Algebra Boole’a i jej zastosowania

Wprowadzenie

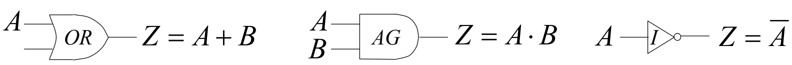
Niech dany będzie zbiór dwuelementowy Ω, którego elementy oznaczymy symbolami 0 oraz 1, tj. $\Omega = \{0, 1\}$. W zbiorze tym określamy działania <u>*sumy*</u> $+ : \Omega \times \Omega \to \Omega$, <u>*iloczynu*</u> $\cdot : \Omega \times \Omega \to \Omega$ oraz <u>*dopełnienia*</u> $\bar{} : \Omega \to \Omega$. A mianowicie, przyjmujemy:

A	B	$A+B$
0	0	0
0	1	1
1	0	1
1	1	1

A	B	$A \cdot B$
0	0	0
0	1	0
1	0	0
1	1	1

A	$\overline{A}$
0	1
1	0

Zbiór Ω wraz z powyższymi działaniami nazywa się <u>*algebrą Boole’a*</u>. Jedną z możliwych jej realizacji są układy elektroniczne zwane <u>*bramkami*</u>. Analogicznie do w/w działań wprowadzamy bramki *OR (Or Gate)*, *AG (And Gate)*, *I (Inverter)* oznaczane na schematach następującymi symbolami:



i zdefiniowane takimi samymi tabelami, jak powyżej.

<u>*Twierdzenie*</u>.
Dla dowolnych elementów *X*, *Y*, *Z* algebry Ω zachodzą następujące równości:

(1) $0 + X = X$;
(2) $1 + X = 1$;
(3) $X + X = X$;
(4) $X + \overline{X} = 1$;
(5) $0 \cdot X = 0$;
(6) $1 \cdot X = X$;
(7) $X \cdot X = X$;
(8) $X \cdot \overline{X} = 0$;
(9) $\overline{(\overline{X})} = X$;
(10) $X + Y = Y + X$;
(11) $X + (Y + Z) = (X + Y) + Z$;
(12) $X \cdot Y = Y \cdot X$;
(13) $X \cdot (Y \cdot Z) = (X \cdot Y) \cdot Z$;

(14) $X \cdot (Y + Z) = (X \cdot Y) + (X \cdot Z)$;

(15) $X + (Y \cdot Z) = (X + Y) \cdot (X + Z)$;

(16) $\overline{(X + Y)} = \overline{X} \cdot \overline{Y}$; (<u>*prawo de'Morgana*</u>)

(17) $\overline{(X \cdot Y)} = \overline{X} + \overline{Y}$; (<u>*prawo de'Morgana*</u>).

Powyższe własności można udowodnić m.in. <u>*metodą zero-jedynkową*</u> polegającą na sprawdzeniu wszystkich możliwych przypadków, których jest skończona ilość. Oto przykładowy dowód własności (11):

X	*Y*	*Z*	*Y*+*Z*	*X*+(*Y*+*Z*)	*X*+*Y*	(*X*+*Y*)+*Z*
0	0	0	0	0	0	0
0	0	1	1	1	0	1
0	1	0	1	1	1	1
0	1	1	1	1	1	1
1	0	0	0	1	1	1
1	0	1	1	1	1	1
1	1	0	1	1	1	1
1	1	1	1	1	1	1

Dzięki równościom (11) i (13) uogólniamy indukcyjnie dodawanie i mnożenie na dowolną skończoną liczbę argumentów. A mianowicie, jeżeli $n > 2$, to przyjmujemy:

$$A_1 + \ldots + A_n = (A_1 + \ldots + A_{n-1}) + A_n ;$$

$$A_1 \cdot \ldots \cdot A_n = (A_1 \cdot \ldots \cdot A_{n-1}) \cdot A_n .$$

W analogiczny sposób uogólniamy bramki *OR* i *AND*. Ponadto przyjmujemy taką samą umowę dotyczącą kolejności działań i opuszczania znaku mnożenia, jak w tradycyjnej algebrze, co pozwala na zredukowanie zbędnych nawiasów i uproszczenie zapisów.

<u>*Twierdzenie*</u>.

Dla dowolnych elementów *X*, *Y*, *Z* algebry Ω zachodzą następujące równości:

(18) $X + X\,Z = X$;

(19) $X(X + Z) = X$;

(20) $X + \overline{X}\,Y = X + Y$;

(21) $X\,Y + Y\,Z + \overline{Y}\,Z = X\,Y + Z$.

<u>*Dowody*</u>. Zastosujemy bardziej efektywną metodę polegającą na wykorzystaniu własności już udowodnionych, do których zaliczymy własności (1) - (17).

Ad (18)

$$X + X\,Z = X \cdot 1 + X \cdot Z = X \cdot (1 + Z) = X \cdot 1 = X .$$

Ad (19)

$$X \cdot (X + Z) = (X \cdot X) + (X \cdot Z) = X + X\,Z \overset{(18)}{=} X .$$

Ad (20)

$$X + Y = 1 \cdot (X + Y) = (X + \overline{X}) \cdot (X + Y) = X \cdot X + X \cdot Y + \overline{X} \cdot X + \overline{X} \cdot Y =$$

$$(X + X\,Y) + (0 + \overline{X} \cdot Y) \overset{(18)}{=} X + \overline{X} \cdot Y .$$

Ad (21)

$$X\,Y + Y\,Z + \overline{Y}\,Z = X\,Y + (Y\,Z + \overline{Y}\,Z) = X\,Y + Z(Y + \overline{Y}) = X\,Y + Z \cdot 1 = X\,Y + Z .$$

W zagadnieniach dotyczących zastosowań algebry Boole'a kluczowe znaczenie ma przekształcanie wyrażeń do najprostszej postaci i to odpowiedniego kształtu. Okazuje się, że każde takie wyrażenie zawsze daje się sprowadzić do jednej z dwóch tzw. <u>*postaci kanonocznych*</u>: sumy iloczynów bądź iloczynu sum pojedynczych argumentów bądź ich dopełnień.

Przykłady. Uprościmy dwa wyrażenia algebry Boole'a.

a)

$$W=(X+Y)(X+\overline{Y})(\overline{X}+Z)=(X\,X+X\,\overline{Y}+Y\,X+Y\,\overline{Y})(\overline{X}+Z)=(X+X\,\overline{Y}+X\,Y)(\overline{X}+Z)=$$
$$X\,\overline{X}+X\,Z+X\,\overline{Y}\,\overline{X}+X\,\overline{Y}\,Z+X\,Y\,\overline{X}+X\,Y\,Z=X\,Z+(X\,\overline{Y}\,Z+X\,Y\,Z)=X\,Z+X\,Z\,(\overline{Y}+Y)=$$
$$X\,Z+X\,Z=X\,Z.$$

Inne, prostsze rozwiązanie:

$$W=\big((X+Y)(X+\overline{Y})\big)(\overline{X}+Z)=(X+Y\,\overline{Y})(\overline{X}+Z)=X\,(\overline{X}+Z)=X\,\overline{X}+X\,Z=X\,Z.$$

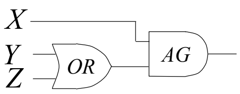
b)

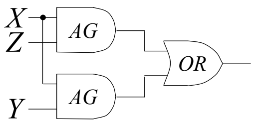
$$V=X(Y\,Z+\overline{Y}\,Z+Y\,\overline{Z})=X\big((Y\,Z+\overline{Y}\,Z)+(Y\,Z+Y\,\overline{Z})\big)=X\big(Z\,(Y+\overline{Y})+Y\,(Z+\overline{Z})\big)=X\,(Z\cdot 1+Y\cdot 1)=$$
$$X\,(Y+Z)$$

lub

$$V=X\,Y+X\,Z.$$

Pierwszy z wyników przedstawia wyrażenie V w postaci kanonicznej iloczynu sum (trzeba założyć, że X jest sumą jednoskładnikową), a drugi z tych wyników – w postaci kanonicznej sumy iloczynów. Przy okazji warto postawić pytanie, która wersja rozwiązania jest prostsza? Aby na nie odpowiedzieć, przedstawmy ich realizację przy pomocy bramek:





Teraz jest jasne, że pierwsze z rozwiązań jest prostsze, gdyż jego realizacja wymaga jednej bramki mniej.

Następnym ważnym problem z uwagi na zastosowania w informatyce jest projektowanie wyrażeń algebry Boole'a, które spełniają wymagane warunki wejścia/wyjścia. Wyjaśnimy to przy pomocy serii przykładów.

Przykład. Znajdziemy wyrażenie zależne na wejściu od zmiennych *X*, *Y* oraz wyprowadzające na wyjściu wyrażenie *Z* tak, aby zachodziły zależności:

Input		*Output*
X	*Y*	*Z*
0	0	1
0	1	0
1	0	1
1	1	1

Aby otrzymać rozwiązanie w postaci sumy iloczynów dodajemy dodatkową kolumnę zawierającą utworzone w odpowiedni sposób iloczyny:

Input		*Output*	*Iloczyny*
X	*Y*	*Z*	
0	0	1	$\overline{X}\cdot\overline{Y}$
0	1	0	$\overline{X}\cdot Y$
1	0	1	$X\cdot\overline{Y}$
1	1	1	$X\cdot Y$

Szukanym rozwiązaniem jest suma wyrażeń z ostatniej kolumny wybranych z tych wierszy, dla których $Z=1$:

$$Z = \overline{X} \cdot \overline{Y} + X \cdot \overline{Y} + X \cdot Y .$$

Uprośćmy je:

$$Z = (\overline{X} \cdot \overline{Y} + X \cdot \overline{Y}) + (X \cdot \overline{Y} + X \cdot Y) = \overline{Y}\,(\overline{X} + X) + X\,(\overline{Y} + Y) = \overline{Y} + X = X + \overline{Y} .$$

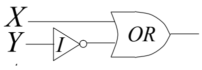
Aby otrzymać rozwiązanie w postaci iloczynu sum dodajemy dodatkową kolumnę zawierającą utworzone w odpowiedni sposób sumy:

Input		*Output*	*Sumy*
X	*Y*	*Z*	
0	0	1	$X + Y$
0	1	0	$X + \overline{Y}$
1	0	1	$\overline{X} + Y$
1	1	1	$\overline{X} + \overline{Y}$

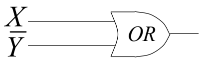
Szukanym rozwiązaniem jest iloczyn wyrażeń z ostatniej kolumny wybranych z tych wierszy, dla których $Z = 0$:

$$Z = X + \overline{Y} .$$

Widać, że otrzymaliśmy dokładnie takie samo rozwiązanie. Jego realizacja przy pomocy bramek wygląda następująco:



lub



<u>*Przykład*</u>. Znajdziemy wyrażenie zależne na wejściu od zmiennych *X*, *Y*, *Z* oraz wyprowadzające na wyjściu wyrażenie *A* tak, aby zachodziły zależności:

Input			*Output*
X	*Y*	*Z*	*A*
0	0	0	1
0	0	1	0
0	1	0	1
0	1	1	0
1	0	0	1
1	0	1	0
1	1	0	1
1	1	1	0

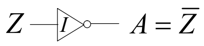
Znajdziemy oba rozwiązania w postaciach kanonicznych:

Input			*Output*	*Iloczyny*	*Sumy*
X	*Y*	*Z*	*A*		
0	0	0	1	$\overline{X}\,\overline{Y}\,\overline{Z}$ *	$X+Y+Z$
0	0	1	0	$\overline{X}\,\overline{Y}\,Z$	$X+Y+\overline{Z}$ *
0	1	0	1	$\overline{X}\,Y\,\overline{Z}$ *	$X+\overline{Y}+Z$
0	1	1	0	$\overline{X}\,Y\,Z$	$X+\overline{Y}+\overline{Z}$ *
1	0	0	1	$X\,\overline{Y}\,\overline{Z}$ *	$\overline{X}+Y+Z$
1	0	1	0	$X\,\overline{Y}\,Z$	$\overline{X}+Y+\overline{Z}$ *
1	1	0	1	$X\,Y\,\overline{Z}$ *	$\overline{X}+\overline{Y}+Z$
1	1	1	0	$X\,Y\,Z$	$\overline{X}+\overline{Y}+\overline{Z}$ *

Rozwiązanie w postaci sumy iloczynów:
$A=(\overline{X}\,\overline{Y}\,\overline{Z}+\overline{X}\,Y\,\overline{Z})+(X\,\overline{Y}\,\overline{Z}+X\,Y\,\overline{Z})=\overline{X}\,\overline{Z}\,(\overline{Y}+Y)+X\,\overline{Z}\,(\overline{Y}+Y)=\overline{X}\,\overline{Z}+X\,\overline{Z}=\overline{Z}\,(\overline{X}+X)=\overline{Z}$.

Rozwiązanie w postaci iloczynu sum:
$A=\big((X+Y+\overline{Z})(X+\overline{Y}+\overline{Z})\big)\big((\overline{X}+Y+\overline{Z})(\overline{X}+\overline{Y}+\overline{Z})\big)=\big((X+\overline{Z})+(Y\cdot\overline{Y})\big)\big((\overline{X}+\overline{Z})+(Y\cdot Y)\big)=$
$(X+\overline{Z})(\overline{X}+\overline{Z})=\overline{Z}+(X\cdot\overline{X})=\overline{Z}$.

Również teraz otrzymaliśmy dokładnie takie samo rozwiązanie. Jego realizacja wymaga tylko jednego inwertera:



Przykład.
Znajdziemy wyrażenie zależne na wejściu od zmiennych *X*, *Y*, *Z* oraz wyprowadzające na wyjściu wyrażenie *A* tak, aby zachodziły zależności:

Input			*Output*
X	*Y*	*Z*	*A*
0	0	0	0
0	0	1	0
0	1	0	1
0	1	1	1
1	0	0	0
1	0	1	0
1	1	0	1
1	1	1	0

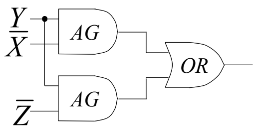
Znajdziemy oba rozwiązania w postaciach kanonicznych:

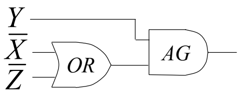
Input			Output	Iloczyny	Sumy
X	Y	Z	A		
0	0	0	0	$\overline{X}\,\overline{Y}\,\overline{Z}$	$X+Y+Z$ *
0	0	1	0	$\overline{X}\,\overline{Y}\,Z$	$X+Y+\overline{Z}$ *
0	1	0	1	$\overline{X}\,Y\,\overline{Z}$ *	$X+\overline{Y}+Z$
0	1	1	1	$\overline{X}\,Y\,Z$ *	$X+\overline{Y}+\overline{Z}$
1	0	0	0	$X\,\overline{Y}\,\overline{Z}$	$\overline{X}+Y+Z$ *
1	0	1	0	$X\,\overline{Y}\,Z$	$\overline{X}+Y+\overline{Z}$ *
1	1	0	1	$X\,Y\,\overline{Z}$ *	$\overline{X}+\overline{Y}+Z$
1	1	1	0	$X\,Y\,Z$	$\overline{X}+\overline{Y}+\overline{Z}$ *

Rozwiązanie w postaci sumy iloczynów:
$A=(\overline{X}\,Y\,\overline{Z}+\overline{X}\,Y\,Z)+(\overline{X}\,Y\,\overline{Z}+X\,Y\,\overline{Z})=\overline{X}\,Y\,(\overline{Z}+Z)+Y\,\overline{Z}\,(\overline{X}+X)=\overline{X}\,Y+Y\,\overline{Z}$.

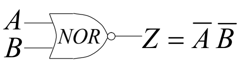
Rozwiązanie w postaci iloczynu sum:
$A=\big((X+Y+Z)(X+Y+\overline{Z})\big)\big((\overline{X}+Y+Z)(\overline{X}+Y+\overline{Z})\big)\big((\overline{X}+Y+\overline{Z})(\overline{X}+\overline{Y}+\overline{Z})\big)=$
$\big((X+Y)+(Z\,\overline{Z})\big)\big((\overline{X}+Y)+Z\,\overline{Z}\big)\big((\overline{X}+\overline{Z})+Y\,\overline{Y}\big)=\big((X+Y)(\overline{X}+Y)\big)(\overline{X}+\overline{Z})=\big(Y+X\,\overline{X}\big)(\overline{X}+\overline{Z})=$
$Y(\overline{X}+\overline{Z})$.

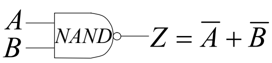
Otrzymaliśmy dwa różne rozwiązania, z których drugie jest prostsze. Widać to z ich realizacji przy pomocy bramek:





Oprócz wcześniej zdefiniowanych bramek duże znaczenie praktyczne mają dwie dodatkowe: bramka *NOR (Not Or Gate)* i bramka *Nand (Not And Gate)*. Oto ich oznaczenia i definicje w wersji dla dwóch zmiennych wejściowych (może być ich więcej):



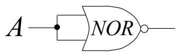


A	B	$Z=\overline{(A+B)}=\overline{A}\cdot\overline{B}$
0	0	1
0	1	0
1	0	0
1	1	0

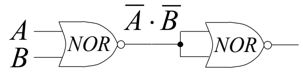
A	B	$Z=\overline{(A\cdot B)}=\overline{A}+\overline{B}$
0	0	1
0	1	1
1	0	1
1	1	0

Istotna zaleta powyższych dwóch bramek polega na tym, że każde wyrażenie algebry Boole'a daje się zrealizować tylko przy pomocy bramek NOR albo tylko przy pomocy bramek NAND. Uzasadniają to poniższe schematy:

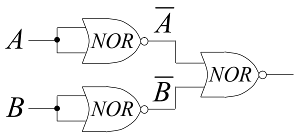
Tylko bramki NOR



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

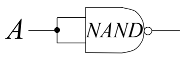


$\overline{\left(\overline{A}\cdot\overline{B}\right)}=A+B$

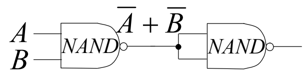


$\overline{\left(\overline{A}\right)}\cdot\overline{\left(\overline{B}\right)}=A\cdot B$

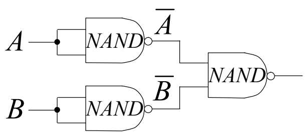
Tylko bramki NAND



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



$\overline{\left(\overline{A}+\overline{B}\right)}=A\cdot B$

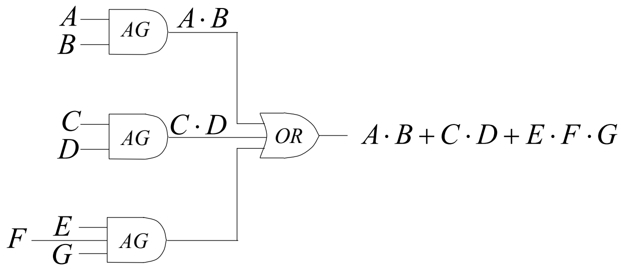


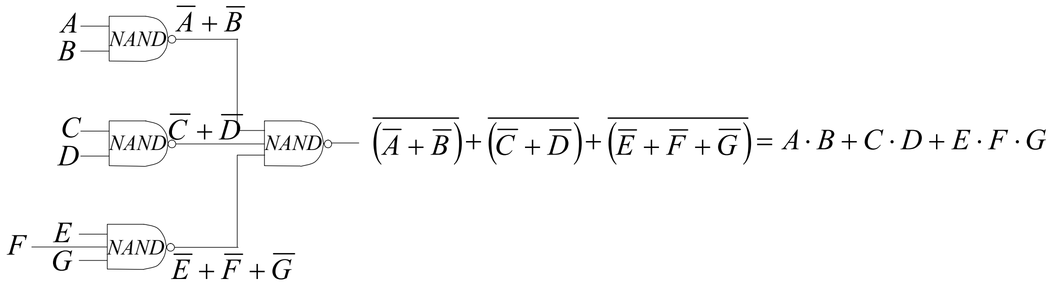
$\overline{\left(\overline{A}\right)}+\overline{\left(\overline{B}\right)}=A+B$

W praktyce nie ma potrzeby stosowania powyższych schematów. Jak już wcześniej zauważyliśmy, wszystkie wyrażenia algebry Boole'a dają się sprowadzić do jednej z dwóch postaci kanonicznych: sumy iloczynów bądź iloczynu sum. W pierwszym z tych przypadków zamiast stosować układ bramek typu *And-to-Or* można użyć układu *Nand-to-Nand* zastępując w pierwszym układzie wszystkie bramki bramkami Nand. W przypadku układu typu *OR-to-And* ten sam efekt uzyskuje się poprzez zastąpienie wszystkich bramek bramkami typu Nor. Daje to układ typu *Nor-to-Nor*. W obu przypadkach otrzymuje się układy logicznie równoważne, ale fizycznie różne. Zagadnienie ilustrują poniższe przykłady.

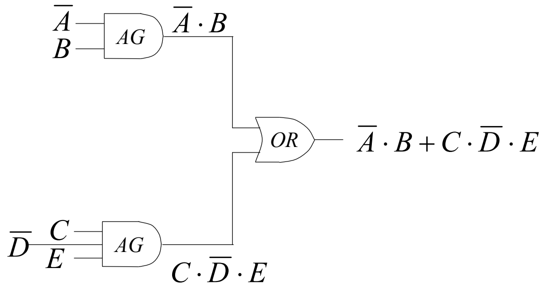
Przykłady

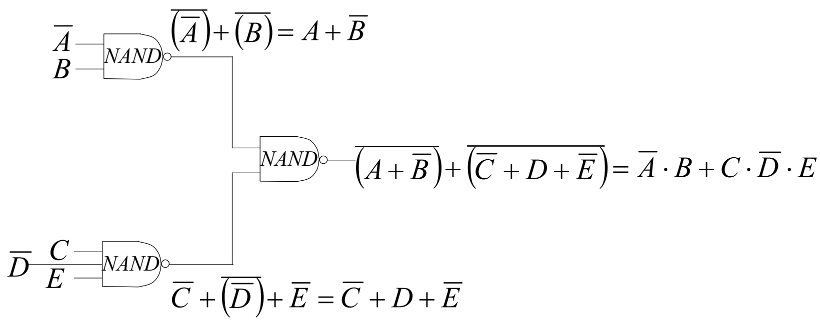
a)



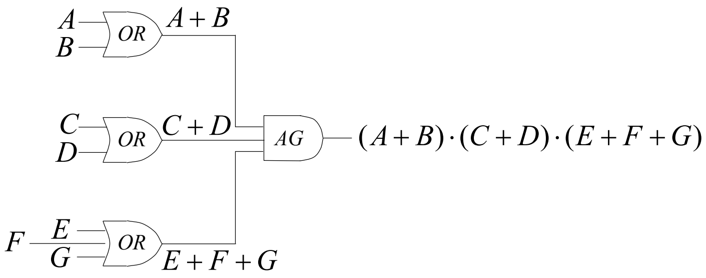


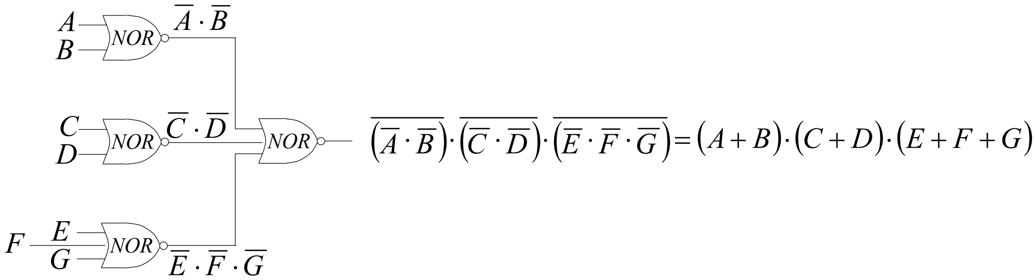
b)



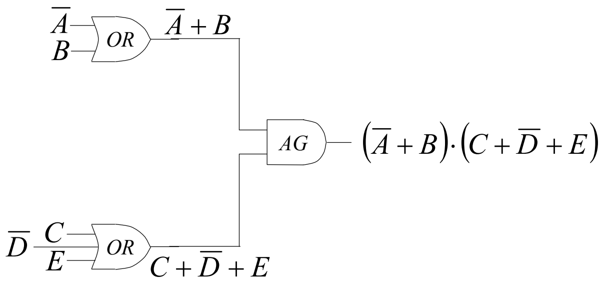


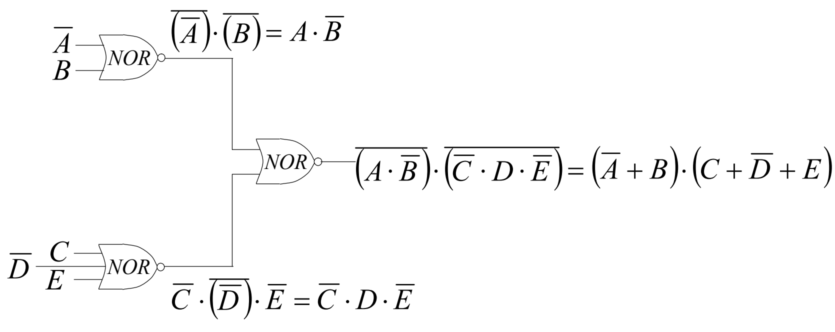
c)


A
B
OR
$A+B$
C
D
OR
$C+D$
AG
$(A+B)\cdot(C+D)\cdot(E+F+G)$
F
E
G
OR
$E+F+G$


A
B
NOR
$\overline{A}\cdot\overline{B}$
C
D
NOR
$\overline{C}\cdot\overline{D}$
NOR
$\overline{\left(\overline{A}\cdot\overline{B}\right)}\cdot\overline{\left(\overline{C}\cdot\overline{D}\right)}\cdot\overline{\left(\overline{E}\cdot\overline{F}\cdot\overline{G}\right)}=\left(A+B\right)\cdot\left(C+D\right)\cdot\left(E+F+G\right)$
F
E
G
NOR
$\overline{E}\cdot\overline{F}\cdot\overline{G}$

d)


$\overline{A}$
B
OR
$\overline{A}+B$
AG
$\left(\overline{A}+B\right)\cdot\left(C+\overline{D}+E\right)$
$\overline{D}$
C
E
OR
$C+\overline{D}+E$


$\overline{A}$
B
NOR
$\left(\overline{\overline{A}}\right)\cdot\left(\overline{B}\right)=A\cdot\overline{B}$
NOR
$\overline{\left(A\cdot\overline{B}\right)}\cdot\overline{\left(\overline{C}\cdot D\cdot\overline{E}\right)}=\left(\overline{A}+B\right)\cdot\left(C+\overline{D}+E\right)$
$\overline{D}$
C
E
NOR
$\overline{C}\cdot\left(\overline{\overline{D}}\right)\cdot\overline{E}=\overline{C}\cdot D\cdot\overline{E}$

Układy zintegrowane

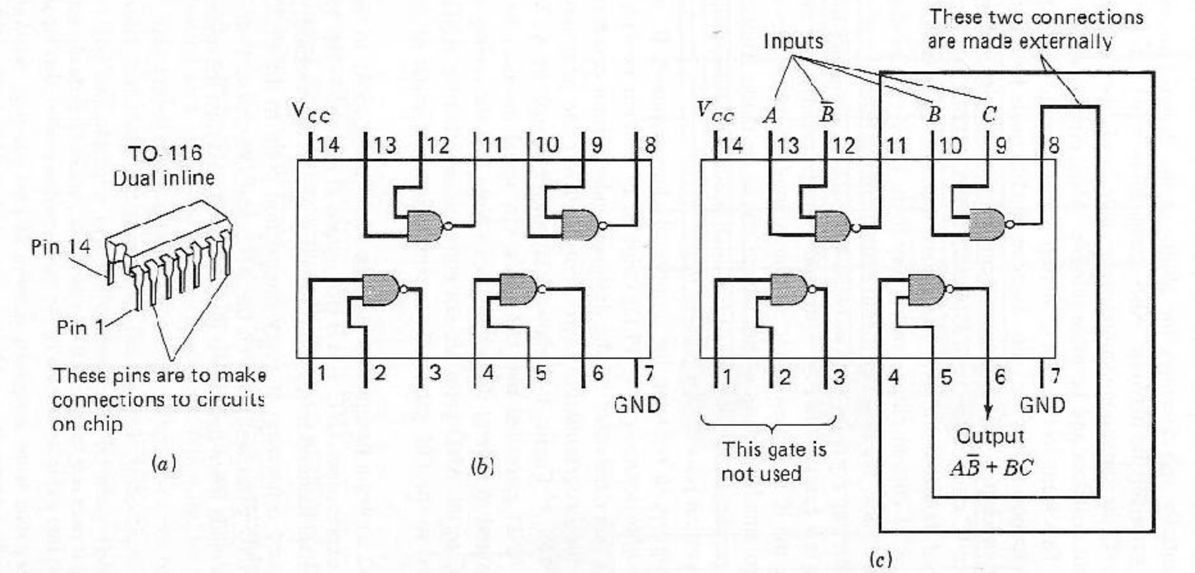
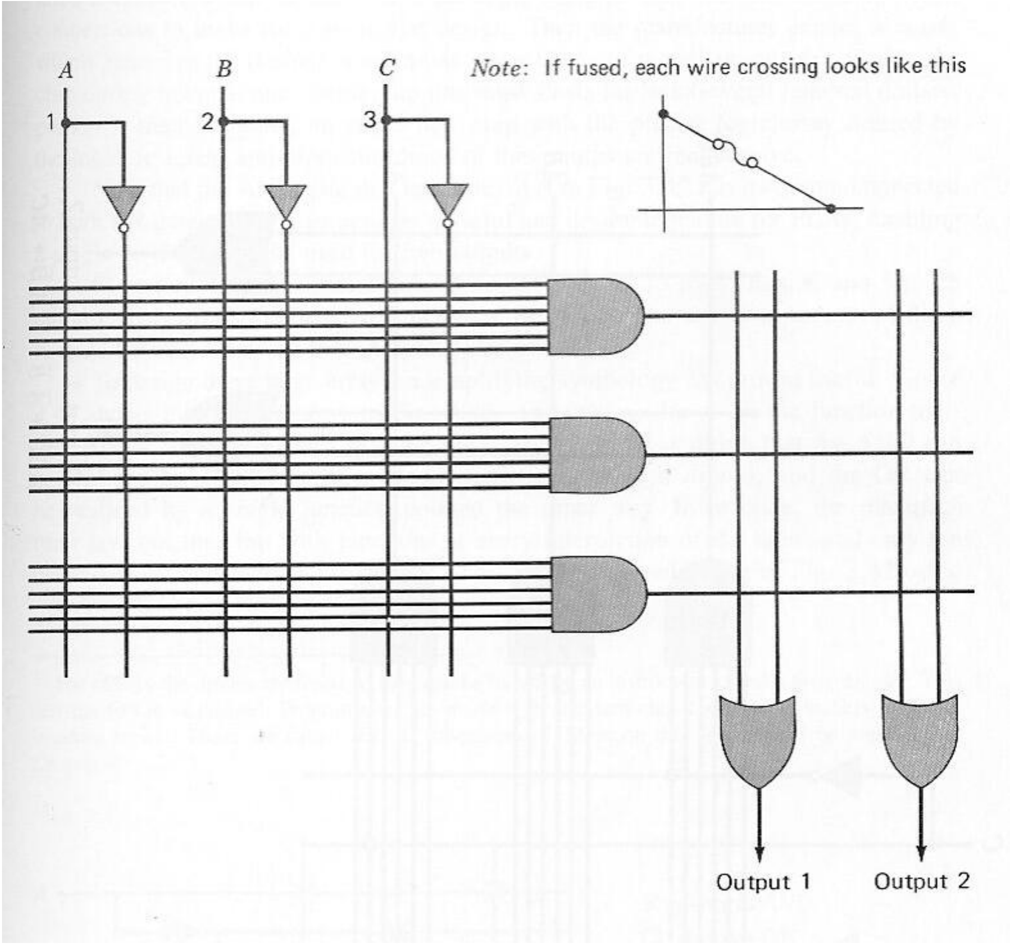


FIGURE 3.40
Integrated-circuit container and pin-out. (*a*) Integrated-circuit container. (*b*) Pin-out showing gate layout in container in (*a*). (*c*) NAND-to-NAND gate set realizing $A\bar{B} + BC$.


A
B
C
1
2
3
Note: If fused, each wire crossing looks like this
Output 1
Output 2

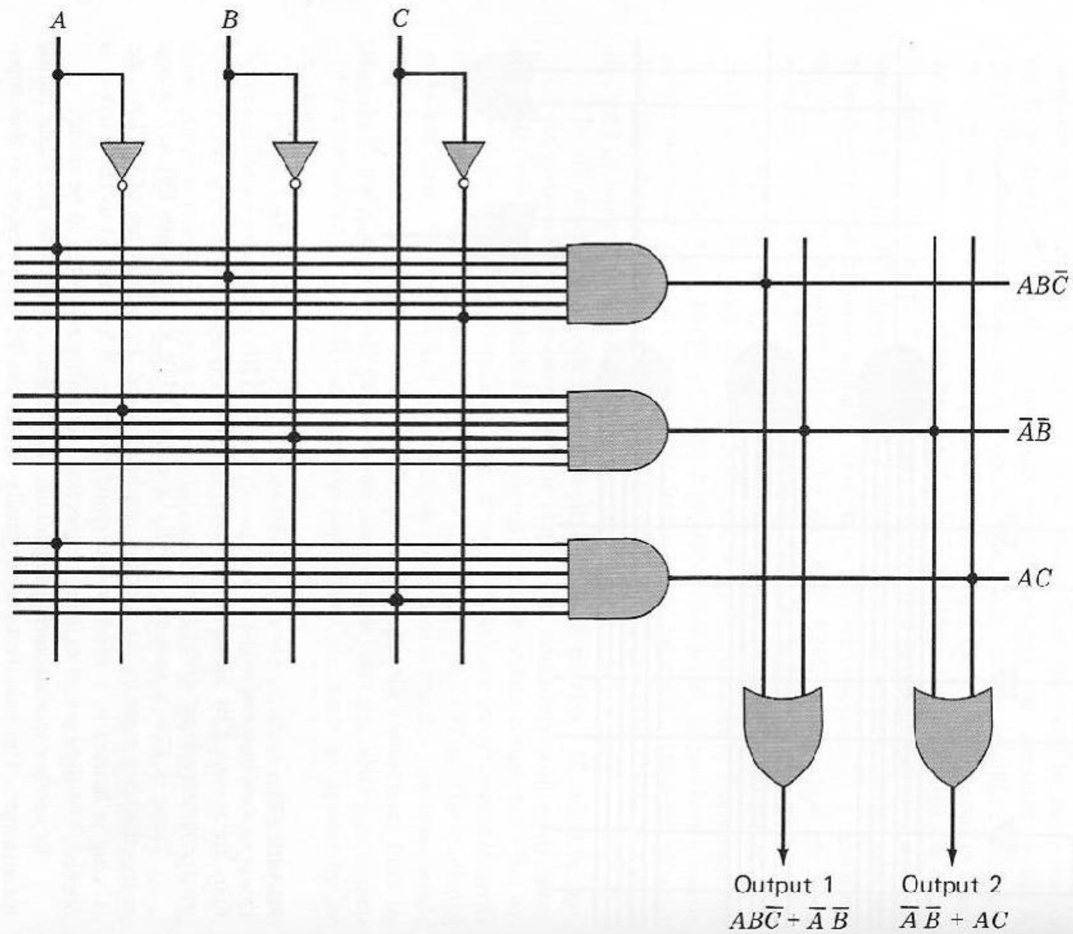


FIGURE 3.42
Connection design for three-input two-output PLA.

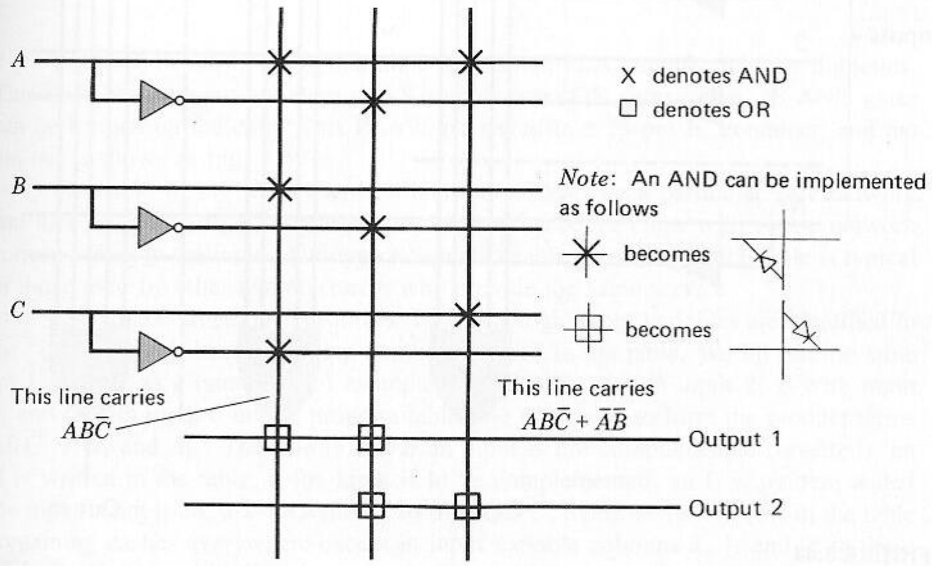


FIGURE 3.43
A frequently used way to draw PLA designs.