Le memorie

Generalità

E

applicazioni

Caratteristiche generali

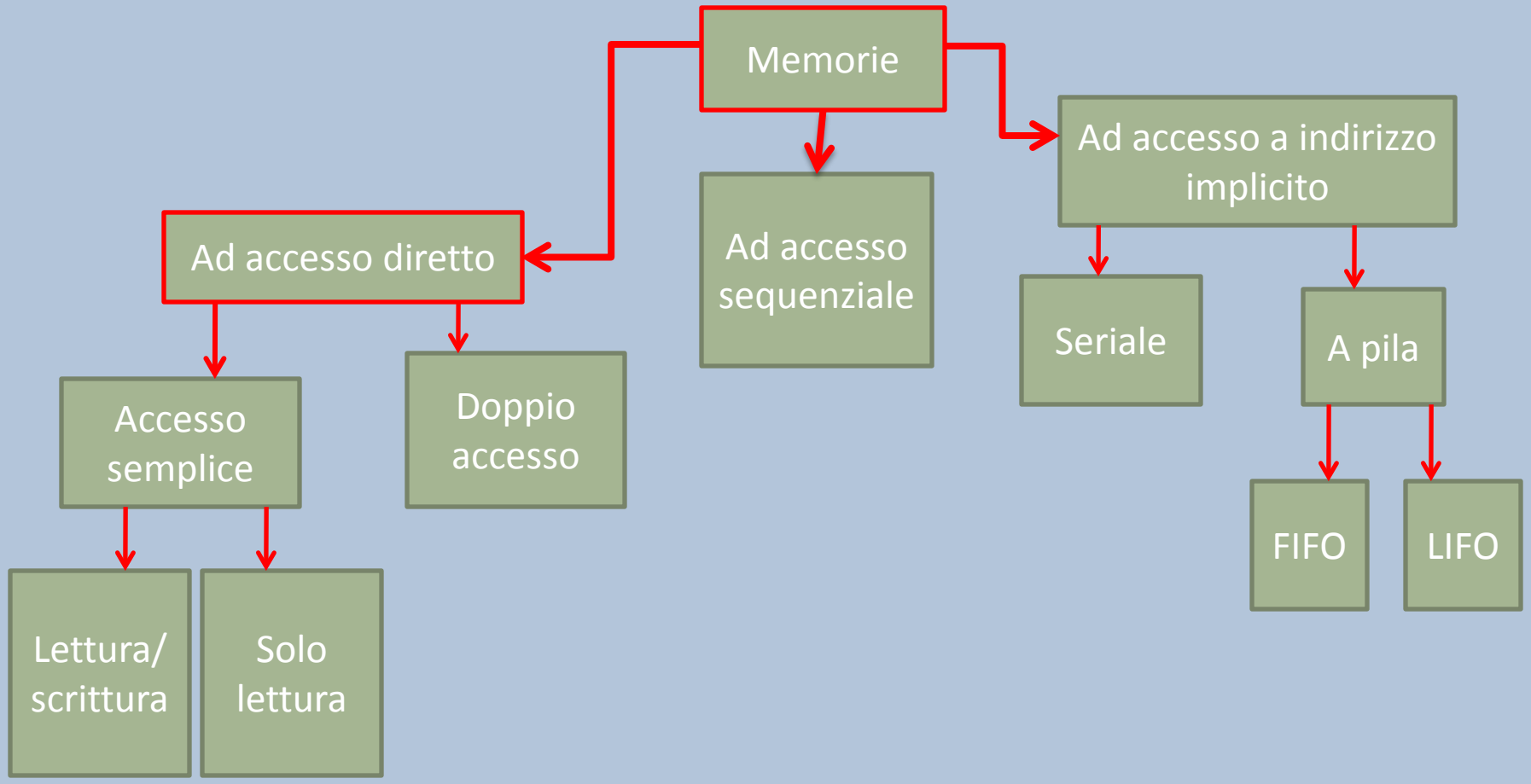
- Tempo di risposta
- Capacità
- Alimentazione
- Dissipazione di potenza
- Numero di pin
- Costo per bit

Modalità di accesso

Per poter scrivere un dato o poter leggere esistono diverse modalità di accesso:

- Casuale o diretto- se l'architettura del dispositivo è tale da fornire un indirizzo di accesso unico sia di lettura che di scrittura
- Seriale- se è possibile accedere al dato che si trova in una determinata posizione; è allora necessario portare il dato nella posizione desiderata tramite uno scorrimento
- A indirizzo implicito- si può accedere solo al dato presente in una determinata posizione senza poter spostare (es. lo stack pointer). In questo caso si distinguono due modalità di accesso: LIFO e FIFO, cioè Last Input First Output e, First Input First Output

Modalità di accesso



Classifica generale

Le memorie si raggruppano in quattro grandi categorie:

1. Di sola lettura:è possibile effettuare solo la lettura ma non la scrittura
2. Di lettura-scrittura: si possono memorizzare dati e leggerli anche
3. Volatili: perdono il loro contenuto una volta tolta l'alimentazione
4. Non volatili: mantengono la memoria anche quando viene tolta l'alimentazione

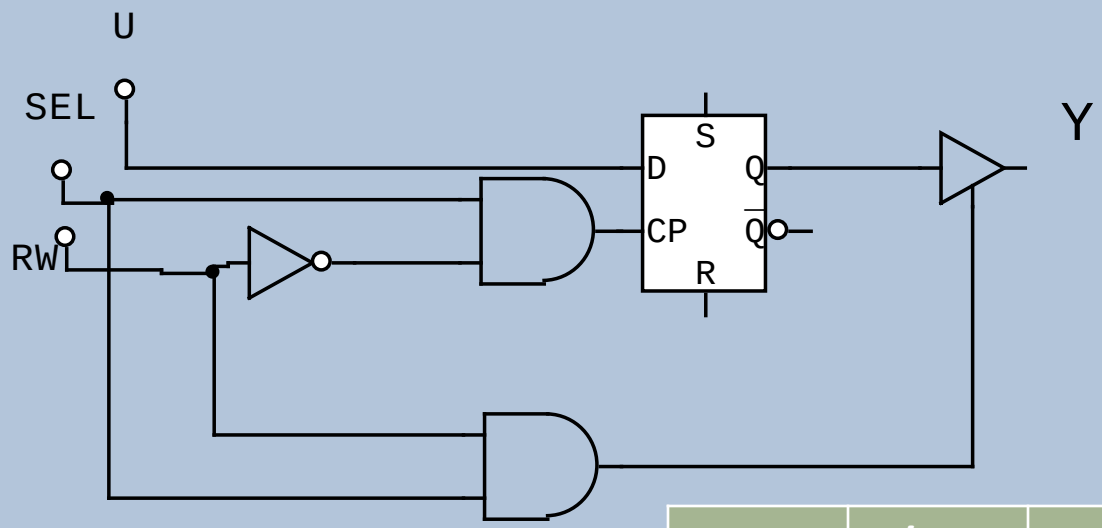
Memorie non volatili

- ROM: Read Only Memory; vengono programmate per firmware di strumenti elettronici programmabili o per programmi non soggetti a modifiche o per decodifiche veloci; sono di basso costo.
- PROM: Programmable ROM; sono programmabili dall'utente ma solo una volta
- EPROM: Erasable PROM. Sono programmabili ma anche cancellabili dall'utente tramite i raggi ultravioletti. Sono per prototipi di frequente cancellazione
- EEPROM: Electrical EPROM si possono cancellare elettricamente
- Flash: sono simili alle EEPROM ma con tempi di cancellazione e scrittura molto più veloci. Sono utilizzate per i modem o per la bios del PC ma anche le memorie di pen drive collegabili alla porta USB. Sono tipiche di alcuni PIC aventi proprio sigla nfxxx

Memorie volatili

- RAM: Random Access Memory; si possono leggere e scrivere in continuazione ma perdono il contenuto nel momento in cui viene tolta l'alimentazione. I tempi di accesso sono molto più brevi delle memorie non volatili. Le memorie RAM si sono evolute in:
    - SRAM
    - DRAM
    - SDRAM
    - DDR SDRAM
    - MRAM

Schema di una cella di memoria lettura/scrittura

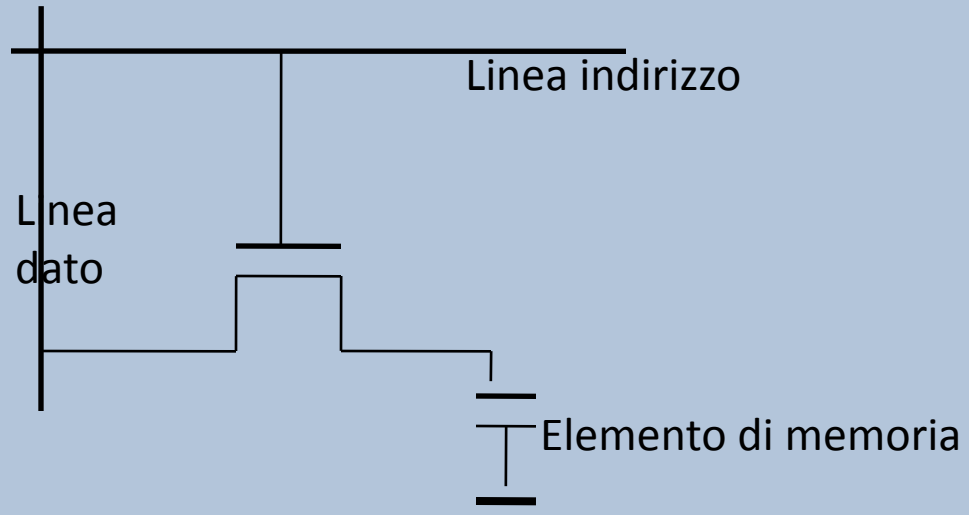


SEL	R/W	U	Qn	Y
0	X	X	Qn-1	HZ
1	1	X	Qn-1	Qn
1	0	1	1	HZ
1	0	0	0	HZ

HZ = alta impedenza

Schema di una cella di memoria DRAM

- L'elemento di memoria è il condensatore mentre il MOSFET isola l'elemento di memoria connettendolo e disconnettendolo. La semplicità della cella permette dispositivi di notevole capacità. La memoria è detta DRAM perché il condensatore si scarica facilmente e deve essere sempre ricaricato. Questa operazione avviene in fase di lettura



Questi tipi di memoria sono oggi i più diffusi nei personal computer e hanno sostituito le memorie SRAM che hanno una più piccola scala di integrazione

Accesso casuale

Le memorie RAM (SRAM, DRAM, VRAM), ROM, EPROM, EEPROM, flash

Sono ad accesso casuale

Si può accedere al dato tramite

Un unico indirizzo definito

Organizzazione di memoria

- L’ **organizzazione** di una memoria a semiconduttore è il modo in cui essa è internamente suddivisa in parole.
- ES: consideriamo una memoria composta da 64 parole da 4 bit ciascuna. L'organizzazione di questa memoria sarà dunque 64 x 4 = 256 bit.
- Ad ogni parola corrisponde un diverso indirizzo compreso fra 0 (indirizzo della prima parola) e 63 (indirizzo dell'ultima parola in memoria). Siccome $2^6 = 64$, dovrebbe essere evidente che occorrono 6 bit per specificare l'indirizzo di ogni singola parola nella nostra memoria.
- In generale il numero di piedini di indirizzo n è tale per cui 2^n è il numero di parole in memoria. Per esempio una memoria con 8 pin di indirizzo conterrà $2^8 = 256$ parole. Siccome si utilizza il sistema binario, il numero di parole in una memoria è sempre una potenza esatta di due.

Capacità di memoria

Riassumendo

La capacità di memoria è data da:

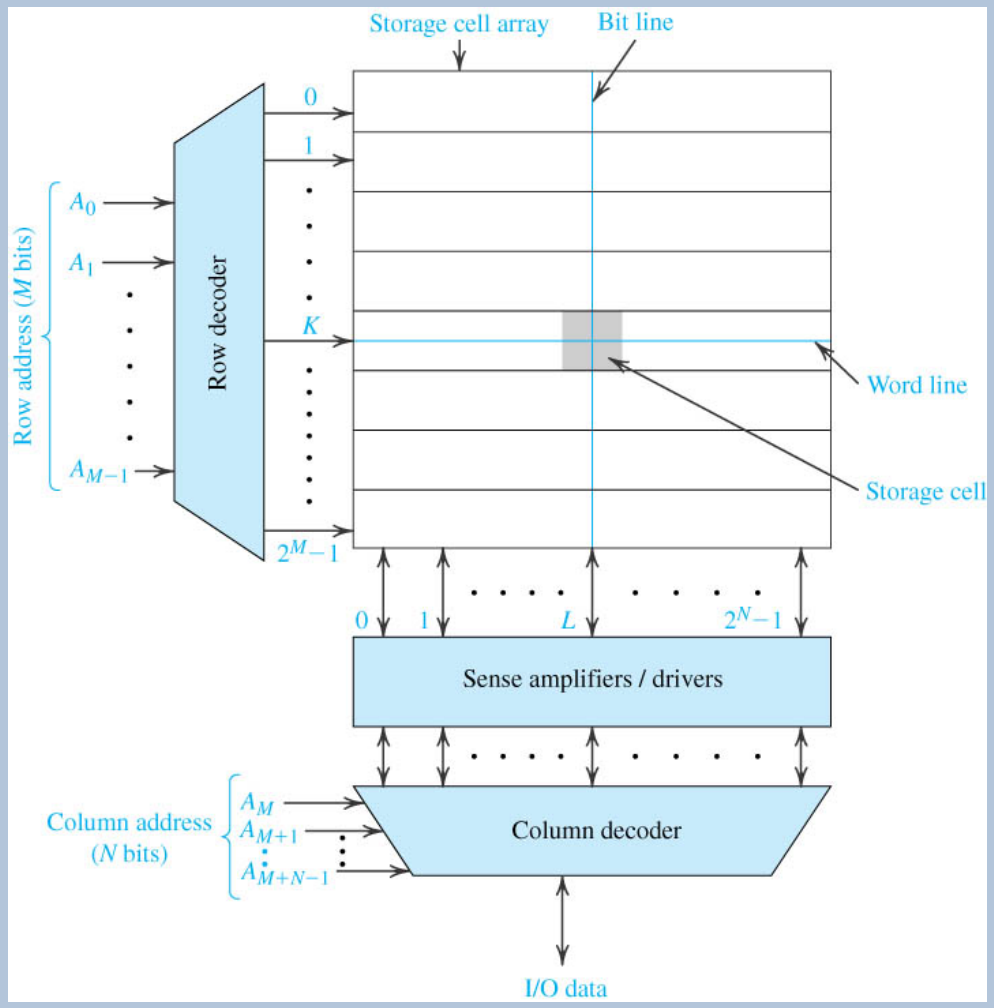
M locazioni di memoria

X

N numero di bit della parola

Memoria a matrice o a indirizzo diretto

Una memoria di
2^(M+N)x1
bit organizzata in una matrice di 2^M righe e 2^N colonne



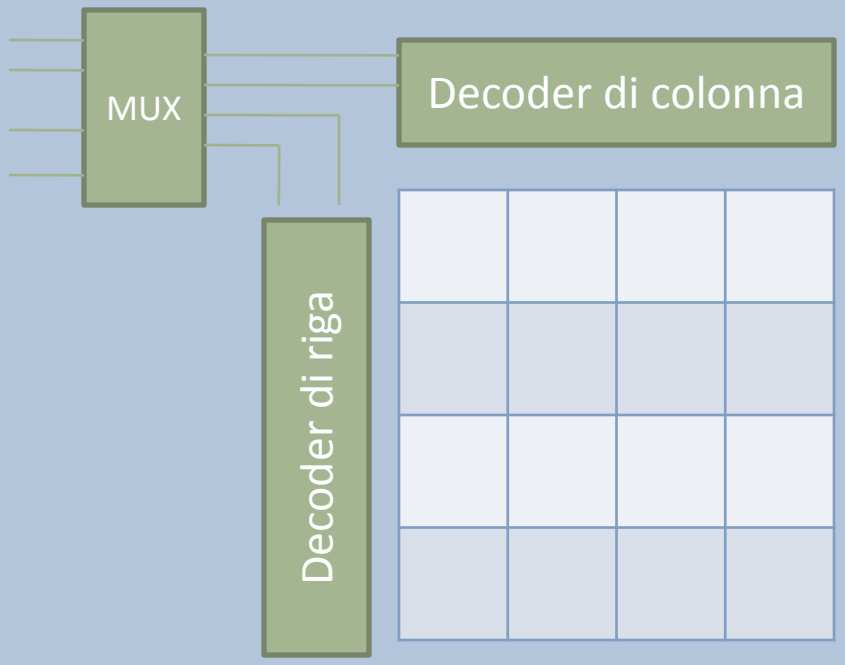
Architettura di un blocco di memoria

- Ogni blocco di memoria è formato da cellette elementari disposte a matrice
- Per ogni celletta esiste un indirizzo che è dato dalla coppia ordinata di due numeri binari
- Il primo numero della coppia individua la colonna il secondo la riga
- Con questo sistema si diminuiscono le linee di indirizzo
- es: sia una memoria con capacità 256 K. Dovremmo scrivere 256x1024 indirizzi differenti. Se trasformiamo gli indirizzi in binario, saranno necessari solo 18 pin per individuare le cellette. Infatti 256 K = 2^8 x2^{10}

Potrebbero così essere necessari 8 pin di riga e 10 di colonna

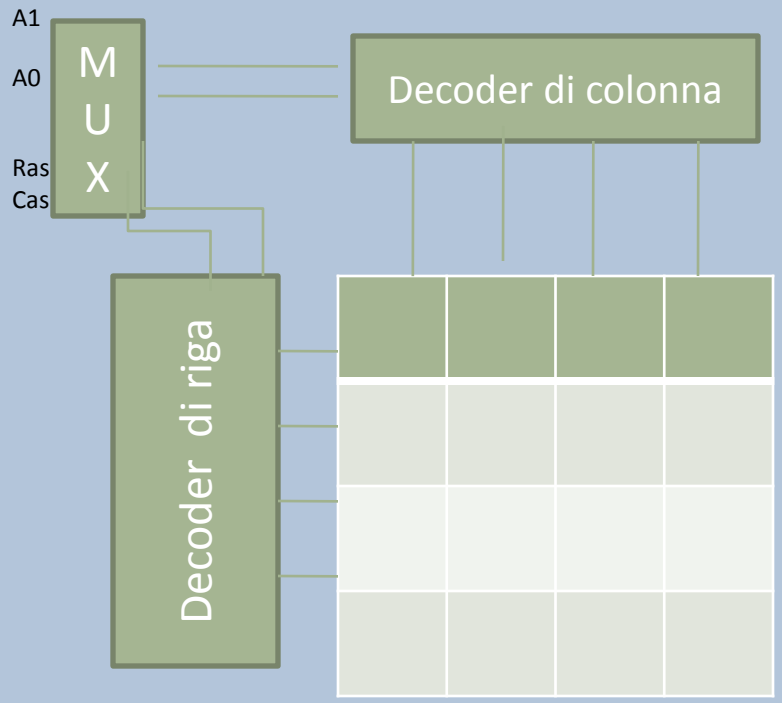
Esempio

Nel seguente esempio è riportata una memoria di capacità 4x4=16 bit
Per poter indirizzare le singole cellette saranno necessari solo 4 linee di indirizzo, 2 per le righe e 2 per le colonne



Esempio

- Memoria con indirizzi di riga e di colonna



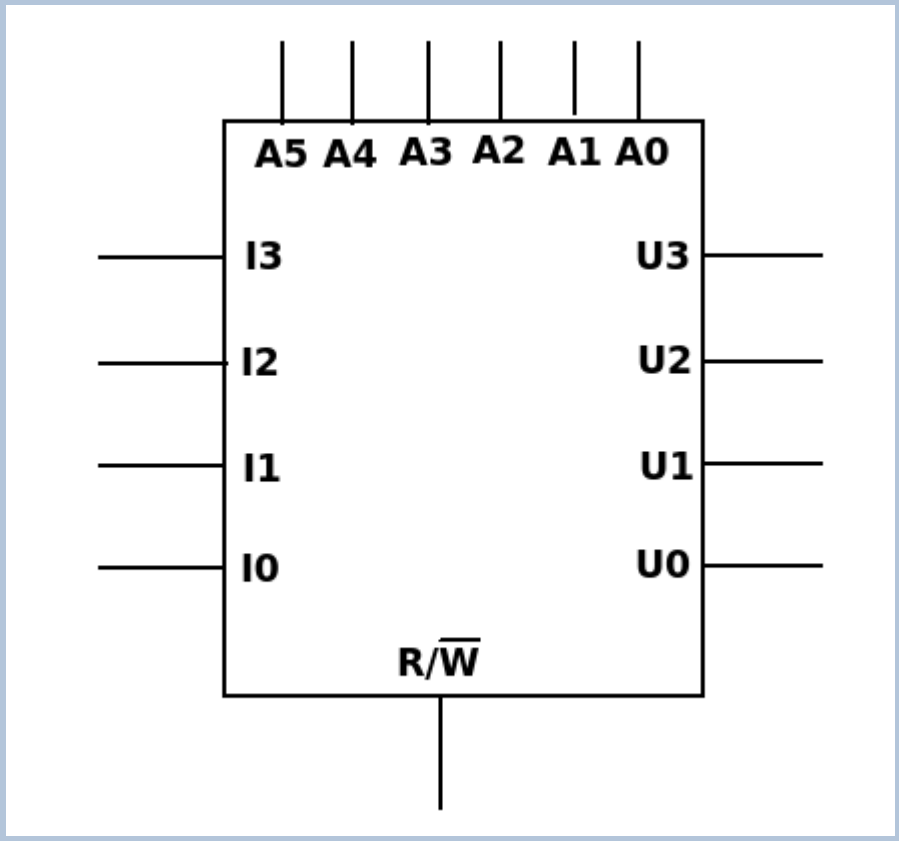
RAS= Row Access Strobe
CAS= Column Access Strobe

Il dispositivo dovrebbe disporre di 8 piedini; con un solo indirizzo di riga o di colonna si possono disporre di soli 2 piedini;si seleziona così prima l'indirizzo di riga e poi quello di colonna

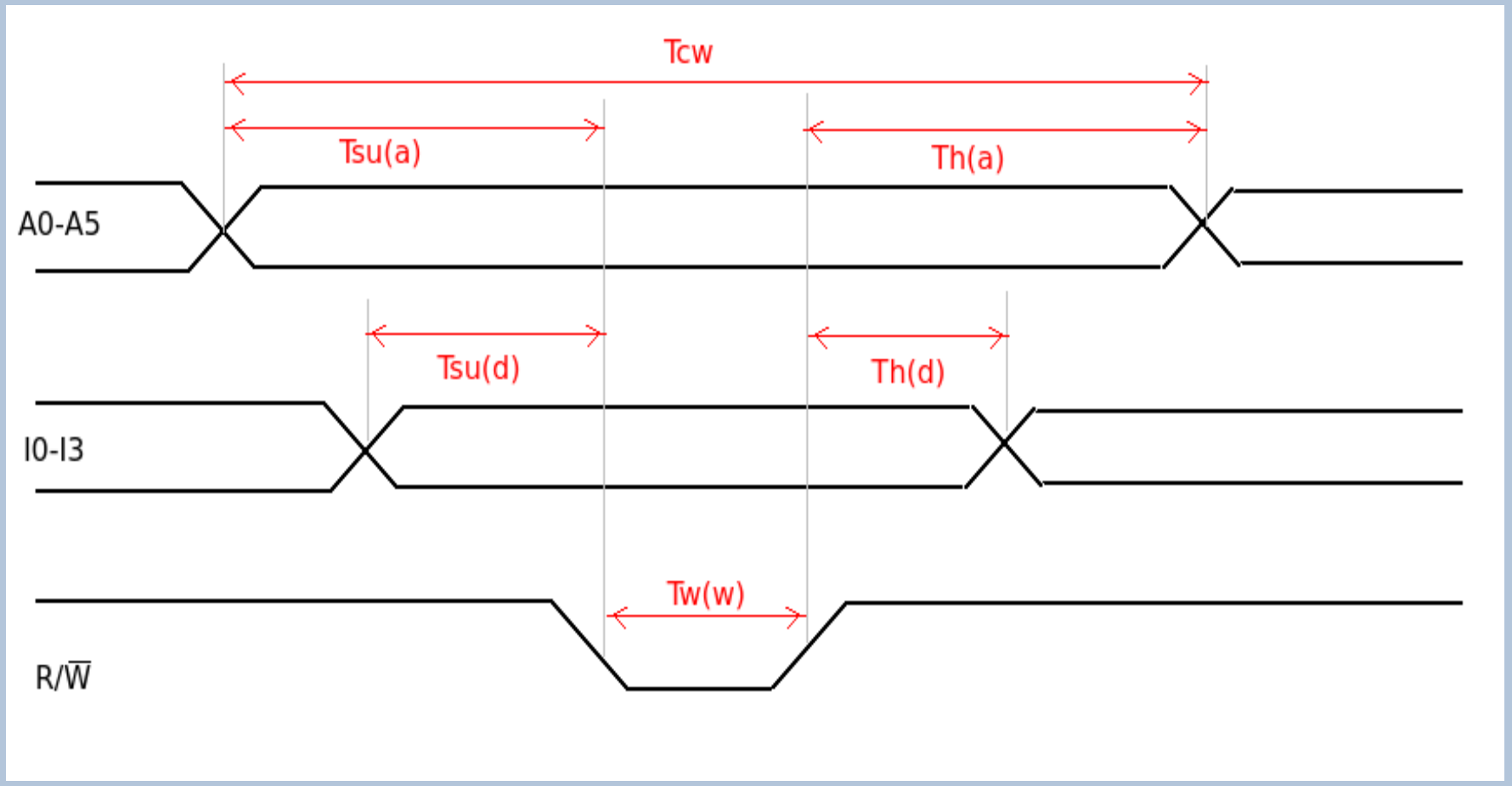
Un esempio di piedinatura

La piedinatura per il nostro integrato

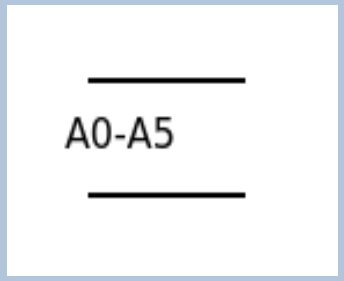
I0, I1, I2, I3 sono i pin di dato di ingresso e
U0, U1, U2, U3 sono i pin di dato in uscita.
A0, A1, A2, A3, A4 e A5 sono i pin di indirizzo.



Ciclo di scrittura in memoria

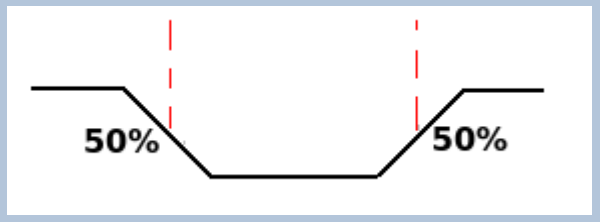


Convenzioni grafiche



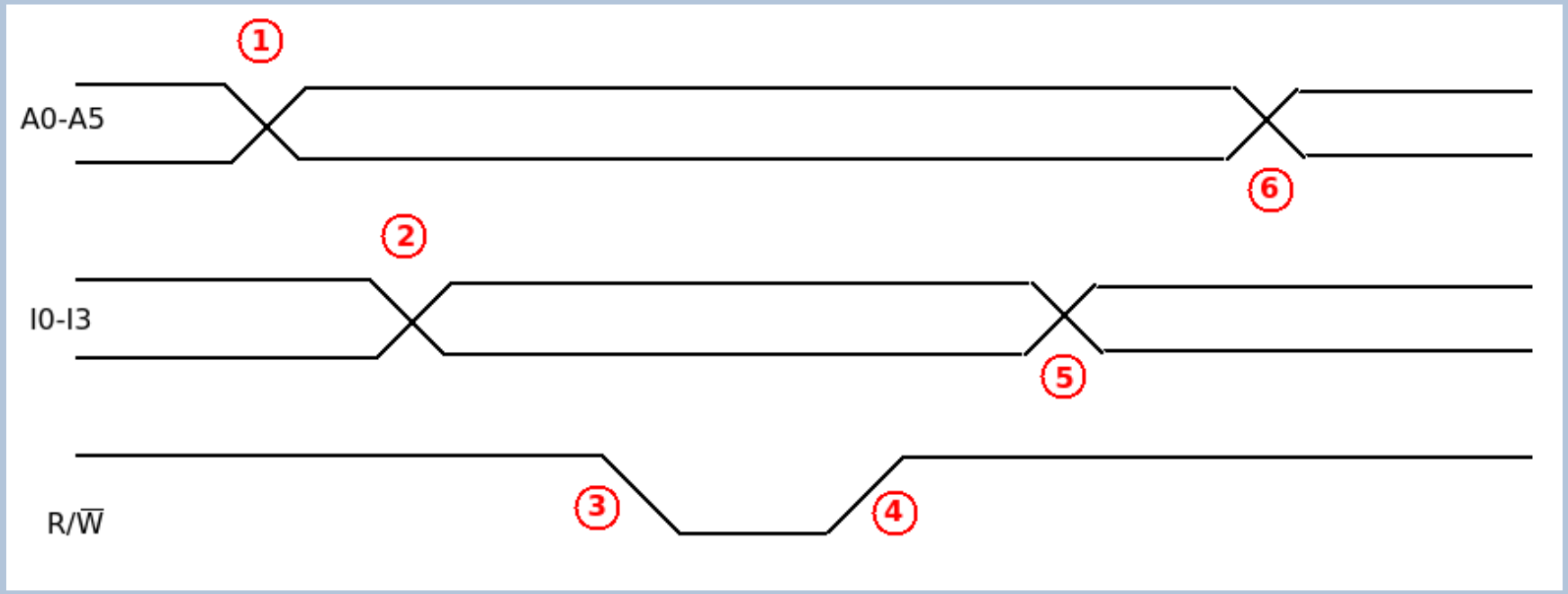
i segnali non sono rappresentati singolarmente, ma a gruppi (per esempio A0-A5 indica le sei linee di indirizzo);

i valori binari effettivamente presenti sui gruppi di segnali (es. sulle linee di indirizzo) non sono importanti (dipendono da quale indirizzo viene fornito) e non vengono rappresentati; ciò che interessa invece è l'istante in cui tali valori commutano, istante rappresentato sul diagramma con un incrocio



I fronti di commutazione sono disegnati inclinati e non verticali, per indicare che la commutazione di un segnale non è istantanea; i tempi sono sempre misurati a partire dal 50% della commutazione:

Scrittura in dettaglio



La sequenza in dettaglio è la seguente:

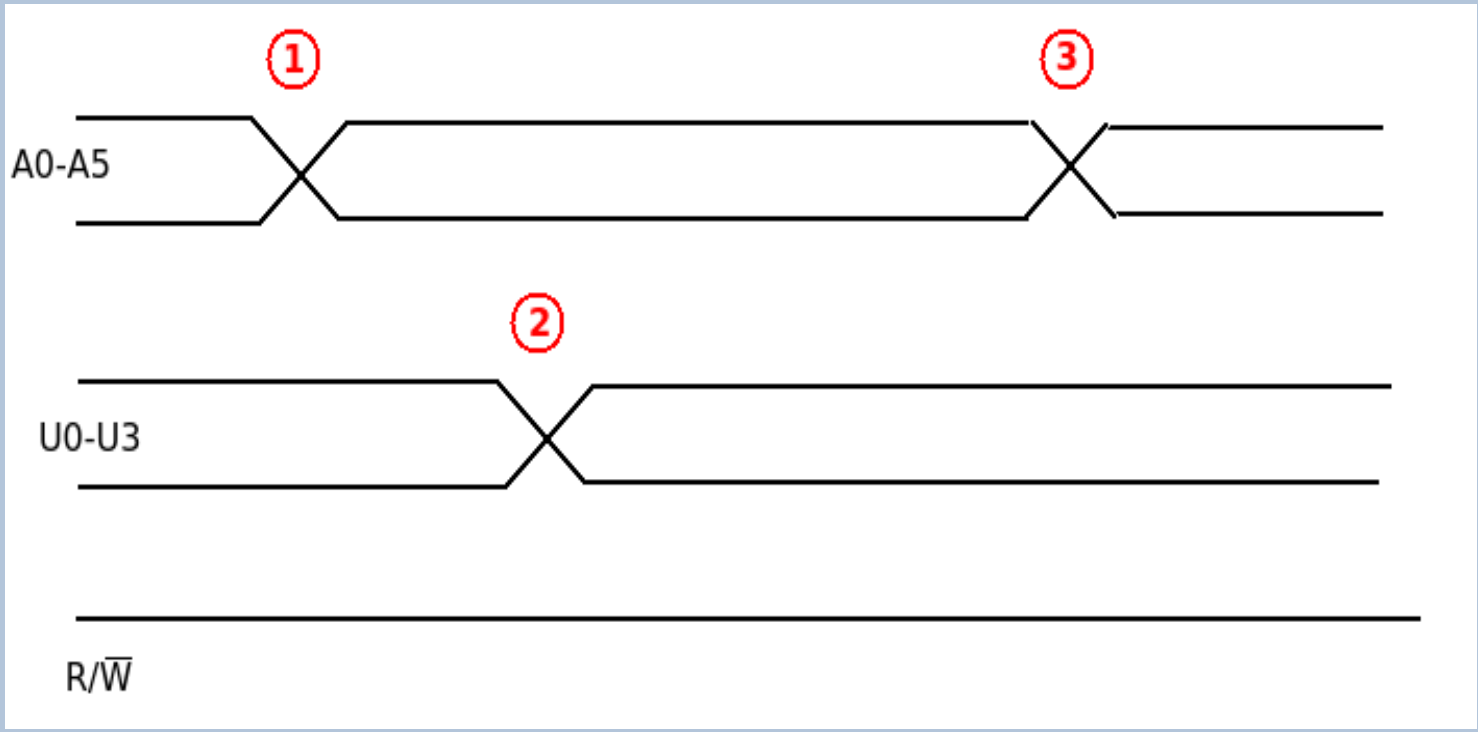
1. viene fornito l'indirizzo della parola che si vuole scrivere;
2. dopo che l'indirizzo è stabile, viene fornito il dato da scrivere;
3. quindi viene inviato un comando di scrittura (R/W = L);
4. dopo un certo tempo il comando di scrittura viene tolto (R/W = H);
5. è possibile far variare il dato;
6. si può togliere anche l'indirizzo.

Temporizzazione della sequenza

Il diagramma temporale fornisce anche i valori dei tempi che devono essere rispettati.

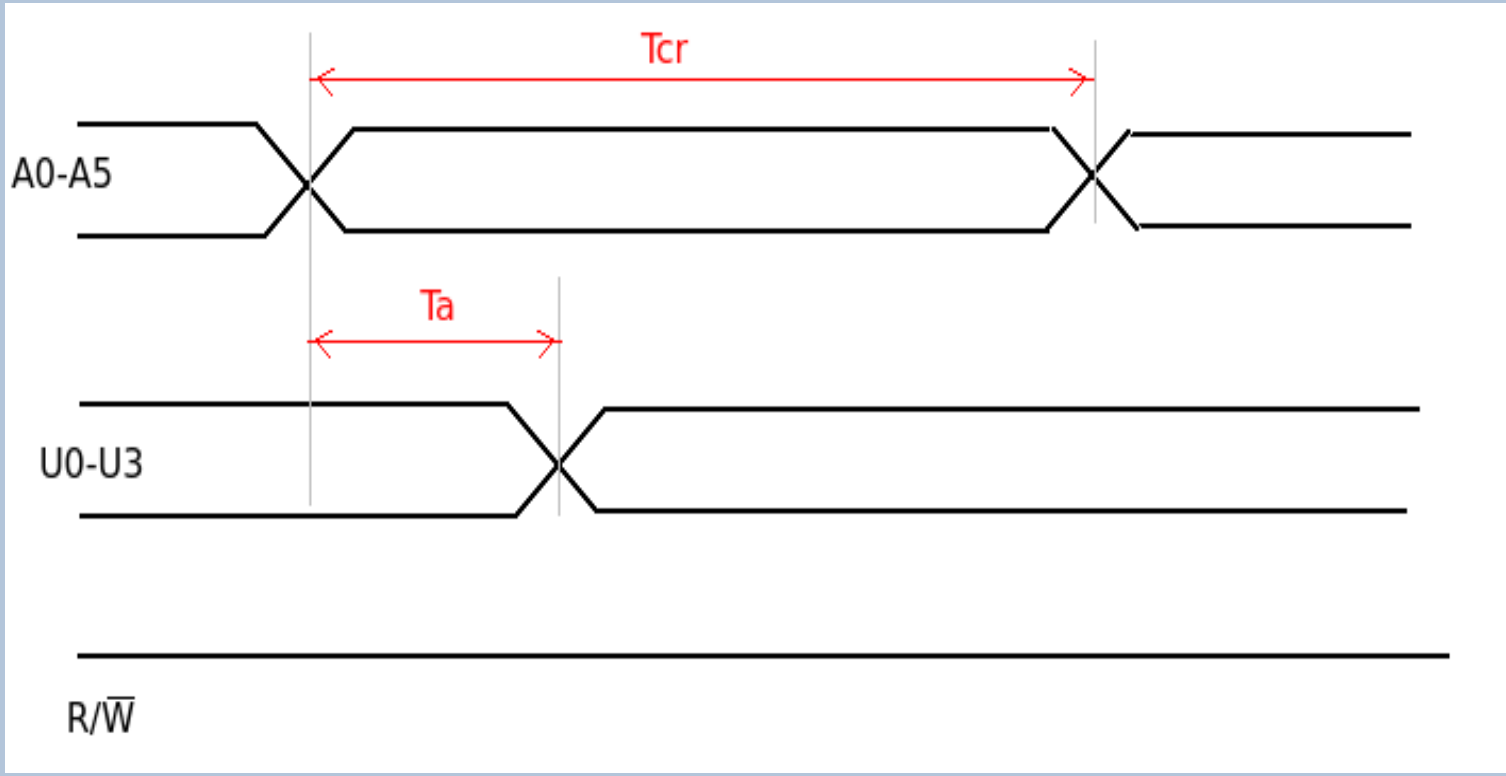
- Tcw (write cycle time, durata del ciclo di scrittura): è il minimo tempo che può intercorrere fra due operazioni successive di scrittura in memoria, cioè fra due successive commutazioni delle linee di indirizzo;
- Tsu(a): è il tempo di set-up dell'indirizzo rispetto al comando di write, cioè quanto tempo prima (al minimo) l'indirizzo deve essere stabile prima che possa essere dato un comando di scrittura;
- Tsu(d): è il tempo di set-up del dato rispetto al comando di write, cioè quanto tempo prima (al minimo) il dato deve essere stabile prima che possa essere dato un comando di scrittura;
- Tw(w) (write pulse width): è la durata minima dell'impulso di scrittura;
- Th(d): è il tempo di hold del dato, cioè quanto tempo (minimo) dev'essere mantenuto stabile il dato dopo che è stato tolto il comando di scrittura;
- Th(a): è il tempo di hold dell'indirizzo, cioè quanto tempo (minimo) dev'essere mantenuto stabile l'indirizzo dopo che è stato tolto il comando di scrittura.
- I fogli tecnici (data sheet) forniscono nel dettaglio i valori di tutti questi tempi. Si noti che si tratta in generale di tempi minimi, che devono essere rispettati affinché l'operazione di scrittura abbia successo.

Ciclo di lettura in memoria



1. viene fornito l'indirizzo della parola che si vuote leggere;
2.dopo un certo tempo sui piedini di uscita compare il dato;
3. l'indirizzo può essere tolto.

Ciclo di lettura in memoria



Ciclo di lettura in memoria

- Tcr (*read cycle time*, durata del ciclo di lettura): è il minimo tempo che può intercorrere fra due operazioni successive di lettura in memoria, cioè fra due successive commutazioni delle linee di indirizzo;
- Ta (*access time*, tempo di accesso): è il tempo (minimo) che intercorre fra l'istante in cui viene fornito l'indirizzo e l'istante in cui il dato è presentato in uscita.

l'integrato IDT71V416S

18 pin di indirizzo (A0..A17) necessari per indirizzare 2^{18} = 262144 = 256K parole.

I 16 piedini I/O0... I/O15 sono invece pin bidirezionali di ingresso e uscita

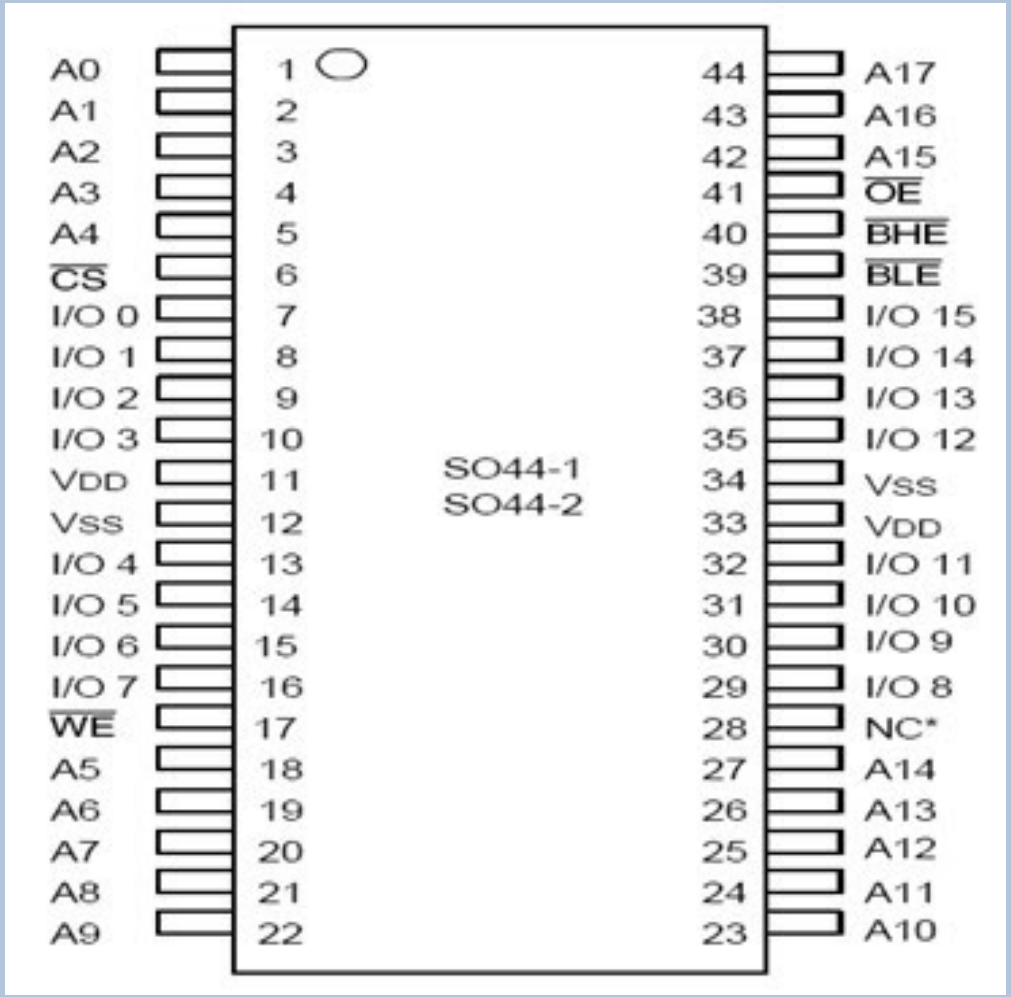


Diagramma temporale di un ciclo di lettura

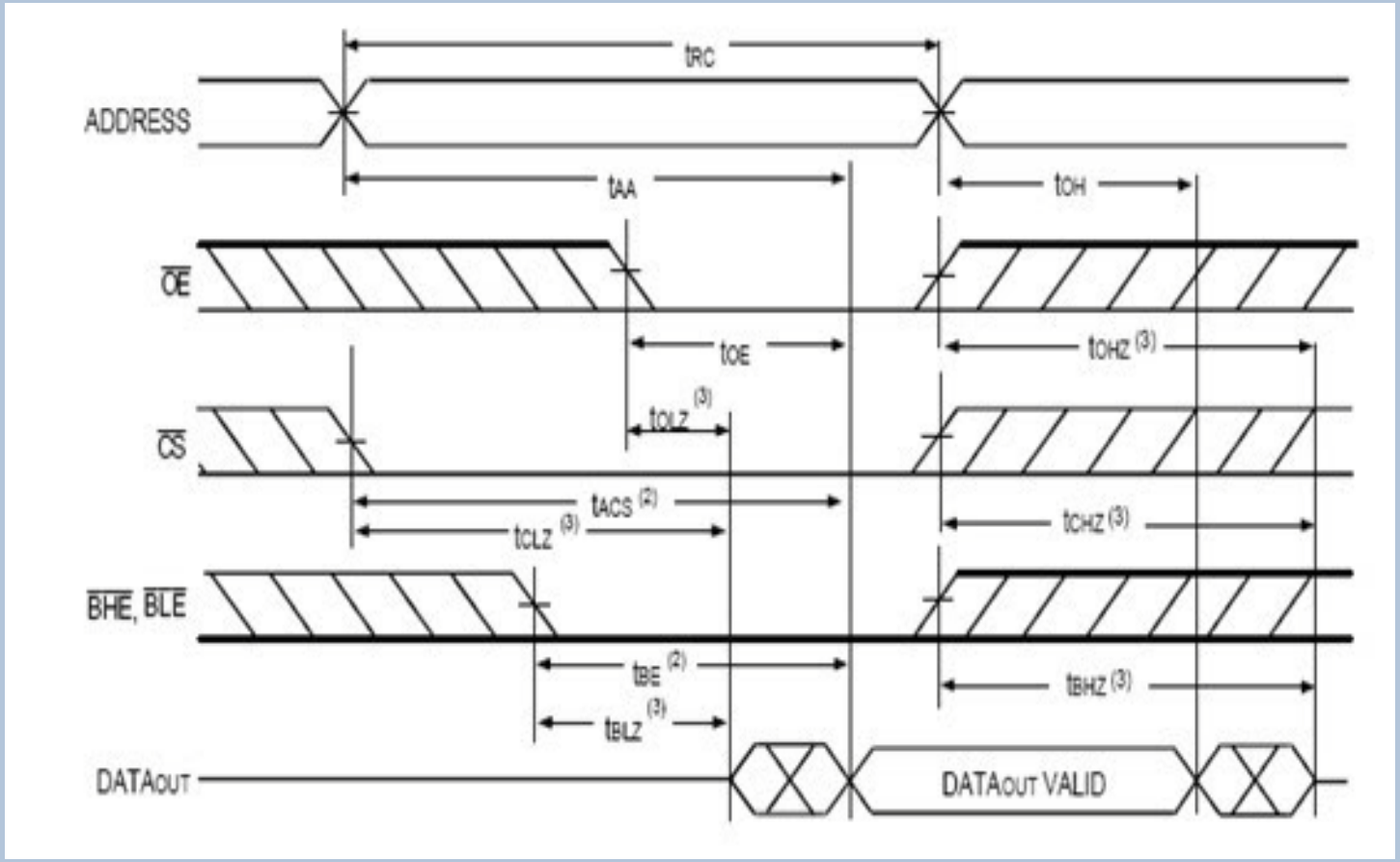
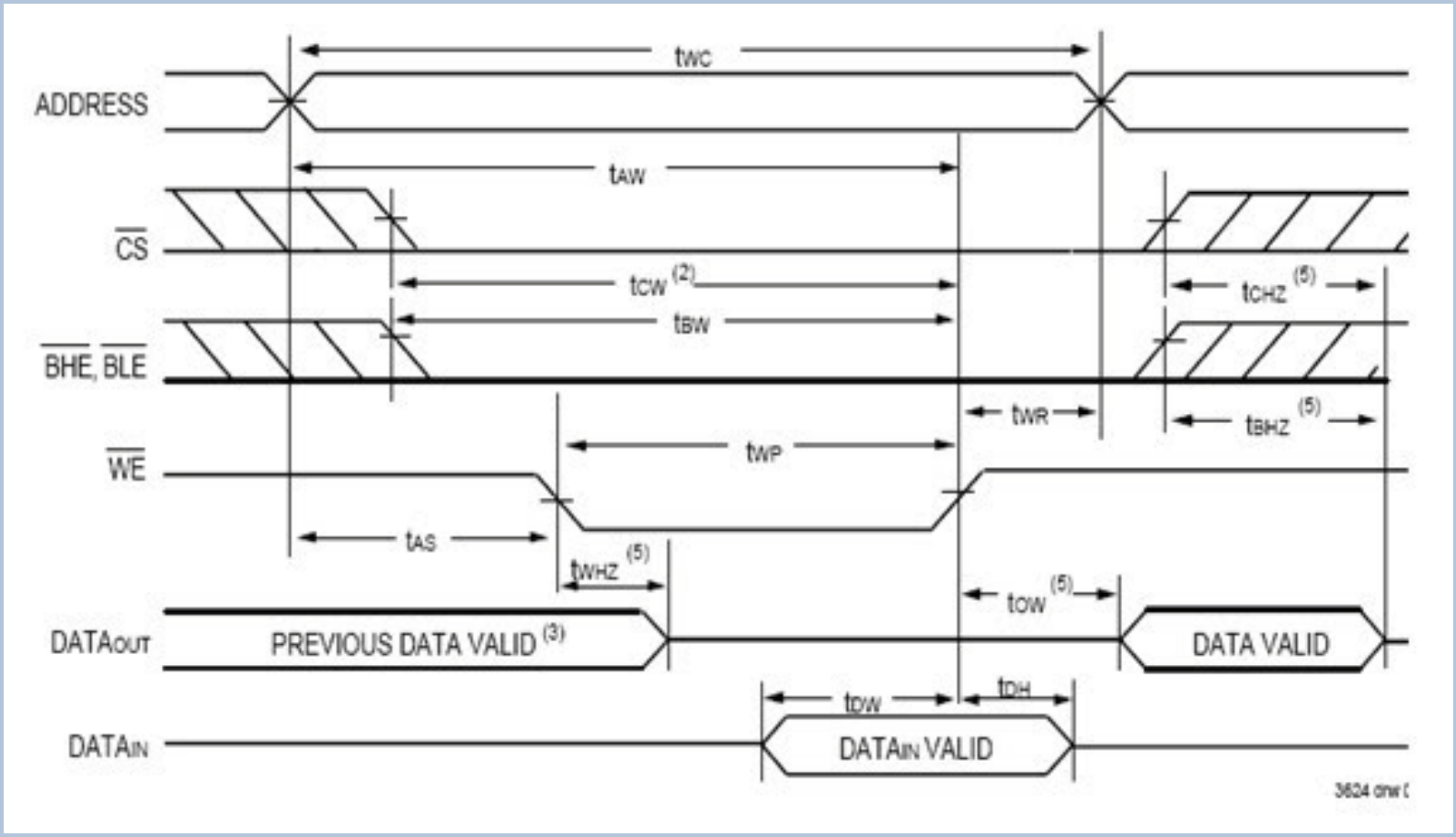


Diagramma temporale di ciclo di scrittura in memoria



Collegamento in banchi di memoria

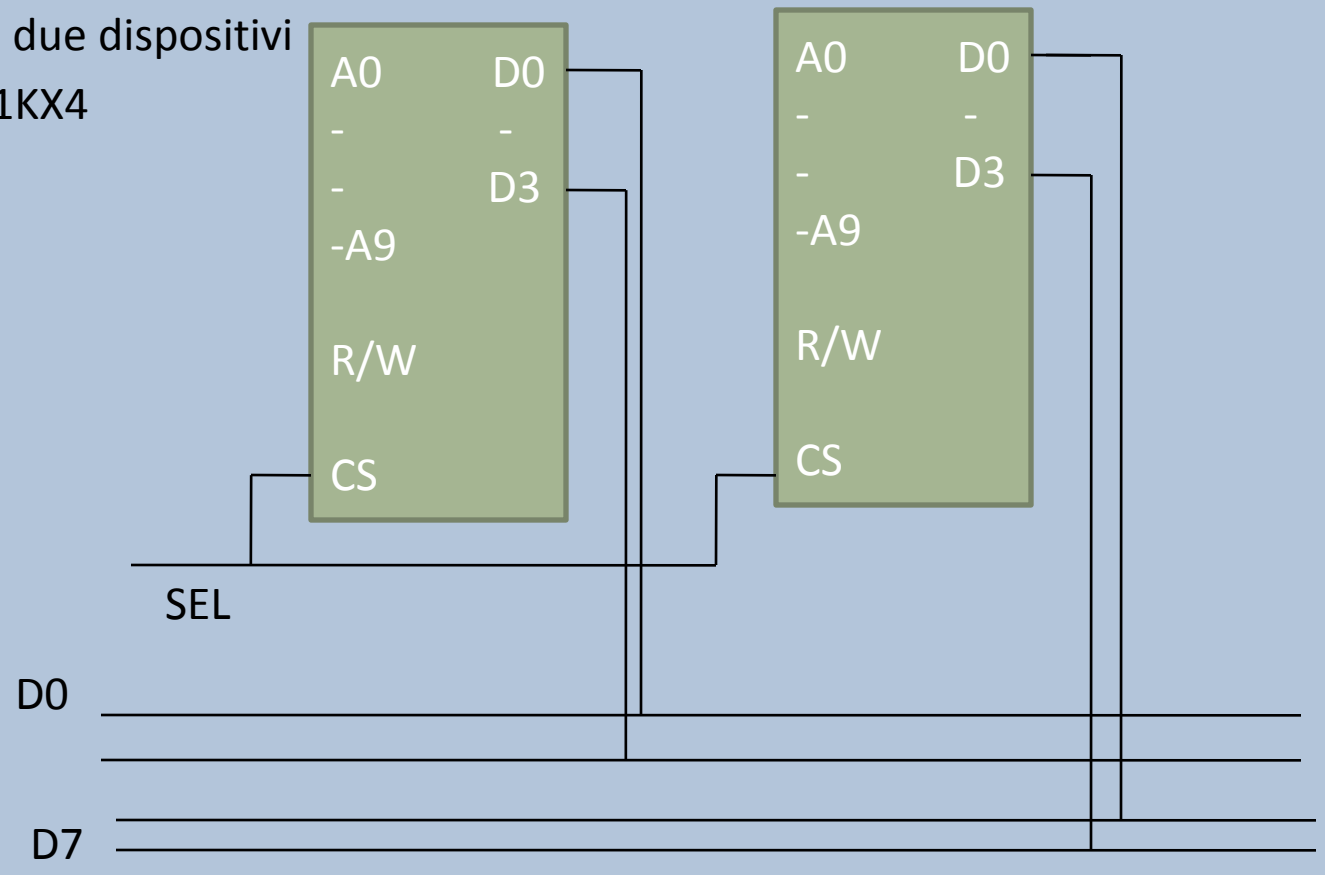
- La memoria di un sistema programmabile si realizza utilizzando i banchi di memoria, cioè una serie di dispositivi connessi in modo tale che la capacità complessiva sia la somma delle capacità dei singoli componenti.
- La connessione può essere fatta nei seguenti modi:
  - Espandendo il numero delle linee dei dati
  - Espandendo il numero delle linee di indirizzo

Espansione del numero linee dati

- Mem da 1KX8

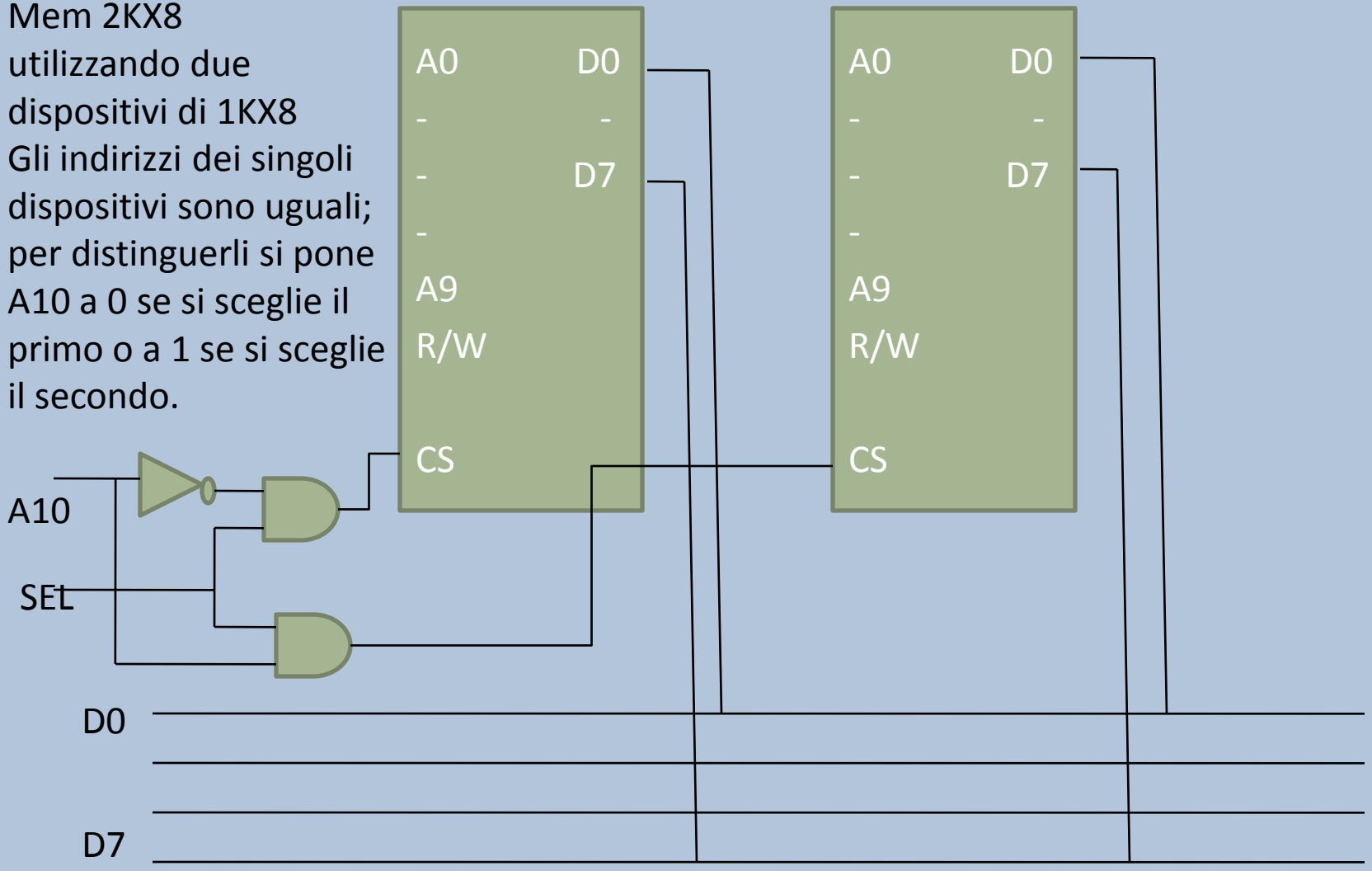
con due dispositivi

da 1KX4

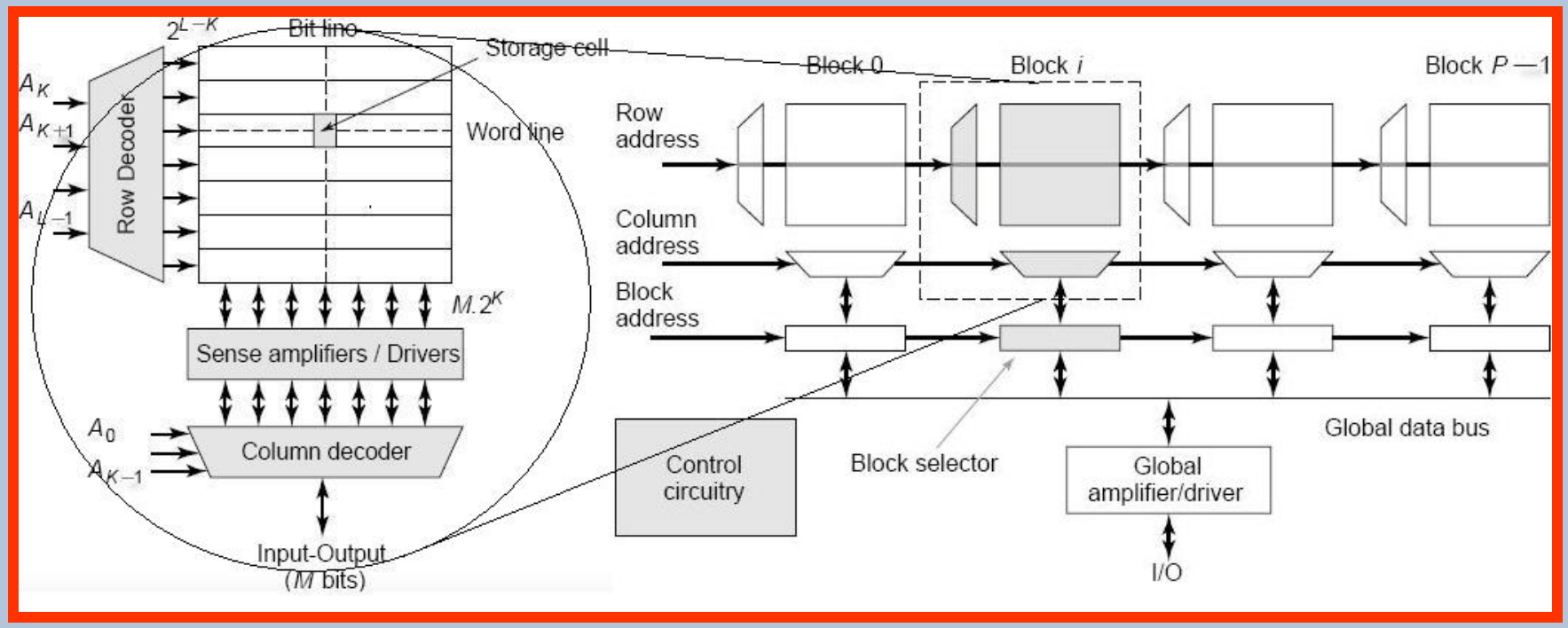


Espansione del numero linee indirizzo

Mem 2KX8 utilizzando due dispositivi di 1KX8
Gli indirizzi dei singoli dispositivi sono uguali; per distinguerli si pone A10 a 0 se si sceglie il primo o a 1 se si sceglie il secondo.



Banchi di memoria



Mappatura di memoria

- Si vuole espandere la memoria da 1kx8 a 4kx8

Disp n.1	**Prima locazione**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**
Disp. N 1	Ultima locazione	0	0	1	1	1	1	1	1	1	1	1
Disp. N 2	Prima locazione	0	1	0	0	0	0	0	0	0	0	0
Disp. N 2	Ultima locazione	0	1	1	1	1	1	1	1	1	1	1
Disp. N 3	Prima locazione	1	0	0	0	0	0	0	0	0	0	0
Disp. N 3	Ultima locazione	1	0	1	1	1	1	1	1	1	1	1
Disp. N 4	Prima locazione	1	1	0	0	0	0	0	0	0	0	0
Disp. N 4	Ultima locazione	1	1	1	1	1	1	1	1	1	1	1

Per poter espandere la memoria da una capacità *Ci* ad una *Cf* , c'è bisogno di un numero di dispositivi pari a *Cf/Ci*. Ogni dispositivo ha le stesse caratteristiche di quello originale