# PROGRAMMABLE LOGIC DEVICES

- ***Read Only Memory* (*ROM*)** - a fixed array of AND gates and a programmable array of OR gates
- ***Programmable Array Logic* (*PAL*) -** a programmable array of AND gates feeding a fixed array of OR gates.
- ***Programmable Logic Array* (*PLA*) -** a programmable array of AND gates feeding a programmable array of OR gates.
- ***Complex Programmable Logic Device* (*CPLD*) /*Field- Programmable Gate Array* (*FPGA*) -** complex enough to be called "architectures"

(a) Programmable read-only memory (PROM)

(b) Programmable array logic (PAL) device

(c) Programmable logic array (PLA) device
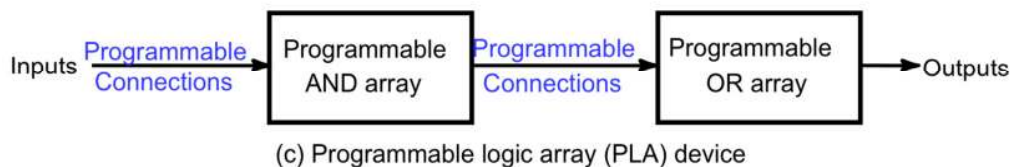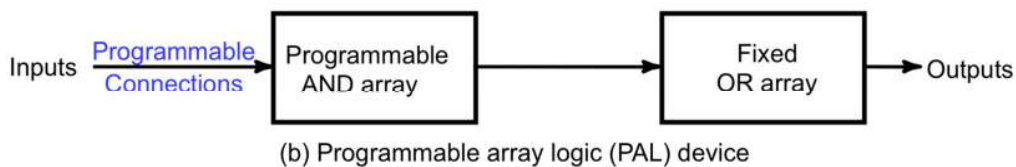
## READ ONLY MEMORY
- Read Only Memories (ROM) or Programmable Read Only Memories (PROM) have:
    - N input lines,
    - M output lines, and
    - 2^N decoded minterms.
- <u>Fixed</u> AND array with 2^N outputs implementing all N-literal minterms.
- <u>Programmable</u> OR Array with M outputs lines to form up to M sum of minterm expressions.
- A program for a ROM or PROM is simply a multiple-output truth table
    - If a 1 entry, a connection is made to the corresponding minterm for the corresponding output
    - If a 0, no connection is made
- Can be viewed as a *memory* with the inputs as *addresses* of *data* (output values), hence ROM or PROM names!

(a)

(b)
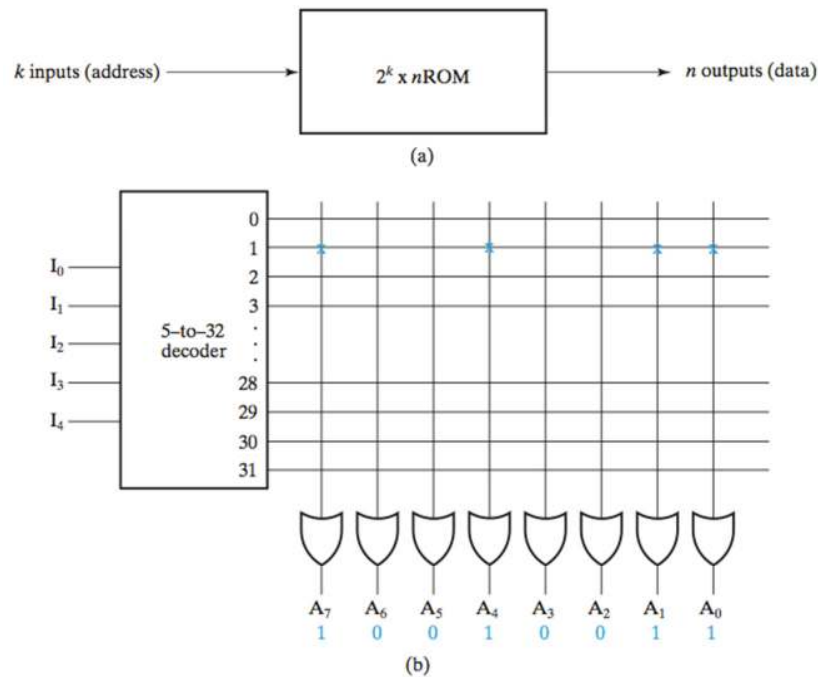
Figure: Block diagram and Internal Logic of a ROM

- Depending on the programming technology and approaches, read-only memories have different names
    1. ROM – mask programmed
    2. PROM – fuse or antifuse programmed
    3. EPROM – erasable floating gate programmed
    4. EEPROM or E^2PROM – electrically erasable floating gate programmed
    5. FLASH memory: electrically erasable floating gate with multiple erasure and programming modes.

- Example: A 8 X 4 ROM (N = 3 input lines, M= 4 output lines)
    - The fixed "AND" array is a "decoder" with 3 inputs and 8 outputs implementing minterms.
    - The programmable "OR" array uses a single line to represent all inputs to an OR gate. An "X" in the array corresponds to attaching the minterm to the OR
    - Read Example: For input (A_2,A_1,A_0) = 001, output is $(F_3,F_2,F_1,F_0$) = 0011.
    - What are functions F_3, F_2 , F_1 and F_0 in terms of (A_2, A_1, A_0)?
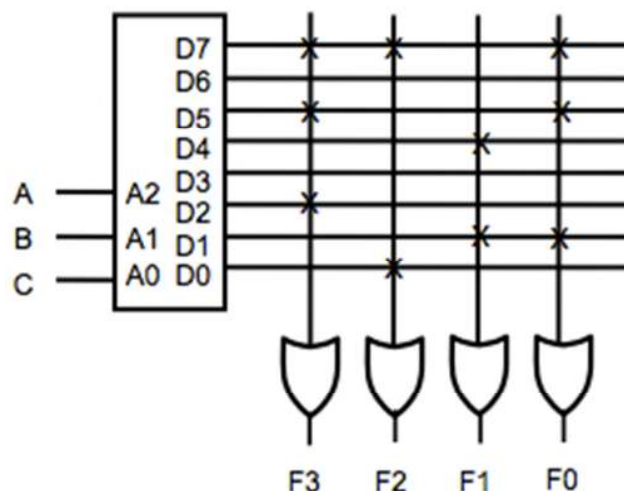
PROGRAMMABLE LOGIC ARRAY (PLA)

- Compared to a ROM and a PAL, a PLA is the most flexible having a <u>programmable</u> set of ANDs combined with a <u>programmable</u> set of ORs.
- Advantages
  - A PLA can have large N and M permitting implementation of equations that are impractical for a ROM (because of the number of inputs, N, required
  - A PLA has all of its product terms connectable to all outputs, overcoming the problem of the limited inputs to the PAL Ors
  - Some PLAs have outputs that can be complemented, adding POS functions
- Disadvantages
  - Often, the product term count limits the application of a PLA.
  - Two-level multiple-output optimization is required to reduce the number of product terms in an implementation, helping to fit it into a PLA.
  - Multi-level circuit capability available in PAL not available in PLA. PLA requires external connections to do multi-level circuits.

Programmable Logic Array Example
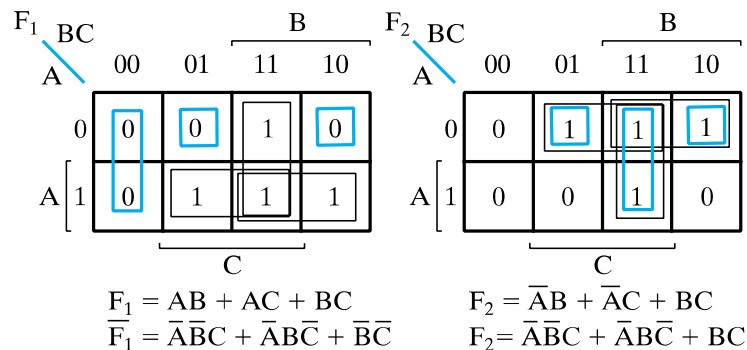
$$F1 = AB' + AC + A'BC'$$
$$F2 = (AC+BC)'$$
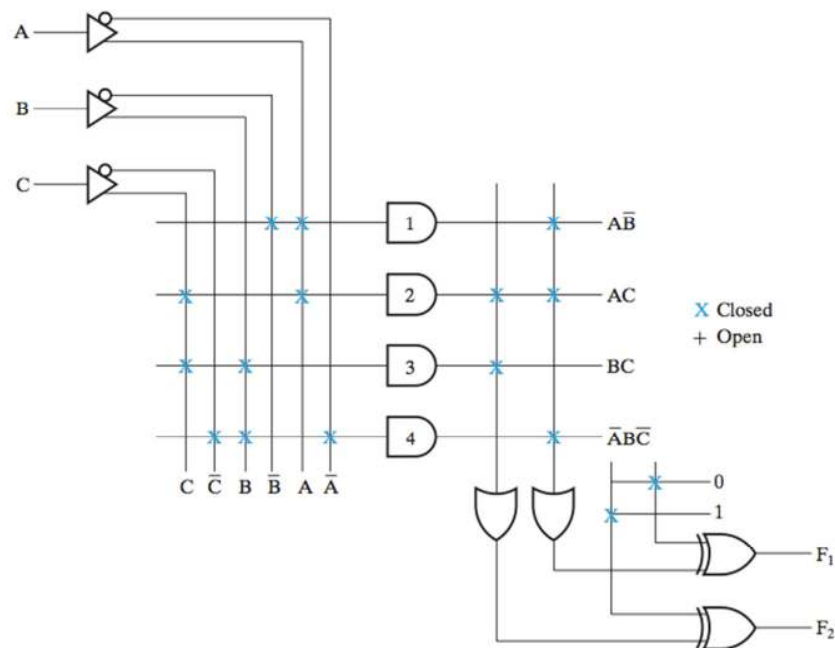
- What are the equations for F_1 and F_2?
- Could the PLA implement the functions without the XOR gates?
- 3-input, 3-output PLA with 4 product terms

Example 6-3 from Mano: Implementing a Combinational Circuit Using a PLA

F1(A,B,C)= Σm(3,5,6,7)

F2(A,B,C)= Σm(1,2,3,7)

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

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

The solution is:

$$F_1 = \overline{\overline{A}\,\overline{B}C + \overline{A}B\overline{C} + \overline{B}\,\overline{C}}$$
$$F_2 = \overline{A}\,\overline{B}C + \overline{A}B\overline{C} + BC$$

PROGRAMMABLE ARRAY LOGIC (PAL)

- The PAL is the opposite of the ROM, having a <u>programmable</u> set of ANDs combined with <u>fixed</u> ORs.
- Disadvantage
  - ROM guaranteed to implement any M functions of N inputs. PAL may have too few inputs to the OR gates.
- Advantages
  - For given internal complexity, a PAL can have larger N and M
  - Some PALs have outputs that can be complemented, adding POS functions
  - No multilevel circuit implementations in ROM (without external connections from output to input). PAL has outputs from OR terms as internal inputs to all AND terms, making implementation of multi-level circuits easier.

Programmable Array Logic Example
- 4-input, 3-output PAL with fixed, 3-input OR terms
- What are the equations for F1 through F4?

W(A,B,C,D) = Σm (2,12,13)

X(A,B,C,D) = Σm (7,8,9,10,11,12,13,14,15)

Y(A,B,C,D) = Σm (0,2,3,4,5,6,7,8,10,11,15)

Z(A,B,C,D) = Σm (1,2,8,12,13)

Simplifying the four function to a minimum number of terms results in the following Boolean functions

W= ABC'+A'B'CD'

X = A+BCD

Y = A'B+CD+B'D'

Z = ABC'+A'B'CD'+AC'D'+A'B'C'D = W+AC'D'+A'B'C'D

Product term

AND gates inputs

A Ā B B̄ C C̄ D D̄ W W̄

1
2
3

A

4
5
6

All Closed
(always = 0)

B

7
8
9

C

10
11
12

D

W
X
Y
Z

X Closed
+ Open

A Ā B B̄ C C̄ D D̄ W W̄