

第五章 電路模擬

5.1 介紹

製造晶片是既昂貴又花時間的工作，所以設計人員需要模擬工具以便在進行製造之前來研究設計空間與確認設計。模擬器可在從製程到架構之間不同層級的抽象化上操作。SUPREME 之類的程序模擬器(Process Simulator)可以預測製程配方中，如時間與溫度這樣的製程因數對元件的物理與電子特性所產生的影響。SPICE 之類的電路模擬器(Circuit Simulator)採用元件模型與線路網清單(netlist)來預測電路的電壓與電流，而這可以顯示性能與電力消耗。邏輯模擬器(Logic Simulator)可以預測數位電路的功能，而且廣泛使用於驗證以硬體敘述語言(Hardware Description Language, HDL)所述之設計的邏輯作業正確性。架構模擬器(Architecture Simulator)在指令層級與暫存器層級上運作以預測吞吐量與記憶存取模式，如此會影響設計決定，例如管線流通與快取記憶體組織。各抽象化層級可以提供細節程度與可以模擬之系統大小之間的取舍條件(tradeoff)。VLSI 設計者主要的考量是電路與邏輯模擬。本章將重點放在利用 SPICE 所作的模擬，而 9.3 節將討論邏輯模擬。

如前面幾章所述，到底是以紙上分析作業來模擬電路行為比較好，還是用模擬的方式比較好呢？VLSI 電路很複雜，而且現代電晶體都具有非線性與非理想行為，因此有必要進行模擬以準確預測詳細的電路行為。即使在延遲或傳導特性有收斂解(closed-form solution)時，用手在大量電路上來作也會因為太花時間而不可行。而另一方面，電路模擬非常容易受到錯誤的影響：也就是垃圾進，垃圾出(Garbage in, garbage out, GIGO)。模擬器可以接受設計所提供的相對性模型，但是很容易就會產生出不精確或不完整的模型。此外，模擬器僅適用於設計者所提供的刺激，而且很容易就會忽略掉最惡劣狀況下的刺激。就如同有經驗的程式設計師了解到在除錯之前程式不會正常運作一般，有經驗的 VLSI 設計師也知道第一次模擬不會反映出相對性。因此，電路設計者需要有很好的直覺才能了解電路的運作，而且應該能夠在模擬進行之前就預測出大概的結果。只有在預期與模擬相符時，對於結果才会有信心。實際上，電路設計者要仰賴手算分析以及模擬兩者，或者就像[Glasser 85]所說的：「分析所得之理解而導引]所得之模擬。」

本章將以範例說明 SPICE 的導覽，然後再討論電晶體的模型與擴散電容。本章其餘的部分將針對描述製程與檢查電路與導線之性能、電力與正確性的模擬技術。

5.2 SPICE 導覽

SPICE (Simulation Program with Integrated Circuit Emphasis，原意為「整合電路重點之模擬程式」)原本是 1970 年代在柏克萊所開發的[Nagel 75]。這套程式可以解描述電晶體、電阻、電容器與電壓來源之類元件的非線性微分方程式。SPICE 可以提供許多種分析電路用的方法，而 VLSI 設計者主要的興趣是在於輸入固定或隨時間變化之數據時可以預測節點電壓的 DC 與瞬間電流分析。SPICE 原本是以 FORTRAN 語言所開發，而且具有某些特性，尤其是在與其傳承相關的檔案格式方面。大多數平台上都有一些可用的免費版本 SPICE，但是商業版本的 SPICE 可以提供更確實的數值收斂。尤其 HSPICE 廣泛使用於工業界因為收斂性較佳，可以支援最新的元件與導線模型，並且具有大量測量與最佳化電路的加強功能。PSPICE 則是另外一種商業用版本，另外還有學生用的有限功能版本。本章中所有範例都是用 HSPICE，而且通常都無法在一般的 SPICE 上執行。

使用 SPICE 的細節隨著版本與平台而有所不同，而所有版本 SPICE 的執行方式都是讀取一個輸入檔，再產生一個含有結果、警告與錯誤訊息的清單檔。輸入檔案通常都被稱為 SPICE 輸入檔(疊, deck)，而每一行都被稱為卡(card)，因為以前就是用一疊打好洞的卡一張一張插入主機裡。輸入檔中有一個包含各元件與節點的線路網清單，另外還有模擬選項、分析指令以及元件模型。線路網清單可以用手輸入或者從 CAD 裡面的電路圖或配置圖中選取。

好的 SPICE 輸入檔就像一個好的軟體，應該要容易讀取，可維修而且可以重複使用。註解以及空白區域可以讓檔案更容易閱讀。通常寫 SPICE 輸入檔的最佳方式就是用一個與所需差不多的輸入檔來修改成自己要的東西。

本節後面部分提供有一系列範例，說明數位 VLSI 電路所需的關鍵語法以及 SPICE 的能力。欲知更多細節，請參閱柏克萊 SPICE 手冊[Johnson 91]、冗長的 HSPICE 手冊或者是有關 SPICE 的教科書(例如[Kielkowsli 95, Foty 96])。

5.2.1 來源與被動元件

假設想找出圖 5.1(a)中 RC 電路的反應，我們輸入一個持續 50ps 從 0 上升到 1.8V 的輸入值。由於 RC 時間常數 $100\text{fF} \cdot 2\text{k}\Omega = 200\text{ps}$ 比輸入時間大許多，我們直覺地預期輸出會是一個時間常數 200ps 的指數函數，以不對稱方式趨近於 1.8V 的最終值。圖 5.2 為本模擬的 SPICE 輸入檔，而圖 5.1(b)則是輸入與輸出的反應。

開頭是星號(*)的卡是註解。SPICE 輸入檔的第一張卡必須是註解，通常都用來說明模擬的標題。把 SPICE 輸入檔視為電腦程式，而依循相同的步驟為輸入檔加註解是不錯的做法。尤其是在一開始就說明模擬的作者、日期與目的將有助於以後必須要修改輸入檔的作業(例如晶片在晶片試調(silicon debug)階段，而我們檢視舊模擬以找出可能失敗的原因的時候。)

控制卡的開頭是一個點(.)。option post 這張卡下令要 HSPICE 把結果用波形顯示器寫到一個檔案裡。SPICE 的最後一張卡必須是.end。

線路網清單中每一張卡的開頭是顯示電路元件型式的一個字母。請注意 SPICE 與

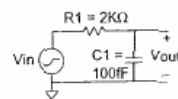
字母的大小寫無關。表 5.1 所示為一般元件。在此案例中，電路包含有一個叫做 V_{in} 的電壓來源、一個叫 R1 的電阻以及一個叫 C1 的電容。電路中的節點被稱為 in、out 以及 gnd。gnd 是一個被定義成 0V 參考點的特殊節點名稱。單位中有一或兩個字母，第一個字母表示強度的等級，如表 5.2 所示，而第二個字母則是為了方便所用的單位(例如 F 是法拉[farad]，而 s 是秒)，而且 SPICE 會把它忽略掉。例如 100fF 的電容可以標示為 100fF、100f 或直接寫成 100e-15。

電壓來源定義為片段線性(Piecewise Linear, PWL)來源。波形是以(時間，電壓)對的任意數來描述。其他的一般來源包括 DC 電源以及脈衝源。稱為 Vdd 並且將 vdd 設定在 2.5V 的 DC 電壓電源可以表示如下：

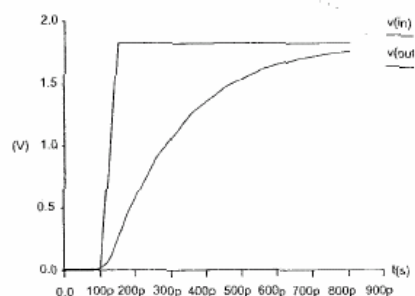
```
Vdd      vdd      gnd      2.5
```

脈衝源很適合像時鐘這一類的重複訊號。圖 5.3 所示為脈衝源的一般型式。例如，一個具有 1.8V 波動、800ps 週期、100ps 升降時間以及 50% 工作時間(Duty Time，亦即上升下降時間相等)的時鐘可以表示為：

```
Vck      clk      gnd      PULSE 0 1.8 0ps 100ps 100ps 300ps 800ps
```



(a)



(b)

圖 5.1 RC 電路反應

```

* rc.sp
* David_Harris@hmc.edu 2/2/03
* Find the response of RC circuit to rising input

-----
* Parameters and models
-----
.option post

-----
* Simulation netlist
-----
Vin  in  gnd  pwl  0ps 0 100ps 0 150ps 1.8 800ps 1.8
R1   in  out  2k
C1   out  gnd  100f

-----
* Stimulus
-----
.tran 20ps 800ps
.plot v(in) v(out)
.end

```

圖 5.2 RC spice 輸入檔

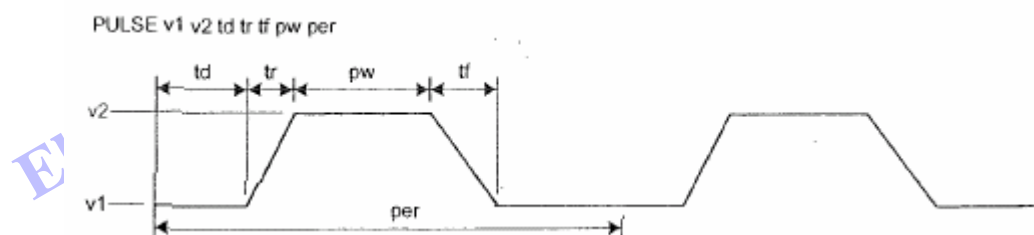


圖 5.3 PULSE 波形

這項刺激可以說明瞬間電流分析(.tran)應以最大步進大小(Step Size)20ps來進行 800ps。當繪製節點電壓時，步進大小決定點與點之間的時間。

.plot 指令可以產生所述節點變數的文字圖形(此處為節點 in 與 out 的電壓)，就如同圖 5.4 所示。同樣地，.print 敘述可以將結果列印成多行的表格。這兩個

字母	元件名稱
R	電阻
C	電容
L	電感
K	互感
V	獨立電壓來源
I	獨立電流來源
M	MOSFET
D	二極體
Q	雙極電晶體
W	有損傳輸線
X	子電路
E	電壓控制電壓來源
G	電壓控制電流來源
H	電流控制電壓來源
F	電流控制電流來源

字母	單位	大小
a	atto	10 ⁻¹⁸
f	femto	10 ⁻¹⁵
p	pico	10 ⁻¹²
n	nano	10 ⁻⁹
u	micro	10 ⁻⁶
m	mili	10 ⁻³
l	kilo	10 ³
x	mega	10 ⁶
g	giga	10 ⁹

指令都可以顯示 FORTRAN 以及印表機的工作歷程。在配備圖形使用者介面的現代電腦中，通常都比較常用 .option post 指令，它可以產生出一個含有所述(瞬間電流)分析結果的檔案(此處是 rc.tr0)。然後可以用另外的圖形瀏覽器來觀看與處理波形。AWAVES 與 COSMOSCOPE 是與 HSPICE 相容的圖形瀏覽器。

legend:
a: v(in)
b: v(out)

time (ab)	v(in)	0.	500.0000m	1.0000	1.5000	2.0000
0.	0.	+	+	+	+	+
20.0000p	0.	2	+	+	+	+
40.0000p	0.	2	+	+	+	+
60.0000p	0.	2	+	+	+	+
80.0000p	0.	2	+	+	+	+
100.0000p	0.	2	+	+	+	+
120.0000p	720.0000m +b	+	+	a+	+	+
140.0000p	1.440	+	b	+	+	a
160.0000p	1.800	+	+b	+	+	+a
180.0000p	1.800	+	+	b	+	+a
200.0000p	1.800	+	+	+b	+	+a
220.0000p	1.800	+	+	+	b	+a
240.0000p	1.800	+	+	+	+b	+a
260.0000p	1.800	+	+	+	+	b
280.0000p	1.800	+	+	+	+	b+
300.0000p	1.800	+	+	+	+	+b
320.0000p	1.800	+	+	+	+	+
340.0000p	1.800	+	+	+	+	b
360.0000p	1.800	+	+	+	+	b
380.0000p	1.800	+	+	+	+	+b
400.0000p	1.800	+	+	+	+	+b
420.0000p	1.800	+	+	+	+	+
440.0000p	1.800	+	+	+	+	b
460.0000p	1.800	+	+	+	+	b
480.0000p	1.800	+	+	+	+	b
500.0000p	1.800	+	+	+	+	+b
520.0000p	1.800	+	+	+	+	+b
540.0000p	1.800	+	+	+	+	+b
560.0000p	1.800	+	+	+	+	+b
580.0000p	1.800	+	+	+	+	+b
600.0000p	1.800	+	+	+	+	+b
620.0000p	1.800	+	+	+	+	+
640.0000p	1.800	+	+	+	+	b
660.0000p	1.800	+	+	+	+	b
680.0000p	1.800	+	+	+	+	b
700.0000p	1.800	+	+	+	+	b
720.0000p	1.800	+	+	+	+	b
740.0000p	1.800	+	+	+	+	b
760.0000p	1.800	+	+	+	+	b
780.0000p	1.800	+	+	+	+	b
800.0000p	1.800	+	+	+	+	ba

Ele

圖 5.4 RC 電路反應的文字圖形

5.2.2 電晶體 DC 分析

在熟悉新的CMOS製程中前幾個步驟之一就是要看電晶體的I-V特徵。圖 5.5(a)所示為一個單位(4/2λ) nMOS電晶體在 180nm製程中於 $V_{DD} = 1.8V$ 時的測試電路。圖 5.5(b)是用圖 5.6 的SPICE輸入檔所繪製而成的I-V特徵圖。

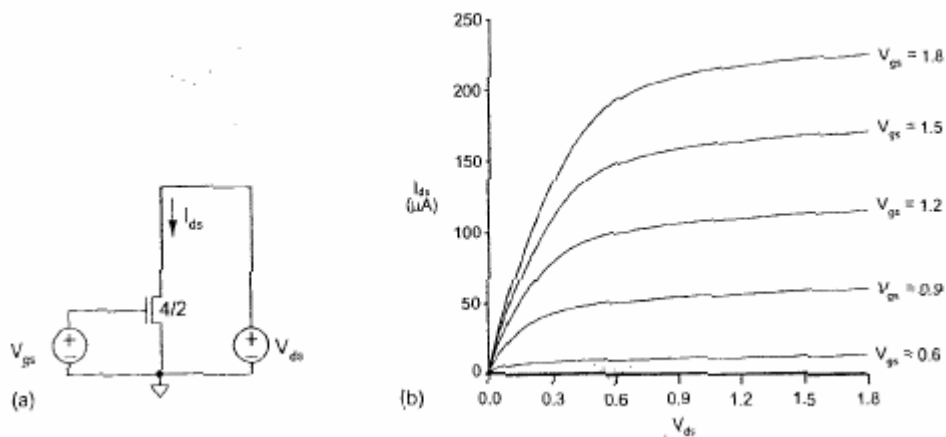


圖 5.5 MOS I-V 特徵

```

* mosiv.sp
-----
* Parameters and models
-----
.include '../models/tsmc180/models.sp'
.temp 70
.option post

-----
* Simulation netlist
-----
*nmos
Vgs  g  gnd  0
Vds  d  gnd  0
M1   d  g    gnd  gnd  NMOS  W=0.36u L=0.18u

-----
* Stimulus
-----
.dc Vds 0 1.8 0.05 SWEEP Vgs 0 1.8 0.3
.end

```

圖 5.6 MOS I-V SPICE 輸入檔

.include 會從磁碟中讀取另一個 SPICE 檔案。在這個範例中，它會載入第 5.3 節討論的元件模型。這個電路使用 2 個內定為 0V 的獨立電壓來源；這些電壓可以用 .dc 指令來改變。NMOS 電晶體是用以下這個語法以 MOSFET 元件來定義：

```
Mname drain gate source body type W=<width> L=<length>
```

.dc 指令以 0.05V 的增加量將電壓源 V_{ds} 的 DC 電壓從 0 改變到 1.8V。這個步驟會隨著 V_{gs} 以 0.3V 的增加量從 0 改變到 1.8V 而重複好幾次，以計算各個 V_{gs} 值時的 I_{ds} 對 V_{ds} 曲線。

5.2.3 換流器瞬間電流分析

圖 5.7 為無負載單位換流器的步進反應，並且標註有傳遞延遲 (Propagation Delay) 與 20%-80% 升降時間。請注意由於沒有負載而因拔靴法 (Boot-Strapping) 所發生的超越量。圖 5.8 為模擬用的 SPICE 輸入檔。

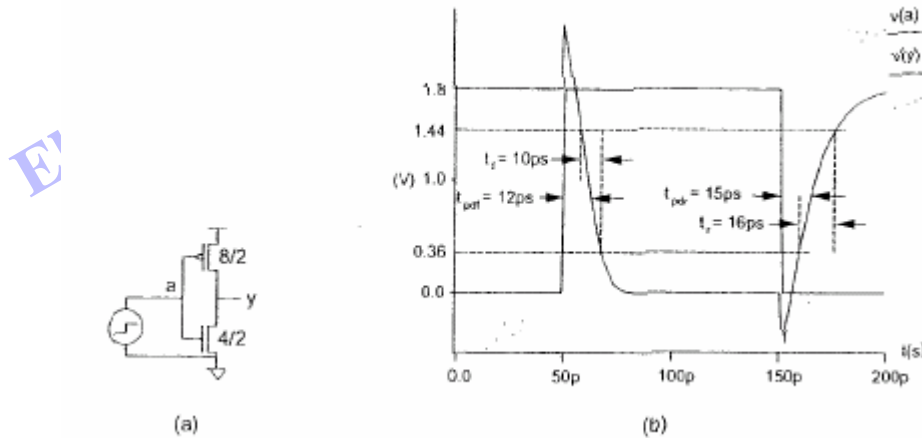


圖 5.7 無負載換流器

這個輸入檔介紹各參數與比例。 .param 敘述定義一個叫做 SUPPLY 的參數，讓它具有 1.8 的數值，接著再用來設定 V_{dd} 以及輸入脈衝的幅度。如果我們想評估不同供應電壓時的反應，我們只需要修改 .param 即可。 .scale 可以為所有尺寸設定一個比例因數，並內定為以公尺測量。在這個案例中，比例設定為 $\lambda = 90\text{nm}$ 。現在電晶體的寬度與長度便可以用 λ 來量測，而不需要用公尺。


```

* inv.sp
-----
* Parameters and models
-----
.param SUPPLY=1.8
.option scale=90n
.include ../models/tsmc180/models.sp
.temp 70
.option post

-----
* Simulation netlist
-----
vdd vdd gnd 'SUPPLY'
Vin a gnd PULSE 0 'SUPPLY' 50ps 0ps 0ps 100ps 200ps
M1 y a gnd gnd NMOS W=4 L=2
+ AS=20 PS=18 AD=20 PD=18
M2 y a vdd vdd PMOS W=8 L=2
+ AS=40 PS=26 AD=40 PD=26

-----
* Stimulus
-----
.tran 1ps 200ps
.end

```

圖 5.8 INV SPICE 輸入檔

請記得寄生延遲(Parasitic Delay)與擴散電容(Diffusion Capacitance)關係密切，而擴散電容又與源極與汲極的面積與周長有關。由於換流器中的每一個擴散區域都必須要有接觸，所以其外型看起來就像圖 2.9(a)所示。擴散寬度等於電晶體的寬度，而擴散長度為 5λ 。因此，源極與汲極的面積就等於 $AS = AD = 5W\lambda^2$ ，而周長等於 $PS = PD = (2W + 10)\lambda$ 。請注意卡中第一列的加號(+)表示這一行是從前一張卡延續下來的。這些尺寸同樣也受比例因數所影響。

5.2.4 子電路與測量

製程固有速度(inherent speed)最簡單的度量之一就是 4 換流器延遲的扇出(fanout-of-4 inverter delay)。圖 5.9(a)為測量這項延遲的一個電路。nMOS與pMOS電晶體大小(單位 $4/2\lambda$ 電晶體的倍數)分別註記在各閘門的下方與上方。X3 是測試中的換流器，而X4 是其負載，而這個負載要比X3 大 4 倍。對第一級來說，這兩個換流器就已經足夠。然而X3 的延遲就如同 4.2.5.1 節所述一般與輸入斜率有關。要取得實際的輸入斜率可以用一對FO4 換流器X1 與X2 來驅動節點c。此外，如 4.2.5.4 所述，X4 的輸入電容同時取決於其 C_{gs} 與 C_{gd} 。當節點e切換時， C_{gd} 會乘以米勒倍數(Miller-Multiplied)，而且如果e突然切換時， C_{gd} 就會加倍。當用X5 施加在e上時，它會用一個較慢較實際的速率切換，稍微將節點上所示之有效電容減少X4。圖 5.9(b)標示有上升與下降的延遲。

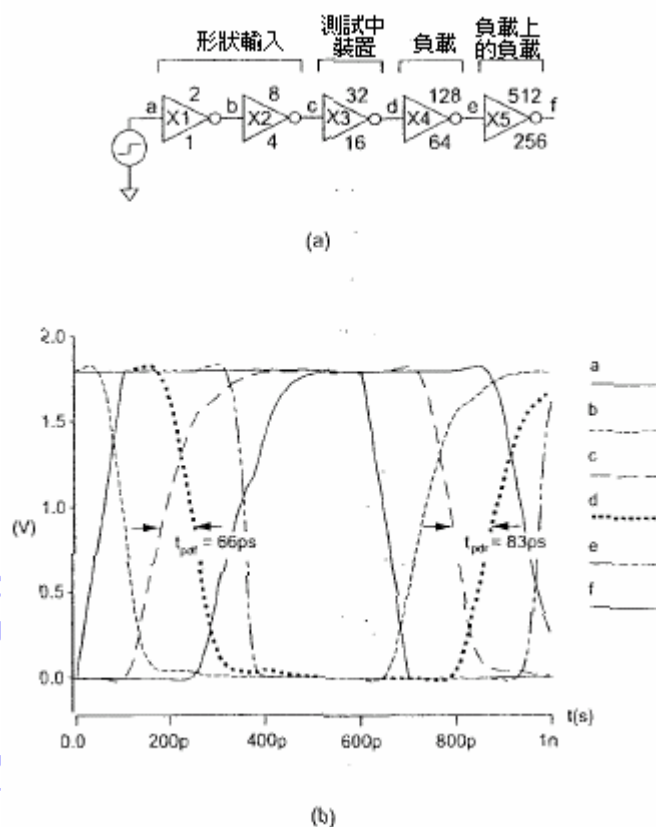


圖 5.9.4 換流器延遲的扇出

當把一般電路元件萃取為子電路時，SPICE 輸入檔就比較容易檢視與維護。例如，圖 5.10 所示的輸入檔就是用一個換流器子電路來計算 FO4 換流器延遲。

.global 敘述將 vdd 與 gnd 定義為可以從子電路裡面參考的總體節點。換流器被宣告為具有兩個端子 a 與 y 的子電路。另外它也接受兩個敘述 nMOS 與 pMOS 電晶體寬度的參數，而這兩個參數分別被內定為 4 與 8。源極與汲極面積與周長是電晶體寬度的方程式。HSPICE 會評估有單引號註記的方程式，而這些方程式可能包括參數、常數、括號、+、-、*、/以及**(乘方)。

```

* fo4.sp
-----
* Parameters and Models
-----
.param SUPPLY=1.8
.param N=4
.option scale=90n
.include '../models/tsmc180/models.sp'
.temp 70
.option post
-----
* Subcircuits
-----
.global vdd gnd
.subckt inv a y N=4 P=8
M1 y a gnd gnd NMOS W='N' L=2
+ AS='N*5' PS='2*N+10' AD='N*5' PD='2*N+10'
M2 y a vdd vdd PMOS W='P' L=2
+ AS='P*5' PS='2*P+10' AD='P*5' PD='2*P+10'
.ends
-----
* Simulation netlist
-----
Vdd vdd gnd 'SUPPLY'
Vin a gnd PULSE 0 'SUPPLY' 0ps 100ps 100ps 500ps 1000ps
X1 a b inv * shape input waveform
X2 b c inv M='N' * reshape input waveform
X3 c d inv M='N*2' * device under test
X4 d e inv M='N*3' * load
X5 e f inv M='N*4' * load on load
-----
* Stimulus
-----
.tran 1ps 1000ps
.measure tpdr * rising prop delay
+ TRIG v(c) VAL='SUPPLY/2' FALL=1
+ TARG v(d) VAL='SUPPLY/2' RISE=1
.measure tpdf * falling prop delay
+ TRIG v(c) VAL='SUPPLY/2' RISE=1
+ TARG v(d) VAL='SUPPLY/2' FALL=1
.measure tpd param='(tpdr+tpdf)/2' * average prop delay
.measure trise * rise time
+ TRIG v(d) VAL='0.2*SUPPLY' RISE=1
+ TARG v(d) VAL='0.8*SUPPLY' RISE=1
.measure tfall * fall time
+ TRIG v(d) VAL='0.8*SUPPLY' FALL=1
+ TARG v(d) VAL='0.2*SUPPLY' FALL=1
.end

```

圖 5.10 FO4 SPICE 輸入檔

模擬線路網清單包含電力供應、輸入波形以及 4 個換流器。每一個換流器都是一個子電路(X)元件。由於沒有規定 N 和 P 值，所以就用內定值。M 參數把子電路裡的所有電流乘以所述之因數，就相當於有 M 個元件以並聯連接。此處扇出值(Fanout)是以參數 H 表示。因此 X2 具有 4 個單位換流器的電容與輸出電流，而 X3 相當於 16。將換流器建立到模型裡的另一個方式就是用 N 與 P 參數：

```

X1  a  b  inv  N=4    P=8    * shape input waveform
X2  b  c  inv  N=16   P=32   * reshape input waveform
X3  c  d  inv  N=64   P=128  * device under test
X4  d  e  inv  N=256  P=512  * load
X5  e  f  inv  N=1024 P=2048 * load on load

```

然而，4 倍單位寬度的電晶體的輸入電容或輸出電流與 4 個並聯在一起的單位換流器並不會完全相同，所以 M 參數比較常用。

在此例中，子電路的宣告以及模擬的線路網清單都是 SPICE 輸入檔的一部分。當採用標準元件庫(Standard Cell Library)時，通常會將子電路宣告保留在其本身的檔案中，再用 .include 敘述來呼叫。當從圖形或配置 CAD 系統中節錄模擬線路網清單時，通常會把線路網清單放在一個檔案中，再用 .include 敘述來呼叫。

.measure敘述可以測量模擬結果，並把結果列印到檔案中。輸入檔會測量出上升傳遞延遲 t_{pdr} 作為輸入值c第一次降到 $V_{\text{DD}}/2$ 以下的時間以及輸出值d第一次超過 $V_{\text{DD}}/2$ 的時間之間的差異。TRIG與TARG是觸發事件(trigger event)與目標事件(target event)，而延遲就是在這兩者之間測得的。.measure敘述也可以用來計算其他測量值的方程式。例如，平均FO4 換流器傳遞延遲 t_{pd} 是 t_{pdr} 與 t_{pdf} 的平均值，亦即 $t_{\text{pd}} = 75\text{ps}$ 。20%-80%上升時間是 $t_r = 94\text{ps}$ ，而下降時間是 $t_f = 67\text{ps}$ 。

5.2.5 最佳化

在許多範例中，我們都假設 P/N 比 2:1 可以讓上升與下降延遲大略相等。FO4 換流器模擬顯示 2:1 的比例可以得到比下降延遲為慢的上升延遲，這是因為 pMOS 的機動性比 nMOS 機動性的一半還要少。您可以用不同的內定 P 值來重複執行模擬來找出相等延遲的比例。HSPICE 有內建的最佳化能力可以自動微調參數來達成某些目的，並且會報告什麼樣的參數值可以得到最佳的結果。圖 5.11 顯示用最佳化工具所作的 FO4 換流器模擬的修改版本。

```

* fo4opt.sp
*-----
* Parameters and models
*-----
.param SUPPLY=1.8
.option scale=90n
.include '../models/tsmc180/models.sp'
.temp 70
.option post
*-----
* Subcircuits
*-----
.global vdd gnd
.subckt inv a y N=4 P=8
M1 y a gnd gnd NMOS W='N' L=2
+ AS='N*5' PS='2*N+10' AD='N*5' PD='2*N+10'
M2 y a vdd vdd PMOS W='P' L=2
+ AS='P*5' PS='2*P+10' AD='P*5' PD='2*P+10'
.ends
*-----
* Simulation netlist
*-----
vdd vdd gnd 'SUPPLY'
Vin a gnd PULSE 0 'SUPPLY' 0ps 100ps 100ps 500ps 1000ps
X1 a b inv P='P1' * shape input waveform
X2 b c inv P='P1' M=4 * reshape input waveform
X3 c d inv P='P1' M=16 * device under test
X4 d e inv P='P1' M=64 * load
X5 e f inv P='P1' M=256 * load on load
*-----
* Optimization setup
*-----
.param P1=optrange(8,4,16) * search from 4 to 16, guess 8
.model optmod opt itropt=30 * maximum of 30 iterations
.measure bestratio param='P1/4' * compute best P/N ratio
*-----
* Stimulus
*-----
.tran 1ps 1000ps SWEEP OPTIMIZE=optrange RESULTS=diff MODEL=optmod
.measure tpdr * rising propagation delay
+ TRIG v(c) VAL='SUPPLY/2' FALL=1
+ TARG v(d) VAL='SUPPLY/2' RISE=1
.measure tpdf * falling propagation delay
+ TRIG v(c) VAL='SUPPLY/2' RISE=1
+ TARG v(d) VAL='SUPPLY/2' FALL=1
.measure tpd param='(tpdr+tpdf)/2' goal=0 * average prop delay
.measure diff param='tpdr-tpdf' goal = 0 * diff between delays
.end

```

圖 5.11 FOAOPT SPICE 輸入檔

X1 到X4 的子電路會用P1 的寬度來蓋過自己內定的pMOS寬度。在最佳化設定中會測量 t_{pdr} 與 t_{pdf} 的差異值。最佳化的目的就是要將這個差值壓到 0，而做法是P1 可能會從 4 變到 16，而初始值是 8。最佳化工具會進行最多 30 次的重複運算來找出最佳的P1 值。因為nMOS的寬度是固定在 4，所以最佳的P/N比就會是 P1/4。瞬間電流分析有一個包含變數參數的SWEEP敘述、所需的結果以及重複運算的次數。

HSPICE 可以計算出上升與下降延遲相等的 P/N 比是 3.6:1，而所得的上升與下降延遲為 84ps。這比 2:1 比例所提供的要慢，而且需要用到耗費面積與電力的大型 pMOS 電晶體，所以通常很少用到這麼高的比例。

有一個類似的情形就是要找出可以提供最低平均延遲的 P/N 比。我們把.tran 卡改成 RESULTS=tpd，就可以找到最佳 P/N 比 1.4:1，而上升、下降與平均傳遞延遲分別是 87、59 與 73ps。每當您要進行最佳化時，一定要記得在考慮最適條件以外，還要考慮到對從這一點所產生之偏差的敏感度。後續的模擬可以得到 1.2:1 到 1.7:1 之間的任何一個 P/N 比都可以得出 73ps 的平均傳遞延遲，所以不需要死守著 1.4:1 這個「最適條件」不放。實際上最佳的 P/N 比是採用較小 pMOS 元件來節省面積與電力，以及採用較大元件來達到最接近相等上升/下降時間，並避免因用較弱的 pMOS 電晶體而產生電路中很慢的上升邊緣所引發的熱電子可靠性問題這兩者之間的妥協。P/N 比將在以下的 6.2.1.6 節繼續討論。

5.2.6 其他 HSPICE 指令

整本 HSPICE 手冊有兩千多頁，而且功能多到無法在此全部說明。本節將另外再討論一些非常有用的指令。5.3 節將說明電晶體模型與資料庫呼叫，而 5.6 節將討論用有損傳輸線(lossy transmission lines)來建立導線的模型。

.option accurate

縮小整合誤差以取得更準確的結果。在振盪器與高增益類比電路上，或者是當結果看起來有問題時很有用。

.option autostop

當所有.measure 的結果全都已取得時就會停止，而非一直把整個.tran 卡都跑完。這可以大量減少模擬時間。

.temp 0 70 125

以 0°、70°與 125°這三個溫度重複模擬三次。元件模型可以納入一些訊息說明改變溫度會如何改變元件的性能。

.op

列印在 DC 操作點時的電壓、電流與電晶體偏壓狀態(transistor bias condition)。

5.3 元件模型

5.2 節裡大多數的範例中都有一個包含電晶體模型的檔案。SPICE 可以提供許多種具有各種複雜度與精確性之間妥協的 MOS 電晶體模型。第 1 級與第 3 級的電晶體具有歷史上的重要性，但是它們已經不適合用來建立非常小型的現代電晶體模型。BSIM 模型比較精確，而且也是目前最廣泛使用的模型。有些公司會使用他們自己的專用模型。本節將簡短說明各個模型的主要特色，另外也會敘述如何建立擴散電容的模型以及如何在各種不同的製程邊界(process corner)上執行模擬。模型的說明只是要提供一個模型在功能與限制方面的大概；如需要更詳細的說明，請參考 SPICE 的使用手冊。

5.3.1 第 1 級模型

SPICE 第 1 級，或者是 Shichman-Hodges 模型[Shichman 68]與 EQ(2.10)所述的 Shockley 模型關係非常密切，並且用通道長度調變效應(channel length modulation)以及本體效應(body effect)加強過。基本的電流模型是：

$$I_{ds} = \begin{cases} 0 & V_{gs} < V_t & \text{切點} \\ KP \frac{W_{eff}}{L_{eff}} (1 + LAMBDA \cdot V_{ds}) \left(V_{gs} - V_t - \frac{V_{ds}}{2} \right) V_{ds} & V_{ds} < V_{gs} - V_t & \text{線性} \\ \frac{KP}{2} \frac{W_{eff}}{L_{eff}} (1 + LAMBDA \cdot V_{ds}) (V_{gs} - V_t)^2 & V_{ds} > V_{gs} - V_t & \text{飽和} \end{cases} \quad (5.1)$$

SPICE 模型裡的參數都是用大寫字母表示。請注意 $KP(W_{eff}/L_{eff})$ 是用 β 來代替，其中 KP 是扮演 EQ(2.7) 中 k' 角色的模型參數，而 W_{eff} 與 L_{eff} 則是 2.4.8 節所述的有效寬度與長度。LAMBDA 可以建立通道長度調變效應的模型(請見 2.4.2 節)。

臨界電壓(threshold voltage)是由源極到基體電壓(source-to-body voltage)利用本體效應來調變(請見 2.4.3 節)。對於非負(non-negative)電壓 V_{sb} 而言，其臨界電壓為：

$$V_t = V_{TO} + GAMMA \left(\sqrt{PHI + V_{sb}} - \sqrt{PHI} \right) \quad (5.2)$$

請注意這跟 EQ(2.30) 一模一樣，其中 V_{TO} 是「零偏壓(zero-bias)」臨界電壓 V_0 ， $GAMMA$ 是本體效應係數 γ ，而 PHI 是表面電位 ϕ_s 。

閘極電容(Gate Capacitance)是從氧化物厚度 TOX 計算而得。HSPICE 內定的閘極電容模型適合用來找出數位電路的瞬間電流反應。另外還有更精細的模型可以來捕捉不可倒置效應(nonreciprocal effect)，而這個效應對類比設計來說很重要。

第 1 級模型在教學方面非常有用，因為很容易和手算分析配合，不過對於現代設計而言過於簡化。圖 5.12 是一個第 1 級模型卡的範例，顯示輸入卡的語法。這張卡也包含計算擴散電容用的用語，如 5.3.4 節所述一般。

```
.model NMOS NMOS (LEVEL=1 TOX=40e-10 KP=155E-6 LAMBDA=0.2
+
+           VTO=0.4 PHI=0.93 GAMMA=0.6
+           CJ=9.8E-5 PB=0.72 MJ=0.36
+           CJSW=2.2E-10 PHP=7.5 MJSW=0.1)
```

圖 5.12 第 1 級模型卡範例

5.3.2 第 2 級與第 3 級模型

SPICE 的第 2 級與第 3 級模型可以增加速度飽和(velocity saturation)、遷移率衰減(mobility degradation)、次臨界區導電(subthreshold conduction)以及源極引發能障衰退(drain-induced barrier lowering，亦稱為汲極偏壓導致通道能障降低效應)等等各種效應。第 2 級模型來自於 Grove-Frohmman 方程式 [Frohmman 69]，而第 3 級模型來自於可以提供類似精確度、較短模擬時間以及較佳收斂度的經驗公式。然而這些模型仍舊無法與現代電晶體中所測得之 I-V 特徵作良好配合。

5.3.3 BSIM 模型

柏克萊的短通道IGFET¹模型(BSIM)是一種非常精密的模型，目前已廣泛使用於電路模擬。模型是從以下的元件物理所衍生而來，但採用了大量的參數來配合現代電晶體的行為。BSIM 的第 1、第 2、第 3v3 與第 4 版執行時分別同 SPICE 的第 13、39、49 與 54 級。

BSIM 第 3v3 版需要一整本書[Cheng 99]來描述這個模型，其中包含了 100 多種參數，而且元件方程式長達 27 頁。這對於數位電路模擬來說非常好，只不過無法將閘極漏電流(gate leakage)建立到模型中。本模型的功能包括：

- ⊙ 跨次臨界區、線性區以及飽和區的連續與可辨認 I-V 特徵可提供較佳收斂性
- ⊙ 如 V_{th} 等各種參數對電晶體長度與寬度的敏感度
- ⊙ 包含本體效應與源極引發能障衰退的詳細臨界電壓模型
- ⊙ 速度飽和、遷移率衰減與其他短通道效應
- ⊙ 多重閘極電容模型
- ⊙ 擴散電容與電阻模型

¹IGFET 是絕緣閘場效電晶體(Insulated-Gate Field Effect Transistor)的縮寫，這是一種 MOSFET 系統。

BSIM 第 4 版在本書著作時還非常地新穎，它增加了對超薄閘極的閘極漏電流以及其他效應的支援。由於這些效應很快就變得很重要而且描述也比較完整，設計者必須要在盲目相信閘極漏電流模型之前先與製程工程師溝通。

有些元件參數，像臨界電壓，會隨著元件尺寸而有很大的變化。BSIM 模型可以分裝成幾個涵蓋以 LMIN、LMAX、WMINM 與 WMAX 等參數所敘述之長度與寬度範圍的模型。例如，一個模型可以涵蓋通道寬度 0.18-0.25 μm 的電晶體，另一個可以涵蓋 0.25-0.5 μm ，而第三個可以涵蓋 0.5-5 μm 。如果有電晶體不符合這幾個分裝模型時，SPICE 會發出警告。

由於 BSIM 模型太複雜，要從以下方程式中找出傳遞延遲、切換臨界 (switching threshold)、雜訊界限(noise margins)等的封閉方程式並不實際，不過要透過電路模擬來找出這些性質卻不難。第 5.4 節將說明簡單的模擬來在操作區域中繪製大多數數位設計者感興趣的元件特徵，並且萃取出在介面轉換 (switching transition) 中平均的有效電容與電阻。簡單的 RC 模型會持續給設計者有關邏輯閘極的重要想法。

5.3.4 擴散電容模型

源極或汲極擴散與本體之間的 p-n 接面 (p-n junction) 會形成一個逆偏壓二極體 (reverse-biased diode)。我們已經看過擴散電容決定閘極的寄生延遲，並且取決於擴散的面積與周長。HSPICE 提供一些方法來說明這種幾何關係，這是由 ACM 參數 (Area Calculation Method, 面積計算法) 所控制，而這個參數是電晶體模型卡的一部分。模型卡中也必須要如第 2.2.2.3 節所述具有接面與側壁擴散 (sidewall diffusion) 的數值。擴散電容模型在大多數元件模型中都很常見，包括第 1-3 級模型與 BSIM 在內。

根據預設值，HSPICE 模型採用 ACM=0。在此模型中，設計者必須詳細說明各電晶體源極與汲極的面積與周長。例如，表 5.3 所示為圖 2.9 的各擴散區尺寸 (面積以 λ^2 為單位，周長以 λ 為單位)。圖 5.13 所示為 SPICE 有關分享接觸擴散案例的敘述，其中假設 .option scale 設定為 λ 值。

	AS1 / AD2	PS1 / PD2	AD1 / AS2	PD1 / PS2
(a) 隔離接觸擴散	$W \cdot 5$	$2 \cdot W + 10$	$W \cdot 5$	$2 \cdot W + 10$
(b) 分享接觸擴散	$W \cdot 5$	$2 \cdot W + 10$	$W \cdot 3$	$W + 6$
(c) 合併接觸擴散	$W \cdot 5$	$2 \cdot W + 10$	$W \cdot 1.5$	$W + 3$

```

* (b): Shared contacted diffusion
M1 mid b bot gnd NMOS W='w' L=2
+ AS='w*5' PS='2*w+10' AD='w*3' PD='w+6'
M2 top a mid gnd NMOS W='w' L=2
+ AS='w*3' PS='w+6' AD='w*5' PD='2*w+10'

```

圖 5.13 分享接觸擴散電晶體的 SPICE 模型

SPICE 模型中也應該要包含 CJ、CJSW、PB、PHP、MJ 以及 MJSW 這幾個參數。假設擴散是逆偏壓的，而且面積與周長也有說明，則源極與本體之間的擴散電容的計算就如 2.3.3 節所示一般。

$$C_{ib} = AS \cdot CJ \cdot \left(1 + \frac{V_{ib}}{PB}\right)^{-MJ} + PS \cdot CJSW \cdot \left(1 + \frac{V_{ib}}{PHP}\right)^{-MJSW} \quad (5.3)$$

汲極方程式也很類似，其中模型參數的 S 換成 D。

BSIM3 模型可提供類似的面積計算模型(ACM=10)，而這個模型會考慮閘極旁的邊緣上的不同側壁電容。請注意 PHP 參數被改名為 PBSW 以保持一致性。

$$C_{ib} = AS \cdot CJ \cdot \left(1 + \frac{V_{ib}}{PB}\right)^{-MJ} + (PS - W) \cdot CJSW \cdot \left(1 + \frac{V_{ib}}{PBSW}\right)^{-MJSW} + W \cdot CJSWG \cdot \left(1 + \frac{V_{ib}}{PBSWG}\right)^{-MJSWG} \quad (5.4)$$

如果未指明面積與周長時，則內定值在 ACM=1 或 10 時為 0，大略低估了閘極的寄生延遲。HSPICE 也支援 ACM=1、2、3 與 12，而這些值可以在未指明面積與周長時提供非零的內定值。請檢查您的模型並熟讀 HSPICE 手冊。

擴散面積與周長也可以用來計算接面漏電流(junction leakage current)。不過這個電流與現代電晶體的次臨界漏電流相比通常都會被忽略掉。

5.3.5 設計邊界(Design corners)

工程師時常會模擬多重設計邊界中的電路來驗證橫跨元件特徵與環境變異的操作。HSPICE 有可以輕易改變程式庫的 .lib 卡。例如，圖 5.4 裡的輸入檔可以在 TT、FF 與 SS 這幾個邊界中執行無負載換流器步進反應的三項模擬。

```

* corner.sp
* Step response of unloaded inverter across process corners

-----
* Parameters and models
-----
.option scale=90n
.param SUP=1.8 * Must set before calling .lib
.lib '../models/tsmc180/opconditions.lib' TT
.option post

-----
* Simulation netlist
-----
vdd vdd gnd 'SUPPLY'
Vin a gnd PULSE 0 'SUPPLY' 200ps 0ps 0ps 500ps 1000ps
M1 y a gnd gnd NMOS W=4 L=2
+ AS=20 FS=18 AD=20 PD=18
M2 y a vdd vdd PMOS W=9 L=2
+ AS=40 FS=26 AD=40 PD=26

-----
* Stimulus
-----
.tran 1ps 1000ps
.alter
.lib '../models/tsmc180/opconditions.lib' FF
.alter
.lib '../models/tsmc180/opconditions.lib' SS
.end

```

圖 5.14 CORNER SPICE 輸入檔

輸入檔首先設定 SUP 為 1.8V 的標稱供電電壓，然後再呼叫.lib 卡在程式庫中讀取 TT 條件的敘述。在刺激中，.alter 敘述被用來重複改變後的模擬。總的來說，一共執行了三次模擬，而且產生出三個設計邊界的三組波形。

圖 5.15 為程式庫檔案。根據所述之程式庫的不同可以設定溫度(用.temp設定為攝氏溫度)，而且 V_{DD} 值 SUPPLY 也可由標稱 SUP 計算而得。程式庫會載入適當的 nMOS 與 pMOS 電晶體模型。快速製程檔具有較低的標稱臨界電壓 V_{t0} 、較大的橫向擴散 L_D 以及較低的擴散電容值。

```

* opconditions.lib
* For TSMC 180 nm process

* TT: Typical nMOS, pMOS, voltage, temperature
.lib TT
.temp 70
.param SUPPLY='SUP'
.include 'modelsTT.sp'
.endl TT

* SS: Slow nMOS, pMOS, low voltage, high temperature
.lib SS
.temp 125
.param SUPPLY='0.9 * SUP'
.include 'modelsSS.sp'
.endl SS

* FF: Fast nMOS, pMOS, high voltage, low temperature
.lib FF
.temp 0
.param SUPPLY='1.1 * SUP'
.include 'modelsFF.sp'
.endl FF

* FS: Fast nMOS, Slow pMOS, typical voltage and temperature
.lib FS
.temp 70
.param SUPPLY='SUP'
.include 'modelsFS.sp'
.endl FS

* SF: Slow nMOS, Fast pMOS, typical voltage and temperature
.lib SF
.temp 70
.param SUPPLY='SUP'
.include 'modelsSF.sp'
.endl SF

```

圖 5.15 OPCONDITIONS 程式庫

5.4 元件特徵

現代 SPICE 模型具有許多參數，而設計者無法輕易從模型檔案中找出關鍵的性能特徵。有一個比較方便的方式就是執行一組模擬來萃取有效電阻與電容、4 換流器延遲的扇出、I-V 特徵值以及其他有用的資料。本節將說明這些模擬，並且將比較各種 CMOS 製程的結果。