

Master II livello 2006

Problematiche di simulazione di circuiti integrati complessi

- 1) Progettazione di circuiti integrati
- 2) Verifica elettrica di un circuito integrato
 - 2.1) il processo
 - 2.2) la tensione di alimentazione
 - 2.3) la temperatura
- 3) Verifica elettrica di un circuito integrato
 - 3.1) verifica mono-variabile e topologia concentrata
 - problematiche (precisione, carichi)
 - esempi
 - 3.2) verifica multi-variabile e topologia distribuita
 - problematiche (precisione, interconnessioni, back-annotazione)
 - esempi
 - 3.3) verifica di funzionalita' elettrica globale
 - problematiche di precisione, velocita' di esecuzione, modellizzazione e semplificazione di netlist
- 4) back annotazione
- 5) simulatori elettrici SPICE e HSIM
- 6) Verifica logica di un circuito integrato
 - 6.1) stimoli e interattivita' (stimoli su condizione e verifica automatica)
 - 6.2) modellizzazione di circuiti analogici
- 7) Simulazioni mixed mode : cosim e verilog-A

Naso Giovanni



in partnership with

Università degli Studi dell'Aquila



1) Progettazione di circuiti integrati

La progettazione di un circuito integrato richiede una operazione di 'virtualizzazione' della realta' attraverso l'uso di modelli fisici e comportamentali che permettono di descrivere il comportamento del circuito integrato (fisiologia).

L'aumentare della complessita' dei circuiti integrati introduce degli elementi che non possono essere trascurati se non si vuole incorrere in errori di 'eccesso di virtualizzazione' (patologia).

Esempi di tali elementi sono :

- contemporanea presenza di circuiti analogici e digitali
- contemporanea presenza di blocchi sincroni e asincroni
- contemporanea presenza di blocchi a bassa e alta tensione
- presenza di parassiti resistivi e capacitivi in topologie distribuite
- accoppiamenti induttivi e capacitivi in strutture a scala di integrazione molto spinta

Una corretta metodologia di verifica deve tener quindi conto del fatto che la virtualizzazione della realta' non e' un processo automatico ma richiede una opportuna integrazione di concetti che appartengono sia all'area della fisiologia che della patologia.



1) Progettazione di circuiti integrati (cont.)

- "Fisiologia"

R. Jacob Baker, Harry W. Li, David E. Boyce
CMOS circuit design, Layout and Simulation
IEEE press 1997

- "Patologia"

Masakazu Shoji
Theory of CMOS digital circuits and circuit failures
Princeton University Press 1992

- "Metodologia di verifica" (simulazioni)

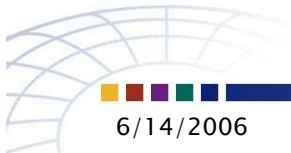
Tecniche numeriche nei simulatori elettrici :
T.L. Pillage, R.A. Rohrer, C. Visweswariah
Electronic Circuit & System Simulation Methods
McGraw-Hill 1994

Verilog :

S. Palnitkar
Verilog HDL - A Guide to Digital Design and Synthesis
SunSoft Press (A Prentice Hall Title)

Mixed mode (Verilog-A) :

D. Fitzpatrick, I. Miller
Analog Behavioral Modeling With The Verilog-A Language
Kluwer Academic Publishing. 1998



1) Progettazione dei circuiti integrati (cont.)

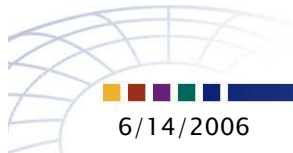
Gli esempi riportati nel seguito di questa dispensa sono citati solo per rendere piu' chiari alcuni concetti relativi alla "metodologia di verifica". In essi non vi sono quindi approfondimenti teorici ne' riferimenti a prodotti o tecnologie specifiche.

Gli esempi riportati, pur se a volte sono molto specifici del design di memorie, permettono di trarre spunti di validita' generale e sono frutto del lavoro di colleghi del dipartimento R&D di Micron Italia. In particolare :

- Girolamo Gallo per valutazioni relative ad output buffer
- Ercole Di Iorio per valutazioni relative a caratterizzazione della pompa v5
- Michele Incarnati per valutazioni di sovratensioni di tub e analisi dei margini
- Stefano Perugini per valutazioni relative alla frequenza di burst machine

Gli altri esempi riportati sono stati da me personalmente curati :

- caratterizzazione di un one shot
- caratterizzazione di VIH/VIL
- accesso diretto di bit line
- valutazione di accoppiamento
- ground bouncing
- accesso da CE_



2) Verifica elettrica di un circuito integrato

Il comportamento elettrico di un circuito integrato in simulazione e' influenzato da:

- processo
- tensione di alimentazione
- temperatura
- layout (parassiti)



6/14/2006



2.1) Il processo

Il processo di fabbricazione di un circuito integrato CMOS ha delle variazioni nella realizzazione della densita' di drogaggio, profondita' di diffusione (energia/bake), geometrie di mascheramento (litografia/etch) che si riflettono in variazioni nei parametri elettrici.

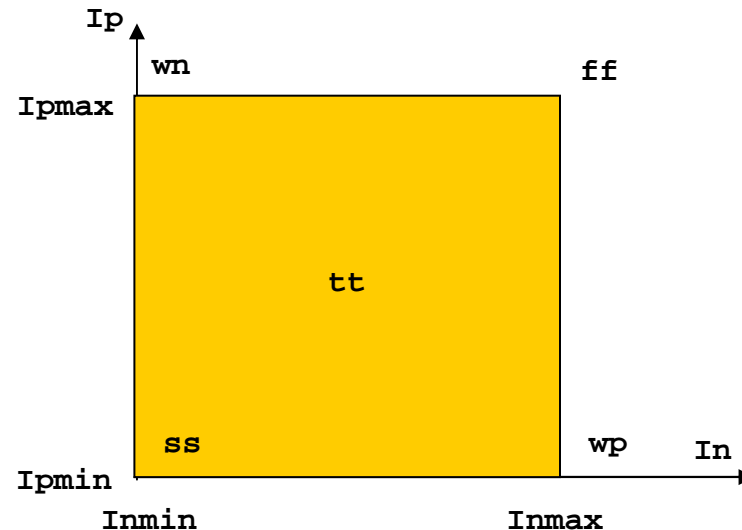
	slow	typical	fast
In Nch Idrive (uA/um)	Inmin	Intyp	Inmax
Ip Pch Idrive (uA/um)	Ipmin	Iptyp	Ipmax
msr metal sheet resistance (ohms/square)	msrmin	msrtyp	msrmax
ctr contact resistance (ohms)	ctrmin	ctrtyp	ctrmax

Per quanto riguarda i transistori nch/pch, il punto in cui puo' collocarsi uno specifico wafer e' all'interno di un quadrilatero definito da quattro punti :

ss : (Inmin;Ipmin) slow nch/slow pch
 ff : (Inmax;Ipmax) fast nch/fast pch
 wp : (Inmax;Ipmin) fast nch/slow pch
 wn : (Inmin;Ipmax) slow nch/fast pch

Il processo tt : (Intyp;Iptyp) sara' collocato in posizione centrale nel quadrilatero di processo.

Per quanto riguarda i transistors sono quindi forniti 5 modelli : ss, ff, wp, wn, tt che verranno opportunamente usati a seconda delle situazioni. Non e' quindi possibile girare simulazioni in situazioni diverse dai 5 punti.



2.1) Il processo (cont.)

Per valutare un ritardo di propagazione e vedere se si rientra in un valore massimo di specifica, si adottera' un modello ss in modo tale da valutare il caso piu' lento di propagazione.

Per valutare un hold time si usera' ff per vedere se comunque un certo segnale e' garantito permanere ad un certo valore per almeno un tempo specificato.

Per valutare se un pull-up di precarica puo' essere vinto da una catena di pull-down di segnale si usera' un wn.

Per valutare quanto uno specifico circuito integrato rispecchia le simulazioni si puo' procedere nel seguente modo :

- si misura In e Ip in strutture di test collocate nel wafer quanto piu' possibile in prossimita' allo specifico circuito integrato (chip) che si vuole valutare.
- si traccia il punto (In;Ip) nel piano di processo
- si valuta quale vertice nel piano di processo e' piu' vicino al punto (In;Ip)
- si considerano le simulazioni relative a quel vertice e si confrontano I risultati con i valori misurati nel circuito integrato

Tanto piu' il punto di processo si trova vicino ad un vertice, tanto piu' ci si aspetta che le misure siano prossime alle simulazioni relative a quel vertice (a parita' di temperatura e alimentazione).

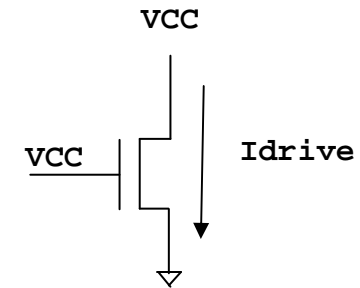
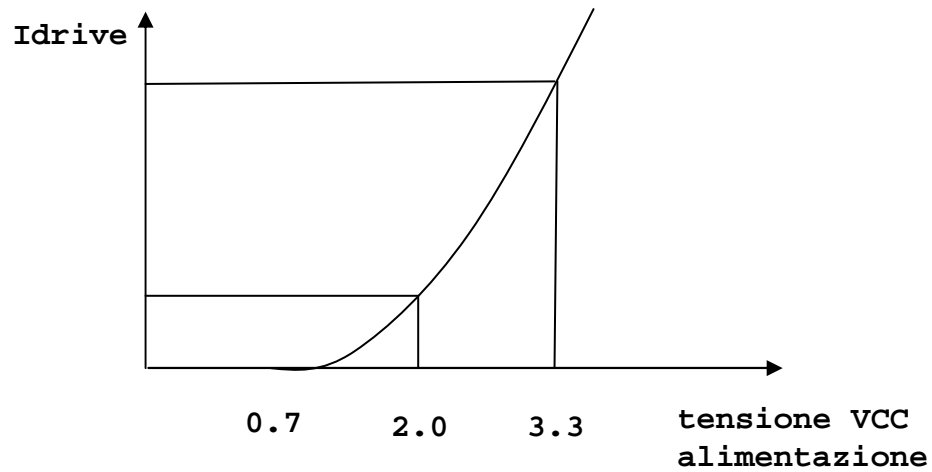
Un modo differente e piu' preciso di associare modelli a processo e' quello adottato in TI spice3 (Texas Instruments). In TI spice3 viene fornito il solo modello nn e 2 parametri PNSIG e PPSIG che permettono di girare simulazioni per uno qualsiasi dei punti all'interno del quadrilatero di processo.

Dato uno specifico chip su un wafer si calcola PNSIG e PPSIG ad esso associati in base a misure di drive current su strutture di test ad esso piu' vicine. Si gira poi una simulazione con la specifica coppia di PNSIG e PPSIG ed il risultato, a parita' di temperatura e tensione di alimentazione, e' esattamente quello che ci si deve attendere dal silicio.



2.2) la tensione di alimentazione

La tensione di alimentazione influenza le prestazioni di un transistor perche' la I_{drive} di un transistor aumenta all'aumentare della tensione di alimentazione (punto di lavoro).



2.2) la tensione di alimentazione (cont.)

In genere all'aumentare della tensione di alimentazione la velocità di propagazione in un circuito integrato aumenta.

Se quindi si vogliono valutare le prestazioni in velocità di un circuito integrato si adottano le seguenti condizioni:

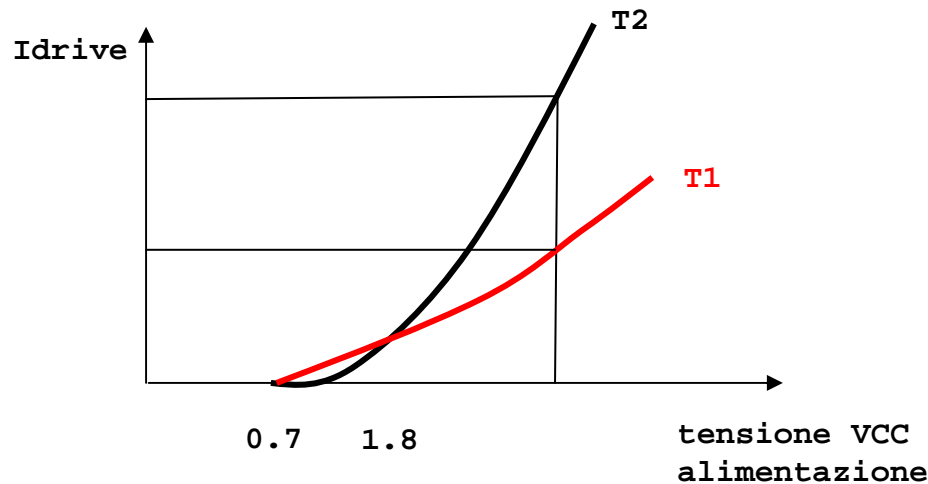
propagazione veloce	:	processo ff, alta VCC
propagazione lenta	:	processo ss, bassa VCC



2.3) la temperatura

La temperatura influenza in vario modo le caratteristiche dei componenti nei circuiti integrati :

- 1) all'aumentare della temperatura aumenta il valore delle resistenze di interconnessione
- 2) All'aumentare della temperatura diminuisce la mobilita' (e quindi il k) di un transistor. L'effetto e' riportato in una tipica curva IV



$T1 > T2$

all'aumentare della temperatura diminuisce l' I_{drive} di un transistor alimentato a tensioni superiori a 1.8V

Se quindi si vogliono valutare le prestazioni in velocita' di un circuito integrato si adottano le seguenti condizioni:

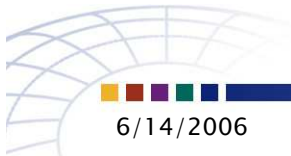
propagazione veloce	:	processo ff, alta VCC bassa T (se $V_{CC} > 1.8V$); alta T (se $V_{CC} < 1.8V$)
propagazione lenta	:	processo ss, bassa VCC alta T (se $V_{CC} > 1.8V$); bassa T (se $V_{CC} < 1.8V$)



3) Verifica elettrica di un circuito integrato

La verifica elettrica di un circuito integrato puo' suddividersi in 3 grandi categorie per quanto riguarda gli scopi, la complessita' dei circuiti e l'uso di diversi simulatori :

- a) valutazione di grandezze associate ad un singolo parametro e topologia concentrata
- in questo caso si tratta di valutare il valore di una singola grandezza (per esempio un rise time) associata ad un singolo parametro (per esempio la larghezza WP di un transistor) in diverse condizioni di processo, tensione di alimentazione e temperatura.
 - in genere il circuito da simulare e' molto limitato nel numero di componenti (decine o centinaia di componenti) e puo' essere considerato con buona approssimazione indipendente da tutti gli altri circuiti (topologia concentrata; per esempio un output buffer)
 - l'interazione con il resto del sistema puo' essere semplificata con l'uso di un semplice modello di carico (per esempio RC da pilotare)
 - in questi casi il tempo di simulazione e' molto breve e la precisione richiesta e' elevata (circuiti di tipo analogico)
 - il simulatore da usare e' HSPICE che tiene conto di tutti gli aspetti di modellistica dei componenti (tutte le capacita' di giunzione e di ossido, body bias, correnti di subthreshold, effetto Miller)



3) Verifica elettrica di un circuito integrato (cont.)

b) valutazione di grandezze associate a diversi parametri e topologie distribuite

- in questo caso si tratta di valutare il valore di una o piu' grandezze (per esempio un tempo di accesso da CE) associata a piu' parametri contemporaneamente (per esempio tensione di soglia di una cella FLASH e durata della precarica di una bit line e Idrive della scarica e istante di sense) in diverse condizioni di processo, tensione di alimentazione, temperatura.
- in genere il circuito da simulare puo' essere semplificato da un numero limitato di circuiti concentrati interconnessi tra di loro con modelli di propagazione RC (per esempio speed path). La netlist che ne risulta e' di dimensioni medio/grandi dell'ordine di migliaia di componenti.
- a volte il numero di interconnessioni e' cosi' elevato che non si puo' pensare di effettuare una modellistica manuale. In questi casi si procede ad una prima valutazione con interconnessioni modellate manualmente e quando il layout e' disponibile si procede ad una operazione di back-annotazione.
- l'uso di simulatori precisi come HSPICE puo' portare in questi casi a simulazioni che durano un tempo eccessivamente lungo. Si procede in questi casi all'uso di simulatori quali HSIM che permettono di effettuare una scelta di compromesso tra precisione e velocita'.



3) Verifica elettrica di un circuito integrato (cont.)

c) valutazione di comportamento di un intero chip

- in questo caso si tratta di valutare il comportamento elettrico di un intero circuito integrato (per esempio una operazione di programmazione in una memoria FLASH che richiede l'esecuzione di un algoritmo, la presenza di parti analogiche e un comportamento adattativo).
- il circuito da simulare sarebbe eccessivamente grande (milioni di componenti) se non si adottassero delle tecniche di semplificazione che lo riducono ad un numero di componenti dell'ordine di qualche centinaio di migliaia. In questo caso la semplificazione viene operata eliminando le parti ripetitive di circuiti che non interessa selezionare (per esempio eliminare tutti i blocchi di una memoria e lasciarne solo alcuni) e che vengono sostituiti da parassiti.
- anche dopo la semplificazione il circuito potrebbe essere comunque oneroso in termini di tempo di simulazione. In questo caso si puo' procedere a tecniche di modellizzazione di circuiti analogici che tipicamente richiedono lunghi tempi di simulazione. (per esempio modelli di Thevenin per generatori di tensione).
- il simulatore da usare in questi casi e' senza dubbio HSIM che permette di abbattere i tempi di simulazione perche' e' basato su tecniche di partizionamento delle netlist, e' un simulatore event driven con diversa sensibilita' agli eventi, permette di lavorare su modelli di transistor a diversi gradi di semplificazione (da modelli HSPICE fino a tabelle IV).
- con una certa accortezza, HSIM puo' essere portato a lavorare da un'area a descrizione prettamente analogica ad un'area a descrizione essenzialmente digitale.
- la back-annotazione (eventualmente parziale) e' spesso usata quando si valuta il comportamento elettrico di un intero chip.



3.1) Esempi di verifica elettrica mono-variabile e topologia concentrata

- a) valutazione di rise/fall time e ringing in un output buffer
- b) valutazione di 'spread' di parametri quali ritardi, durate di impulsi, generazione tensioni (al variare di processo, alimentazione, temperatura, trim)
- c) soddisfacimento di VIH/VIL
- d) test mode di accesso a bit lines
- e) dimensionamento componenti e/o percorsi e/o distanze per rendere minimo il cross-coupling
- f) curve di carico per charge pumps

3.2) Esempi di verifica elettrica multi-variabile e topologia distribuita

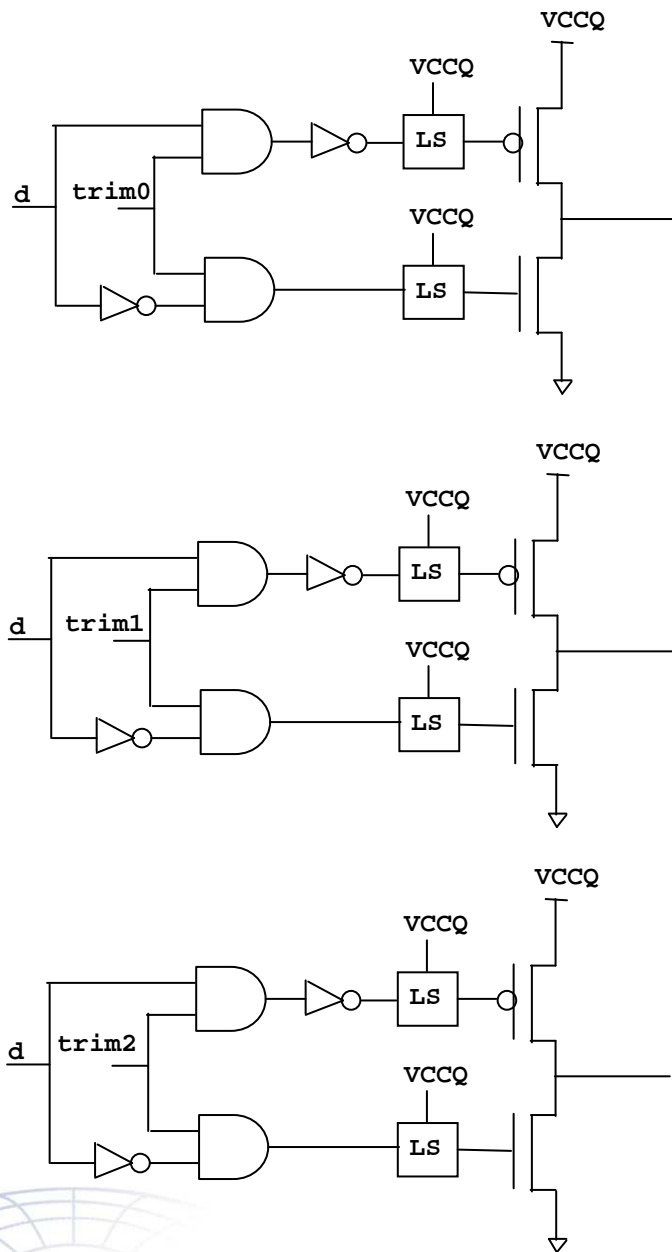
- a) valutazioni delle connessioni di gnd per rendere minimo il 'bouncing' in output buffers
- b) calcolo di tempi di accesso da address e da CE
- c) valutazione di accoppiamento di segnali con diversa drive strength dovuto a topologia distribuita e layout automatico
- d) valutazione efficacia di ground in carichi estremamente distribuiti (tub)
- e) analisi dei margini in una operazione di lettura di una memoria FLASH
- f) frequenza di funzionamento di una burst machine

3.3) Esempi di verifica elettrica di un intero chip

- a) algoritmo di lettura (scrittura) in una memoria FLASH



3.1.a) Valutazione rise/fall time e ringing in un output buffer



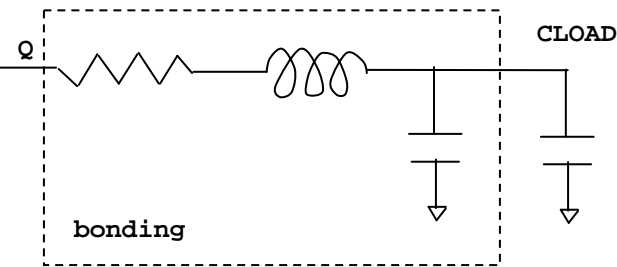
Si consideri un output buffer costituito da 3 stadi CMOS selezionabili da fusibili di trim (trim0, trim1, trim2) connessi ad una tensione di alimentazione esterna VCCQ in genere differente dalla tensione di alimentazione interna VCC.

E' importante valutare per questo circuito i tempi di salita e discesa (rise/fall) della uscita Q in funzione della configurazione di trim e a diverse condizioni di processo, alimentazione e temperatura.

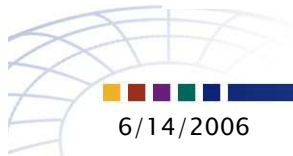
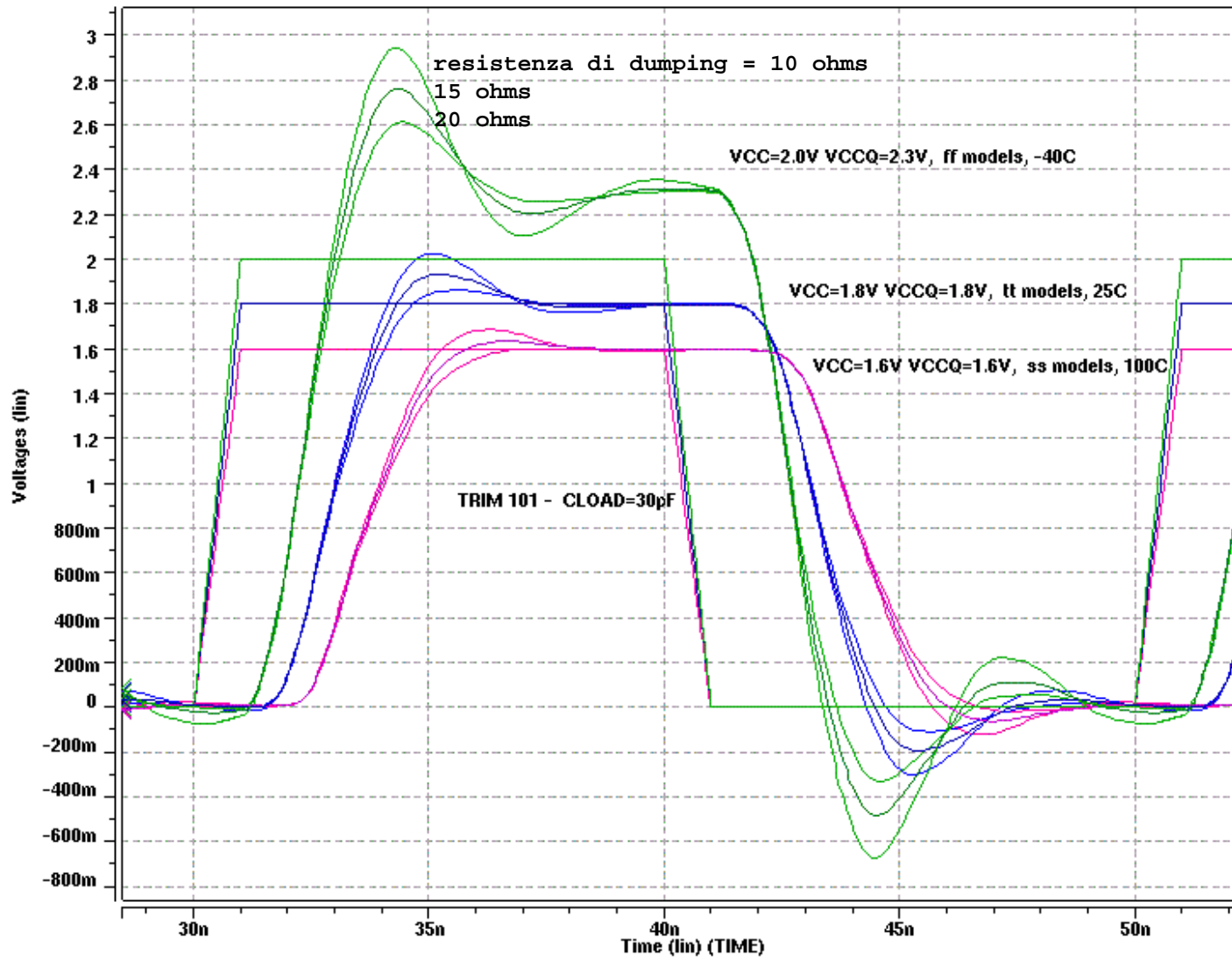
E' anche importante valutare gli effetti di oscillazione e diversi valori della resistenza di dumping che influiscono sull'ampiezza delle oscillazioni.

Questo tipo di circuiti ha un numero limitato di componenti e possono essere esaminati separatamente dal resto del circuito integrato pur di tenere in conto un modello di bonding e il valore del carico.

resistenza
di dumping



3.1.a) Valutazione rise/fall time e ringing in un output buffer (cont.)



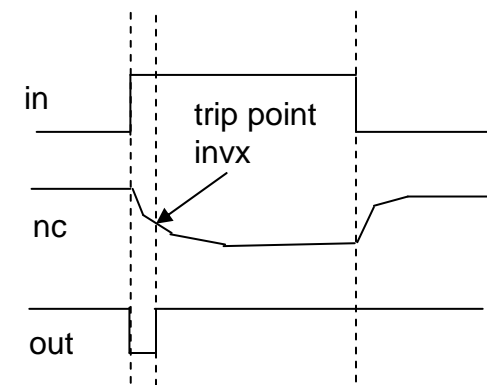
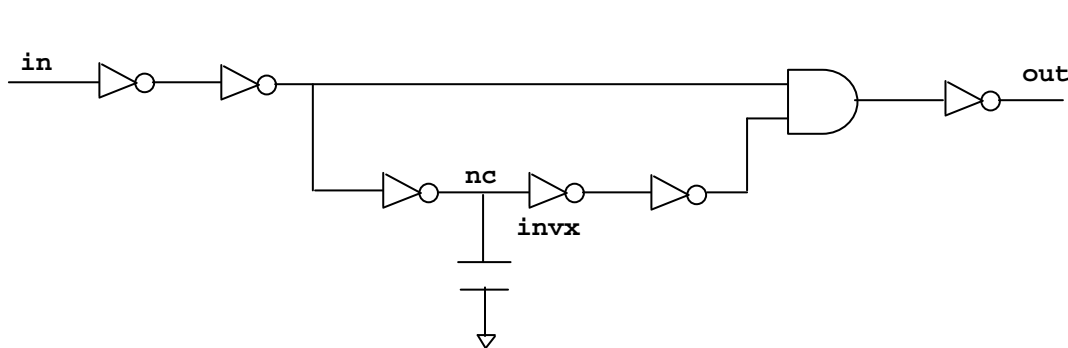
3.1.b) Valutazione di 'spread' di impulso

Si consideri un generatore di impulso (oshi) che genera un impulso negativo in uscita (out) in corrispondenza di un fronte positivo in ingresso (in). Tale circuito puo' essere usato all'ingresso di un flip-flop (ff) set-reset.

E' importante valutare per questo circuito la durata dell'impulso al variare del processo, alimentazione e temperatura in modo da essere sicuri che sia abbastanza largo da far commutare il ff ma non troppo da invadere le due aree di funzionamento di set e reset.

In caso si voglia aumentare o diminuire tale valore basta aumentare o diminuire il valore della capacita' inserita nel nodo di carica nc.

Questo tipo di circuiti ha un numero molto limitato di componenti e possono essere esaminati separatamente dal resto del circuito integrato.



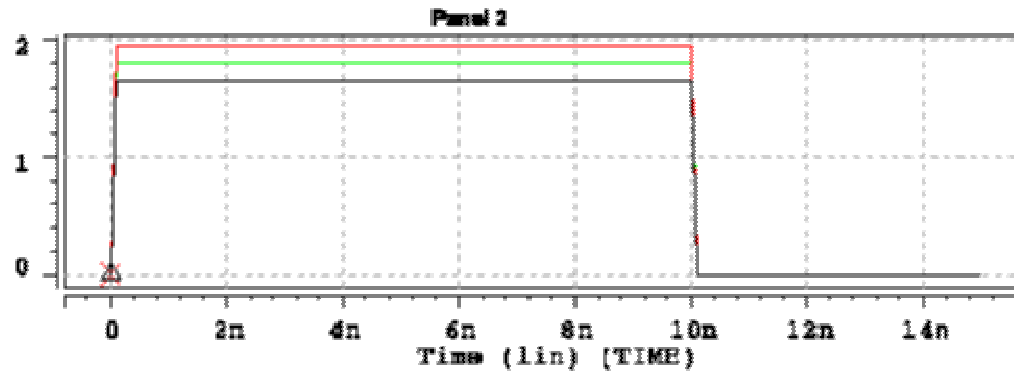
3.1.b) Valutazione di 'spread' di impulso (cont.)

Rosso : ff - VCC=1.9V - T=-40C / Verde : tt - VCC=1.8V - T= 25C / Nero : ss - VCC=1.6V - T= 90C

Wave	Symbol
D0:A0:v(in)	X
D1:A0:v(in)	X
D2:A0:v(in)	△

in

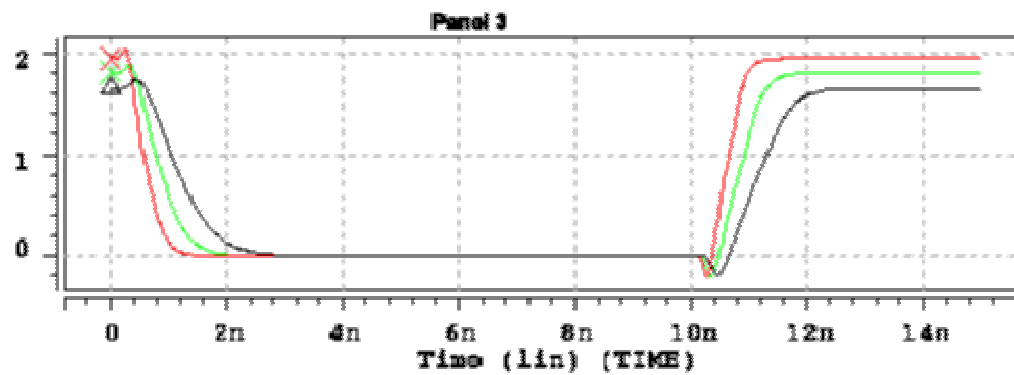
Voltages (lin)



Wave	Symbol
D0:A0:v(nc)	X
D1:A0:v(nc)	X
D2:A0:v(nc)	△

nc

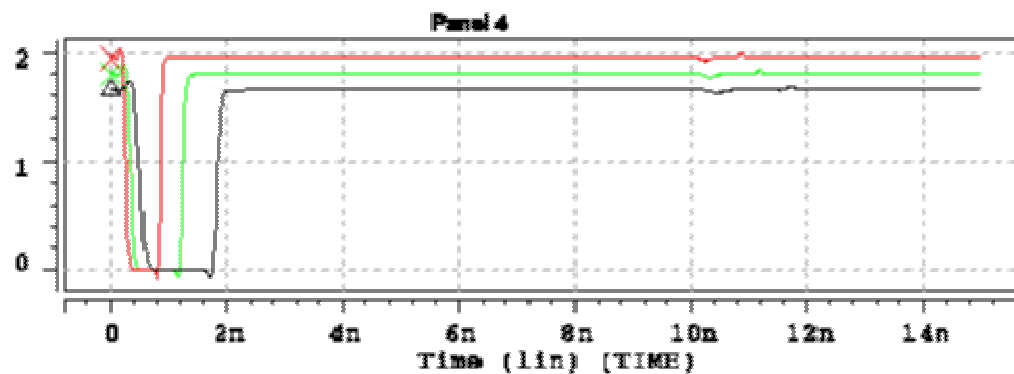
Voltages (lin)



Wave	Symbol
D0:A0:v(out)	X
D1:A0:v(out)	X
D2:A0:v(out)	△

out

Voltages (lin)



3.1.c) Soddisfacimento di VIH/VIL

Le simulazioni mirate alla verifica di soddisfacimento di VIH/VIL coinvolgono il solo circuito di input buffer.

Una rampa molto lenta e' applicata all'ingresso di un input buffer ed il suo valore varia da 0V a VCC.

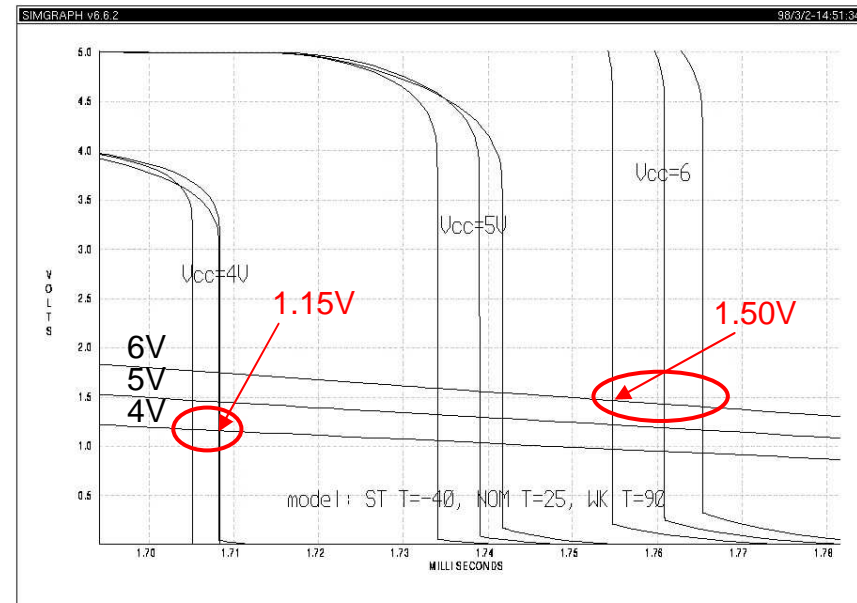
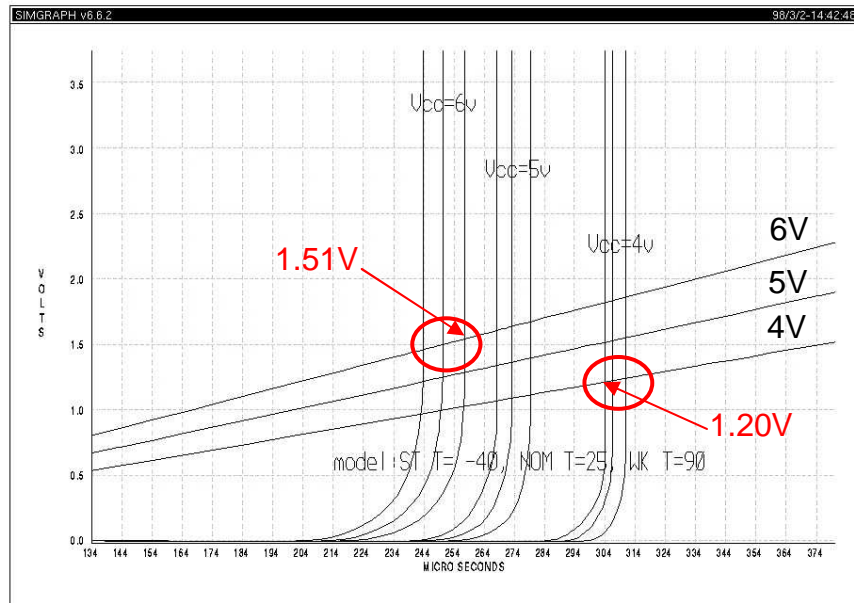
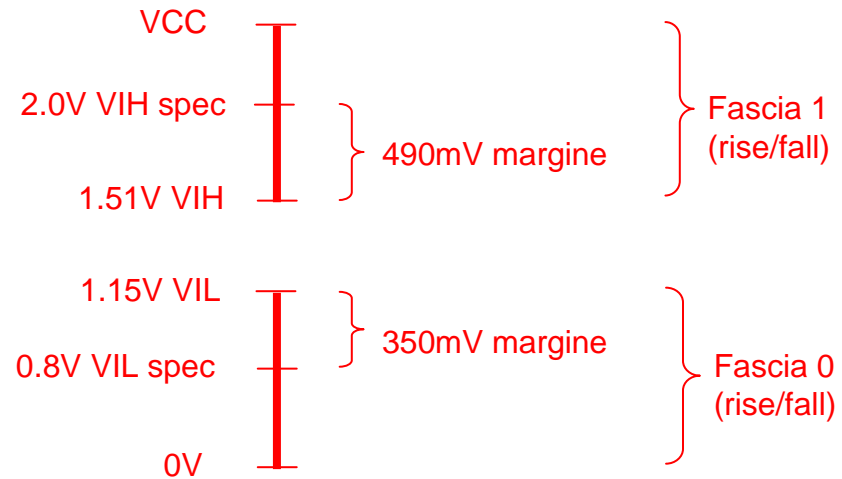
Si vede per quale valore di ingresso si ha la commutazione dell'input buffer e si deve garantire che questo 'trip point' si trovi il piu' possibile centrato rispetto ai valori di specifica in tutte le condizioni di processo, alimentazione e temperatura.

Se per esempio la specifica dice $V_{IH} = 2V \text{ min}$ e $V_{IL} = 0.8V \text{ max}$ (per VCC compresi tra 4V e 6V) e il trip point occupa una fascia pari a 1.15V-1.51V si puo' dire di rispettare la specifica con buon margine.



3.1.c) Soddisfacimento di VIH/VIL (cont.)

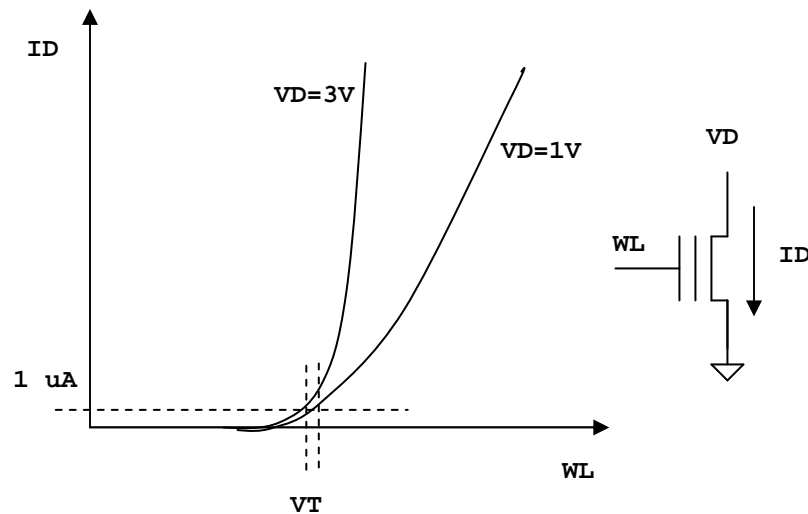
model	Vcc	T	rise	fall
wk	4V	90	1.25	1.15
wk	5V	90	1.40	1.30
wk	6V	90	1.51	1.49
nom	4V	25	1.22	1.15
nom	5V	25	1.37	1.30
nom	6V	25	1.50	1.50
st	4V	-40	1.20	1.16
st	5V	-40	1.35	1.30
st	6V	-40	1.49	1.50



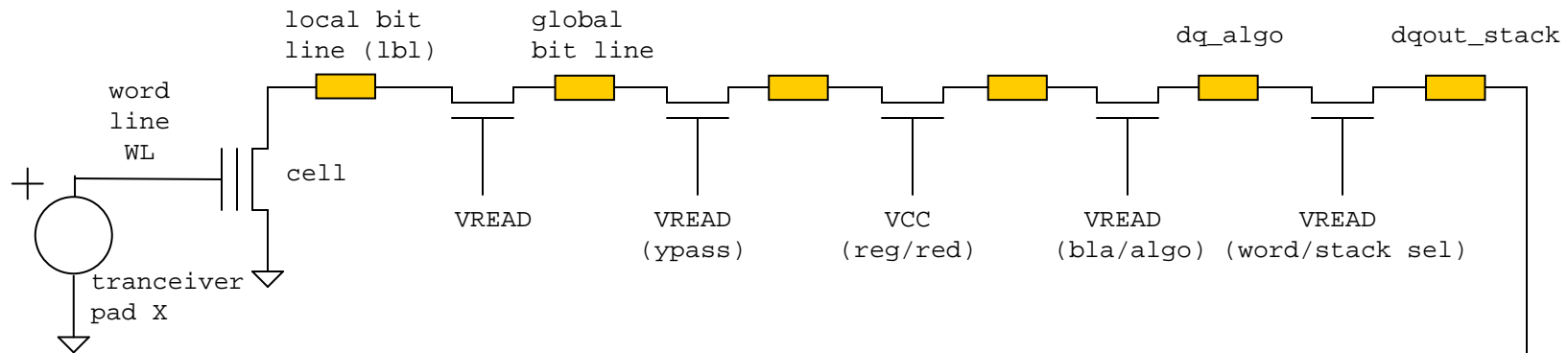
3.1.d) Test mode di accesso a bit lines

Si consideri la tecnica di misura della tensione di soglia V_T di una cella di memoria FLASH.

Tale tecnica consiste nel polarizzare il drain della cella ad una tensione che non produce disturbo di program (tipicamente $\sim 1V$ anche se la misura a tensioni maggiori sarebbe piu' precisa) e nel far variare la tensione WL di gate fino ad un valore a cui la cella comincia a condurre (tipicamente $1\mu A$). Tale valore di WL puo', con buona approssimazione essere considerato uguale a V_T .



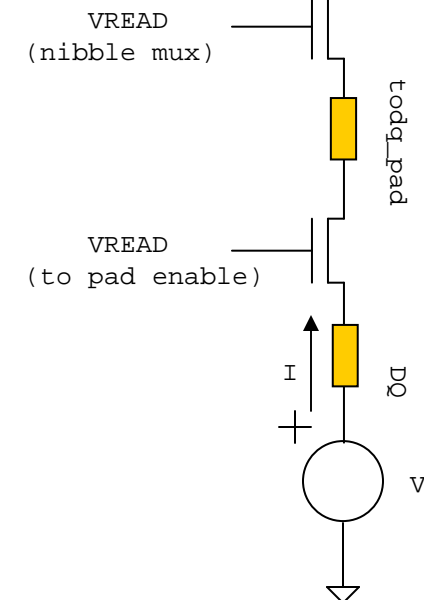
3.1.d) Test mode di accesso a bit lines (cont.)



Siccome la cella di memoria e' annidata all'interno del chip bisogna effettuare l'accesso diretto (bit line access) attraverso la catena di circuiti di decodifica presenti nel chip.

La polarizzazione della bit line a ~1V viene effettuata attraverso un pad esterno e bisogna calcolare quale valore di tensione bisogna applicare all'esterno, e in quali condizioni di alimentazione e temperatura, per garantire ~1V sulla bit line. La tensione di alimentazione X della WL viene anch'essa fornita da un apposito pad Esterno.

Il circuito da simulare in questo caso e' di dimensioni ridotte ed e' una selezione specifica di un percorso costruito componente per componente inclusi i parassiti resistivi.

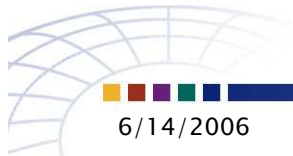
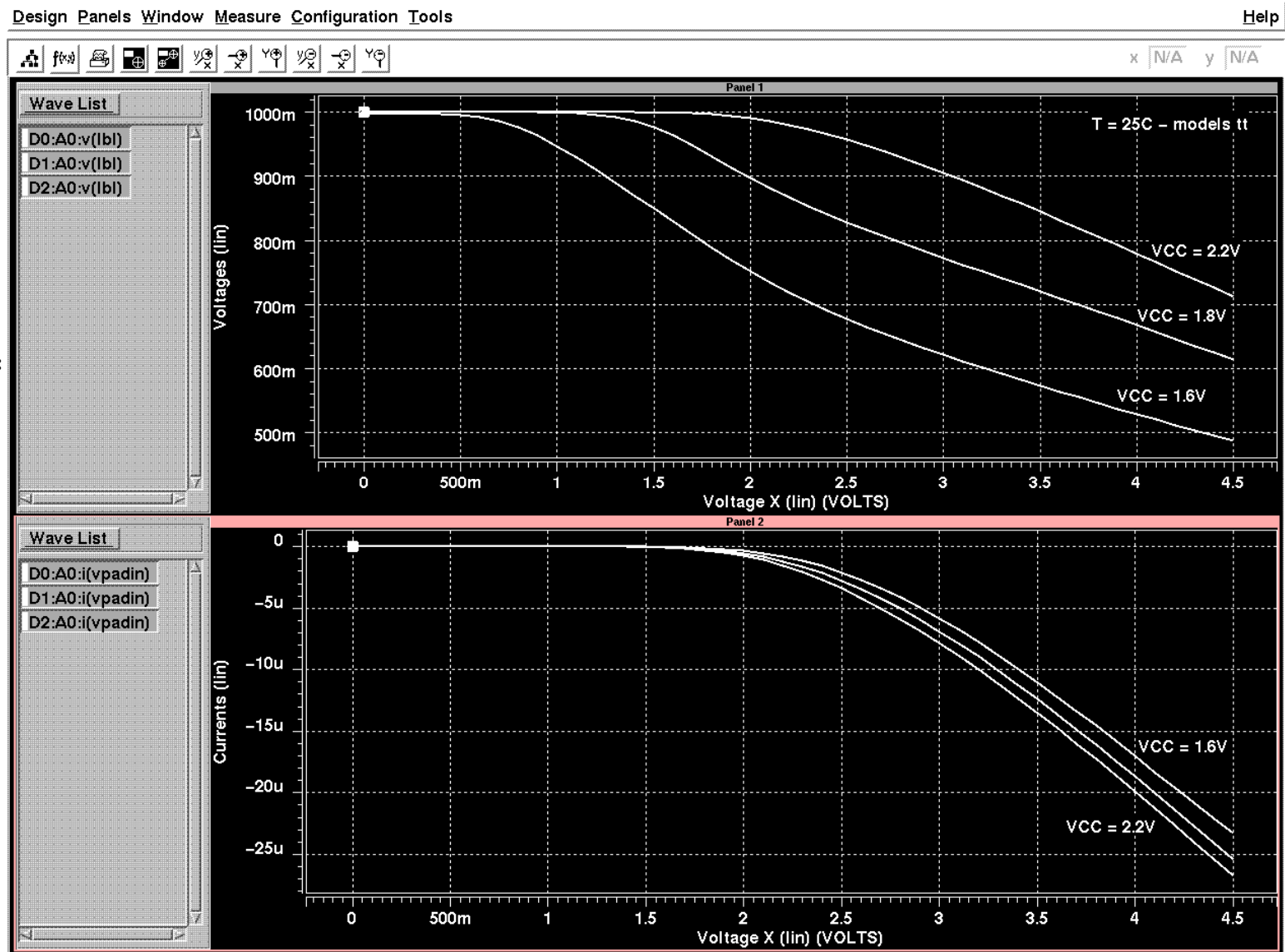


3.1.d) Test mode di accesso a bit lines (cont.)

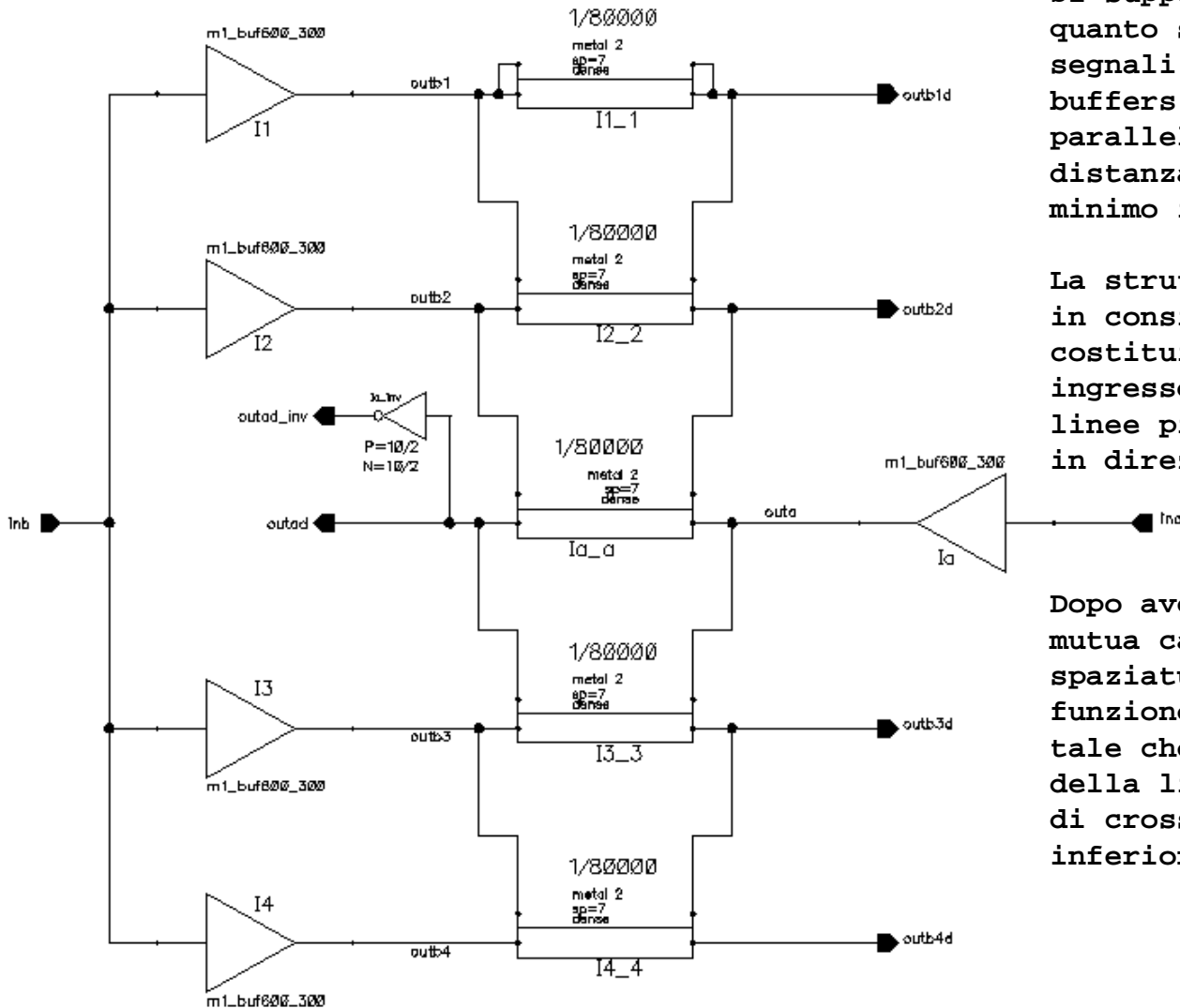
Se si sceglie $V_{READ}=4.5V$ e $V=1V$, si ottengono le forme d'onda riportate a lato.

Esse dicono che:

- 1) pur di scegliere $VCC > 2.2V$, la tensione l_{bl} in condizioni tipiche e' $\sim 1V$.
- 2) la tensione di soglia V_T della cella e' $2V$.



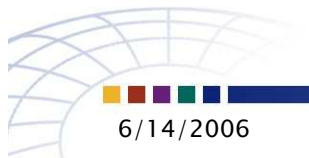
3.1.e) Dimensionamento di componenti e/o topologia per rendere minimo il cross coupling



Si supponga di voler valutare di quanto si devono distanziare segnali che sono pilotati da buffers potenti e che viaggiano parallelamente per una lunga distanza in modo tale da rendere minimo il cross coupling.

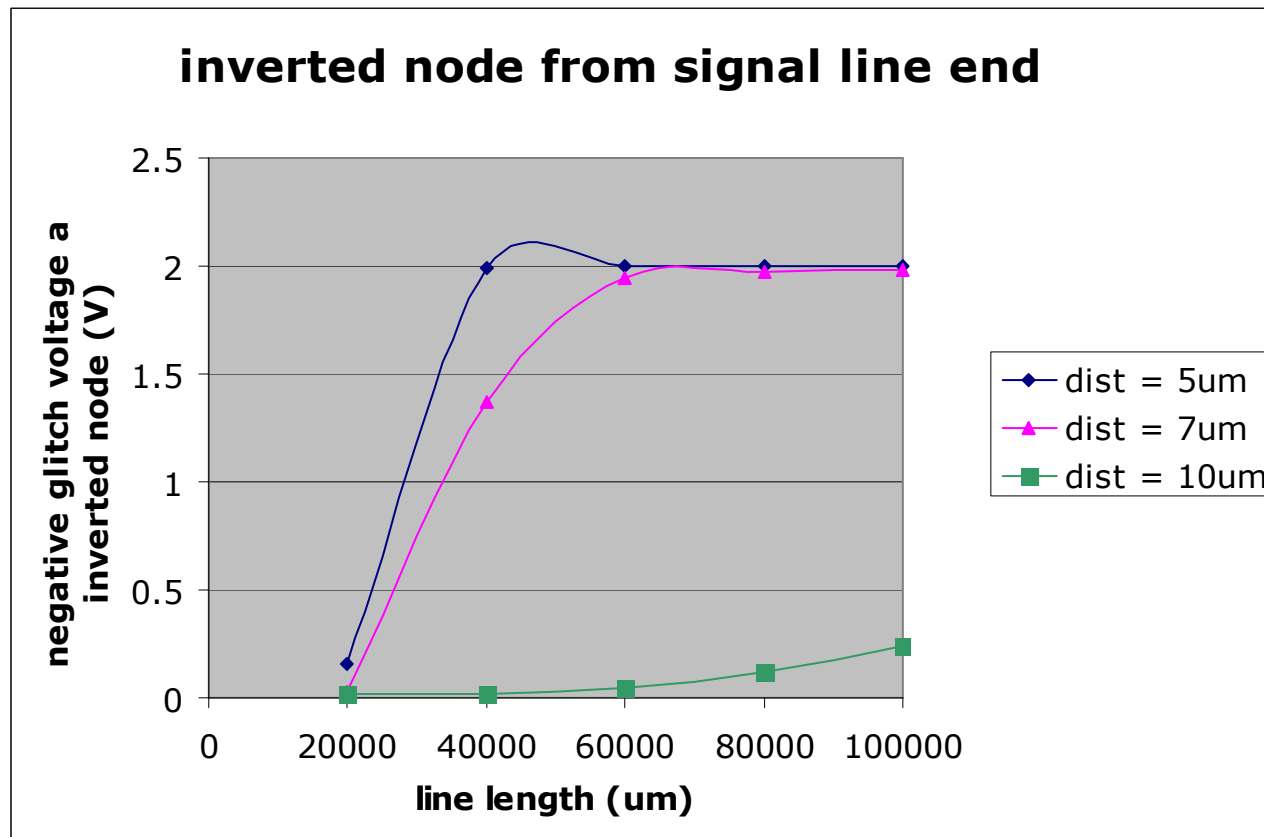
La struttura che si puo' prendere in considerazione e' quella costituita da una linea (il cui ingresso e' ina) affiancata da 4 linee pilotate contemporaneamente in direzione opposta (segnale inb).

Dopo aver modellato l'effetto della mutua capacita', si valuta la spaziatura delle connessioni in funzione della lunghezza, in modo tale che l'inverter alla fine della linea intermedia non risente di cross coupling (la sua uscita e' inferiore al punto di scatto).



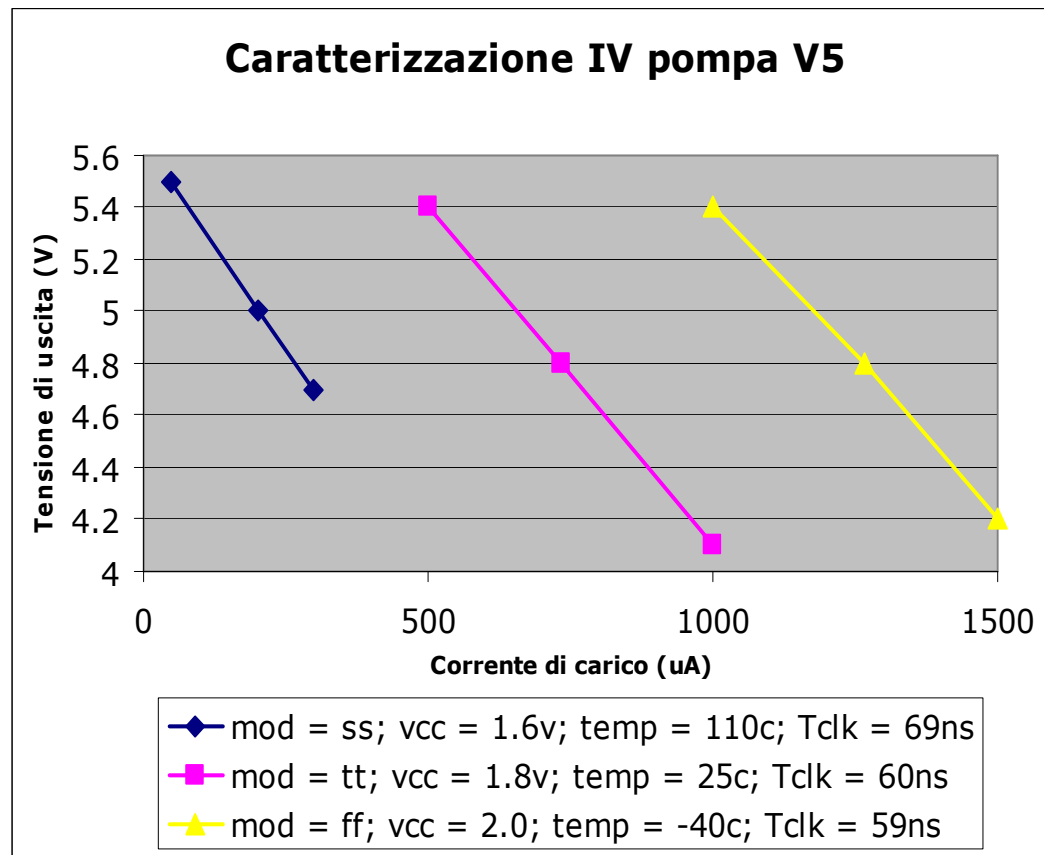
3.1.e) Dimensionamento di componenti e/o topologia per rendere minimo il cross coupling (cont.)

I risultati di questa simulazione effettuata in una sola combinazione di modelli (tt) tensione di alimentazione (2V) e temperatura (T=25C) dicono che, fissate le dimensioni dei driver, l'uscita dell'inverter non ha glitch pericolosi (cioe' sono inferiori alla soglia di scatto) si si mantengono le linee spaziate di almeno 10 μm (drawn) e cio' e' vero fino a percorrenze di 100000 μm (drawn). Se invece si usano spaziature inferiori a 7 μm (drawn) le lunghezze di interconnessione non possono superare il valore di 20000 μm (drawn).



3.1.f) Curve di carico per charge pumps

Una pompa (charge pump) e' un circuito che a partire da una tensione di alimentazione e' in grado di generare una tensione di valore superiore di cui il circuito integrato puo' aver bisogno per eseguire diverse operazioni senza che queste tensioni vengano fornite dall'esterno. In una memoria FLASH, alimentata a 1.8V nominali, tipicamente si ha bisogno di varie tensioni interne di valore superiore : 5V, 7V, 12V.



Questi generatori di tensioni interne devono soddisfare diversi requisiti di corrente.

Alcuni devono fornire alta corrente senza variare il loro valore di tensione, altri devono fornire una corrente minore senza variare il loro valore di tensione.

E' quindi necessario valutare quanta corrente e' in grado di fornire una pompa prima che il suo valore di tensione si degradi a limiti inaccettabili.

Tali simulazioni richiedono di usare la intera pompa e di modellare carichi esterni di corrente variabile. Il numero di componenti e' dell'ordine di qualche centinaia con topologia concentrata.



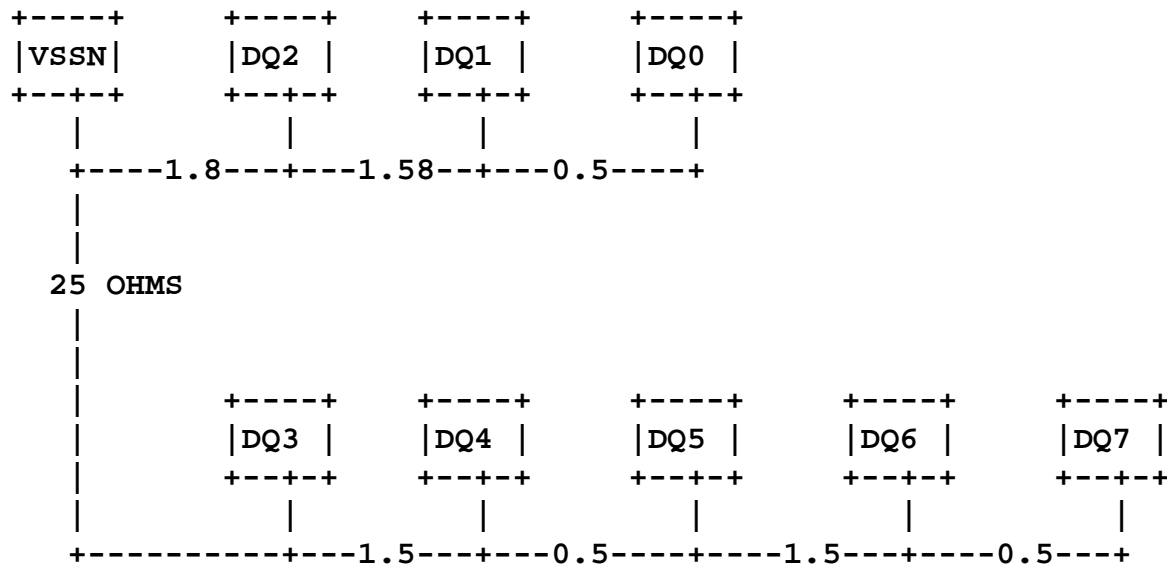
3.2.a) valutazioni delle connessioni di gnd per rendere minimo il 'bouncing' in output buffers

L'esempio precedentemente riportato a proposito delle simulazioni di rise/fall time di un output buffer richiede un approfondimento.

Quando si deve effettuare una simulazione di rise/fall si puo' partire dalla semplice simulazione di un singolo stadio, ma si possono commettere anche gravi errori se non si tiene conto della topologia che coinvolge tutti gli stadi di uscita del chip.

Viene riportato di seguito il caso di un fallimento e le simulazioni elettriche usate per effettuare il debug.

Le misure sul silicio dei tempi di propagazione in un circuito integrato avevano un valore uguale per tutte le uscite DQ7...DQ0 per transizioni da 0 a 1 mentre erano differenti tra i diversi DQ per transizioni da 1 a 0 essendo DQ2 il piu' rapido e DQ7 il piu' lento con una Differenza di 15nsec tra i due.



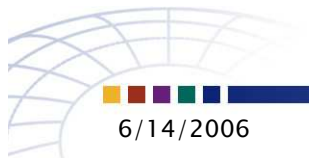
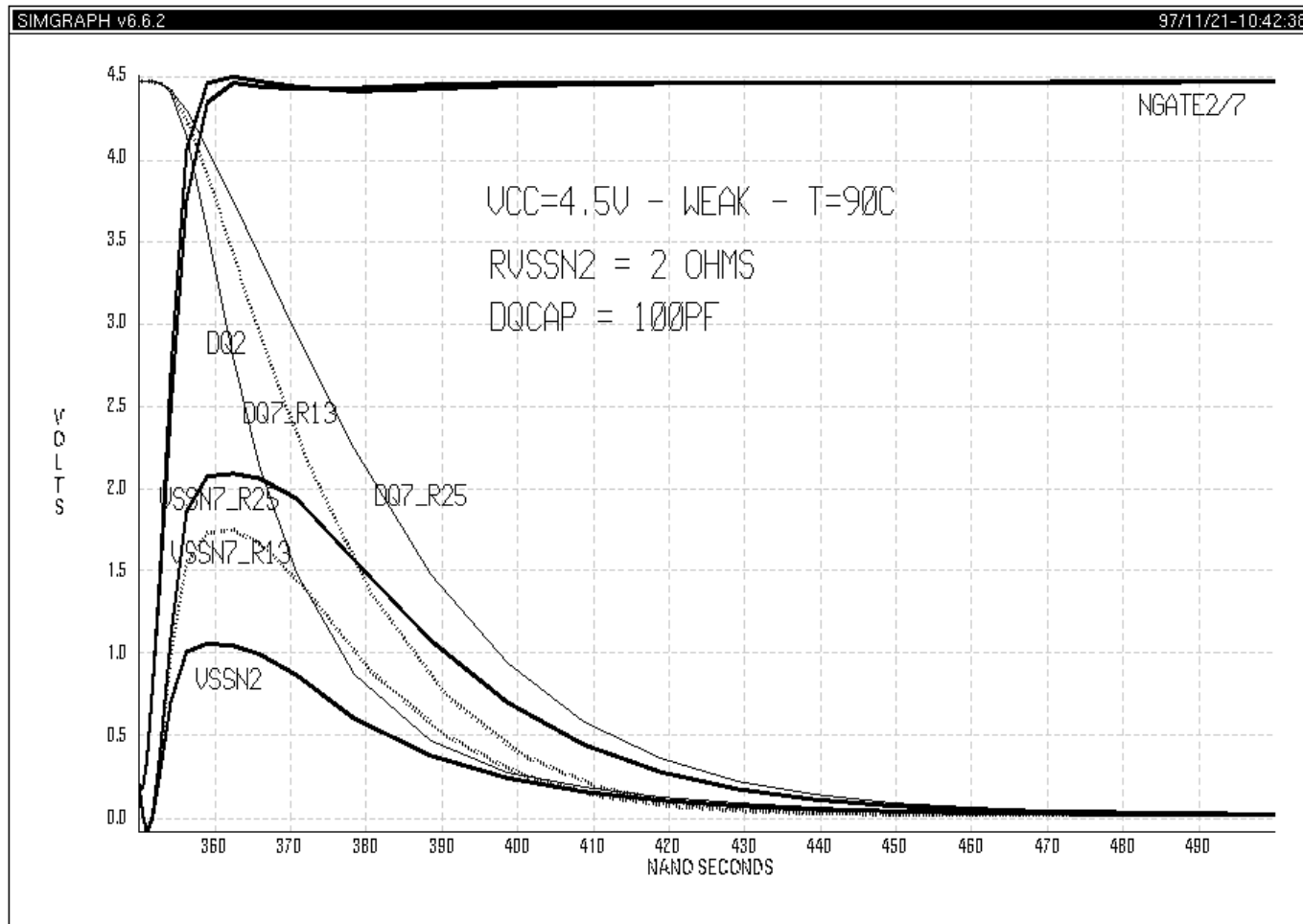
Un esame del layout mostro' la topologia di ground a lato riportata tra i pad DQ2...DQ0 collocati al top del chip e i pad DQ7...DQ3 collocati al bottom del chip.

Le simulazioni di fall time furono girate di nuovo con tutti i DQ che commutavano contemporaneamente e con la reale topologia di ground ottenendo una coincidenza tra simulazioni e silicio.



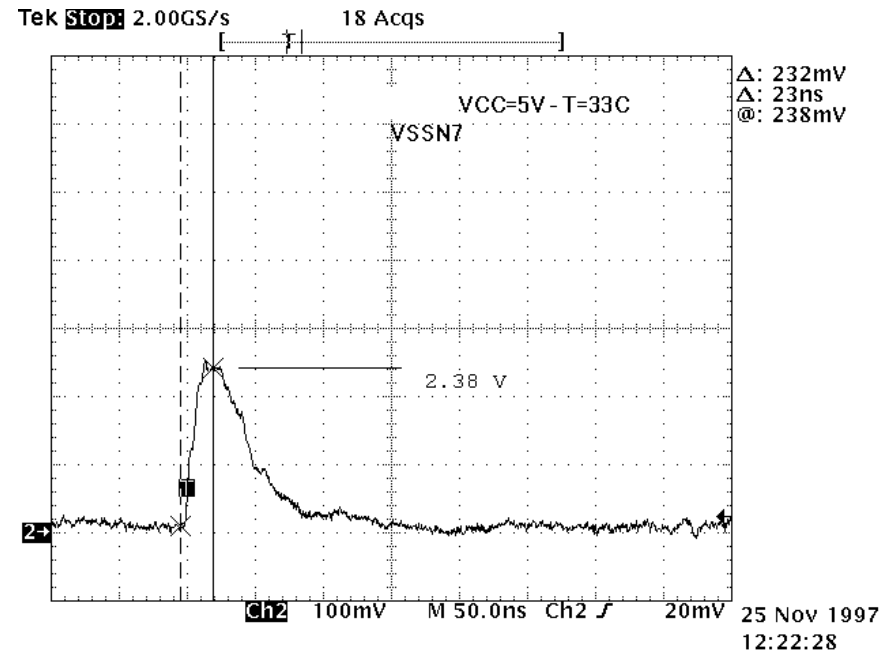
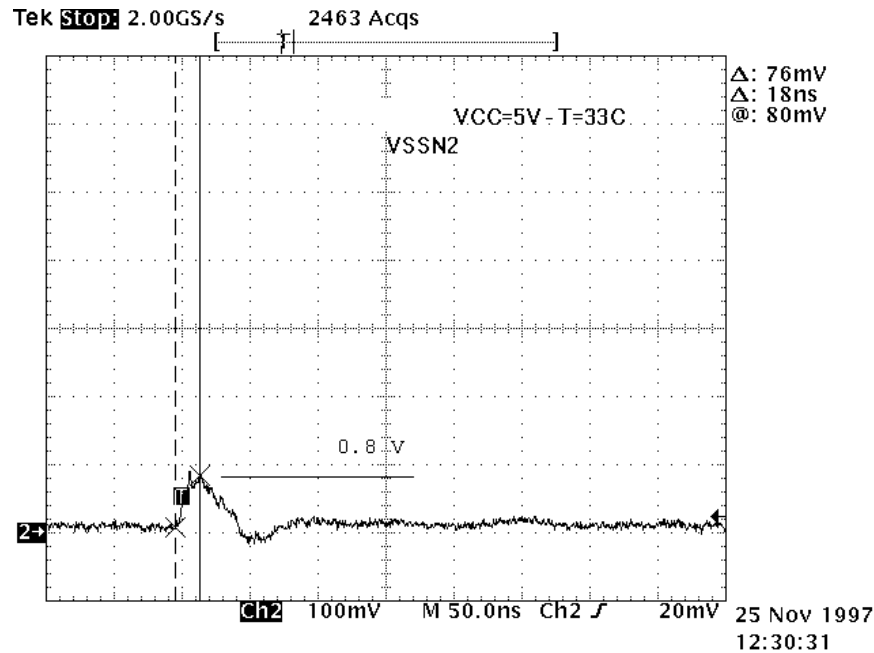
3.2.a) valutazioni delle connessioni di gnd per rendere minimo il 'bouncing' in output buffers

Simulazioni di ground bouncing per : 1) il ground (VSSN2) del pad DQ2; 2) per il ground (VSSN7_R25) del pad DQ7; 3) del ground (VSSN7_R13) del pad DQ7 dopo aver ridotto la resistenza di VSSN tra top e bottom da 25 ohms a 13 ohms.



3.2.a) valutazioni delle connessioni di gnd per rendere minimo il 'bouncing' in output buffers (cont.)

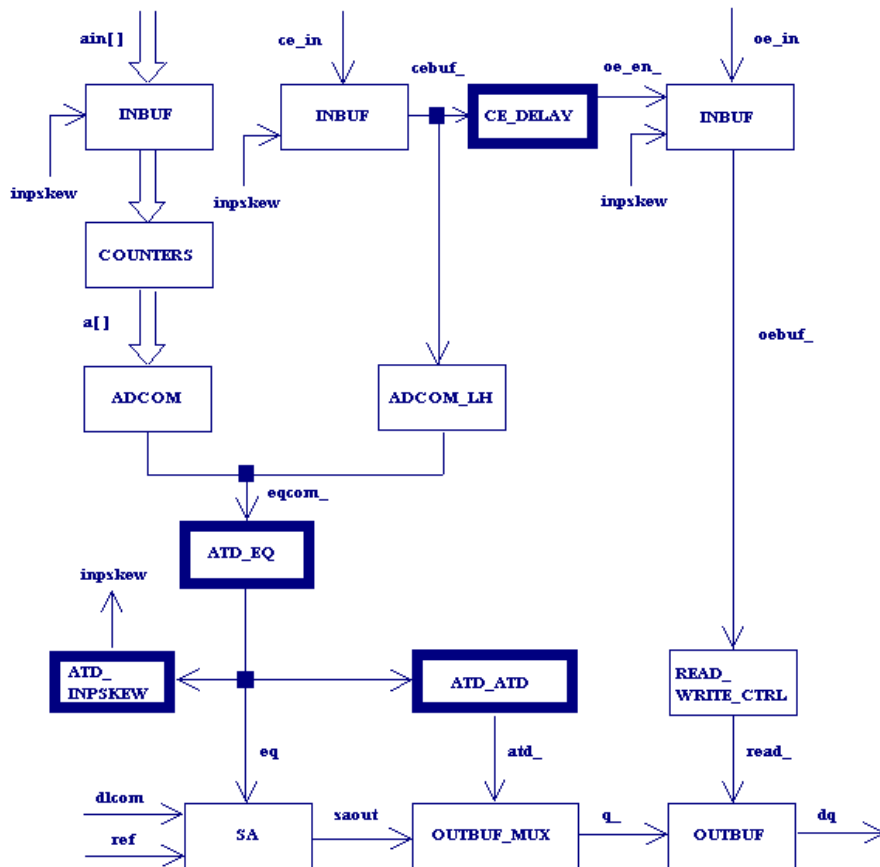
Misure sul silicio del ground bouncing per il ground VSSN2 relativo al pad DQ2 e per il ground (VSSN7) relativo al pad DQ7.



3.2.b) calcolo di tempi di accesso da address e da CE_

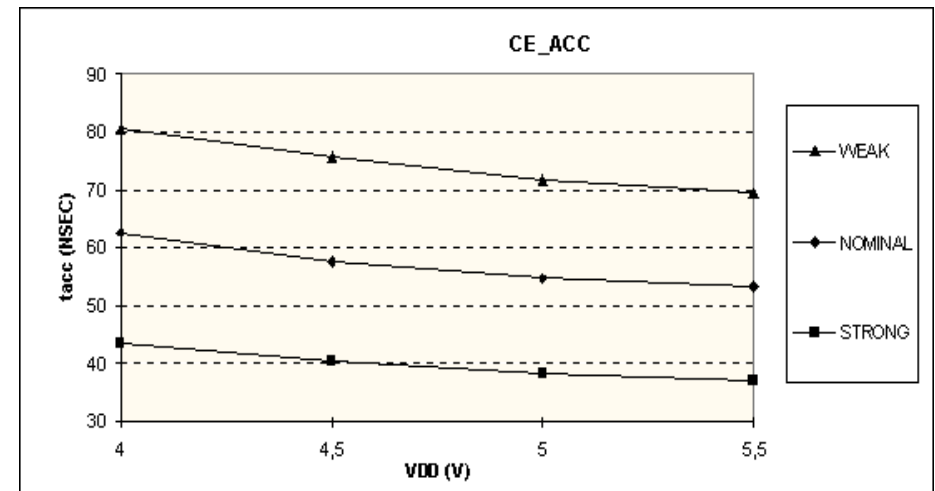
La valutazione di un tempo di accesso da CE_ richiede di impostare una simulazione elettrica che coinvolge tutto il percorso di propagazione del segnale di chip enable che abilita il passaggio della corrente dalla cella FLASH al pad di uscita.

READ PATH DELAYS

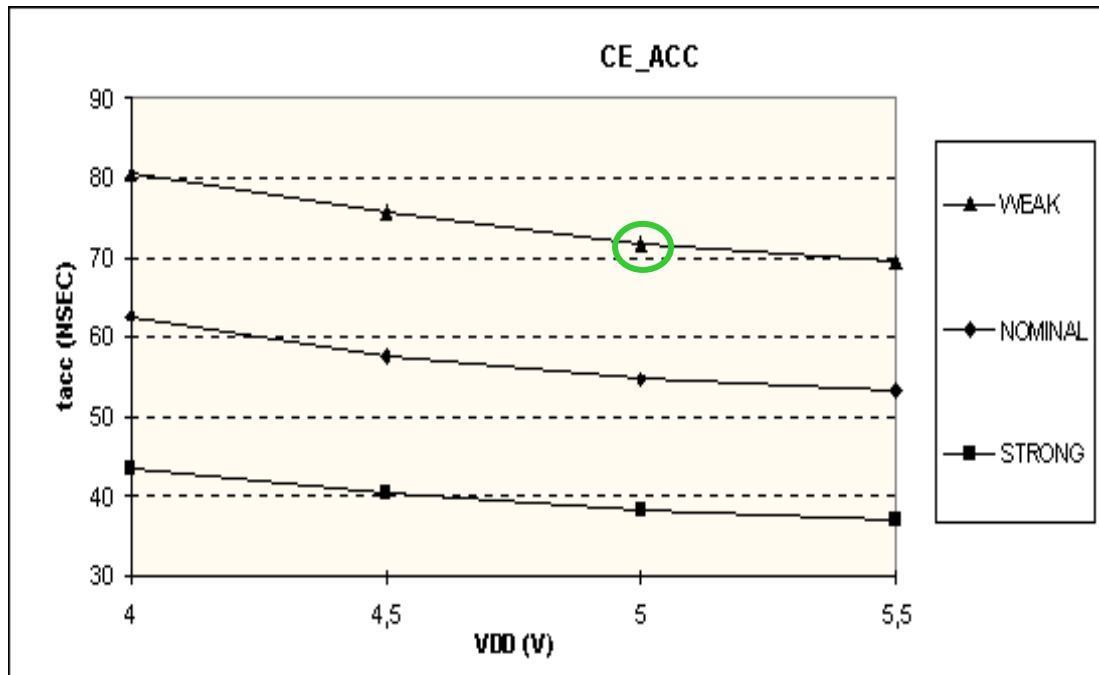


Una tale simulazione richiede di isolare i singoli blocchi componenti il percorso di propagazione e di ottenere uno specifico circuito in cui verranno inseriti i modelli RC di interconnessione e di carico.

In genere un tale circuito ha dimensioni di qualche migliaia di componenti e richiede una cura particolare nell'effettuare la modellistica di interconnessione.



3.2.c) Valutazione di accoppiamento di segnali con diversa drive strength dovuto a topologia distribuita e layout automatico



Si riporta ora il caso di un fallimento dovuto a non corretta modellizzazione dei parassiti di accoppiamento durante la valutazione di un tempo di accesso da CE_.

I dati misurati sul silicio a 5V e alta temperatura sono ~95.5ns per DQ3...DQ7 e ~85ns per DQ0 e DQ2 contro i ~72ns delle simulazioni. Cio' e' dovuto ad una cattiva connessione di ground particolarmente evidente per DQ3...DQ7.

In aggiunta a questo problema dovuto ad una cattiva connessione di ground si osserva una penalizzazione specifica nella propagazione relativa al dato su DQ1 che ha una propagazione di 101.5ns.

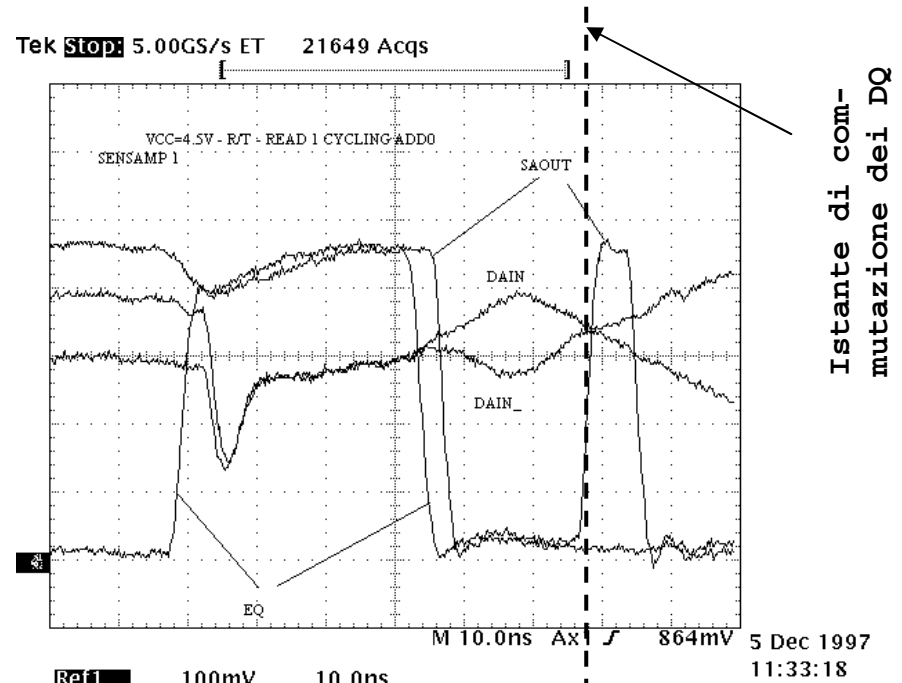
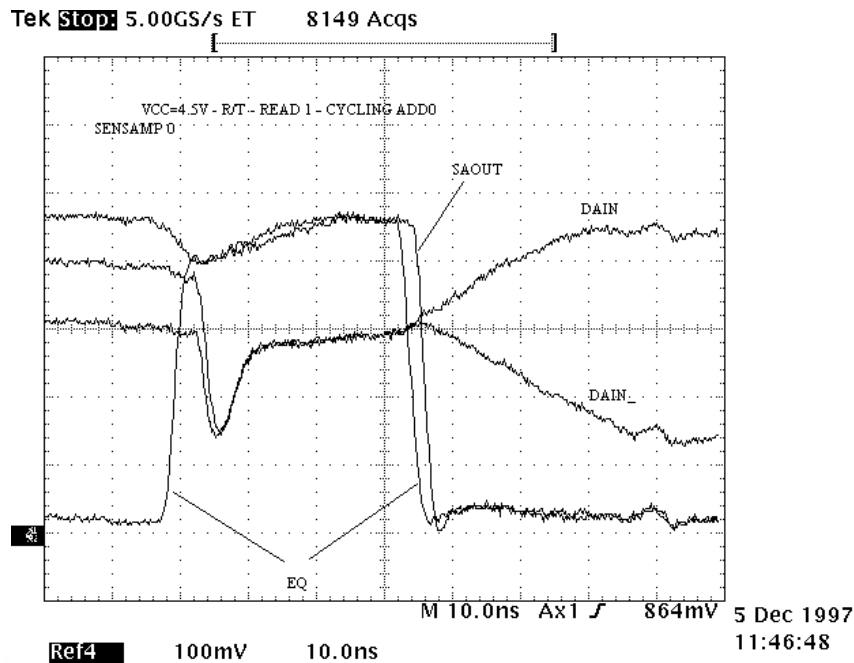
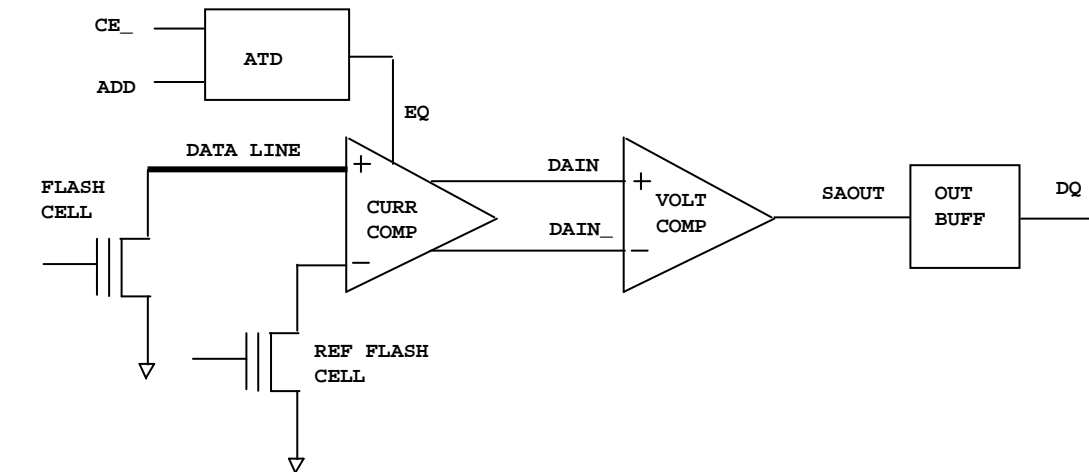
	DQ0	DQ1	DQ2	DQ3	DQ4	DQ5	DQ6	DQ7
1b) ta() (5V) - CMOS lvs,Byte Mode [Addr Comp,RPX pat]								
4.00V:	92,93	Fail	94,96	101,05	101,05	101,56	101,56	120,86
4.25V:	89,88	123,4	91,91	99,02	99,53	99,53	100,04	111,21
4.50V:	87,34	113,24	89,38	97,5	98,01	98,01	98,01	102,07
4.75V:	85,31	106,13	87,34	96,48	96,99	96,99	96,99	97,5
5.00V:	83,79	101,56	85,31	95,47	95,98	95,98	95,98	96,48
5.25V:	82,77	98,52	84,3	94,96	94,96	94,96	95,47	95,47
5.50V:	82,27	95,47	83,28	94,45	94,96	94,45	94,45	94,96
5.75V:	81,25	93,44	82,77	93,95	94,45	93,95	94,45	94,45
6.00V:	80,74	91,41	81,76	93,44	93,95	93,95	93,95	94,45

A causa della topologia distribuita e dell'esecuzione automatica del layout, e' capitato che la data line del sensamp 1 (che ha una bassa capacita' di drive : 20uA) si e' accoppiata per un lungo tratto con una linea DQ (pilotata da un buffer molto forte)

3.2.c) Valutazione di accoppiamento di segnali con diversa drive strength dovuto a topologia distribuita e layout automatico (cont.)

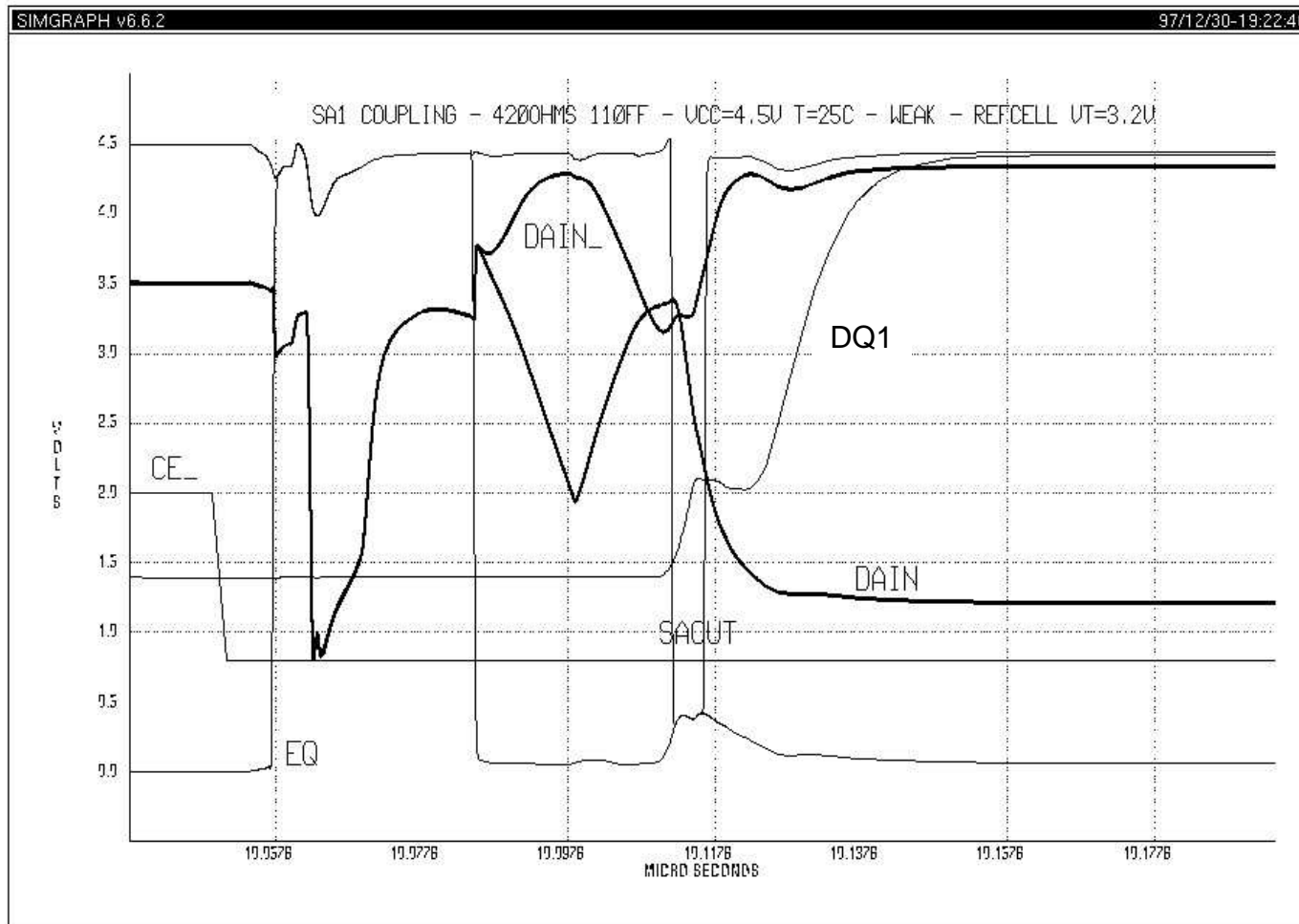
Il normale funzionamento relativo a DQ diversi da DQ1 e' riportato nelle forme d'onda di sinistra misurate sul silicio.

A causa dell'accoppiamento di DATA LINE la propagazione relativa a DQ1, riportata nelle forme d'onda di destra misurate sul silicio, presenta un impulso anomalo che ritarda l'intero accesso.



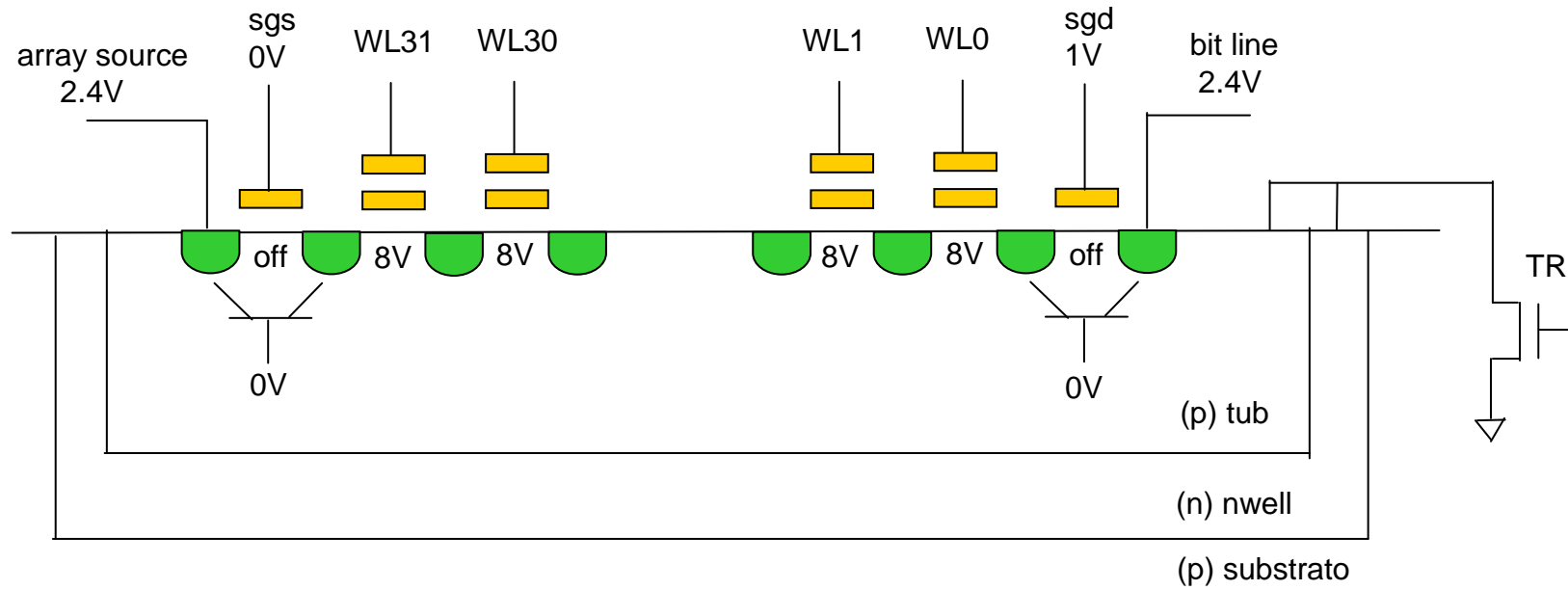
3.2.c) Valutazione di accoppiamento di segnali con diversa drive strength dovuto a topologia distribuita e layout automatico (cont.)

Dopo aver opportunamente modellato l'accoppiamento tra la data line della line DQ1 e il DQ aggressore, le forme d'onda misurate sul silicio si sono riprodotte con precisione notevole anche in simulazione.

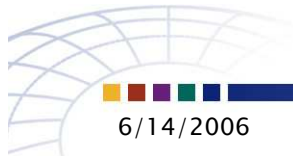


3.2.d) Valutazione efficacia di ground in carichi estremamente distribuiti (tub)

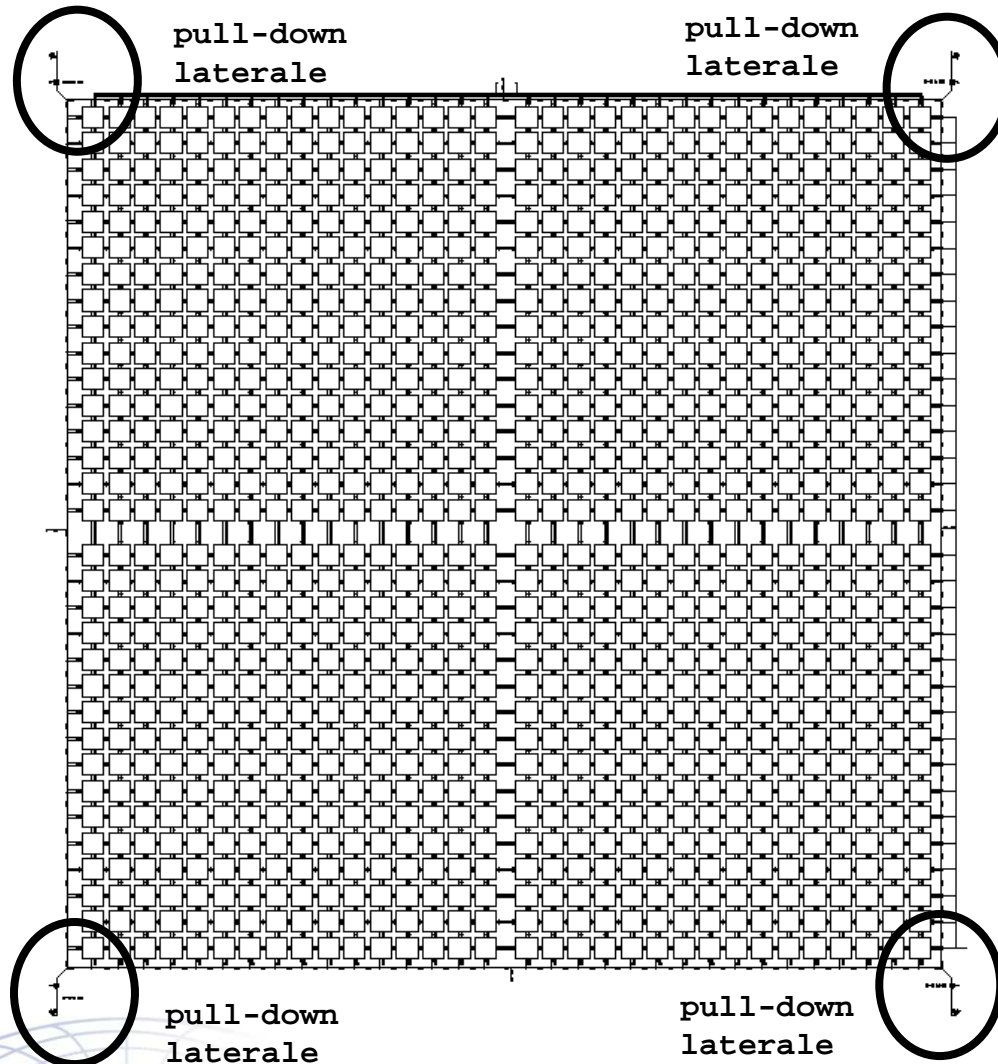
Un evidente effetto di picco di circa 650mV nel tub di una memoria NAND FLASH e' stato osservato in corrispondenza della salita del source e delle bit lines in una operazione di program. Tale valore deve considerarsi molto rischioso perche' coincidente con con la tensione di innesco di transistor npn di selezione di una stringa NAND. Tale innesco potrebbe far scaricare il canale di una stringa inibita producendo un disturbo di programmazione.



Il canale sotto ogni cella della stringa inibita e' a 8V per accoppiamento con la WL che si vuole programmare (WL= \sim 20V). Il tub e' tenuto a 0V dai transistor di pull down TR. Se si dovesse avere un glitch nel tub superiore a 0.7V, i transistor npn di selezione sgs/sgd potrebbero andare ON scaricando il canale e producendo un disturbo di programmazione.



3.2.d) Valutazione efficacia di ground in carichi estremamente distribuiti (tub) (cont.)



Per un corretto funzionamento, il tub deve mantenersi a livello di 0V e comunque non superare il valore di 200mV.

Allo scopo di valutare il valore di picco del tub, e' stato costruito un modello dell'intero array di una memoria FLASH.

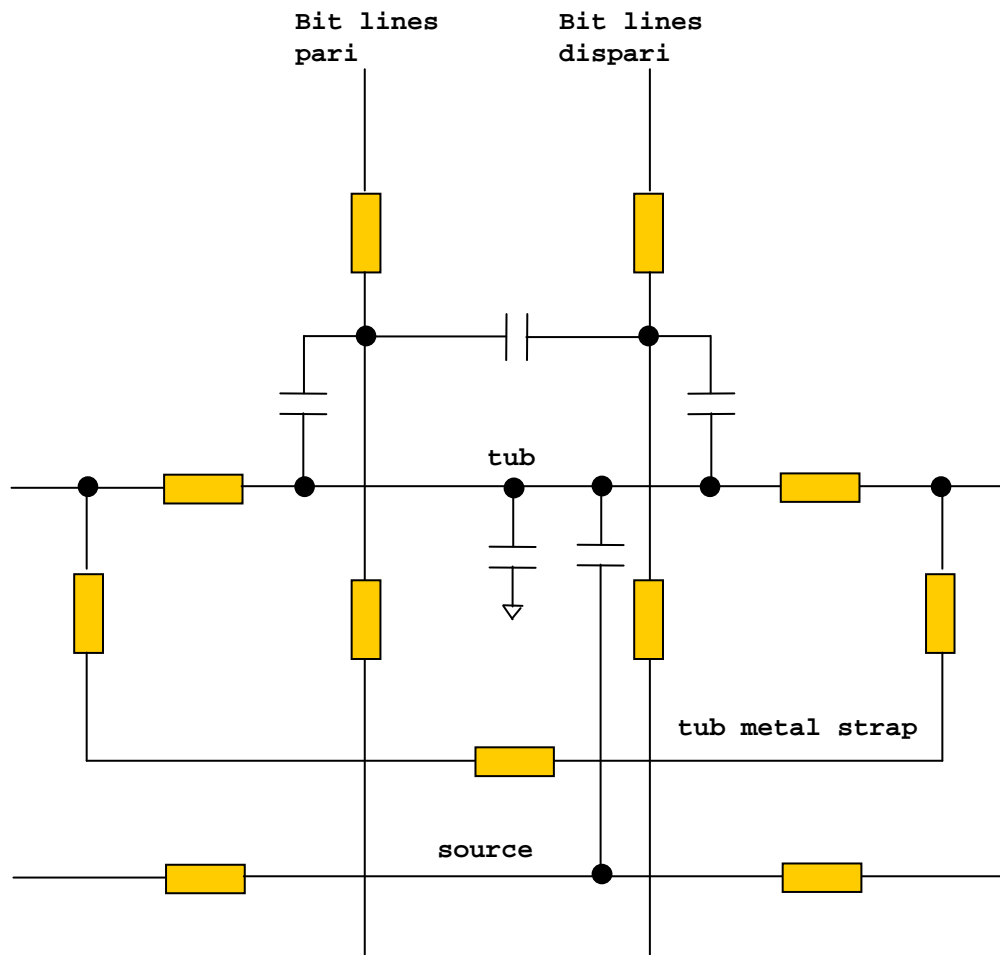
Tutto l'array e' stato scomposto in 32x32 moduli con i 4 pull down laterali che dovrebbero ancorare il tub a ground.

Le bit lines ed il source sono stati fatti salire al loro valore di regime ed il valore di accoppiamento del tub e' stato valutato per diversi valori dei 4 pull down laterali trovando il miglior valore che garantisce il soddisfacimento del requisito di 200mV.

La simulazione ha permesso inoltre la valutazione di diverse strategie di pull down.



3.2.d) Valutazione efficacia di ground in carichi estremamente distribuiti (tub) (cont.)



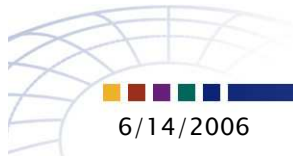
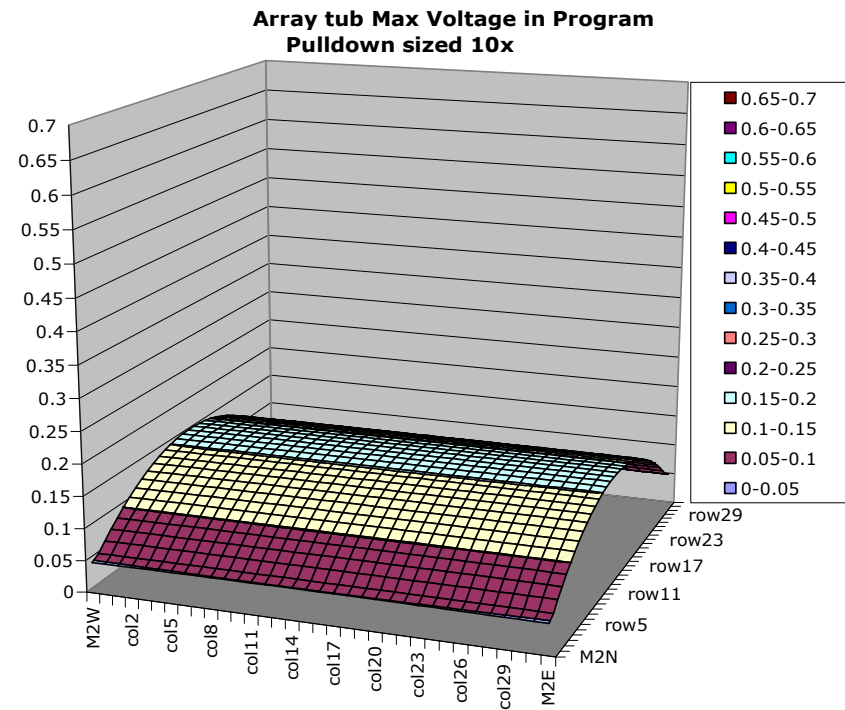
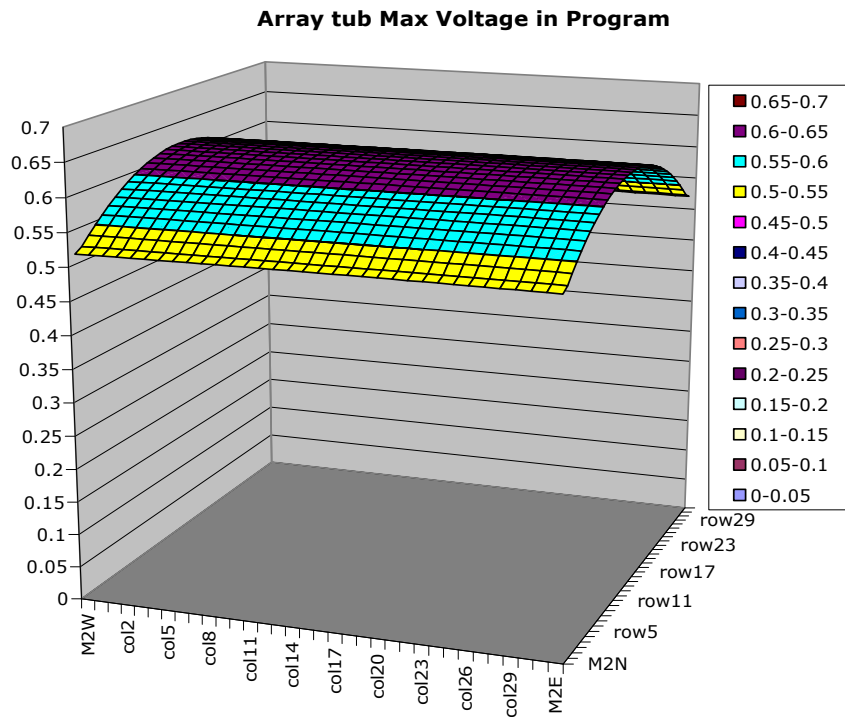
Il modello del singolo modulo in cui e' stato diviso l'intero array di memoria tiene conto di :

- resistenza orizzontale del tub e sua capacita' verso il substrato
- resistenza e capacita' delle bit lines pari e dispari
- resistenza e capacita' del source verso il tub
- effetto dello strap di metallo per la diminuzione della resistenza del tub



3.2.d) Valutazione efficacia di ground in carichi estremamente distribuiti (tub) (cont.)

Le simulazioni hanno permesso di confermare che con la scelta originaria dei pull down si aveva un accoppiamento di tub di 620mV mentre aumentando la dimensione dei pull down a 10 volte si ottiene un accoppiamento del tub di 150mV.

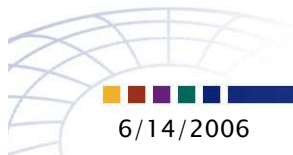


3.2.e) Analisi dei margini in una operazione di lettura di una memoria flash

Per 'analisi dei margini di read' si intende un insieme di simulazioni elettriche dei circuiti di lettura di una memoria FLASH tese a stabilire per un dato valore di tensione applicato alla word line (WL) quali sono i valori della tensione di soglia V_T della cella FLASH che vengono letti come 1 e 0 al variare delle diverse condizioni operative che possono influenzare la lettura : tensione di alimentazione, processo, temperatura, polarizzazione di bit line, istante di lettura (strobe), diverse leggi (tecniche) di compensazione della temperatura, spread delle tensioni di riferimento.

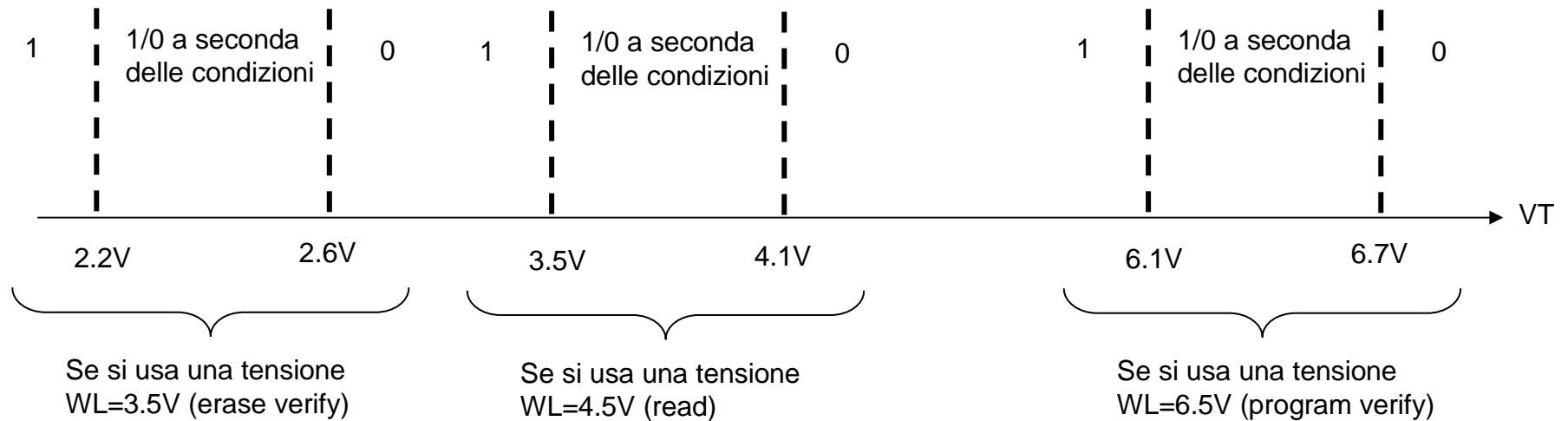
Tale simulazione deve essere appositamente costruita includendo tutte le parti che concorrono a formare la catena di sensing a partire dalla generazione della WL fino al circuito che genera lo strobe del sens amplifier :

- array di celle in cui si possa variare la V_T del modello da valori cancellati a valori programmati
- modello di word line che permetta di tener conto dell'attivazione della cella piu' vicina e di quella piu' lontana dal row decoder
- modello delle bit lines che permetta di tener conto dell'attivazione della cella piu' lontana e di quella piu' vicina al column decoder
- circuito di sensamp e circuiteria di controllo associata (pre-charge; read timer)



3.2.e) Analisi dei margini in una operazione di lettura di una memoria flash (cont.)

Nello specifico esempio di seguito riportato per un memoria FLASH NOR, durante l'operazione di cancellazione si e' usato 3.5V sulla WL nella fase di erase verify e si e' trovato che tutte le celle hanno $V_T < 2.6V$ mentre durante una operazione di programmazione si e' usato 6.5V sulla WL nella fase di program verify e si e' trovato che tutte le celle hanno $V_T > 6.1V$.



Se durante una operazione di lettura si ipotizza di usare $WL=4.5V$, si leggono sicuramente come 1 tutte le celle aventi $V_T < 3.5V$ (con un margine di $3.5V - 2.6V = 0.9V$ rispetto alle celle cancellate) e si leggono sicuramente come 0 tutte le celle aventi $V_T > 4.1V$ (con un margine di $6.1V - 4.1V = 2.0V$ rispetto alle celle programmate). In genere si sceglie un margine maggiore verso le celle programmate che potrebbero perdere carica nel tempo (data retention).



3.2.f) Frequenza di funzionamento di una burst machine

BIT #	DESCRIPTION	FUNCTION
15	Read Mode	0 = Synchronous Burst Access Mode 1 = Asynchronous/Page Access Mode (default)
14	Reserved	Default = 1
13-11	Latency Code	Sets the number of clock cycles before valid data out (see Figure 6): 000 = Code 0 - reserved 001 = Code 1 - reserved 010 = Code 2 011 = Code 3 100 = Code 4 101 = Code 5 110 = Code 6 - reserved 111 = Code 7 - reserved (default)
10	Wait Signal Polarity	0 = WAIT# signal is active LOW 1 = WAIT# signal is active HIGH (default)
9	Hold Data Out	Sets the data output configuration: 0 = Hold data for one clock 1 = Hold data for two clocks (default)
8	Wait Configuration	Controls the behavior of the WAIT# output signal: 0 = WAIT# asserted during delay 1 = WAIT# asserted one data cycle before delay (default)
7	Burst Sequence	Specifies the order in which data is addressed in synchronous burst mode: 0 = Reserved 1 = Linear (default)
6	Clock Configuration	Defines the clock edge on which the BURST operation starts and data is referenced: 0 = Falling edge 1 = Rising edge (default)
5-4	Reserved	Default = 0
3	Burst Wrap	0 = Burst wraps within the burst length 1 = Burst no wrap (default)
2-0	Burst Length	Sets the number of words the device will output in burst mode: 001 = 4 words 010 = 8 words 011 = 16 words 111 = Continuous burst (default)

Per aumentare la quantità di dati nell'unità di tempo che si possono leggere da una memoria FLASH e' stato definito un sistema di lettura seriale dei dati (burst) che a partire da un indirizzo iniziale permette di scandire tutte le locazioni della memoria al commutare di un clock.

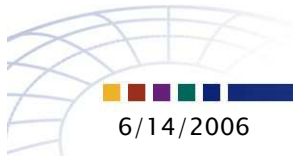
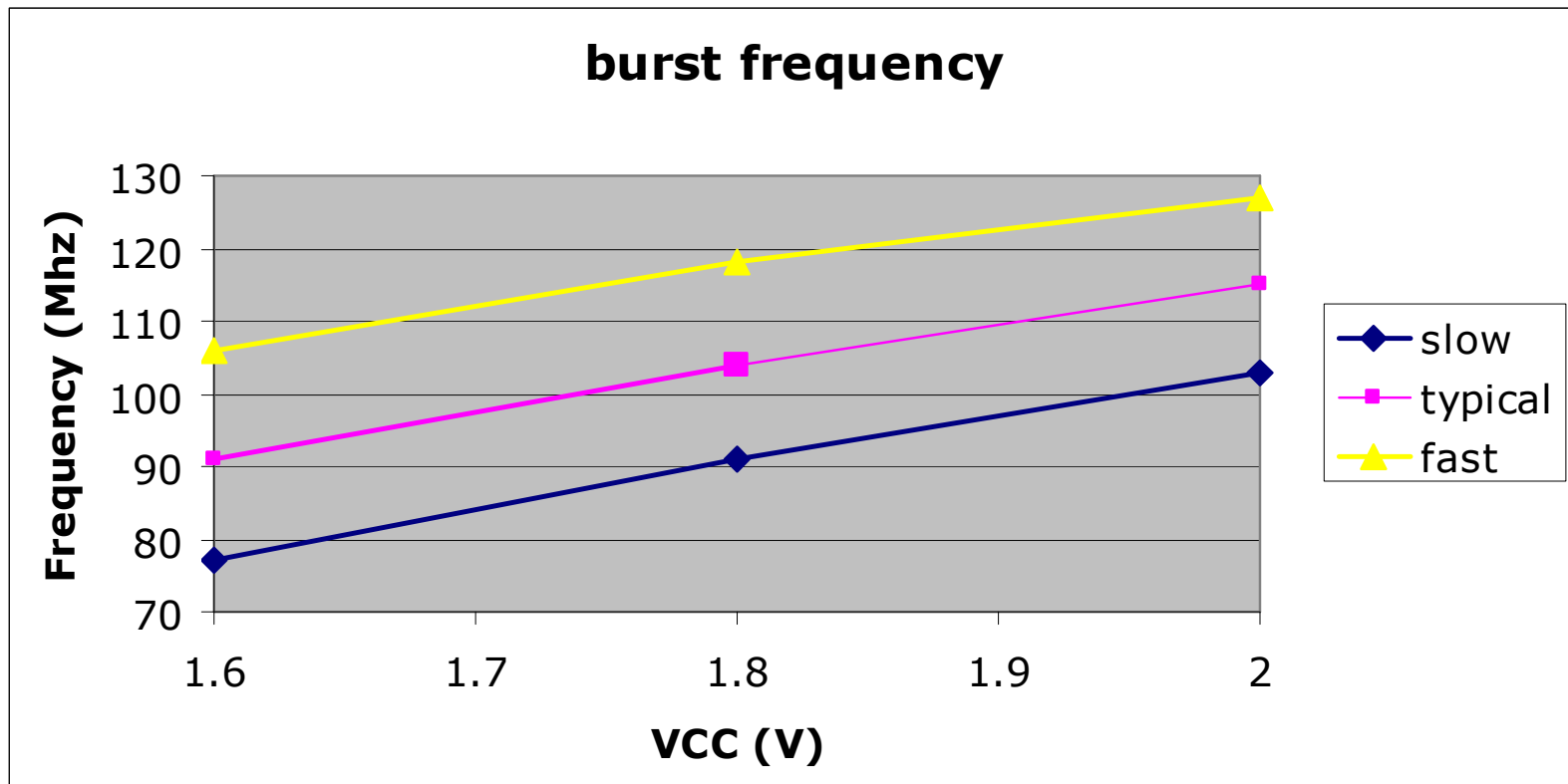
Tale sistema burst può essere configurato con un sistema di 15 bits (rcr) che definiscono varie configurazioni di burst : latenza iniziale, polarità del wait, numero di hold, fronte attivo del clock, modalità di wrap.

Le simulazioni elettriche per la caratterizzazione di una burst machine sono quindi complesse perché coinvolgono innumerevoli configurazioni, richiedono l'uso di grandi circuiti in cui si deve tener conto dei parassiti distribuiti.

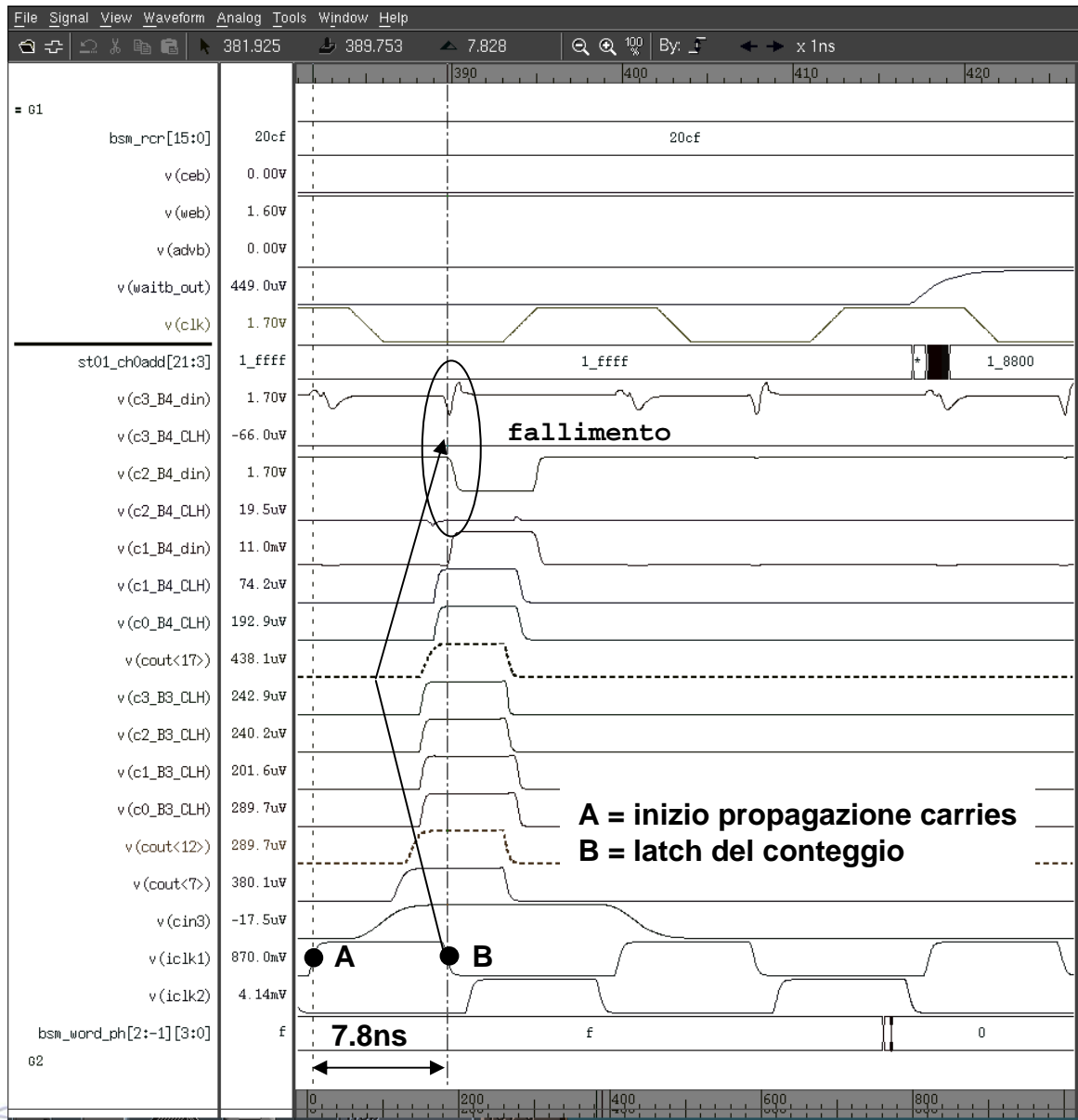


3.2.f) Frequenza di funzionamento di una burst machine (cont.)

Un esempio di simulazione di burst machine in cui si doveva garantire almeno 66Mhz di frequenza di funzionamento e' di seguito riportato. La simulazione e' stata girata in varie condizioni di modello, temperatura e tensione di alimentazione ma si sono adottate due assunzioni che si sono dimostrate fonte di errore. La prima assunzione e' stata che i parassiti all'interno del contatore di burst fossero trascurabili. La seconda e' stata quella di valutare la frequenza relativamente ai primi clocks di burst. Con queste assunzioni la frequenza di funzionamento risultante dalle simulazioni era ben al di sopra delle aspettative anche nel caso peggiore.



3.2.f) Frequenza di funzionamento di una burst machine (cont.)



Misure di frequenza effettuate sul silicio erano invece mediamente piu' basse dei valori di simulazione ed in particolare il valore minore si otteneva in corrispondenza all'attraversamento del chip (boundary), quando si ritornava dall'ultimo indirizzo al primo.

L'errore era dovuto al fatto che il layout, ottenuto per sintesi automatica del contatore, presentava gli ultimi stadi del contatore molto distanti dagli altri per cui il riporto era piu' lento a propagarsi di quanto il semiperiodo (7.8ns) permettesse.

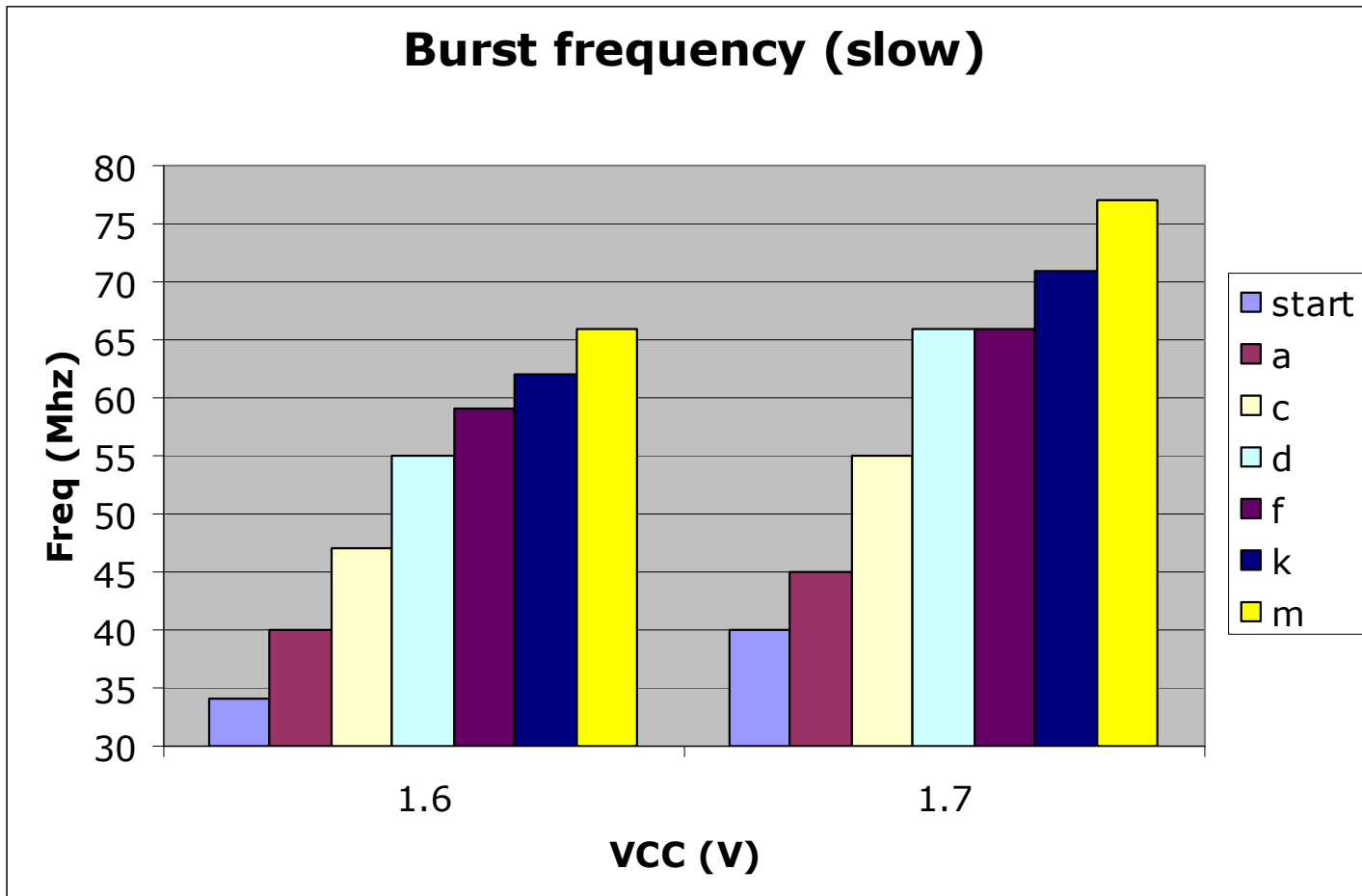
L'errore di boundary era dovuto al fatto che il caso peggiore era proprio il boundary in quanto i riporti dovevano attraversare tutti gli stadi del contatore entro il semiperiodo.

A lato e' riportata una simulazione con tutti i parassiti estratti in back annotazione e con indirizzo iniziale tale da imporre l'attraversamento di boundary.



3.2.f) Frequenza di funzionamento di una burst machine (cont.)

Dopo aver rigirato le simulazioni con back annotazione e includendo la condizione di boundary si sono ottenuti valori di frequenza di 34Mhz (a 1.6V) compatibili con i valori misurati sul silicio (start). Dopo aver potenziato i driver dei carry superiori (a), i driver del carry di word (c), la generazione e propagazione del carry di word (d), il bypass dei mux x16/x32 (f), ed l'avvicinamento del layout degli ultimi stadi (k,m) si e' potuto ottenere il risultato dei 66Mhz, poi confermato sul nuovo silicio.



3.3) Verifica di funzionalita' elettrica globale

La verifica di funzionalita' elettrica globale ha lo scopo di simulare la funzionalita' dell'intero chip coinvolgendo sia le parti analogiche che le parti logiche che le parti di memoria.

L'intento non e' quello di coprire tutti i casi funzionali, oppure tutte le parti del chip, oppure tutte le situazioni di decodifica ma di coinvolgere in singole operazioni tutte quelle parti del chip che per brevitaa di simulazione non si sono potute simulare insieme ma sono state scorporate usando generatori fissi di tensione oppure sostituendo intere porzioni di chip con valori statici.

In questo modo si mettono in evidenza anomalie quali : sovraccarichi, corti, inversione di segnali, tempi di assestamento.

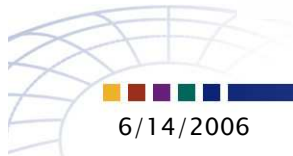
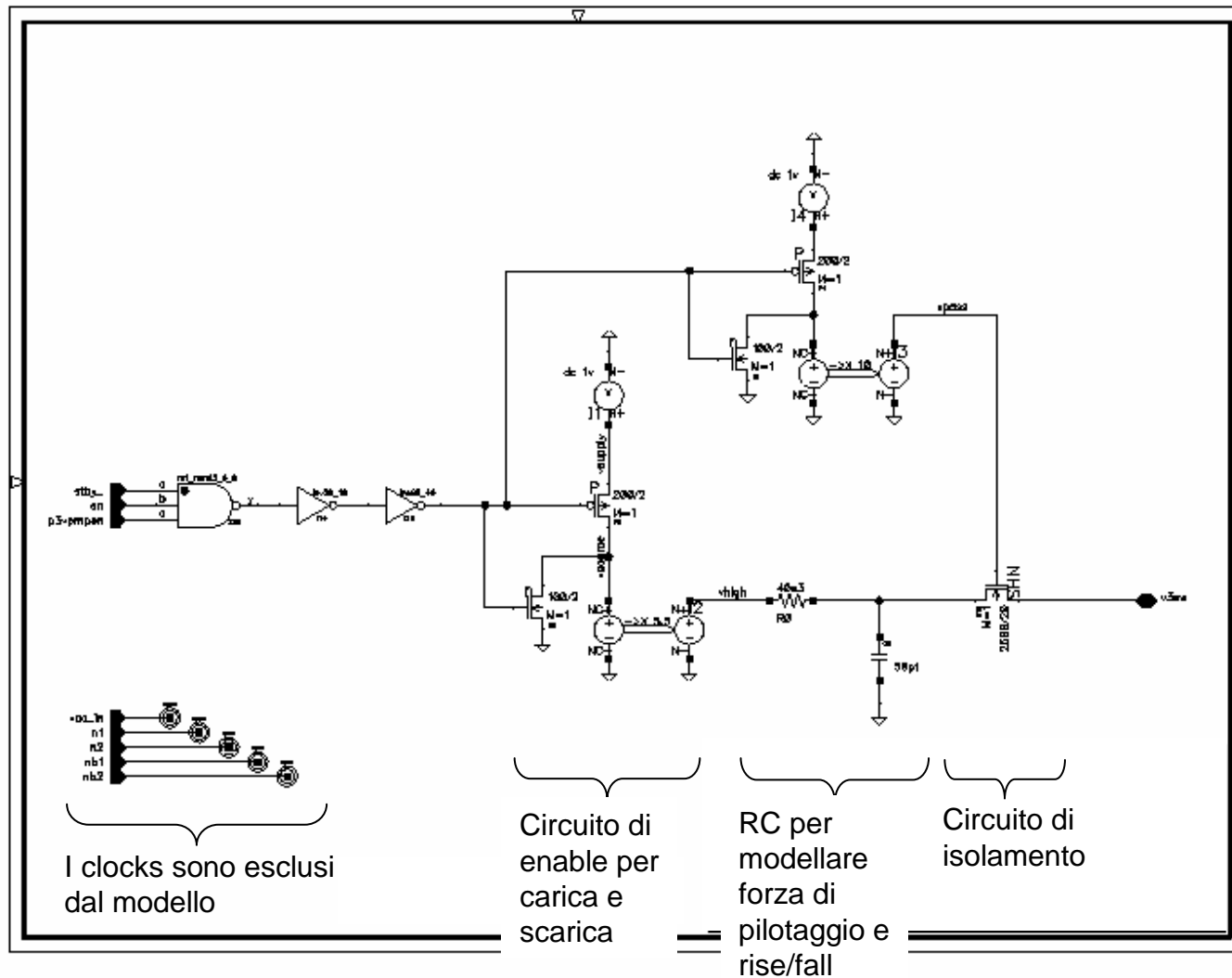
Pur con questi intenti, la verifica di funzionalita' elettrica globale impone delle semplificazioni nel chip. Tipicamente porzioni di generatori di tensione vengono sostituiti con circuiti equivalenti di Thevenin per evitare l'uso di clocks che altrimenti appesantirebbero la simulazione e di circuiti ripetitivi (array di sottocircuiti tutti uguali) che vengono semplificati usando solo il primo e l'ultimo e modellando gli intermedi con parassiti di interconnessioni.

In genere la verifica di funzionalita' elettrica globale richiede lunghi tempi di simulazione (decine di ore) e simulatori in grado di gestire grandi netlist gerarchiche con la possibilita' di effettuare compromessi tra precisione e tempi di simulazione.



3.3) Verifica di funzionalita' elettrica globale (cont.)

Un tipico esempio di modellizzazione di un charge pump e' riportato di seguito



4) Back annotazione

Di fondamentale importanza e' avere la possibilita' di includere nelle simulazioni i parassiti di interconnessione associati un uno specifico layout (back annotazione). A tale proposito e' importante disporre di un tool di estrazione dei parassiti che permetta una gestione accurata e flessibile dei parassiti.

I parassiti da estrarre sono : resistenze di interconnessione (R) e capacita' verso gnd (C) dei diversi rami di layout.

Un tool flessibile di gestione possiede le seguenti caratteristiche :

- 1) disponibilita' di due netlist : una senza parassiti (pre-layout) e una contenente solo i parassiti.
- 2) possibilita' di specificare valori minimi di resistenza (HSIMRMIN) al di sotto dei quali una resistenza parassita viene sostituita da un cortocircuito.
- 3) Possibilita' di specificare valori minimi di capacita' (HSIMCMIN) al di sotto dei quali una capacita' viene considerata un circuito aperto.
- 4) possibilita' di specificare l'uso o meno di back annotation per singolo sotto circuito per ognuno dei quali si puo' specificare se usare solo R, oppure solo C o entrambi.

Un tipico uso di back annotazione puo' essere organizzato in questo modo :

- 1) Un intero chip viene semplificato modellando elettricamente alcuni componenti e semplificando i circuiti che sono ripetuti e non si suppone di dover abilitare
- 2) i vari blocchi componenti si suppongono in prima approssimazione di piccola topologia e quindi privi di parassiti al loro interno
- 3) si modellano le interconnessioni fra i vari blocchi con circuiti RC stimati in base alla distanza di connessione
- 4) man mano che i layout dei singoli blocchi componenti sara' disponibile si ripeteranno le simulazioni includendo la back annotazione di questi singoli blocchi



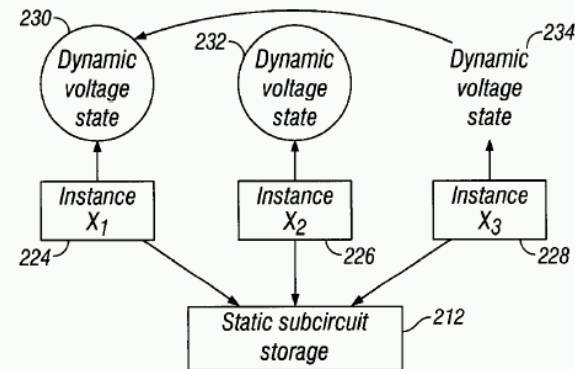
(12) **United States Patent**
Tcherniaev et al.

(10) **Patent No.:** **US 6,577,992 B1**
(45) **Date of Patent:** **Jun. 10, 2003**

(54) **TRANSISTOR LEVEL CIRCUIT SIMULATOR USING HIERARCHICAL DATA**

(75) **Inventors:** **Andrei Tcherniaev**, Mountain View, CA (US); **Iouri Feinberg**, Saratoga, CA (US); **Walter Chan**, Milpitas, CA (US); **Jeh-Fu Tuan**, Saratoga, CA (US); **An-Chang Deng**, Saratoga, CA (US)

(73) **Assignee:** **Nassda Corporation**, Santa Clara, CA (US)



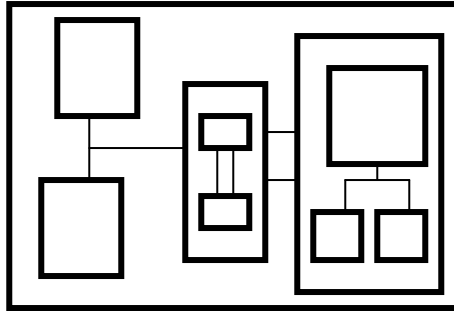
Un simulatore quale SPICE e' basato sul trattamento 'flatten' della netlist di un circuito integrato in cui tutti i componenti sono trattati allo stesso livello e l'evoluzione temporale di tutti i nodi deve essere ricalcolata ad ogni istante con grande occupazione di memoria macchina e lunghi tempi di calcolo. Un circuito simulabile con spice non puo' realisticamente superare i centinaia di migliaia di componenti.

Un simulatore quale HSIM e' basato sul trattamento gerarchico della netlist in cui la topologia di un sottocircuito viene memorizzata una sola volta e la sua evoluzione viene registrata dinamicamente usando il concetto di isomorfismo (solo condizioni elettriche diverse vengono registrate separatamente). Circuiti naturalmente strutturati in modo gerarchico, traggono grande vantaggio da questo concetto e si possono simulare circuiti aventi un numero di componenti dell'ordine di milioni di componenti.

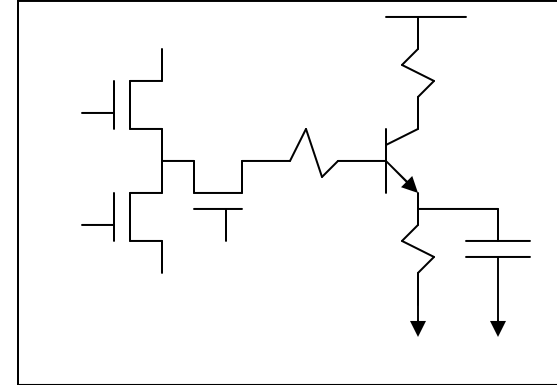
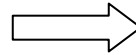


5) Simulatori SPICE e HSIM (cont.)

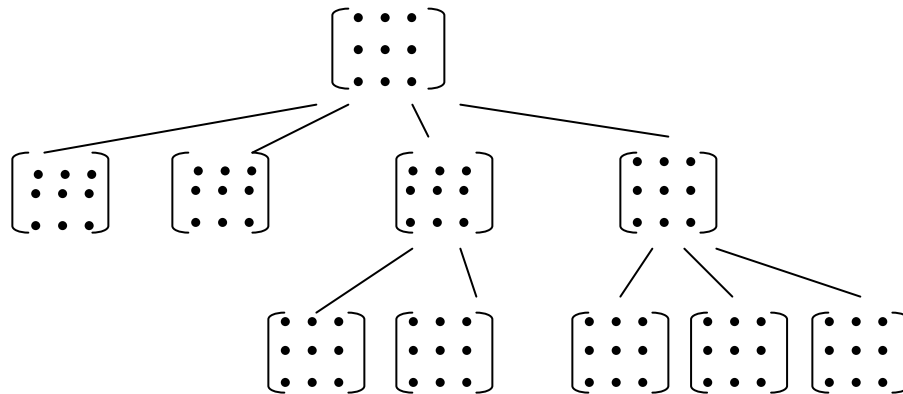
L'organizzazione gerarchica di una netlist permette di avere una struttura di matrice descrittiva del circuito molto piu' rapida da risolvere. Invece di risolvere una unica grande matrice si risolvono solo un numero limitato di piccole matrici : solo quelle per cui sono intervenute delle variazioni. Questo comporta un considerevole risparmio nel tempo di simulazione.



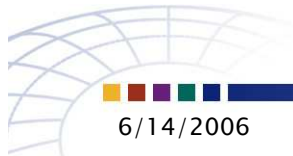
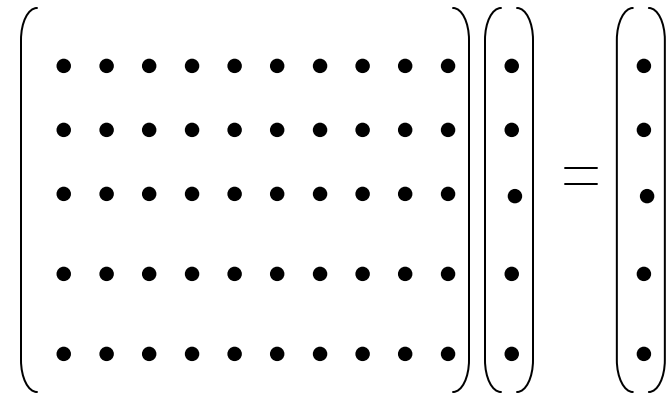
Operazione di
'flattening' di
una netlist



HSIM : organizzazione gerarchica delle matrici che descrivono il circuito



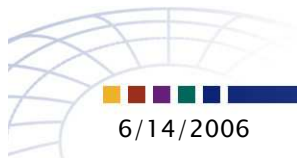
SPICE : organizzazione 'flatten'
della matrice che descrive il circuito



5) Simulatori SPICE e HSIM (cont.)

Il flattening puo' essere controllato dal parametro HSIMFLAT. Se HSIMFLAT=0 non viene applicato il processo di flattening nella generazione della netlist riportandosi ad una organizzazione tipo SPICE.

L'isomorfismo e' controllato dai parametri HSIMPORTV e HSIMPORTCR. Due circuiti sono considerati isomorfi, e quindi possono dividere le stesse risorse per la memorizzazione dei dati, se la tensione alle loro porte differisce per meno di HSIMPORTV e se il rapporto delle capacita' di porte corrispondenti e' inferiore a HSIMPORTCR.



5) Simulatori SPICE e HSPICE (cont.)

Per aumentare la velocità di simulazione, HSPICE permette di effettuare dei compromessi sulla precisione dei risultati della simulazione variando alcuni parametri che intervengono nel corso della esecuzione della simulazione.

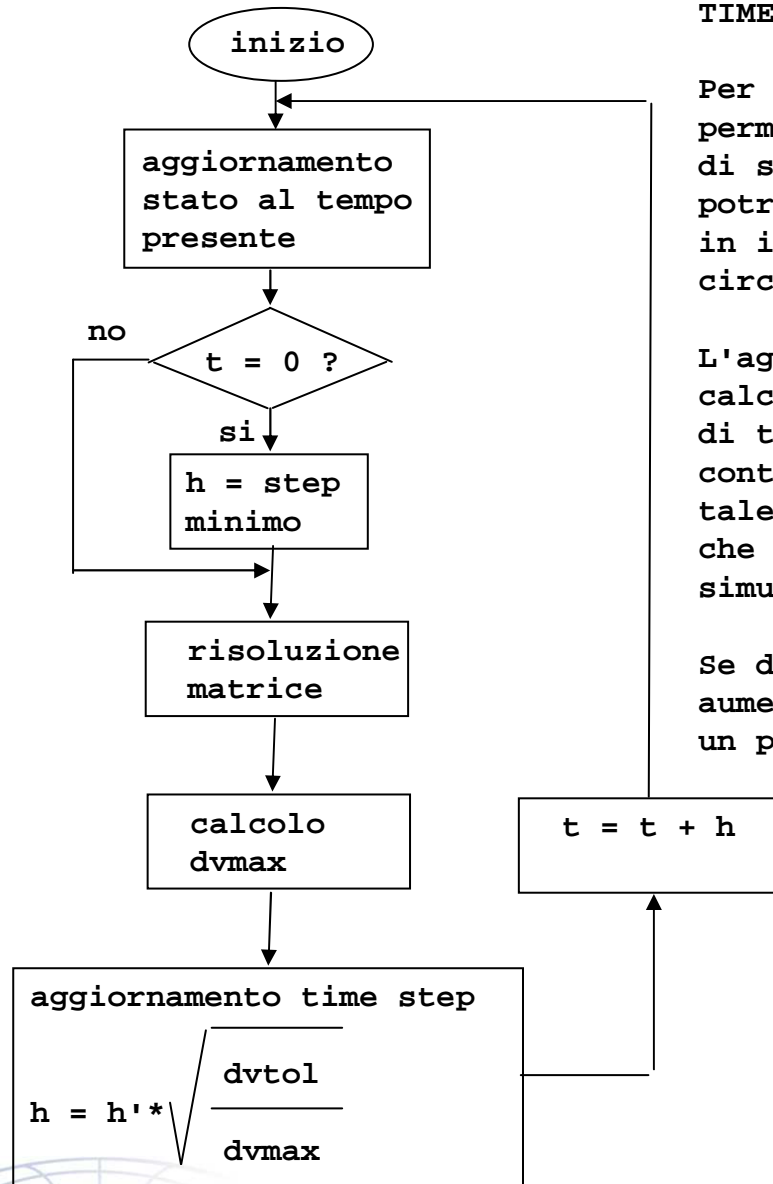
Se si scelgono opportunamente questi parametri in base alle caratteristiche del circuito da simulare, si può ottenere una elevata velocità di esecuzione e allo stesso tempo una buona precisione dei risultati.

I fattori che influenzano la velocità di esecuzione e la precisione sono :

- 1) l'uso del 'time step' di simulazione :
 - costante o variabile (HSPICEMQS)
 - (HSPICEMQS=0 --> time step uguale per tutti i nodi in ogni istante.
HSPICEMQS=1 --> time step variabile a seconda della dinamica del nodo; multiplo del minimo time step tra tutti i nodi del circuito ad un certo istante.
HSPICEMQS=2 --> diverso per ogni nodo e dipendente dalla pendenza di tensione al nodo)
 - massimo incremento di tensione per singolo time step (HSPICEMAXDDV)
 - massimo valore di time step (HSPICEMAX)
- 2) i modelli dei componenti :
 - modelli MOS tabulati oppure equazioni analitiche dei MOS (HSPICESPICE)
 - categorizzazione dei sottoblocchi in analogici e digitali (HSPICEMANALOG)
- 3) algoritmi di calcolo :
 - algoritmi innescati da eventi (in blocchi modellati)
 - latenza (i valori di tensione di un circuito non vengono ricalcolati se la massima variazione di tensione è inferiore ad una tolleranza. Parametri HSPICEMSNCS e HSPICEMSTEADYCURRENT)



5) Simulatori SPICE e HSIM (cont.)



TIME STEP :

Per aumentare la velocità di simulazione, HSIM permette di gestire dinamicamente il time step (h) di simulazione. Infatti avere un time step statico potrebbe produrre inutili lunghi tempi di simulazione in intervalli di tempo in cui i nodi interni di un circuito variano poco.

L'aggiustamento dinamico del time step è effettuato calcolando il valore dv_{max} della massima deviazione di tensione tra tutti i nodi del circuito e tenendo conto di una tolleranza $dvtol$ che si vuole imporre a tale dv_{max} . $dvtol$ è un parametro globale `HSIMALLOWEDDV` che può essere variato in varie fasi temporali della simulazione.

Se $dv_{max} > dvtol$, h viene diminuito altrimenti viene aumentato. Per evitare aumenti eccessivi di h, esiste un parametro `HSIMTAUMAX` che ne limita il valore.



5) Simulatori SPICE e HSIM (cont.)

MODELLIZZAZIONE :

Il parametro HSIMSPICE permette di definire la tipologia di modellizzazione dei componenti MOS

- HSIMSPICE = 0 comporta la generazione di una tabella di valori descrittivi di IV a partire dalle equazioni SPICE. La tabella viene generata all' inizio di una simulazione e permette in seguito un notevole risparmio di tempo di simulazione

	Vgs	Vds
Ids	●	●
⋮		
Ids	●	●
Ids	●	●

	Vbs
data	●
⋮	
data	●
data	●

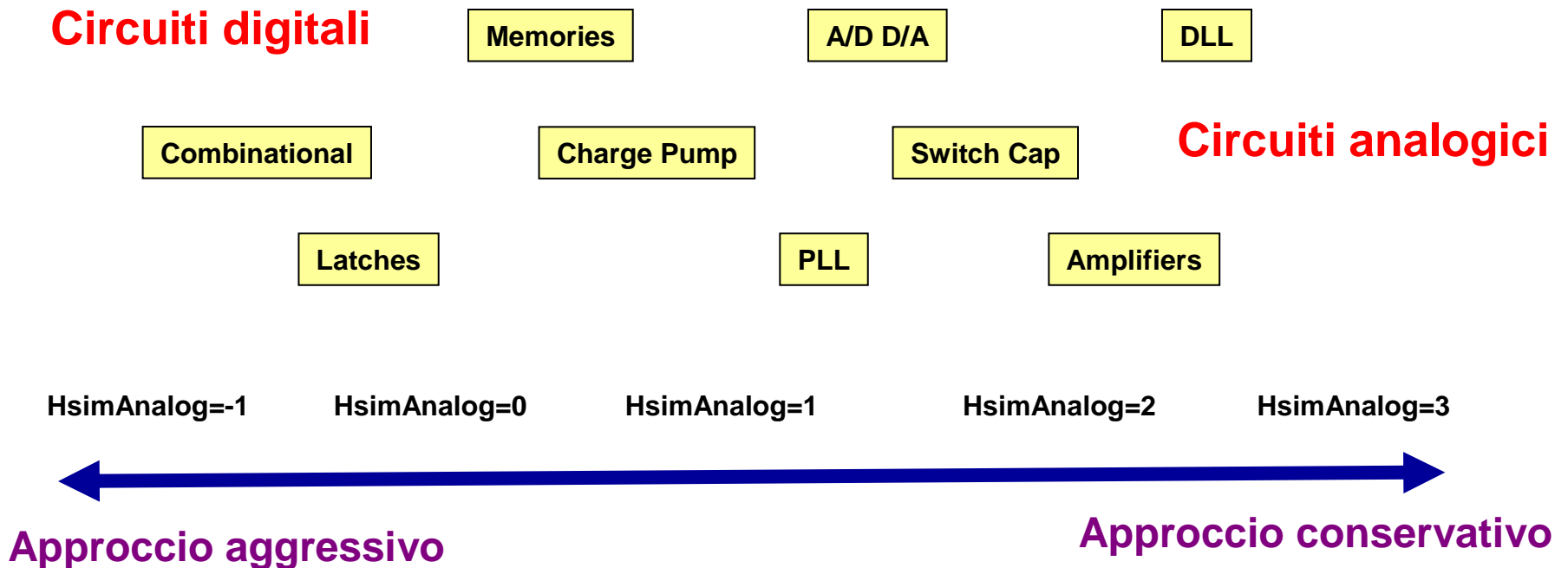
- HSIMSPICE = 1,2,3 comportano invece l'uso di equazioni SPICE per i componenti MOS



5) Simulatori SPICE e HSIM (cont.)

Per ottenere le migliori prestazioni da HSIM e' opportuno considerare l'uso del parametro HSIMANALOG, che permette di simulare alcuni sottoblocchi di un circuito complesso con una precisione adeguata alla sua funzionalita'. In particolare, circuiti tipicamente analogici, in cui e' molto importante tener conto di controreazioni o regolazioni, possono essere trattati con una precisione tipica di SPICE. a scapito del tempo di simulazione.

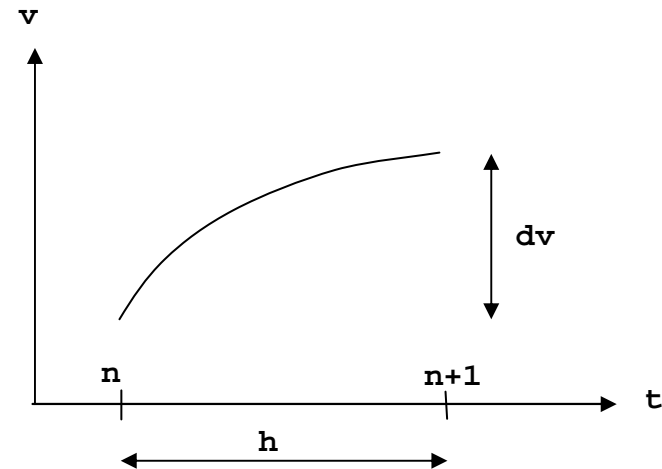
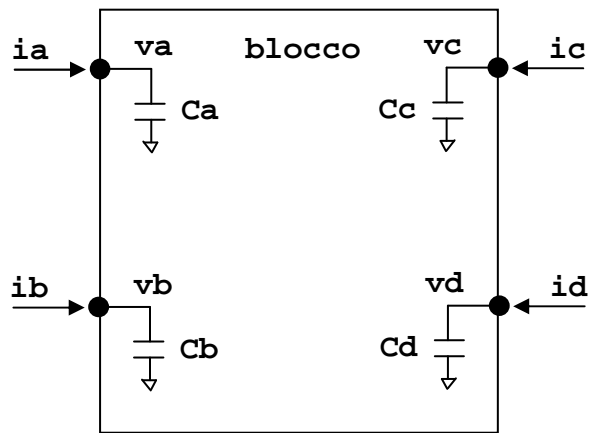
Una situazione tipica di categorie di circuito per cui e' il caso di applicare il parametro HSIMANALOG e' riportata di seguito.



5) Simulatori SPICE e HSIM (cont.)

LATENZA :

Usando il concetto di latenza si puo' aumentare la velocita' di esecuzione di una simulazione HSIM. Se infatti il parametro HSIMSNCS e' posto uguale a 1, tutti i nodi interni ad un blocco non verranno aggiornati se il blocco e' 'idle', cioe' se durante un time step le variazioni di tensione ai bordi del blocco sono inferiori ad un certo valore.



$$\text{Se } i = C * \frac{dv}{dt} = C * \frac{|v_{n+1} - v_n|}{h} < \text{HSIMSTEADYCURRENT}$$

$$\text{cioe' se } |v_{n+1} - v_n| < h * \frac{\text{HSIMSTEADYCURRENT}}{C}$$

per ogni nodo al bordo del blocco, il blocco e' considerato 'idle' e non vengono aggiornati i valori di tensione all'interno del blocco.



5) Simulatori SPICE e HSIM (cont.)

C'e' un parametro in HSIM che permette di definire sinteticamente il compromesso tra precisione e velocita'. Tale parametro e' HSIMSPEED e raggruppa una serie di altri parametri ognuno dei quali definisce una proprieta' di compromesso tra precisione e velocita'.

← Precisione
→ Velocita' →

HsimSpeed	0	1	2	3	4	5	6	7	8
HsimSNCS	0	1	1	1	1	1	1	1	1
HsimFlat	0	0	1	1	1	1	1	1	1
HsimGCSC	1	1	1	1	1	0	0	0	0
HsimMqs	0	0	0	1	1	1	2	1	1
HsimAllowedDv	0.1	0.3	0.3	0.3	0.3	0.5	0.3	0.5	0.5
HsimSteadyCurrent	10nA	10nA	10nA	10nA	100nA	100nA	100nA	100nA	100nA



6) Verifica logica di un circuito integrato

La verifica logica di un circuito integrato si propone di effettuare verifiche esaustive di numerose (eventualmente tutte) situazioni di possibile funzionamento del circuito.

Tipici esempi sono :

- tutte le possibili situazioni di configurazione di una burst machine
- tutti i possibili comandi di una command user interface (CUI) di una memoria FLASH
- varie possibilita' di concorrenza (read in una partizione mentre si programma un'altra partizione) in una memoria FLASH per applicazioni wireless
- scansione di tutti i possibili indirizzi di settore per verificare il decoder di blocco in una memoria
- scansione di tutti i possibili indirizzi di riga per verificare il word decoder di una memoria

Questa esaustivita' e' possibile perche' le simulazioni logiche non tengono conto del comportamento elettrico del circuito e quindi sono molto rapide. Lo svantaggio e' che percorsi e concomitanze di segnali non sono realistici e quindi i risultati di una simulazione logica offrono garanzia di bonta' della verifica solo quando il design elettrico ha gia' garantito una integrita' dell'intero circuito.



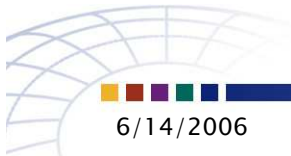
6) Verifica logica di un circuito integrato (cont.)

Per ovviare all'inconveniente delle simulazioni logiche di non tener conto di aspetti elettrici del circuito si e' di recente introdotta anche per i simulatori logici la possibilita' di tener conto di una forma di back annotazione a partire dal layout. Infatti e' possibile estrarre automaticamente dal layout i ritardi di propagazione dei segnali in base a drive, lunghezza di routing a fan-out e di introdurre questi ritardi nelle simulazioni logiche.

La esaustivita' di una simulazione logica non puo' esser comunque totale. Si pensi ad esempio al caso in cui si volesse verificare la correttezza di scrittura/lettura in una memoria coprendo tutti i possibili valori di dato. Anche in questo caso, pero', si sono introdotte di recente delle tecniche di simulazione che permettono un tale tipo di copertura (simulazioni simboliche).

Una potente caratteristica dei simulatori logici, oltre che alla esaustivita' e alla velocita', e' la versatilita' degli stimoli. Infatti :

- e' possibile definire un insieme di stimoli (tasks) che possono essere richiamati sotto forma di macro in file di stimoli
- e' possibile dare comandi in un file di stimoli solo dopo che si sono verificate opportune condizioni in segnali interni al circuito
- e' possibile effettuare delle verifiche automatiche sui risultati delle simulazioni



6.1) Verifica logica di un circuito integrato : stimoli e interattivita'

programma principale

```
File Edit Search Preferences Shell Macro
Windows Help

// operazione di read in una
// memoria NAND FLASH

// richiesta di read
cmd_latch(8'h00);
add_latch(8'h00);
add_latch(8'h00);
add_latch(8'h00);
add_latch(8'h00);
add_latch(8'h02);
cmd_latch(8'h30);

// attesa del download di una pagina
// da array a cache

wait(test.top.mid.Lb.algo_busy);
wait(!test.top.mid.Lb.algo_busy);

// verifica automatica del
// risultato atteso

#1000 serial_access(8'h00);
#1000 serial_access(8'h01);
#1000 serial_access(8'h02);
#1000 serial_access(8'h03);
```

task di comandi

```
File Edit Search Preferences Shell Macro
Windows Help

task cmd_latch;
input [15:0] data;
fork
    p_reb_optale= 1'b1;
    p_ceb_optcle = 1'b0;
    #(`tcs+`tch) p_ceb_optcle = 1'b1;
    #(`twp-`tcls) p_cle_optceb = 1'b1;
    #(`twp+`tclh) p_cle_optceb = 1'b0;
    p_we_ = 1'b0;
    #(`twp) p_we_ = 1'b1;
    #(`twp-`tds) io_padq = data;
    #(`twp+`tdh) io_padq = 16'hz;
join
endtask
```

attesa di condizioni

task per la verifica automatica

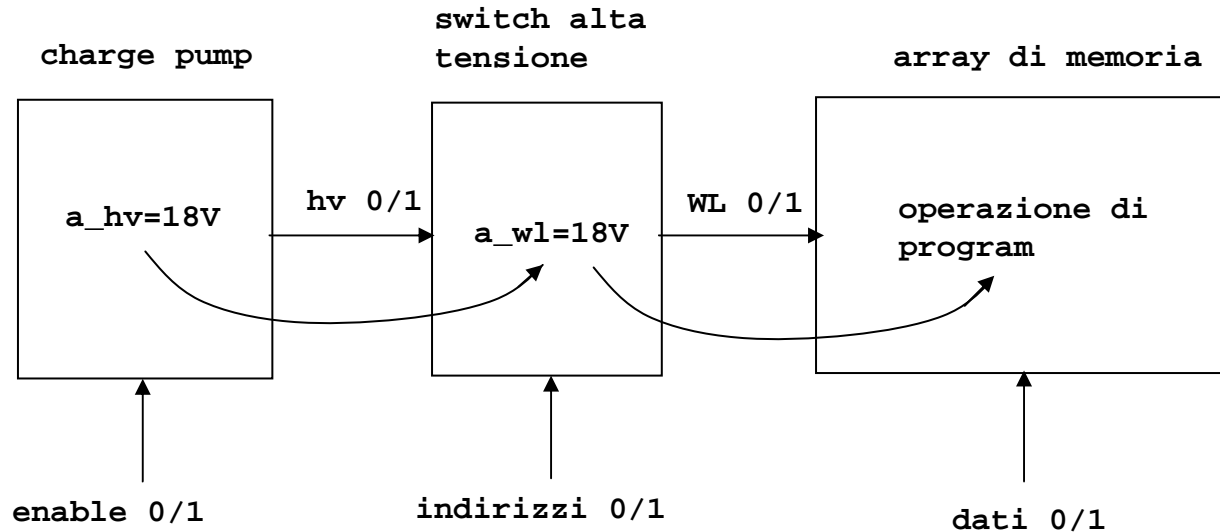
```
File Edit Search Preferences Shell Macro Windows Help

task serial_access;
input [7:0] data;
fork
    p_ceb_optcle = 1'b0;
    p_reb_optale = 1'b0;
    #(`trp) p_reb_optale = 1'b1;
    #(`trp) if (test.top.padq[7:0] != data[7:0])
        $display($time,"DATA ERROR : READ(DQ)=%h EXPECTED(DQ)=%h", test.top.padq[7:0], data);
    else
        $display($time,"DATA OK : READ(DQ)=%h EXPECTED(DQ)=%h", test.top.padq[7:0], data);
join
endtask
```



6.2) Verifica logica di un circuito integrato : modellizzazione di circuiti analogici

Quando in un circuito integrato sono presenti dei blocchi analogici quali ad esempio charge pumps, sense amplifiers, decodifiche che coinvolgono alte tensioni, si puo' procedere ad una loro modellizzazione per includerli in ambiente di simulazioni logiche.



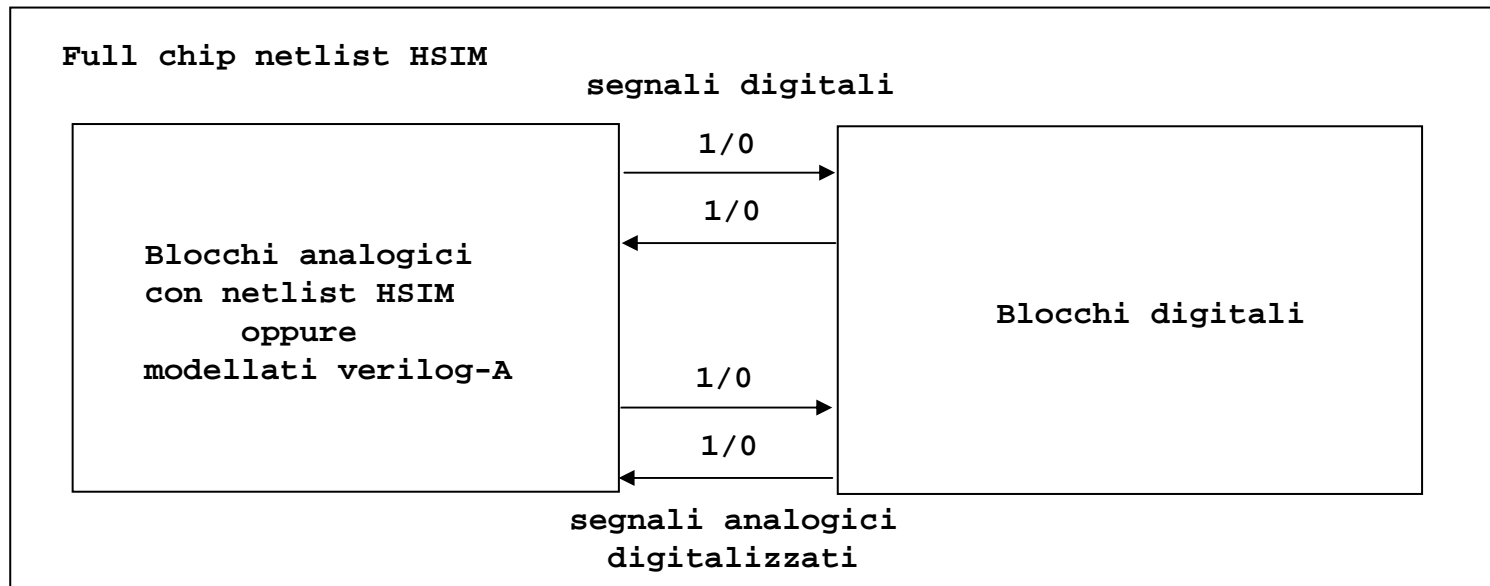
Tale modellizzazione puo' essere illustrata facendo riferimento all'esempio di una operazione di program di una array di memoria FLASH attraverso un charge pump e uno switch di alta tensione :

- il blocco analogico 'charge pump', in conseguenza del segnale di ingresso digitale enable produce una uscita digitale hv a cui e' associata una variabile reale a_hv interna al modello che assume un valore analogico di 18V.
- quando il blocco analogico 'switch alta tensione' riceverà il valore logico 1 al suo ingresso digitale hv, la variabile a_hv verrà letta ed il suo valore di 18V verrà trasferito alla variabile reale a_wl interna al modello e contemporaneamente il segnale logico WL verrà posto ad 1.
- quando il blocco 'array di memoria' riceverà il valore logico 1 al suo ingresso WL effettuerà una lettura di a_wl e se a_wl=18V avvierà una operazione di trasferimento dei dati all'interno della memoria (program).



7) Simulazioni mixed-mode : cosim e verilog-A

Un circuito complesso costituito da blocchi analogici e digitali puo' essere simulato con simulatori opportuni (cosimulatori) che tengono conto di questa duplice natura e simulano la parte analogica con simulatori elettrici (HSIM) e la parte digitale con simulatori tipo verilog.



In questo modo si trae il maggior beneficio in termini di precisione e velocita' dai due ambienti. L'intero circuito e' diviso in due grandi aree : analogica e digitale che si scambiano segnali. Alcuni di questi segnali sono digitali per propria natura altri sono analogici ma vengono digitalizzati da una opportuna interfaccia. Quando un segnale analogico supera certe prefissate soglie viene considerato come 0 oppure 1.

I blocchi analogici possono essere anche opportunamente modellati usando il linguaggio verilog-a in grado di generare segnali analogici che poi vengono digitalizzati all'interfaccia.

