Programmable Logic Devices

Ayoush Johari
Assistant Professor
Department of Electronics and Communication Engineering
Lakshmi Narain College of Technology and Science, Bhopal

- Text Book:
  - Rapid prototyping of digital systems – SOPC edition by James O. Hamblen, Tyson S. Hall and Michael D. Furman
- Reference Books:
  - Xilinx and altera logic design handbooks

Implementation Options for Digital Logic

- Assembly of SSI and MSI parts on PC boards.
  - Mostly obsolete; still useful when just a few parts needed
- Programmable Logic Devices (PLD)
  - Variety of types, with different size and performance characteristics; largest have over 10^6 gate "equivalents"
  - CAD tools enable simulation and automate device programming

Implementation Options for Digital Logic (Cont.)

- Application Specific Integrated Circuits (ASIC)
  - design methods similar to PLDs
    - HDLs and simulation with synthesis using standard cell library
    - plus physical design - placement of logic components and routing
  - can augment with custom design of critical components
  - higher performance, greater logic density
  - custom IC fabrication -- suitable for high production volumes

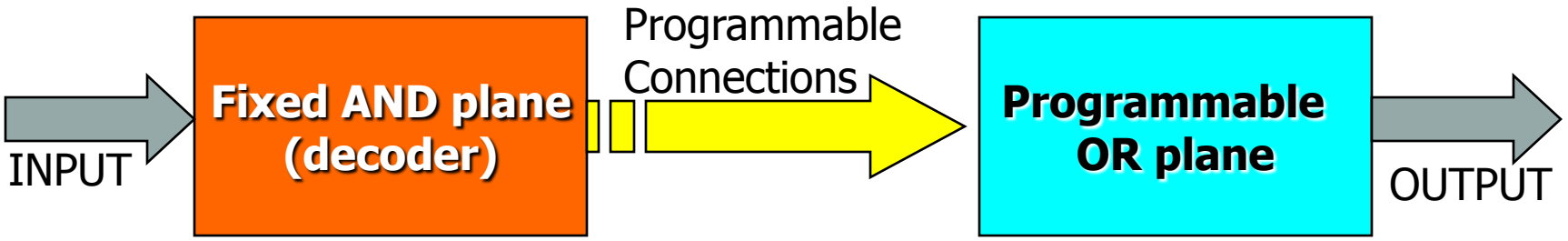
Programmable Logic Devices

- Simple logic arrays
  - Implement 2 level logic circuits (AND/OR)
  - Based on regular array structure
  - Several types
    - Read Only Memories (ROMs and PROMs)
    - Programmable Logic Array (PLA)
    - Programmable Array Logic (PAL)
- Complex Programmable Logic Devices (CPLD)
  - Collection of individual PLDs (e.g. PALs, PLAs) on a single chip.
  - Programmable interconnects.

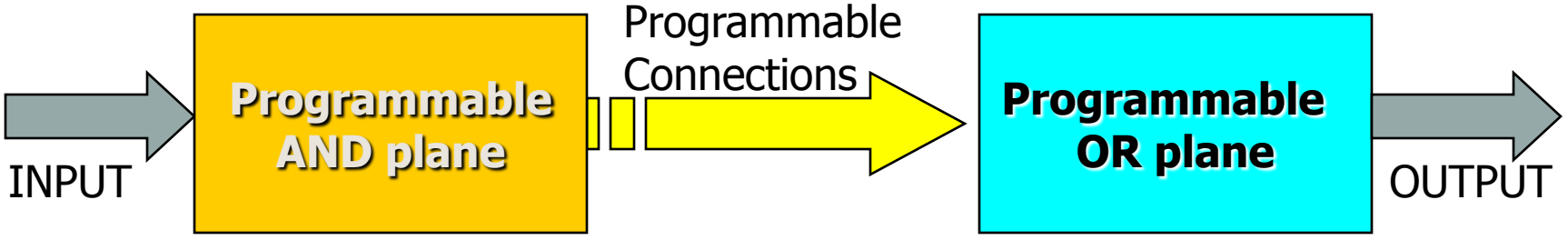
Programmable Logic Devices (Cont.)

- Field Programmable Gate Arrays (FPGA)
  - many copies of common building block
  - each block can be configured for different logic functions and typically includes a flip flop and a 4 input function generator
  - programmable interconnect
  - often includes SRAM blocks
  - largest FPGAs have about 100K flip flops, 100K function generators and 10 MB of SRAM

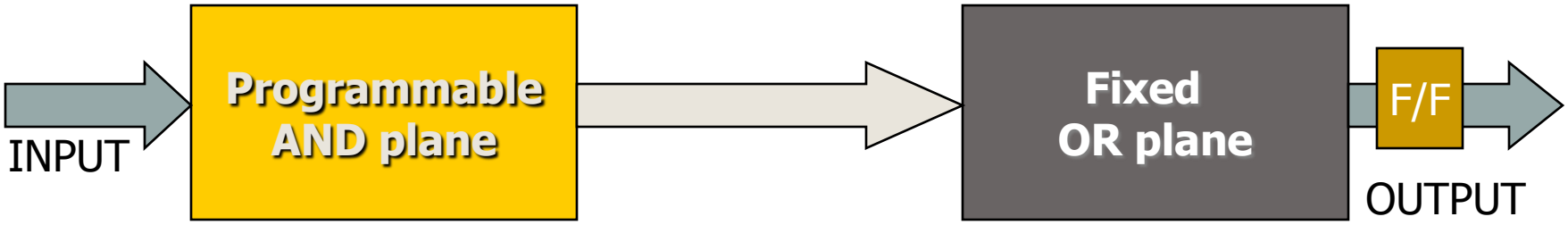
Classifying Three Basic PLDs



(Programmable) Read-Only Memory (ROM)



Programmable Logic Array (PLA)

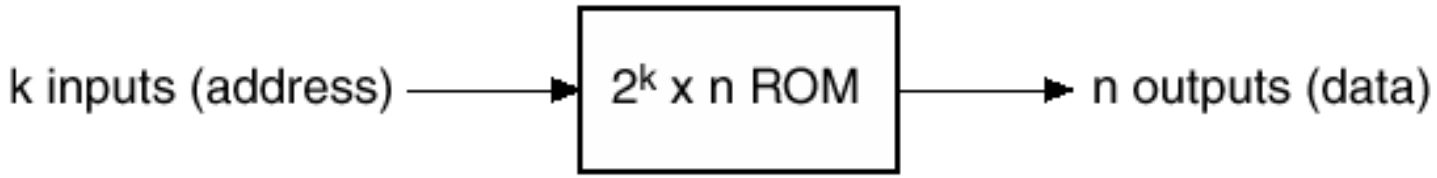


Programmable Array Logic (PAL) Devices

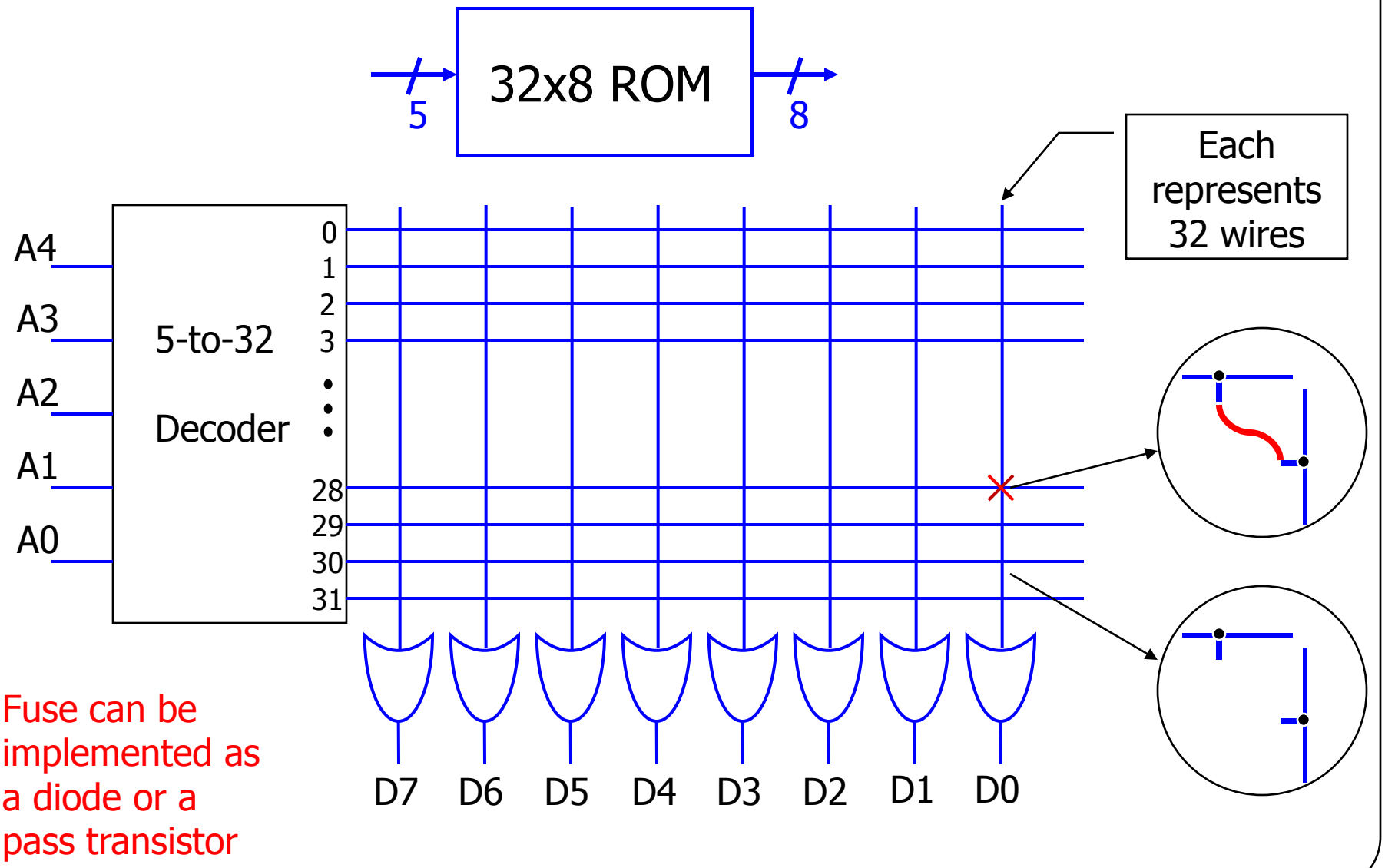
PAL: trademark of AMD, use PAL as an adjective or expect to receive a letter from AMD's lawyers

Read-Only Memory

- "Permanent" binary information is stored
- Non-volatile memory
  - Power off does not erase the information stored.

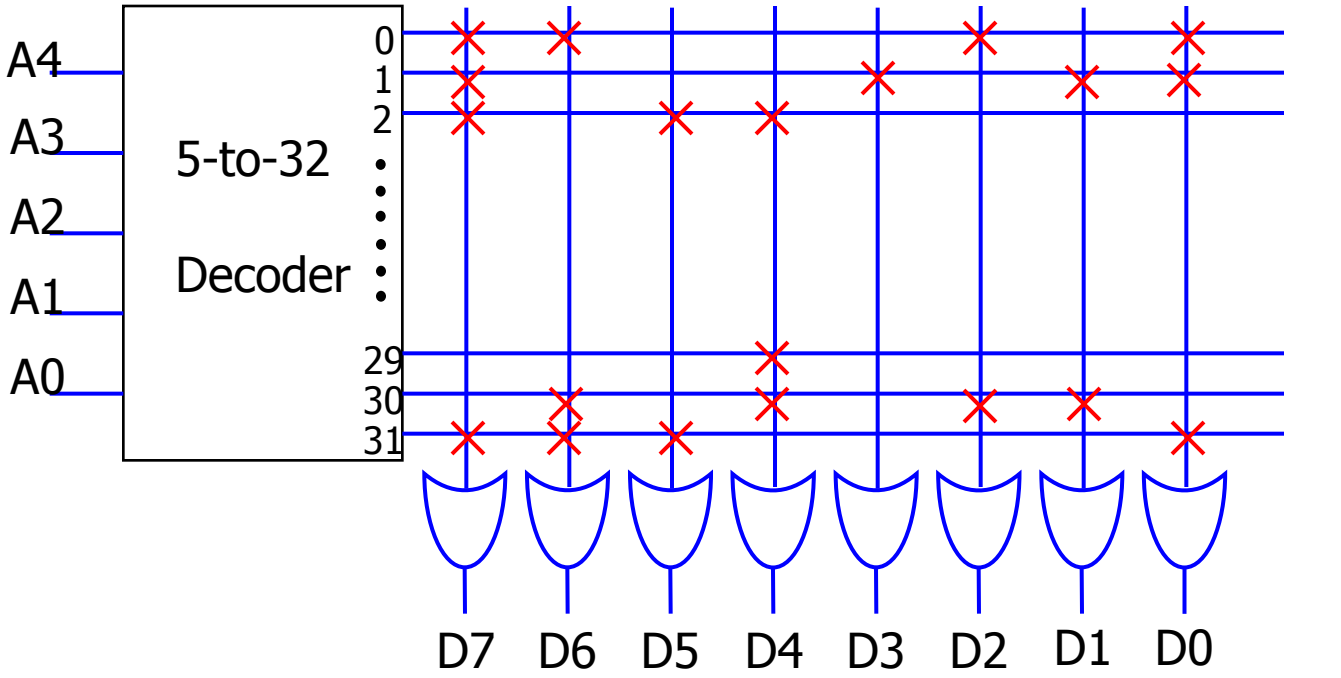


Implementation of ROM



Programming the 32x8 ROM

A4	A3	A2	A1	A0	D7	D6	D5	D4	D3	D2	D1	D0
0	0	0	0	0	1	1	0	0	0	1	0	1
0	0	0	0	1	1	0	0	0	1	0	1	1
0	0	0	1	0	1	0	1	1	0	0	0	0
…	…	…	…	…	…	…	…	…	…	…	…	…
1	1	1	0	1	0	0	0	1	0	0	0	0
1	1	1	1	0	0	1	0	1	0	1	1	0
1	1	1	1	1	1	1	1	0	0	0	0	1



Example: Lookup Table

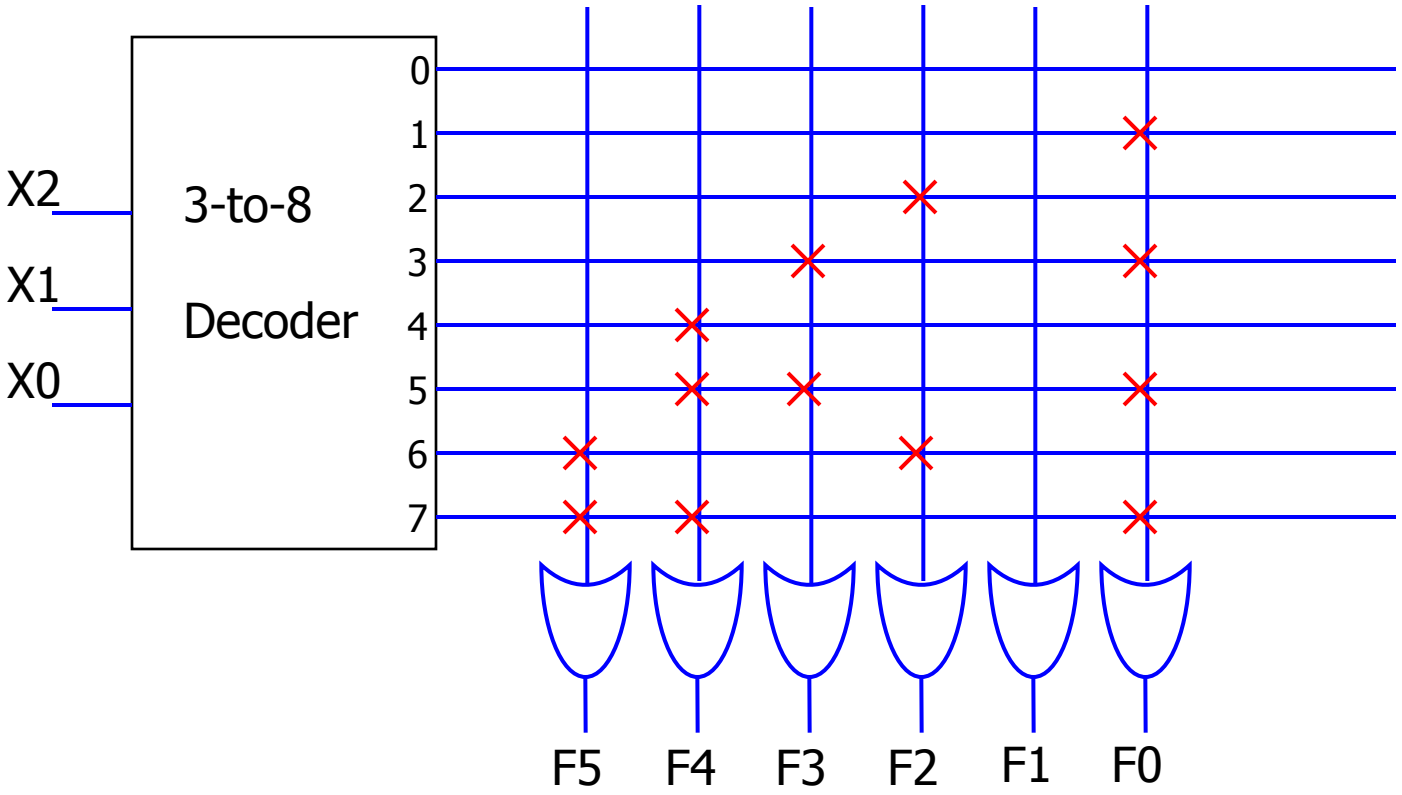
- Design a square lookup table for $F(X) = X^2$ using ROM

X	$F(X) = X^2$
0	0
1	1
2	4
3	9
4	16
5	25
6	36
7	49

X	$F(X) = X^2$
000	000000
001	000001
010	000100
011	001001
100	010000
101	011001
110	100100
111	110001

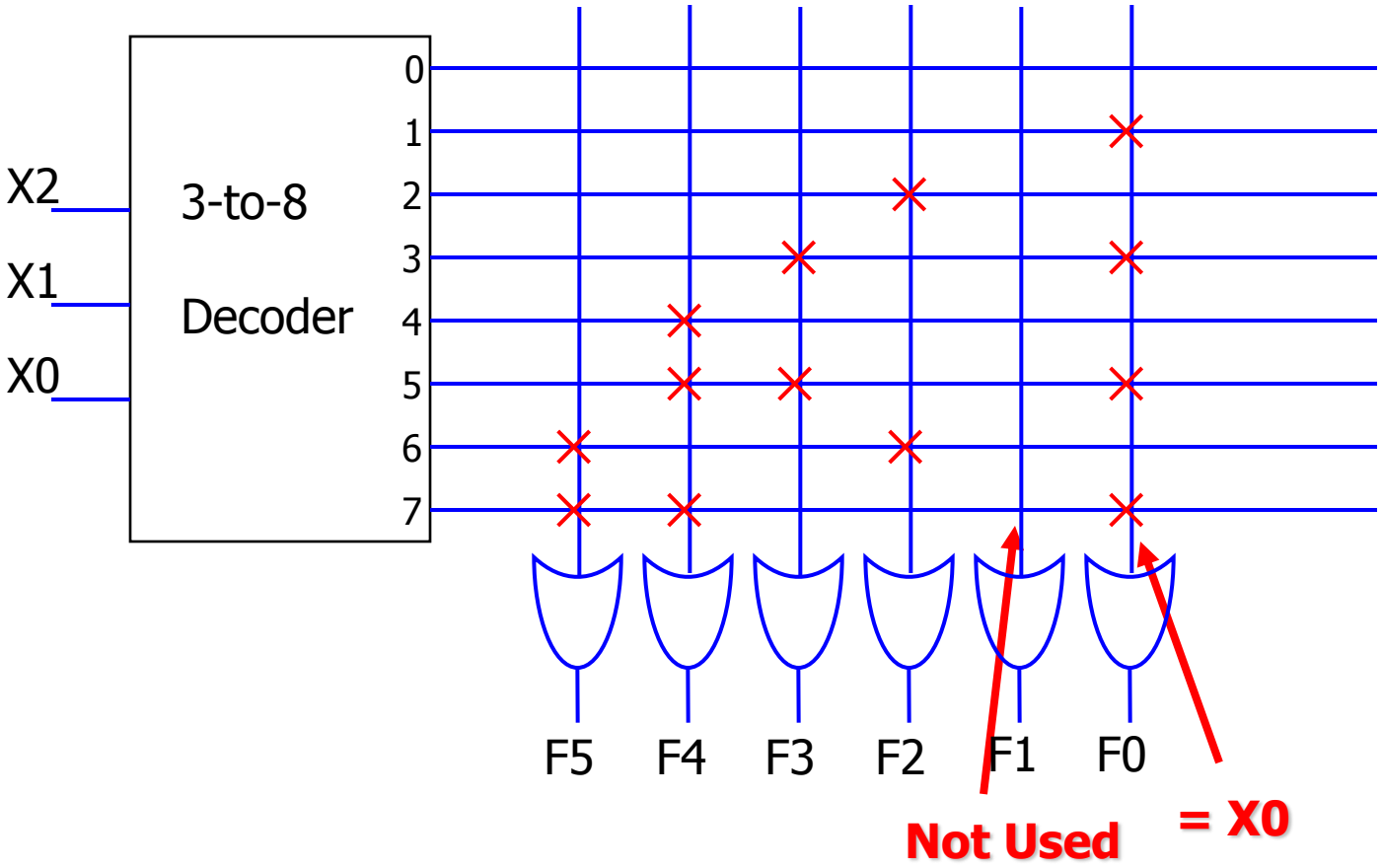
Square Lookup Table using ROM

X	$F(X)=X^2$
000	000000
001	000001
010	000100
011	001001
100	010000
101	011001
110	100100
111	110001



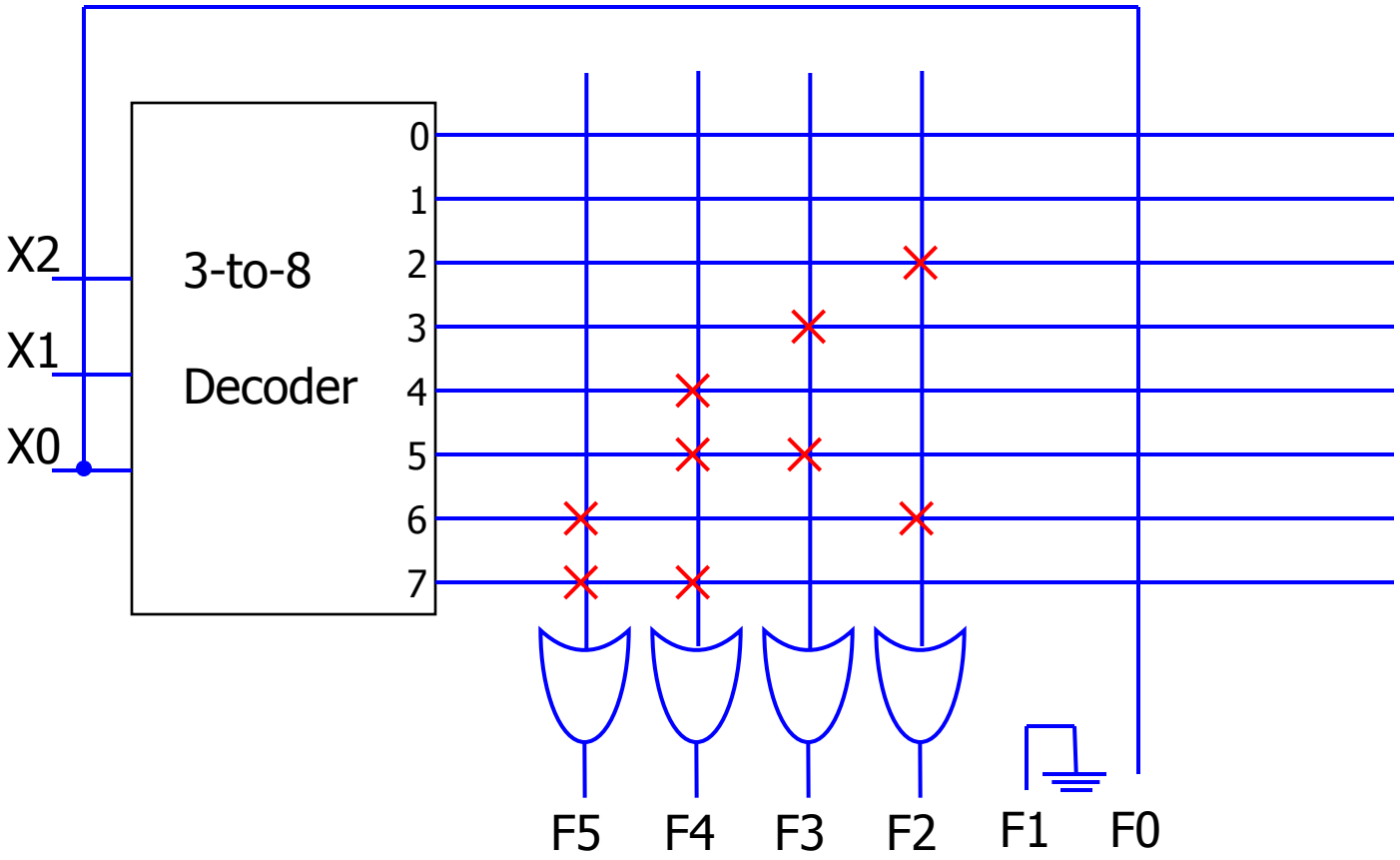
Square Lookup Table using ROM

X	$F(X)=X^2$
000	000000
001	000001
010	000100
011	001001
100	010000
101	011001
110	100100
111	110001

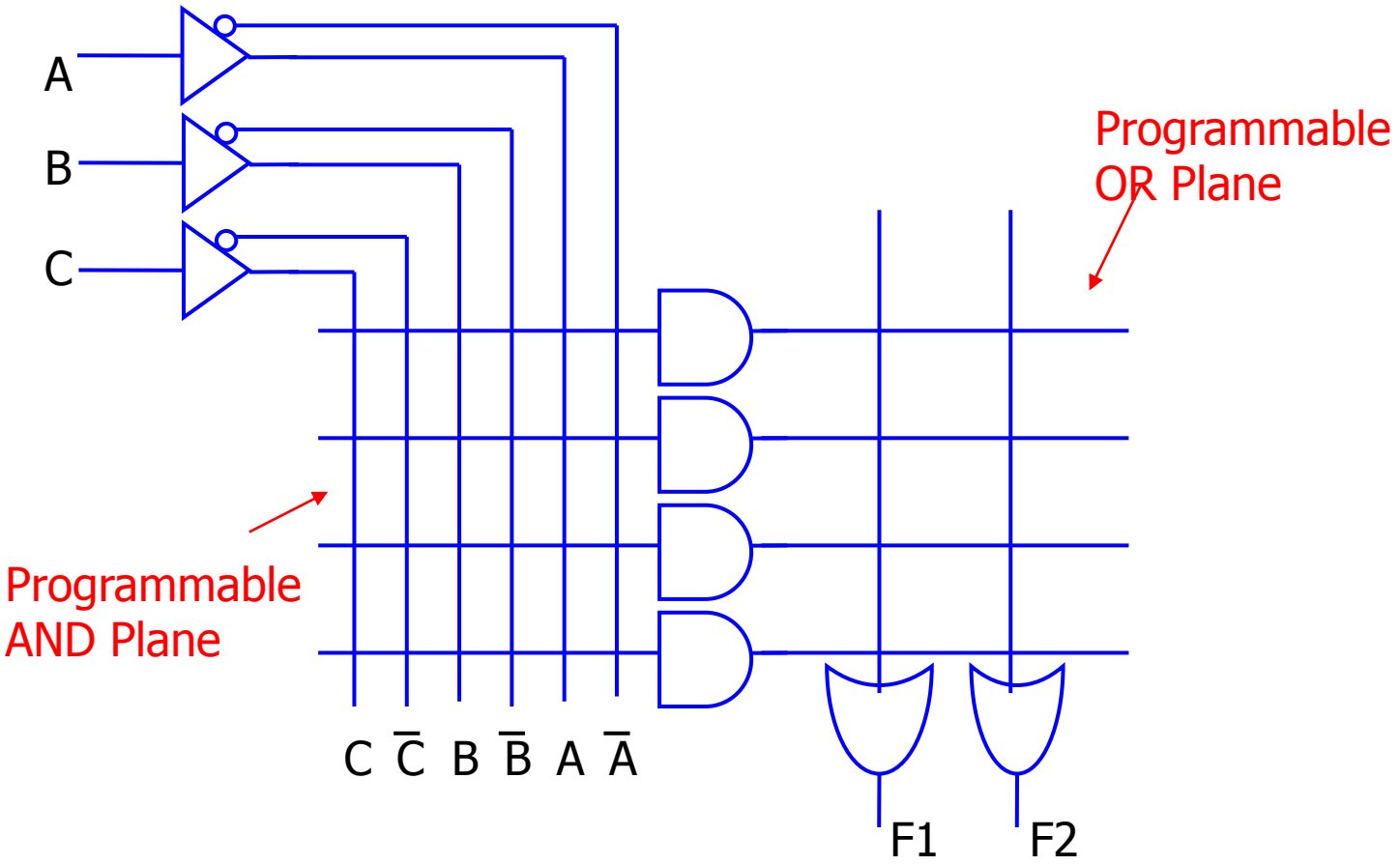


Square Lookup Table using ROM

X	$F(X)=X^2$
000	000000
001	000001
010	000100
011	001001
100	010000
101	011001
110	100100
111	110001



Programmable Logic Arrays



- PLAs have configurable "AND-plane" & "OR-plane"
- Can implement any 2-level AND-OR circuit.

Example using PLA

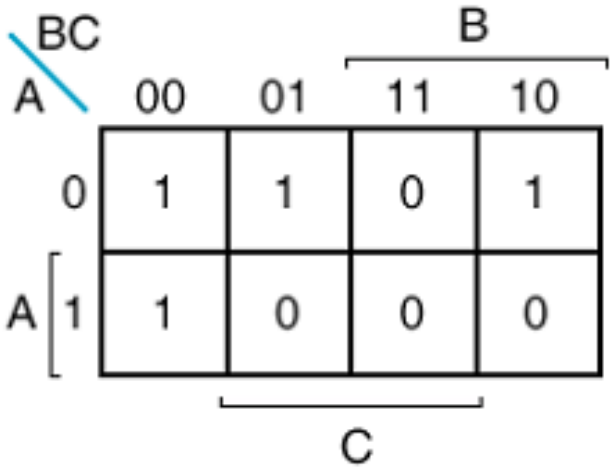
- **Functions to implement are**

$$F_1(A,B,C) = \sum m(0,1,2,4)$$

$$F_2(A,B,C) = \sum m(0,5,6,7)$$

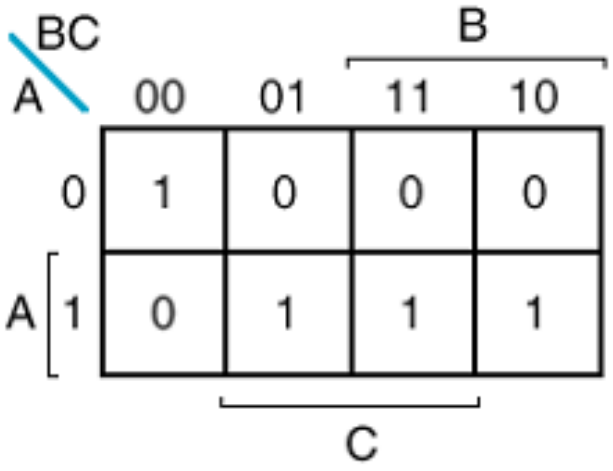
Example using PLA (Cont.)

- **Minimum product terms are**



$F_1 = \overline{A}\,\overline{B} + \overline{A}\,\overline{C} + \overline{B}\,\overline{C}$

$\overline{F}_1 = AB + AC + BC$

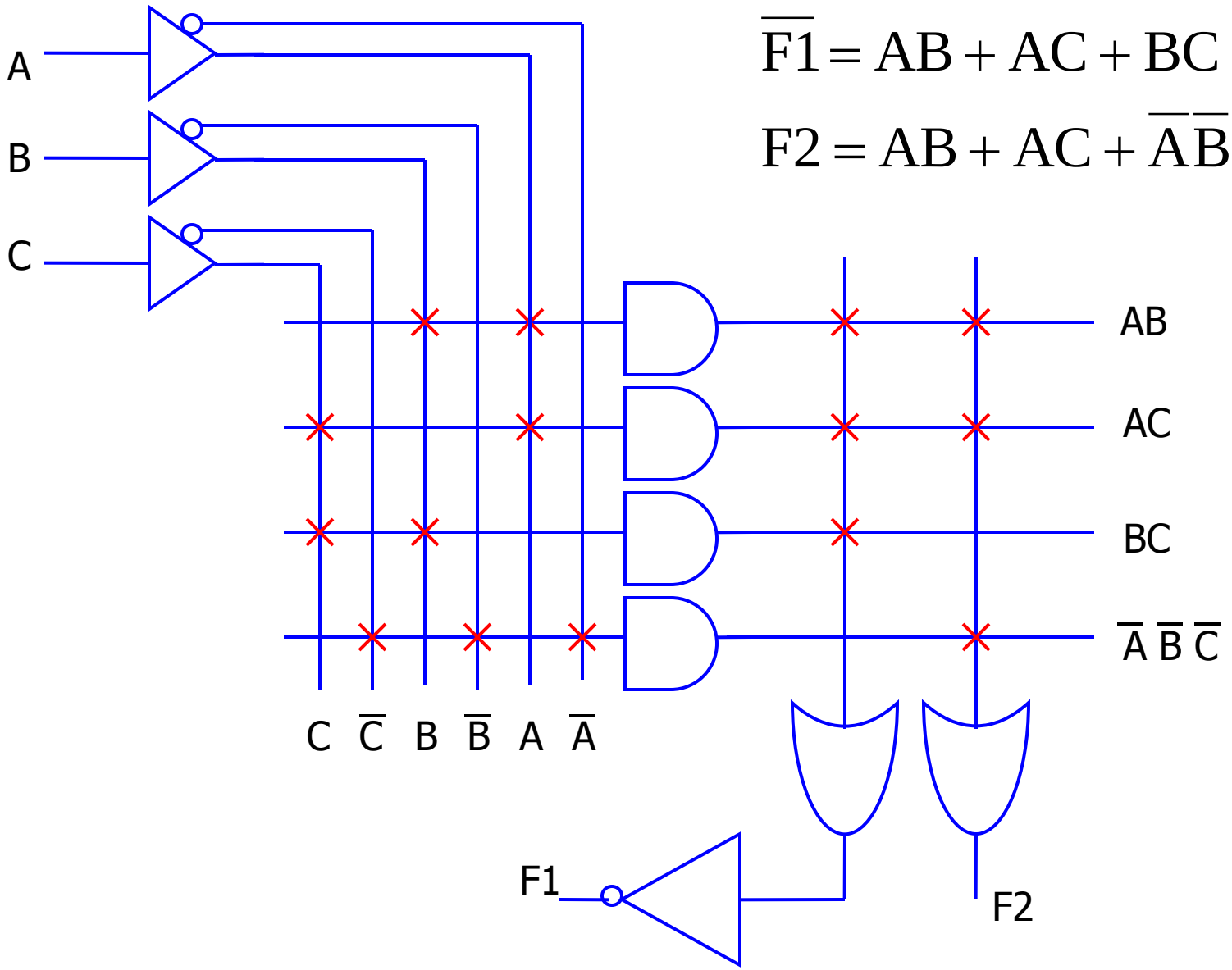


$F_2 = AB + AC + \overline{A}\,\overline{B}\,\overline{C}$

Example using PLA (Cont.)

$$\overline{F1} = AB + AC + BC$$

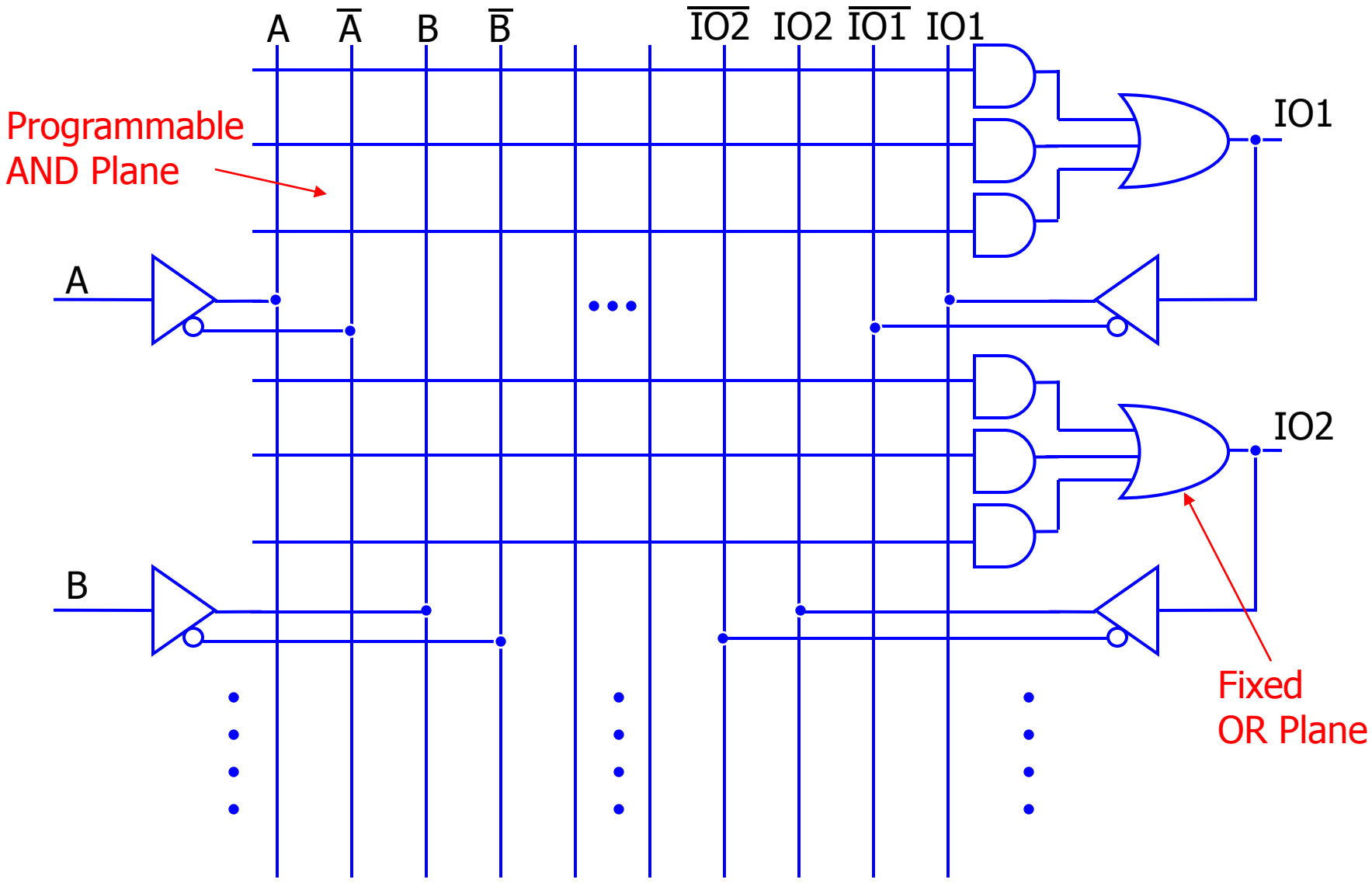
$$F2 = AB + AC + \overline{A}\,\overline{B}\,\overline{C}$$



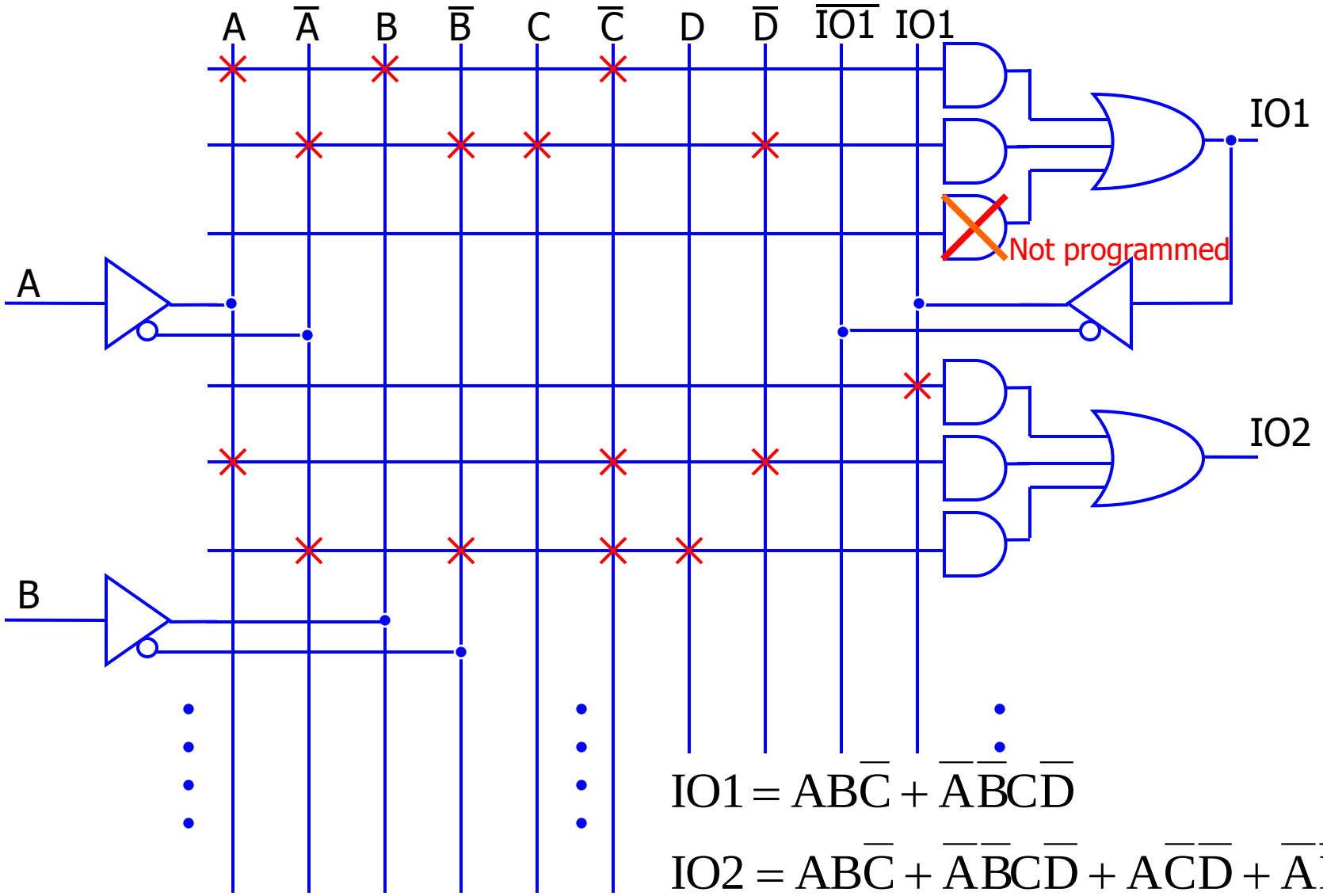
Programmable Array Logic

- PAL is similar to PLA but fixed OR-plane.
- Simpler to program and cheaper implementation.
- Faster than PLA as OR-plane is hard wired.
- Limited number of terms in each output.

PAL Structure



PAL Device Design Example



$$IO1 = AB\overline{C} + \overline{A}\,\overline{B}C\overline{D}$$

$$IO2 = AB\overline{C} + \overline{A}\,\overline{B}C\overline{D} + A\overline{C}\,\overline{D} + \overline{A}\,\overline{B}\,\overline{C}\,\overline{D}$$

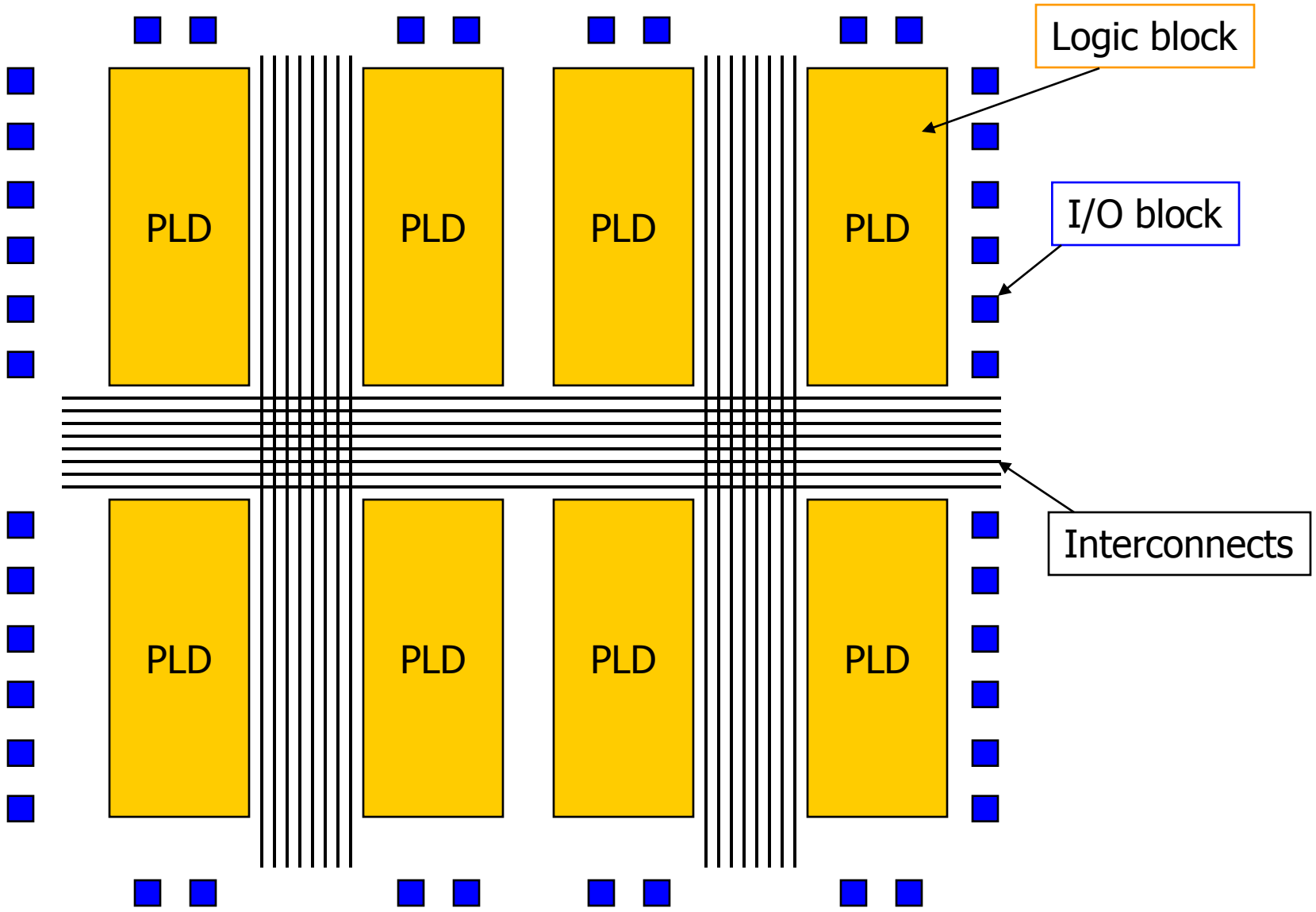
Comparison of PROMs, PLAs & PALs

- Can view PROMs and PALs as restricted forms of PLA.
  - PROMs are logically equivalent to PLA with AND-plane that generates all minterms and configurable OR-plane
  - PAL is logically equivalent to PLA with fixed OR-plane in which each output is the OR of a subset of the ANDs
- Most parts include flip flops, for implementing sequential circuits.

Why CPLD ?

- 16V8 PLD (20 Pins)
  - can have 16 inputs (max) and/or 8 outputs
  - has 32 inputs to each of the AND gates
- 22V10 PLD (24 pins)
  - can have 22 inputs and/or 10 outputs (max)
  - has 44 inputs to each of the AND gates
- How about a "128V64" for larger applications?
  - It will be much slower and will more wasted silicon space
  - *Solution?* Use ***CPLDs***

Top Level CPLD Architecture



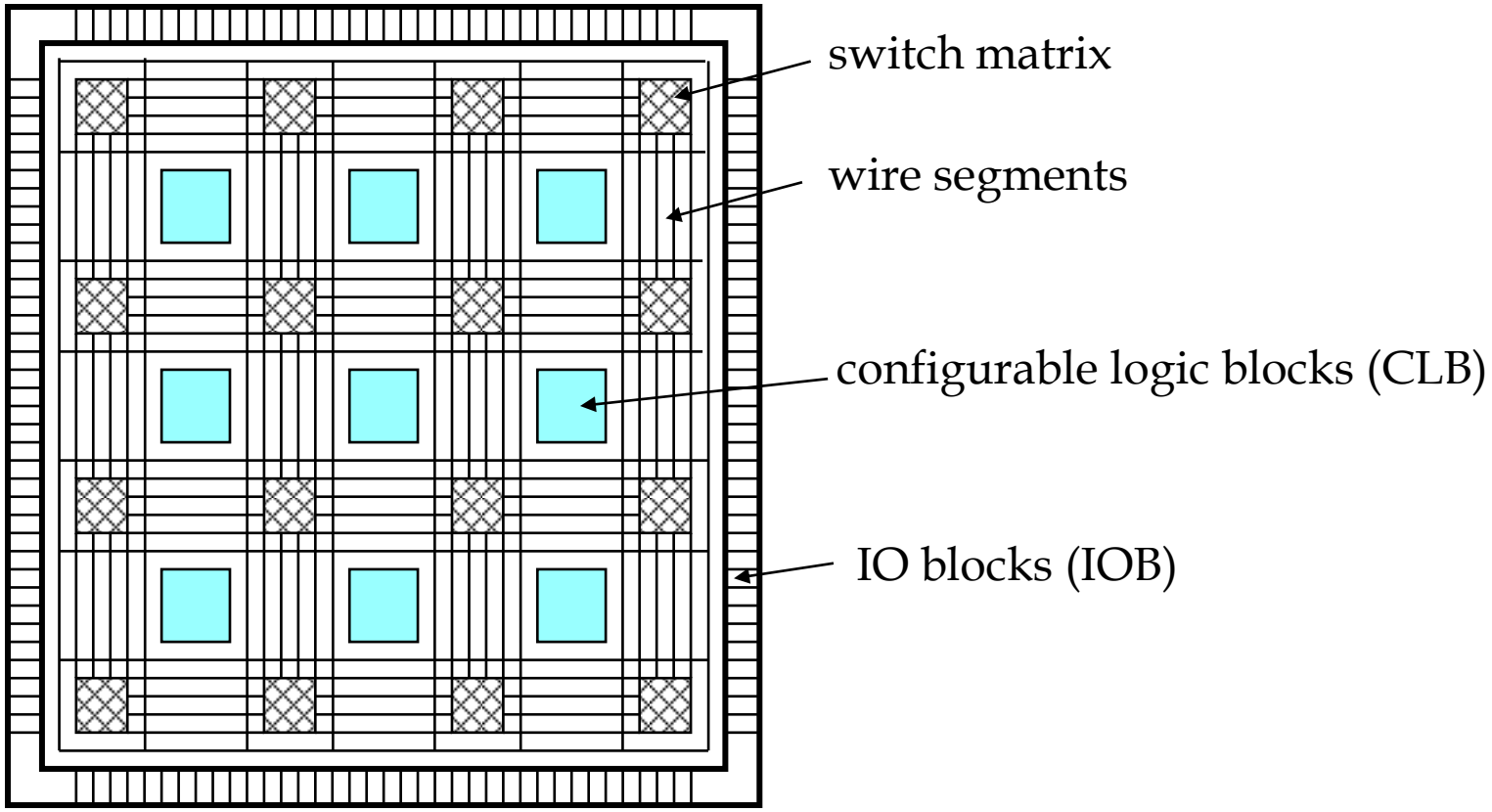
Field Programmable Gate Arrays

- FPGAs can be used to construct more complex circuits.
- Chip contains a large number (tens of thousands) of configurable logic building blocks.
  - Typically each block includes a 4 input function generator, a flip flop and some "glue" logic.
  - CAD tools map high level circuit to basic blocks, configuring function generators & other configurable elements as needed.

Field Programmable Gate Arrays (Cont.)

- Programmable interconnect used to wire logic blocks.
  - Wire segments connected to logic blocks and to other wire segments by configurable switches.
  - CAD tools determine switch configuration needed to provide right connectivity.
- CAD tools perform *mapping*, *placement*, *routing*.
  - Routing information used in timing analysis & simulation.

Top Level FPGA Architecture



Top Level FPGA Architecture (Cont.)

- CLBs can be connected to "passing" wires.
- Wire segments connected by switch matrix.
- Long wire segments used to connect distant CLBs.
- Configuration information stored in SRAM bits that are loaded when power turns on.