

Programowalne Układy Logiczne

Wykład I

dr inż. Paweł Russek



Literatura

- www.actel.com
- www.altera.com
- www.xilinx.com
- www.latticesemi.com
- Field Programmable Gate Arrays J.V. Oldfield, R.C. Dorf
- Field Programable Gate Array Technology ed. S.M. Trimberger
- Digital System Design And Prototyping Using Field Programmable Logic, Z. Salcic, A. Smailagic

Program wykładu

- Dlaczego PLD?
- Systematyka
- Geneza współczesnych układów PLD
 - Trochę teorii
 - Układy PAL/GAL, PLA
- Współczesne układy PLD
 - CPLD
 - FPGA
 - CPLD vs FPGA

Systemy cyfrowe



Dwa podejścia

?

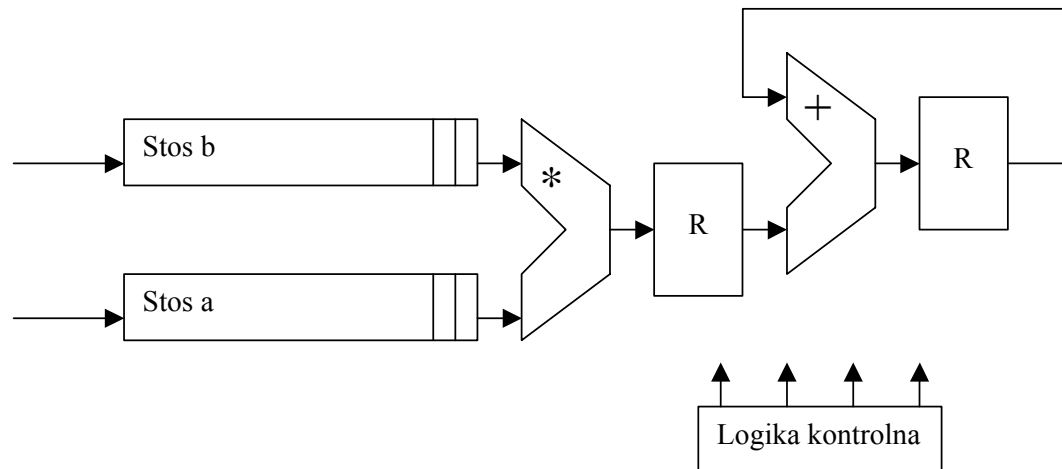
$$\sum_{i/0}^{n-1} a_i * b_i$$

1

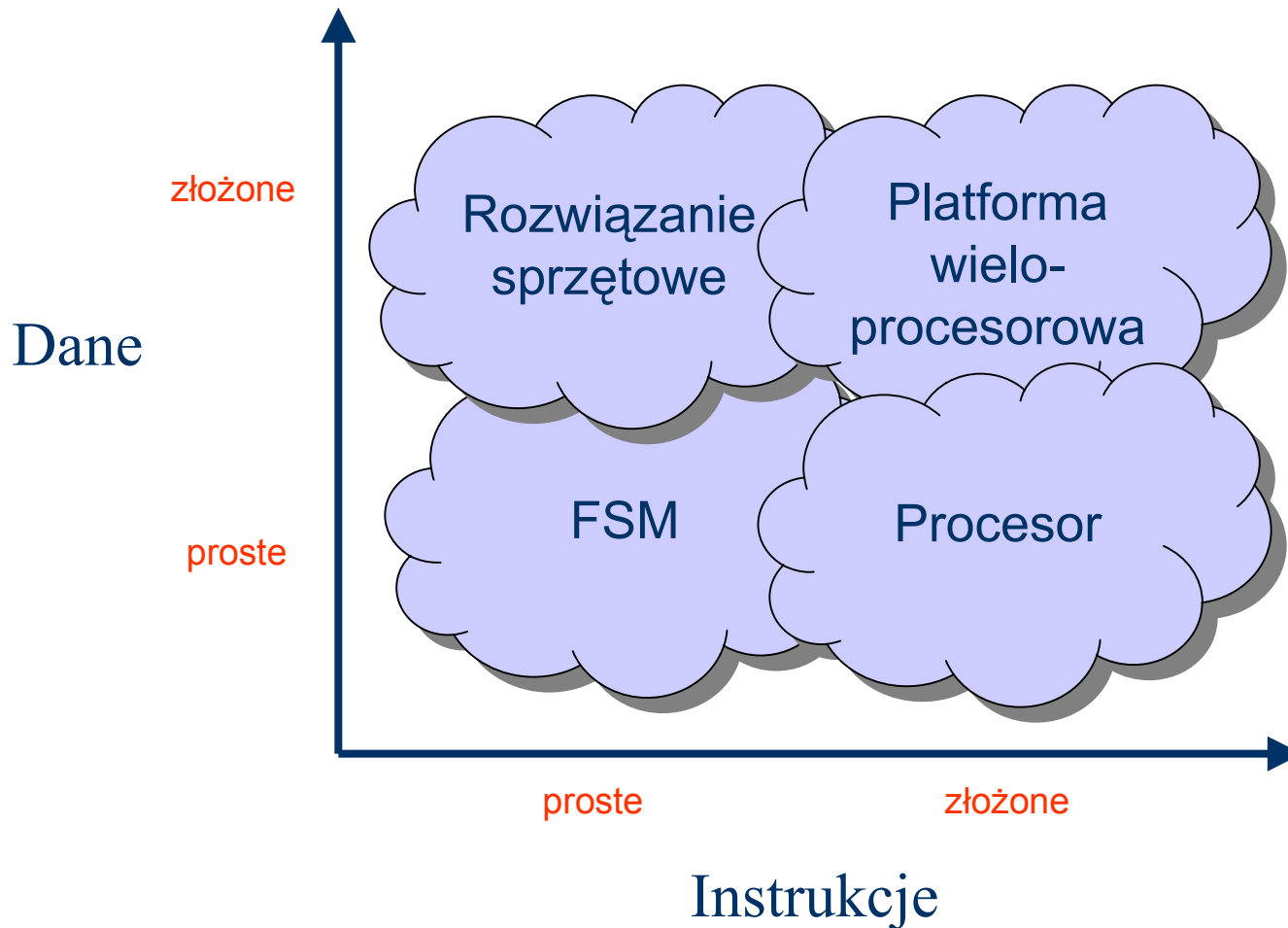
sum=0

for i=0 to n-1 do sum=sum+a(i)*b(i)

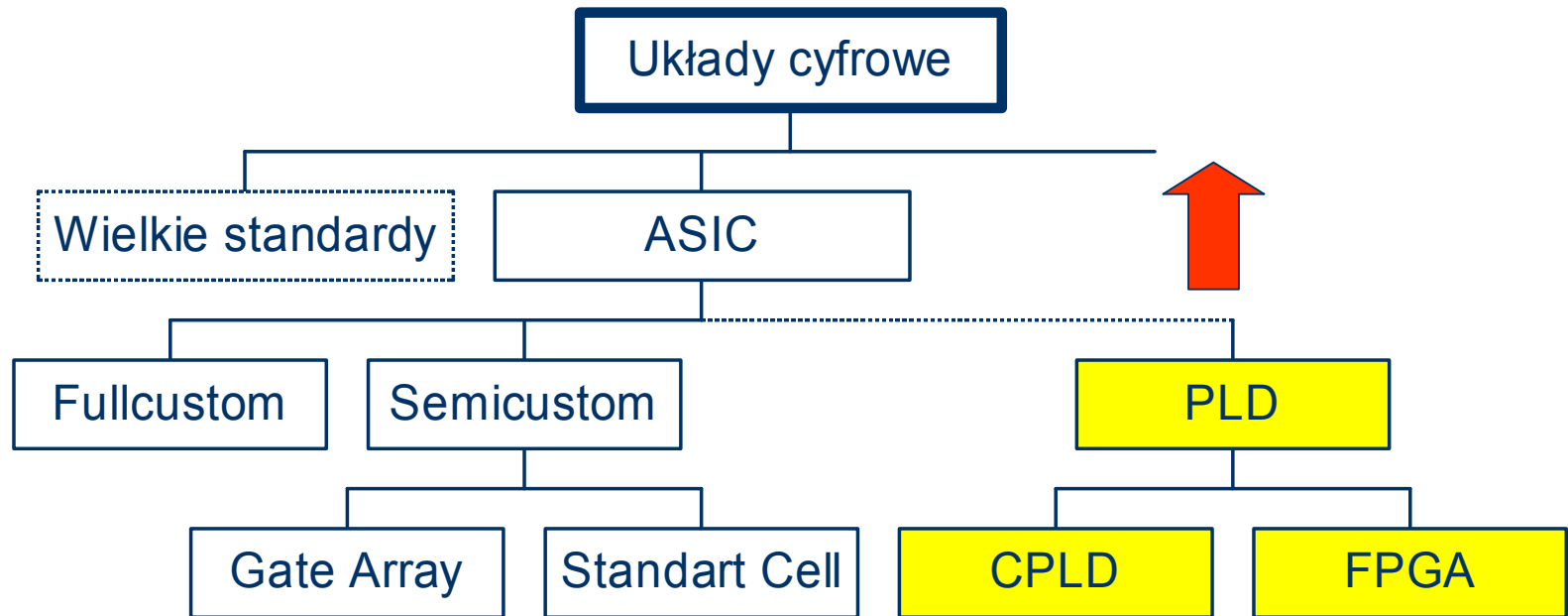
2



Które podejście?



Układy cyfrowe

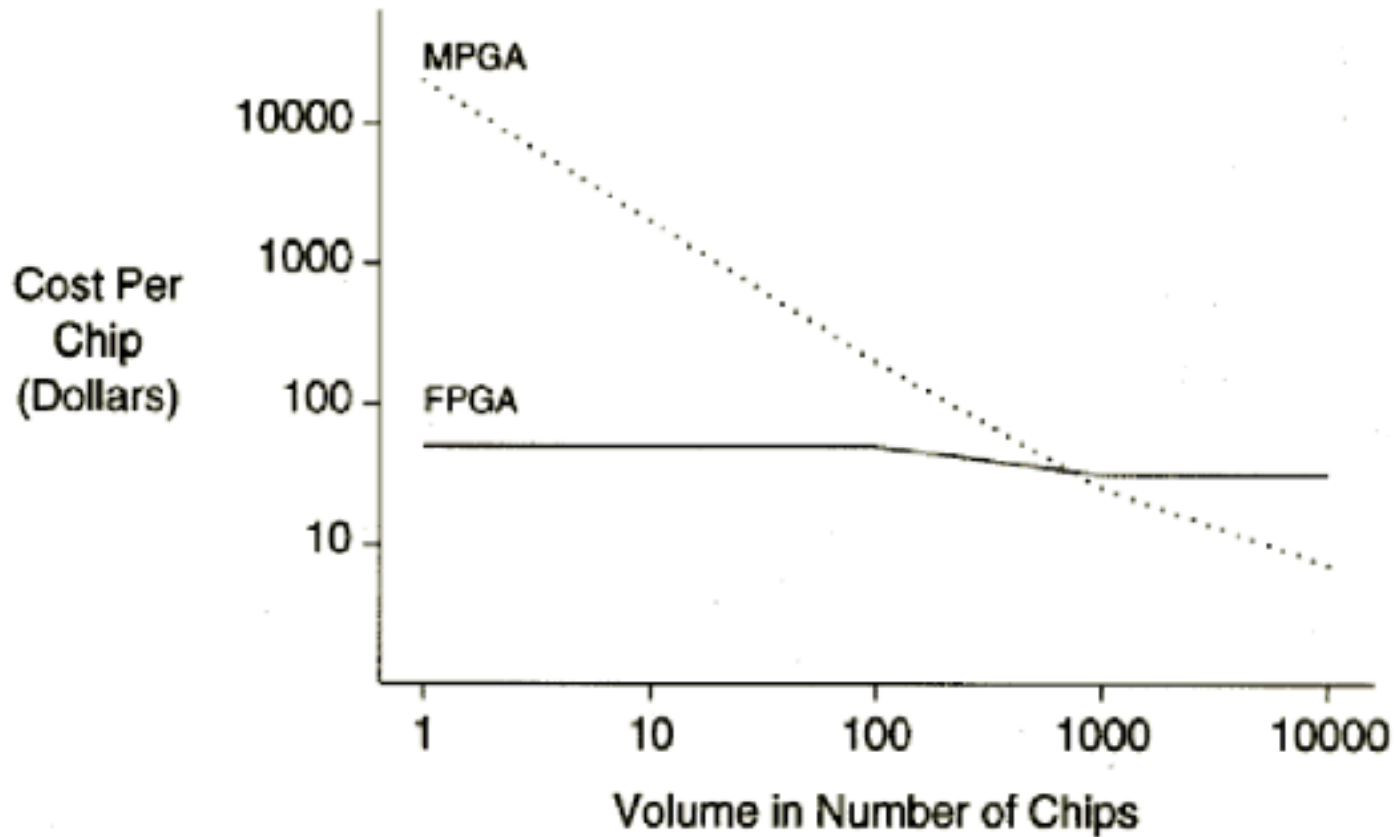


PLD vs ASIC

- Koszt implementacji projektu
- Modyfikacja projektu
- Koszt NRE
- Ryzyko projektowe
- Weryfikacja projektu
- Testowanie
- Koszt/rozmiary chipu
- Szybkość
- Jakość projektowania



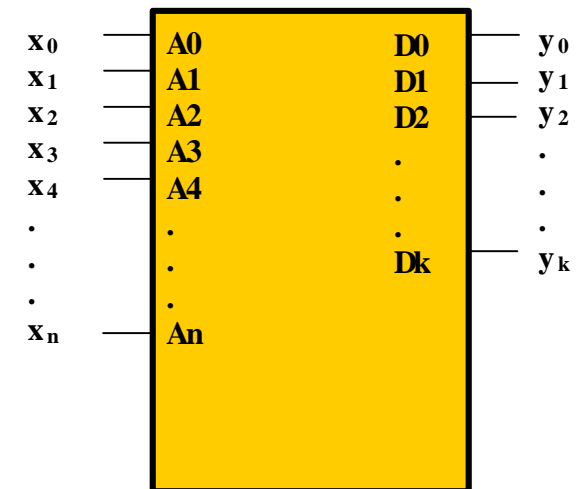
PLD vs ASIC – Opłacalność stosowania



Pamięć jako kombinacyjny PLD

- $(y_1, y_2, \dots, y_k) = F(x_1, x_2, \dots, x_n)$

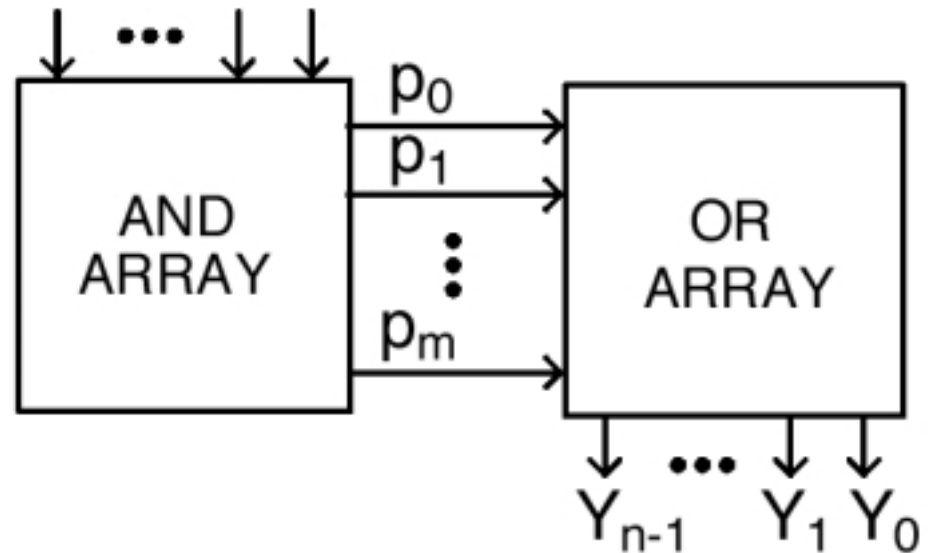
X	y ₁	y ₂		y _k
....00000	0	0		0
....00001	1	1		1
....00010	1	1		1
....00011	0	0		0
....00100	0	0		0
....00101	0	0		0
....00110	0	0		0
....00111	1	1		1



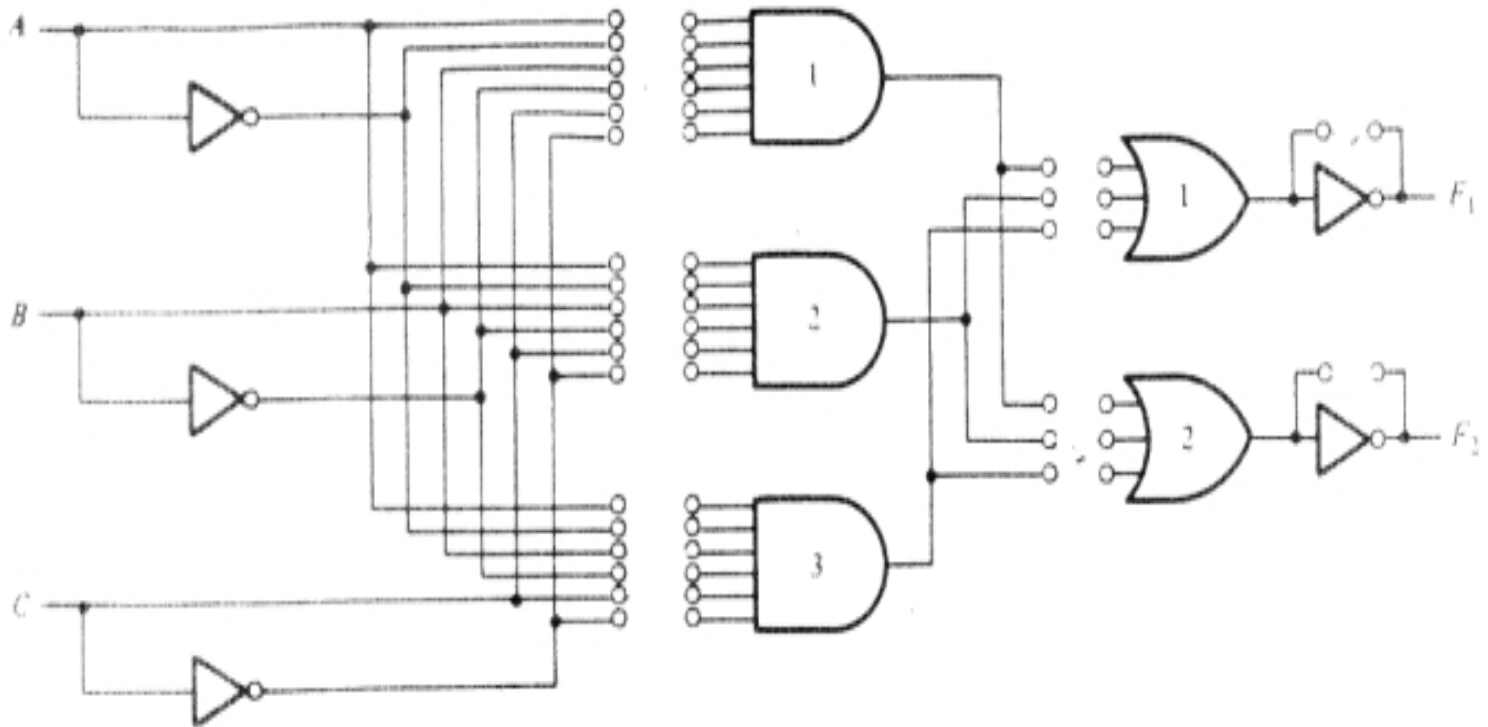
Rozwiązanie bardzo rozrzutne.

Postać kanoniczna funkcji logicznej

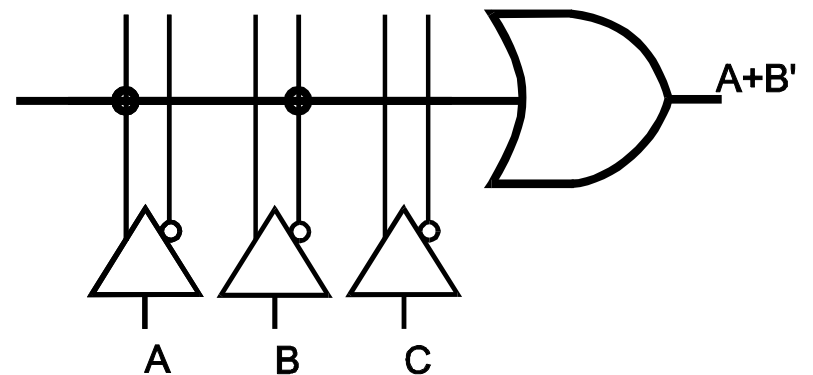
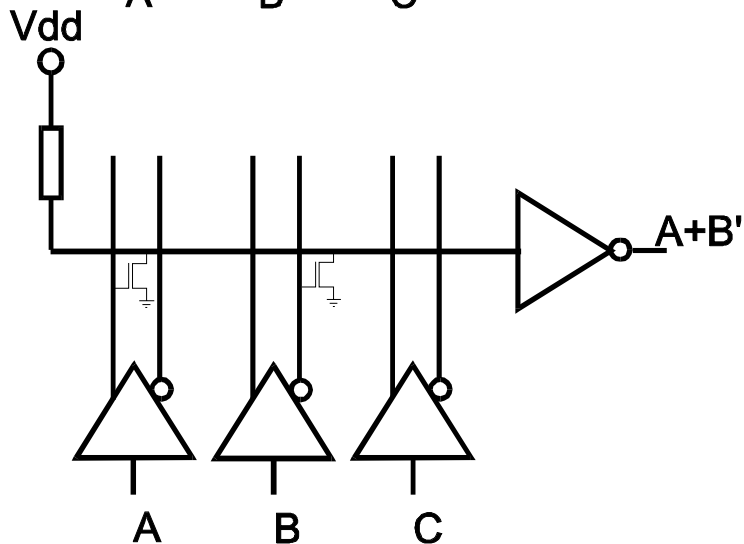
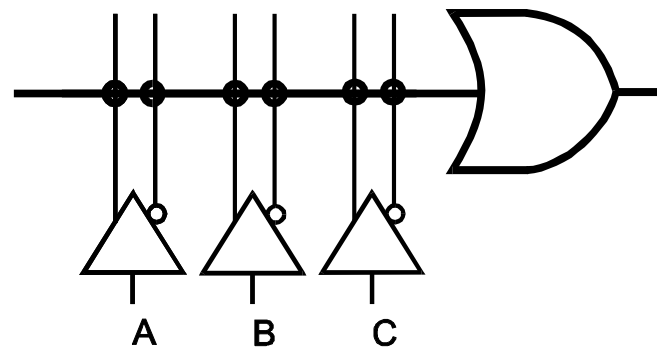
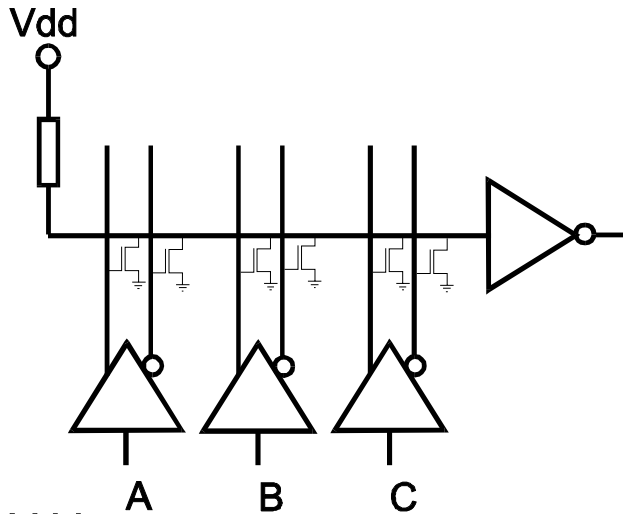
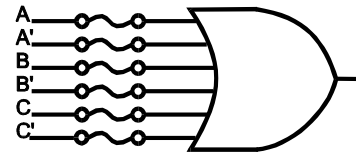
- $F(A,B,C,\dots)$
- Literały: $A, A', B, B', C, C', \dots$
- Mintermy: $A \bullet B' \bullet C, A' \bullet C, \dots$
- Postać kanoniczna:
 $A \bullet B + A \bullet C' + A \bullet B' \bullet C + \dots$



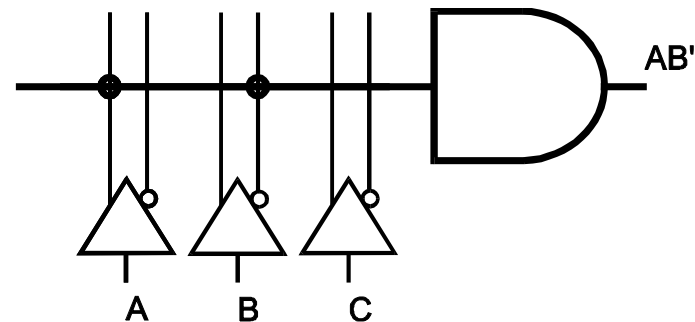
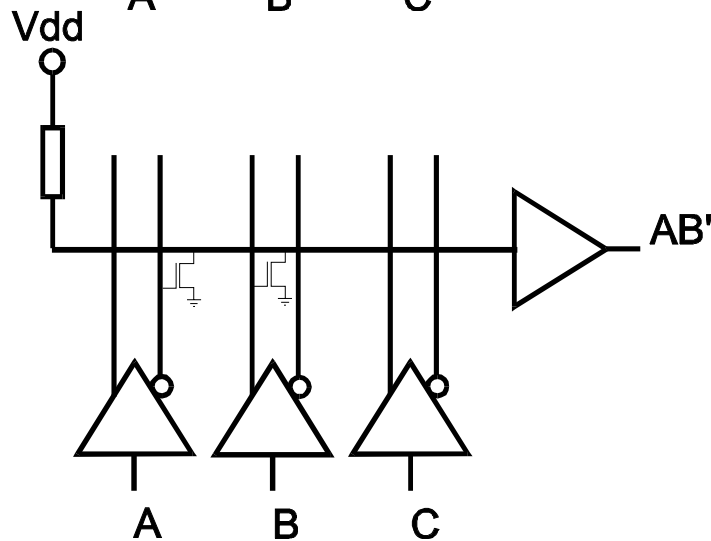
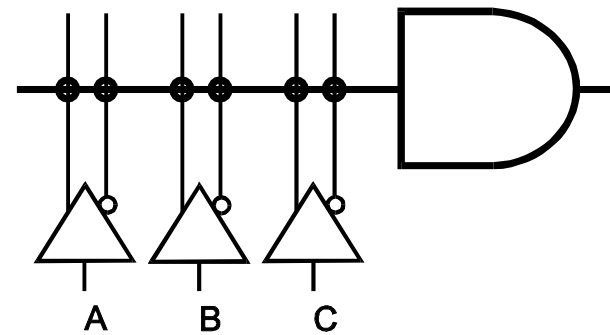
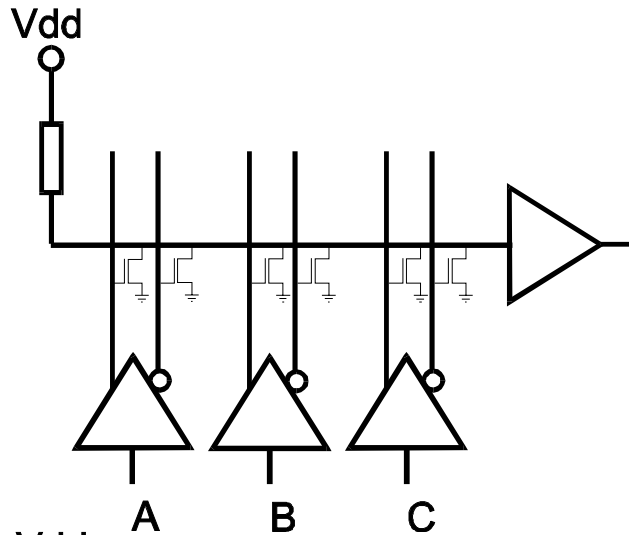
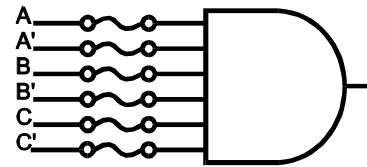
Programowalna sieć AND/OR



Realizacja matrycy OR



Realizacja matrycy AND

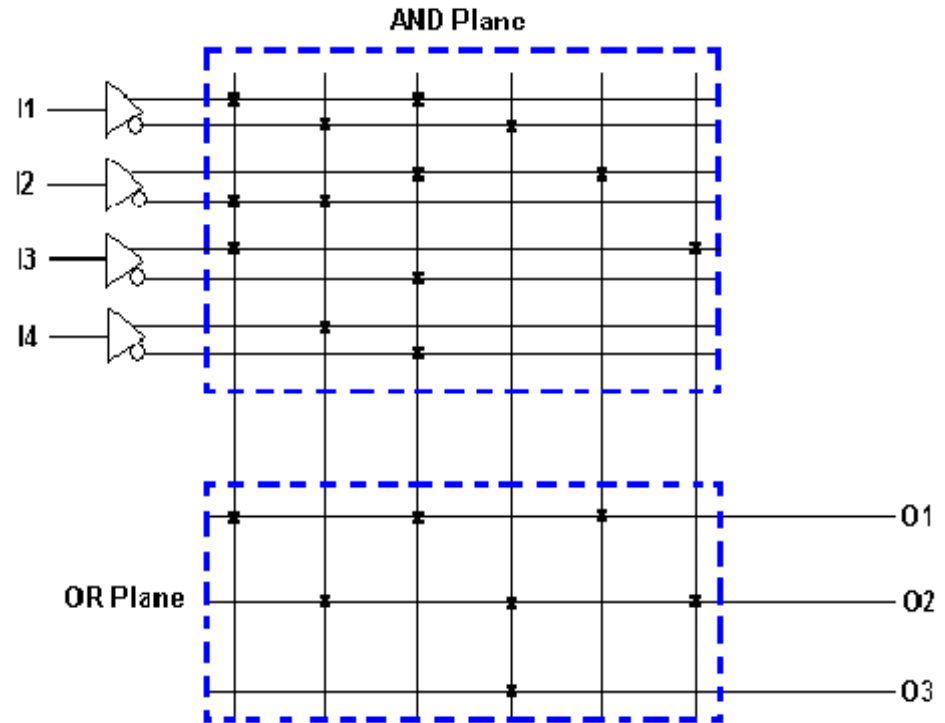


Układy typu PLA

- Programmable Logic Array

Cechy:

- Matryca OR i matryca AND programowalne
- Możliwe współdzielenie (min)termów przez funkcje wyjściowe
- Wejścia i wyjścia są jednokierunkowe.

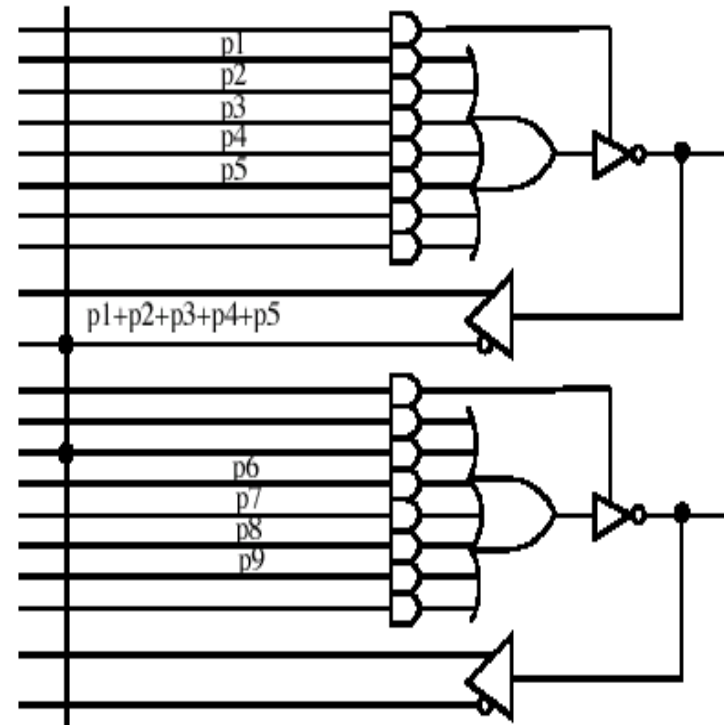


Układy typu PAL

- Programmable Array Logic

Cechy:

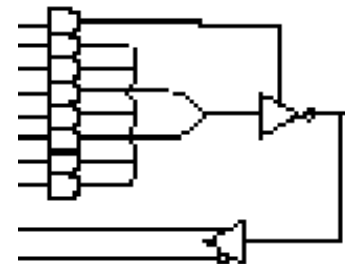
- Matryca AND programowalna
- Matryca OR stała
- Nie możliwe współdzielenie (min)termów przez funkcje wyjściowe
- Posiada dwukierunkowe wejścia/wyjścia
- Możliwość zwiększania liczby dostępnych dla jednego wyjścia termów



Rodzina układów PAL

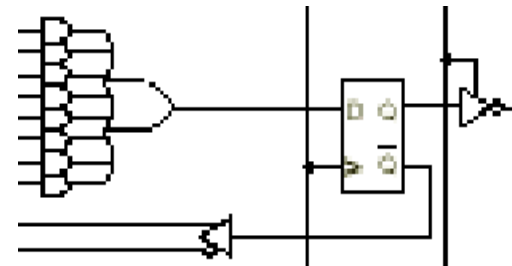
- **Kombinacyjne (PAL16L8[†])**

Do realizacji logiki kombinacyjnej



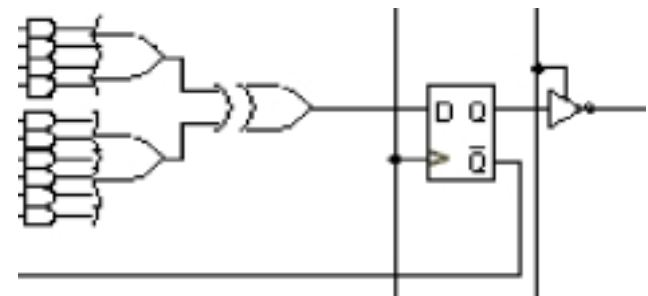
- **Sekwencyjne (PAL16R4[†])**

Do realizacji logiki sekwencyjnej (FSM)

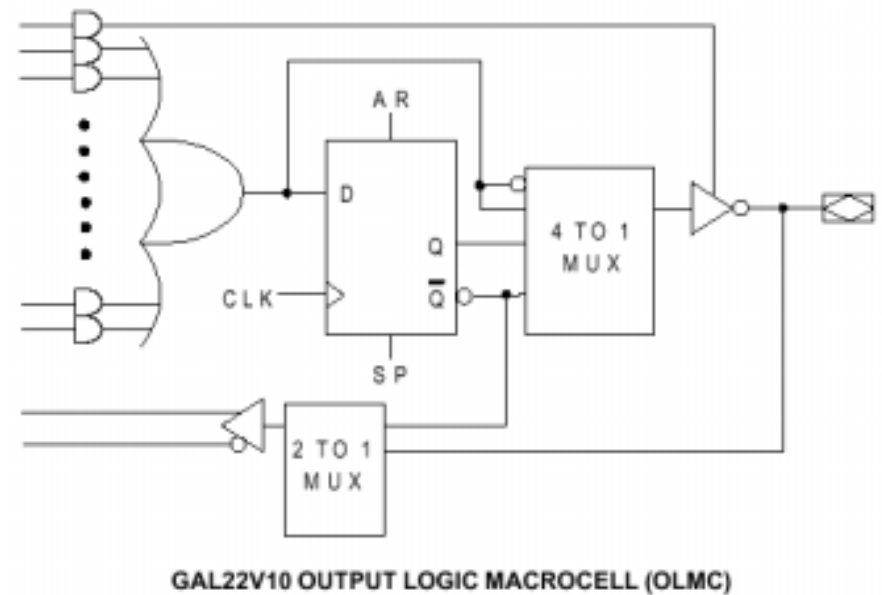
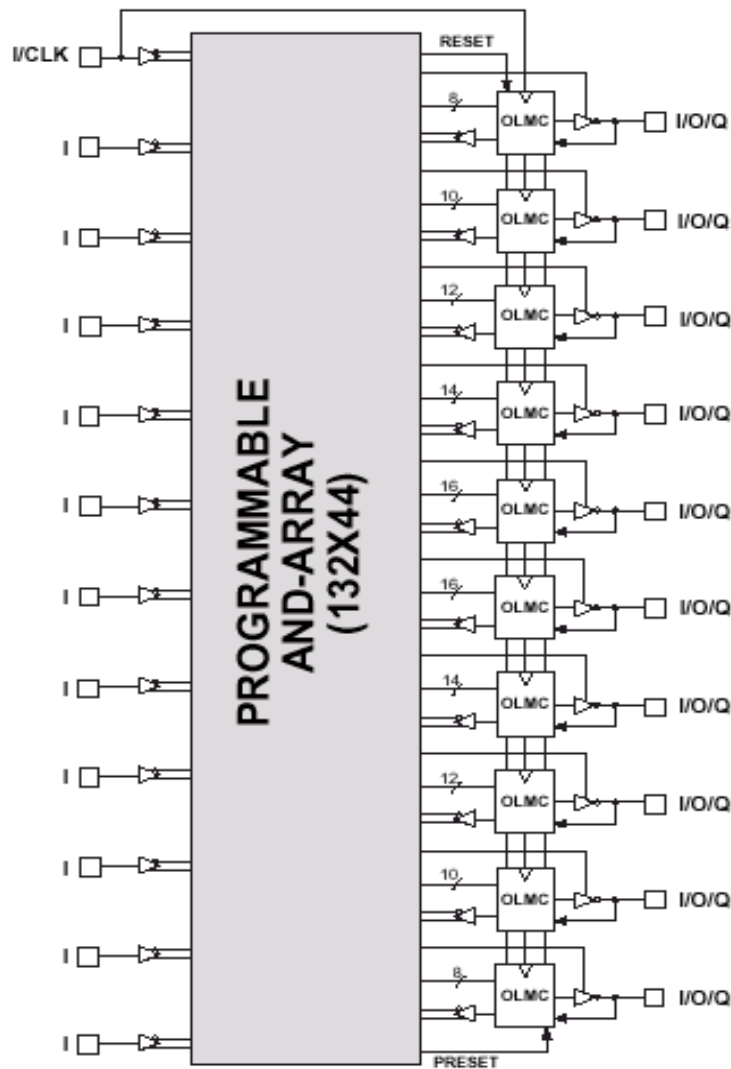


- **Arytmetyczne (PAL16A4[†])**

Sumy termalne mogą być łączone przez XOR

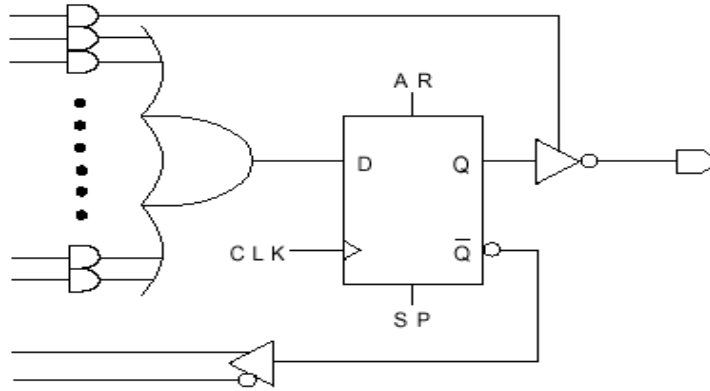


Legendarny 22V10

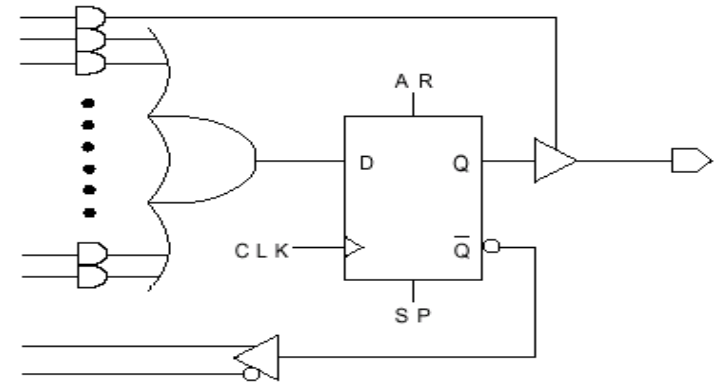


GAL22V10 OUTPUT LOGIC MACROCELL (OLMC)

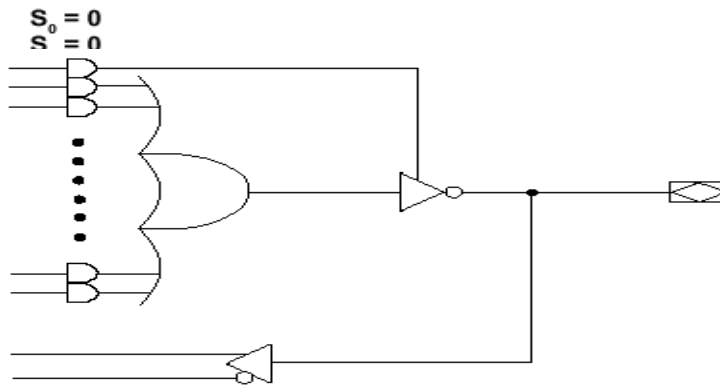
22V10 tryby pracy OLMC



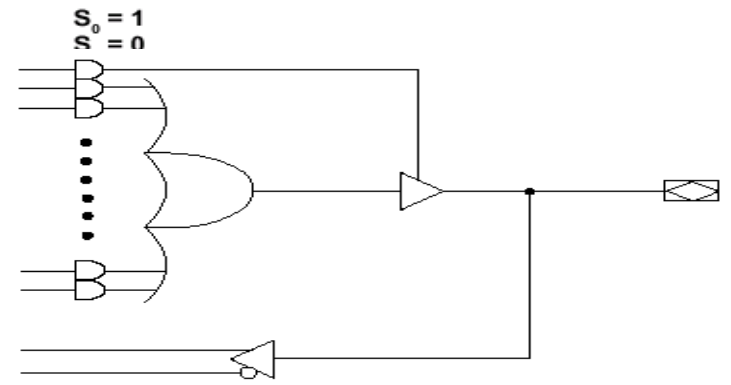
ACTIVE LOW



ACTIVE HIGH



ACTIVE LOW

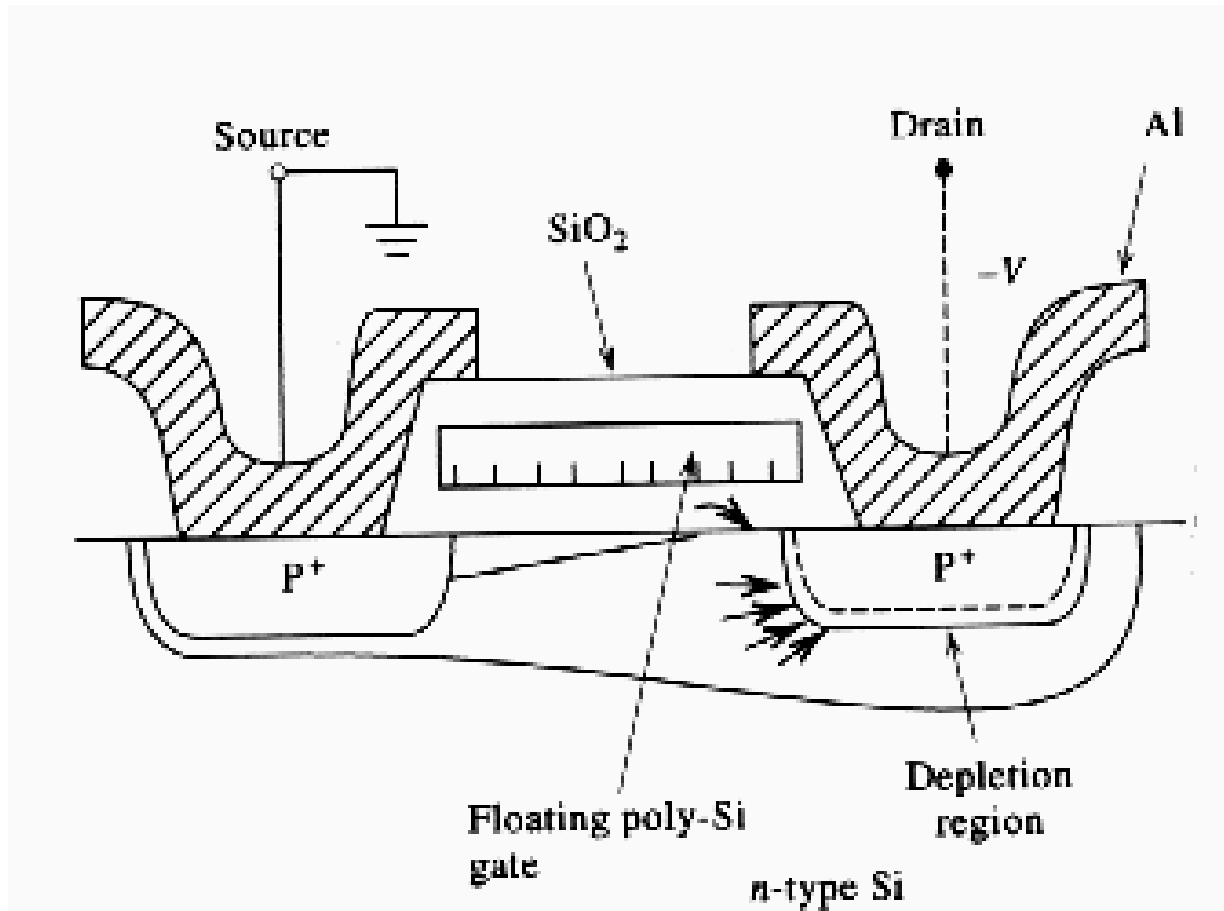


ACTIVE HIGH

$S_0 = 0$
 $S_1 = 1$

$S_0 = 1$
 $S_1 = 1$

Element programowania EPROM



Współczesne PAL

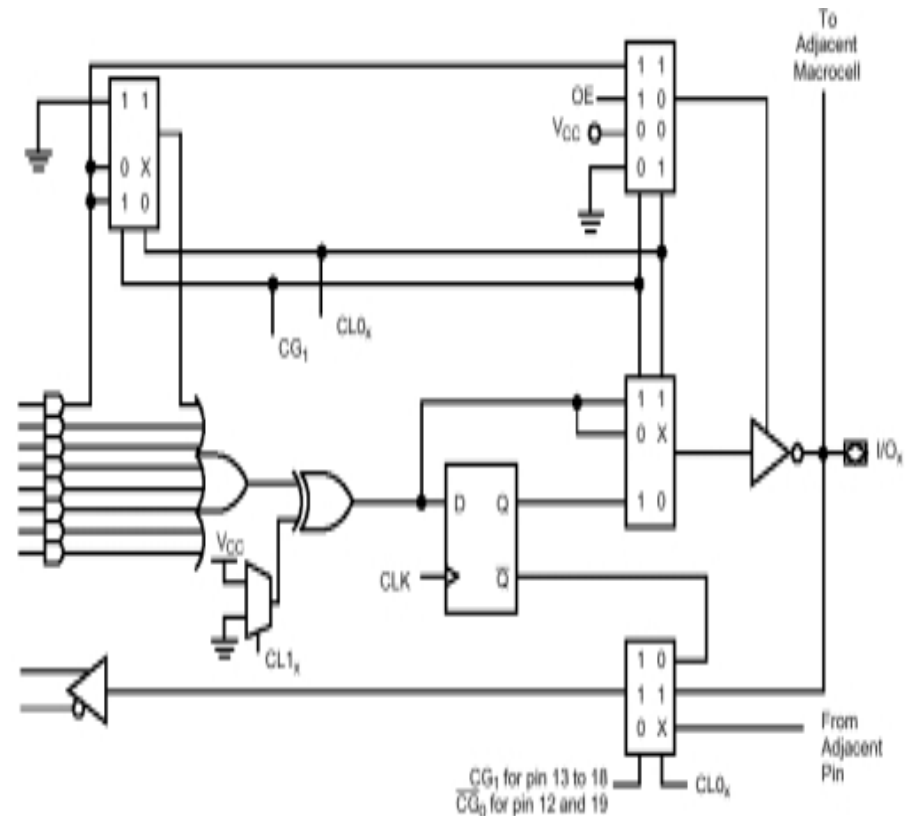
- PAL 16V8

16 wejść

8 wyjść

Wyjścia typu uniwersalnego
ang. variable

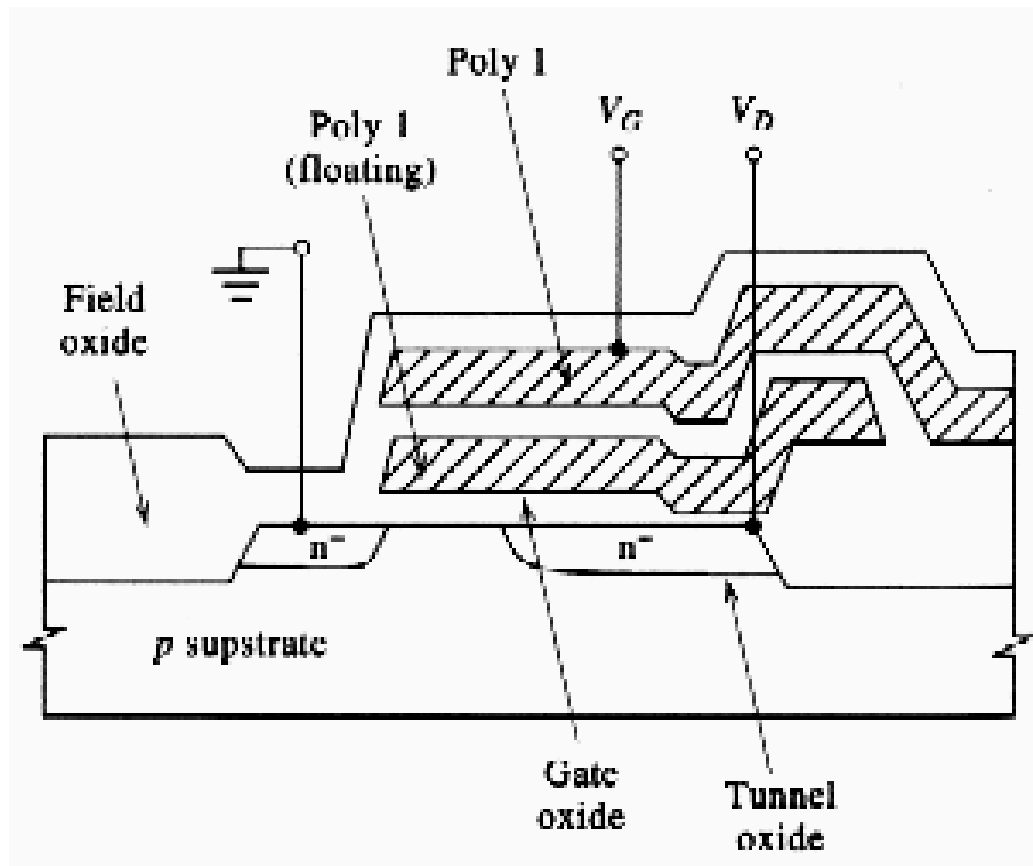
- 8 termów na wyjście
- 32 wejścia do matrycy AND
- Kontrolne funkcje 1 termu



OLMC

Output Logic Macro Cell

Element programowania EEPROM



PAL16V8/GAL16V8

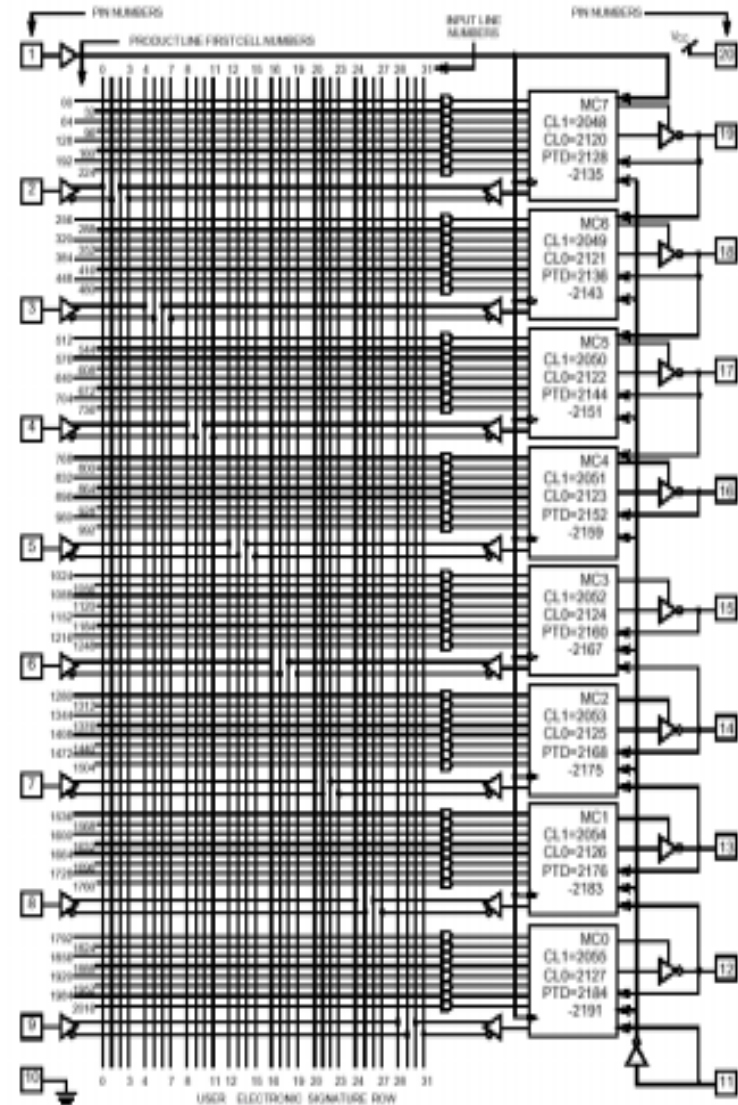
- GenericArray Logic

PAL

Programowanie oparte o technologię EPROM

GAL

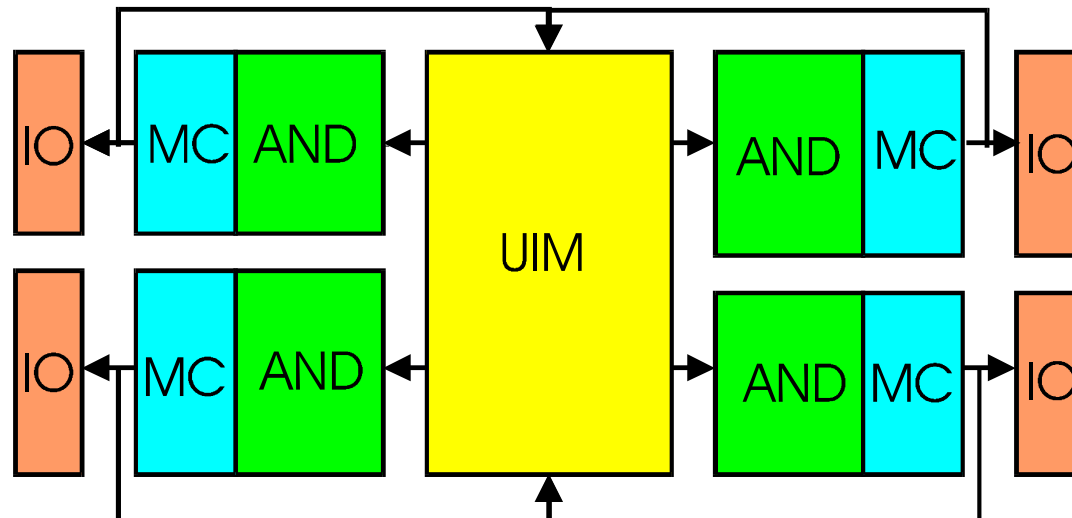
Programowanie oparte o technologię EEPROM



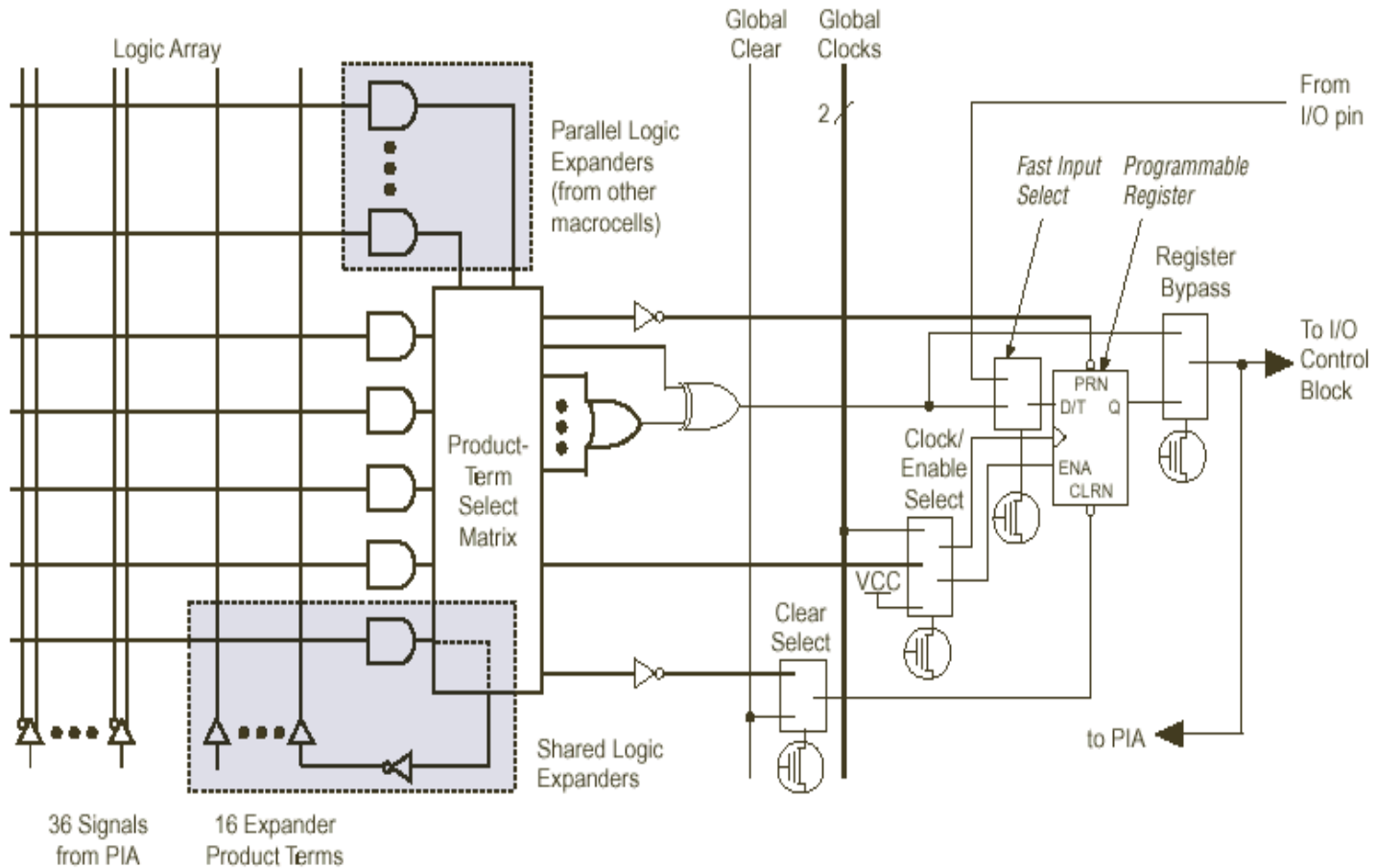
Układy CPLD

- Complex PLD

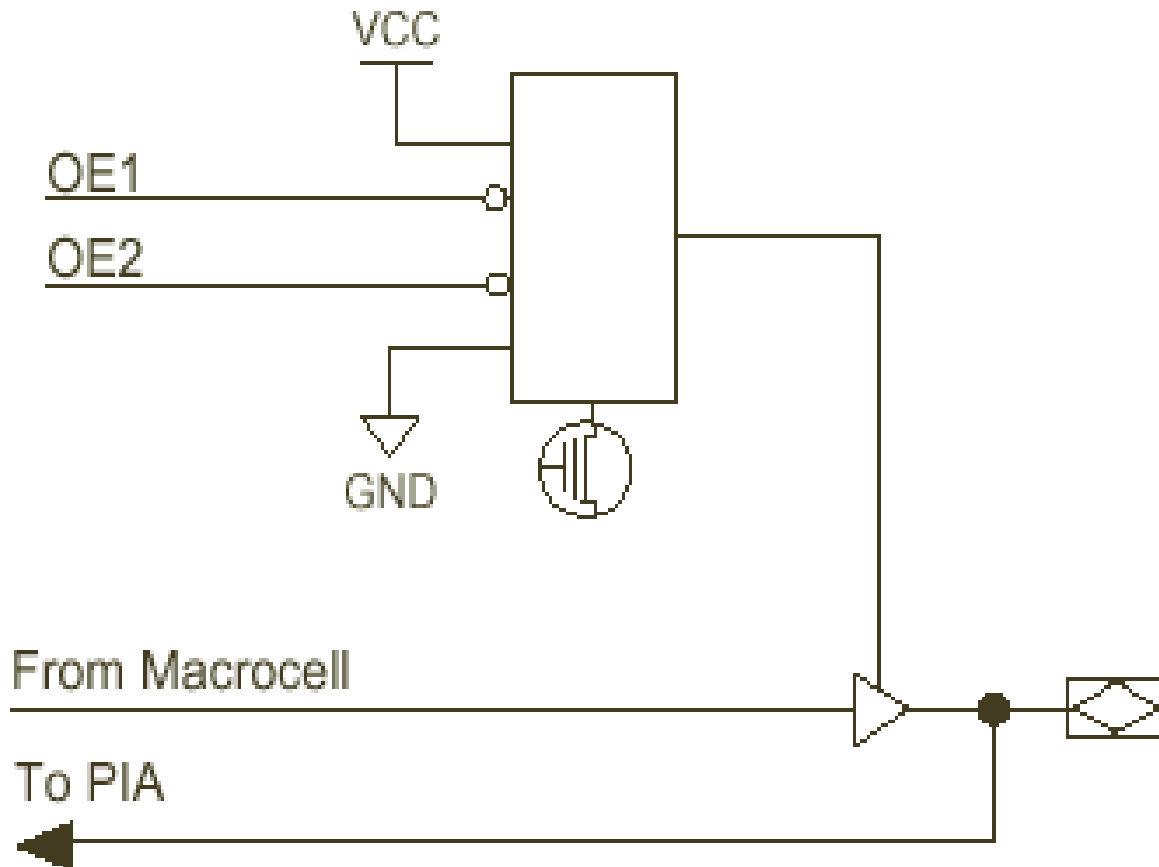
Układy CPLD to kombinacja kilku struktur PAL w jednym układzie. Typowo zawierają: struktury PAL (MC+AND), matrycę połączeniową (UIM/PIA) i bloki we/wy (IO).



CPLD. Macrocell

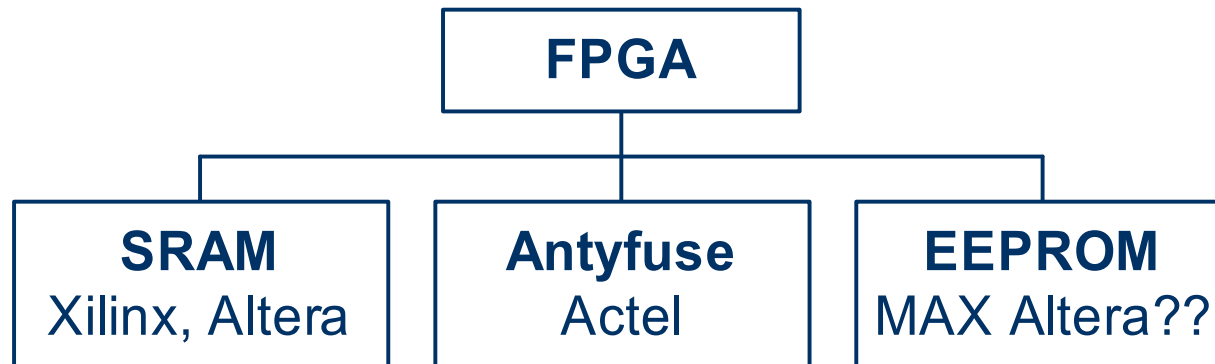


CPLD. Input Output Block



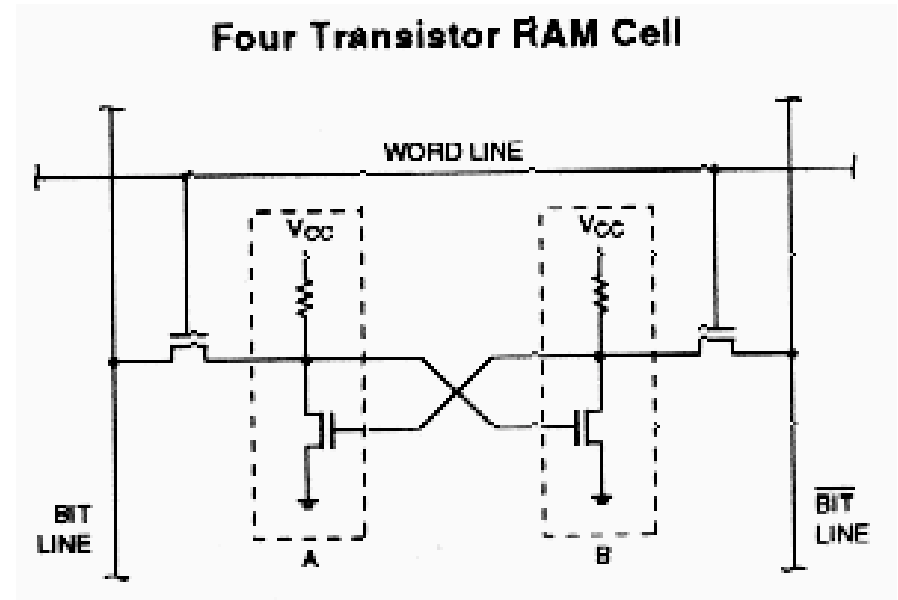
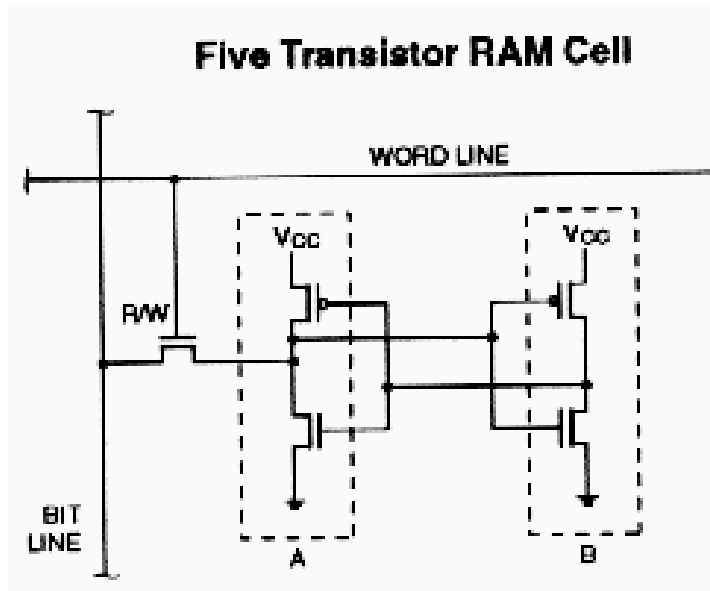
Układy FPGA

- **F**ield **P**rogrammable **G**ate **A**rrays
- Gate Array → FPGA
- Stanowią jednorodne medium realizacji układów cyfrowych
- Odpowiadają hierarchicznemu modelowi projektowania



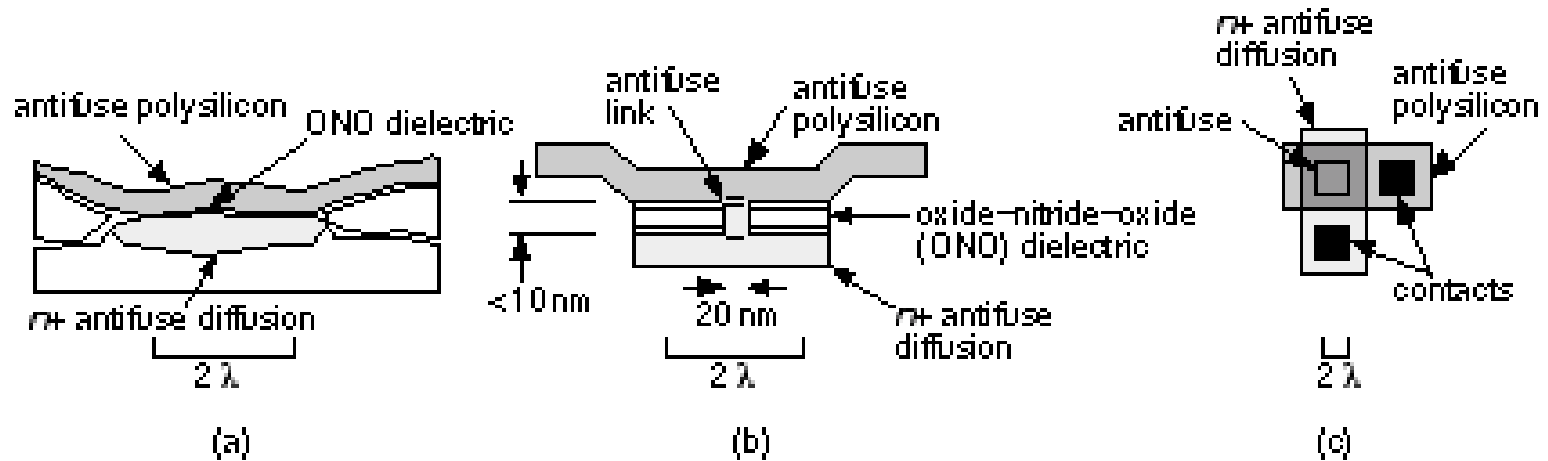
Element programowania SRAM

- Ulotność
- Pamięć zewnętrzna
- Reprogramowalność
- Jakość układów
- Technologia
- Moc



Element programowania „Antifuse”

- Rezystancja
- Rozmiary
- Czas programowania
- Testowalność
- Technologia
- Moc
- OTP



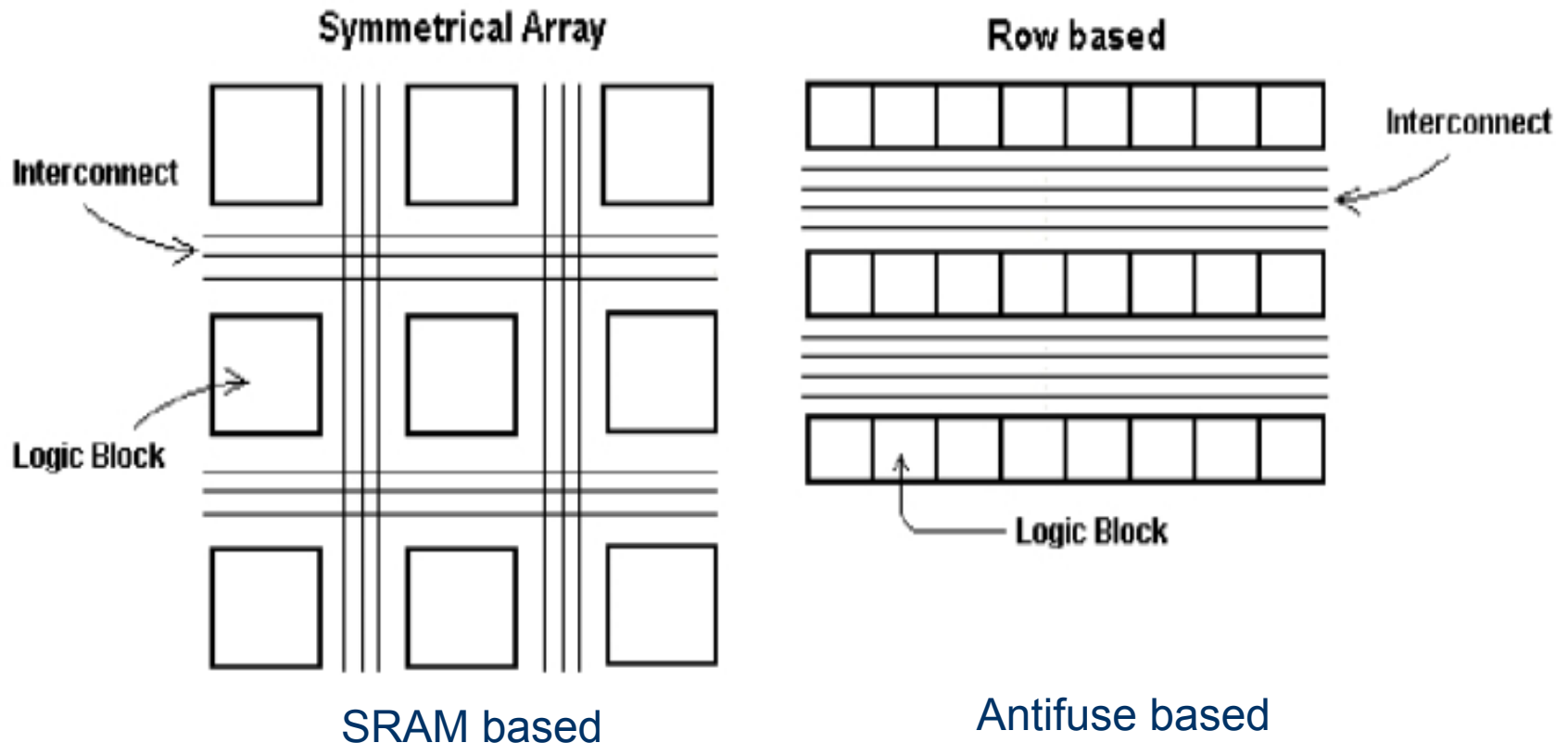
Actel antifuse

Elementy składowe FPGA

- Bloki logiczne. Realizują cyfrowe układy kombinacyjne i sekwencyjne.
- Matryce komutacyjne. Do realizacji sieci połączeń
- Zasoby połączeniowe
- Bloki we/wy. Połączenie bloków logicznych z wyprowadzeniami zewnętrznymi

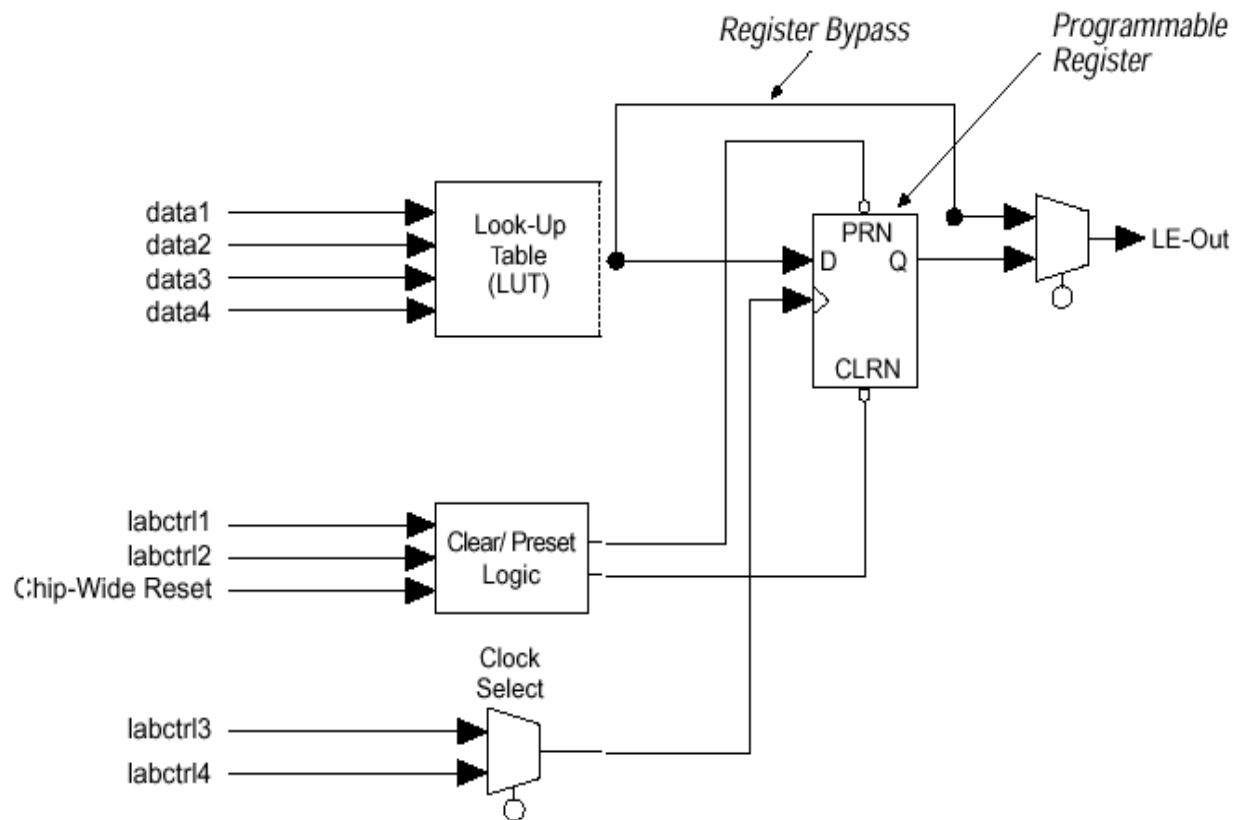
Klasyczna struktura FPGA

- Technologia programowania wymusza strukturę



Bloki logiczne (CLB, LE, MC)

Logic Element



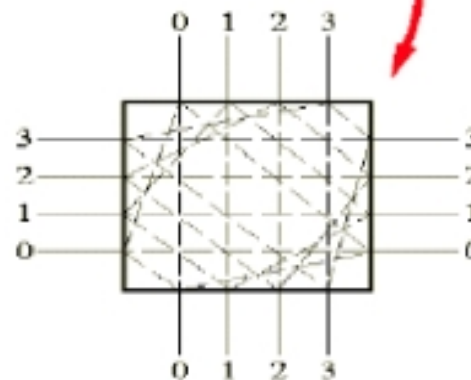
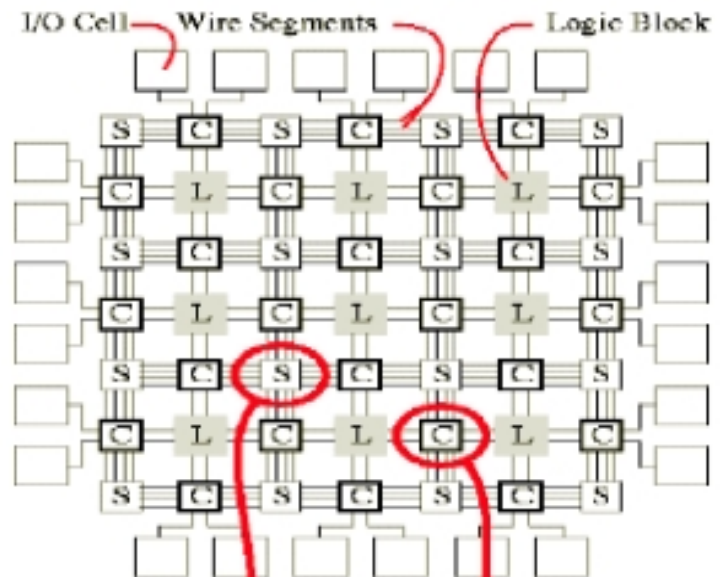
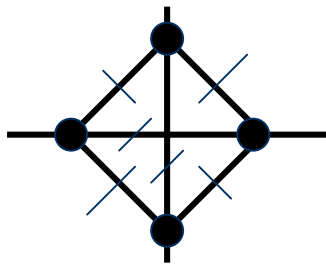
Matryca połączeń

Połączenia typu:

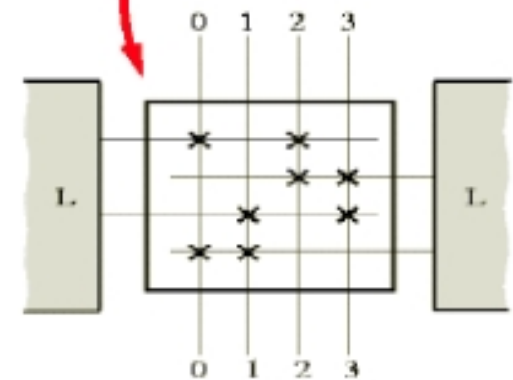
- Programmable interconnect „C”



- Switch matrix „S”



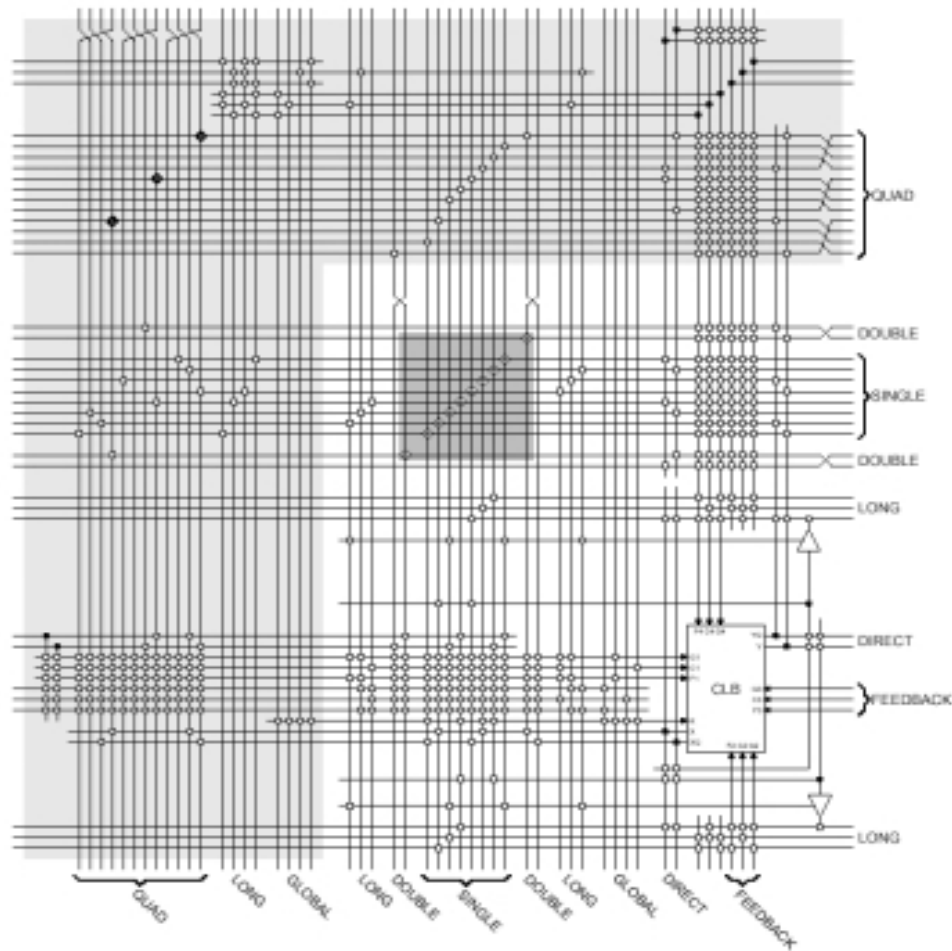
a) S block detail.



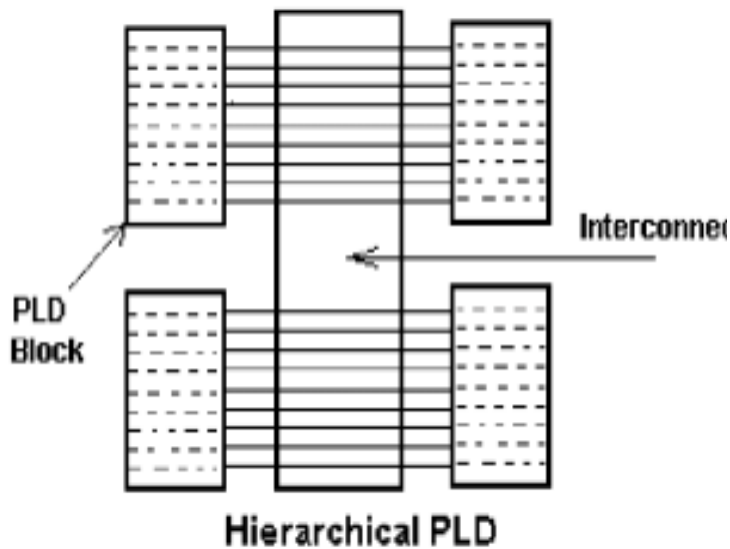
b) C block detail.

Sieć połączeń

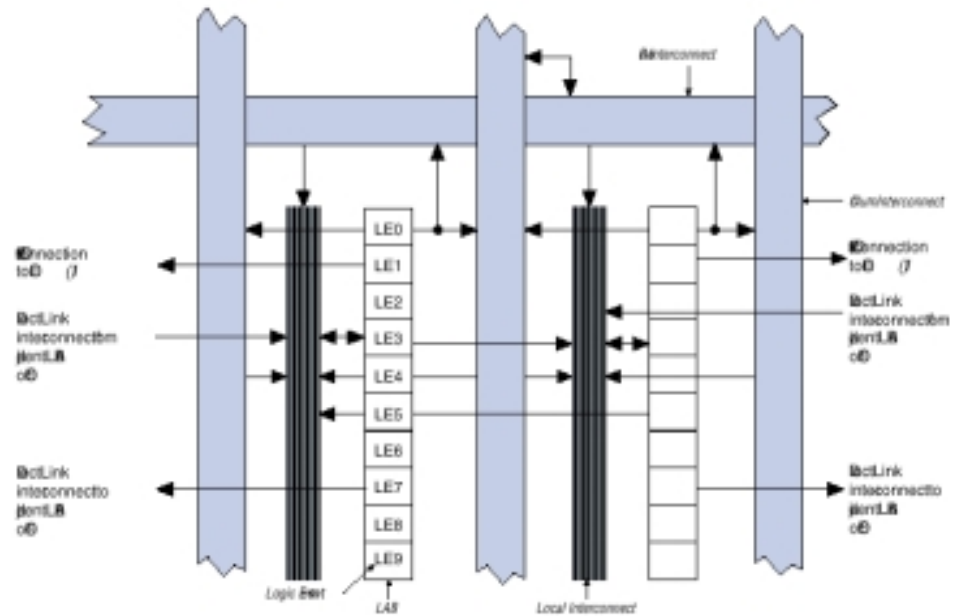
Zasoby połączeniowe decydują o zaletach i wadach FPGA.



CPLD czy FPGA?



EEPROM based



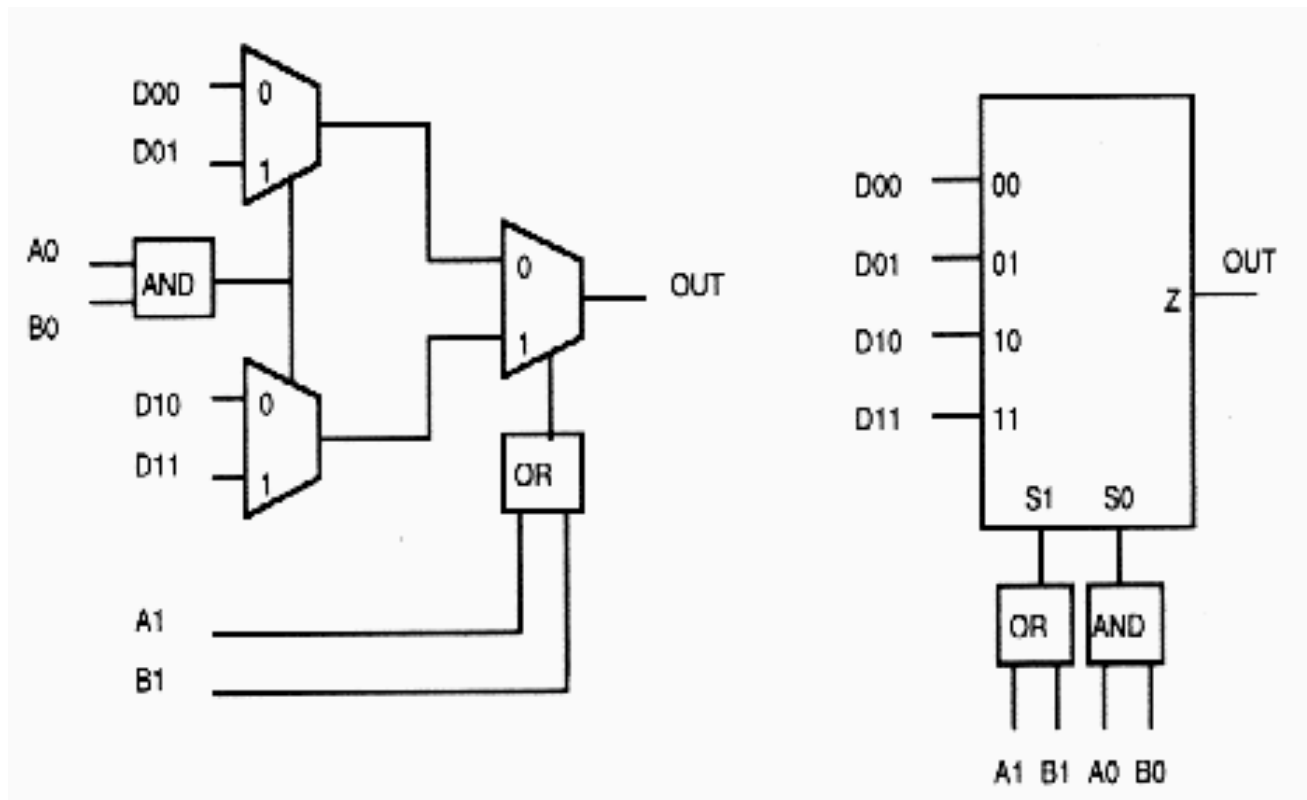
ALTERA MAX II structure

Multiplekser zamiast LUT

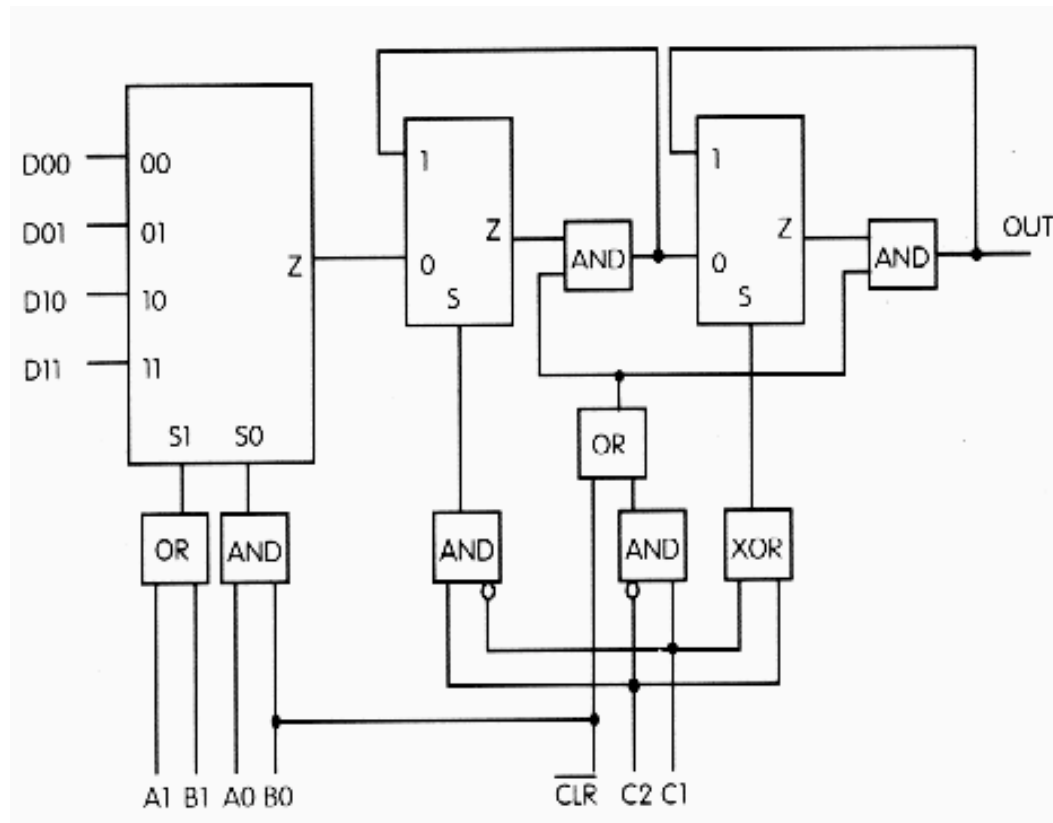
- Actel
- Sposób atrakcyjny przy małym rozmiarze klucza programowalnego - antifuse
- MUX 2:1 realizuje dowolną funkcję logiczną 2 zmiennych
- Realizacja technologiczna zgodna z realizacją sieci połączeń FPGA
- Duża funkcjonalność niewielkim kosztem

Przykład z MUX firmy Actel

- Realizuje wszystkie funkcje 3 zmiennych
- Realizuje wiele funkcji 4 zmiennych



Multipleksery w funkcji przerzutników



C1=0, C2=clk: rising edge flip-flop

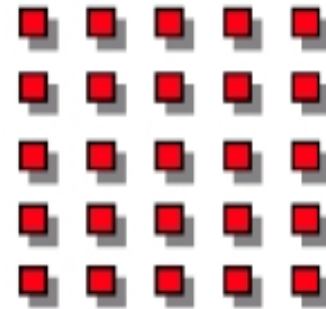
CPLD vs FPGA

- CPLD



- Struktura niejednorodna
- Średnia ilość zasobów
- Narzucona struktura połączeń
- Duża szybkość

- FPGA



- Struktura tablicowa
- Duża ilość zasobów
- Duża dowolność łączenia zasobów
- Średnia szybkość