НОВ БЪЛГАРСКИ УНИВЕРСИТЕТ

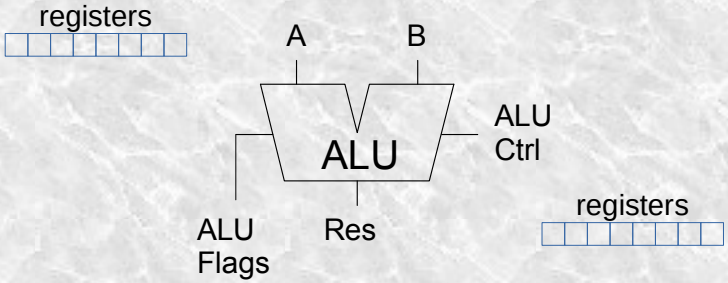
Компютърни архитектури
CSCB008

Логика, конфигурирана от потребителя

доц. д-р Ясен Горбунов
2021

Интегрални схеми с висока степен на интеграция

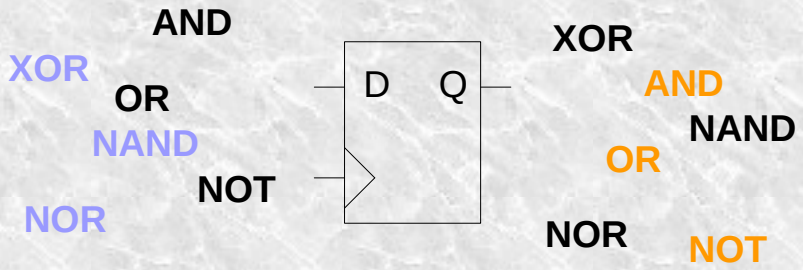
„твърда“ логика (микропроцесори)



фиксирана логика

- лесна конфигурация чрез <u>софтуер</u>
- последователно изпълнение на инструкции
- ниско ниво на паралелизъм
- **адаптиране на софтуера към хардуера**

„мека“ логика (CPLD и FPGA)

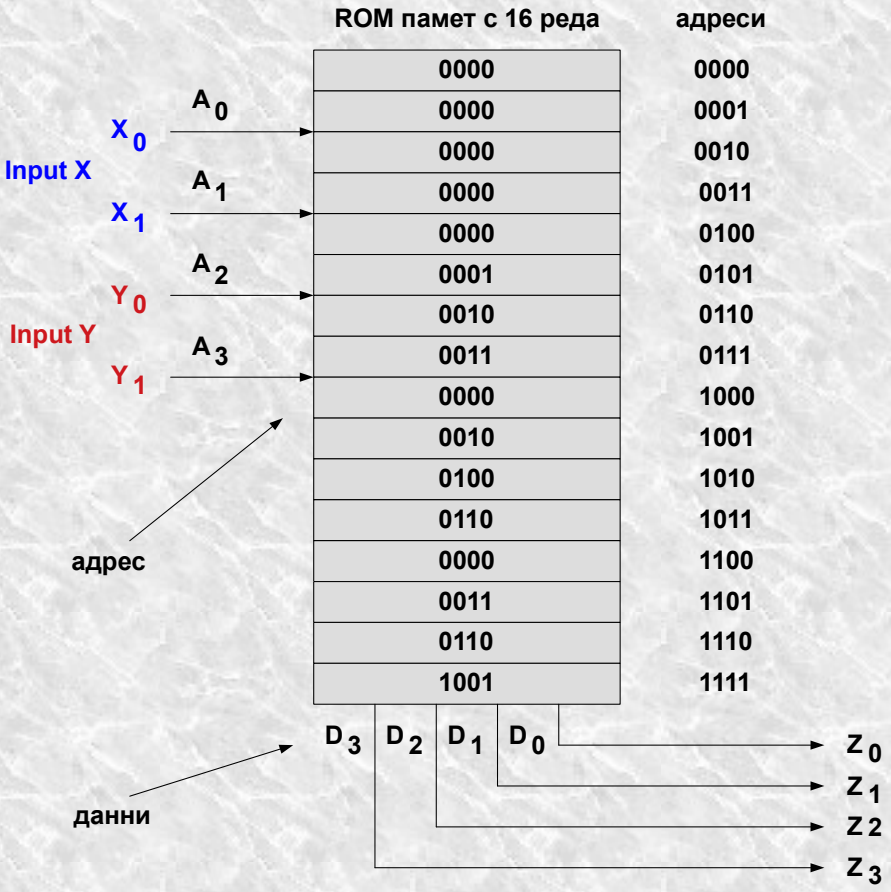


програмируема логика

- сложно конфигуриране – познаване на хардуера
- програмиране чрез език за хардуерно описание
- най-високо ниво на паралелизъм и гъвкавост
- **адаптиране на хардуера към софтуера**

Логика, конфигурирана от потребителя

- висока степен на интеграция при запазване на висока универсалност
- съставени от типични, регулярни елементи, вместо от специализирани схеми

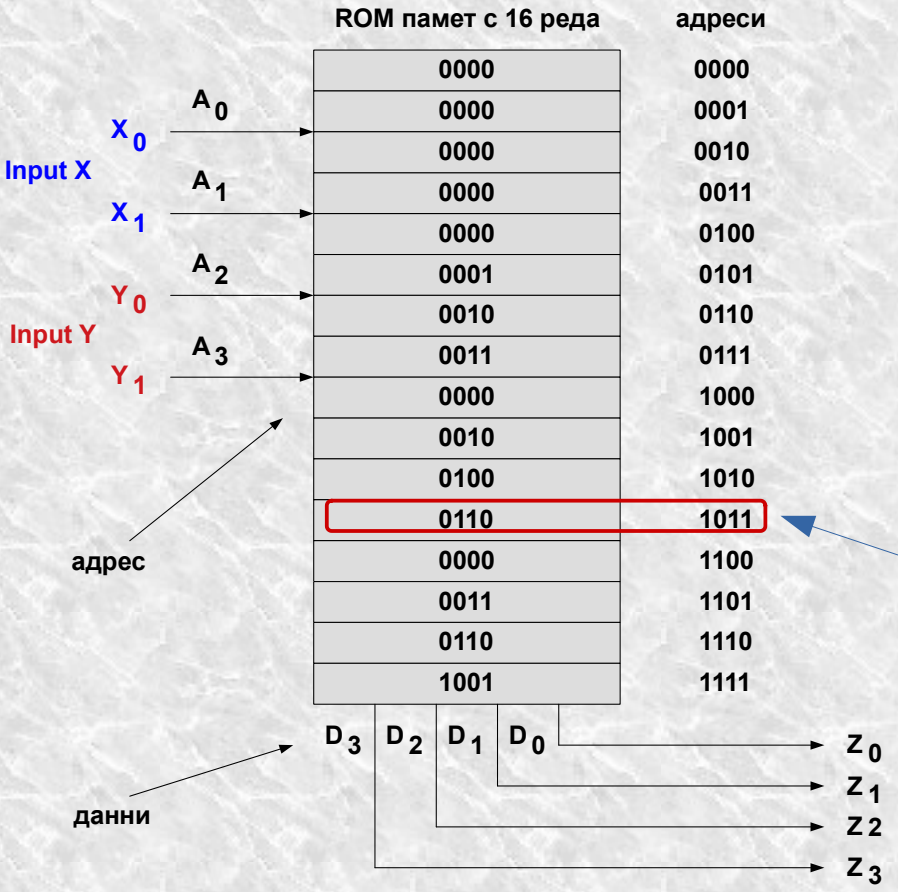


Типична схема с регулярна структура – ROM (Read Only Memory)

- Look-Up Table (LUT) – реализира директно таблицата на истинност
- n адресни входа, които определят 2^n клетки
- описва се с броя клетки – например 16 x 4 ROM има 16 клетки, всяка с по 4 бита

Логика, конфигурирана от потребителя

- висока степен на интеграция при запазване на висока универсалност
- съставени от типични, регулярни елементи, вместо от специализирани схеми



Типична схема с регулярна структура – ROM (Read Only Memory)

- Look-Up Table (LUT) – реализира директно таблицата на истинност
- n адресни входа, които определят 2^n клетки
- описва се с броя клетки – например 16 x 4 ROM има 16 клетки, всяка с по 4 бита

Пример

$2_{DEC} \times 3_{DEC} = 6_{DEC}$ $(10_{BIN} \times 11_{BIN} = 0110_{BIN})$

→ адрес 1011

→ стойност на клетката 0110

Логика, конфигурирана от потребителя

три базови типа програмируеми логически схеми

PROM (Programmable Read-Only Memory)

- лесно заменяеми
- ограничени до около 20 входа x 8 изхода (2^{20} x 8 = 8 Mbit)
- непрограмируема AND равнина и програмируема OR равнина
- OTP (One-Time Programmable) – програмирани от производителя
- PROM – изискват програматор

Програмируеми логически схеми

Логика, конфигурирана от потребителя

три базови типа програмируеми логически схеми

PROM (Programmable Read-Only Memory)

- лесно заменяеми
- ограничени до около 20 входа x 8 изхода (2^{20} x 8 = 8 Mbit)
- непрограмируема AND равнина и програмируема OR равнина
- OTP (One-Time Programmable) – програмирани от производителя
- PROM – изискват програматор

PLA (Programmable Logic Array) – Texas Instruments 1970, Philips

- наследник на PROM
- пълното име е FPLA (Field Programmable Logic Array)
- две отделно програмируеми равнини (AND и OR)
- по-трудно производство, по-големи закъснения на сигналите

Програмируеми логически схеми

Логика, конфигурирана от потребителя

три базови типа програмируеми логически схеми

PROM (Programmable Read-Only Memory)

- лесно заменяеми
- ограничени до около 20 входа x 8 изхода (2^{20} x 8 = 8 Mbit)
- непрограмируема AND равнина и програмируема OR равнина
- OTP (One-Time Programmable) – програмирани от производителя
- PROM – изискват програматор

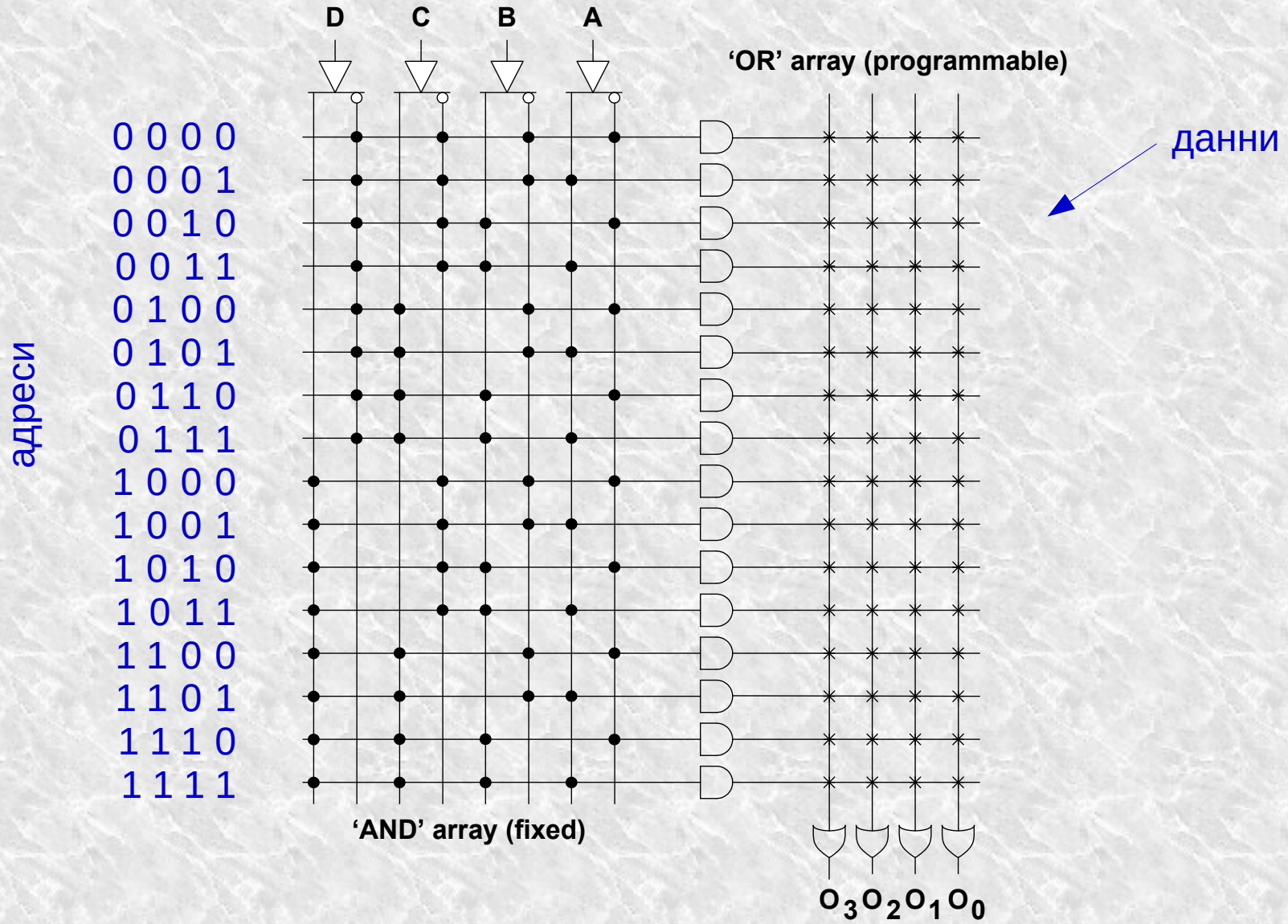
PLA (Programmable Logic Array) – Texas Instruments 1970, Philips

- наследник на PROM
- пълното име е FPLA (Field Programmable Logic Array)
- две отделно програмируеми равнини (AND и OR)
- по-трудно производство, по-големи закъснения на сигналите

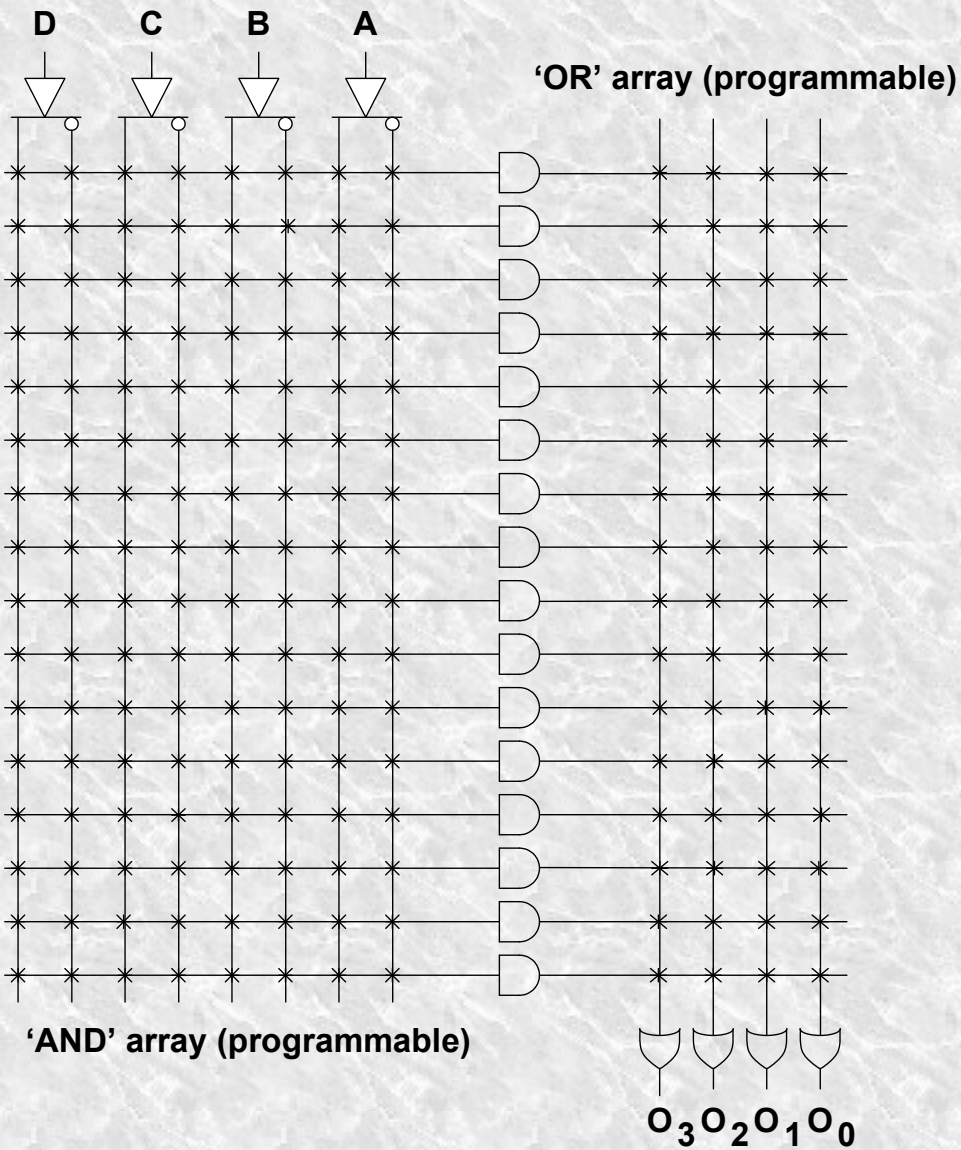
PAL (Programmable Array Logic) – Monolithic Memories Inc. (MMI) 1978, AMD

- програмируема AND равнина и непрограмируема OR равнина

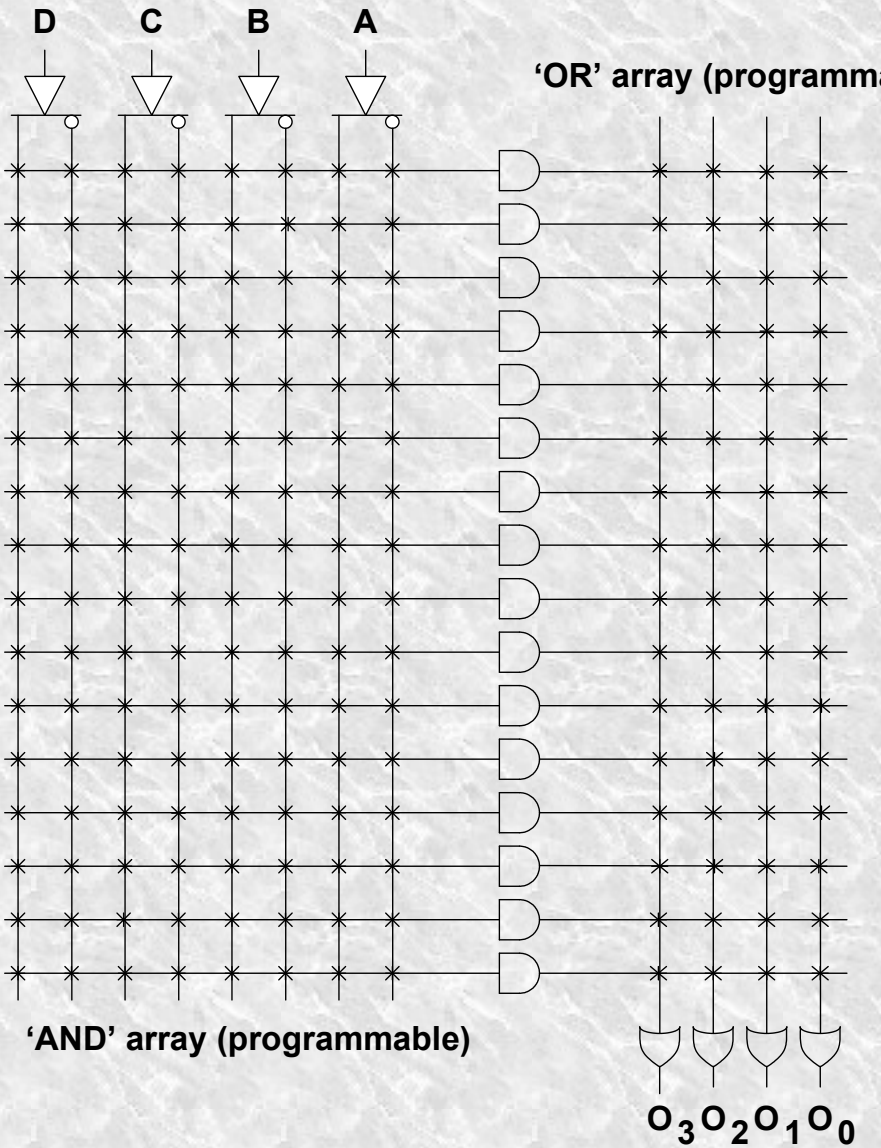
PROM (Programmable Read-Only Memory)



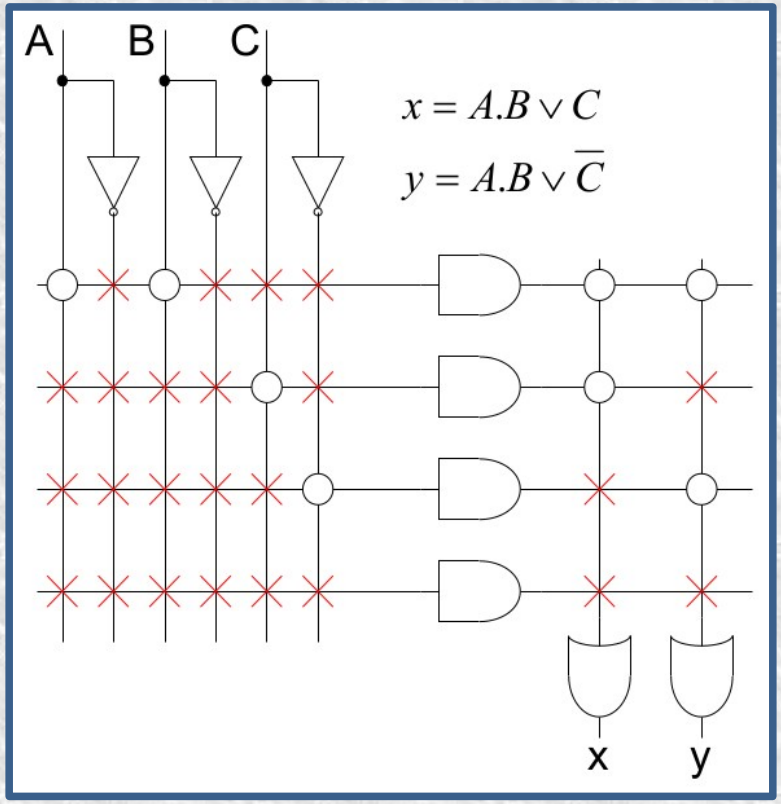
PLA (Programmable Logic Array)

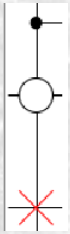


PLA (Programmable Logic Array)



Пример



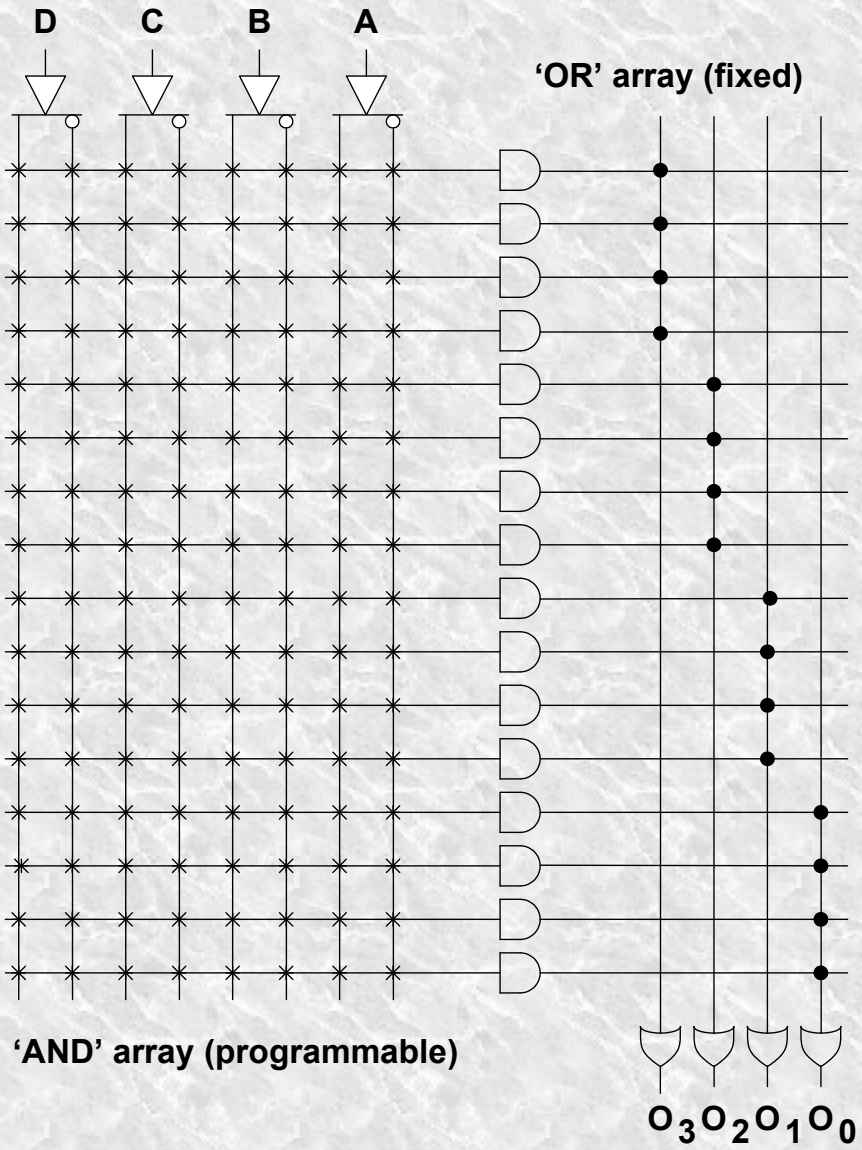


непрограмируеми връзки

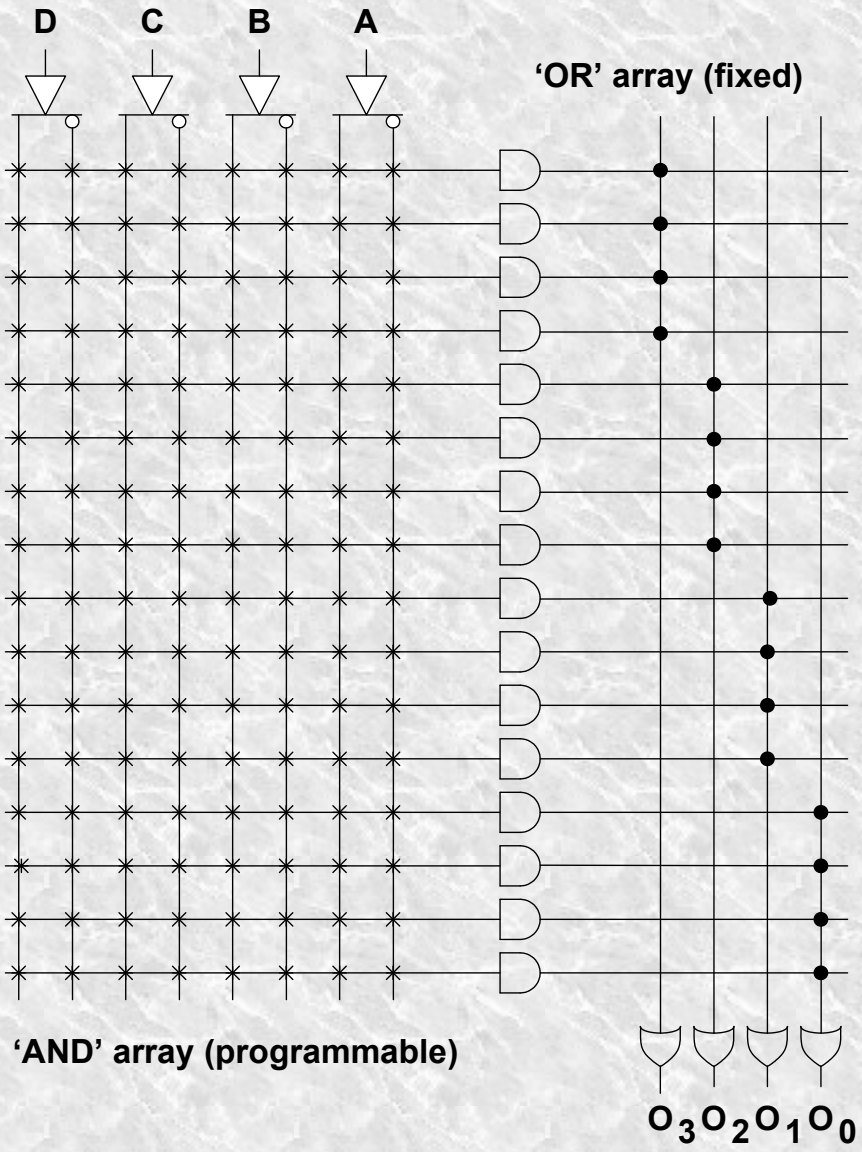
програмирани връзки

непрограмирани връзки

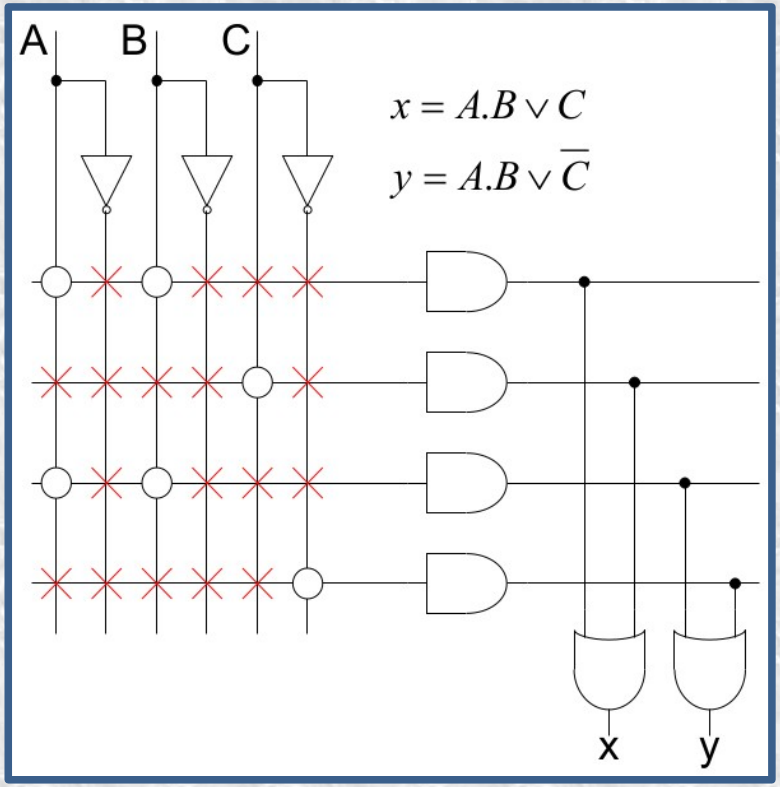
PAL (Programmable Array Logic)

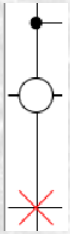


PAL (Programmable Array Logic)



Пример





непрограмируеми връзки

програмирани връзки

непрограмирани връзки

Други програмируеми логически схеми

SPLD (Simple Programmable Logic Device)

- PLA (Programmable Logic Array) и PAL (Programmable Array Logic)

CPLD (Complex Programmable Logic Device)

- включват до 50 SPLD устройства върху един чип
- създадени от Altera под името Classic EPLD
- предназначени да заместят големи масиви от TTL логика (glue logic)
- FLASH базирани (non-volatile)

FPGA (Field Programmable Gate Array)

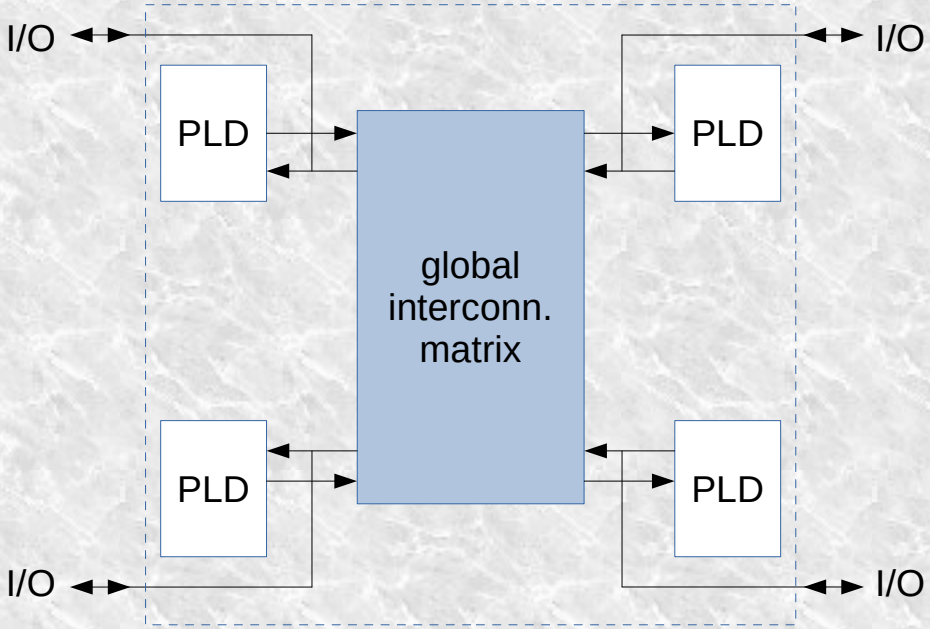
- програмируеми матрични кристали
- повтарящи се полета от малки логически блокове
- огромен логически капацитет
- RAM базирани (volatile)

Основни производители

- Xilinx (през 2020 AMD купува Xilinx за 40 милиарда USD)
- Altera (през 2015 Intel купува Altera за 16.7 милиарда USD)
- Lattice
- Actel (Microsemi)
- Quicklogic
- Cypress

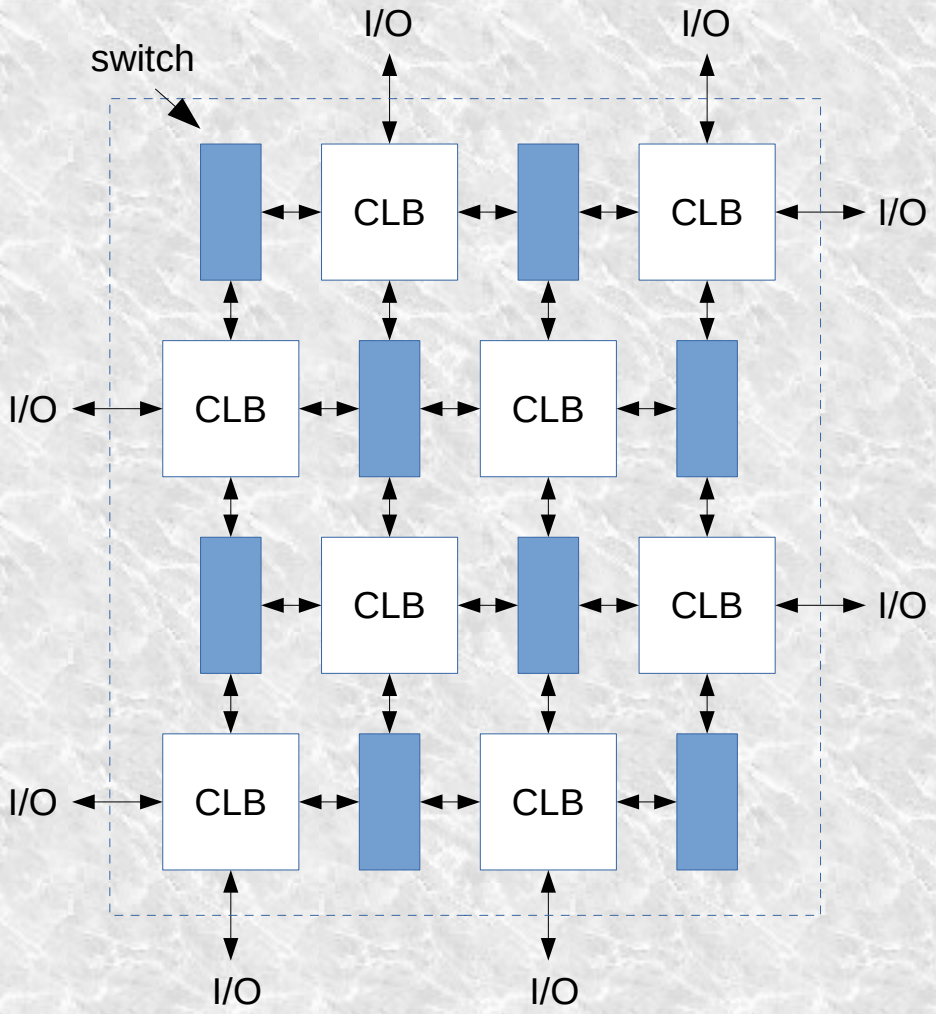
Програмируеми логически схеми

CPLD и FPGA архитектури



PLD – функционален блок PLA или PAL

CLB – конфигурируем логически блок

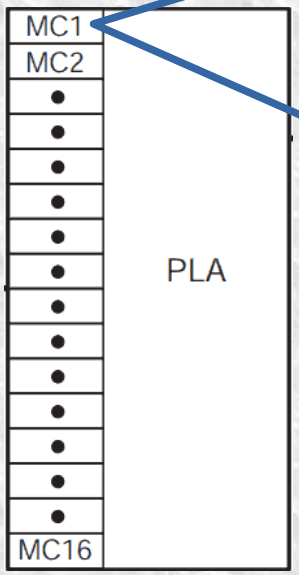


Сложни програмируеми логически устройства
Complex Programmable Logic Device
CPLD

Полеви програмируеми матрици
Field Programmable Gate Array
FPGA

НОВ БЪЛГАРСКИ УНИВЕРСИТЕТ

CPLD – Макроклетка (Macrocell)



Функционален блок (PLD)

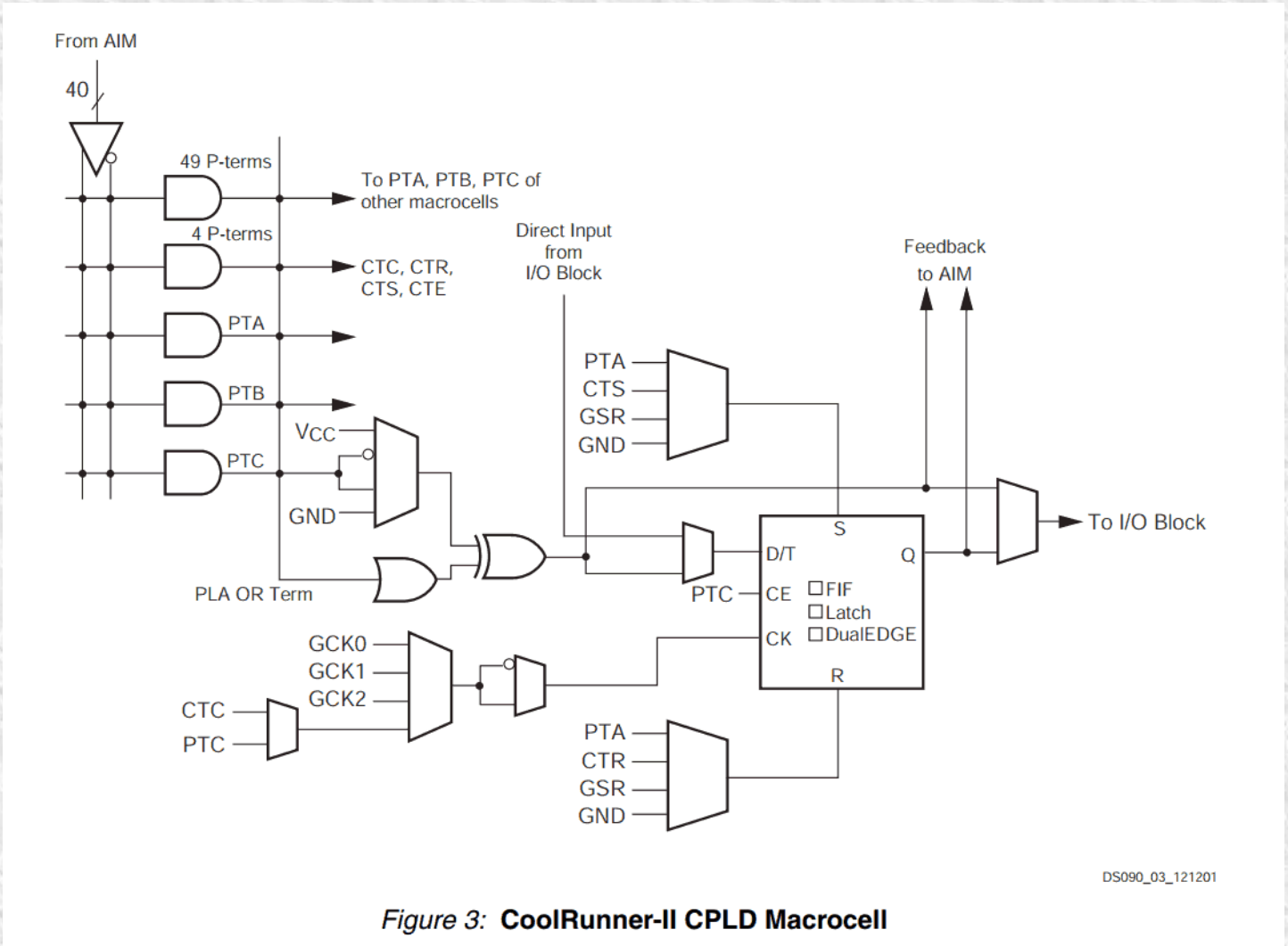
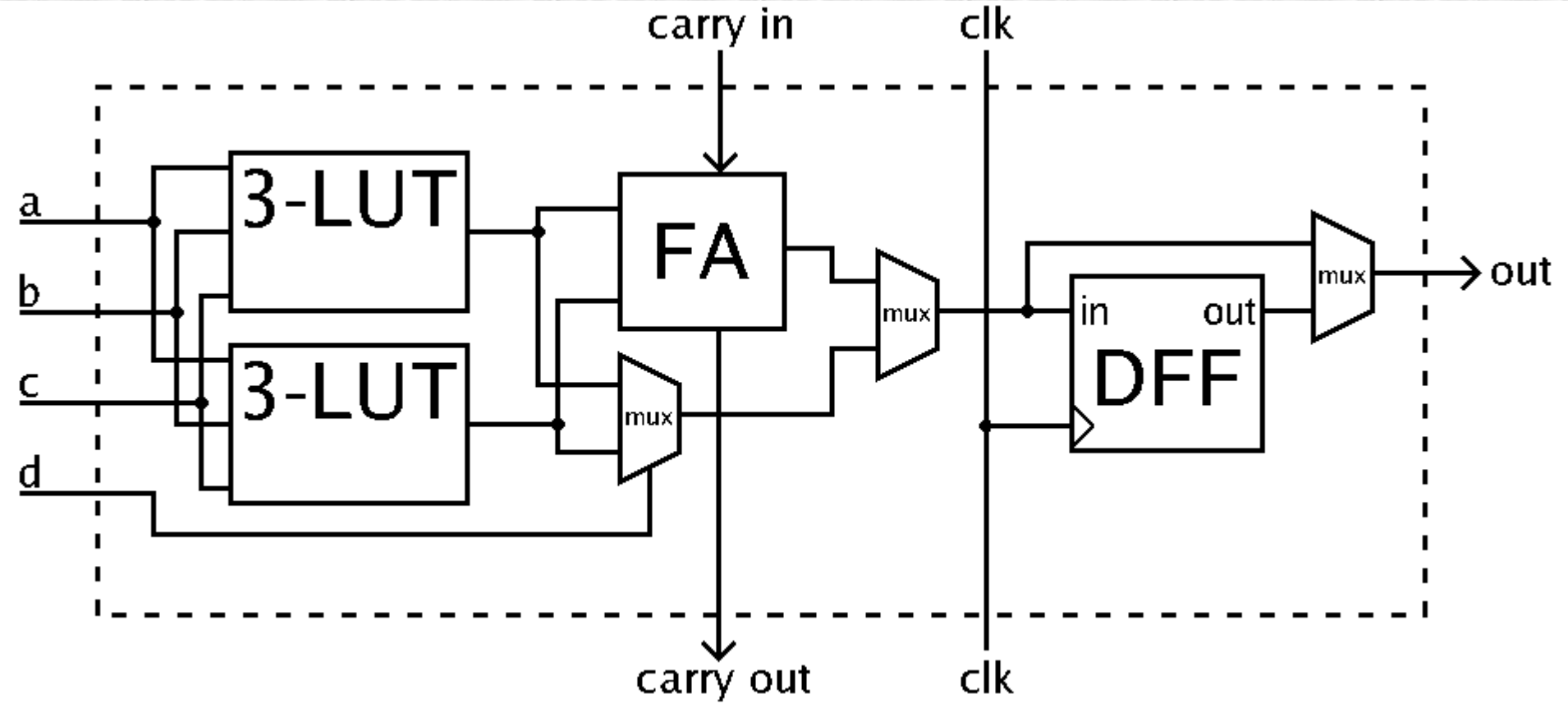


Figure 3: **CoolRunner-II CPLD Macrocell**

FPGA – Конфигурируем логически блок (Configurable Logic Block – CLB)



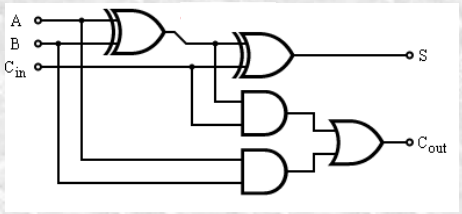
Проектиране с програмируеми логически схеми (design flow)

HDL source code

```
module
full_adder(i_bit1,i_bit2,i_carry,o_sum,o_carry);
  input  i_bit1, i_bit2;
  input  i_carry;
  output o_sum, o_carry;

  wire   w_WIRE_1, w_WIRE_2, w_WIRE_3;

  assign w_WIRE_1 = i_bit1 ^ i_bit2;
  assign w_WIRE_2 = w_WIRE_1 & i_carry;
  assign w_WIRE_3 = i_bit1 & i_bit2;

  assign o_sum   = w_WIRE_1 ^ i_carry;
  assign o_carry = w_WIRE_2 | w_WIRE_3;
endmodule
```

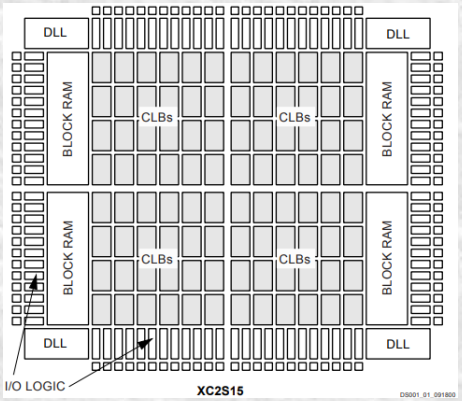
synthesize

netlist



map, place & route

FPGA



generate bitstream

bitstream

```
01010101010101010010101010101010010100101010
10101010101001010101010101001010010101111100
01010101010101010010101010101010010100101010
10101010101001010101010101001010010101111100
01010101010101010010101010101010010100101010
10101010101001010101010101001010010101111100
01010101010101010010101010101010010100101010
10101010101001010101010101001010010101111100
01010101010101010010101010101010010100101010
10101010101001010101010101001010010101111100
01010101010101010010101010101010010100101010
10101010101001010101010101001010010101111100
```

download and test