

**INVERTER CMOS**

**Analisi del funzionamento dell'inverter CMOS in condizioni statiche**

Quest'analisi suppone che i transistori si siano esauriti. Vedremo successivamente il comportamento nel tempo.

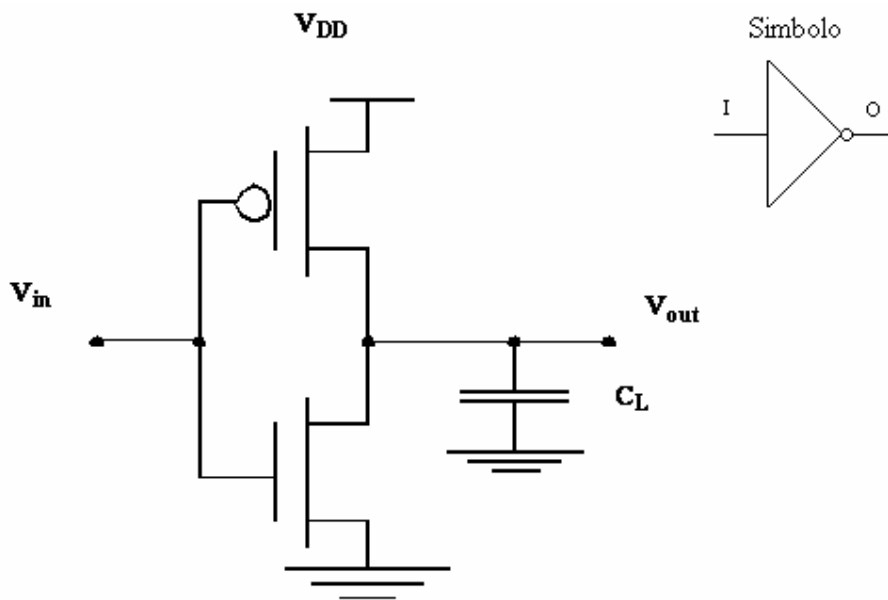


figura 1.1 - Inverter CMOS

**CASO A:**  $0 \leq V_{in} \leq V_{TN}$

In questo caso il transistor NMOS si trova in interdizione in quanto  $V_{GSN} < V_T$ , mentre il PMOS conduce perché  $V_{GSP} = V_{DD}$ .

Inoltre in condizioni statiche, cioè a transistori esauriti, la corrente che circola nei due transistor è quasi nulla. Quindi la tensione in uscita è pari a  $V_{DD}$  (l'uscita si trova quindi al livello logico ALTO), cioè  $V_{DSP} = 0$ ; se ne deduce che il transistor PMOS lavora in zona lineare.

Il circuito in figura 1.1 risulta dunque equivalente a quello mostrato in figura 1.2.

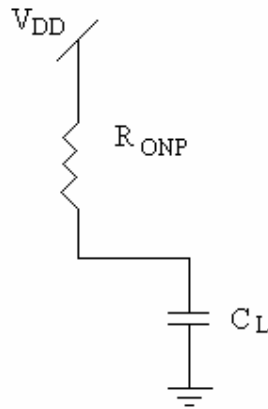


figura 1.2 - Circuito equivalente per il caso 1

**CASO B:**  $V_{TN} < V_{in} < V_{DD}/2$

Aumentando la tensione d'ingresso in modo tale da superare la tensione di soglia del transistor NMOS si ha che il PMOS rimarrà ancora nella situazione lineare mentre il transistor NMOS si trova in saturazione in quanto la  $V_{DSN}$  è molto alta e il canale risulta essere strozzato.

In questo caso il circuito di figura 1.1, esauriti i transistori, è equivalente a quello mostrato in figura 1.3.

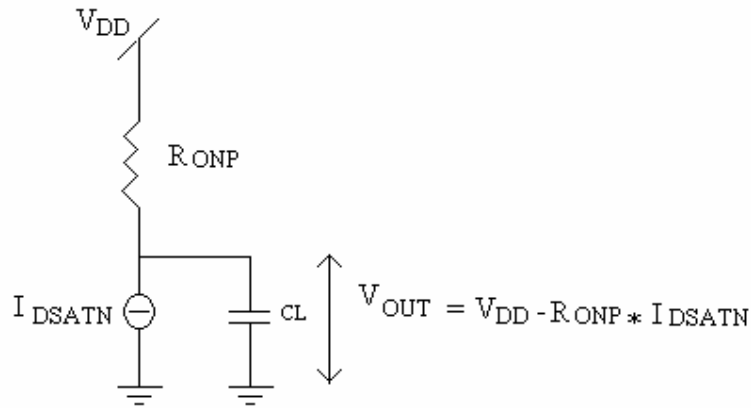


figura 1.3 - Circuito equivalente per il caso 2

Ad un certo punto anche il PMOS andrà nella zona di saturazione e si avrà un circuito equivalente a quello di figura 1.4, dove i due generatori di corrente hanno in realtà resistenze parassite molto alte in parallelo, che determinano il punto di lavoro  $V_{OUT}$  e rendono la caratteristica di figura 1.5 quasi verticale in questa zona.

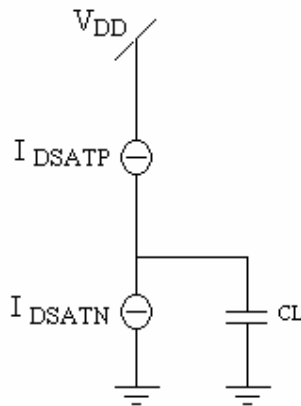


figura 1.4 - Circuito equivalente quando entrambi i transistor sono in saturazione

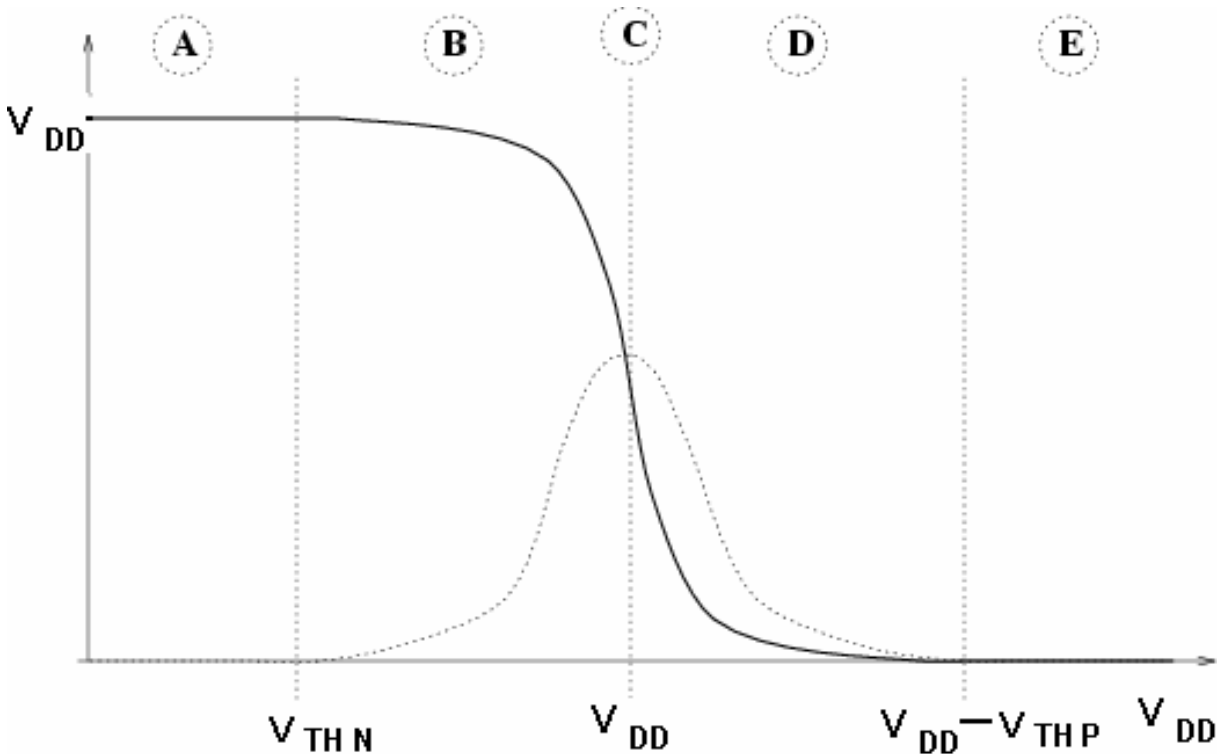
Successivamente si passa ad avere il transistor NMOS che conduce linearmente e il PMOS in zona di saturazione.

**CASO C:**  $V_{in} = V_{DD}/2$

In questo caso sia l'NMOS sia il PMOS sono saturi e conducono la stessa corrente.

I casi D ed E sono come i casi B e A rispettivamente, scambiando i ruoli di PMOS e NMOS.

A seguito dell'analisi precedente è possibile tracciare l'andamento qualitativo della caratteristica di trasferimento in tensione del CMOS (figura 1.5).



La linea tratteggiata nella figura precedente illustra l'andamento della corrente (identica, in condizioni statiche) nei transistor NMOS e PMOS.

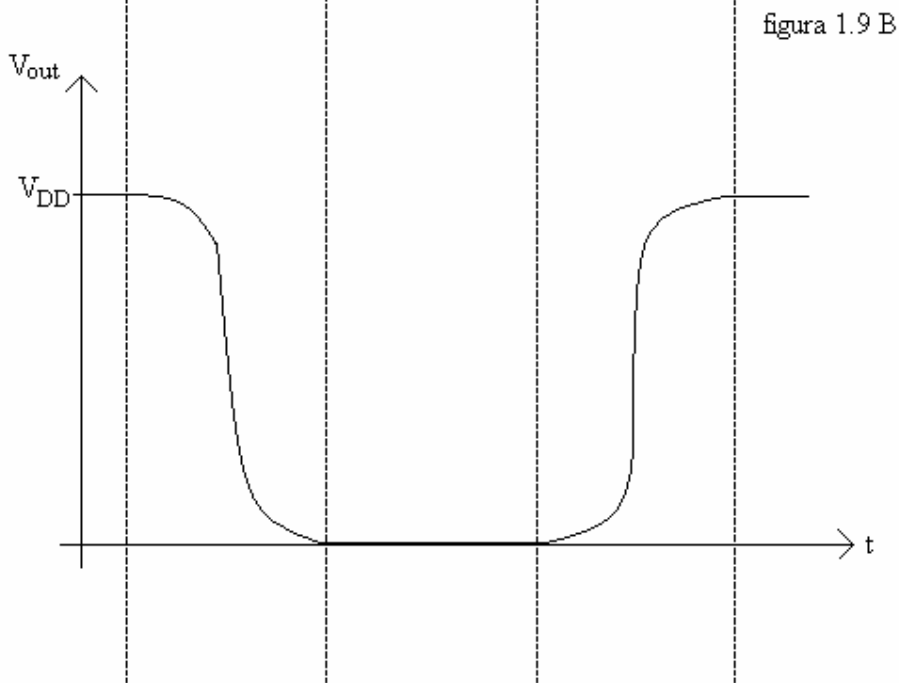
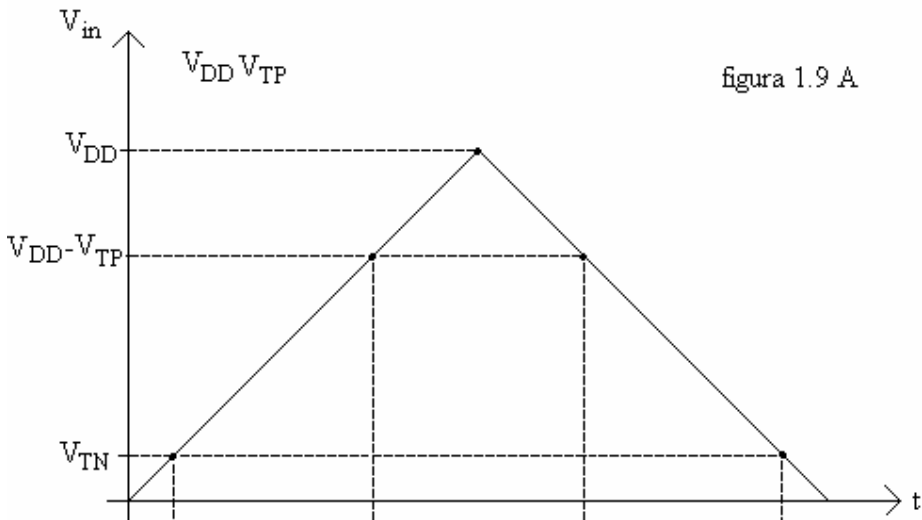
In prima approssimazione possiamo affermare che il CMOS conduce tra  $V_{DD}$  e massa solo in commutazione, ovvero nell'intervallo in cui  $V_{TN} < V_{IN} < V_{DD} - V_{TP}$ .

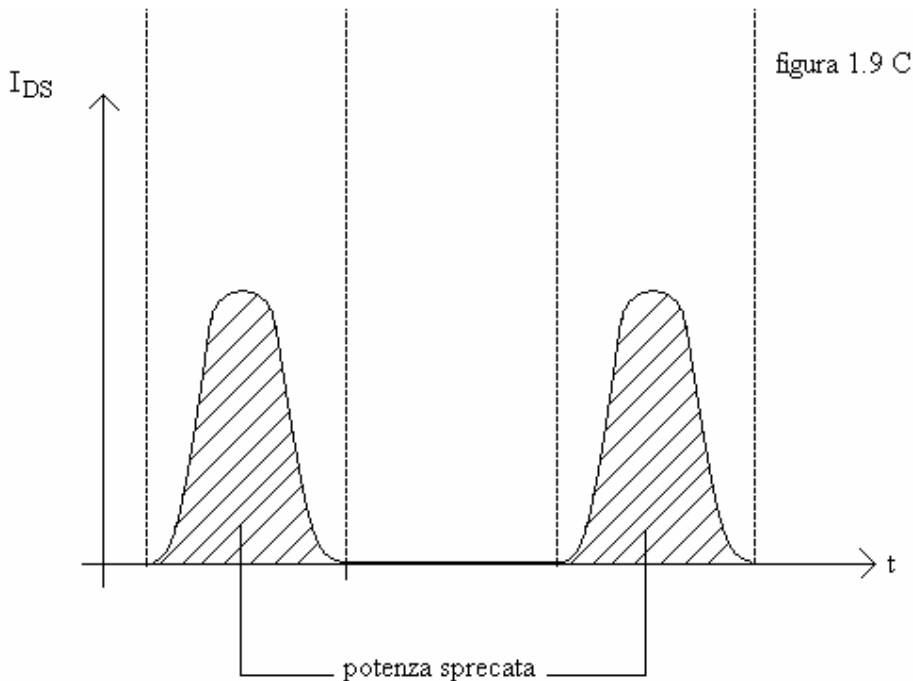
Porre  $V_{THINV} = V_{DD}/2$  è conveniente innanzi tutto perché la simmetria semplifica molte analisi, per esempio l'analisi dei ritardi (uguali in salita e discesa) e perché miglioriamo la robustezza al rumore.

La tensione di rumore può essere positiva o negativa in modo casuale e fa spostare la tensione  $V_{IN}$  rispettivamente a destra o a sinistra. Quindi ponendo la soglia a  $V_{DD}/2$  otteniamo il margine di rumore massimo.

## ANALISI NEL DOMINIO DEL TEMPO

1) VARIAZIONI MOLTO LENTE: la quantità di corrente che la capacità parassita ( $C_L$ ) richiede per seguire le variazioni della tensione è trascurabile rispetto alla corrente fornita dai transistor. Dato l'ingresso in figura 1.9 A, le figure 1.9 B-C mostrano l'andamento della tensione d'uscita e della corrente dei transistor in funzione del tempo, seguendo essenzialmente la caratteristica statica analizzata precedentemente.





Questo regime di funzionamento comporta due problemi fondamentali:

- a) si spreca troppa energia,
- b) si ha una elevata sensibilità al rumore nella zona intorno alla soglia.

2) **VARIAZIONI VELOCI**: la maggior parte della corrente serve a caricare o scaricare la capacità di carico.

Dato l'ingresso in figura 1.10A, le figure 1.10 B C e D mostrano l'andamento della tensione d'uscita e della corrente nei transistor in funzione del tempo, supponendo di essere in regime di variazioni veloci. In questo caso, uno dei due transistor si spegne quasi subito (il PMOS nel primo transistorio, l'NMOS nel secondo transistorio) e non si ha quasi consumo di corrente di corto circuito tra i due transistor.

L'andamento, in prima approssimazione, è inizialmente lineare in funzione del tempo, finché il transistor NMOS nel primo transistorio (o il transistor PMOS nel secondo transistorio) rimane in saturazione, in quanto corrisponde a una capacità scaricata a corrente costante nel primo transistorio (o caricata a corrente costante nel secondo transistorio).

Poi, quando il transistor che conduce (l'NMOS nel primo transistorio, il PMOS nel secondo) entra in zona lineare, l'andamento è esponenziale con costante di tempo  $R_{ON} C_L$ .

Per garantire tempi di salita e discesa uguali, che in generale consentono di realizzare un circuito ben equilibrato e di ottimizzare il caso peggiore, è necessario dimensionare NMOS e PMOS in modo che erogino circa la stessa corrente di saturazione e abbiano circa la stessa  $R_{ON}$ . In pratica, siccome in ogni generazione tecnologica quasi tutti i parametri dei transistor che determinano le correnti di saturazione e le resistenze sono circa uguali, ad eccezione delle mobilità, il rapporto tra la resistenza di un PMOS e un NMOS di dimensioni uguali è dato dal rapporto delle loro mobilità, chiamato  $\beta$ . Quindi per avere un inverter con tempi di salita e discesa uguali è necessario (siccome la lunghezza dei transistor è sempre la minima), avere un pull-up che sia  $\beta$  volte più largo del pull-down.

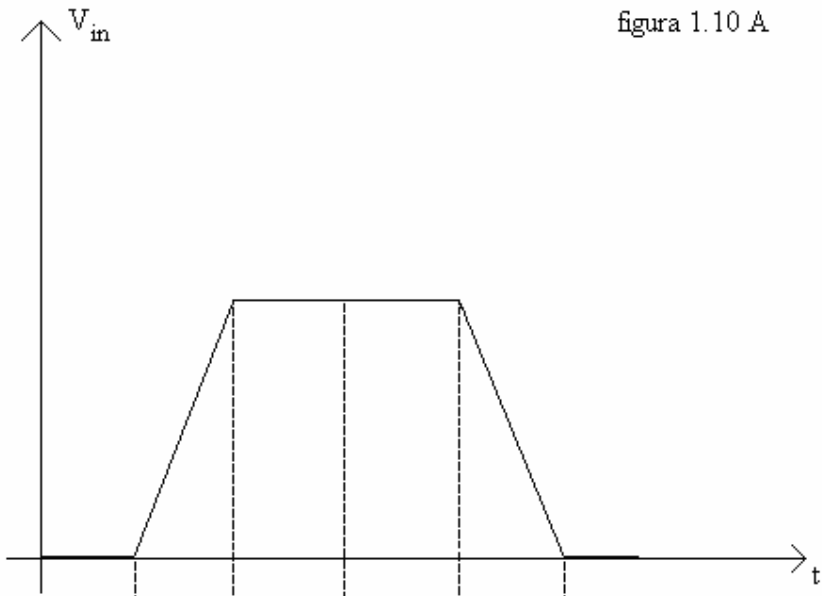


figura 1.10 A

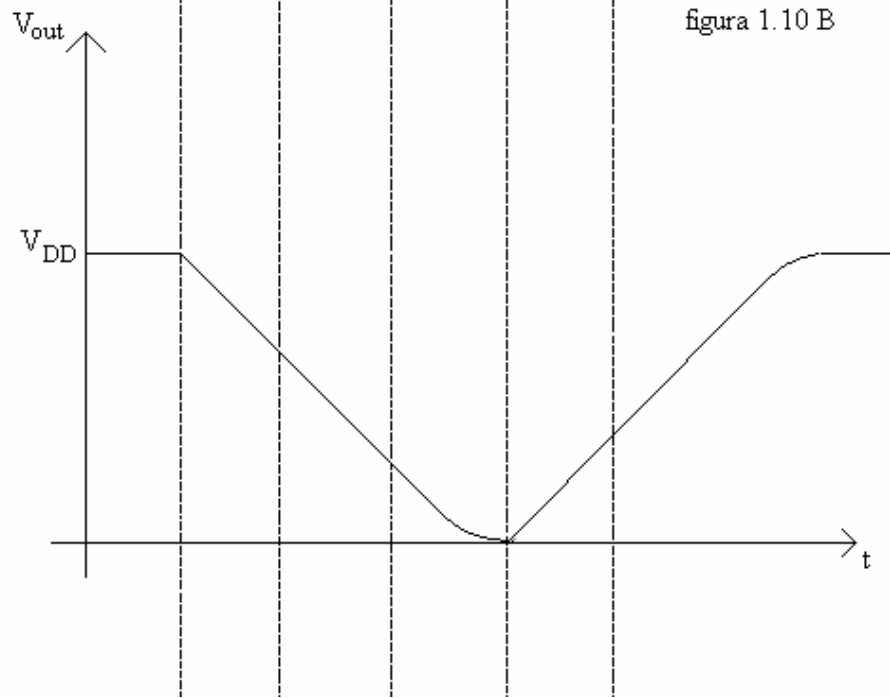
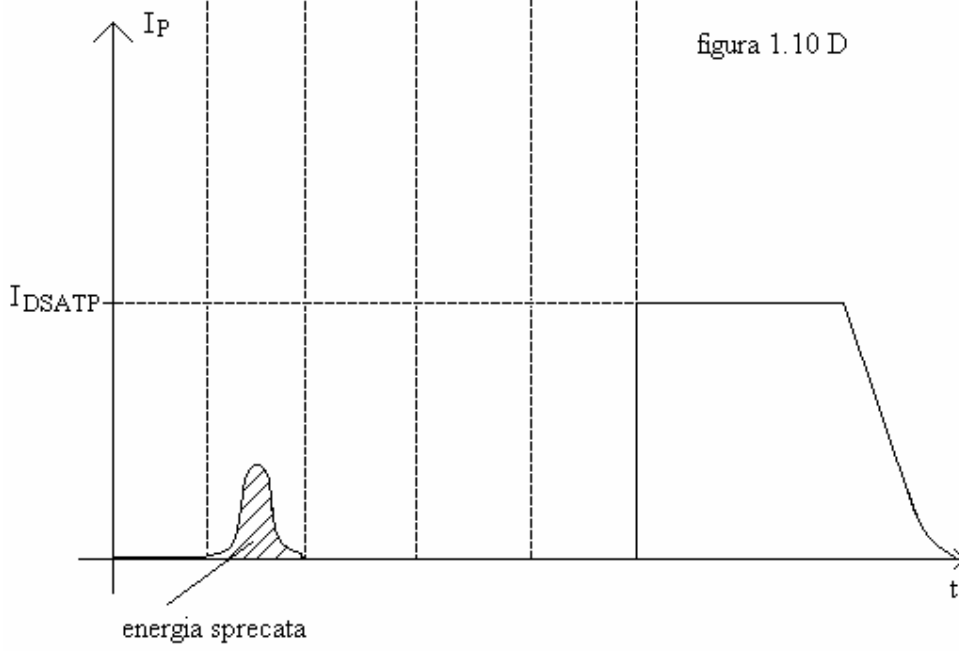
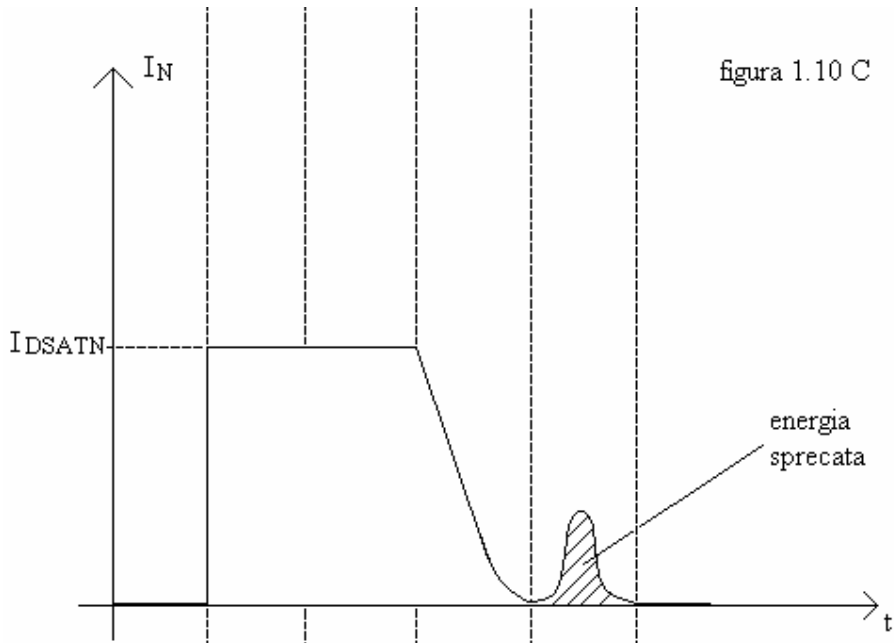


figura 1.10 B



## ANALISI DEI RITARDI NELLE INTERCONNESSIONI

Ogni componente possiede una capacità parassita che introduce ritardi nel circuito e una resistenza parassita che dissipa energia. Le induttanze nei circuiti integrati digitali attuali sono essenzialmente trascurabili (si notano solo nei collegamenti con package e circuito stampato).

In passato le resistenze dei fili di interconnessione venivano trascurate.

Attualmente le interconnessioni hanno un'influenza sul ritardo molto significativa rispetto ai ritardi delle port (in media del 50% entro blocchi di logica combinatoria di qualche centinaio di migliaia di porte, molto maggiori per le interconnessioni globali).

### Capacità parassite delle interconnessioni

In tecnologie "vecchie", la sezione delle interconnessioni era essenzialmente piatta. Quindi predominavano le capacità tra metallo e metallo, e tra metallo e substrato. La capacità tra metallo e substrato, più significativa per i livelli più bassi, era riducibile solo riducendo la lunghezza dell'interconnessione. La capacità tra metallo e metallo era riducibile anche riducendo la distanza per cui la coppia di interconnessioni di cui si vuole ridurre la capacità mutua viaggiano parallele.

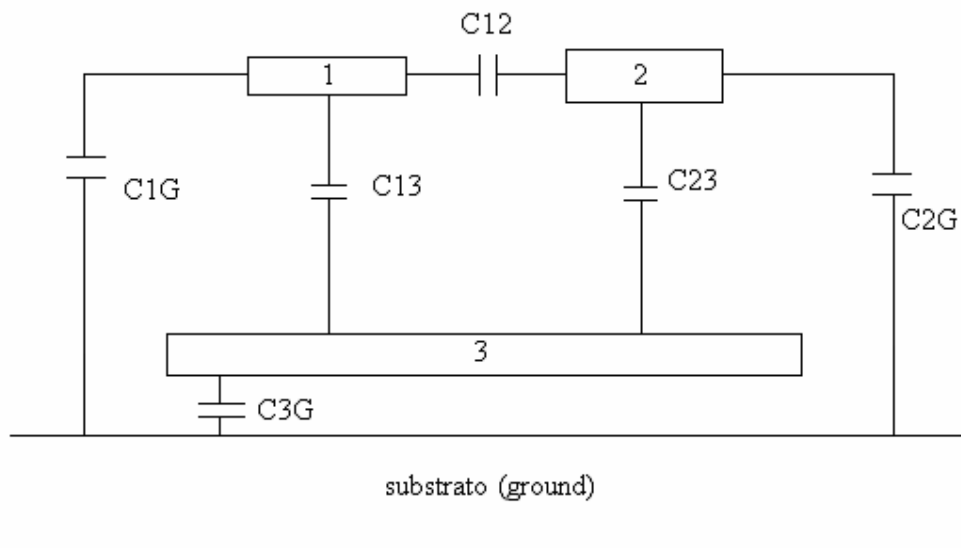


figura 1.24 - Capacità parassite ( vecchie tecnologie )

Con le nuove tecnologie, la sezione delle interconnessioni diventa sempre più verticale, allo scopo di mantenere una sezione accettabile, e quindi una resistenza bassa, nonostante il diminuire della

larghezza delle interconnessioni. Quindi la capacità all'interno di un layer, tra interconnessioni adiacenti, sta diventando dominante. Un ulteriore modo per ridurla è di aumentare la distanza tra interconnessioni parallele; di conseguenza se vogliamo realizzare un circuito più veloce abbiamo bisogno di più area.

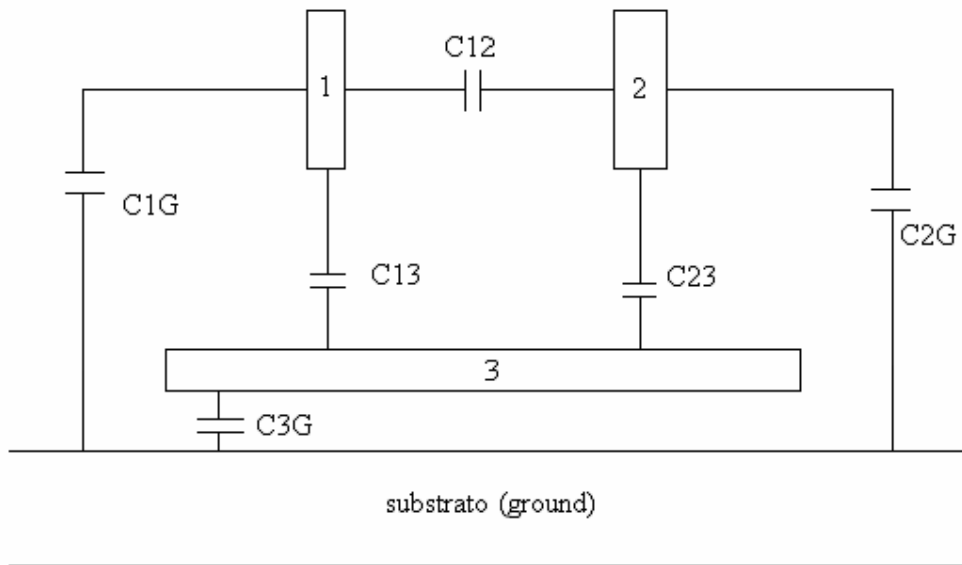


figura 1.25 - Capacità parassite (nuove tecnologie)

### Resistenze parassite delle interconnessioni

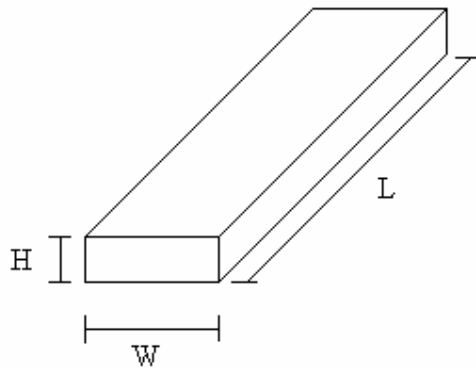


figura 1.11 C - Filo di interconnessione

$$R = \rho L / WH$$

Dove:

- L è la lunghezza del filo
- H è l'altezza del filo
- W è la larghezza del filo
- $\rho$  è la resistenza caratteristica del materiale.

L'altezza e la resistenza caratteristica non possono essere variate, perché dipendono dalla tecnologia. Quindi un progettista può variare solo i parametri L e W per avere una resistenza parassita bassa.

Una resistenza parassita bassa comporta:

- nei segnali una riduzione dei ritardi
- nelle alimentazioni una riduzione sia dei ritardi sia del rumore

Avere rumore sulle alimentazioni (anche detto "ground bounce") si traduce in un aumento del ritardo, in quanto una riduzione di VDD riduce la corrente nei MOS, ma soprattutto può comportare degli errori logici.

Supponiamo di avere il circuito mostrato in figura 1.12.

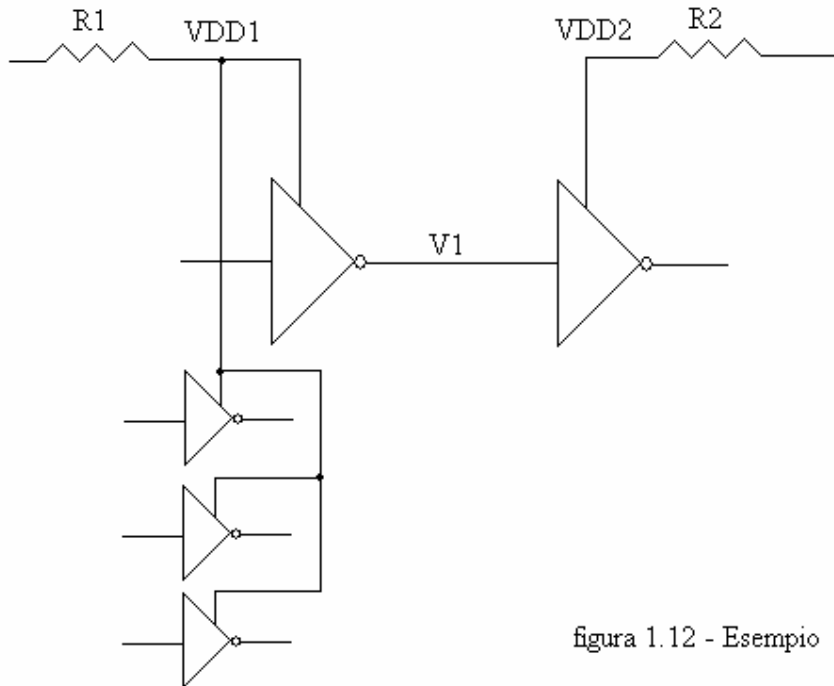


figura 1.12 - Esempio

A causa del fatto che VDD1 alimenta molti componenti, la corrente che circola sulla resistenza R1 sarà elevata.

Questo comporterà una diminuzione di VDD1 e, di conseguenza, anche della tensione V1(perché si comporta come VDD1).

Se V1 scende sotto la soglia dell'inverter successivo si ha un comportamento diverso del circuito rispetto a quello aspettato (errore logico).

Ecco perché è necessario mantenere bassa la resistenza parassita nelle alimentazioni.

### Calcolo della resistenza parassite delle interconnessioni

Supponiamo adesso di prendere un filo che abbia  $L=W$ .

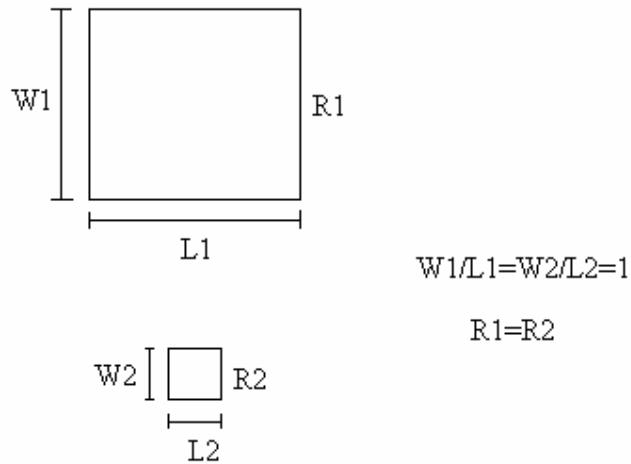


figura 1.13 - Fili di interconnessione con  $L=W$

Come mostra la figura 1.13, qualunque siano i valori di  $W$  e di  $L$  il loro rapporto è sempre pari ad 1; ne deriva che  $R1=R2$ . La resistenza in  $\Omega$  di un quadrato di materiale conduttore su un certo livello di interconnessione (diffusione, silicio policristallino, metallo) e' costante data la tecnologia. Dividendo un'interconnessione in quadrati (figura 1.14) è possibile calcolare la resistenza complessiva semplicemente moltiplicando il numero di quadrati per la resistenza tipica misurata in  $\Omega/\square$  (tabella 1).

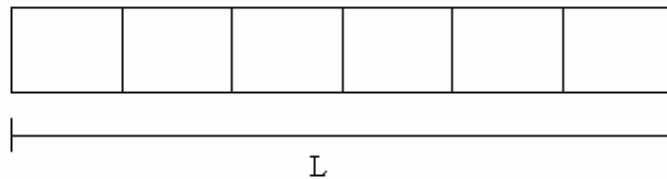


figura 1.14 - Suddivisione del filo in quadrati

Materiale	Resistenza $\Omega/\square$
Diffusione well p,n	1
Diffusione p+,n+	50-100
Silicio drogato p+,n+	150-200
Alluminio	0,05-0,1

### Effetti complessivi di capacità e resistenze parassite

Possiamo concludere dunque che, per quanto riguarda i segnali sia la capacità che la resistenza parassita influiscono negativamente.

Per le alimentazioni la resistenza parassita è dannosa (è per questo che bisogna cercare di allargare le interconnessioni di alimentazioni il più possibile) ma la capacità parassita influisce positivamente, in quanto fornisce una “carica locale” durante i transitori che contribuisce a ridurre gli effetti della resistenza parassita.

### MODELLI DI INTERCONNESSIONE

Prendendo in esame due diversi modelli delle interconnessioni, uno a parametri distribuiti e uno a parametri concentrati.

### Linea di trasmissione RC

In questo caso consideriamo resistenze e capacità infinitesime di segmenti di lunghezza infinitesima (figura 1.16).

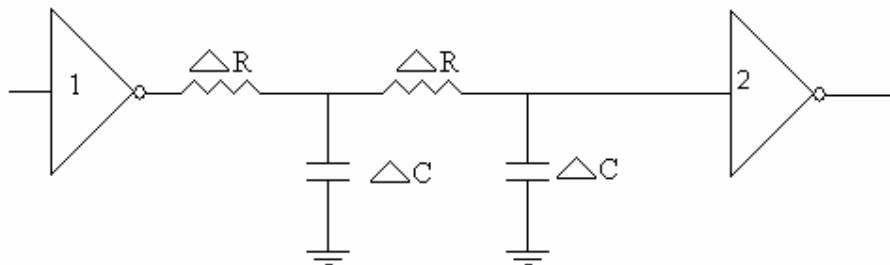


figura 1.16 - Linea di trasmissione RC

In questo caso il ritardo di propagazione di una linea senza diramazioni tra gli inverter 1 e 2, con capacità e resistenza costante, vale

$$\tau_{12} = rcL^2 / 2$$

dove:

- r è la resistenza/unità di lunghezza
- c è la capacità/unità di lunghezza

- L e' la lunghezza totale.

### Modello di Elmore

In questo caso semplifichiamo il modello, permettendoci di analizzare velocemente interconnessioni anche complesse e molto numerose, considerando capacità e resistenze su segmenti di lunghezza finita, abbastanza piccola da poter trascurare gli errori (figura 1.17).

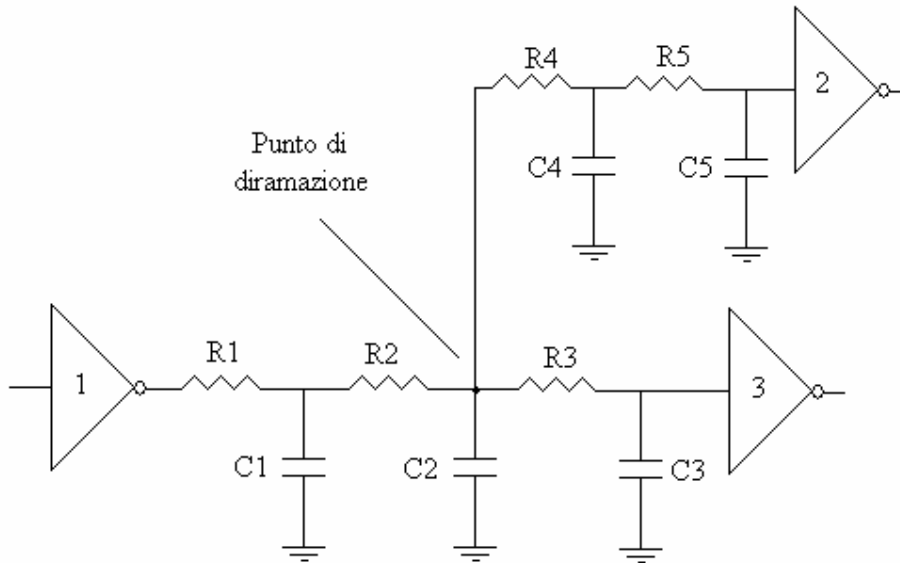


figura 1.17 - Modello di Elmore

$$\tau_{12} = R1(C1+C2+C4+C5) + R2(C2+C4+C5) + R4(C4+C5) + R5C5$$

$$\tau_{13} = R1(C1+C2+C3) + R2(C2+C3) + R3C3$$

Questo modello è più semplice del precedente ma discretizzando commettiamo un errore perché si sottostima il ritardo e, inoltre, nel calcolare  $\tau_{12}$  trascuriamo il tratto che va dal punto di diramazione al dispositivo 3.

## LOGICA CMOS STATICA

Nella logica CMOS statica, dopo aver esaurito i transistori, l'uscita è sempre e solo collegata alla tensione di alimentazione VDD o a massa.

Inoltre, sempre dopo aver esaurito i transistori, la tensione d'uscita corrisponde sempre al valore booleano della funzione implementata dal circuito.

Questa è la caratteristica che distingue la logica statica da quella dinamica dove l'uscita viene memorizzata su una capacità parassita; questo approccio consente di avere porte logiche più veloci ma un circuito dinamico è, come vedremo, più sensibile ai disturbi.

Una porta CMOS statica nasce dalla combinazione di due reti di transistori: una rete di pull-up (PU) e una di pull-down (PD).

Il PU ha lo scopo di connettere il nodo di uscita a VDD (quando l'uscita deve assumere il valore logico 1) mentre il PD connette l'uscita a massa (quando l'uscita deve assumere il valore logico 0). Le reti PD e PU sono costruite in modo tale che una e solo una delle due sia attiva.

La rete PD è costruita usando solo dispositivi NMOS mentre la rete PU usa dispositivi PMOS; la ragione di ciò nasce dal fatto che gli NMOS trasmettono bene il valore 0 (il valore 1 verrebbe trasmesso a meno di una  $V_t$ , quindi riducendo i margini di rumore e aumentando i ritardi), mentre i PMOS trasmettono bene il valore 1.

Due o più NMOS collegati in serie corrispondono alla funzione booleana AND (conducono se entrambi gli ingressi sono alti) mentre due o più NMOS in parallelo corrispondono alla funzione OR (figura 1.18). Inoltre, se sono collegati in ingresso allo zero logico, di fatto realizzano NAND or NOR rispettivamente (la serie porta l'uscita a zero se entrambi gli ingressi sono a uno, il parallelo se almeno uno degli ingressi è a uno).

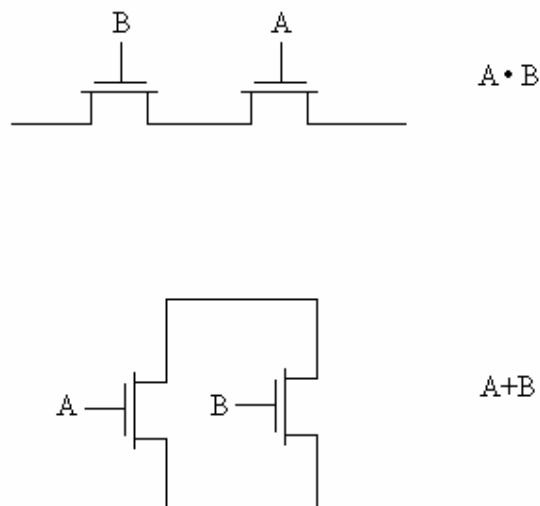


figura 1.18 - Funzioni booleane AND e OR

Due o più PMOS in serie corrispondono alla funzione NOR (conducono se entrambi gli ingressi sono a zero) mentre uno o più PMOS in parallelo implementano la funzione NAND.

Questo significa che per realizzare una porta CMOS basta costruire una delle due reti usando le connessioni serie e parallelo; l'altra rete può essere ottenuta applicando il principio di dualità logica (De Morgan) o grafica (scambiando serie e parallelo).

La porta logica così realizzata è invertente e si possono quindi realizzare, per esempio, le funzioni NAND e NOR, mentre non è possibile realizzare le AND e le OR utilizzando un singolo stadio complementare.

Il numero di transistor per una porta NAND o NOR a N ingressi è pari a 2N.

**Porta Nand a due ingressi**

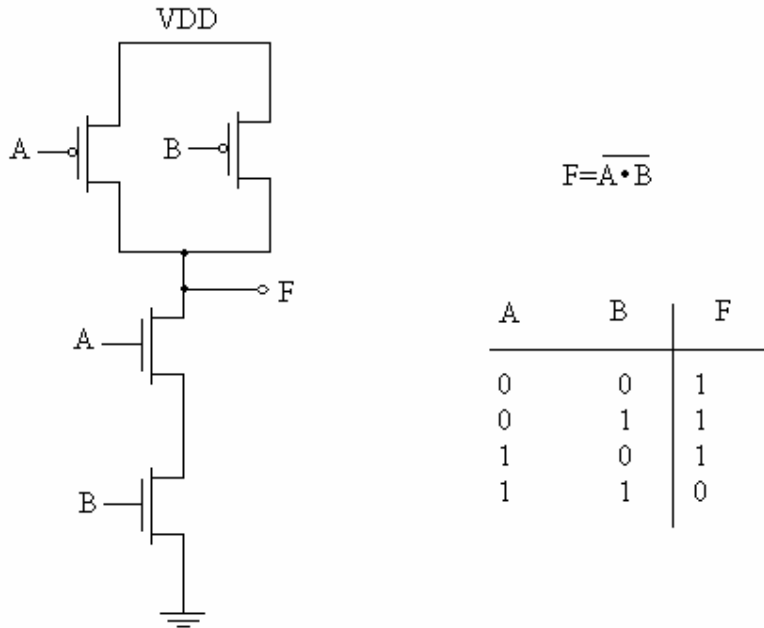


figura 1.19 - Porta NAND

La rete PD è costruita con la serie di due transistor NMOS e realizza il valore d’uscita 0 della funzione booleana A NAND B.

La rete PU è duale alla precedente, consiste nel parallelo di due transistor PMOS e realizza il valore d’uscita 1 della funzione booleana A NAND B.

Per garantire tempi di salita uguali nel caso peggiore si deve considerare:

- per il pull-down, il caso in cui entrambi conducono,
- per il pull-up, il caso in cui uno solo conduce (se entrambi conducono, il tempo di salita e’ approssimativamente dimezzato).

Supponiamo di voler ottenere la stessa resistenza o corrente di saturazione di un inverter in cui il PD ha larghezza W. La serie di due NMOS con larghezza W avra’ resistenza doppia di quella dell’inverter. Quindi e’ necessario raddoppiare la larghezza degli NMOS, scegliendola 2W.

L’unico PMOS attivo nel caso peggiore invece puo’ avere una larghezza  $\beta W$ . Quindi una porta NAND a 2 ingressi di dimensione minima avra’ gli NMOS di larghezza  $W_{min}$ , e i PMOS di larghezza  $W_{min} \beta/2$ .

E’ da notare come nell’inverter usare i pull-up e pull-down con un rapporto  $\beta$  tra le larghezze risultava sia nello stesso tempo di salita e discesa, sia nel porre la soglia della porta a  $V_{dd}/2$ . Nelle porte complesse lo stesso non e’ possibile, in quanto per la NAND la soglia dipende da quanti ingressi hanno il valore logico 1. La soglia del NAND e’ piu’ bassa quando un solo PMOS e’ acceso, in quanto deve avere una  $V_{GS}$  piu’ elevata, e quindi una  $V_{IN}$  piu’ bassa, per “contrastare” l’NMOS e far commutare l’uscita verso l’alto, mentre quando entrambi i PMOS sono accesi, basta una  $V_{GS}$  minore, e quindi una  $V_{IN}$  piu’ alta, per far commutare l’uscita verso l’alto.

Il discorso per la porta NOR e’ duale.

**Porta NOR a due ingressi**

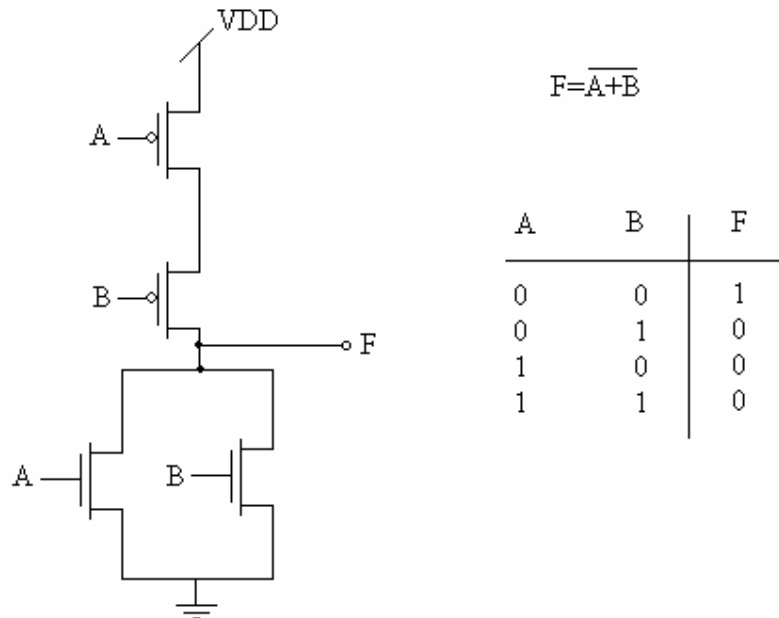


figura 1.20 - Porta NOR

La rete PD è costruita con due transistor NMOS in parallelo e realizza il valore d'uscita 0 della funzione booleana A NOR B.

La rete PU è duale rispetto alla PD, consiste in due transistor PMOS in serie, e realizza il valore d'uscita 1 della funzione booleana A NOR B.

**Sintesi di una porta logica CMOS arbitraria**

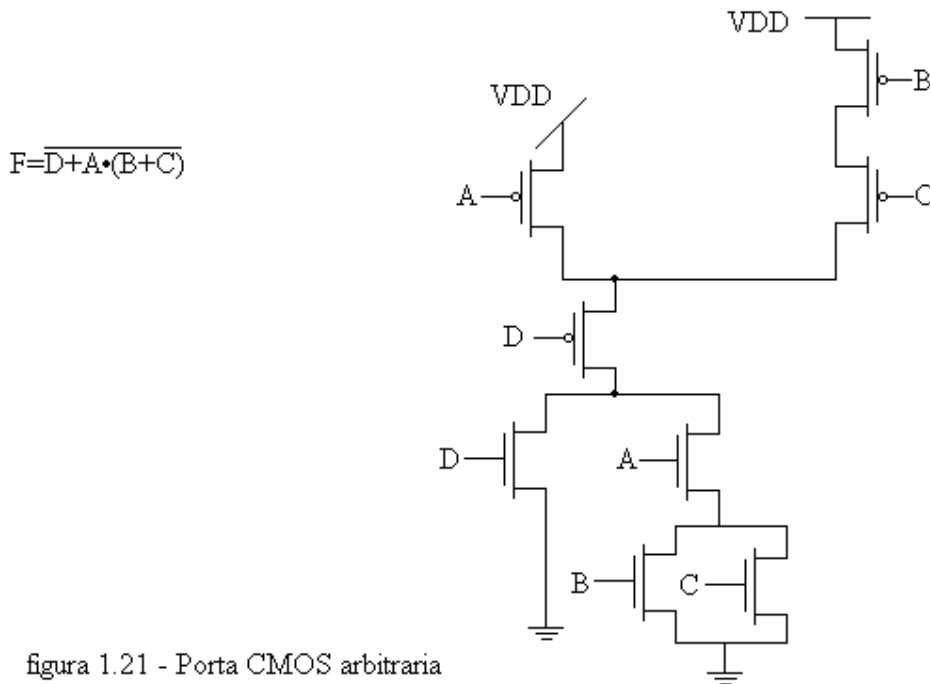


figura 1.21 - Porta CMOS arbitraria

La figura 1.21 mostra la porta logica che realizza la funzione booleana  $F = (D + A(B + C))'$ .

Inizialmente viene creata la rete di NMOS; I transistor B e C in parallelo realizzano la funzione  $(B+C)'$ , il parallelo tra B e C viene messo in serie con A per realizzare  $(A*(B+C))'$  e infine il tutto viene messo in parallelo con D per ottenere  $(D+A*(B+C))'$ .  
Successivamente viene realizzata la rete di PMOS applicando il principio di dualità.

### **PROBLEMI DELLE PORTE CMOS STATICHE**

La logica CMOS statica, seppur robusta e semplice da realizzare, comporta due problemi. In primo luogo il fatto che per realizzare un porta sono necessari  $2N$  transistor, di cui i pull-up sono in generale  $\beta$  volte piu' grandi dei pull-down a parita' di funzione logica realizzata, e quindi si ha un elevato consumo di area.

Un secondo problema è il tempo di propagazione che aumenta al crescere degli ingressi; nel caso peggiore il tempo di propagazione di una porta con transistor in serie cresce, per il modello di Elmore, con il quadrato del numero di ingressi (o fan-in) della porta.

### **TECNICHE PER LA PROGETTAZIONE DI PORTE CMOS CON FAN-IN ELEVATO**

#### **Dimensionamento dei transistor**

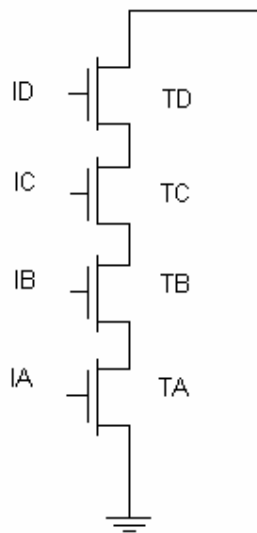
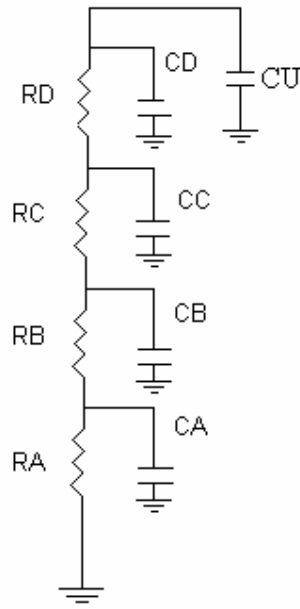


figura 1.22 - Porta NAND a 4 ingressi



1.23 - Schema fisico NAND a 4 ingressi

La figura 1.22 mostra la rete di pull-down (in questo caso la piu' critica dal punto di vista dei ritardi, in quanto presenta transistor in serie) di una porta NAND a 4 ingressi, il cui corrispondente modello elettrico semplificato RC è mostrato in figura 1.23. In questo tipo di modello si rappresenta la transizione in uscita non con il modello esatto, lineare in zona di saturazione e esponenziale in zona lineare del transistor, con un'esponenziale dato da una resistenza "equivalente" che, quando pilota la stessa capacita' di carico, risulta nello stesso tempo di propagazione (definito come la distanza temporale tra quando l'ingresso supera  $V_{dd}/2$  e quando l'uscita supera  $V_{dd}/2$ ).

Il tempo di propagazione della porta logica NAND a 4 ingressi, calcolato con l'equazione di Elmore, risulta essere:

$$\tau = RA(CA+CB+CC+CD+CU)+RB(CB+CC+CD+CU)+RC(CC+CD+CU)+RD(CD+CU)$$

Per ridurre il tempo di propagazione al massimo non serve usare un dimensionamento uniforme, ovvero aumentare le dimensioni di tutti transistor per diminuire le resistenze dei dispositivi e le relative costanti di tempo, perche' in questo modo si introducono capacita' parassite maggiori che influenzano negativamente il tempo di propagazione delle porte precedenti.

La soluzione è quella del dimensionamento progressivo: come si può notare, la resistenza RA compare N volte nel tempo di ritardo, la resistenza RB compare N-1 volte e così via.

Quindi bisogna fare in modo che RA sia la resistenza più piccola (perché compare più volte), la resistenza RB la successiva e via discorrendo.

Intuitivamente, il transistor TA che deve fare piu' "lavoro" viene reso piu' largo e quindi conduce meglio.

## Ordinamento dei transistor nelle porte logiche

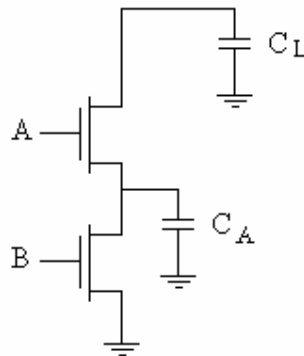


figura 1.24 - Porta logica NAND a due ingressi (si è trascurata la rete di pull-up)

Prendiamo ad esempio il circuito in figura 1.24, dove come al solito ci concentriamo sui transistor in serie.

Supponiamo inizialmente che l'ultimo segnale a stabilizzare (cioè a passare a livello logico 1) sia B. Il segnale A era già stabilizzato quindi le capacità  $C_L$  e  $C_A$  sono cariche.

Quindi quando B si stabilizza, il transistore corrispondente deve scaricare due capacità facendo più lavoro.

Se invece supponiamo che sia A a stabilizzare per ultimo, quando quest'ultimo arriva  $C_A$  è già scarica quindi c'è meno lavoro da fare.

La regola generale è dunque quella di mettere più in alto il segnale che stabilizza per ultimo.