UNIVERSITÀ
DEGLI STUDI
DI PADOVA

DIPARTIMENTO
DI INGEGNERIA
DELL'INFORMAZIONE

DIPARTIMENTO DI INGEGNERIA DELL'INFORMAZIONE

CORSO DI LAUREA IN INGEGNERIA INFORMATICA

"CARATTERISTICHE E STRUTTURE DELLE MEMORIE A SEMICONDUTTORE"

Relatore:
Dott. Gaudenzio Meneghesso

Laureando:
Giuseppe Labate
Matricola: 1226050

ANNO ACCADEMICO 2021 – 2022

Data di laurea 25/11/2022

Alla mia famiglia che ha sempre creduto in me.

Ma, soprattutto, a mio nonno.

SOMMARIO

In questo documento verranno esaminate le diverse tipologie di memorie a semiconduttore, partendo dall'elemento base, il transistor MOS, e andando ad approfondire tutte le diverse memorie sviluppate negli anni, considerando particolarità, vantaggi e svantaggi per ognuna di esse.

Le memorie trattate di seguito saranno illustrate dal punto di vista elettrico, mostrando ed analizzando le operazioni di lettura, scrittura e cancellazione, ove possibili, a livello circuitale.

INDICE

Capitolo 1

Transistor MOSFET

1.1 Definizione

Il transistor MOSFET (**M**etal-**O**xide-**S**emiconductor **F**ield-**E**ffect **T**ransistor) è, al giorno d'oggi, il transistor più comune nei circuiti digitali.
Ideato da Lilienfeld nel 1925 e realizzato da M. Atalla e Dawon Kahng dei Bell Labs nel 1959, è composto da un substrato di materiale drogato, comunemente il silicio, a cui vengono collegati tre terminali: gate, source e drain.

1.2 Struttura

Il gate è realizzato da un materiale conduttore poiché, nel processo di produzione, non esiste una tecnologia per allineare il gate metallico al resto del transistor.
Tra gate e substrato è presente un sottile strato di biossido di silicio, detto ossido di gate, che ha la funzione di ridurre la potenza dissipata, isolando il gate in modo che non perda carica.
I terminali di source e drain sono anch'essi composti da un semiconduttore drogato però in maniera opposta: se il substrato ha un drogaggio di tipo p i due terminali avranno un drogaggio di tipo n e viceversa.
A seconda che il tipo di drogaggio del substrato sia p o n, il MOSFET prende il nome rispettivamente di nMOS (*Figura 1.1 a*) e pMOS (*Figura1.1 b*), i quali hanno un comportamento speculare.

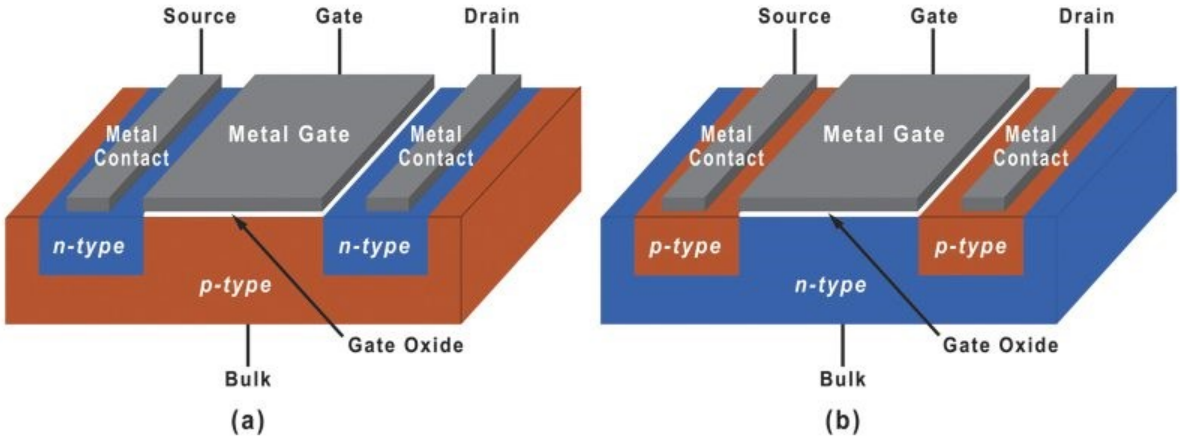

Figura 1.1: nMOS(a) e pMOS(b) a confronto

1.3 Funzionamento

1.3.1 Accensione

La particolarità del transistor MOS risiede nelle sue regioni di funzionamento, le quali dipendono dalla tensione applicata ai capi del substrato sotto al gate (V_{GS}) e dalla tensione tra drain e source (V_{DS}).

Quando un nMOS passa alla fase di funzionamento attivo, le lacune al di sotto del gate si allontanano e gli elettroni del substrato, detti portatori maggioritari, si avvinano al gate.

Si viene quindi a formare una sottile regione, detta di inversione, composta da portatori di carica liberi n.

In questa zona, il semiconduttore si comporta come se fosse drogato n, permettendo il passaggio di corrente tra source e drain, poiché anch'essi con lo stesso drogaggio.

1.3.2 Tensione di soglia

Perché l'nMOS risulti acceso, la tensione tra gate e source deve superare quella di soglia (V_{th})

$$V_{th} \leq V_{GS}$$

Dove V_{th} , che dipende da V_{SB} per effetto body, per un nMOS vale:

$$V_{th} = V_{th0} + \gamma\left(\sqrt{V_{SB} + |2\varphi_F|} - \sqrt{|2\varphi_F|}\right).$$

Ciò accade perché se vi è una differenza di tensione tra source e body, per ottenere la regione di inversione è necessaria una maggiore differenza di potenziale, il che equivale ad un aumento della tensione di soglia del transistor.

Se si definisce pertanto la tensione di soglia senza considerare l'effetto body, nel canale risulta una carica indotta minore di quella aspettata, e questo comporta un errore in eccesso nella valutazione della corrente del canale.

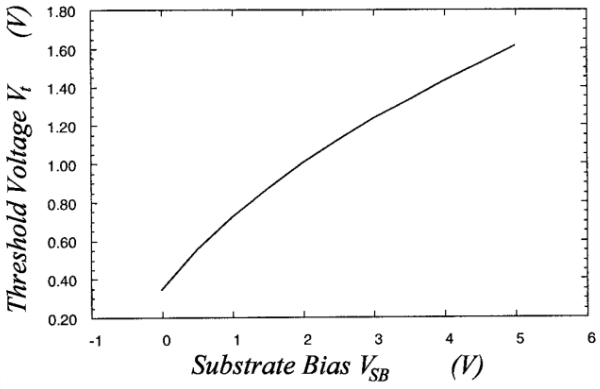

Figura 1.3.1: Rappresentazione della dipendenza tra V_{th} e V_{SB}

Una volta in accensione, esistono due possibili regioni di funzionamento: la regione lineare e la regione di saturazione.

1.3.3 Regione di saturazione

All'aumentare di V_{DS}, la tensione $V_{GD} = V_{GS} - V_{DS}$ diminuisce sempre di più, provocando una strozzatura del canale in prossimità del drain (pinch-off).

Il pinch-off si verifica nel punto $L' < L$ (lunghezza di canale), in cui il potenziale vale $V_{GS} - V_{th}$.

Per questo motivo, un nMOS è in zona di saturazione quando

$$V_{DS} \geq V_{GS} - V_{th}$$

Un altro aspetto da non trascurare è l'indipendenza da V_{DS} della I_D (corrente che va da drain a source), dovuta per l'appunto al pinch-off.

$$I_D = \frac{\mu_n C_{ox}}{2} \frac{W}{L} (V_{GS} - V_{th})^2$$

Con μ_n mobilità dei portatori di carica, C_{ox} capacità parassita dell'ossido di gate per unità di area e W e L larghezza e lunghezza del gate.

Considerando l'effetto di modulazione di lunghezza di canale, si ha una dipendenza lineare tra I_D e V_{DS}. Questa linearità è dovuta alla diminuzione della lunghezza di canale, rappresentata con λ coefficiente, detto fattore di pinch-off.

$$I_D = \frac{\mu_n C_{ox}}{2} \frac{W}{L} (V_{GS} - V_{th})^2 (1 + \lambda V_{DS})$$

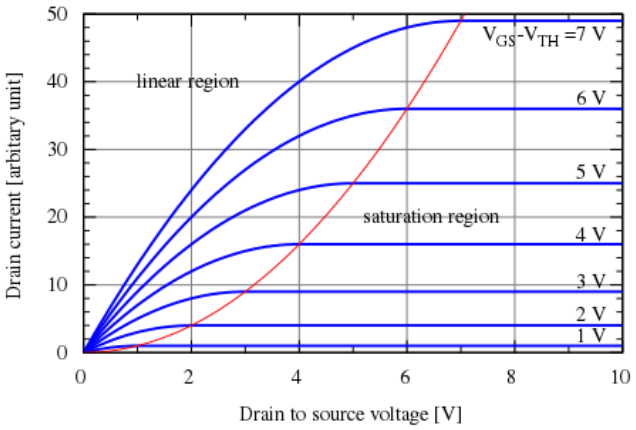

Figura 1.3.2: Grafico della corrente di un nMOS in funzione della V_{DS} (senza modulazione di canale)

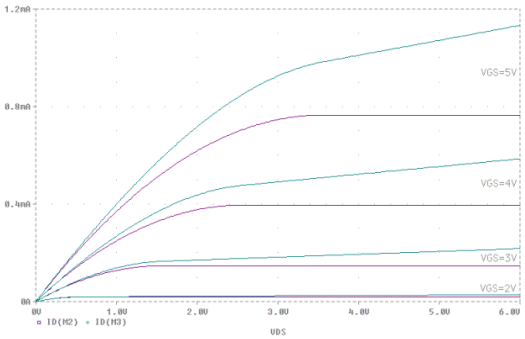

Figura 1.3.3: Confronto tra I_D con modulazione di canale (in blu) e I_D senza modulazione di canale (in rosso)

1.3.4 Regione lineare o di triodo

Un nMOS viene quindi osservato in zona lineare quando è acceso e

$$V_{DS} \leq V_{GS} - V_{th}$$

In questa regione il canale riesce ad allungarsi fino ad arrivare a raggiungere il drain.

Non è più presente il pinch-off e il canale risulta quindi omogeneo.

La corrente dipenderà ora quadraticamente da V_{DS}

$$I_D = \mu_n C_{ox} \frac{W}{L}\left((V_{GS} - V_{th})V_{DS} - \frac{{V_{DS}}^2}{2}\right)$$

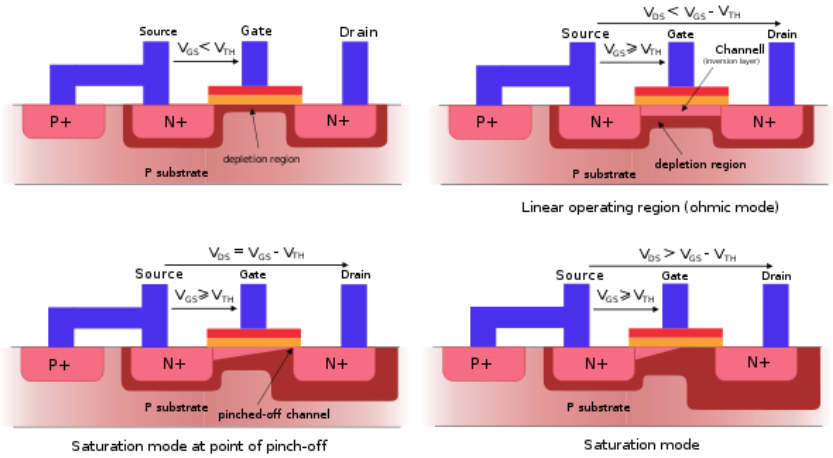

Figura 1.3.4: Canale di inversione nelle varie zone di funzionamento

1.4 Invertitori in retroazione positiva e logica CMOS

Per invertitore logico si intende una porta in logica CMOS (complementary mos) che, dato un ingresso V_{in}, avrà in uscita l'ingresso stesso negato $V_{out} = \overline{V_{in}}$.

È composto da un pMOS con source collegato a V_{DD} e drain in comune al drain dell'nMOS, che avrà il source a massa.

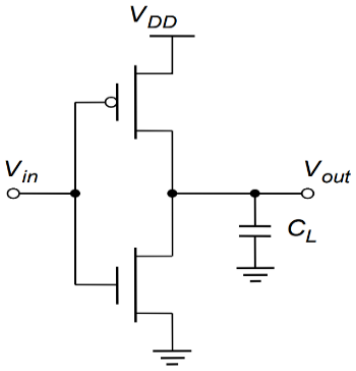

Figura 1.4.1: Inverter CMOS

La particolarità di questa porta sta nel fatto che può acquisire più stati, a seconda della tensione posta in ingresso.

Se essa, infatti, è compresa tra $0\ V$ e V_{th}, l'nMOS sarà spento in quanto $V_{GSn} \leq V_{thn}$ e il pMOS acceso in zona lineare, poiché $|V_{GSp}| \geq |V_{thp}|$ e $|V_{DSp}| \leq |V_{GSp} - V_{thp}|$.

Analogamente per $V_{DD} - |V_{tp}| \leq V_{in} \leq V_{DD}$, si avrà l nMOS in zona lineare e pMOS spento.

In questi due stati, l'invertitore mantiene il valore logico senza dissipare corrente.

Nell'invertitore CMOS, infatti, la corrente di un transistor è limitata da quella dell'altro, perché in ogni caso la corrente che scorre deve essere sempre la stessa ($I_{Dn} = I_{Dp}$).

Perciò se uno dei due transistor risulta spento e quindi con $I = 0\ A$, la corrente dell'intero invertitore sarà nulla, anche se il secondo transistor è acceso.

Ci sono poi altri due stati speculari in cui il pMOS è in zona lineare e l'nMOS in saturazione e viceversa.

In questi stati, grazie alla robustezza dell'invertitore, pur non avendo un valore logico pieno, lo si assume come tale, approssimandolo se cade all'interno di un certo intervallo, noto come margine a rumore.

Infine, quando sia il pMOS che l'nMOS si trovano in regione di saturazione, la corrente, e quindi anche la potenza dissipata, è massima.

In questa regione, inoltre, la V_{in} coincide o cade nell'intorno di V_M, detta tensione di soglia logica, per la quale si ha il massimo grado di indeterminazione.

Quando la tensione d'ingresso si avvicina a V_M, infatti, non si può più approssimare la V_{out} ad uno dei due valori logici ('0' o '1'), in quanto equidistante da entrambi.

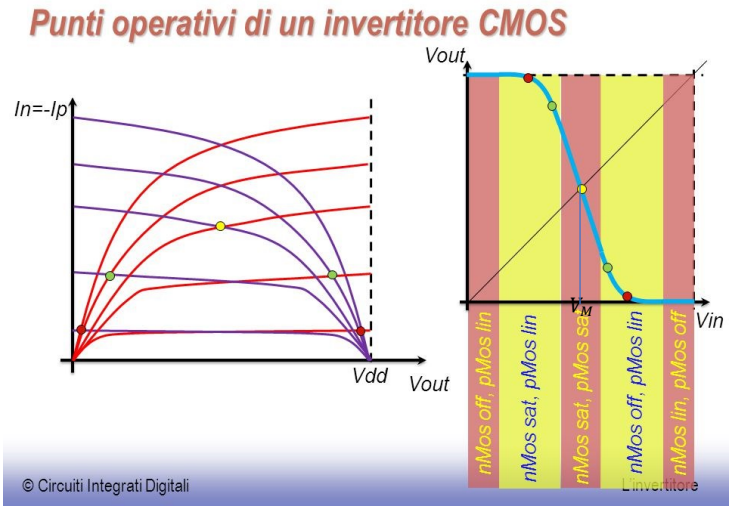

Figura 1.4.2: Punti operativi di un invertitore CMOS

Ponendo due invertitori in retroazione, si sfrutta la proprietà rigenerativa dell'invertitore.
Essa consente, dopo un breve transitorio, di sanare progressivamente l'errore in ingresso, portando il segnale verso i due punti di lavoro possibili ('A' e 'B', che corrispondono a '1' e '0').
Dato il doppio punto d'equilibrio, questa proprietà prende il nome di bistabilità ed è utilizzata da SRAM, latch e registri statici.
Qualsiasi sia l'errore, entro pochi cicli, si verrà ricondotti al valore corretto, ciò conferisce grande robustezza ai rumori.
Questa robustezza è supportata anche dalla memorizzazione del bit e del suo negato (V_{o1} *e* V_{o2}), così da poter fare un double check della sua correttezza.
Se però il rumore, porta l'ingresso su V_M (punto 'C' del grafico), il valore risulta irrecuperabile, come spiegato precedentemente.
Date tutte queste proprietà, la catena di invertitori in logica CMOS a retroazione positiva è alla base dei meccanismi di memorizzazione dei bit.

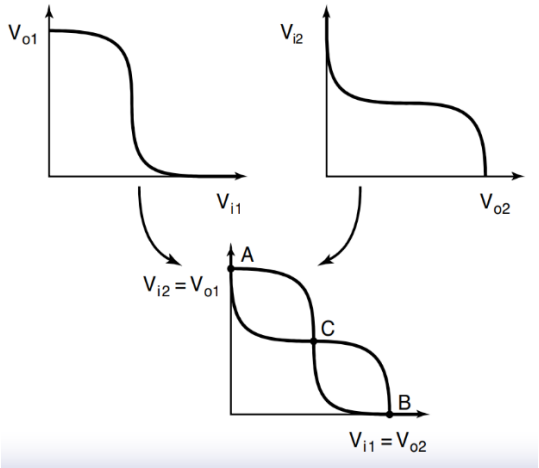

Figura 1.4.3: Caratteristica di 2 invertitori in retroazione

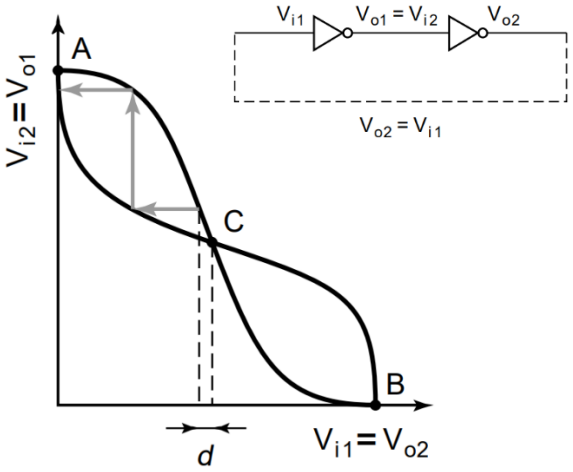

Figura 1.4.4: Bistabilità e proprietà rigenerativa degli invertitori in retroazione

Capitolo 2

Tipologie di memorie a semiconduttore

2.1 Definizione

Le memorie a semiconduttore sono apparati elettronici che conservano dati e programmi attraverso l'utilizzo di dispositivi a semiconduttore, tipicamente transistor di diverse tipologie con tecnologia MOSFET.

2.2 I vari tipi di memoria a semiconduttore

La distinzione principale nelle memorie a semiconduttore è data dalla volatilità.
Quest'ultima indica l'incapacità della memoria nel mantenere i dati, in essa contenuti, in mancanza di alimentazione.
Questo comportamento, sebbene accettabile per le RAM, risulta inaccettabile per le memorie di archiviazione.
In quanto, mentre la RAM contiene programmi ed istruzioni in esecuzione, le memorie di archiviazione custodiscono dati che non devono assolutamente essere persi, tra cui il software di sistema e il BIOS (programma che permette il corretto avvio del sistema operativo e del computer).
Memorie volatili sono SRAM e DRAM.
Mentre alle memorie non volatili appartengono ROM, PROM, EPROM, EEPROM (memorie ROM) e flash a NOR e a NAND (memorie Flash).

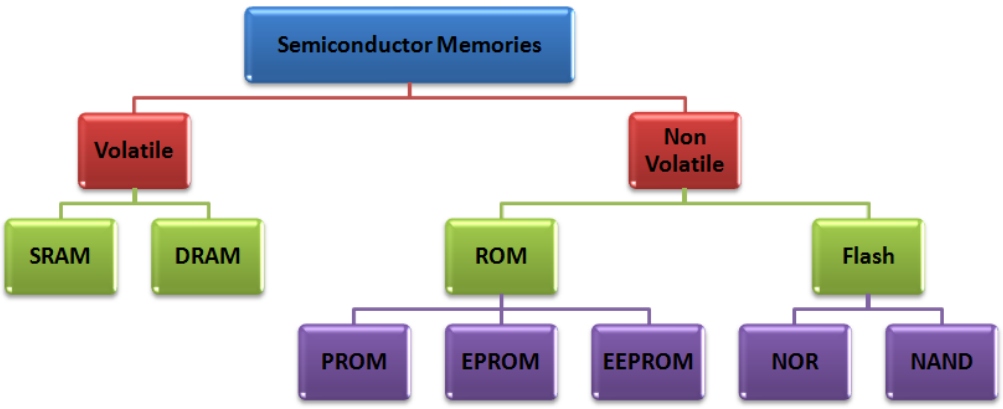

Figura 2.1: Rappresentazione delle varie tipologie di memoria a semiconduttore

2.3 Perché utilizzare semiconduttori?

Gli apparati a semiconduttore vengono sfruttati per la costruzione delle memorie moderne per via della loro incredibile rapidità in accesso.

Per accedere ai dati, infatti, le ssd impiegano circa 0.1 ms e le DRAM, che riescono a sincronizzarsi col clock, $7 - 12$ ns.

Negli hard-disks, invece, l'accesso è nell'ordine delle decine di millisecondi, un divario enorme.

Ciò è dovuto non solo alle prestazioni migliori dei semiconduttori, ma anche alla loro differente tipologia di accesso.

Mentre gli hard-disks accedono sequenzialmente ai dati, le memorie a semiconduttore sono dotate di un accesso randomico.

Per accesso randomico, non si intende che i dati ottenuti siano dettati dal caso, ma che i tempi d'accesso, sia in lettura che in scrittura, non dipendono dalla posizione del dato nella memoria, come avveniva precedentemente per gli hard-disks.

In questo modo ogni dato in qualsiasi locazione viene raggiunto nello stesso tempo, il che risulta di grande vantaggio.

Oltre alle spiccate prestazioni, si predilige l'uso della memoria MOS perché essa è meno costosa e consuma meno elettricità della memoria a nucleo magnetico (magnetic core memory).

Inoltre, l'accesso alle celle di memoria può essere regolato dalla frequenza di clock del processore, per ottimizzare ulteriormente le operazioni.

Per tutti questi motivi, al giorno d'oggi, è consuetudine utilizzare memorie a semiconduttore per la memorizzazione dei dati nei calcolatori.

Per il meccanismo di caching, inoltre, si predilige utilizzare le memorie più veloci e costose per le memorie più interne, a diretto contatto con la CPU, e quelle più lente ed economiche per le memorie secondarie e di archiviazione (seguendo l'ordine della *Figura 2.2*).

In questo modo si riesce a velocizzare l'intera macchina, mantenendo il prezzo contenuto.

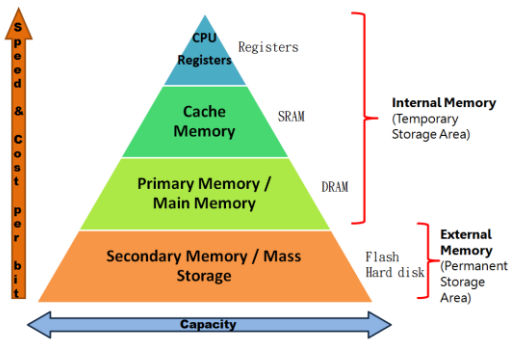

Figura 2.2: Tipologie di memorie utilizzate per le memorie della macchina

2.4 Memorie con struttura a matrice

Le memorie trattate presentano tutte una struttura a matrice.

Questo caratteristico sistema di memorizzazione è il motivo del loro accesso randomico.

Infatti, una volta selezionata la riga e la colonna, tramite un indirizzo (address), si ottiene la cella di memoria desiderata.

L'indirizzo è composto da due parti ($da\ A_K\ a\ A_{L-1}\ e\ da\ A_0\ a\ A_{K-1}$).

La prima per indirizzare la riga, che sarà sfruttata dal decoder di riga (row decoder) per selezionare la riga richiesta (word line WL).

E la seconda, per selezionare la colonna (bit line BL) tramite il decoder di colonna (column decoder).

I decoder, in elettronica digitale, hanno lo scopo di convertire una stringa di '1' e '0' in un segnale di attivazione diretto all'indirizzo richiesto.

Le celle di memoria da un bit vengono raggruppate in piccole unità chiamate parole (words), solitamente composte da M bit (con M = 1,2,4, o 8 bit).

Per questo motivo le righe prendono il nome di word line.

Una memoria, con un indirizzo di riga di $L - K$ bit, creerà 2^{L-K} word line, composte da 2^K words per ogni riga, poiché, avendo $L - K$ bit di riga, si avranno K bit di colonna e quindi 2^K colonne.

Il totale di words presenti sarà allora $2^{L-K} \cdot 2^K = 2^L$, con L lunghezza totale dell'indirizzo.

Ricordando, inoltre, che ogni word è composta da M bit, si può calcolare il numero di bit memorizzabili, che risulta pari a $M \cdot 2^L$.

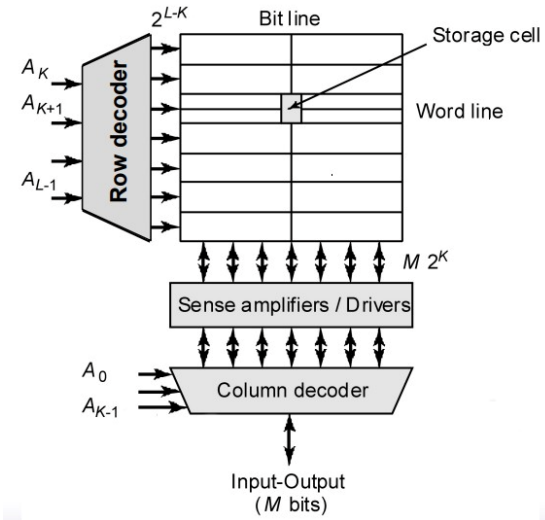

Figura 2.3: Memoria con struttura a matrice

Capitolo 3
Memorie volatili: RAM

3.1 SRAM

3.1.1 Definizione e proprietà

Il funzionamento della SRAM (Static Random Access Memory) si basa su numerosi transistor che formano un flip-flop digitale per la memorizzazione di ogni bit.
Il termine statico, che differenzia la SRAM dalla DRAM, indica la capacità di conservare i dati memorizzati senza il bisogno di un refresh periodico.
Ovviamente, quanto ciò affermato, è valido finché permane l'alimentazione, in quanto la SRAM rimane pur sempre una RAM e quindi volatile.
Un ulteriore importante vantaggio è la sua rapidità sia in lettura, che in scrittura (10 ns contro i 60ns della DRAM).
Tuttavia, non è esente da svantaggi.
La SRAM, infatti, consuma più energia, è meno densa (6 transistor per cella) ed è più costosa per bit della cugina DRAM.
Per queste sue caratteristiche, viene utilizzata per memorie cache più piccole dei computer, che richiedono una grande velocità.

Figura 3.1.1: SRAM Alliance Memory da 1Mbit,
128000 byte x 8 bit, 32 Pin

3.1.2 Struttura e funzionamento

Le celle di una SRAM sono costituite da un circuito in retroazione positiva, formato da due invertitori logici.

Le uscite di questi ultimi sono collegate, alle due estremità, alle linee dei dati (WL e BL) tramite due transistor, che prendono il nome di porte di trasmissione.

Le singole coppie di porte di trasmissione vengono abilitate a seconda della cella su cui deve essere effettuata la lettura o scrittura.

L' uscita della SRAM è differenziale, in quanto vengono resi disponibili sia il bit memorizzato Q che la sua negazione $\bar{Q}$, al fine di migliorare il controllo dei margini di rumore.

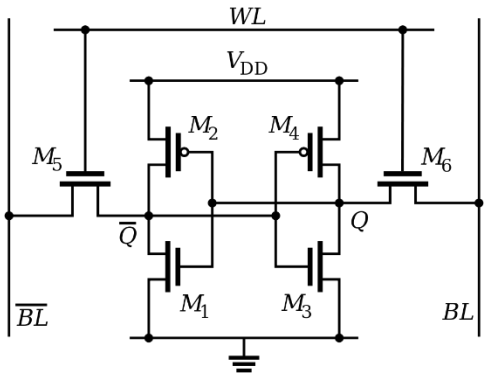

Figura 3.1.2: SRAM CMOS a 6 transistor

3.1.3 Lettura

Assumiamo che la cella stia memorizzando un 1 ($Q = V_{DD}$ e $\bar{Q} = 0V$).

A V_{DD} corrisponde il valore logico '1', mentre a $0V$, lo '0' logico.

Prima che le operazioni di lettura comincino, la $\overline{BL}$ e la BL, vengono precaricate ad una tensione tra $V_{DD}/2$ e V_{DD}.

Per comodità, assumeremo che il loro voltaggio sia esattamente V_{DD}.

Quando la WL viene attivata (e quindi portata a V_{DD}), i pass transistor M_5 e M_6 si accendono e permettono il fluire della corrente.

In particolare, osservando la parte di sinistra del circuito, la corrente scorrerà dalla capacità di $\overline{BL}$ ($C_{\overline{BL}}$) a $C_{\bar{Q}}$ e M_1 (che risulta acceso, perché con il gate collegato a Q).

La SRAM può ora essere rappresentata in maniera semplificata, come in *Figura 3.1.3,* con degli aperti al posto dei pass transistor M_2 e M_3, perché spenti.

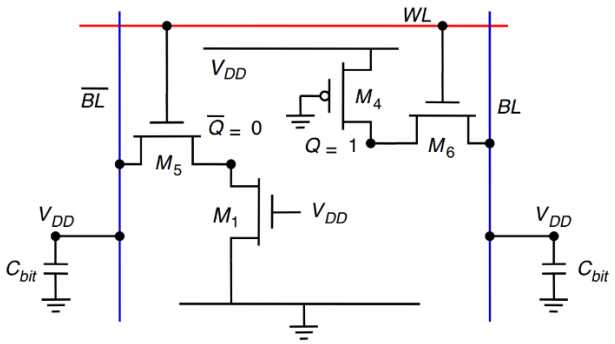

Figura 3.1.3: Operazione di lettura

La corrente scorre fino a quando non si raggiunge lo stato di equilibrio, nel quale $C_{\bar{Q}}$ arriva ad una tensione di $V_{\bar{Q}}$.

Perché l'operazione di lettura risulti non distruttiva, bisogna che $V_{\bar{Q}} \leq V_{tn}$ con V_{tn} tensione di soglia di M_3 M_4.

In questo modo, tramite la bistabilità degli invertitori a retroazione positiva, si riuscirà a riportare $\bar{Q}$ al suo corretto valore, in questo caso $V_{\bar{Q}} = 0V$.

Per questo motivo, c'è bisogno di tarare le $Z = \frac{W}{L}$ dei transistor M_1 e M_5.

Aumentando la Z_1 o la Z_5, si sbilancia il rapporto tra i transistor, che devono avere correnti uguali ($I_5 = I_1$), proprio come nel caso dell'invertitore CMOS.

Perciò cambiando la loro relazione, si regola la quantità di corrente che può scorrere nel transitorio e quindi anche la $V_{\bar{Q}}$ che si avrà alla fine di esso.

Una volta giunti alla fine del processo, si avrà una $V_{\overline{BL}}$ minore di quella di partenza, perché le cariche sono state trasportate via dalla corrente I_5.

La V_{BL} risulta poi invariata , perché sia V_{BL} che V_Q sono a V_{DD} e quindi non c'è stato alcuno scambio di carica.

Alla luce di queste considerazioni, si può calcolare la ΔV variazione di tensione delle bitline, che corrisponde al dato memorizzato all'interno della cella appena letta.

$$\Delta V = \frac{I_5 \Delta t}{C_{\overline{BL}}}$$

Da notare è la grandezza della capacità $C_{\overline{BL}}$ (1-2 pF), dovuta alle numerose celle collegate alla $\overline{BL}$ e la minuscola variazione di tensione ($\Delta V = 0.1 - 0.2V$), dovuta alla necessità di rendere la lettura non distruttiva.

Per questo motivo si sfrutta il sense amplifier, che porta la ridotta variazione di tensione ΔV a '0' o a '1'.

Nonostante sia stato considerato solo il caso della lettura dell' '1', le operazioni di lettura dello '0' risultano identiche.

Ovviamente, anziché leggere un decremento di ΔV sulla $\overline{BL}$, che viene poi trasformato in un '1' dal sense amplifier, si leggerà un decremento di ΔV sulla BL, che col sense amplifier verrà tradotto in uno '0'.

Poiché, inoltre, la SRAM è una struttura simmetrica, le considerazioni sul rapporto $\frac{Z_5}{Z_1}$, ricadono analogalmente anche su $\frac{Z_6}{Z_3}$.

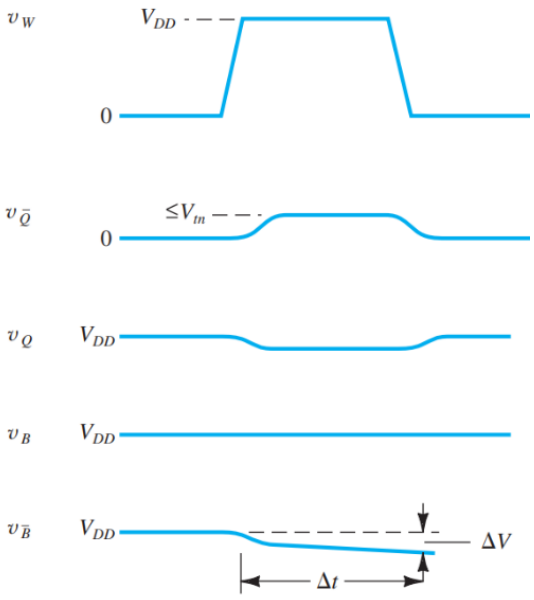

Figura 3.1.4: Variazioni delle tensioni

3.1.5 Scrittura

Si consideri ora l'operazione di scrittura di uno '0' in una cella che conteneva precedentemente un '1'.

Avendo un '1' memorizzato, la cella è disposta proprio come nel caso precedente di lettura. Ora, però, la BL viene portata a $0V$ e la $\overline{BL}$ a V_{DD}, con l'obiettivo di far arrivare almeno uno dei due nodi (Q o $\bar{Q}$) oltre la tensione di soglia logica del rispettivo invertitore ($M_1 - M_2$ per Q e $M_3 - M_4$ per $\bar{Q}$).

In questo modo, tramite la proprietà rigenerativa e la bistabilità, si riesce a invertire il valore memorizzato nella cella.

Il fatto che, sia sufficiente che solo uno dei due Q prenda il valore imposto dalla scrittura, è molto utile.

Infatti, nel caso in cui si volesse scrivere '1' su $\bar{Q}$, come abbiamo già visto, al fine di mantenere la lettura non distruttiva, i transistor $M_5 - M_1$ erano stati precedentemente tarati perché il $\bar{Q}$ non superasse la V_{th} e quindi $\bar{Q}$ non potrà essere portato da '0' a '1'.

Per fortuna, la scrittura dello '0' su Q funziona egregiamente e non rende la lettura distruttiva. Perciò, cambiando valore a Q, cambierà anche il valore di $\bar{Q}$, perché in feedback tra loro.

È necessario stare attenti, però, a come tarare i transistor anche nel caso della scrittura per le motivazioni opposte a quelle della lettura, cioè in modo che il valore salvato possa, in questo caso, virare.

Si otterranno quindi degli intervalli di Z che saranno validi sia per le operazioni di scrittura che di lettura.

I meccanismi di carica e scarica funzionano in modo analogo a quelli delle operazioni di lettura.

Ad esempio, nella parte destra del circuito in *Figura 3.1.5*, si viene a creare una corrente $I_4 = I_6$, che scarica la Q, portandola, dopo un breve transitorio, da V_{DD} a $0V$.

Poiché la carica e scarica è riferita a condensatori di dimensioni ridotte ($C_Q \; e \; C_{\bar{Q}}$), l'operazione di scrittura risulta più veloce di quella di lettura (che doveva caricare e scaricare le grandi $C_{BL} \; e \; C_{\overline{BL}}$).

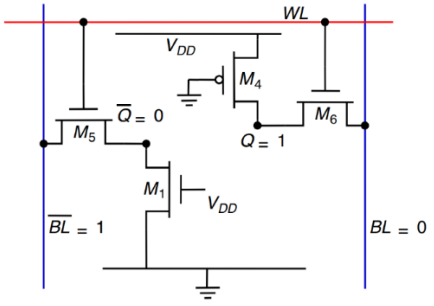

Figura 3.1.5: scrittura di uno '0' in una SRAM che conservava un '1'

3.2 DRAM

3.2.1 Definizione e proprietà

La DRAM (Dynamic Random Access Memory) è una memoria RAM in grado di salvare il valore dei bit attraverso l'uso di uno o tre transistor e un piccolo condensatore.
Per via di questo suo meccanismo di memorizzazione, essa ha bisogno di un refresh periodico, in quanto la carica immagazzinata in un condensatore viene persa con il tempo.
Un'altra caratteristica fondamentale che distingue la DRAM dalla SRAM è il tipo di uscita. Nella DRAM, infatti, essa è singola, non differenziale e ciò comporta la perdita di una controprova della veridicità del valore memorizzato.
D'altra parte, però, le DRAM hanno dimensioni molto contenute.
Esistono, infatti, due tipi di architetture per le DRAM: a 1 o a 3 transistor per cella.

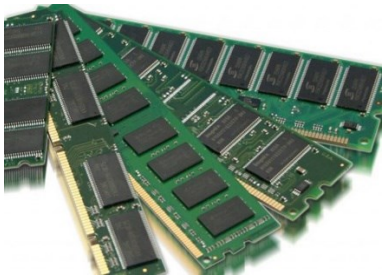

Figura 3.2.1: DRAM

3.2.2 DRAM-1T

La DRAM-1T è composta da un nMOS con il gate collegato a WL e gli altri terminali a BL e a C_S.
Quest' ultimo sarà il condensatore che manterrà il valore logico della cella, mentre l'nMOS avrà la funzione di pass transistor, regolando l'accesso al C_S.

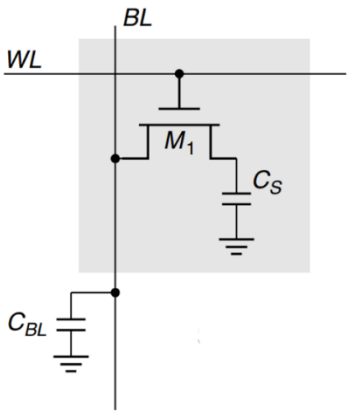

Figura 3.2.2: DRAM-1T visione circuitale

3.2.2.1 Scrittura

Si consideri l'operazione di scrittura di un '1' in una cella che conservava precedentemente uno '0'.

La WL viene portata a V_{DD} per la selezione della cella, si attiva di conseguenza l'nMOS, che permette il passaggio di corrente tra BL e C_S.

La BL, posta a V_{DD}, carica il C_S tramite una corrente I_1.

Si presenta ora un problema dato dall'utilizzo della logica a pass transistor.

L'nMOS, infatti, non può caricare il source oltre $V_{DD} - V_{th}$, in quanto c'è bisogno che $V_{GS} \geq V_{th}$ perché l'nMOS risulti acceso.

Andando a sostituire V_{DD} a V_G e $V_{DD} - V_{th}$ a V_S, la condizione di accensione diventa $V_{DD} - (V_{DD} - V_{th}) \geq V_{th}$ e quindi $V_{th} \geq V_{th}$.

Un nMOS, se usato come pass transistor, potrà quindi caricare o scaricare il source solo da 0 a $V_{DD} - V_{th}$.

Secondo la stessa logica, un pMOS avrà come escursione da $V_{th}\ fino\ a\ \ V_{DD}$.

Per ovviare quindi a questo difetto, la WL può essere caricata a $V_{DD} + V_{th}$.

In questo modo il source, e quindi anche il C_S, potranno avere tensione uguale a V_{DD} e, di conseguenza, valore logico '1'.

Per via degli effetti di perdita di carica del condensatore, sarà mecessario ripetere periodicamente la scrittura del bit nella cella, con un refresh ogni 5/10ms.

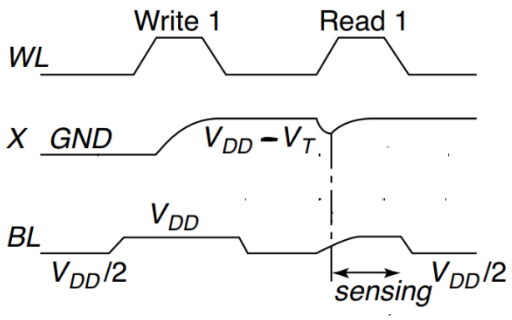

Figura 3.2.3: Tensioni durante le operazioni di scrittura e lettura su una DRAM-1T

3.2.2.2 Lettura

Si valuti l'operazione di lettura di un valore V_{CS}, con $V_{CS} = V_{DD}$ se la cella immagazzina un '1', oppure $V_{CS} = 0V$ se si tratta di uno '0'.

La BL viene precaricata a $V_{DD}/2$ e la WL, posta a V_{DD}, attiva l'nMOS.

Per la conservazione della carica, vale la seguente equazione.

$$C_S V_{CS} + C_{BL}\frac{V_{DD}}{2} = (C_{BL} + C_S)\left(\frac{V_{DD}}{2} + \Delta V\right)$$

Che, una volta rielaborata, diventa:

$$\Delta V = \frac{C_S}{C_{BL} + C_S}\left(V_{CS} - \frac{V_{DD}}{2}\right)$$

Da qui, notando che la $C_{BL} \gg C_S$, in quanto somma dei condensatori delle BL in parallelo, si può approssimare la formula precedente come segue:

$$\Delta V \approx \frac{C_S}{C_{BL}}\left(V_{CS} - \frac{V_{DD}}{2}\right)$$

Adesso, se $V_{CS} = V_{DD}$ (e quindi se la cella conserva un '1'), $\Delta V \approx \frac{C_S}{C_{BL}}\left(\frac{V_{DD}}{2}\right)$.

Altrimenti, per $V_{CS} = 0V$, $\Delta V \approx -\frac{C_S}{C_{BL}}\left(\frac{V_{DD}}{2}\right)$.

Questi ΔV, però, risultano abbastanza bassi per via della grande differenza tra C_{BL} e C_S.

Come soluzione, si adotta il sense amplifier (come nel caso della SRAM), che permette di ottenere come output un '1' se il ΔV è lievemente positivo e uno '0' nel caso opposto (*Figura 3.2.4*).

Da notare la distruttività della lettura che, per ottenere il valore memorizzato nella cella, di fatto lo fa virare, seppur di un voltaggio minimo.

Per riportare la cella al suo valore originario, il sense amplifier scrive l'output appena letto ed effettua un refresh di tutta la riga, precedentemente selezionata dal decoder di riga.

Questo refresh avviene anche durante le operazioni di scrittura su tutta la riga selezionata, in modo da parallelizzare le operazioni.

Nonostante lo svantaggio del refresh periodico, la DRAM 1-T risulta accessibile per qualsiasi operazione il 98% del tempo.

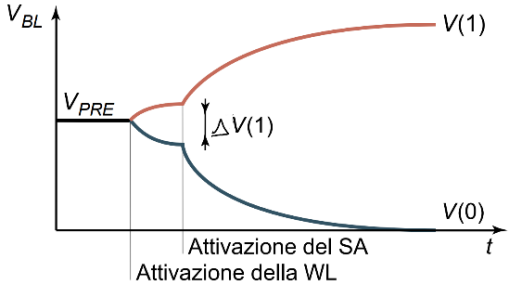

Figura 3.2.4: Sense amplifier

3.2.2 DRAM-3T

La DRAM-3T è composta da due BL, due WL, un condensatore C_S e tre nMOS.

La BL_1 è collegata ad M_1 e la BL_2 a M_3, mentre le due WL hanno scopi diversi.

La WWL, collegata al gate di M_1, si occupa delle operazioni di scrittura, mentre la RWL, collegata al gate di M_3, di quelle di lettura.

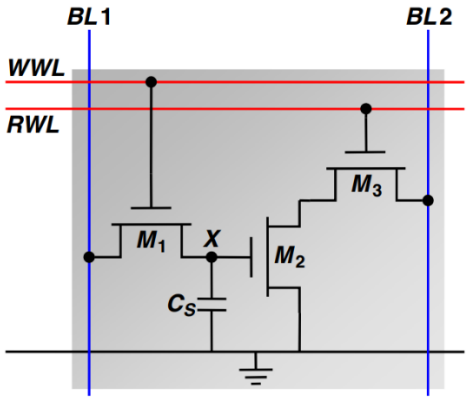

Figura 3.2.5: DRAM-3T

3.2.2.1 Scrittura

Per dare il via alle operazioni di scrittura, la WWL si attiva, accendendo di conseguenza M_1.

La BL_1 viene posta al valore desiderato e, a seconda di esso, carica o scarica il C_s tramite l'utilizzo di M_1 (perciò si potrà caricare C_s solo fino a $V_{DD} - V_{th}$, come visto precedentemente).

Una volta terminata l'operazione di scrittura, la WWL viene disattivata e M_1 si spegne.

Il nodo X risulterà quindi flottante, perché collegato a sinistra con un nMOS spento e a destra al gate di M_2 che, dal punto di vista resistivo, si comporta da resistenza infinita.

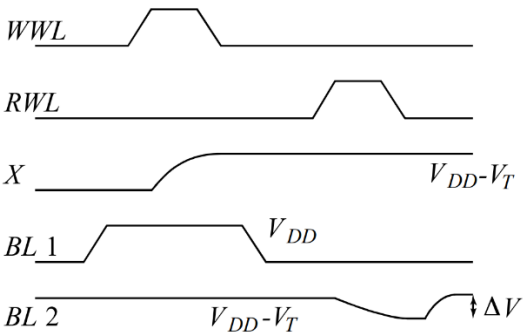

Figura 3.2.6: Tensioni durante le operazioni di scrittura e lettura su una DRAM-3T

3.2.2.2 Lettura

In lettura viene sfruttata la restante parte del circuito.

La RWL si attiva accendendo M_3 e collegando la BL_2 (posta a V_{DD}) alla rete di pulldown composta dalla serie di M_3 e M_2.

Ora, se M_2 risulta acceso (ovvero se $X = '1'$), la BL_2 si scaricherà restituendo in output $0V$ (‘0’).

Altrimenti, per $X = '0'$, l’ M_2 interrompe il pulldown, evitando che la BL_2 si scarichi e, di conseguenza, restituendo in output un ‘1’.

Da notare è il fatto che la cella DRAM-3T sia invertente, in quanto restituisce il valore opposto a quello memorizzato.

Questo problema risulta però facilmente risolvibile con l’utilizzo di un invertitore.

Inoltre, a differenza della DRAM-1T, la lettura è non distruttiva e il refresh, quindi, non è più necessario ad ogni operazione, ma solo periodicamente.

Capitolo 4

Memorie non volatili: ROM

Le ROM (Read Only Memory) sono memorie sfruttate prettamente per la memorizzazione di massa, per via della loro non volatilità.
Esistono vari tipi di ROM: le ROM a MOS, che si possono solo accedere in lettura, le PROM, ovvero le ROM programmabili, le EPROM, che sono sia programmabili che cancellabili e le EEPROM, che sono EPROM cancellabili elettricamente.

4.1 ROM a MOS

Le ROM a MOS sono delle memorie a matrice non riprogrammabili e di sola lettura.
I bit vengono salvati all'interno delle memorie durante il processo di fabbricazione.
Ci sono più modi per immagazzinare i bit e ognuno ha vantaggi e svantaggi.
Durante la progettazione si sceglie quindi il più consono per l'utilizzo della memoria e si procede poi alla produzione.

4.1.1 ROM-NOR

Le ROM-NOR si basano sulla porta in logica CMOS NOR.
Essa presenta sulla rete di pull-up una serie di pMOS e sulla rete di pull-down nMOS in parallelo (*Figura 4.1.1*).
In logica CMOS una porta logica ha la rete di pull-up con gli estremi collegati a V_{DD} e a V_{out}, mentre quella di pull-down a V_{out} e massa.

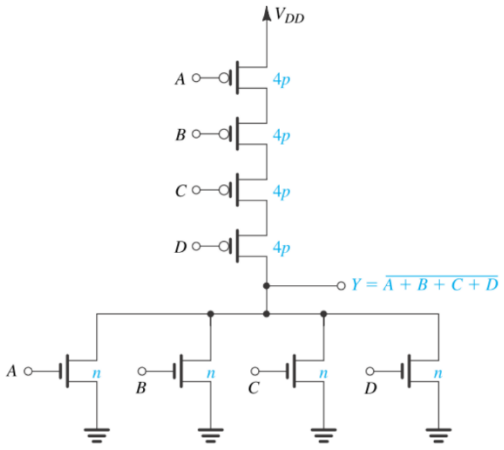

Figura 4.1.1: Porta NOR in logica CMOS

La memoria ROM-NOR è costituita da un pMOS con il source connesso a V_{DD}, il gate a massa, per mantenerlo costantemente acceso e il drain collegato alla BL.
Per ogni BL è presente un pMOS con questa configurazione, mentre gli nMOS vengono posti sulle intersezioni tra BL e WL e conferiscono alla cella il valore del bit.
Gli nMOS nella configurazione ROM-NOR sono posti con il drain sulla BL, il gate sulla WL e il source a massa.
Per questo motivo, una volta selezionata, tramite la WL, una cella col transistor, esso si accenderà scaricando completamente la BL ad esso collegata e portandola a '0'.
Dove invece non è rappresentato un transistor, la BL non verrà scaricata e manterrà il valore a '1'.

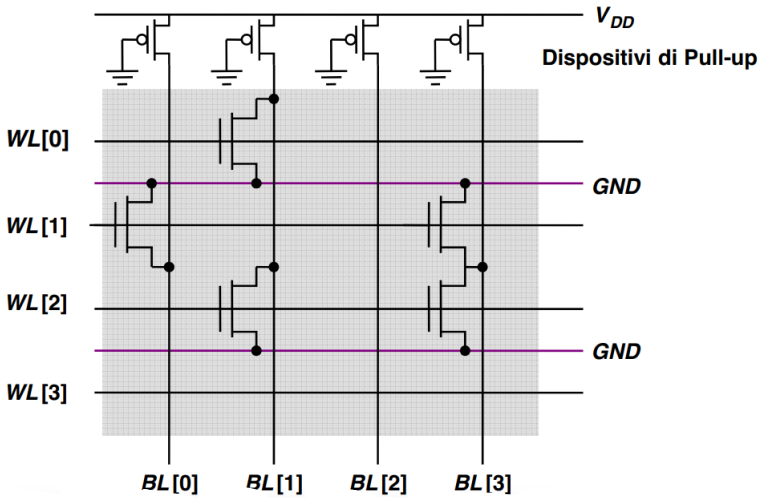

Figura 4.1.2: Rappresentazione di una ROM-NOR

Ci sono due modi per programmare questo tipo di ROM: mediante la regione attiva (*Figura 4.1.3 a*) e mediante la maschera dei contatti(*Figura 4.1.3 b*).
Entrambe le regioni indicano il passo della produzione del transistor MOSFET nel quale avviene la programmazione.
Programmando a livello della regione attiva, gli '0' si ottengono tramite la creazione di un transistor e gli '1' tramite la mancata fabbricazione del transistor in quella cella.
La regione attiva, primo passo per la fabbricazione di un transistor, non viene proprio deposta e, al suo posto, viene semplicemente collocato del metallo per condurre.
Le ROM programmate nella regione attiva pur essendo veloci e poco costose, non sono esenti da problemi.
Programmandole in uno dei primi passi della fabbricazione, risultano l'una diversa dall'altra dal primo istante e questa personalizzazione si paga.
Inoltre, gli errori costano caro perché in caso di bit sbagliato, c'è bisogno di ricostruire da zero un'altra ROM con il valore del bit corretto.

Nelle ROM programmate mediante maschera dei contatti o, più semplicemente, nelle MROM (Mask Read Only Memory), la storia è diversa.

I transistor vengono prodotti per ogni cella, ma solo le celle che devono contenere uno '0' avranno la BL collegata al drain del transistor, mentre le celle che devono mantenere un '1' avranno, invece, un aperto sul drain.

In questo modo si può produrre una ROM uguale per tutti fino allo scavo dei contatti (tra gli ultimi passi della fabbricazione).

L'MROM sarà quindi meno costosa, poiché la produzione fino alla maschera dei contatti sarà uguale per tutte le catene e ciò smorza decisamente il costo di realizzazione.

Inoltre, essendo programmabile quasi alla fine della sua fabbricazione, il progettista avrà più tempo per accertarsi della correttezza dei bit richiesti.

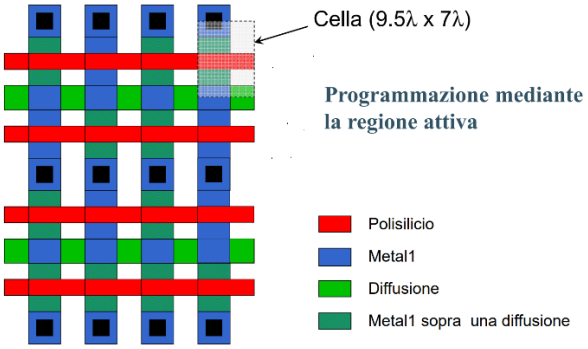

Figura 4.1.3a: ROM-NOR programmata con la regione attiva

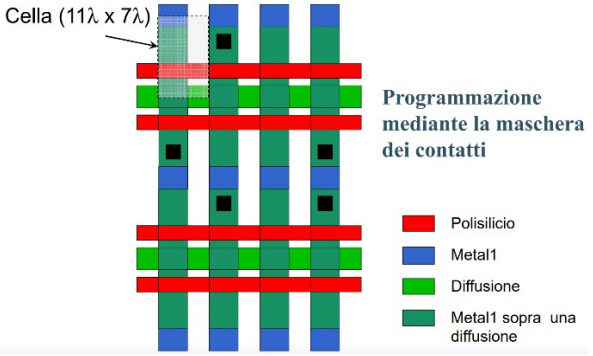

Figura 4.1.3b: ROM-NOR programmata mediante la maschera dei contatti

Uno svantaggio del circuito ROM è la dissipazione di potenza statica.

In particolare, quando viene selezionata una word, i transistor di questa particolare riga condurranno corrente statica che viene fornita dai transistor di carico PMOS.

La dissipazione di potenza statica può essere eliminata con una semplice modifica.

Invece di collegare i terminali di gate dei transistor PMOS a massa, è possibile connetterli a una linea di precarica φ che viene normalmente tenuta alta.

Poco prima di un'operazione di lettura, φ si abbassa e le BL vengono precaricate a V_{DD}, attraverso i transistor PMOS.

Il segnale di precarica φ diventa quindi alto e la WL viene selezionata, dopodiché si eseguono i passi precedentemente descritti.

4.1.2 ROM-NAND

Le ROM-NAND si basano invece sulla porta in logica CMOS NAND.

Essa presenta sulla rete di pull-up dei pMOS in parallelo e sulla rete di pull-down una serie di nMOS.

La rete di pull-up ha gli estremi collegati a V_{DD} e a V_{out}, mentre quella di pull-down a V_{out} e massa (*Figura 4.1.4*).

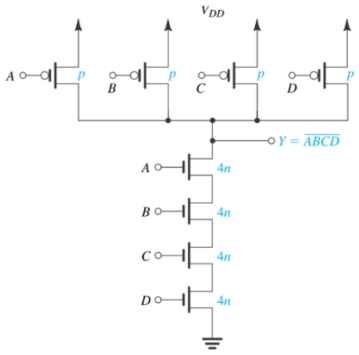

Figura 4.1.4: Porta NAND in logica CMOS

Questa variante della classica ROM-NOR consente una maggiore compattezza ed economicità a scapito della ridotta velocità.

La ROM-NAND appare compatta poiché i transistor nMOS sono posti in serie e non più in parallelo.

Ciò evita la creazione di linee per collegare ogni transistor a massa e a V_{out}.

Anche questa tipologia di ROM (come la MROM), in fase di realizzazione, presenta transistor su tutte le celle.

Per distinguere le celle con uno ‘0’ dalle celle con un ‘1’, si cortocircuitano i transistor delle celle che devono contenere uno ’0’, infatti, questa memoria funziona in modo speculare alla ROM-NOR.

Le WL sono sempre poste a ‘1’ e per selezionare una word, la WL corrispondente viene portata a ‘0’.

In questo modo i transistor nMOS risultano sempre accesi e scaricano la BL a ‘0’.

Quando però viene selezionato un transistor, il suo gate viene portato dalla WL a ‘0’ e quindi si spegne, interrompendo la serie che scarica la BL a massa e, grazie al pMOS che carica la BL, riportando in uscita un ‘1’.

Nei cortocircuiti, invece, che la WL sia a ‘0’ o a ‘1’ cambia poco, in quanto in ogni caso la BL viene scaricata a massa e restituisce a V_{out} uno ‘0’.

Il cortocircuito viene creato nel passo di fabbricazione MET-1, un passo successivo a quello dello scavo per la maschera dei contatti.
Ciò consente alla ROM-NAND di essere più versatile ed economica anche della MROM.
La velocità ridotta, infine, è dovuta alla serie degli nMOS, che portano tempi di scarica e di carica ad alzarsi notevolmente.

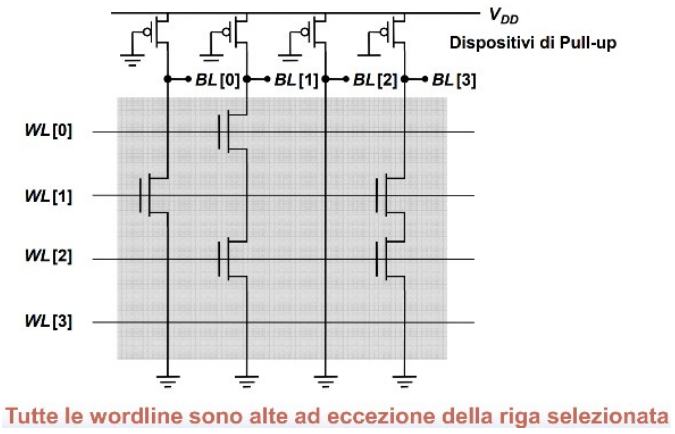

Figura 4.1.5: Rappresentazione di una ROM-NAND

4.2 PROM

Le PROM (Programmable Read Only Memory) sono memorie OTP (One Time Programmable) programmabili dall'utente un'unica volta, in quanto non sono cancellabili.
Vengono sfruttate per la memorizzazione di programmi a basso livello come firmware o microcodici.
La differenza tra ROM a MOS e PROM sta nel fatto che, mentre la ROM a MOS viene programmata durante la sua fabbricazione, la PROM può essere programmata dopo di essa, rendendola più versatile.
Vengono infatti prodotte solitamente PROM vuote, che saranno poi programmate a seconda dell'ambito nel quale verranno utilizzate.
Un'ulteriore vantaggio delle PROM è il loro poco consumo di potenza e la piccola area per cella.

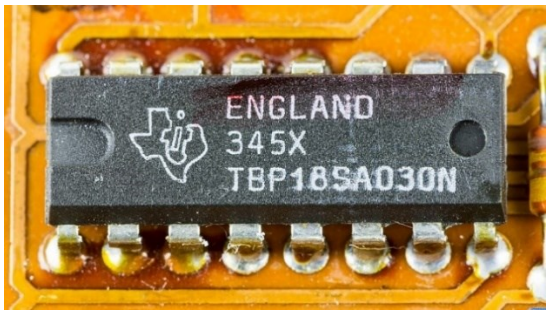

Figura 4.2.1: PROM TBP18SA030N della Texas Instruments

4.2.1 Programmazione e lettura

Per programmare le PROM si sfruttano i fusibili in essa presenti.

Una PROM nasce con ogni bit posto a ‘1’, poi, tramite delle correnti di sovraccarico, si bruciano i fusibili in corrispondenza delle celle che dovranno contenere uno ‘0’.

Ora, dopo la programmazione, la memoria si comporta come una ROM a MOS, in quanto non più modificabile e accessibile in modo analogo.

Questo processo di programmazione, oltre ai vantaggi precedentemente elencati, ne ha altri. Infatti, bruciando direttamente i fusibili, si rende impossibile contraffare la memoria e i dati in essa contenuti e ciò costituisce una prova della sicurezza e dell’autenticità del prodotto.

Un altro vantaggio è l’esistenza di specifici software per programmare le PROM, senza il bisogno di dover cablare a mano la memoria.

Esistono anche tipi di PROM nelle quali, anziché bruciare i fusibili per gli ‘0’, si creano collegamenti, tramite antifusibili, per gli ‘1’.

In questo tipo di PROM, lo stato di partenza è, ovviamente, con tutti i bit posti a ‘0’.

A seconda delle esigenze dell’utente, verrà usato questo tipo di memoria se si vuole vietare la riprogrammazione della stessa o, altrimenti, la EPROM o la EEPROM, che permettono di essere sovrascritte numerose volte.

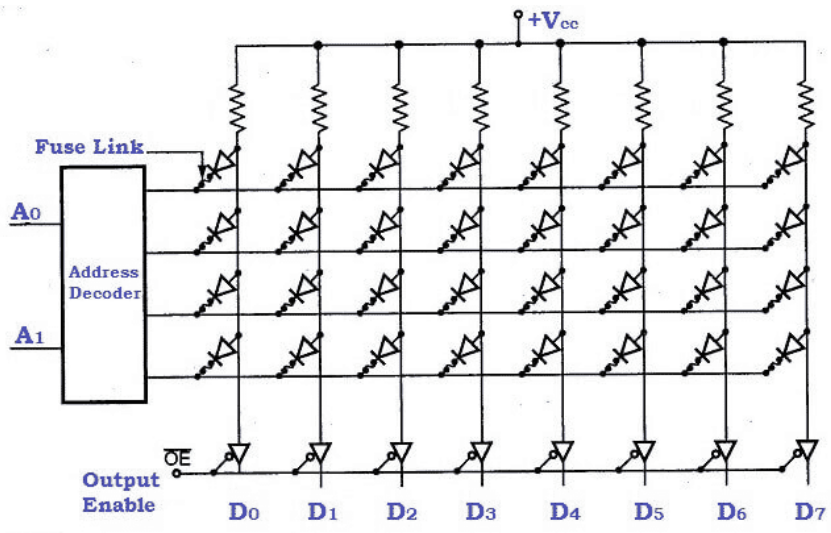

Figura 4.2.2: Matrice di memorizzazione di una PROM

4.3 EPROM

A differenza delle precedenti memorie, la EPROM (Eresable Programmable Read Only Memory) è più versatile.

Essa, infatti, permette la lettura e la scrittura dei dati più di una volta, rendendo di fatto la memoria illimitatamente sovrascrivibile.

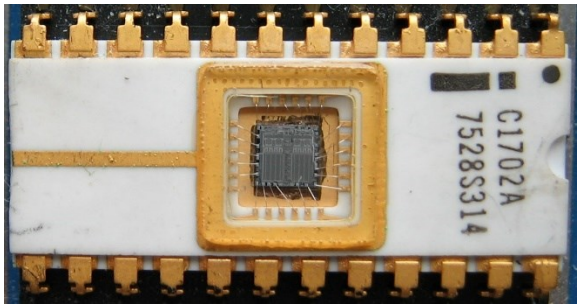

Figura 4.3.1: Memoria EPROM Intel C1702A.

Da notare la finestra al quarzo al centro della memoria, fondamentale per l'operazione di cancellazione a onde ultraviolette

4.3.1 Transistor a gate flottante FGMOS

Per poter essere così versatili, le EPROM sfruttano un tipo diverso di nMOS: l'nMOS a gate flottante, FGMOS (Floating Gate MOS).

Esso sfrutta il principio dell'iniezione di portatori caldi attraverso il breakdown a valanga, un processo che permette lo scorrere della corrente in materiali solitamente considerati isolanti e generato da una tensione moderata applicata su una distanza molto corta.

L' FGMOS è un nMOS che presenta due gate in polisilicio, separati dall'ossido (*Figura 4.3.2*).

Un gate non è connesso elettricamente al resto del circuito ed è lasciato flottante, stato che gli conferisce appunto il nome di gate flottante.

L'altro gate, invece, detto gate di selezione, ha lo stesso compito e funzionamento del normale gate di un nMOS.

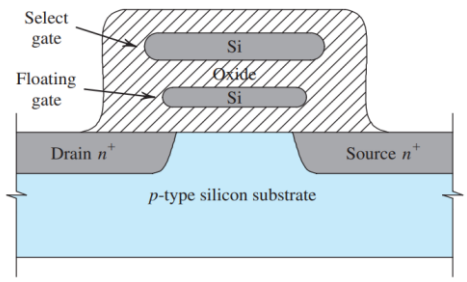

Figura 4.3.2: Sezione trasversale di un nMOS a gate flottante

4.3.2 Programmazione

Per programmare il transistor, viene incrementata la V_{DS} e la tensione tra gate di selezione e source V_{GS}.

Il transistor si comporta da normale nMOS, dando il via alla corrente I_{DS}, che trasporta gli elettroni dal source al drain.

Quando gli elettroni si avvicinano al pinch-off sul drain, acquisiscono energia cinetica.

Ora questi elettroni, noti come portatori caldi, sfruttando l'energia cinetica appena ottenuta, attraversano l'ossido, attirati dalla forte V_{GS} verticale.

La barriera di ossido, non essendo un isolante ideale, permette agli elettroni con un'energia sufficiente di oltrepassarla senza problemi, per essere poi catturati dal gate flottante.

Una volta raggiunto quest'ultimo, i portatori caldi, che hanno esaurito tutta l'energia che avevano a loro disposizione, rimangono nel gate flottante, che acquisisce quindi carica negativa.

La carica, a differenza delle DRAM rimane fissa nel tempo, anche in assenza di alimentazione, in quanto il gate flottante è isolato dall'ossido.

Il processo di carica del gate flottante è fortunatamente limitato.

La carica negativa, che a mano a mano viene accumulata, infatti, riduce il campo elettrico verticale fino al punto in cui esso non perde la capacità di attirare con sufficiente forza i portatori caldi.

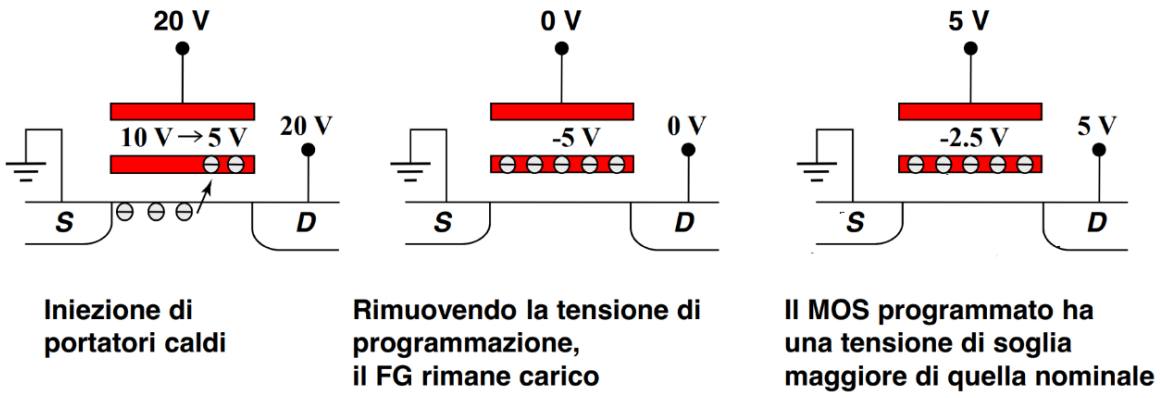

Figura 4.3.3: Programmazione a iniettori caldi

Questa carica causa un ritardo nell'accensione del nMOS, quando utilizzato normalmente. La V_{GS}, infatti, dovrà essere, come al solito, maggiore della tensione di soglia perché il transistor risulti acceso.
Avendo però una carica negativa nel gate flottante, la V_{GS} dovrà compensarla per poter accendere l'nMOS.
La V_{GS} dovrà quindi essere maggiore della tensione di soglia sommata alla tensione per compensare la carica negativa del gate flottante.
Si viene di fatto a creare una nuova tensione di soglia $V_{th'} = V_{th} + V_{FG}$ che varrà V_{th} quando il gate flottante è scarico e che sarà invece pari ad un valore $V_{th'} > V_{th}$ quando il gate flottante è carico.
A seconda del tipo di architettura ROM usata, verranno scelti valori logici diversi per gli stati. Ad esempio, considerando una ROM a NOR, si utilizza lo '0' per lo stato non programmato, in quanto, una volta acceso il FGMOS, tramite V_{WL}, la tensione verrà scaricata completamente e sulla BL rimarranno 0V.
Analogamente, viene assegnato l''1' allo stato programmato poiché, posta sempre la stessa V_{WL}, il transistor non si accenderà e non scaricherà la tensione; pertanto, sulla BL verrà restituita V_{DD}.

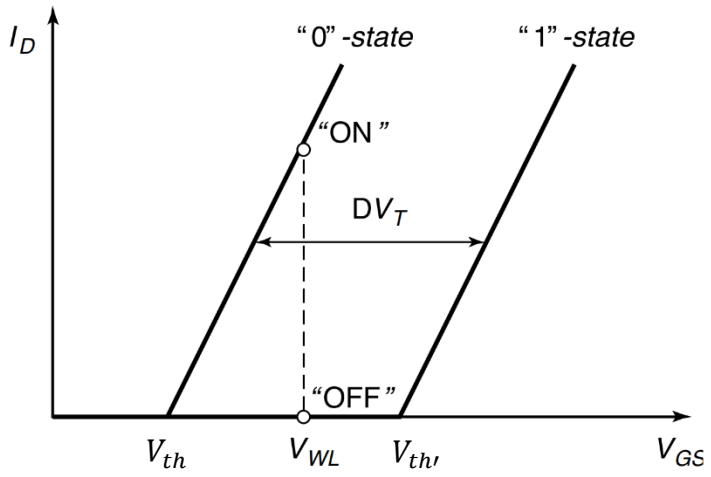

Figura 4.3.4: Caratteristica del nMOS a gate flottante.
La scala è logaritmica, per questo motivo le caratteristiche assumono la forma di rette e non di curve

4.3.3 Lettura

In lettura si applica una tensione V_{GS} (V_{WL} in *Figura 4.3*.4) posta tra V_{th} e $V_{th'}$ e si osservano i due casi possibili.

Se il transistor era stato programmato, risulterà spento in quanto non ha una $V_{GS} \geq V_{th'}$.

Altrimenti, se non era stato programmato, sarà acceso e in conduzione, poiché $V_{GS} \geq V_{th}$.

Allo stato programmato viene assegnato un '1', in quanto, non accendendosi tramite la V_{WL}, il transistor non scarica la BL, che quindi mantiene il suo valore a '1'.

Allo stato non programmato, invece, si assegna uno '0', poiché si comporta esattamente come un transistor in una ROM-NOR.

Esso, infatti, dopo essersi acceso per via della V_{WL}, scarica la tensione del la BL a '0'.

4.3.4 Cancellazione

La cancellazione dei dati avviene tramite l'uso di una luce ultravioletta con lunghezza d'onda pari a $253.7\ nm$.

Illuminando infatti la cella con quest'onda per uno specifico lasso di tempo (tipicamente nell'ordine delle decine di minuti), si riesce a trasmettere sufficiente energia agli elettroni intrappolati nel gate flottante, per attraversare l'ossido e ritornare nel substrato.

Poiché un'operazione simile comporta un dispendio di tempo considerevole, risulta conveniente utilizzare l'EPROM solo nei casi in cui la cancellazione avvenga poco frequentemente.

Come si vede dalla *Figura 4.3.1*, le EPROM dispongono di una finestra al quarzo che permette l'interazione degli ultravioletti con le celle della memoria.

Un ulteriore difetto riscontrato nella cancellazione della EPROM è che, per poter cambiare il valore anche di un solo byte, c'è bisogno ugualmente di cancellare l'intero contenuto della memoria, a differenza della EEPROM.

4.4 EEPROM

Per porre rimedio alla dispendiosa operazione di cancellazione della EPROM, è stata sviluppata la EEPROM (Electrically Eresable Programmable Read Only Memory).
Essa è capace, infatti, di essere programmata molte volte come la EPROM, ma ha il vantaggio di poter essere cancellabile elettricamente, evitando l'uso delle onde ultraviolette.
Ciò comporta un enorme risparmio in termini di tempo, si passa infatti dai $5 - 30$ minuti per la cancellazione dell'EPROM, agli $0.1 - 5$ ms della EEPROM.
Un altro vantaggio è la possibilità di cancellare singoli byte, senza dover cancellare per forza l'intera memoria.

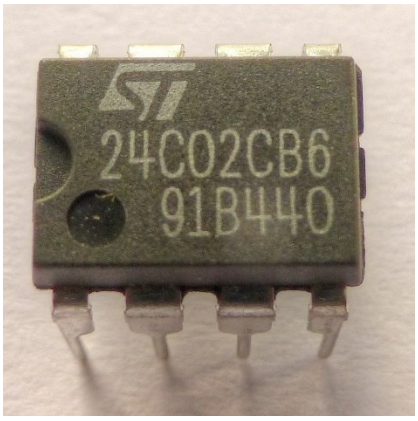

Figura 4.4.1: EEPROM STMicro M24C02 I²C

4.4.1 Transistor FLOTOX

Per poter cancellare elettricamente i dati, la memoria EEPROM sfrutta una diversa tipologia di transistor, il FLOTOX (Floating Gate Tunnel Oxide).
Questo nuovo tipo di transistor è identico al FGMOS, con l'eccezione di una regione, in prossimità del drain, dove l'ossido che separa il gate flottante dal canale è particolarmente sottile (*Figura 4.4.2*).

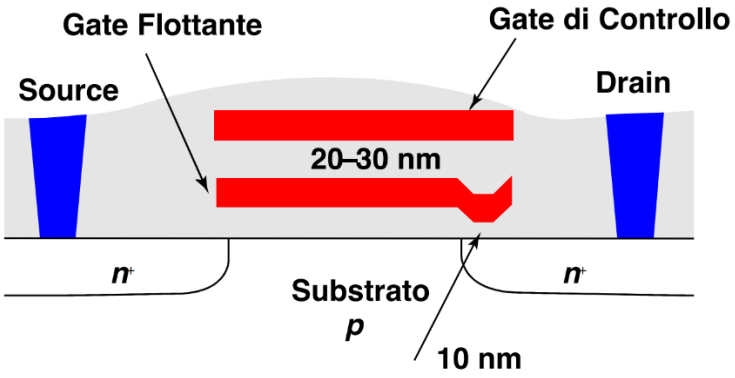

Figura 4.4.2: Transistor FLOTOX

Questa scelta strutturale è dovuta al diverso meccanismo di funzionamento delle due memorie.
Infatti, mentre la EPROM si basa sull'iniezione dei portatori caldi tramite breakdown a valanga, l'EEPROM sfrutta l'effetto di corrente di tunnel Fowler-Nordheim per iniettarli.
Questo effetto quantistico, che si viene a creare quando è presente un campo elettrico sufficientemente forte e uno strato sottile di isolante, permette ad alcuni portatori di attraversare il sottile strato di ossido per poi essere catturati dal gate flottante.
L'effetto è bidirezionale, ovvero, a seconda del valore della tensione del gate di selezione V_{GD}, si riesce a caricare o a scaricare il gate flottante, cioè a programmare o a cancellare la cella.

4.4.2 Programmazione

Come precedentemente affermato, il transistor viene pilotato attraverso V_{GD}.
Variando questa tensione, si riesce a controllare il manifestarsi dell'effetto tunnel.
Si vengono a creare, dunque, due effettive tensioni di soglia, per le quali avviene la cancellazione o la programmazione (la prima ha segno negativo e la seconda positivo, *Figura 4.4.3*).
La programmazione del FLOTOX è molto simile a quella della EPROM.
Infatti, nonostante cambi il principio per il quale i portatori caldi vengono iniettati, essi saranno comunque immagazzinati all'interno del gate flottante che, di conseguenza, alzerà la tensione di soglia (*vedi paragrafo 4.3.2*).
Quindi, anche per quanto riguarda le EEPROM, la cella potrà essere programmata o non programmata e assumerà '1' o '0' a seconda della presenza o meno di carica nel gate flottante.

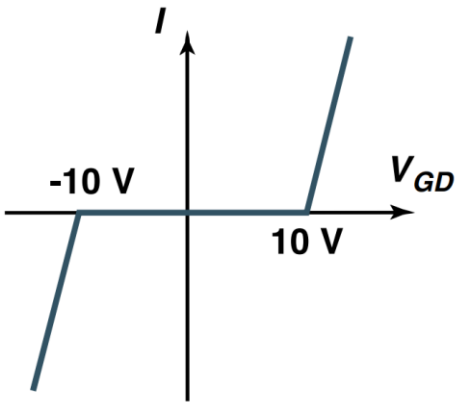

Figura 4.4.3: Corrente di tunnel di Fowler-Nordheim
in funzione della tensione V_{GD}.

4.4.3 Lettura

Anche la lettura avviene similmente alle celle EPROM.
Si applica una tensione compresa tra V_{th} e $V_{th'}$ e se il FLOTOX risulta acceso, allora non era stato programmato, altrimenti era già stato programmato.
A seconda dello stato, si assegnano '1' e '0' in modo analogo agli assegnamenti dei valori logici nelle EPROM (*vedi paragrafo 4.3.3*).

4.4.4 Cancellazione

La cancellazione è la prima operazione che risulta differente dalla EPROM.
Invece di usare macchinari per cancellare le celle tramite la luce ultravioletta, infatti, le EEPROM sfruttano l'effetto tunnel per scaricare il gate flottante.
Applicando una tensione negativa su V_{GD}, l'intero meccanismo, che prima ha permesso la programmazione della cella, si inverte.
L'effetto tunnel, infatti, come si vede nella *Figura 4.4.3*, crea una corrente opposta e riporta nel substrato gli elettroni che erano stati precedentemente catturati dal gate flottante.
Questa procedura, come già osservato precedentemente, è capace non solo di cancellare una cella di un byte alla volta, ma di farlo estremamente più velocemente e senza il bisogno di macchinari esterni.
Analogamente alle memorie EPROM, però, anche le EEPROM presentano una criticità legata alla fase di cancellazione.
Il processo di scarica del FLOTOX può infatti causare, come effetto indesiderato, l'accumulo di una carica positiva con conseguente variazione della tensione di soglia del dispositivo.
Il problema è stato risolto mediante l'introduzione di un transistor d'accesso, che è parte integrante della cella (*Figura 4.4.4*).
La presenza di questo transistor determina, tuttavia, un maggiore impiego di area rispetto alle EPROM, perché ora per ogni cella dovranno essere posti due transistor in serie.
Questo costo in termini di area è ulteriormente aggravato dal fatto che la cella FLOTOX sia intrinsecamente più grande della cella FGMOS, a causa della presenza della zona di ossido di tunnel.
La presenza del transistor d'accesso può essere, però, sfruttata al massimo per compensare la maggiore superficie occupata della cella.
Infatti, può essere usato come filtro per poter porre il FLOTOX a delle tensioni di soglia infime per lo'0' e a tensioni di soglia maggiori di V_{DD} per l''1'.

In questo modo il FLOTOX rimane sempre acceso quando contiene uno ‘0’ e sempre spento quando contiene un ‘1’, indipendentemente dalla tensione che gli si applica.
Così si velocizza di gran lunga la cella, che presenterà già i condensatori parassiti scarichi o carichi, a seconda del valore logico salvato.
Perciò, una volta acceso il transistor d’accesso tramite la V_{WL}, la BL potrà essere scaricata rapidamente, nel caso di uno ‘0’ salvato nel FLOTOX, oppure rimarrà a ‘1’ senza il bisogno di caricare la capacità parassita.

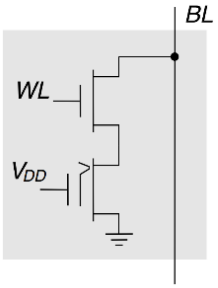

Figura 4.4.4: Cella di una EEPROM con transistor d’accesso

4.4.5 Limitazioni

Ci sono due principali limitazioni nell’uso della EEPROM che le danno di fatto una data di scadenza: resistenza e ritenzione.
Durante la riscrittura, l'ossido del gate accumula gradualmente elettroni.
Il campo elettrico degli elettroni intrappolati si aggiunge a quello degli elettroni nel gate flottante, abbassando la finestra tra le tensioni di soglia per gli ‘0’ e gli ‘1’ (V_{th} e $V_{th\prime}$).
Dopo un numero sufficiente di cicli di riscrittura, la differenza diventa troppo piccola per essere riconosciuta e la cella rimane bloccata nello stato programmato.
Questo problema è noto come resistenza.
Durante l'immagazzinamento, invece, gli elettroni iniettati nel gate flottante possono andare alla deriva attraverso l'isolante (soprattutto a temperature elevate) e causare una perdita di carica, riportando la cella nello stato di cancellazione.
Questo fallimento della cella è noto invece come ritenzione.
Ci sono, inoltre, anche altri aspetti negativi della EEPROM.
La fabbricazione di uno strato di ossido molto sottile è, difatti, un processo difficile e quindi costoso, perciò un banco di memoria EEPROM integra meno bit ad un costo addirittura più alto rispetto alla memoria EPROM.
Inoltre, per poter programmare, cancellare e leggere la cella bisogna usare più voltaggi.
Ovviamente questi difetti, che vengono comunque bilanciati dalla maggiore versatilità e affidabilità della EEPROM, verranno poi risolti dalla più moderna e popolare EEPROM FLASH.

Capitolo 5

Memorie non volatili: FLASH

5.1 La cella EEPROM FLASH

La memoria EEPROM Flash è una memoria non volatile, programmabile e cancellabile.
A differenza della EEPROM classica, questo nuovo tipo di memoria presenta un FLOTOX con l'intero gate flottante molto più vicino al substrato.
In questo modo l'effetto tunnel non è più legato al drain, ma può avvenire lungo tutta la superficie del gate.

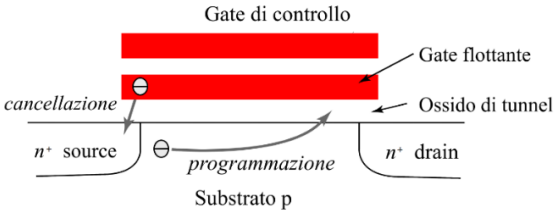

Figura 5.1.1: Esempio di cella EEPROM Flash

Le memorie Flash presentano tutti i vantaggi delle EEPROM classiche, con l'aggiunta di proprietà uniche che attribuiscono loro il nome flash.
Infatti, queste memorie possiedono la capacità di cancellare e salvare dati in un unico passo e, soprattutto, possono cancellare e programmare più bit contemporaneamente, grazie all'uso di celle multilivello.
Tutto ciò conferisce alla memoria una grande velocità e affidabilità, oltre ad un costo minore di quello delle precedenti EEPROM.
Si pensi che per cancellare o programmare un intero gruppo di celle, il tempo si aggira intorno ai $0{,}01 - 1$ ms, contro i $0.1 - 5$ ms della EEPROM per la cancellazione di un solo byte.
Rimane, però, il problema di ritenzione dei dati, che limita la vita delle memorie flash a 10000 cicli di scrittura.
Ad oggi esistono comunque algoritmi ad hoc che permettono di prolungare ulteriormente la vita di queste velocissime memorie.
Per questi motivi, attualmente le memorie flash sono le memorie dominanti per la memorizzazione della gran parte dei dispositivi.
Le memorie flash ssd (solid state storage), infatti, sono presenti come memorie secondarie ormai in tutti i pc e console e segnano di fatto una nuova era.
Esse, infatti, rendono i tempi di caricamento di qualsiasi programma di ordini di grandezza inferiori rispetto a quanto avveniva con l'uso degli hard disks.

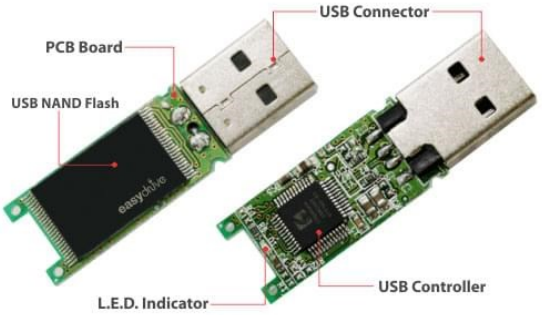

Figura 5.1.2: Esempio di memoria Flash
usata per salvare i dati di una chiavetta usb

Come precedentemente osservato per le ROM a MOS, esistono due architetture diverse anche per le memorie Flash: le Flash-NOR e le Flash-NAND

5.2 Flash-NOR

Per quanto riguarda le Flash-NOR, esse sono disposte in modo analogo alle ROM-NOR classiche.
Le celle, ovvero i transistor a gate flottante, sono poste con il drain sulla bit line BL, il gate di selezione connesso alla word line WL e il source a una tensione comune a blocchi, chiamata common source.
In questo modo, ponendo celle in parallelo, i byte possono essere cancellati a blocchi e non più singolarmente.
Data la loro velocità e accesso in lettura randomico e a singoli bit, vengono sfruttate per l'esecuzione di parti di codice (ad esempio può essere usata per il BIOS, in quanto soggetta a pochissimi errori, praticamente zero).

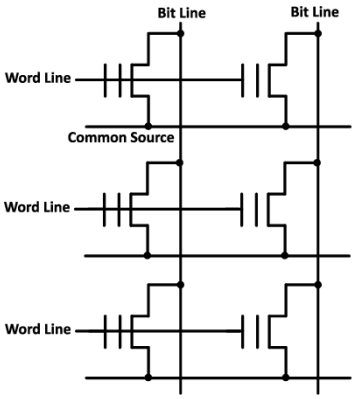

Figura 5.2.1: Rappresentazione della memoria Flash-NOR

5.2.1 Programmazione

La programmazione avviene tramite l'iniezione di portatori caldi attraverso il breakdown a valanga, come accadeva per le EPROM.

La WL, selezionando la word corrispondente, pone il gate delle celle ad un valore alto e la common source viene, invece, collegata a massa.

La programmazione delle Flash-NOR avviene bit a bit e non a blocchi di bit come la cancellazione.

Infatti, si riesce a programmare unicamente la cella tratteggiata in *Figura 5.2.2*, poiché l'unica ad avere le tensioni corrette applicate ai terminali.

Siccome si sfrutta l'iniezione di portatori caldi con breakdown a valanga, la tensione V_{GS} deve essere elevata, per permettere di creare un campo elettrico sufficientemente forte da attirare le cariche, ma deve essere elevata anche la V_{DS}, tensione che deve accelerare le cariche lungo il canale, per permettere loro di acquisire abbastanza energia cinetica.

Per fare ciò, la BL, in corrispondenza alla cella desiderata, viene posta ad un voltaggio sufficientemente alto (minore della V_{GS}).

La programmazione prosegue poi nello stesso modo in cui avveniva nelle EPROM.

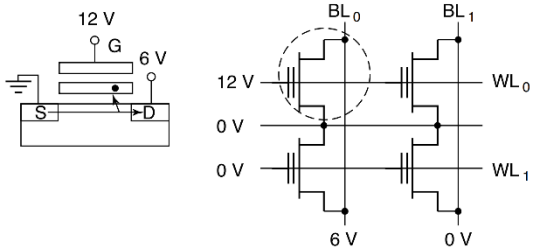

Figura 5.2.2: Cella Flash predisposta alla programmazione

5.2.2 Lettura

La lettura avviene bit per bit, come la programmazione e risulta praticamente identica alla lettura della EPROM e della EEPROM.

Verrà letta la cella con le tensioni corrette applicate ai terminali, ovvero la cella tratteggiata in *Figura 5.2.3*.

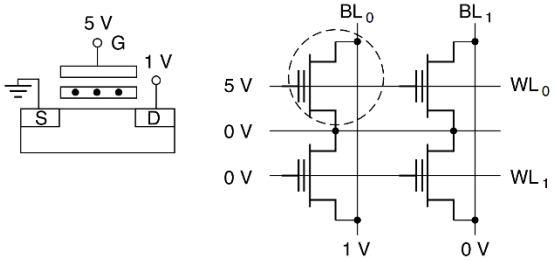

Figura 5.2.3: Cella Flash predisposta alla lettura

Selezionando la word desiderata, viene applicata una V_{WL} compresa tra la tensione di soglia della cella programmata e di quella non programmata (come avveniva per la lettura delle celle EPROM e EEPROM).
In questo modo il transistor scaricherà a '0' la BL, inizialmente posta a '1', se la cella non era stata programmata e, viceversa, la manterrà a '1' se era già stata programmata.

5.2.3 Cancellazione

La cancellazione in una memoria Flash avviene tramite l'effetto di corrente di tunnel Fowler-Nordheim, precedentemente osservato nelle memorie EEPROM classiche.
La particolarità della cancellazione delle memorie Flash è il fatto che essa avvenga a blocchi, proprietà che velocizza non di poco l'interazione con la memoria.
Come avveniva per la cancellazione delle EEPROM, anche in questo caso l'obiettivo è cercare di invertire la tensione posta sul transistor.
Questa volta però, poiché la cancellazione avviene sul source e non sul drain, bisognerà invertire la V_{GS}, portandola ad un valore negativo sufficiente perché abbia luogo l'effetto di corrente di tunnel.
Perciò il gate di tutti i transistor del blocco viene collegato a massa e il source, tramite il common source, comune a tutto il blocco, viene portato a V_{DD} (nel caso della *Figura 5.2.4*, 12V).
In questo modo, adesso, la tensione V_{GS} avrà un valore sufficientemente grande in modulo, da permettere la cancellazione di tutto il blocco, che, come preannunciato, avverrà esattamente come nella EEPROM.

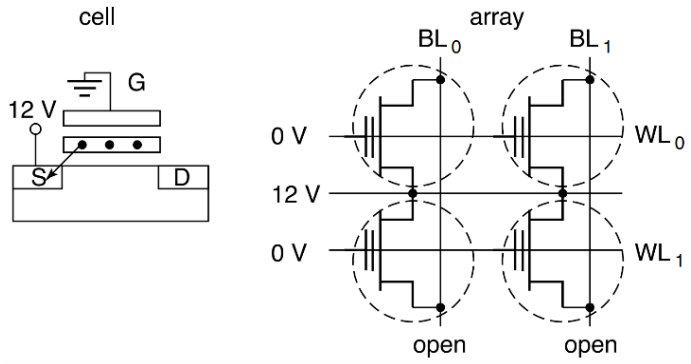

Figura 5.2.4: Cella Flash predisposta alla cancellazione

5.3 Flash-NAND

Le memorie Flash-NAND hanno tempi di cancellazione e scrittura ridotti in confronto alle Flash-NOR, in quanto le operazioni avvengono su blocchi di byte contemporaneamente. Ciò rende, di fatto, le Flash-NAND memorie perfette per l'aggiornamento rapido dei dati, anche per via del ridotto blocco di byte da aggiornare ogni volta (8Kb della NAND contro i 64Kb della NOR).

Come accadeva per le ROM-NAND, inoltre, le Flash-NAND richiedono una minore superficie per cella, consentendo così una maggiore densità di memorizzazione e quindi un costo per bit inferiore rispetto alle memorie Flash-NOR.

Questa economicità e densità si paga con un accesso in lettura sempre a blocchi, che, in questo caso, limita le memorie nell'essere utilizzabili unicamente come memorie secondarie di massa (ssd).

Ciò perché i microprocessori hanno bisogno di accessi random a livello byte per funzionare, come gli accessi in lettura delle Flash-NOR.

Passando alla struttura, invece, essa risulta molto simile a quella della ROM-NAND, essendo formata da una serie di transistor collegati fra loro.

Il primo e l'ultimo di questi transistor, che collegano la serie di transistor a gate flottante alla bit line BL da un capo e alla source line dall'altro, prendono il nome di transistor di selezione. I transistor a gate flottante, inseriti tra quelli di selezione, invece, sono normalmente collegati e accessibili tramite la word line WL.

Come per la ROM-NOR classica, tutte le WL sono inizialmente poste a '1', mentre quella che seleziona la cella desiderata è posta a '0'.

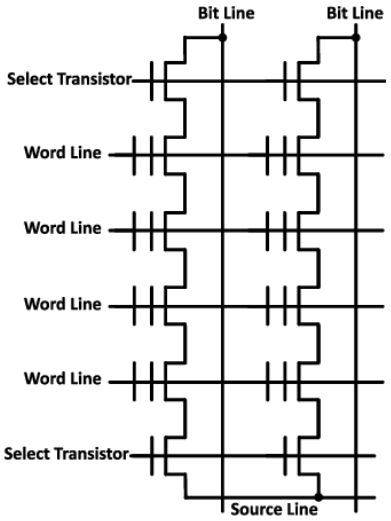

Figura 5.3.1: Rappresentazione della memoria Flash-NAND

5.3.1 Programmazione

La programmazione e la cancellazione avvengono tramite l'effetto di corrente di tunnel di Fowler-Nordheim, in modo simile a quanto avveniva nelle EEPROM.
La differenza sostanziale sta nell'organizzazione dei dati, infatti, nelle Flash-NAND i dati sono allestiti in stringhe, pagine, blocchi, piani e matrici.
Le stringhe sono le sequenze di celle di memoria connesse in serie e variano dalle 32 alle 128 celle per stringa.
Queste ultime sono poi raggruppate in pagine, che si riuniscono in blocchi.
I piani, invece, sono composti da un certo numero di blocchi collegati alla stessa BL, mentre la matrice è l'intera memoria, compresi i circuiti che permettono lettura, scrittura e cancellazione dei dati.
Una Flash-NAND può essere scritta e letta in pagine, ma cancellata in blocchi.
In questo tipo di memorie, inoltre, la programmazione può avvenire unicamente su pagine che non sono programmate.
Perciò, per poter scrivere in una pagina che contiene già dati, il contenuto attuale della pagina e i nuovi dati devono essere copiati in una nuova pagina completamente vuota.
Se è disponibile una pagina adatta non sorge alcun problema, altrimenti, un blocco viene cancellato completamente per far spazio alla scrittura.
La vecchia pagina viene quindi contrassegnata come non valida ed è disponibile per la cancellazione e il riutilizzo.

5.3.2 Lettura

La lettura delle Flash-NAND è un misto tra la lettura delle EEPROM e delle ROM-NAND.
Infatti, tutte le WL sono poste a '1' e quando si seleziona una particolare WL, essa viene portata ad una tensione compresa tra V_{th} e $V_{th'}$, tensioni di soglia rispettivamente della cella non programmata e di quella programmata.
In questo modo, se il transistor selezionato si accende, scaricando la BL a '0', vuol dire che esso non era stato programmato e viceversa nel caso opposto.

5.3.3 Cancellazione

Come precedentemente asserito, anche la cancellazione delle Flash-NAND avviene tramite l'effetto di corrente di tunnel di Fowler-Nordheim.
La cancellazione di queste memorie è rapida e a blocchi e avviene ponendo una tensione sufficientemente negativa sui terminali delle celle, affinché si verifichi l'effetto tunnel.

5.4 Limitazioni

Essendo le memorie Flash, di fatto delle memorie EEPROM, esse presentano gli stessi svantaggi e limitazioni come, ad esempio, la ritenzione dati.

Questi problemi rendono le memorie Flash riscrivibili un numero finito di volte e causano, nelle Flash-NAND, errori di lettura.

Per risolvere queste complicanze si utilizzano dei controllori, che limitano l'accesso in lettura ad una cella ad un numero finito di volte, dopo il quale, non si ha più la certezza che il dato in essa contenuto sia corretto.

L'accesso in scrittura e lettura, infatti, provoca gravi danni alle celle di memoria, che faticheranno sempre più a mantenere i dati corretti.

Inoltre, esistono algoritmi ad hoc che permettono di distribuire la scrittura su tutti i settori della memoria, per evitare l'eccessiva riscrittura della stessa zona di memoria e il non utilizzo di un'altra.

Tramite questi mezzi, la vita della memoria viene allungata e preservata.

Feature	NOR Flash		NAND Flash	
	General	S70GL02GT	General	S34ML04G2
Capacity	8MB – 256MB	256MB	256MB – 2GB	256MB
Cost per bit	Higher	$6.57x10^{-9}$ USD/bit for 1ku	Lower	$2.533x10^{-9}$ USD/bit for 1ku
Random Read speed	Faster	120ns	Slower	30µS
Write speed	Slower		Faster	
Erase speed	Slower	520ms	Faster	3.5ms
Power on current	Higher	160mA (max)	Lower	50mA (max)
Standby current	Lower	200µA (max)	Higher	1mA (max)
Bit-flipping	Less common		More common	
Bad blocks while shipping	0%		Up to 2%	
Bad block development	Less frequent		More frequent	
Bad block handling	Not mandatory		Mandatory	
Data Retention	Very high	20 years for 1K program-erase cycles	Lower	10 years (typ)
Program-erase cycles	Lower	100,000	Higher	100,000
Preferred Application	Code storage & execution		Data storage	

Figura 5.4: Vantaggi e svantaggi delle due architetture

Capitolo 6
CONCLUSIONI

Lo scopo principale del presente lavoro era quello di descrivere al meglio le fondamentali strutture di memorizzazione a semiconduttore, evidenziandone caratteristiche negative e positive e di tentare di fornire una risposta al perché, al giorno d'oggi, esse abbiano preso il sopravvento.

Attraverso i vari capitoli, il MOSFET, l'elemento base delle memorie a semiconduttore, è stato descritto con le sue fasce di funzionamento e particolarità.
Si è passati poi alla struttura a matrice della memoria, configurazione usata da tutte le memorie a semiconduttore trattate.
Dopodiché sono stati descritti i principali tipi di RAM e ROM, paragonandoli in velocità, prezzo, capacità e affidabilità.
Inoltre, sono state ampiamente esposte le modalità di scrittura, lettura e cancellazione per ognuna di queste memorie a livello circuitale.

Per rispondere, infine, al quesito iniziale, le memorie a semiconduttore, nonostante le operazioni limitate in scrittura e un tempo massimo di immagazzinamento dei dati, sono attualmente le memorie più utilizzate per una serie di ragioni.
Risultano essere estremamente veloci, dotate di un accesso randomico e sincronizzabile col clock, di una grande densità di memorizzazione dei bit, di un minore utilizzo di energia, ma soprattutto di una grandissima varietà con proprietà diverse, che soddisfa qualsiasi domanda.

SITOGRAFIA

- What is Semiconductor Memory? Definition, Functional Block Diagram and Types of Semiconductor Memory - Electronics Desk
- Memoria a semiconduttori - Okpedia
- Memoria a semiconduttore (wikiita.com)
- Classification of Semiconductor Memories and Computer Memories - VLSIFacts
- https://www.radioamatore.info/attachments/798_Transistor_MOSFET.pdf
- Read Only Memory (ROM) - Working, Types, Applications, Advantages & Disadvantages (electricalfundablog.com)
- Looking inside a 1970s PROM chip that stores data in microscopic fuses (righto.com)
- What Is EEPROM and How Does it Work? - Programming Insider
- What is EEPROM (electrically erasable programmable read-only memory)? (techtarget.com)

BIBLIOGRAFIA

- Circuiti Integrati Digitali L'ottica del progettista di Jan M. Rabaey Anantha Chandrakasan Borivoje Nikolic
- Microelectronic circuits di Adel S. Sedra, Kenneth C. Smith