

Sławomir Kulesza

# Technika cyfrowa

## Inżynieria dyskretna cz. 1

Wykład dla studentów III roku Informatyki

Wersja 5.0, 10/10/2015

# Rachunek zdań

Zdanie logiczne – każde zdanie, któremu można przyporządkować wartość logiczną (T/F).

Funktory logiczne – spójniki łączące zdania logiczne.

10100101001101010101011011000101101011110101010010100010101010010010101010010101001010100110010101001

		Koniunkcja (iloczyn, AND)	Alternatywa (suma, OR)	Implikacja	Równoważność	Negacja (Inwersja, NOT)
A	B	$A \cdot B$	$A + B$	$A \Rightarrow B$	$A \Leftrightarrow B$	$\bar{A}$
0	0	0	0	1	1	1
0	1	0	1	1	0	1
1	0	0	1	0	0	0
1	1	1	1	1	1	0

# Algebra Boole'a

Zerojedynkowa algebra sygnałów binarnych:

$\langle \{0, 1\}, +, \cdot, \neg, 0, 1 \rangle$ :

- zdefiniowane działania zeroargumentowe: 0, 1
- zdefiniowane działanie jednoargumentowe: NOT
- zdefiniowane działania dwuargumentowe: OR, AND
- element neutralny sumy: 0
- element neutralny iloczynu: 1

# Prawa algebry Boole'a

Idempotentność	$A + A = A$	$A \cdot A = A$
Przemienność	$A + B = B + A$	$A \cdot B = B \cdot A$
Łączność	$A + (B + C) = (A + B) + C$	$(A \cdot B) \cdot C = A \cdot (B \cdot C)$
Rozdzielność	$A + (B \cdot C) = (A + B)(A + C)$	$A \cdot (B + C) = (A \cdot B) + (A \cdot C)$
Pochłanianie	$A + (A \cdot B) = A$	$A \cdot (A + B) = A$
Element neutralny	$A + 0 = A$	$A \cdot 1 = A$
Element stały	$A + 1 = 1$	$A \cdot 0 = 0$
Negacja	$A + \bar{A} = 1$	$A \cdot \bar{A} = 0$
Podwójna negacja	$\bar{\bar{A}} = A$	
Prawa de Morgana	$\overline{A + B} = \bar{A} \cdot \bar{B}$	$\overline{A \cdot B} = \bar{A} + \bar{B}$

# Zasada dualności

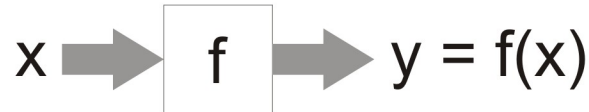
W obrębie danego prawa wzory są dualnie równoważne: każdy związek można otrzymać z drugiego poprzez zamianę operatorów koniunkcji i alternatywy oraz stałych 0 i 1.



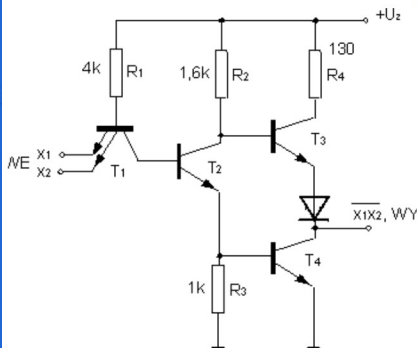
# Poziomy abstrakcji opisu UC

$$f = \bar{X}_1\bar{X}_2X_3 + \bar{X}_1X_2X_3 + X_1\bar{X}_2X_3 + X_1X_2\bar{X}_3 + X_1X_2X_3$$

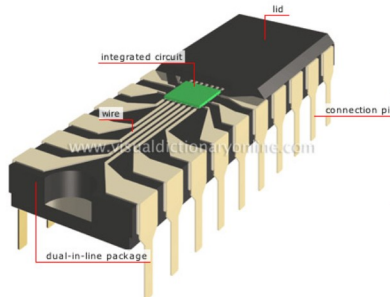
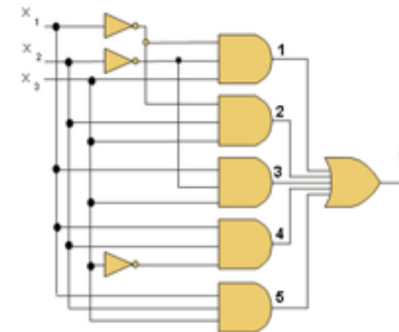
Opis funkcjonalny



	$x_1$	$x_2$	$x_3$	$f$
0	0	0	0	0
1	0	0	1	1
2	0	1	0	0
3	0	1	1	1
4	1	0	0	0
5	1	0	1	1
6	1	1	0	1
7	1	1	1	1



Opis strukturalny



Opis fizyczny

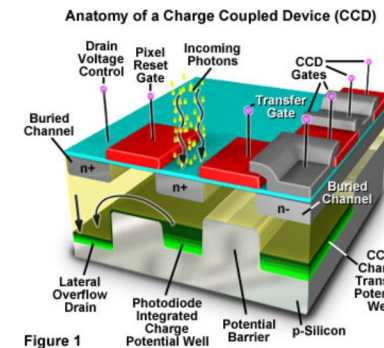


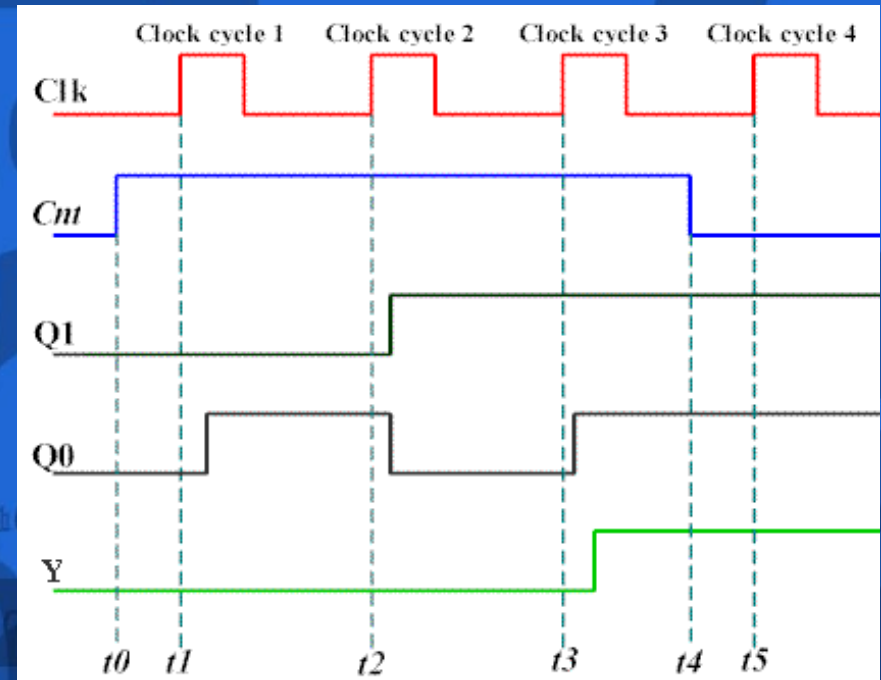
Figure 1

# Abstrakcja funkcjonalna

Abstrakcja funkcjonalna dotyczy zachowania układu jako „czarnej skrzynki”: interesuje nas zależność wej→wyj, a nie konkretna realizacja układu lub jego prostota.

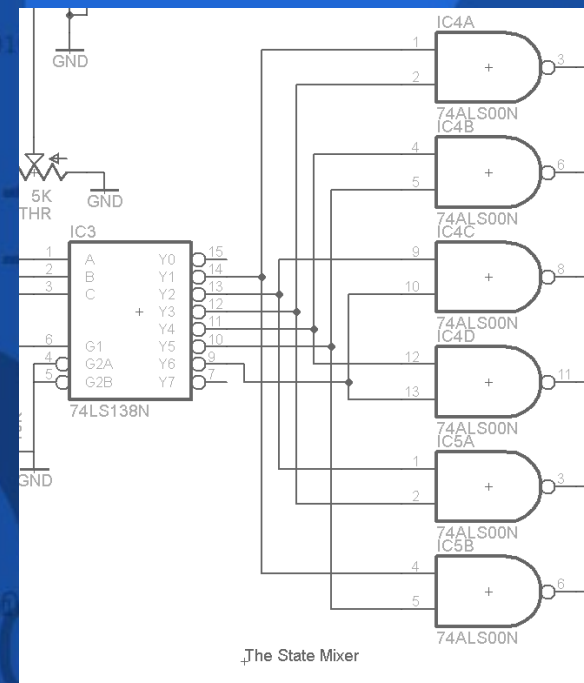
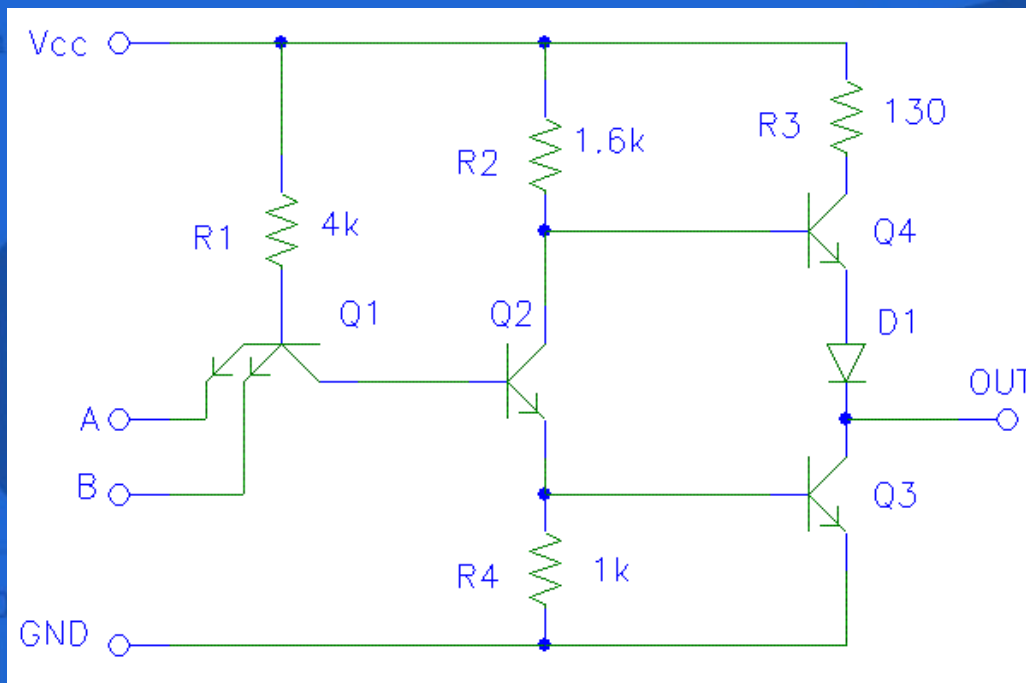
$$f = A \cdot \bar{B} \cdot D + \bar{A} \cdot B + \bar{C}$$

Input		Output			
		After Multiplication		After byte-wise XOR	
Hex	Bin $x_1x_2x_3x_4$	Hex	Bin $y_1y_2y_3y_4y_5y_6y_7y_8$	Hex	Bin $y_1y_2y_3y_4y_5y_6y_7y_8$
0x0	0000	0x00	0000 0000	0xff	1111 1111
0x1	0001	0x11	0001 0001	0xee	1110 1110
0x2	0010	0x22	0010 0010	0xdd	1101 1101
0x3	0011	0x33	0011 0011	0xcc	1100 1100
0x4	0100	0x44	0100 0100	0xbb	1011 1011
0x5	0101	0x55	0101 0101	0xaa	1010 1010
0x6	0110	0x66	0110 0110	0x99	1001 1001
0x7	0111	0x77	0111 0111	0x88	1000 1000
0x8	1000	0x88	1000 1000	0x77	0111 0111
0x9	1001	0x99	1001 1001	0x66	0110 0110
0xa	1010	0xaa	1010 1010	0x55	0101 0101
0xb	1011	0xbb	1011 1011	0x44	0100 0100
0xc	1100	0xcc	1100 1100	0x33	0011 0011
0xd	1101	0xdd	1101 1101	0x22	0010 0010
0xe	1110	0xee	1110 1110	0x11	0001 0001
0xf	1111	0xff	1111 1111	0x00	0000 0000



# Abstrakcja strukturalna

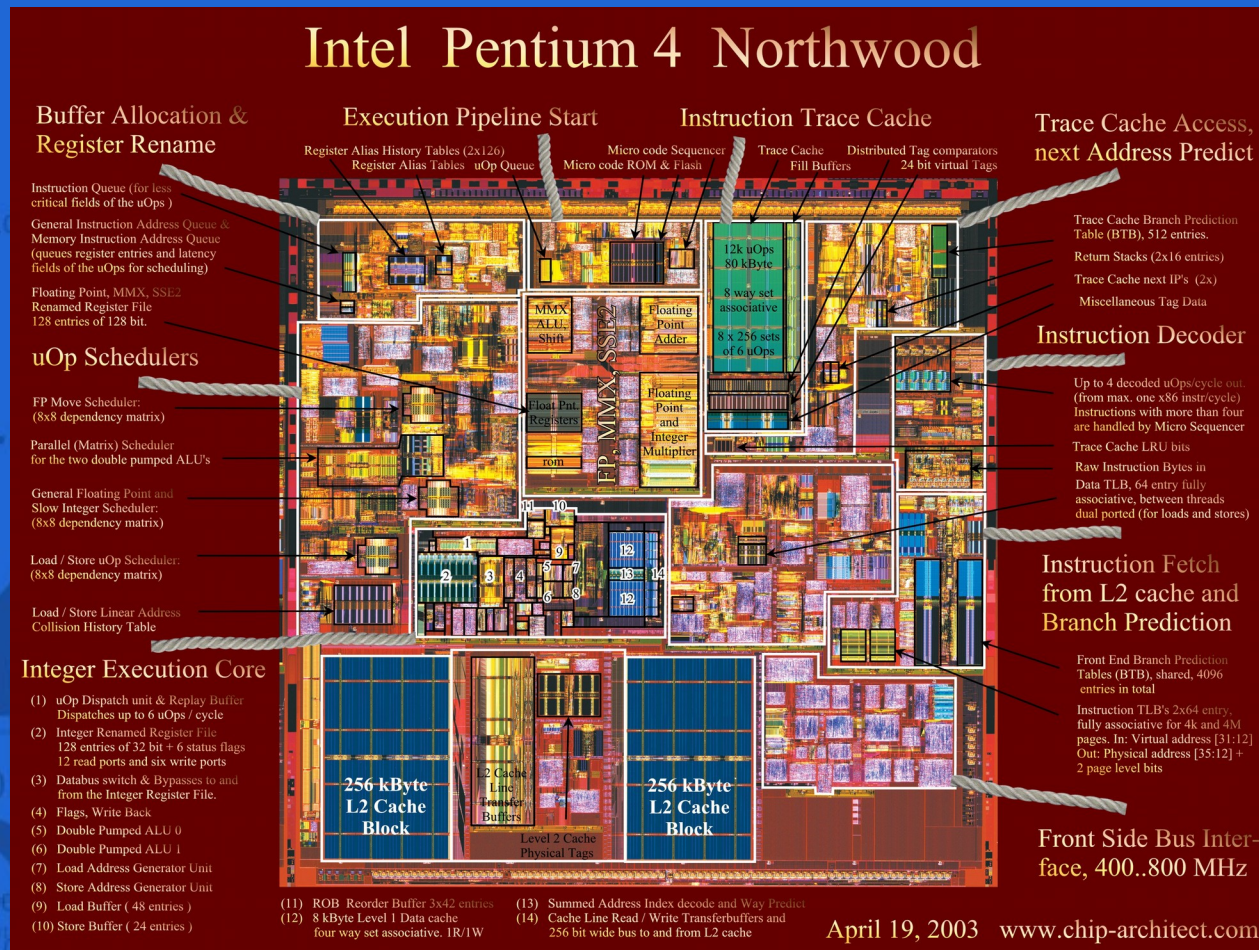
Abstrakcja strukturalna traktuje układ jak sieć połączonych elementów funkcjonalnych: od schematu elektronicznego układu do schematu elementów logicznych.





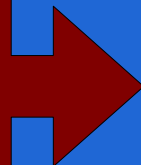
# Abstrakcja fizyczna

Abstrakcja fizyczna to najniższy poziom opisu układu: fizyczna budowa elementów, topografia ścieżek, warstwowość struktury itp.



# Realizacje rachunku zdań binarnych

Układy  
mechaniczne



Układy  
elektromechaniczne



Układy  
elektroniczne



Układy  
elektroniczne  
scalone

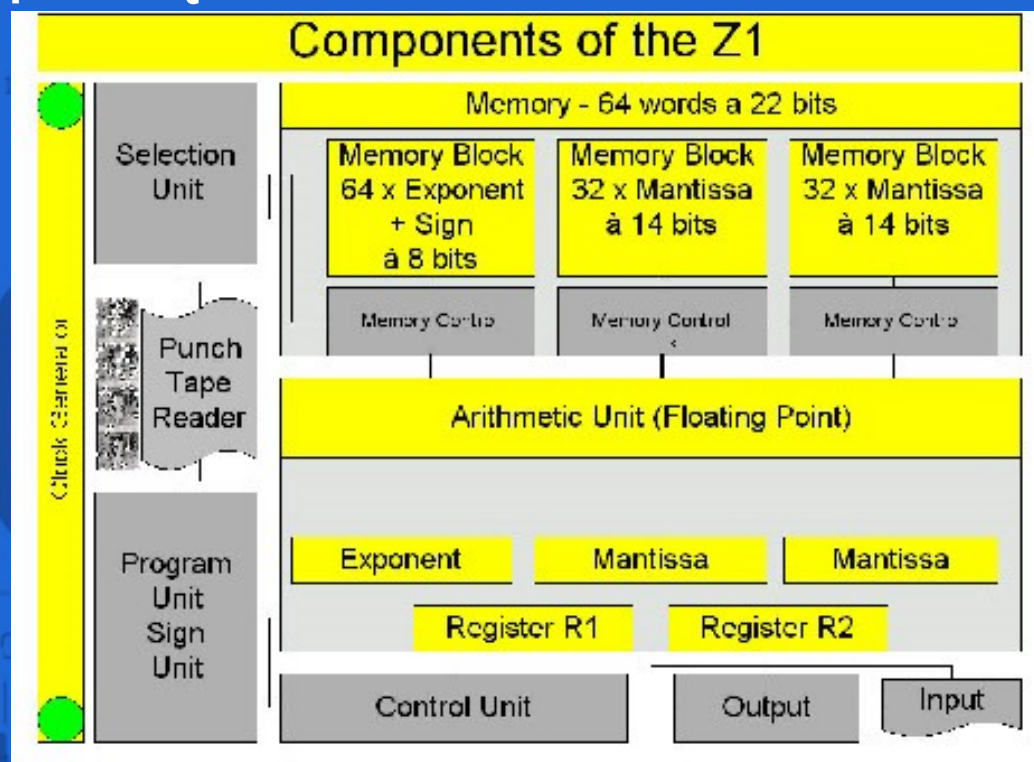


Układy  
kwantowe



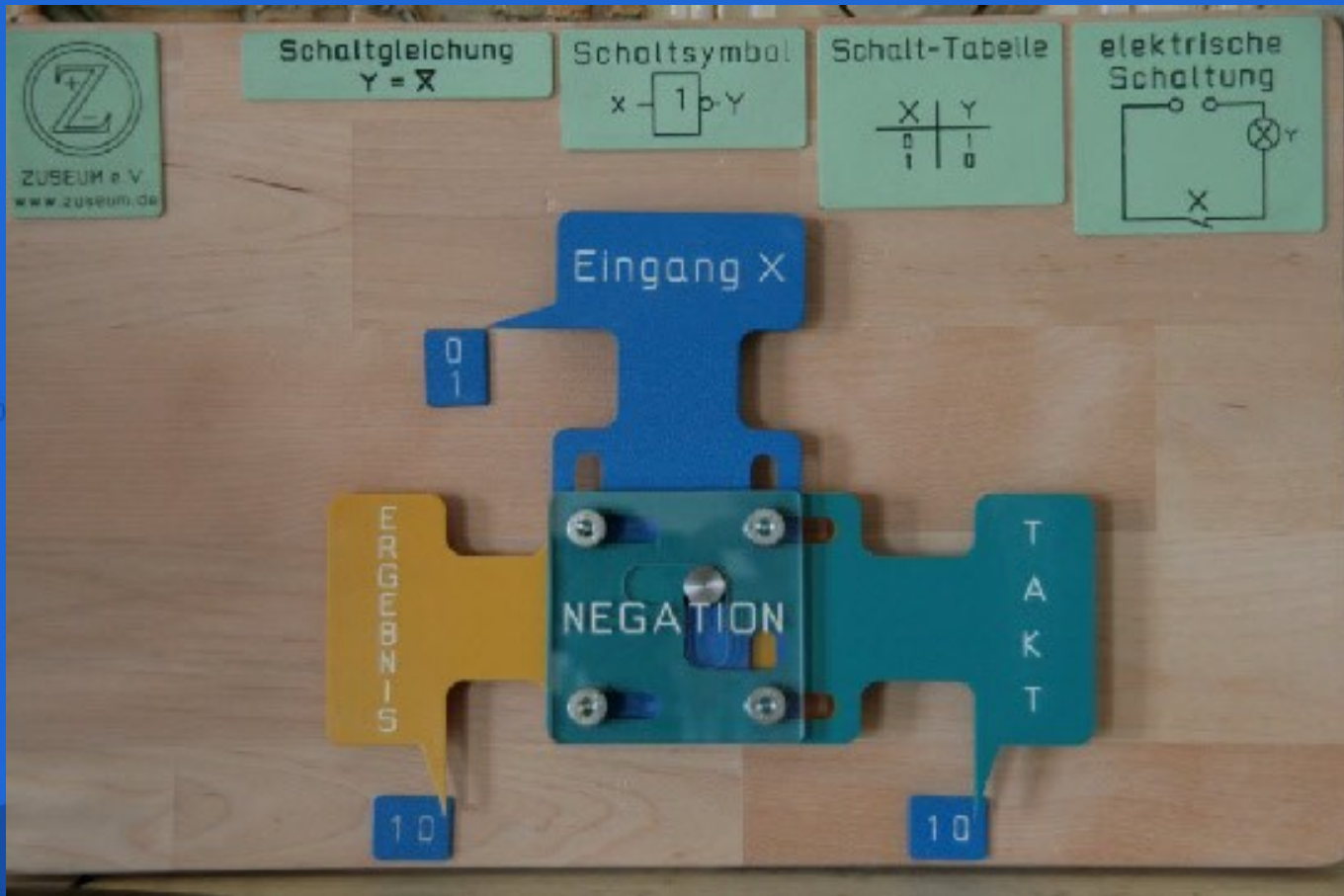
# Komputery mechaniczne

Pierwszym programowalnym komputerem binarnym z arytmetyką zmiennoprzecinkową był Z1 stworzony przez Konrada Zuse (1938 r.):  $f_{\text{takt}} = 1 \text{ Hz}$ , 22 bit, pamięć  $64 \times 22 \text{ bit}$ , dodawanie 5 s, mnożenie 10 s.

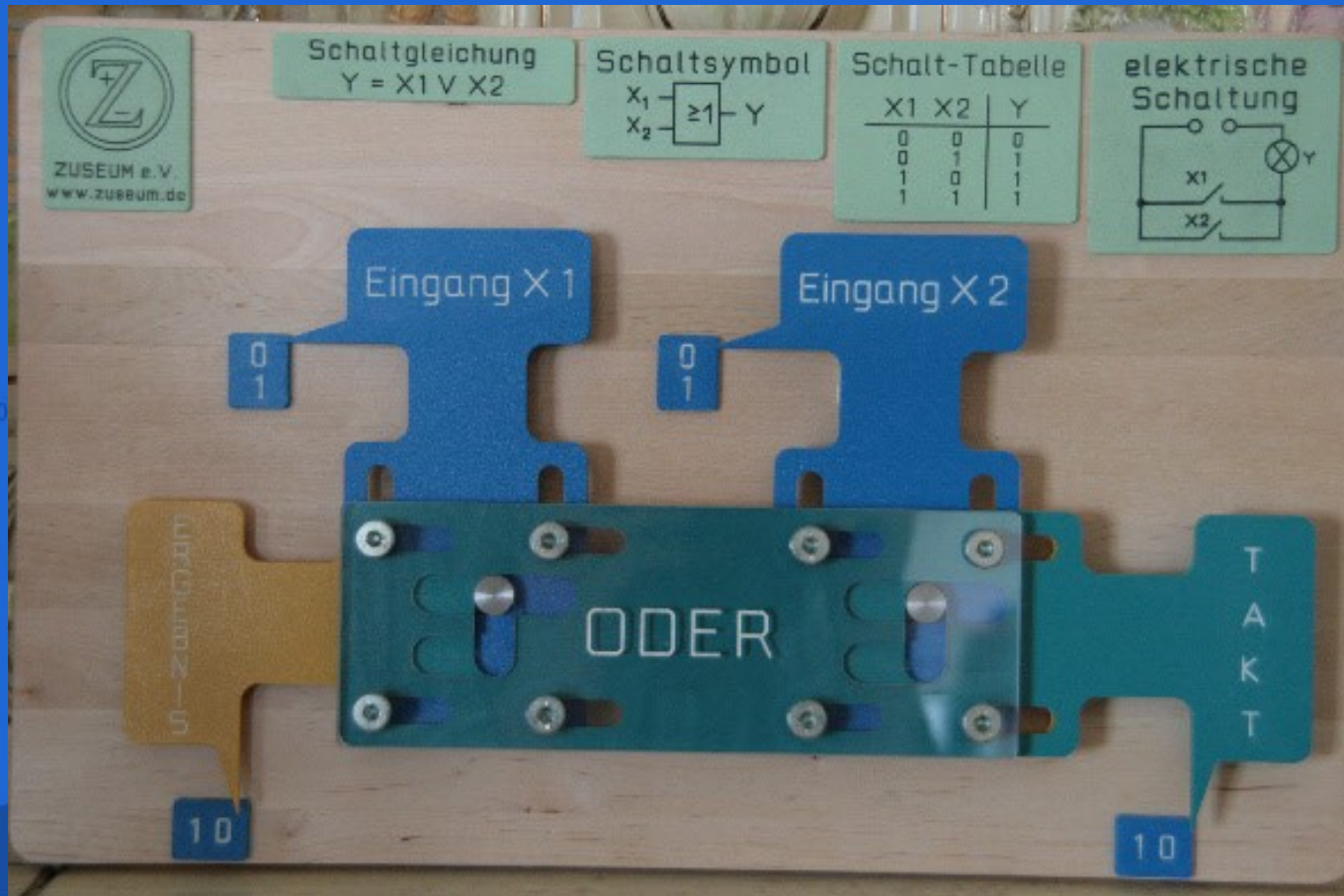




# Mechanische bramka NOT



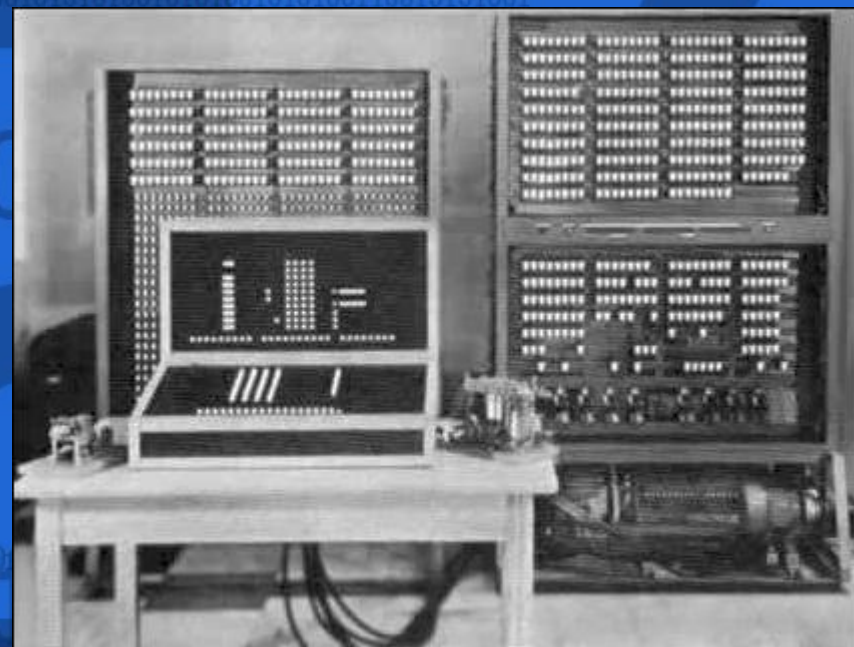
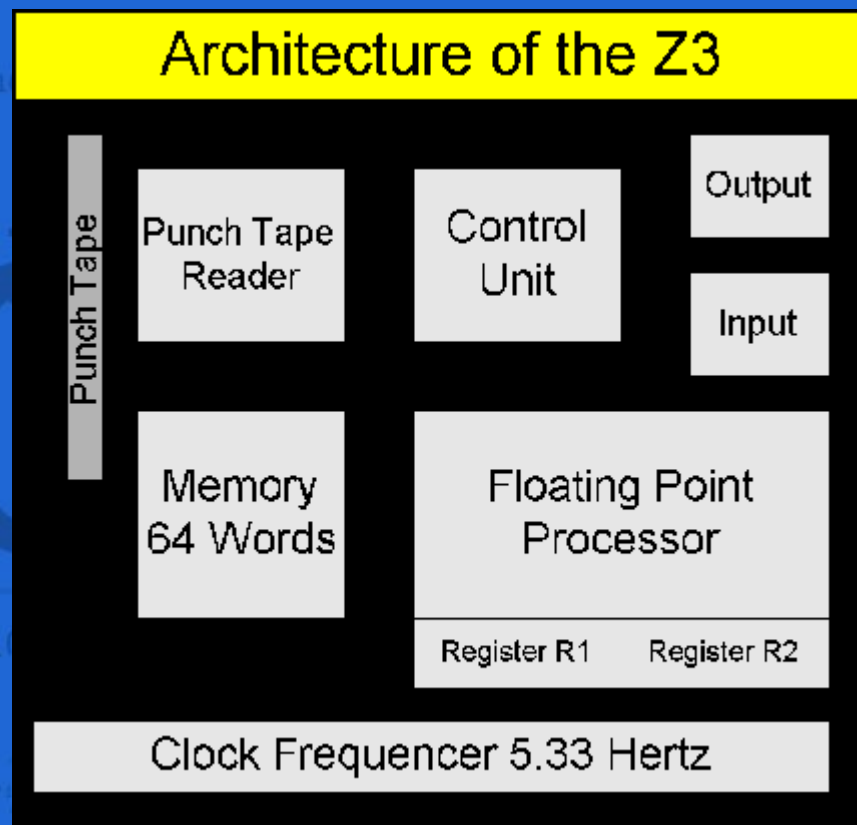
# Mechanische bramka OR





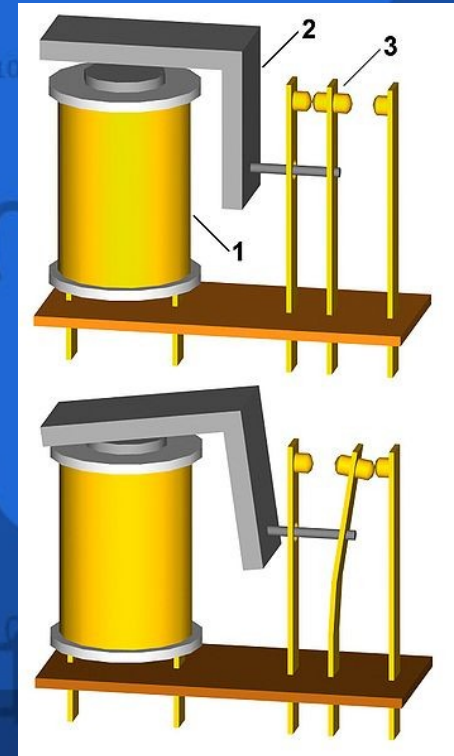
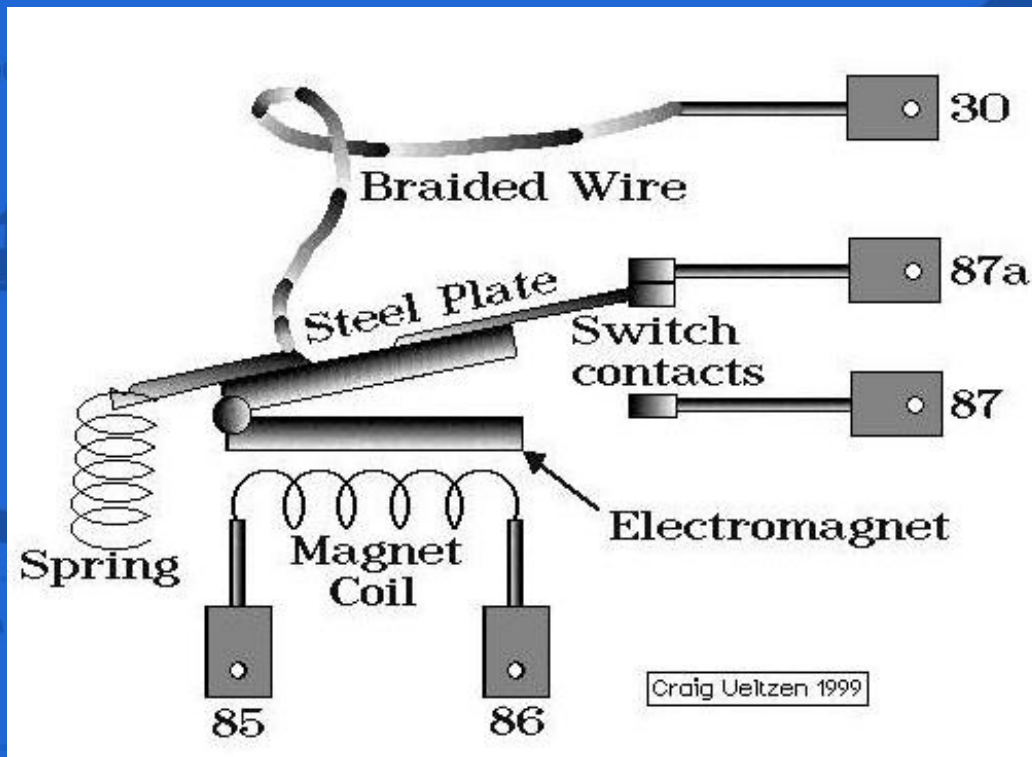
# Maszyny przekąźnikowe

Konrad Zuse w roku 1941 stworzył maszynę przekąźnikową Z3: arytmetyka zmiennoprzecinkowa,  $f_{\text{takt}} = 5.3 \text{ Hz}$ , 22 bit, pamięć  $64 \times 22 \text{ bit}$ , dodawanie 0.8 s, mnożenie 3 s.

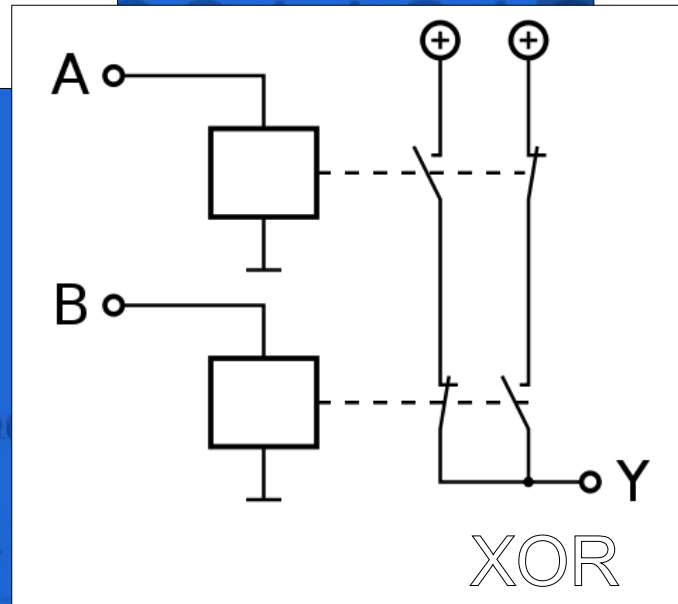
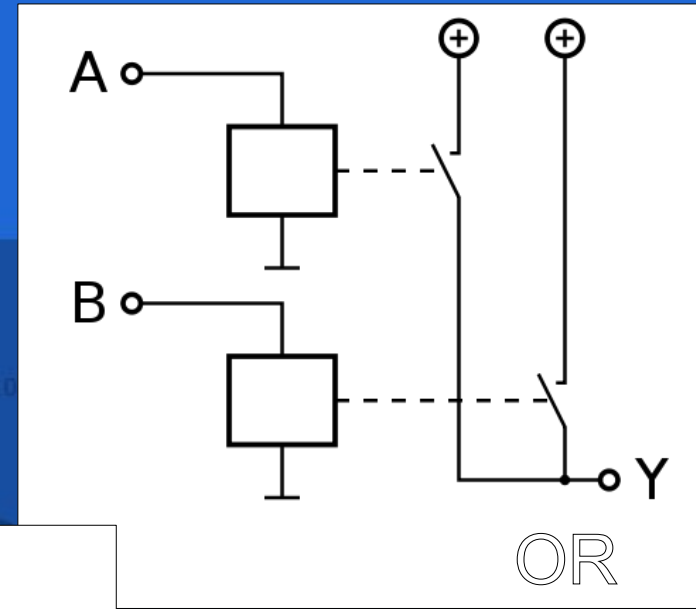
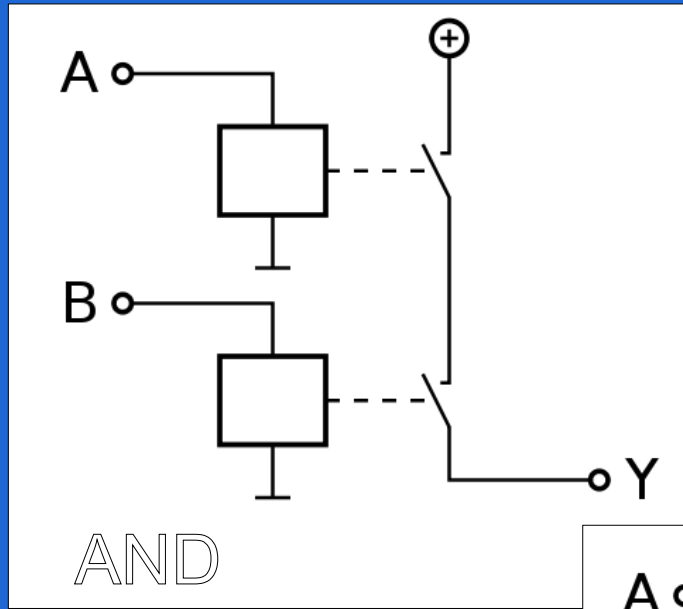


# Przełącznik elektromechaniczny

Przełącznik jest przełącznikiem sterowanym prądem elektrycznym.



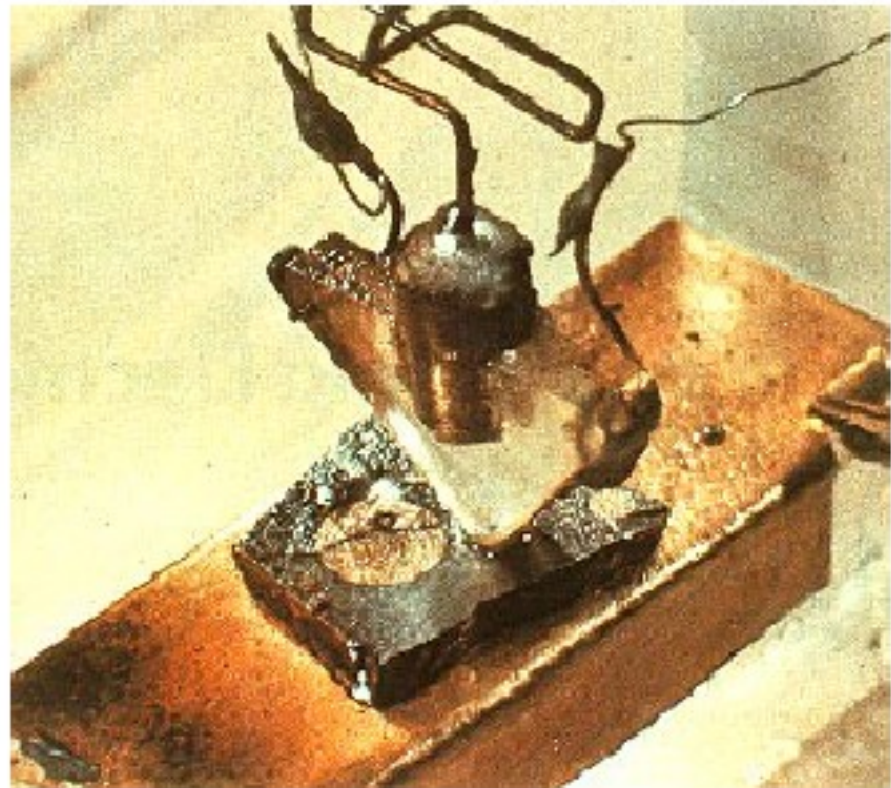
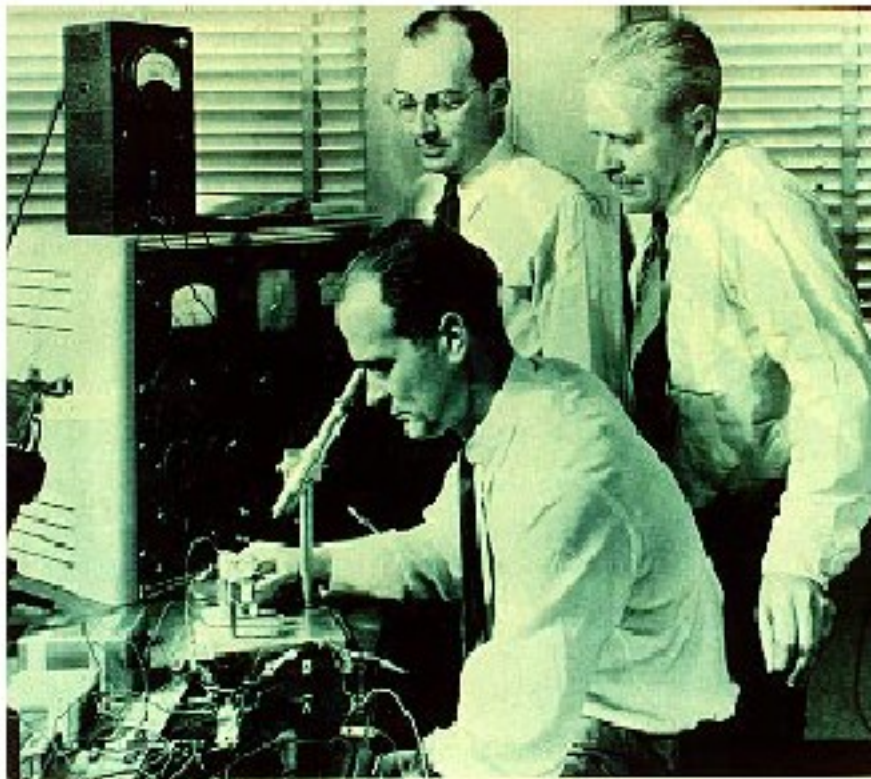
# Przełącznikowa realizacja wyrażeń logicznych



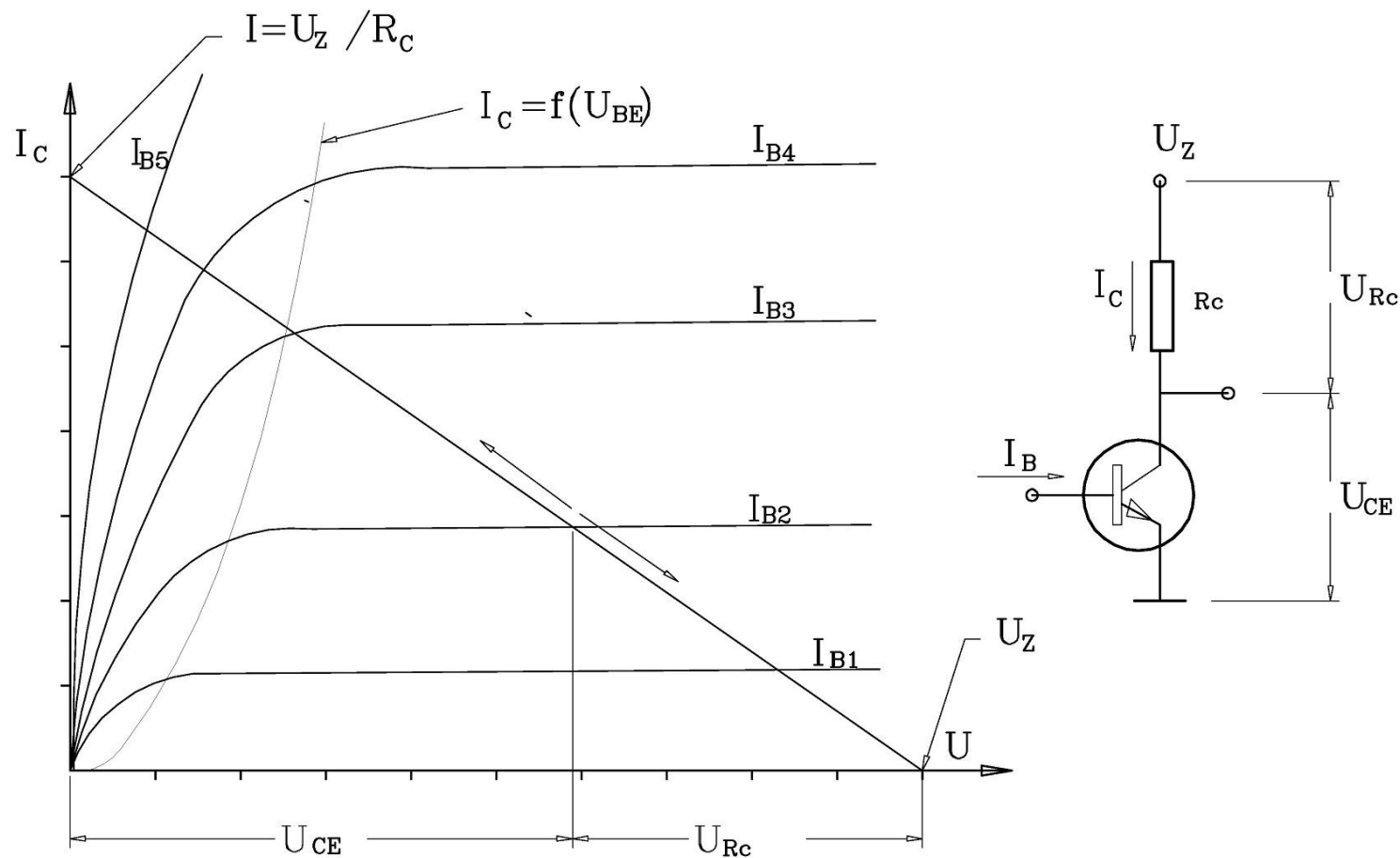


# Tranzystor ostrzowy 1947 r.

Shockley, Bardeen, Brattain – nagroda Nobla (1956 r.)

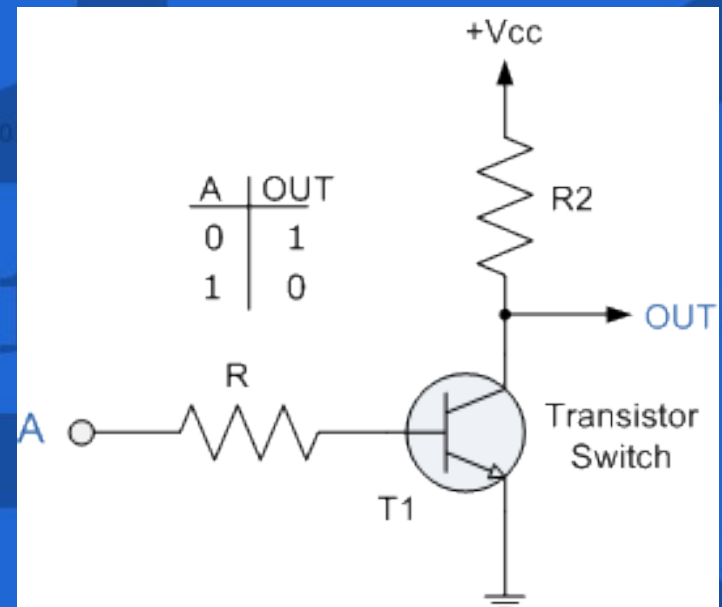
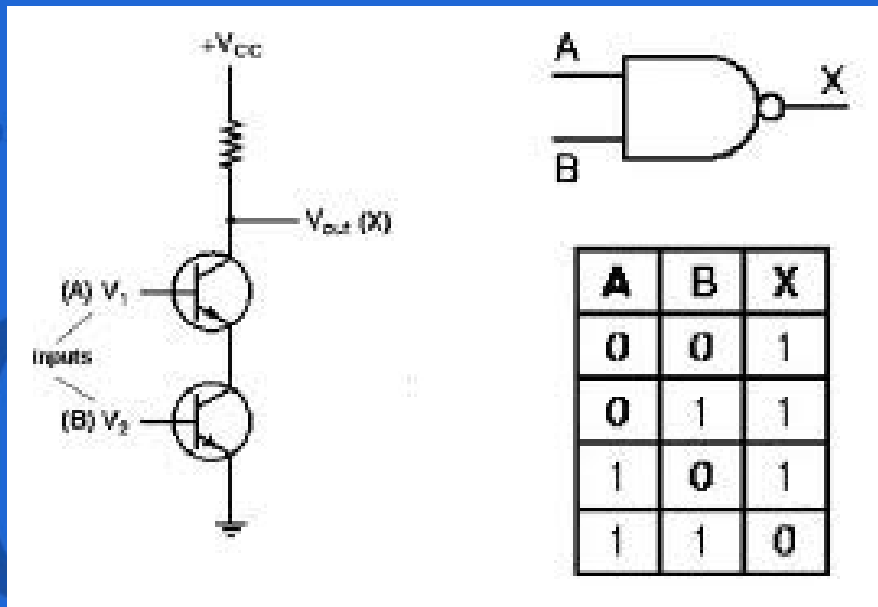


# Tranzystor jako klucz elektroniczny

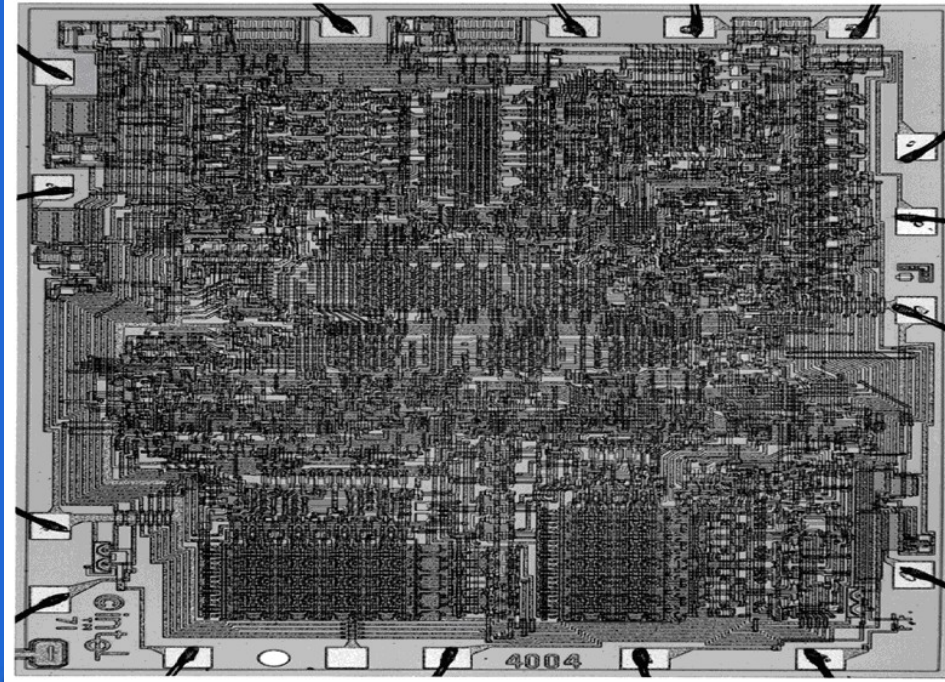




# Tranzystorowe bramki logiczne



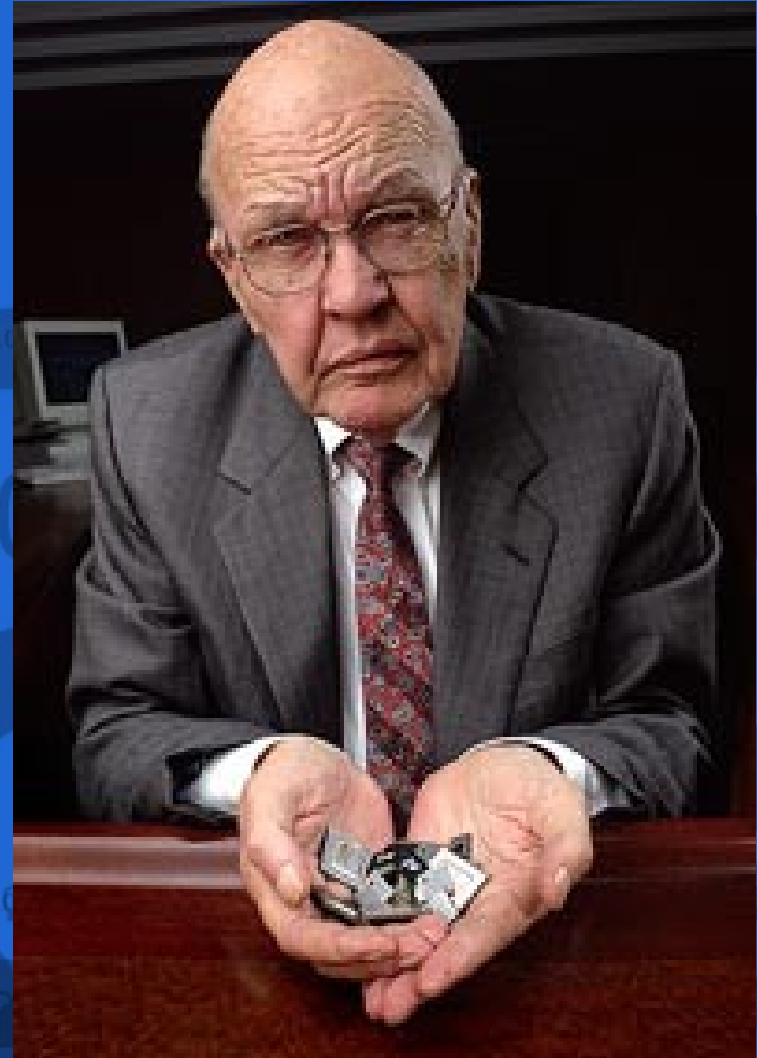
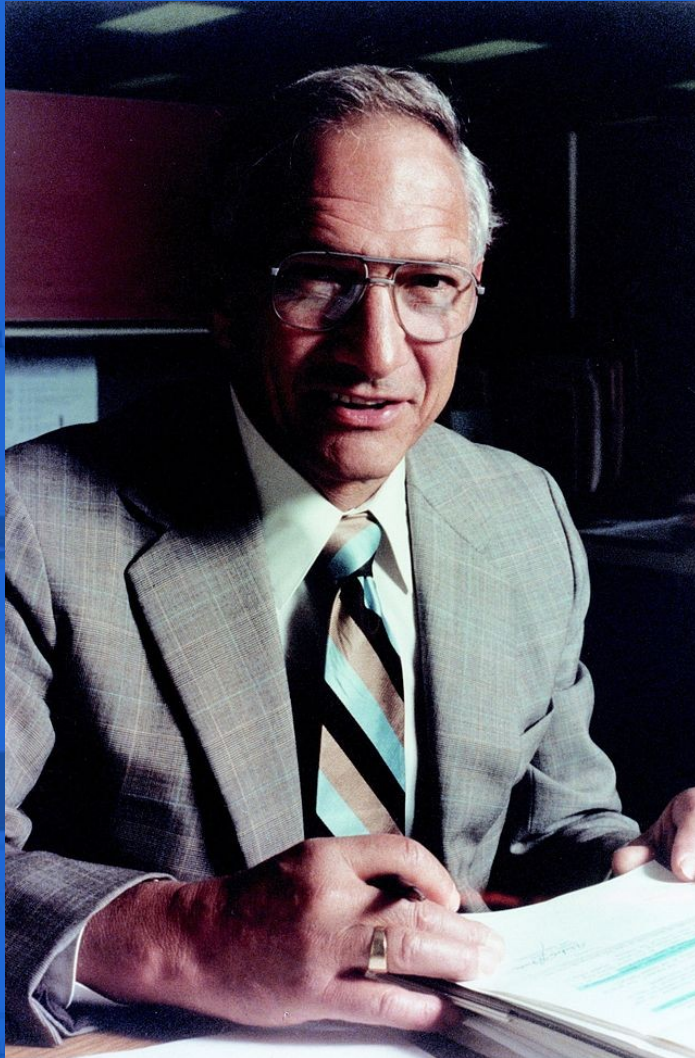
# Integracja elementów półprzewodnikowych – układy scalone



# Układy scalone:

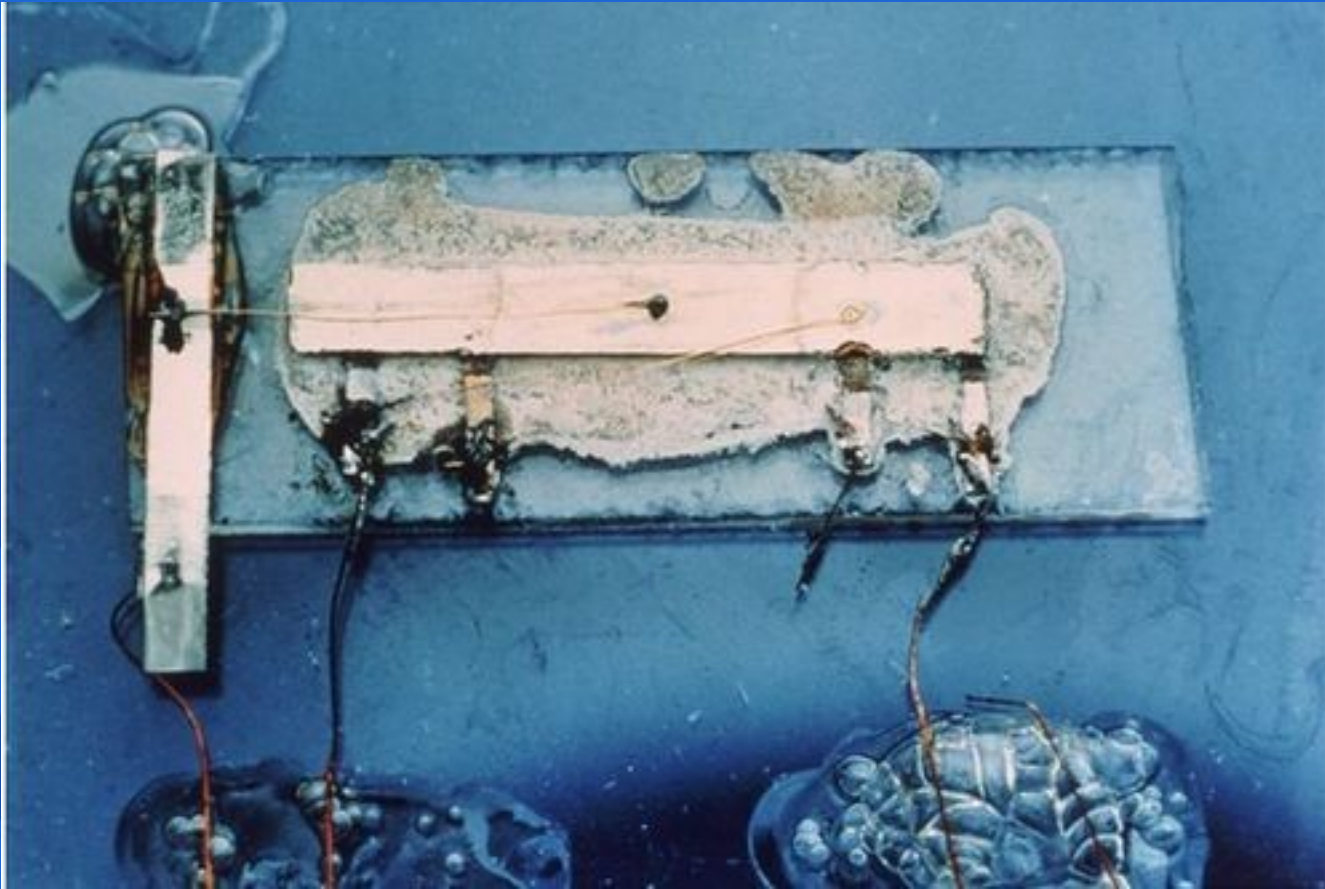
## Robert Noyce (Fairchild Semiconductor)

## Jack Kilby (Texas Instruments)





Jack Kilby 1958 r.  
nagroda Nobla z fizyki w 2000 r.



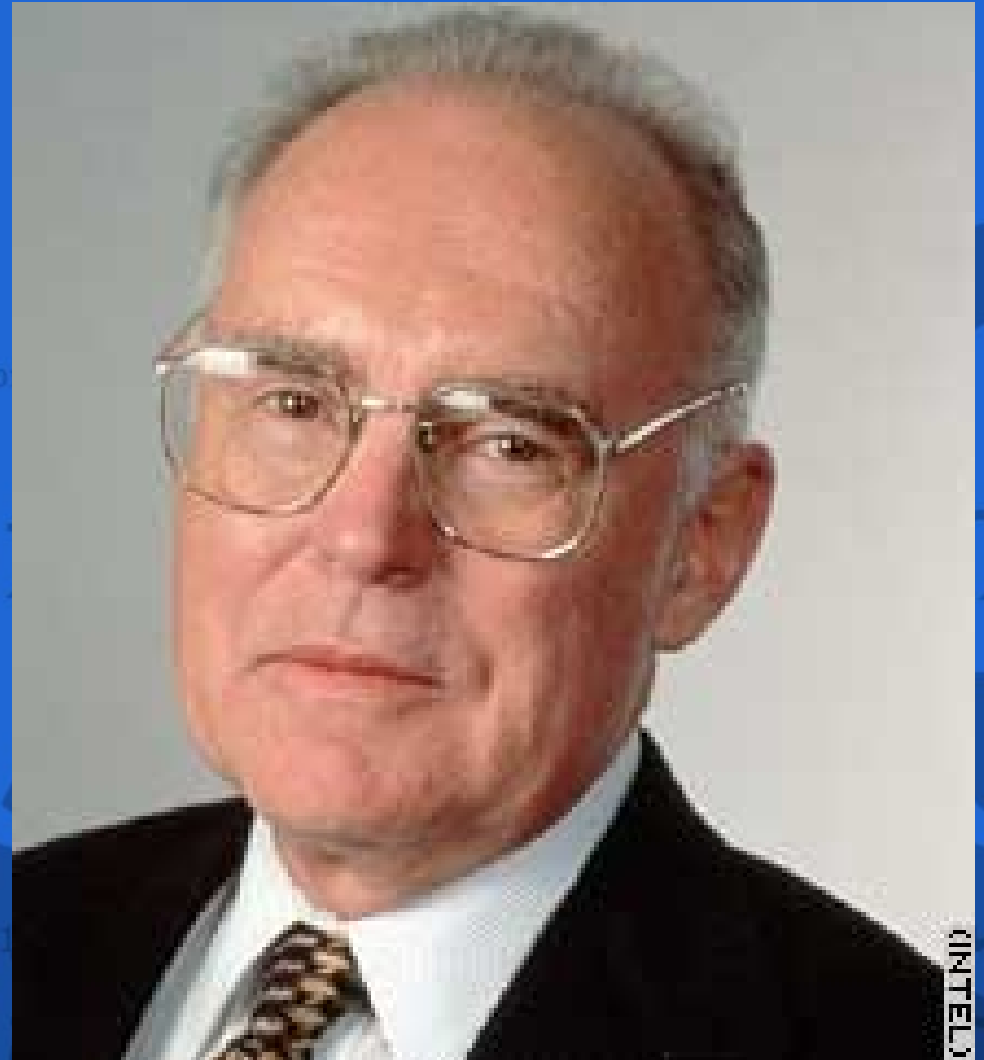
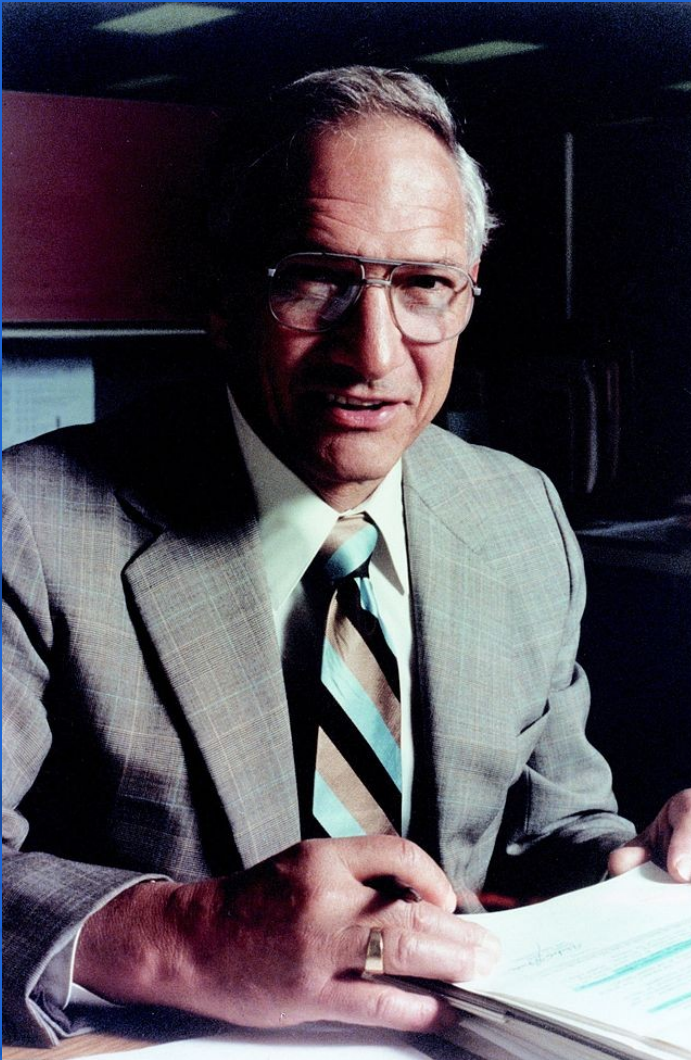
# Zdradziecka ósemka



From left to right: Gordon Moore, C. Sheldon Roberts, Eugene Kleiner, Robert Noyce, Victor Grinich, Julius Blank, Jean Hoerni and Jay Last. (1960)



# 1968 Intel (Integrated Electronics) Robert Noyce, Gordon H. Moore



# Generacje układów scalonych

Stopień scalenia	Liczba elementów aktywnych	Zastosowania	Rok
SSI	1-100	Bramki, wzmacniacze operacyjne	1960
MSI	100-1000	Rejestry, filtry	1965
LSI	1000-100000	Mikroprocesory, przetworniki AC, CA	1970
VLSI	100000-1000000	Pamięci, procesory, DSP	1975
(ULSI)	>1000000	Pamięci, procesory	2001