

Prawa Boole'a

Zmienne określamy literami alfabetu np. a, b, c, d ...

Operatory mogą przybierać różne oznaczenia

Suma	Iloczyn	Negacja
OR	AND	NOT
U	\cap	\sim
+	.	\bar{a}
+	.	—
&		¬

Dziewięć własności które będą potrzebne do przeliczania

$$a + 0 = a$$

$$1 + a = 1$$

$$0 * a = 0$$

$$1 * a = a$$

$$a + a = a$$

$$a + \bar{a} = 1$$

$$a * a = a$$

$$a * \bar{a} = 0$$

$$\bar{a} * \bar{a} = a$$

Przemienność

$$a + b = b + a$$

$$a * b = b * a$$

Łączność

$$a+(b+c) = (a+b)+c$$

$$a*(b*c) = (a*b)*c$$

Absorpcja

$$a+(a*b) = a$$

$$a*(a+b) = a$$

Rozdzielność

$$a+(b*c)=(a+b)*(a+c)$$

$$a*(b+c)=(a*b)+(a*c)$$

Bramki logiczne

Bramka Not

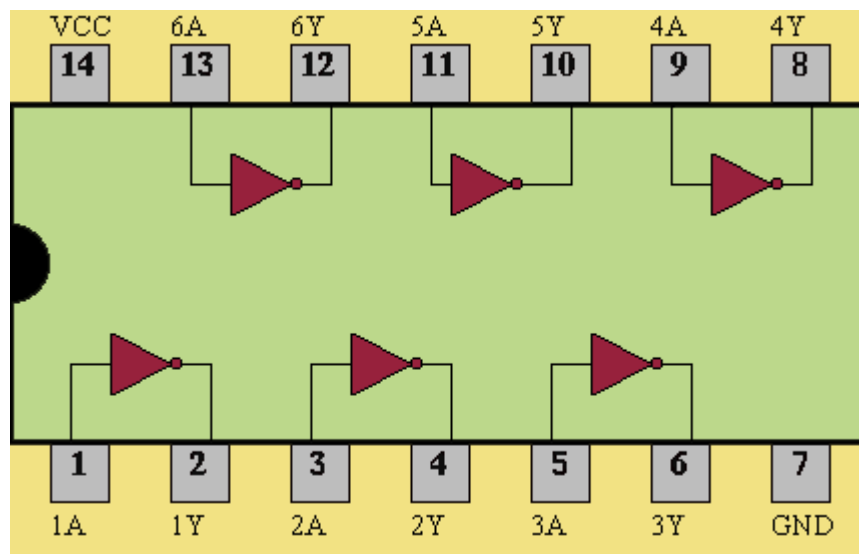
Bramka NOT jest najprostrzą w działaniu bramką logiczną. Jej zadaniem jest odwracanie (negowanie) sygnału wejściowego. Gdy na wejściu ustawimy sygnał "1" to na wyjściu otrzymamy "0", a gdy na wejściu ustawimy "0" to na wyjściu pojawi się "1". Wejście bramki, wg przedstawionego poniżej symbolu graficznego, znajduje się po lewej stronie; po prawej jest jej wyjście. Bramka ta posiada tylko jedno wejście.



Tabela prawdy bramki NOT

we	wy
0	1
1	0

Układ logiczny z bramkami Not



Bramka AND

Wyjście bramki logicznej AND jest w stanie wysokim tylko wtedy, gdy oba wejścia są w stanie wysokim. Symbol logiczny tej bramki i tablicę prawdy pokazano poniżej. Podobnie jak w przypadku bramek OR, dostępne są bramki AND 3- i 4-wejściowe (czasem o większej liczbie wejść). Na przykład, jeżeli 8-wejściowa bramka AND będzie miała wyjście w stanie wysokim tylko wtedy, gdy wszystkie wejścia będą w stanie wysokim.

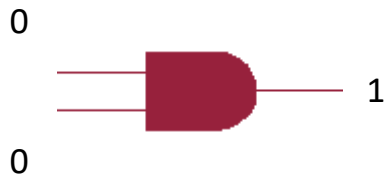
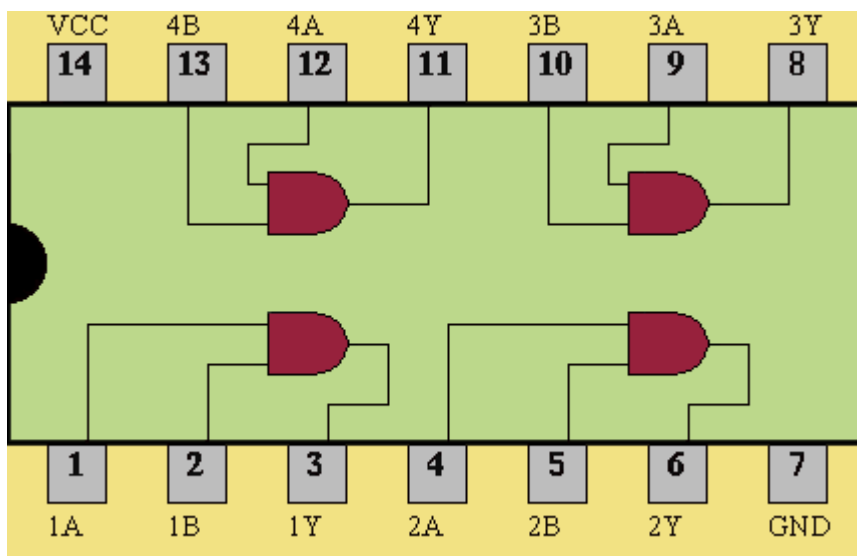


Tabela prawdy bramki NAND

we 1	we 2	wy
0	0	1
0	1	1
1	0	1
1	1	0

Układ logiczny z bramkami AND



Bramk NAND

Bramka NAND stanowi jakby połączenie bramki AND i NOT. Zero logiczne "0" na wyjściu jest ustawiane tylko wtedy gdy na obu wejściach jest jedynka logiczna "1". W pozostałych przypadkach na wyjściu zawsze jest stan "1". Można zauważyć, że jest ona dokładną odwrotnością bramki AND - porównaj tablice prawdy dla obu bramek. Również i ta bramka może mieć wiele wejść i tylko jedno wyjście.

0

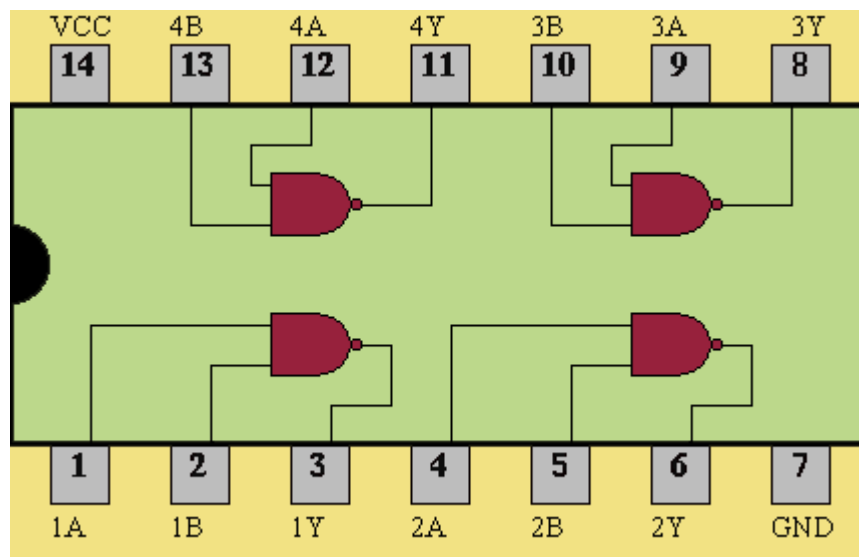


0

Tabela prawdy bramki NAND

we 1	we 2	wy
0	0	1
0	1	1
1	0	1
1	1	0

Układ logiczny z bramkami NAND



Bramka OR

Wyjście brami OR (czyli LUB) jest w stanie wysokim, jeżeli jedno z wejść (lub oba) jest w stanie wysokim. Można to wyrazić za pomocą "tablicy prawdy". Narysowana bramka to 2-wejściowa bramka OR. W przypadku ogólnym bramki mogą mieć dowolną liczbę wejść, ale typowy układ scalony zawiera zwykle cztery bramki 2-wejściowe, trzy bramki 3-wejściowe lub dwie bramki 4-wejściowe. Na przykład, wejście 4-wejściowej bramki OR będzie w stanie wysokim, jeżeli przynajmniej jedno jej wejście będzie w stanie wysokim.

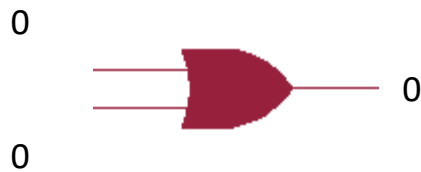
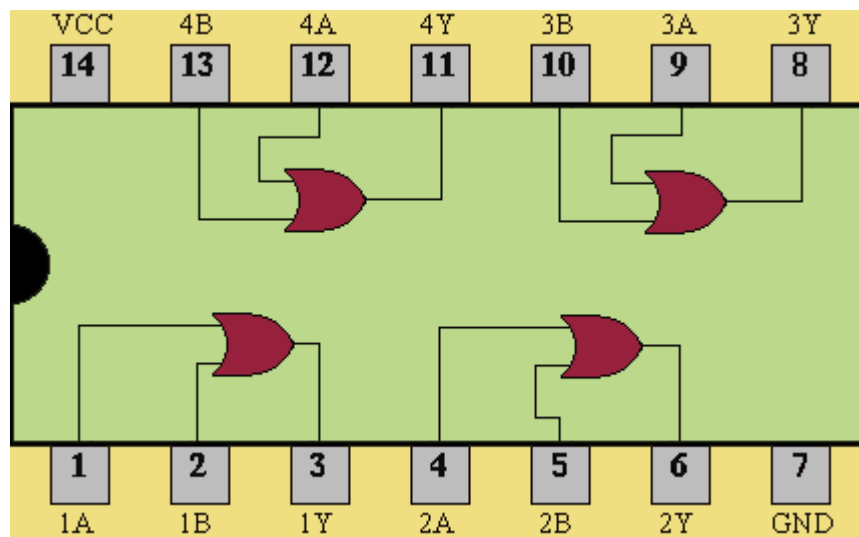


Tabela prawdy bramki OR

we 1	we 2	wy
0	0	0
0	1	1
1	0	1
1	1	1

Układ logiczny z bramkami OR



Bramka NOR

Bramka NOR jest odwrotnością bramki OR. Zero na wyjściu pojawia się zawsze wtedy, gdy choćby na jednym z wejść jest jedynka logiczna. Tylko wtedy, gdy wszystkie wejścia są ustawione w stan "0" na wyjściu pojawia się "1".

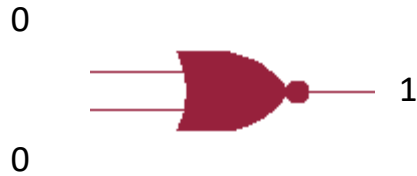
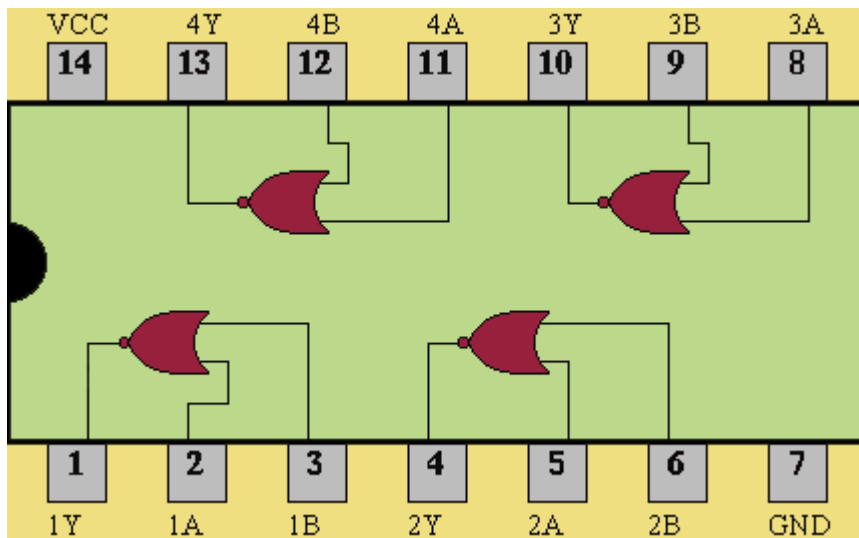


Tabela prawdy bramki NOR

we 1	we 2	wy
0	0	1
0	1	0
1	0	0
1	1	0

Układ logiczny z bramkami NOR



Bramka XOR

Exclusive-OR (XOR, czyli WYŁĄCZNIK NIE) jest ciekawą funkcją, chociaż mniej podstawową niż AND i OR. Wyjście bramki XOR jest w stanie wysokim, jeżeli jedno albo drugie wejście jest w stanie wysokim (jest to zawsze funkcja dwóch zmiennych). Bramka XOR realizuje dodawanie bitów modulo-2.

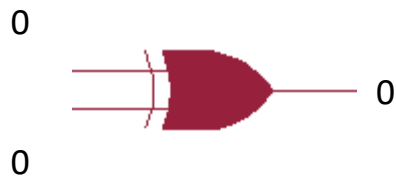
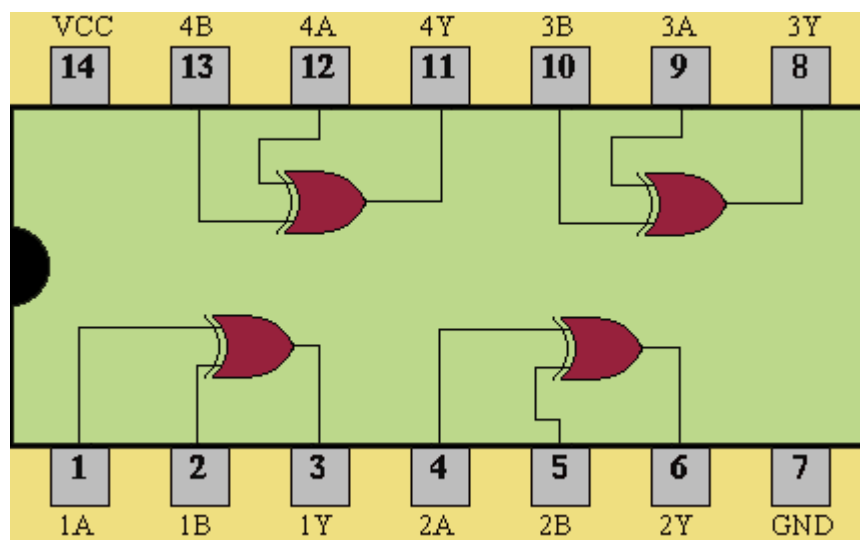


Tabela prawdy bramki XOR

we 1	we 2	wy
0	0	0
0	1	1
1	0	1
1	1	0

Układ logiczny z bramkami XOR



[illegible]

