

Corso di Calcolatori Elettronici I

A.A. 2011-2012

Macchine combinatorie: encoder/decoder e multiplexer/demultiplexer

Lezione 12

Prof. Antonio Pescapè

Università degli Studi di Napoli Federico II
Facoltà di Ingegneria
Corso di Laurea in Ingegneria Informatica



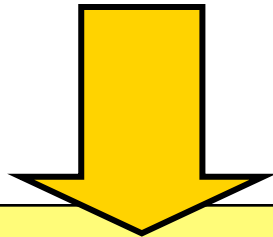
Analisi e Sintesi di un sistema 1/2

- Per *analisi* di un sistema si intende l'individuazione delle relazioni di causa/effetto tra i segnali di ingresso e uscita, attraverso l'esame di una rappresentazione schematica dei suoi componenti elementari e dei collegamenti che li interconnettono, ovvero:
 - *data la rappresentazione schematica del sistema, individuarne il comportamento.*
 - Per *sintesi* di un sistema si intende l'individuazione dei componenti e delle interconnessioni necessarie per realizzarlo seguendo la preassegnata specifica funzionale:
 - *data la specifica funzionale individuarne la struttura.*
-

Analisi e Sintesi di un sistema 2/2

Analisi

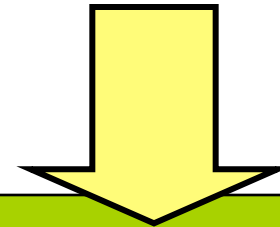
Data la descrizione
della
STRUTTURA
(come è fatta)



Determinarne il
COMPORAMENTO
(cosa fa)

Sintesi

Data la descrizione
del
COMPORAMENTO
(cosa deve fare)



Determinarne la
STRUTTURA
(come è fatta)

Tassonomia dei circuiti digitali

- I circuiti digitali possono essere classificati in due categorie
 - **Circuiti combinatori**
 - Il valore delle uscite ad un determinato istante dipende unicamente dal valore degli ingressi in quello stesso istante.
 - **Circuiti sequenziali**
 - Il valore delle uscite in un determinato istante dipende sia dal valore degli ingressi in quell'istante sia dal valore degli ingressi in istanti precedenti
 - Per definire il comportamento di un circuito sequenziale è necessario tenere conto della storia passata degli ingressi del circuito
 - La definizione di circuito sequenziale implica due concetti:
 - **Il concetto di tempo**
 - **Il concetto di stato**
-

Macchine combinatorie

Reti logiche con n ingressi x_1, x_2, \dots, x_n e m uscite y_1, y_2, \dots, y_m che realizzano la corrispondenza:

$$y_1 = f_1(x_1, x_2, \dots, x_n)$$

.....

$$y_m = f_m(x_1, x_2, \dots, x_n)$$



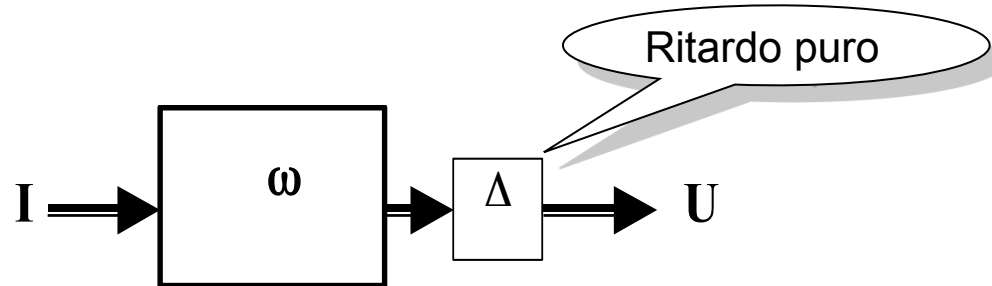
Macchine combinatorie

- Una macchina combinatoria è una rete logica con n ingressi (x_1, x_2, \dots, x_n) ed m uscite (y_1, y_2, \dots, y_m) ed è tale che ad ogni insieme di valori degli ingressi corrisponde un preciso insieme di valori delle uscite
 - Il comportamento di una rete combinatoria $n \times m$ può essere descritto tramite:
 - » una tabella di verità in cui viene specificato il valore dell'uscita per ognuna delle possibili combinazioni dei valori degli ingressi
 - » m funzioni booleane, una per ogni uscita, ciascuna delle quali esprime il valore della corrispondente variabile di uscita in funzione delle n variabili di ingresso
-

I limiti delle macchine reali

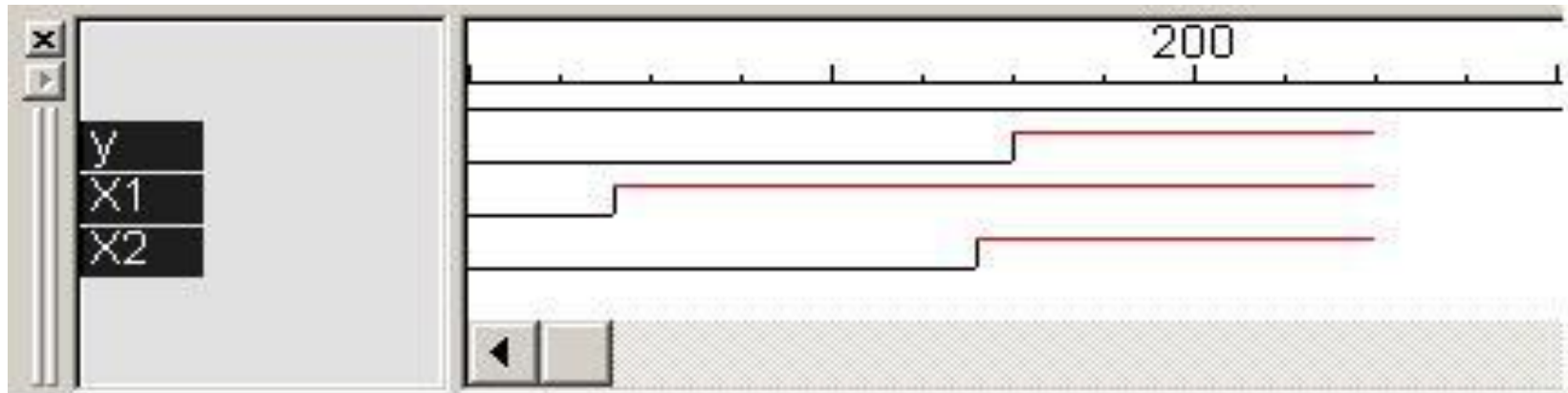
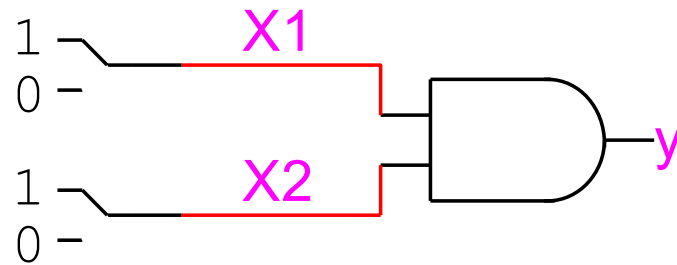
Tempo di risposta

- Una rete ideale reagisce “istantaneamente” ad ogni sollecitazione in ingresso, ovvero $U(t) = \omega(I(t))$
- In una rete reale la variazione dell'uscita a fronte di una variazione degli ingressi avviene con un ritardo Δ (**tempo di risposta**):
$$U(t+\Delta) = \omega(I(t))$$



Il Tempo di risposta

Il Tempo di risposta di una macchina è il ritardo $d = t_f - t_i$ con il quale una variazione sull'ingresso è seguita da una variazione sull'uscita

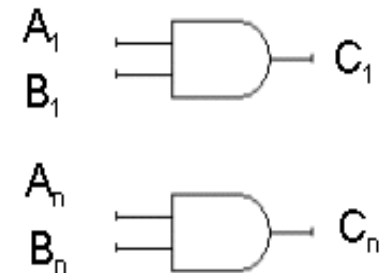
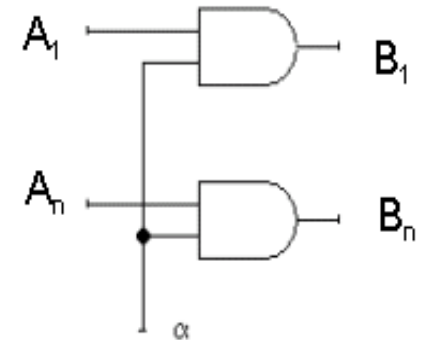
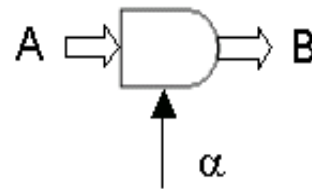


Macchine combinatorie

- In una macchina combinatoria i valori delle uscite dipendono esclusivamente dai valori degli ingressi
 - macchina combinatoria ideale: tale dipendenza è istantanea
 - macchina combinatoria reale: presenza di ritardo tra l'istante in cui c'è una variazione in uno degli ingressi e l'istante in cui l'effetto di questa variazione si manifesta sulle uscite
 - E' importante notare come
 - ciascuna y_i può essere decomposta in funzioni componenti
 - due distinte y_i possono contenere una identica funzione componente
 - Ciò comporta, ad esempio, una potenziale diminuzione di porte elementari rispetto ad una realizzazione indipendente delle y_i
-

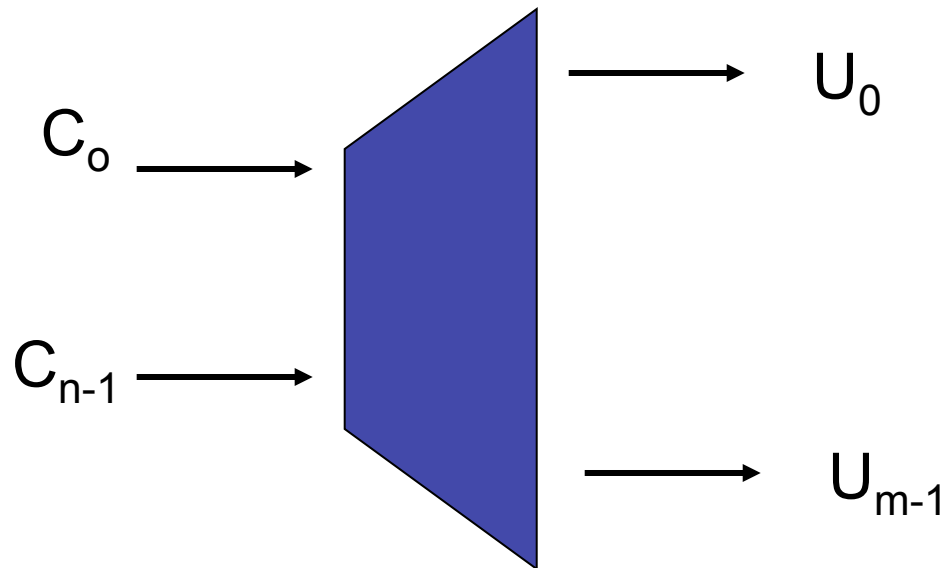
Porte di parola

- Porte con abilitazione:
 - $B = \alpha A = \alpha \text{ AND } A$
- Parola:
 - Vettore di bit
 - $V = \{v_0, v_1, \dots, v_n\}$
- Porta di parola con abilitazione:
 - $\alpha V = \{\alpha v_0, \dots, \alpha v_n\}$
- Porta generica di parola:
 - $A \text{ AND } B =$
 $= \{a_0 \text{ AND } b_0, \dots, a_n \text{ AND } b_n\}$



Decodificatore (decoder) 1 su m

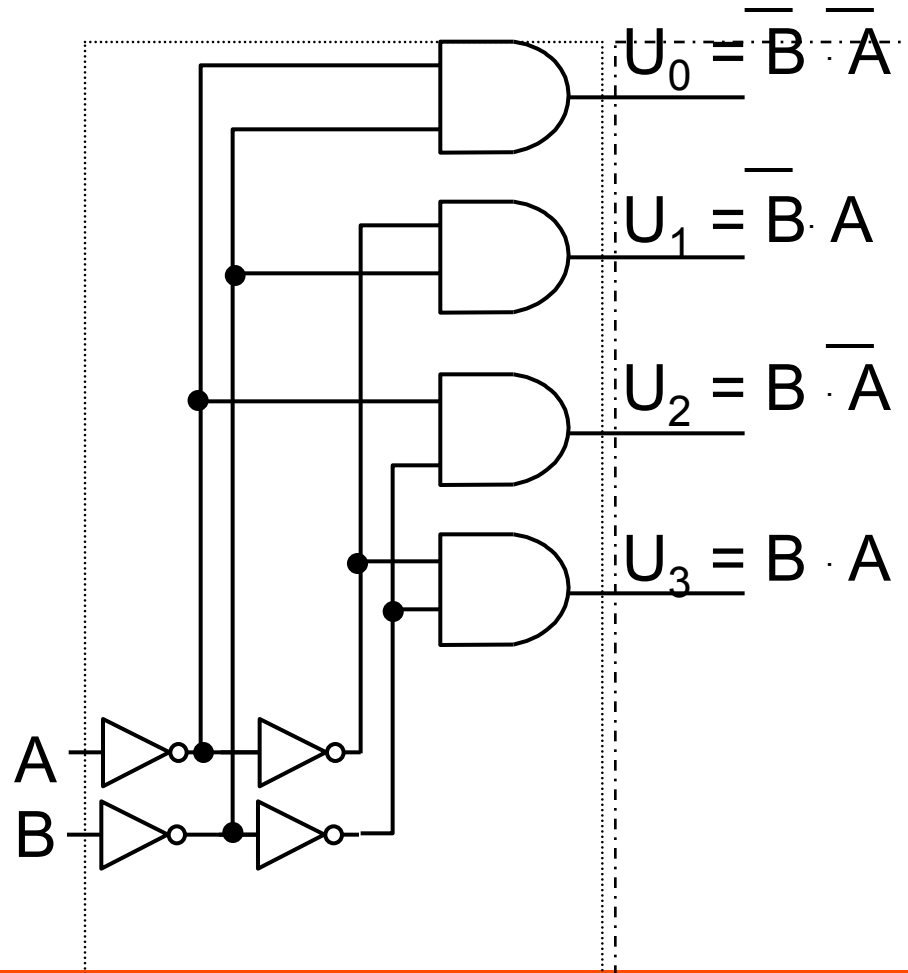
- Un decodificatore è una macchina che riceve in ingresso una parola codice (C) su n bit e presenta in uscita la sua rappresentazione decodificata (linee U_0, \dots, U_{N-1}) su $m=2^n$ bit



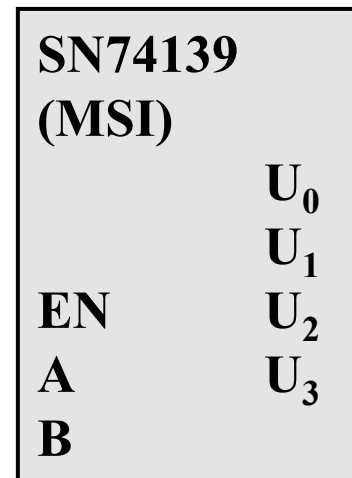
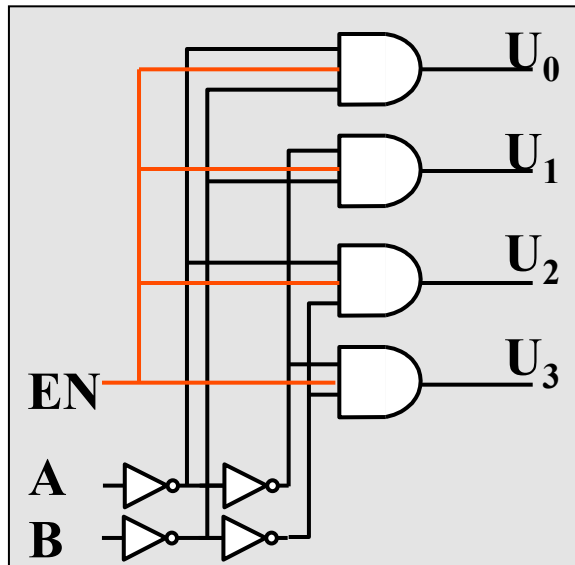
Decoder 1 su 4

Esempio: decoder 2:4

	B	A	U_0	U_1	U_2	U_3
1	0	0	1	0	0	0
2	0	1	0	1	0	0
3	1	0	0	0	1	0
4	1	1	0	0	0	1

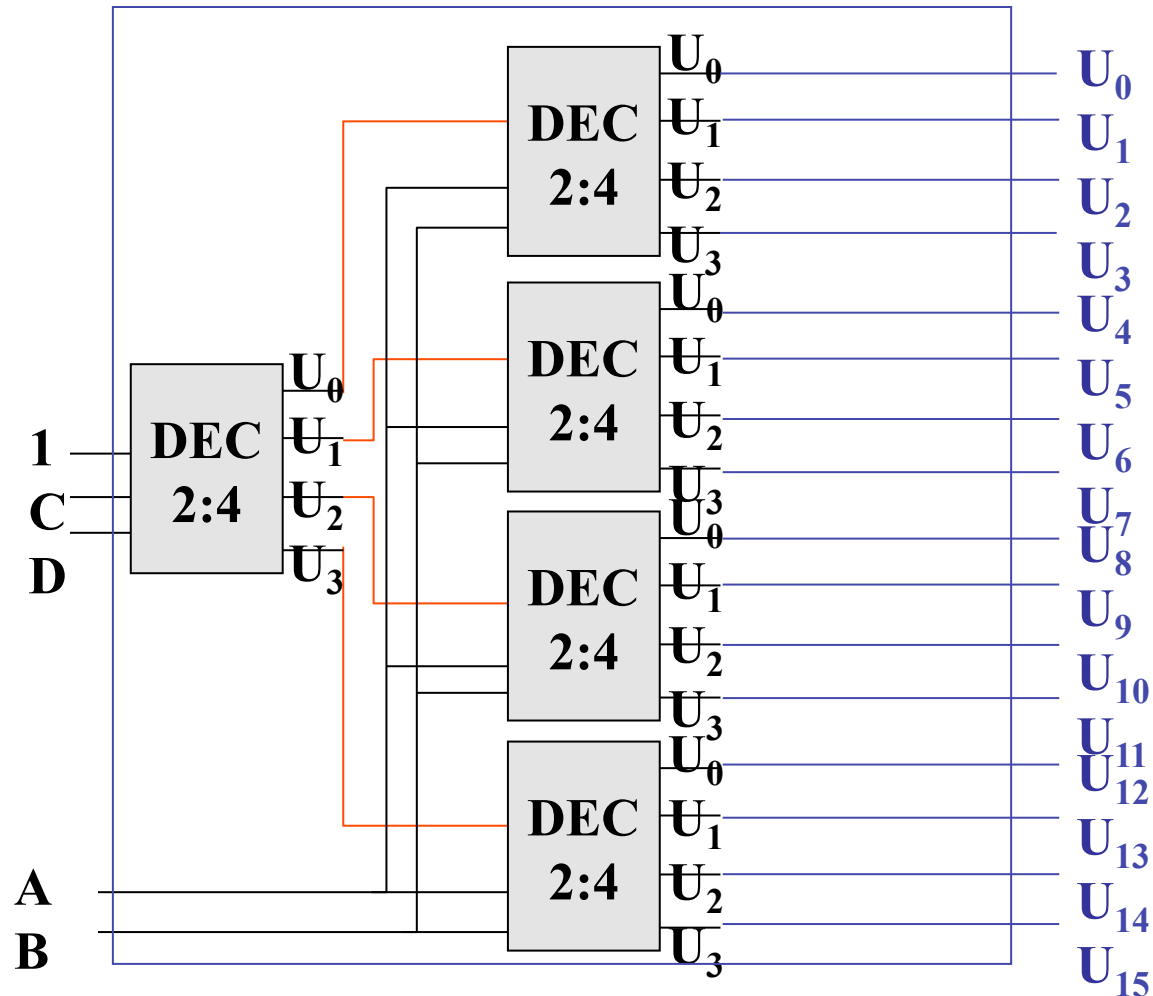


Il circuito integrato DECODER



Quando EN=1 (segnale di abilitazione), vale 1 l'uscita il cui pedice, in decimale, corrisponde al numero binario in ingresso (A bit di minor peso).

Composizione modulare di decoder: decoder 4:16 con decoder 2:4

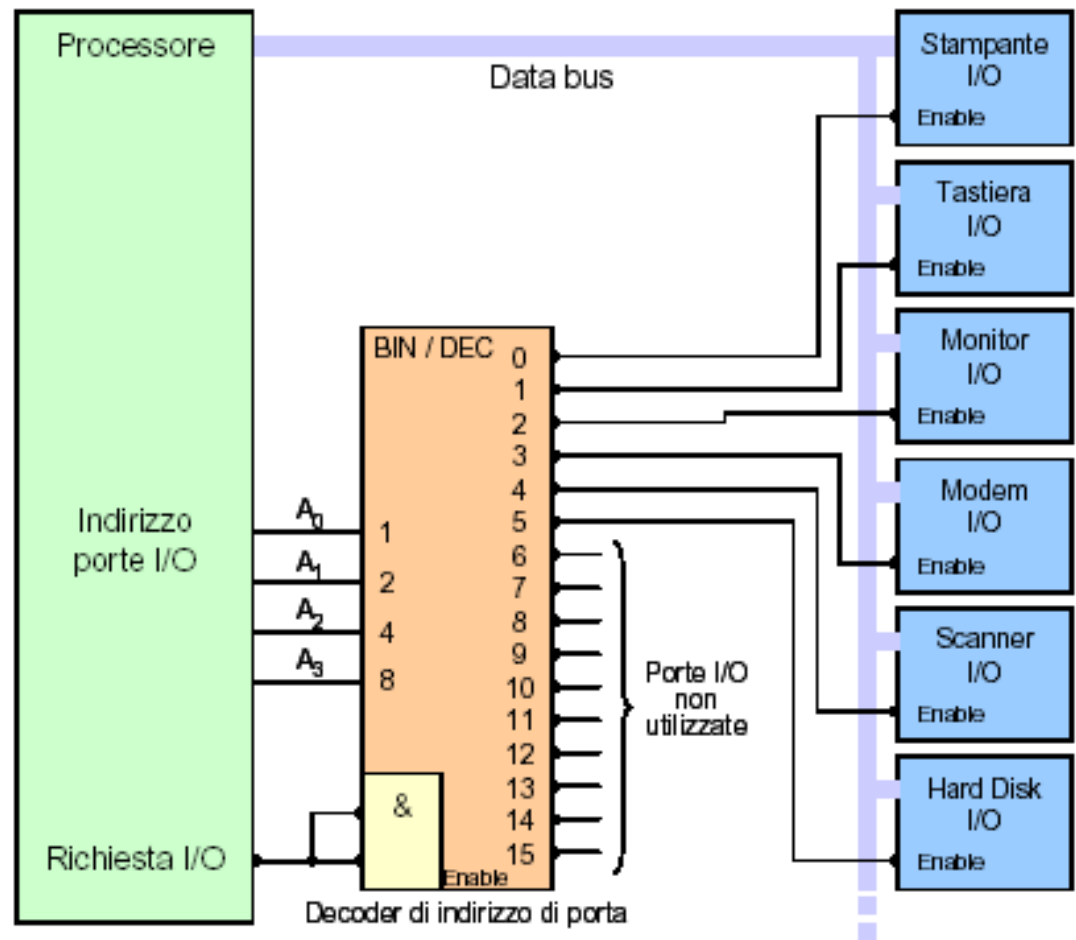


Esempio di utilizzo di Decoder (1/2)

- I decoder sono utilizzati in moltissime applicazioni.
 - Un esempio è la selezione dell'input/output da parte del processore.
 - Ogni porta di I/O possiede un indirizzo (univoco).
 - Quando il processore vuole comunicare con una determinata periferica esso produce il codice dell'indirizzo della porta di I/O al quale la periferica è connessa.
 - L'indirizzo (binario) della porta è fornito in input al decoder il quale attraverso il suo output abilita la corrispondente porta di I/O.
-

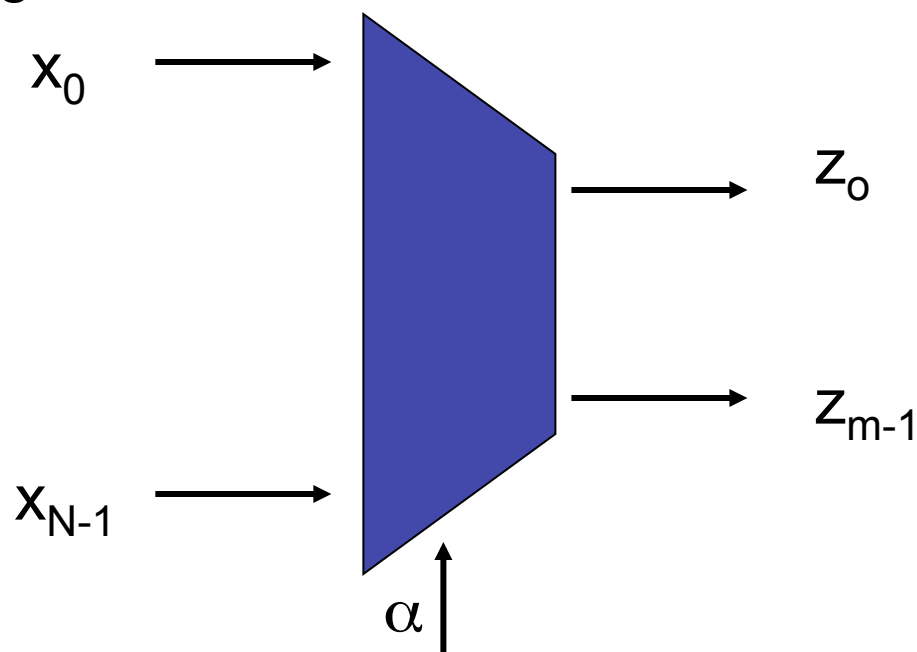
Esempio di utilizzo di Decoder (2/2)

- Come mostrato in Figura l'informazione binaria è trasferita su un **bus di dati**, che è un set di fili paralleli.
- **Esempio:** un bus di 8-bit consiste in 8 fili paralleli che trasferiscono un byte di dati alla volta.
- Il bus è connesso a tutte le porte I/O, ma ogni dato in transito passerà attraverso la porta soltanto se questa è abilitata dal decoder.

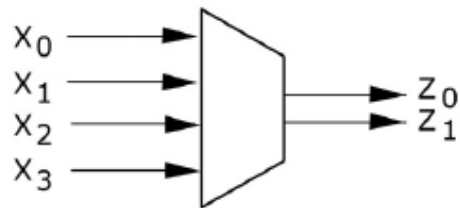


Encoder o codificatore

- Un codificatore riceve in ingresso una rappresentazione decodificata (linee x_0, \dots, x_{m-1}) e fornisce in uscita una rappresentazione con un codice a lunghezza fissa di n bit
- L'uscita è la parola codice associata a x_i
se $x_i=1$
ed $\alpha=1$ (abilitazione)
- Vincolo su ingressi:
 $x_i \cdot x_j = 0$ per $i \neq j$



Codificatore a 4 ingressi



X_3	X_2	X_1	X_0	Z_1	Z_0
0	0	0	1	0	0
0	0	1	0	0	1
0	1	0	0	1	0
1	0	0	0	1	1

Figura 3.31 - Schematizzazione di un codificatore a quattro ingressi e parte della tabella di verità delle funzioni di uscita.

Codificatore 8-4-2-1

- $Z_3 = X_8 + X_9$
- $Z_2 = X_4 + X_5 + X_6 + X_7$
- $Z_1 = X_2 + X_3 + X_6 + X_7$
- $Z_0 = X_1 + X_3 + X_5 + X_7 + X_9$

cifra	
	8-4-2-1
0	0000
1	0001
2	0010
3	0011
4	0100
5	0101
6	0110
7	0111
8	1000
9	1001

- Espressioni ottenute considerando opportunamente le configurazioni di ingresso *dont'care*
-

Arbitro di priorità

- Un codificatore può essere preceduto da una “rete a priorità” che, in caso di più ingressi contemporaneamente alti, filtra quello con priorità assegnata maggiore

- Rete a priorità

- n ingressi X_i
- n uscite corrispondenti F_i , che rappresentano gli ingressi del codificatore
- fra gli ingressi è definita una priorità, ad esempio:
 - per fissare le idee
« X_i è prioritario su X_j se $i < j$ »
- **L'uscita Y_i è alta se e solo se X_i è alto e tutti gli altri ingressi prioritari su X_i sono bassi.**

$$F_1 = X_1$$

$$F_2 = X_2 \overline{X_1}$$

.....

$$F_n = X_n \overline{X_{n-1}} \cdots \overline{X_1}$$

Arbitro di priorità a 4 ingressi

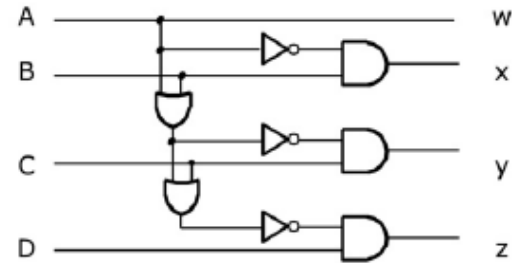
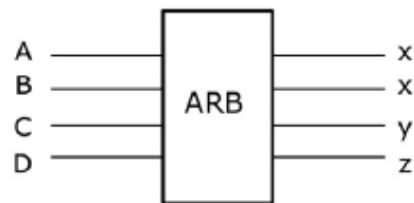


Figura 3.34 - Schematizzazione di un arbitro di priorità a quattro ingressi e corrispondente rete "a scala".

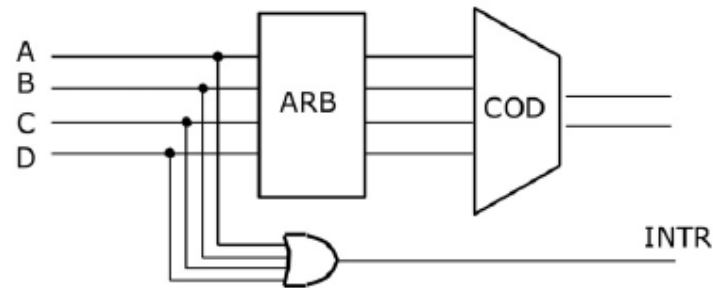
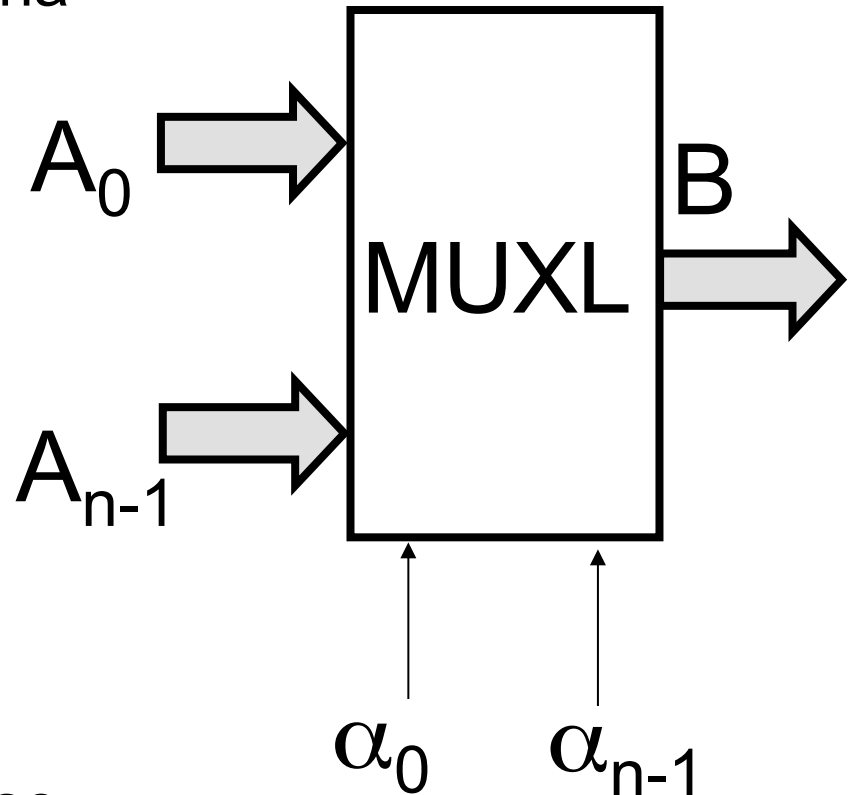


Figura 3.35 - Costruzione di un codificatore di priorità a partire da un arbitro di priorità e da un codificatore. La linea INTR indica che almeno una delle linee tra A, B, C e D è 1. Quando INTR è 0, nessuna linea di ingresso risulta asserita e l'uscita del codificatore è senza significato.

Multiplexer lineare

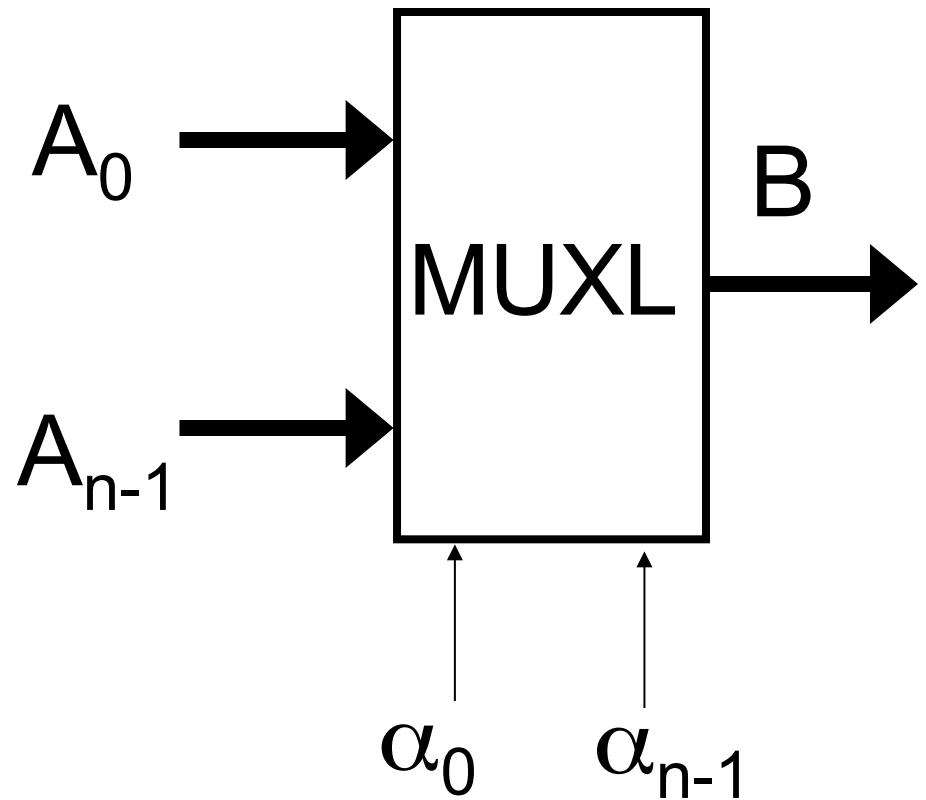
- Un *Multiplexer lineare* (ML) è una macchina con:
 - n ingressi-dati (A_0, \dots, A_{n-1})
 - n segnali binari di selezione ($\alpha_0, \dots, \alpha_{n-1}$),
dei quali al più uno è attivo
 - una uscita-dati B, che assume
 - valore A_i se è attivo α_i
 - neutro se nessuna delle selezioni è attiva
- Utilizzato quando più linee devono essere convogliate verso un' unica linea di uscita (bus)



Multiplexer binario

- Se i dati A_i e B sono semplici bit si parla di multiplexer *binario*

$$B = \sum_{i=0}^{n-1} \alpha_i \cdot A_i$$



Multiplexer binario - realizzazioni

- $B = A_0 \alpha_0 + A_1 \alpha_1 + \dots + A_{n-1} \alpha_{n-1}$
 $n=4$

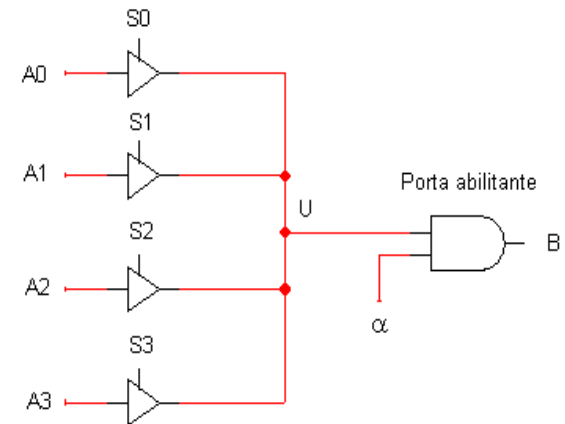
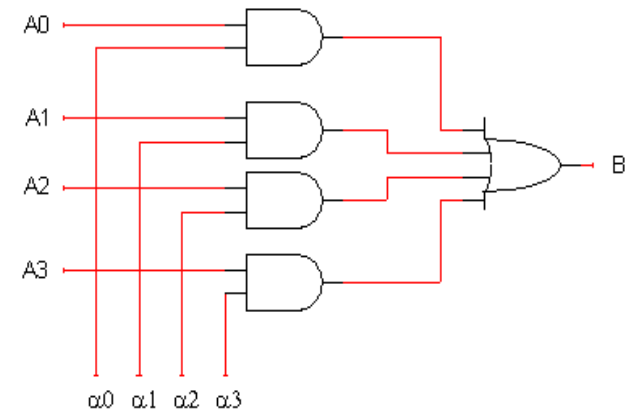
- **Realizzazione I**

- Con porte AND e OR

- **Realizzazione II**

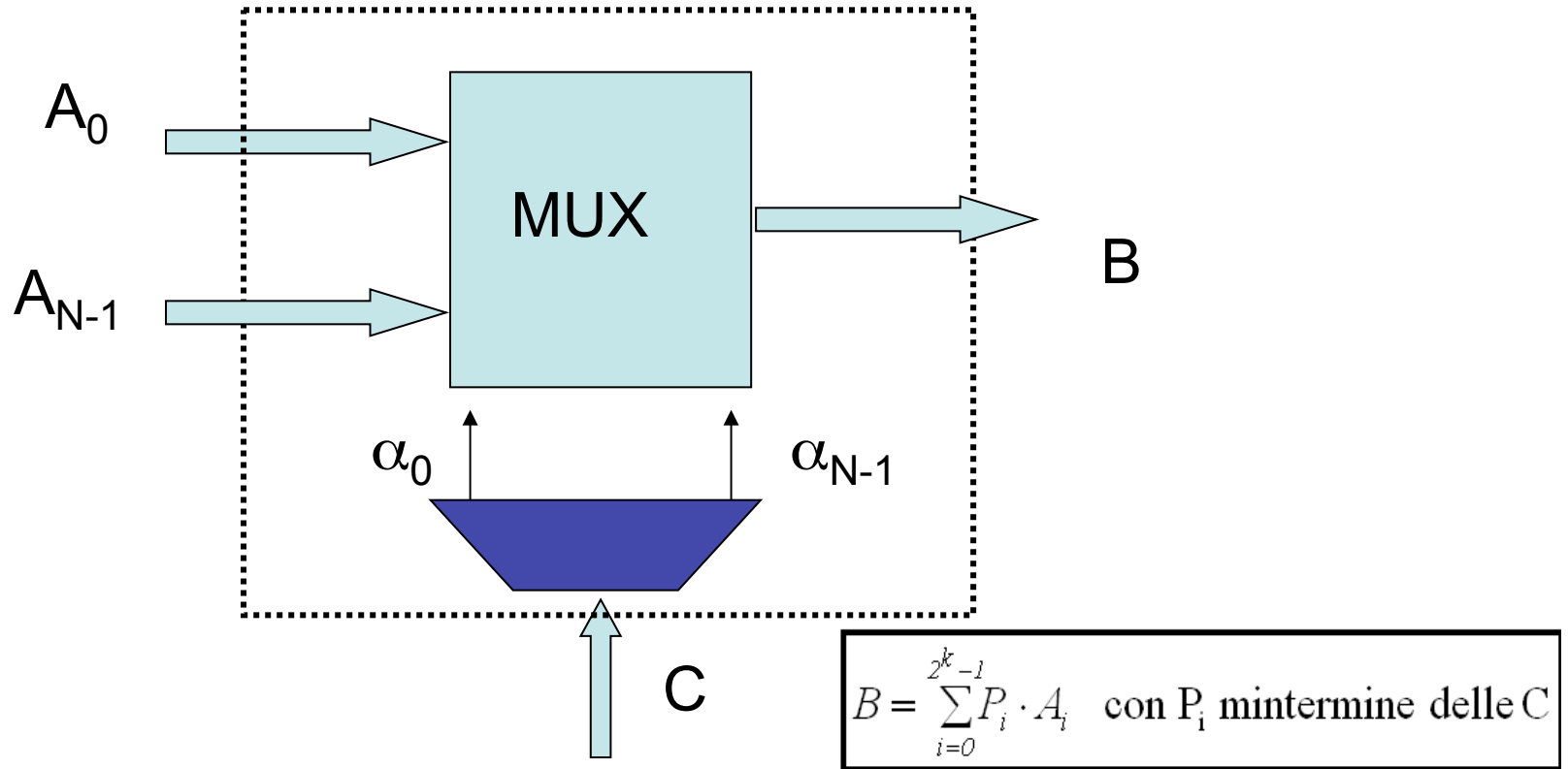
- Con porte 3-state
 - $S=1$, restituisce il valore di A
 - $S=0$, restituisce un'alta

A	S	Uscita	impedenza (apre il circuito)
0	0	z	
0	1	0	
1	0	z	
1	1	1	



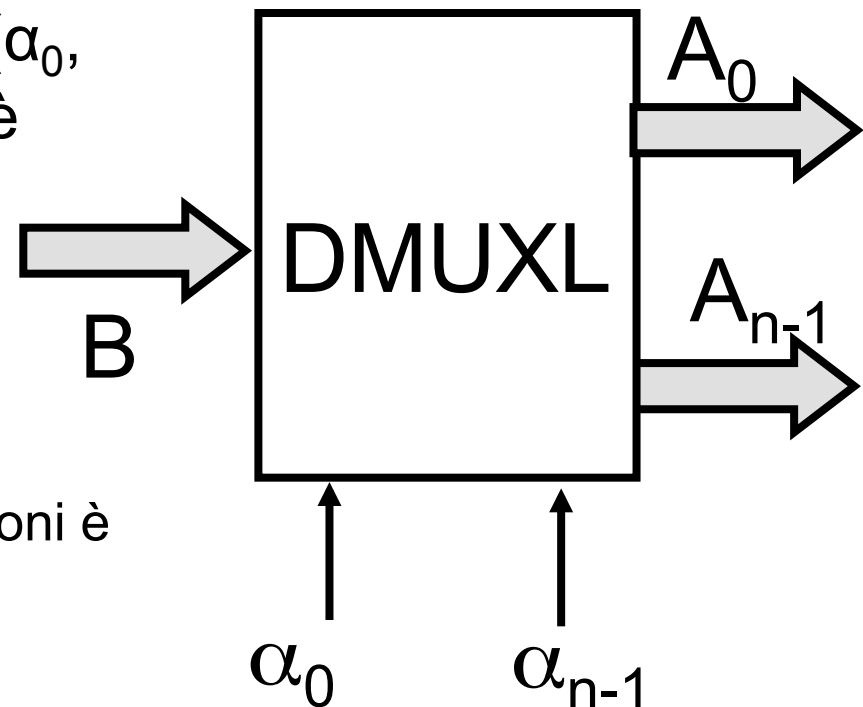
Multiplexer (indirizzabile)

- Multiplexer Lineare i cui segnali di abilitazione sono collegati con le uscite di un decodificatore



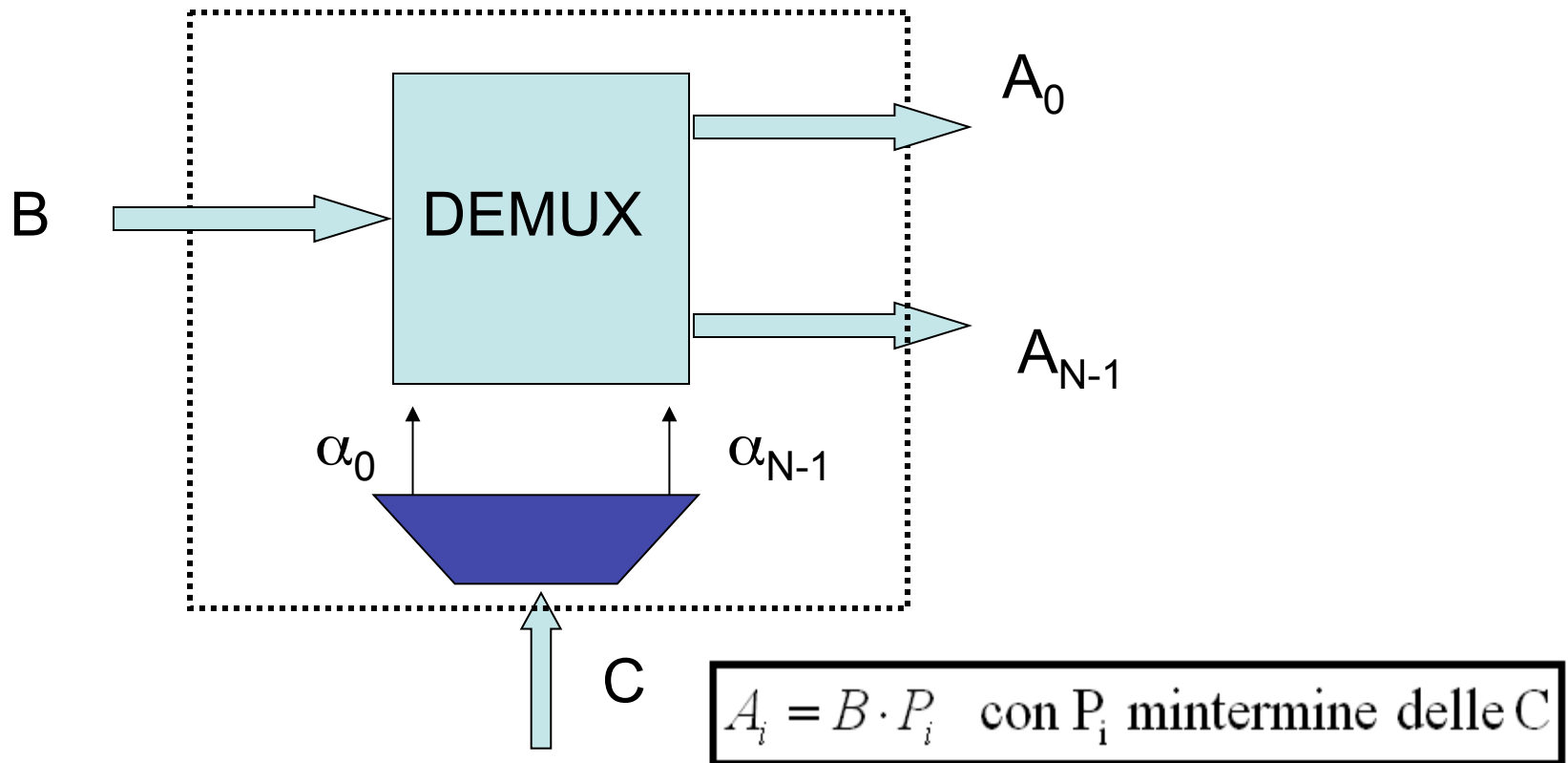
Demultiplexer lineare

- Un *Demultiplexer Lineare* è una macchina con:
 - 1 ingresso-dati B
 - n segnali binari di selezione ($\alpha_0, \dots, \alpha_{n-1}$), dei quali al più uno è attivo
 - n uscite-dati (A_0, \dots, A_{n-1}), con
 - $A_i=B$ se è attivo α_i
 - neutro se nessuna delle selezioni è attiva



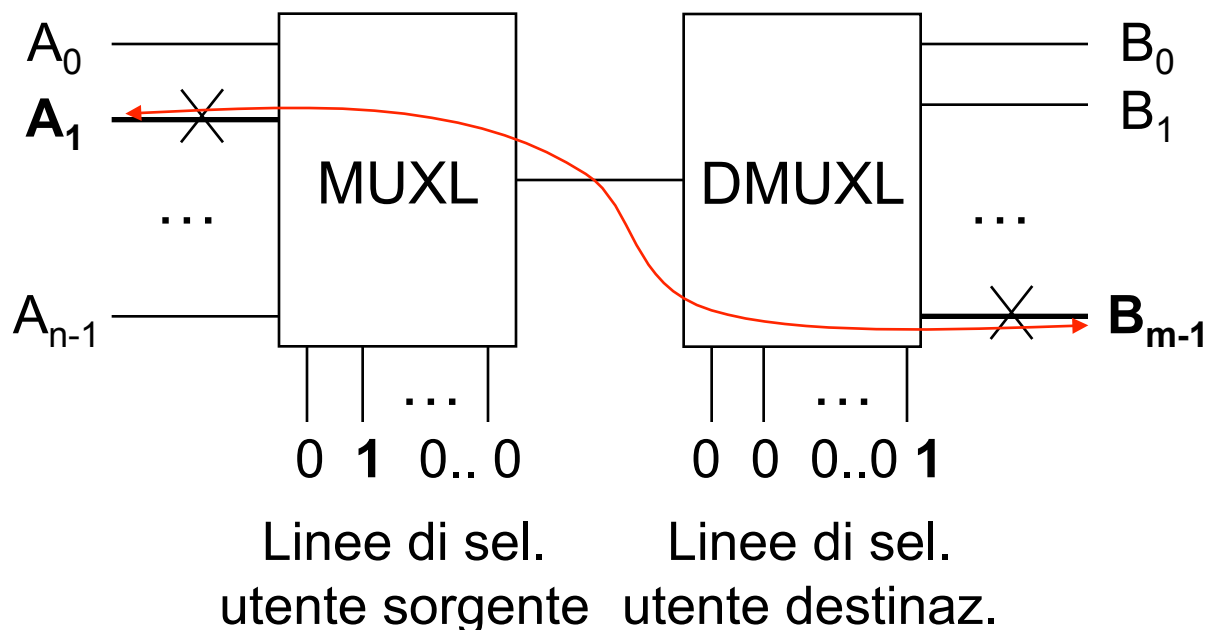
Demultiplexer (indirizzabile)

- Demultiplexer Lineare i cui segnali di abilitazione sono collegati con le uscite di un decodificatore



Muxl/Dmuxl: un esempio

- Supponiamo di avere un “centralino telefonico” in cui n utenti sorgente vogliono parlare con m utenti destinazione
 - **vincolo:** l’utente di destinazione abilitato deve sentire solo l’utente sorgente abilitato



L’utente A_1 è abilitato a parlare con l’utente B_{m-1}