HAZARDY V LOGICKÝCH SYSTÉMECH

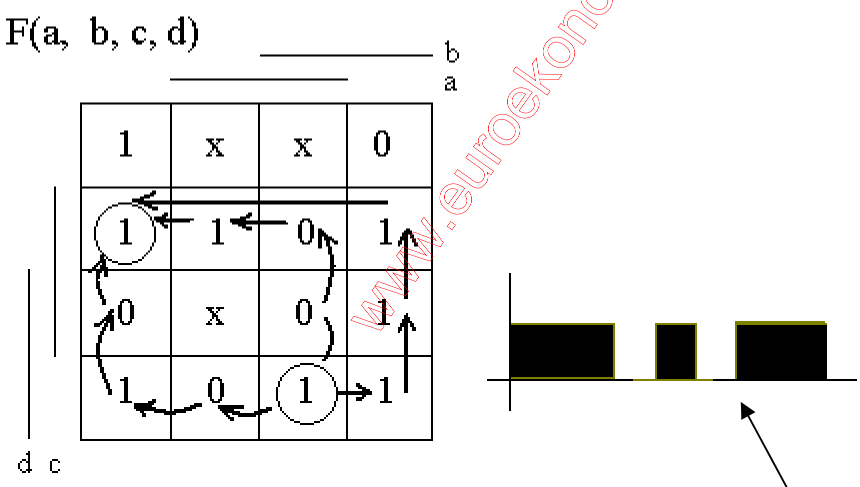
1. FUNKČNÍ HAZARD :

Při změně vstupního stavu vstupních proměnných, kdy se bude měnit více jak jedna proměnná - v reálné praxi však současná změna nenastává a ke změnám hodnot vstupních proměnných dochází posloupně.

<u>Příklad</u> : Je dána logická funkce F(a, b, c, d) = Σ(0, 4, 5, 6, 8, 10, 11, 14) + + Σ_x(1, 3, 13)

Uvažujme následující změnu vstupního stavu (vektoru) :

$\mathbf{d\,\bar{c}\,b\,a} \longrightarrow \mathbf{\bar{d}\,c\,\bar{b}\,\bar{a}}$ **při které má být zachován výstupní stav 1**



Nyní závisí výsledek na sledu změn vstupních proměnných :

a) pořadí změn bude : b → a → c → d , pak F bude : 10101
b) pořadí změn bude : c → d → b → a , pak F bude : 10011
c) pořadí změn bude : a → c → d → b , pak F bude : 11111 - bez hazardu!!

<u>Odstraňování funkčních hazardů:</u>

Nejefektivnější je použití <u>filtrů</u> !!

2. LOGICKÉ HAZARDY

2.1 STATICKÝ HAZARD

D: Jestliže pro dva sousední vstupní stavy (vektory) má být hodnota výstupní proměnné 1 nebo 0 , a jestliže existuje přechodný stav, během něhož se na výstupu objeví na krátkou dobu rozdílné hodnoty 0 nebo 1, pak hovoříme, že vznikl statický hazard v jedničce resp. v nule.

Statický hazard je způsobován časovým rozdílem průchodu signálu od vstupů k výstupním proměnným, tzn. projeví se reálné chování logických členů (a je použita negace vstupní proměnné). Pro zjištění podmínky vzniku použijme Shannonovy věty o rozkladu pro proměnnou x_1

$$y = F(x_1, x_2, \ldots, x_n) = F(x_1, X)$$ X je vstupní vektor bez x_1 $\{x_2, x_3, \ldots, x_n\}$

tedy : $$y = x_1 . F(1, X) + \overline{x_1} . F(0, X)$$

resp. : $$y = [x_1 + F(0, X)] . [\overline{x_1} + F(1, X)]$$

2.1 a) Statický hazard v jedničce - má být zachovaná jedničková hodnota na výstupu

Tedy : $F(1, X) = 1$, $F(0, X) = 1$

Pak $$\boxed{y = x_1 + \overline{x_1}}$$

a podmínka vzniku statického hazardu v jedničce je

$$\boxed{x_1 + \overline{x_1} = 0}$$ Pozor, zákon vyloučeného třetího

2.1 b) Statický hazard v nule - má být zachována nulová hodnota na výstupu

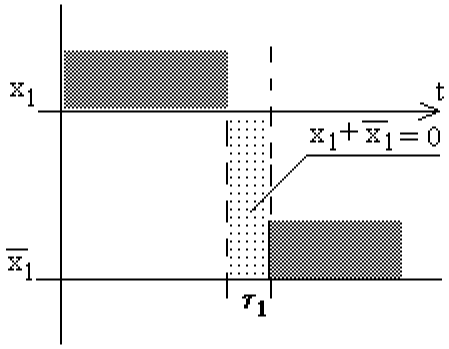
Tedy : $F(0, X) = 0$, $F(1, X) = 0$

Pak : $$\boxed{y = x_1 . \overline{x_1}}$$

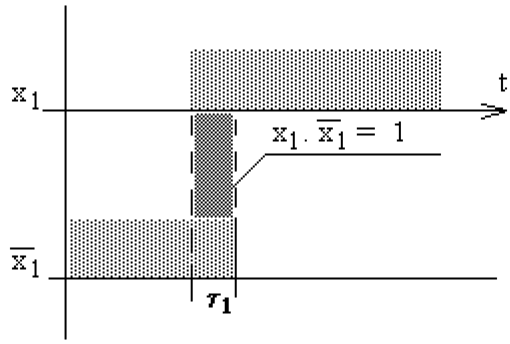
a podmínka vzniku statického hazardu v nule je

$$\boxed{x_1 . \overline{x_1} = 1}$$ Pozor, zákon vyloučeného třetího

Podmínky vzniku statických hazardů v časových diagramech



a)

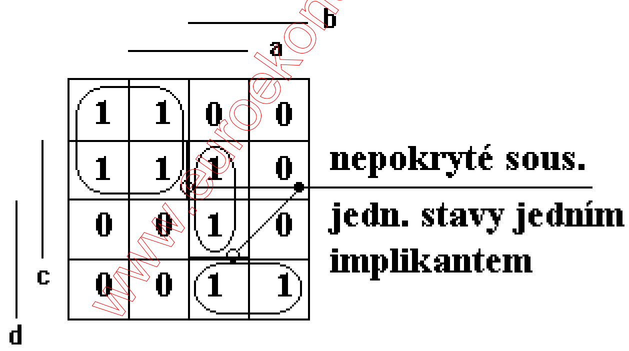


b)

<u>**Vyšetřování hazardů :**</u>

1) pomocí zobrazení funkce do mapy

F(a, b, c, d)



2) pomocí časových diagramů

3) u rozsáhlejších logických systémů jedině simulací, tj, pomocí vhodného simulačního systému na počítači (ku příkladu OrCAD)

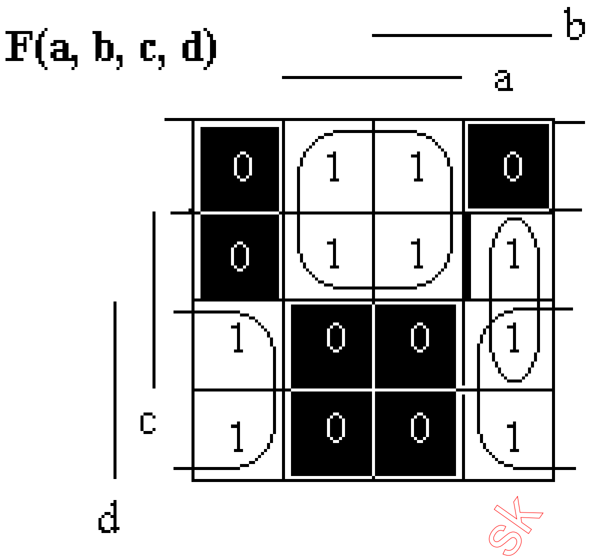
<u>**Odstraňování statických hazardů :**</u>

1) doplnění zpožďovacího členu do kratší větve - méně vhodné !

2) přidáním dalšího implikantu, který pokryje sousední jedničkové stavy. Potom již není funkce minimální!

PŘÍKLAD:

Mějme zadanou následující logickou funkci se 4 vstupními proměnnými:

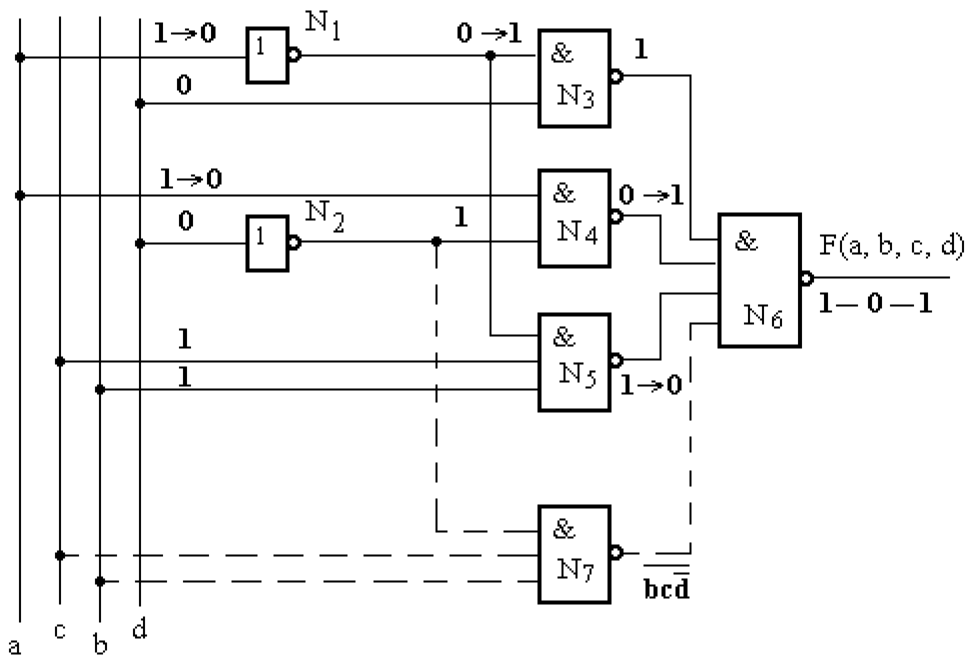


Minimální normální disjunktní forma(MNDF) :

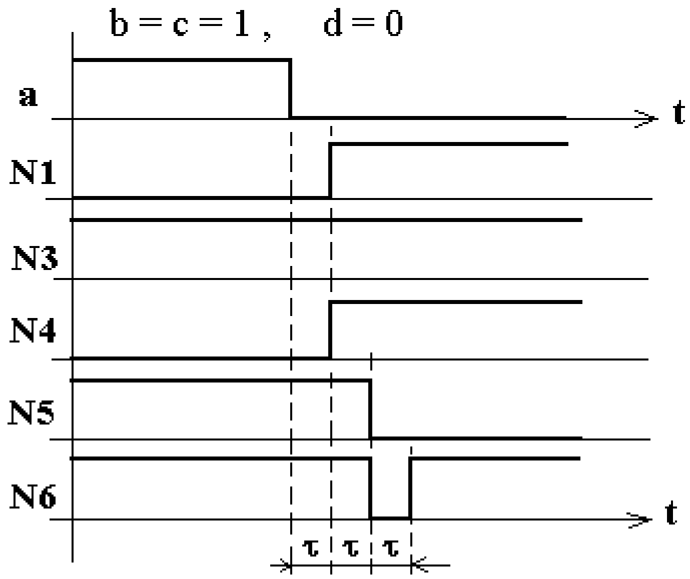
$$F(a, b, c, d) = a\bar{d} + \bar{a}d + \bar{a}bc$$

Úprava pro realizaci s logickými členy NAND :

$$F(a, b, c, d) = \overline{\overline{a\bar{d}} \cdot \overline{\bar{a}d} \cdot \overline{\bar{a}bc}}$$



Kontrola statického hazardu v časovém diagramu



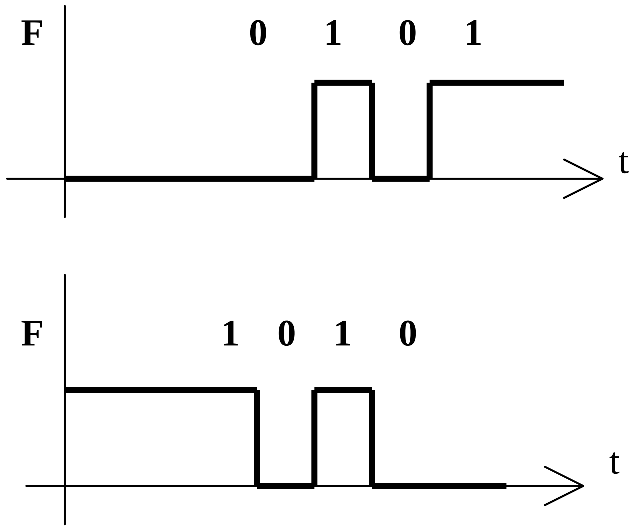
Odstranění statického hazardu v jedničce -- nejlépe redundantním přímým implikantem

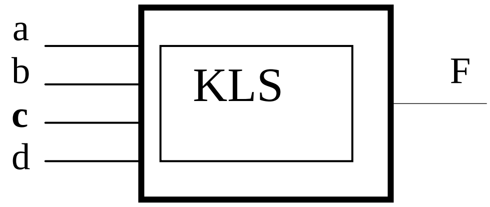
$$F'(a, b, c, d) = a\bar{d} + \bar{a}d + \bar{a}bc + bc\bar{d} \longrightarrow$$ **úprava schematu**

2.2 DYNAMICKÝ HAZARD

D : Kombinační systém může vykazovat dynamický hazard, jestliže pro dva sousední stavy resp. dvě sousední vstupní písmena, pro která má výstupní funkce přejít z hodnoty logické nuly (0) na hodnotu logické jedničky nebo obráceně, existuje přechodný stav, během něhož se na výstupu objeví na výstupu posloupnost změn výstupních hodnot **0 1 0 1** resp. **1 0 1 0.**

Znázornění časovým diagramem:





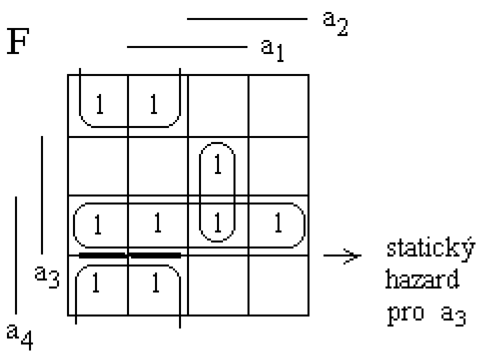
Vyšetřování dynamického hazardu

- nelze použít zobrazení v mapě ani jiný způsob jako u statického hazardu
- je třeba použít intuice a časových diagramů
- velmi často pomůže nejdříve vyhledání ***lokálního statického hazardu*** , který když odstraníme, odstraní se i dynamický hazard. Statický hazard a hlavně lokální statický hazard podmiňuje vznik dynamického hazardu.

<u>Příklad na dynamický hazard :</u>

Mějme zadanou následující logickou funkci $F(a_1, a_2, a_3, a_4) = \Sigma(0, 1, 7, 8, 9, 12, 13, 14, 15)$. Realizaci je třeba provést pomocí dvouvstupových NANDů a invertorů. Pokud existuji statické hazardy, je třeba nejdříve tyto odstranit.
Při existenci dynamického hazardu navrhněte jeho nejvýhodnější odstranění.

ŘEŠENÍ : Zobrazení funkce F v mapě a její minimalizace:



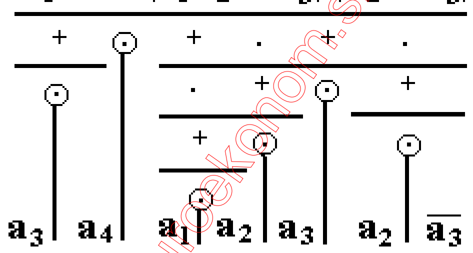
$F(a_1, a_2, a_3, a_4) = \bar{a}_2\bar{a}_3 + a_3a_4 + a_1a_2a_3 =$
$= a_3a_4 + (a_1a_2 + \bar{a}_3)(\bar{a}_2 + a_3)$

Po odstranění statického hazardu :

$F(a_1, a_2, a_3, a_4) = \bar{a}_2\bar{a}_3 + a_3a_4 + a_1a_2a_3 + \bar{a}_2a_4 = a_3a_4 + \bar{a}_2a_4 + (a_1a_2 + \bar{a}_3)(\bar{a}_2 + a_3)$

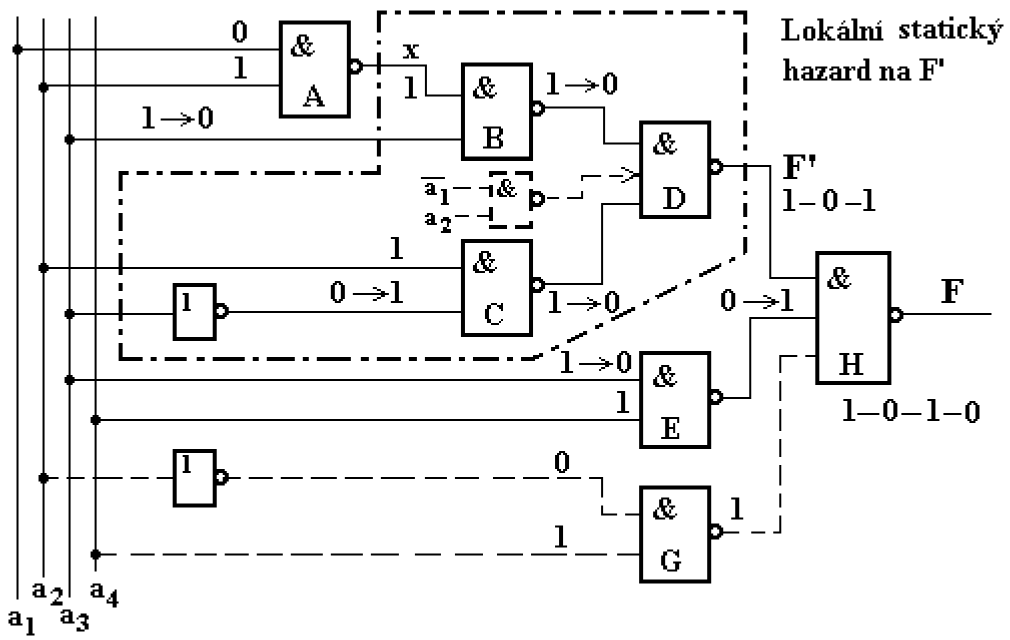
Úprava funkce pro realizaci s členy NAND - Rottovou mřížkou , resp.De Morganovým zákonem

$\mathbf{F(a_1, a_2, a_3, a_4) = a_3a_4 + (a_1a_2 + \bar{a}_3)(\bar{a}_2 + a_3)}$



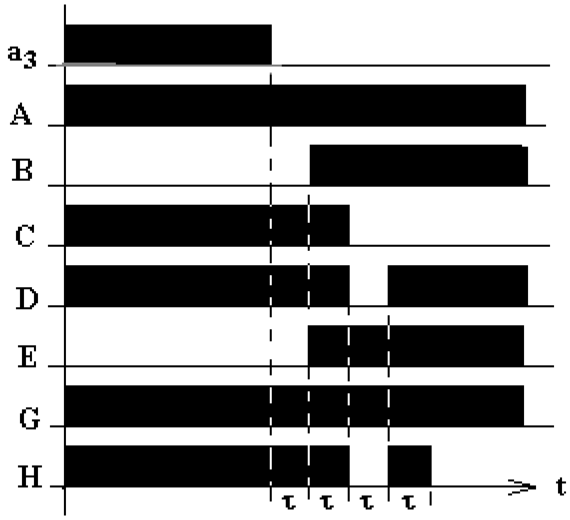
$F(a_1, a_2, a_3, a_4) = \overline{\overline{a_3a_4}\;\overline{a_2\bar{a}_3}\;\overline{a_3\overline{a_1a_2}}}$

Schéma zapojení:



Kontrola dynamického hazardu v časovém diagramu:

Uvažujeme stejná zpoždění τ pro všechny typy logických členů



Vyhledání lokálního statického hazardu pro $\overline{F'}$ $(F = \overline{F'} + a_3 a_4)$