

ARCHITETTURA DEGLI ELABORATORI

A.A. 2020-2021

Università di Napoli Federico II
Corso di Laurea in Informatica

Docenti

Proff.

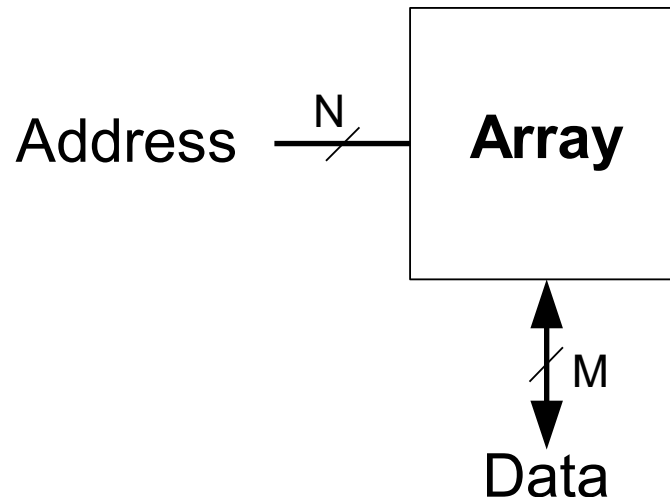
Luigi Sauro gruppo 1 (A-G)

Silvia Rossi gruppo 2 (H-Z)



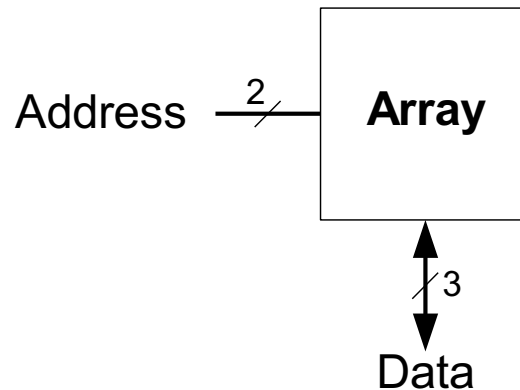
Memory Arrays

- Memorizzare efficacemente una grossa quantità di dati
- 3 tipologie:
 - Dynamic random access memory (DRAM)
 - Static random access memory (SRAM)
 - Read only memory (ROM)
- dato a M -bit letto/scritto su di un unico indirizzo a N -bit



Esempio

- $2^2 \times 3$ -bit array
- Numero parole: 4
- Lunghezza parole: 3-bits
- All'indirizzo 10 corrisponde la parola 100

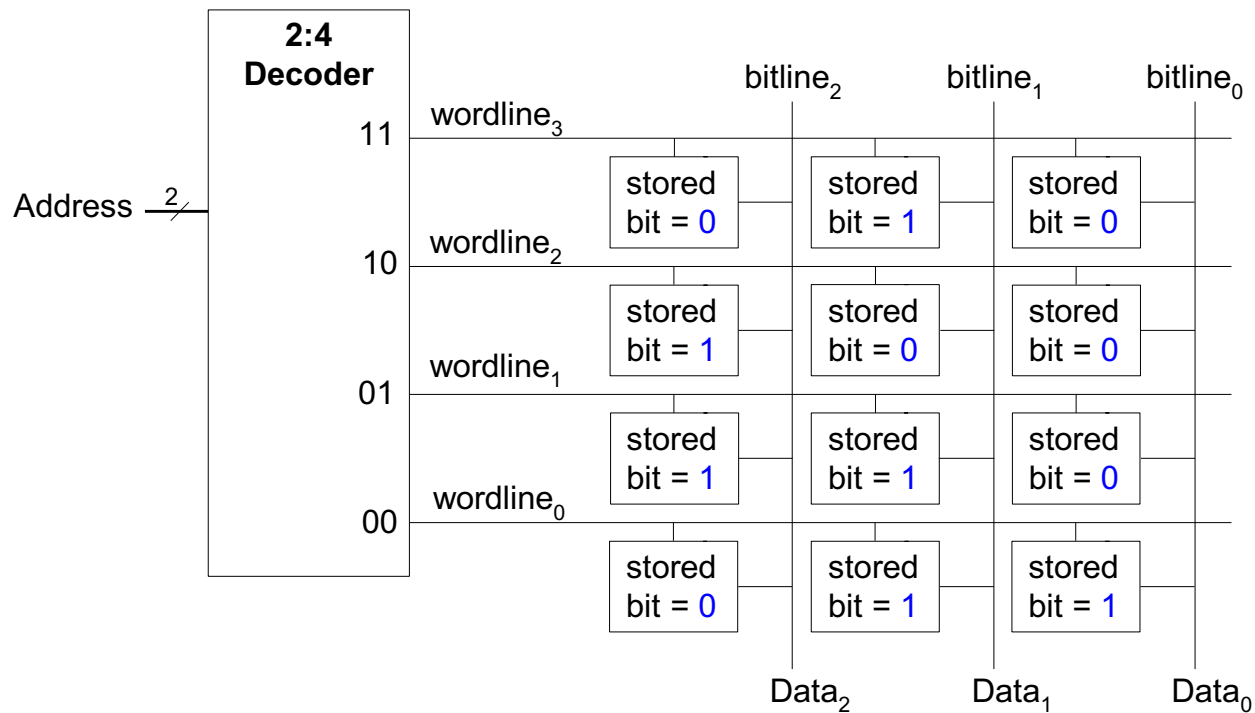


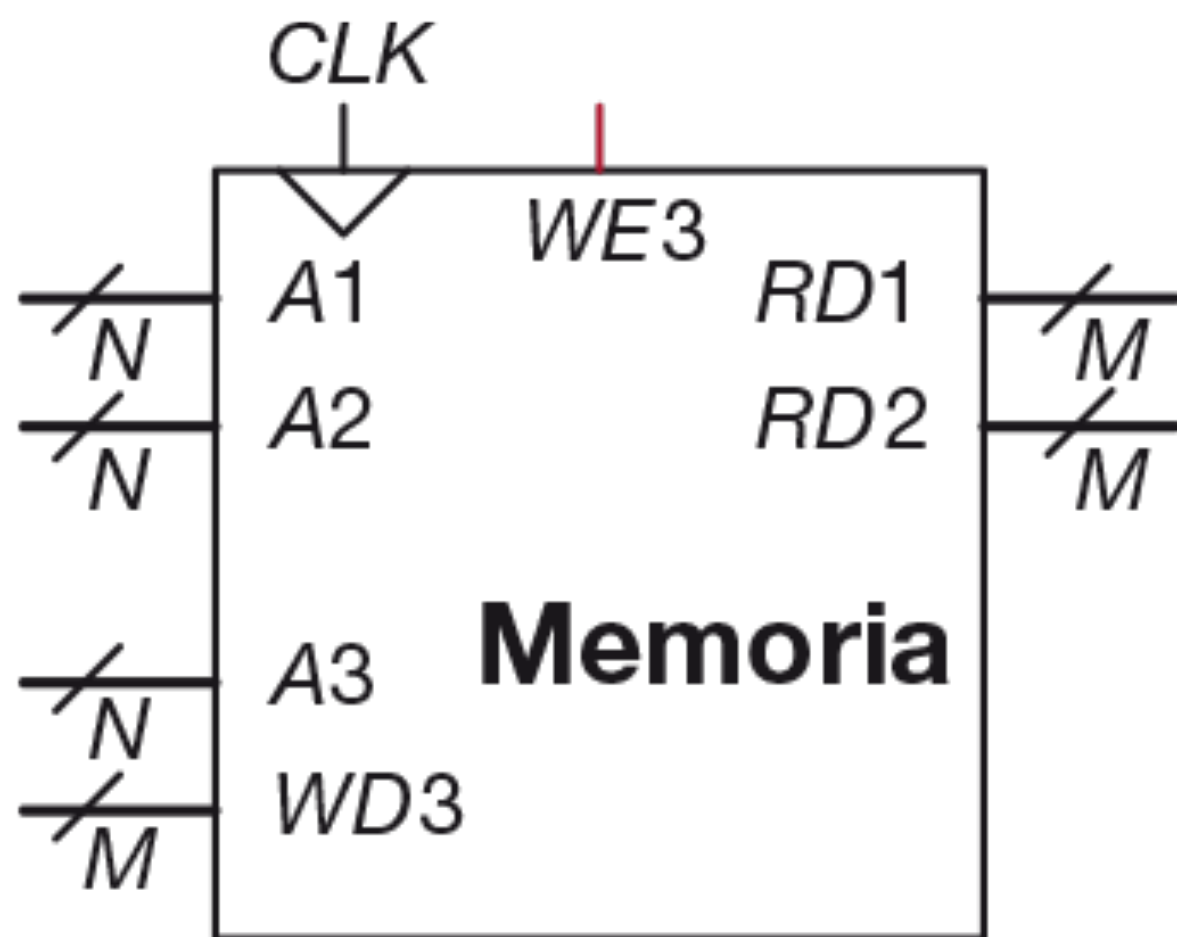
Address	Data			
11	0	1	0	depth ↑ ↓
10	1	0	0	
01	1	1	0	
00	0	1	1	
	width ←→			

Memory Array

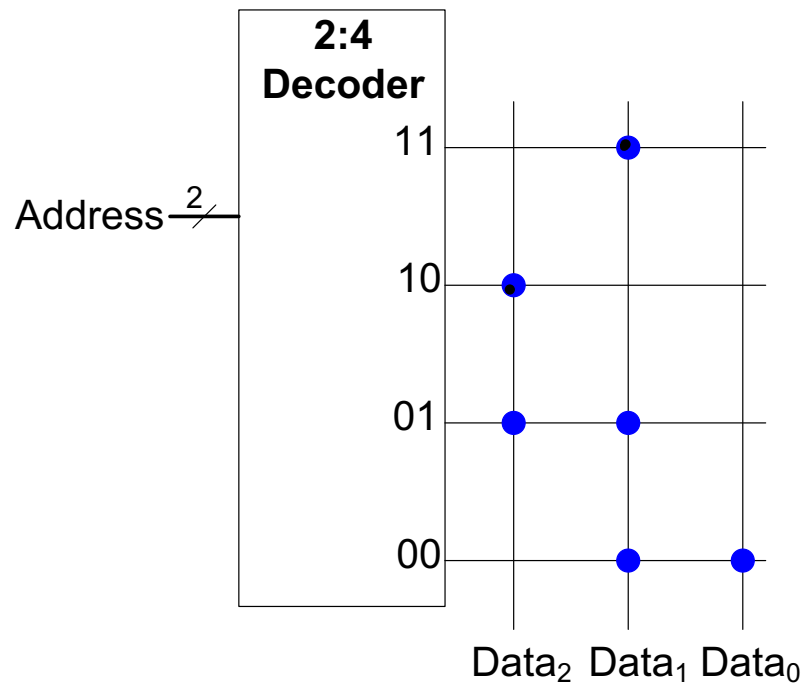
■ Wordline:

- Agisce come un enable
- Seleziona una riga nella memoria
- Corrisponde ad un unico indirizzo
- Solo una wordline per volta è attivata





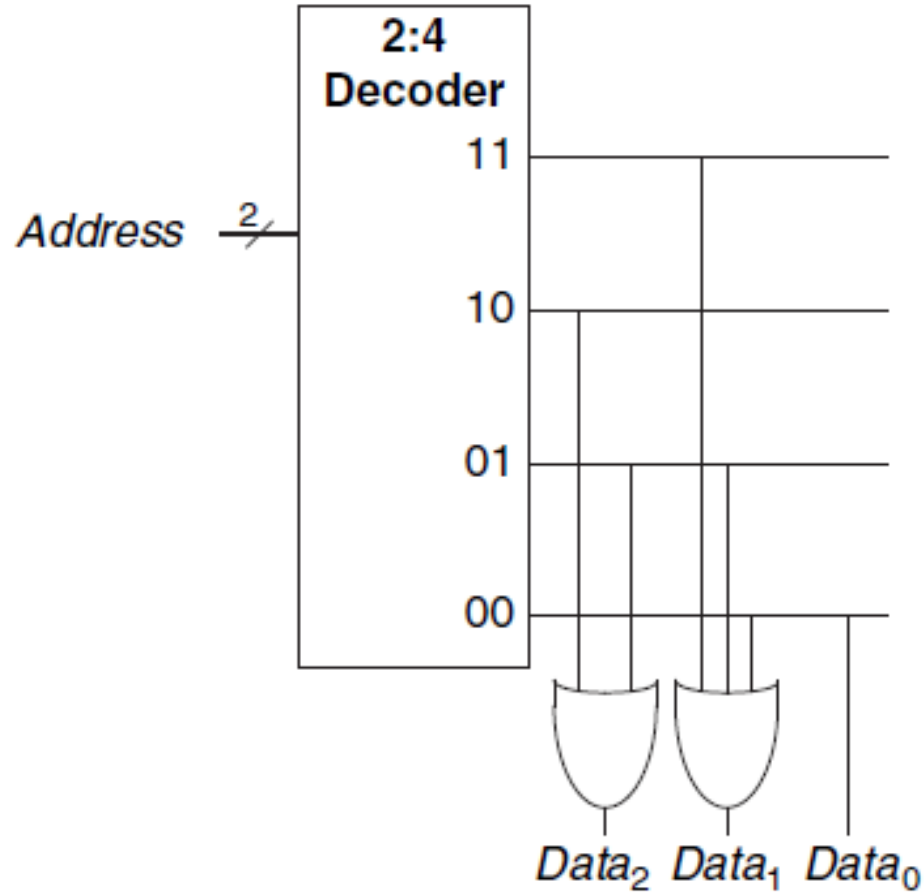
ROM Storage



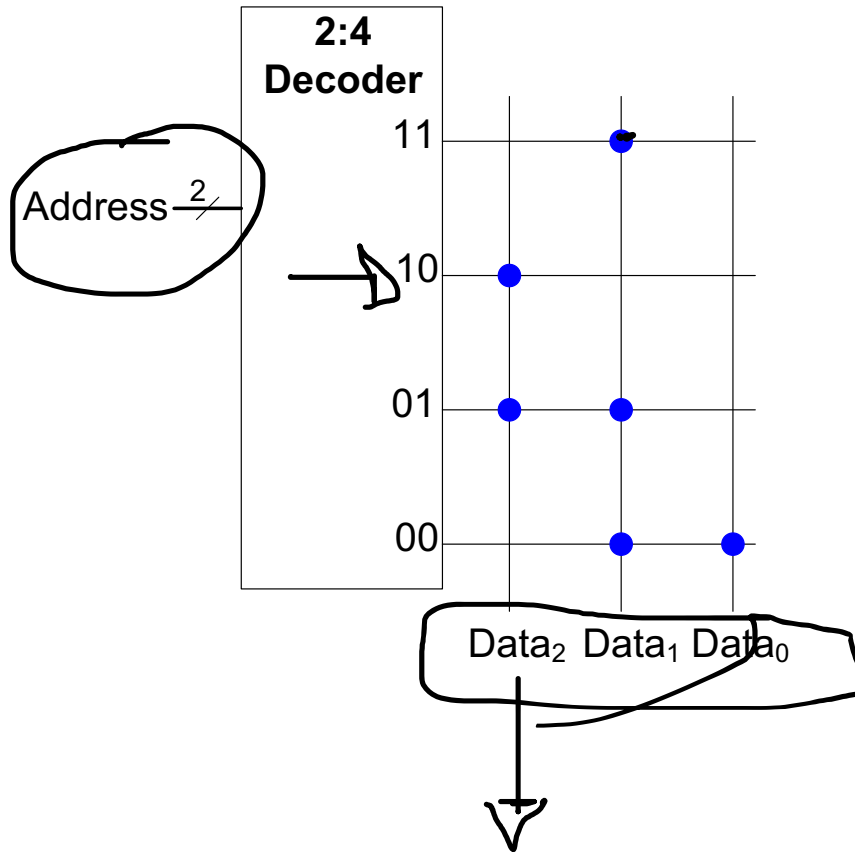
1

Address	Data			depth ↑ ↓
11	0	1	0	
10	1	0	0	
01	1	1	0	
00	0	1	1	
width ←→				

ROM via porte logiche



ROM Logic



A ₁	A ₀	D ₂
1	1	0
1	0	1
0	1	1
0	0	0

$$Data_2 = \underline{A_1 \oplus A_0}$$

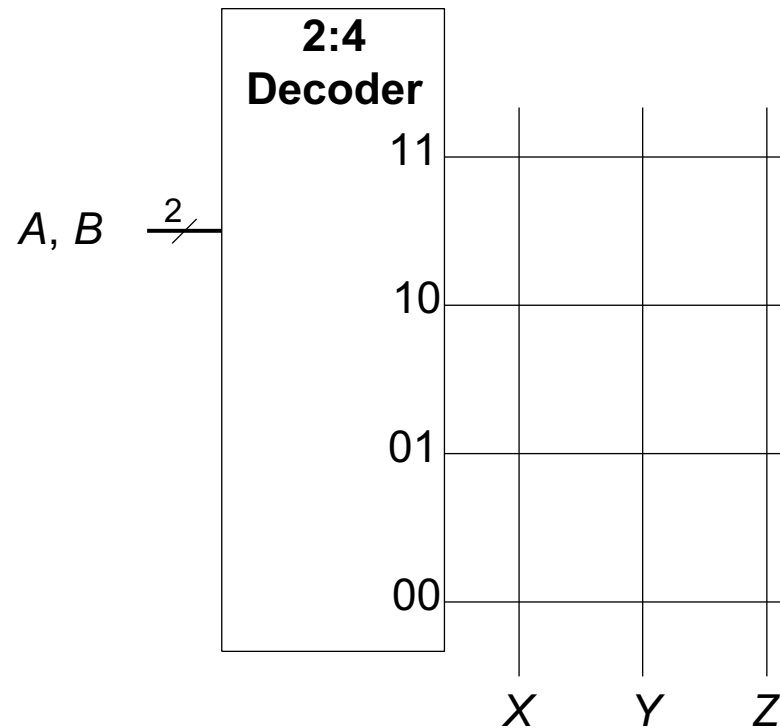
$$Data_1 = \overline{A_1} + A_0$$

$$Data_0 = \overline{A_1} \overline{A_0}$$

Esempio

Usare un $2^2 \times 3$ -bit ROM per implementare funzioni booleane:

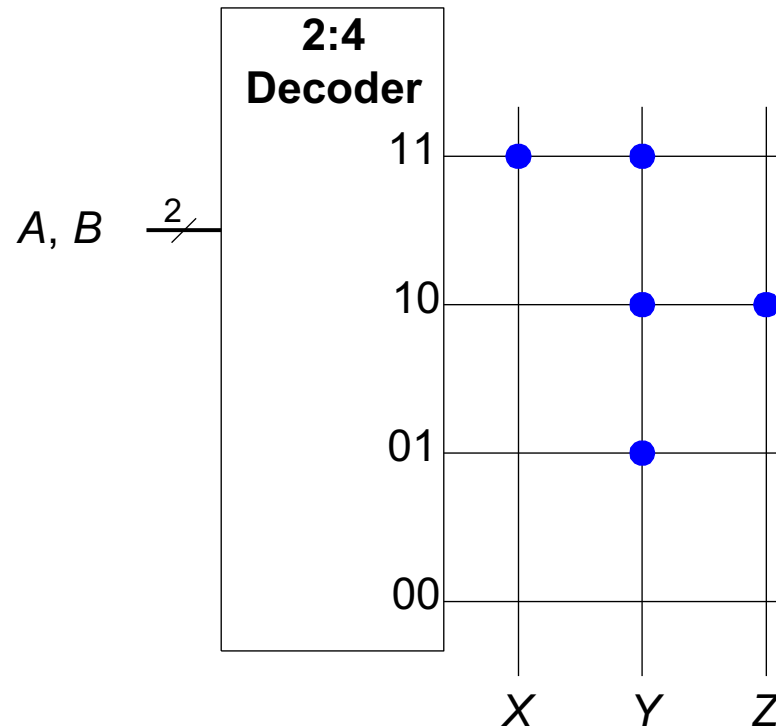
- $X = AB$
- $Y = A + B$
- $Z = A \overline{B}$



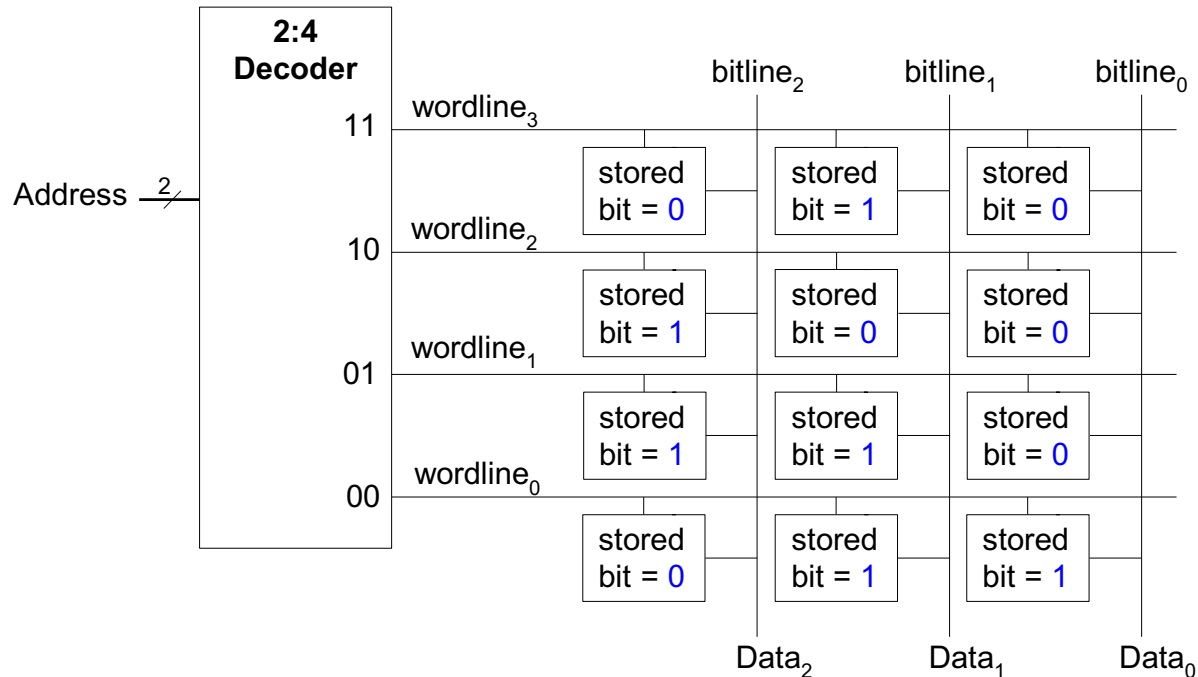
Esempio

Usare un $2^2 \times 3$ -bit ROM per implementare funzioni booleane:

- $X = AB$
- $Y = A + B$
- $Z = A \overline{B}$



Lookup tables (LUTs)



$$Data_2 = A_1 \oplus A_0$$

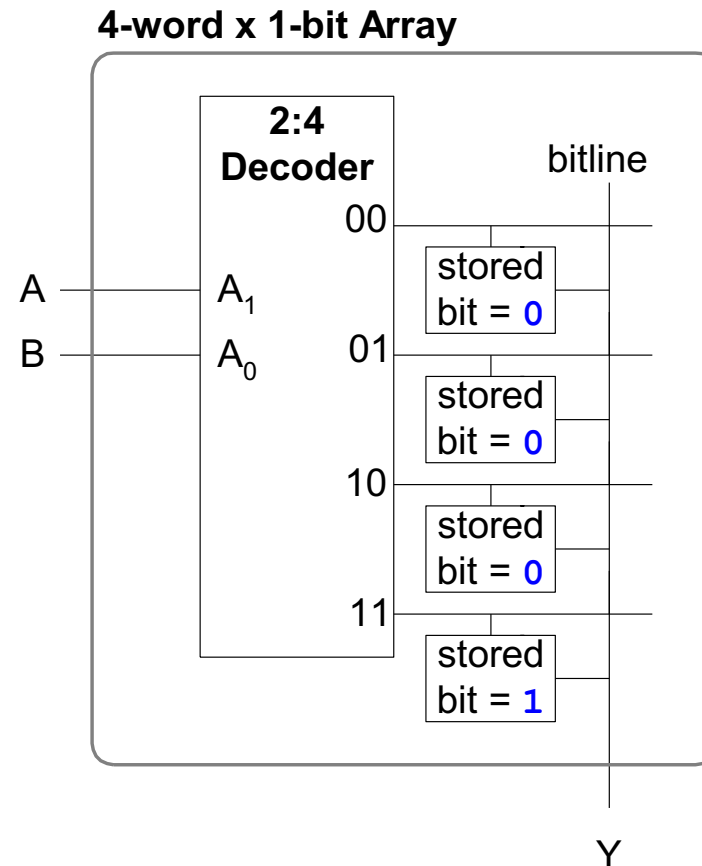
$$Data_1 = \bar{A}_1 + A_0$$

$$Data_0 = \bar{A}_1 \bar{A}_0$$

Logiche con memory array

Sono chiamate *lookup tables* (LUTs): si “guardano” gli output per ogni combinazione di input (indirizzo)

Truth Table		
A	B	Y
0	0	0
0	1	0
1	0	0
1	1	1

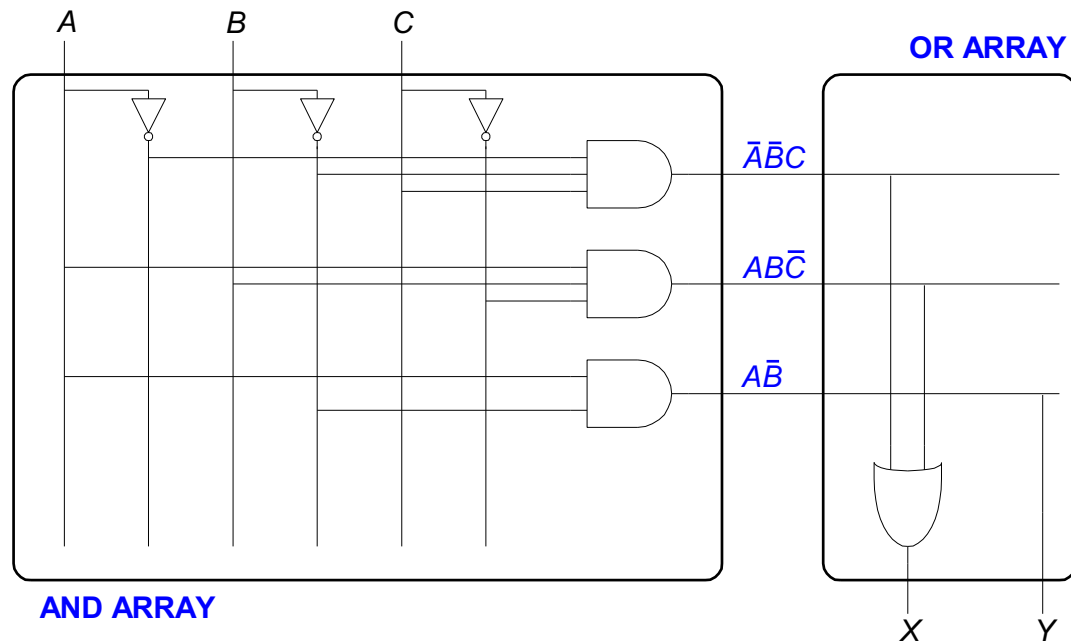
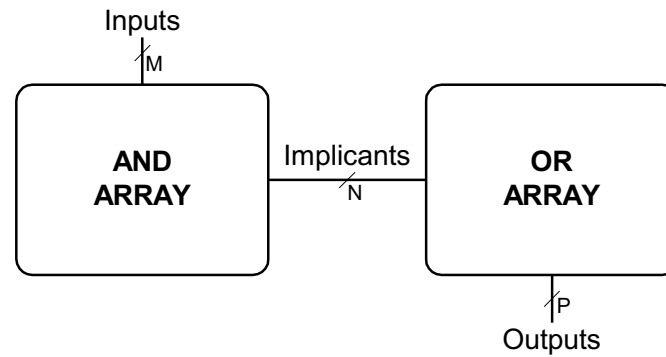


Logic Arrays

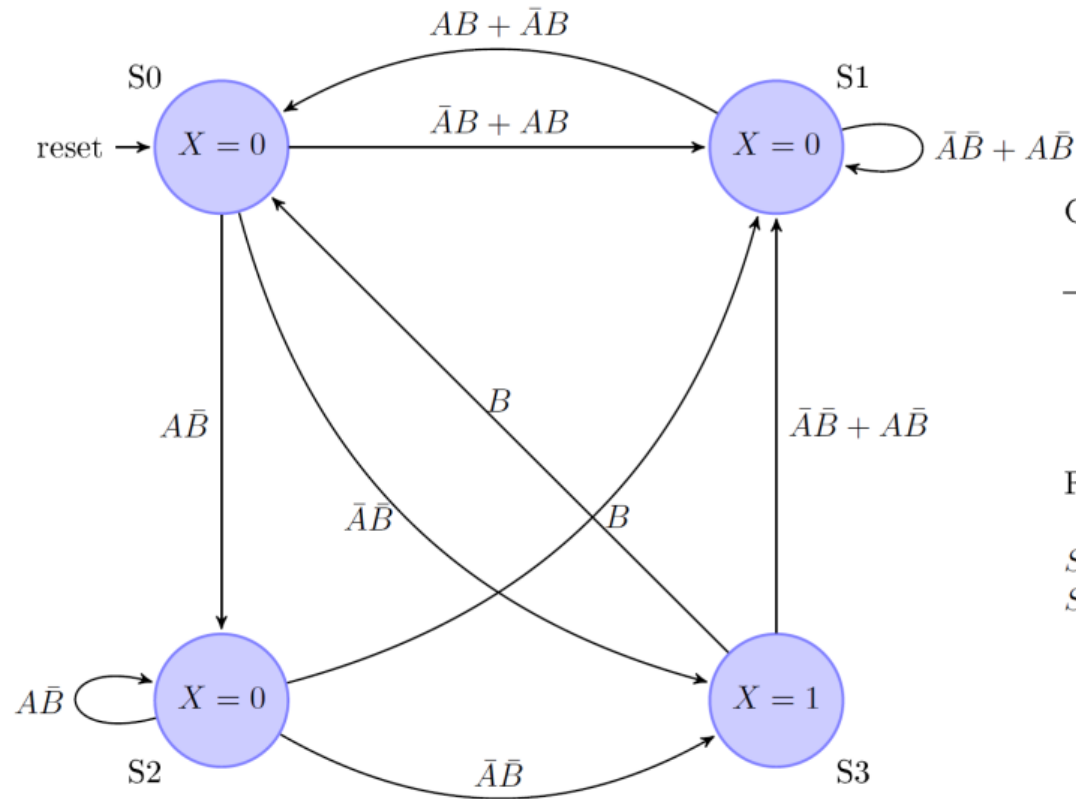
- **PLAs** (Programmable logic arrays)
 - AND array seguito da un OR array
 - solo circuiti combinatori (SOP)
 - Le connessioni interne sono fisse
- **FPGAs** (Field programmable gate arrays)
 - Array elementi logici (LE)
 - Circuiti combinatori e sequenziali
 - Programmable internal connections

PLAs

- $X = \overline{A}\overline{B}C + A\overline{B}\overline{C}$
- $Y = A\overline{B}$



Il seguente diagramma di transizione per una macchina di Moore ha due input A e B e un output X . Indicare le formule SOP **minime** relative alle due variabili di stato (S_1 e S_0).



Codifica dello stato:

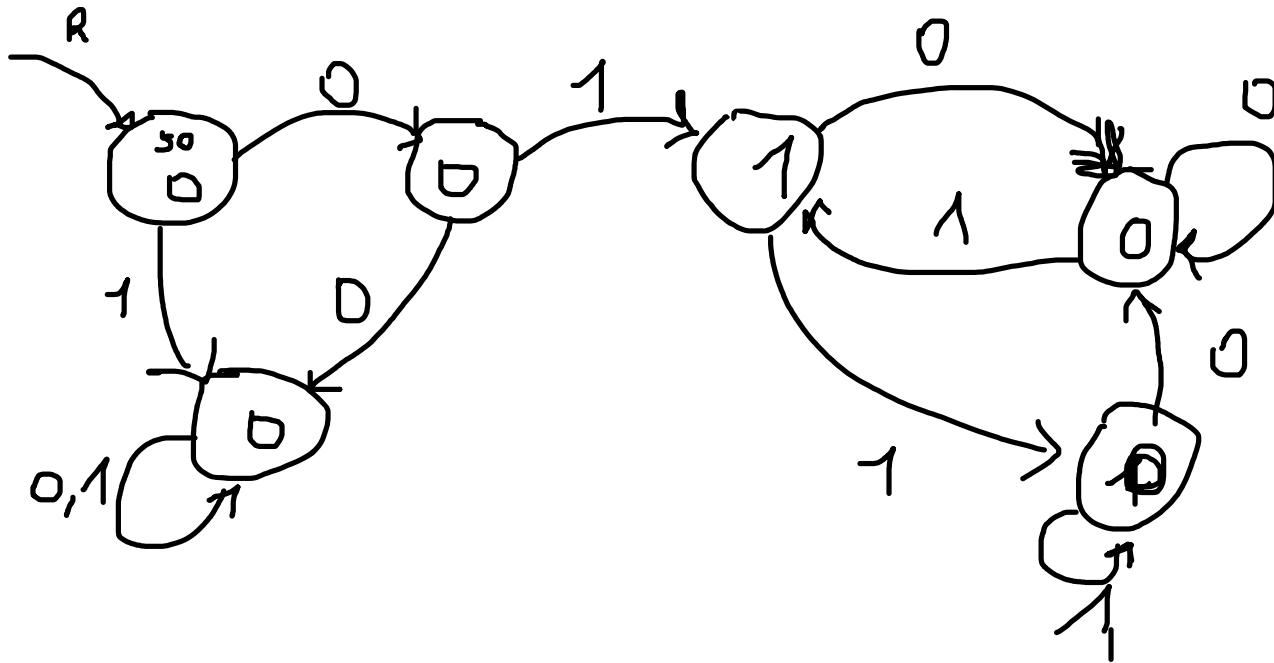
stato	S_1	S_0
S0	0	0
S1	0	1
S2	1	0
S3	1	1

Formule minime SOP:

$$S'_1: \bar{B}\bar{S}_0$$

$$S'_0: \bar{A}\bar{B} + B\bar{S}_0 + \bar{B}S_0$$

Si disegni il grafo di un automa di Moore con un solo input ed un solo output che ritorna 1 sse ~~gli ultimi~~ lo stream di ingresso inizia e finisce con la sequenza 01. [Esempi positivi: 01; 0110111101;] [Esempi negativi: 1101; 0111011; 010010]



Si disegni il grafo di un automa di Moore con un solo input ed un solo output che ritorna 1 sse gli ultimi 3 input letti sono 1. [Esempi positivi: 0111; 00111111;] [Esempi negativi: 1011; 01110; 11]