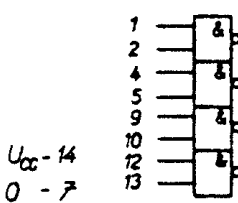
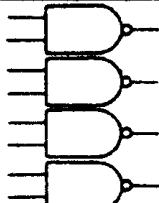
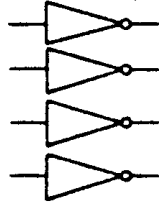
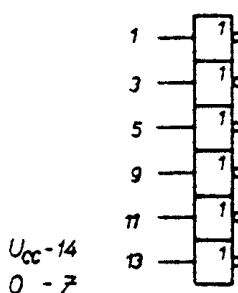
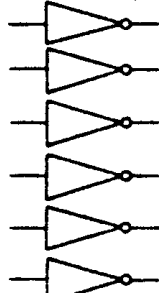
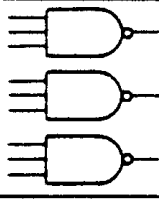
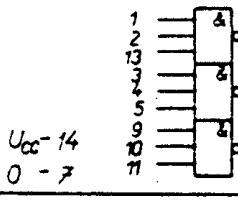
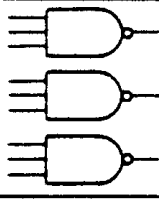
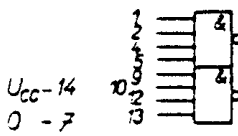
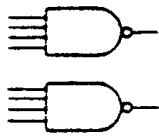
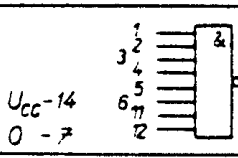
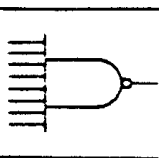
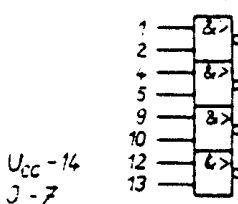
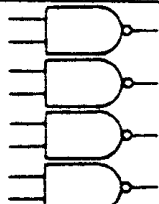
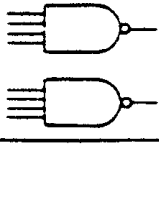
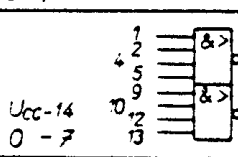
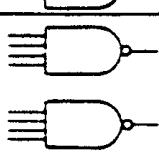


5. Kombinační obvody

Obvod se nazývá kombinační, jestliže jeho výstupy závisí pouze na vstupních kombinacích a ne na jejich předchozích hodnotách. Jediné kombinaci vstupních proměnných odpovídá jediná výstupní kombinace. Obvod nemá žádnou paměť předchozích stavů. Chceme-li navrhnout kombinační log. obvod, musíme především vědět, jakou činnost má uskutečňovat. Ze zadání bychom měli především vyčíst, kolik vstupních a výstupních proměnných má navrhovaný obvod. Tyto proměnné pak vhodně označíme písmeny a přiřadíme jim symbolické označení 0 a 1, vyjadřující, že daná činnost existuje nebo neexistuje. Příklady jednoduchých kombinačních obvodů jsou uvedeny v příkladech 15 - 18.

Základní integrované obvody

Typ	Název — funkce	Schematický znak	Schematický znak dřívější
MH7400	čtveřice dvouvstupových členů NAND $Y = \overline{A \cdot B}$		
MH7403	čtveřice dvouvstupových členů NAND s otevřeným kolektorem	$U_{cc} - 14$ $0 - 7$	
MH7404	šestice invertorů $Y = \overline{A}$		
MH7405	šestice invertorů s otevřeným kolektorem	$U_{cc} - 14$ $0 - 7$	
MH7410	trojice 3 vstupu členů NAND $Y = \overline{A \cdot B \cdot C}$		
MH7420	dvojice čtyřvstupových členů NAND $Y = \overline{A \cdot B \cdot C \cdot D}$		
MH7430	osmivstupový člen NAND $Y = \overline{A \cdot B \cdot C \cdot D \cdot E \cdot F \cdot G \cdot H}$		
MH7437	čtveřice dvouvstupových výkonových členů NAND $Y = \overline{A \cdot B}$		
MH7438	čtveřice dvojevstupových výkonových členů NAND s otevřeným kolektorem	$U_{cc} - 14$ $0 - 7$	
MH7440	dvojice čtyřvstupových výkonových členů NAND $Y = \overline{A \cdot B \cdot C \cdot D}$		

5.1. Kodéry a dekodéry

Typickým představitelem kombinačních log. obvodů jsou různé převodníky kódů, které převádějí informace z jedné kódované formy do jiné kódované formy. Kódem se rozumí pravidlo, podle něhož určité kombinaci nul a jedniček (nebo stavu L a H) přiřazujeme nějaké desítkové číslo. Kódy můžeme rozdělit do dvou hlavních skupin. Jsou to dvojkový (binární) kód a kód desítkový (BCD).

Dvojkový kód: přirozenému pořadí dvojkových čísel je přiřazeno přirozené pořadí čísel desítkových.

Kód BCD (Binary Coded Decimal – tj. dvojkově kódované desítkové číslo): v těchto kódech je desítkovým číslům 0 až 9 (tj. deset hodnot) přiřazeno deset dvojkových čísel o čtyřech bitech. Čísla jsou organizována v dekádách. Např. pro vyjádření desítkového čísla řádu 10^2 je nutno použít tři čtyřbitová čísla. Jedno (nejvíce vpravo) vyjadřuje jednotky, druhé desítky, třetí stovky.

Nejjednodušším kódem BCD je takový kód, v němž je přirozenému pořadí dvojkových čísel přiřazeno přirozené pořadí desítkových čísel v celém rozsahu tj. do čísla 9. Tento kód se označuje jako **BCD 1248**. Např. číslo 127 bude v tomto kódu vyjádřeno číslem 0001 0010 0111.

Kodéry převádějí desítková čísla do binární soustavy a dekodéry z binární soustavy do desítkové. Nejčastější úlohou je převod desítkového čísla na dvojkové v binárním kódu.

Příklad 19:

Navrhněte obvod pro převod dekadických čísel 0, 1, 2, 3 do soustavy binární.

Řešení:

Nejprve si sestavíme tabulku stavů: Jeden sloupec (des.) vyhradíme pro desítková čísla. Do dalších sloupců těmto desítkovým číslům přiřadíme kombinace stisknutého tlačítka (např. tlačítko C představuje číslo 1, tlačítko B představuje číslo 2 a tlačítko A číslo 3). Stisk tlačítka představuje log. 1

Des.	A	B	C	Y	Z
0	0	0	0	0	0
1	0	0	1	0	1
2	0	1	0	1	0
	0	1	1	-	-
3	1	0	0	1	1
	1	0	1	-	-
	1	1	0	-	-
	1	1	1	-	-

Jedná se o neúplnou tabulku stavů. Tabulku prepíšeme do dvou map, napíšeme rovnice a navrhne obvod.

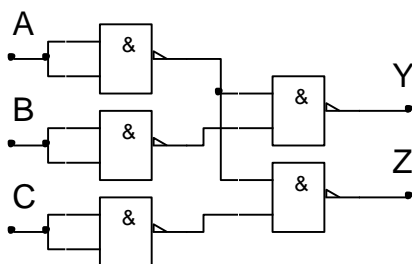
		C	
		0	1
AB	00	0	0
	01	1	1
	11	1	1
	10	1	1

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

$$Y = A + B \quad Z = A + C$$

převáděno na Scheffer. funkci:

$$Y = \overline{\overline{A \cdot B}} \quad Z = \overline{\overline{A \cdot C}}$$



Opačnou úlohou je převod dvojkového čísla v určitém kódu na číslo desítkové.

Příklad 21: Navrhněte převodník dvojkového čísla do desítkového 1 ze 4.

Řešení: Sestavíme si tabulku stavů. Každému dvojkovému číslu na vstupu zde odpovídá jeden výstup, jehož stav je odlišný od stavu výstupů ostatních, tj. je aktivní. V daném uspořádání dle tabulky je aktivní výstup ve tvaru log. 0 (L), což je výhodné z hlediska součástkové základny. Výstupem ve stavu log. 0 lze také dobře řídit (spínat) zátěž např. zobrazovací prvek. Ty výstupy, které nejsou aktivní mají hodnotu log. 1 (H).

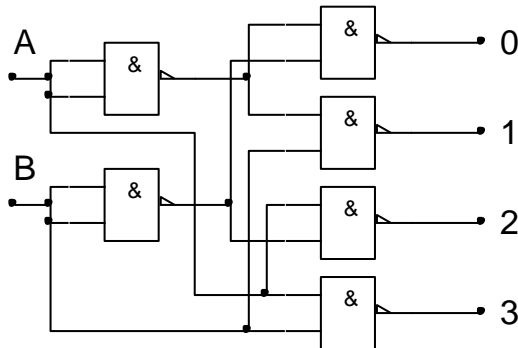
A	B	0	1	2	3
0	0	0	1	1	1
0	1	1	0	1	1
1	0	1	1	0	1
1	1	1	1	1	0

$$0 = \overline{\overline{A \cdot B}}$$

$$2 = \overline{\overline{A \cdot B}}$$

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

$$3 = \overline{\overline{A \cdot B}}$$



Vztahy určují takové negace v součinech veličin **A**, **B**, aby byl stav daného výstupu log. **0 (L)**.
Např. pro výstup 2 musíme nejprve hodnotu **A (1)** násobit negovanou hodnotou **B (0)**. Výsledek tohoto součinu je roven **1 (H)** a tento pak musíme negovat, abychom dostali výsledný stav log. **0 (L)**.

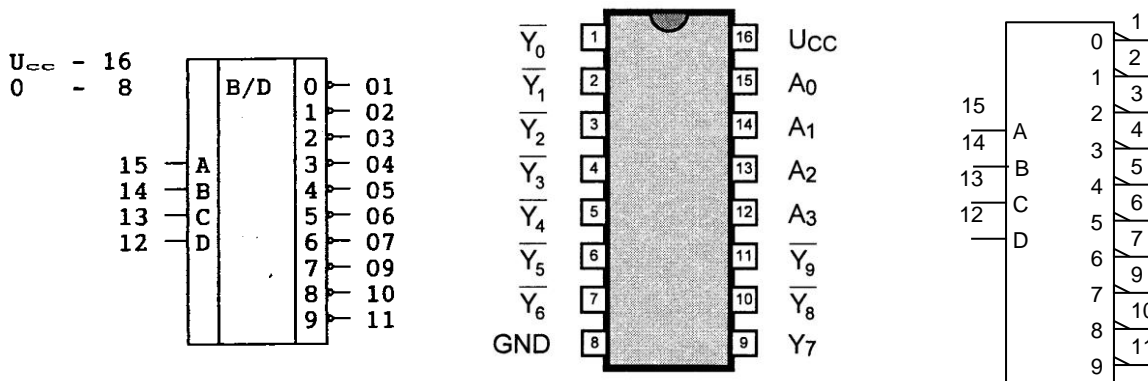
Protože je ve všech vztazích použita negace součinu, použijeme k realizaci převodníku logické členy NAND.

5.2. Integrované dekodéry

Dekodéry se často v zařízeních číslicové techniky opakují. To platí zejména o dekodérech 1 z 10. Proto byly některé dekodéry realizovány formou integrovaných obvodů.

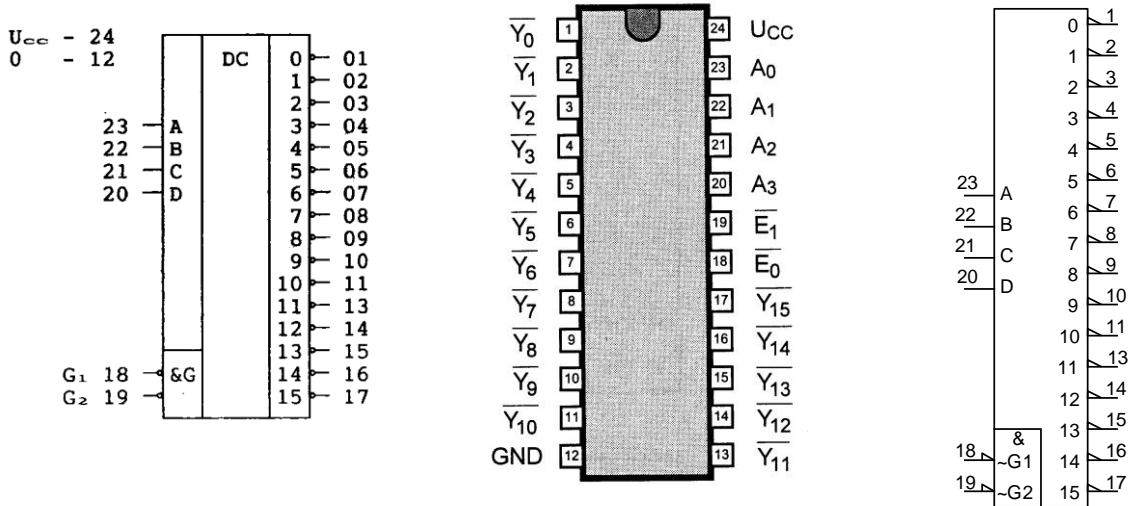
Dekodér 7442: je to převodník z kódu **BCD** na kód **1 z 10**, má tedy obdobnou funkci jako dekodér podle obr. 5. Na rozdíl od výše popsaného dekodéru je však v obvodu **7442** zajištěno úplné dekódování, může tedy zpracovávat vstupní kombinace odpovídající číslům 10 až 15. Jsou-li tato čísla přivedena, zůstávají všechny výstupy obvodu ve stavu **H**, tj., nedojde k chybnému dekódování. Příslušná pravdivostní tabulka je podobná tab. 4. Všem následujícím kombinacím vstupních proměnných odpovídají výstupní proměnné ve stavu **H**.

Dekodér obsahuje 10 log. členů NAND a 8 invertorů. Je montován v plastickém pouzdře s šestnácti vývody. Schématický znak je na obr. 6.



Obr. 6: Schématický znak dekodéru MH7442 a číslování vývodů pouzder TTL

Dekodér 74154: umožňuje převádět dvojkové číslo o čtyřech bitech na kód **1 z 16**. Vyčerpává tedy všechny možnosti čtyřbitového čísla. Dekódování je založeno na zcela shodném principu, jak byl dříve popsán. Obvod 74154 může kromě funkce dekodéru vykonávat též funkci demultiplexeru - proto je opatřen dvěma vybavovacími vstupy **G₁** a **G₂**. Má-li obvod pracovat jako dekodér, musí být tyto vstupy na úrovni **L**. Je-li alespoň jeden z těchto vstupů na úrovni **H**, jsou všechny výstupy na úrovni **H**.



Obr. 7: Schématický znak dekodéru 74154 a číslování vývodů pouzdra s dvacetičtyřmi vývody

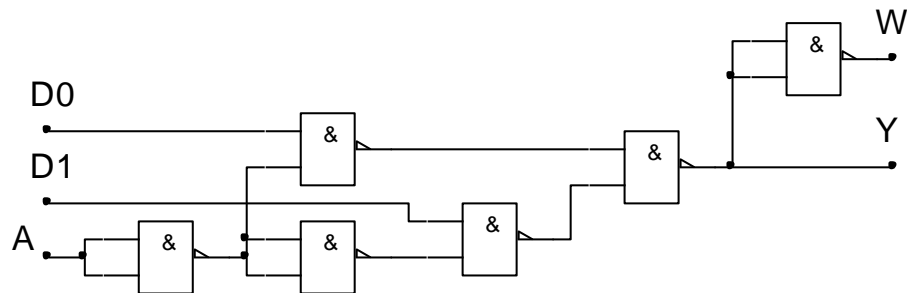
5.3. MULTIPLEXERY A DEMULTIPLEXERY

Multiplexer je vlastně obdoba přepínače, kterým se *informace přítomná na jednom z několika vstupů přenáší na jediný výstup (který byl zvolen adresou)*. Vstup, ze kterého má být informace přenesena, se určí adresou ve dvojkovém kódu na adresovacím vstupu. Jde tedy o obvod pro výběr dat. Dále mohou být doplněny vstupy pro odstavení obvodu (používají se pro kaskádní řazení) **Multiplexery** jsou kombinačními sítěmi a jsou podobné dekodérům (někdy je označujeme jako *selektory dat*).

Příklad 22: Navrhněte dvoukanálový multiplexer

Řešení: Daný multiplexer má dva vstupy datové a jeden vstup adresovací

A	D ₀	D ₁	Y	W
L	L	X	L	H
L	H	X	H	L
H	X	L	L	H
H	X	H	H	L

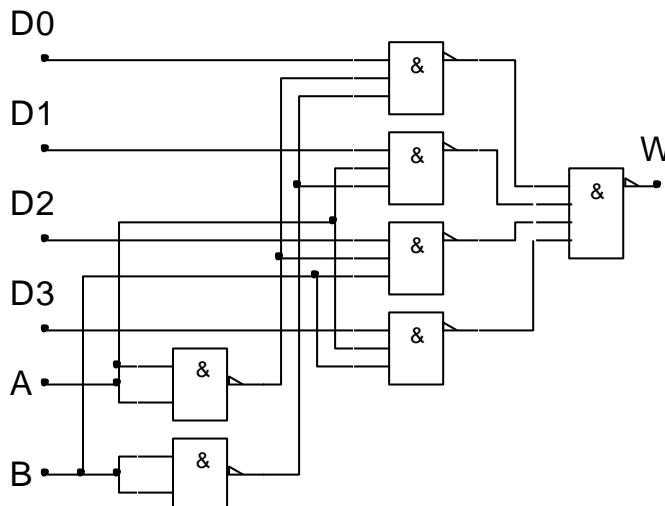


Z uvedené tabulky a odpovídajícího zapojení vyplývá, že výstupy může po zaadresování pomocí vstupu **A** (výběrový vstup) ovlivnit vždy jen jeden z datových vstupů. Je-li **A = L**, ovlivňuje výstupy hodnota vstupního datového signálu **D₀**, je-li **A = H**, ovlivňuje výstupy hodnota vstupního datového signálu **D₁**. Dále si ještě uvedeme čtyřkanálový multiplexer (obr. 8)

Funkce **demultiplexeru** je rovněž podobná funkci přepínače, ale je opačná proti multiplexeru.

Zde z jednoho vstupu převádíme signál do několika výstupů, kde volbu výstupu opět zajišťujeme adresovacími vstupy.

B	A	D ₀	D ₁	D ₂	D ₃	W
L	L	L	X	X	X	L
L	L	H	X	X	X	H
L	H	X	L	X	X	L
L	H	X	H	X	X	H
H	L	X	X	L	X	L
H	L	X	X	H	X	H
H	H	X	X	X	L	L
H	H	X	X	X	H	H

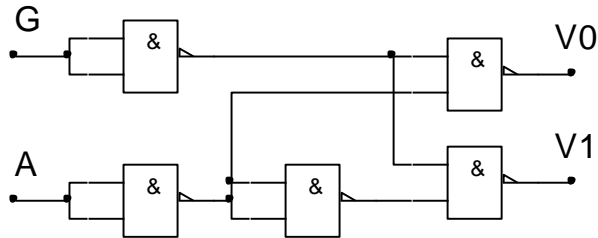


Obr.8: Čtyřkanálový multiplexer

Příklad 23. Navrhněte dvoukanálový demultiplexer

Řešení: Uvažovaný demultiplexer bude mít jeden vstup vybavovací (označen: **G**), jeden adresovací (označen: **A**) a dva výstupy (označeny: **V₀** a **V₁**)

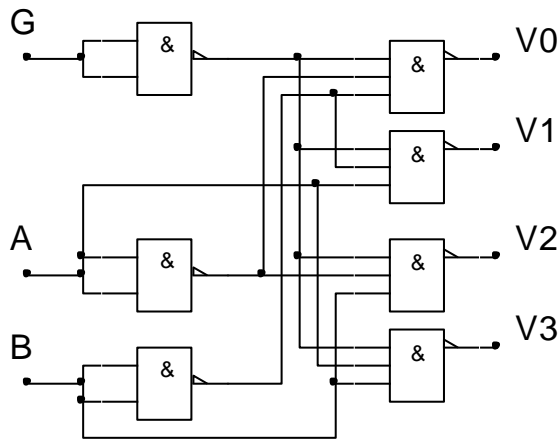
G	A	V ₀	V ₁
L	L	L	H
L	H	H	L
H	X	H	H



Z uvedeného schématu a tabulky je zřejmé nastavení jednoho z výstupů do úrovně **L** v závislosti na nastavení adresovacího vstupu a přivedení úrovně **L** na vybavovací vstup.

Dále si ještě uvedeme čtyřkanálový demultiplexer (obr. 9) a jeho funkční tabulku.

G	A	B	V ₀	V ₁	V ₂	V ₃
L	L	L	L	H	H	H
L	H	L	H	L	H	H
L	L	H	H	H	L	H
L	H	H	H	H	H	L
H	X	X	H	H	H	H



Obr. 9: Čtyřkanálový demultiplexer

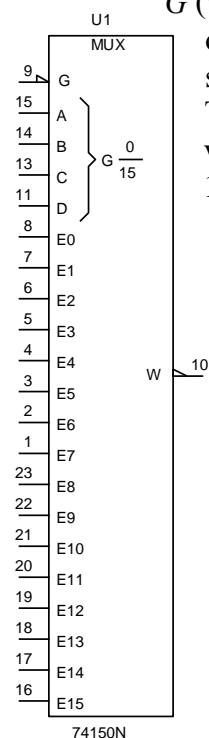
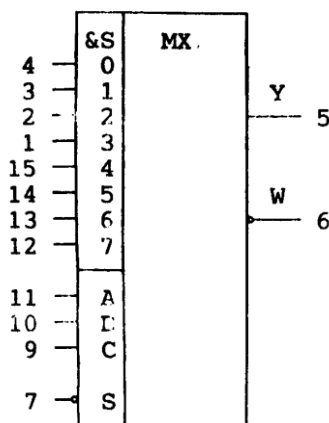
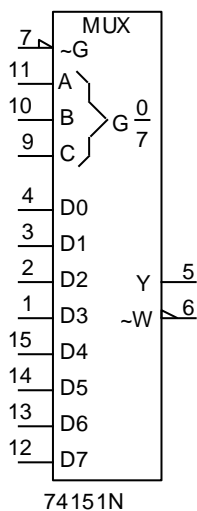
5.4. Integrované multiplexery

Tak jako dekodéry, tak jsou také formou integrovaných obvodů realizovány některé multiplexery a demultiplexery.

Multiplexer 74151: je osmikanálový. Má tedy osm vstupů, označených **D₀** až **D₇**, na něž se přivádějí informace. Vstup, z něhož má být přenos uskutečněn se volí pomocí tří adresových vstupů

A, B, C. Obvod je také opatřen vybavovacím vstupem řazení), který umožňuje ovládat výstup nezávisle na stavu vstup umožňuje propojovat větší počet multiplexerů. Má-li vstup na výstup, musí být vybavovací vstup na úrovni **L**. výstupy vzájemně opačné. Lze tedy odebírat i negovanou informaci. Uspořádání je patrné z obr. 10 a

\bar{G} (vstupem kaskádního ostatních vstupů. Tento se přenést informace ze Tento obvod má dva výstupní 11 a funkce z tab. 6.



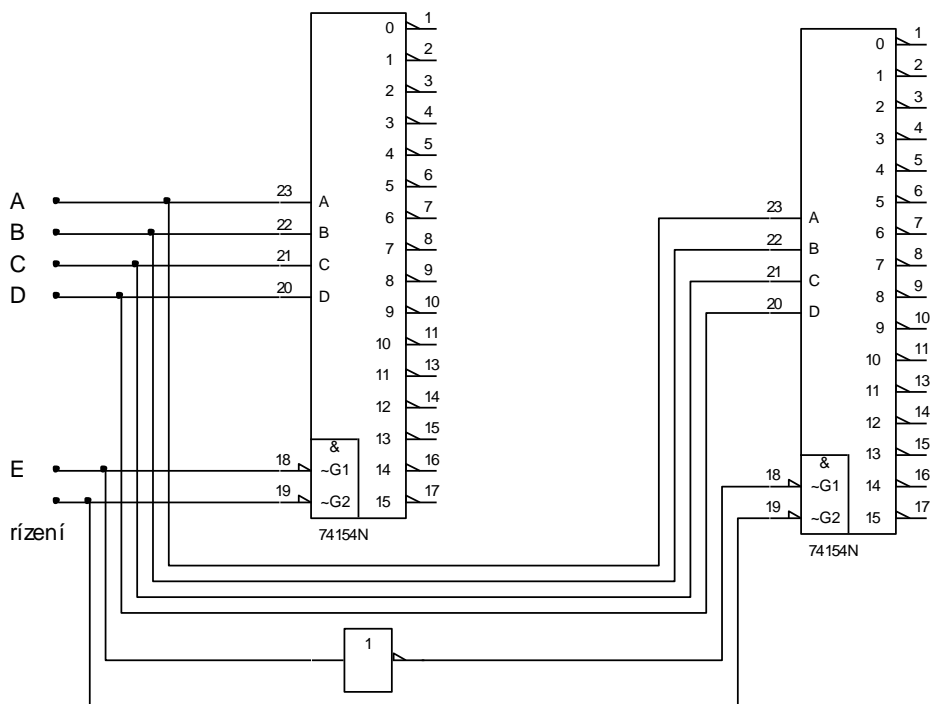
Obr. 10: Schematický znak multiplexeru 74151 a 74150

Multiplexer 74150: tento obvod má shodnou funkci jako obvod předchozí, počet informačních vstupů je však rozšířen na 16. Na rozdíl od MH74151 má však jen jeden výstup. Pouzdro tohoto obvodu má 24 vývodů.

Demultiplexer 74154: jeho schéma je uvedeno na obr. 7 a pravdivostní tabulka v tab. 5.

Tento obvod umožňuje rovněž funkci dekodéru, jak již bylo popsáno dříve.

S pomocí tohoto obvodu lze realizovat větší dekodéry nebo demultiplexery. Na obrázku je ukázáno zapojení dekodéru demultiplexeru pro kód **1 z 32**, sestavený ze dvou obvodů 74154. Signály vstupů **A, B, C, D** jsou přivedeny přímo na oba obvody. Další signál **E** (pátý bit dvojkového čísla) je přiveden přímo na vstup **G₁** prvního obvodu a přes inverter na vstup **G₁** druhého obvodu. Činnost dekodéru je možno ovládat dalším signálem (signál „**Rízení**“) na vstupu **G₂**. Při funkci demultiplexeru se signál ze vstupu **G₂** přenesou na jeden z 32 výstupů, který je vybrán adresou **A, B, C, D, E**.



6. Hazardy

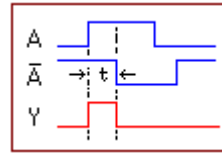
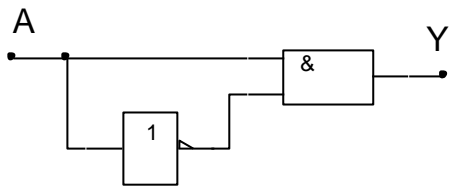
Hazardní stavy jsou nežádoucí stavy v číslicových obvodech, projevující se výskytem nesprávné hodnoty log. veličiny na výstupu, a to buď dočasně nebo trvale. Příčinou jejich vzniku je obvykle vliv různých zpoždění v jednotlivých částech logické soustavy, způsobující odchylku skutečného chování obvodu od ideálního, předepsaného chování. Rozlišujeme hazardy statické, které způsobují jednorázové, dočasné chyby hodnoty výstupní funkce, a hazardy dynamické, které mohou vznikat ve vícecestupňových kombinačních logických obvodech a způsobují několik po sobě následujících změn hodnoty výstupní proměnné. V kombinačních logických obvodech mohou vznikat oba druhy hazardů.

Vznik hazardu si objasníme na jednoduchém obvodu, uvedeném na obr.12. Na jeden vstup součinného logického členu je přiveden signál A a na druhý vstup signál \bar{A} , který jsme získali v invertoru. Při realizaci funkce: $Y = A \cdot \bar{A}$ ideálními prvky by na výstupu měla být vždy hodnota $Y = 0$, protože platí zákon doplňku: $A \cdot \bar{A} = 0$.

Při realizaci skutečnými elektronickými log. členy se však projevují zpoždění, vlivem kterých se změni veličina z A na \bar{A} až za určitou dobu τ . Ve skutečném případě na výstupu vznikne tedy impuls, který by v ideálním případě nemohl nastat. Během přechodného děje vznikne po určitou dobu paradoxní výsledek:

$$Y = A \cdot \bar{A} = 1$$

což je postačující podmínka pro vznik hazardu.



Obr. 12: Objasnění vzniku hazardu

Statické hazardy nastávají při realizaci logických funkcí, v jejichž algebraických vztazích se vyskytují tzv. sousední stavy. Při syntéze obvodů musíme dbát na vyloučení možnosti jejich vzniku. Existuje několik způsobů:

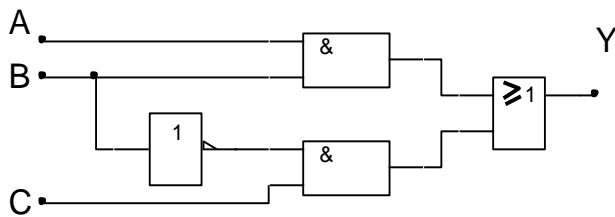
- Logické soustavy realizujeme tak, aby se nevyskytovaly situace, které vedou k hazardům.
- Do algebraické rovnice zavedeme další kombinace vstupních proměnných, které zajistí setrvání výstupu v log. 0, nebo 1 během přechodového děje, jak to správná funkce obvodu vyžaduje.
- Na výstup zapojíme časový filtr, nepropouštějící impulsy kratší než je přípustná hodnota. Jsou-li impulsy projevující se jako hazardy kratší než je přípustná hodnota, neprojeví se na výstupu. Touto úpravou se však snižuje maximální operační kmitočet daného logického obvodu.

Příklad24: Sestavte logické schéma pro realizaci logické funkce:

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

Tato funkce se dá realizovat jednoduchým obvodem na obr. 13.

Obr. 13: Realizace log. funkce: $Y = A \cdot B + \bar{B} \cdot C$



Tento obvod není odolný proti hazardu v log. 1 při změně B, protože členy $A \cdot B$ a $C \cdot \bar{B}$ jsou "sousední". Vztah ale můžeme upravit rozšířením na úplný tvar:

$$Y = AB(C + \bar{C}) + C\bar{B}(A + \bar{A})$$

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

Po úpravě (pomocí Karnaughovy mapy):

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

Schéma realizované podle této upravené logické funkce obsahuje člen AC, který zajistí odolnost obvodu (obr.14) proti vzniku hazardu, protože při $A = C = 1$ bude $AC = 1$, a tedy také $Y = 1$, i kdyby během přechodného děje nastala paradoxní situace, že: $B = \bar{B} = 0$

Obr. 14: Realizace log. funkce $Y = A \cdot B + C \cdot \bar{B} + A \cdot C$

