
# Digital Logic Design

**Hajar Falahati**

**Department of Computer Engineering**
**IRAN University of Science and Technology**

**hfalahati@iust.ac.ir**

# Outline

- Integrated Circuits
- Programmable Logic Device
- Read Only Memory (ROM)
- Programmable Logic Array (PLA)
- Programmable Array Logic (PAL)

# Integrated Circuits

# Realization

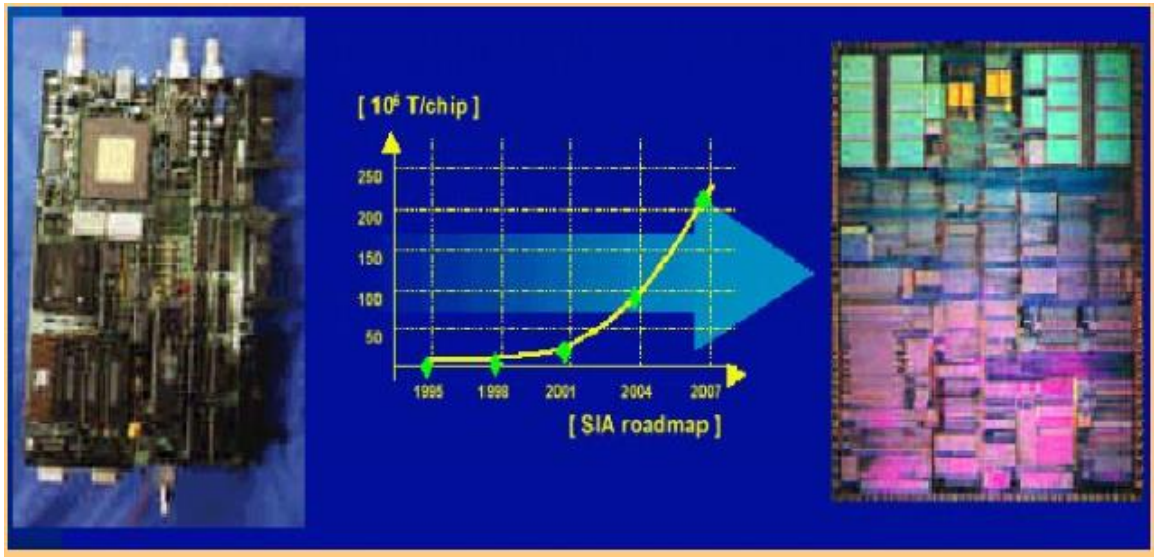
- AND-OR
- OR-AND
- NAND
- NOR

# Realization Types

- Application Specific Integrated Circuit (ASIC)
- Programmable Logic Device (PLD)

# ASIC

- Application Specific Integrated Circuit (ASIC)
- AND-OR Plane
  - Customized implementation
  - Efficient
  - High cost



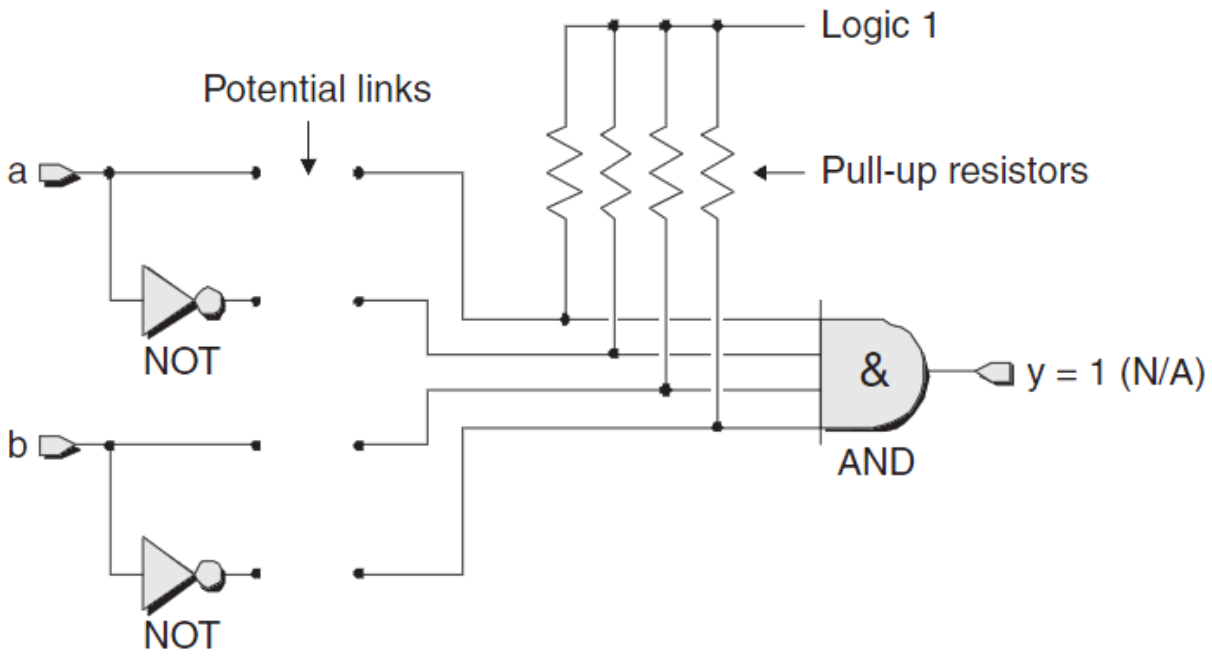
# ASIC (cont'd)

- Why Programable Logic Device (PLD)?
  - Custom computing
  - Reconfigurable computing
  - Reuse the device for a different design

# Programmable Logic Design

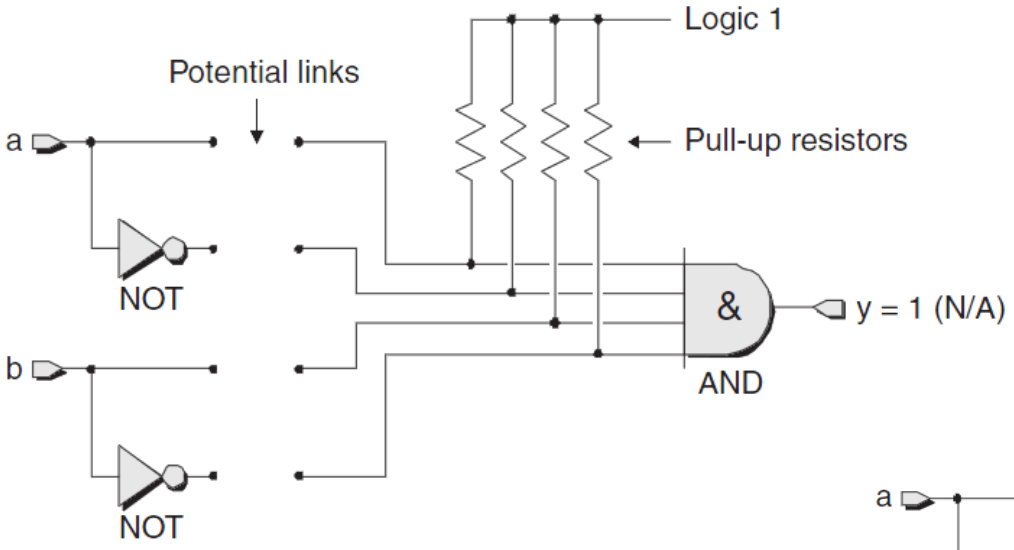
# Programmable Realization

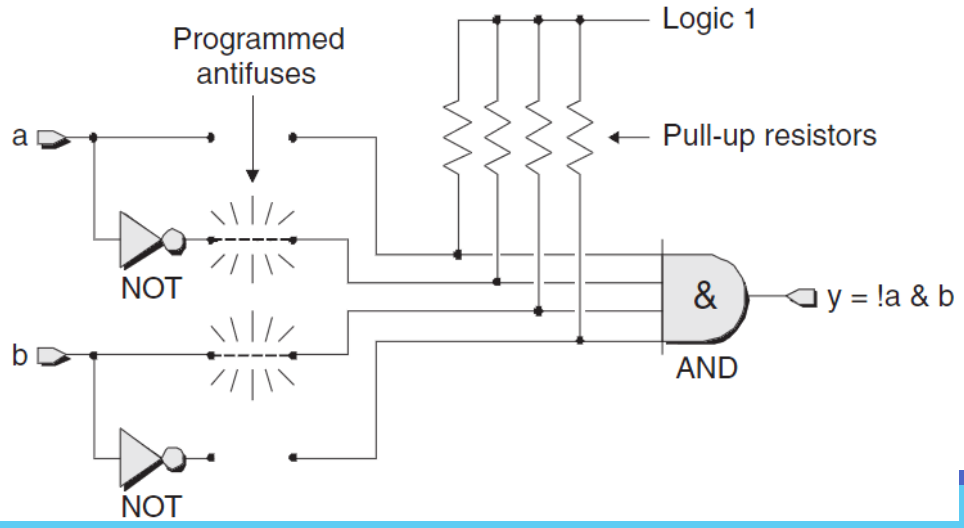
- Programmable gates



# Programmable Realization (cont'd)

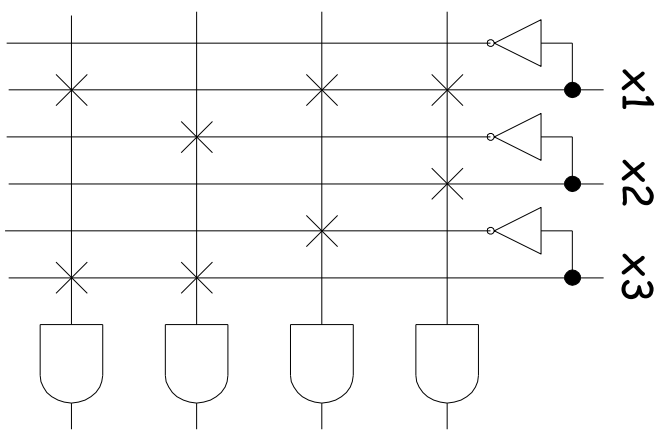
- Programmable gates

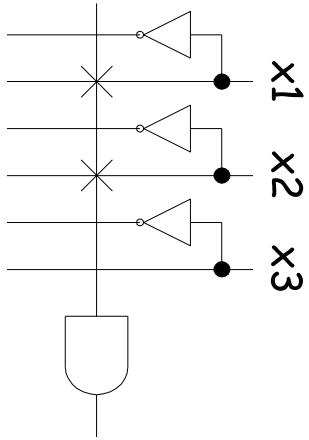


# Programmable Realization (cont'd)

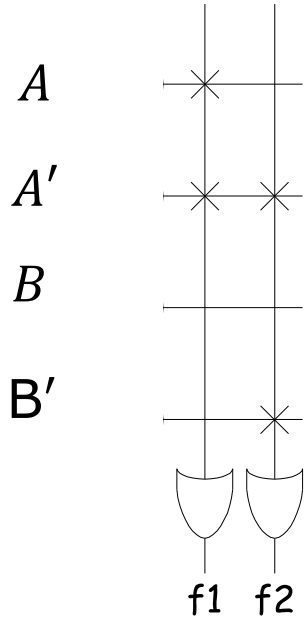
- Programmable AND



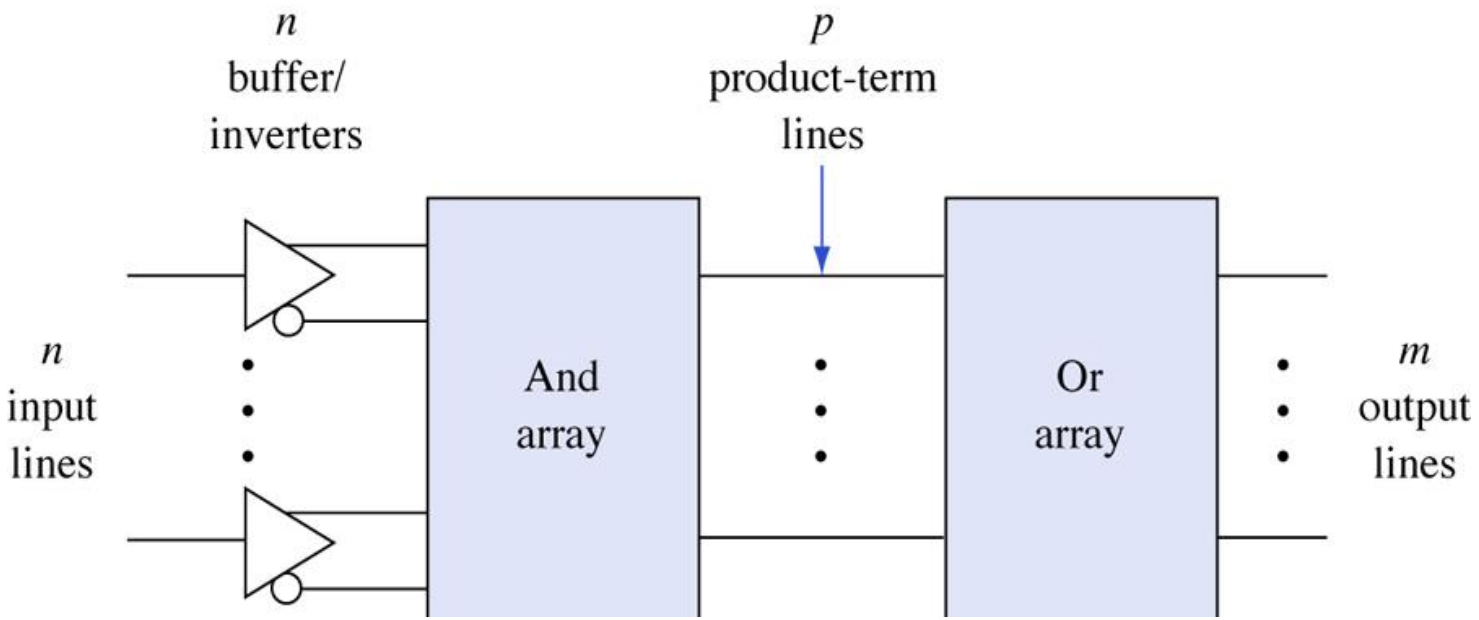
# Programmable Realization (cont'd)

- Programmable OR



# Programmable Logic Design

- Includes programmable gates

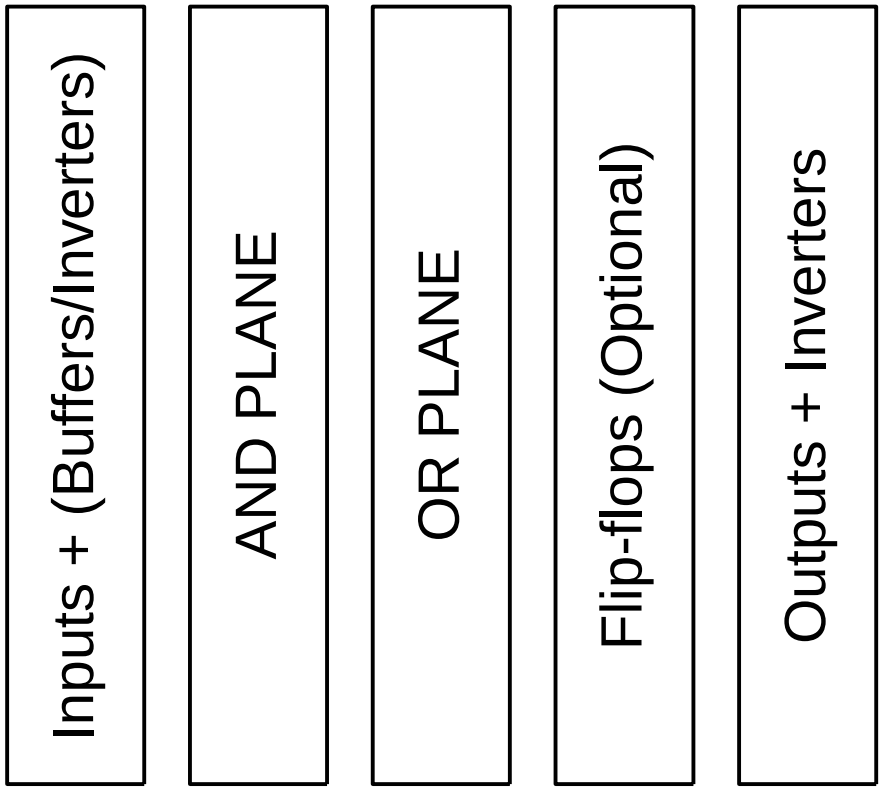


# PLD: Types

- SPLD Structure
- CPLD Structure
- FPGA

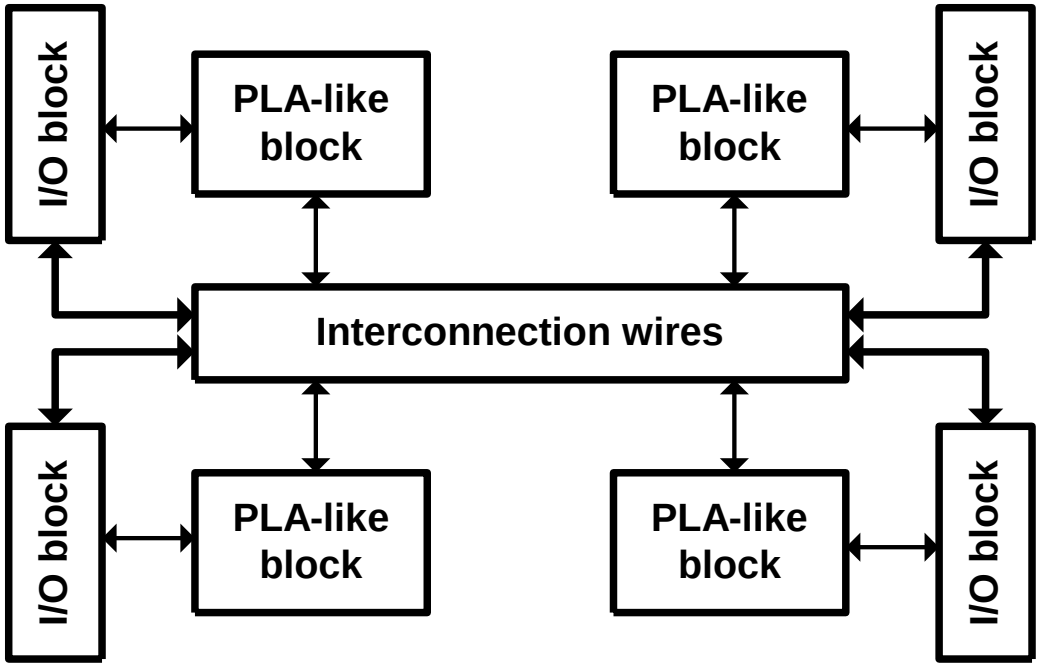
# SPLD Structure

- Simple Programmable Logic Design (SPLD)



# CPLD Structure

- Complex Programmable Logic Design (SPLD)



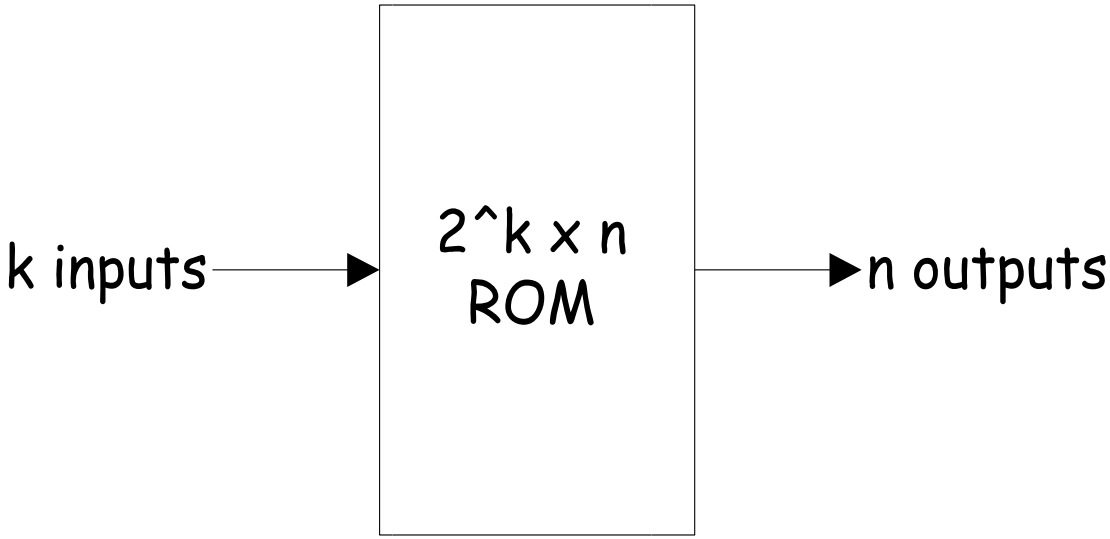
# SPLD

# SPLD Types

- Read Only Memory (ROM)
  - Fixed And-Plane, Programmable Or-Plane
- Programmable Array Logic (PAL)
  - Programmable And-Plane, Semi-Programmable Or-Plane
- Programmable Logic Array (PLA)
  - Programmable And-Plane and Or-Plane

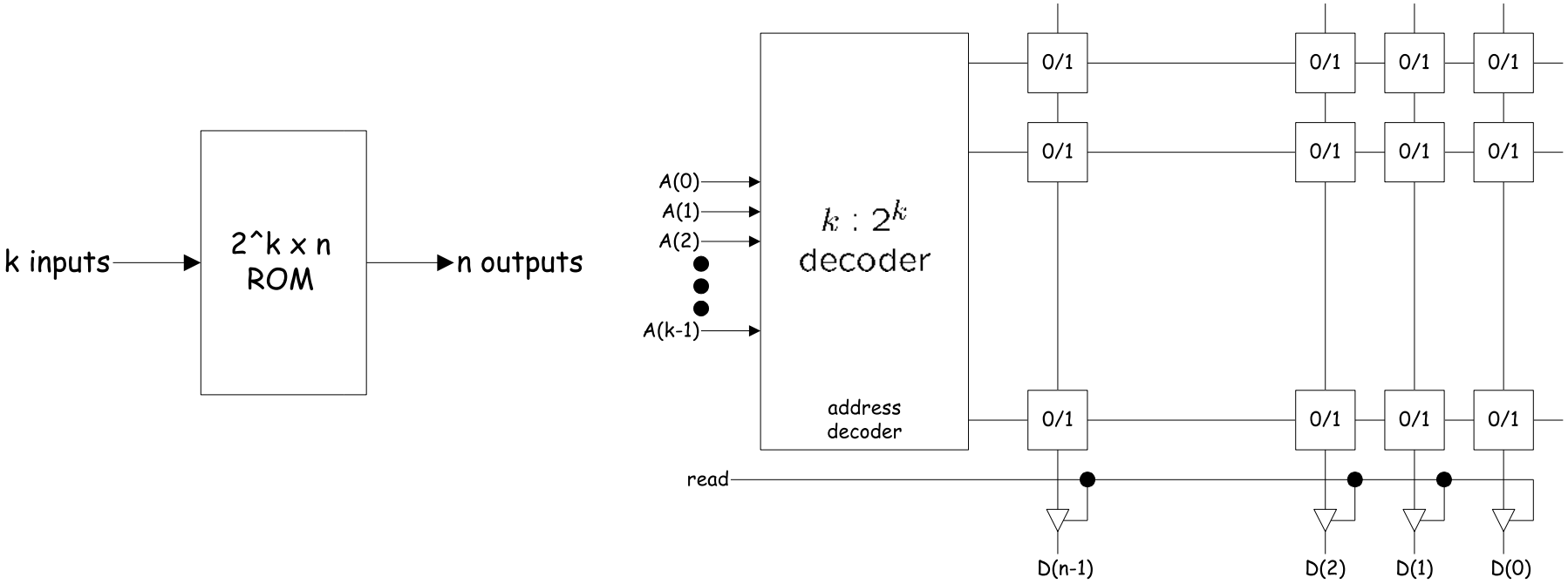
# ROM

- ROM
  - Store truth table in a memory
  - Store minterms/maxterms

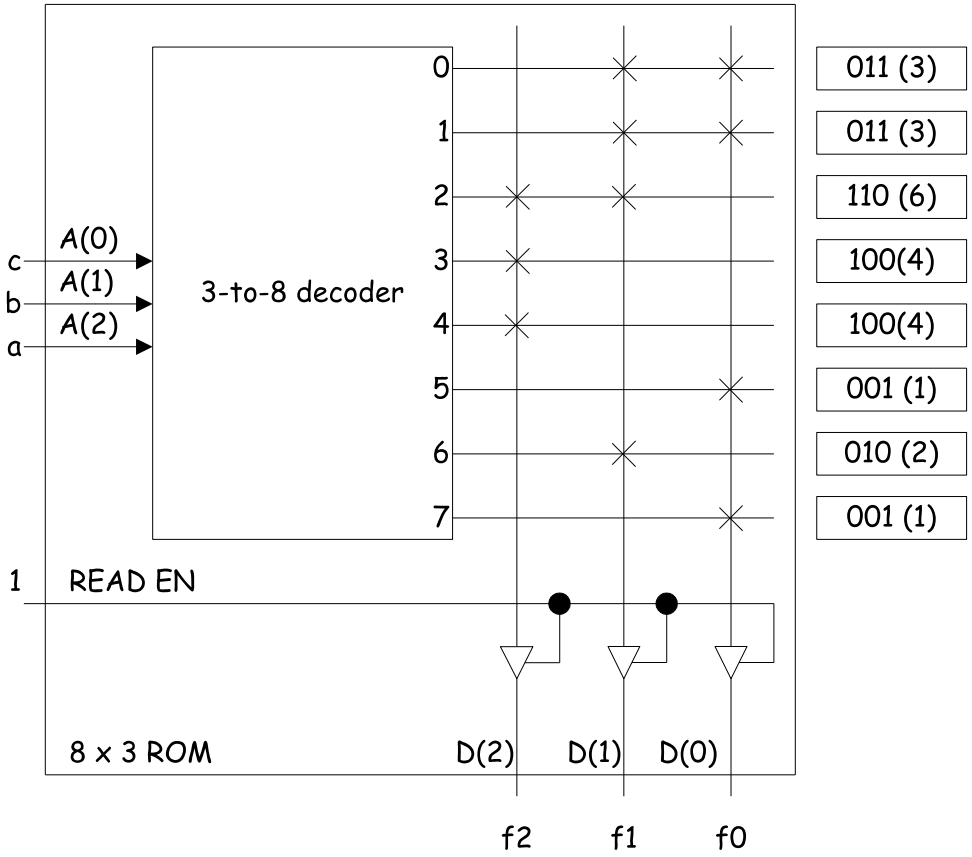


# Programmable ROM (PROM)

- PROM
  - Select which minterms/maxterms as inputs to OR/AND gates

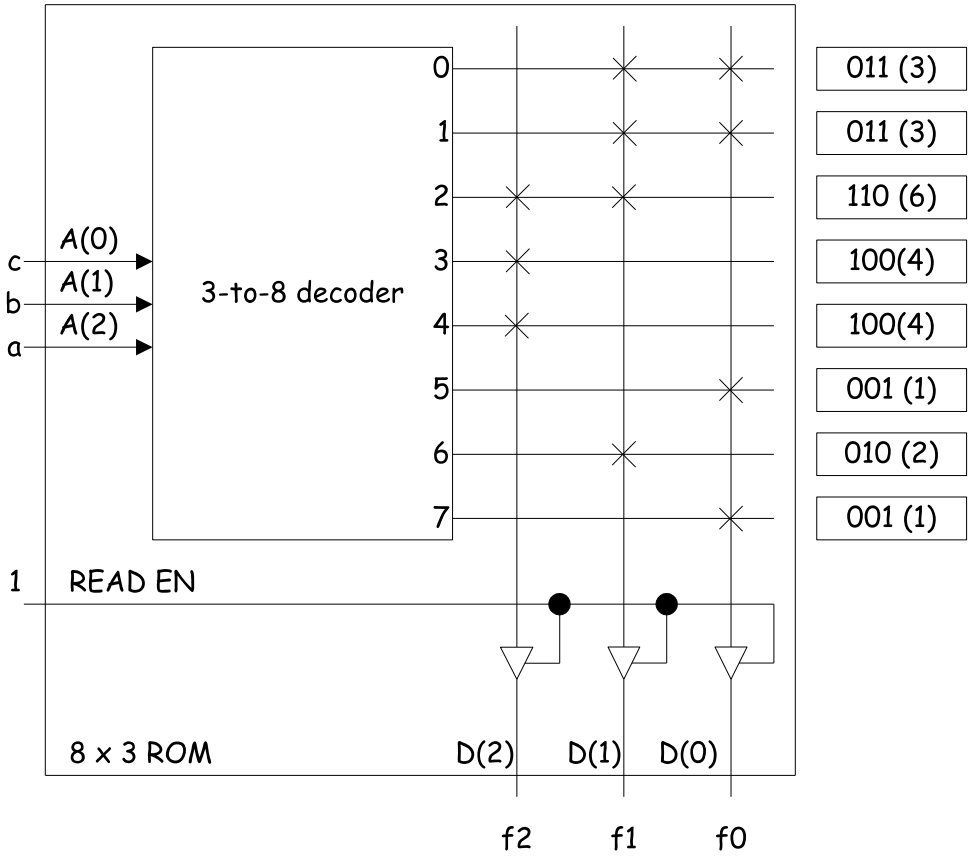


# PROM (cont'd)


A(0)
A(1)
A(2)
c
b
a
3-to-8 decoder
0
1
2
3
4
5
6
7
011 (3)
011 (3)
110 (6)
100(4)
100(4)
001 (1)
010 (2)
001 (1)
1
READ EN
8 x 3 ROM
D(2)
D(1)
D(0)
f2
f1
f0

# PROM (cont'd)

| $a$ | $b$ | $c$ | $f_2$ | $f_1$ | $f_0$ |
|---|---|---|---|---|---|
| 0 | 0 | 0 | 0 | 1 | 1 |
| 0 | 0 | 1 | 0 | 1 | 1 |
| 0 | 1 | 0 | 1 | 1 | 0 |
| 0 | 1 | 1 | 1 | 0 | 0 |
| 1 | 0 | 0 | 1 | 0 | 0 |
| 1 | 0 | 1 | 0 | 0 | 1 |
| 1 | 1 | 0 | 0 | 1 | 0 |
| 1 | 1 | 1 | 0 | 0 | 1 |



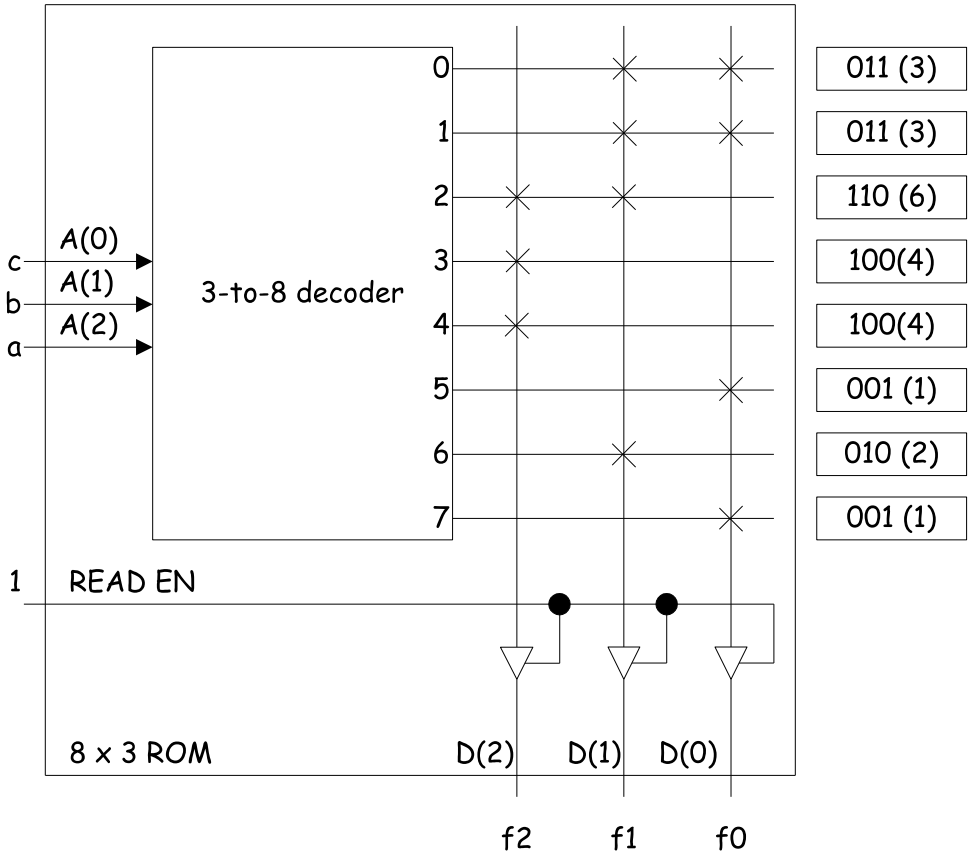
# Programable Logic: PROM (cont'd)

| $a$ | $b$ | $c$ | $f_2$ | $f_1$ | $f_0$ |
|---|---|---|---|---|---|
| 0 | 0 | 0 | 0 | 1 | 1 |
| 0 | 0 | 1 | 0 | 1 | 1 |
| 0 | 1 | 0 | 1 | 1 | 0 |
| 0 | 1 | 1 | 1 | 0 | 0 |
| 1 | 0 | 0 | 1 | 0 | 0 |
| 1 | 0 | 1 | 0 | 0 | 1 |
| 1 | 1 | 0 | 0 | 1 | 0 |
| 1 | 1 | 1 | 0 | 0 | 1 |

$f_0 = \sum (0,1,5,7)$

$f_1 = \sum (0,1,2,6)$

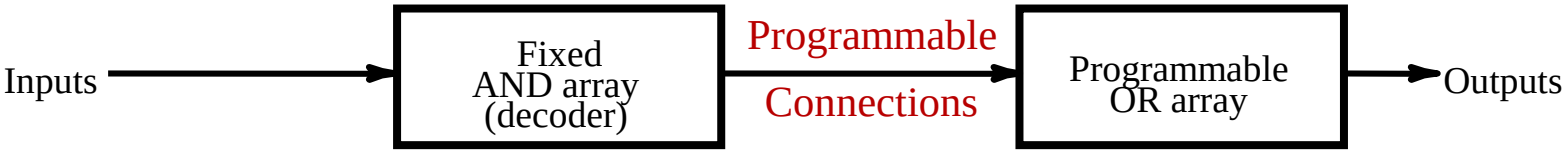
$f_2 = \sum (2,3,4)$



# PROM

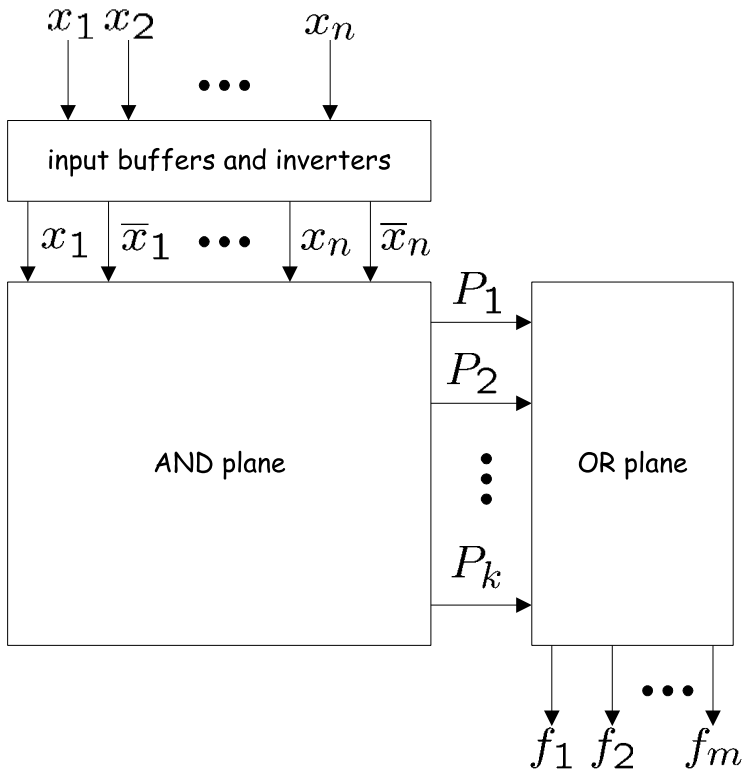
- PROM
  - Fixed And-Plane, Programmable Or-Plane

- Disadvantages
  - Canonical forms
  - => Not optimized



# Programmable Logic Array (PLA)

- PLA
  - Programmable And-Plane, Programmable Or-Plane



# PLA (cont'd)

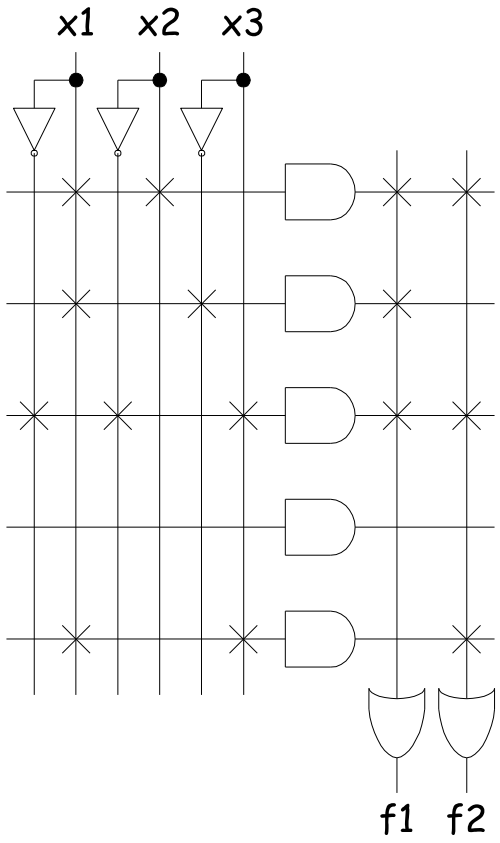
- 2 logic functions of 3 inputs

$$f_1 = x_1x_2 + x_1\overline{x}_3 + \overline{x}_1\overline{x}_2x_3$$

$$f_2 = x_1x_2 + \overline{x}_1\overline{x}_2x_3 + x_1x_3$$

# Programable Logic: PLA (cont'd)

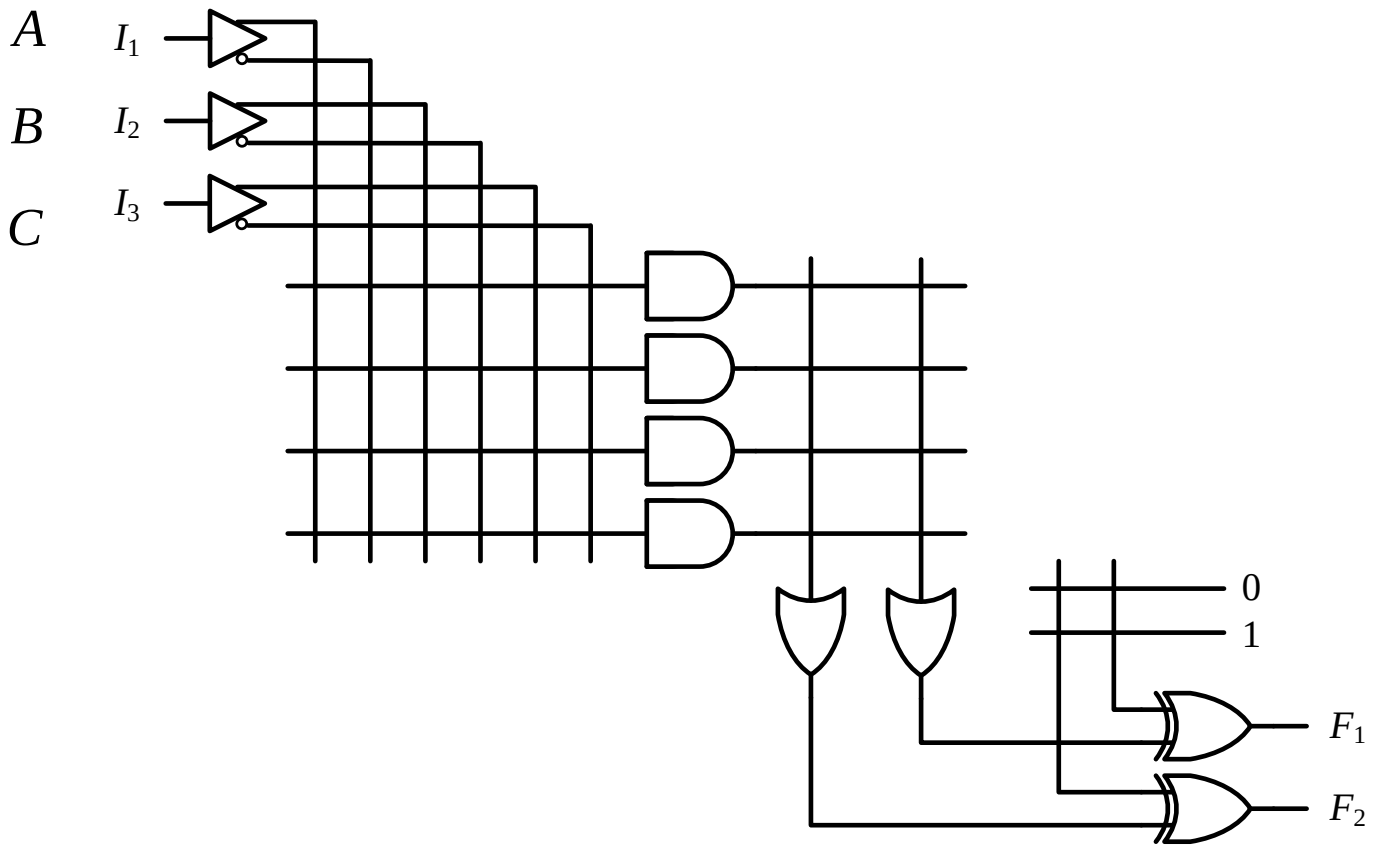
- 2 logic functions of 3 inputs



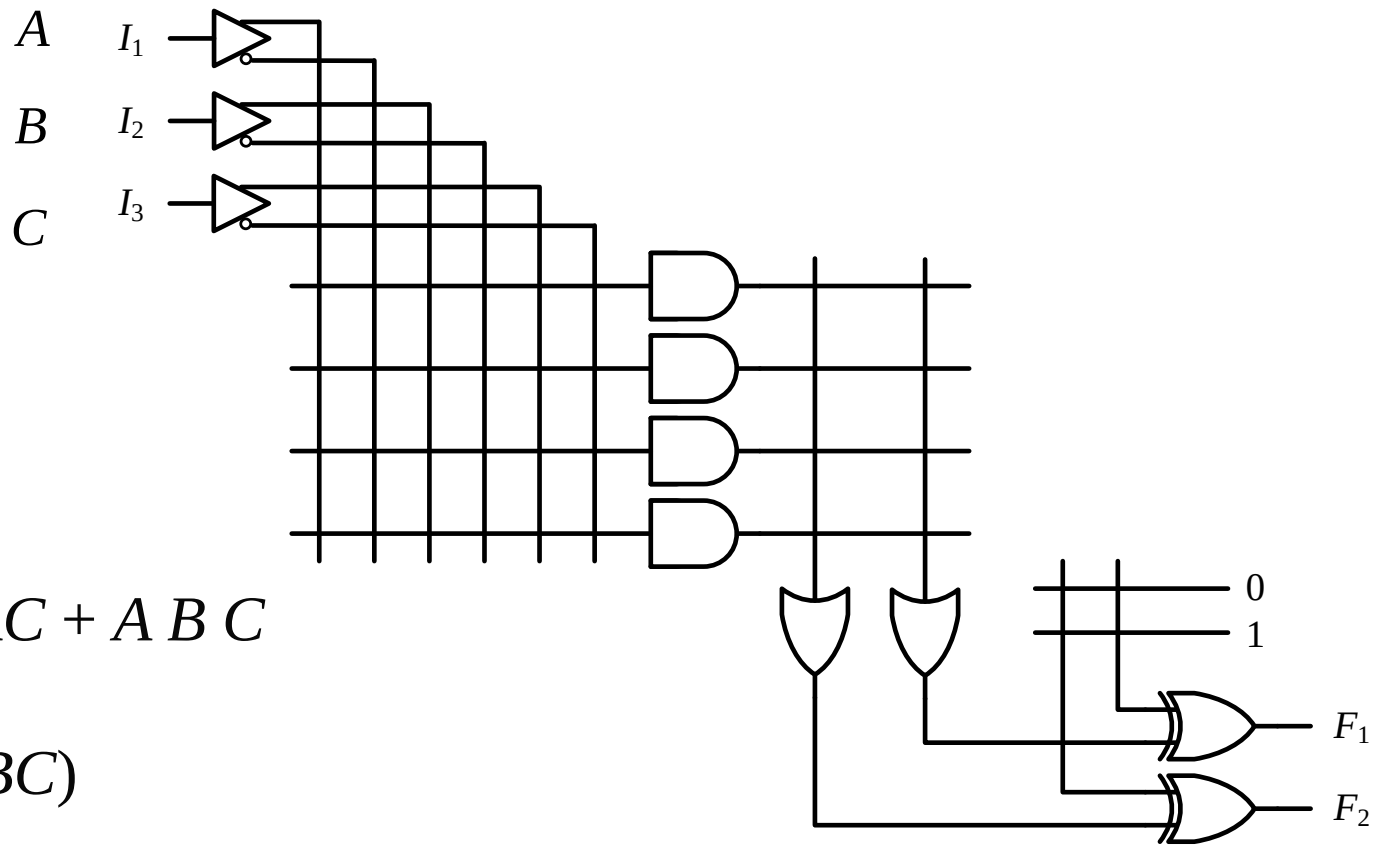

$$f_1 = x_1x_2 + x_1\overline{x}_3 + \overline{x}_1\overline{x}_2x_3$$

$$f_2 = x_1x_2 + \overline{x}_1\overline{x}_2x_3 + x_1x_3$$

# Programable Logic: PLA (cont'd)



# Programable Logic: PLA (cont'd)

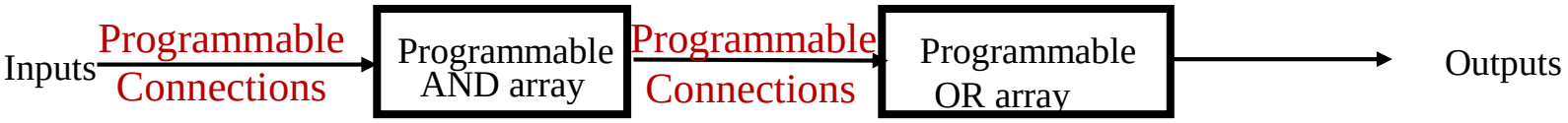



$$F_1 = A\,B + AC + A\,B\,C$$

$$F_2 = (AC + BC)$$

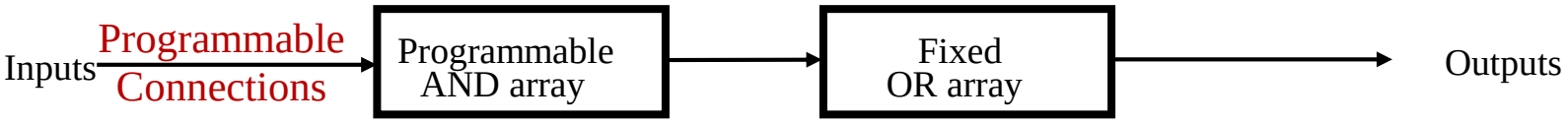
# PLA

- PLA
  - Programmable And-Plane and Or-Plane
- Disadvantages
  - => High cost

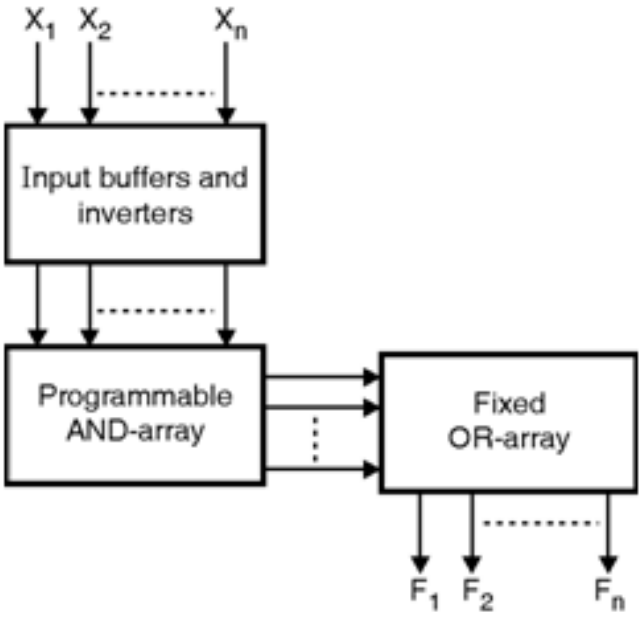


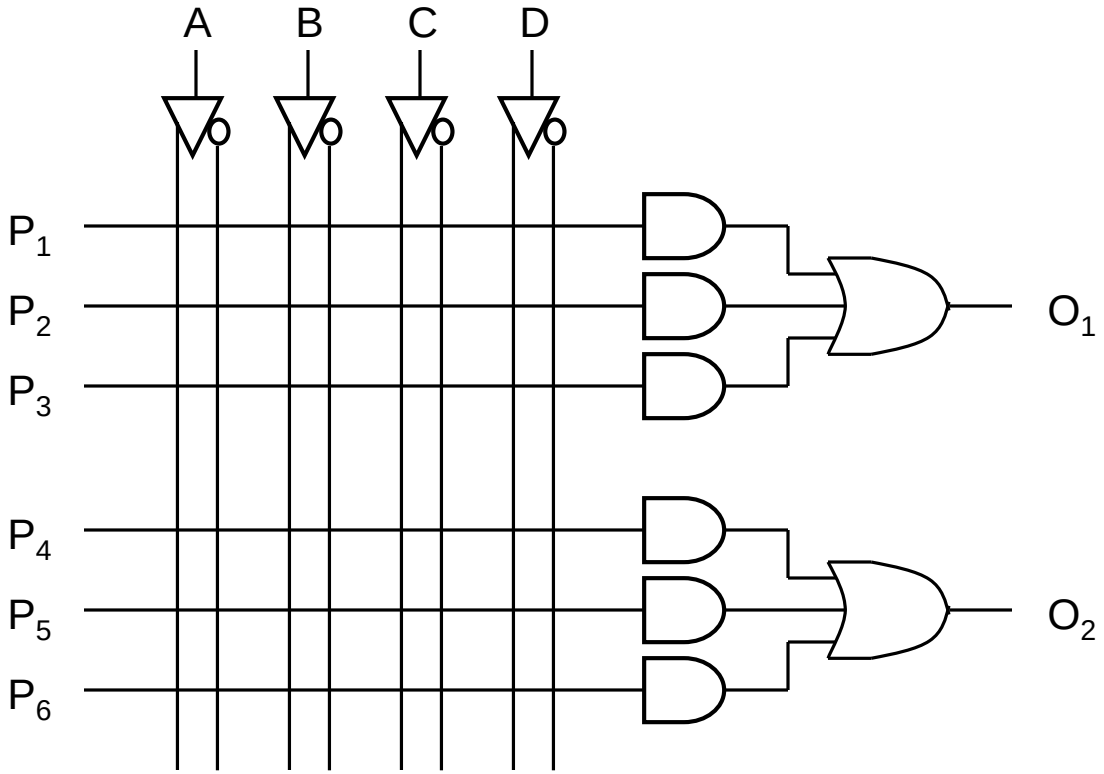
# Programmable Array Logic (PAL)

- PAL
  - Programmable And-Plane
  - Number of PT is limited
  - Optimized like as PLA
  - Lower cost than PLA



# PAL (cont'd)


X1 X2 Xn
Input buffers and inverters
Programmable AND-array
Fixed OR-array
F1 F2 Fn


A
B
C
D
P1
P2
P3
P4
P5
P6
O1
O2

# PAL (cont'd)

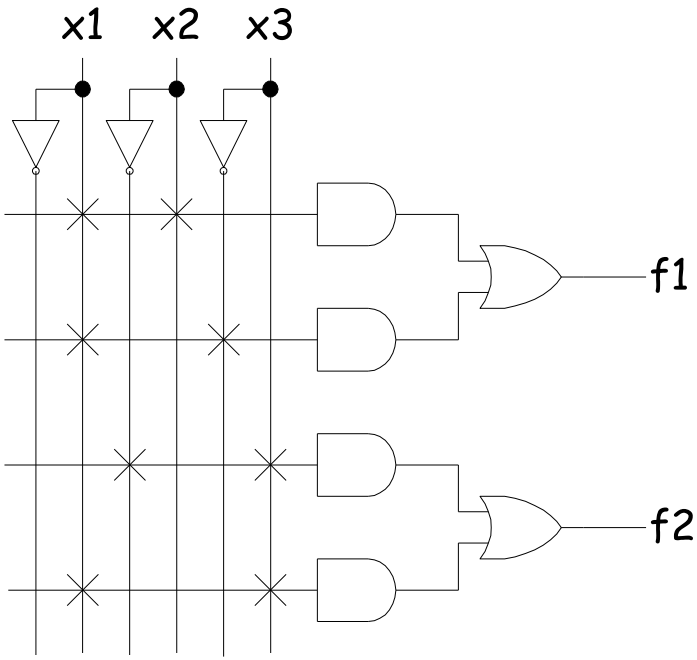
- 2 logic functions of 3 inputs

$$f_1 = x_1 x_2 + x_1 \overline{x}_3$$

$$f_2 = \overline{x}_2 x_3 + x_1 x_3$$

# PAL (cont'd)

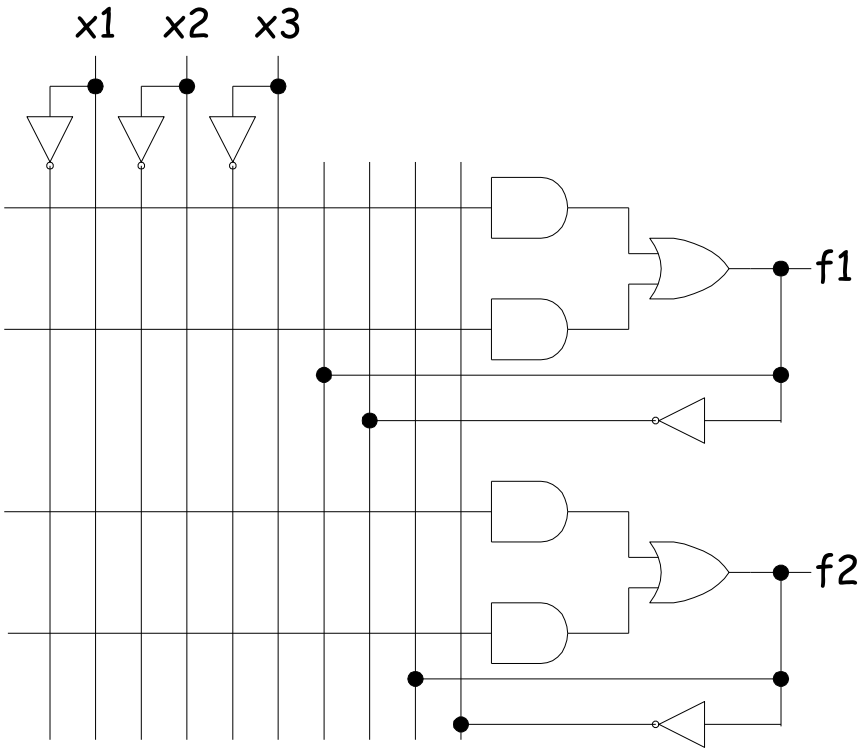
- 2 logic functions of 3 inputs




$$f_1 = x_1x_2 + x_1\overline{x}_3$$

$$f_2 = \overline{x}_2x_3 + x_1x_3$$

# PAL (cont'd)

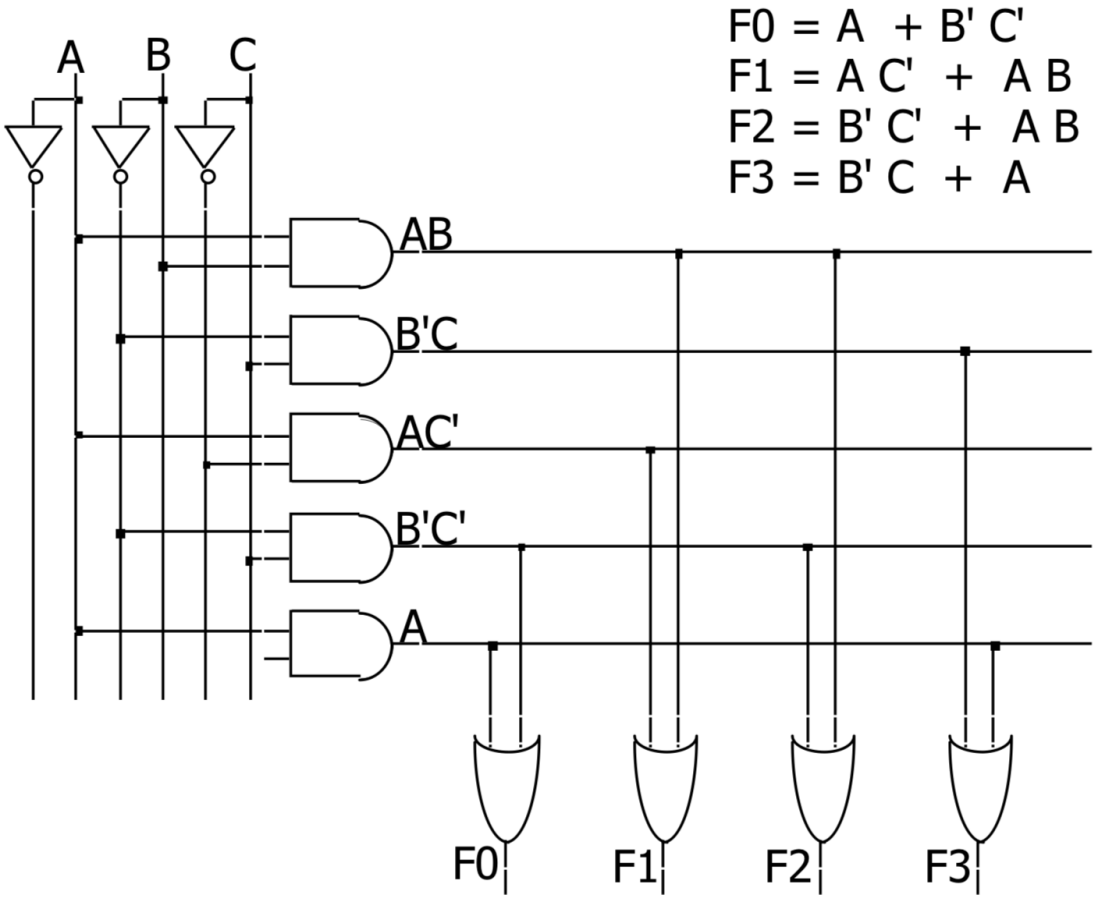


# Question 1

- PLA implementation

$$F0 = A + B'C'$$
$$F1 = AC' + AB$$
$$F2 = B'C' + AB$$
$$F3 = B'C + A$$

# Question1: Answer



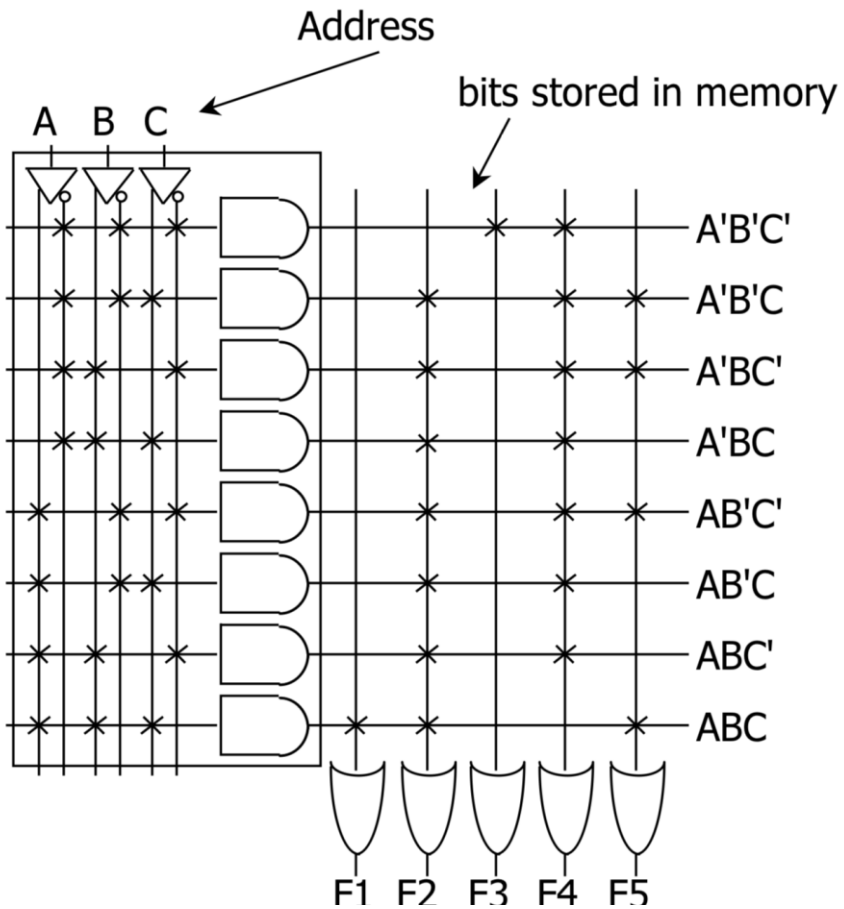
# Question 2

- ROM implementation

F1 = A B C
F2 = A + B + C
F3 = A' B' C'
F4 = A' + B' + C'
F5 = A xor B xor C

# Question 2: Answer

F1 = A B C
F2 = A + B + C
F3 = A' B' C'
F4 = A' + B' + C'
F5 = A xor B xor C



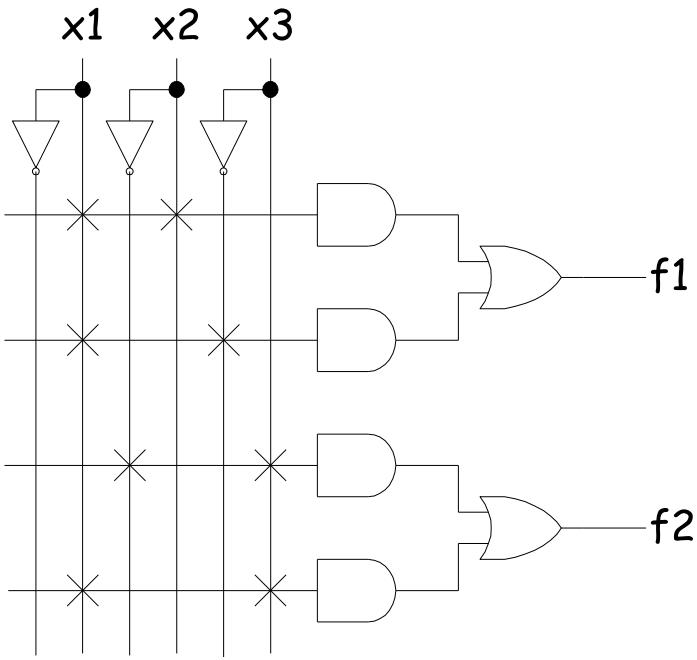
# Question 3

- PAL implementation

$$f_1 = x_1x_2 + x_1\overline{x}_3$$

$$f_2 = \overline{x}_2x_3 + x_1x_3$$

# Question 3: Answer




$$f_1 = x_1x_2 + x_1\overline{x}_3$$

$$f_2 = \overline{x}_2x_3 + x_1x_3$$

# Thank You