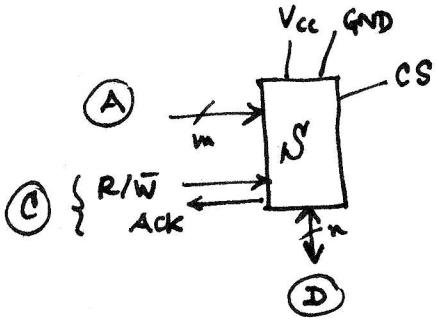
29 sett. 2017

2ª lezione: Banchi di memoria, funzioni booleane, architettura vs organizzazione, decoder, realizzazione circuitale vs schema logico



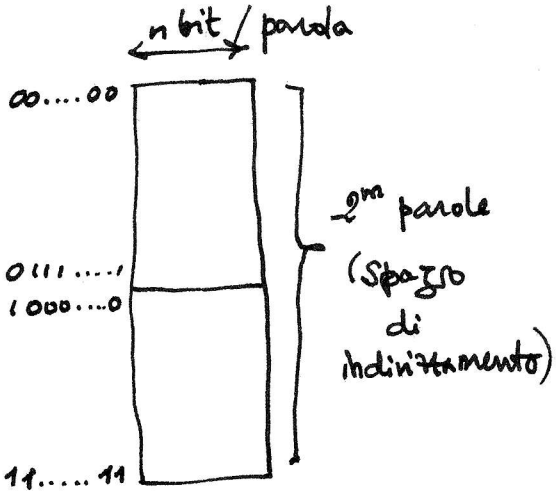

chip di memoria

Read/Write = segnali mutuamente esclusivi ⇒ codificati in uno solo

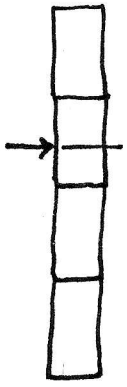
Cosa vuol dire "asserire" un segnale

Struttura logica della memoria: (nastro)

[cfr. lez. 25/9/17]

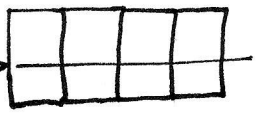



ESPANSIONI DI MEMORIA (con chip tutti uguali)



di indirizzo

$S' = 2^{m+k} \times n$



dati

$S' = 2^m \times (n*k)$

Tutti CS asseriti

per raddoppio: k=1, h=2

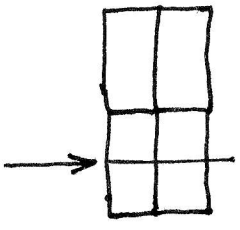
4X : k=2, h=4

di indirizzo & dati

$S' = 2^{m+k} \times (n \cdot h)$

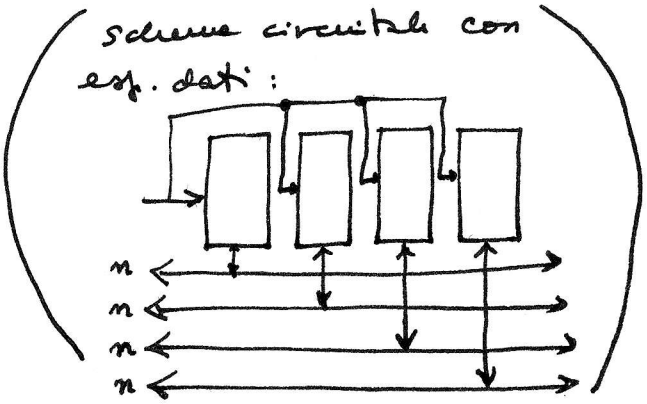
(vedi pag. seg.)

Solo 1 CS asserito per volta

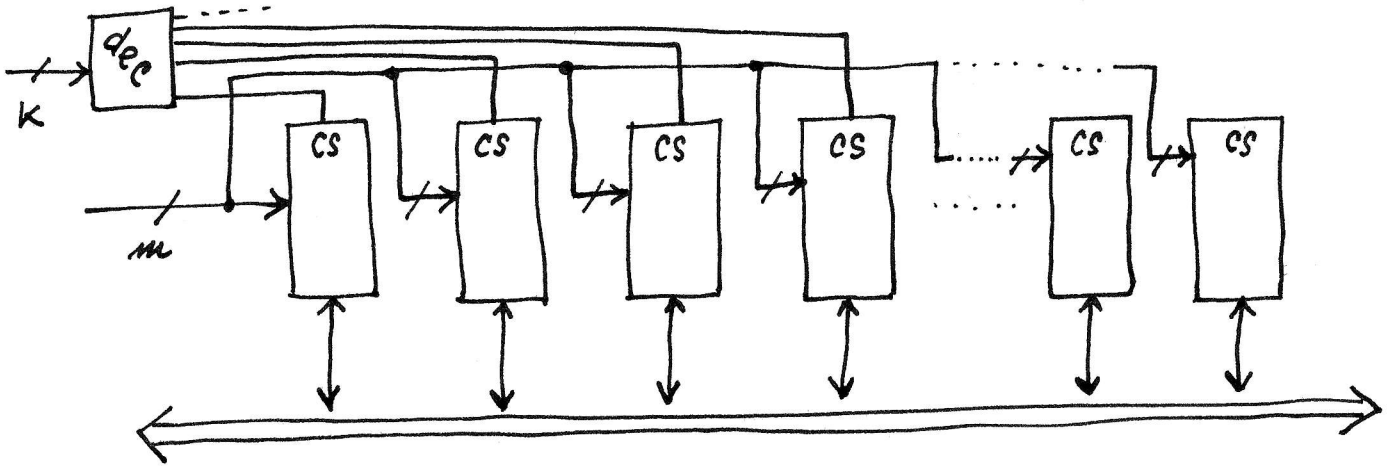


Due CS asseriti, due no

schema circuitale con esp. dati:

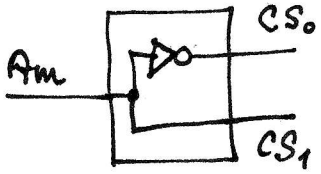


## Schema generale $2^{m+k} \times n$




$k = 1$

| Am | $CS_0$ | $CS_1$ |
|---|---|---|
| 0 | 1 | 0 |
| 1 | 0 | 1 |

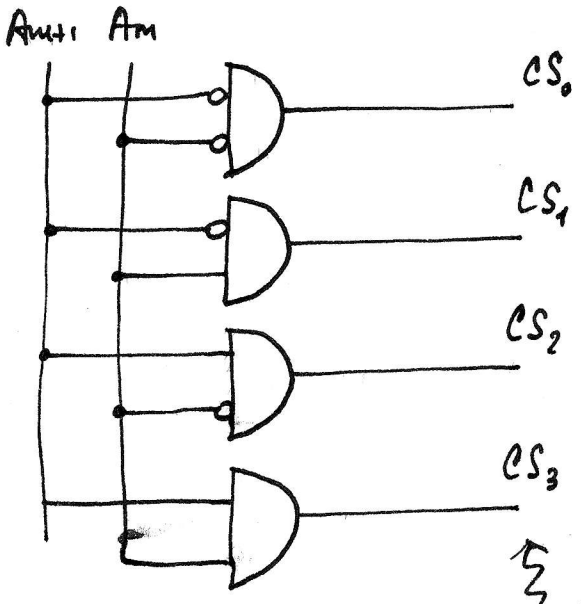


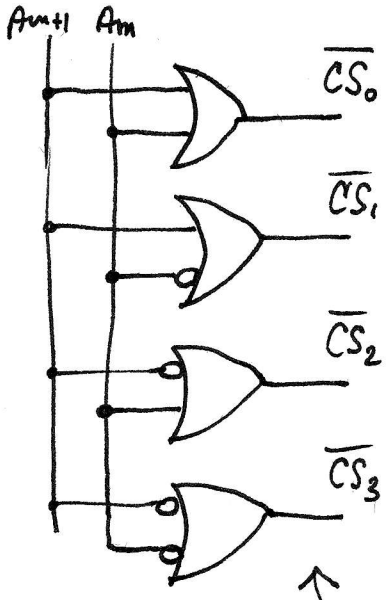

(poi ho fatto l'enumerazione di funzioni booleane, k=1 e k=2, con tabelle e formula generale $2^{2^k}$ ($\neq 4^{k!}$))

$k = 2$

| | | logica positiva | | | | logica negativa | | | |
|---|---|---|---|---|---|---|---|---|---|
| Am+1 | Am | $CS_0$ | $CS_1$ | $CS_2$ | $CS_3$ | $\overline{CS_0}$ | $\overline{CS_1}$ | $\overline{CS_2}$ | $\overline{CS_3}$ |
| 0 | 0 | 1 | 0 | 0 | 0 | 0 | 1 | 1 | 1 |
| 0 | 1 | 0 | 1 | 0 | 0 | 1 | 0 | 1 | 1 |
| 1 | 0 | 0 | 0 | 1 | 0 | 1 | 1 | 0 | 1 |
| 1 | 1 | 0 | 0 | 0 | 1 | 1 | 1 | 1 | 0 |

Il DECODER è un componente standard che "riconosce" codici binari in ingresso




solo uno asserito

Notiamo che

$\overline{CS_0}$ = OR  ($CS_0$ = NOR)

$CS_3$ = AND  ($\overline{CS_3}$ = NAND)

con, ad es.,

$$NOR = CS_0 = \overline{A_{m+1} + A_m} = \overline{A_{m+1}} \cdot \overline{A_m} \neq \overline{A_{m+1} \cdot A_m} = NAND$$

ossia le funzioni si possono esprimere sia in forma di prodotti che di somme.

## Architettura vs Organizzazione  (nel caso $2^{m+k} \times n$)

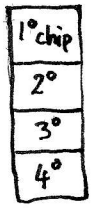
Il processore vede la parola di indirizzo tutta intera:

| $A_{m+1}$ $A_m$ $A_{m-1}$ ..... $A_1$ $A_0$ |
|---|

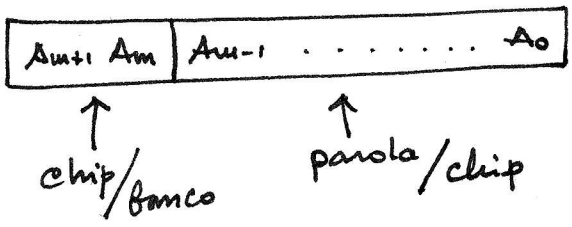
parola/banco

L'organizzazione del banco induce però una suddivisione logica in campi dell'indirizzo.

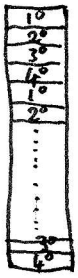
Due esempi di organizzazione:




1]

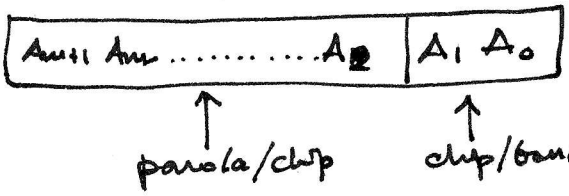



cfr. i casi visti prima: qui i chip contengono ciascuno un quarto della memoria totale, ed esattamente il 1°, 2°, 3° e 4° quarto



2]

(cfr. MEMORY INTERLEAVING, pag. 122 Bucci[4])




qui invece il 1° chip contiene gli indirizzi $A \equiv 0 \pmod 4$
2° " $A \equiv 1 \pmod 4$
3° " $A \equiv 2 \pmod 4$
4° " $A \equiv 3 \pmod 4$