Introduction to PLDs

Digital Logic Design (CC131)

BSCS 2nd Semester Spring-2026

By

Syed Zulfiqar Ali Shah,

Associate Professor (CS) / HoD

Govt. College of Management Sciences, Kohat

www.linkedin.com/in/syedzulfiqaralishah

LOGIC DEVICES

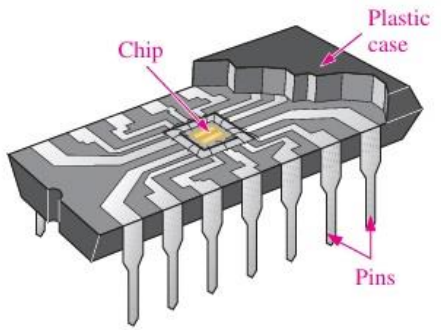
Logic circuits can be broadly classified into two categories:

1. Fixed-function Logic Devices
2. Programmable Logic Devices

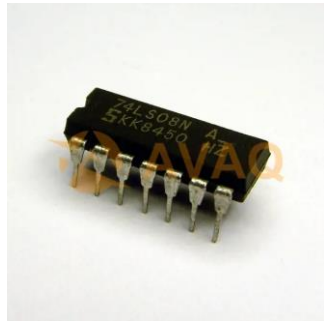
1. Fixed-function Logic Devices

A **fixed-function logic device** is an integrated circuit (IC) whose logic function is permanently determined by the manufacturer and cannot be changed by the user.

Examples



- AND gate IC (e.g., 7408)
- OR gate IC (e.g., 7432)
- NAND gate IC (e.g., 7400)
- Multiplexers
- Decoders
- Counters
- Adders



Example

Suppose you buy a 7408 IC (Quad 2-input AND gate).
No matter what you do, it will always behave as an AND gate. You cannot reconfigure it into an OR gate or a decoder.

Advantages

- Simple to use
- Usually cheaper
- Optimized for a specific function
- Often faster for simple applications

Disadvantages

- Inflexible
- Large circuits require many different ICs
- Any design change may require replacing hardware

2. Programmable Logic Devices

- A **Programmable Logic Device (PLD)** is an IC whose internal connections can be programmed by the user to implement different logic functions.

- The hardware is manufactured in a general form, and the desired logic is configured later.

Example

You can buy one PLD and program it to behave as:

- An AND-OR circuit today
- A decoder tomorrow
- A counter next week

without changing the physical chip.

Common Types

- PROM (Programmable Read-Only Memory)
- PLA (Programmable Logic Array)
- PAL (Programmable Array Logic)
- CPLD (Complex Programmable Logic Device)
- FPGA (Field-Programmable Gate Array)

Example

Suppose you need:

$$F=AB+AC$$

You program the PLD to create:

- Product term AB
- Product term AC
- OR them together

Later, if you need

$$F=\overline{A}B+BC$$

you simply reprogram the device.

Advantages

- Highly flexible
- Easy to modify designs
- Fewer ICs required
- Faster design and prototyping

Disadvantages

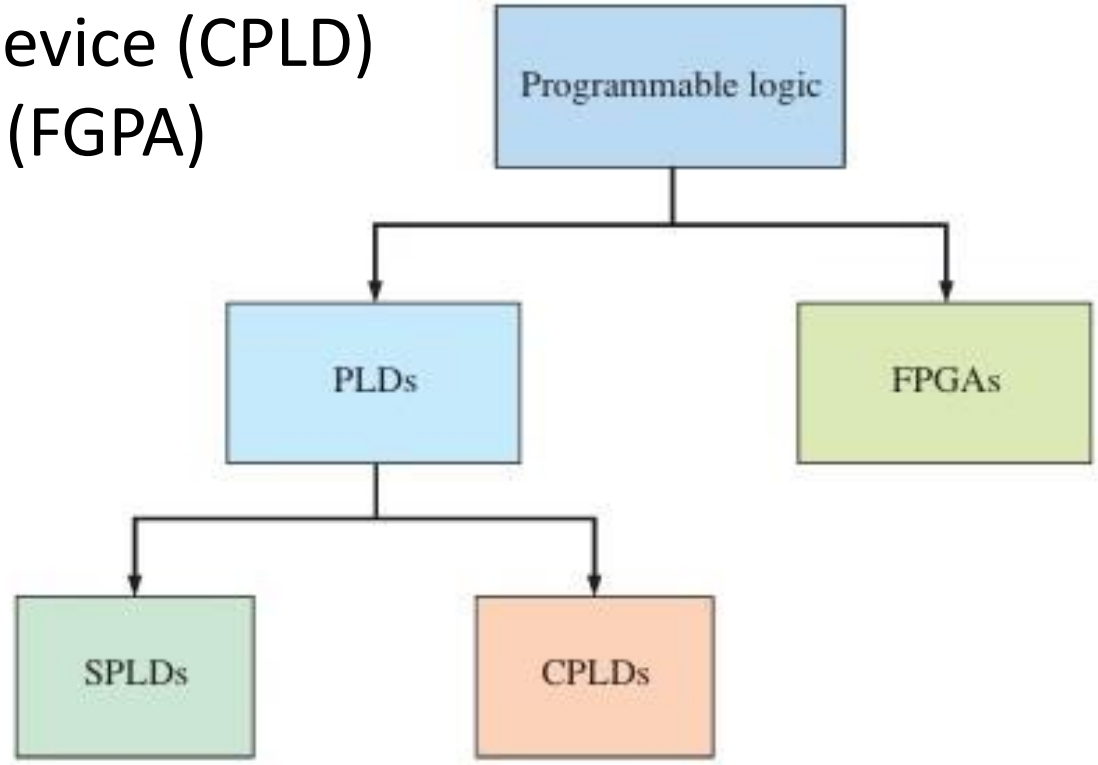
- More expensive than simple fixed-function ICs
- Requires programming tools
- Can be more complex to learn

Types of Programmable Logic Devices

Types of Programmable Logic Devices

There are three types of PLDs;

1. Simple Programmable Logic Device (SPLD)
2. Complex Programmable Logic Device (CPLD)
3. Field Programmable Gate Array (FGPA)

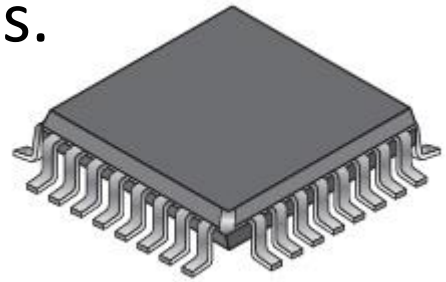


Programmable logic hierarchy.

1. Simple Programmable Logic Device

"An SPLD is a programmable logic device that contains one programmable logic block capable of implementing small combinational and sequential circuits."

- An **SPLD** is the smallest and simplest type of programmable logic device.
- It contains a limited number of programmable logic gates and is designed to implement relatively simple digital functions.



1. Simple Programmable Logic Device

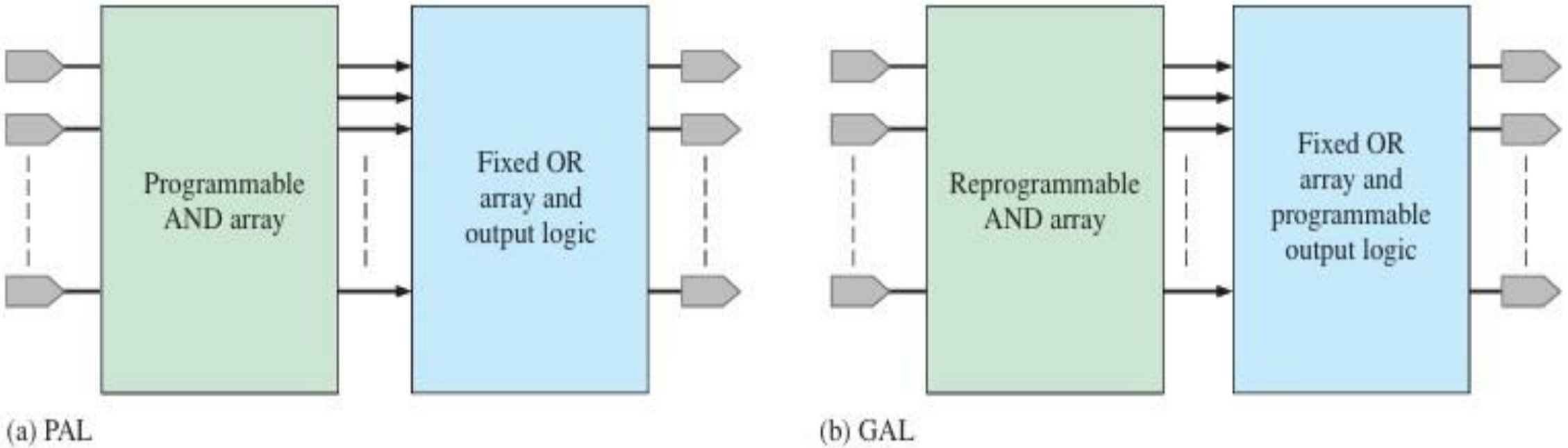
Examples:

- PAL (Programmable Array Logic)
- PLA (Programmable Logic Array)
- GAL (Generic Array Logic)

Characteristics:

- Small number of inputs and outputs
- Implements Boolean expressions directly
- Suitable for simple Sum-of-Products (SOP) or Product-of-Sums (POS) designs
- Usually replaces a few SSI/MSI chips

1. Simple Programmable Logic Device



Block diagrams of simple programmable logic devices (SPLDs).

1. Simple Programmable Logic Device

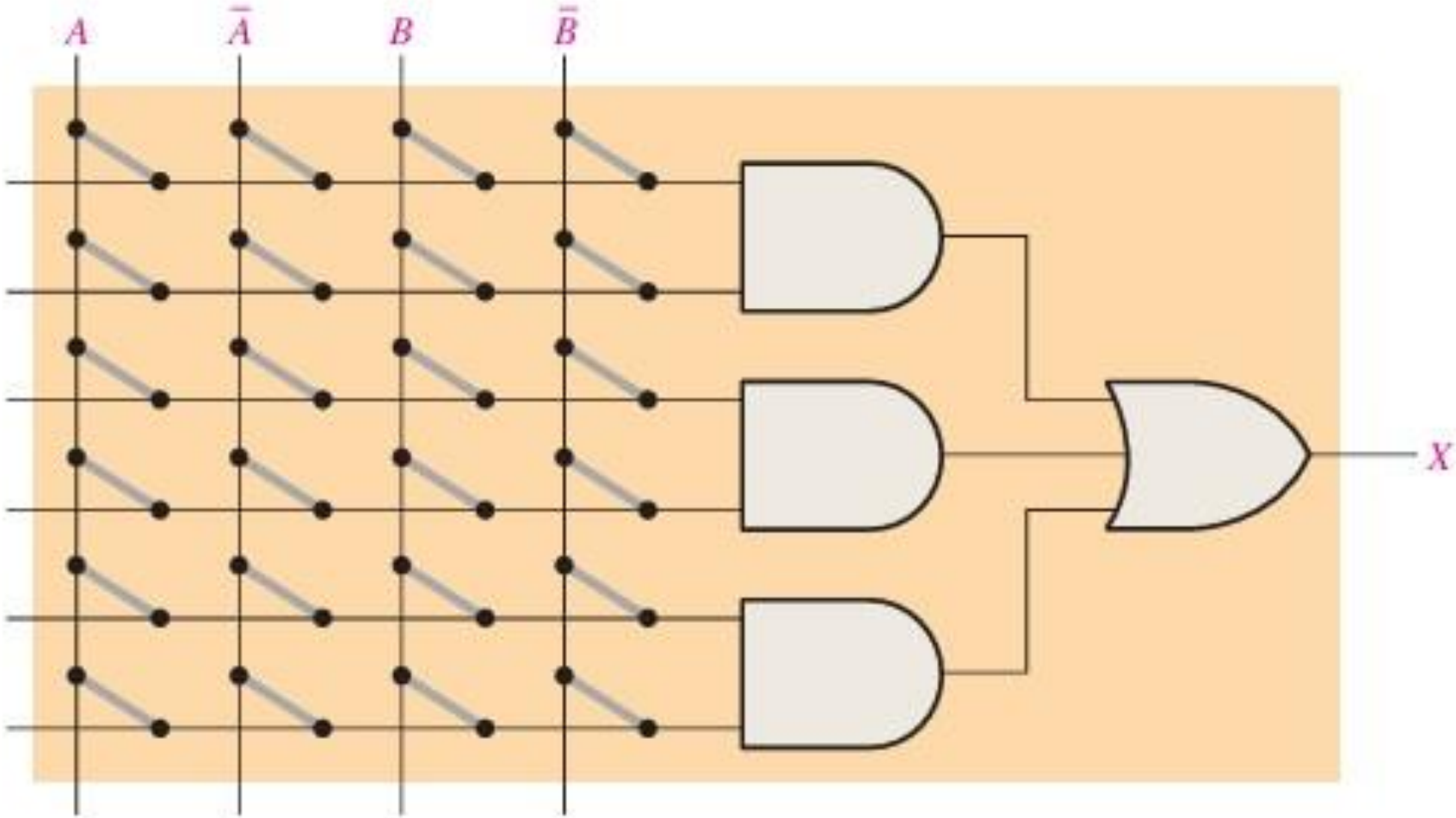
Uses:

- Designing a decoder
- Designing a simple state machine
- Implementing a few logic equations

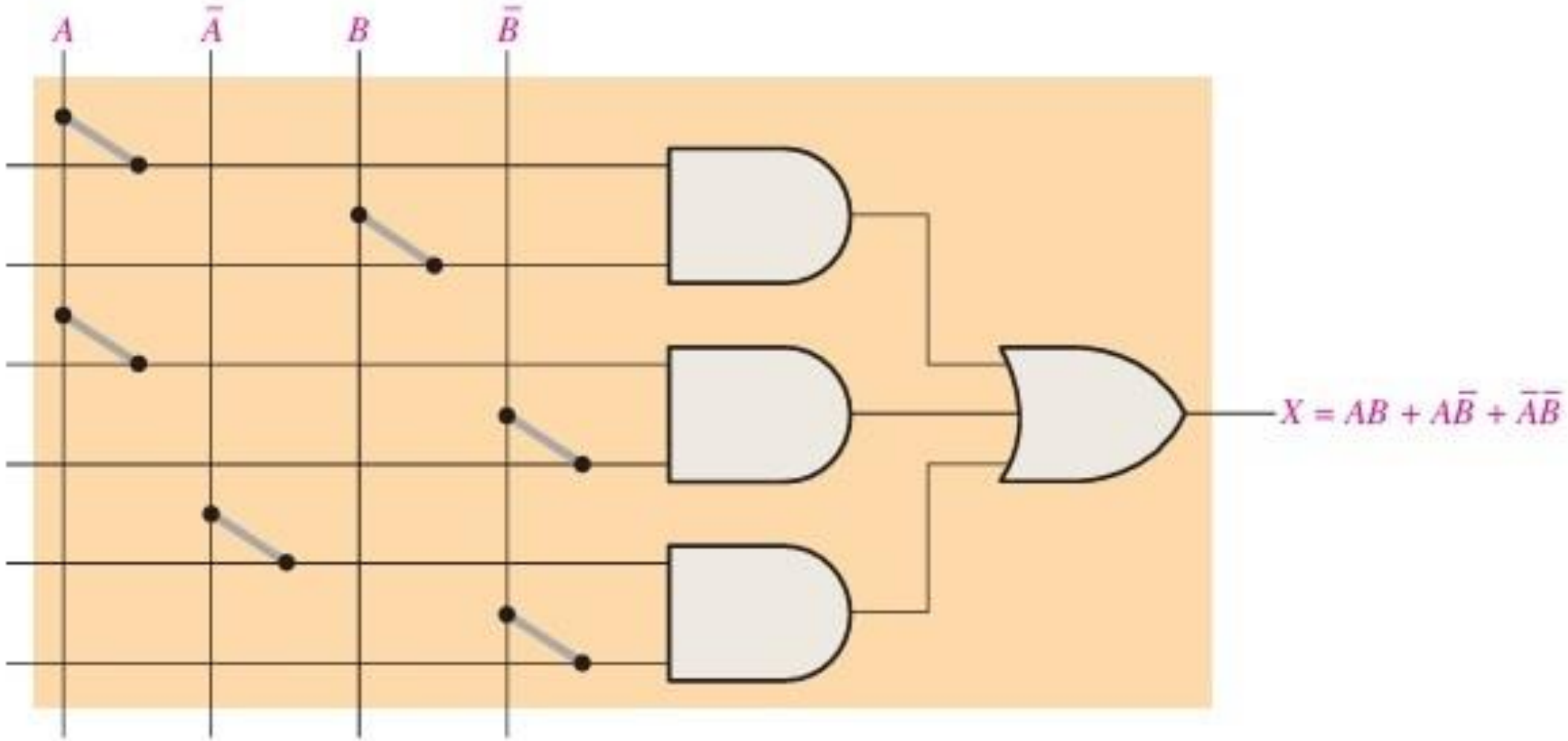
Programmable Array Logic (PAL)

"A PAL is a programmable logic device in which the **AND array is programmable** and the **OR array is fixed**."

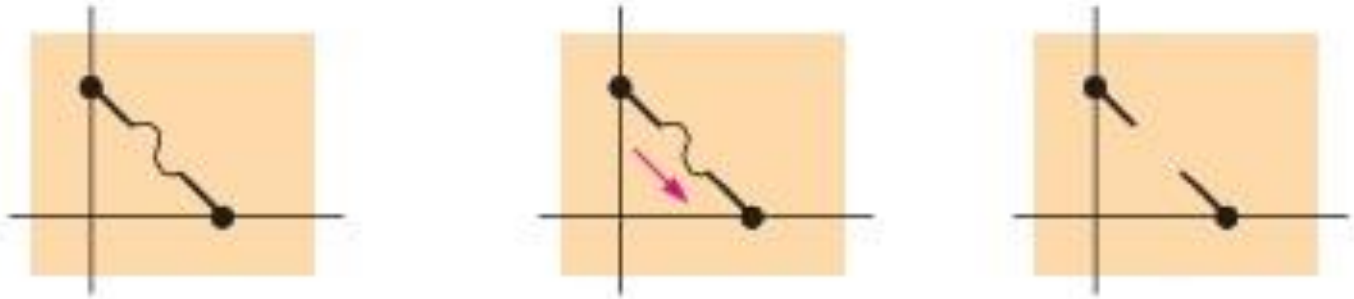
- A PAL is an early type of Simple Programmable Logic Device (SPLD) used to implement Boolean logic functions.



Basic AND/OR structure of a PAL.



PAL implementation of a sum-of-products expression.



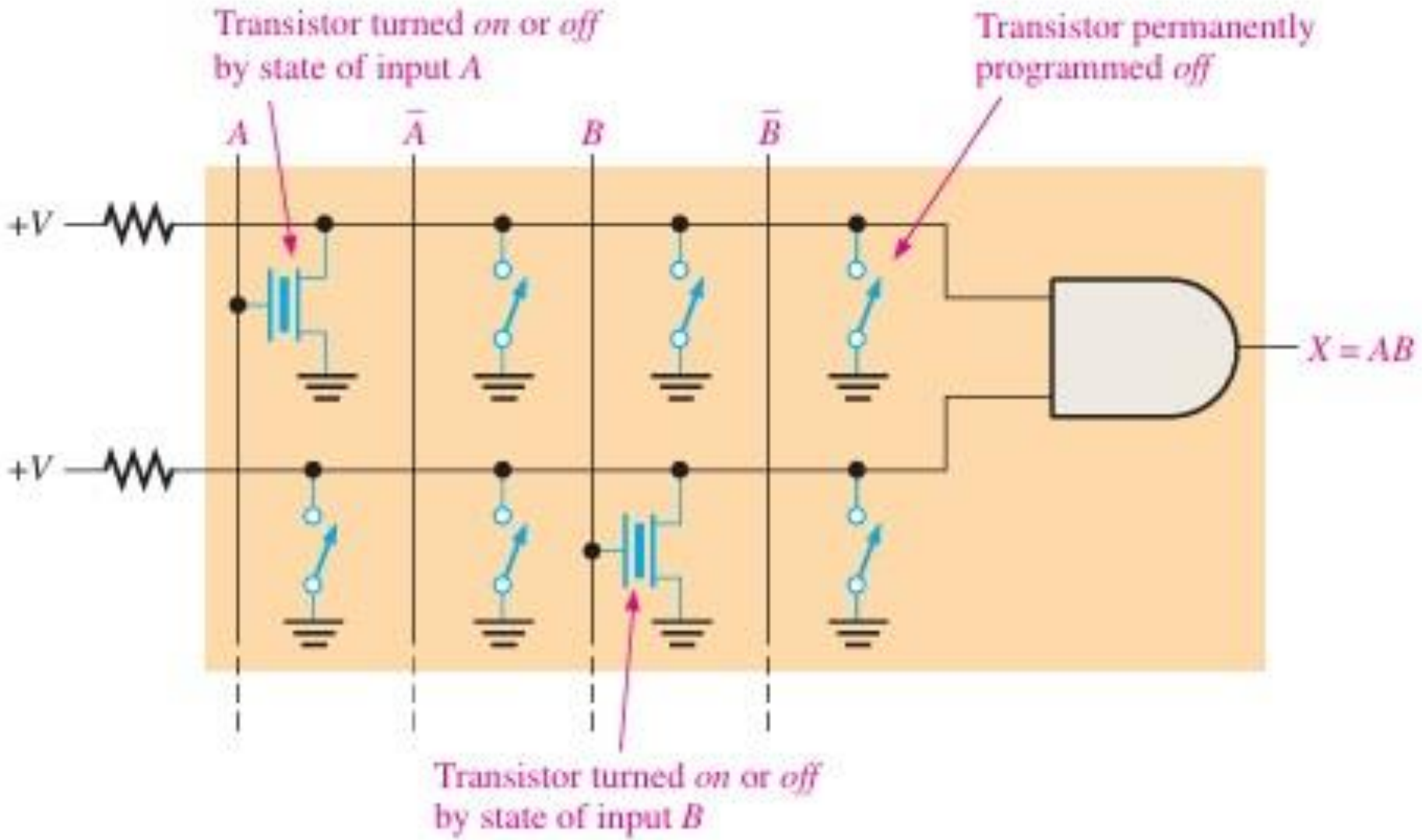
(a) Fuse intact before programming

(b) Programming current

(c) Fuse open after programming

The programmable fuse link.

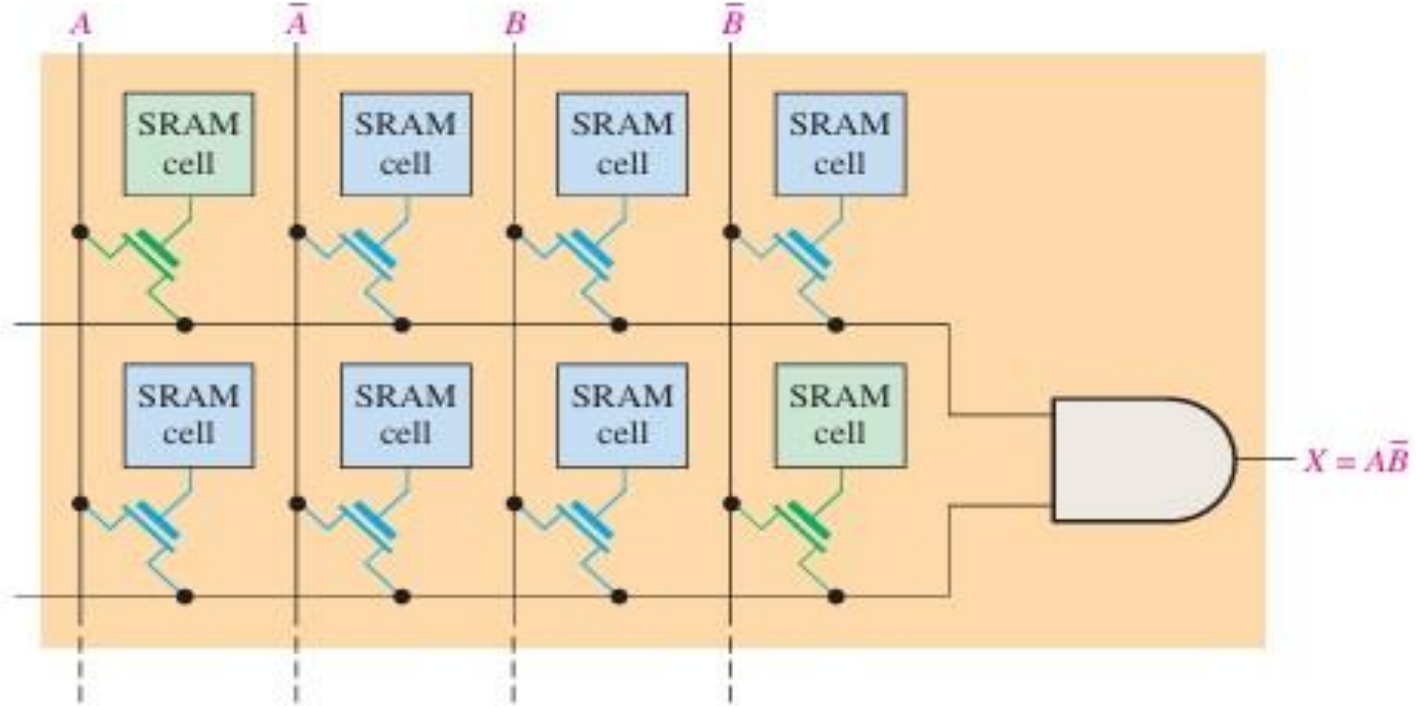
Programmable logic devices that use fuse technology are one-time programmable (OTP)



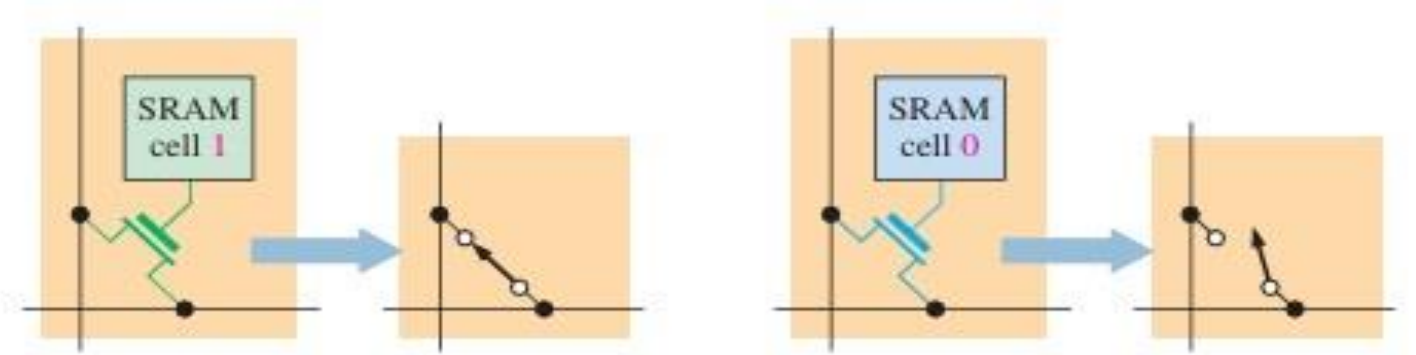
A simple AND array with EPROM technology.

Programmable logic devices that use EPROM technology are mostly one-time programmable (OTP)

Programmable logic devices that use SRAM technology are reprogrammable.



(a) SRAM-based programmable array



(b) Transistor *on* (c) Transistor *off*

Concept of an AND array with SRAM technology.

For example,

if you want to implement:

$$F=AB+AC$$

The programmable AND array creates the product terms:

- AB
- AC

The fixed OR array then combines them:

$$F=(AB)+(AC)$$

Advantages of PAL

- Faster than PLA
- Simpler architecture
- Easier and cheaper to manufacture

Limitation

- The fixed OR array restricts flexibility.

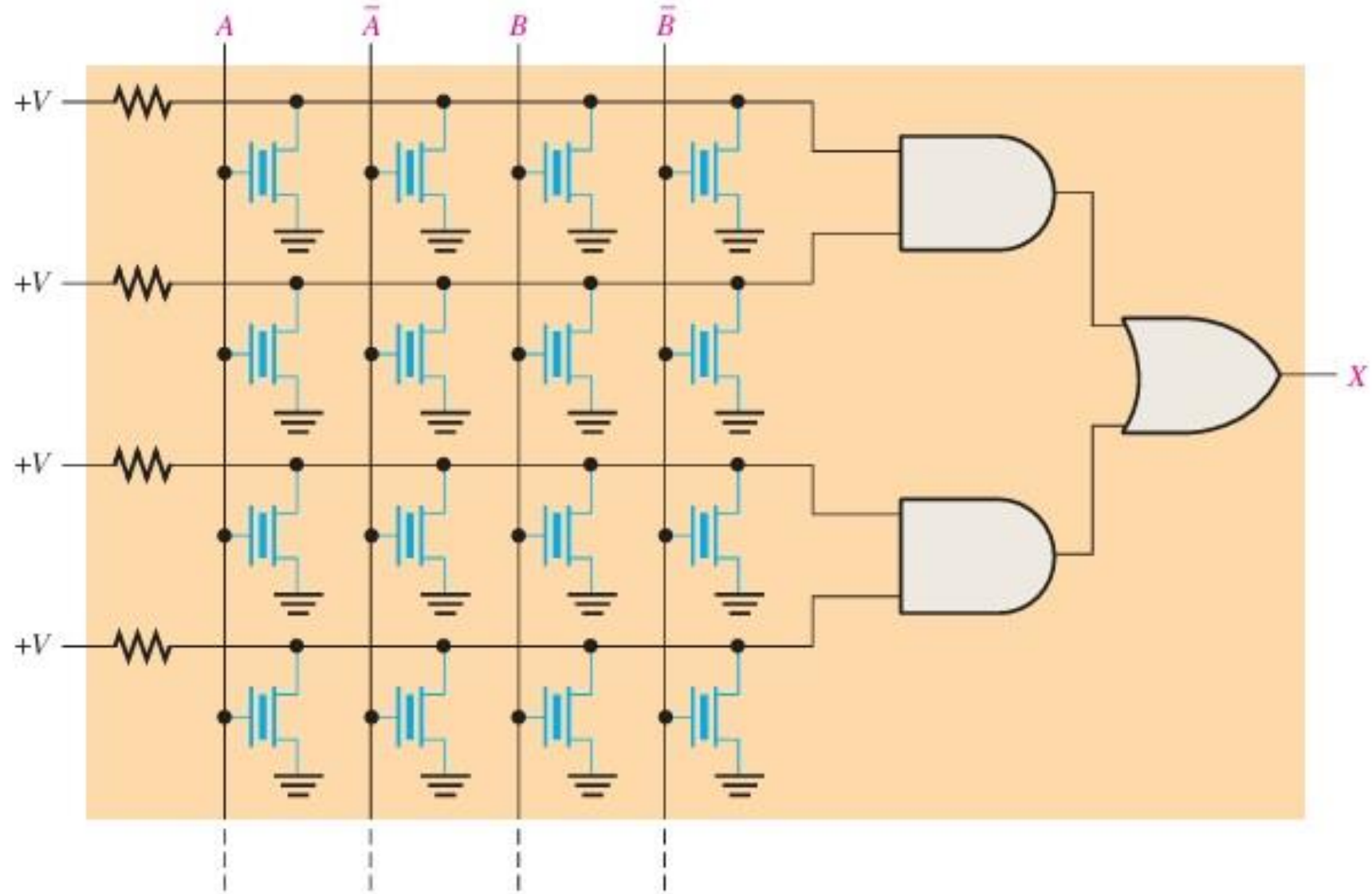
Generic Array Logic (GAL)

A GAL is a reprogrammable SPLD that is functionally similar to a PAL but can be erased and programmed multiple times.

A **GAL** is an improved version of a PAL.

Because GAL uses EEPROM cells, you can:

1. Program it.
2. Test your circuit.
3. Erase it.
4. Program a different circuit.



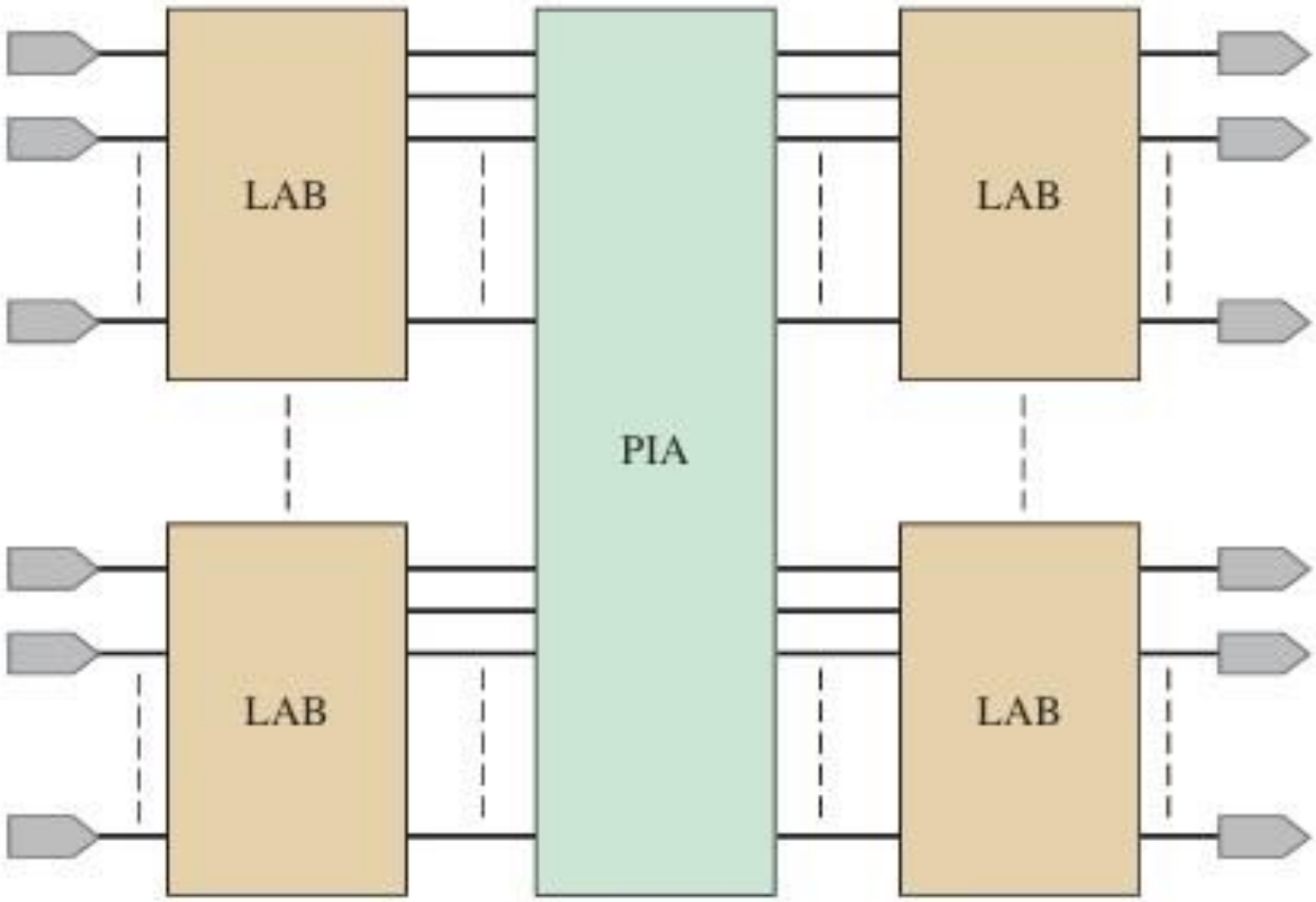
Simplified GAL array.

2. Complex Programmable Logic Device

2. Complex Programmable Logic Device

- A **CPLD** (complex programmable logic device) consists basically of multiple SPLD arrays with programmable interconnections.
- Each SPLD array in CPLD is referred to as **LAB**(Logic Array Block).
- The programmable interconnections are called the **PIA** (programmable interconnect array).

2. Complex Programmable Logic Device

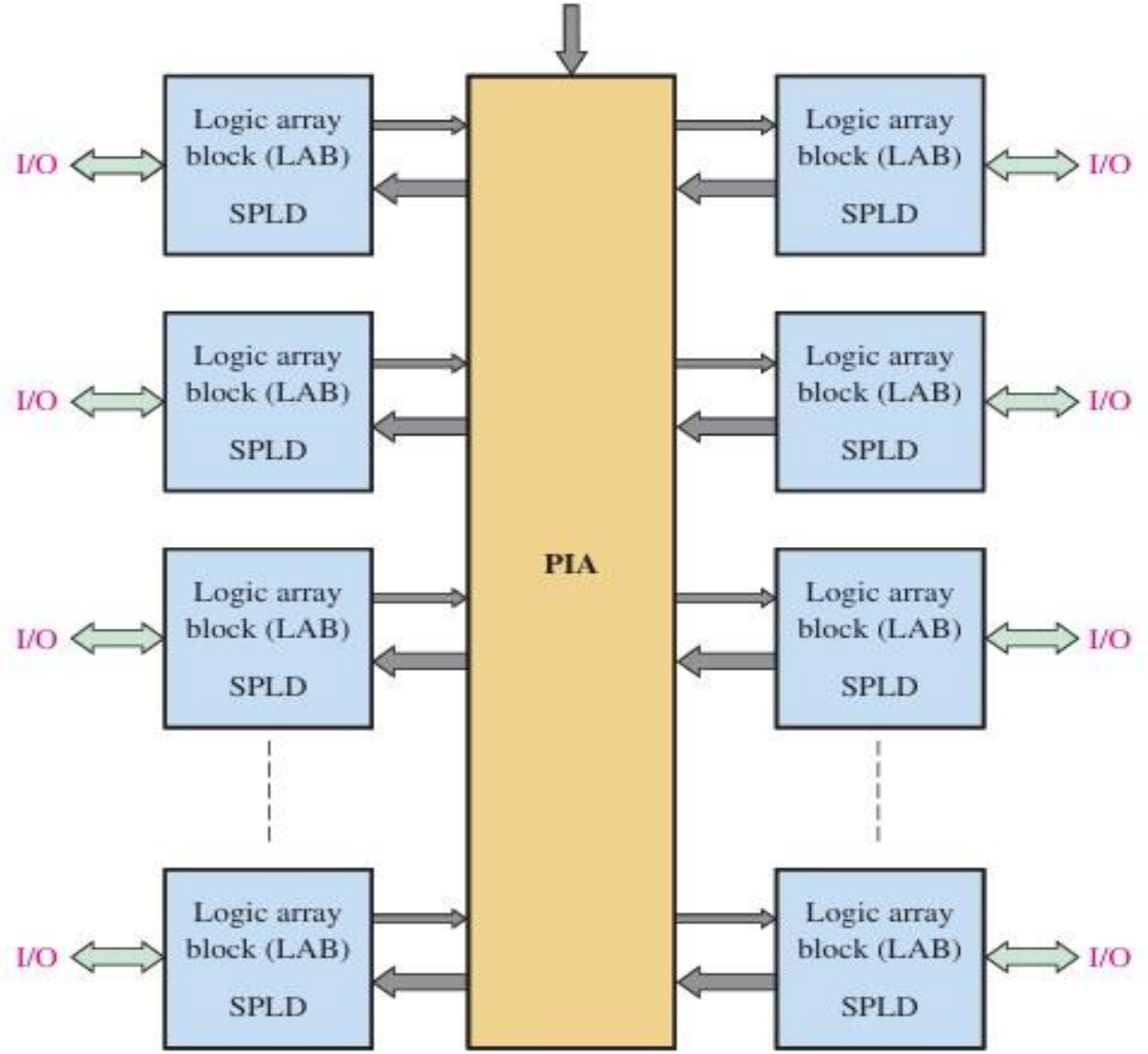


General block diagram of a CPLD.

2. Complex Programmable Logic Device

- The LABs and the interconnections between LABs are programmed using software.
- A CPLD can be programmed for complex logic functions based on the SOP structure of the individual LABs (actually SPLDs).
- Inputs can be connected to any of the LABs, and their outputs can be interconnected to any other LABs via the PIA.
- The CPLD density is usually specified by manufacturers in terms of macrocells.
- Densities can range from tens of macrocells to over 1500 macrocells in packages

2. Complex Programmable Logic Device



Basic block diagram of a generic CPLD.

2. Complex Programmable Logic Device

Characteristics:

- Much larger than an SPLD
- Contains many logic blocks (often called macrocells)
- Predictable timing behavior
- Non-volatile configuration (usually retains programming when power is removed)

2. Complex Programmable Logic Device

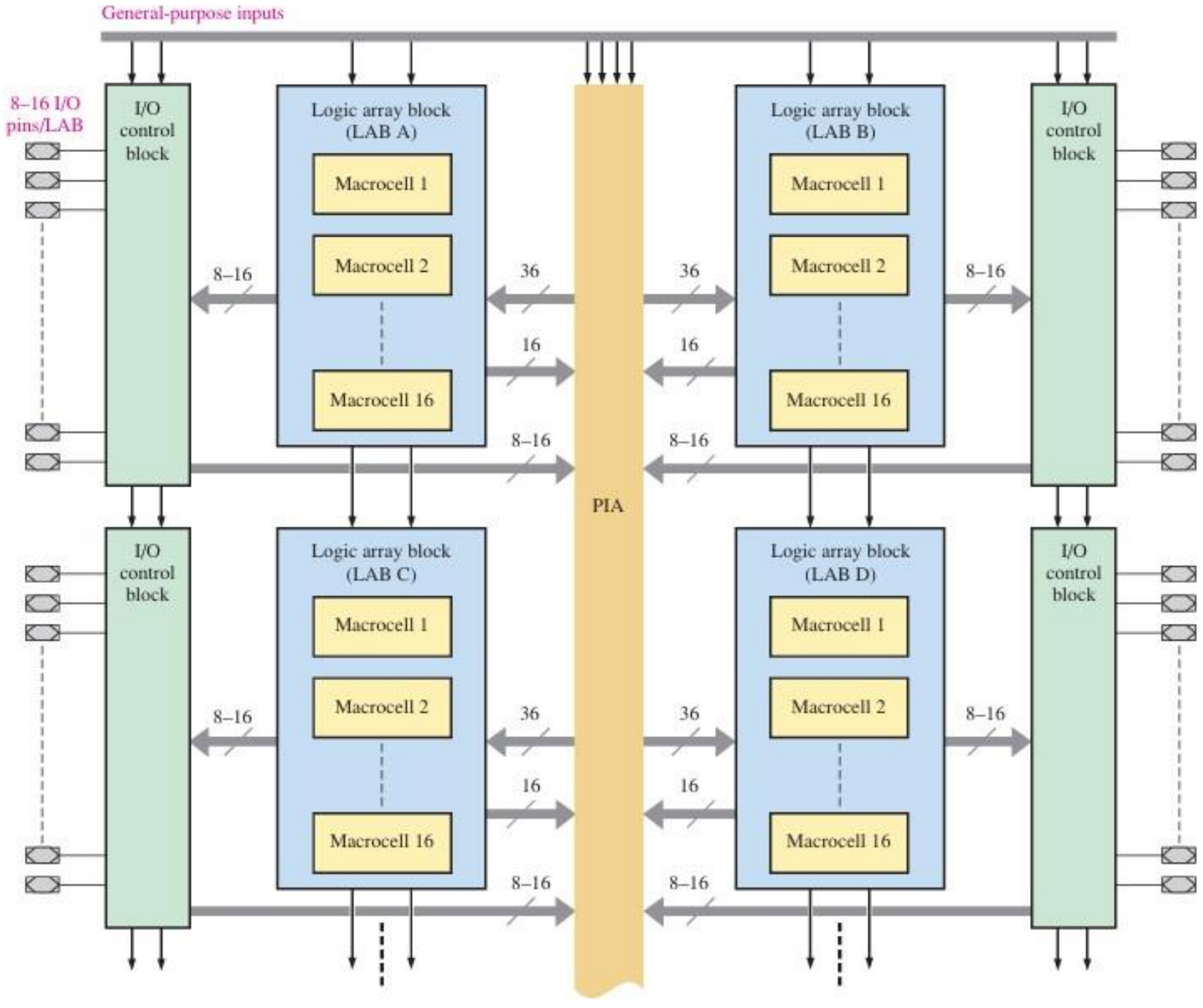
Uses:

- Control systems
- Medium-sized finite state machines
- Address decoding
- Glue logic between processors and peripherals

CPLD Architecture

- The architecture of a CPLD is the way in which the internal elements are organized and arranged.
- A general block diagram of a typical CPLD is shown.
- It has the classic PAL/GAL structure that produces SOP functions.
- Four LABs are shown, but there can be up to sixteen.
- Each of the four LABs consists of sixteen macrocells, and multiple LABs are linked together via the PIA.
- PIA is a programmable global (goes to all LABs) bus structure to which the general-purpose inputs, the I/Os, and the macrocells are connected.

CPLD Architecture



EXPANATION Of
CPLD Architecture Block Diagram

1: Inputs Enter the CPLD

At the top, the **general-purpose inputs** enter the chip.
These signals can be sent to any logic block through the PIA

2: PIA Distributes Signals

The **PIA (Programmable Interconnect Array)** is the large vertical block in the center.
Its job is to route signals:

- From inputs to LABs
- Between LABs
- From LABs to outputs

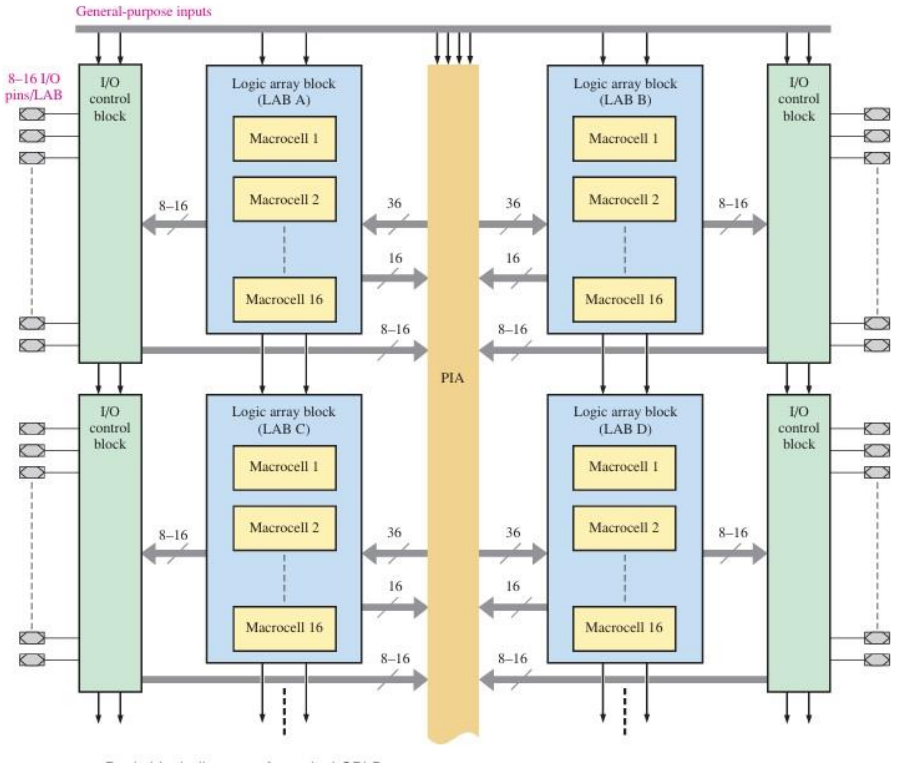
Think of it as the **traffic network** of the CPLD.

EXPANATION Of
CPLD Architecture Block Diagram

3: LABs Perform Logic

The CPLD contains four **Logic Array Blocks**:

- LAB A
- LAB B
- LAB C
- LAB D



Each LAB contains **16 macrocells**.

The macrocells implement your Boolean equations, counters, state machines, registers, etc.

EXPANATION Of
CPLD Architecture Block Diagram

4: LABs Exchange Information

If a macrocell in LAB A produces a signal that LAB D needs:
The PIA carries that signal between the blocks.

5: Outputs Go to I/O Control Blocks

After the logic is computed, the results are sent to the **I/O Control Blocks** located around the edges.
These blocks:

- Connect internal logic to external pins
- Configure pins as input/output
- Control output drivers

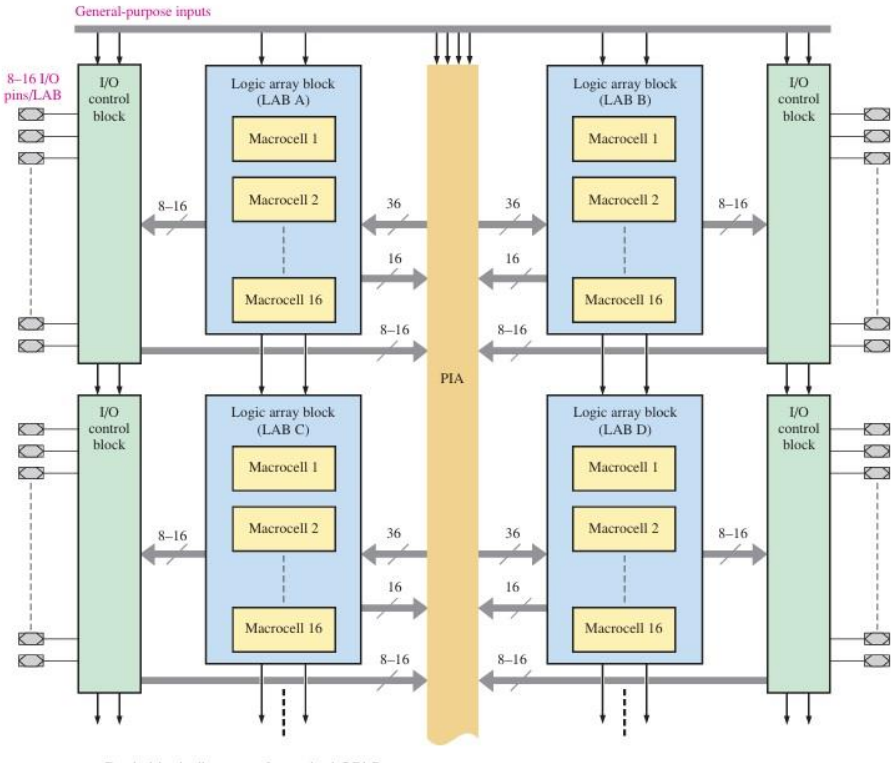
EXPANATION Of
CPLD Architecture Block Diagram

6: Signals Can Be Fed Back

Notice some arrows returning from the LABs to the PIA.

This feedback is essential for:

- Counters
- Registers
- Finite State Machines (FSMs)



MACROCELL

A macrocell is the basic programmable building block inside a CPLD's Logic Array Block (LAB).

So:

- **CPLD** consists of multiple **LABs**
- Each **LAB** consists of multiple **macrocells**
- Each **macrocell** implements one logic function or output

MACROCELL

Main components:

1.Combinational Logic

- Implements Boolean expressions such as:
- F=AB+CD

2. Flip-Flop (Optional)

- Stores a bit.
- Used for sequential circuits, counters, registers, state machines, etc.

3. Multiplexer

- Selects either:
  - Direct combinational output, or
  - Registered (flip-flop) output

4. Output Control

- Can send the signal to:
  - An output pin
  - Another macrocell through the PIA

3. Field Programmable Gate Array

3. Field Programmable Gate Array

An FPGA is a programmable integrated circuit containing a large array of configurable logic blocks and programmable interconnections that can be configured by the user to perform different digital tasks.

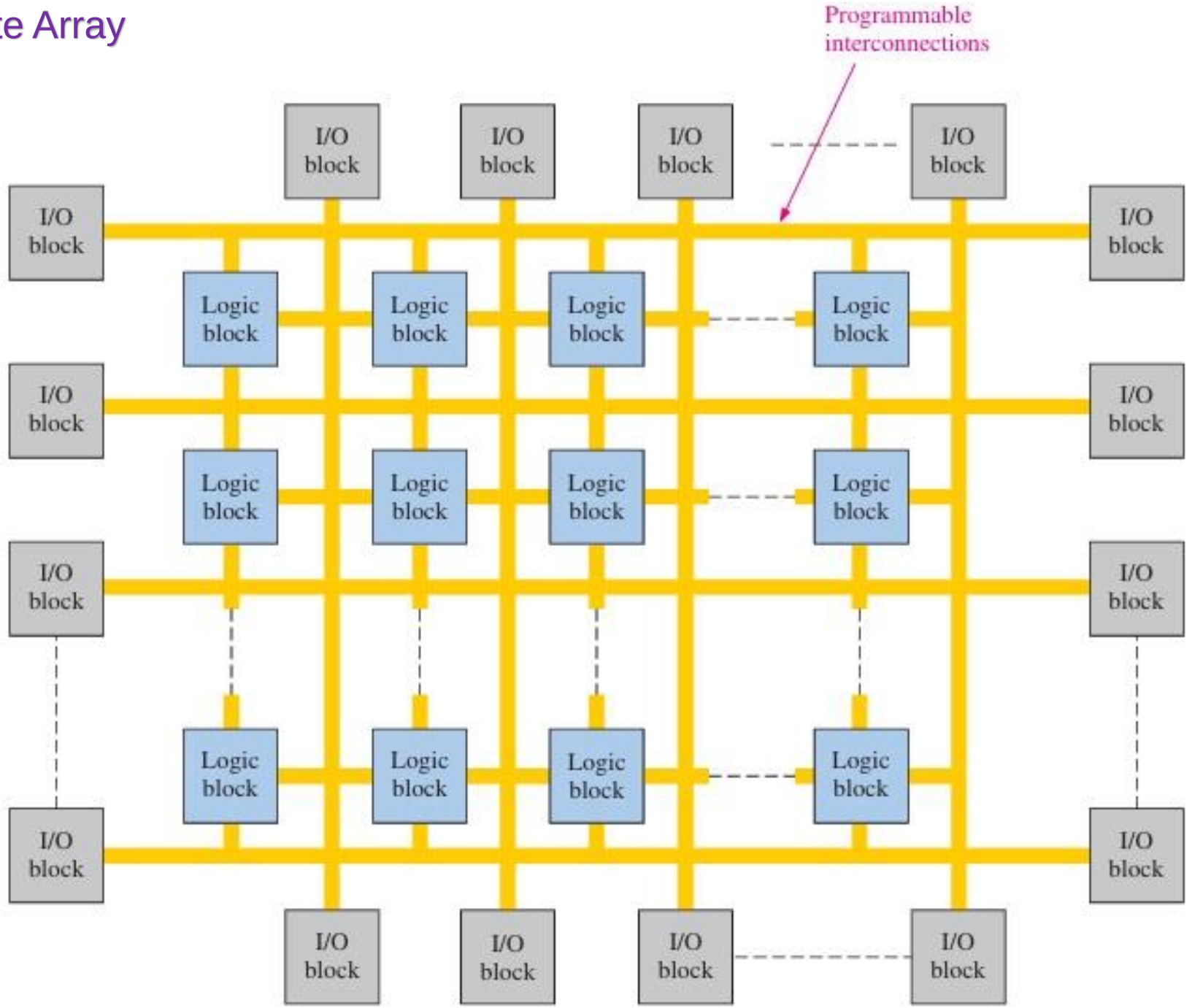
- An **FPGA** is the largest and most powerful programmable logic device.
- It contains thousands or millions of small configurable logic blocks connected by programmable routing.

Basic Structure of an FPGA

An FPGA consists of three main parts:

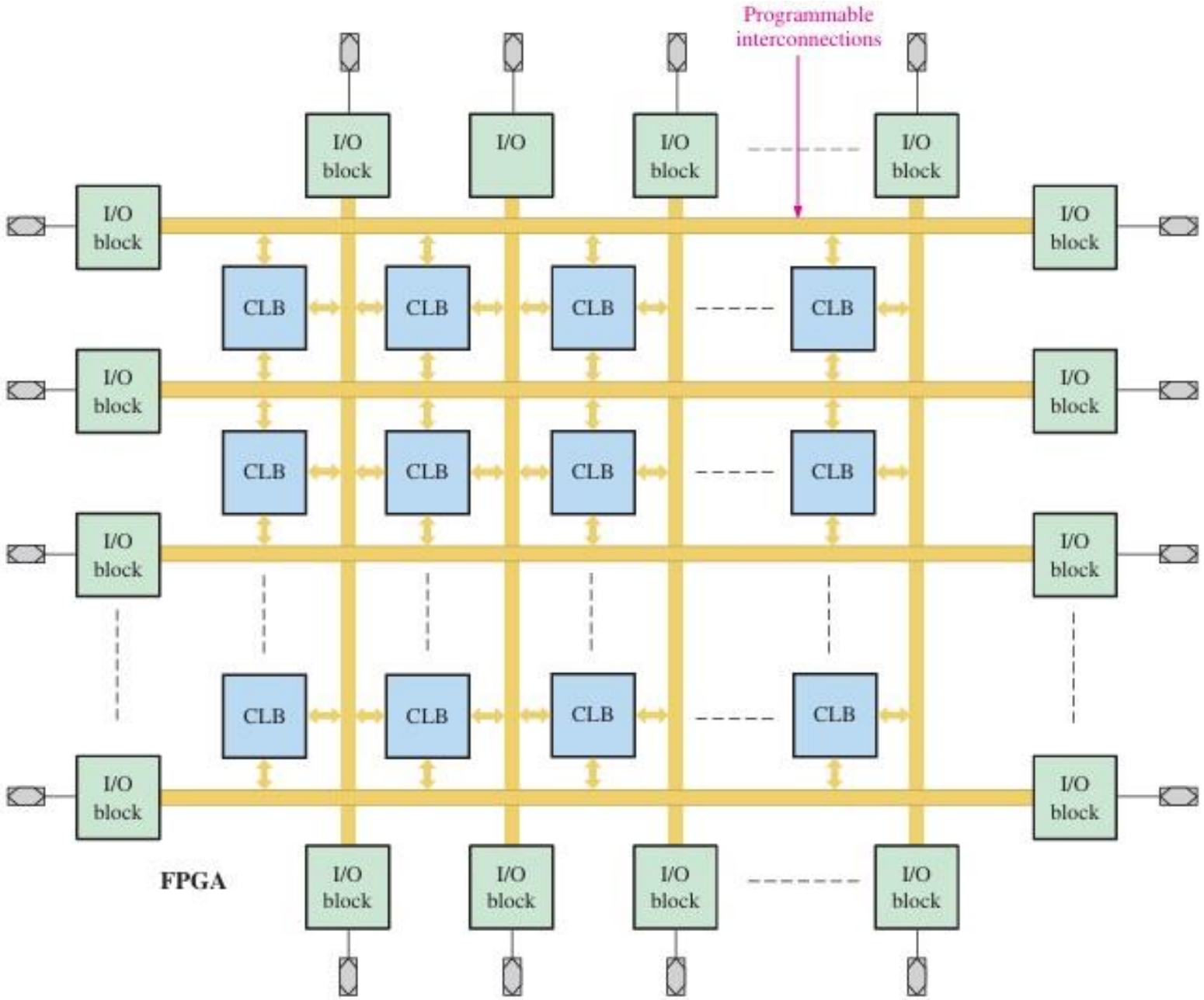
1. **Configurable Logic Blocks (CLBs)**
   - These are the small logic-building units.
   - They perform logic operations such as AND, OR, XOR, addition, counting, etc.
   - A large FPGA contains many CLBs.
2. **Programmable Interconnections**
   - These are programmable wires that connect the CLBs together.
   - The user decides how the blocks should be connected.
3. **Input/Output (I/O) Blocks**
   - Located around the edges of the FPGA.
   - Connect the FPGA to external devices such as switches, sensors, LEDs, displays, memories, and processors.

3. Field Programmable Gate Array



Basic Structure of an FPGA

3. Field Programmable Gate Array



Basic Structure of an FPGA

What is a LUT?

A **LUT (Look-Up Table)** is a small memory inside a logic block. Instead of building logic gates directly, the FPGA stores the desired output values in the LUT.

A	B	Output
0	0	0
0	1	1
1	0	1
1	1	0

This table represents an XOR gate. The LUT simply looks up and outputs the correct value.

Characteristics:

- Extremely large capacity
- Can implement entire digital systems
- Contains:
  - Logic blocks
  - Flip-flops
  - RAM blocks
  - DSP blocks
  - Clock-management circuits

Uses:

- CPU design
- Digital signal processing
- Artificial intelligence accelerators
- Video processing
- Communication systems

How Does an FPGA Work?

Step 1:

You write a digital circuit using **VHDL**, **Verilog**, or a schematic.

Step 2:

FPGA software converts your design into a configuration file.

Step 3:

The configuration file is loaded into the FPGA.

Step 4:

The FPGA configures:

- the logic inside CLBs,
- the connections between CLBs,
- the I/O pins.

Step 5:

The FPGA now behaves like the circuit you designed.

Introduction to PLDs — Lecture Summary

Logic Devices

Fixed-Function ICs
- Logic permanently set by maker
- Simple, cheap, fast
- Cannot be modified (e.g., 7408)

Programmable Logic Devices
- Internal connections user-defined
- Reprogrammable without hardware swap
- Flexible, fewer ICs needed

Types of PLDs

SPLD
- Simplest PLD — PAL, PLA, GAL
- Implements Boolean SOP/POS
- PAL: programmable AND, fixed OR
- GAL: reprogrammable (EEPROM)

CPLD
- Multiple SPLDs (LABs) + PIA bus
- Non-volatile, predictable timing

FPGA
- Largest: CLBs + Interconnects + I/O
- Uses LUTs; VHDL/Verilog design

Key Concepts

Programming Technologies
- Fuse — one-time programmable (OTP)
- EPROM — mostly OTP
- SRAM — fully reprogrammable

CPLD Internals
- LABs hold macrocells (SOP logic)
- PIA routes signals between LABs
- Macrocell = AND logic + Flip-flop + MUX

FPGA Internals
- CLBs contain LUTs (truth-table memory)
- Programmed via VHDL or Verilog
- Applications: DSP, AI, CPU design

Thank You