



**POLITECNICO  
DI TORINO**

*Laurea Magistrale in Ingegneria Elettronica*

*Anno Accademico 2018/2019*

Sistemi Elettronici a basso consumo

*prof. Maurizio Zamboni, prof. Mariagrazia Graziano*



## Relazione Laboratori

Raffaele Tuzzo 263722

Giulio Pecoraro 266391

Luigi Massari 265396

# Contents

<b>1</b>	<b>Laboratorio 1:</b>	
	<b>Power Estimation: probabilistic techniques</b>	<b>1</b>
1.1	Probability and Activity Calculation: Simple Logic Gates . .	1
1.2	Probability and Activity Calculation: Half and full adder . .	3
1.3	RCA synthesis and power analysis . . . . .	5
<b>2</b>	<b>Laboratorio 2:</b>	
	<b>FSM State Assignment and VHDL Synthesis</b>	<b>8</b>
2.1	FSM State Assignment . . . . .	8
2.2	VHDL synthesis . . . . .	9

# 1. Laboratorio 1:

## Power Estimation: probabilistic techniques

### 1.1 Probability and Activity Calculation: Simple Logic Gates

Durante la prima parte dell'esercitazione è stata calcolata la probabilità di avere '1' logico in uscita di alcuni gate elementari, con la relativa Switching Activity.

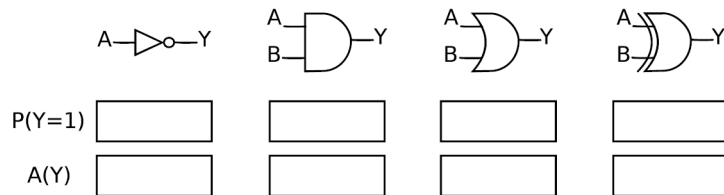


Figure 1.1: *Probabilità e Switching Activity stimati manualmente*

La probabilità di '1' logico è stata stimata semplicemente andando a valutare il rapporto fra il numero di possibili combinazioni con '1' logico diviso il numero di combinazioni totali. Invece per il calcolo della Switching Activity è stata utilizzata la formula vista a lezione:

$$A = P_1P_0 + P_1P_0 = 2P_1(1 - P_1)$$

Dove  $P_1$  e  $P_0$  sono le probabilità di avere '1' e '0' logici in uscita dalla mia

## 1.1. Probability and Activity Calculation: Simple Logic Gates

---

porta.

In seguito, tramite il programma *ModelSim* è stato analizzato il numero di toogle delle varie porte utilizzando un testbench sviluppato appositamente dai docenti. Si è andato a variare il numero di colpi di clock, come richiesto dalla traccia ed in seguito si sono comparati i valori ottenuti dalla simulazione con ciò che si era calcolato manualmente.

Tramite appositi comandi di Modelsim (**-power report**), sono stati stilati dei report relativi ad una stima delle commutazioni delle varie porte, della quale se ne riporta un esempio in Figura 1.2. Questi report consentono di stimare l'attività delle porte come verrà descritto in seguito.

Power Report			Node	Tc	Ti	Time At
1	Time At 0	Time At X				
-----						
			/tbprob/clk	20000	0	5000000
ps	5000000 ps	0 ps	/tbprob/reset	1	0	1000
ps	9999000 ps	0 ps	/tbprob/ld	1	0	2000
ps	9998000 ps	0 ps	/tbprob/dout (15)	4924	0	4929000
ps	5071000 ps	0 ps	/tbprob/dout (14)	4923	0	4928500
ps	5071500 ps	0 ps	/tbprob/dout (13)	4922	0	4928000
ps	5072000 ps	0 ps	/tbprob/dout (12)	4922	0	4928000
ps	5072000 ps	0 ps				

Figure 1.2: *Probabilità e Switching Activity stimati manualmente*

Si riportano nella tabella 1.1, i risultati ottenuti dalle varie simulazioni.

Tc(CK)	Tc(INV)	Tc(AND)	Tc(OR)	Tc(XOR)
20	1	?	4	4
200	43	40	42	44
2000	533	418	352	470
20000	4916	3606	3784	4876

Table 1.1: *Risultati simulazione*

Dai seguenti valori è facile ricavare i valori di Switching Activity simulale, in quanto si possono stimare da:

$$A = \frac{Tc(PORT)}{T_{CLK}}$$

Come ci si aspettava, essendo la Switching Activity il numero di toogle avvenuti in un periodo, i valori delle simulazioni vengono molto simili ai

## 1.2. Probability and Activity Calculation: Half and full adder

---

valori calcolati analiticamente. Aumentando il tempo di simulazione, i valori di Switching Activity diventano sempre più precisi, arrivando ad avere un errore tra 0,01-0,5.

### 1.2 Probability and Activity Calculation: Half and full adder

Per prima cosa sono state calcolate le probabilità di avere un '1' logico sull'uscita sia dell'half adder e sia del full adder e le probabilità di avere un '1' logico come carry out degli stessi blocchi. Una volta eseguiti questi calcoli sono state calcolate anche le corrispettive switching activity.

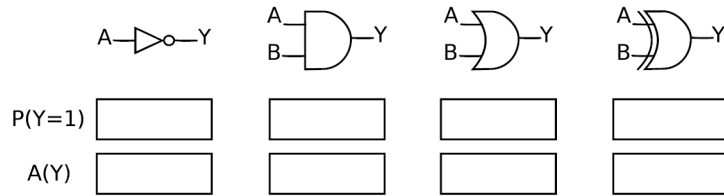


Figure 1.3: *Probabilità e Switching Activity stimati manualmente*

La probabilità di '1' logico è stata stimata semplicemente andando a valutare il rapporto fra il numero di possibili combinazioni con '1' logico diviso il numero di combinazioni totali. Invece per il calcolo della Switching Activity è stata utilizzata la formula vista a lezione:

$$A = P_1P_0 + P_1P_0 = 2P_1(1 - P_1)$$

α Dove  $P_1$  e  $P_0$  sono le probabilità di avere '1' e '0' logici in uscita dalla mia porta.

In seguito, si sono calcolate sempre manualmente le probabilità di uscita con le rispettive switching activity del Ripple carry adder, valutando la probabilità per ogni singolo Half adder come riportato in figura. Per questo calcolo iniziale gli ingressi sono stati considerati scorrelati ed equiprobabili.

## 1.2. Probability and Activity Calculation: Half and full adder

---

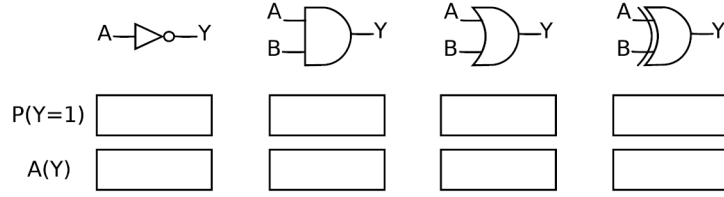


Figure 1.4: *Probabilità e Switching Activity stimati manualmente con ingressi equiprobabili*

Nel secondo caso, invece, gli ingressi sono stati considerati sempre scorrelati ma con probabilità diverse, infatti si ha che:

$$P(A = 1) = 0.4 \text{ e } P(B = 1) = 0.6$$

I risultati ottenuti risultano uguali ai precedenti, per quanto riguarda la probabilità dell'uscita e del carry out. In seguito, tramite il programma *ModelSim* è stato simulato il Ripple carry adder (giusto?) utilizzando un testbench sviluppato appositamente dai docenti. Il testbench è stato costruito appositamente per assegnare ritardi diversi al bit di somma, DRCAS, e al bit di carry, DRCAC. Inoltre per garantire una simulazione generale si è utilizzato l'LFSR per generare ingressi randomici. Per una corretta visualizzazione dei risultati si è impostata una risoluzione di 1ps. Dopo aver visualizzato il power report, come fatto già in precedenza, si sono comparati i valori ottenuti dalla simulazione con ciò che si era calcolato manualmente, seguendo lo stesso ragionamento del punto precedente.

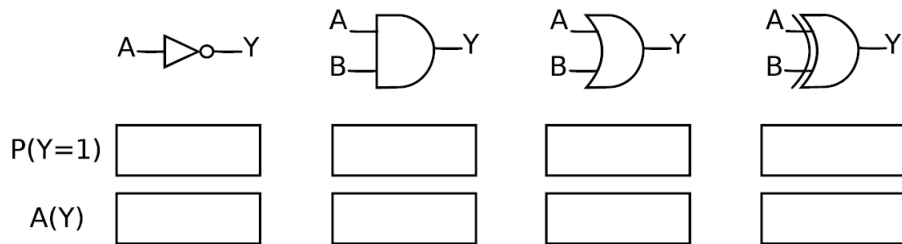


Figure 1.5: *Power report*

### 1.3. RCA synthesis and power analysis

---

(cosa noto?) e bisogna mettere le waveforms?

j- ATTENZIONE Si è poi simulato il caso in cui i due ritardi riguardanti il bit di somma e il bit di carry fossero uguali, e anche per questo è stato visualizzato il power report.

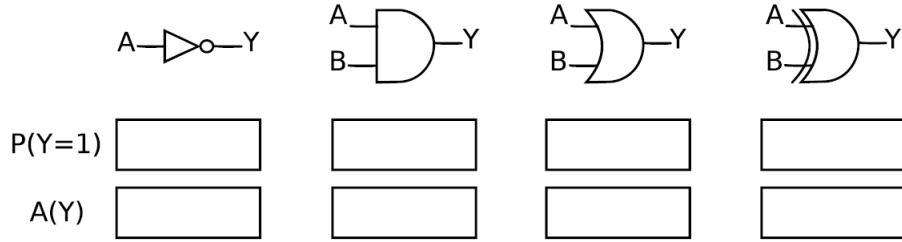


Figure 1.6: *Power report con ritardi uguali*

Da un analisi e confronto tra i risultati ottenuti manualmente e quelli ottenuti con le simulazioni, si nota come questi coincidano dato che avendo uguali ritardi, non riesco a simulare la presenza di eventuali glitch.

In seguito si è calcolata la switching activity totale dei due sommatore, utilizzando la seguente formula:

$$A = \sum_{i=1}^{N-1} A(S_i)$$

What is the overhead computation of the second adder? <- cioè?

Come ultima cosa è stato simulato il secondo testbench che ci è stato fornito, dove si simulava sempre un Ripple carry adder, ma questa volta in maniera puramente combinatoria. Analizzando i risultati, si può concludere come non avendo un segnale di temporizzazione, lavoro alla massima velocità ma ho la presenza di glitch.

### 1.3 RCA synthesis and power analysis

Nella seguente sezione dell'esercitazione è stata analizzata la potenza del sommatore RCA già analizzato in precedenza, tramite il software *Synopsys*. Dopo aver analizzato ed elaborato i file che descrivono la struttura dell'RCA, il tutto è stato sintetizzato e sono stati raccolti i vari report relativi alla

### 1.3. RCA synthesis and power analysis

---

potenza. Un primo report di potenza, riportato in Figura 1.7, descrive i contributi di potenza relativi alle 8 istanze dei Full-Adder che compongono il sommatore RCA.

Come ci si aspettava, i contributi dei vari Full Adder sono tutti simili tra di

Design	Wire Load Model	Library
RCA	SK_hvratio_1_1	NangateOpenCellLibrary
Global Operating Voltage = 1.1		
Power-specific unit information :		
Voltage Units = 1V		
Capacitance Units = 1.000000ff		
Time Units = 1ns		
Dynamic Power Units = 1uW (derived from V,C,T units)		
Leakage Power Units = 1nW		

Figure 1.7: *Power report*

loro, ad eccezione dell'istanza *FAI\_8*: il motivo consiste nel fatto che il Carry Out dell'ultimo Full Adder non è connesso a nessun'altra porta, dunque il carico da pilotare è decisamente minore.

Diventa ora interessante andare ad analizzare la singola istanza, per andare a valutare l'origine dei singoli contributi di potenza. Tramite il comando *current\_instance FAI\_1* vado ad analizzare l'istanza relativa al primo Full Adder. Viene riportato il report in Figura 1.8.

Come ci si aspettava i valori di potenza risultano assolutamente identici al report trovato in precedenza e riportato in Figura 1.7. Il passo finale è comprendere come avvenga la stima della potenza dinamica (*switching power*) dei singoli nodi del FA



### 1.3. RCA synthesis and power analysis

---

Cell		Cell	Driven Net	Tot Dynamic
Leakage		Internal	Switching	Power
Cell	Attrs	Power	Power	(% Cell/Tot)
Power				
-----				
U1		0.8205	0.0512	0.872 (94%)
36.0111				
U4		0.5889	0.5488	1.138 (52%)
36.1637				
U3		0.3374	0.1749	0.512 (66%)
32.5747				
U2		0.1377	0.3964	0.534 (26%)
14.2499				
-----				
Totals (4 cells)		1.884uW	1.171uW	3.056uW (62%)
118.999nW				

Figure 1.8: *Power report*

## 2. Laboratorio 2: FSM State Assignment and VHDL Synthesis

### 2.1 FSM State Assignment

Durante la prima parte dell'esercitazione di laboratorio, viene richiesto di implementare un circuito per sommare 6 numeri

$$s = a + b + c + d + e + f$$

utilizzando un unico sommatore, due multiplexer e un registro. Viene richiesto di valutare e minimizzare il consumo di potenza, andando a modificare la connessione degli input A-H, considerando esclusivamente l'attività della FSM e i bit di selezione del MUX S0-S3.

Il circuito completo è riportato in Figura 2.4, mentre la FSM è presente in Figura ??.

Dopo varie ottimizzazioni, si è arrivati ad avere un'attività totale pari a 8 per il multiplexer e 6 per la State transition della macchina a stati, andando a considerare che la macchina a stati e il multiplexer ricomincino le operazioni una volta terminate. Nella tabella 2.1 viene riportata la configurazione degli stati e dei bit del multiplexer scelta:

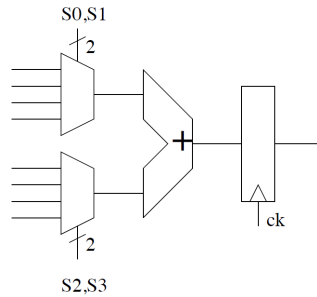


Figure 2.1: *Probabilità e Switching Activity stimati manualmente*

STATI	$S_3S_2S_1S_0$
000	0000
001	0101
011	0111
010	1110
110	1010

Table 2.1: *Risultati simulazione*

## 2.2 VHDL synthesis

Il secondo punto del laboratorio prevede di sintetizzare l'FSM tramite synopsys e studiarne le caratteristiche in termini di area, potenza e timing in modo da ricercare possibili ottimizzazioni. Si è utilizzata la libreria a 45 nm, definito un segnale di clock di periodo corrispondente a 10 ns, si è verificato il corretto inserimento tramite il comando *report\_clock* e si è sintetizzato il circuito.

clock	period	waveform
CLK	10.00 ns	{0 5} V

Di seguito è riportato lo schema generato da synopsys:

## 2.2. VHDL synthesis

---

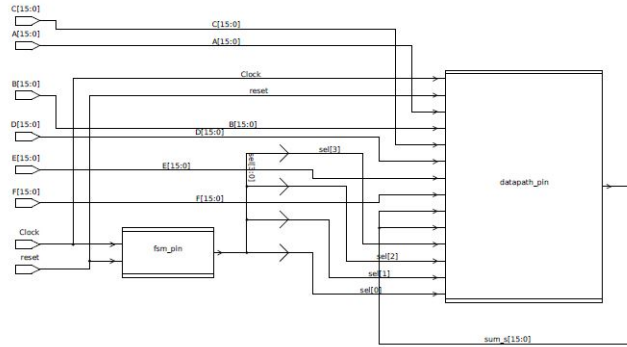


Figure 2.2: *Schematico del circuito sintetizzato*

Dal report sull'area si sono ottenute informazioni riguardanti la quantità di componenti, le connessioni, l'area relativa occupata dalla logica combinatoria, circa il doppio rispetto a quella non combinatoria e quindi dell'area totale.

type	number
ports	114
nets	118
cells	2
references	2

area type	value
combinational	195.244003
noncombinational	101.08003
total cell	296.324006

Successivamente, dopo aver verificato la corretta codifica degli stati della FSM si è analizzato il timing del circuito, dal quale si sono ottenute importanti informazioni riguardo ai ritardi delle varie porte e allo Slack time nel caso del percorso peggiore che è di 8.03 ns, parametro che consente di ottimizzare frequenza di funzionamento del circuito, visto che la condizione necessaria è che lo slack time sia positivo. Inoltre si è eseguito il timing per i peggiori 10 percorsi e si sono ricavati i valori di slack.

## 2.2. VHDL synthesis

---

Slack (MET)	value [ns]
1	8.04
2	8.04
3	8.04
4	8.04
5	8.04
6	8.05
7	8.05
8	8.05
9	8.05
10	8.05

Non si evidenziano rilevanti differenze tra i diversi slack, ciò è dovuto alla simmetria dei percorsi critici che presentano la stessa struttura. Ecco come sono distribuiti i peggiori slack:

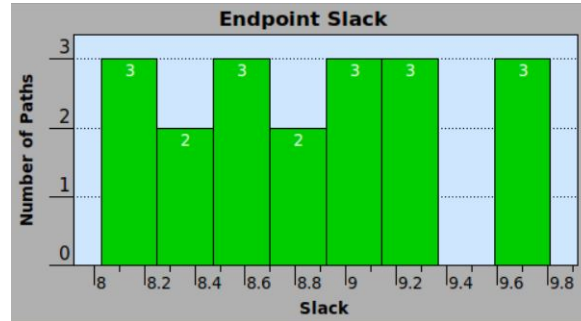


Figure 2.3: *Grafico distribuzione slacks*

Dunque si è analizzata la potenza dissipata sia dall'intera logica che da ogni singola cella. Il report sulla potenza distingue la potenza dissipata dinamicamente, staticamente e quella dovuta alle correnti di leakage, fornendo la percentuale rispetto alla potenza dissipata totale e analizzando il circuito a livello gerarchico. In questo modo è possibile capire quanto influiscono i diversi contributi di potenza dissipata e se necessario intervenire opportunamente per consumare meno.

## 2.2. VHDL synthesis

---

hierarchy	switch	int	leak	tot	%
m_adder	11.943	28.443	5.46e+3	45.844	100
datapath_adder	10.930	25.532	5.03e+3	41.495	90.5
add_78	1.765	4.921	1.19e+3	7.879	17.2
fsm	1.012	2.911	425.171	4.349	9.5

Inoltre si è analizzata l'attività delle singole celle in modo da studiare i consumi di ogni singola cella. Da quest'ultimo report si è ricavato che le celle corrispondenti ai registri hanno dei consumi più elevati di potenza statica e consumano più del doppio della corrente di leakage rispetto alle altre celle del circuito.

cell	cell internal	driven net switching	tot dynamic [% cell/tot]	cell leakage
REG[0]	1.0163	0.0584	1.075 (95%)	87.1072
REG[1]	0.8660	0.1134	0.979 (88%)	81.4649
REG[2]	0.7468	0.1083	0.855 (87%)	84.7325
U8	0.0465	0.0297	7.62e-2 (61%)	33.6813
U9	0.0386	0.1368	0.175 (22%)	31.6341
U6	0.0375	0.1356	0.173 (22%)	18.0848
U5	0.0314	0.0193	5.07e-2 (62%)	19.3118
U4	0.0304	0.0174	4.77e-2 (64%)	17.9767
U10	0.0285	0.0746	0.103 (28%)	12.9020
U7	0.0233	0.1282	0.151 (15%)	15.8344
U3	0.0190	0.1236	0.143 (13%)	17.0242
U11	9.993e-3	0.0392	4.92e-2 (20%)	14.3532
tot (12 cells)	2.894 uW	984.349 nW	3.878 uW (75%)	434.107 nW

E' utile inoltre valutare il numero di commutazioni delle singole uscite valutando sia la capacità del carico che il tasso di commutazioni. In accordo con i risultati precedenti si denota un'attività più intensa per i registri per hanno un carico capacitivo molto più alto delle altre uscite anche se il toggle rate è praticamente equivalente per tutte le uscite e la probabilità statistica è comunque minore.

hierarchy	switch	int	leak	tot	%
m_adder	11.943	28.443	5.46e+3	45.844	100
datapath_adder	10.930	25.532	5.03e+3	41.495	90.5
add_78	1.765	4.921	1.19e+3	7.879	17.2
fsm	1.012	2.911	425.171	4.349	9.5

## 2.2. VHDL synthesis

---

Inoltre si è analizzata l'attività delle singole celle in modo da studiare i consumi di ogni singola cella. Da quest'ultimo report si è ricavato che le celle corrispondenti ai registri hanno dei consumi più elevati di potenza statica e consumano più del doppio della corrente di leakage rispetto alle altre celle del circuito.

net	total net load	static prob.	toggle rate	switching power
S[2]	11.104	0.326	0.0244	0.1641
S[0]	9.304	0.228	0.0244	0.1375
S[1]	10.166	0.295	0.0221	0.1360
S[3]	9.541	0.186	0.0200	0.1153
n21	10.518	0.088	0.0181	0.1153
n5	6.169	0.814	0.0200	0.0746
n6	3.949	0.772	0.0244	0.0584
n8	4.078	0.706	0.0222	0.0546
n25	3.843	0.500	0.0221	0.0514
n28	6.482	0.772	0.0100	0.0392
n27	1.980	0.706	0.0244	0.0293
N8	1.438	0.098	0.0221	0.0193
n7	1.438	0.0392	0.0200	0.0174
tot (13 nets)				1.0125 uW

Si è adesso focalizzata l'attenzione sui consumi della Macchina a Stati. La FSM è caratterizzata da una leakage current di 418 nW a fronte dei 434 nW totali e anche per gli altri contributi di consumo di potenza dinamica i dati tendono ad evidenziare il ruolo preponderante del componente sul totale consumo del circuito.

## 2.2. VHDL synthesis

Cell	Cell Internal Power	Driven Net Switching Power	Tot Dynamic Power (% Cell/Tot)	Cell Leakage Power	Attrs
CURRENTSTATE_reg[0]	0.8986	0.0979	0.997 (90%)	89.1728	
CURRENTSTATE_reg[1]	0.5562	0.0461	0.602 (92%)	81.9647	
CURRENTSTATE_reg[2]	0.3711	2.719e-03	0.374 (99%)	79.5076	
U11	0.0999	0.3922	0.492 (20%)	14.3532	
U6	0.0989	0.0453	0.144 (69%)	30.0052	
U5	0.0644	0.0595	0.124 (52%)	17.0430	
U7	0.0384	0.0491	8.75e-02 (44%)	11.6295	
U4	0.0193	1.813e-03	2.12e-02 (91%)	34.6953	
U10	0.0109	8.310e-03	1.92e-02 (57%)	20.1295	
U9	2.905e-03	1.593e-03	4.50e-03 (65%)	16.3885	
U8	2.689e-03	6.834e-03	9.52e-03 (28%)	5.3388	
U3	1.734e-03	3.431e-04	2.08e-03 (83%)	18.5247	
Totals (12 cells)	2.165uW	711.655nW	2.877uW (75%)	418.753nW	

Figure 2.4: *Potenza dissipata dalla FSM*

Dopodiché si è provato a variare la frequenza di lavoro del circuito, provando a sintetizzare il circuito in modo da lavorare alla massima frequenza di funzionamento consentita dal percorso critico. Dall'analisi sul timing si è trovato lo slack peggiore di circa 8.02 ns lavorando a 10 ns di periodo di clock. Si può allora decrementare il periodo di clock fino a  $10 - 8.02 = 1.98$  ns. Difatti si è scelto un periodo di clock di 2 ns.

cell internal power	141.4894 uW (71%)
net switching power	58.8064 uW (29%)
total dynamic power	200.2958 uW (100%)
cell leakage power	5.5273 uW

Si nota come sia aumentata la Total Dynamic Power, questo poiché aumentando la frequenza operativa aumentano anche il numero di commutazioni interne e quindi viene dissipata maggiore potenza dinamica. Resta invece invariata la corrente di leakage che infatti dipende solo dalla tecnologia usata.

Infine è stato posto al sintetizzatore un ulteriore vincolo sulla massima potenza dinamica dissipabile a 200 uW, considerando che nell'ultimo report la potenza totale dissipata ammonta a 200.2958 uW.



## 2.2. VHDL synthesis

---

cell internal power	140.3547 uW (70%)
net switching power	59.0397 uW (30%)
total dynamic power	199.3944 uW (100%)
cell leakage power	5.5832 uW

Adesso la potenza dinamica totale dissipata è di 199.3944 uW e rispetta il vincolo. Inoltre, si nota che stavolta il parametro della leakage power è leggermente variato, questo poichè stavolta per rispettare il vincolo sul consumo di potenza è stata variata la topologia del circuito usando differenti porte logiche.