Memorie

Memorizzazione di un bit di informazione

Può essere effettuata mediante due distinti meccanismi fisici.

- Utilizzando circuiti con feedback positivo: in questo caso il dato immagazzinato viene conservato fin quando la cella di memoria è alimentata. Si parla di **memorie statiche**

- Utilizzando la carica immagazzinata in una capacità per preservare il dato memorizzato: in questo caso la carica tende a modificarsi nel tempo a causa di correnti di perdita indesiderate (leakage). Il dato può essere conservato solo per un tempo limitato. Si parla di **memorie dinamiche**

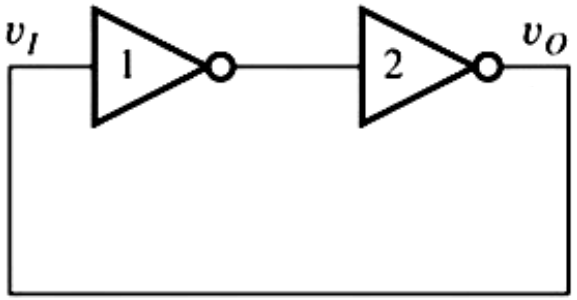
Memorizzazione di un bit di informazione

Le memorie statiche sono più “robuste”.

Le memorie dinamiche sono più semplici ed occupano quindi meno area.

Il circuito bistabile

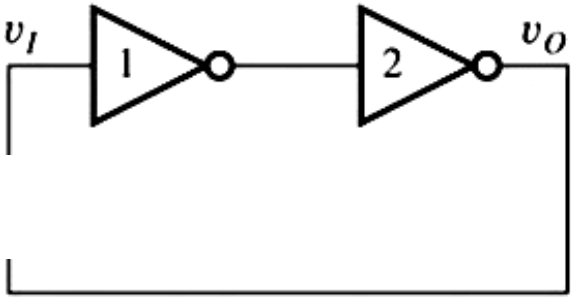
L'elemento base dei circuiti di memoria statici è il bistabile elementare.
Il circuito è costituito da due invertitori collegati come in figura:



Questo semplice sistema possiede **tre** possibili stati stazionari**. Due di questi stati sono stabili, l'altro è instabile**.

Punti di equilibrio

Supponiamo di interrompere il collegamento come mostrato in figura:

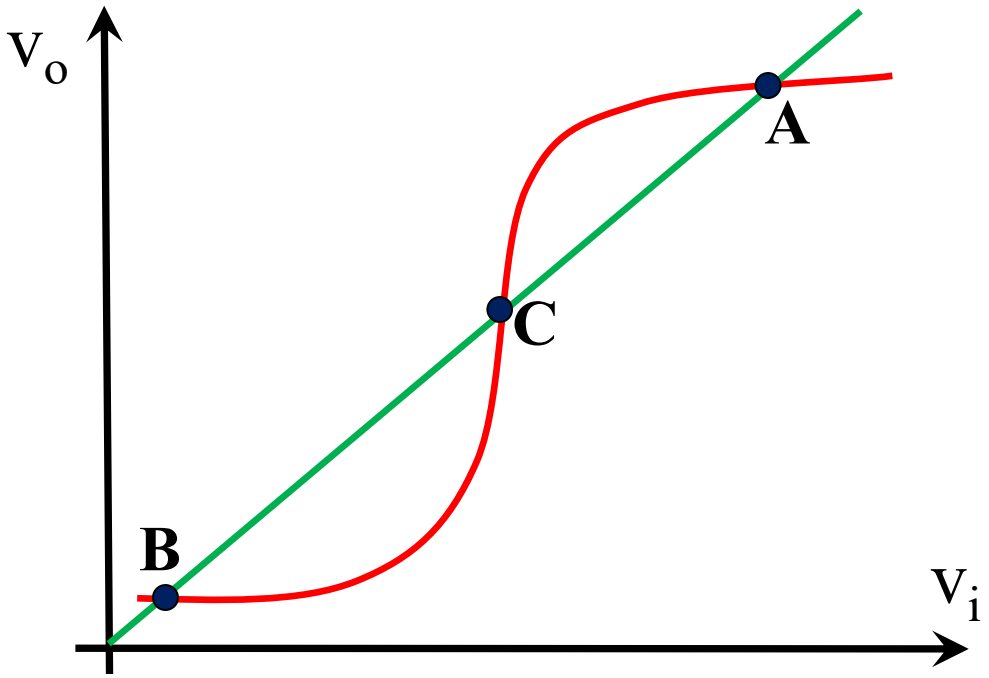


La relazione che lega v_o con v_i **sarà di tipo non-invertente** (poiché ci sono due inverter in cascata):

Ad ingresso basso corrisponde uscita bassa e ad ingresso alto corrisponde uscita alta

Punti di equilibrio

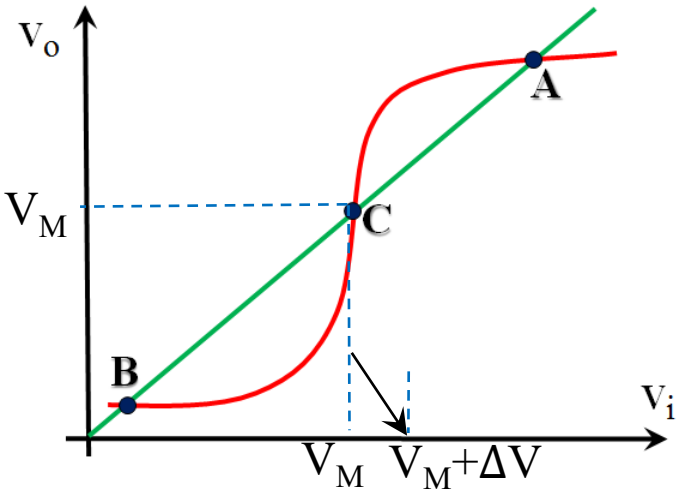
Per ottenere i punti di lavoro dobbiamo considerare che nel circuito effettivo $v_o = v_i$. Questa relazione rappresenta nel piano v_o, v_i una retta inclinata a 45°:

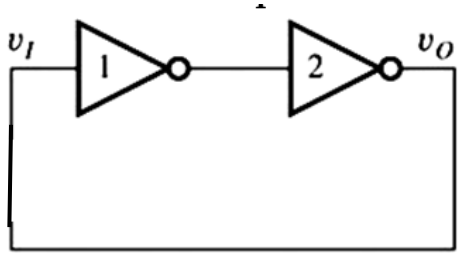


I punti di equilibrio del circuito sono tre, dati dalle intersezioni delle due curve.
A corrisponde alla condizione $v_o=v_i=V_{OH}$.
B corrisponde alla condizione $v_o=v_i=V_{OL}$.

Punti di equilibrio

Mostriamo che il punto **C** è di equilibrio instabile, mentre **A** e **B** sono punti di equilibrio stabile

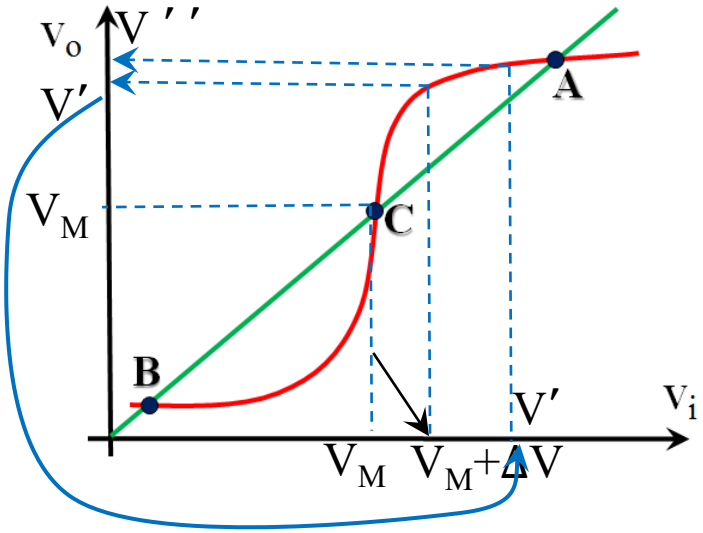


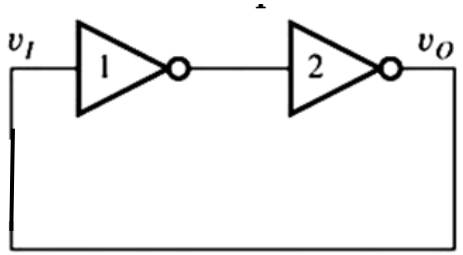


Supponiamo di trovarci inizialmente in **C**, con $v_o=v_i=V_M$.
A causa di un disturbo, la tensione v_i aumenti, divenendo: $V_M+\Delta V$

Punti di equilibrio

La tensione di uscita aumenta, diventando:V′
Poiché $v_o=v_i$, la tensione V′ si ritrova in ingresso:

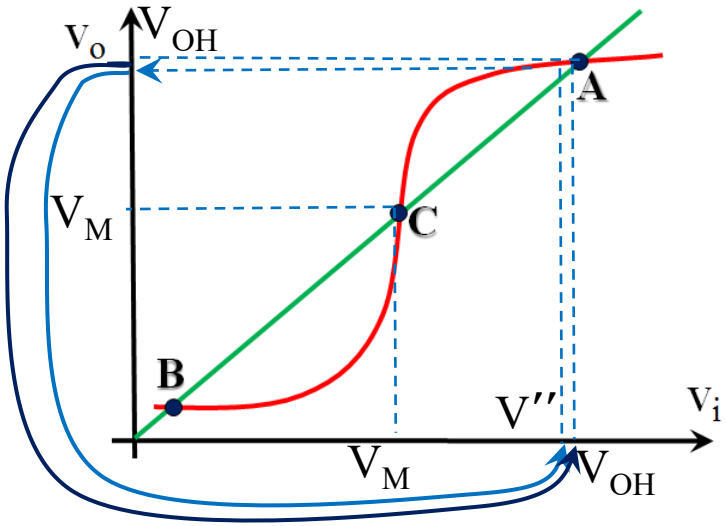


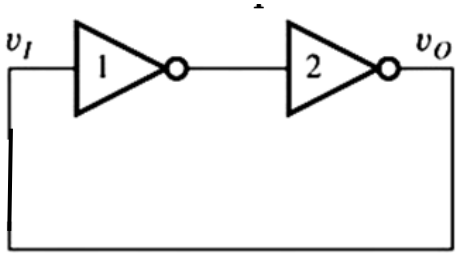


L’uscita diviene V"

Punti di equilibrio

La tensione di uscita aumenta ulteriormente, diventando:V′′
Il processo prosegue fin quando il punto di lavoro raggiunge **A**

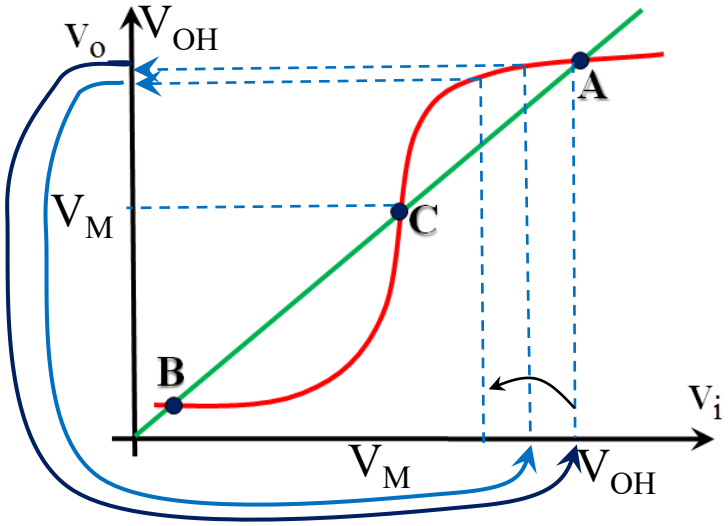




Ragionando in modo analogo, si può vedere che partendo dal punto **C**, la presenza di un disturbo negativo sulla tensione v_i porta il sistema nel punto **B**

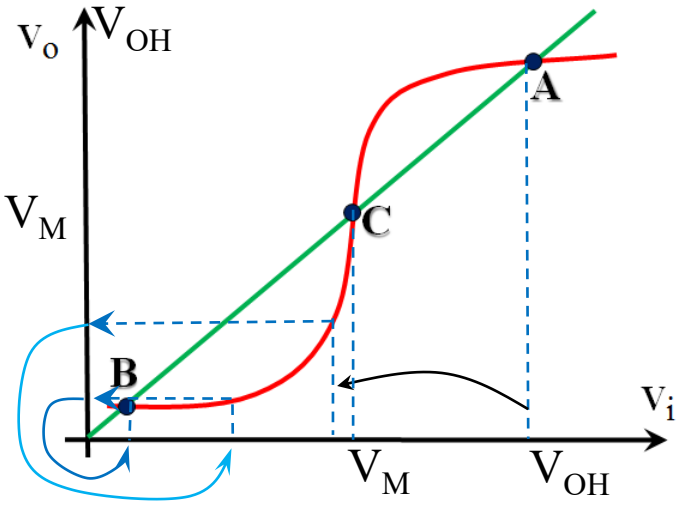
Punti di equilibrio

I punti **A** e **B** sono stabili: se mi trovo ad esempio in **A**, la presenza di un disturbo sposta temporaneamente il punto di lavoro che converge nuovamente in **A**



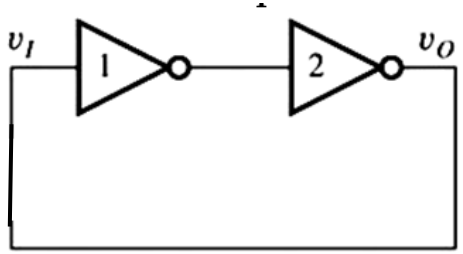
Punti di equilibrio

Se mi trovo in **A** e la tensione v_i si abbassa al di sotto di V_M, il punto di lavoro converge in **B** (cambio lo stato del bistabile)



Latch

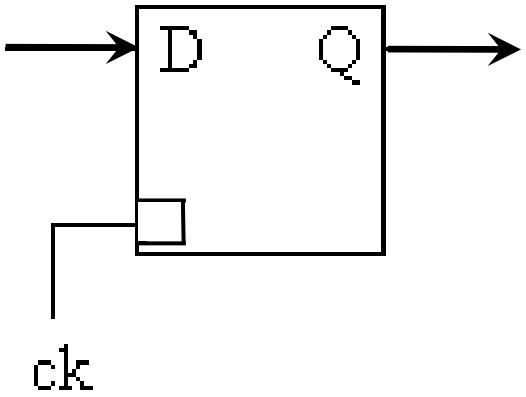
Un bistabile elementare non dispone di segnali di controllo che consentano di posizionarlo in uno dei due stati stabili. Quando il circuito viene alimentato, dopo un transtorio, raggiunge casualmente uno dei due stati



Un latch (letteralmente: “chiavistello” è un circuito che consente invece di controllare lo stato in cui il bistabile si trova. In un latch è possibile immagazzinare un bit di informazione utile.

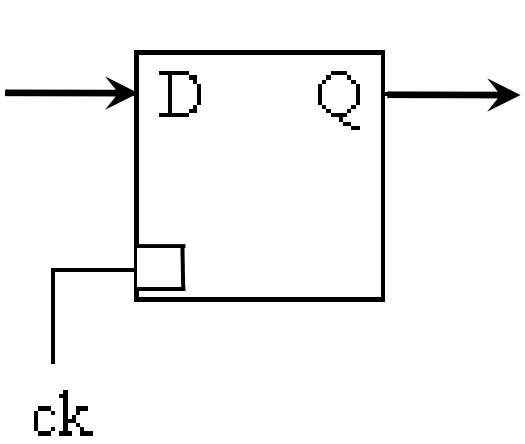
D-latch

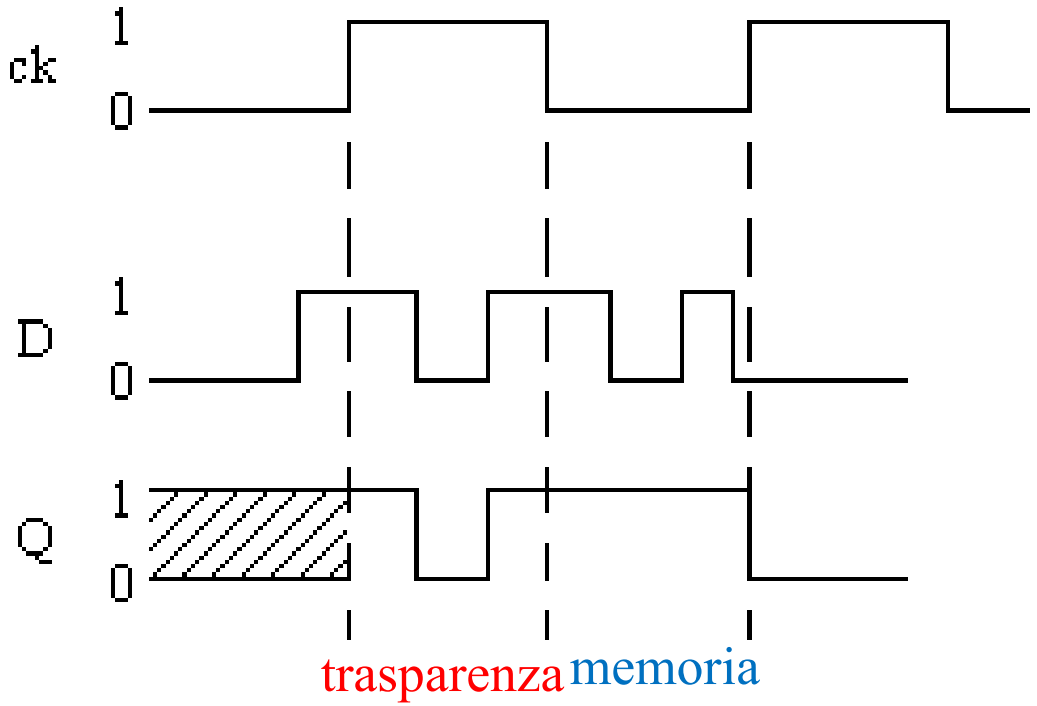
Un D-latch ha due segnali di ingresso D e ck ed un segnale di uscita, Q.
ck è il segnale di temporizzazione o **clock**



Quando il clock è attivo (ck=1) il valore presente sull'ingresso D viene riportato sull'uscita Q. Questa è la fase di "trasparenza" (l'uscita segue l'andamento dell'ingresso).
Quando il clock non è attivo (clock=0) il valore presente sull'ingresso D non modifica l'uscita Q. Questa è la fase di "memorizzazione" (il sistema memorizza l'ultimo valore di D acquisito in fase di trasparenza).

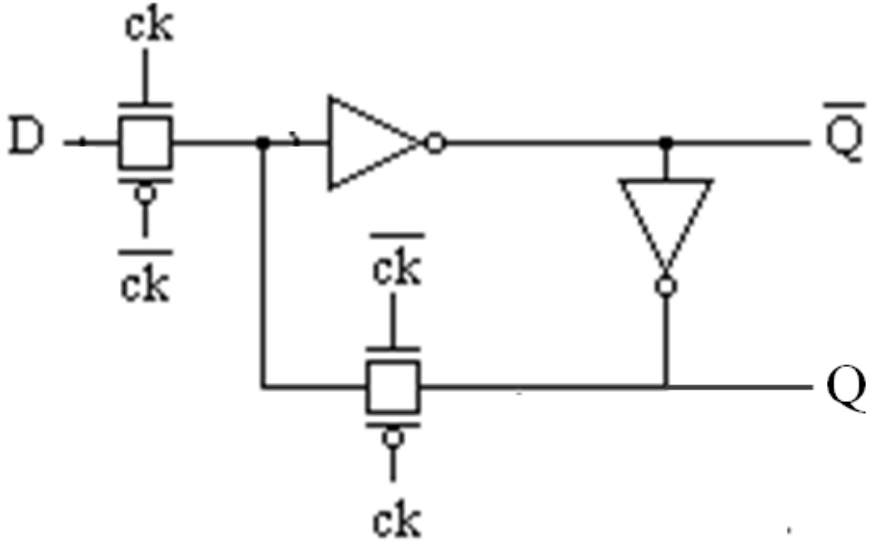
D-latch





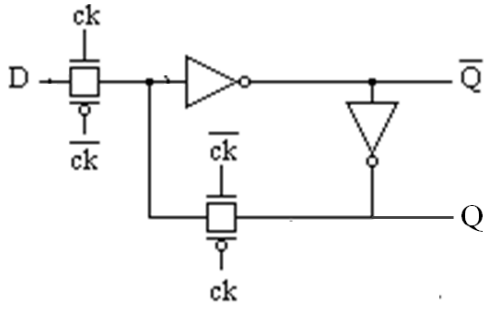
Realizzazione di un D-latch

Un D-latch può essere realizzato utilizzando due porte di trasmissione e due invertitori:

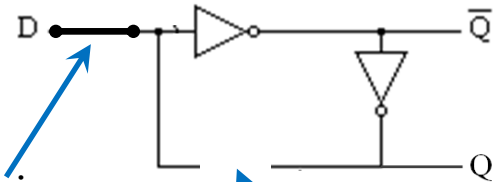


Sono disponibili in uscita sia Q che il negato

D-latch a porte di trasmissione



Configurazione per ck=1 (trasparenza):

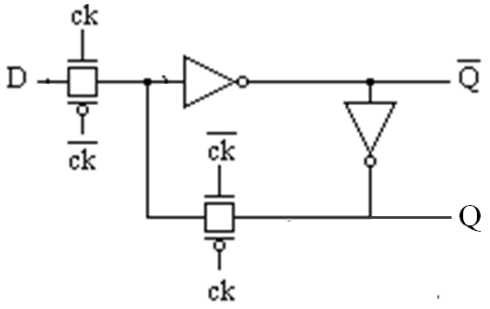


Porta di trasmissione in conduzione

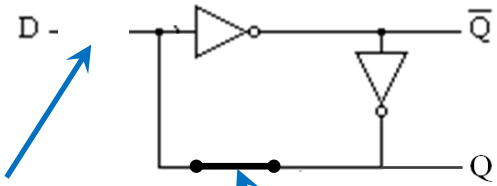
Porta di trasmissione interdetta

Il dato d'ingresso si propaga verso l'uscita

D-latch a porte di trasmissione



Configurazione per ck=0 (memoria):

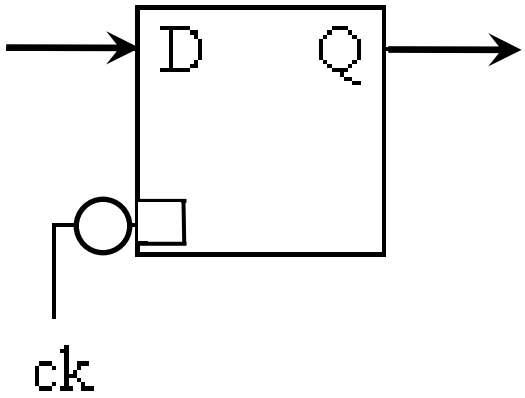


Porta di trasmissione interdetta

Porta di trasmissione in conduzione

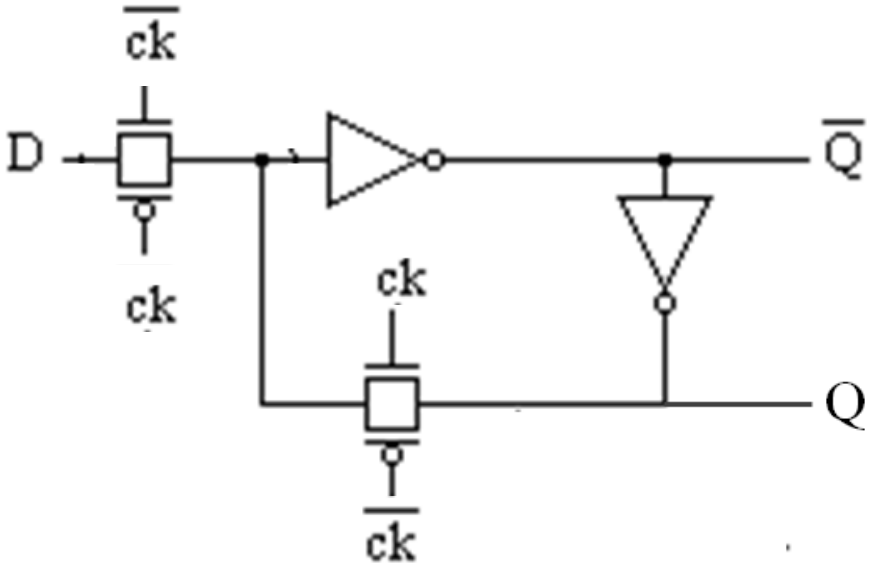
Il sistema si configura come un bistabile

D-latch trasparente sul livello basso del clock



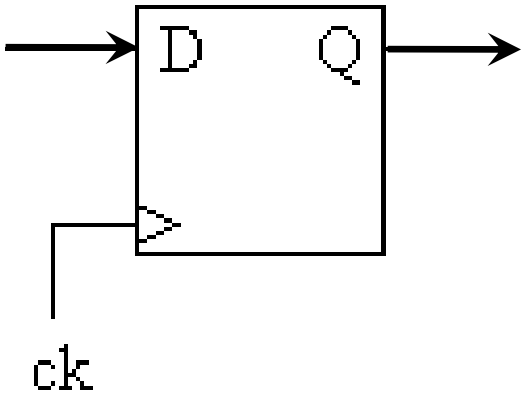
Trasparenza per ck=0;
memorizzazione per ck=1

Per realizzarlo, è sufficiente scambiare di posto i segnali ck e ck negato:



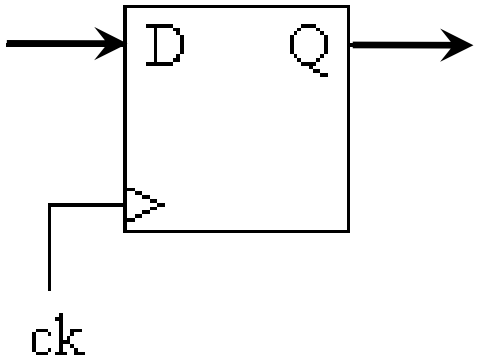
Flip-flop o registro

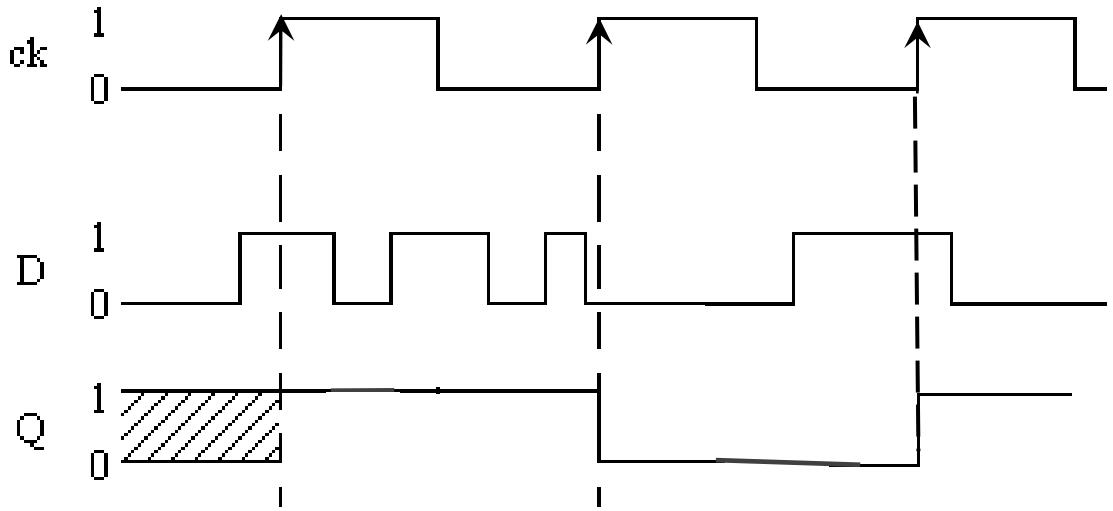
Un flip-flop D ha due segnali di ingresso D e ck ed un segnale di uscita, Q.
ck è il segnale di temporizzazione o **clock**



Mentre un latch è sensibile al livello del clock (ck=1 oppure ck=0) un flip-flop è sensibile al fronte di commutazione del clock.
Il valore del dato d'ingresso viene memorizzato in corrispondenza del fronte di salita del clock.
Un flip-flop non è mai trasparente

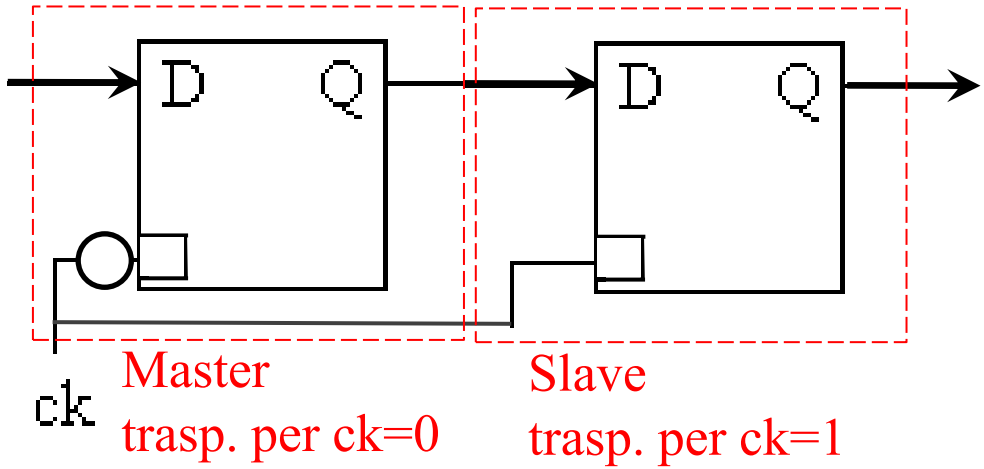
Flip-flop o registro



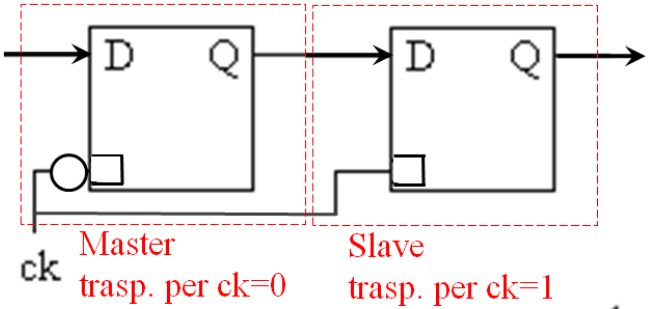


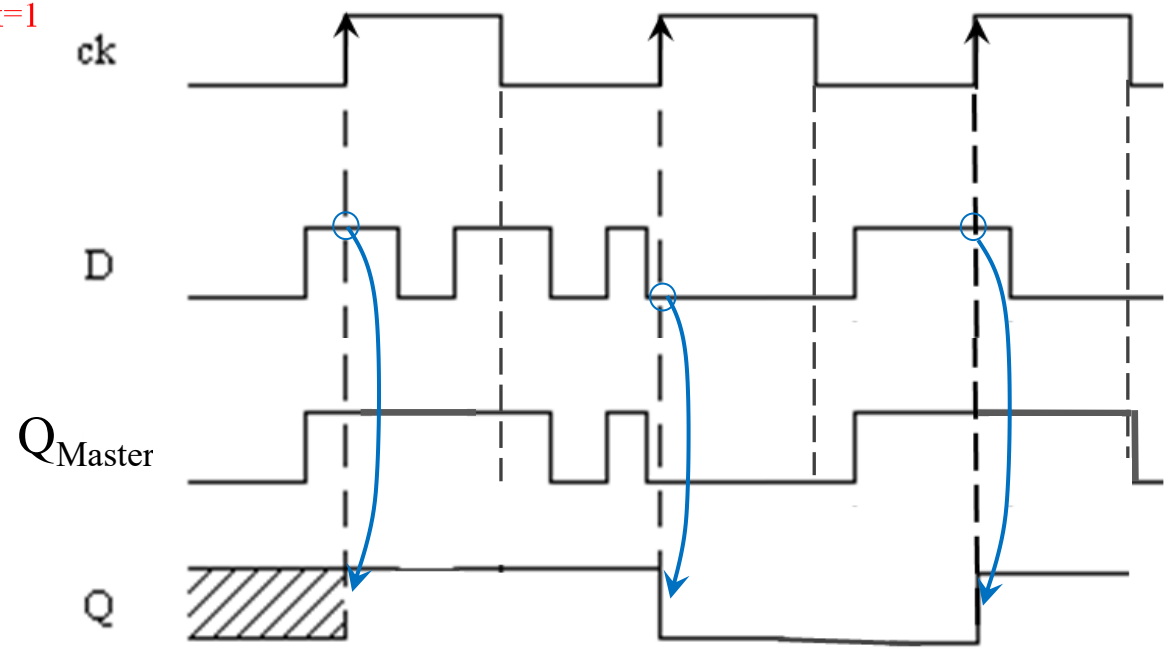
Flip-flip master/slave

Un registro si può realizzare ponendo in cascata due latch, trasparenti su i livelli opposti del clock

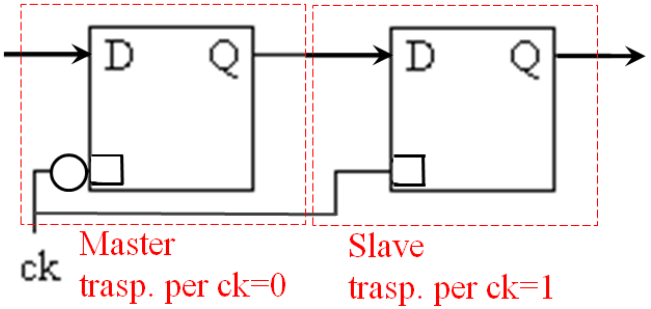


Flip-flip master/slave

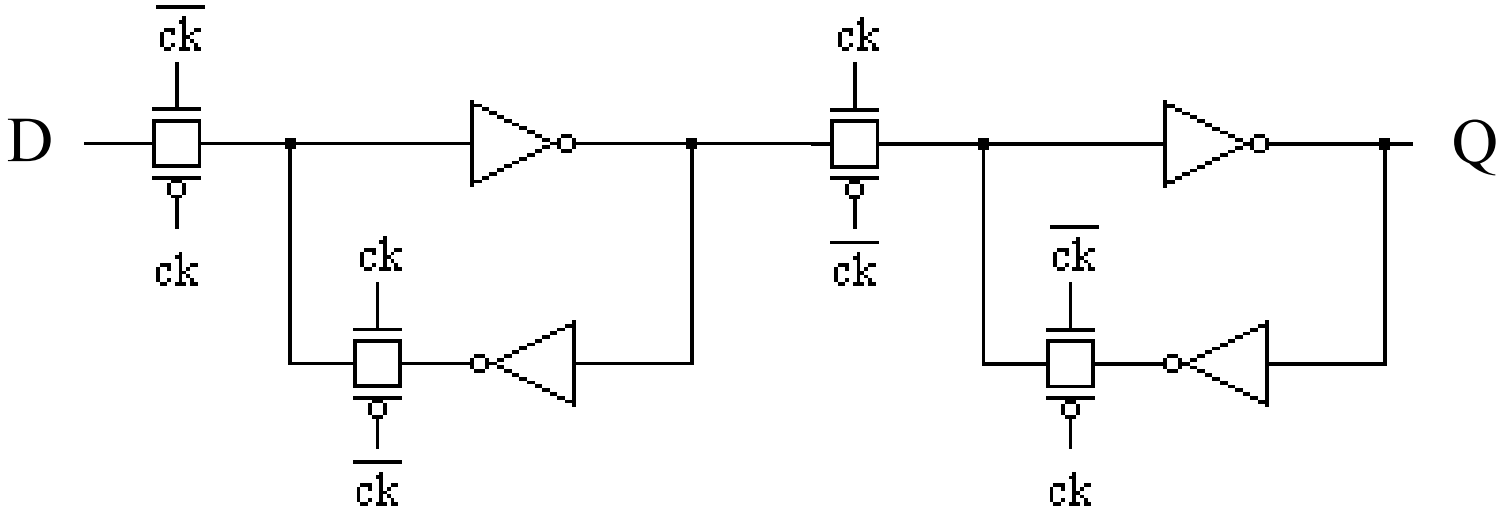




Flip-flip con porte di trasmissione

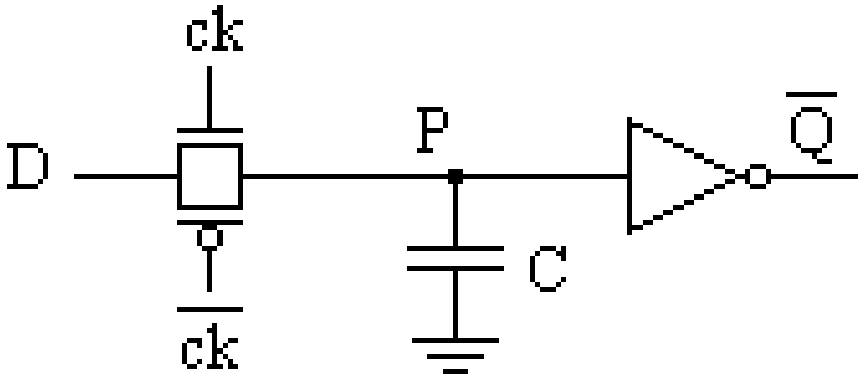


Utilizziamo le uscite negate dei due latch



D-latch dinamico

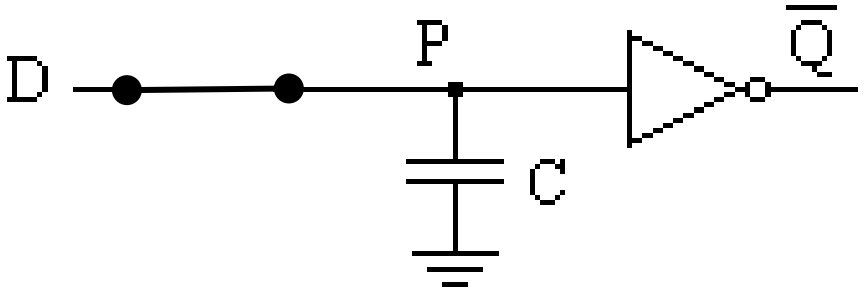
La memoria è data dalla carica accumulata in una capacità



Da notare che **il condensatore C non è un componente aggiuntivo**

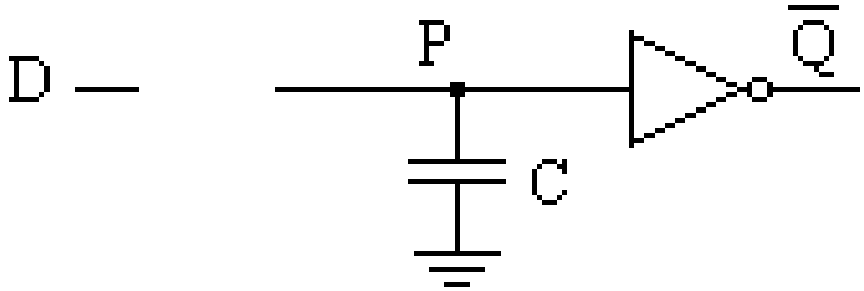
C è dato dalla capacità di ingresso dell'invertitore.

D-latch dinamico, ck=1



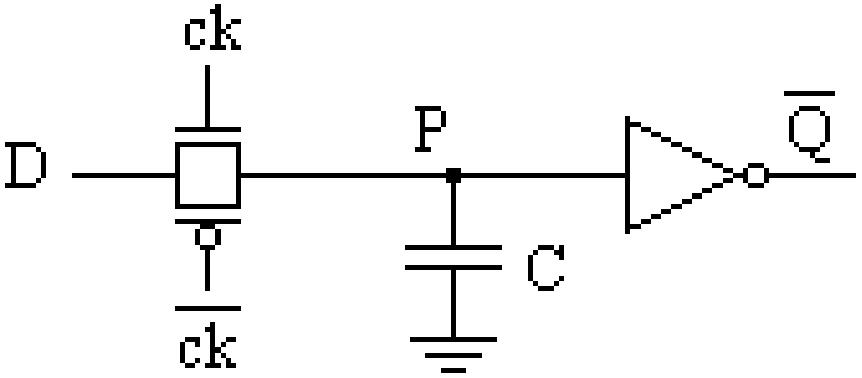
Per ck=1 il latch è trasparente; il potenziale sul nodo P coincide con quello sul nodo D.
L'uscita segue l'andamento di D (a parte l'inversione)

D-latch dinamico, ck=0



Per ck=0 il latch è in memorizzazione; il potenziale sul nodo P è costante poiché la capacità e scollegata dall'ingresso (è flottante).
L'uscita memorizza l'ultimo valore di D acquisito in fase di trasparenza.

D-latch dinamico, lekage



La fase di memorizzazione deve essere di durata breve. La porta di trasmissione non è infatti un circuito aperto ideale.
Le correnti di perdita (leakage) della porta di trasmissione modificano nel tempo la carica immagazzinata nella capacità.
Da dove proviene il leakage? Ricordiamo che i MOS presentano dei diodi parassiti verso il substrato, la cui corrente inversa (sebbene molto piccola) è in grado di alterare la carica della C

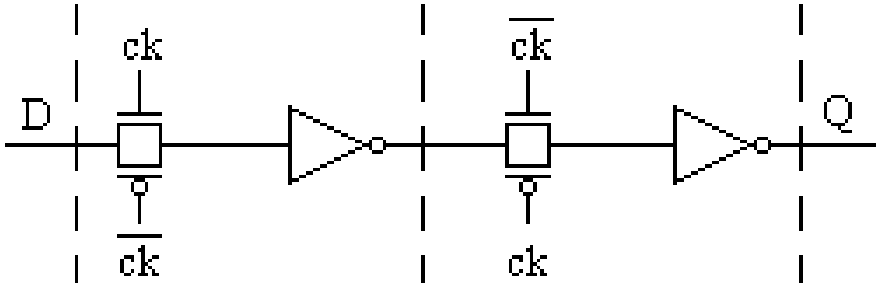
D-latch dinamico, leakage

$$\frac{dV}{dt} = \frac{I}{C}$$

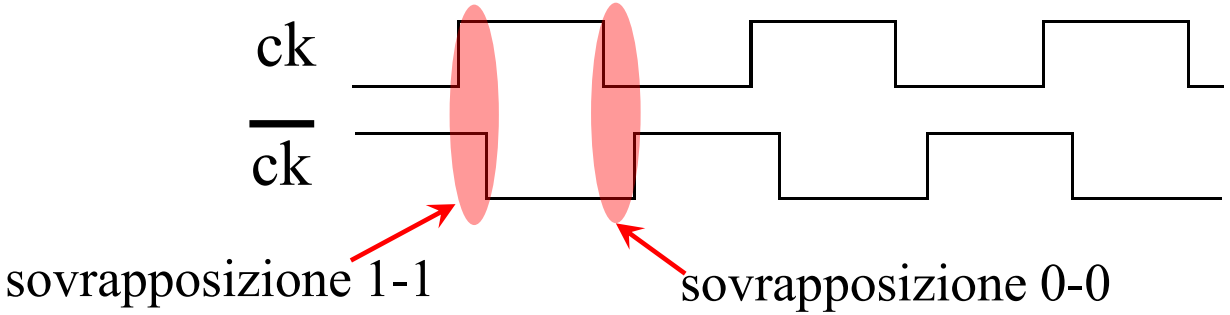
Es: una corrente di leakage di 0.1fA induce un gradiente di tensione di 1V/ms su una capacità di 0.1pF.
La durata massima della fase di memorizzazione è dell'ordine del millisecondo.
E' sufficiente che il clock abbia una frequenza superiore al KHz per soddisfare questo vincolo.

Flip-flop dinamico

Struttura master-slave

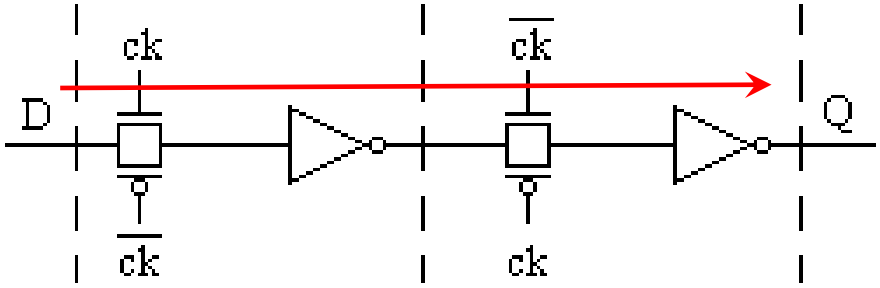


Problema della sovrapposizione fra ck ed il suo negato:



Flip-flop dinamico

Struttura master-slave



Durante la sovrapposizione 1-1 conducono entrambi gli NMOS: il sistema è trasparente ed il dato D raggiunge direttamente l'uscita.
Problema analogo per la sovrapposizione 0-0 quando conducono entrambi i PMOS.
Per risolvere questo problema si adottano logiche a più fasi di clock.

Memorie

Memorie

I flip-flop vengono utilizzati per realizzare sistemi sequenziali, come registri in cui memorizzare temporaneamente delle informazioni ecc.

Quando si devono memorizzare elevate quantità di dati si adottano circuiti più semplici, in cui si sacrificano velocità, margini di rumore ecc. per raggiungere la massima compattezza.

Classificazione delle memorie

Memorie a sola lettura: ROM (read-only memory):
I dati contenuti nella memoria sono definiti una volta per tutte all'atto della realizzazione del circuito. La memoria conserva i suoi dati anche in assenza di alimentazione.

Memorie a lettura-scrittura: RWM (read write memory):
I dati contenuti nella memoria possono essere riscritti. La memoria perde i dati immagazzinati in assenza di alimentazione. Si utilizza comunemente l'acronimo RAM (random-access memory) che evidenzia come i tempi di accesso della memoria non dipendano dalla particolare locazione su cui si opera.

Le RAM si suddividono in SRAM (static-RAM) e DRAM (dynamic RAM) a seconda se utilizzano un bistabile oppure la carica in una capacità per memorizzare un bit d'informazione.

Memorie non-volatili

Memorie non-volatili: sono a metà strada fra ROM e RAM.
I dati contenuti nella memoria possono essere aggiornati, sebbene i tempi di scrittura siano maggiori rispetto a quelli di lettura. La memoria conserva i suoi dati anche in assenza di alimentazione.

Memorie non-volatili

Si suddividono in:

PROM: (Programmable-ROM) sono programmabili dall'utente una sola volta.
EPROM: (Erasable PROM) sono programmabili dall'utente più volte ma necessitano di una fase di cancellazione che utilizza esposizione a radiazione ultravioletta.
EEPROM (Electrically Erasable PROM) sono programmabili dall'utente più volte grazie ad opportuni segnali di controllo (non necessitano di particolari operazioni tecnologiche).
FLASH: caratteristiche simili alle EEPROM, consentono di ottenere maggiori densità di integrazione.

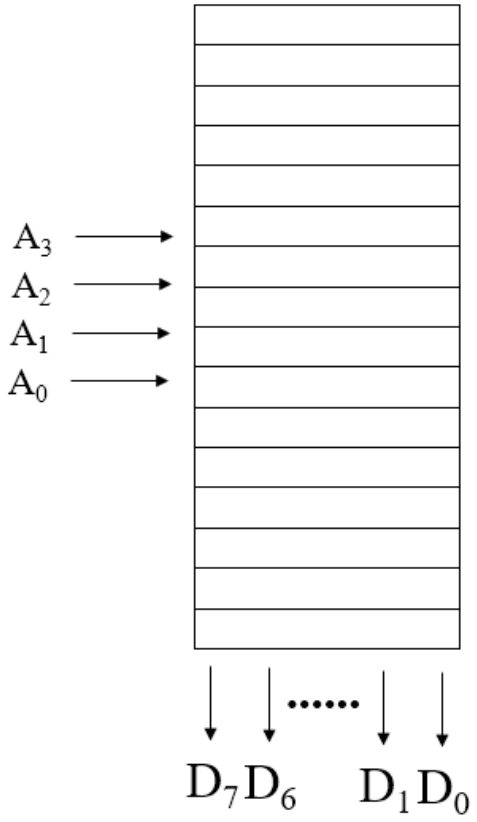
Terminali di ingresso/uscita di una memoria

Ogni memoria deve disporre di:
Un indirizzo binario costituito da P-bit, con il quale si puo' selezionare una fra 2^P locazioni di memoria.
Un bus dati da Q-bit che contiene il dato letto dalla locazione di memoria indirizzata, o il dato da scrivere nella locazione di memoria.
Alcuni segnali di controllo, per definire l'operazione da compiere (lettura o scrittura) e per temporizzarla.

Memorie non volatili

Memoria ROM

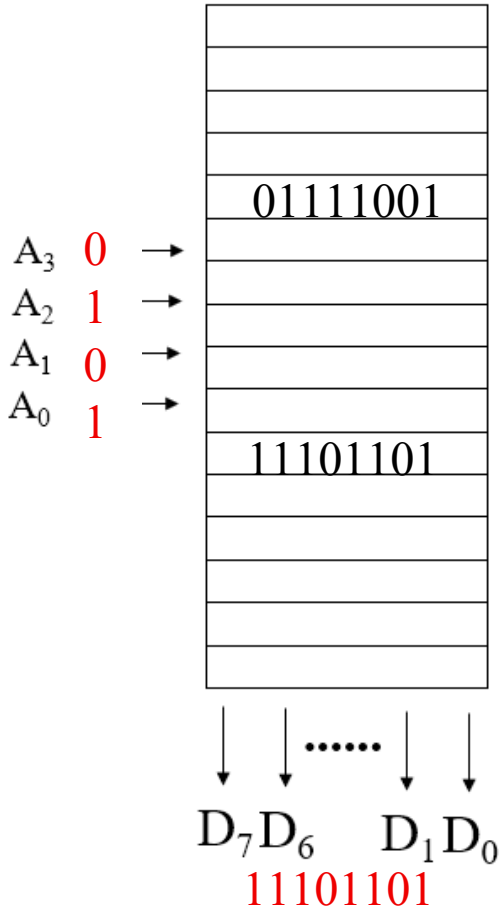
Una ROM possiede *P* linee di ingresso, che costituiscono l'*indirizzo* della memoria.
Con *P* ingressi è possibile indirizzare 2^P *locazioni di memoria*, ognuna delle quali memorizza un dato codificato su *Q* bit.
Inserendo in ingresso uno specifico indirizzo, si ottiene in uscita il dato (la "parola") immagazzinato nella locazione di memoria selezionata.
La Figura a destra mostra il caso di una ROM con P=4 linee di indirizzo e Q=8.
La capacità complessiva della memoria è:
C=16 byte (ricordiamo che un insieme di 8bit è denominato byte). La capacità in bit è data da:
C=16×8=128bit. In generale: $\mathbf{C_{BIT}=Q\ 2^P}$



Memoria ROM

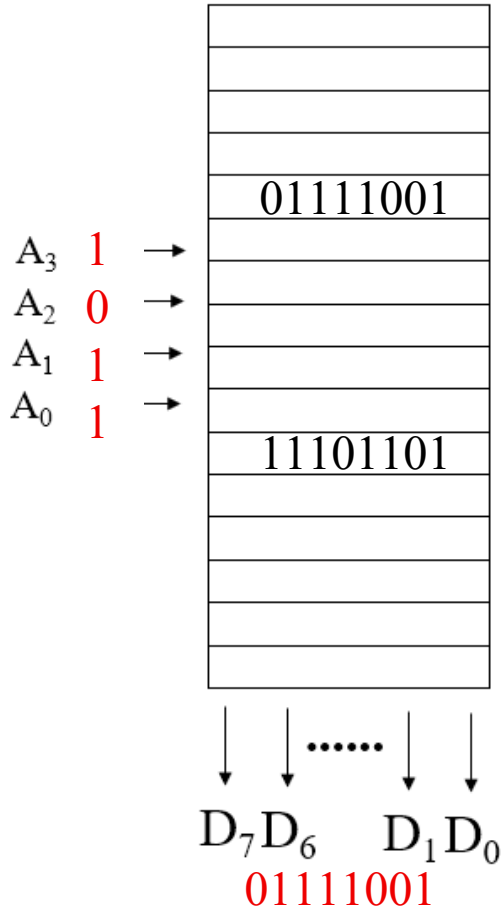
Una ROM viene chiamata "memoria" in maniera impropria, comportandosi come un circuito combinatorio: dato un indirizzo di ingresso fornisce un dato di uscita, funzione dell'indirizzo.

Nell'esempio di destra, con l'ingresso: $A_3A_2A_1A_0 = 0101$ si seleziona la locazione di memoria #5, cui è memorizzato (ad esempio) il dato 11101101. L'uscita assume il valore: $D_7D_6D_5D_4D_3D_2D_1D_0 = 11101101$



Memoria ROM

Nell'esempio di destra, con l'ingresso: $A_3A_2A_1A_0 = 1011$ si seleziona la locazione di memoria #11, cui è memorizzato (ad esempio) il dato 01111001. L'uscita assume il valore: $D_7D_6D_5D_4D_3D_2D_1D_0 = 01111001$



Struttura di una ROM

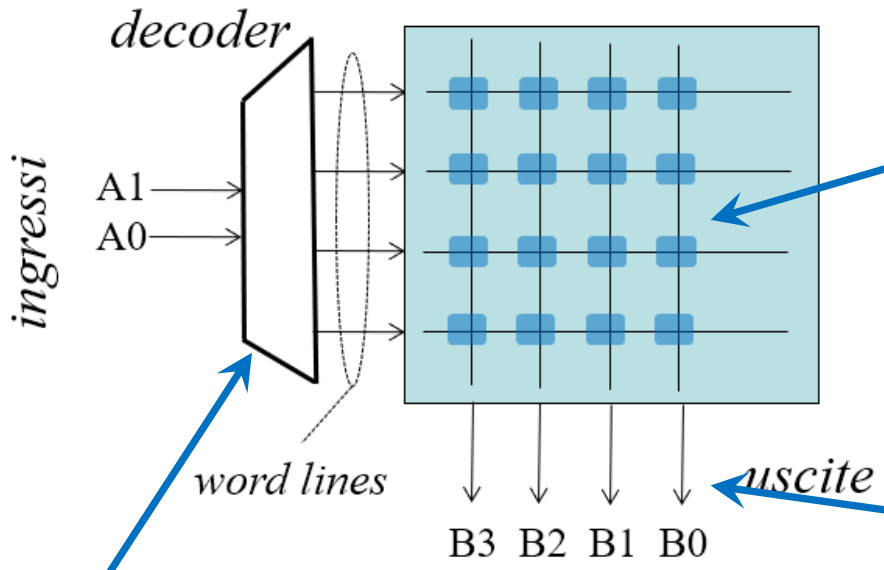
Consideriamo, ad esempio, una piccola ROM con 4 locazioni di memoria, ognuna da 4 bit.
I bit di indirizzo saranno 2.
I bit di uscita saranno 4.

Contenuto della memoria:

ingressi			uscite			
A1	A0		B3	B2	B1	B0
0	0		0	0	1	0
0	1		1	0	0	0
1	0		0	1	1	0
1	1		0	1	1	0

Struttura di una ROM

Consideriamo, per semplicità, un caso particolarmente semplice: P=2, Q=4



La memoria include una **matrice di celle elementari**, ognuna delle quali è in grado di memorizzare un bit di informazione.

Le uscite della matrice di celle elementari sono denominate **bit lines** e corrispondono alle uscite della memoria stessa.

L'indirizzo della locazione di memoria da leggere viene applicato ad un **decodificatore**. Il decodificatore ha P bit di ingresso e 2^P uscite, denominate **word lines.** Una sola delle uscite ha il valore logico 1.

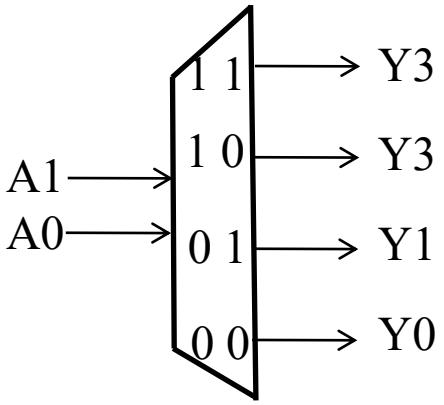
Struttura di una ROM

Abbiamo, dunque, due blocchi fondamentali: il **decodificatore** e la **matrice di programmazione**. Iniziamo a vedere la possibile struttura di un decodificatore

Decodificatore

Un decodificatore ha P bit di ingresso e 2^P uscite.
Una sola delle uscite ha il valore logico 1.
Esempio: decoder 2→4

ingressi			uscite			
A1	A0		Y3	Y2	Y1	Y0
0	0		0	0	0	1
0	1		0	0	1	0
1	0		0	1	0	0
1	1		1	0	0	0



Decodificatore

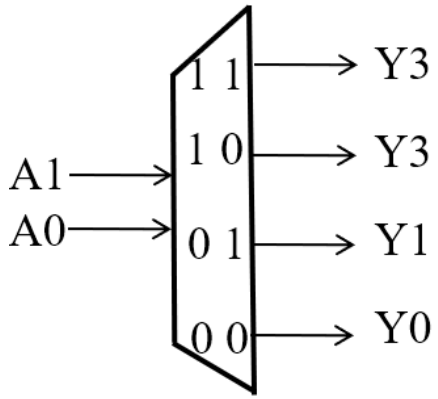
ingressi			uscite			
A1	A0		Y3	Y2	Y1	Y0
0	0		0	0	0	1
0	1		0	0	1	0
1	0		0	1	0	0
1	1		1	0	0	0

$$Y0 = \overline{A1} \cdot \overline{A0} = \overline{A1 + A0}$$

$$Y1 = \overline{A1} \cdot A0 = \overline{A1 + \overline{A0}}$$

$$Y2 = A1 \cdot \overline{A0} = \overline{\overline{A1} + A0}$$

$$Y3 = A1 \cdot A0 = \overline{\overline{A1} + \overline{A0}}$$



Il decoder può essere realizzato con porte NOR, disponendo degli ingressi e dei negati degli ingressi

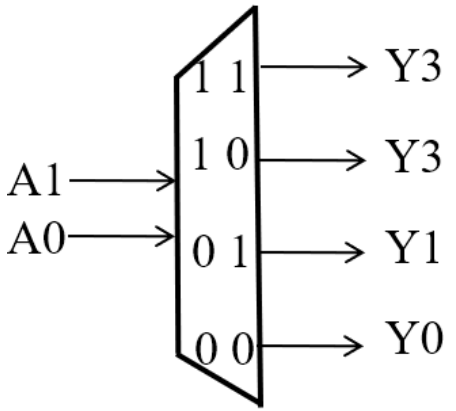
Decodificatore

$$Y0 = \overline{A1} \cdot \overline{A0} = \overline{A1 + A0}$$

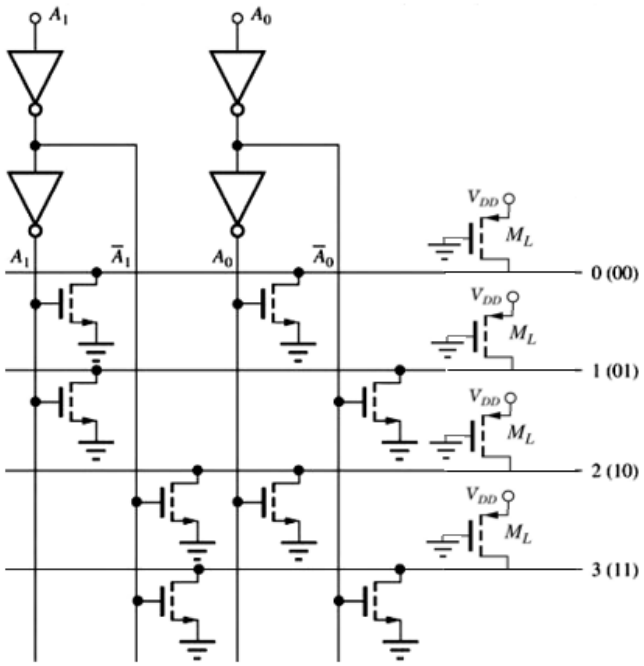
$$Y1 = \overline{A1} \cdot A0 = \overline{A1 + \overline{A0}}$$

$$Y2 = A1 \cdot \overline{A0} = \overline{\overline{A1} + A0}$$

$$Y3 = A1 \cdot A0 = \overline{\overline{A1} + \overline{A0}}$$

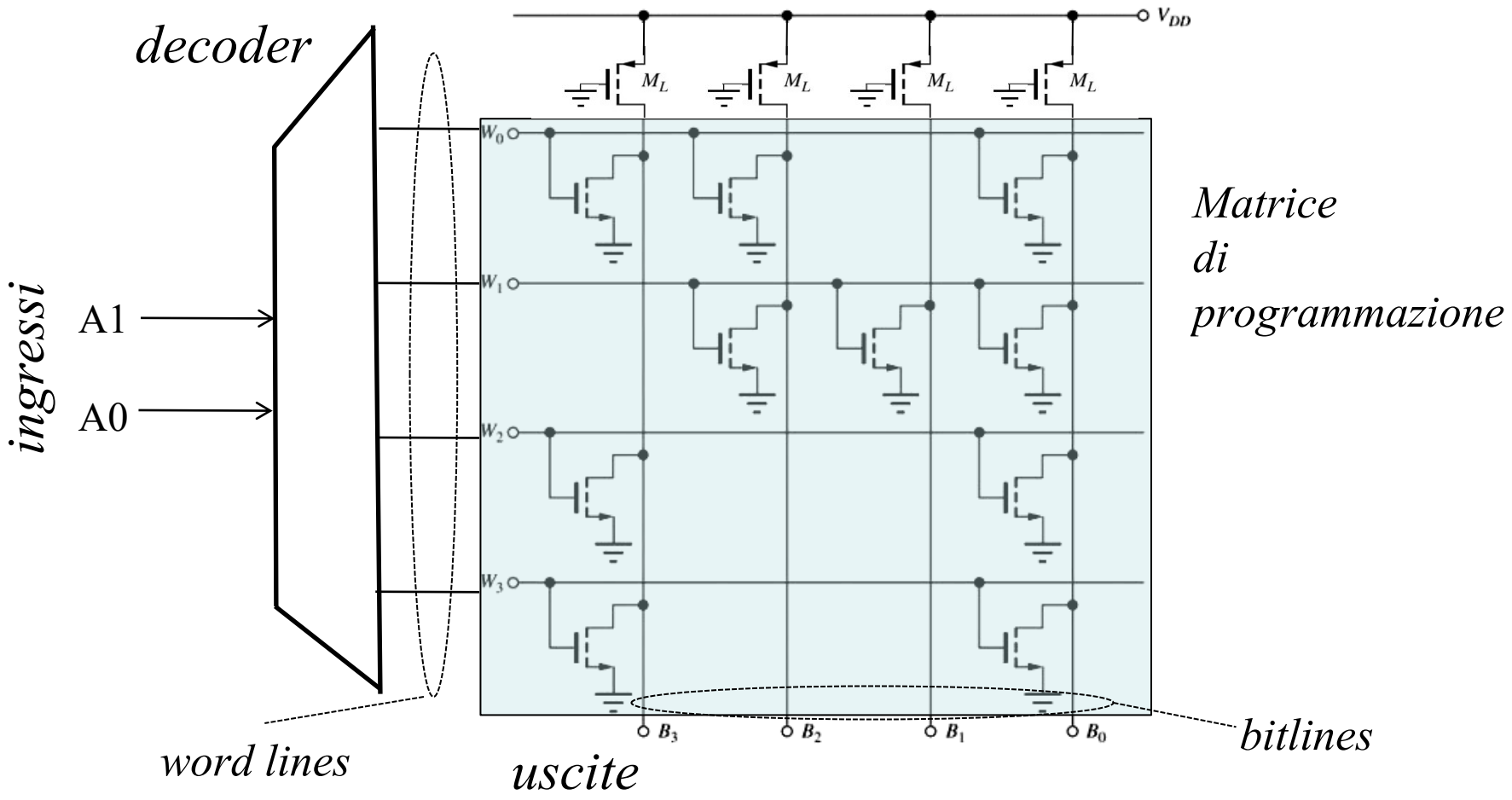


Il decoder in logica pseudo-NMOS:



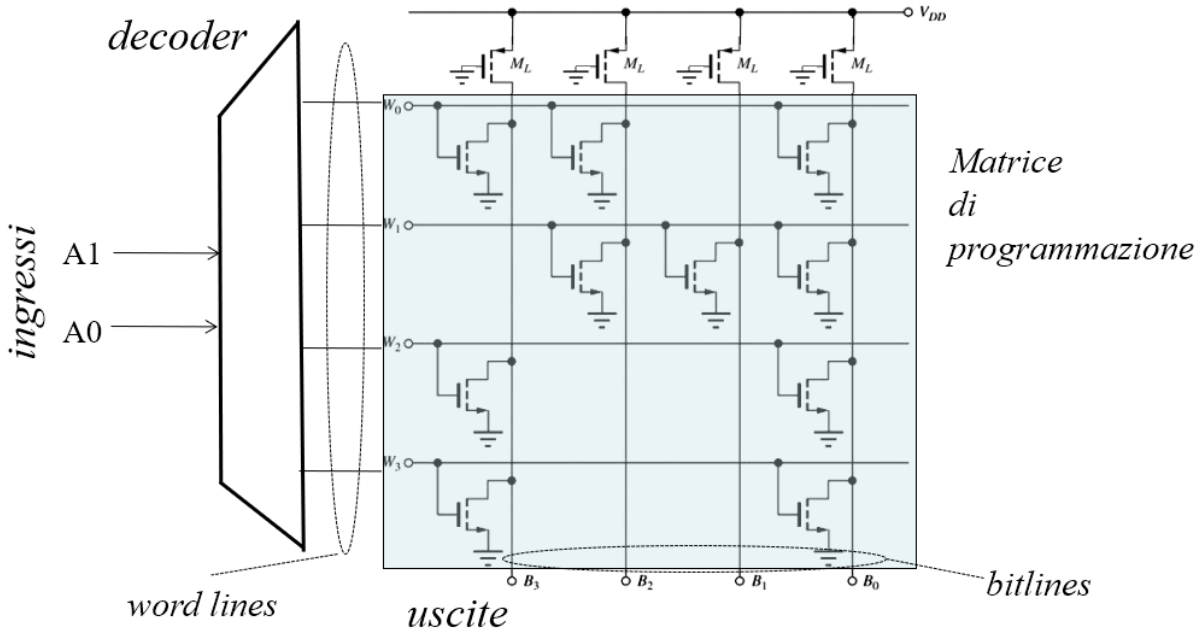
Il decoder consiste in questo esempio di 4 porte NOR a due ingressi. La Figura mostra che i dispositivi che costituiscono le porte NOR possono essere arrangiati in una struttura abbastanza regolare.

Struttura di una memoria



Struttura di una memoria

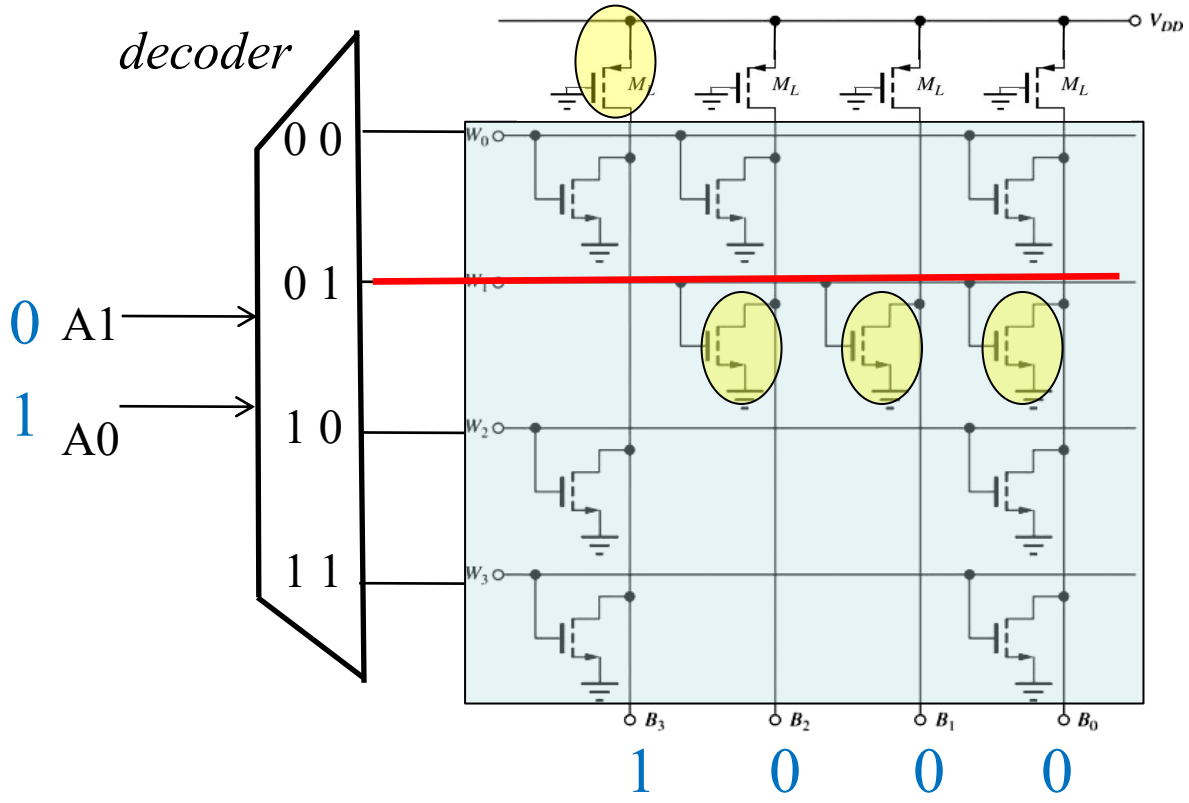
Come si può vedere dalla Figura, anche la matrice di programmazione è costituita da un insieme di porte NOR. In questo esempio abbiamo 4 porte NOR (quante sono le uscite), ognuna delle quali ha 4 ingressi (quante sono le bitlines).



I dispositivi NMOS che compaiono nella rete di pull-down della matrice di programmazione vengono inseriti opportunamente, in modo da ottenere la programmazione desiderata.

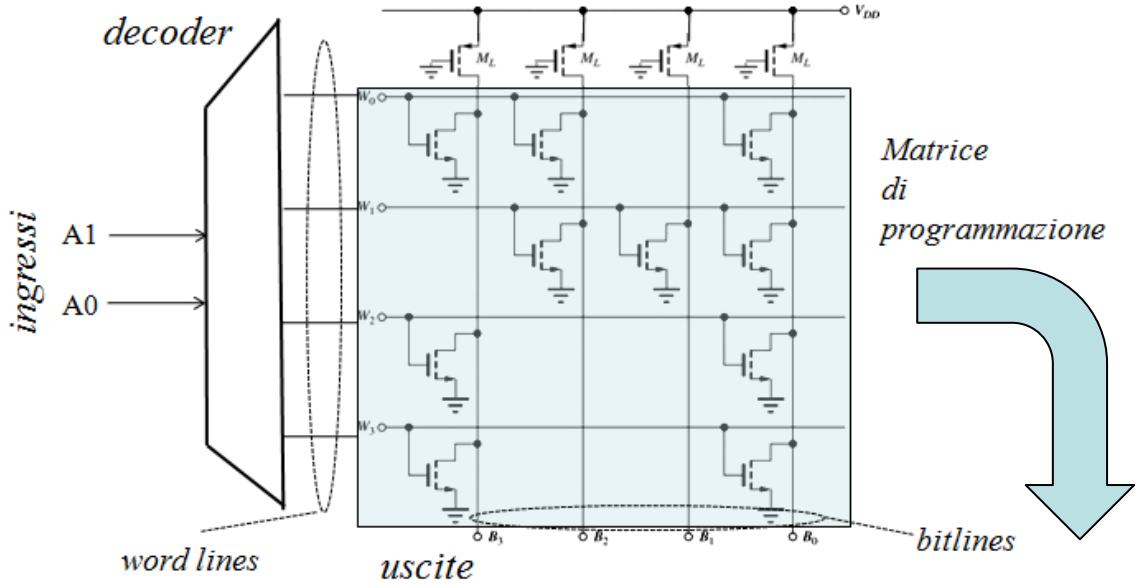
Esempio di funzionamento

Supponiamo: A1=0; A0=1. L'uscita alta del decoder è W1

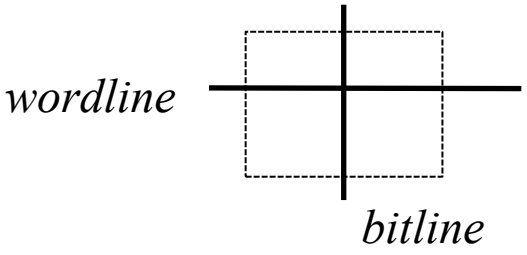


I dispositivi evidenziati si attivano e portano a zero le uscite B2,B1,B0
L'uscita B3 è ad 1 grazie al dispositivo di pull-up

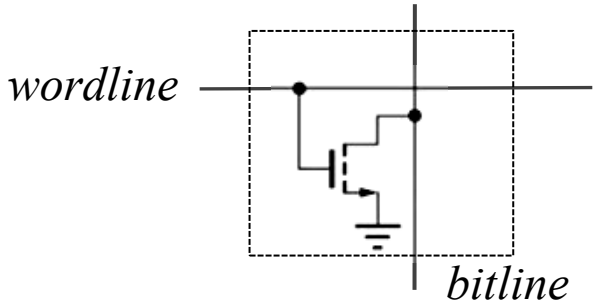
Struttura di una memoria



Memorizzo un “1”



Memorizzo uno “0”



Read-Only Memory (ROM)

Nelle ROM programmate “a livello maschera” il contenuto della memoria viene definito all’atto della costruzione del chip, creando o non creando i dispositivi MOS nella varie locazioni della matrice di memoria.

Svantaggio: approccio non versatile: la modifica di un singolo bit della memoria richiede il progetto di un nuovo chip, con i costi ed i tempi relativi.

Progammable ROM (PROM)

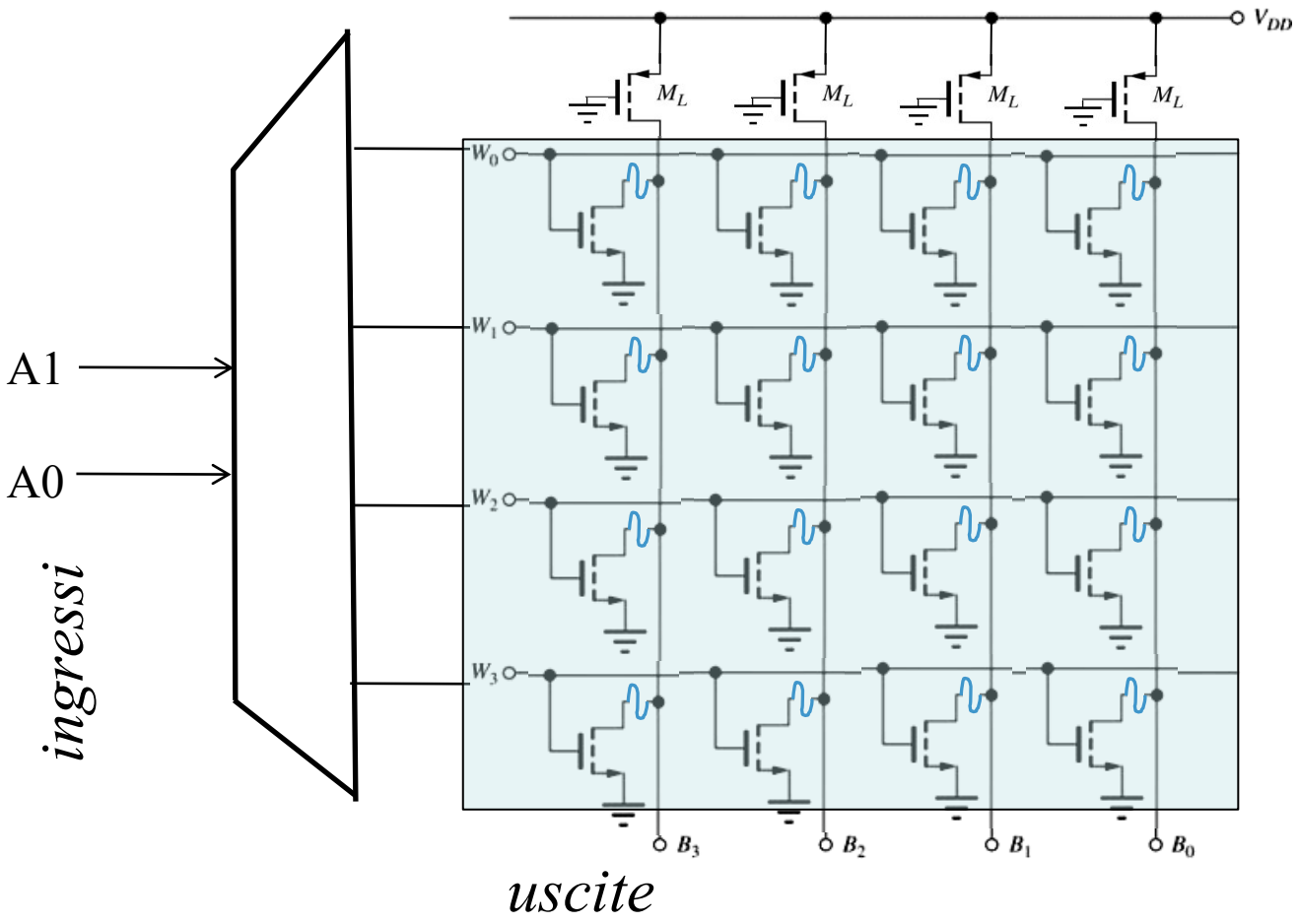
Memorie PROM: la matrice di memoria è completamente popolata di dispositivi.

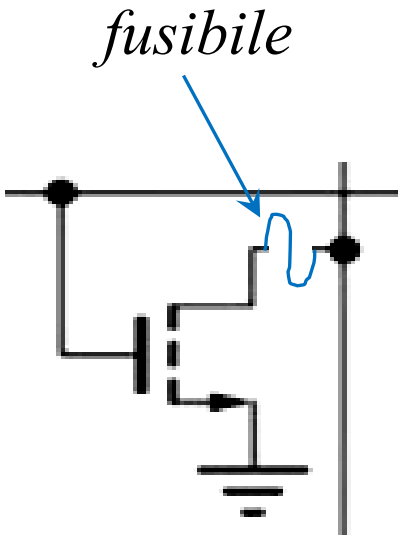
In serie ad ogni dispositivi c'è un microfusibile.

In fase di programmazione l'utente, applicando opportuni impulsi di tensione, può interrompere i fusibili nelle locazioni desiderate e personalizzare in questo modo la memoria.

Le PROM sono programmabili una sola volta.

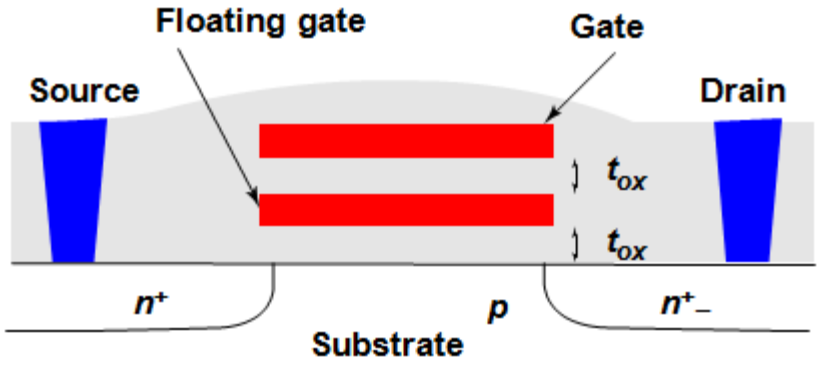
Struttura di una memoria

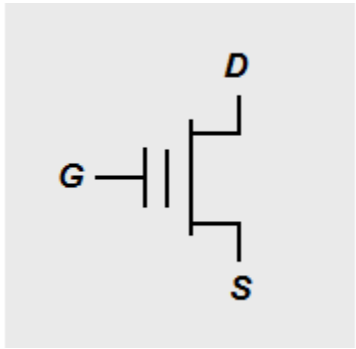




Memorie non-volatili

Le memorie EPROM, EEPROM e FLASH utilizzano dei dispositivi MOS particolari che includono una gate flottante.





Solo la control gate è accessibile dall'esterno. La gate flottante è invece isolata dall'ossido di silicio. Lo spessore dell'ossido che separa la floating gate del silicio sottostante è di pochi nanometri.

Memorie non-volatili

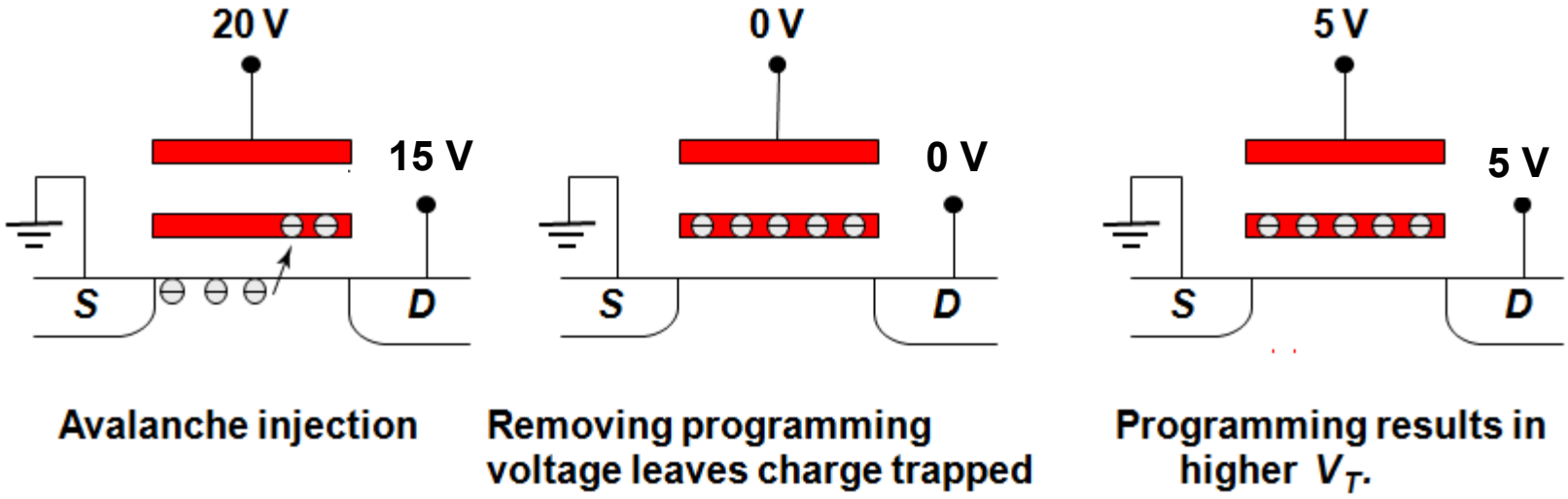
Il dispositivo si comporta come un MOS standard, caratterizzato da una certa tensione di soglia.
Il valore della tensione di soglia può tuttavia essere modificato, effettuando una opportuna "programmazione " del dispositivo.
A tale scopo, gate e drain vengono collegati ad una tensione positiva, di valore superiore a quella di normale funzionamento del dispositivo.

Memorie non-volatili

Si innesca un fenomeno denominato “iniezione a valanga” a causa del quale alcuni elettroni che fluiscono nel canale vengono intrappolati nella gate flottante.
La gate flottante si carica negativamente.
Al termine del processo, il dispositivo esibisce una tensione di soglia più alta di quella iniziale.

La carica immagazzinata nella gate flottante resta intrappolata per un tempo lunghissimo (decine di anni);

Memorie non-volatili



La carica immagazzinata nella gate flottante permane anche in assenza di tensione di alimentazione (memoria non-volatile) e resta intrappolata per un tempo lunghissimo (decine di anni).

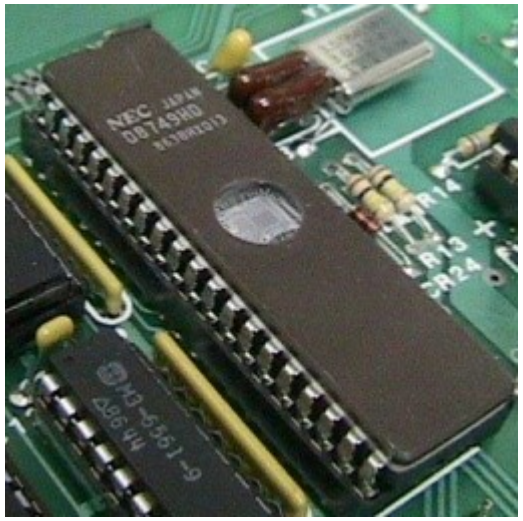
EPROM

Nelle EPROM: la matrice di memoria è completamente popolata di dispositivi a gate flottante.
In fase di programmazione, applicando le opportune tensioni, si innalzano le tensioni di soglia dei dispositivi nelle locazioni desiderate.
Poiché **la tensione di soglia dei dispositivi programmati è maggiore rispetto alla tensione di alimentazione**, tali dispositivi non entreranno mai in conduzione (è l'equivalente della interruzione del fusibile nelle PROM)

EPROM

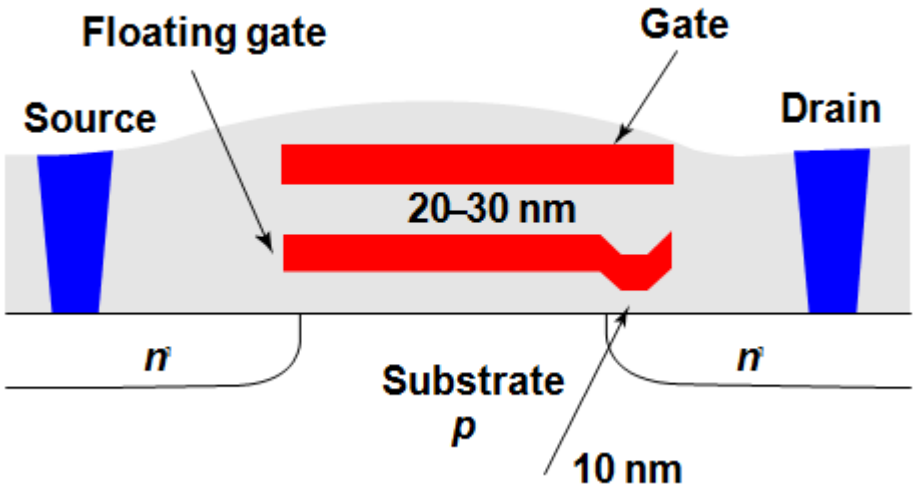
Le EPROM possono essere cancellate, esponendo il chip a radiazione ultravioletta che rende l'ossido di silicio debolmente conduttore disperdendo la carica immagazzinata nella gate flottante.

Le EPROM dispongono a tale scopo, di contenitori (packages) dotati di una finestra trasparente.

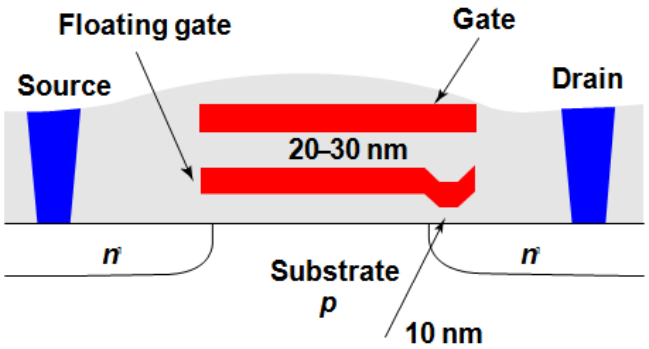


EEPROM

Le EEPROM sono cancellabili elettricamente. Utilizzano dispositivi a gate flottante in cui la gate isolata si estende in parte anche sopra la zona di drain.



EEPROM

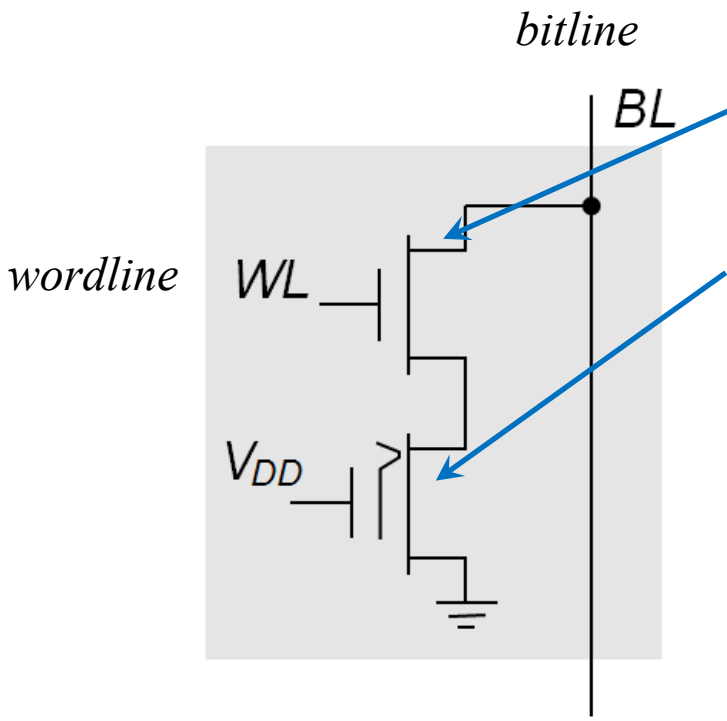


Il dispositivo viene programmato per effetto tunnel.
Applicando una tensione positiva fra gate e drain si immagazzinano elettroni sulla floating gate (**fase di programmazione**: aumento la tensione di soglia del dispositivo).
Applicando una tensione negativa fra gate e drain gli elettroni vengono espulsi dalla floating gate (**fase di cancellazione**: riduco la tensione di soglia del dispositivo)

EEPROM

La fase di cancellazione è critica: vorremmo idealmente riportare la tensione di soglia al suo valore originario, ma la V_T non è facilmente controllabile e può divenire negativa (sovracancellazione).
In questo modo il MOS si comporta come un dispositivo a svuotamento, che conduce anche quando la gate di controllo è a 0V e la memoria perde la sua funzionalità.
Per risolvere questo problema, **le celle di memoria EEPROM utilizzano due MOS in serie.**

EEPROM

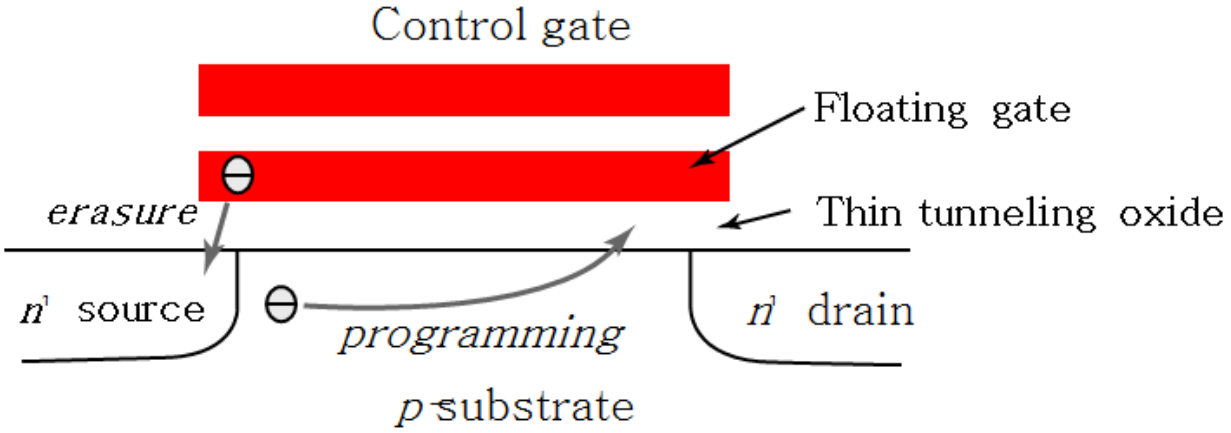


Il dispositivo superiore nello schema è un MOS standard, che porta la bitline a 0 se la wordline è attivata.

Il dispositivo a gate flottante si utilizza come una sorta di interruttore.
È sempre in conduzione ($V_T<0$) (approssima un interruttore ON) se il dispositivo è cancellato.
È sempre spento ($V_T>V_{DD}$) (approssima un interruttore OFF) se il dispositivo è programmato.

FLASH

Le memorie flash superano le limitazioni delle EEPROM, utilizzando un solo MOS per cella (invece di due).



La fase di programmazione avviene come nelle EPROM per iniezione a valanga (più veloce).
La fase di cancellazione utilizza l'effetto tunnel (in questo esempio dalla parte del source.

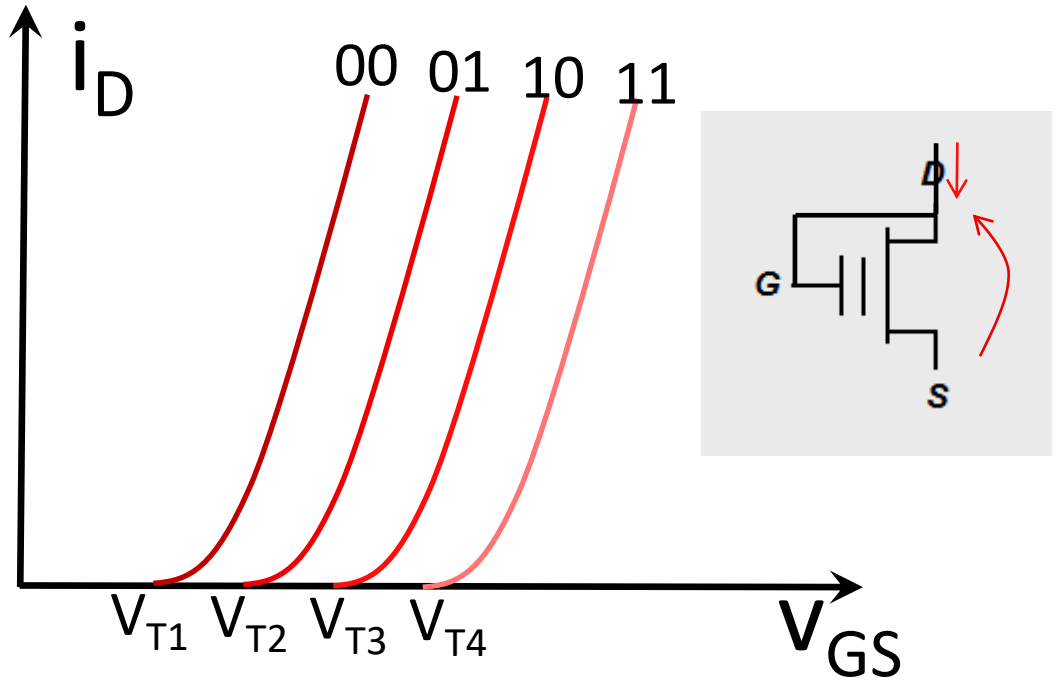
FLASH

Per risolvere il problema della sovracancellazione, la circuiteria di controllo interna al chip verifica la tensione di soglia durante la procedura di cancellazione, regolandone dinamicamente la durata.
Si ottiene in questo modo un controllo accurato della tensione di soglia, a prezzo di una complessa circuiteria aggiuntiva.
Questa tecnica *è efficiente solo se svolta parallelamente su un elevato numero di celle*.
In pratica, la memoria è suddivisa in blocchi. Se si deve modificare qualche locazione di memoria in un blocco, l'intero blocco viene cancellato e tutte le sue locazioni vengono riscritte.

FLASH Multilivello

Le memorie FLASH richiedono un solo MOS per ogni cella di memoria. Le FLASH multilivello sono in grado di immagazzinare più bit per ogni cella di memoria.

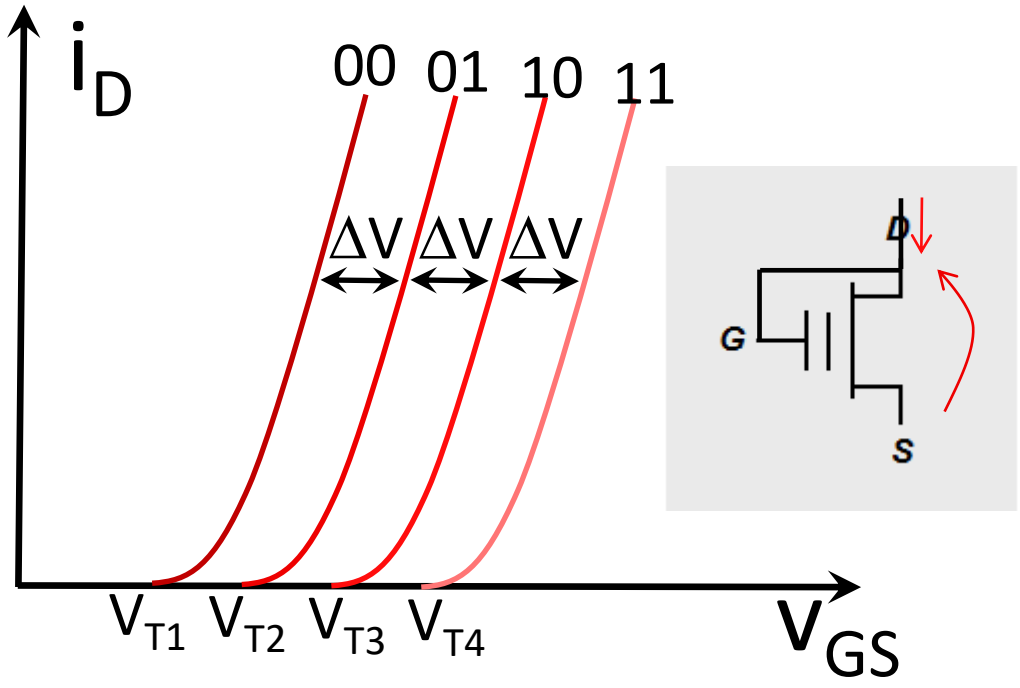
L'informazione è codificata nel valore della tensione di soglia dei dispositivi.
Se il dispositivo ha la tensione di soglia V_{T1}, il dato memorizzato è "00"; per V_{T2}, il dato è "01" ecc.



FLASH Multilivello

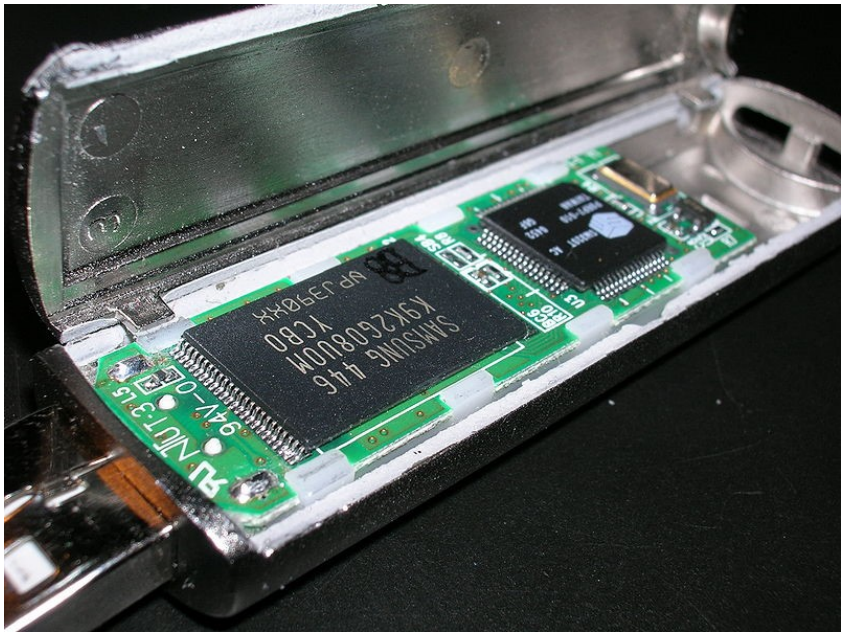
In fase di scrittura si deve poter controllare il valore della tensione di soglia con un certa precisione ΔV.
Analogamente, in fase di lettura si deve essere in grado di discriminare la V_T del transistor con la stessa precisione.

Il numero di livelli cresce esponenzialmente con il numero di bit della cella; ΔV decresce esponenzialmente con il numero di bit della cella, rendendo lettura e scrittura sempre più complesse.



FLASH

Le memorie FLASH, grazie alla loro densità, sono sempre più utilizzate nell'elettronica di consumo



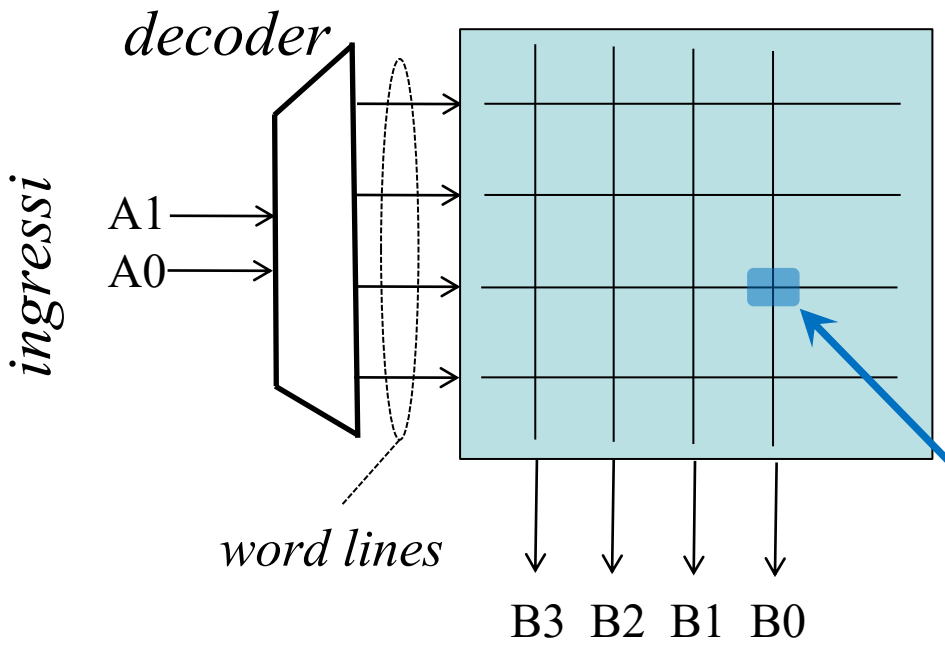
Da notare che per ogni cella di una FLASH si può realizzare un numero finito di operazioni di cancellazione/programmazione, dopo il quale la cella si logora. Il numero di cicli è tipicamente compreso fra 10^5 e 10^6

Memorie SRAM

SRAM

Le memorie SRAM sono le più veloci (sono utilizzate, ad esempio, nella cache dei processori).
Sono in grado di memorizzare l'informazione fin quando sono alimentate.
Utilizzano 6 oppure 4 MOS per cella (sono costose, richiedendo un'elevata area di silicio)

SRAM

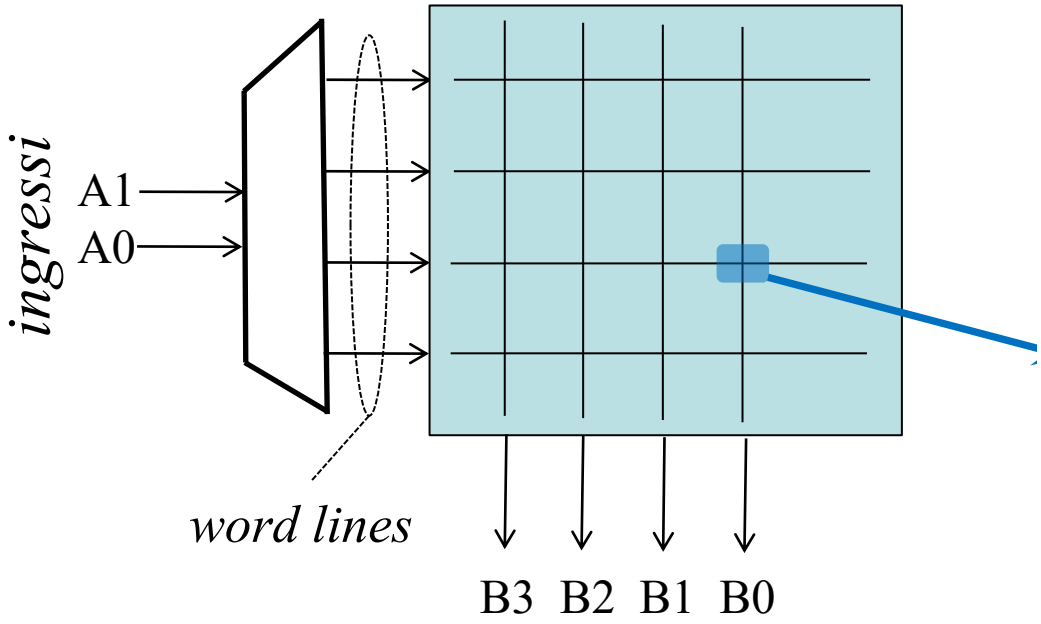


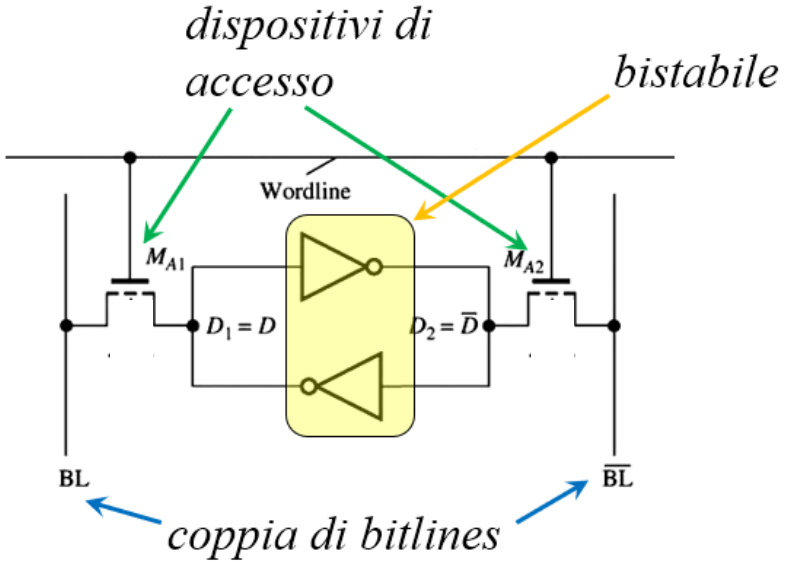
Abbiamo già visto la struttura di principio di una memoria. Nelle SRAM è presente anche una circuiteria apposita per **controllare l'operazione da effettuare** (se lettura o scrittura). Sono presenti anche degli opportuni **amplificatori di lettura.**

Ci focalizziamo solo sulla **cella elementare**, che memorizza il singolo bit. Da notare che nelle SRAM **c'è una coppia di bitlines** per ogni colonna della memoria.

SRAM

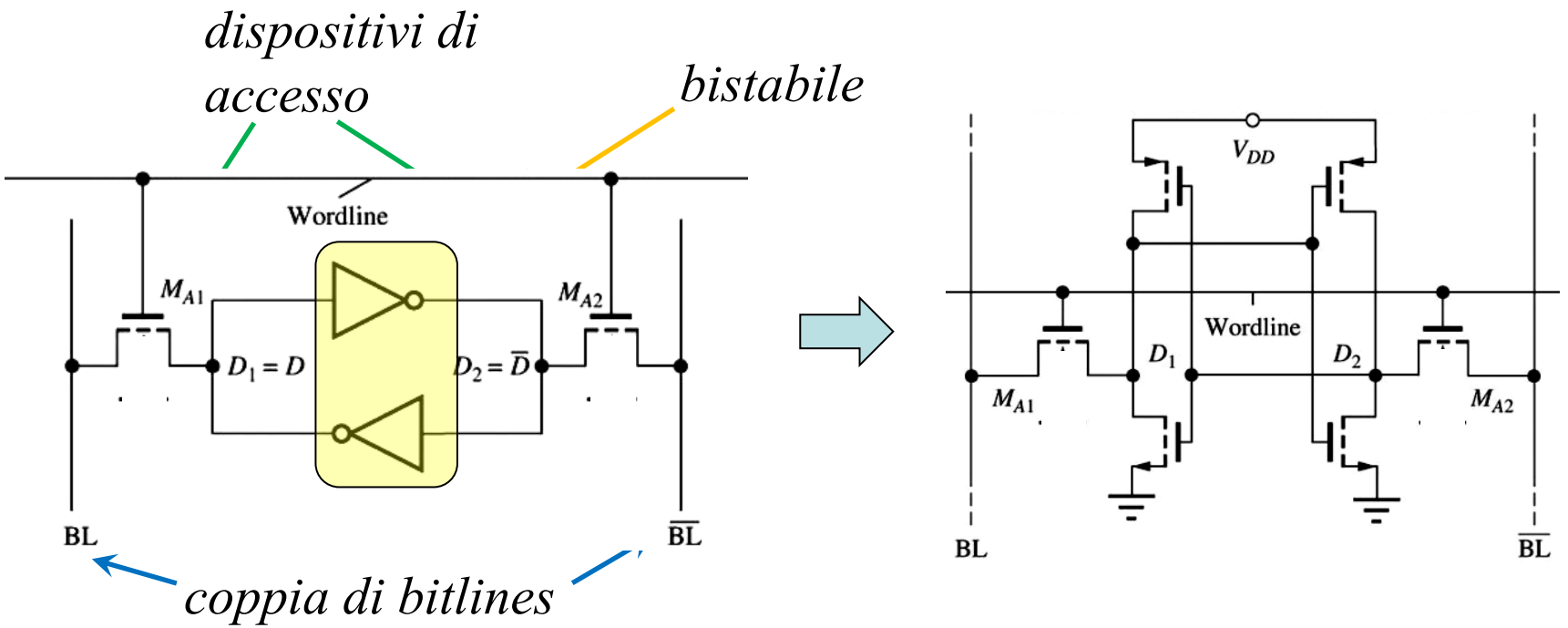
La cella SRAM è costituita da un **bistabile**, cui si aggiungono **due dispositivi di accesso**, grazie ai quali si può leggere o modificare il contenuto della cella. Da notare che nelle SRAM **c'è una coppia di bitlines** per ogni colonna della memoria.





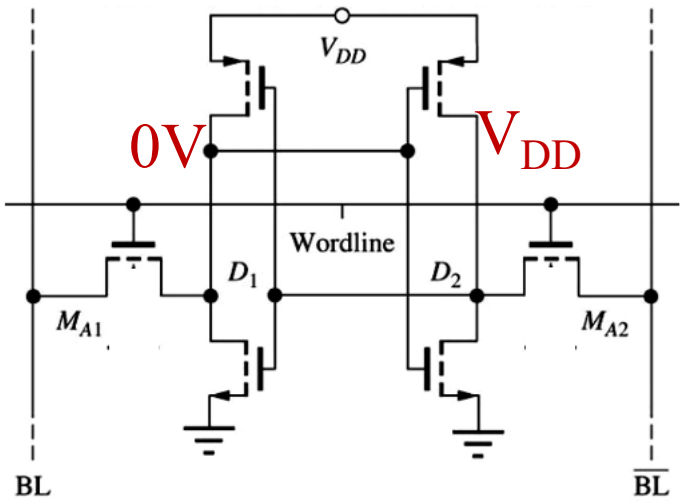
SRAM 6T

Nella cella SRAM 6T il bistabile è composto da due invertitori CMOS.

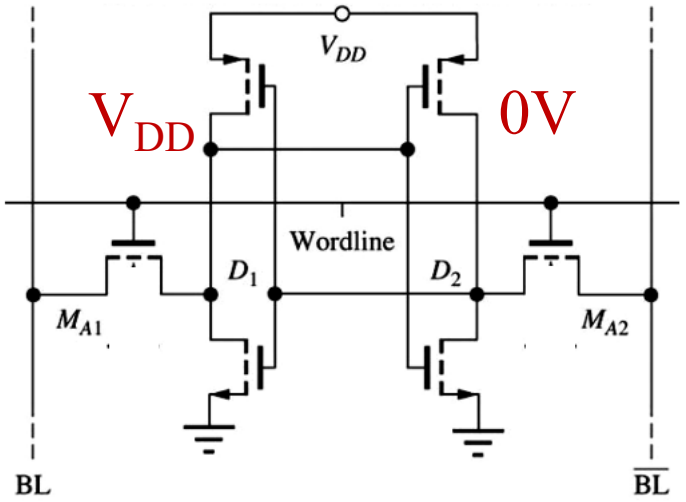


SRAM 6T

Diremo convenzionalmente che la cella memorizza uno “0” se nodo D_1 è a livello logico basso e D_2 a livello alto.
Nell’altro caso, la cella memorizza un “1”



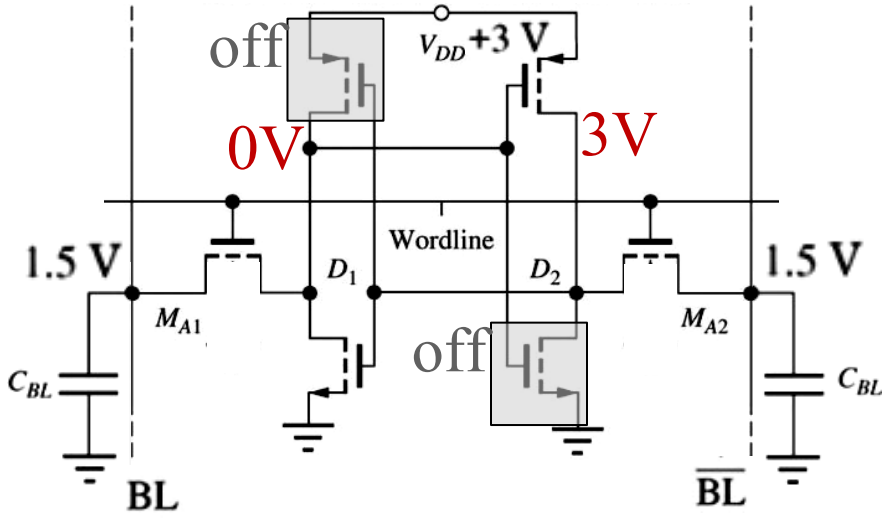
stato “0”



stato “1”

Fase di lettura

Mediante un circuito ausiliario, le capacità parassite delle due bitlines vengono **precaricate** ad un valore di tensione prefissato, ad esempio $V_{DD}/2$. In questa fase la wordline è a zero, in modo da non disturbare la cella di memoria.

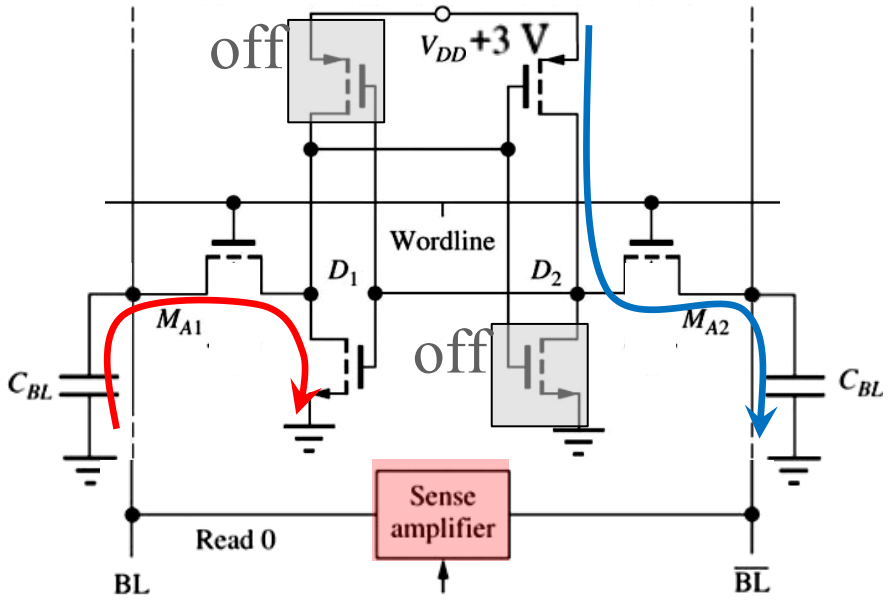


Nell'esempio, supponiamo che nella cella sia memorizzato uno "0". Notare i dispositivi interdetti.

Operazione di lettura di una cella 6T

Completata la precarica, si attiva la wordline.

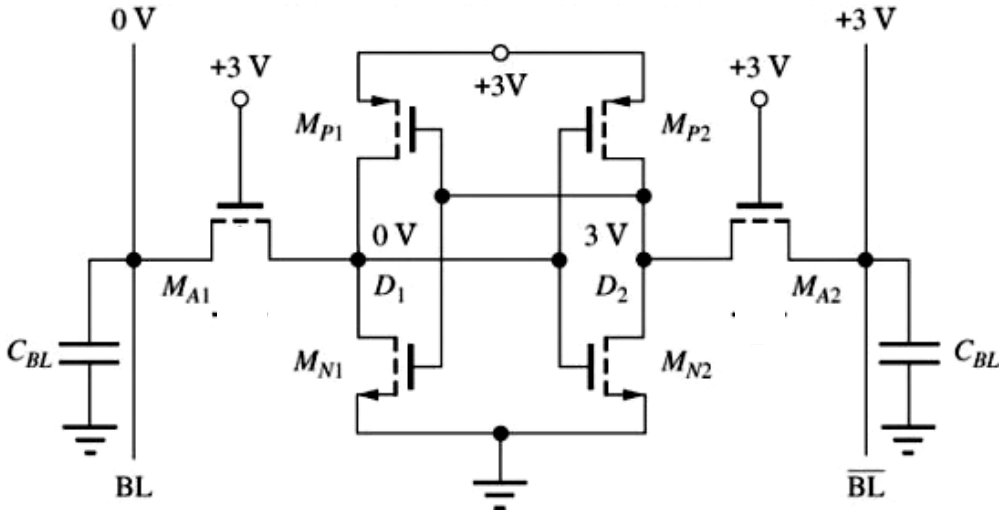
La capacità sulla bitline di sinistra si scarica a 0V, mentre quella a destra si carica a Vdd.



I potenziali sulle due bitline si sbilanciano.
La differenza di potenziale è amplificata da un amplificatore di lettura che fornisce l'uscita.

Operazione di lettura di una cella 6T

Situazione a regime,
completata la lettura



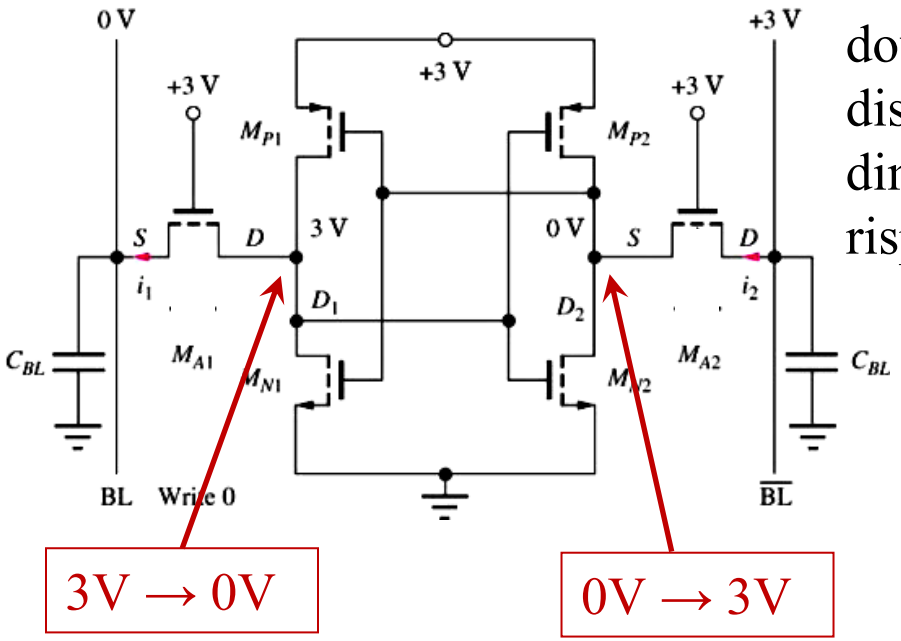
Operazione di scrittura nella cella 6T

Per scrivere un dato nella cella di memoria, pilotiamo BL con il dato stesso e BL con il suo complemento.

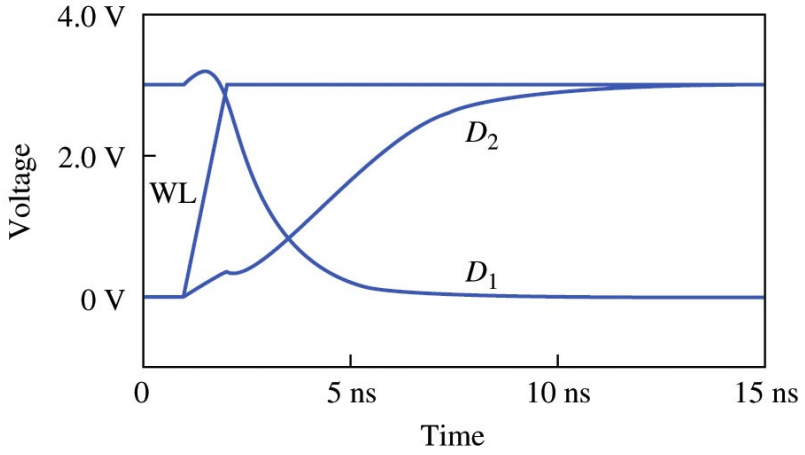
Quindi si attiva la wordline.

Operazione di scrittura nella cella 6T

Esempio, scrittura di uno “0” in una cella in cui è memorizzato “1”

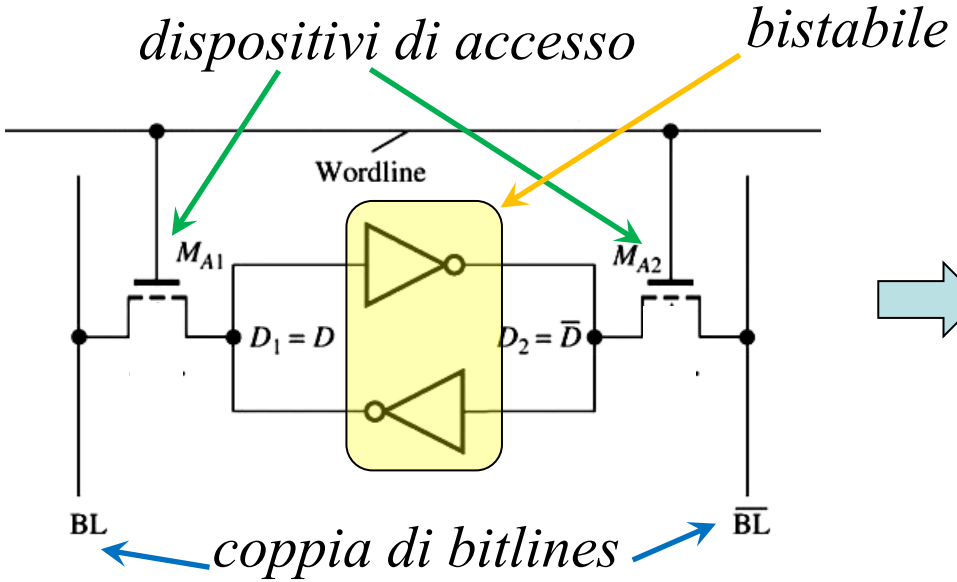


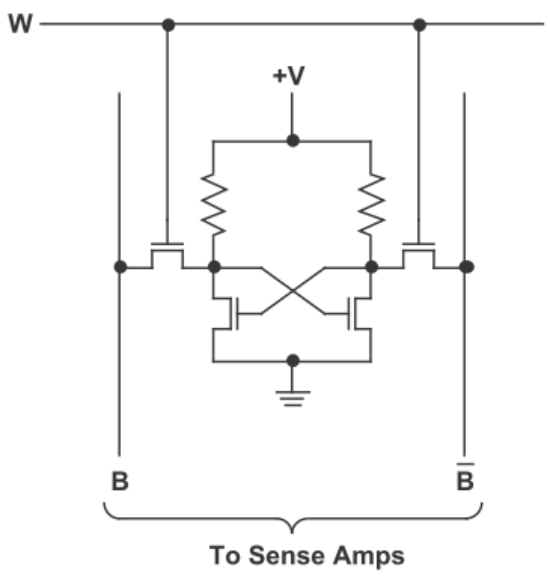
Il nodo D_1 dovrà portarsi a 0, mentre D_2 dovrà portarsi ad “1”. A tal fine, i dispositivi di accesso devono essere dimensionati in modo da “prevalere” rispetto ai MOS del bistabile.



SRAM 4T

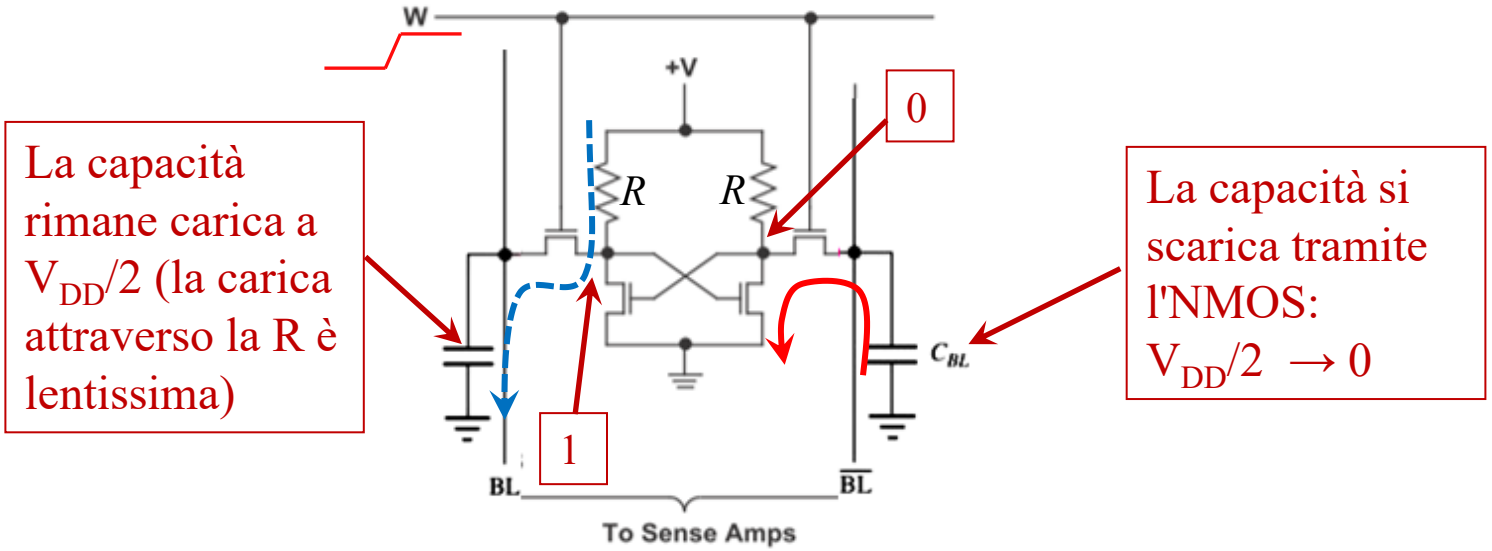
Il bistabile è composto da due invertitori con carico resistivo, cui si aggiungono due dispositivi di accesso. Le resistenze sono di valore estremamente elevato ($10^{12}\Omega$) per ridurre la dissipazione di potenza. Si adottano tecnologie particolari per realizzare resistenze così elevate con una minima occupazione di area.





SRAM 4T – operazioni di lettura e scrittura

Le fasi di lettura e scrittura sono analoghe a quanto visto per la memoria 6T.
Per quanto riguarda la lettura, dopo la precarica delle bitlines e l'attivazione della wordline, solo la bitline collegata al ramo del bistabile a libello basso modifica il suo potenziale scaricandosi. L'altra capacità rimane carica a $V_{DD}/2$ (a causa del valore enorme della R, la carica è infatti lentissima).
L'amplificatore di lettura è comunque in grado di individuare la differenza di potenziale fra le due bitlines, fornendo in uscita il dato memorizzato nella cella.



Memorie DRAM

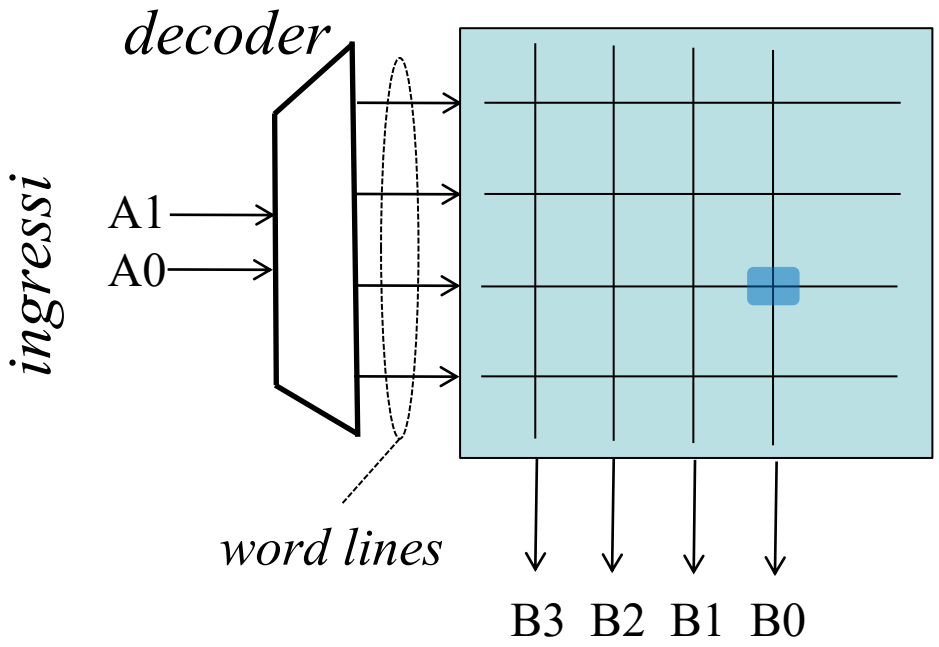
DRAM

Le memorie dinamiche, DRAM, sono le più compatte (sono le memorie principali dei processori).

Sono in grado di memorizzare l'informazione solo per un ridotto intervallo di tempo (qualche millisecondo); richiedono poi una periodica operazione di *refresh* per evitare che le informazioni memorizzate vengano perse.

Utilizzano 1 MOS ed 1 Capacità per cella

DRAM

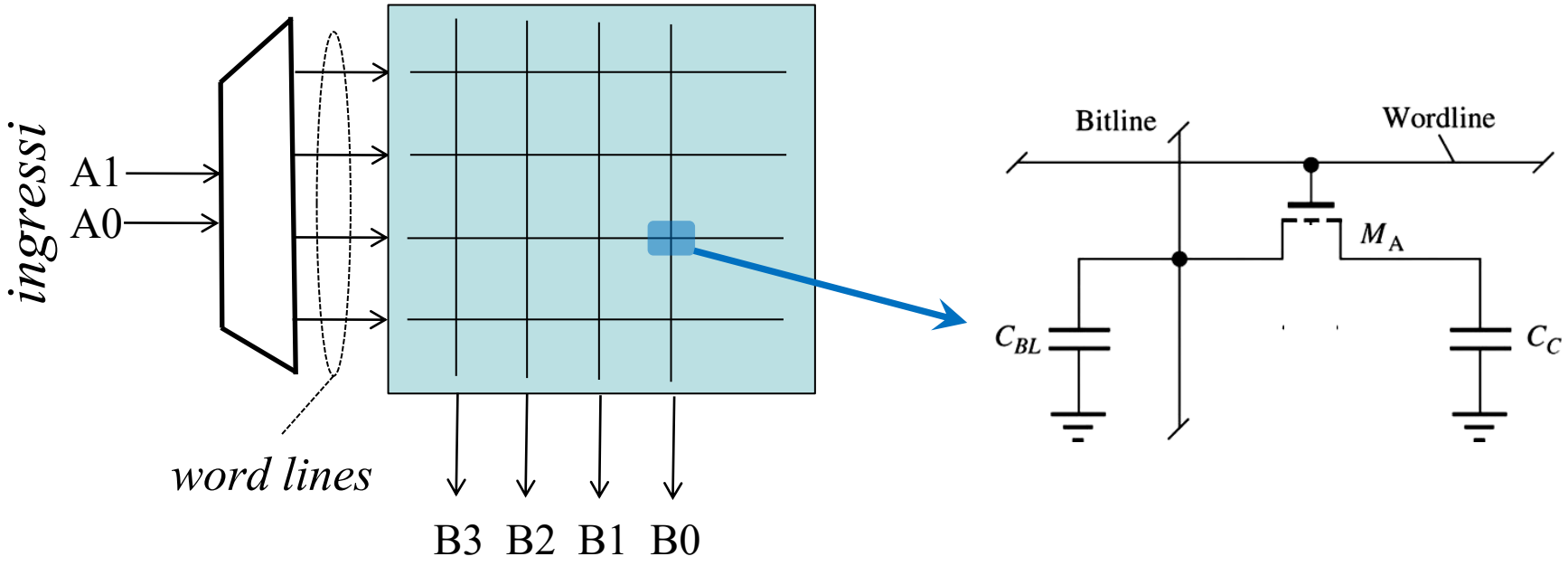


Anche in questo caso ci focalizziamo solo sulla **cella elementare**, che memorizza il singolo bit.
Da notare che nelle DRAM **c'è una singola bitline** per ogni colonna della memoria.

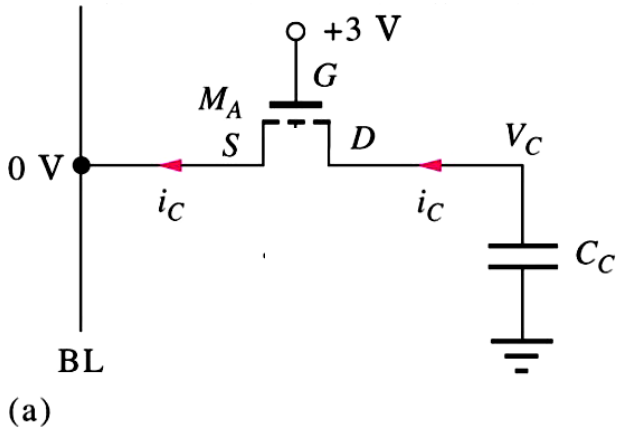
Nelle DRAM sono presenti i circuiti di lettura/scrittura; **l'amplificatore di lettura** ha un ruolo fondamentale. È presente anche un circuito per il *refresh* dei dati memorizzati, come vedremo fra breve,

Cella di memoria dinamica

La cella di memoria dinamica utilizza un solo transistor (1T), oltre ad una capacità; il dato è rappresentato come presenza o assenza di carica sulla capacità
A causa delle correnti di perdita di M_A, il dato può essere memorizzato solo per un tempo relativamente breve, trascorso il quale è necessario effettuare un'operazione di *refresh*

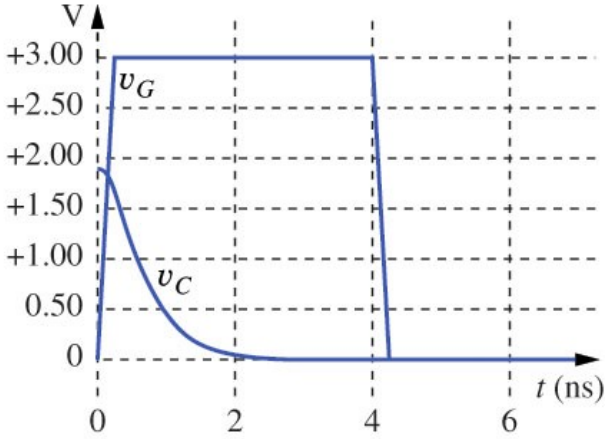


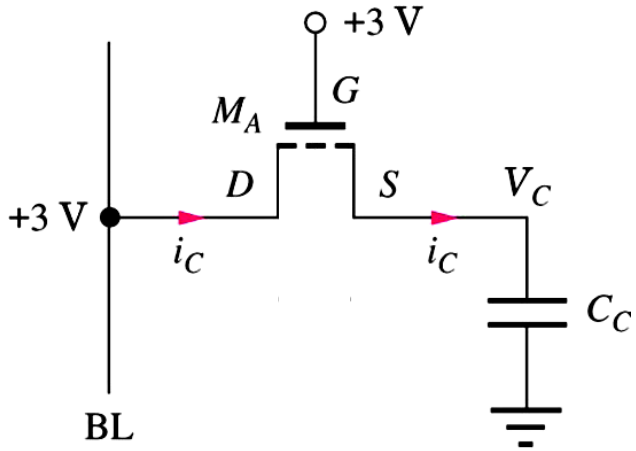
Memorizzazione di un dato nella cella 1T



(a)

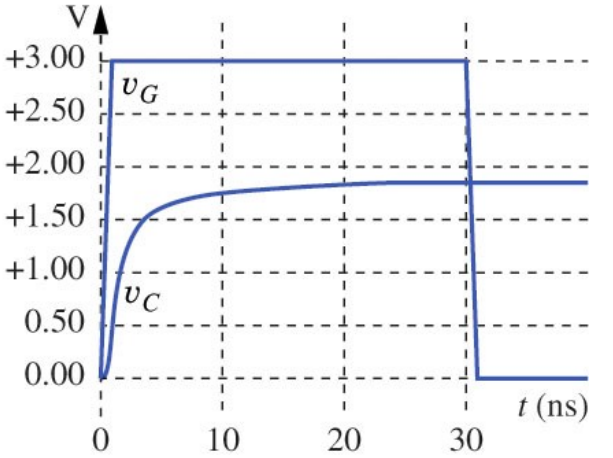
Memorizzo“0”





(a)

Memorizzo “1”



Memorizzazione di un dato nella cella 1T

Un livello logico basso corrisponde ad una tensione di 0V sulla capacità.

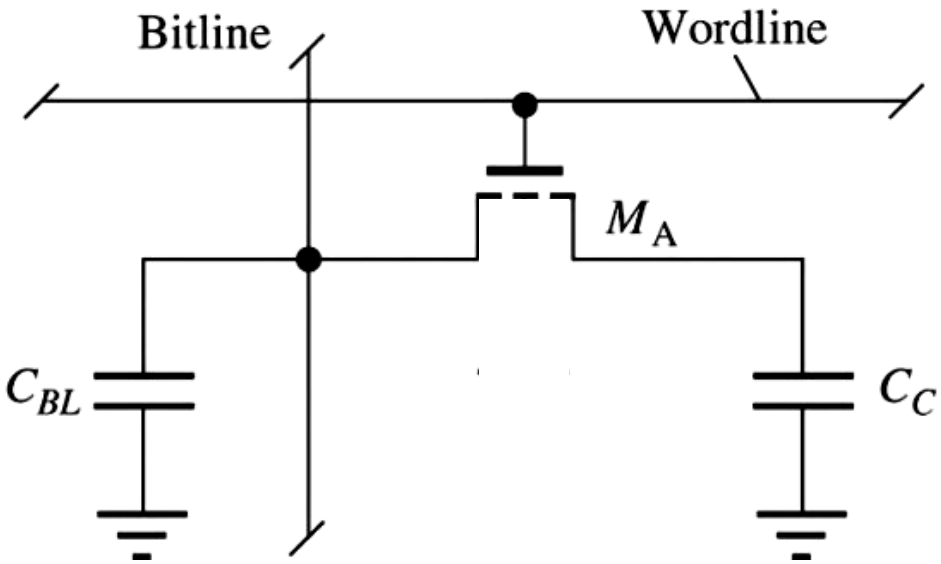
Un livello logico alto corrisponde ad immagazzinare sulla capacità una tensione di V_{DD}-V_T (perdita della soglia).

Boosted Wordline

Spesso la wordline viene pilotata ad una tensione maggiore rispetto a quella di alimentazione; in questo modo si risolve il problema della perdita della soglia.

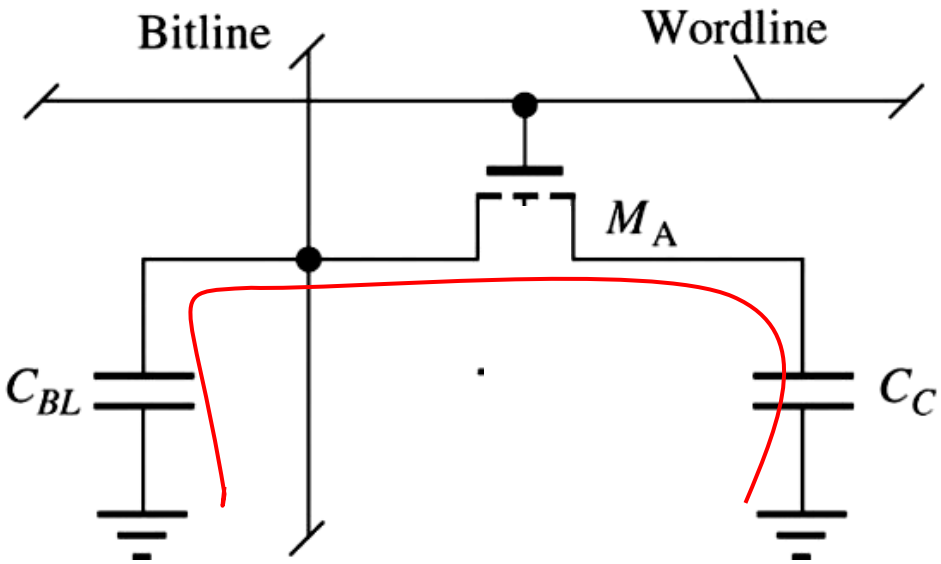
Lettura nella DRAM 1T

La capacità parassita della Bitline viene dapprima precaricata ad una tensione V_{BL} (ad esempio, $V_{BL}=V_{DD}/2$). Viene quindi attivata la wordline



Lettura nella DRAM 1T

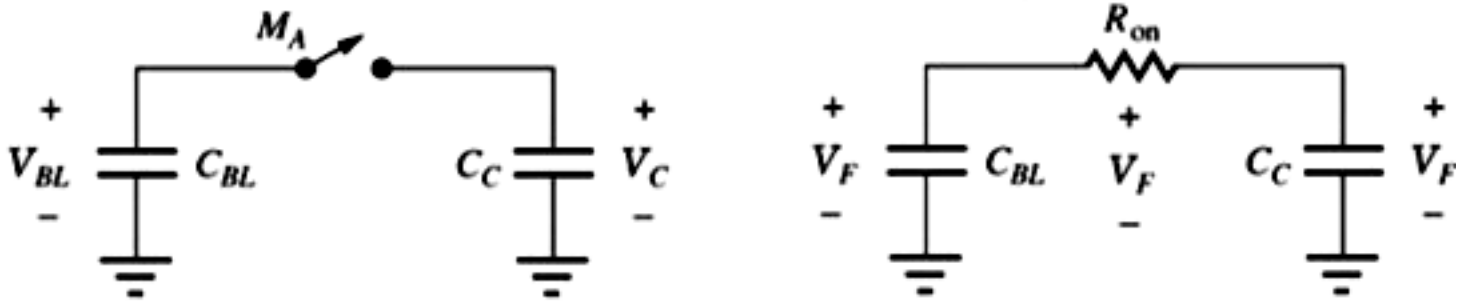
Quando la wordline è attivata, le due capacità C_{BL} e C_C sono collegate fra di loro e si ha un fenomeno di *ridistribuzione di carica*



Lettura nella DRAM 1T

Indichiamo con V_{BL} e V_C le tensioni prima dell'attivazione della wordline.

La tensione dopo l'attivazione della wordline sarà V_F.

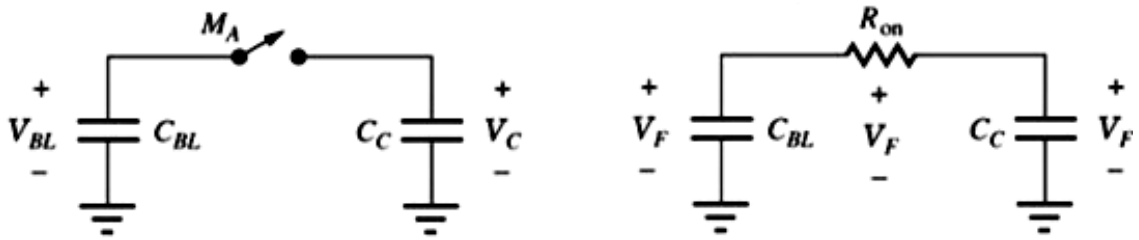


Carica iniziale nel sistema = Carica finale

$$V_{BL}C_{BL} + V_C C_C = V_F C_{BL} + V_F C_C$$

$$V_F = \frac{V_{BL}C_{BL} + V_C C_C}{C_{BL} + C_C}$$

Lettura nella DRAM 1T



$$V_F = \frac{V_{BL}C_{BL} + V_C C_C}{C_{BL} + C_C}$$

La variazione di tensione sulla bitline è:

$$\Delta V_{BL} = V_F - V_{BL} = \frac{V_{BL}C_{BL} + V_C C_C}{C_{BL} + C_C} - V_{BL} =$$

$$= \frac{C_C}{C_{BL} + C_C}\left(V_C - V_{BL}\right)$$

Lettura nella DRAM 1T

Sfortunatamente, la capacità parassita della bitline è molto maggiore della capacità della cella e si ha:

$$\Delta V_{BL} = \frac{C_C}{C_{BL} + C_C}\left(V_C - V_{BL}\right) \simeq \frac{C_C}{C_{BL}}\left(V_C - V_{BL}\right)$$

Il valore di ΔV_{BL} è di qualche decina di mV.
Il segno di ΔV_{BL} consente di individuare se $V_C > V_{BL}$ (il dato in memoria è "1") oppure se $V_C < V_{BL}$ (il dato in memoria è "0").
Un amplificatore di lettura amplifica ΔV_{BL} e fornisce il dato in uscita.

Lettura nella DRAM 1T

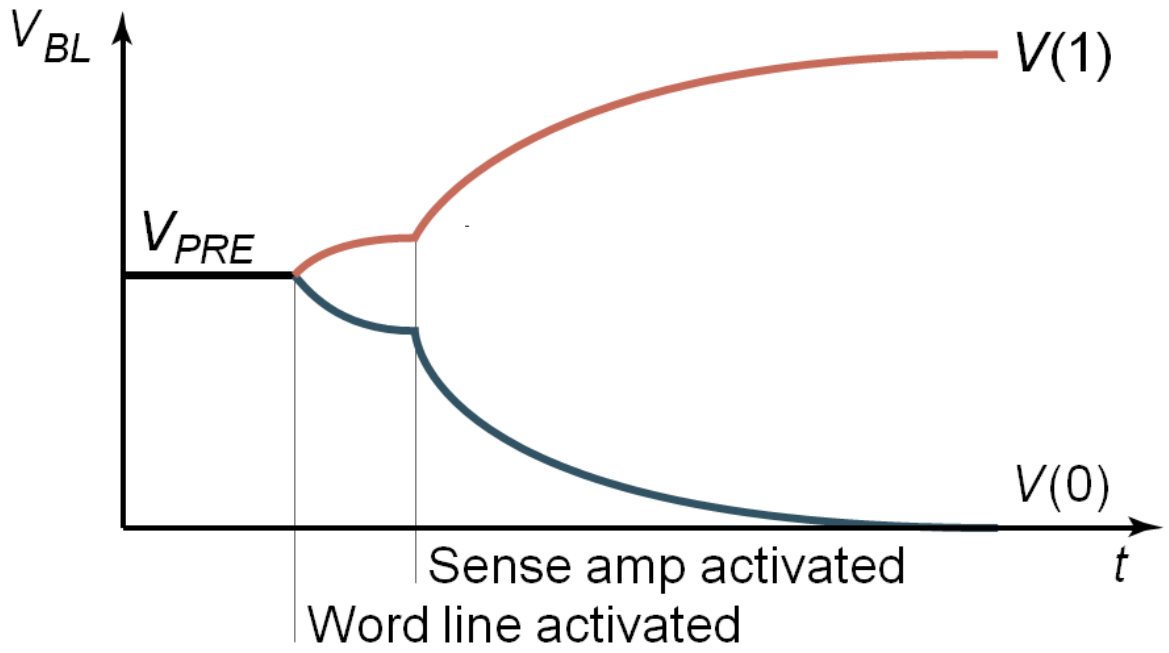
La lettura è distruttiva. La tensione sulla capacità dopo l'attivazione della wordline è infatti:

$$V_F = \frac{V_{BL}C_{BL} + V_C C_C}{C_{BL} + C_C} \simeq V_{BL}$$

Dunque, indipendentemente dal valore immagazzinato nella capacità, dopo l'attivazione della wordline la capacità è carica ad un potenziale è pari alla tensione di precarica.

Amplificatore di Lettura

La presenza dell'amplificatore di lettura è fondamentale nelle DRAM. La tensione sulla bitline viene infatti riportata a 0V oppure a V_{DD} dall'amplificatore di lettura e ciò ripristina la carica sulla capacità.



Refresh

La lettura, grazie alla presenza dell'amplificatore, consente in effetti di ripristinare la carica nella capacità.

L'operazione di refresh deve essere effettuata anche quando non è richiesta la lettura di nessun dato, per evitare che le correnti di perdita alterino la carica immagazzinata nelle capacità.