

Corso di Calcolatori Elettronici

Reti Universali

Prof. Roberto Canonico



Università degli Studi di Napoli Federico II
Dipartimento di Ingegneria Elettrica
e delle Tecnologie dell'Informazione

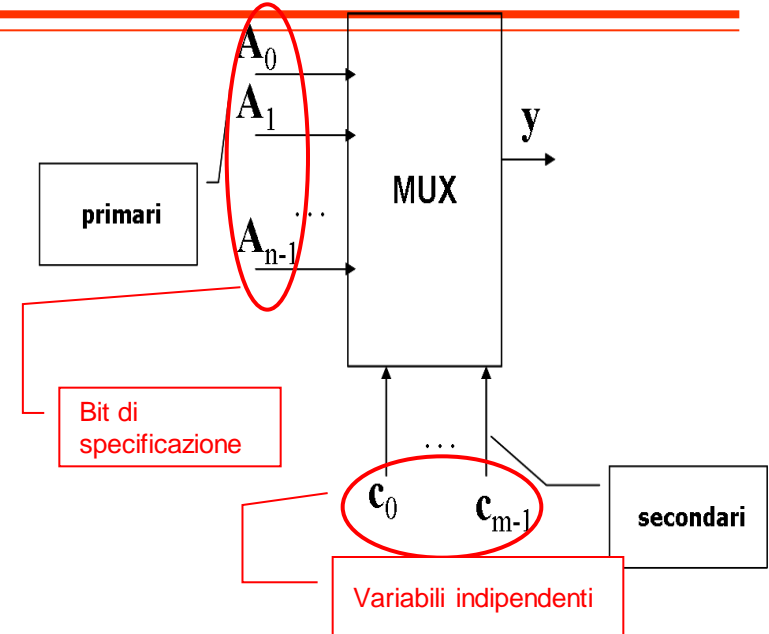
Reti universali: multiplexer

Reti con multiplexer

- Un multiplexer convoglia sulla uscita y quello fra gli $n=2^m$ “ingressi primari” A_i selezionato dagli n “ingressi secondari” C_j

$$y = \sum_{i=0}^{n-1} A_i \cdot P_i$$

- Ma questa è la forma normale di una funzione delle C_j con P_i mintermini e A_i bit che specificano la funzione. Si ottiene allora una rete universale per funzioni di m variabili ponendo:
 - Ingressi primari = bit di specificazione
 - Ingressi secondari = variabili indipendenti



Multiplexer come generatore di funzioni

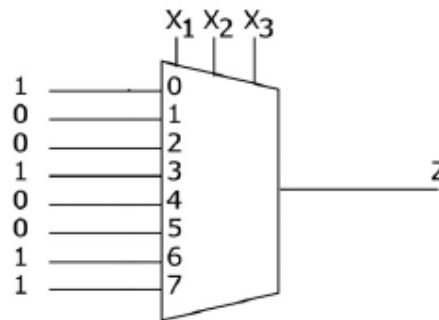
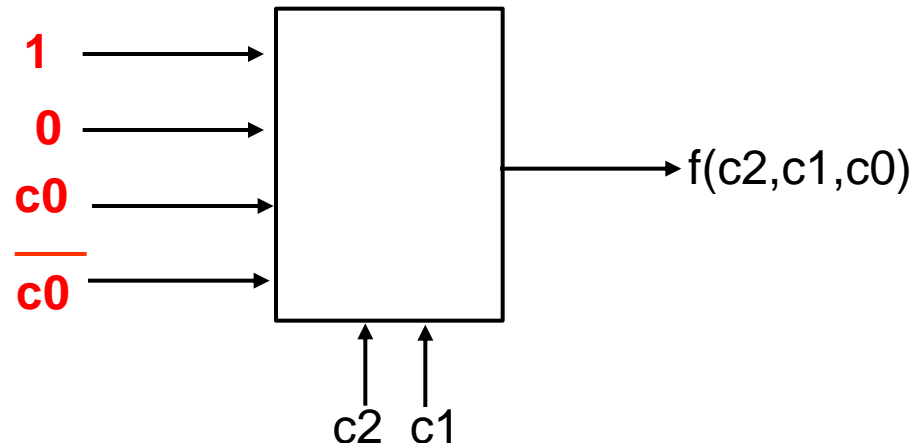


Figura 3.33 - Realizzazione della funzione $z = \sum_3(0, 3, 6, 7)$ tramite multiplexer a otto vie.

Reti universali: multiplexer con logica folded

Logica “folded”

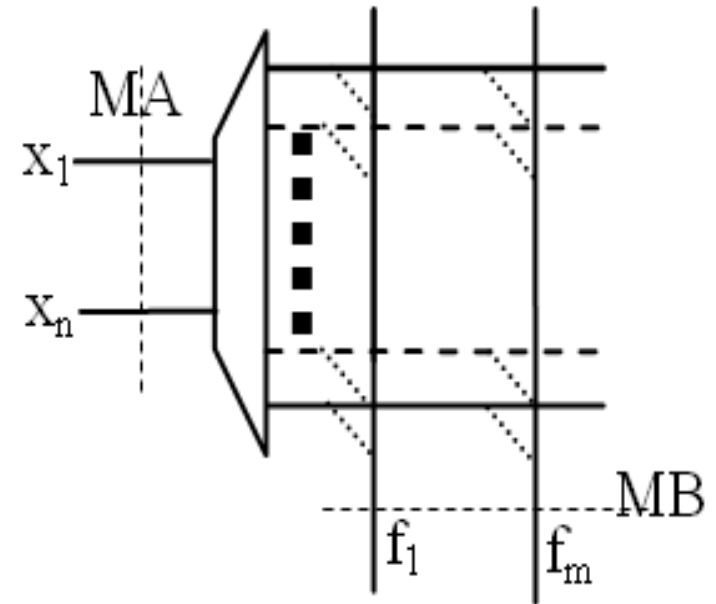
- È possibile usare un multiplexer con n-1 ingressi secondari per generare funzioni di n variabili
- Esempio: $f(c_2, c_1, c_0) = P_0 + P_1 + P_5 + P_6$
- $$f = \overline{c_2} \cdot \overline{c_1} \cdot \overline{c_0} + \overline{c_2} \cdot \overline{c_1} \cdot c_0 + c_2 \cdot \overline{c_1} \cdot c_0 + c_2 \cdot c_1 \cdot \overline{c_0} =$$
$$= \overline{c_2} \cdot \overline{c_1} \cdot \mathbf{1} + \overline{c_2} \cdot c_1 \cdot \mathbf{0} + c_2 \cdot \overline{c_1} \cdot \mathbf{c_0} + c_2 \cdot c_1 \cdot \mathbf{\overline{c_0}}$$



Reti universali: ROM

Logica ROM-Read Only Memory

- Una ROM implementa la funzione **$MB = M(MA)$**
- Se MA è di n bit
- e la memoria ha parallelismo di m bit
- Si realizzano m funzioni di n variabili
- I contenuti della memoria (colonne di tabelle di verità) sono
 - specificati dall'utente
 - realizzati dal costruttore “bruciando” i collegamenti fra linee orizzontali e verticali



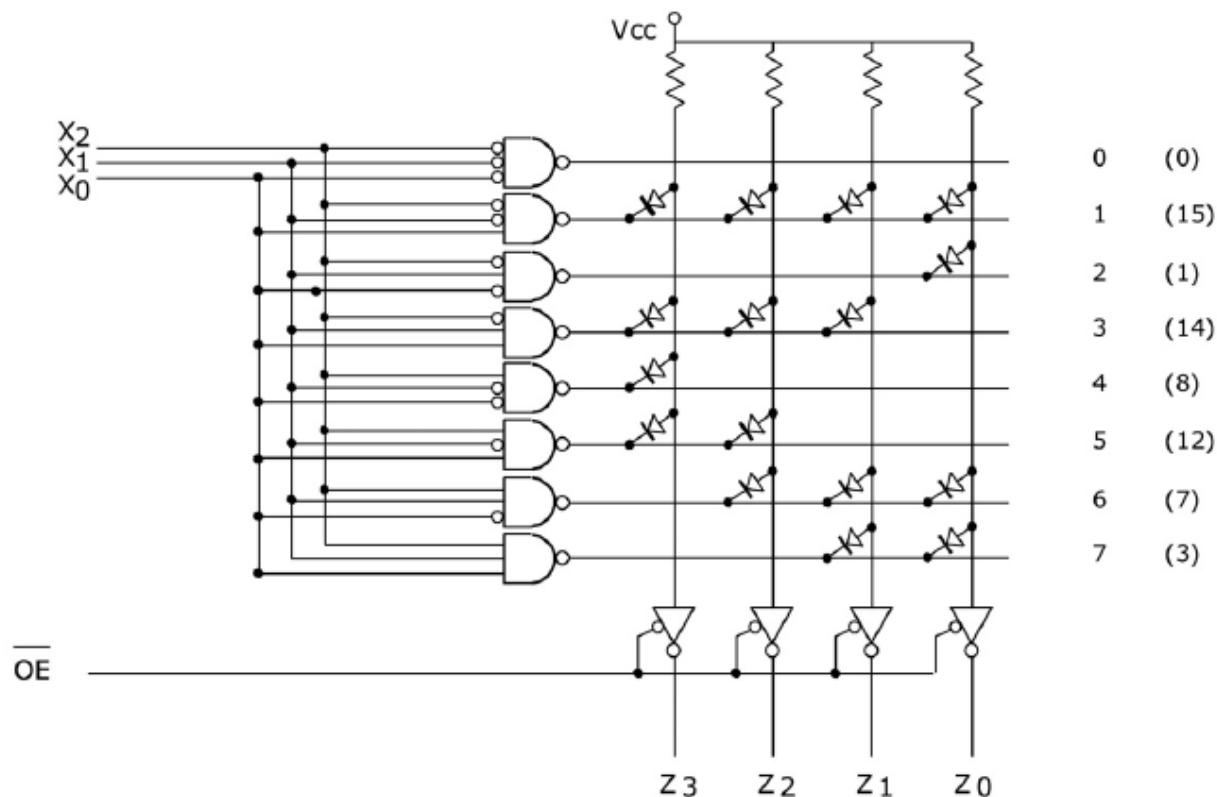


Figura 3.38 - Schema di una ROM di otto posizioni di 4 bit. A destra viene riportato l'indirizzo di ciascuna cella e, tra parentesi, il suo contenuto. Per capirne il funzionamento si consideri, per esempio, il caso in cui l'indirizzo $x_2x_1x_0$ sia 2. In tal caso solo la riga 2 della matrice è asserita bassa, per cui la tensione sull'anodo del diodo a essa collegato (colonna z_0) risulta bassa (il diodo ha una caduta di tensione trascurabile). Gli altri diodi sulla colonna z_0 risultano interdetti. I diodi sulle restanti colonne sono in conduzione, ma la tensione sui loro anodi è alta essendo alta la tensione su tutte le righe diverse dalla 2. Si noti che la memoria ha un segnale di abilitazione delle uscite.

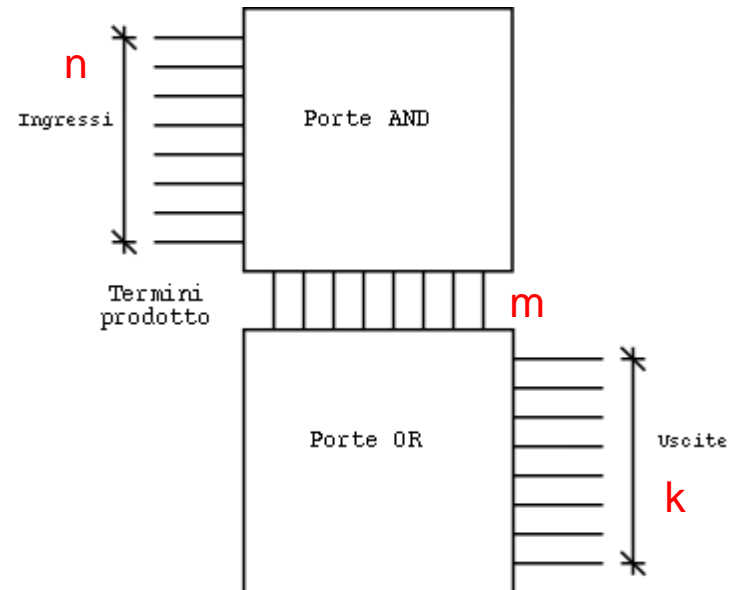
Reti universali: PLA

PLA – Programmable Logic Array

- Fanno parte della più ampia classe dei “Dispositivi Logici Programmabili”
- Realizzare k funzioni di n variabili in forma and-or a 2 livelli

$$f_j = \sum_{i=1}^m a_{ij} \cdot \gamma_i \quad a_{ij} = 0,1; i \in [1,m], j \in [1,k]$$

- 2 sezioni
 - AND: realizza le m clausole
 - OR: realizza le k sommatorie
 - Le specifiche (γ_i e a_{ij})
 - fornite da utente
 - realizzate dal costruttore attivando i collegamenti per le AND e le OR



PLA

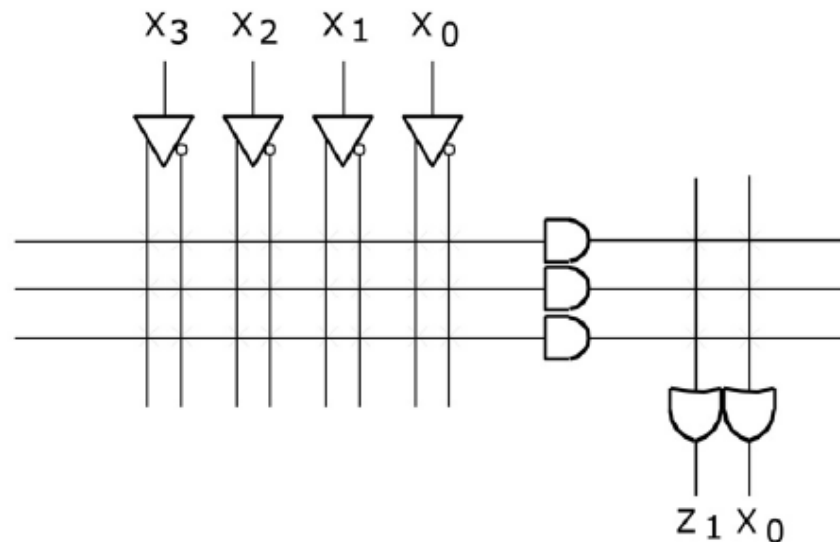
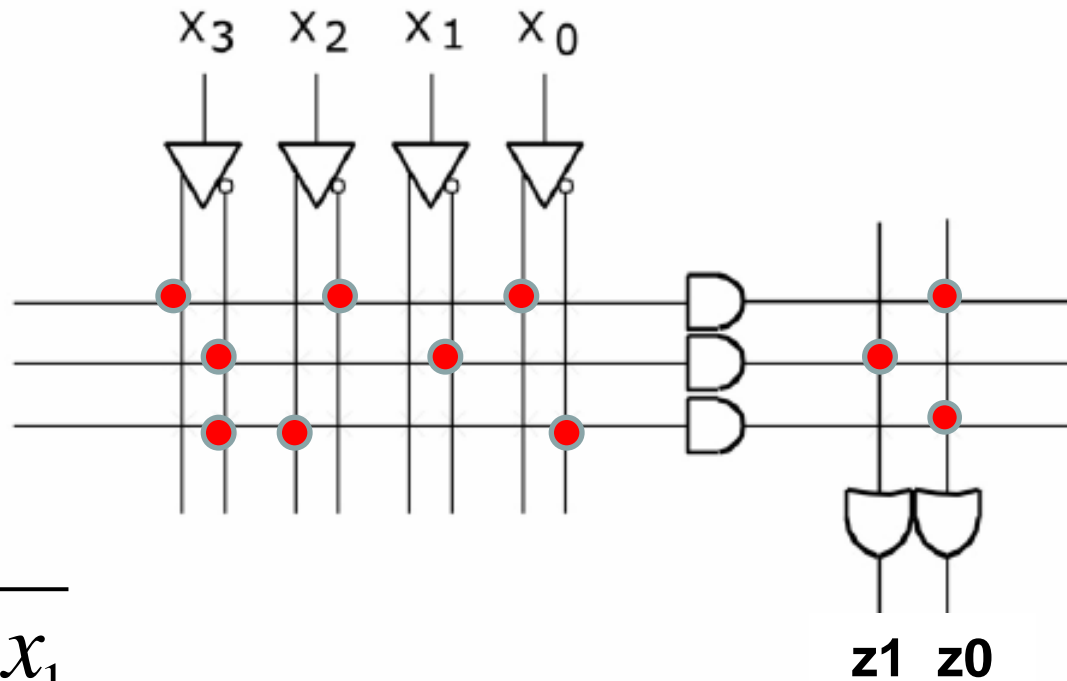


Figura 3.41 - Schema di un dispositivo PLA a quattro ingressi e due uscite. Il dispositivo prevede una matrice di AND e una matrice di OR, ambedue programmabili in campo.

PLA: esempio di programmazione



$$z_1 = \overline{x_3} \cdot \overline{x_1}$$

$$z_0 = x_3 \cdot \overline{x_2} \cdot x_0 + \overline{x_3} \cdot x_2 \cdot \overline{x_0}$$
