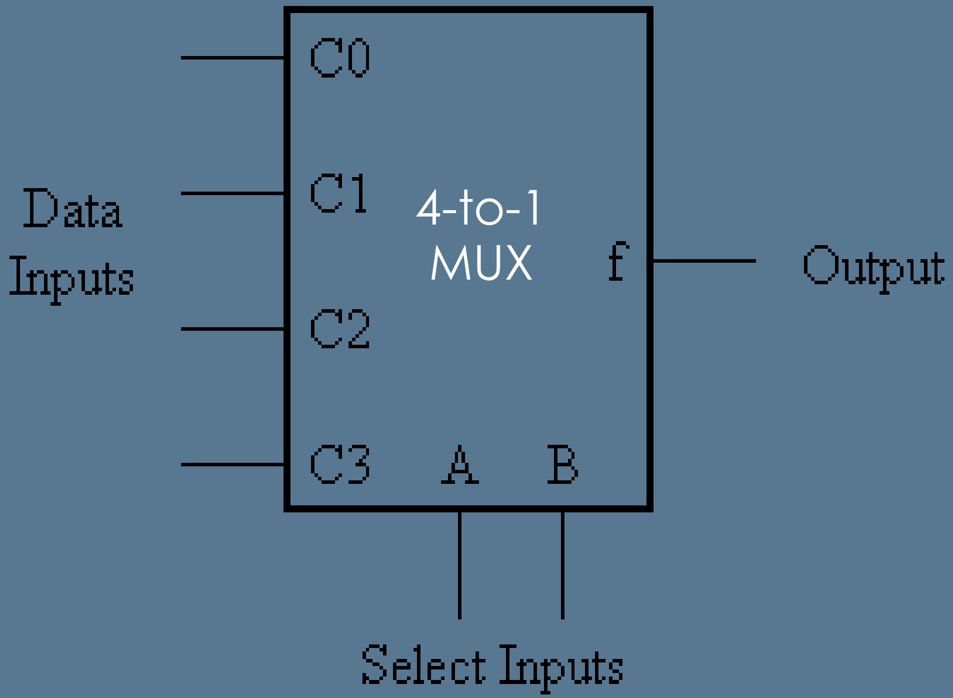
Chapter 7

COMBINATIONAL LOGIC BUILDING BLOCKS

Multiplexer (Data selectors)

- A device that selects binary information from one of many input lines and directs it to a single output line.
- Control signal pattern forms binary index of input connected to output
- Two forms:
  – Logical form (with n selection lines)
  – Functional form (with n-1 selection lines)

Example: MUX

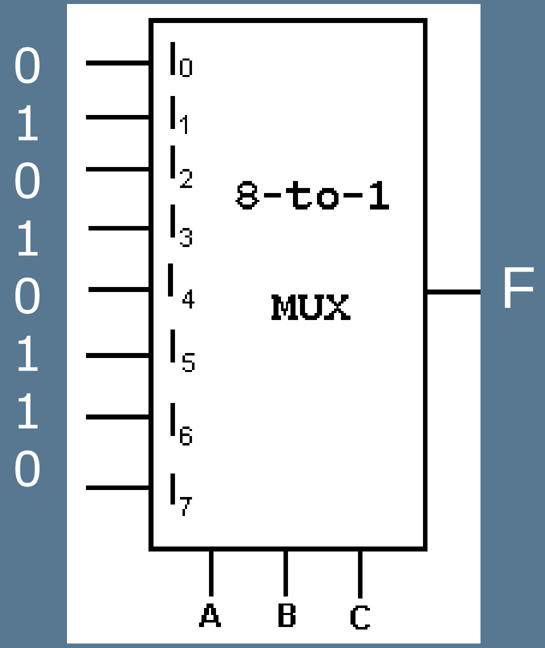


Example: MUX

- Implement the function $F = \sum(1,3,5,6)$ using a MUX in (a) Logical form and (b) Functional form

Example: MUX

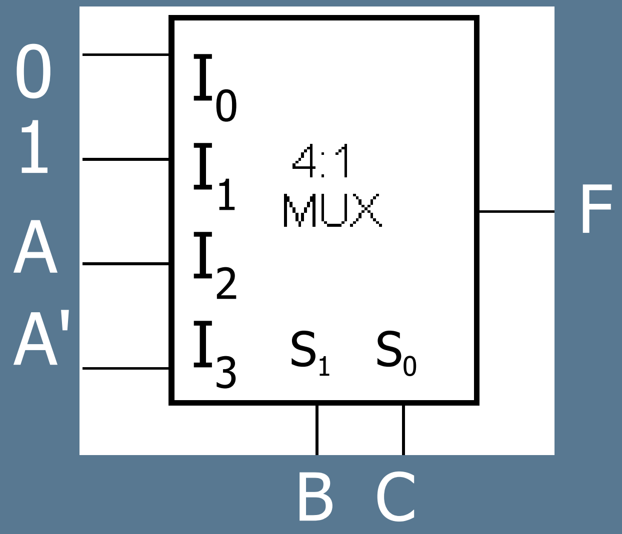
(a) $F = \sum(1,3,5,6)$ in Logical form (Use an 8-to-1 MUX)



Example: MUX

(b) $F = \sum(1,3,5,6)$ in Functional form (Use a 4-to-1 MUX)

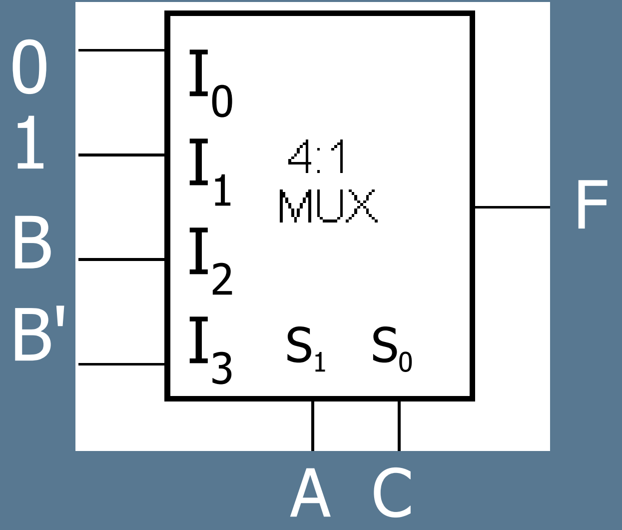
	I_0	I_1	I_2	I_3
A'	0	1	2	3
A	4	5	6	7
	0	1	A	A'



Example: MUX

(b) $F = \sum(1,3,5,6)$ in Functional form (Use a 4-to-1 MUX)

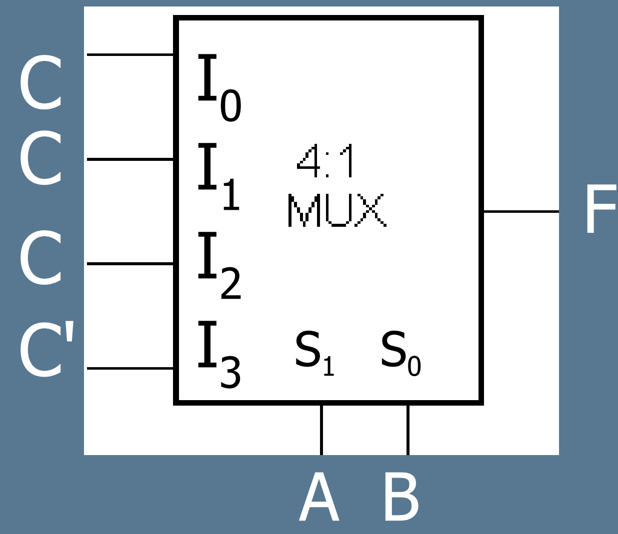
	I_0	I_1	I_2	I_3
B’	0	1	4	5
B	2	3	6	7
	0	1	B	B’



Example: MUX

(b) $F = \sum(1,3,5,6)$ in Functional form (Use a 4-to-1 MUX)

	I_0	I_1	I_2	I_3
C’	0	2	4	6
C	1	3	5	7
	C	C	C	C’

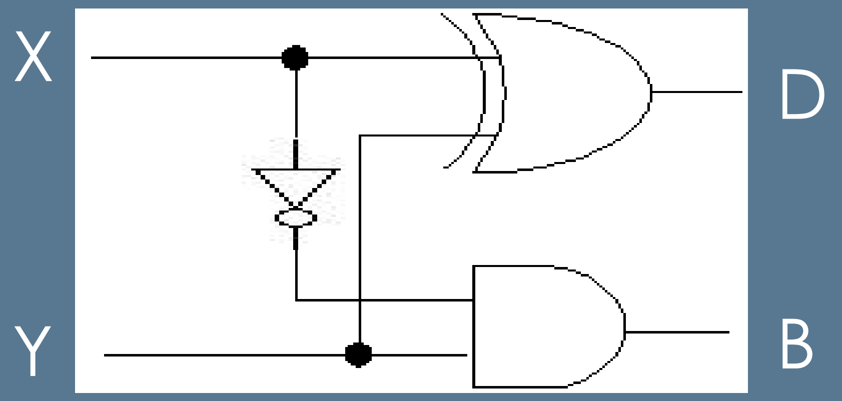


Subtracter

- Half-Subtracter
  - combinational circuit that subtracts two bits and produces their difference
- Full-Subtracter
  - combinational circuit that performs subtraction between bits, taking into account that a 1 may have been borrowed by a lower significant stage

Half-Subtracter

x	y	B	D
0	0	0	0
0	1	1	1
1	0	0	1
1	1	0	0



$B = x'y$

$D = x'y + xy'$

$\quad = x \oplus y$

D = Difference

B = Borrow

Full-Subtracter

x	y	z	B	D
0	0	0	0	0
0	0	1	1	1
0	1	0	1	1
0	1	1	1	0
1	0	0	0	1
1	0	1	0	0
1	1	0	0	0
1	1	1	1	1

where:

x = minuend

y = subtrahend

z = previous borrow

B = Borrow

D = Difference

$B = x'y + x'z + yz$

$D = x'y'z + x'yz' + xy'z' + xyz$

Magnitude Comparator

- Combinational circuit that compares two numbers, A and B, and determines their relative magnitudes
- Two ways to compare numbers:
  - Bit-by-bit comparison
  - Subtraction

Magnitude Comparator

- Bit-by-bit comparison
- Given two binary numbers

  $A = A_n A_{n-1} \ldots A_1 A_0$

  $B = B_n B_{n-1} \ldots B_1 B_0$

A_n is compared to B_n, A_{n-1} is compared to B_{n-1}, and so on. If the two bits are the same, $X_i = 1$, where i denotes the position of the bit. If they are different, $X_i = 0$. A = B only if all X_i's are equal to 1. $(X_n X_{n-1} \ldots X_1 X_0)$

Bit-by-bit comparison

- To determine if $A > B$ or $A < B$, compare first the MSBs: A_n and B_n. If $A_n = 1$ and $B_n = 0$, then $A > B$. If $A_n = 0$ and $B_n = 1$, then $A < B$. If $A_n = B_n$, proceed to the next significant pair of bits, comparing them, until two unequal bits are found.

Bit-by-bit comparison

- Example: compare two 4-bit numbers

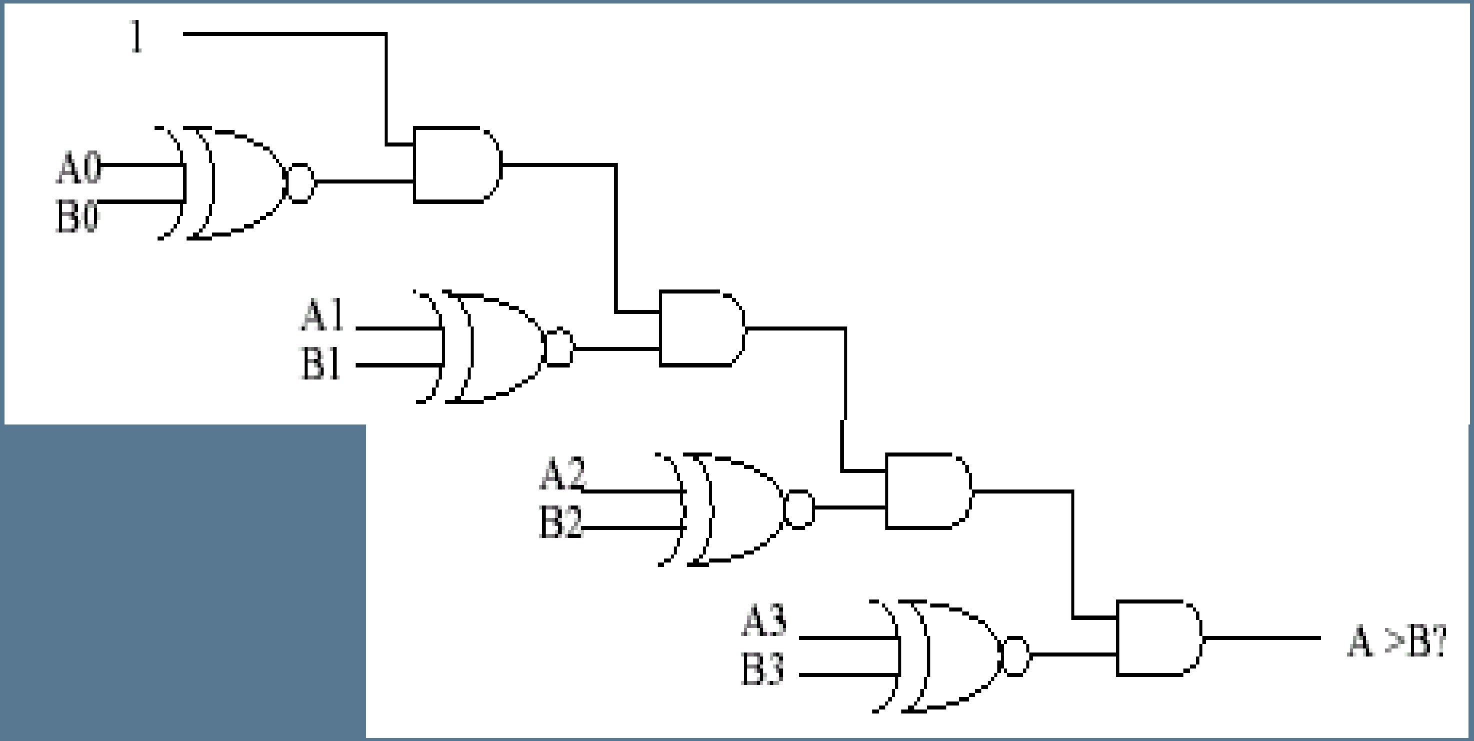
$(A = B) = X_3X_2X_1X_0$

$(A > B) = A_3B_3' + X_3A_2B_2' + X_3X_2A_1B_1' + X_3X_2X_1A_0B_0'$

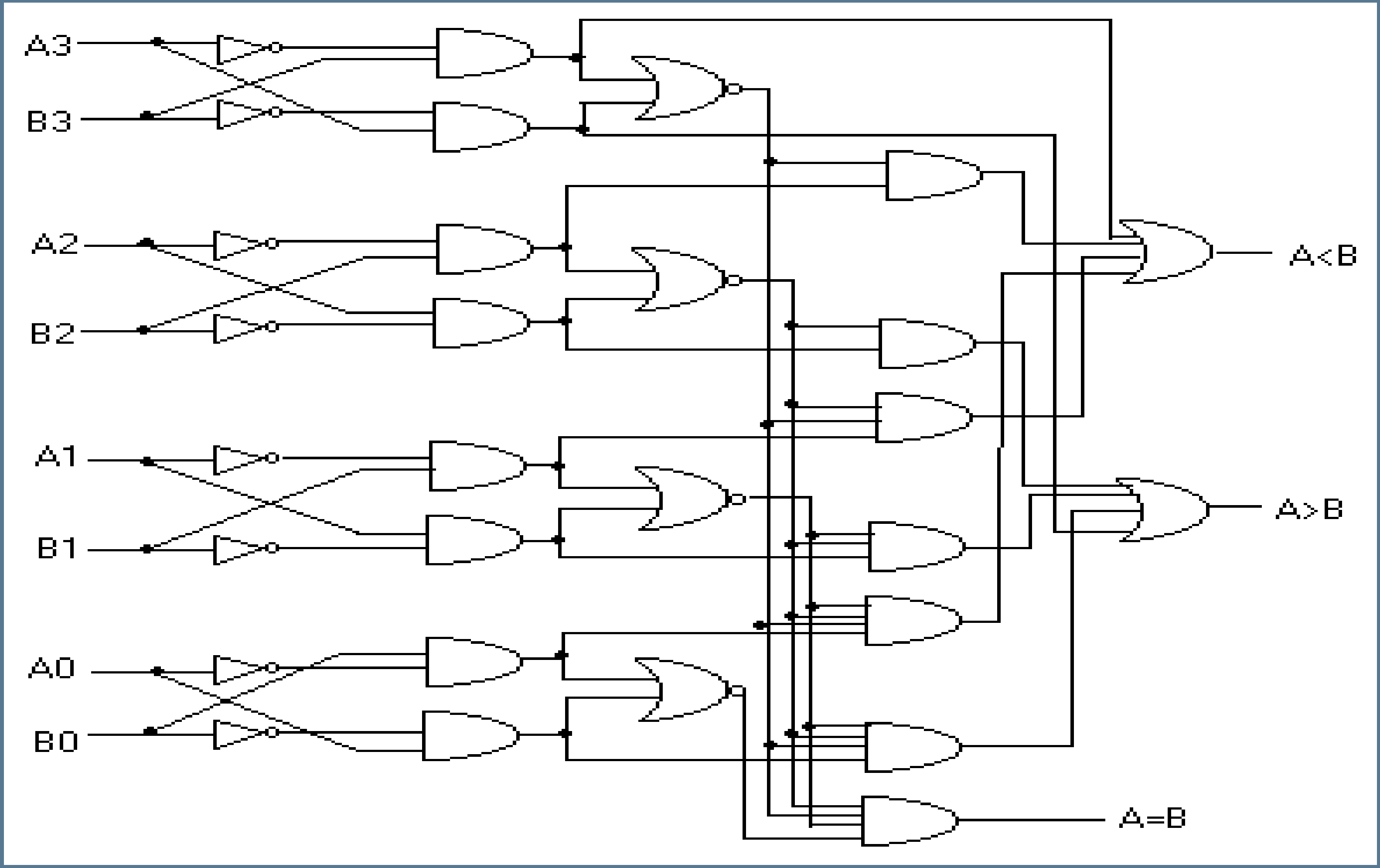
$(A < B) = A_3'B_3 + X_3A_2'B_2 + X_3X_2A_1'B_1 + X_3X_2X_1A_0'B_0$

Cascading Comparators

- an XNOR gate acts as one bit comparator
  - high if equal
  - low if not



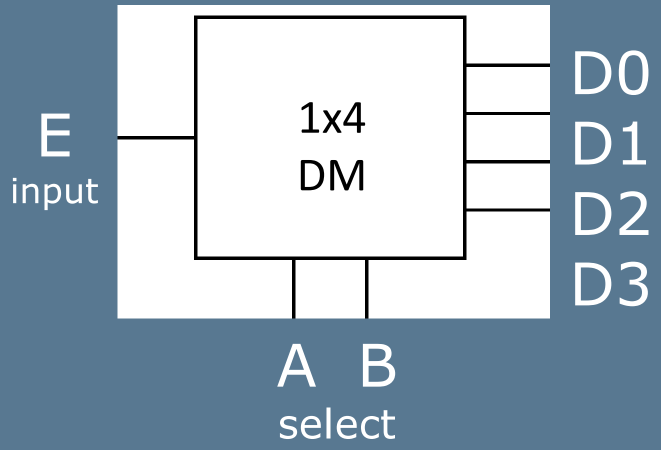
Example: 4-bit comparator



Demultiplexers

- A circuit that receives information on a single line and transmits this information on one of 2^n possible output lines.

- The selection of the specific output line is controlled by the values of n selection lines.



Encoder

- An encoder is a combinational circuit that performs the inverse operation of a decoder function.
- It has 2^n input lines (or fewer) and n output lines.

Example: Encoder

7	6	5	4	3	2	1	0	x	y	z
0	0	0	0	0	0	0	1	0	0	0
0	0	0	0	0	0	1	0	0	0	1
0	0	0	0	0	1	0	0	0	1	0
0	0	0	0	1	0	0	0	0	1	1
0	0	0	1	0	0	0	0	1	0	0
0	0	1	0	0	0	0	0	1	0	1
0	1	0	0	0	0	0	0	1	1	0
1	0	0	0	0	0	0	0	1	1	1

$x = 4 + 5 + 6 + 7$

$y = 2 + 3 + 6 + 7$

$z = 1 + 3 + 5 + 7$

Octal to Binary Encoder

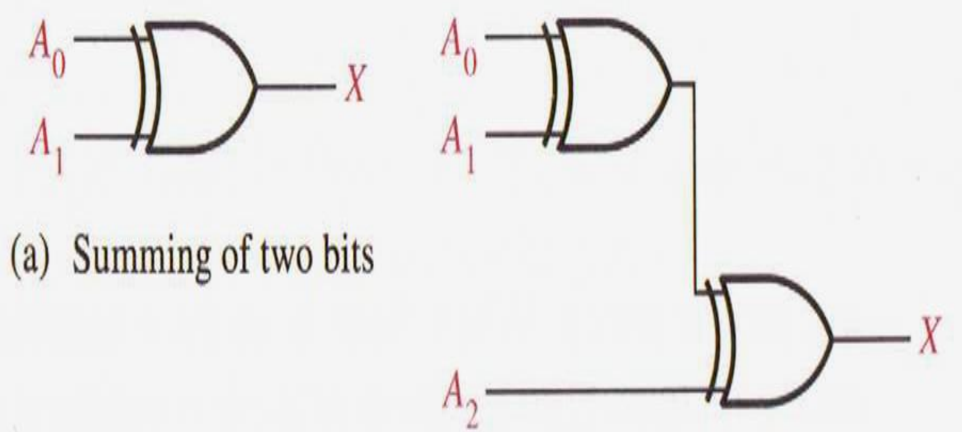
Code Converter

- A combinational circuit that makes two code systems compatible even though each uses a different binary code

- Some types:
  - BCD-to-excess-3 code converter
  - BCD-to-binary code converter

Parity Checker

- The sum (disregarding carries) of an even number of 1s is always 0 and the sum of odd number of 1s is always 1.



(a) Summing of two bits

(b) Summing of three bits

Programmable Logic Devices

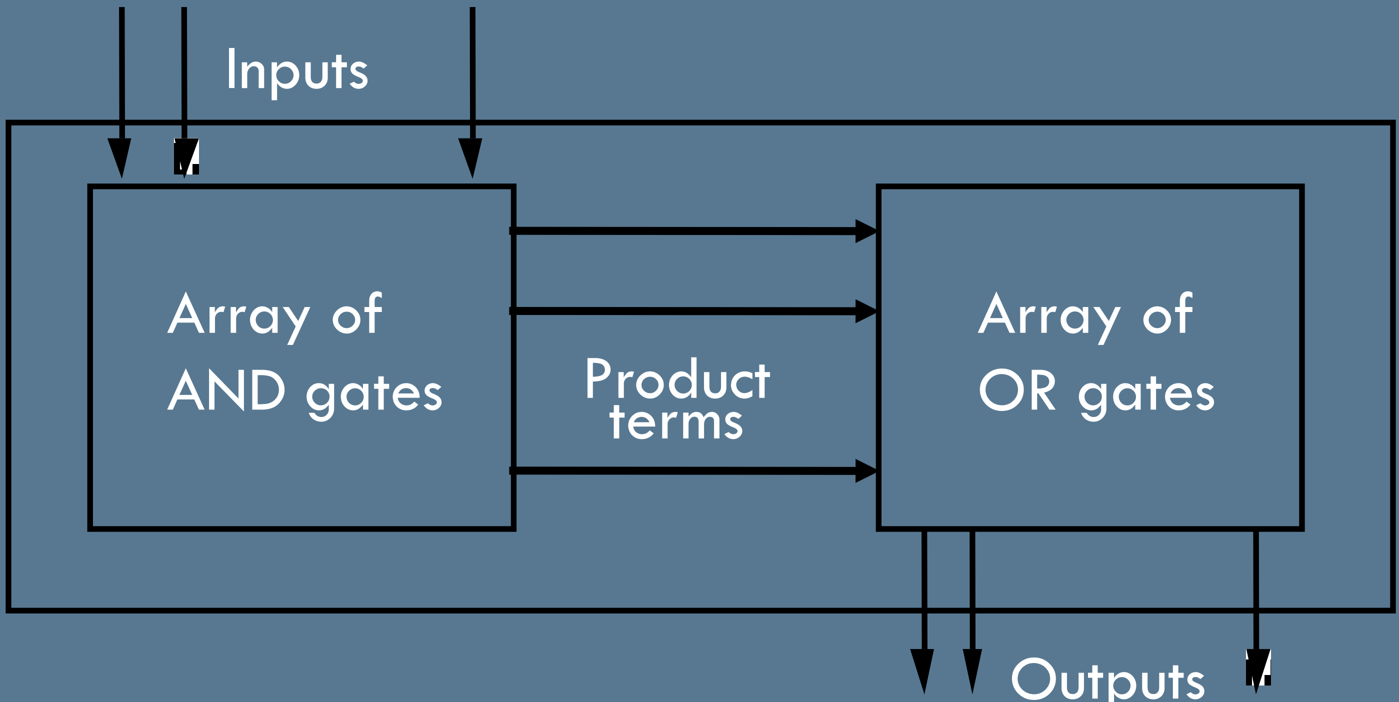
- Any Boolean function can be expressed in a Sum of Product form
- SOP form => AND-OR implementation

Programmable Logic Devices:

- Pre-fabricated building blocks of many AND/OR gates (or NOR, NAND)
- Programmed by making or breaking connections among the gates

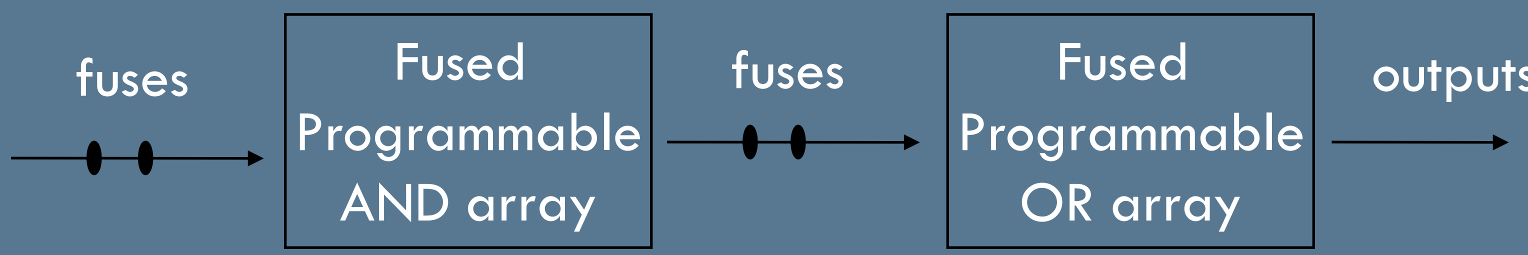
Programmable Logic Devices

Block Diagram of a Programmable Array of AND and OR for Sum of Products Form

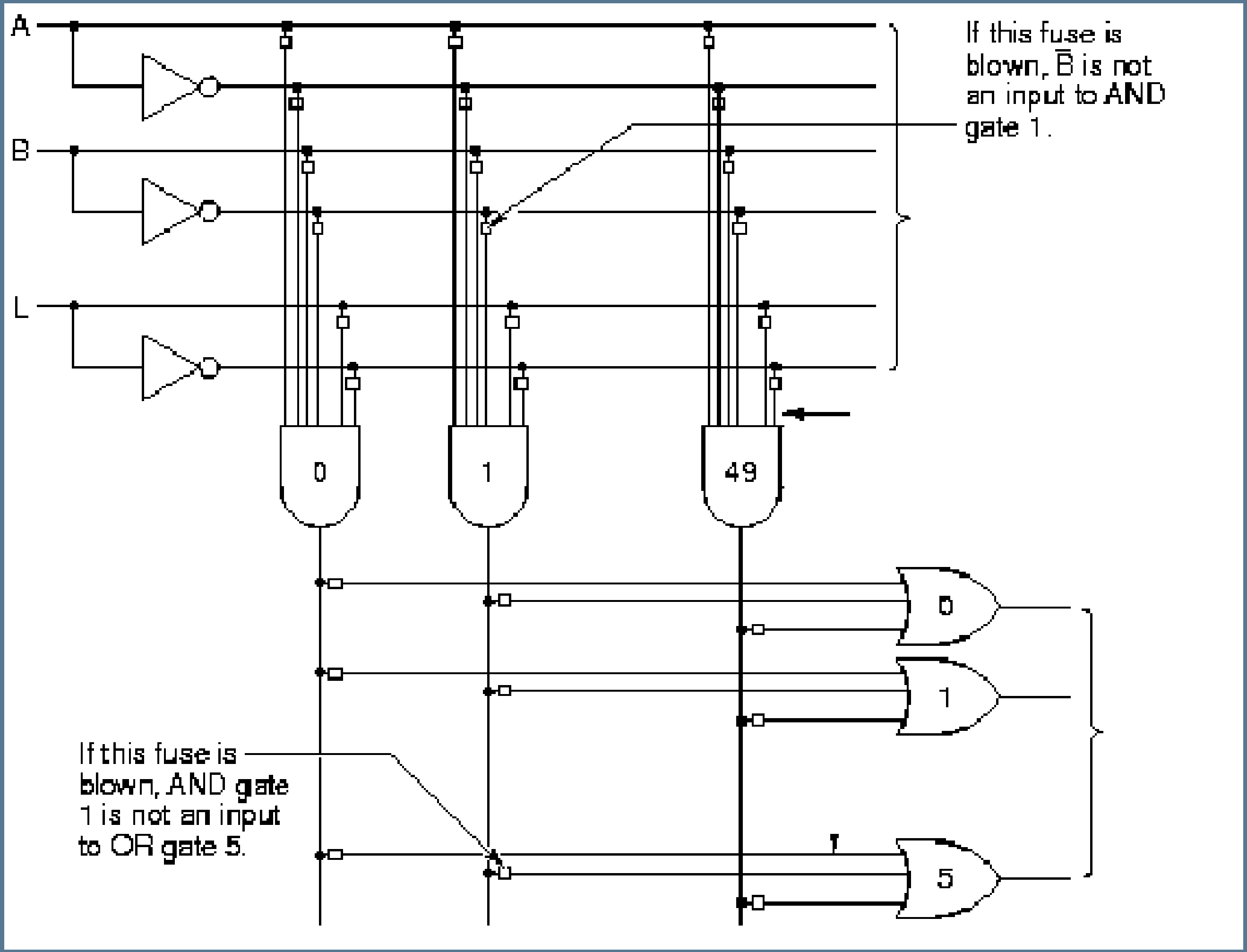


Programmable Logic Devices

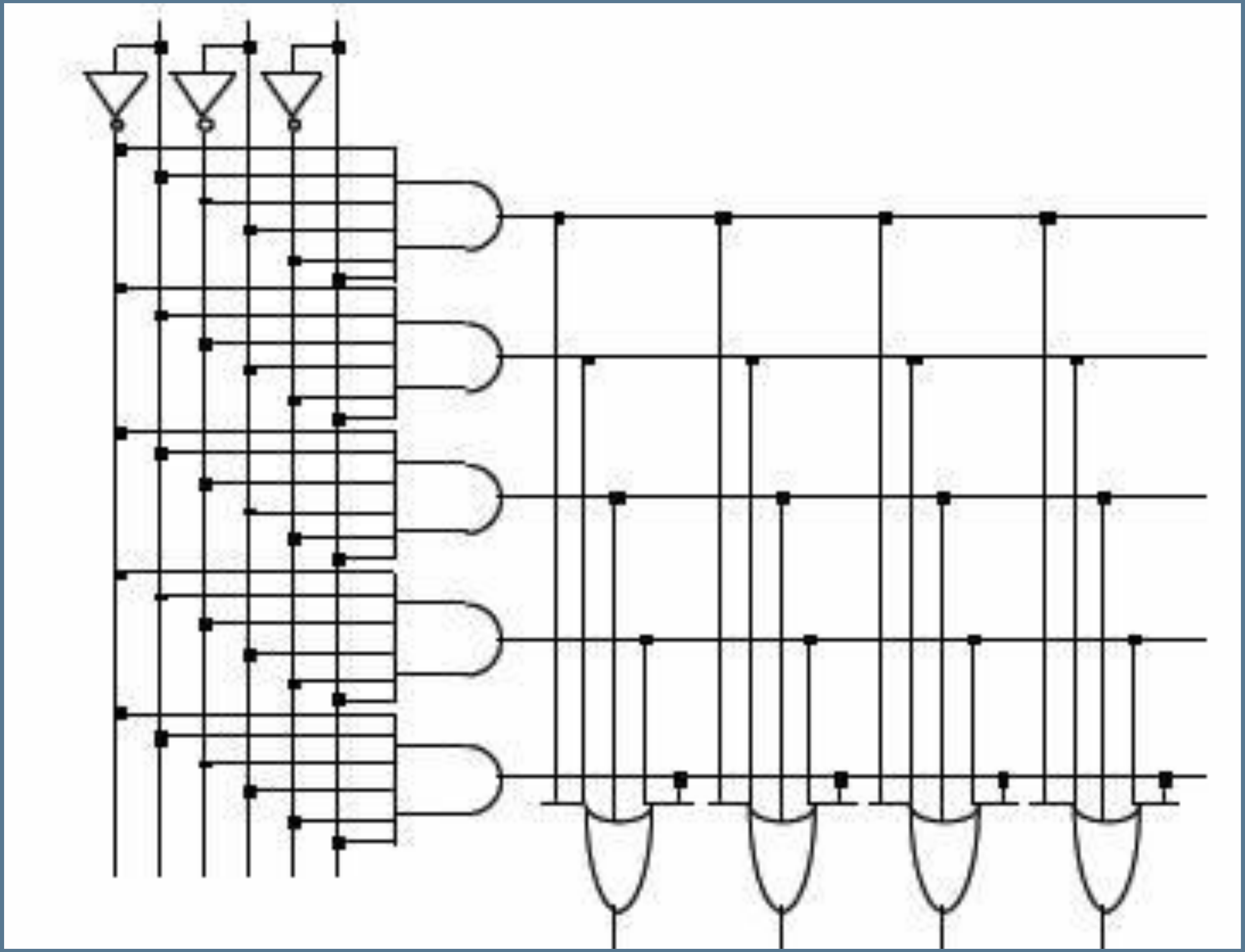
Programmable Logic Array (PLA): The AND array and the OR array are *programmable.*



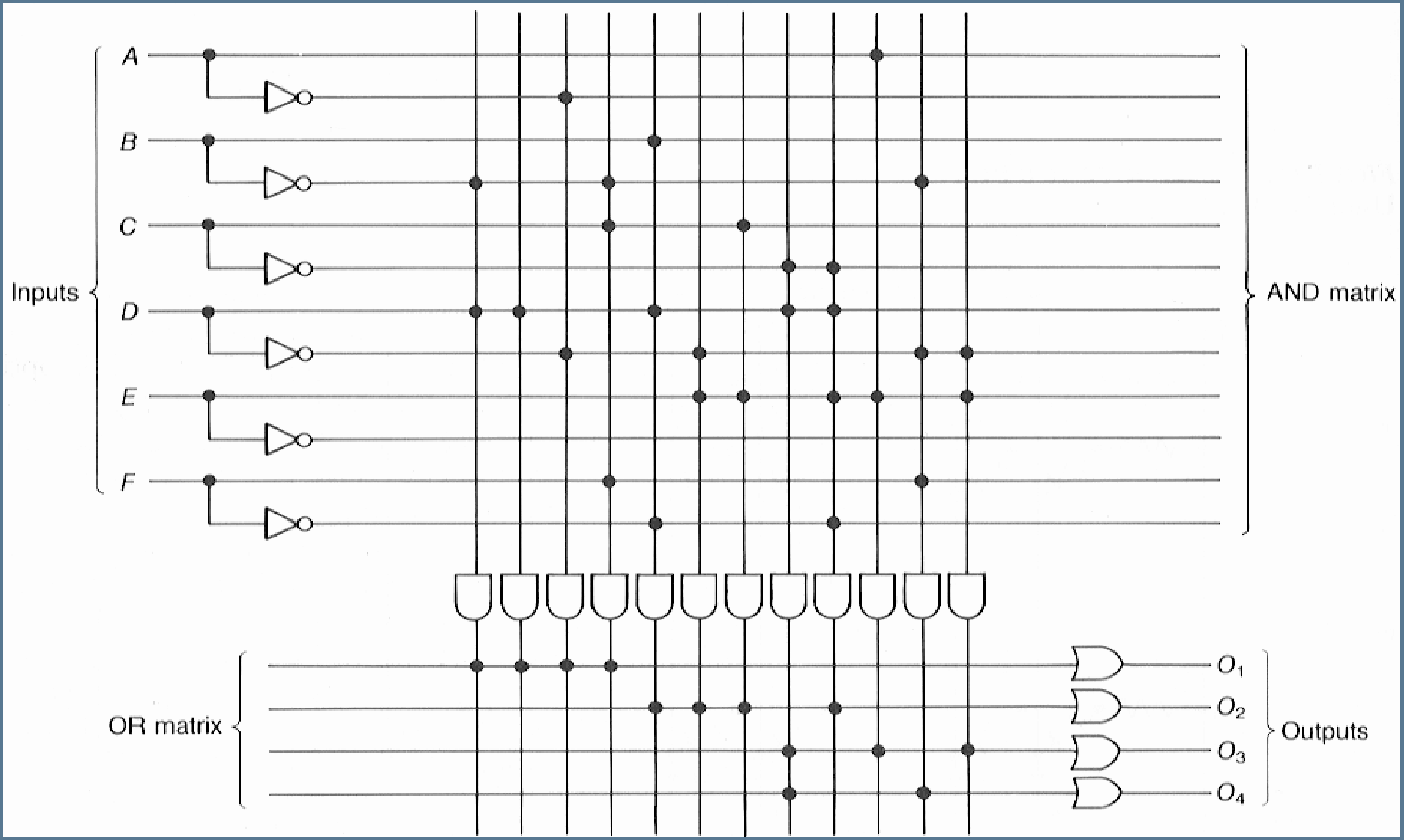
PLA


A
B
L
0
1
49
If this fuse is blown, $\overline{B}$ is not an input to AND gate 1.
0
1
5
If this fuse is blown, AND gate 1 is not an input to OR gate 5.

PLA



PLA



PLAs

- Metal fuses are used by the manufacturer to establish the connections among the gates
- Kinds of metal fuses:
  - blown fuse
  - intact fuse
- Programming a PLA: “blowing” fuses for unwanted connections.

Programming a PLA

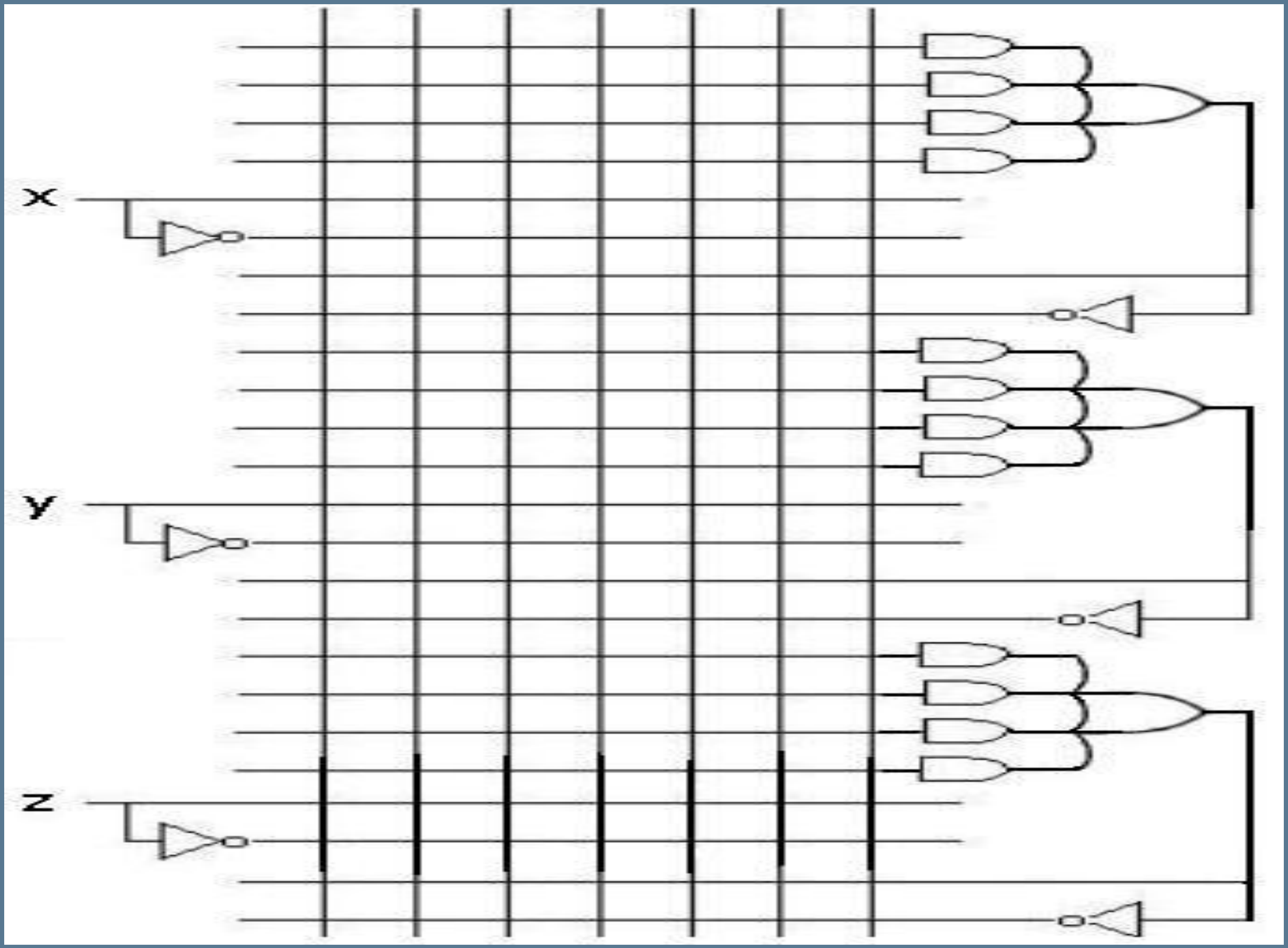
- A PLA program table is needed to program a PLA
- The size of a PLA is determined by three factors:
  - # of inputs, # of product terms, # of outputs

- Types of PLA
  - mask programmable
    - designer submits a PLA program table to manufacturer
  - field programmable
    - PLA can be programmed by the designer using special instruments

Programmable Logic Devices

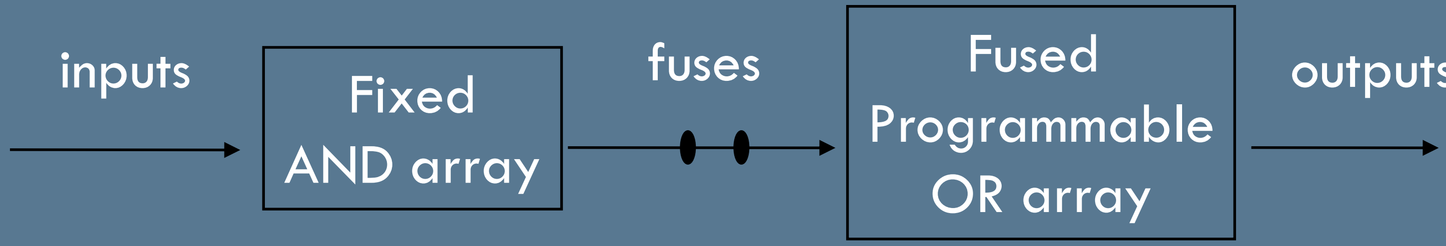
Programmable Array Logic (PAL): The AND array is programmable but the OR array is fixed

PAL



Programmable Logic Devices

- Programmable Read-Only Memory(PROM): The AND array is fixed and the OR array is programmable.



Conclusions/Key ideas

- Combinational circuit outputs depend only on current input.
- Design aims: cheaper circuit, faster circuit, simpler to design
- Simple circuits can be designed using function simplification methods (K-map or more complicated computer aided methods)
- More complicated circuits can be designed by decomposing it into simpler circuits
- Signals suffer propagation delays as they go through gates