Fundusze Europejskie
Wiedza Edukacja Rozwój

Rzeczpospolita Polska

Unia Europejska
Europejski Fundusz Społeczny

Politechnika Śląska jako Centrum Nowoczesnego Kształcenia
opartego o badania i innowacje

POWR.03.05.00-IP.08-00-PZ1/17

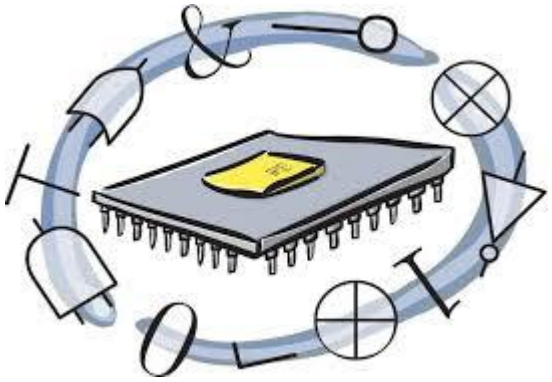
Projekt współfinansowany przez Unię Europejską ze środków Europejskiego Funduszu Społecznego

Digital Circuits Design

Faculty of Automatic Control, Electronics and Computer Science, Informatics, Bachelor Degree

Lecture 9.

Programmable Logic Devices



Ph.D. Eng. **Adam Opara**

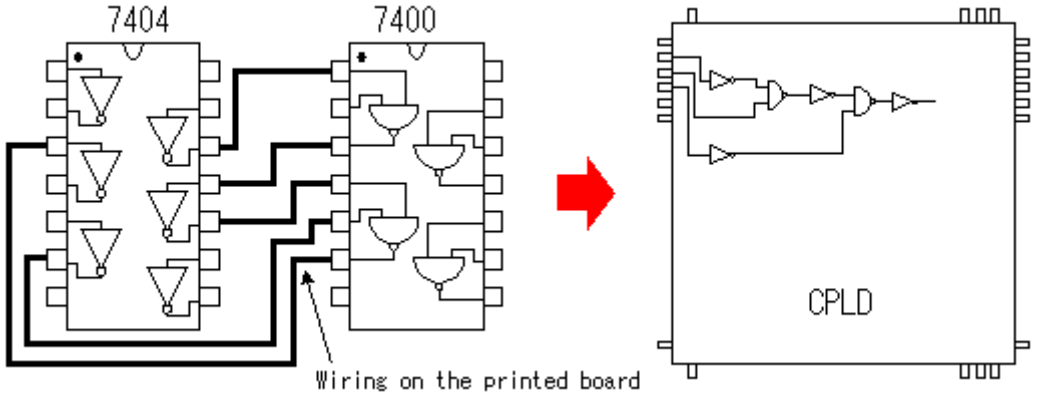
Logic Devices

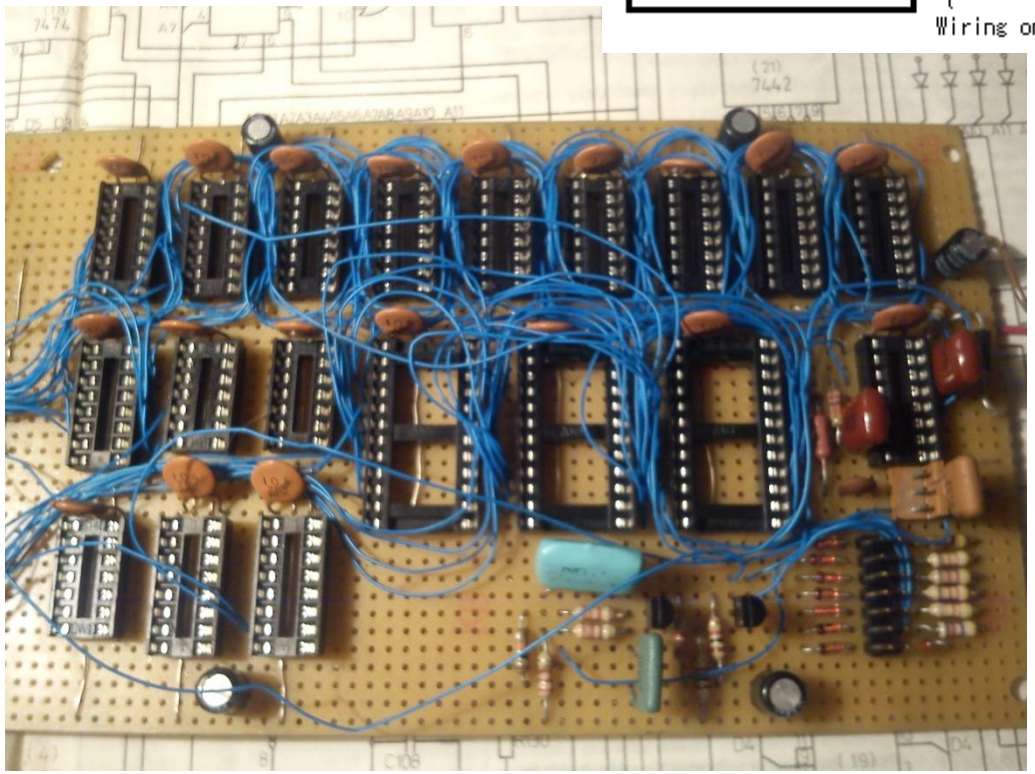
Agenda:

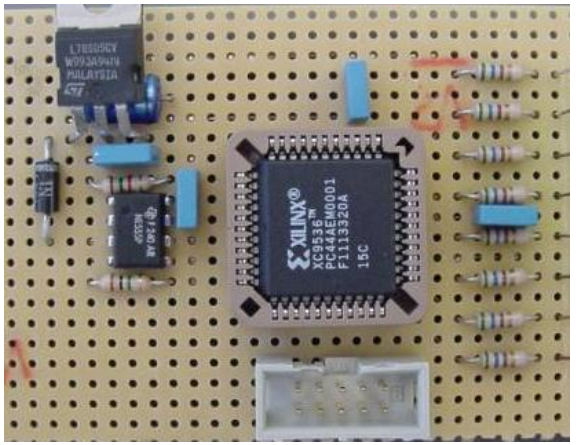
- Programmable Logic Devices (PLD) programmed by user:
  - Simple PLD: PAL, PLE, PLA
  - CPLD (Complex PLD)
  - FPGA (Field Programmable Gate Array)
- Application Specific Integrated Circuit (ASIC)
  - semi custom – designed by user (Gate Array, Std. Cell)
  - full custom – ordered by user

Programmable Logic Devices

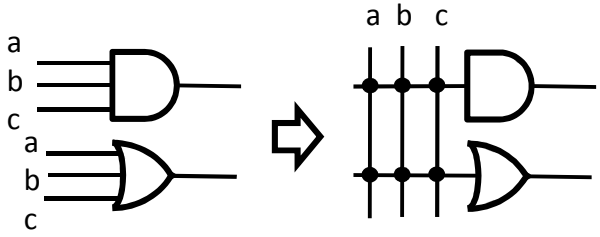
Why use PLD?

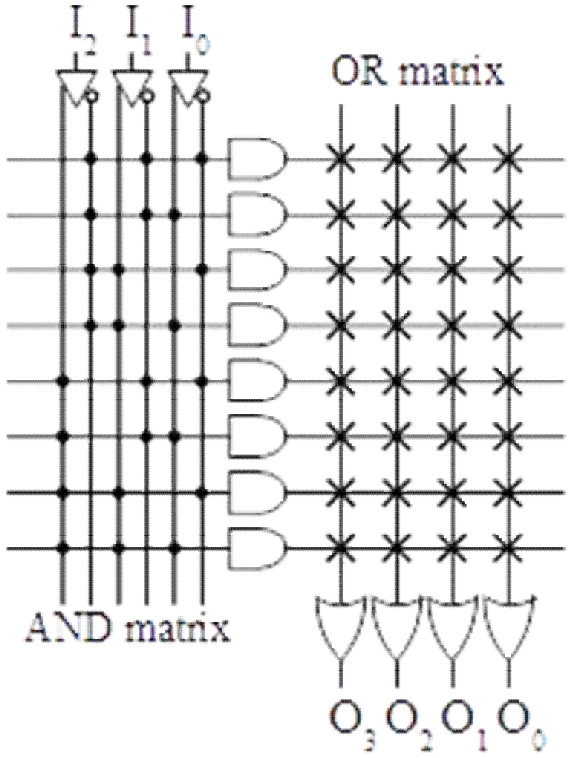






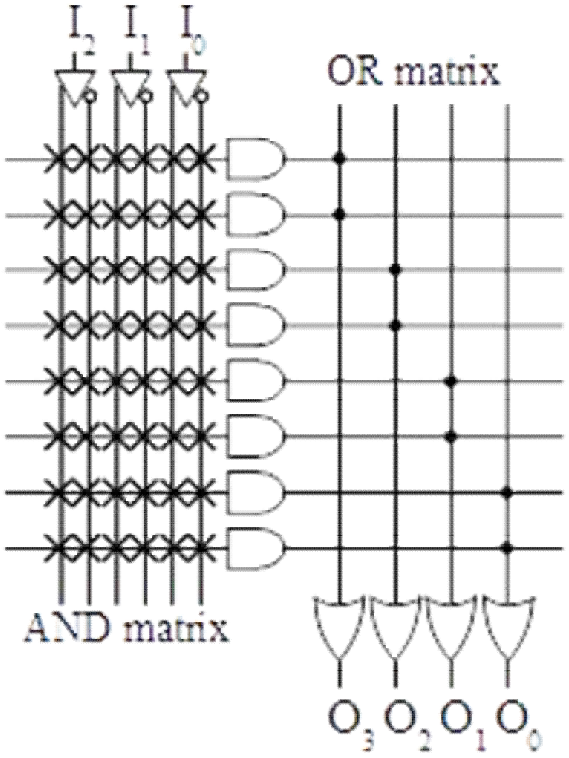
Simple PLD





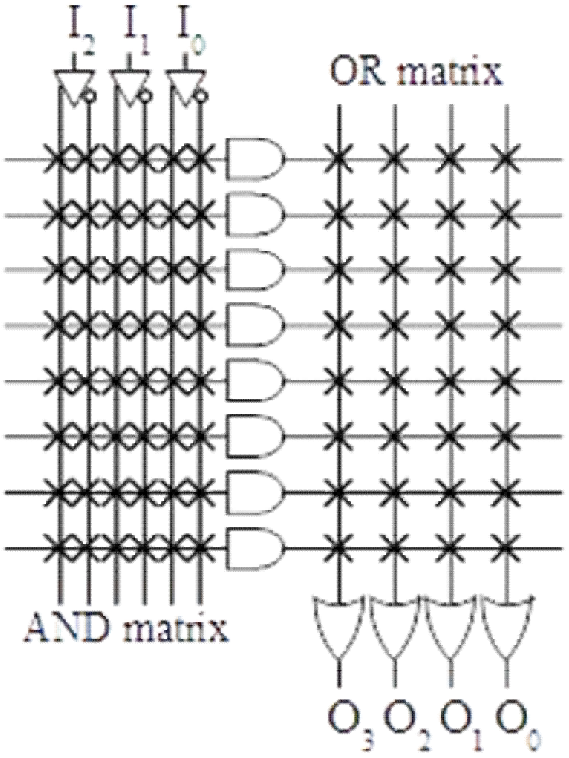
PLE (PROM)

Programmable OR matrix



PAL

Programmable AND matrix

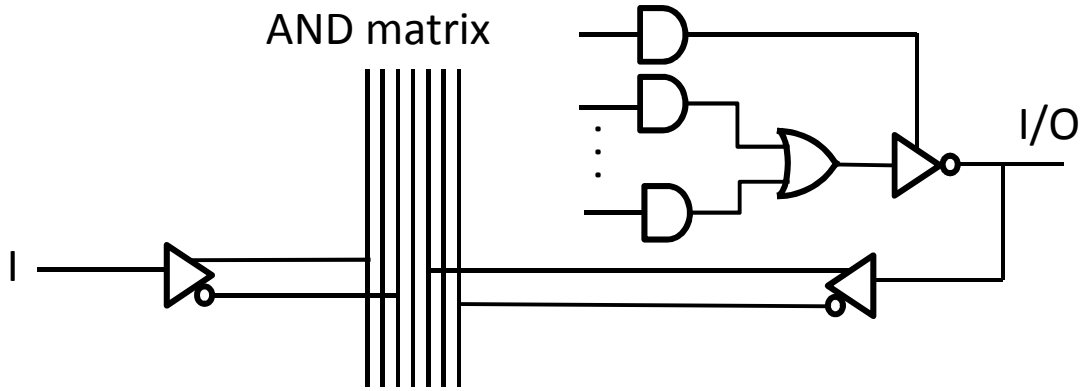


PLA

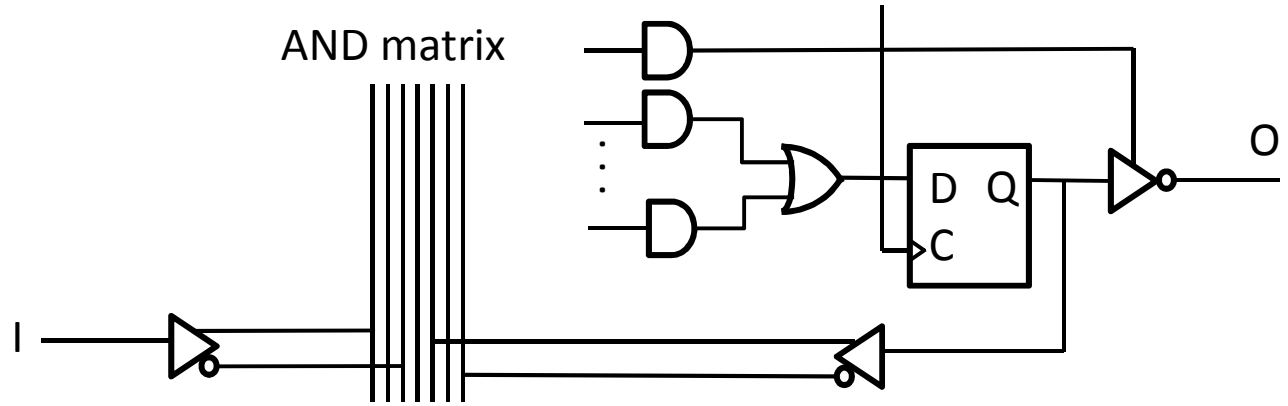
Programmable
OR and AND matrix

Simple PLD

Combinatorial circuit output

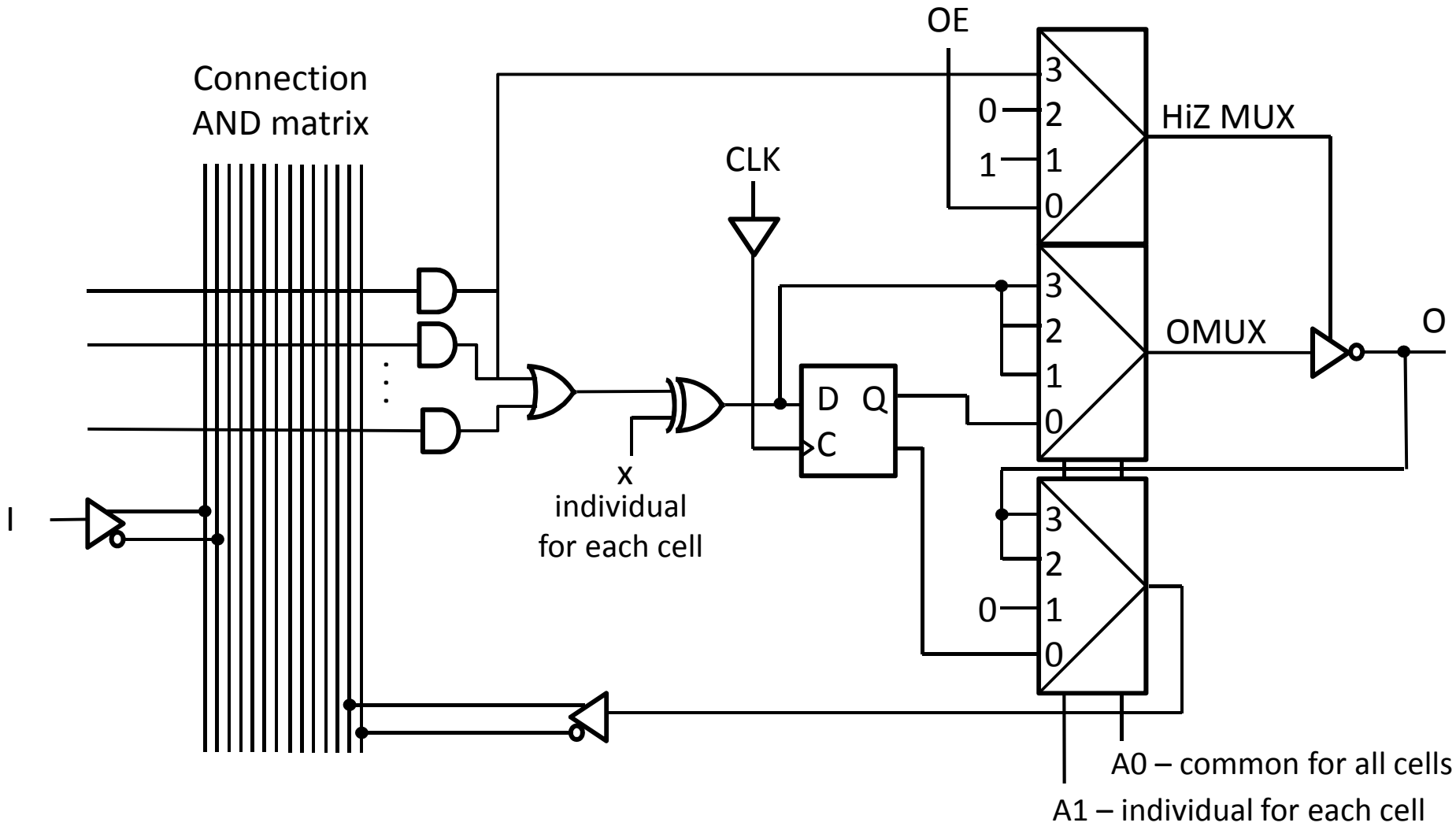


Sequential circuit output



Simple PLD

An example of GAL Output Logic Macrocell (OLMC)



Simple PLD

An example of Simple PLD naming:

PAL 16L8 – max **16** in, max **8** out,

but *not* 16 *and* 8 !, 10 in, 2 out, 6 in/out

L - out negated (Low)

programmed output – **L**ow, **H**igh, **R**egister, **V**ariable

One **PAL16V8** supersedes PAL 16L8, 16H8, 16R8,

-> 8 outputs (functions)

-> sequential circuit with 256 states

Simple PLD

An example of functions implementation:

a	b	c	W	X	Y	Z
0	0	0	1	0	1	0
0	0	1	1	0	1	0
0	1	0	0	0	1	1
0	1	1	0	1	0	1
1	0	0	1	1	0	1
1	0	1	0	0	0	0
1	1	0	1	1	1	1
1	1	1	0	1	0	1

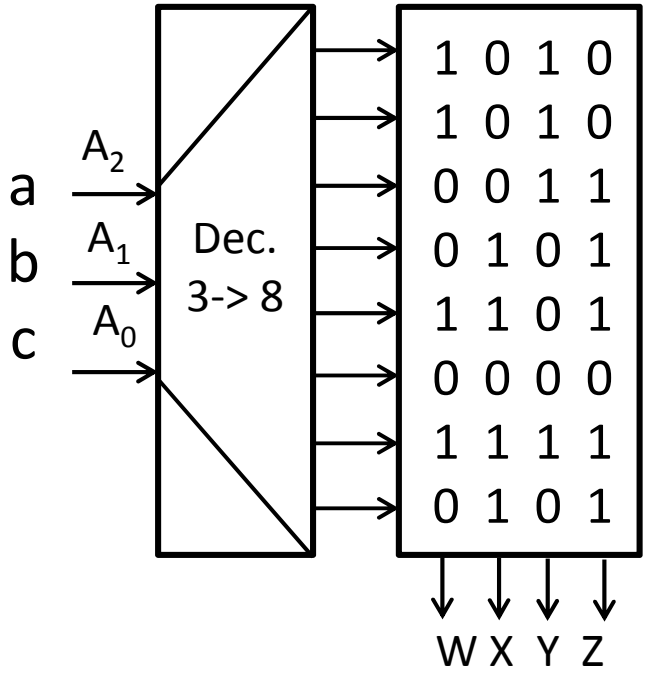
Simple PLD

1. PLE (PROM) implementation

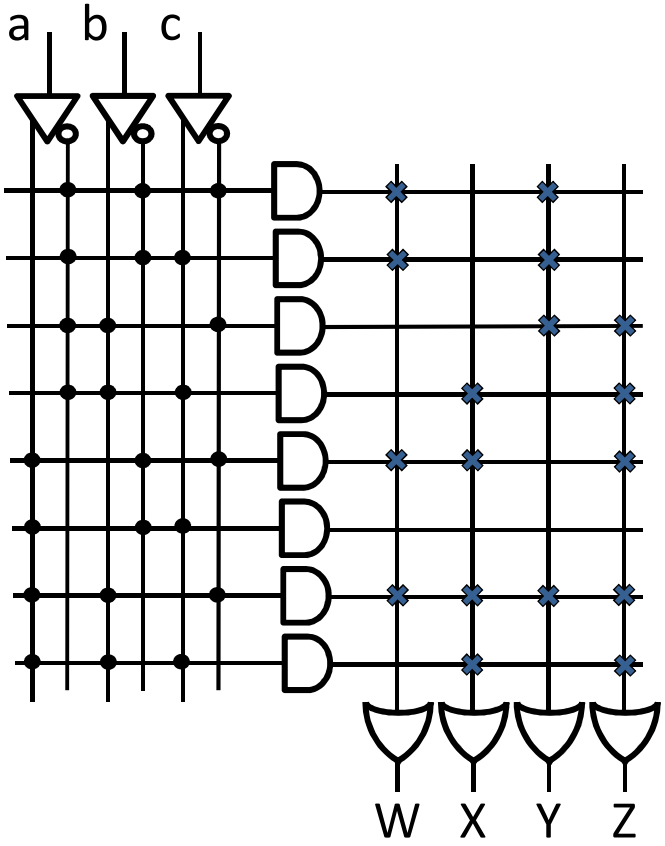
abc -> address,

WXYZ -> val. of memory cell

a	b	c	W	X	Y	Z
0	0	0	1	0	1	0
0	0	1	1	0	1	0
0	1	0	0	0	1	1
0	1	1	0	1	0	1
1	0	0	1	1	0	1
1	0	1	0	0	0	0
1	1	0	1	1	1	1
1	1	1	0	1	0	1



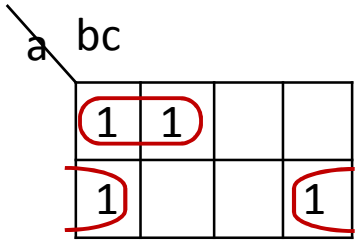
PLE Logic schematic.
AND-const., OR-prog.



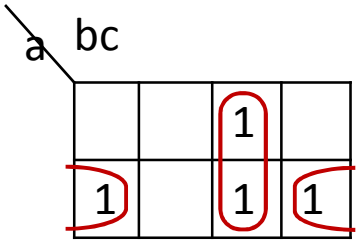
Simple PLD

2. PLA implementation

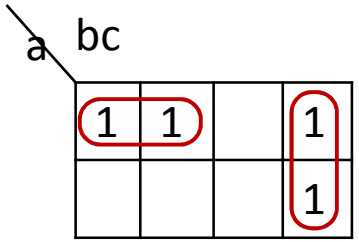
WXYZ -> function minimization/factorisation



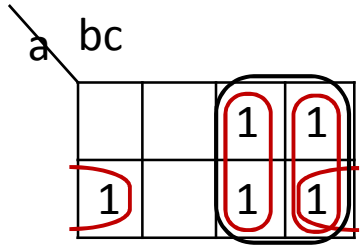
W=$\bar{a}\bar{b} + a\bar{c}$



X=$bc + a\bar{c}$



Y=$\bar{a}\bar{b} + b\bar{c}$



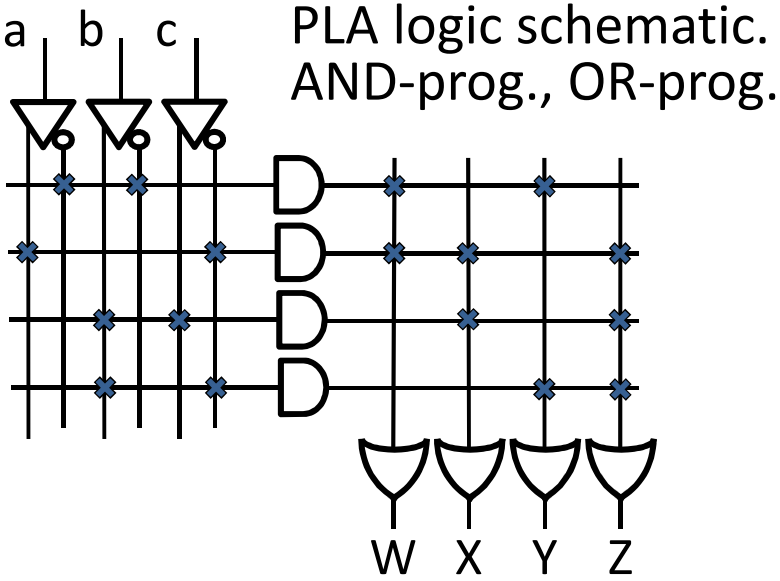
Z=$a\bar{c} + b$=
$a\bar{c} + bc + b\bar{c}$

shared/common terms

Term	abc	WXYZ
$\bar{a}\bar{b}$	00-	1010
$a\bar{c}$	1-0	1101
bc	-11	0101
$b\bar{c}$	-10	0011

AND matrix

OR matrix



PLA logic schematic.
AND-prog., OR-prog.

Simple PLD

3. PAL implementation

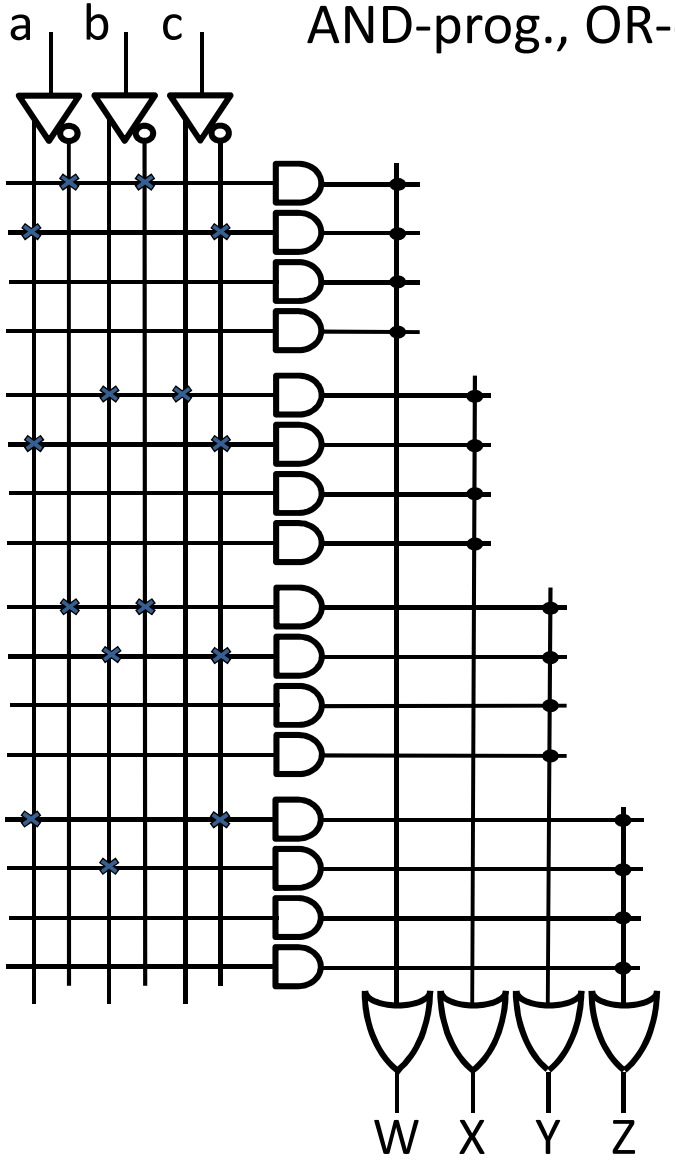
WXYZ -> function minimization /factorisation

Term	abc	WXYZ
$\bar{a}\bar{b}$	00-	1010
$a\bar{c}$	1-0	1101
bc	-11	0101
$b\bar{c}$	-10	0011

AND matrix

OR matrix

PAL logic schematic.
AND-prog., OR-const.



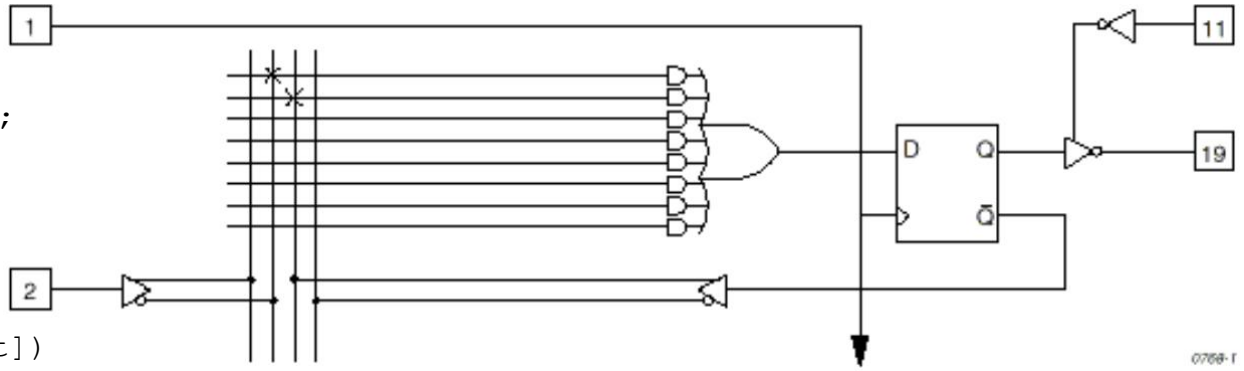
Simple PLD

Hardware Description Language (HDL) - ABEL

```
module simple_pin2pin
Clk       pin 1;
Toggle    pin 2;
Ena       pin 11;
Qout      pin 19 istype 'reg';

equations
     Qout := !Qout.FB & Toggle;
     Qout.CLK = Clk;
     Qout.OE = !Ena;

test_vectors
     ([Clk,Ena,Toggle] -> [Qout])
     [.c., 0 , 0 ] -> 0;
     [.c., 0 , 1 ] -> 1;
     [.c., 0 , 1 ] -> 0;
     [.c., 0 , 1 ] -> 1;
     [.c., 0 , 1 ] -> 0;
     [.c., 1 , 1 ] -> .Z.;
     [ 0 , 0 , 1 ] -> 1;
     [.c., 1 , 1 ] -> .Z.;
     [ 0 , 0 , 1 ] -> 0;
end
```



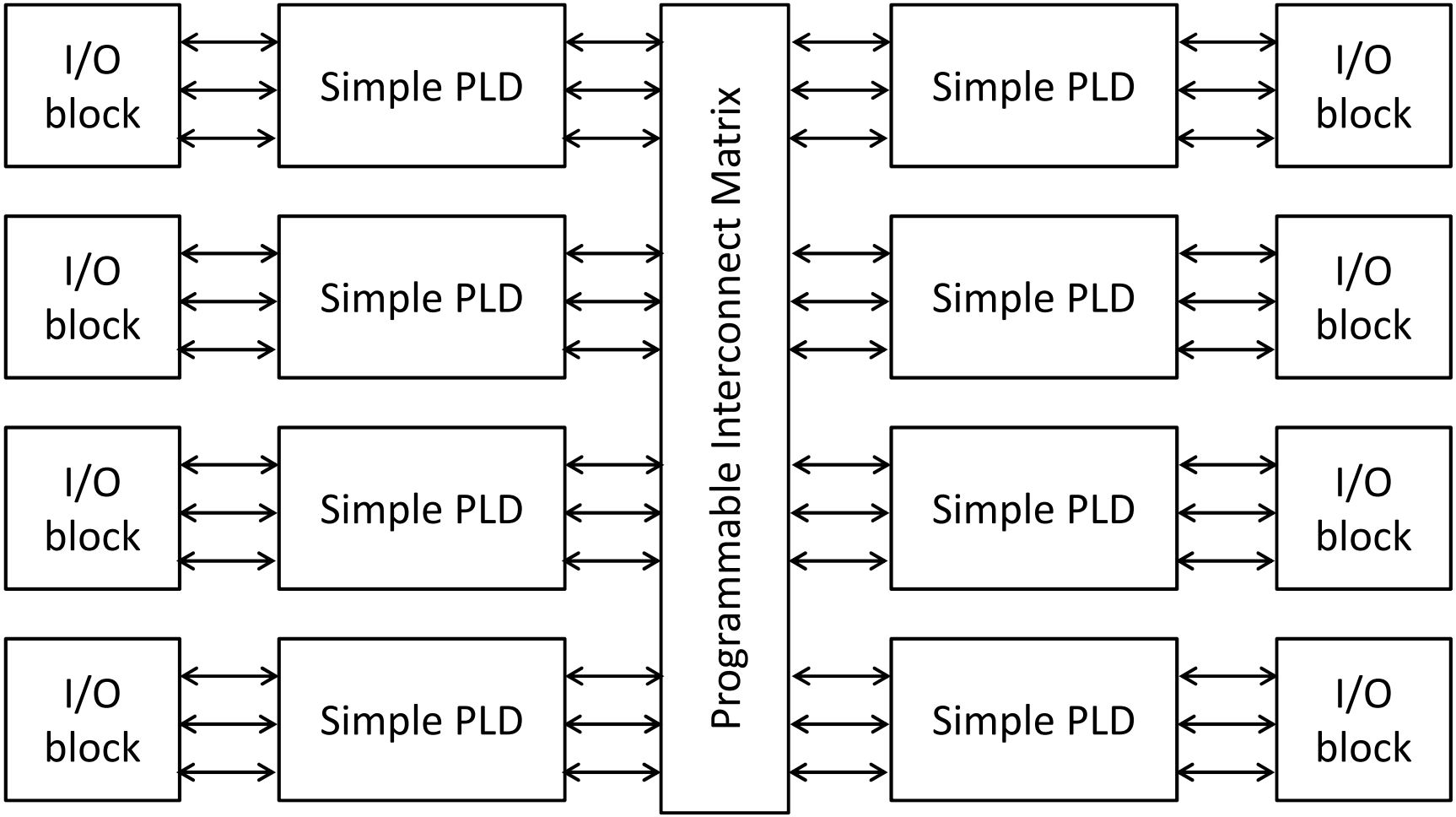
CPLD

Complex Programmable Logic Devices

(different architecture than PLD)

- Functional Blocks (Macrocell)
  small simple PLD
- Programmable Interconnect Matrix (PIM)
  - Full connection matrix
    mesh structure, const. propagation time (~7..15ns)
  - Could have multiplexers/expanders
    n cascaded MUXs, var. propagation time *n* x(~0.2…2ns)
- I/O blocks
  3 state output, different input level standards
- Often in-system programmable
  (EEPROM technology)

CPLD



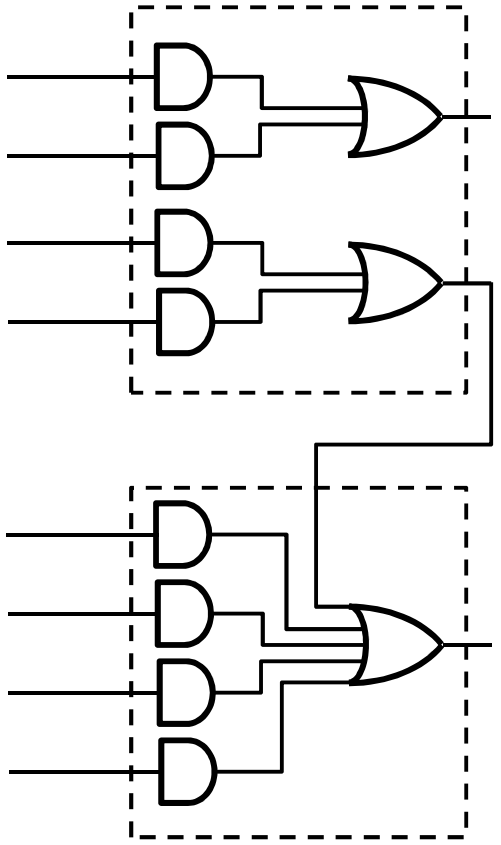
Complex PLD

CPLD

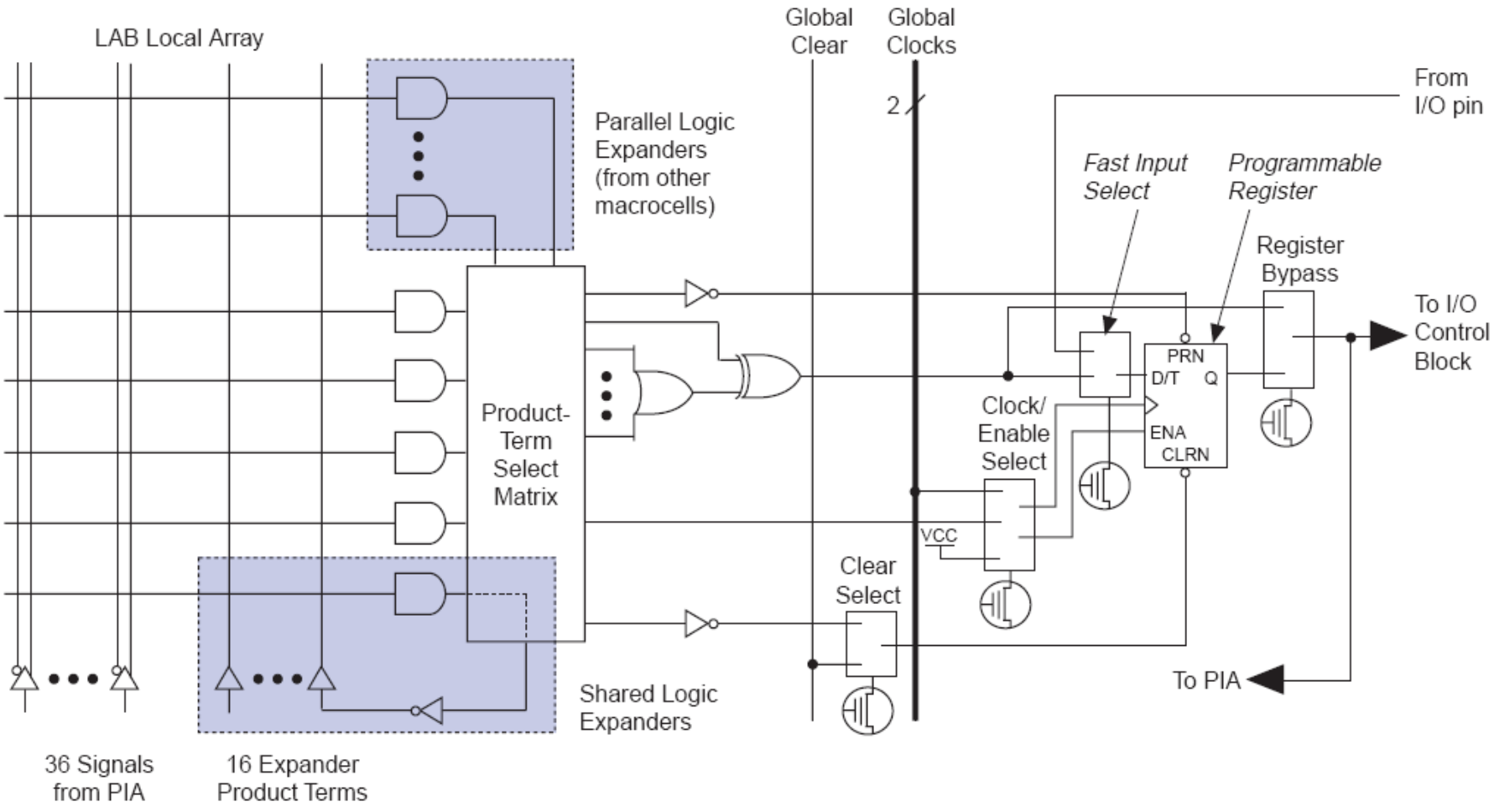
Functional Blocks (Macrocells)

includes a “term allocator”

to expand number of terms (borrow ones)

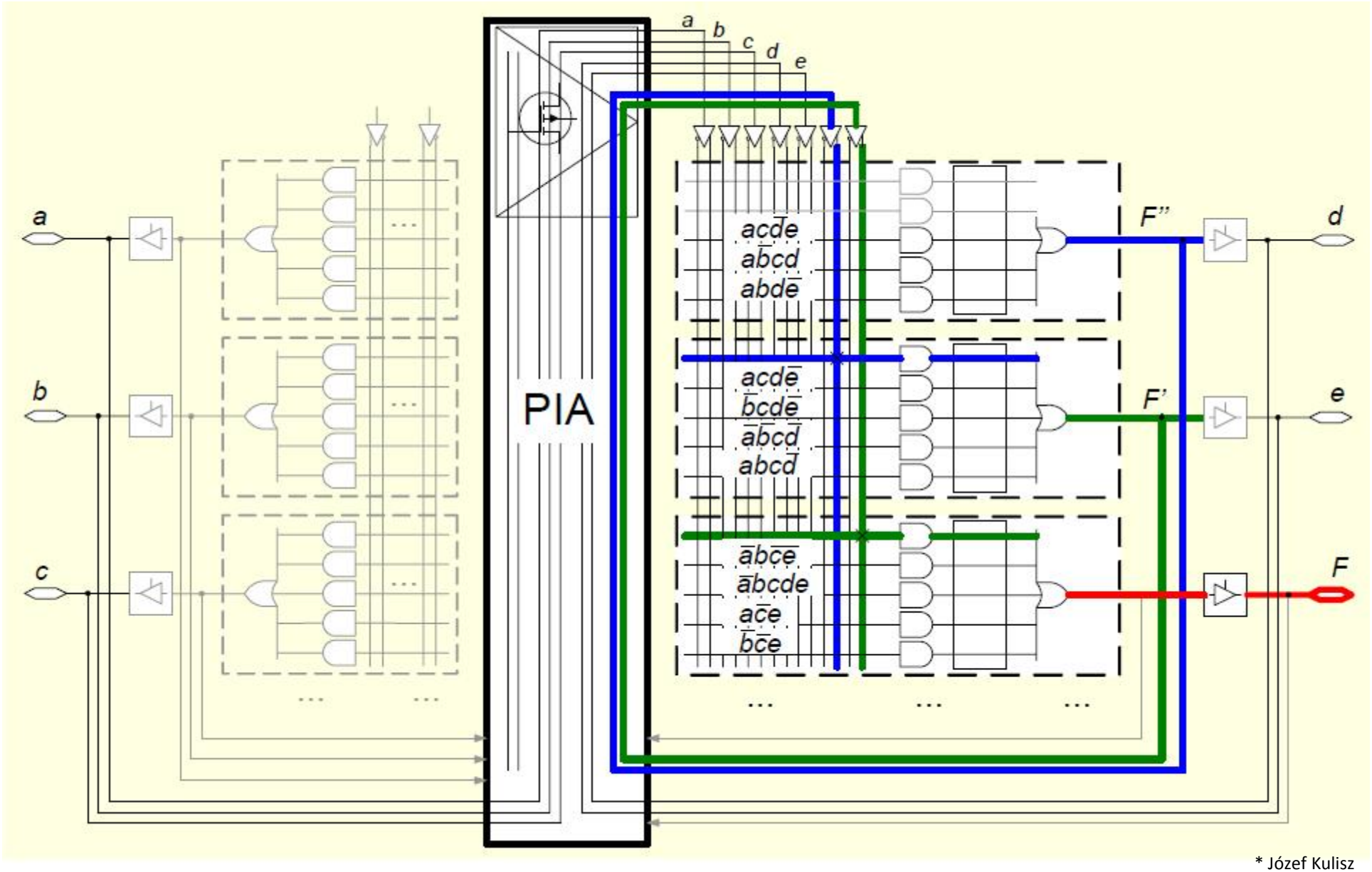


CPLD



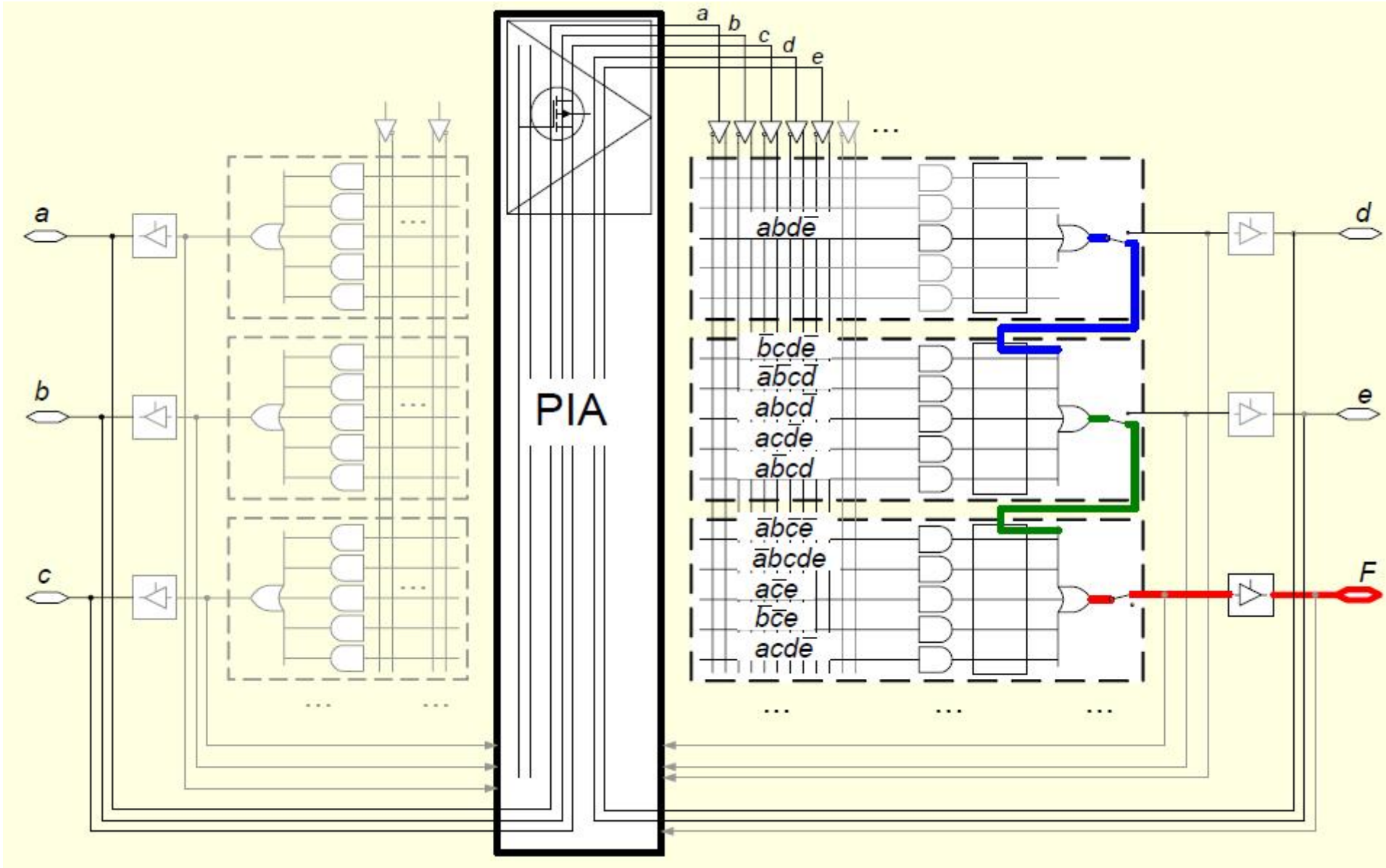
PIA – Programmable Interconnect Array

CPLD



* Józef Kulisz

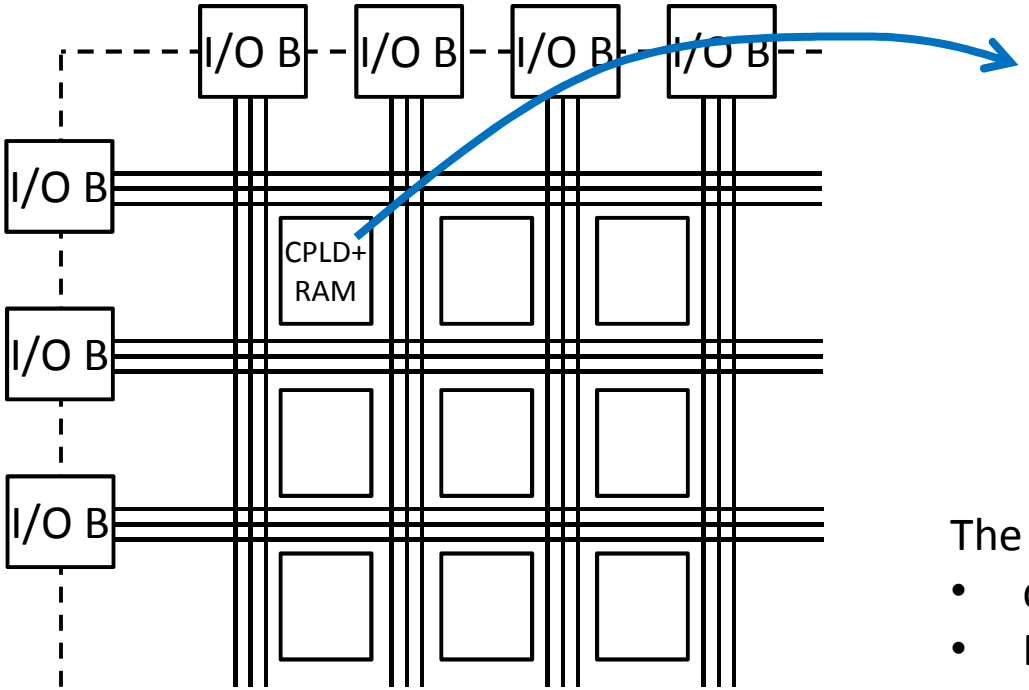
CPLD

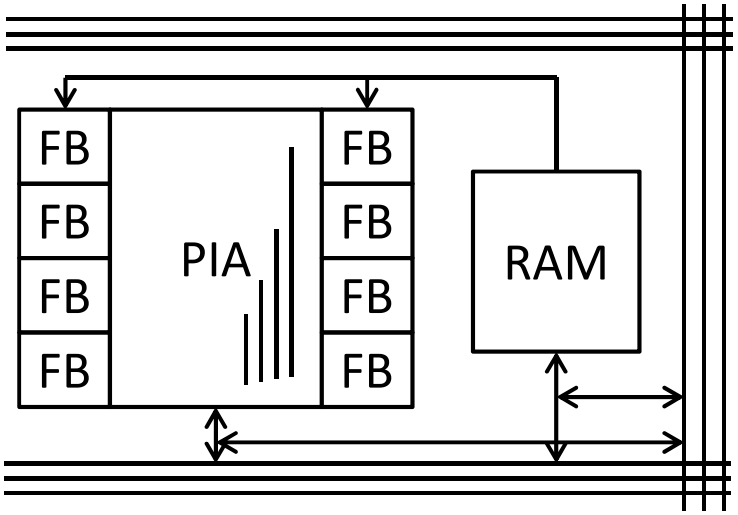


* Józef Kulisz

More complex devices:

- A regular matrix of logical blocks connected by vertical and horizontal channels
- Channels crossing includes SRAM/ROM memory





The memory:

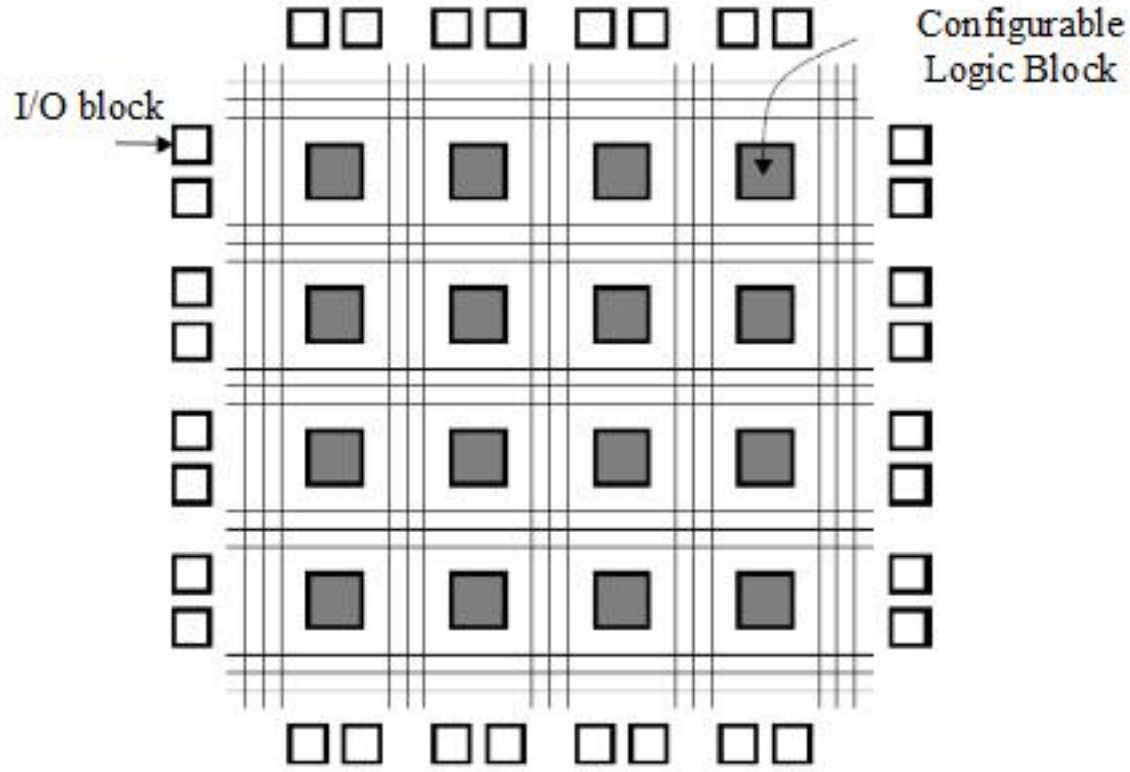
- defines connections
- RAM needs to be loaded after power on
- some has autoinitialization from buildin Flash

FPGA

Even more and more complex devices ->

Field Programmable Gate Array

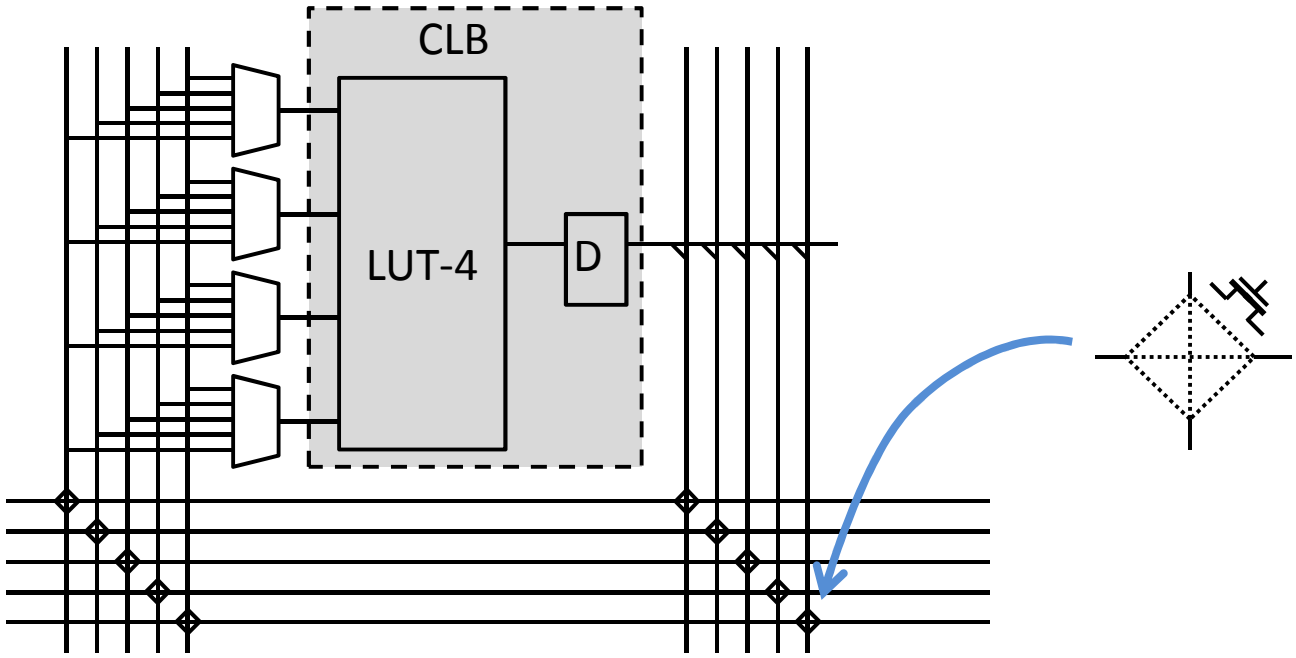
- Large number of very simple logical blocks (CLB - Configurable Logic Blocks)
- Many small connection matrixes



FPGA

CLB - Configurable Logic Block

- Look Up Table (LUT) small RAM implementing any 4 input logic function



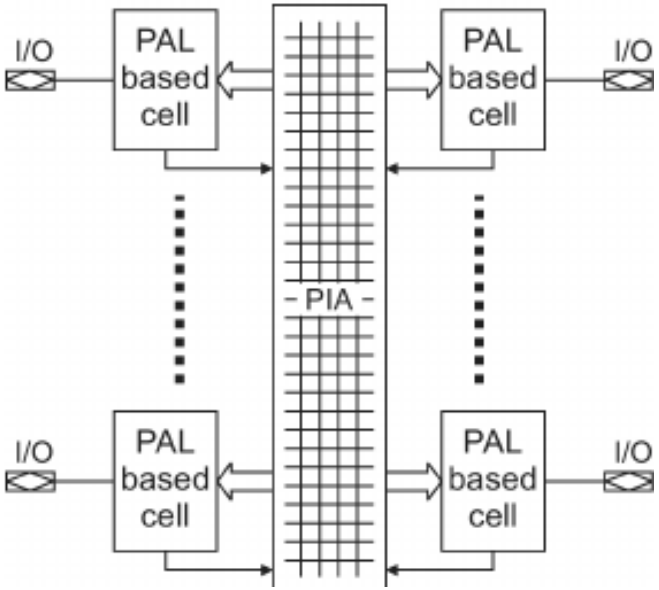
CPLD vs FPGA

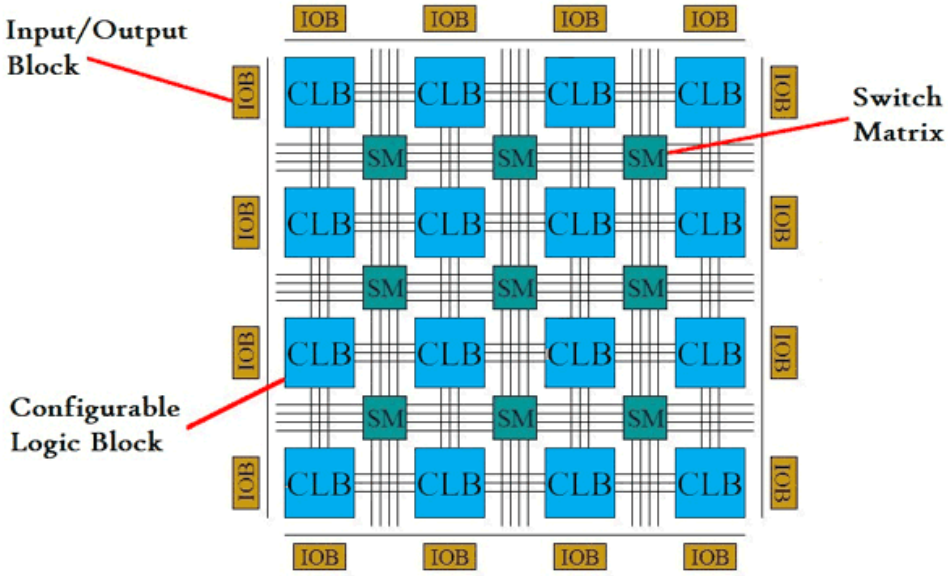
CPLD

- Logic Block - PAL
- One full connection matrix (const. propagation time)
- 100% connectivity
- EEPROM

FPGA

- Logic Block - LUT
- Many connection matrixes (var. propagation time)
- <100% connectivity
- RAM





ASIC

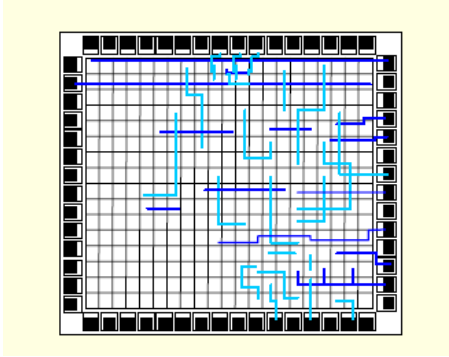
Application Specific Integrated Circuit:

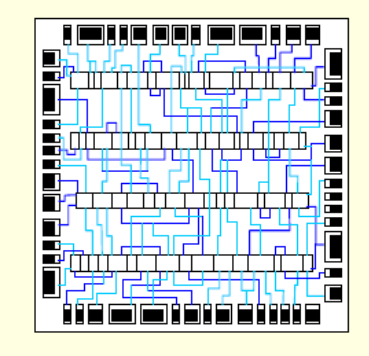
- Individual application for specific use
- Design and implementation cost – large
  -> profitable only in high volume production

ASIC

Types:

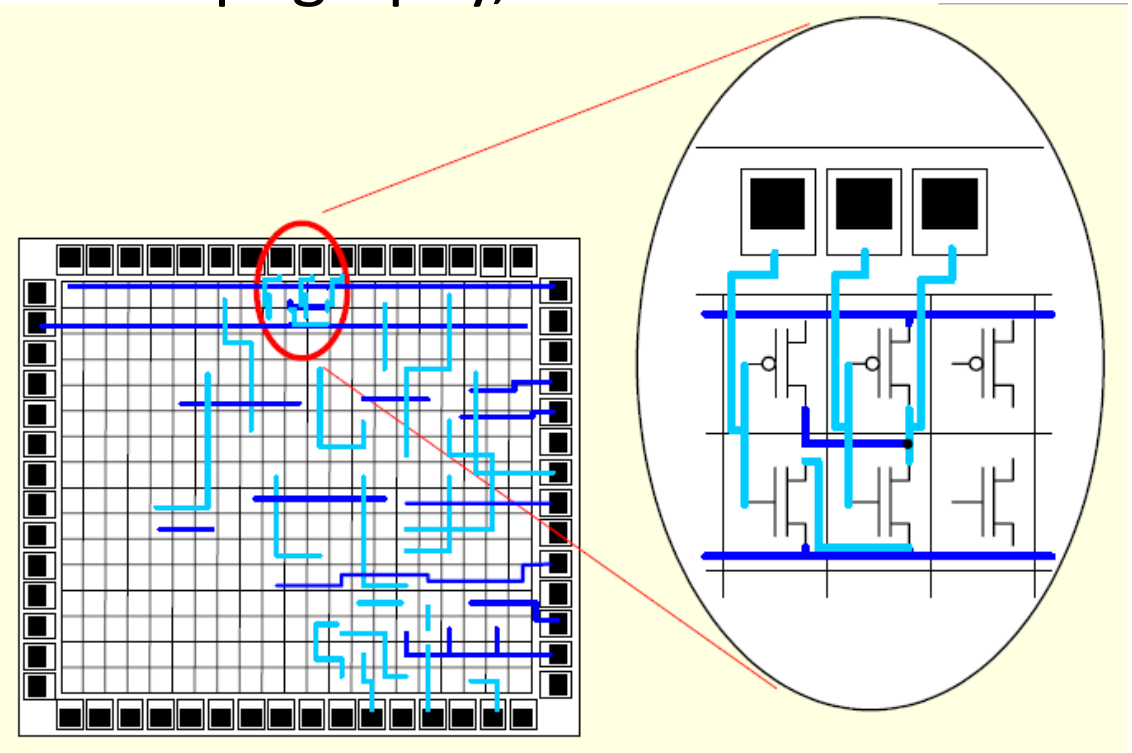
- Gate Array (GA)
- Standard Cell (SC), Embedded Array (EA)
- Full Custom (FC), Cell Based (CB)





ASIC

- Gate Array (GA) CMOS, BiCMOS, ECL technology.
  Two stage design:
  - Prefabricated silicon substrate with not connected elements (transistors, resistors)
  - Design of connection topography, creating masks



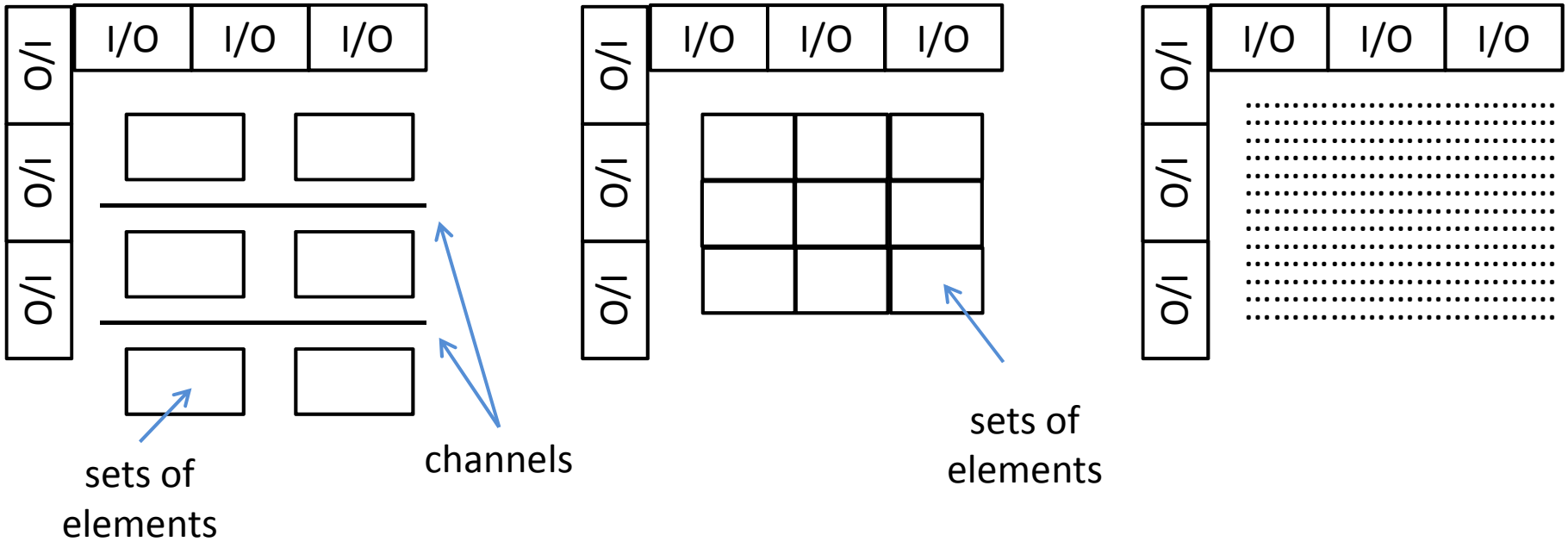
ASIC

- Gate Array (GA) - 3 structures

Channeled

Unchanneled sea of gates

Sea of transistors

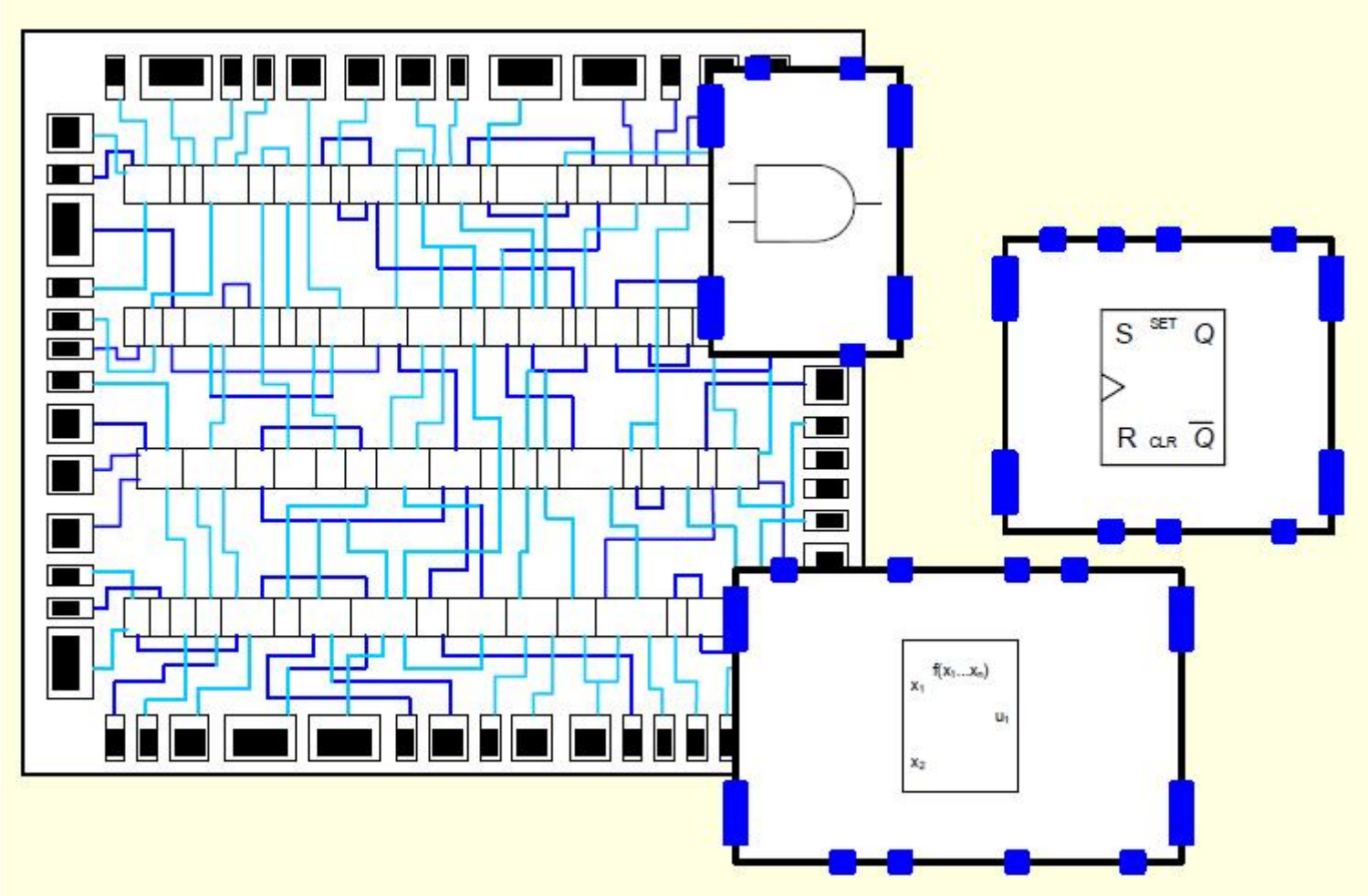


ASIC

- Standard Cell (SC)
  - Project based on a clean silicon surface
  - Tested ready to use libraries of elements
  - Cells placed in rows between connection channels
  - Cell height = const., width = var.

ASIC

- Standard Cell (SC)



* Józef Kulisz

ASIC

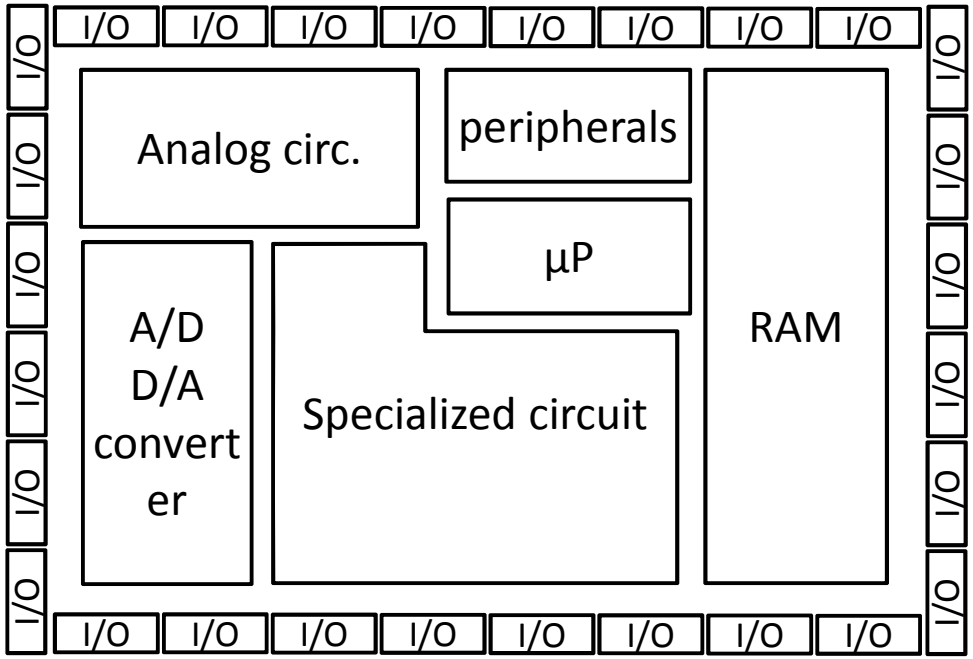
- Embedded Array (EA)
  - Two areas Standard Cell and Gate Array in one chip
    - SC – functional blocks e.g. µP
    - GA – custom logic
  - Only GA part designed

ASIC

- Full Custom (FC)
  - Typical VLSI IC
  - Expensive and long-term design process
  - Difficult design of connections
  - High optimization possibility (speed)

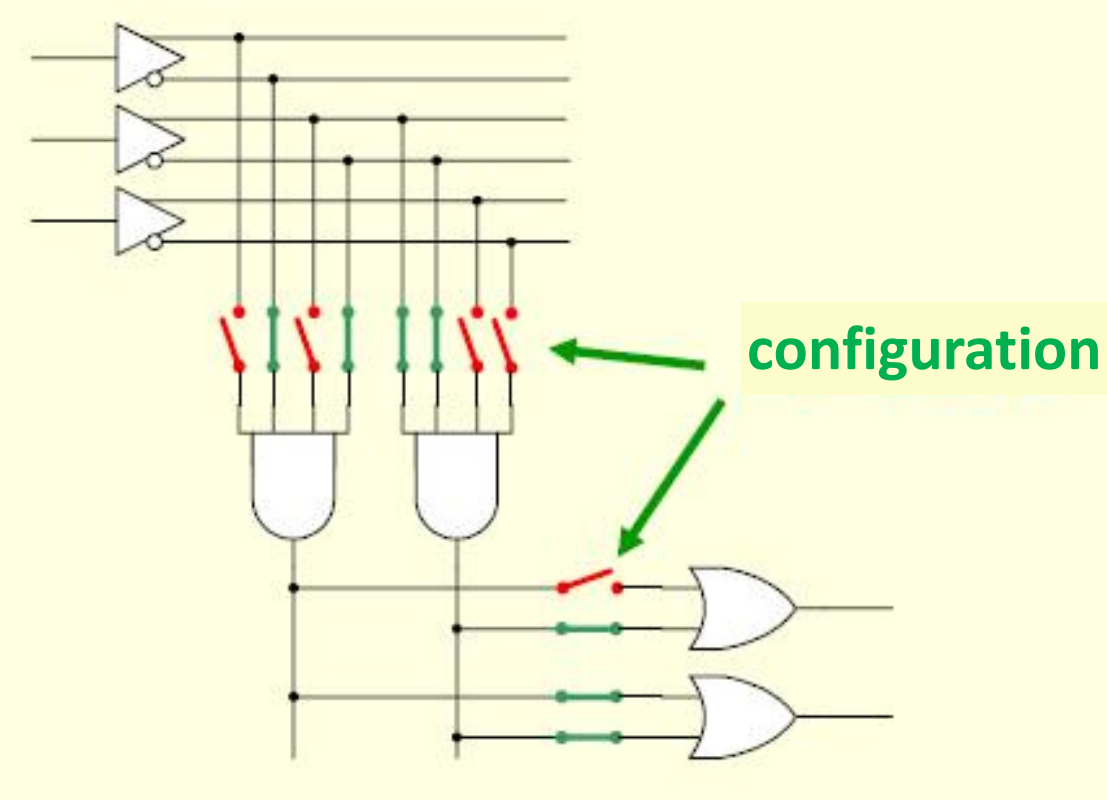
ASIC

- Cell Based (CB) (variant of Full Custom)
  - Cell libraries
  - Faster and cheaper design process



ASIC

- PLD is also ASIC



* Józef Kulisz