Booleova algebra

Logické proměnné a logické funkce

- Logická proměnná je veličina, která může nabývat pouze dvou hodnot, označených 0 a I (tedy dvojková proměnná) a nemůže se spojitě měnit
- Logická funkce *n* proměnných x_1, x_2, x_3,… x_n je funkce, která může nabývat, stejně jako všechny logické proměnné, pouze dvou hodnot

Logické proměnné a logické funkce

- Funkce rovnosti platí, když dvě logické proměnné A , B se sobě rovnají, tzn. , jestliže A = I , B = I nebo A = 0 , B = 0 , což zapisujeme A = B
- Dvě veličiny A = $a_1, a_2, a_3, \ldots a_n$; B = $b_1, b_2, b_3, \ldots b_n$ se sobě rovnají, když platí $a_i = b_i$ pro všechna *i*.

V Booleově algebře jsou definovány tři základní operace

- Logická negace $\overline{0} = I$

  $\overline{I} = 0$

- Logický součin $Y = A \,.\, B$

- Logický součet $Y = A + B$

Zákony a pravidla Booleovy algebry

- Komutativní zákon

$$A + B = B + A$$
$$A \,.\, B = B \,.\, A$$

- Asociativní zákon

$$A + (B + C) = (A + B) + C$$
$$A \,.\, (B \,.\, C) = (A \,.\, B) \,.\, C$$

- Distributivní zákon

$$A + B \,.\, C = (A + B) \,.\, (A + C)$$
$$A \,.\, (B + C) = A \,.\, B + A \,.\, C$$

Zákony a pravidla Booleovy algebry

- Zákon o agresívnosti prvku I a O

$$A + I = I$$
$$A \,.\, 0 = 0$$

- Zákon o neutrálnosti prvku I a O

$$A + 0 = A$$
$$A \,.\, I = A$$

- Zákon o vyloučení třetího

$$A + \overline{A} = I$$
$$A \,.\, \overline{A} = 0$$

- Zákon dvojité negace

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

Zákony a pravidla Booleovy algebry

- Zákon absorpce

$$A + A = A$$
$$A \, . \, A = A$$
$$A + A \, . \, B = A$$
$$A \, . \, (A + B) = A$$

Zákon absorpce negace

$$A + \overline{A} = I$$
$$A \, . \, \overline{A} = 0$$
$$A + \overline{A} \, . \, B = A + B$$
$$A \, . \, (\overline{A} + B) = A \, . \, B$$
$$\overline{A} + A.B = \overline{A} + B$$
$$\overline{A}(A + B) = \overline{A}.B$$

De Morganův zákon

$$\overline{A + B + C + \ldots} = \overline{A} \,.\, \overline{B} \,.\, \overline{C} \,.\, \ldots$$

$$\overline{A \,.\, B \,.\, C \,.\, \ldots} = \overline{A} + \overline{B} + \overline{C} + \ldots$$

- negaci funkce získáme nahrazením každé proměnné její negací a záměnou značek součtu a součinu navzájem

$$A + B \,.\, C = A + (B \,.\, C)$$

Tedy :

$$\overline{A + B \,.\, C} = \overline{A} \,.\, (\overline{B} + \overline{C})$$

a neplatí :

$$\overline{A \,.\, B} + \overline{C}$$

Definice logické funkce

- Pravdivostní tabulkou
  - Pravdivostní tabulka je tabulka , do které se zapisuje logická (Booleovská funkce) . Pravdivostní tabulka má r+n sloupců a 2^n řádků . Číslo r je počet sloupců výsledných funkcí (obyčejně bývá jedna výsledná funkce – tedy jeden sloupec). Číslo n udává počet proměnných. Číslo 2^n udává počet všech možných kombinací proměnných, kde číslo n je počet proměnných. Tyto kombinace reprezentuje počet řádků .
- Zápisem logické funkce
  - úplné disjunktivní normální formy – ÚDNF
  - úplné konjunktivní normální formy – ÚKNF

Způsoby zápisu a zobrazení kombinačních logických funkcí

Abychom mohli s kombinačními logickými funkcemi pracovat, musíme je nejprve zapsat či zobrazit.

Nejčastěji se používají tyto způsoby zápisu, popř. zobrazení kombinačních logických funkcí:

- zápis pomocí pravdivostní tabulky,
- zápis logickým výrazem,
- zobrazení pomocí mapy,
- zobrazení pomocí logického schématu.

Úplně zadaná funkce

- Logická funkce je úplně zadaná , jestliže je známa její hodnota I nebo 0 pro všechny možné kombinace hodnot proměnných

A	B	C	f
0	0	0	0
0	0	I	I
0	I	0	I
0	I	I	0
I	0	0	0
I	0	I	I
I	I	0	0
I	I	I	I

Neúplně zadaná funkce

- Logická funkce je neúplně zadaná , když její hodnota pro některé kombinace hodnot proměnných je libovolná nebo není určena

A	B	C	f
0	0	0	0
0	0	I	I
0	I	0	X
0	I	I	0
I	0	0	X
I	0	I	I
I	I	0	0
I	I	I	X

Logická funkce n-proměnných

- Logická funkce f_n n-proměnných nabývá všech možných hodnot, pro všechny možné kombinace n-proměnných .
- Počet funkcí je $(2^2)^n$

Funkce jedné proměnné

A	f_0	f_1	f_2	f_3
0	0	0	I	I
I	0	I	0	I

- konstanty $f_0 = 0$ a $f_3 = I$
- proměnná sama $f_1 = A$
- negace proměnné $f_2 = \overline{A}$

Funkce dvou proměnných

A	B	f_0	f_1	f_2	f_3	f_4	f_5	f_6	f_7	f_8	f_9	f_{10}	f_{11}	f_{12}	f_{13}	f_{14}	f_{15}
0	0	0	0	0	0	0	0	0	0	I	I	I	I	I	I	I	I
0	I	0	0	0	0	I	I	I	I	0	0	0	0	I	I	I	I
I	0	0	0	I	I	0	0	I	I	0	0	I	I	0	0	I	I
I	I	0	I	0	I	0	I	0	I	0	I	0	I	0	I	0	I

Funkce dvou proměnných

- součet modulo 2 $A \oplus B$ $f_6 = A\overline{B} + \overline{A}B$
- ekvivalence $A \otimes B$ $f_9 = AB + \overline{A}\,\overline{B}$
- Piercova funkce $f_8 = \overline{A}\,\overline{B} = \overline{A + B}$
- Schefferova funkce $f_{14} = \overline{A} + \overline{B} = \overline{AB}$
- inhibice $f_2 = A\overline{B}$
- Negace obrácené implikace $f_4 = \overline{A}B$
- Obrácená implikace $f_{11} = A + \overline{B}$
- Implikace $f_{13} = \overline{A} + B$

Úplná disjunktivní normální forma

- ÚDNF – je to součet základních součinů přímých nebo negovaných proměnných . Každý základní součin (minterm – z ang. minimal polynomial term) nabývá hodnoty I pro určitou kombinaci, kdy funkce má hodnotu I a hodnoty 0 pro všechny ostatní kombinace. ÚDNF vyjadřuje funkci jako součet případů, kdy má hodnotu I.

Úplná konjunktivní normální forma

- ÚKNF – je to součin základních součtů přímých nebo negovaných proměnných . Každý základní součet nabývá hodnoty 0 pro určitou kombinaci, kdy funkce má hodnotu 0 a hodnoty I pro všechny ostatní kombinace. ÚKNF vyjadřuje funkci jako součin případů, kdy má hodnotu 0.

Výpis logických funkcí z pravdivostní tabulky

A	B	C	f
0	0	0	I
0	0	I	0
0	I	0	I
0	I	I	0
I	0	0	I
I	0	I	0
I	I	0	I
I	I	I	0

$$f = \overline{A}\,\overline{B}\,\overline{C} + \overline{A}\,B\,\overline{C} + A\,\overline{B}\,\overline{C} + A\,B\,\overline{C}$$

$$f = (A + B + \overline{C}) . (A + \overline{B} + \overline{C}) . (\overline{A} + B + \overline{C}) . (\overline{A} + \overline{B} + \overline{C})$$

Karnaughova mapa

A

I	0	0	I	B
I	0	0	I	

C

A	B	C	f
0	0	0	I
0	0	I	0
0	I	0	I
0	I	I	0
I	0	0	I
I	0	I	0
I	I	0	I
I	I	I	0

Zjednodušování zápisu logické funkce

- Algebraická minimalizace
  - upravování logického výrazu podle zákonů a pravidel Booleovy algebry
- Grafická – Karnaughova metoda
  - Pro zjednodušení funkce pomocí algebraické minimalizace spojujeme součiny (mintermy) , které se liší v jediné proměnné

Algebraická minimalizace

A	B	f
0	0	I
0	I	0
I	0	I
I	I	I

$$f = \overline{A}\,\overline{B} + A\overline{B} + AB$$

$$f = \overline{A}\,\overline{B} + A\overline{B} + AB = A(B + \overline{B}) + \overline{A}\;\overline{B} = A + \overline{A}\;\overline{B} = A + \overline{B}$$

Minimalizace logická funkce

- V Karnaughově mapě vytváříme **podmapy**
- Podmapou rozumíme sjednocení 2^k sousedních stavů, ve kterých nabývá logická funkce hodnoty 1 pro k = 0, 1, 2, ... , n-1
- Každou podmapou vyloučíme k proměnných z dvou, čtyř až 2^{n-1} základních součinů
- Snažíme se vytvářet co největší podmapy, abychom vyloučili co největší počet proměnných
- Využíváme k tomu také **neurčité stavy**

Pravidla vytváření podmap

- vybranými podmapami musí být pokryty všechny jednotkové stavy logické funkce,
- do podmapy spojujeme stejné stavy, které spolu sousedí hranou, a to i přes okraje mapy. Rohy mapy jsou též sousedními stavy. Členy dvou sousedních polí se od sebe liší jednou proměnnou a tuto proměnnou můžeme vyloučit,
- podmapu pravidelného tvaru (čtverec, obdélník) vytváříme co největší, aby se ze skupiny stavů vyloučilo co nejvíce proměnných,
- podmapy se mohou prolínat,
- nevytváříme zbytečné podmapy, tzn. že nespojujeme ty stavy, které už byly předtím pokryty jinou podmapou,
- čím větší bude podmapa, tím jednodušší bude výsledný výraz

Zjednodušení úplně zadané funkce

A	B	C	D	f
0	0	0	0	I
0	0	0	I	0
0	0	I	0	I
0	0	I	I	0
0	I	0	0	0
0	I	0	I	I
0	I	I	0	0
0	I	I	I	0
I	0	0	0	I
I	0	0	I	I
I	0	I	0	I
I	0	I	I	I
I	I	0	0	0
I	I	0	I	I
I	I	I	0	0
I	I	I	I	0

A

	I	0	0	I
D	0	I	I	I
D	0	0	0	I
	I	0	0	I

B C

$$\overline{A}\,B\,\overline{C}\,D + A\,B\,\overline{C}\,D = B\,\overline{C}\,D\,(\overline{A} + A) = B\,\overline{C}\,D$$

$$A\,\overline{B}\,\overline{C}\,(D + \overline{D}) + A\,\overline{B}\,C\,(D + \overline{D}) = A\,\overline{B}\,\overline{C} + A\,\overline{B}\,C = A\,\overline{B}\,(\overline{C} + C) = A\,\overline{B}$$

$$\overline{A}\,\overline{B}\,\overline{D}\,(\overline{C} + C) + A\,\overline{B}\,\overline{D}\,(\overline{C} + C) = \overline{A}\,\overline{B}\,\overline{D} + A\,\overline{B}\,\overline{D} = \overline{B}\,\overline{D}\,(\overline{A} + A) = \overline{B}\,\overline{D}$$

$$f = B\,\overline{C}\,D + A\,\overline{B} + \overline{B}\,\overline{D}$$

$$f = \overline{A}\,\overline{B}\,\overline{C}\,\overline{D} + \overline{A}\,\overline{B}\,C\,\overline{D} + \overline{A}\,B\,\overline{C}\,D + A\,\overline{B}\,\overline{C}\,\overline{D} + A\overline{B}\,\overline{C}D + A\,\overline{B}\,C\,\overline{D} + A\,\overline{B}\,C\,D + A\,B\,\overline{C}\,D$$

Příklad 1. řešení

1) Minimalizace úpravou logické funkce:

( ) = m( ) = minterm( )

D	x_2	x_1	x_0	f
0	0	0	0	1
1	0	0	1	1
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

(0) (1) (3) (4) (7)

$$f_1 = \overline{x_2}\,\overline{x_1}\,\overline{x_0} + \overline{x_2}\,\overline{x_1}\,x_0 + \overline{x_2}\,x_1\,x_0 + x_2\,\overline{x_1}\,\overline{x_0} + x_2\,x_1\,x_0 =$$

(0)(4) (1)(3) (3)(7)

$$= \overline{x_2}\,\overline{x_1}\,\overline{x_0} + \overline{x_2}\,\overline{x_1}\,x_0 + \overline{x_2}\,x_1\,x_0 + \overline{x_2}\,x_1\,x_0 + x_2\,\overline{x_1}\,\overline{x_0} + x_2\,x_1\,x_0 =$$

a + a = a

(0)(4) (1)(3) (3)(7)

$$= \overline{x_1}\,\overline{x_0}\,(\overline{x_2} + x_2) + \overline{x_2}\,x_0\,(\overline{x_1} + x_1) + x_1\,x_0\,(\overline{x_2} + x_2) =$$

$$= \overline{x_1}\,\overline{x_0} + \overline{x_2}\,x_0 + x_1\,x_0$$

1. řešení

$$f_1 = \overline{x_1}\,\overline{x_0} + \overline{x_2}\,x_0 + x_1\,x_0$$

Příklad 2. řešení

1) Minimalizace úpravou logické funkce (pokrač.):

( ) = m( ) = minterm( )

D	x_2	x_1	x_0	f
0	0	0	0	1
1	0	0	1	1
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

(0) (1) (3) (4) (7)

$$f_2 = \underbrace{\bar{x}_2 \bar{x}_1 \bar{x}_0}_{(0)} + \underbrace{\bar{x}_2 \bar{x}_1 x_0}_{(1)} + \underbrace{\bar{x}_2 x_1 x_0}_{(3)} + \underbrace{x_2 \bar{x}_1 \bar{x}_0}_{(4)} + \underbrace{x_2 x_1 x_0}_{(7)} =$$

(0)(1) (3)(7) (0)(4)

$$= \bar{x}_2 \bar{x}_1 \bar{x}_0 + \bar{x}_2 \bar{x}_1 \bar{x}_0 + \bar{x}_2 \bar{x}_1 x_0 + \bar{x}_2 x_1 x_0 + x_2 \bar{x}_1 \bar{x}_0 + x_2 x_1 x_0 =$$

a + a = a

(0)(4) (0)(1) (3)(7)

$$= \bar{x}_1 \bar{x}_0 (\bar{x}_2 + x_2) + \bar{x}_2 \bar{x}_1 (\bar{x}_0 + x_0) + x_1 x_0 (\bar{x}_2 + x_2) =$$

$$= \bar{x}_1 \bar{x}_0 + \bar{x}_2 \bar{x}_1 + x_1 x_0$$

2. řešení

$$f_2 = \bar{x}_1 \bar{x}_0 + \bar{x}_2 \bar{x}_1 + x_1 x_0$$

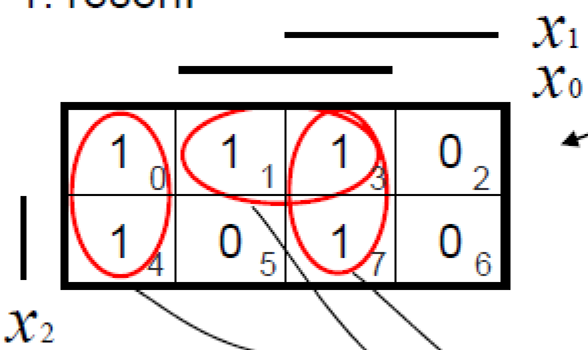
Příklad

2) Minimalizace z K-mapy (Karnaughova mapa):

$$f = \sum m(0,1,3,4,7)$$

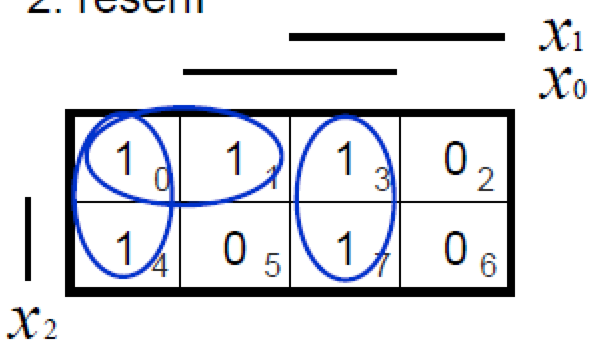
D	x_2	x_1	x_0	f
0	0	0	0	1
1	0	0	1	1
2	0	1	0	0
3	0	1	1	1
4	1	0	0	1
5	1	0	1	0
6	1	1	0	0
7	1	1	1	1

1. řešení



$$f_1 = \overline{x_1}\,\overline{x_0} + \overline{x_2}\,x_0 + x_1 x_0$$

2. řešení



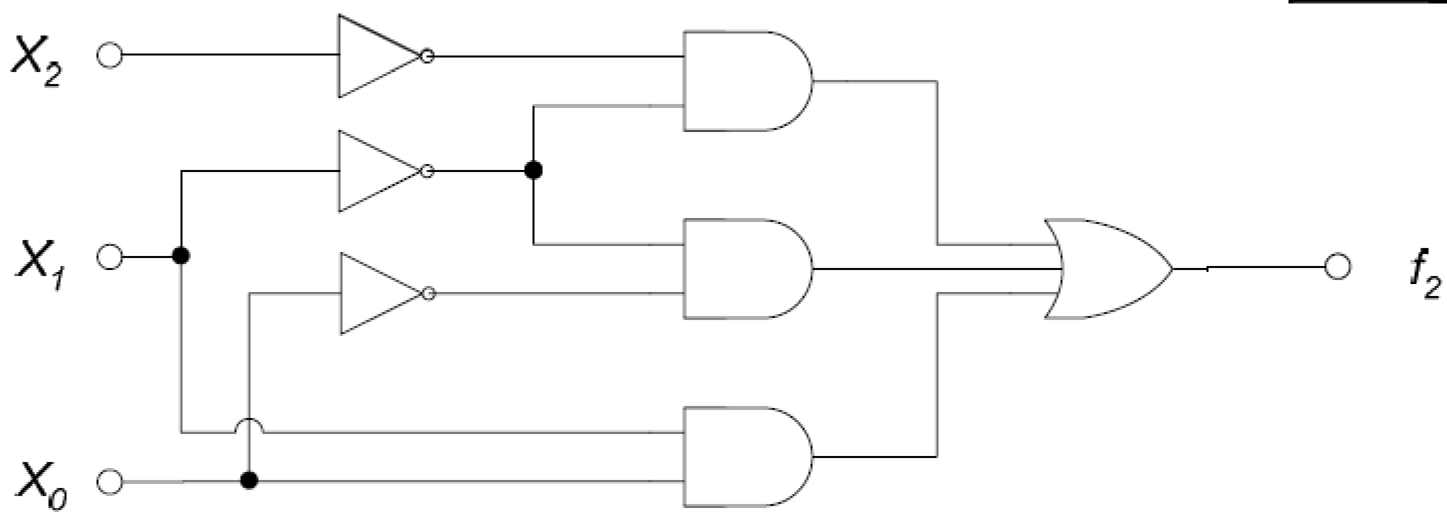
$$f_2 = \overline{x_1}\,\overline{x_0} + \overline{x_2}\,\overline{x_1} + x_1 x_0$$

Porovnej s řešením dle 1)

Realizace příkladu

$$f_2 = \bar{x}_1 \bar{x}_0 + \bar{x}_2 \bar{x}_1 + x_1 x_0$$

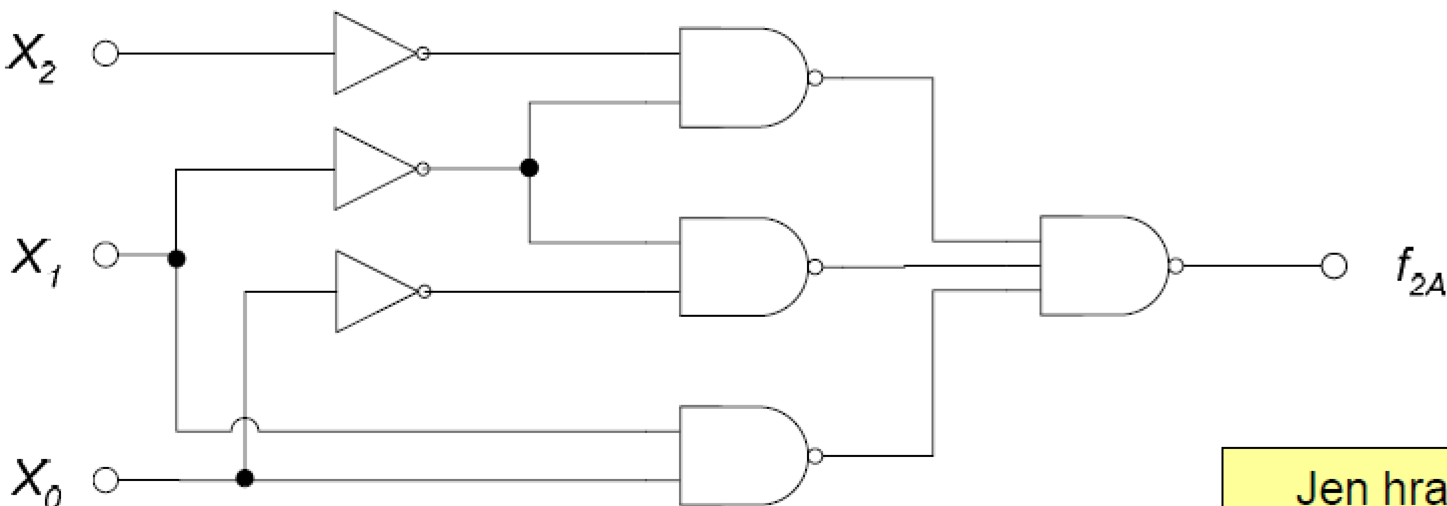
D	x_2	x_1	x_0	f
0	0	0	0	1
1	0	0	1	1
2	0	1	0	0
3	0	1	1	1
4	1	0	0	1
5	1	0	1	0
6	1	1	0	0
7	1	1	1	1



Realizace příkladu - NAND

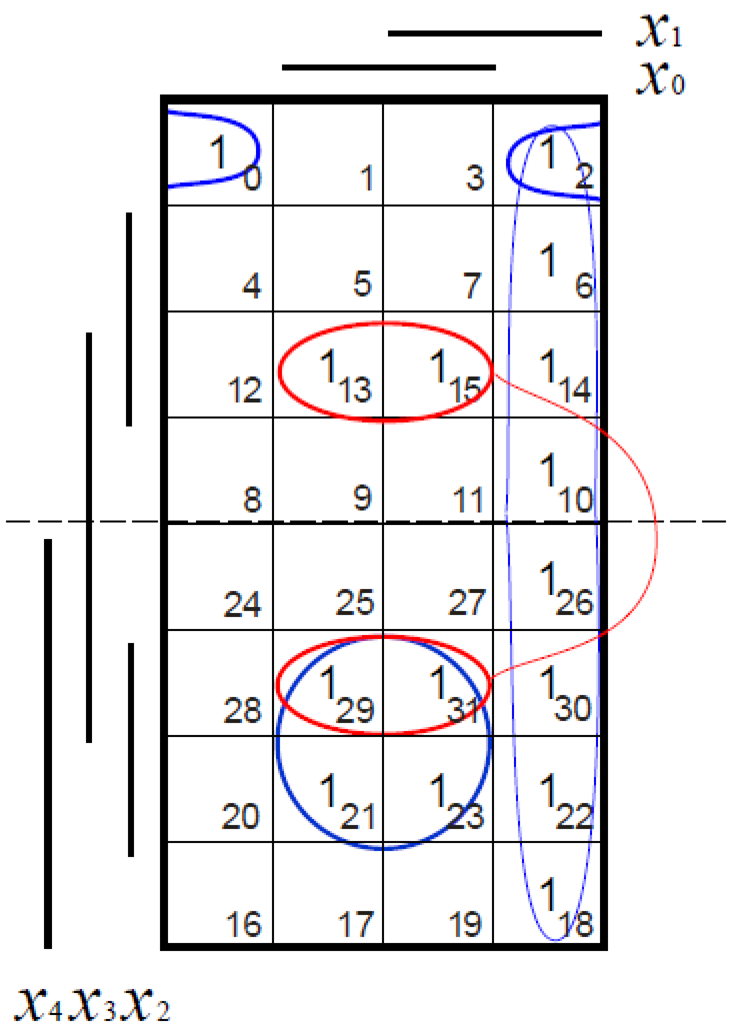
$$f_{2A} = \overline{\overline{f}}_2 = \overline{\overline{\bar{x}_1 \bar{x}_0 + \bar{x}_2 \bar{x}_1 + x_1 x_0}} = \overline{\overline{\bar{x}_2 \bar{x}_1}\;\overline{\bar{x}_1 \bar{x}_0}\;\overline{x_1 x_0}}$$

D	x_2	x_1	x_0	f
0	0	0	0	1
1	0	0	1	1
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



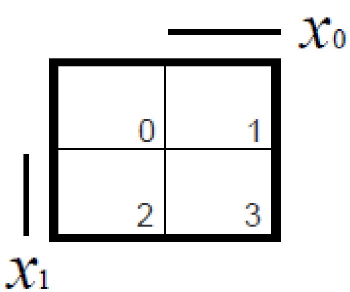
Jen hradla NAND

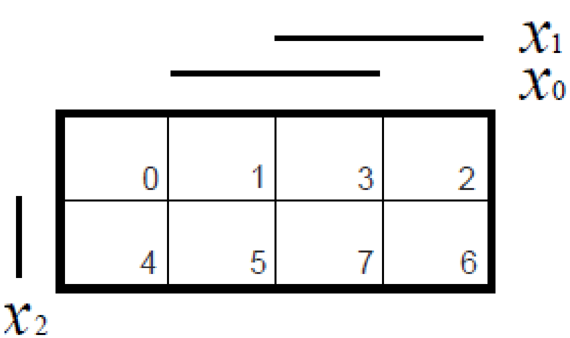
Karnaughova mapa pro 5 proměnných

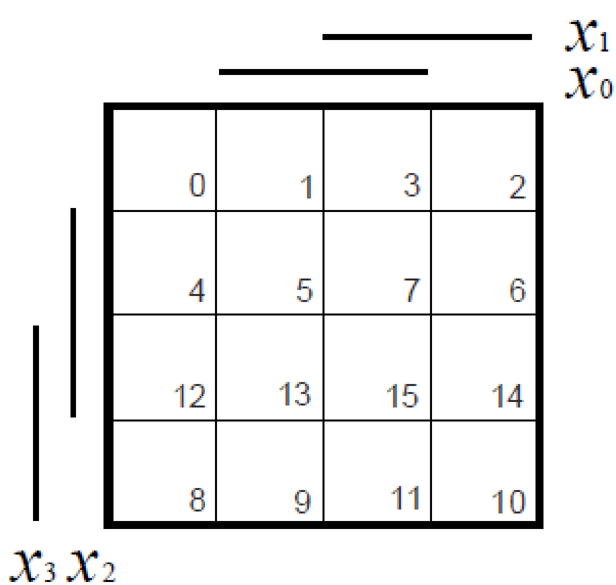


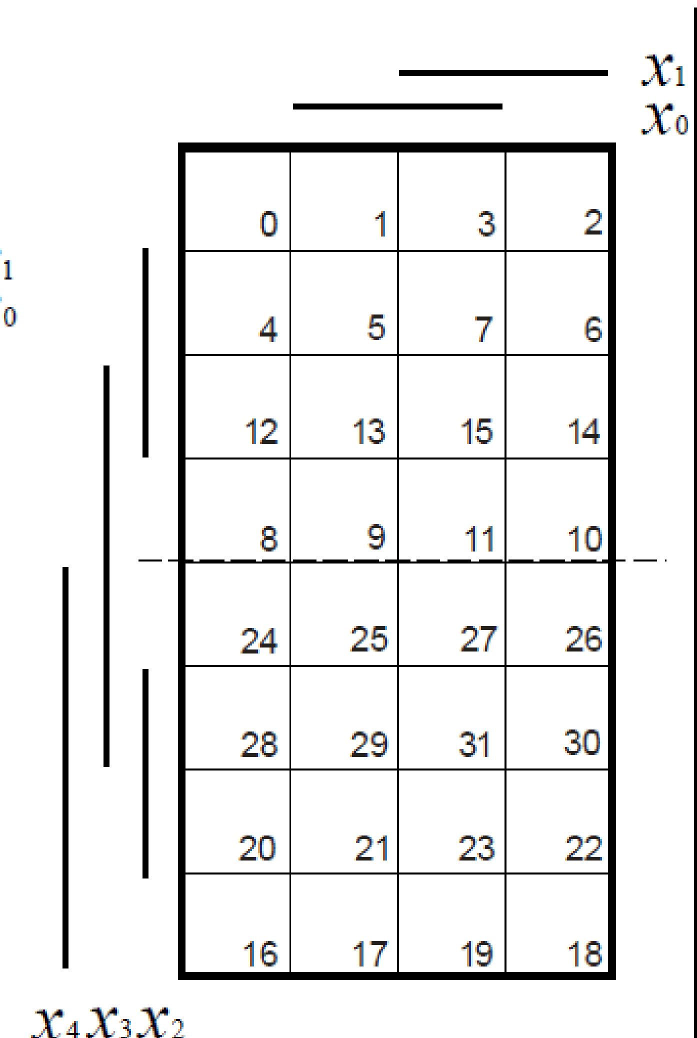
$$f = \bar{x}_4 \bar{x}_3 \bar{x}_2 \bar{x}_0 + x_3 x_2 x_0 +$$
$$+ x_1 \bar{x}_0 + x_4 x_2 x_0$$

Šablony Kaurnaghovy mapy









Zjednodušení neúplně zadané funkce

A	B	C	D	f
0	0	0	0	I
0	0	0	I	X
0	0	I	0	I
0	0	I	I	0
0	I	0	0	0
0	I	0	I	I
0	I	I	0	X
0	I	I	I	0
I	0	0	0	I
I	0	0	I	I
I	0	I	0	I
I	0	I	I	I
I	I	0	0	0
I	I	0	I	I
I	I	I	0	0
I	I	I	I	X

I	0	0	I
X	I	I	I
0	0	X	I
I	X	0	I

A

D

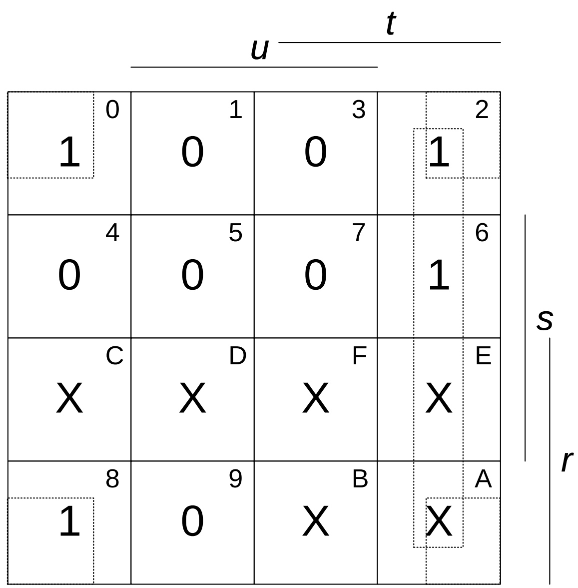
C

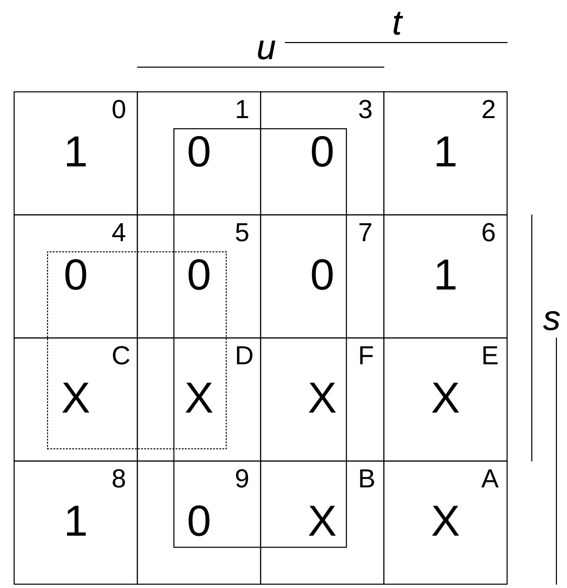
B

$$f = \overline{B}\,\overline{D} + \overline{C}D + AD$$

Minimalizace neúplně určených funkcí

Pravdivostní tabulka neúplně určené funkce neobsahuje všechny řádky, které má tabulka úplně určené funkce se stejným počtem proměnných. Tedy pro některé kombinace vstupních proměnných není hodnota funkce definována. Pro tyto kombinace můžeme hodnotu funkce definovat dodatečně tak, aby vyjádření funkce bylo co nejjednodušší.

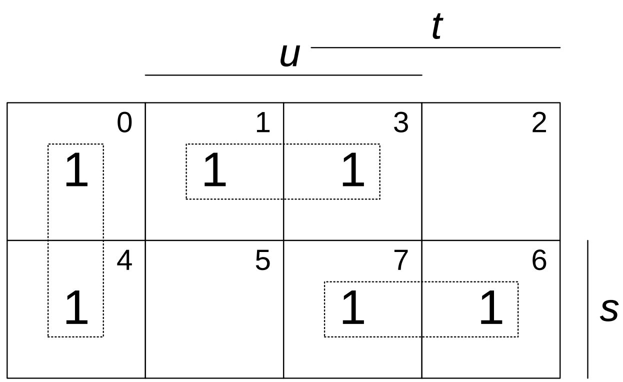


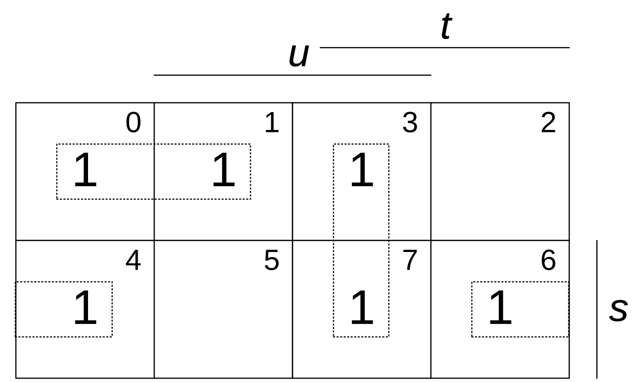


$$e = \overline{s} \cdot \overline{u} + t \cdot \overline{u} \qquad\qquad e = \overline{u} \cdot \left(\overline{s} + t\right)$$

Minimalizace funkce e s využitím neúplnosti její definice

Dvojí výběr:





$$y = \bar{t} \cdot \bar{u} + \bar{s} \cdot u + s \cdot t \qquad y = \bar{s} \cdot \bar{t} + t \cdot u + s \cdot \bar{u}$$

Funkce se dvěma možnými minimálními součtovými tvary

Konzultace k tématu semestrální práce Návrh logického obvodu

- Popis funkce logického obvodu s min. 4 vstupy a min. 1 výstupem
  - Slovní popis a pomocí pravdivostní tabulky
- Minimalizace logické funkce
  - Pomocí zákonů Booleovy algebry a pomocí Karnaughovy mapy
- Návrh realizace minimalizované logické funkce
  - Pomocí logických členů NAND

Příklad

- Chceme určit logickou funkci zařízení které:
  - rozsvítí zelenou žárovku – F1, když v nějakém výrobním procesu překročí kritickou hodnotu pouze jedna ze sledovaných veličin, např. tlak (x), nebo teplota (y), nebo vlhkost (z), nebo žádná,
  - rozsvítí červenou žárovku - F2, když je překročena kritická hodnota kterýchkoli dvou veličin současně,
  - zapne sirénu - F3, když jsou překročeny kritické hodnoty všech tří veličin současně.

Příklad

x	y	z	F1	F2	F3
0	0	0	1	0	0
0	0	1	1	0	0
0	1	0	1	0	0
0	1	1	0	1	0
1	0	0	1	0	0
1	0	1	0	1	0
1	1	0	0	1	0
1	1	1	0	0	1

$$F1 = \bar{x}\bar{y}\bar{z} + \bar{x}\bar{y}z + \bar{x}y\bar{z} + x\bar{y}\bar{z}$$

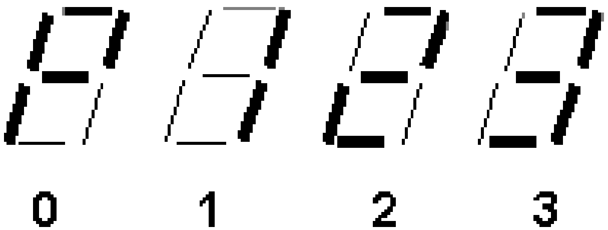
$$F2 = \bar{x}yz + x\bar{y}z + xy\bar{z}$$

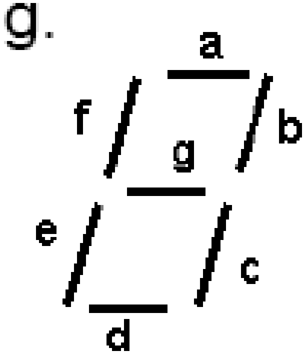
$$F3 = xyz$$

Dekodér z kódu 2bit. BCD na 7-mi segmentový displej

Vstupní proměnné: Q2,Q3

Výstupní proměnné a,b,c,d,e,f,g.





		Q2
	P	1
Q3	2	3

Q3	Q2	N	a	b	c	d	e	f	g
0	0	P	1	1	0	0	1	1	1
0	1	1	0	1	1	0	0	0	0
1	0	2	1	1	0	1	1	0	1
1	1	3	1	1	1	1	0	0	1

$a = \overline{Q2} + Q3$

$b = 1$

$c = Q2$

$d = Q3$

$e = \overline{Q2}$

$f = \overline{Q2}.\overline{Q3}$

$g = \overline{Q2} + Q3$

a

		Q2
	1	0
Q3	1	1

c

		Q2
	0	1
Q3	0	1

d

		Q2
	0	0
Q3	1	1

e

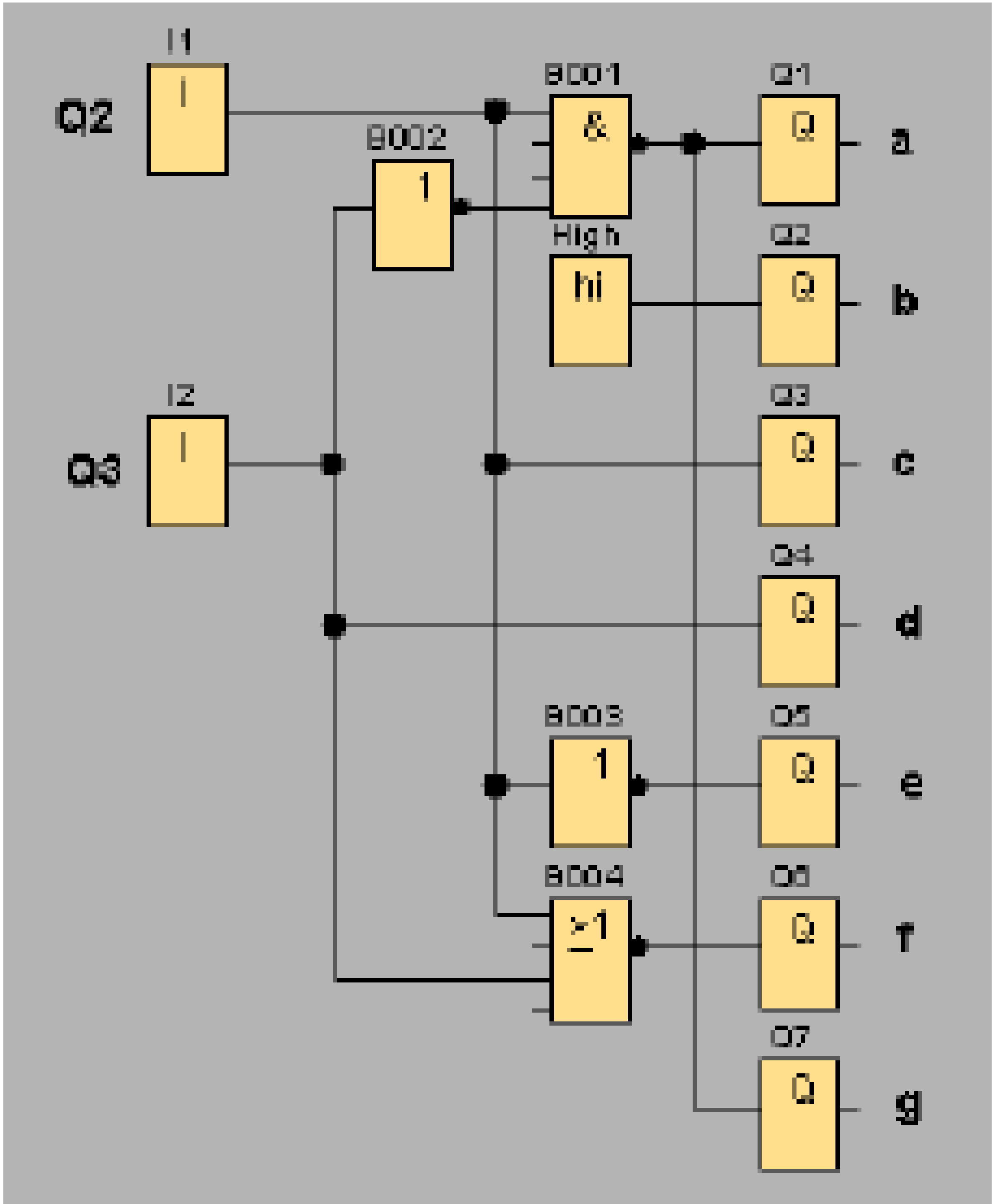
		Q2
	1	0
Q3	1	0

f

		Q2
	1	0
Q3	0	0

g

		Q2
	1	0
Q3	1	1

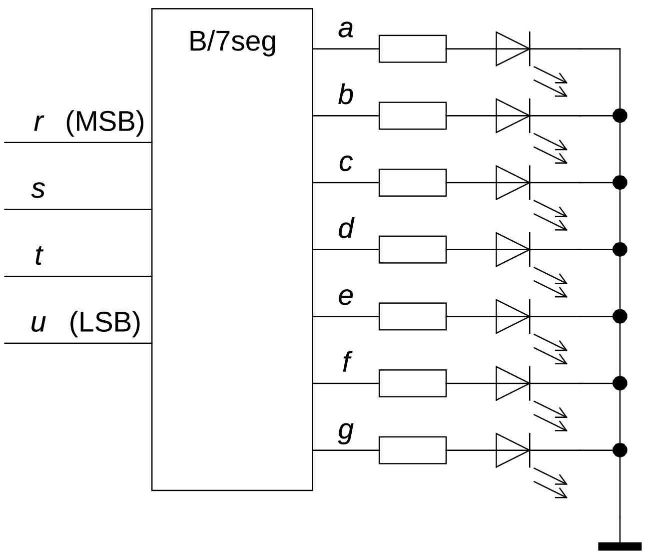


I1
Q2
B002
B001
Q1
a
High
hi
Q2
b
I2
Q3
Q3
c
Q4
d
B003
Q5
e
B004
Q6
f
Q7
g

Další příklad

Jako příklad tohoto způsobu zápisu uvedeme popis převodníku čtyřbitového binárního kódu na kód sedmisegmentového displeje s hexadecimálním zobrazením. Náčrt zapojení a zobrazované znaky (hexadecimální číslice) jsou na obrázku.

Vstupní proměnné **MSB** (Most Significant Bit) a **LSB** (Least Significant Bit) označují nejvýznamnější a nejméně významný bit.



Převodník čtyřbitového binárního kódu na kód sedmisegmentového displeje

Převodník čtyřbitového binárního kódu na kód sedmisegmentového displeje, při hodnotě **1** proměnných ***a*** až ***g*** odpovídající segmenty svítí.

Pravdivostní tabulka převodníku

číslo (stavový index)	číslo (hex)	vstupy r	s	t	u	výstupy a	b	c	d	e	f	g
0	0	0	0	0	0	1	1	1	1	1	1	0
1	1	0	0	0	1	0	1	1	0	0	0	0
2	2	0	0	1	0	1	1	0	1	1	0	1
3	3	0	0	1	1	1	1	1	1	0	0	1
4	4	0	1	0	0	0	1	1	0	0	1	1
5	5	0	1	0	1	1	0	1	1	0	1	1
6	6	0	1	1	0	1	0	1	1	1	1	1
7	7	0	1	1	1	1	1	1	0	0	0	0
8	8	1	0	0	0	1	1	1	1	1	1	1
9	9	1	0	0	1	1	1	1	1	0	1	1
10	A	1	0	1	0	1	1	1	0	1	1	1
11	B	1	0	1	1	0	0	1	1	1	1	1
12	C	1	1	0	0	1	0	0	1	1	1	0
13	D	1	1	0	1	0	1	1	1	1	0	1
14	E	1	1	1	0	1	0	0	1	1	1	1
15	F	1	1	1	1	1	0	0	0	1	1	1

Příklad (displej)**: segment e**

$$t\overline{u} + rt + rs + \overline{u}\,\overline{s}$$