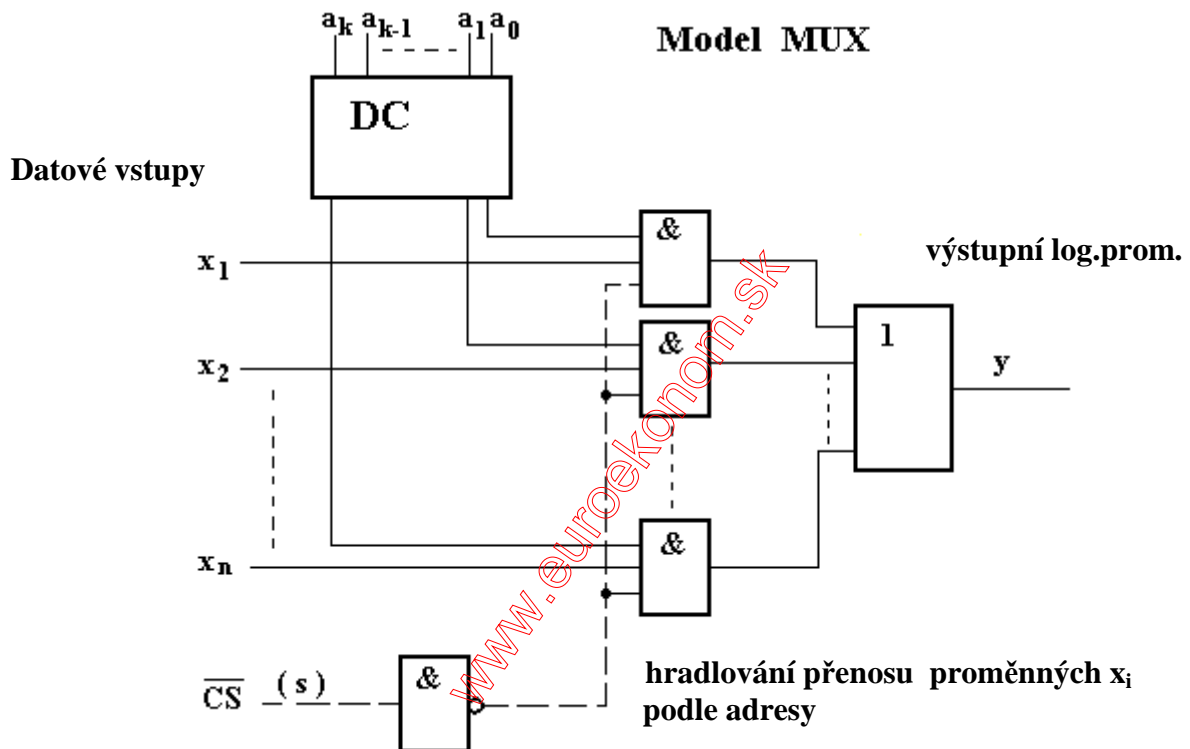


Návrh kombinačních logických obvodů s IO střední hustoty int.

I. Použití multiplexorů jako univerzálního logického bloku

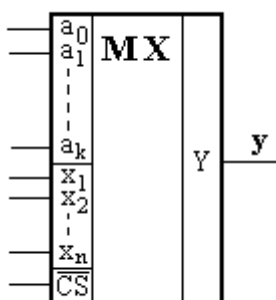
Strukturní model multiplexoru a jeho funkce :

řídící signály a_i – adresa, která datová hodnota x_i má být přenášena na výstup

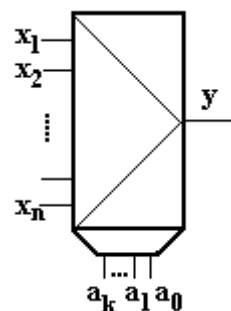


Schematické značky :

a) Podle oborové normy



b) Mnemotechnická používaná sch. značka

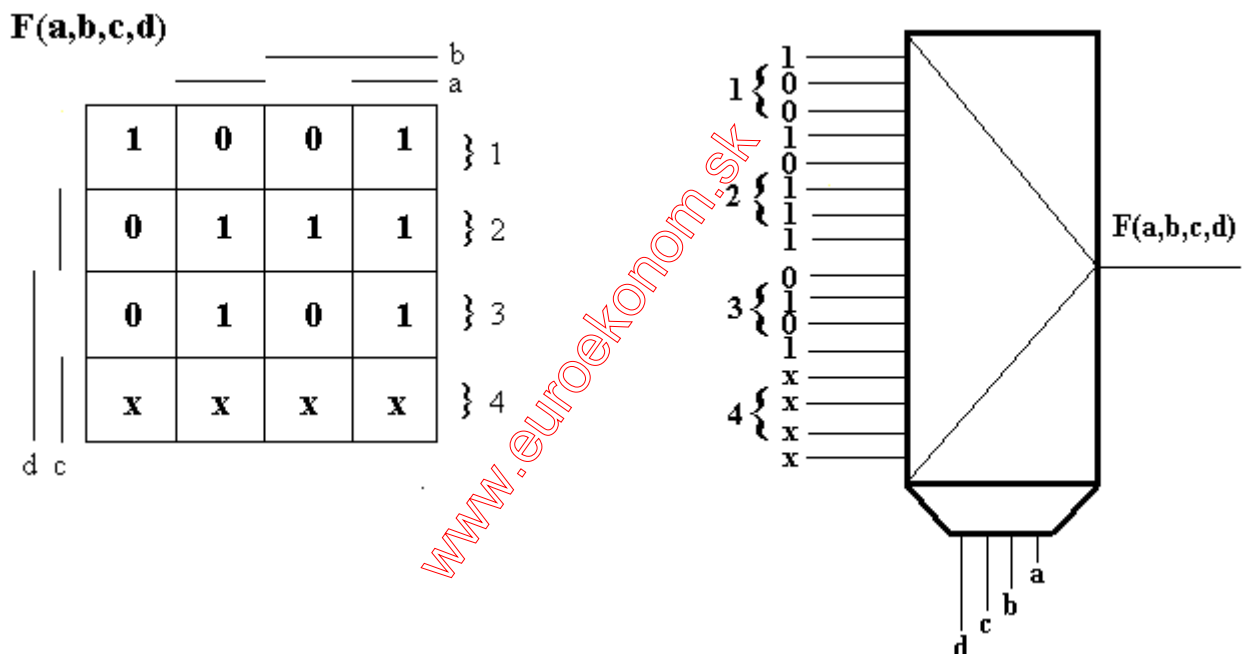


1. Použití MUX pro realizaci libovolné kombinační logické funkce – jako univerzální funkční blok:

- Vstupní logické proměnné zadané logické funkce zapojíme na adresní vstupy multiplexoru
- Konkrétní stav na těchto vstupních logických proměnných bude vlastně vytvářet adresu, která bude určovat, která hodnota z datového vstupu se bude přenášet na výstup (log. konstanta 0 nebo 1 – jako funkční hodnota logické funkce)

Příklad: Logická funkce $F(a, b, c, d)$ je zadána Svobodovou mapou.

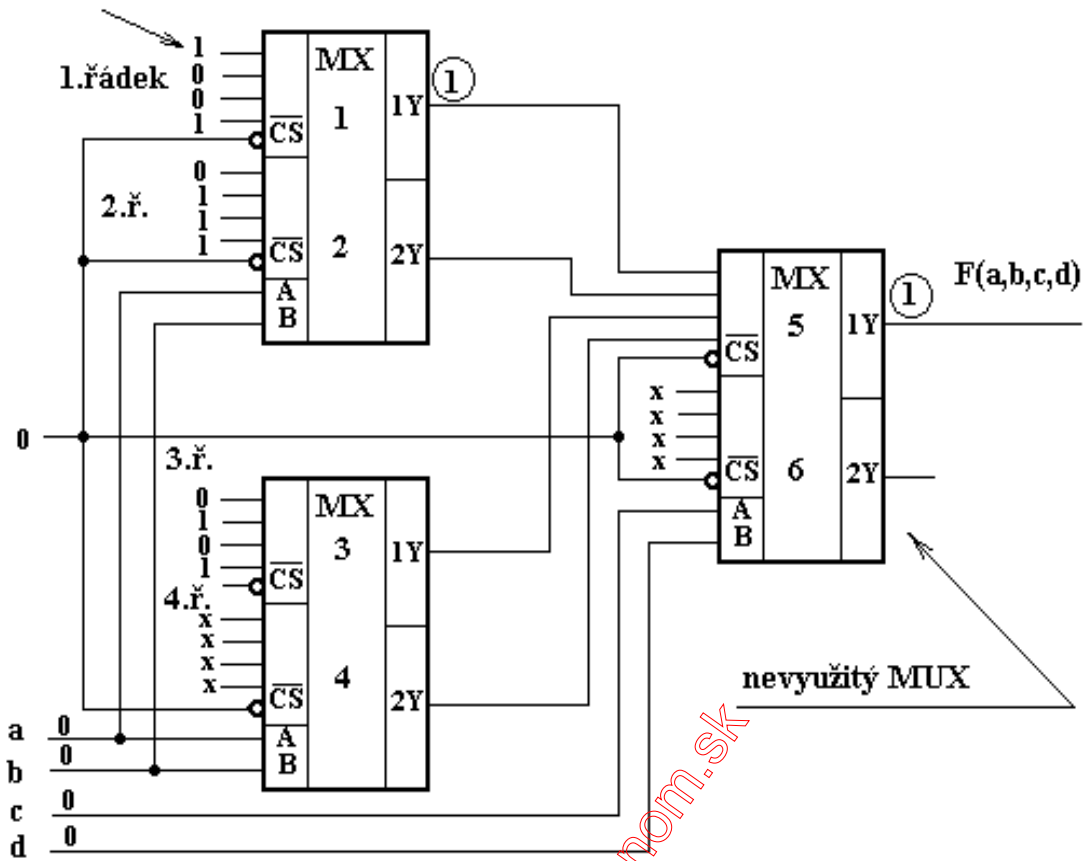
Řešení s 16 – vstupovým multiplexorem SN 74151 (4 adresní vstupy)



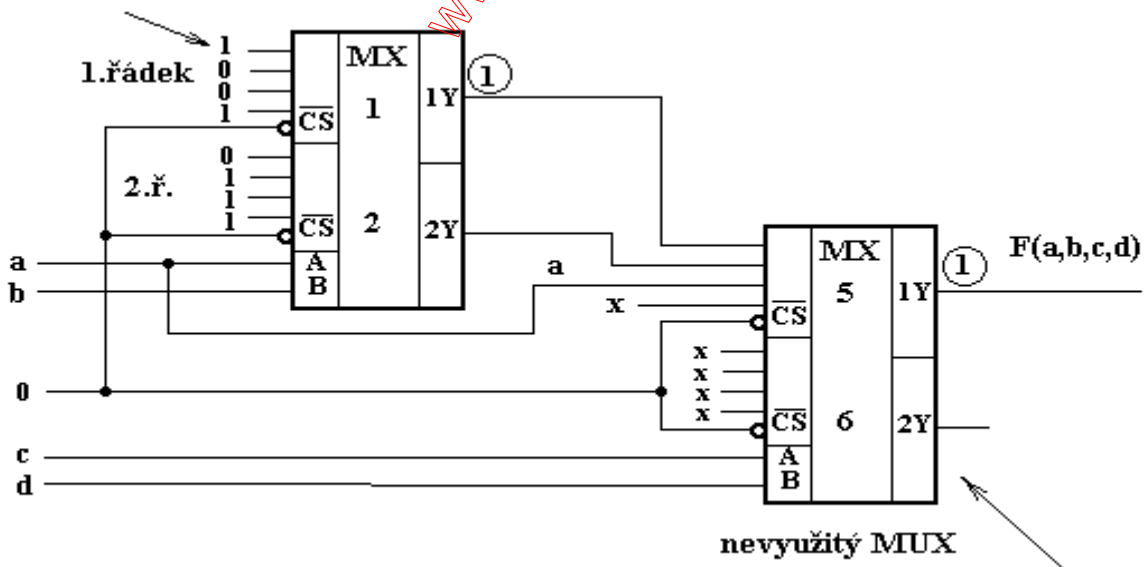
Minimalizace logické funkce s tímto multiplexorem je bezpředmětná - realizuje se tudíž ÚNF !!

2. Řešení s MUX s menším počtem adresních vstupů (74153)

- kaskádou multiplexorů - řádkům nebo sloupcům se přiřadí pomocné funkce F_1, F_2, F_3, F_4



Úprava s menším počtem multiplexorů

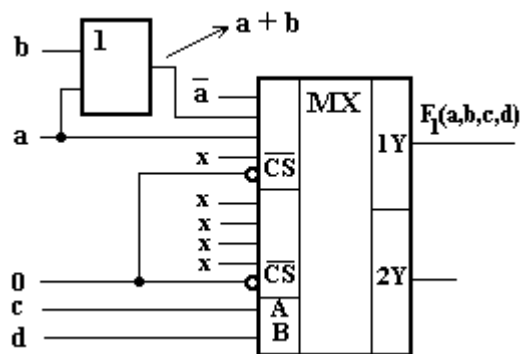


b) Řešení s jedním MUX resp. s jedním IO 74153

Zadaná logická funkce:

$$F_1(a, b, c, d)$$

		a		
		b		
	1	0	1	0
0	1	1	1	1
0	1	0	1	1
x	x	x	x	x
d	c			



II) Použití dekodéru (DC)

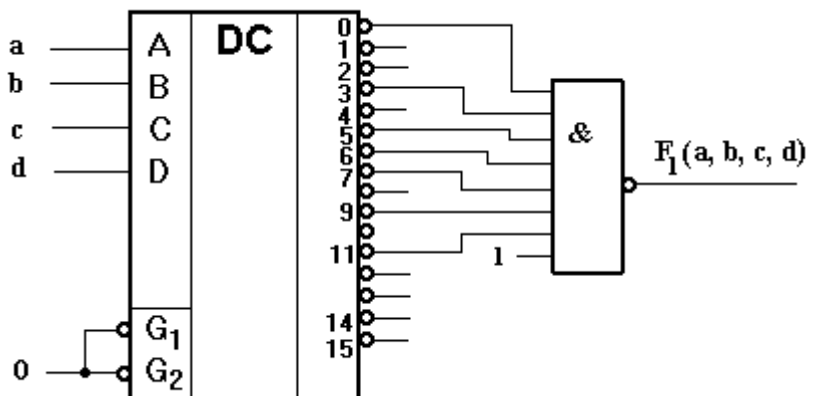
- užite se dekodéru z binárního kódu do kódu 1 z N (74154) aktivovaná výstupní proměnná vždy jednoznačně určuje existující vstupní stav
- kombinační obvod realizujeme tak, že vytvoříme logický součet hodnot výstupních proměnných odpovídajících těm vstupním stavům, které produkují jedničkovou hodnotu zadané logické funkce

Příklad: Realizujte zadanou logickou funkci $F_1(a, b, c, d)$ pomocí dekodéru 74154

Zadaná logická funkce:

$$F_1(a, b, c, d)$$

		a		
		b		
	1	0	1	0
0	1	1	1	1
0	1	0	1	1
x	x	x	x	x
d	c			



Minimalizace není možná!!

Realizace vícevýstupové logické funkce pomocí dekodéru 74154

Zadané logické funkce:

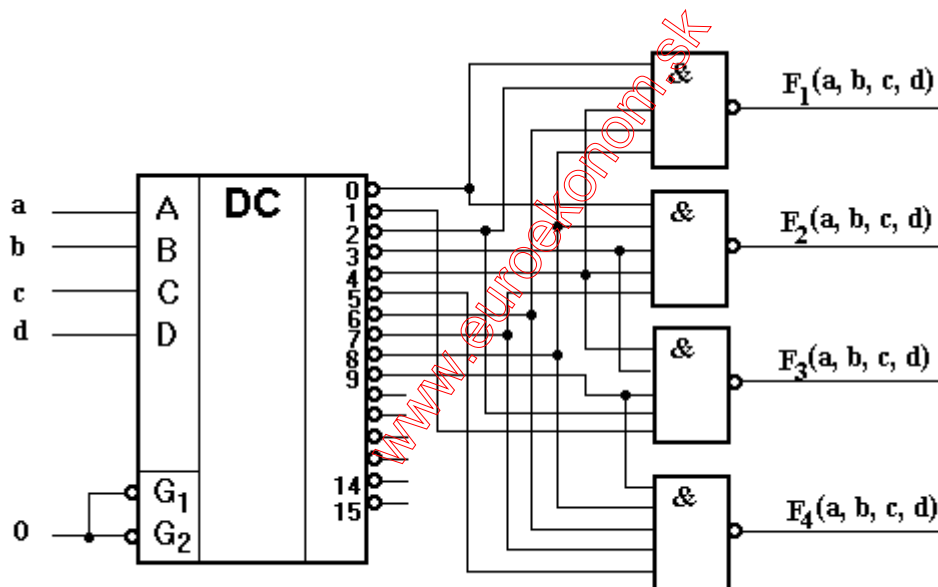
$$F_1(x_1, x_2, x_3, x_4) = \sum(0, 2, 4, 6, 8) + \sum_x(10, 14, 15)$$

$$F_2(x_1, x_2, x_3, x_4) = \sum(0, 3, 4, 7, 8) + \sum_x(11, 15)$$

$$F_3(x_1, x_2, x_3, x_4) = \sum(1, 2, 3, 4, 9) + \sum_x(10, 14)$$

$$F_4(x_1, x_2, x_3, x_4) = \sum(5, 6, 7, 8, 9) + \sum_x(10, 11, 14)$$

Řešení:



Byla by ještě nutná úprava s log.členy NAND např. na 4 vstupové a dvou-
vstupové – takže by bylo třeba : 4 x 4vstupové
8 x 2vstupové

Toto řešení se nemusí jevit jako nejvýhodnější.