

Topic 16

Programmable Logic Device

Outline

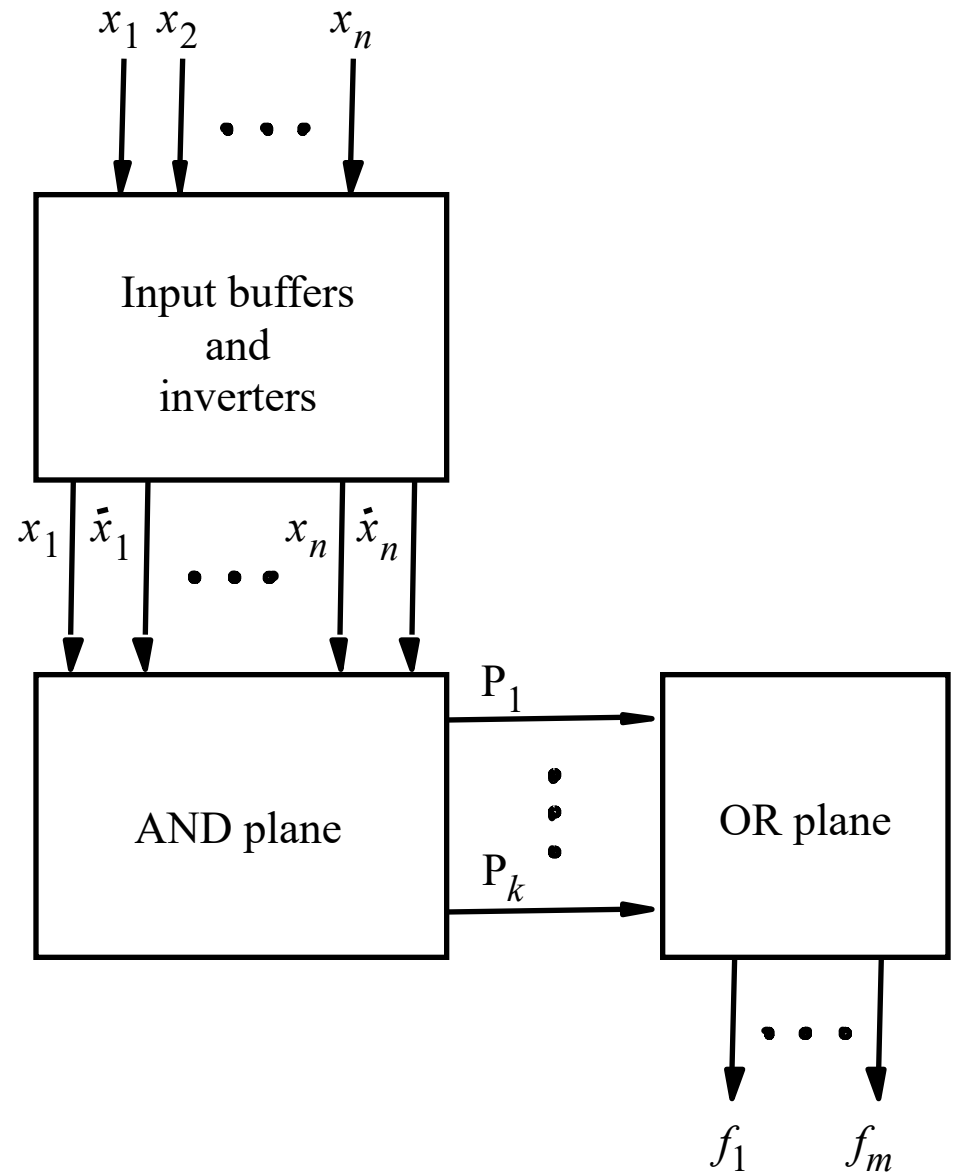
- Programmable Logic Devices (PLD)
- Programmable Logic Array (PLA)
- Programmable Array Logic (PAL)
- Complex Programmable Logic Device (CPLD)
- Field Programmable Gate Array (FPGA)

Programmable Logic Devices

- **PLD**
 - First introduced in 1970s
 - Can be viewed as a “black box” containing logic gates and programmable switches
 - The logic gates and programmable switches can be customized to implement specific logic circuit
 - Simple programmable logic devices (SPLD)
 - Programmable logic array (PLA)
 - Programmable array logic (PAL)
 - Complex programmable logic array (CPLD)
 - Field-programmable gate array (FPGA)

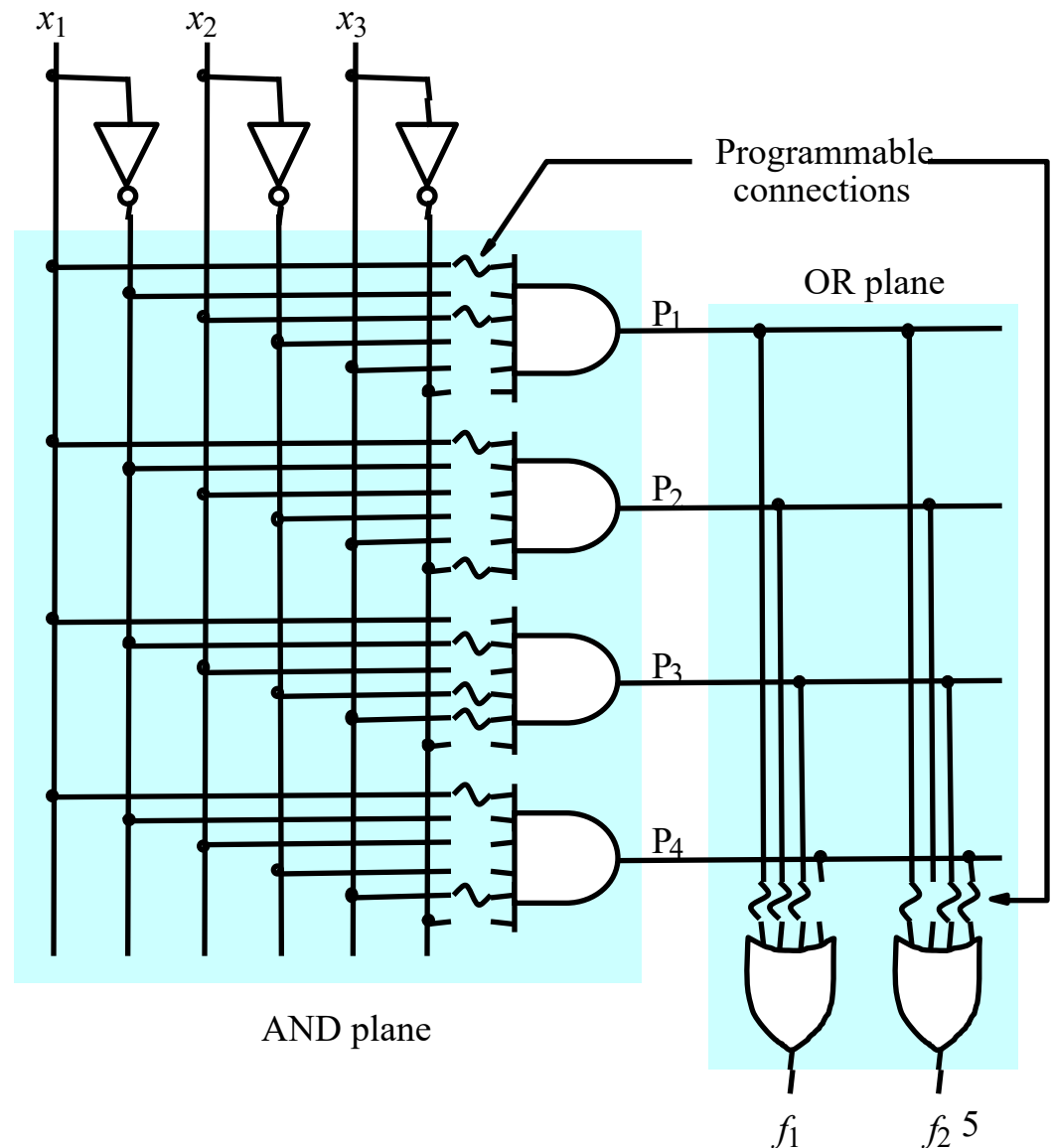
Programmable Logic Array (PLA)

- Comprises a collection of buffers, inverters, AND gates, OR gates
- Can be used to realize logic circuit in sum-of-products (SOP) form,
- Example:
$$f = x'yz + xy'z'$$



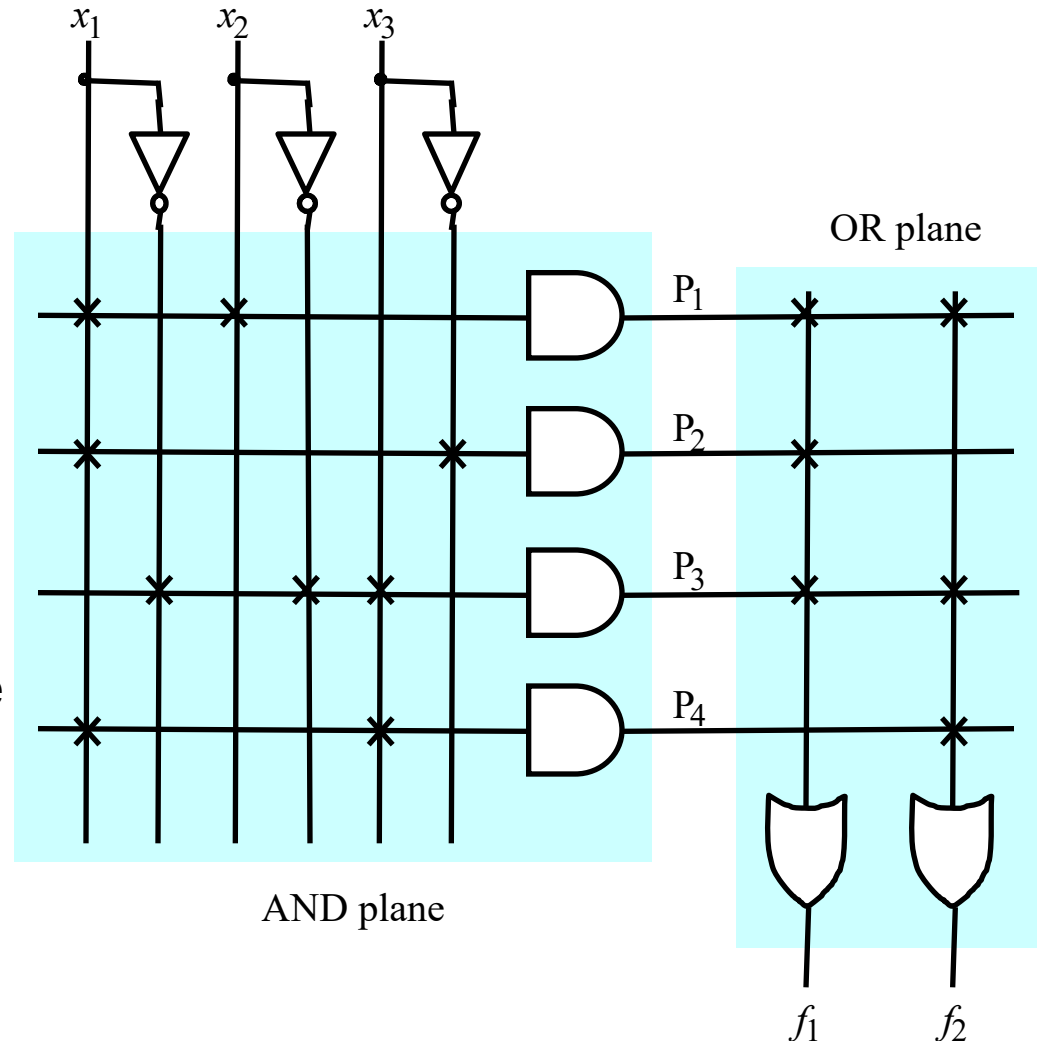
Programmable Logic Array (PLA)

- Buffers and inverters provide both true value and complement of each input
- AND plane provides the product terms
- OR plane provides the sum of the product terms
- Example:
 - $P_1 = x_1x_2$
 - $P_2 = x_1x_3'$
 - $P_3 = x_1'x_2'x_3$
 - $P_4 = x_1x_3$
 - $F_1 = P_1 + P_2 + P_3$
 - $F_2 = P_1 + P_3 + P_4$



Programmable Logic Array (PLA)

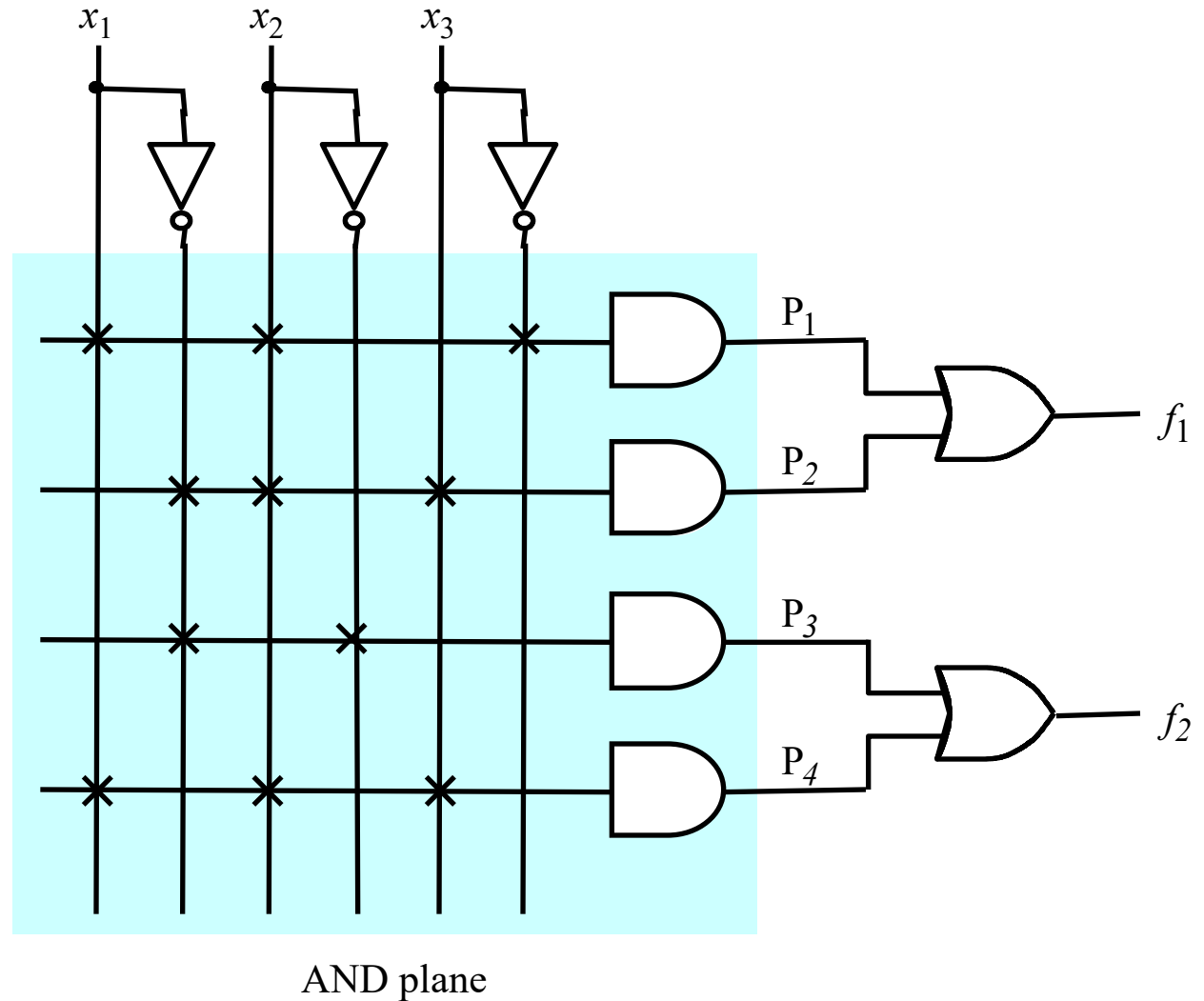
- Each AND gate has $2 \times N$ inputs
 - N , number of primary inputs
- Each OR gate has M inputs
 - M , number of and gates
- **Problem:** size of the inputs
- Commercially available PLAs typically have:
 - 16 inputs
 - 32 AND gates
 - 8 OR gates
- Connections replaced by single lines, “x” indicates a connected input to the gate



Programmable Array Logic (PAL)

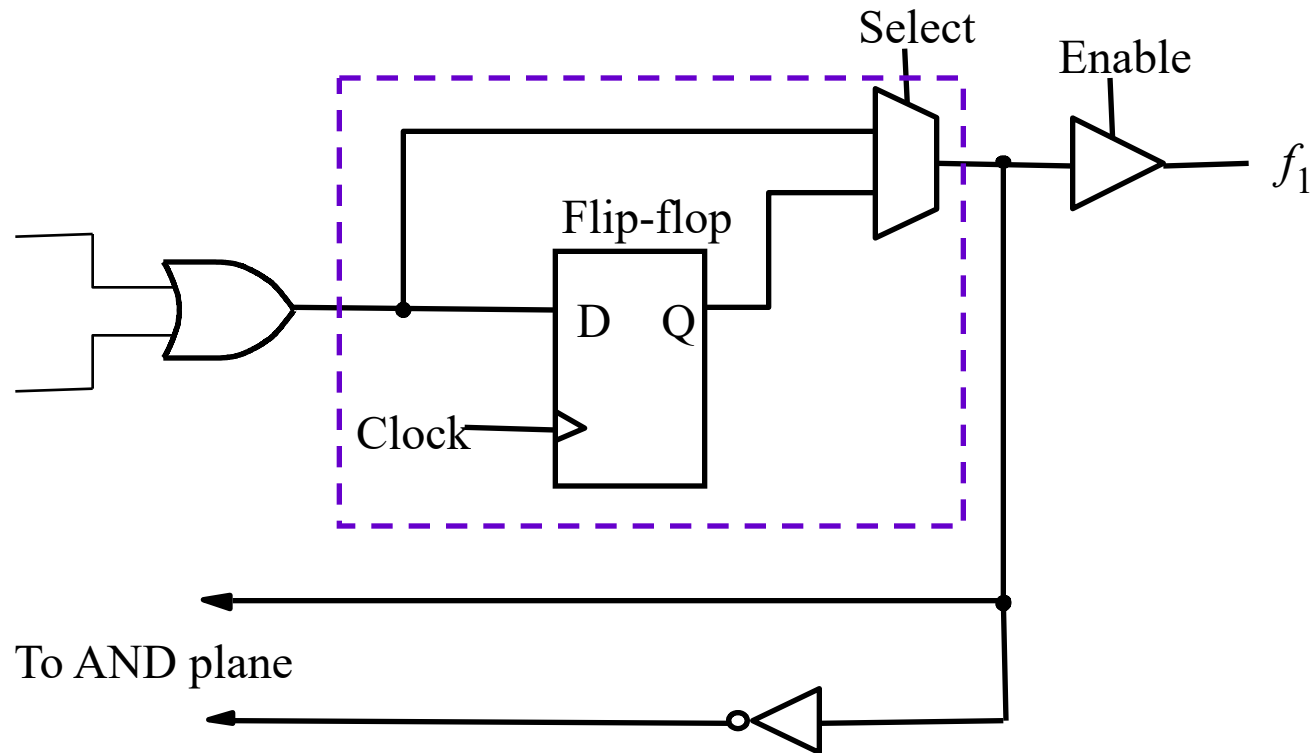
- Drawbacks of PLA
 - Hard to fabricate correctly due to the programmable connections
 - Special implementation of the programmable connections reduce the speed of circuits in PLA
- Solution: fix the OR plane – PAL
 - Less expensive
 - Better performance
 - Became popular in practical applications

Programmable Array Logic (PAL)



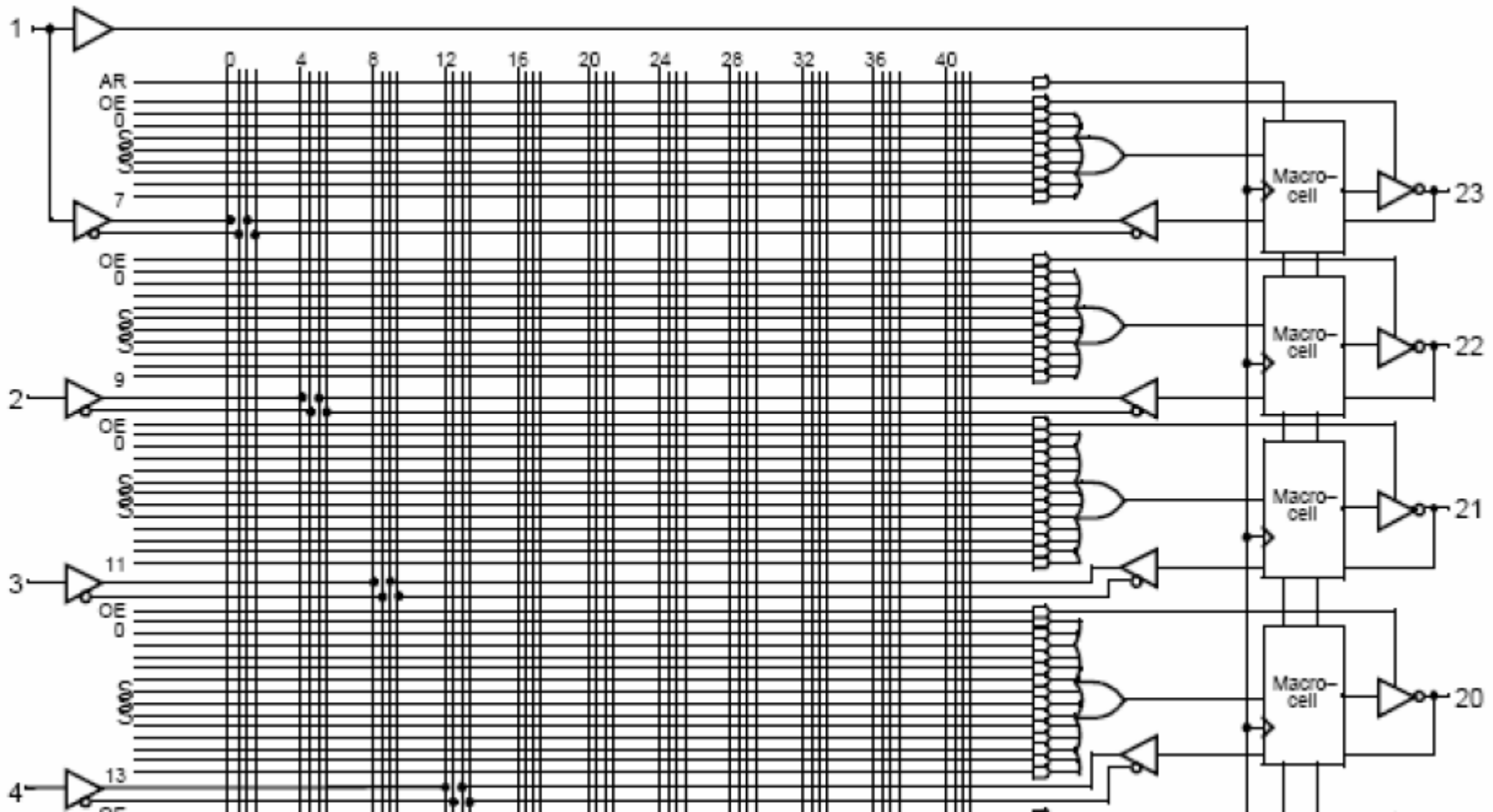
Programmable Array Logic (PAL)

- In order to provide additional flexibility, an extra circuit is inserted between the OR output and the chip pin - *Macrocell*



PAL Example

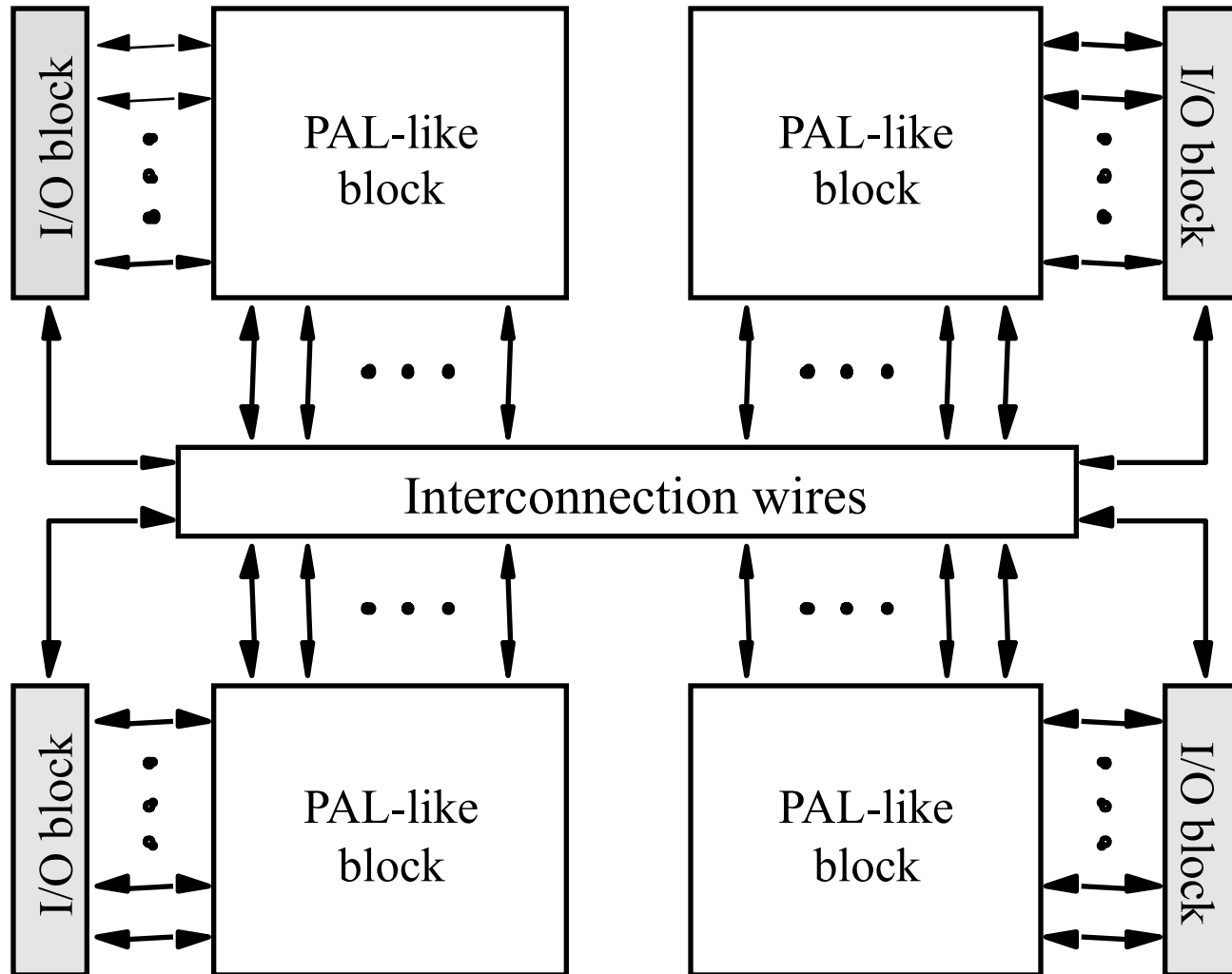
- Compensate for the reduced flexibility
 - Various numbers of inputs to the OR gates

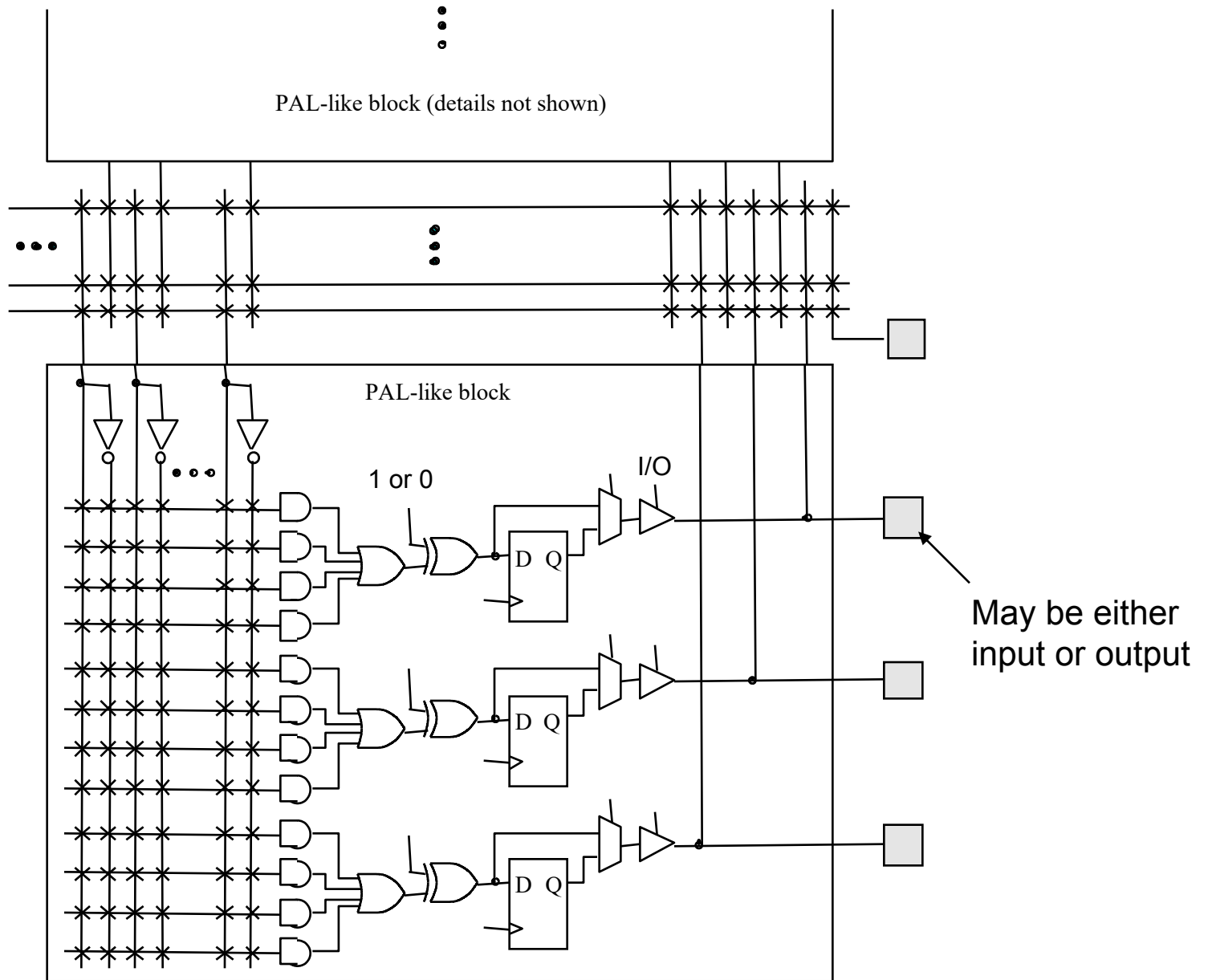


Complex Programmable Logic Devices (CPLD)

- Composed of multiple PAL/PLA-like circuit blocks
 - Blocks are connected through a set of interconnection wires
 - Blocks are connected to the IC chip pins through a set of I/O blocks
 - Number of blocks may vary from 2 to over 100
- Provides more inputs and outputs
- Provides more flexibility
- May accommodate bigger size circuit

CPLD





Field Programmable Gate Array (FPGA)

- First introduced by Xilinx in 1985
- Most FPGA providers are "fabless", allows
 - focus on device capability
 - improvement of design software
 - offering IP cores

Types of FPGA

- Reprogrammable
 - SRAM-based FPGA
 - Volatile, often the best choice for prototyping and development
 - Supports in-system-programming (ISP)
 - What we used in the labs
 - EEPROM-based (Flash-based) FPGA
- One-Time Programmable (OTP)
 - Anti-Fuse-based FPGA
 - EPROM-based FPGA

Configuration Technologies FPGA

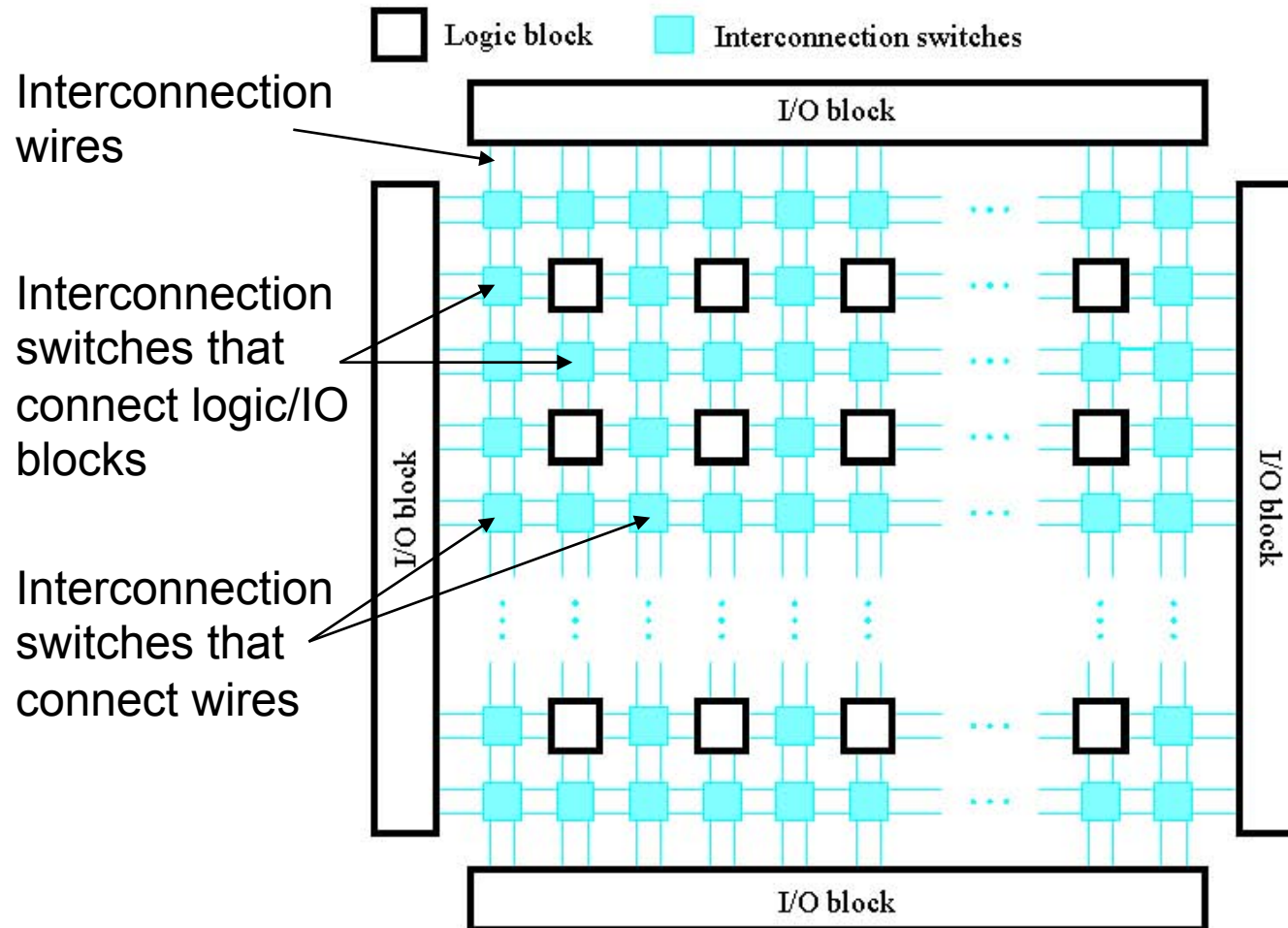
- **Configure SRAM-based FPGA**
 - ISP, fast reconfiguration, but volatile
 - External device (EEPROM, may be onboard), programming on power up
- **Configure flash-based FPGA**
 - Similar to EEPROM programming devices
 - nonvolatile, may or may not support ISP
- **Configure EPROM-based FPGA**
 - Similar to EPROM programming devices
 - nonvolatile, off-board one-time configuration
- **Configure Anti-Fuse-based FPGA**
 - Fuse burning devices (electronically or optically)
 - nonvolatile, off-board one-time configuration

Internal Structure

- Composed of logic blocks and wires
 - Configurable Logic blocks (CLB)
 - I/O blocks (IOB)
 - Interconnection wires and switches

FPGA Architecture

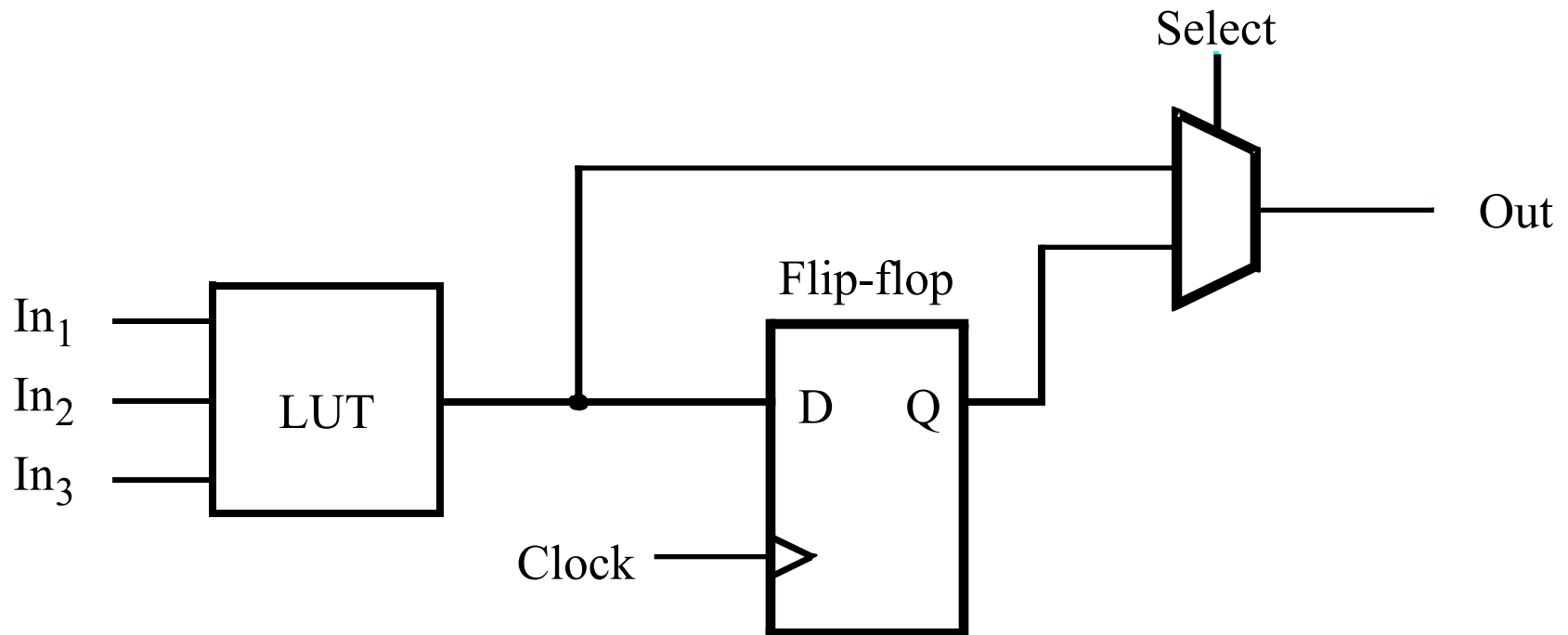
- Typical FPGA architecture



Courtesy of Xilinx Inc.

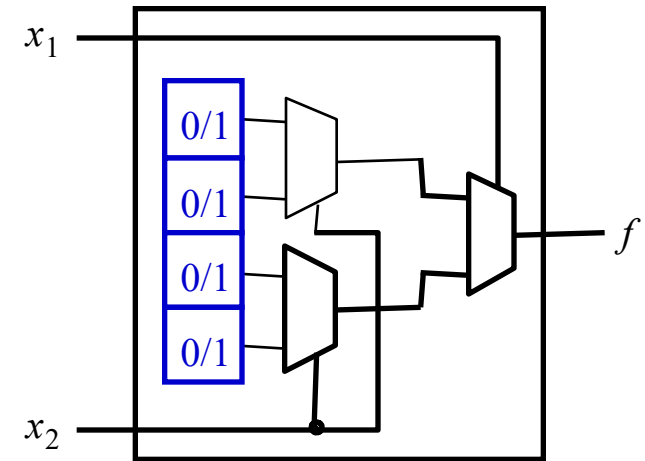
Programmable (Configurable) Logic Block

- PLB (or CLB), like in SPLD and CPLD, macrocell is added to provide more flexibility



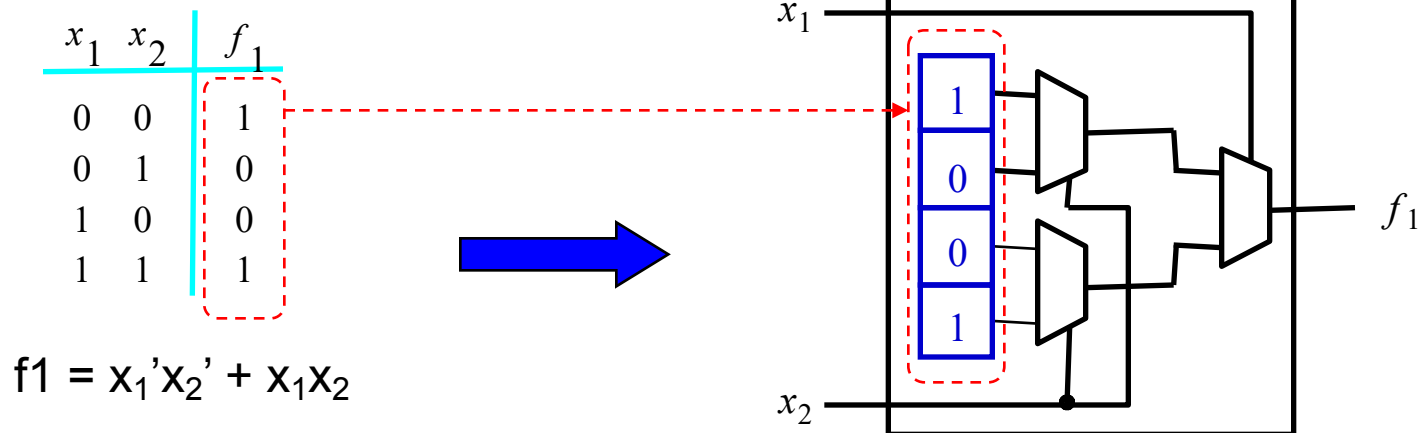
Look Up Table (LUT)

- A typical PLB has a LUT
- Each LUT contains 2^N storage cells, N is the number of inputs to the LUT
 - **SRAM** for storage cell
 - Each storage cell can hold a value, either “1” or “0”
 - The cells are programmed to implement particular logic functions
 - The cells may be reconfigured to implement a different logic function in the same LUT
- N input LUT can implement any N variable logic function



LUT Example

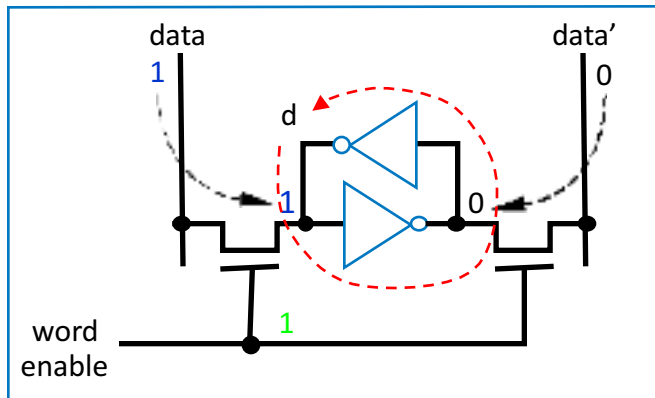
- Implement the logic function specified by the truth table using a 2-input LUT



Static RAM (SRAM) Cell

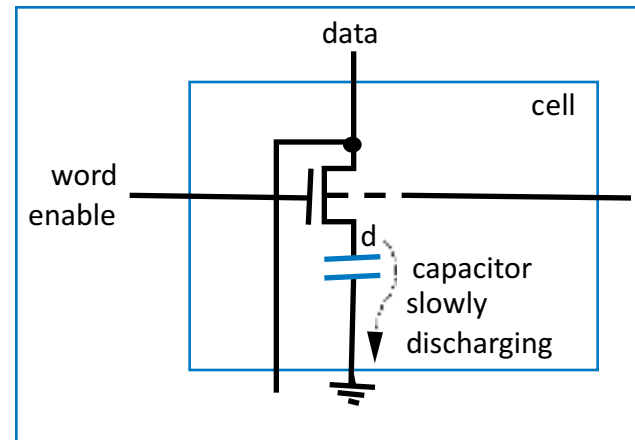
- Static Random Access Memory (SRAM) cell

SRAM cell



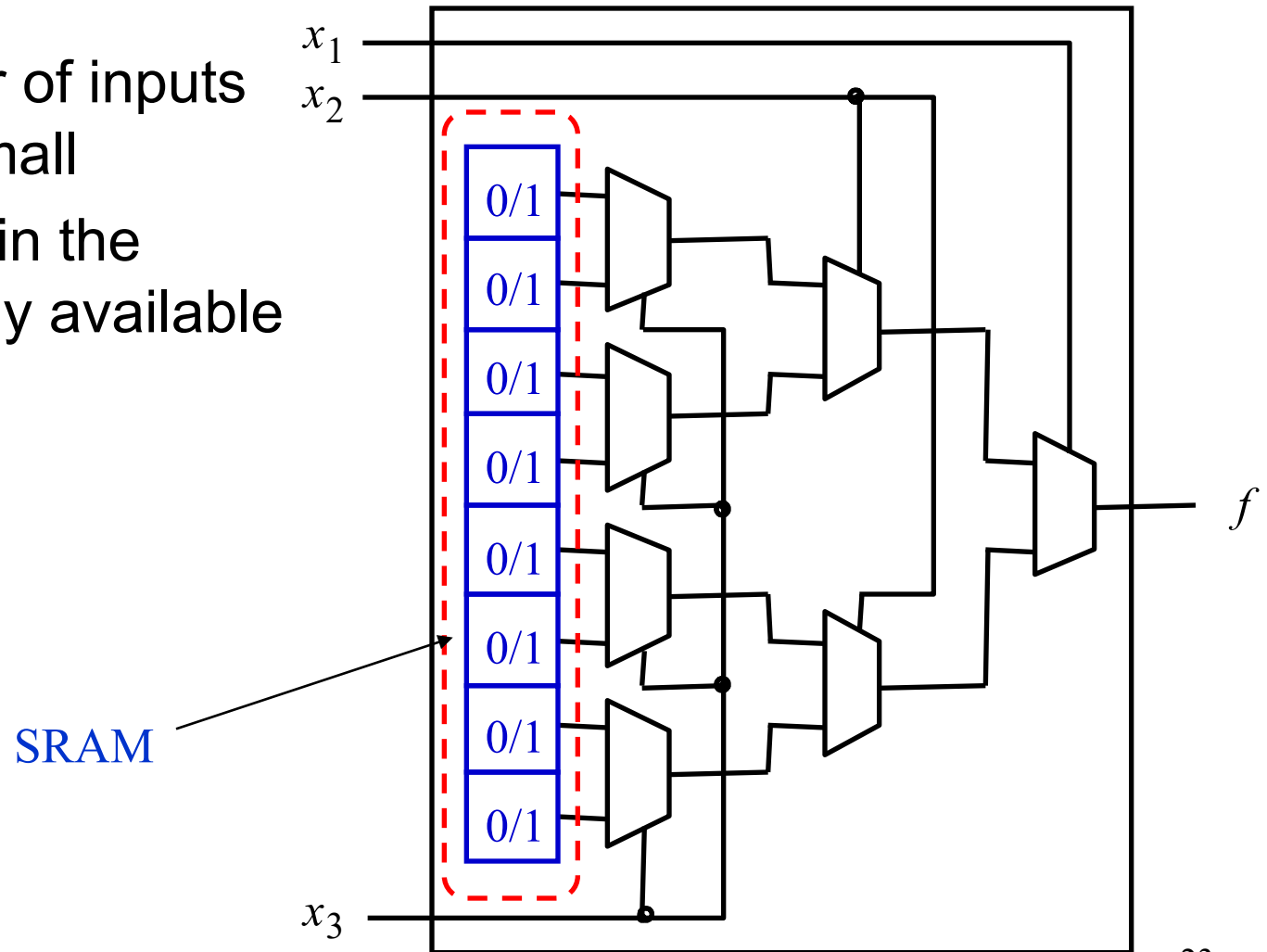
- Dynamic RAM (DRAM) cell

DRAM cell



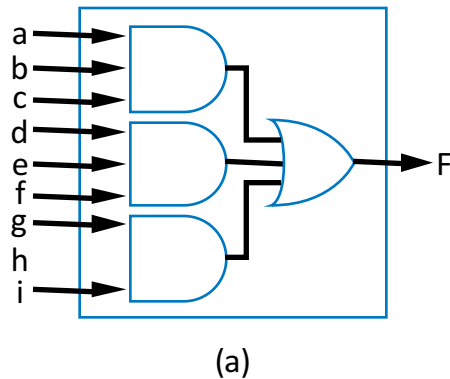
3-input LUT

- The number of inputs to LUT is small
- 4 - 6 inputs in the commercially available FPGAs

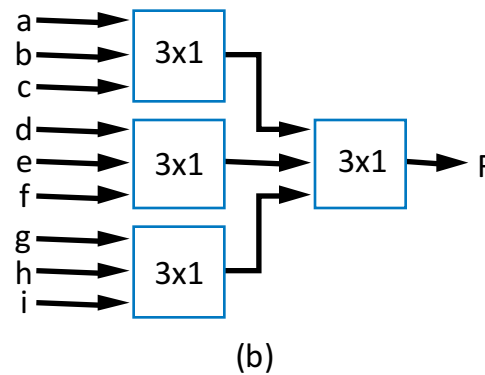


FPGA Internals: Lookup Tables (LUTs)

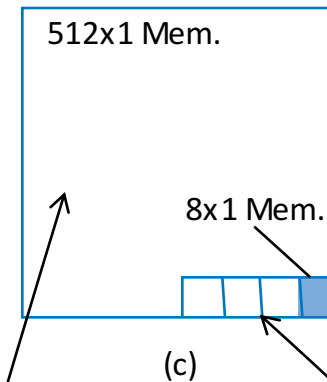
- Partitioning among smaller LUTs is more size efficient
 - Example: 9-input circuit



Original 9-input circuit



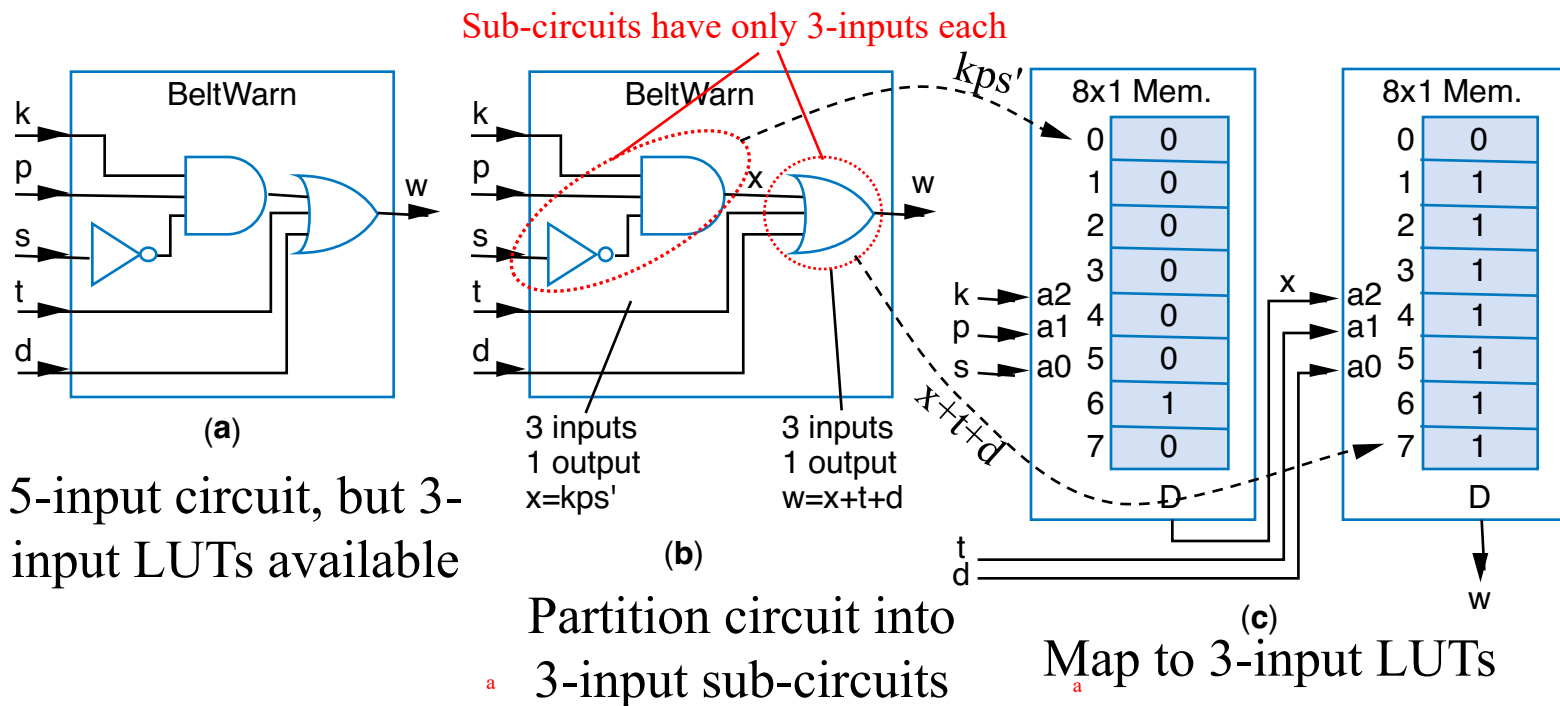
Partitioned among
3x1 LUTs



Requires only 4
3-input LUTs
(8x1 memories) –
much smaller than
a 9-input LUT
(512x1 memory)

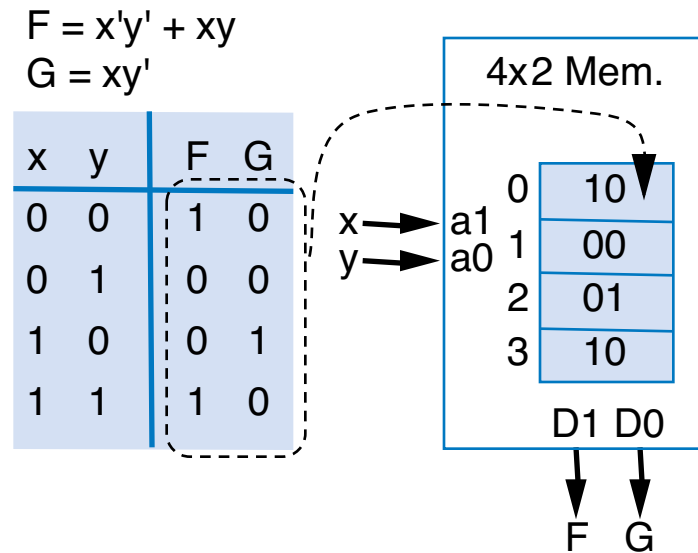
Logic Implemented with LUTs

- Circuits with more inputs



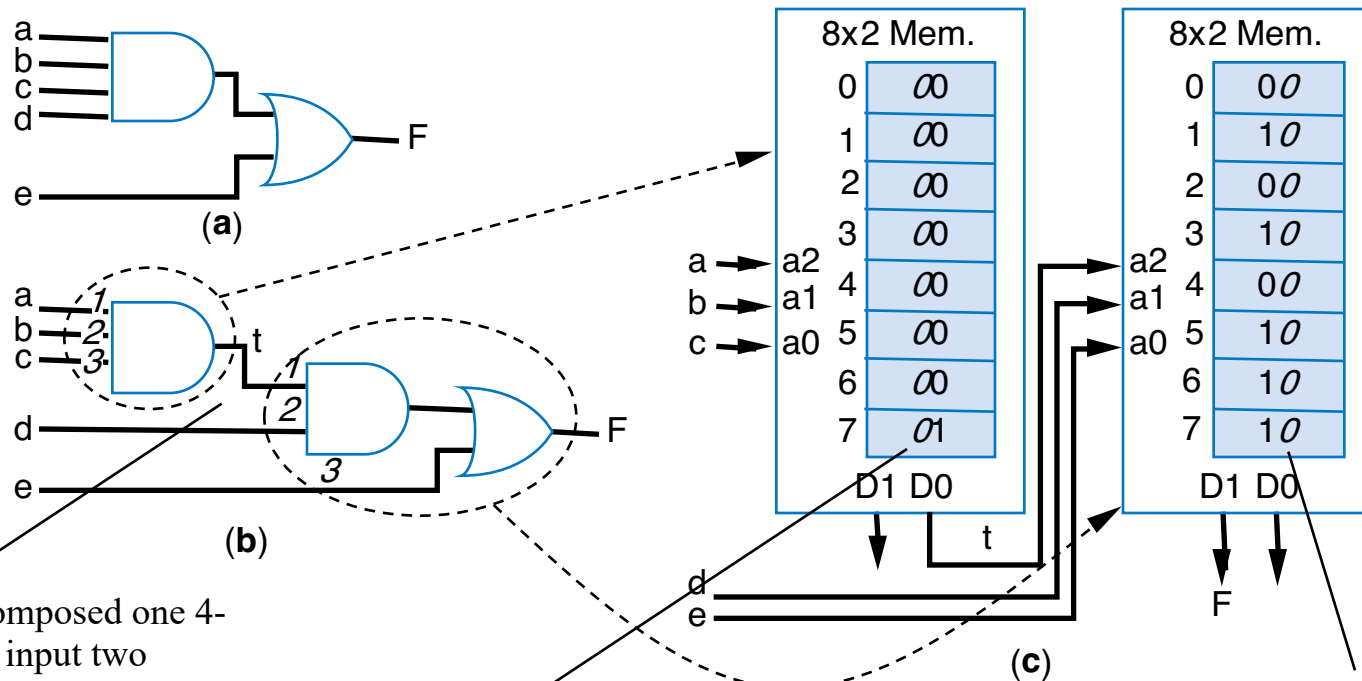
2-Output LUTs

- LUT typically has 2 (or more) outputs, not just one



2-Output LUTs

- Example: Partitioning a circuit among 3-input 2-output lookup tables



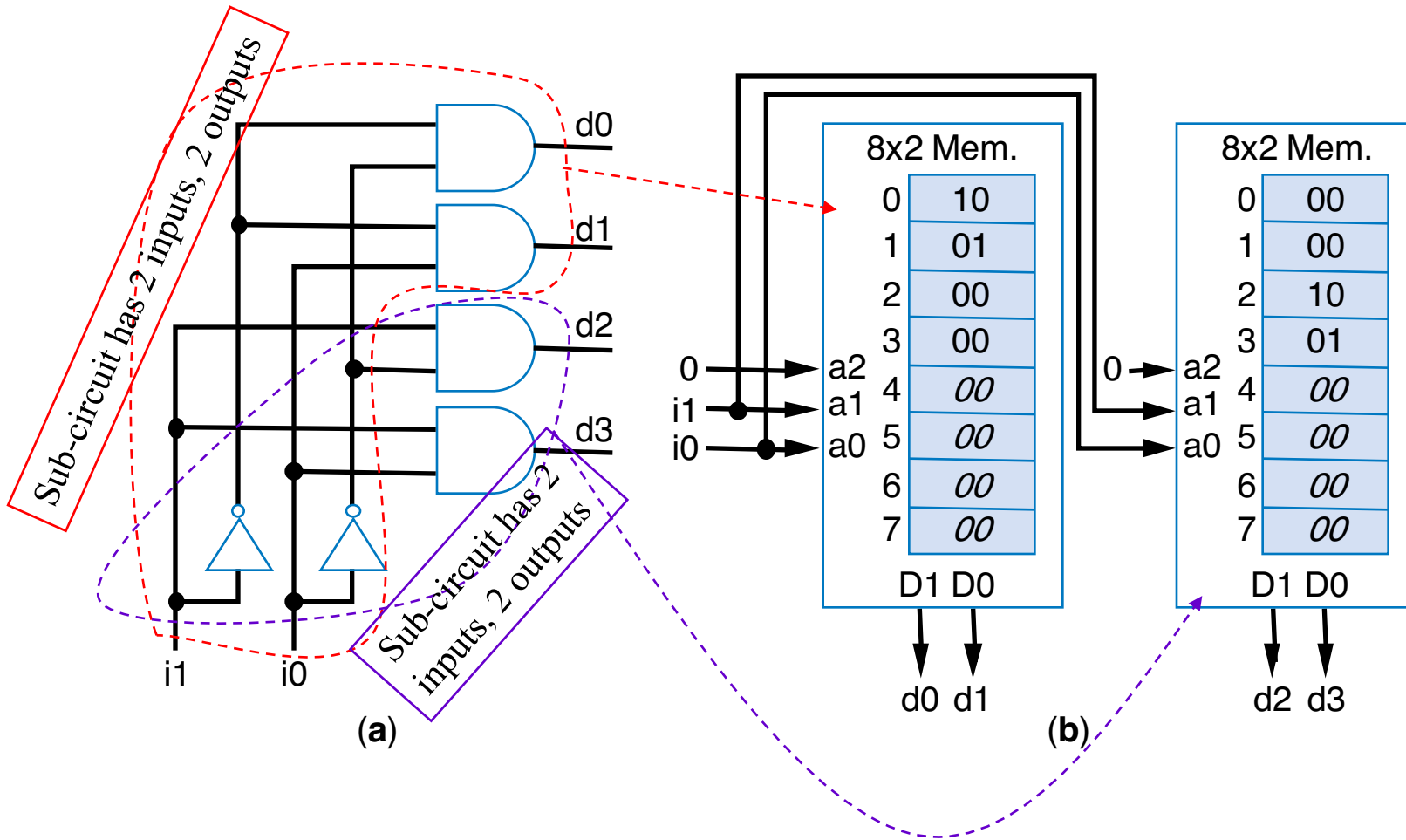
(Note: decomposed one 4-input AND into two smaller ANDs to enable partitioning into 3-input sub-circuits)

First column unused;
second column
implements AND

Second column unused;
first column implements
AND/OR sub-circuit

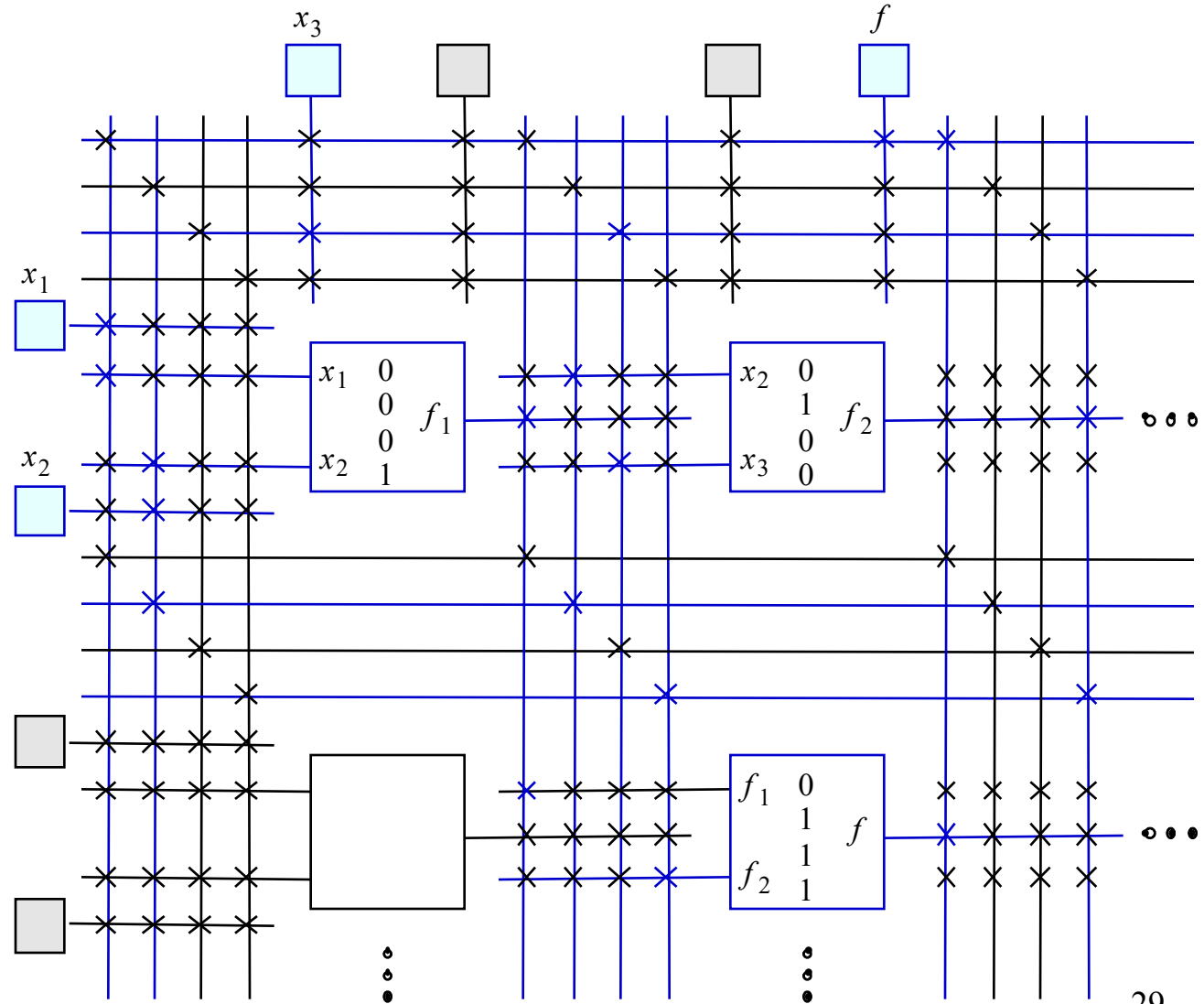
2-Output LUTs

- Example: Mapping a 2x4 decoder to 3-input 2-output LUTs

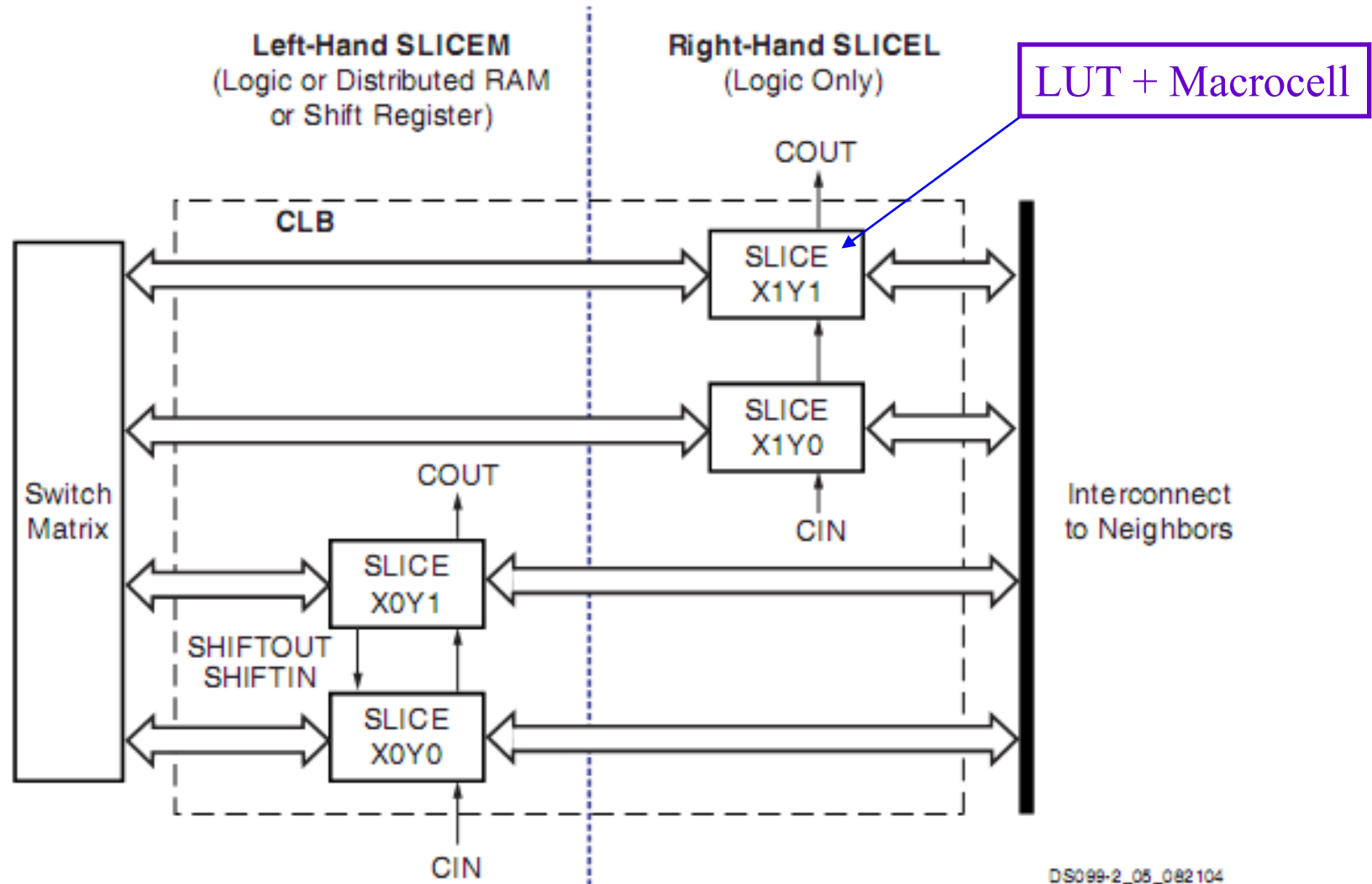


FPGA Configuration Example

- Blue x indicates a connection
- $f1 = x_1x_2$
- $f2 = x_2'x_3$
- $f = f1 + f2$

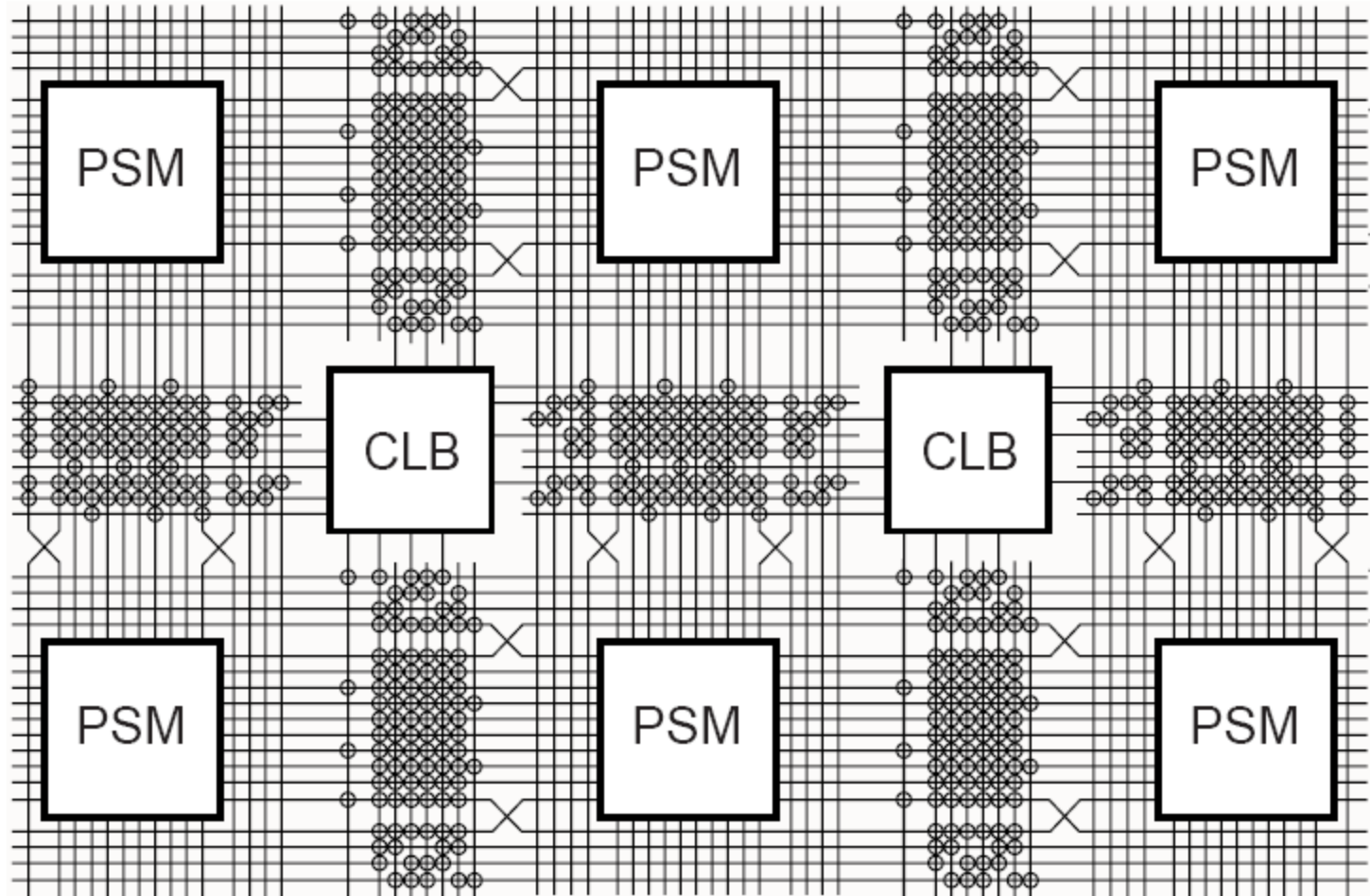


Xilinx Spartan 3 FPGA CLB Structure



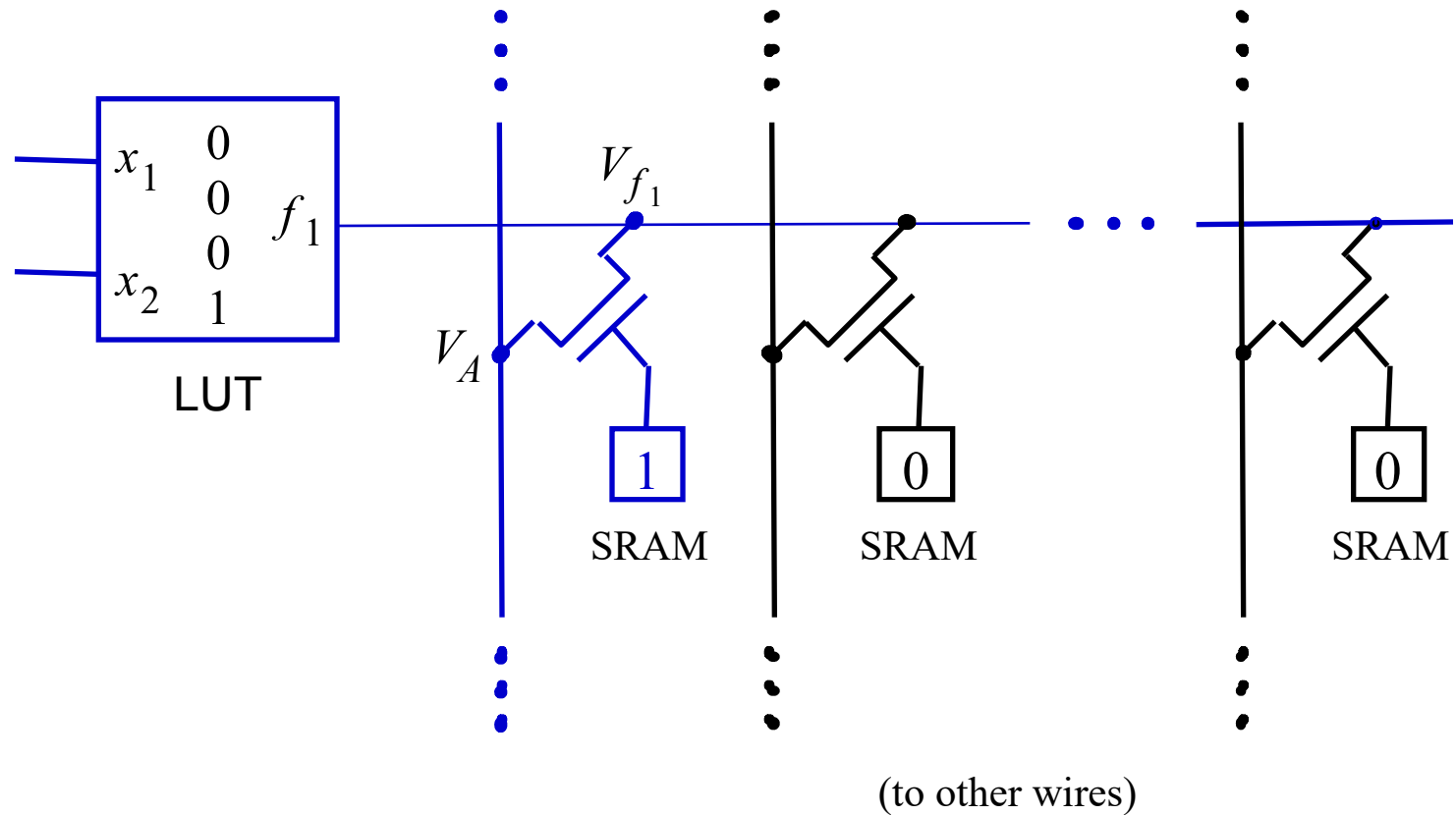
Courtesy of Xilinx Inc.

Xilinx FPGA Switches and Wires

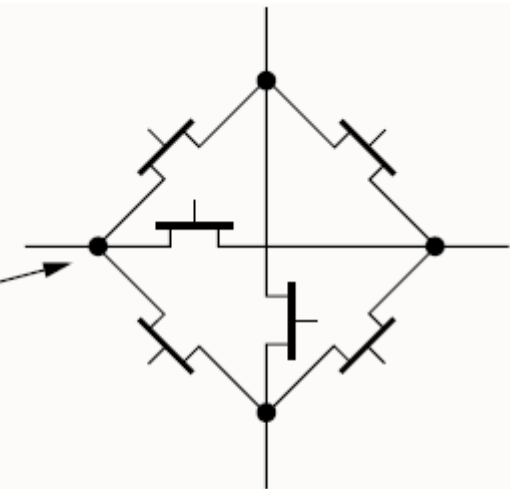
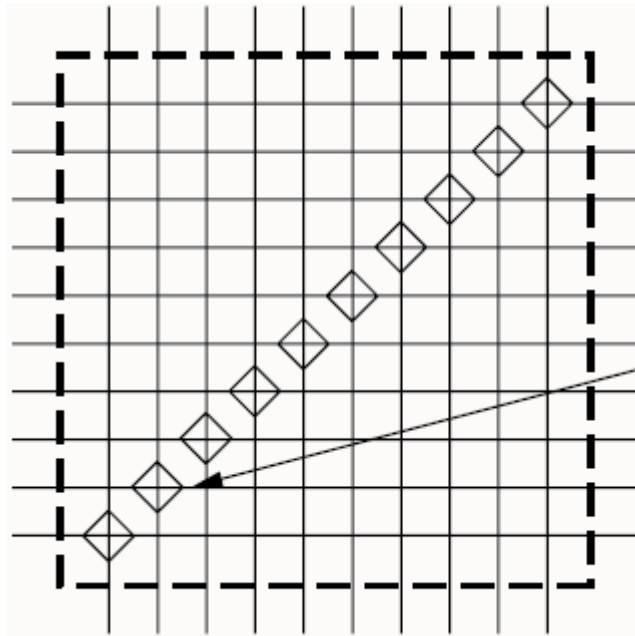


Courtesy of Xilinx Inc.

Programmable Switch

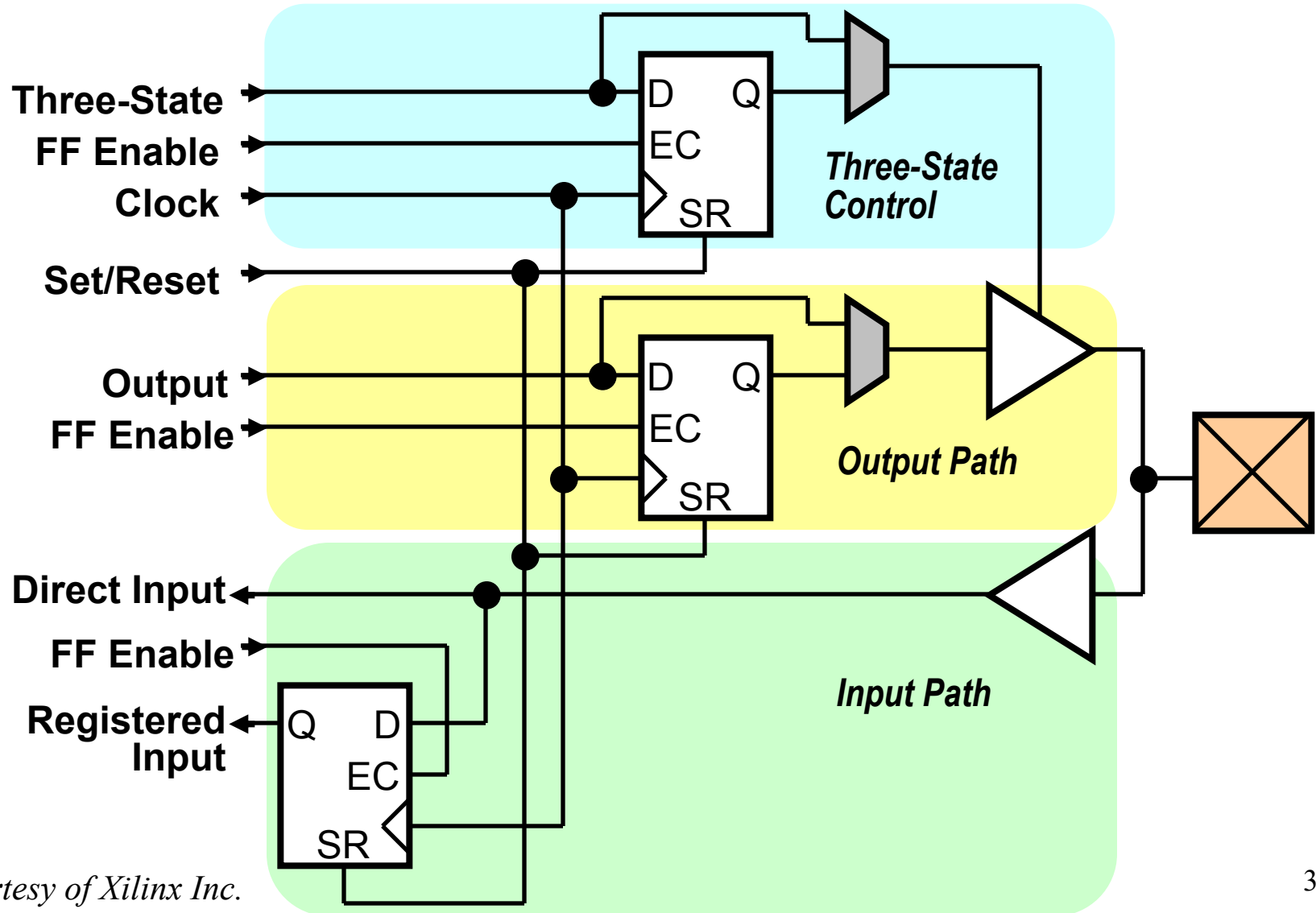


Xilinx FPGA PSM



Programmable Switch Matrix (PSM)

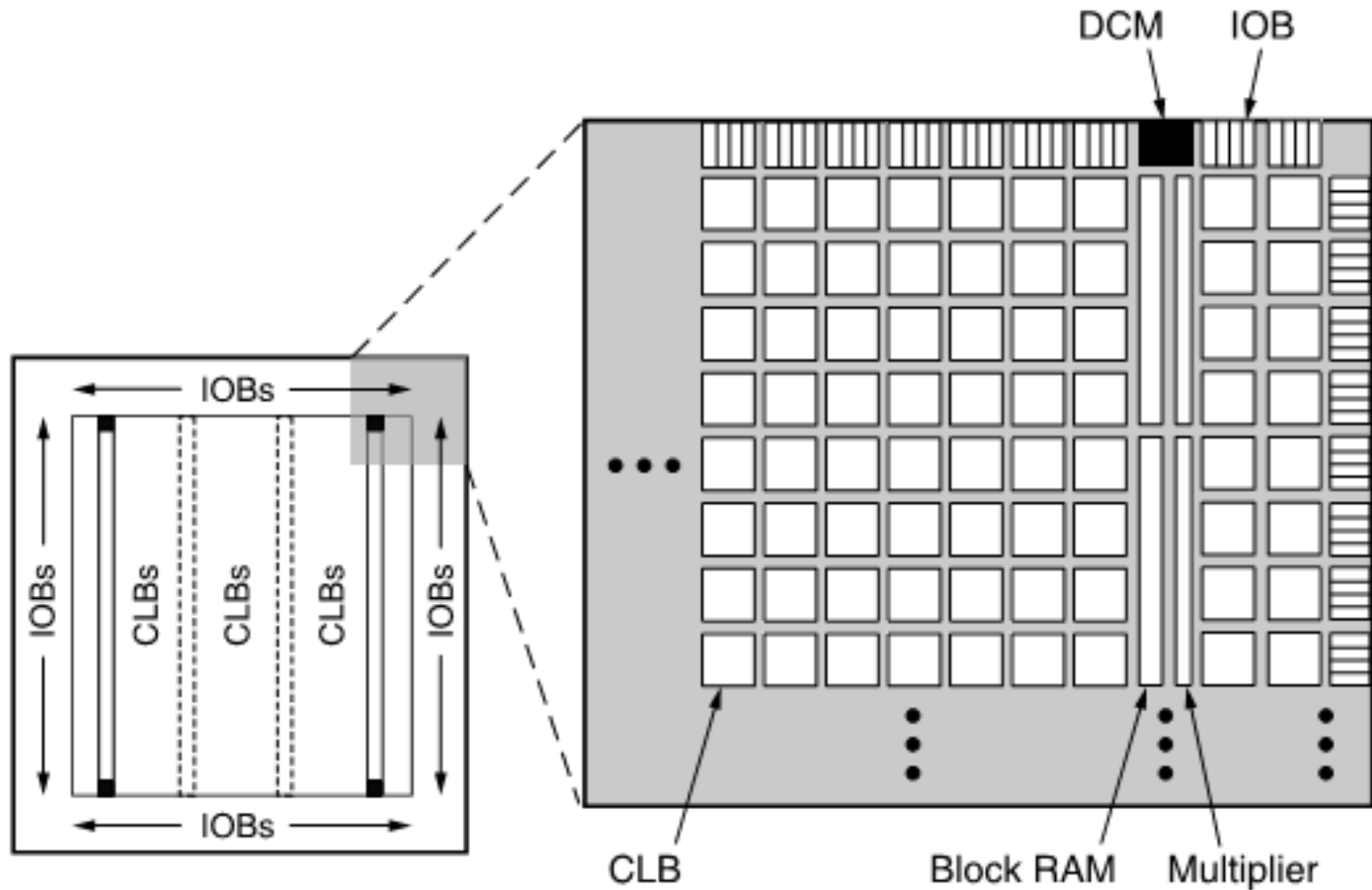
Basic I/O Block Structure



Advanced FPGA Features

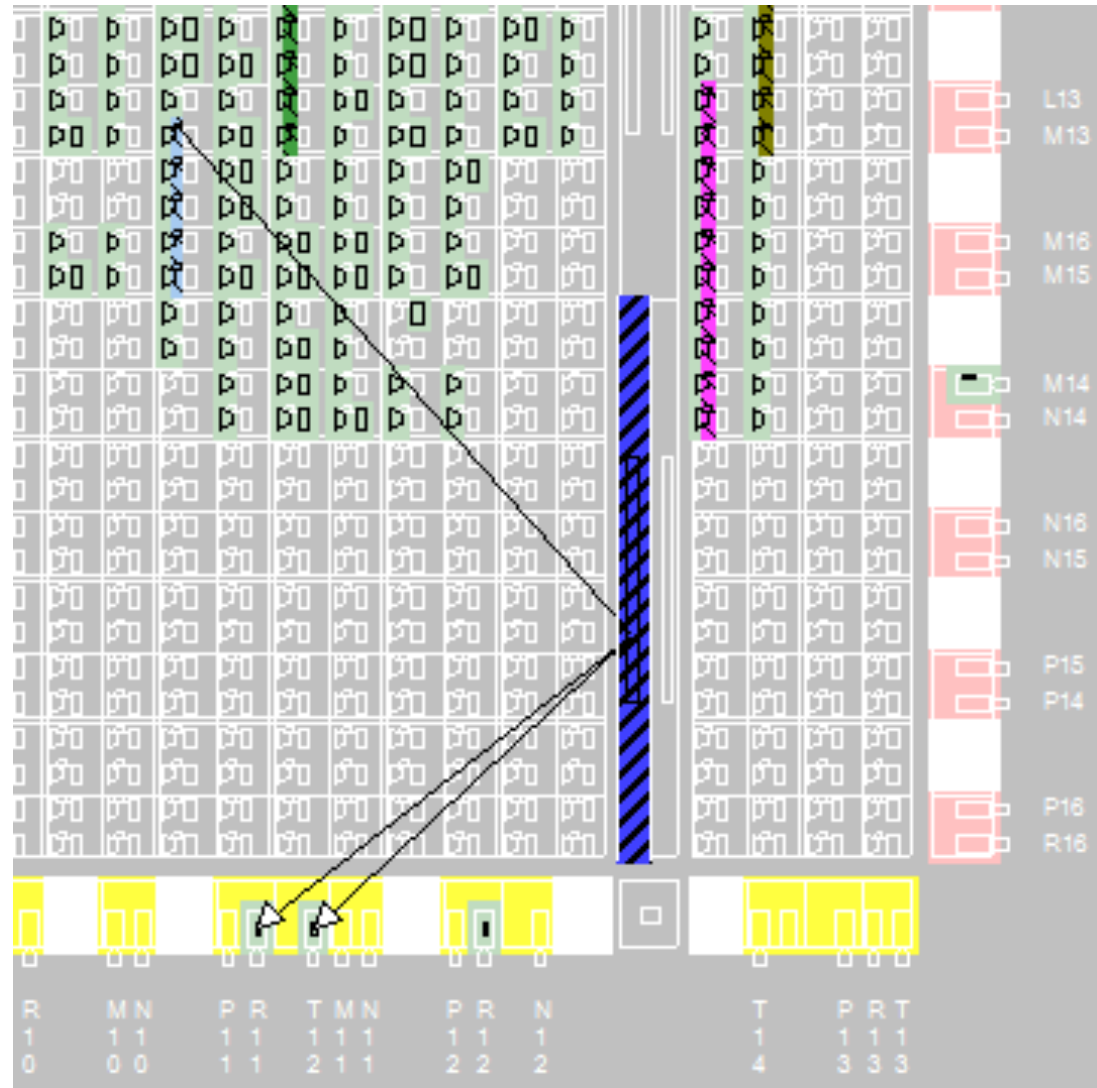
- Dozens of millions of equivalent logic gates
- Enhanced clock features
- Flexible embedded memory blocks
- Intellectual property (IP) cores
- Embedded processors (hard and soft)
- Digital signal processing (DSP) blocks, tools, design flows (specific FPGA vendors)
- Dedicated hardware multipliers
- high-speed communication capabilities
- Advanced I/O standards and protocol support

FPGA Example – Spartan 3



Courtesy of Xilinx Inc.

Example of FPGA Implementation



Pros and Cons of FPGA

- Pros
 - Fast turnaround.
 - Low NRE (non-recurring engineering) cost.
 - Low risk.
 - Effective design verification.
 - Low testing cost.
- Cons (compare to regular IC chip implementation, and are improving)
 - *Bigger chip size*
 - *Higher cost*
 - *Higher power consumption*
 - *Slower speed*
- Technology is still advancing