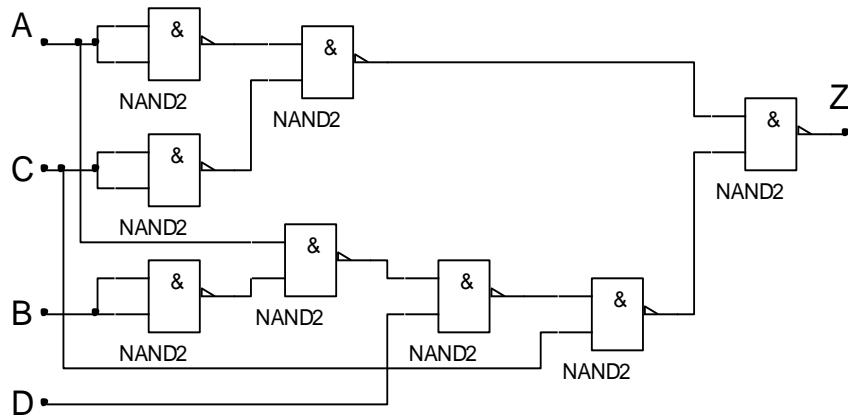
$$Z = \bar{A} \cdot \bar{C} + C(\bar{D} + A \cdot \bar{B})$$

$$Z = \bar{A} \cdot \bar{C} + C \cdot \bar{D} \cdot A \cdot \bar{B}$$

$$Z = \bar{A} \cdot \bar{C} + C \cdot D \cdot A \cdot \bar{B}$$

$$Z = \bar{A} \cdot \bar{C} + C \cdot D \cdot A \cdot \bar{B}$$

$$Z = \bar{A} \cdot \bar{C} \cdot C \cdot D \cdot A \cdot \bar{B}$$



3.2. Realizace obvodu Pierceovou funkcí

Pierceův člen nám uskutečňuje negaci disjunkce nebo negaci. Abychom mohli algebraický výraz vyjádřený ve tvaru součtu součinů nebo součinu součtů realizovat Pierceovými členy, musíme jej upravit na postupné negace disjunkcí. Způsob úpravy si ukážeme na příkladě.

Příklad 14: Upravte následující výraz na výraz vhodný pro realizaci pomocí Pierceových členů.

$$Z = CD + \bar{C} \cdot \bar{A} + ABC$$

$$Z = \bar{A} \cdot \bar{C} + C(\bar{D} + A \cdot \bar{B})$$

$$Z = \bar{A} \cdot \bar{C} + C(\bar{D} + A \cdot \bar{B})$$

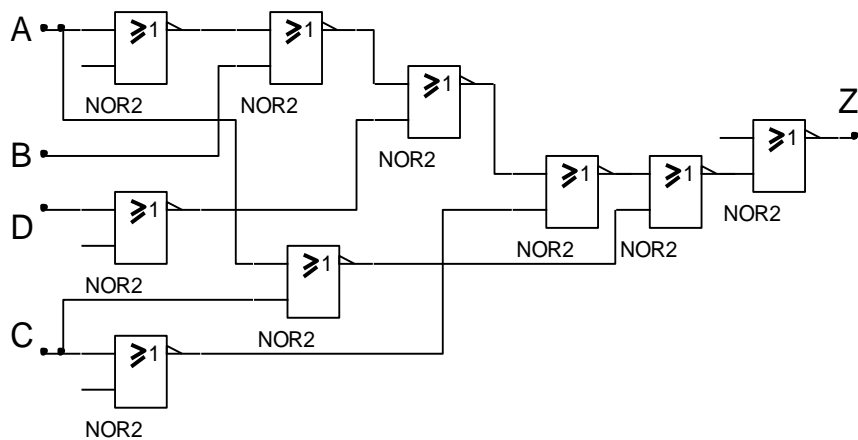
$$Z = \bar{A} \cdot \bar{C} + C(\bar{D} + \bar{A} + \bar{B})$$

$$Z = \bar{A} \cdot \bar{C} + C(\bar{D} + \bar{A} + \bar{B})$$

$$Z = \bar{A} \cdot \bar{C} + C(\bar{D} + \bar{A} + \bar{B})$$

$$Z = \bar{A} + C + \bar{C} + \bar{D} + \bar{A} + \bar{B}$$

$$Z = \bar{A} + C + \bar{C} + \bar{D} + \bar{A} + \bar{B}$$



4. NÁVRHY A ŘEŠENÍ KOMBINAČNÍCH LOGICKÝCH OBVODŮ

Při návrhu jednoduchých ovládacích obvodů volíme zpravidla tento postup:

1. Vyjádříme podmínky činnosti zařízení.
2. Sestavíme rovnice log. funkce a provedeme její minimalizaci.
3. Navrhujeme log. členy, které použijeme pro realizaci obvodu.
4. Nakreslíme schéma log. obvodu.

4.1. Návrh log. obvodu s jedním výstupem

Příklad 15: Navrhněte obvod pro ovládání žárovky ze dvou míst.

Je-li sepnut lichý počet spínačů, žárovka svítí (sepnutý spínač má hodnotu log. 1 - tj. žárovka svítí).

Řešení: Sestavíme tabulku stavů a Karnaughovu mapu.

| A | B | Z |
|---|---|---|
| 0 | 0 | 0 |
| 0 | 1 | 1 |
| 1 | 0 | 1 |
| 1 | 1 | 0 |

| | | | |
|---|---|---|---|
| | | B | |
| | | 0 | 1 |
| A | 0 | 0 | 1 |
| | 1 | 1 | 0 |

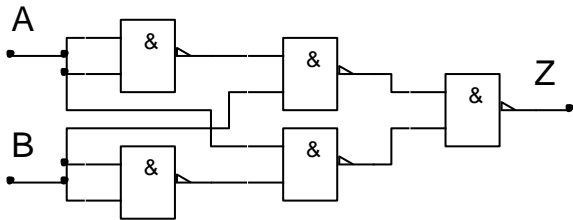
Z mapy vyplývá, že nemůžeme minimalizovat
Napišeme algebraický výraz:

$$Z = A\bar{B} + \bar{A}B$$

Daný výraz převedeme do úplné Schefferovy funkce:

$$Z = \overline{\overline{A\bar{B} + \bar{A}B}} = \overline{\overline{A\bar{B}} \cdot \overline{\bar{A}B}} = \overline{\overline{A} \cdot \overline{\bar{B}} \cdot \overline{\bar{A}} \cdot \overline{B}} = \overline{\overline{A} \cdot \overline{\bar{B}} \cdot \overline{\bar{A}} \cdot \overline{B}}$$

Z uvedené rovnice navrhujeme schéma obvodu:



Příklad 16: Navrhněte obvod (schodišťový vypínač) pro ovládání žárovky ze tří míst.

Je-li sepnut lichý počet spínačů, žárovka svítí (sepnutý spínač má hodnotu log. 1)

Řešení: Sestavíme tabulku stavů a Karnaughovu mapu.

| A | B | C | Z |
|---|---|---|---|
| 0 | 0 | 0 | 0 |
| 0 | 0 | 1 | 1 |
| 0 | 1 | 0 | 1 |
| 0 | 1 | 1 | 0 |
| 1 | 0 | 0 | 1 |
| 1 | 0 | 1 | 0 |
| 1 | 1 | 0 | 0 |
| 1 | 1 | 1 | 1 |

| | | | |
|----|----|---|---|
| | | C | |
| | | 0 | 1 |
| AB | 00 | 0 | 1 |
| | 01 | 1 | 0 |
| | 11 | 0 | 1 |
| | 10 | 1 | 0 |

Z mapy vyplývá, že nemůžeme minimalizovat.

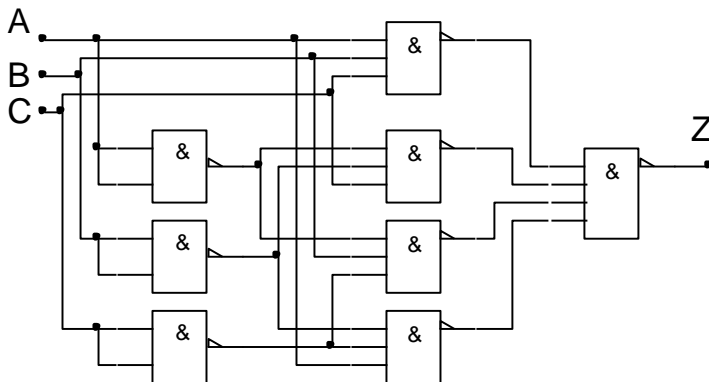
Napišeme algebraický výraz:

$$Z = \bar{A} \cdot \bar{B} \cdot C + \bar{A} \cdot B \cdot \bar{C} + ABC + A \cdot \bar{B} \cdot \bar{C}$$

Převod do Schefferovy funkce:

$$Z = \overline{\overline{\bar{A} \cdot \bar{B} \cdot C + \bar{A} \cdot B \cdot \bar{C} + ABC + A \cdot \bar{B} \cdot \bar{C}}}$$

$$Z = \overline{\overline{\bar{A} \cdot \bar{B} \cdot C} \cdot \overline{\bar{A} \cdot B \cdot \bar{C}} \cdot \overline{ABC} \cdot \overline{A \cdot \bar{B} \cdot \bar{C}}}$$

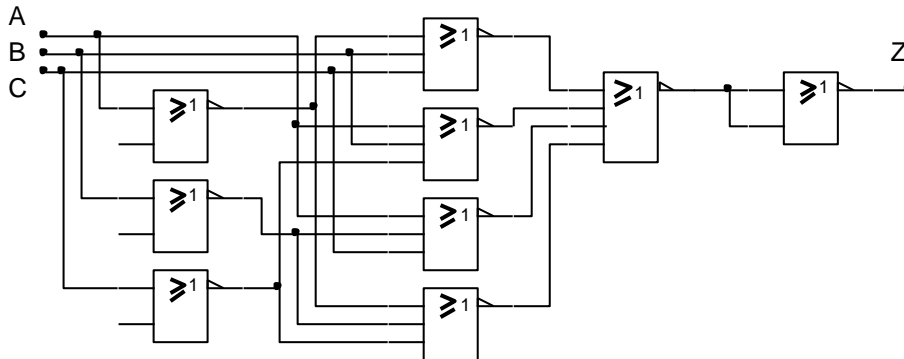


Převod do Pieceovy funkce:

$$Z = \overline{\overline{A \cdot B \cdot C} + \overline{A \cdot B \cdot \overline{C}} + \overline{A \cdot B \cdot C} + \overline{A \cdot \overline{B} \cdot \overline{C}}}$$

$$Z = \overline{A + B + \overline{C} + A + \overline{B} + C + \overline{A} + \overline{B} + \overline{C} + \overline{A} + B + C}$$

$$Z = \overline{\overline{A + B + \overline{C} + A + \overline{B} + C + \overline{A} + \overline{B} + \overline{C} + \overline{A} + B + C}}$$



4.2. Návrh logického obvodu s větším počtem výstupů

Při návrhu log. obvodů se často vyskytují obvody, které mají větší počet výstupů. Funkci, kterou musí realizovat navrhovaný obvod, budeme zadávat tabulkou stavů a tuto tabulku přepíšeme do Karnaughových map. Z nich získáme minimalizované výrazy, podle kterých následně navrhne schéma logických obvodů s jedním výstupem. Log. obvod s větším počtem výstupů pak dostaneme sloučením jednotlivých log. obvodů s jedním výstupem do jediného schématu. Jestliže ve schématech s jedním výstupem existuje log. člen, který realizuje stejnou funkci v několika z nich, pak ve výsledném schématu bude tento člen nakreslen pouze jednou.

Příklad 17: Navrhněte log. obvod, který má sčítat tři dvojková čísla o jednom řádu.

Řešení: Při aritmetickém sčítání tří dvojkových čísel o jednom řádu bude výsledek vyjádřen dvojkovým číslem o dvou řádech. Z tabulky stavů sestavíme Karnaughovy mapy pro jednotlivé výstupní proměnné a z nich napíšeme příslušné algebraické výrazy.

| A | B | C | Y | Z |
|---|---|---|---|---|
| 0 | 0 | 0 | 0 | 0 |
| 0 | 0 | 1 | 0 | 1 |
| 0 | 1 | 0 | 0 | 1 |
| 0 | 1 | 1 | 1 | 0 |
| 1 | 0 | 0 | 0 | 1 |
| 1 | 0 | 1 | 1 | 0 |
| 1 | 1 | 0 | 1 | 0 |
| 1 | 1 | 1 | 1 | 1 |

→ Zde algebraicky sčítáme dvojková čísla: 0+1=01
1+1=10
1+1+1=11

| | | C | |
|----|----|---|---|
| | | 0 | 1 |
| AB | 00 | 0 | 0 |
| | 01 | 0 | 1 |
| 11 | 11 | 1 | 1 |
| | 10 | 0 | 1 |

| | | C | |
|----|----|---|---|
| | | 0 | 1 |
| AB | 00 | 0 | 1 |
| | 01 | 1 | 0 |
| 11 | 11 | 0 | 1 |
| | 10 | 1 | 0 |

$$Y = AB + AC + BC$$

$$Z = \overline{A} \cdot \overline{B} \cdot C + \overline{A} \cdot B \cdot \overline{C} + ABC + A \cdot \overline{B} \cdot \overline{C}$$

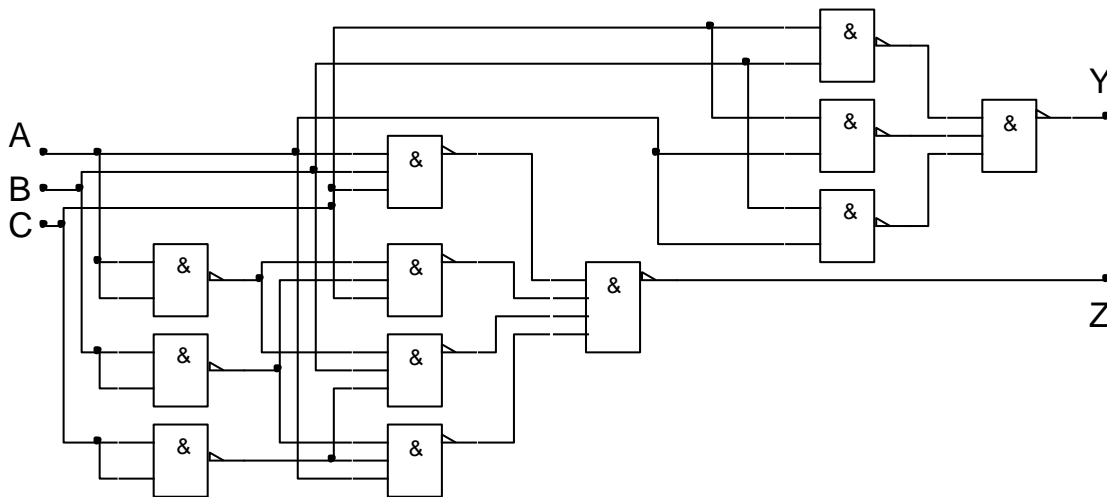
Převod do Shefferovy funkce:

$$Y = \overline{\overline{AB + AC + BC}}$$

$$Y = \overline{\overline{AB} \cdot \overline{AC} \cdot \overline{BC}}$$

$$Z = \overline{\overline{A \cdot \overline{B} \cdot C + \overline{A} \cdot B \cdot \overline{C} + ABC + A \cdot \overline{B} \cdot \overline{C}}}$$

$$Z = \overline{\overline{A \cdot \overline{B} \cdot C} \cdot \overline{\overline{A \cdot B \cdot \overline{C}}} \cdot \overline{ABC} \cdot \overline{A \cdot \overline{B} \cdot \overline{C}}}$$



4.3. Návrh obvodu zadaného neúplnou tabulkou stavů

Doposud jsme se setkali pouze s případy, kdy log. obvod byl zadán úplnou tabulkou stavů. Často se však vyskytne případ, že se některé kombinace vstupních proměnných nemohou na vstupu navrhovaného obvodu vyskytnout, nebo výstupní proměnná není pro danou kombinaci určena. V tom případě je v tabulce stavů na místě hodnoty výstupní proměnné pomlčka. Abychom mohli z této neúplné tabulky vytvořit algebraický výraz, musíme místo každé pomlčky napsat buď jedničku nebo nulu. Neúplnou tabulku stavů můžeme doplnit různým způsobem. Doplnění však provedeme tak, abychom dostali co nejjednodušší algebraický výraz. Proto je vhodné doplnit až Karnaughovu mapu se zřetelem na minimalizaci. Návrh log. obvodu zadaného neúplnou tabulkou stavů si vysvětlíme na příkladě.

Příklad 18:

Ve slévárně jsou tři pece a plní se postupně v libovolném pořadí. Při plnění pece vždy musíme otevřít její uzávěr. Navrhněte log. obvod, který z bezpečnostních důvodů signalizuje otevření uzávěru.

Řešení:

Každý uzávěr má kontakt, který vydává signál, když je otevřen. Podle podmínek sestavíme tabulku. Protože více uzávěrů nemůže být otevřeno současně, napíšeme do sloupce **Z** pro dané kombinace vstupních proměnných pomlčky. Nyní přepíšeme tabulku do Karnaughovy mapy a doplníme ji, v našem případě jedničkami. Z mapy pak sestavíme odpovídající algebraický výraz.

| A | B | C | Z |
|---|---|---|---|
| 0 | 0 | 0 | 0 |
| 0 | 0 | 1 | 1 |
| 0 | 1 | 0 | 1 |
| 0 | 1 | 1 | - |
| 1 | 0 | 0 | 1 |
| 1 | 0 | 1 | - |
| 1 | 1 | 0 | - |
| 1 | 1 | 1 | - |

| | | C | |
|----|----|---|---|
| | | 0 | 1 |
| AB | 00 | 0 | 1 |
| | 01 | 1 | - |
| 11 | 11 | - | - |
| | 10 | 1 | - |

| | | C | |
|----|----|---|---|
| | | 0 | 1 |
| AB | 00 | 0 | 1 |
| | 01 | 1 | 1 |
| 11 | 11 | 1 | 1 |
| | 10 | 1 | 1 |

$$Z = A + B + C$$