

UNIVERSITA' DEGLI STUDI DI GENOVA

SCUOLA POLITECNICA



Corso di Studi in Ingegneria Elettronica

Progetto e simulazione di un circuito Full Adder TSPC

Alessio Caligiuri, Emanuele Anfuso

Indice

1	Introduzione	1
1.1	Fasi di lavoro	1
2	Analisi	5
2.1	Le caratteristiche del MOS	5
2.1.1	Il modello semplificato del MOS	5
2.1.2	La caratteristica reale del MOS	6
2.1.3	Il modello <i>level 3</i> di <i>MicroWind</i>	7
2.2	Funzionamento dinamico dei circuiti CMOS	8
2.3	Il Full Adder TSPC	11
3	Progettazione circuitale	14
3.1	Schema completo	14
3.2	Dimensionamento di massima	15
3.2.1	Vincoli temporali e <i>inverter equivalente</i>	15
3.2.2	Dall' <i>inverter equivalente</i> alle dimensioni dei MOS	16
3.2.3	Dimensioni geometriche dei MOS: risultati di <i>MATLAB</i>	17
3.2.4	Prima simulazione <i>pre-layout</i>	17
3.3	Fitting dei parametri: dimensioni definitive	19
3.3.1	Simulazione <i>pre-layout</i> soddisfacente	20
4	Layout	22
4.1	Disegno dei singoli stadi	23
4.1.1	Stadio <i>1</i> - Generazione di !CARRY	23
4.1.2	Stadio <i>2.1</i> - Generazione di !SUM	24
4.1.3	Stadi finali <i>2.2</i> e <i>3</i> - Generazione di CARRY e SUM	25
4.2	Full design	25
5	Conclusioni finali	28

Capitolo 1

Introduzione

Questa relazione illustra il progetto del layout di un circuito sommatore CMOS integrato che sfrutti la tecnica TSPC (*True Single Phase Clock*) presentata da *Yuan e Svensson* nell'articolo [1]. Il lavoro si inserisce nell'ambito del corso di Microelettronica tenuto dal Prof. Daniele Caviglia nella Laurea Magistrale in Ingegneria Elettronica dell'Università degli Studi di Genova.

L'elaborato vuole descrivere nel dettaglio il lavoro svolto, i risultati raggiunti e le problematiche riscontrate durante lo stesso. Inizieremo dunque con l'illustrare i vari passi percorsi, per poi scendere nel dettaglio delle fasi di analisi (sia della tecnologia utilizzata che del circuito sommatore) e di progettazione vera e propria.

1.1 Fasi di lavoro

In fig. 1.1 è rappresentato il diagramma di flusso che abbiamo seguito durante il nostro lavoro. Tale sequenza di passi è generalizzabile a qualsiasi progetto di circuiti digitali, quando si parte da uno schema circuitale e si vuole giungere al disegno delle maschere di produzione che lo realizzano rispettando le specifiche desiderate. Nel diagramma ogni rettangolo indica l'operazione da eseguire, mentre il rombo comporta un controllo che determina il passo successivo da intraprendere.

Prima di scendere nel dettaglio delle varie fasi introduciamo i vari software utilizzati nel corso del progetto. Già indicati nel diagramma di flusso, essi sono:

- *MATLAB*: software che facilita l'esecuzione e la gestione di calcoli matematici;

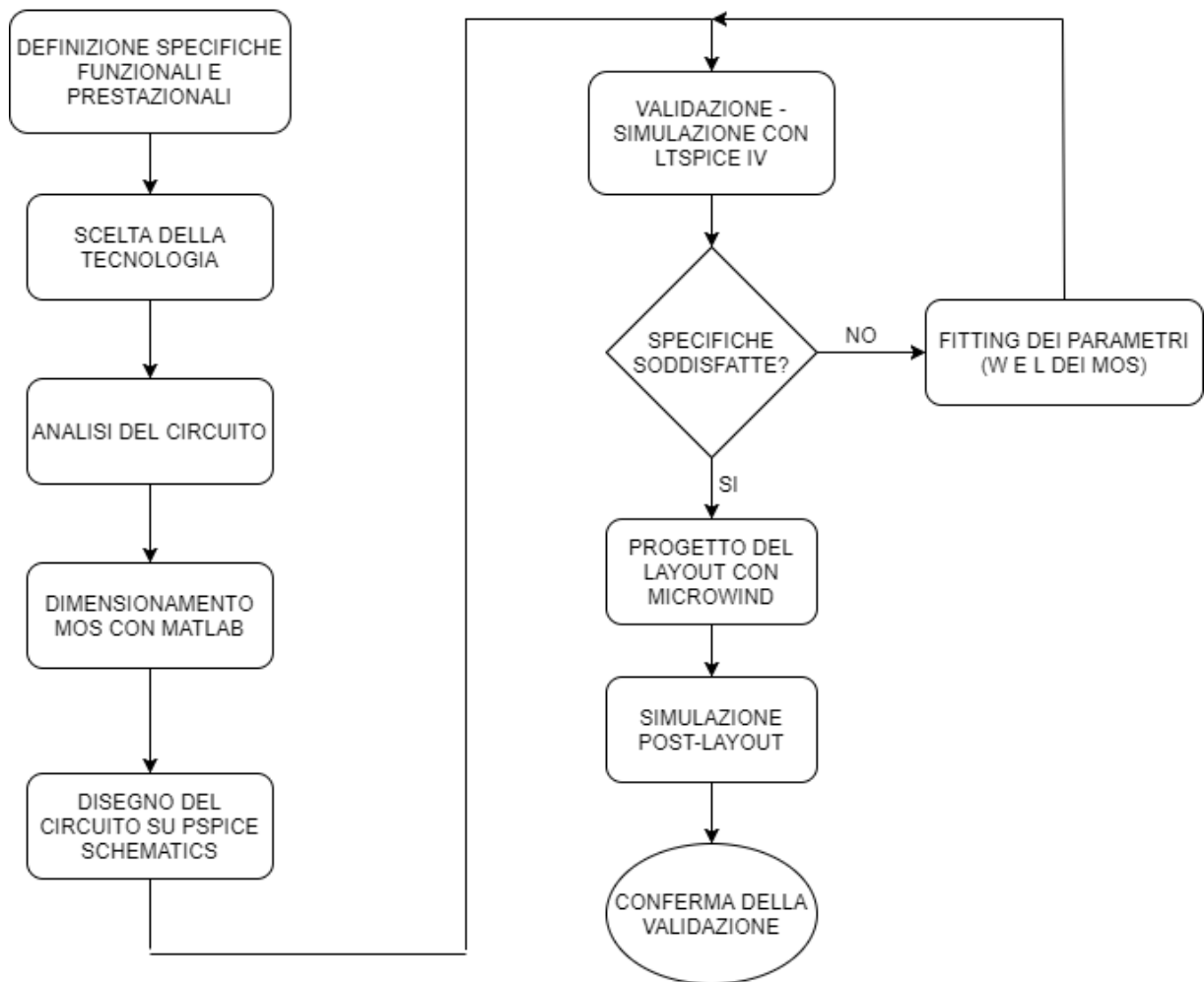


FIGURA 1.1: Flusso di progetto.

- *PSpice*: programma di simulazione circuitale;
- *LTspice IV*: anch'esso dedicato alla simulazione circuitale, lo abbiamo scelto in quanto il precedente PSpice nella versione gratuita presenta alcune restrizioni, in particolare sul numero massimo di transistor presenti nel circuito;
- *Microwind*: software per il disegno di maschere di produzione di circuiti integrati.

Procediamo quindi con la descrizione del flusso di progetto di fig. 1.1 in relazione al nostro lavoro. Il punto di partenza è la definizione delle specifiche di progetto. Queste nel nostro caso ci sono state fornite come consegna e sono riassunte nel seguente elenco:

- Tensione di alimentazione V_{DD} : 1.2 V
- Capacità di carico C_{load} sulle uscite *SUM* e *CARRY*: 100 fF

- Frequenza operativa f : 2 GHz
- Tipologia di progettazione del circuito: *standard cell*
- Segnali di ingresso: tre generatori di tensione ideali, con tempi di salita e di discesa pari a 25 ps

Anche il tipo di tecnologia ci è stato indicato nella consegna del progetto: tecnologia CMOS $0.12\mu m$. Questo dato è molto importante nel progetto di circuiti integrati, in quanto tra le altre cose indica la lunghezza minima di canale che possono avere i transistor MOS.

La terza fase prevede l'analisi dello schema circuitale del Full Adder presente nell'articolo citato nell'introduzione. Nella sezione 2.3 sarà illustrata l'analisi in dettaglio. A partire da quest'ultima, come vedremo, è stato possibile individuare le relazioni analitiche che ci hanno permesso di legare i nostri parametri di progetto, ottenuti a partire dalle specifiche funzionali e prestazionali, con i parametri liberi, ovvero le dimensioni dei transistor MOS presenti nel circuito.

Il dimensionamento dei MOS è per l'appunto la quarta fase del processo e ne rappresenta uno dei momenti più critici, in quanto getta le basi per il funzionamento del circuito finale. Esso tuttavia è basato sull'utilizzo di modelli analitici approssimati, per cui le dimensioni dei MOS trovati a questo passo non sono quelle definitive. Nel cap. 3, dedicato alla progettazione circuitale, parleremo in dettaglio di questo aspetto.

Ottenuta una prima approssimazione delle dimensioni dei transistor si procede con la quinta fase, la quale prevede il disegno del circuito su un software che ne permetta la simulazione. In particolare noi abbiamo disegnato il circuito su *PSpice Schematics*, ne abbiamo esportato la netlist, e abbiamo simulato quest'ultima con *LTspice IV*, per i problemi di cui abbiamo già parlato, impiegando per le simulazioni il modello del MOS fornito da *Microwind* e dipendente dalla tecnologia di processo scelta. Anche questo passaggio sarà esaurientemente descritto nel cap. 3.

Il passo successivo è uno dei più interessanti. La fase precedente ha infatti permesso di ottenere una simulazione del circuito dimensionato per via analitica, ma come abbiamo già detto non ci si deve stupire se il risultato non è soddisfacente. Occorre un'attenta analisi dei segnali presenti in ogni ramo del circuito così da individuare eventuali punti critici e perfezionare di conseguenza le dimensioni dei MOS. Inizia così una fase di *fitting dei parametri*, composta da un ciclo di analisi e ritocco delle dimensioni, che termina soltanto quando ci si ritiene soddisfatti della simulazione circuitale.

Identificate le dimensioni finali dei transistor si procede al progetto e disegno del layout di circuito su *Microwind*, secondo l'approccio *standard cell*. Questa fase sarà descritta nel Cap. 4 e, una volta terminata, è seguita da una simulazione finale *post-layout* che, se in accordo con le specifiche, consente di validare il disegno finale.

Capitolo 2

Analisi

2.1 Le caratteristiche del MOS

2.1.1 Il modello semplificato del MOS

Dato che il circuito trattato in questo progetto è basato su tecnologia CMOS è doveroso iniziare con l'analisi del *mattoncino base* che lo compone, ossia il transistor di tipo MOSFET.

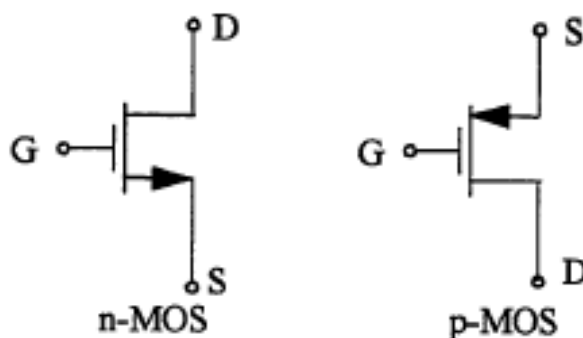


FIGURA 2.1: Simbolo NMOS (a sinistra) e PMOS (a destra)

Il modello semplificato introduce una *tensione di soglia* V_{th} e prevede che il dispositivo possa operare in tre zone di funzionamento, illustrate di seguito. Per semplicità consideriamo un NMOS; per il PMOS i segni di correnti e tensioni sono invertiti.

- *Interdizione* - la V_{GS} è sotto la soglia e non si è formato il canale, quindi la corrente di drain è nulla:

$$V_{GS} < V_{th_n} \quad \Rightarrow \quad I_D = 0. \quad (2.1)$$

- *Zona lineare* - V_{GS} è sopra alla soglia e il canale si estende dal drain al source:

$$\begin{cases} V_{GS} \geq V_{th_n} \\ V_{DS} < V_{GS} - V_{th_n} \end{cases} \Rightarrow I_D = \mu_n C'_{ox} \frac{W}{L} \left[(V_{GS} - V_{th_n}) V_{DS} - \frac{V_{DS}^2}{2} \right] \quad (2.2)$$

se V_{DS} è piccola il MOS si comporta come un resistore controllato da V_{GS} (relazione lineare tra V_{DS} e I_D):

$$I_D \simeq \mu_n C'_{ox} \frac{W}{L} (V_{GS} - V_{th_n}) V_{DS}. \quad (2.3)$$

- *saturazione* - V_{GS} è sopra alla soglia e il canale presenta uno strozzamento in prossimità del drain; il MOS si comporta come un generatore di corrente I_D costante, pilotato dalla tensione V_{GS} :

$$\begin{cases} V_{GS} \geq V_{th_n} \\ V_{DS} > V_{GS} - V_{th_n} \end{cases} \Rightarrow I_D = \frac{1}{2} \mu_n C'_{ox} \frac{W}{L} (V_{GS} - V_{th_n})^2. \quad (2.4)$$

Le grandezze μ_n (*mobilità dei portatori di carica*) e C'_{ox} (*capacità dell'ossido per unità di area tra gate e canale*) dipendono dalla tecnologia impiegata. Nel nostro caso (*Microwind* 0.12 μm) si ha:

- NMOS: $V_{th_{n,0}} = 0.40 \text{ V}; \quad \mu_n = 600 \frac{cm^2}{V \cdot s}$
- PMOS: $V_{th_{p,0}} = -0.45 \text{ V}; \quad \mu_p = 200 \frac{cm^2}{V \cdot s}$

mentre per entrambi l'ossido ha spessore $t_{ox} = 2.0 \cdot 10^{-9} \text{ m}$ e permittività elettrica relativa $\epsilon_{rSiO_2} = 3.9$. Si ricava quindi:

$$C'_{ox} = \frac{\epsilon_0 \cdot \epsilon_r}{t_{ox}} = \frac{8.85 \cdot 10^{-12} [F/m] \cdot 3.9}{2.0 \cdot 10^{-9} [m]} \simeq 17.26 \cdot 10^{-3} \frac{F}{m^2} \quad (2.5)$$

2.1.2 La caratteristica reale del MOS

Il modello del MOS mostrato nella sezione 2.1.1 è ben distante dalla realtà, perché si manifestano:

- *effetti di canale corto*, particolarmente visibili quando i transistor sono realizzati con la lunghezza minima disponibile per la tecnologia, come nel caso dei dispositivi

digitali, in cui si vogliono minimizzare le dimensioni; ciò comporta la progressiva riduzione della velocità dei portatori di carica nel canale, diminuendo il fattore μ e quindi il guadagno;

- *effetto body*, per il quale la V_{th} diminuisce all'aumentare della tensione V_{SB} ; nei dispositivi integrati spesso il source del MOS non è collegato al bulk (ovvero il substrato) e quindi $V_{SB} \neq 0 \Rightarrow |V_{th}| < |V_{th0}|$ (con V_{th0} tensione di soglia per $V_{SB} = 0$).

2.1.3 Il modello *level 3* di *MicroWind*

Il software *MicroWind* fornisce un modello *SPICE* del MOS di *livello 3*, ben distante dalla caratteristica semplificata illustrata in sezione 2.1.1. Di seguito, proponiamo un confronto tra i due modelli, facendo affidamento a curve caratteristiche ottenute mediante simulazione con *LTspice*.

I grafici seguenti fanno riferimento ad un pMOS a dimensioni minime ($W = L = 0.12\mu m$) generato da *MicroWind*.

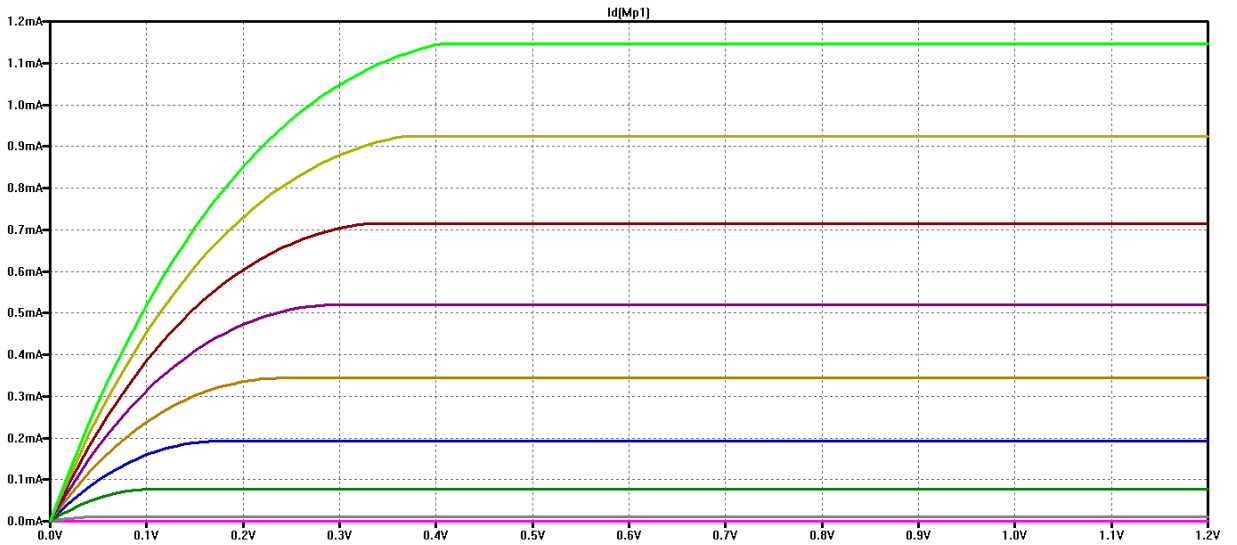


FIGURA 2.2: Curve caratteristiche di un pMOS - modello *level 3* di *MicroWind*.

Inizialmente abbiamo simulato il pMOS con il modello *level 3* esattamente come generato da *MicroWind*. Le curve nel piano $I_{DS} - V_{DS}$ sono ottenute per incrementi costanti di V_{GS} tra 0V e $V_{DD} = 1.2V$. Le correnti sono decisamente più basse rispetto a quanto atteso applicando le formule semplificate presentate in sezione 2.1.1. Ad esempio, per $V_{GS} = 1.2V$, in saturazione si ha $I_D \approx 1.15mA$.

Per riportare il transistor ad un funzionamento più vicino al nostro modello quadratico approssimato, abbiamo provato ad ingannare il simulatore semplificando il modello *SPICE*. In particolare dal listato di fig. 2.3 (per il pMOS, analogo per l'nMOS) abbiamo rimosso le prime due righe rosse, corrispondenti a parametri inesistenti nel nostro modello approssimato che comportano una notevole degradazione delle prestazioni dei transistor.

```
* p-MOS Model 3:
* low leakage
.MODEL P1 PMOS LEVEL=3 VTO=-0.45 U0=200.000 TOX= 2.0E-9
+LD =0.0000 THETA=0.300 GAMMA=0.400
+PHI=0.200 KAPPA=0.060 VMAX=110.00K
+CGSO=100.0p CGDO=100.0p
+CGBO= 60.0p CJSW=240.0p
*
```

FIGURA 2.3: Modello *SPICE* del pMOS fornito da *MicroWind*.

Nelle nuove curve caratteristiche, per $V_{GS} = 1.2V$, in saturazione si ha $I_D \approx 2.9mA$. Il risultato è aderente con quanto atteso applicando le formule semplificate, tuttavia non ha alcuna rilevanza pratica.

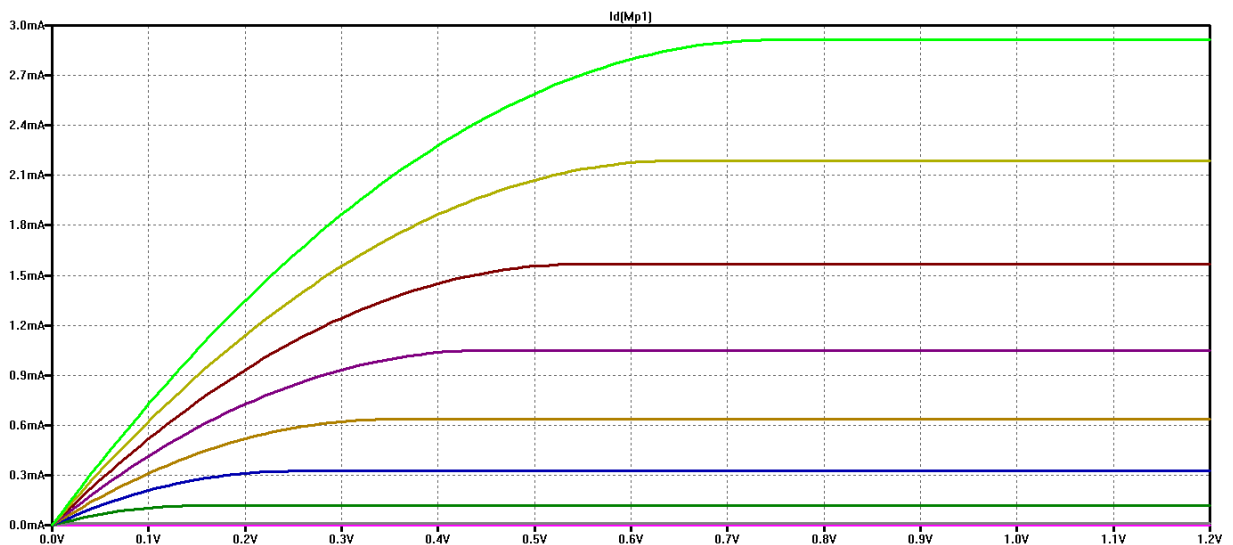


FIGURA 2.4: Curve caratteristiche di un pMOS - modello semplificato.

2.2 Funzionamento dinamico dei circuiti CMOS

Per valutare le prestazioni dinamiche della tecnologia CMOS, consideriamo come caso di studio un inverter con carico capacitivo ed effettuiamo un'analisi temporale fornendo in ingresso un'onda quadra.

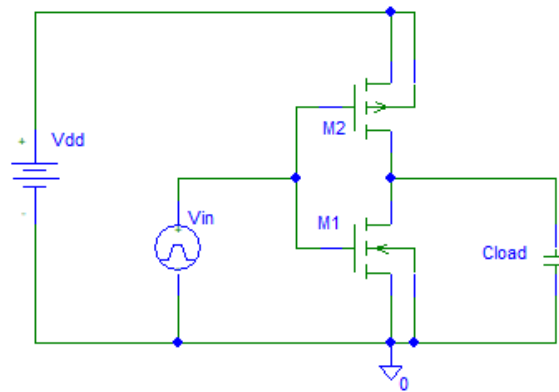


FIGURA 2.5: Inverter CMOS con capacità di carico.

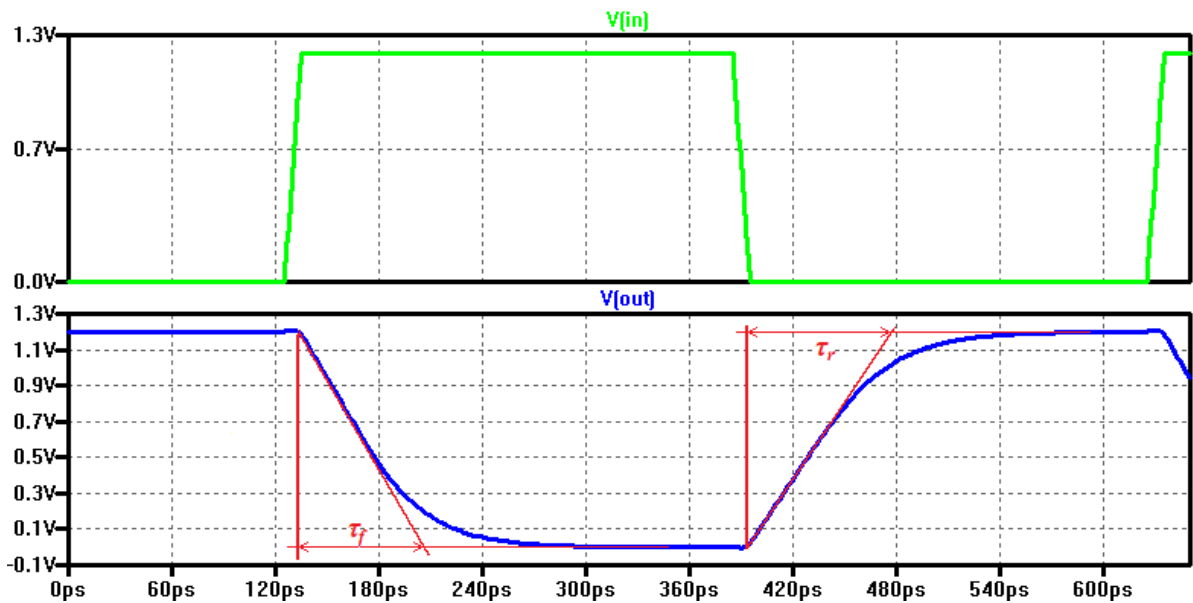


FIGURA 2.6: Simulazione temporale di un inverter CMOS con capacità di carico.

Consideriamo un fronte di discesa della tensione d'uscita: in questo caso il condensatore, precedentemente caricato a V_{DD} viene scaricato dal nMOS che funziona inizialmente in regime di saturazione ($V_{DS} > V_{GS} - V_{thn}$) per poi concludere la scarica in zona lineare.

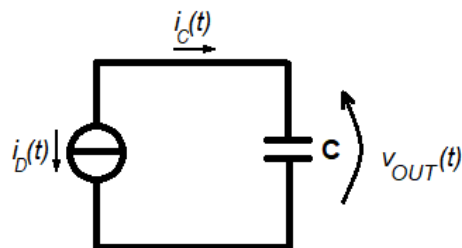


FIGURA 2.7: circuito equivalente per la scarica della capacità.

Questa situazione è schematizzata in figura 2.7. La corrente $i_D(t) = -i_C(t)$ è quella che scorre nel canale del nMOS scaricando il condensatore. Si ha che:

$$i_C(t) = C \frac{d}{dt} v_{OUT}(t) \quad (2.6)$$

da cui, integrando tra t_0 e t , si ottiene l'evoluzione temporale di $v_{OUT}(t)$ a partire dalla condizione iniziale $v_{OUT}(t_0)$:

$$\int_{t_0}^t i_C(\xi) d\xi = C \int_{t_0}^t \frac{d}{d\xi} v_{OUT}(\xi) d\xi \quad (2.7)$$

ovvero, calcolando l'integrale a secondo membro e riordinando i termini:

$$v_{OUT}(t) - v_{OUT}(t_0) = \frac{1}{C} \int_{t_0}^t i_C(\xi) d\xi \quad (2.8)$$

Un'approssimazione valida per semplificare l'analisi si ottiene supponendo che il transistor operi solo in saturazione; esso si comporta quindi come un generatore di corrente costante e $i_C(t) = -i_D(t) = -I_D$; in tal caso l'integrale a secondo membro diventa una retta in t e la scarica di C ha un andamento lineare:

$$v_{OUT}(t) - v_{OUT}(t_0) \approx -\frac{1}{C} \int_{t_0}^t I_D d\xi = -\frac{I_D}{C} (t - t_0) \quad (2.9)$$

Questa formula, chiaramente, ha senso fisico finché $0 < V_{out}(t) < V_{DD}$, ovvero per i t che soddisfano questo vincolo. Scegliendo un istante di tempo $t_1 > t_0$, si ha:

$$\Delta V = |v_{OUT}(t_1) - v_{OUT}(t_0)| \approx \frac{|I_D|}{C} (t_1 - t_0) \quad (2.10)$$

Nel caso del fronte di salita conduce il pMOS e si possono fare considerazioni analoghe; la 2.10 resta valida grazie agli operatori di valore assoluto.

Ai fini progettuali, in riferimento ai fronti di salita e discesa siamo interessati al tempo necessario per passare dal livello logico basso ($0V$) a quello alto ($+V_{DD}$) e viceversa. Nel caso di fronte di discesa, iniziamo l'analisi con C carico a $v_{OUT}(t_0) = V_{DD}$ e scegliamo t_1 t.c. $v_{OUT}(t_1) = 0$; facciamo il viceversa con il fronte di salita. Otteniamo così una comoda formula approssimata, valida in entrambi i casi:

$$\Delta V = \frac{\tau |I_D|}{C} \quad \Leftrightarrow \quad |I_D| = C \frac{\Delta V}{\tau} \quad (2.11)$$

La tecnica TSPC (*True single phase clock*) permette di ottimizzare le performance di circuiti CMOS, soprattutto in termini di frequenza operativa. Tale risultato è raggiunto mediante la temporizzazione del funzionamento dei circuiti CMOS e più precisamente con l'aggiunta di un segnale di clock che, nel caso del circuito in esame, identifica due sole fasi in cui può trovarsi a lavorare quest'ultimo:

- *Fase di precarica*: il segnale di clock Φ è a livello logico basso, per cui i MOS di tipo p pilotati da Φ si trovano in conduzione, mentre quelli di tipo n sono in interdizione. Facendo riferimento allo schema di fig. 2.8 ciò implica che durante questo semiperiodo del segnale di clock il segnale \overline{SUM} vada a livello alto, mentre i segnali SUM e $CARRY$, che tra l'altro sono i segnali di uscita, mantengano il valore cui si trovavano all'inizio del semiperiodo.
- *Fase di valutazione*: Φ è a livello alto, i pMOS da esso pilotati non conducono, a differenza degli nMOS. Nel circuito in esame questo implica che il valore logico dei segnali SUM e $CARRY$ dipenda da tutti i restanti segnali, ovvero A , B e C , secondo la funzione logica tipica del sommatore full adder.

Valutiamo più precisamente cosa accade durante la fase di valutazione. Per farlo consideriamo la fig. 2.9 dove sono stati evidenziati i quattro stadi che compongono il circuito.

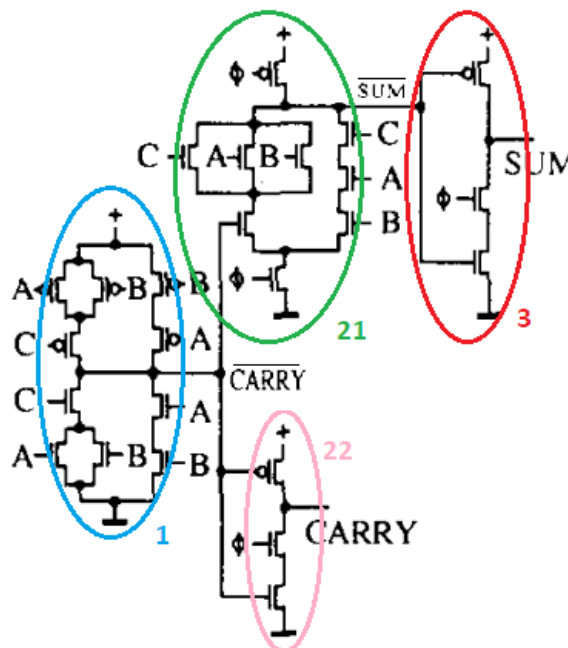


FIGURA 2.9: I quattro stadi che compongono il TSPC full adder.

Per quanto riguarda il segnale SUM lo stadio finale è il 3. La parte n di questo stadio CMOS è costituita da due nMOS, uno pilotato dal segnale di temporizzazione Φ e uno dal segnale $!SUM$ generato dallo stadio precedente. $!SUM$ pilota anche l'unico transistor pMOS presente nella parte p . Sull'uscita di questo stadio si ipotizza essere presente una capacità di carico pari a $100fF$ come da specifiche.

Il segnale $!SUM$ citato nel paragrafo precedente è a sua volta generato dallo stadio 2.1. La parte p consiste in un solo pMOS pilotato da Φ che ha il compito di fornire in uscita un 1 logico durante la fase di precarica mentre delega alla parte n la generazione di $!SUM$ durante la fase di valutazione. Quest'ultimo assumerà lo 0 logico solo se, durante la valutazione, è soddisfatta almeno una delle seguenti condizioni:

- A , B , C sono tutti alti contemporaneamente, cosicché i rispettivi nMOS possano condurre e creare così un percorso per la corrente dal nodo $!SUM$ a massa;
- $!CARRY$ è alto e almeno uno tra A , B , o C è alto, per lo stesso motivo del punto precedente.

Lo stadio 2.2 genera il segnale d'uscita $CARRY$ ed è del tutto analogo allo stadio 3.

Infine lo stadio 1 si occupa di determinare il valore del segnale $!CARRY$. La sua analisi è simile a quella dello stadio 2.1, sebbene questa volta non vi sia alcun transistor pilotato dal segnale di temporizzazione. $!CARRY$ assume livello logico basso quando si crea un percorso per la corrente tra il nodo di uscita e massa, cioè quando A e B sono alti o uno dei due è alto assieme a C , cosicché i canali dei rispettivi nMOS sono presenti e il percorso si può così creare. Discorso duale vale per la parte p : se A e B sono bassi, o uno dei due è basso assieme a C , si forma un percorso di corrente tra l'alimentazione positiva e il nodo di uscita, ovvero $!CARRY$ assume livello logico alto.

Capitolo 3

Progettazione circuitale

3.1 Schema completo

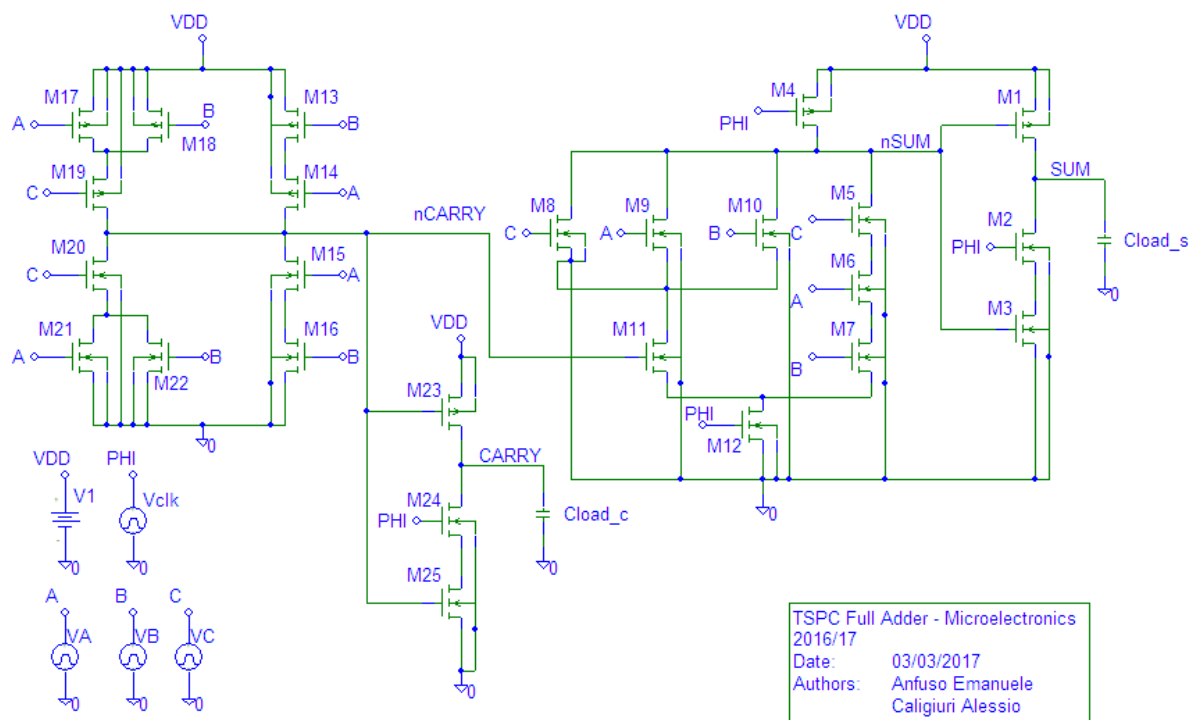


FIGURA 3.1: Schema completo del full-adder.

In fig. 3.1 è mostrato lo schema completo del full-adder, così come simulato in *SPICE*. Tutti i *bulk* degli nMOS e dei pMOS sono stati collegati rispettivamente a massa e all'alimentazione positiva, in modo tale da assicurare la polarizzazione inversa delle giunzioni *bulk - drain* e *bulk - source* in ogni condizione operativa. Questa scelta è inoltre obbligata

dalla realizzazione su silicio nella quale tutti i bulk degli nMOS (pMOS) corrispondono al medesimo substrato di tipo p ("vasca" n -well).

3.2 Dimensionamento di massima

Il dimensionamento dei transistor prevede di iniziare dagli **stadi finali** 2.2 e 3 (in riferimento alla fig. 2.9), relativi rispettivamente ai MOS $M1$, $M2$, $M3$ e $M23$, $M24$, $M25$ dello schema in fig. 3.1; entrambi hanno una capacità di carico $C_{load} = 100fF$, fornita dalle specifiche indicate in sezione 1.1.

Le capacità di gate di $M1$ e $M3$ costituiranno il carico dello stadio 2.1 precedente; dipendendo queste dalle dimensioni dei MOS, si giustifica la necessità di iniziare dai finali e tornare indietro. Si osservi che analogamente il carico dello stadio 1 è dato dal parallelo delle capacità di gate di $M23$, $M25$ (stadio 2.2) e $M11$ (stadio 2.1).

3.2.1 Vincoli temporali e *inverter equivalente*

Dalla frequenza operativa richiesta di $f_{clock} = 2GHz$ (ovvero periodo $T_{clock} = 500ps$) derivano vincoli temporali sulle fasi di valutazione e precarica, che devono andare a regime entro $T_{clock}/2 = 250ps$. Rispetto al formalismo usato in sezione 2.2, questo vuol dire che i tempi di salita e di discesa dei singoli stadi devono restare entro tale limite:

$$\tau = \tau_r = \tau_f < T_{clock}/2 = 250ps \quad (3.1)$$

A titolo precauzionale si fissa $\tau = 200ps$, tenendo conto anche dei $25ps$ dati come tempi di salita/discesa delle tensioni A , B , C , $clock$ in ingresso.

A questo punto si può applicare la formula 2.12, impiegando i valori di μ_n , μ_p e C'_{ox} trovati in sezione 2.1.1, ottenendo così i **rapporti d'aspetto dei MOS di un *inverter equivalente*** che deve pilotare la data capacità di carico completando la carica entro il tempo definito. Si ricorda nuovamente che il risultato fornito da questa formula è decisamente più piccolo del necessario, a causa di approssimazioni ottimistiche sia sul tempo di carica/scarica (assunta lineare per facilità) sia sulla caratteristica del MOS, come già discusso.

3.2.2 Dall'inverter equivalente alle dimensioni dei MOS

Ai fini progettuali, ogni stadio del full-adder è assimilabile ad un *inverter equivalente* dal momento che il segnale in uscita è *alzato* a V_{DD} da una serie/parallelo di pMOS ed *abbassato* a massa da una serie/parallelo di nMOS; per ragioni geometriche entrambe queste combinazioni di transistor possono essere ricondotte ad un singolo componente di dimensioni opportune, considerando il caso peggiore.

Consideriamo per esempio la parte superiore dello stadio 1. Il percorso di corrente più sfavorevole si ha quando conducono solo $M13$ e $M14$ (o, analogamente, solo $M17$ e $M19$). Due MOS di uguale larghezza in serie si possono valutare come un unico MOS con rapporto d'aspetto equivalente ridotto, come mostra la figura seguente.

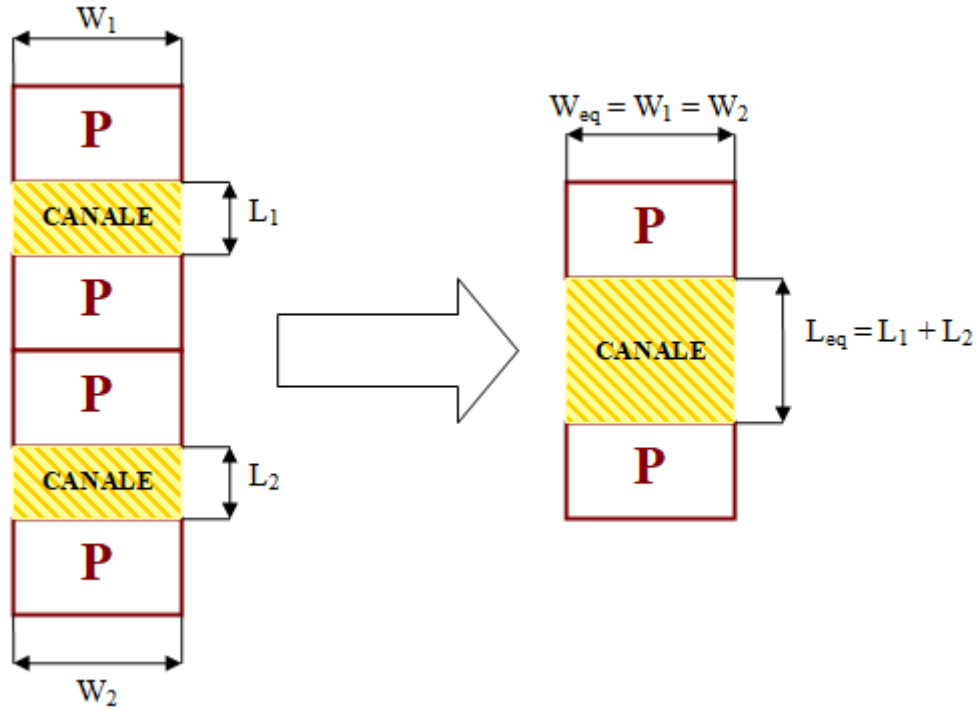


FIGURA 3.2: Considerazioni geometriche sulle dimensioni di due pMOS in serie.

Quando entrambi i transistor sono in conduzione, la corrente di drain percorre due canali, per una lunghezza totale $L_{eq} = L_1 + L_2$ e una larghezza costante $W_{eq} = W_1 = W_2 = W$. Se i due MOS hanno anche lunghezza di canale uguale ($L = L_1 = L_2$), il rapporto d'aspetto risulta dimezzato; infatti:

$$\left(\frac{W}{L}\right)_{eq} = \frac{W}{L_1 + L_2} = \frac{W}{2 \cdot L} = \frac{\left(\frac{W}{L}\right)_1}{2} = \frac{\left(\frac{W}{L}\right)_2}{2} \quad (3.2)$$

Questo vale sia per dispositivi a canale n che a canale p . In generale, si può affermare che, con un numero $n \in \mathbb{N}$ di MOS tutti uguali in serie, il rapporto d'aspetto del transistor equivalente è:

$$\left(\frac{W}{L}\right)_{eq} = \frac{\left(\frac{W}{L}\right)}{n} \quad (3.3)$$

Una considerazione duale si può fare considerando il parallelo di MOS uguali, nel qual caso il rapporto d'aspetto del singolo è moltiplicato per il numero di dispositivi. Tuttavia questa considerazione non è d'interesse nella circostanza considerata, in quanto il caso peggiore in un parallelo si ha sempre quando conduce un solo MOS alla volta.

Per brevità non si riportano i conti sui singoli stadi, rimandando l'analisi degli stessi ai relativi script *MATLAB*.

3.2.3 Dimensioni geometriche dei MOS: risultati di *MATLAB*

Di seguito si riportano le dimensioni di massima dei MOS ottenute applicando le considerazioni precedenti; tali risultati sono stati calcolati con alcuni script *MATLAB*.

MOS	Canale	W/L	$W(\mu\text{m})$	$L(\mu\text{m})$	Stadio
M1	p	10	1.20	0.12	Finale <i>SUM</i>
M2, M3	n	6	0.72	0.12	"
M4	p	2	0.24	0.12	Generazione <i>!SUM</i>
M5, M6, M7	n	2	0.24	0.12	"
M8, M9, M10	n	2	0.24	0.12	"
M11	n	2	0.24	0.12	"
M12	n	2	0.24	0.12	"
M13, M14	p	6	0.72	0.12	Generazione <i>!CARRY</i>
M15, M16	n	2	0.24	0.12	"
M17, M18, M19	p	6	0.72	0.12	"
M20, M21, M22	n	2	0.24	0.12	"
M23	p	8	0.96	0.12	Finale <i>CARRY</i>
M24, M25	n	5	0.60	0.12	"

TABELLA 3.1: Tabella delle dimensioni calcolate dei MOS, non valide.

3.2.4 Prima simulazione *pre-layout*

La prima simulazione *SPICE* (fig. 3.3) del circuito dimensionato come da tabella 3.1 ha restituito dei risultati imbarazzanti: i segnali in uscita *SUM* (in blu) e *CARRY* (in verde)

spesso non riescono a raggiungere le tensioni di alimentazione, perché le commutazioni sono troppo lente.

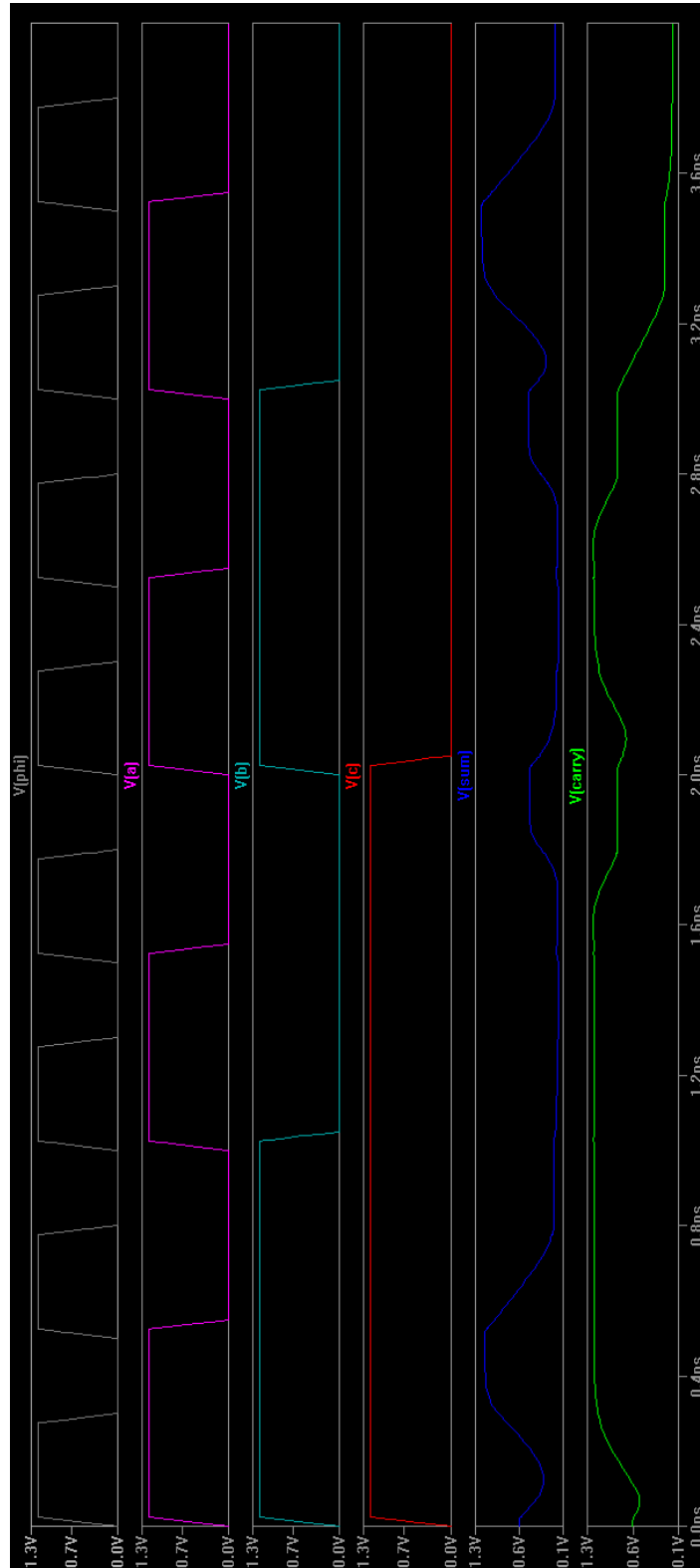


FIGURA 3.3: Prima simulazione *pre-layout* con dimensioni calcolate dei MOS.

Per comprendere le cause di questo fallimento, abbiamo provato ad ingannare il simulatore semplificando il modello del MOS, esattamente come in sezione 2.1.3. Ne è risultata una simulazione, qui omessa, decisamente più aderente ai risultati attesi che ha testimoniato la bontà del dimensionamento effettuato ma chiaramente non attendibile dal momento che non ricalca il reale funzionamento dei MOS. Ad esempio, a causa del forte campo elettrico nella zona di canale dovuto alle ridotte dimensioni dei MOS (tutti a lunghezza minima), μ_n e μ_p non sono più costanti e diminuiscono rispetto ai valori calcolati, come già discusso in sezione 2.1.2.

3.3 Fitting dei parametri: dimensioni definitive

Al fine di ottenere delle formule di progetto adeguate, sarebbe alquanto arduo utilizzare per i MOS il modello matematico *SPICE level 3*. Abbiamo inizialmente provato a ricavare $\mu \cdot C'_{ox}$ e V_{th} dalle curve caratteristiche *level 3* di fig. 2.2 per utilizzare questi valori nella formula 2.12, ottenendo dei transistor di larghezza assai elevata vista l'applicazione in questione. Pertanto abbiamo scelto di cambiare approccio e operare in modo empirico: dopo un'attenta analisi dei segnali intermedi ottenuti mediante simulazioni, abbiamo agito sulle singole dimensioni dei transistor critici e, per approssimazioni successive, ci siamo avvicinati ad una soluzione soddisfacente sia dal punto di vista temporale che geometrico.

È stato necessario aumentare le dimensioni dei singoli transistor, in particolare di quelli finali che pilotano le uscite *SUM* e *CARRY* (stadi 3 e 2.2). La tabella seguente riporta le nuove dimensioni.

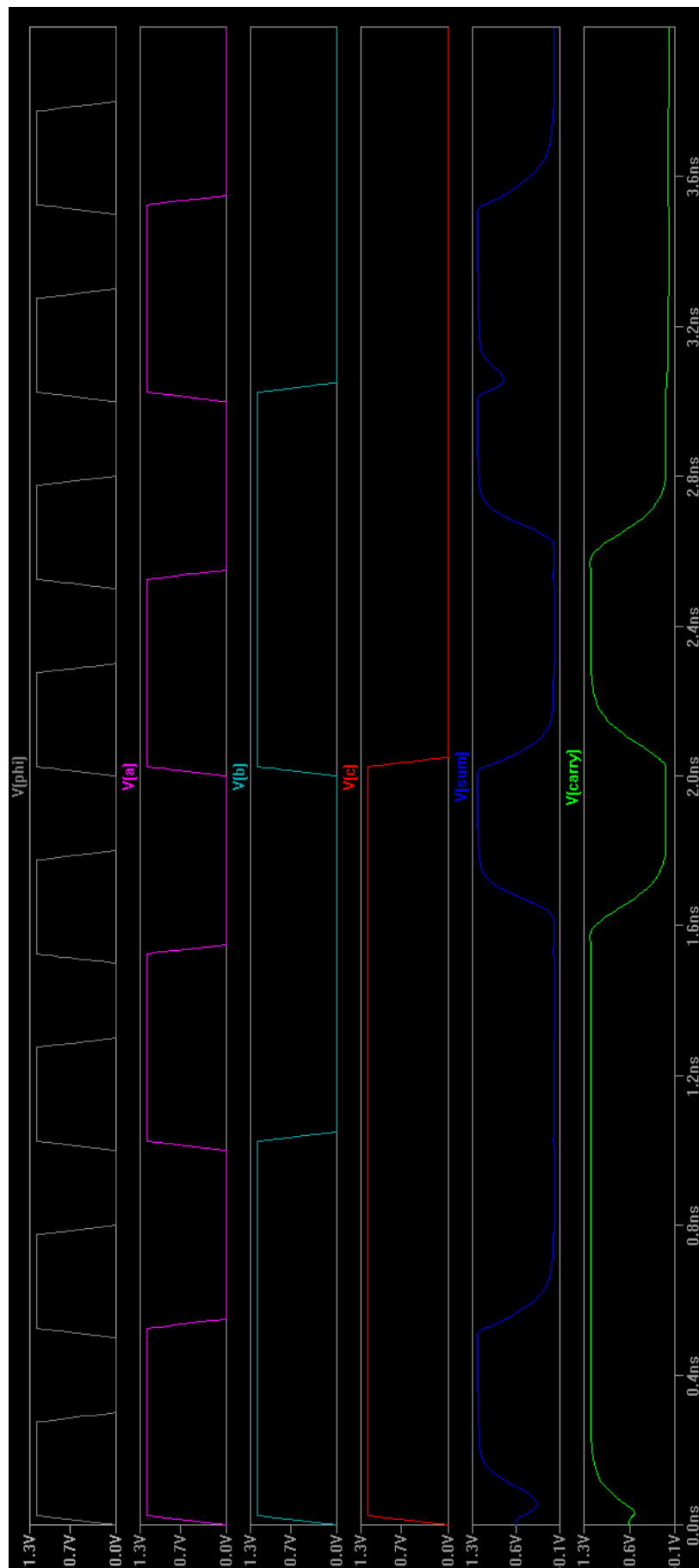
MOS	Canale	W/L	$W(\mu\text{m})$	$L(\mu\text{m})$	Stadio
M1	p	30	3.60	0.12	Finale <i>SUM</i>
M2, M3	n	16	1.92	0.12	"
M4	p	10	1.20	0.12	Generazione <i>!SUM</i>
M5, M6, M7	n	16	1.92	0.12	"
M8, M9, M10	n	10	1.20	0.12	"
M11	n	10	1.20	0.12	"
M12	n	16	1.92	0.12	"
M13, M14	p	24	2.88	0.12	Generazione <i>!CARRY</i>
M15, M16	n	8	0.96	0.12	"
M17, M18, M19	p	24	2.88	0.12	"
M20, M21, M22	n	8	0.96	0.12	"
M23	p	30	3.60	0.12	Finale <i>CARRY</i>
M24, M25	n	20	2.40	0.12	"

TABELLA 3.2: Tabella delle dimensioni dei MOS dopo il *fitting*.

3.3.1 Simulazione *pre-layout* soddisfacente

La simulazione *SPICE pre-layout* finale mostra un corretto funzionamento del circuito. È agevole verificare che a fronte degli ingressi *A*, *B*, *C*, le uscite *SUM* e *CARRY* forniscono i corretti valori logici e tensioni accettabili.

Sarà necessario eseguire un'analoga simulazione *post-layout*, ovvero dopo il disegno su silicio del circuito.

FIGURA 3.4: Simulazione *pre-layout* soddisfacente.

Capitolo 4

Layout

In Fig. 4.1 si può ammirare il layout finale del Full Adder TSPC.

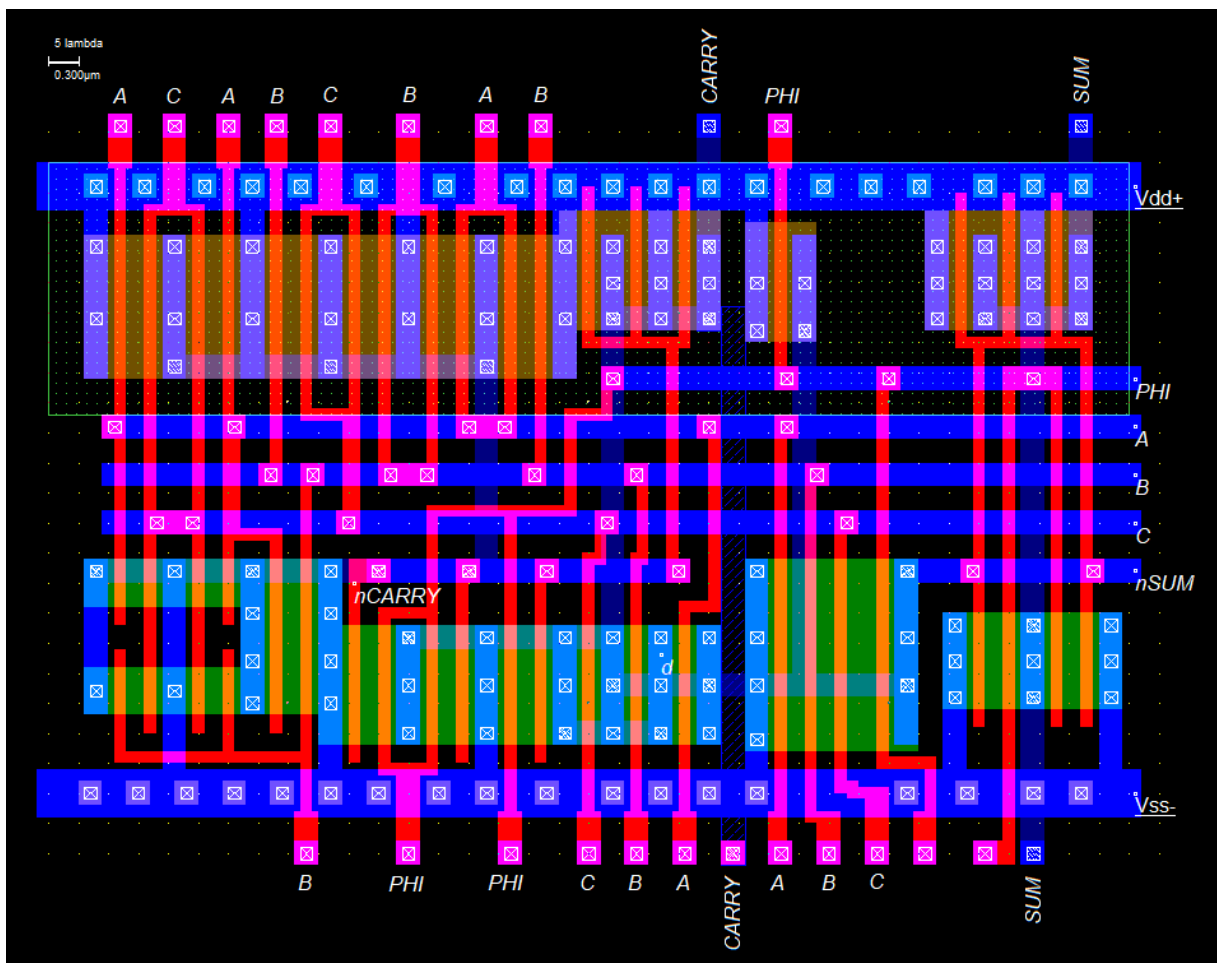


FIGURA 4.1: Layout finale.

Vediamo quali sono stati i passi che hanno portato alla sua realizzazione.

4.1 Disegno dei singoli stadi

Un possibile metodo di lavoro prevede di ottenere il circuito finale in maniera progressiva, realizzando e validando un singolo stadio alla volta, mediante simulazioni *post-layout*. Il circuito finale è ottenuto dall'unione dei vari pezzi ed è a sua volta validato da una simulazione dello stesso tipo che, se soddisfacente, sancisce il raggiungimento degli obiettivi prefissati.

Analizziamo quindi i disegni delle singole parti di circuito riportando alcuni screen tratti da *Microwind* e i cui colori hanno i seguenti significati:

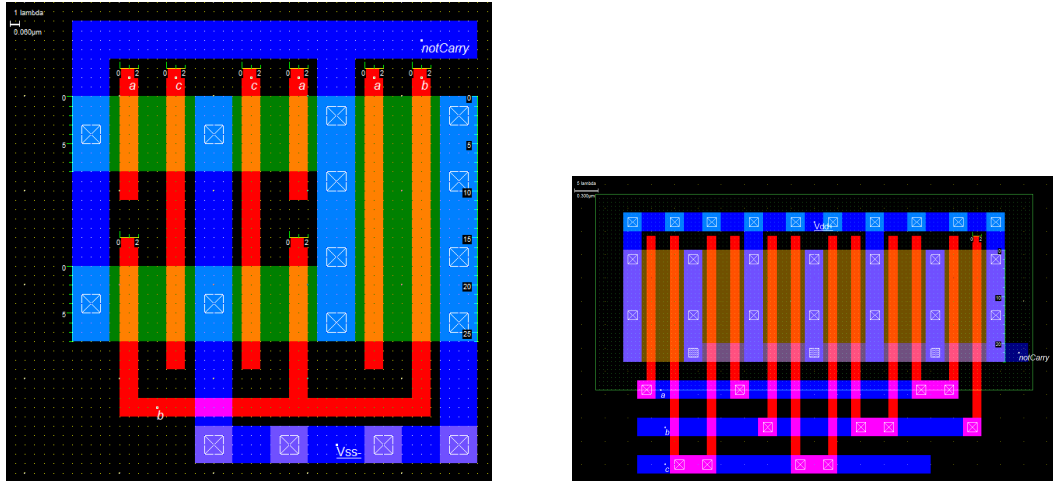
- lo sfondo nero rappresenta il substrato di tipo *p*;
- le aree verdi sono diffusioni di tipo *n*;
- le aree verdi a puntini sono diffusioni di tipo *n-well*;
- le aree rosse costituiscono piste di polisilicio;
- le aree blu sono metalizzazioni di tipo *metal 1*;
- i quadrati con la croce all'interno indicano la presenza di un contatto tra livelli differenti.

4.1.1 Stadio 1 - Generazione di !CARRY

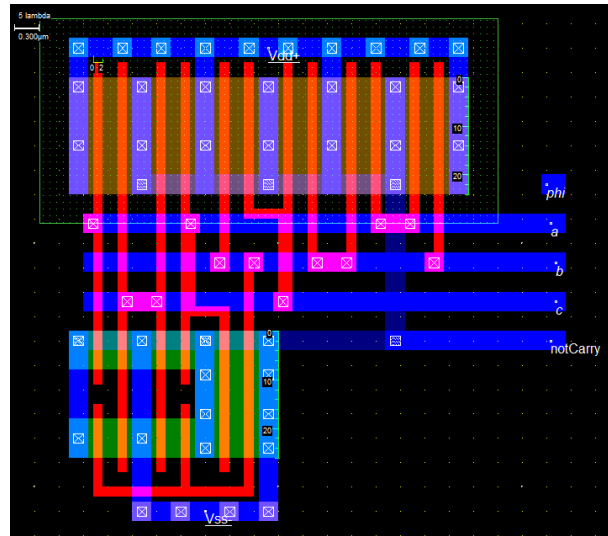
In fig. 4.2 si può osservare la realizzazione della parte *n* (a sinistra) e della parte *p* (a destra) dello stadio 1 dedicato alla generazione del segnale *!CARRY*, indicato nel layout con la label *notCarry*. I MOS più larghi sono disegnati "ripiegati" su se stessi in modo da mantenere le dimensioni di canale desiderate pur potendo riorganizzare la disposizione delle diffusioni per meglio adattarsi alla forma generale che si vuole far assumere al circuito.

Per ciascuno dei due circuiti viene estratta la netlist che, tramite *LTspice*, permette di validare il disegno tenendo conto di tutti i parametri presenti nel modello della tecnologia utilizzata, incluse le capacità parassite.

Le due parti collegate costituiscono il disegno complessivo dello stadio 1, riportato in fig. 4.3. Da tale figura si può notare la scelta di adottare un approccio *standard cell* per la disposizione delle piste di alimentazione e di segnale, come d'altronde viene suggerito nelle specifiche di progetto. In alto e in basso sono quindi presenti due piste orizzontali di *Metal 1* collegate rispettivamente all'alimentazione positiva e negativa. I segnali *A*, *B*, *C*

FIGURA 4.2: Parte n (a sinistra) e parte p (a destra) dello stadio 1.

e *notCarry* sono forniti e prelevati in verticale, tramite piste di polisilicio per i primi tre e una pista di *Metal 2* per l'ultimo. Tutti e quattro, assieme a *phi*, sono poi trasportati nel resto del circuito da alcune piste di *Metal 1*, orizzontali, poste tra le parti n e le parti p di ogni stadio.

FIGURA 4.3: Disegno dello stadio 1 dedicato alla generazione di *notCarry*.

Anche di questo circuito si effettua una simulazione *post-layout* prima di procedere allo stadio successivo.

4.1.2 Stadio 2.1 - Generazione di !SUM

In figura 4.4 è riportato il disegno della parte n dello stadio 2.1 dedicato alla generazione del segnale *!SUM*, etichettato come *notSum*. Una particolarità di questa parte riguarda

la realizzazione dei MOS in serie, ottenuta tramite l'affiancamento di diverse piste di polisilicio pilotate dai segnali d'ingresso; questa tecnica permette di risparmiare spazio evitando di inserire metalizzazioni per i gate dei MOS intermedi tra il nodo di uscita *notSum* e il nodo comune ai due rami che compongono questa parte di circuito: non è infatti necessario prelevare tali segnali.

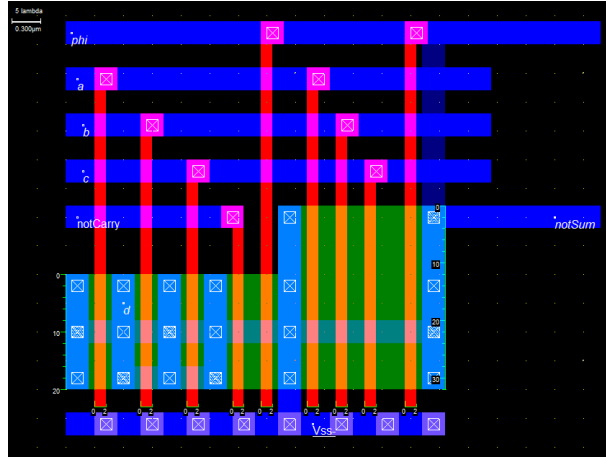


FIGURA 4.4: Parte n dello stadio 2.1 dedicato alla generazione di *notSum*.

La parte p di questo stadio è costituita da un semplice pMOS per cui se ne omette la rappresentazione completa. Considerazioni circa la tipologia di progetto *standard cell* e la procedura di validazione sono analoghe a quelle esposte per lo stadio 1.

4.1.3 Stadi finali 2.2 e 3 - Generazione di CARRY e SUM

Gli stadi finali 2.2 e 3 hanno la stessa identica struttura, per cui per brevità si riporta in fig. 4.5 il disegno del solo stadio 3, dedicato alla generazione del segnale d'uscita *SUM*.

Le tecniche utilizzate per il layout di questi stadi sono le stesse già presentate nei paragrafi precedenti. Terminata la sua validazione, si può procedere a comporre il circuito finale.

4.2 Full design

Il circuito finale, rappresentato in fig. 4.1, è il risultato della composizione dei singoli stadi presentati finora. Essi sono affiancati in modo da ottenere una struttura tanto più compatta possibile pur mantenendo la topologia delle varie parti. Ciò significa avvicinare le varie piste e/o diffusioni in modo da minimizzare gli spazi vuoti pur rispettando le regole di progettazione ovvero le distanze minime necessarie.

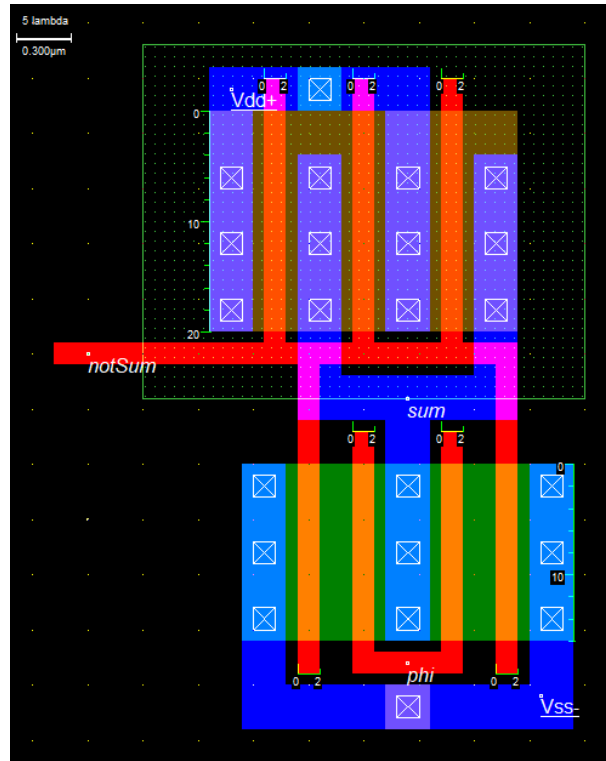


FIGURA 4.5: Disegno dello stadio 3 dedicato alla generazione di *SUM*.

Il risultato è un circuito molto ordinato e di dimensioni $11.04\mu m \times 6.54\mu m$ (ovvero, in *lambda*, 184 x 109), con i segnali di ingresso/uscita prelevati e forniti in verticale e quelli di alimentazione applicati in orizzontale.

L'ultimo passo consiste nel compiere un'ultima validazione del circuito finale. Se ne esporta quindi la netlist che viene simulata tramite *LTspice*. Il risultato della simulazione si osserva in fig. 4.6. Come si può notare a fronte di ogni combinazione degli ingressi *A*, *B*, *C*, i segnali *SUM* e *CARRY* assumono il livello di tensione corretto entro le tempistiche richieste.

Da questa simulazione è anche possibile calcolare la potenza media dissipata dal circuito in esame, che risulta pari a $315\mu W$.

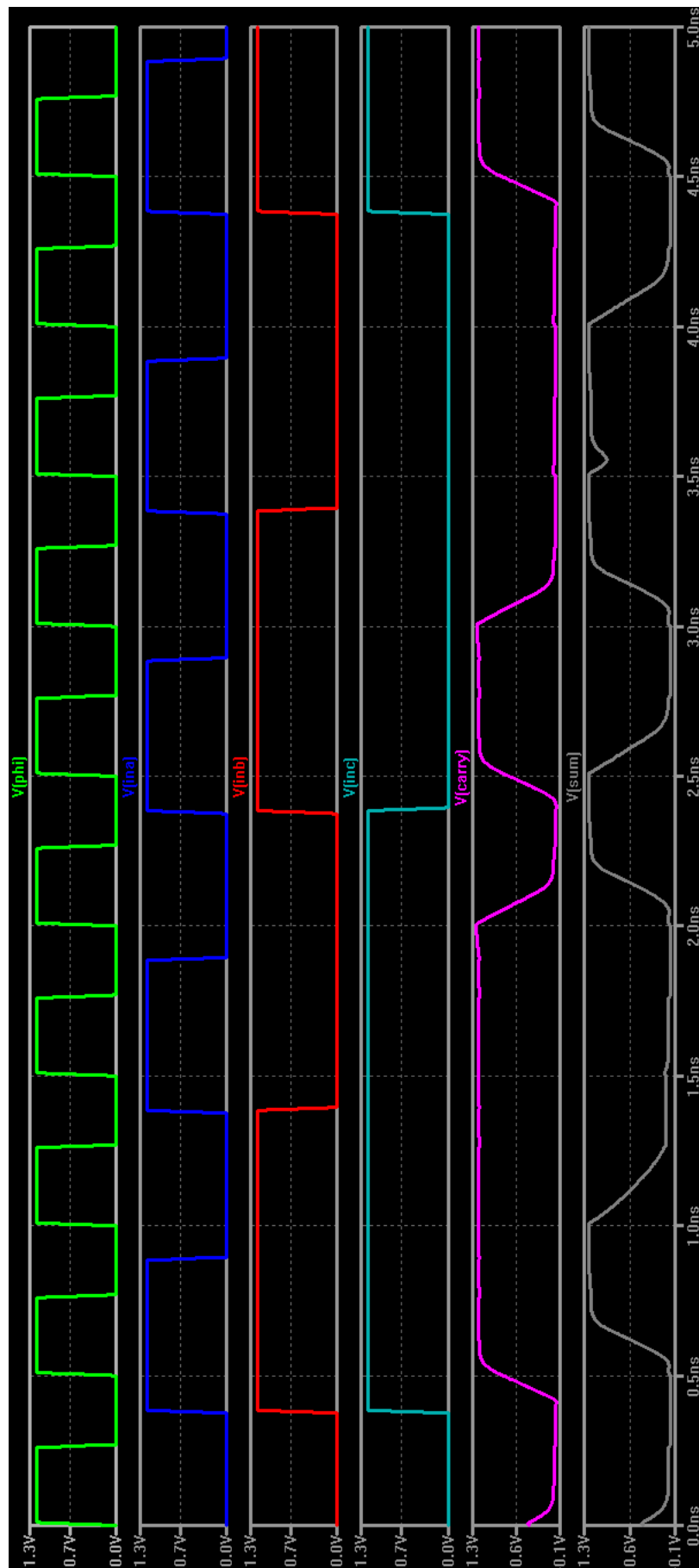


FIGURA 4.6: Simulazione post-layout dello stadio finale.

Capitolo 5

Conclusioni finali

In questo progetto abbiamo realizzato un circuito Full Adder TSPC partendo dallo schema circuitale per ottenere come risultato finale il disegno delle maschere necessarie alla sua produzione.

La simulazione finale *post-layout* ha confermato che le specifiche richieste sono state soddisfatte. La cura e l'attenzione riposte nella realizzazione del layout hanno permesso di ottimizzare l'area occupata e la potenza dissipata.

Da un punto di vista formativo questo lavoro ci ha dato l'opportunità di imparare e adottare una metodologia di progetto chiara e completa. Abbiamo inoltre preso dimestichezza con alcuni software di progettazione elettronica e ci siamo scontrati con alcune problematiche tipiche della modellizzazione di componenti fisici.

Bibliografia

- [1] Jiren Yuan and Christer Svensson. High-speed cmos circuit technique. *IEEE Journal of Solid-State Circuits*, 24(1):62–70, 1989.