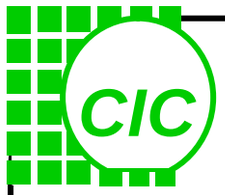


Full Custom IC Design Concepts

Training Manual

國家晶片系統設計製作中心

Jan. 2003

Full Custom IC Design Concepts

Jan. - 2003

國家晶片系統設計製作中心

Chip Implementation Center

王鴻猷 (06)5053041 * 210 ted@tn.cic.edu.tw

謝遠達 (06)5053041 * 107 ythsieh@tn.cic.eud.tw

課程大綱

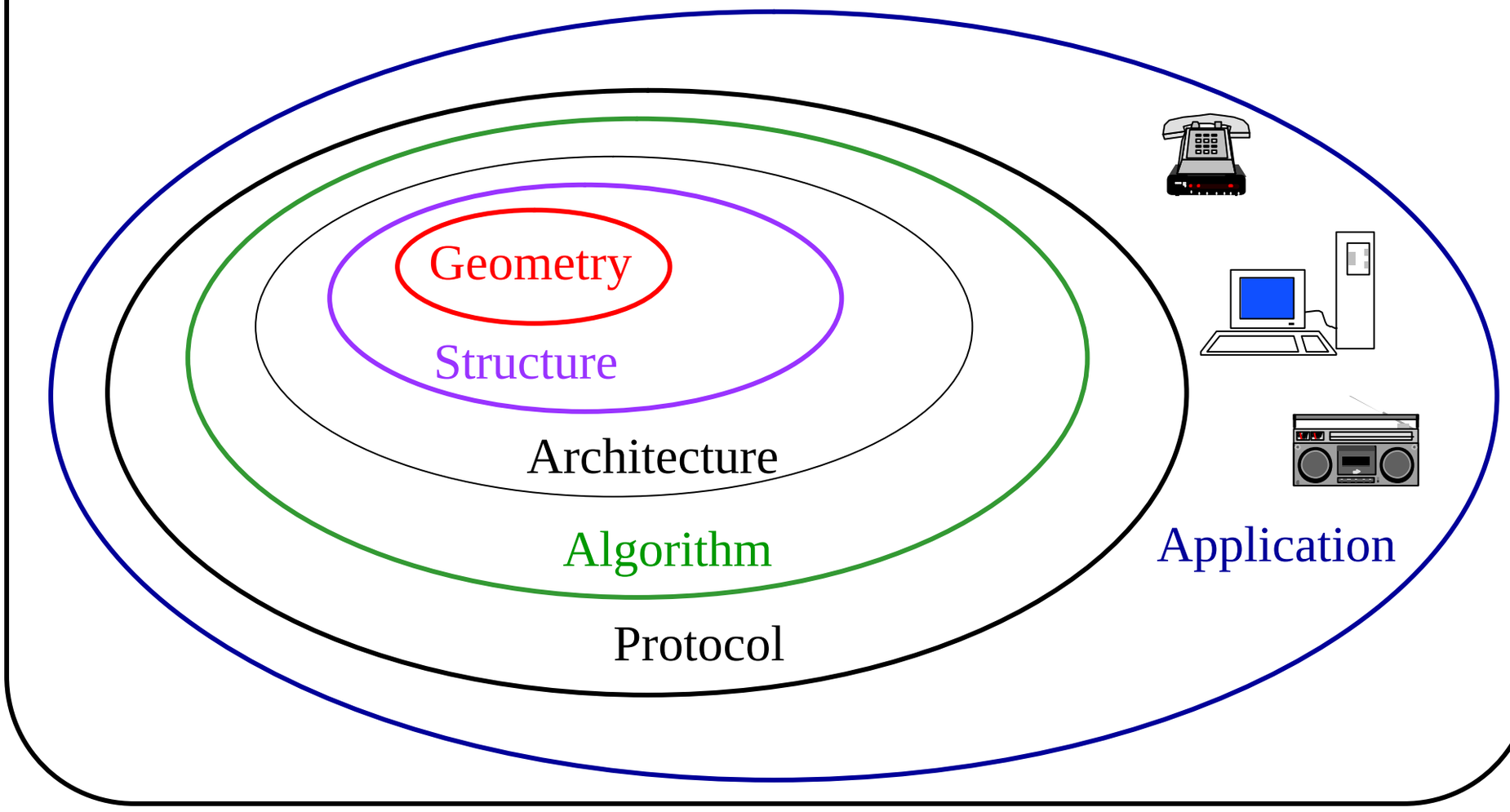
電路設計基本概念

1 電路設計流程

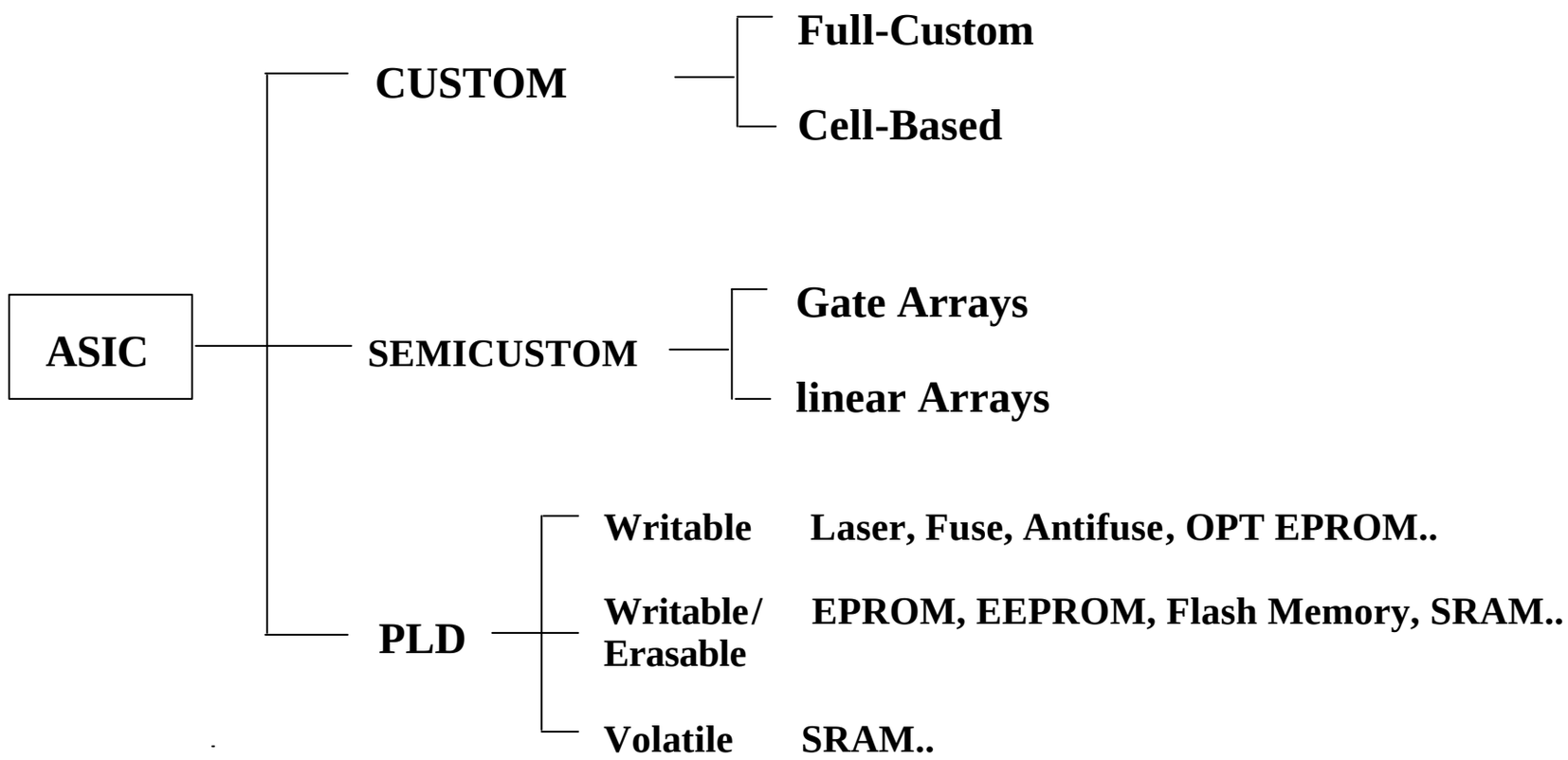
2 設計需求與設計考量
- 規格與需求
- 製程的選擇
- 模擬與驗證

3 CAD設計環境

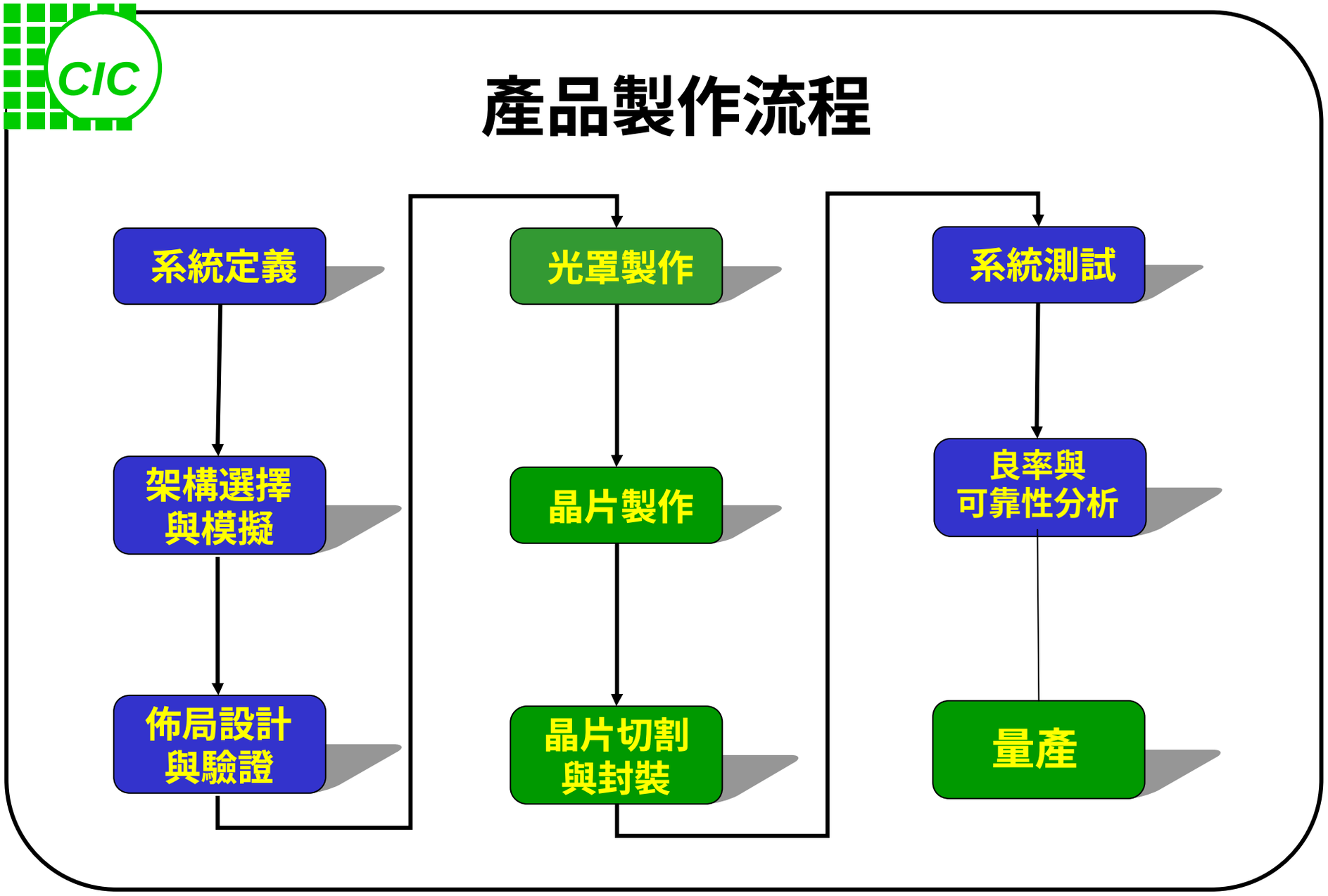
從概念到應用-積體電路的產生



Implementation of IC



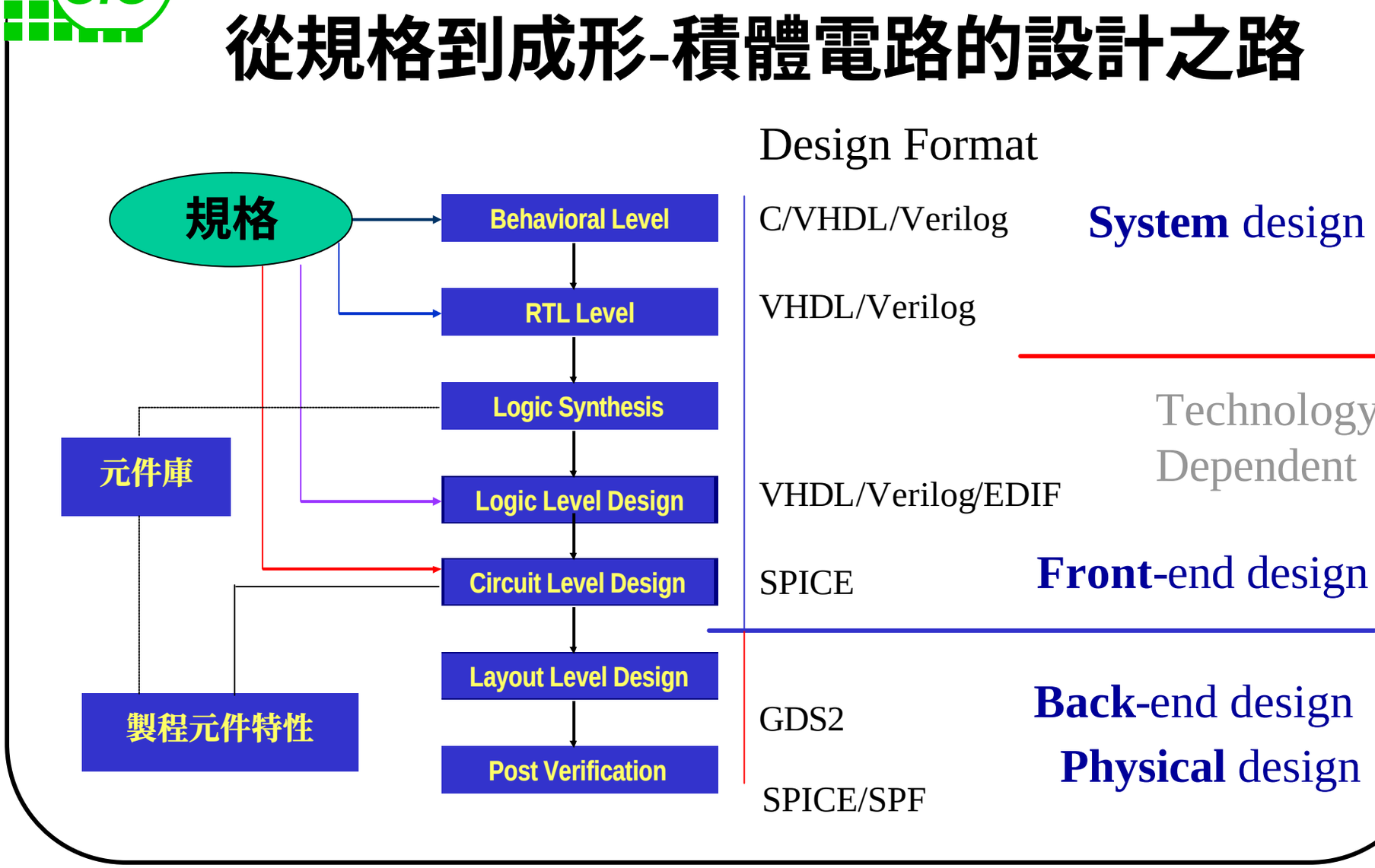
產品製作流程



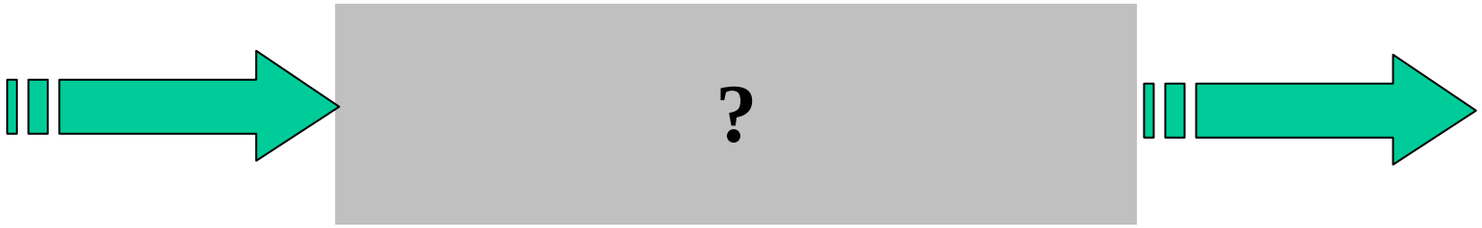
雛型IC設計步驟

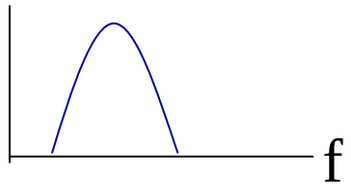
- **規格定義** 需求時脈頻率，輸出入時序、功能對應...
- **製程選擇** (0.35/0.25/0.18, logic, mixed-mode, Embed)CMOS, BiCMOS,GaAs
- **架構選擇** Dynamic/Static logic, Parallel/Serial/Pipelined ...
- **電路設計** 模組分割，需求定義，電路方塊設計與連接
- **電路模擬** 功能模擬，時序驗證...
- **佈局設計** Auto Placement&Route(APR), HandCraft, Using Hard IP
- **佈局驗證** DRC/ERC, LVS, ...
- **佈局後模擬** LPE， Delay Calculation , Backannotation
- **可靠性分析** Electron-migration, ESD, Substrate coupling ...

從規格到成形-積體電路的設計之路

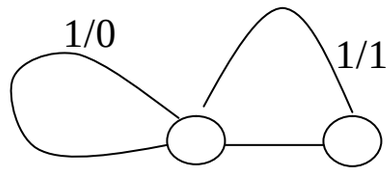


規格與需求



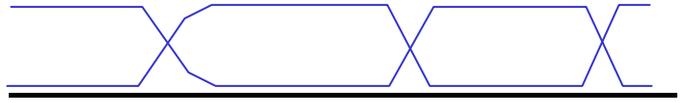


功能

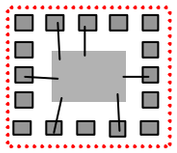


In	Out
0 0 1 0	1 x
1 0 1 1	0 1

時序



面積



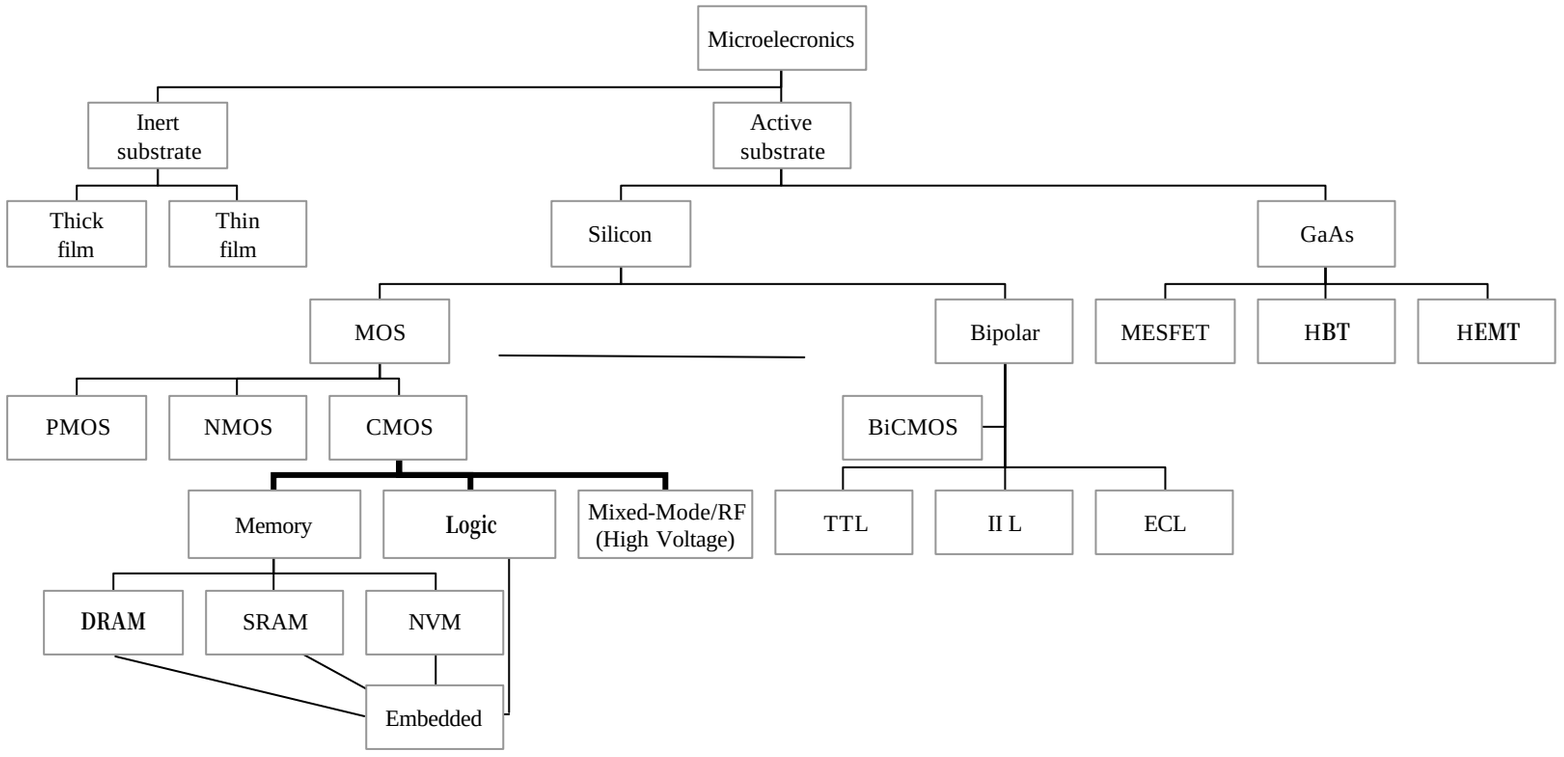
2500um X 2500um

功率

200mW

訊號雜訊比 50dB

Major IC Process



*Thin-film and thick film techniques are used primarily for constructing passive networks such as resistor ladders, filters, attenuators, and phase-shift networks. Such ICs are manufactured by depositing resistive and conductive materials through a series of masking steps, on a nonconducting base such as ceramic. The component tolerances can be made closer than equivalent components made by the monolithic method.

Electrical Characteristic of Process

0.18um CMOS 1.8/3.3V : MOS multiple Vt

- **Metal Capacitor(Q > 50 for C=0.9pF @2.4GHz)**
- **Spiral Inductor(Q >7 for L=4nH @2.4GHz, Al)**
- **High Resistor(1050 Ohm/sq)**
- **Varactor(VCC > 45%/V)**
- **Triple Well(suppress sub. noise coupling > 25dB)**
- **f_t /f_{max} = 62/37 GHz(N) 23/20GHz(P)**
- **Support scalable 1/f noise model**

0.18um SiGe

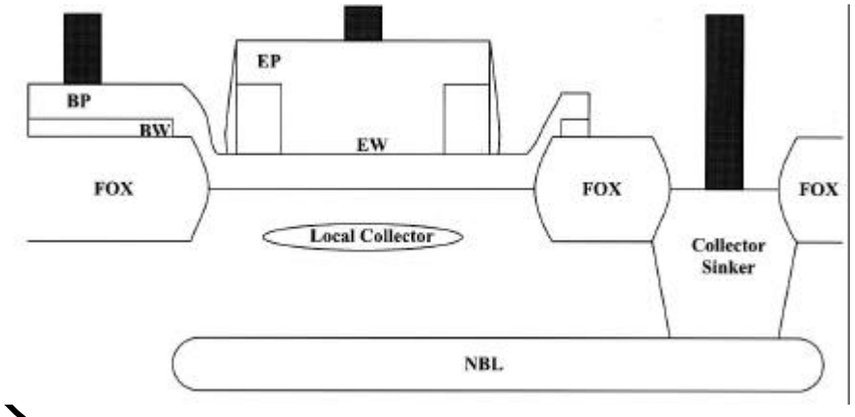
- **High Resistor(1050 Ohm/sq)**
- **Beta: 350**
- **Spiral Inductor**
- **3 Types of NPN available- Hi Speed, Std., and Hi Volt.**
  - **f_t = (120, 65, 35)GHz**
  - **f_{max} = (120, 90, 60)GHz**
- **MiM Capactor**

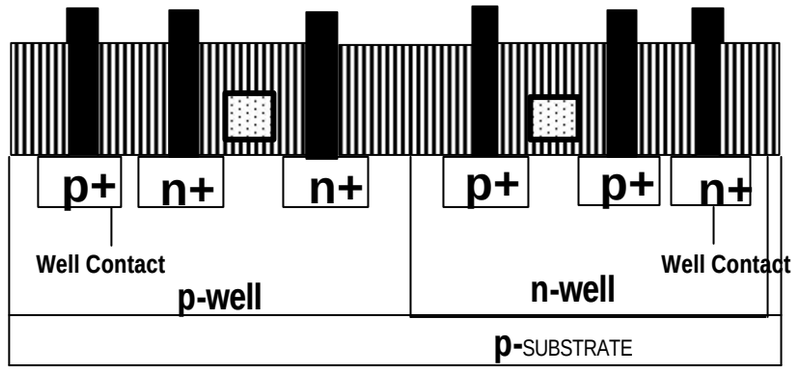
CMOS Technology

High Performance	MOS Speed/delay	Bipolar Bandwidth	Multi-Vt	BiCMOS(SiGe) Mismatch
Driving Capability	Lower	Higher		
Operating Voltage	Low voltage/Low threshold SOI		High voltage	
Signal Integrity	Digital	Mixed Mode	Analog High precision resistor and capacitor	
Design Style	Available cell library	or Full-custom		
Chip Complexity	Many Interconnect layers			
	Embedded Memory			

Comparison between Devices

BJT	MOS
I-controlled device	V-controlled device
Minority carrier device (positive Temperature coefficient)	Majority carrier device (negative Temperature coefficient)
More complicated structure	Simpler structure and smaller area (low power)





製程技術資料

於選定製程後，需取得製程之相關技術資料以利設計之進行，設計資料包含：

1. **製程特性參數**

   各層厚度、深度、*Doping* ***濃度***

2. **製程電性參數**

   元件特性參數(device spice model)

3. Design Rule

   佈局規則，可靠度設計參考規範

4. **設計環境檔**

   使用層次及顏色特性定義、佈局驗證命令檔

Process Design Manual

- **Brief process flow**
- **Masking layers and bias**
- **Layout design rule(RF design rule)**
- **SPICE model(noise model, RF model)**
- **Design guideline(RF design guideline)**
- **Electrical design rule**
- **Characterization reports-**describe the characteristic of devices
- **PCM(process control monitor) specification**
- **WAT(wafer acceptance test) report**
- **Substrate coupling report**
- **Qualification(Reliability)-**describe test methods and the conclusion of testing

Cell Based 設計流程

Behavioral Level

RTL Level

Logic Synthesis

Logic Level Design

Circuit Level Design

Layout Level Design

Post Verification

SPW

BONeS

Visual Architect

Verilog-XL,Polaris,VSS

HDL Debugger

CELL LIBRARY
Compass(0.6um/0.35um)
Smart model

Design-Compiler,ViewSynthesis

Verilog-XL

Timemill,star-time
star-sim

Cell-Based IC

Silicon Ensemble/Dracula,
Apollo/Hercules

RC Extraction(Star-RC, Dracula)

GDSII

Full Custom 設計流程

Behavioral Level

RTL Level

Logic Synthesis

Logic Level Design

Circuit Level Design

Layout Level Design

Post Verification

Composer

SPICE

Virtuoso/Diva, Dracula

RC Extraction(Star-RC, Dracula)

GDSII

Mixed Signal 設計流程

Behavioral Level

RTL Level

Logic Synthesis

Logic Level Design

Circuit Level Design

Layout Level Design

Post Verification

SPW

BONeS

Visual Architect

Verilog

HDL Debugger

Synopsys

CELL LIBRARY
Compass(0.6um/0.35um)
Smart model

Verilog

SPICE

Full-Custom IC

Cell-Based IC

Virtuoso/Dracula

Silicon Ensemble/Dracula,
Apollo/Hercules

RC Extraction(Star-RC, Dracula)

GDSII

電路設計

- **依設計規格選擇適當架構**

  Modularity/Regularity

  Parallel v.s Sequential

  Differential signal / Single-ended signal
- **依架構決定元件之組合**

  Current mirror type / Compensation type
- **依交、直流參數決定適當之電晶體大小及偏壓點**(Channel Width/Length)
- **依環境決定負載型態及負載值**
- Language based design v.s graphical design

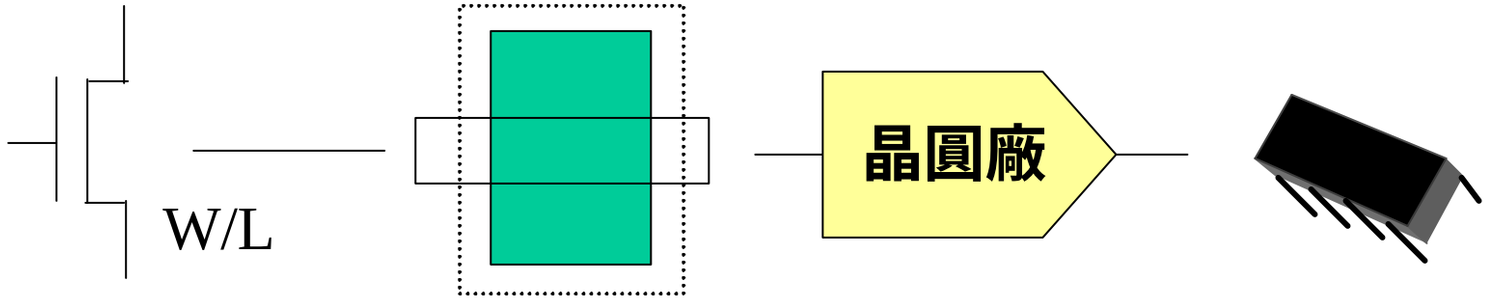
電路模擬

電路模擬的用途：

- **依所給定的元件模型驗證所設計電路的功能及規格。**
- **提供電路架構參數修改的依據。**
- **依模擬所得之結果以供決定佈局之原則如電源線寬度，** Buffer **數量等。**
- **依製程參數的變異以訂定電路工作的區間及限制。**
- **驗證環境因數變化對電路工作特性的影響。**

佈局設計

- 電路設計及模擬的驗證決定電路的組成及相關參數，但仍不是實體的成品
- 積體電路的實際成品需經晶圓廠製作
- 設計者需提供積體電路製作的實體描述稱為佈局
- 佈局設計將所設計的電路轉換為電路製作的圖形描述格式
- Layout Editor 或 P&R 工具提供佈局設計的環境

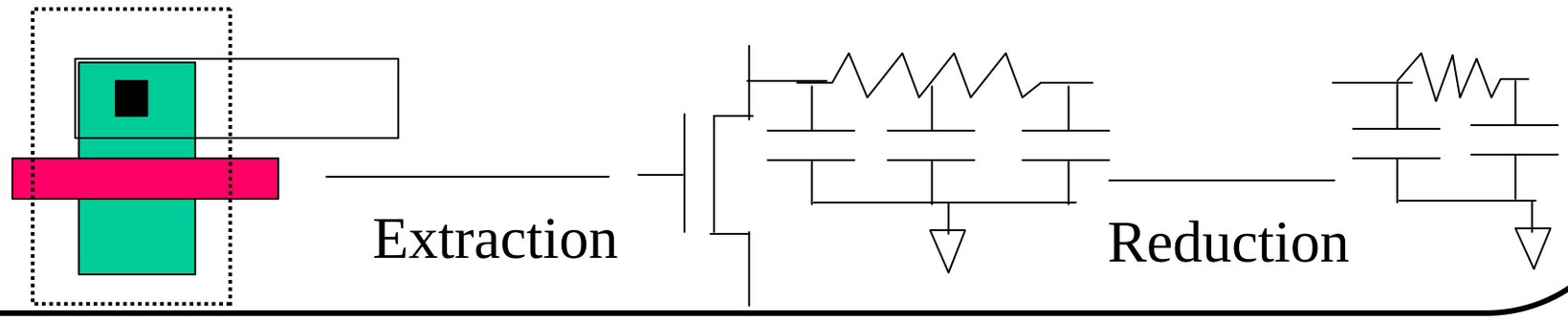


佈局驗證

- **每一製程均有其設備上及控制上的極限，如光解析度、化學藥品濃度劑量、作用時間、溫度等，因此在佈局上需能容忍製程變異的發生。**
- **為讓晶片製作過程的合理變動不致影響製作的結果，電路設計者所設計的電路佈局必需滿足晶圓廠所提供的佈局規範。**(Design Rule Check)
- **電路設計及佈局設計為不同階段的獨立設計過程，必須確保佈局設計及原電路的一致性。**(Layout v.s. Schematic)
- Diva, Dracula,Calibre **及**Hercules **等軟體提供佈局驗證的功能**

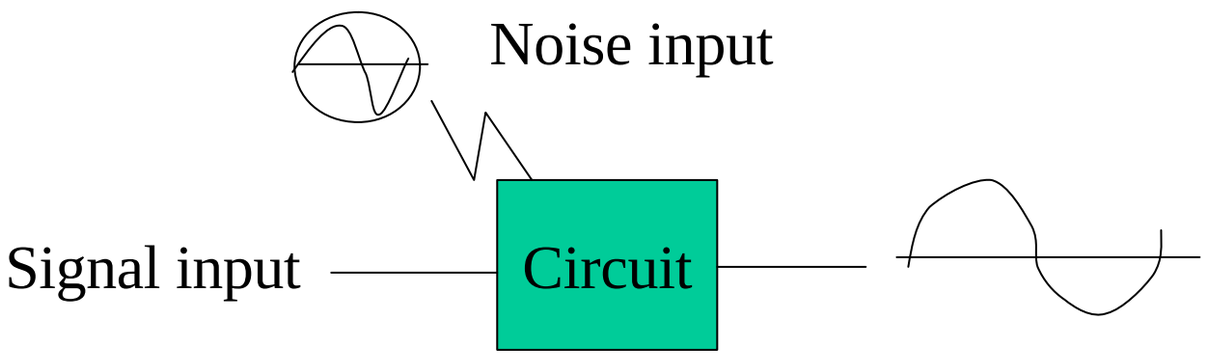
佈局後電路模擬

- 實際的訊號線具有阻抗及負載，對原電路將造成特性上的改變，完整設計應考量訊號線的負載延遲效應。
- 準確的連接線模型方可促成正確的模擬結果。
- 完整的連接線負載包含龐大數量的雜散元件，完整的模擬將增加所需的時間，device reduction 為必須的考量。
- 佈局後模擬包含 電路及雜散元件萃取 + 電路模擬等兩項步驟。



雜訊分析

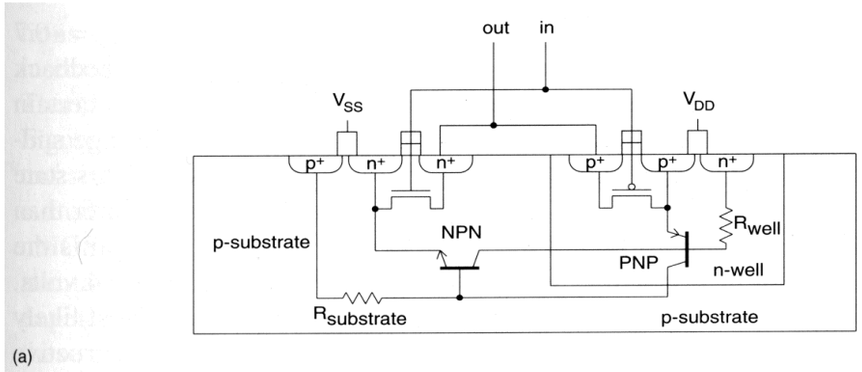
- **由於積體電路中訊號線的距離相當接近，一條訊號線上的訊號轉態會干擾鄰近訊號線的位準**(Interconnect coupling)。
- **由於積體電路的所有元件均位於同一基底上，因此，雜訊可能會透過基底干擾其他電路的運作**(Substrate coupling)。
- **由於電路的電源訊號係由金屬線連至晶片各處，金屬線上的雜散電感值將使電流變化轉換為電壓降產生雜訊影響電路的運作**(IR drop induced power/ground bounce)

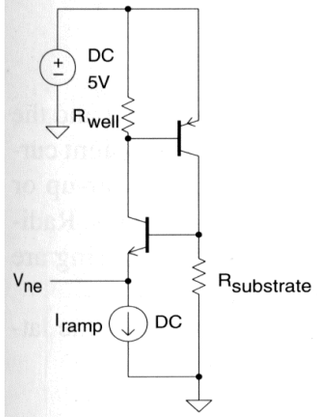


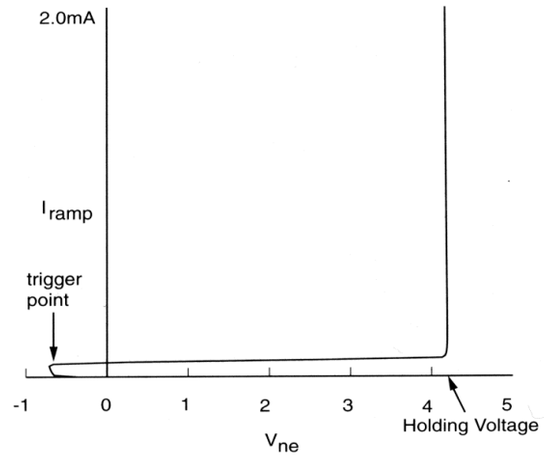
可靠性分析

- **在一般的模擬中僅驗證電路的功能及特性，但電路的電晶體及連接線在正常的工作下會出現老化等性能衰退現象，將影響積體電路工作的壽命。**
- **影響電晶體工作特性的因素** : antenna effect, hot-carrier effect, oxide breakdown**等**
- **影響連接線特性的效應**: electron migration, stress migration
- **可靠性分析需要建立元件或連接線的老化模型**
- **提供足夠的工作區間為提升電路可靠度的最直接方法。**

Latchup防止







- **CMOS電路中的寄生SCR結構運作導致原設計電路無法正常工作。**

Latchup **的防制**

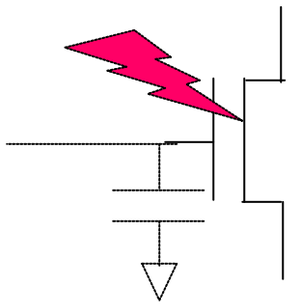
- **製程的加強**
- Layout **的技巧**

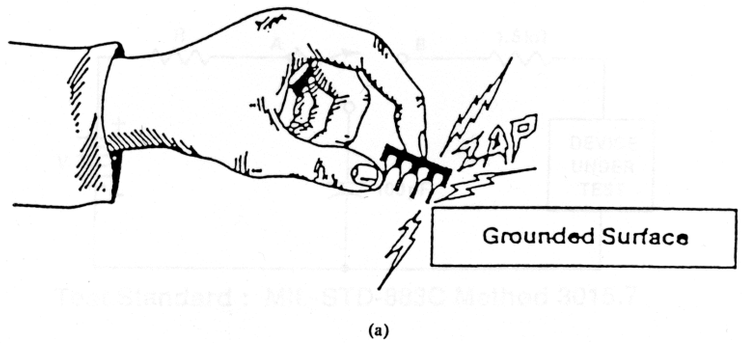
  降低寄生BJT **的** β **值**

  降低 well/substrate **電阻值**

  Guard ring for collecting carrier

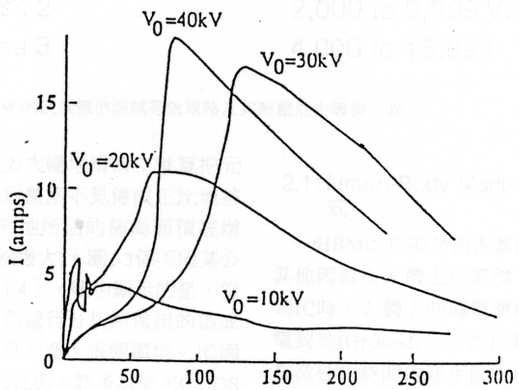
ESD(Electrostatic Discharge)





(a)

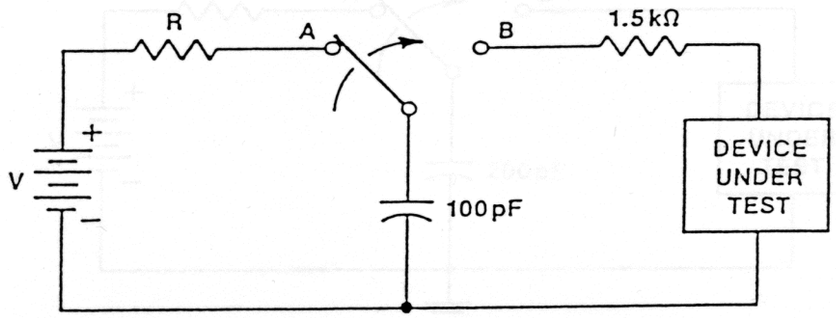
ESD CURRENTS



(a) HBM的ESD發生情形；(b) 在不同HBM靜電電壓下，其靜電放電之電流與時間的關係

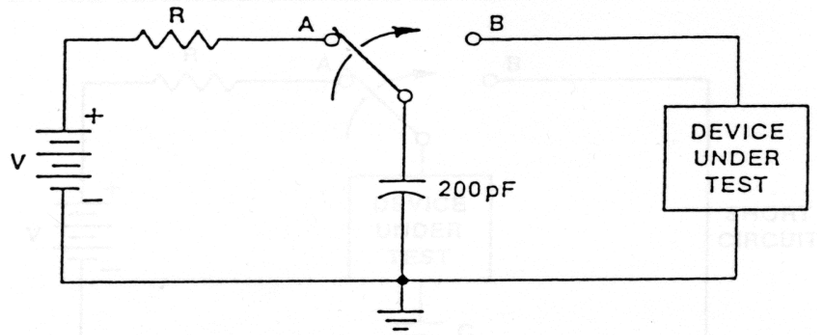
(1) Human-Body Mode
(2) Machine Mode
(3) Charged-Device Mode
(4) Field-Induced Mode

The Specification of ESD



Test Standard : MIL-STD-883C Method 3015.7

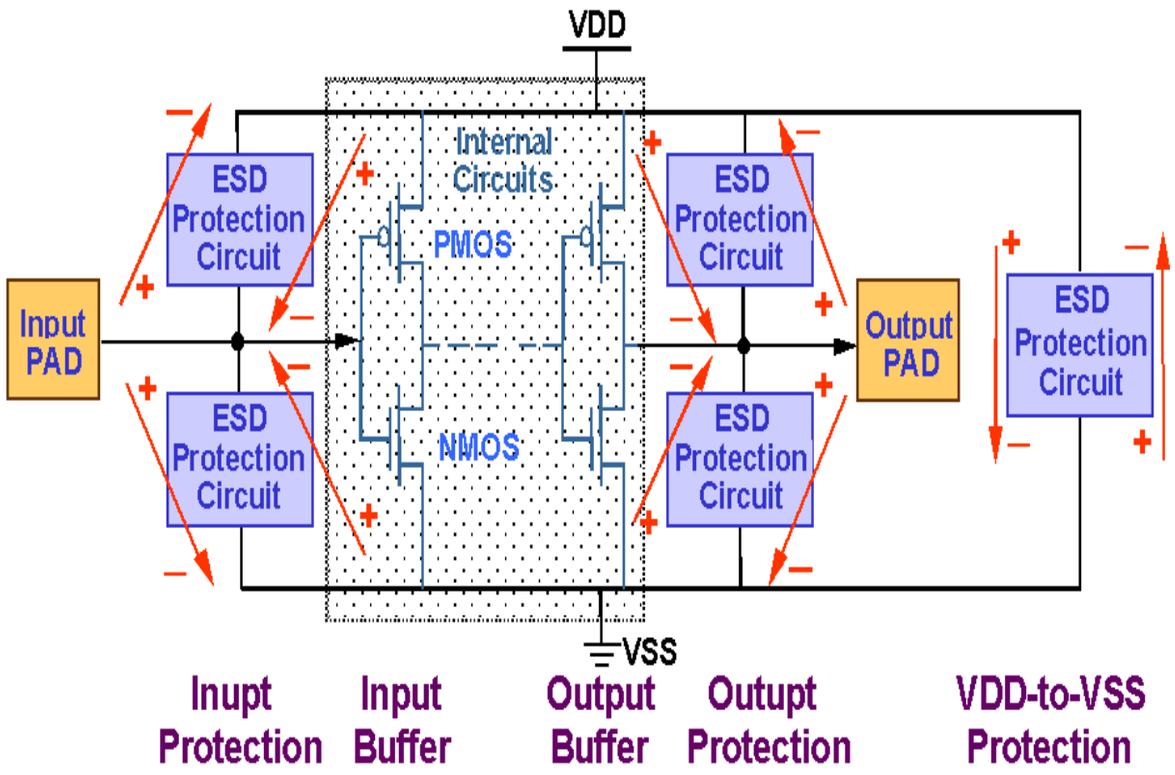
CLASSIFICATION	Sensitivity
Class 1	0 to 1,999 Volts
Class 2	2,000 to 3,999 Volts
Class 3	4,000 to 15,999 Volts



Test Standard : EIAJ-IC-121 Method 20.

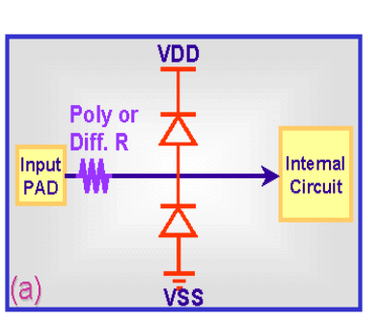
CLASS	STRESS LEVELS
M0	0 TO <50 V
M1	50 TO < 100 V
M2	100 TO <200 V
M3	200 TO <400 V
M4	400 TO <800 V
M5	>800 V

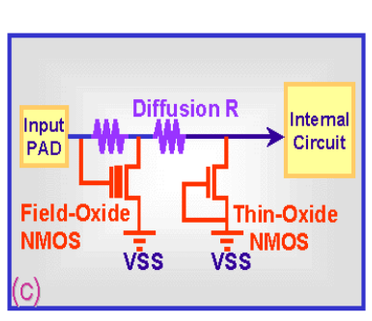
Whole Chip ESD Protection

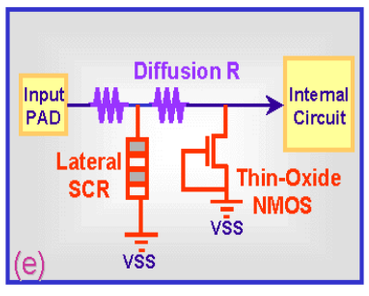


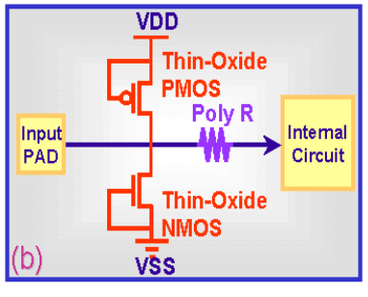
Whole-Chip ESD Protection

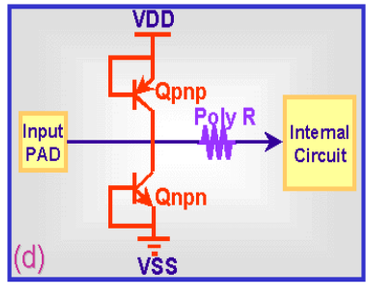
Commonly-Used On Chip ESD Protection Circuits for Input Pad in CMOS IC











晶片封裝選擇

- **封裝提供晶片保護、散熱及系統連接等功能**
- **封裝選擇的考量**
  - Pin count
  - Die cavity
  - **散熱效益**
  - **系統整合之便利性** - Through hole / surface-mount /Chip-scale package
  - **良好的電氣特性**
  - Debugging capability

Full-Custom 設計系統環境

完整的Full-Custom **設計環境包含**

- **設計資料庫** - Cadence Design Framework II
- **電路編輯環境** - Text editor / Schematic editor
- **電路模擬軟體** - SPICE
- **佈局編輯軟體** - Cadence virtuoso, Laker
- **佈局驗證軟體** - Diva, Dracula, Calibre, Hercules

系統環境

- **工作站**
- unix-based **作業系統**

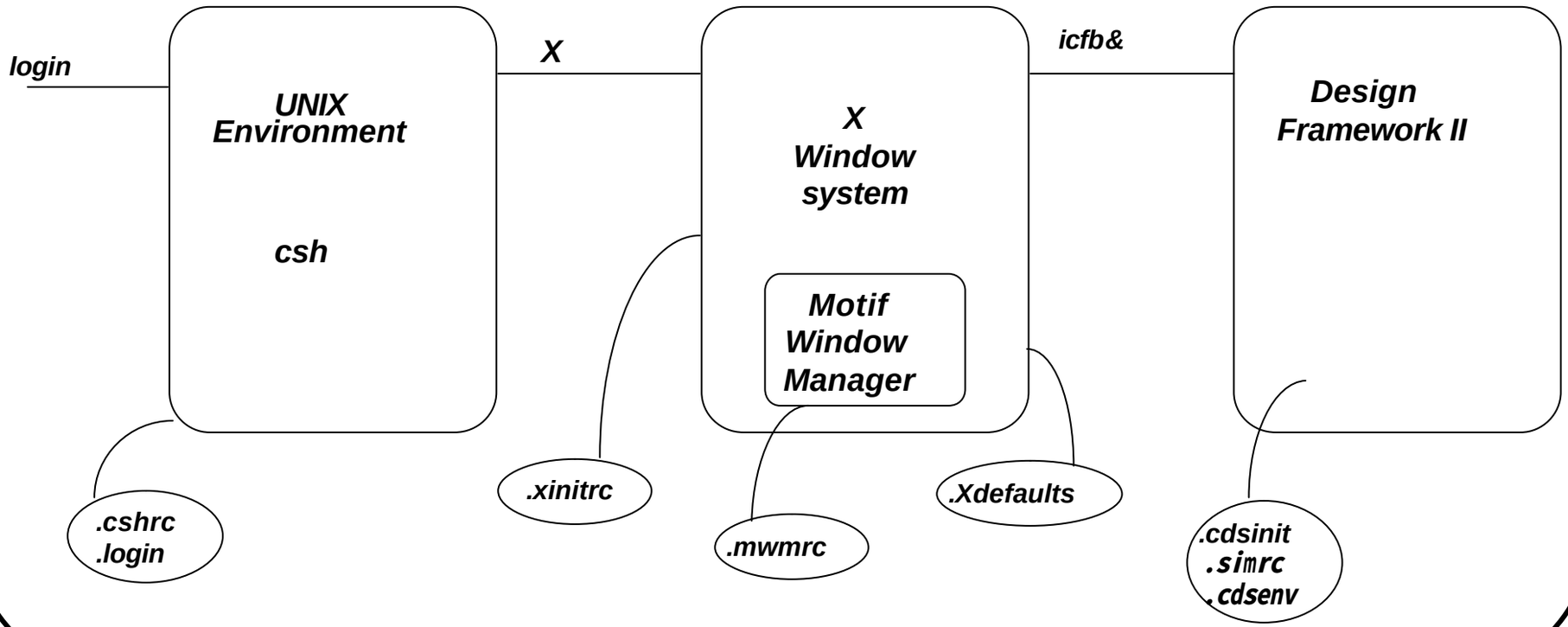
工作站系統與視窗環境

- .login **簽入環境設定檔**
- .cshrc Shell **環境設定執行檔**
- .xinitrc **視窗系統設定檔**
  - .mwmrc motif **視窗環境設定檔**
  - .openwin-init openwin**視窗環境設定**
- .logout **簽出命令批次檔**
- .cdsinit Cadence **環境設定檔**
- cds.lib Cadence **環境資料庫路徑設定檔**
- .simrc Cadence **模擬驗證環境設定檔**
- display.drf Cadence Layout editor **顏色圖樣設定檔**

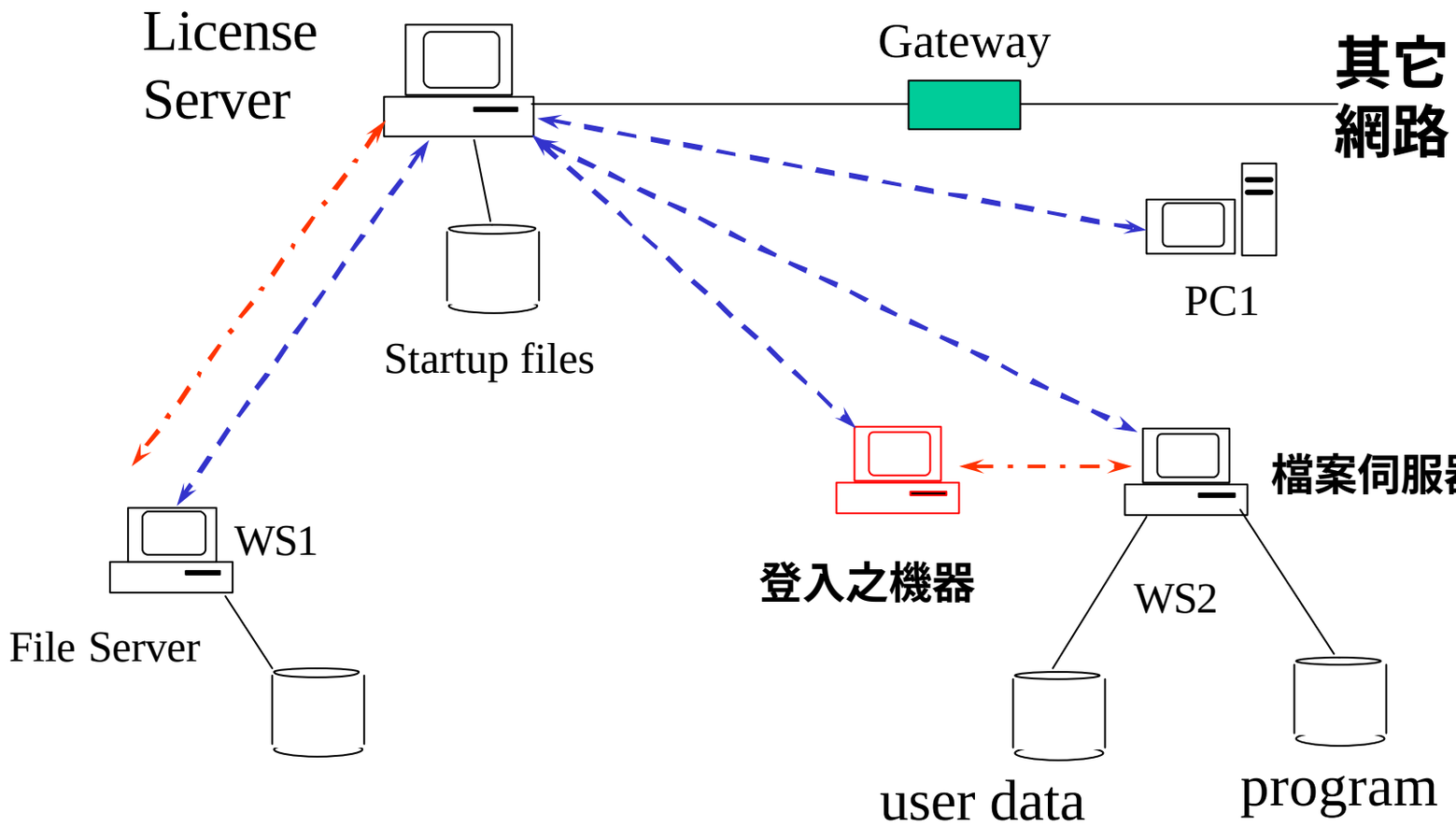
工作站使用環境

Workstation **環境設定檔**

環境設定檔圖解

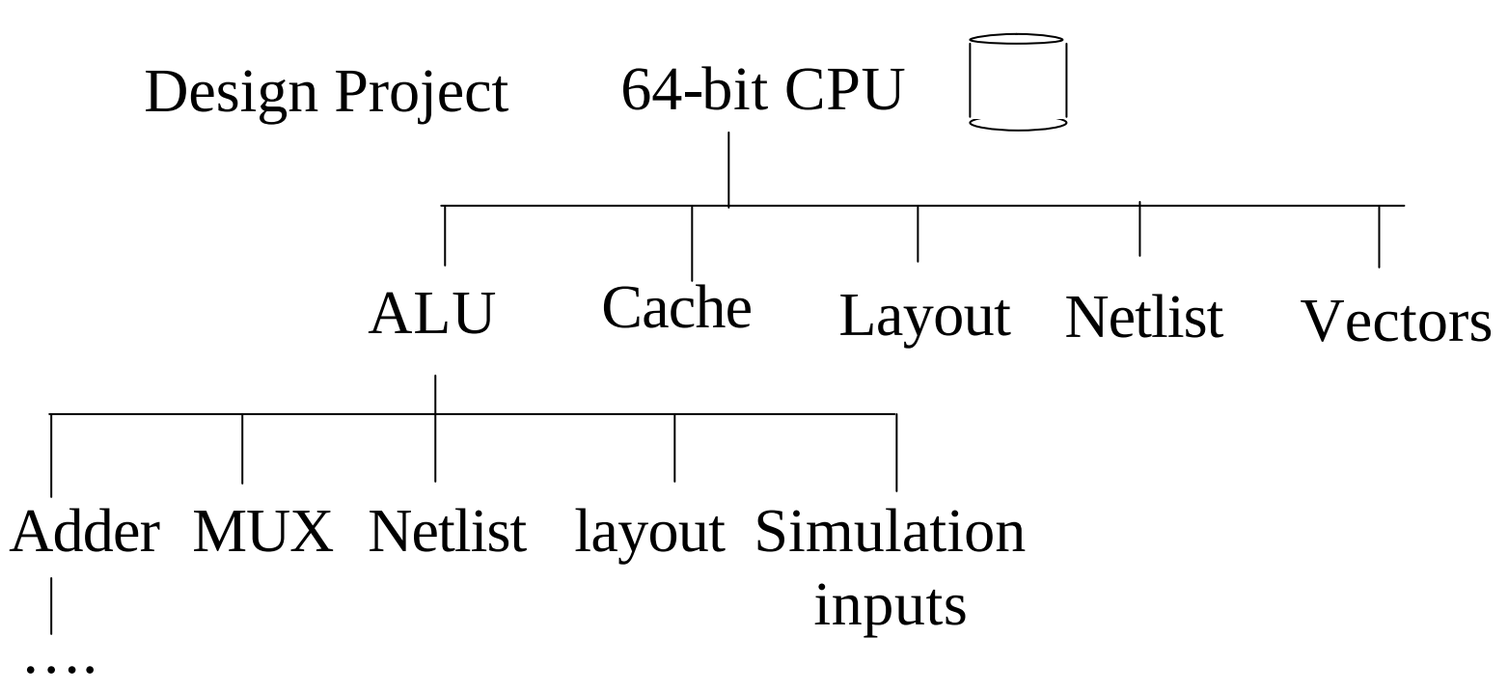


工作站軟體環境

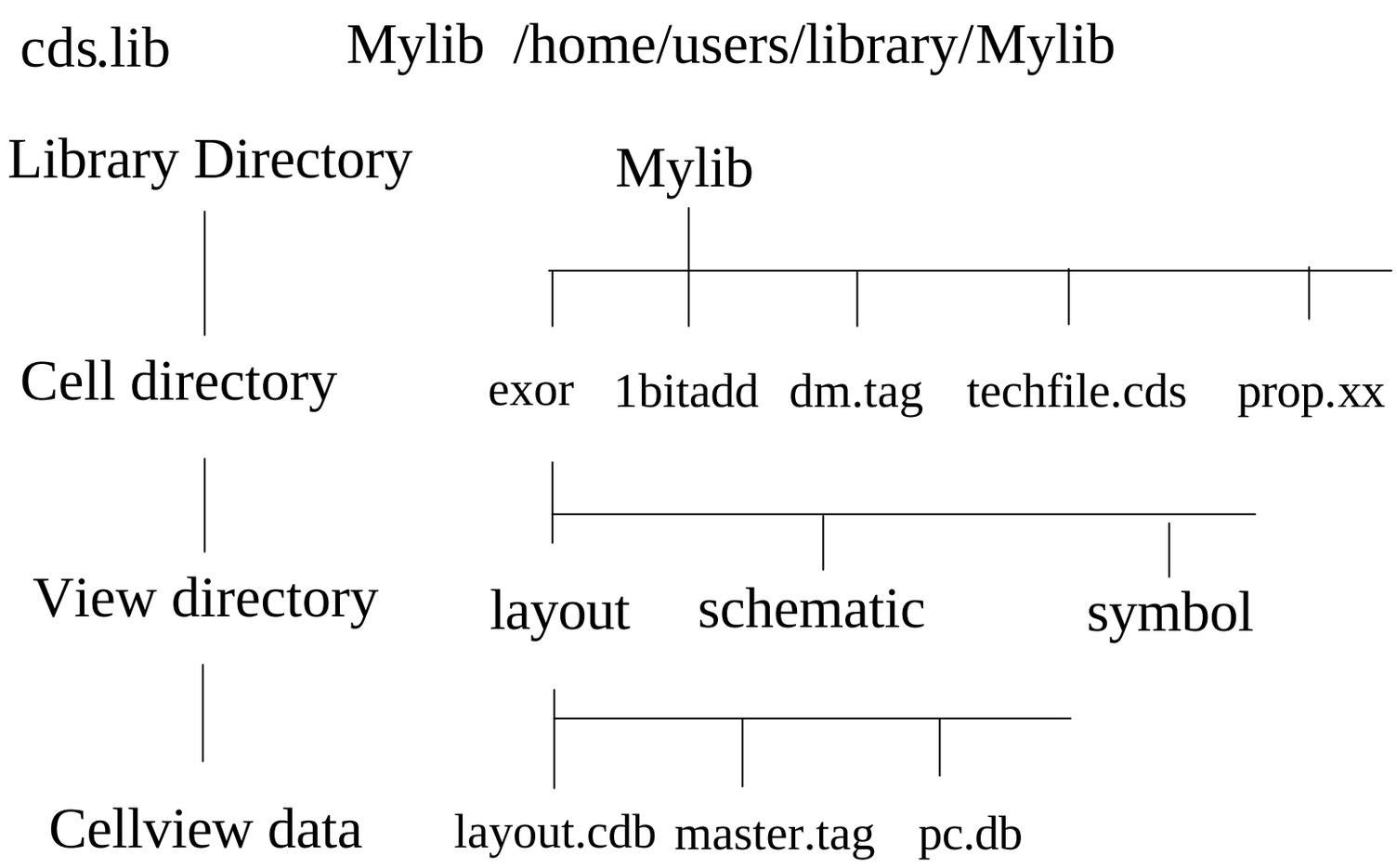

License
Server
Gateway
其它
網路
PC1
Startup files
WS1
File Server
登入之機器
檔案伺服器
WS2
user data
program

設計資料庫

- 階層化資料
- 版本控制
- 存取控制
- 經常備份



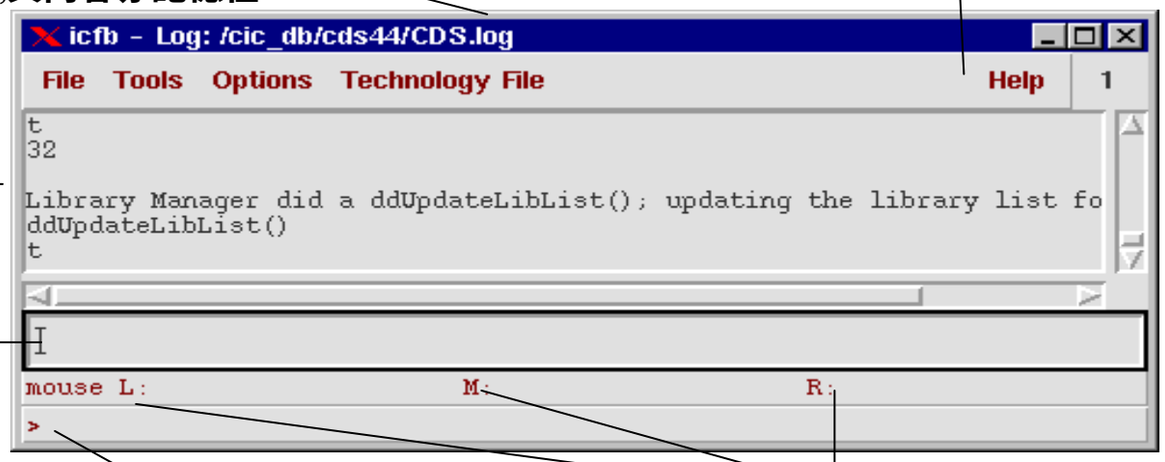
設計資料庫相關檔案


cds.lib
Mylib /home/users/library/Mylib
Library Directory
Mylib
Cell directory
exor
1bitadd
dm.tag
techfile.cds
prop.xx
View directory
layout
schematic
symbol
Cellview data
layout.cdb
master.tag
pc.db

Cadence中常見表格

CIW(Command Interpreter Window)

以 skill format display user 的 command 與system 的 response ,其內容亦記憶在file



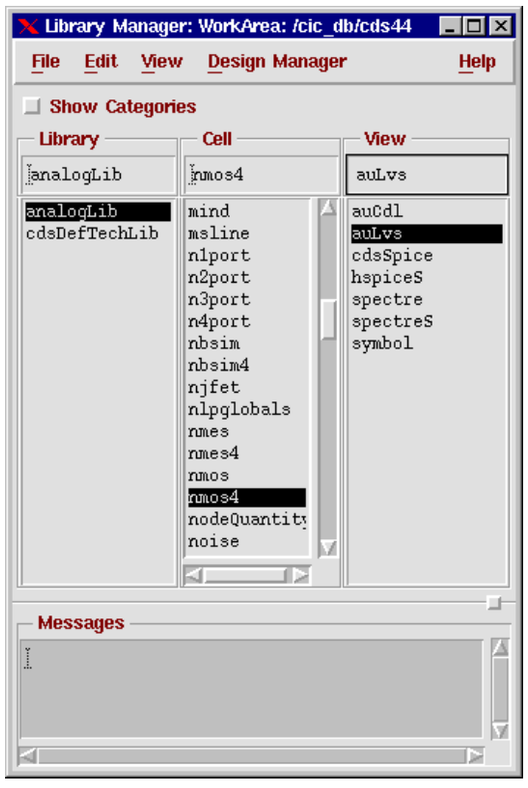
on-line help

提示動作

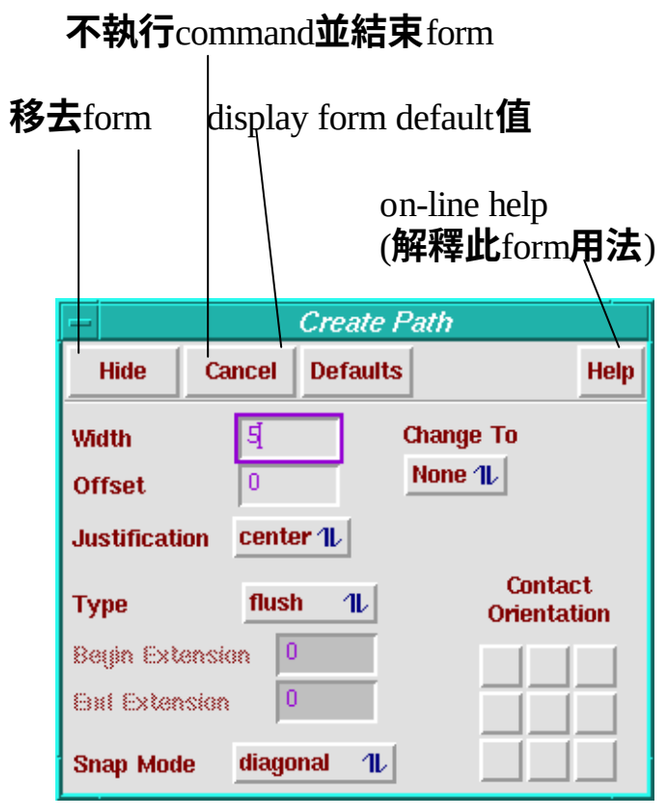
提示 mouse key 功能

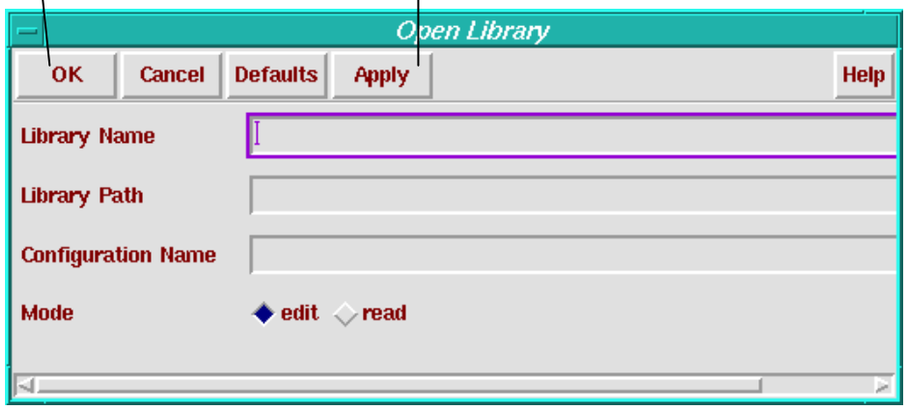
下skill command 處

Library Manager

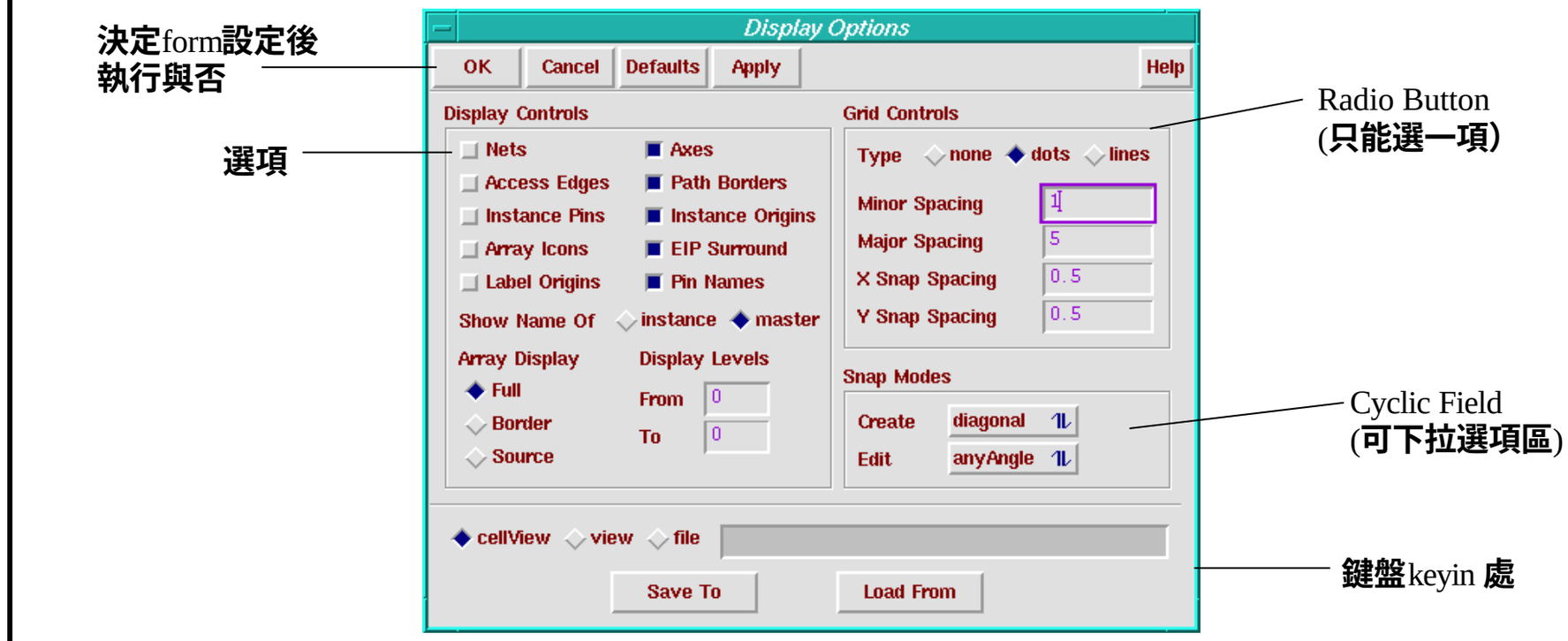


顯示的library為cds.lib中所定義 ,這些library可被expand成cell與 cellview而 read或edit





Enter key**功能同**OK,Escape key**功能同**Cancel
^a**跳**cursor**到最前**,^e**跳**cursor**到最後**

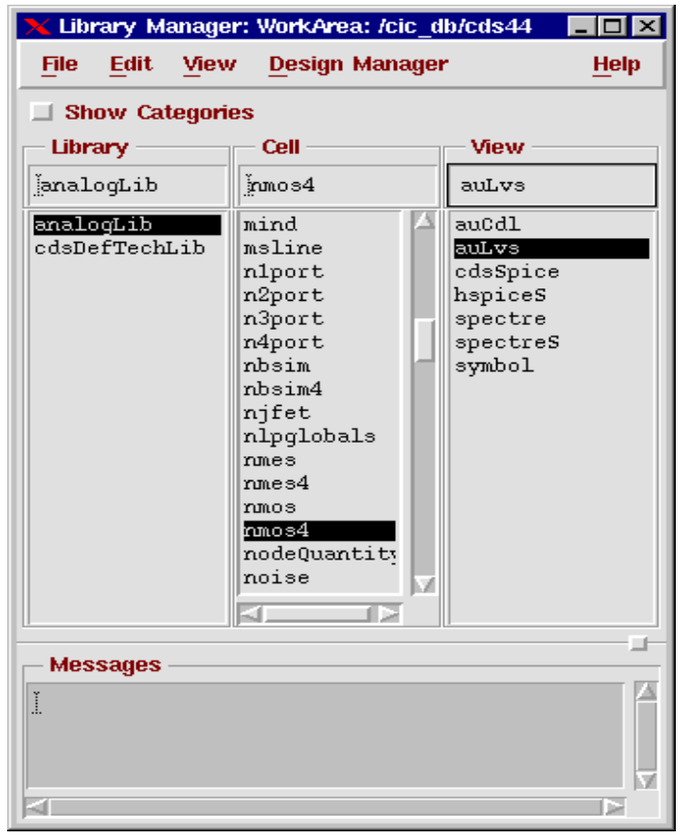


Cyclic Field**功能同**Radio Buttons,**只能選一項**

Library Manager

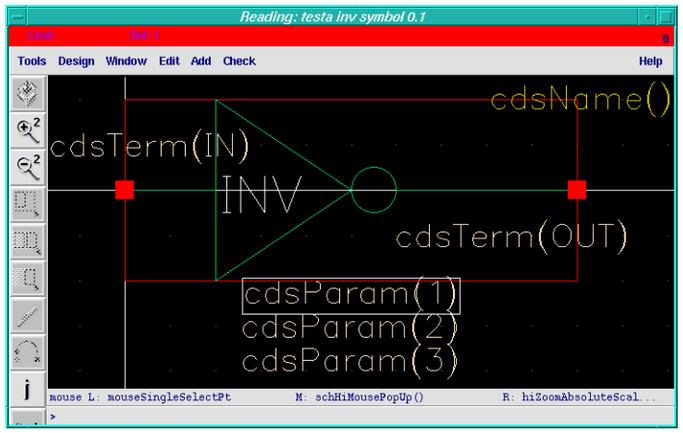
cds.lib**中定義了**library**與所在的找到的** library, **方便** user access design data, **包括**create new library, cell**與**cell view, open**或** read cellview**等**, **其結構 如下**

Library—(Category)— cell— cellview

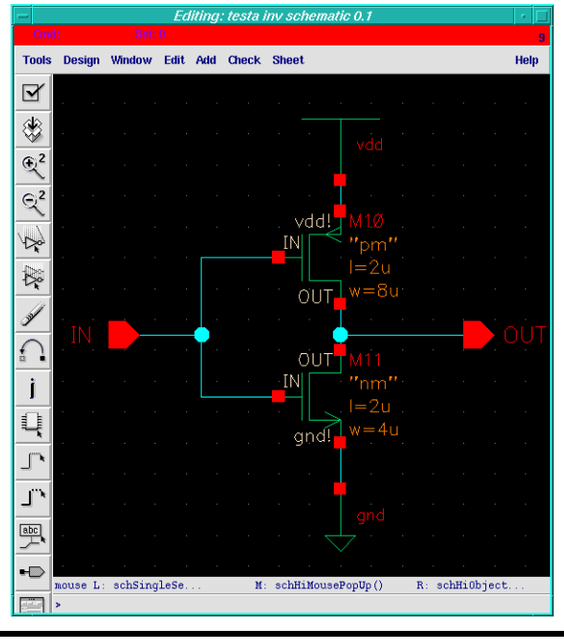


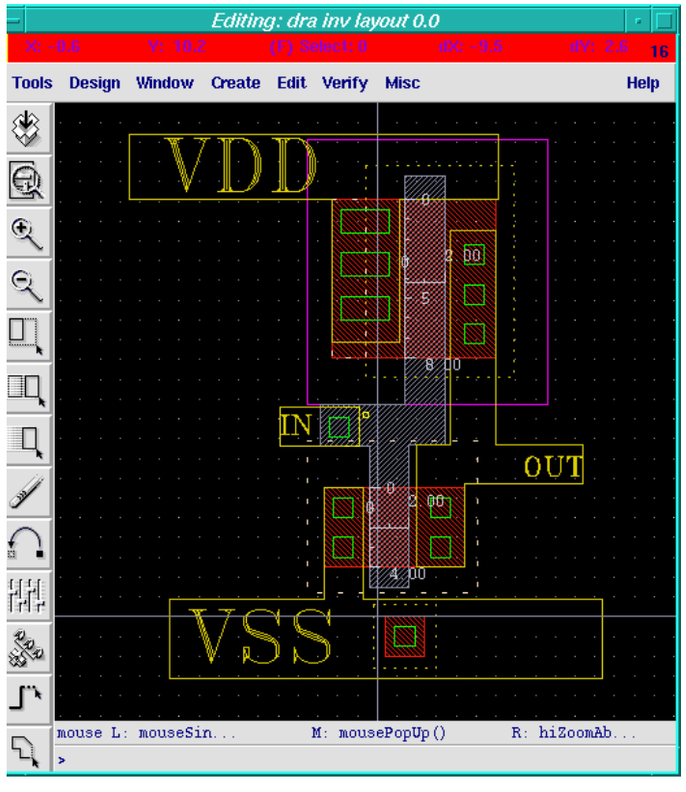
view name 與 view type 定義於 technology file, 使得不同 view name 的 window 中選項也有 所不同, 如下圖 cell inv 的三種 cellview

symbol view



schematic view





layout view

Layer Selection Window (LSW)

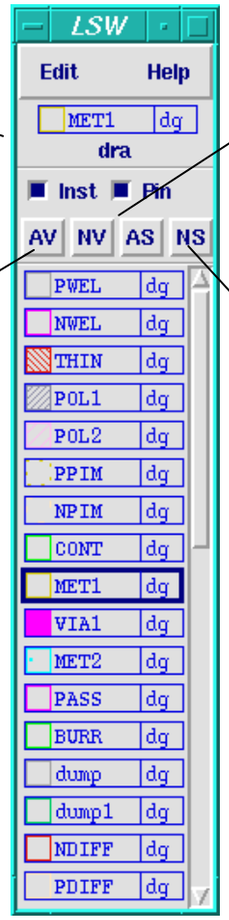
目前所選之層

library name

set Instance, pin **可選否**

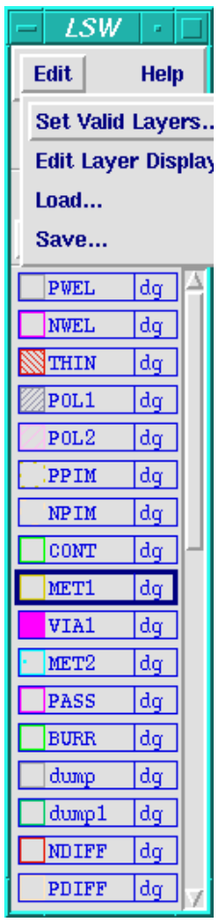
所有層均 show **於** layout cellview **中**

除目前所選之層外在 layout cellview **中均不顯示**



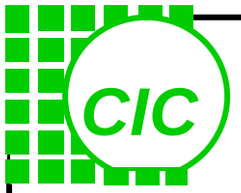
LSW**上所有層在**layout cellview **中均可 選**

除目前所選之層外在layout cellview **中均不可 選**



設 show**於** CIW**上之層**

設LSW**上所選層顏色等**



Technology file : This file is a large data file that specifies all of the technology-dependent parameters associated with that particular library. Design rules, symbolic device definitions, and parasitic values are some of the technology-specific parameters common to all cells in a library

Technology file library: One can share technology file information between different libraries. Use Technology File -> Attach To... **command to attach the design library to the technology file library. A technology file library contains process technology, design rules, and symbolic devices.**

techfile.cds : The techfile.cds file contains the binary technology file

abgen.rul: A ASCII file to generate Abstract view

display.drf containing layer display information. The software reads all the display resource files, including
$ CDS_INST_DIR/share/cdssetup/dfII/default.drf
$ CDS_INST_DIR/tools/dfII/local/display.drf
$CDS_PROJECT/display.drf (if you are working in a TDM environment)
$HOME/display.drf
./display.drf

cds.lib is a file containing library definition. The software reads the first cds.lib defined in the search path file <install_dir>/share/cdssetup/set.loc.
The following is a sample listing of the search path:
./cds.lib
$CDS_WORKAREA/cds.lib
$HOME/cds.lib
$CDS_PROJECT/cds.lib & $CDS_SITE/cds.lib (if you are working in a TDM environment)
$ CDS_INST_DIR/share/cds.lib
(If no cds.lib has been found, a new search path is established from the file $CDS_SITE/cdssetup/setup.loc.)

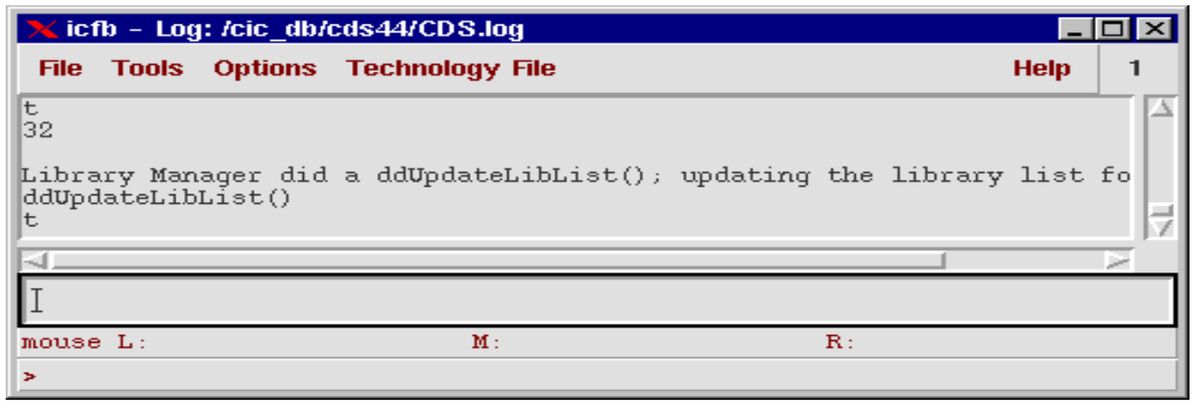
Create library

在選定好製程及相關的 technology file **後**, **再在** OPUS **內** create library,**即可將** schematic, symbol, layout ... **等** view**全建在固定** library **內**

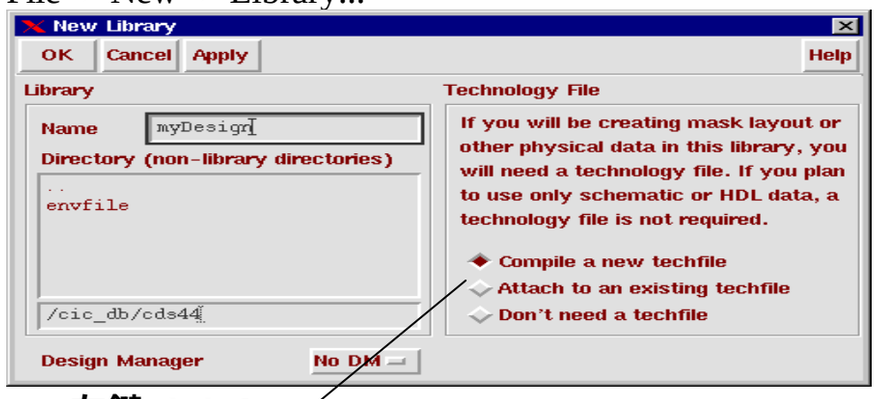
進入 OPUS

% ic fb&

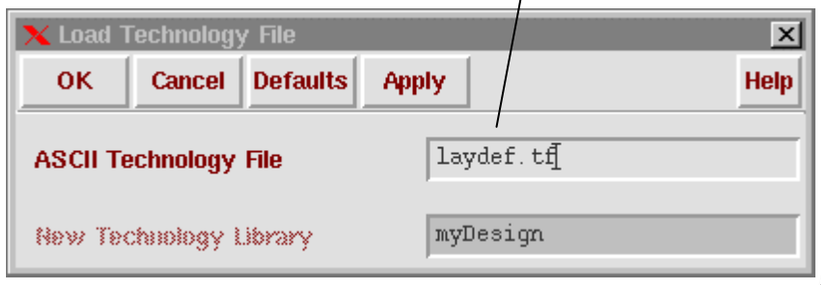
出現 Command Interpreter Window(CIW)



File -> New -> Library...



左鍵 click OK

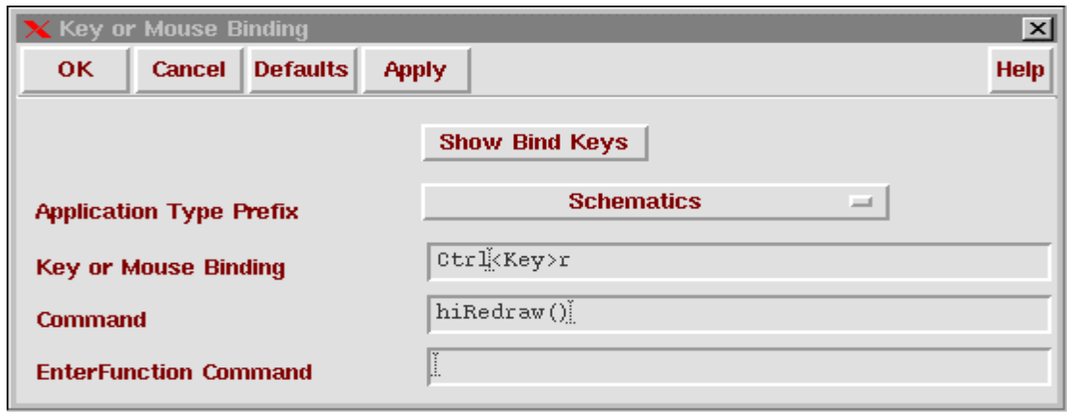


左鍵 click OK

Defining bindkeys

@**為何要設定** user Bindkeys?

對於經常使用的 command, **為了避免多重滑鼠按鍵選擇的麻煩**, **可在鍵盤上設定單一按鍵**,**雙鍵或滑鼠代替抹某一**command,
方便使用者選擇所要下的
Command.



@**設定** user bindkeys **的方式**:
(1)**在** CIW **視窗裡定義**:

選擇 Options -> Bindkey **後**, **將會出現** CIW BindKey Form (**如上圖所示**)
鍵入所要的按鍵及命令.

Application Type Prefix: **可選擇** Command Interpreter, layout, Schematics, etc.
Show Bind Keys : **可顯示出已經定義之** Bind Keys.
Key or Mouse Binding: **單鍵或雙鍵或滑鼠按鍵**, **格式如下圖所示**:
單鍵: <Key>a, <Key>F1.
雙鍵: Shift<Key>b, Ctrl<Key>c, Meta<Key>d.
滑鼠: None<Btn1Down>
Command: **可鍵入使用者所欲定義之** command, **如**: dmbOpenLibBrowser().
EnterFunction Command: **一般不需鍵入若使用者有興趣可參考** Show Bind Keys **中已經定義好之** EnterFunction Command.

Defining bindkeys in SKILL

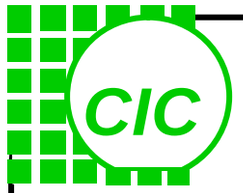
1. **使用** Skill **語言定義** Bindkey **格式如下圖所示：**

hiSetBindKey(“Layout” “<Key>c” “leHiCopy()”)
(**上列即定義當**cursor**在**Layout window**上時按鍵**c**等同為用**copy command,
可用 hiGetBindKey(“Layout” “<Key>c”)**以知是否鍵**c **已定義**)

hiSetBindKey(“Symbol” “<Key>i” “schHiCloneSymbol()”)
hiSetBindKey(“Schematics” “<Key>c” “schHiCopy ()”)

2. **將以上內容存檔,如** :userBindKeys.il, **直接在** CIW**視窗下**
load(“userBindKeys.il”) **命令。**

3.**另外可將已經定義好之** Bindkey **指令的 檔案加在** .cdsinit **檔案中，**
則每次執行時便會自動將 user **定義之** Bindkey **自動載入。**



電路架構建立與模擬

- 電路設計的基本觀念
- 電路模擬與規格驗證
- 電路設計環境
- 電路模擬環境

電路設計的依據

- **電路規格**

  1. **電氣規格**
     環境容忍範圍 : VDD,VIN,TA, Tstorage
     靜態規格 : V_{IH}, V_{IL}, I_{IH}, I_{IL}, V_{OH}, V_{OL}, I_{OH}, I_{OL},I_{OUT}
     動態規格 : T_w, f_{clk}, t_r, t_f, t_h, t_{su}, t_{ph} , …
  2. **功能規格**
     Truth table, state diagram, timing diagram, Flow chart
  3. **類比規格** : **頻率響應，** PSRR, CMRR, Offset, Gain error, settling time….
  4. **其他** : ESD, I/O Capacitance,**量測條件**,**接腳對應**,**包裝型態**...

電路架構選擇

依規格透過系統評估決定電路架構

- Parallel v.s Sequential
- Number of Clock phase
- Number of pipeline stage
- Dynamic Logic / Static Logic
- Current mode /Voltage mode operation
- Gain stage type and stage number
- Differential signal v.s. Single ended
- Compensation scheme

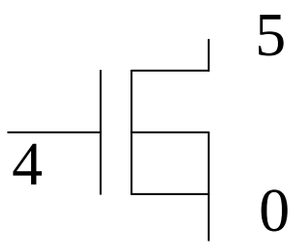
元件參數訂定

- **在全客戶 設計方式中，各元件的值(如電晶體的**W/L**、電容值、電阻值等)均可由設計者依各項規格參數決定。**
- **在直流方面，**KCL, KVL, **及** $I_D = F(VGS, W/L)$
- **在**Timing **方面，** $Tr \approx (L/W)_{driver} * (WL)_{load}$
- **在類比電路中** , gain : $g_m * r_o$ ， $g_m \approx (W/L)^{1/2}$ ， $r_o \approx (L/W)_{load}$
- **依各項關係式預估所需電壓、電流，進而推估所需** W/L
- **參數變動原則**:
  - **選擇主要規格**
  - **決定主要影響元件**
  - **調整主要元件** W/L, **模擬驗證**
  - **調整其他元件參數值**

電路描述建立的途徑

Design Capture

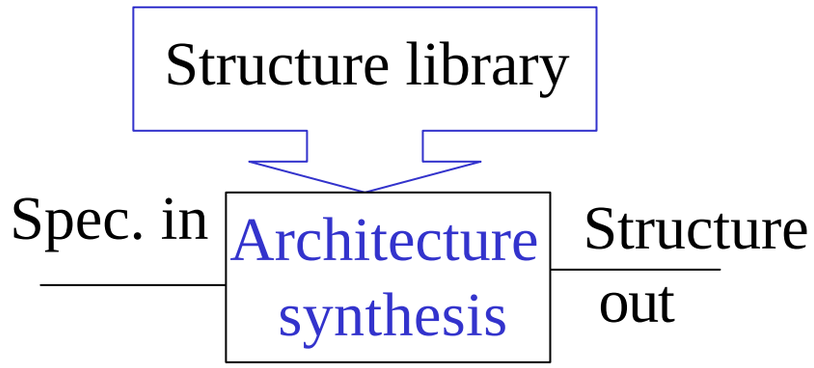
Schematic entry



Text-based netlist entry

M1 5 4 0 0 NMOS ..

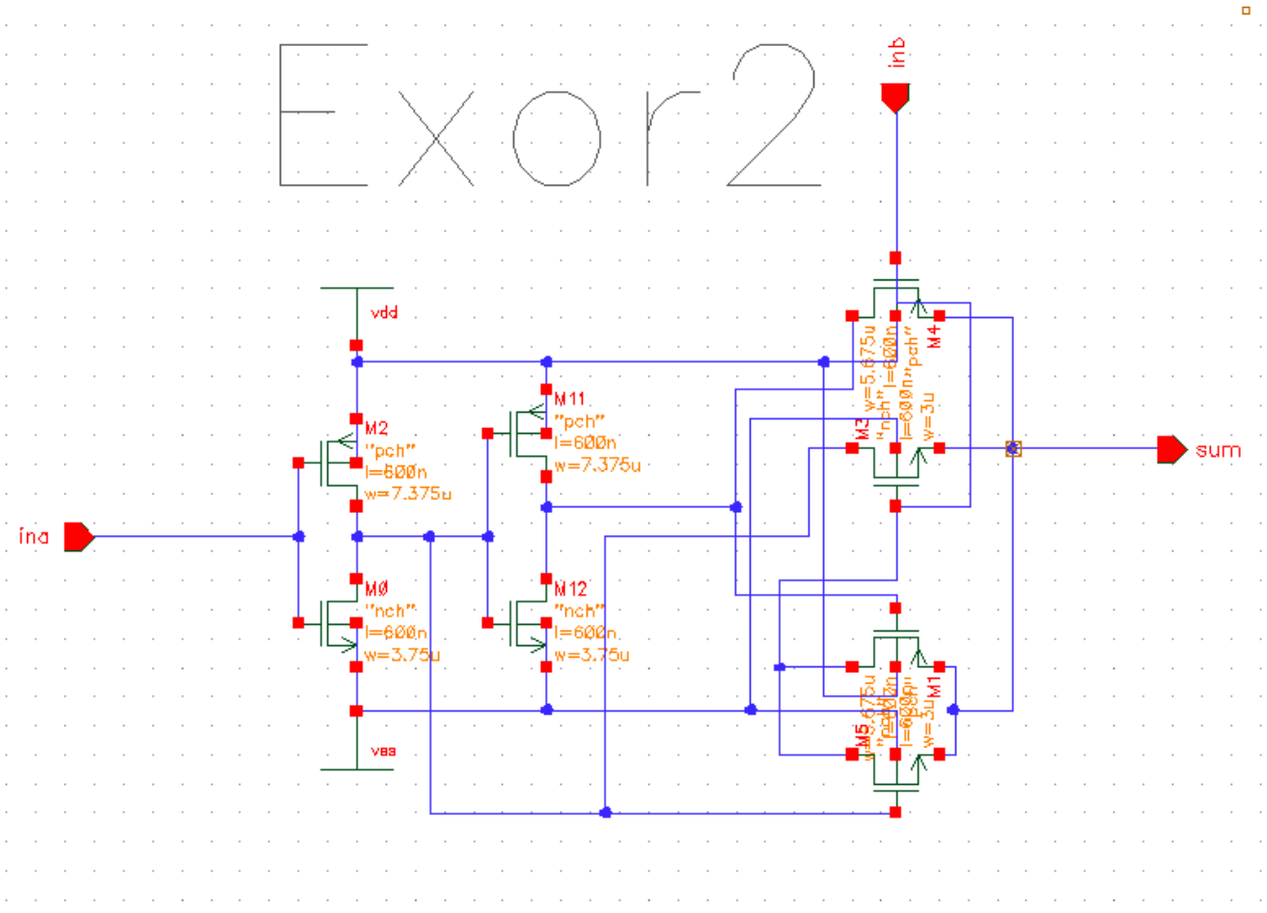
Design Synthesis



SPICE Netlist 範例

```
*Two stage OP design
.lib "umc05.lib" TYP
.options post nomod
.TEMP 27
* Netlist information
M1 3 1 5 0 nmos L=2u W=8u AS=18p AD=18p
+                    PS=18u PD=18u
M2 4 2 5 0 nmos L=2u W=8u AS=18p AD=18p
+                    PS=18u PD=18u
M3 3 3 vdd vdd pmos L=10u W=10u AS=12p AD=12p PS=16u PD=16u
M4 4 3 vdd vdd pmos L=10u W=10u AS=12p AD=12p PS=16u PD=16u
M5 5 vbias vss vss nmos L=2u W=7u AS=49p AD=49p PS=26u PD=26u
M6 vout 4 vdd vdd pmos L=2u W=70u AS=490p AD=490p PS=150u PD=150u
M7 vout vbias vss vss nmos L=2u W=130u AS=930p AD=930p
+                                PS=260u PD=260u
M8 vbias vbias vss vss nmos L=2u W=7u AS=49p AD=49p PS=26u PD=26u

* Feedback CAP
Cc vout 4 0.44pF
Cl vout 0 4pF
Ibias vdd vbias 8.8u

* Voltage sourses
vdd vdd 0 5v
vss vss 0 0v
```

電路圖範例

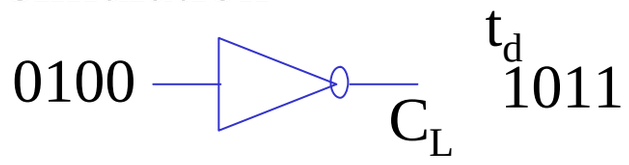

Exor2
inb
vdd
M2
"pch"
l=600n
w=7.375u
M11
"pch"
l=600n
w=7.375u
M4
"pch"
l=600n
w=5.375u
M3
"nch"
l=600n
w=3u
sum
ina
M0
"nch"
l=600n
w=3.75u
M12
"nch"
l=600n
w=3.75u
M5
"nch"
l=600n
w=5.375u
M1
"nch"
l=600n
w=3u
vss

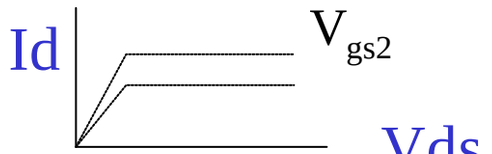
電路模擬與驗證

- 所設計之電路架構其功能及各項特性需經驗證及確認
- 利用成品驗證為最真實之結果，但成本過高
- 透過電路模擬驗證可降低驗證的成本及時間
- 電路模擬為依據一預設之 元件模型 透過 程式 依照元件的組合連接來預估電路的 特定行為。
- 電路模擬的基本要件：電路模擬軟體，元件模型，電路描述，模擬控制命令
- 除了分析電路的特性外，電路模擬亦用於評估電路的操作範圍、消耗功率、可靠度等

電路模擬軟體

- Logic Simulator/Switch level simulation
  - Verilog, SILOS
- Transistor level simulation
  - Timemill, Star-sim
- Time-domain Circuit Simulator
  - Star-HSPICE, SBTSPICE, Eldo
- Freq.-domain Circuit Simulator
  - ADS, Harmonica, EldoRF, SpectreRF
- Device level circuit simulation
  - Medici





$I_d=F(V_{gs},V_{ds},W, L,\ldots)$

模擬的基礎- 元件模型

元件模型 - 描述基本元件的行為特性

- Logic simulation
  - Gate delay model : $Td = A + B*C_L$
- Transistor level simulation
  - Transistor I-V /C-V table model
- Circuit level simulation
  - Transistor device model

電晶體參數模型

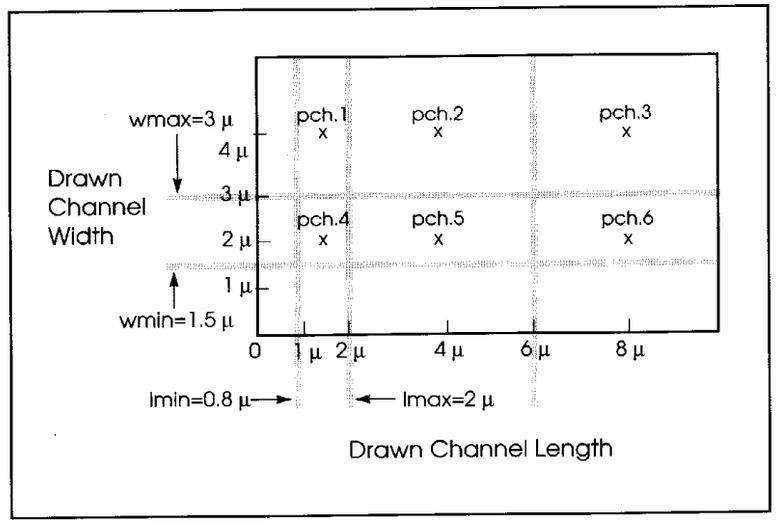
- **透過** Equation **描述電晶體的交、直流特性。不同的**model level **使用不同的**equation **及**equation parameter.
- **經由** device model extraction, **產生對應製程的**Equation coefficients.
- **為提昇**model **的準確度，部份電晶體模型依電晶體大小分組給定參數模型。**

HSPICE Level 49 : BSIM3V3
- UMC 0.5um 1P3M
- TSMC 0.35um 1P4M
- TSMC 0.25 1P5M
- TSMC 0.18 1P6M

Spectre Level 11 : BSIM3V3 TSMC 0.35um 1P4M

電晶體參數模型區間

- **每一組參數區間的適用範圍由**Lmin , Lmax, Wmin, Wmax **等參數指定。**
- **不要使用超出區間範圍之電晶體尺寸，對大尺寸的電晶體使用並或串聯的電晶體代替。**
- Spice **會依據電晶體的尺寸自動選擇對應的** model **名稱**



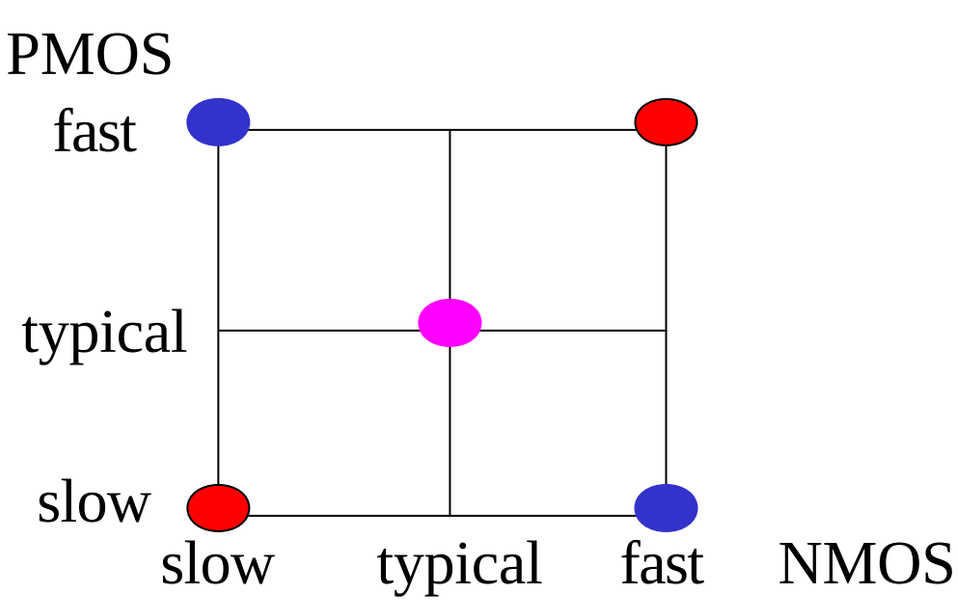
電晶體模型參數之製程變異

- **電晶體的特性參數隨製程的變動會產生漂移。**
- **依據特性漂移的範圍訂定電晶體的參數變異範圍，**NMOS **及** PMOS **分別定義三組參數**(slow, typical, fast)
- **電路的模擬應考慮參數的變動**:5 corners simulation(ss,sf,tt,fs,ff)

Model **的選擇**

.lib “model_file” TT
.lib “model_file” SS

Or
.lib “model_file” mos_tt



電晶體的種類選擇

- **為提高電晶體的使用彈性，及兼顧各種系統應用的需求，新的製程提供多種不同特性的電晶體如** 2.5V/3.3V**等不同工作電壓的電晶體及不同臨界電壓的電晶體**(normal Vt, zero Vt, native Vt…) **等，而為補充電晶體高頻特性的模型特性，另有電晶體的**RF model**。每一種電晶體都有其對應的**model name**，在電路模擬時，需正確使用對應的電晶體模型。**

In Netlist :

M11 Dr Ga Sr Bu **mnch2** L=.. W=…

.lib “model_filename” TT_M2 : 2.5V Medium Vt devices
.lib “model_filename” TT_T2 : 2.5V Zero Vt devices

模擬考量-功能驗證

- 應詳讀元件模型參數檔內之說明
- 模擬輸入應包含足夠的初始化時間(Initializing pattern)
- 模擬輸入的變化應能使電路中各點均有變化的機會(toggle rate)
- 模擬輸入的數量應足夠驗證電路所需的功能。
- 模擬的過程應檢查有否任何的warning或error訊息。
- 儘量使電路結構化以利功能模擬時的除錯。
- 可能的話，採用Top-down 的階層化設計方式，逐步完成電路各不同部份的完整設計。
- 功能驗證時先不考慮製程的參數偏移，以標準參數來模擬，惟各項規格需有足夠的margin.

模擬考量-規格驗證

- **在評估電路的各項特性時，應以** worst case**考量(含**Model, VDD, Temp variation)**，並預留** design margin**。**
- **在電路模擬時應加上預估之輸出負載。**
- **在**timing **上找出** longest delay path**及**pattern**，在**power**方面找出** largest power pattern**。**
- **在電路模擬階段，因未計入連接線的雜散負載效應，因此在考慮時序模擬**(timing simulation)**時，應再預留一些區間**(~30%)**。或於電路模擬時加入預估負載。**
- **不同的規格可能需使用不同的電路架構模擬。**

  例如 slew rate, frequency response...

  Worst case power : VDD_{max}, T_{min}, FF corner

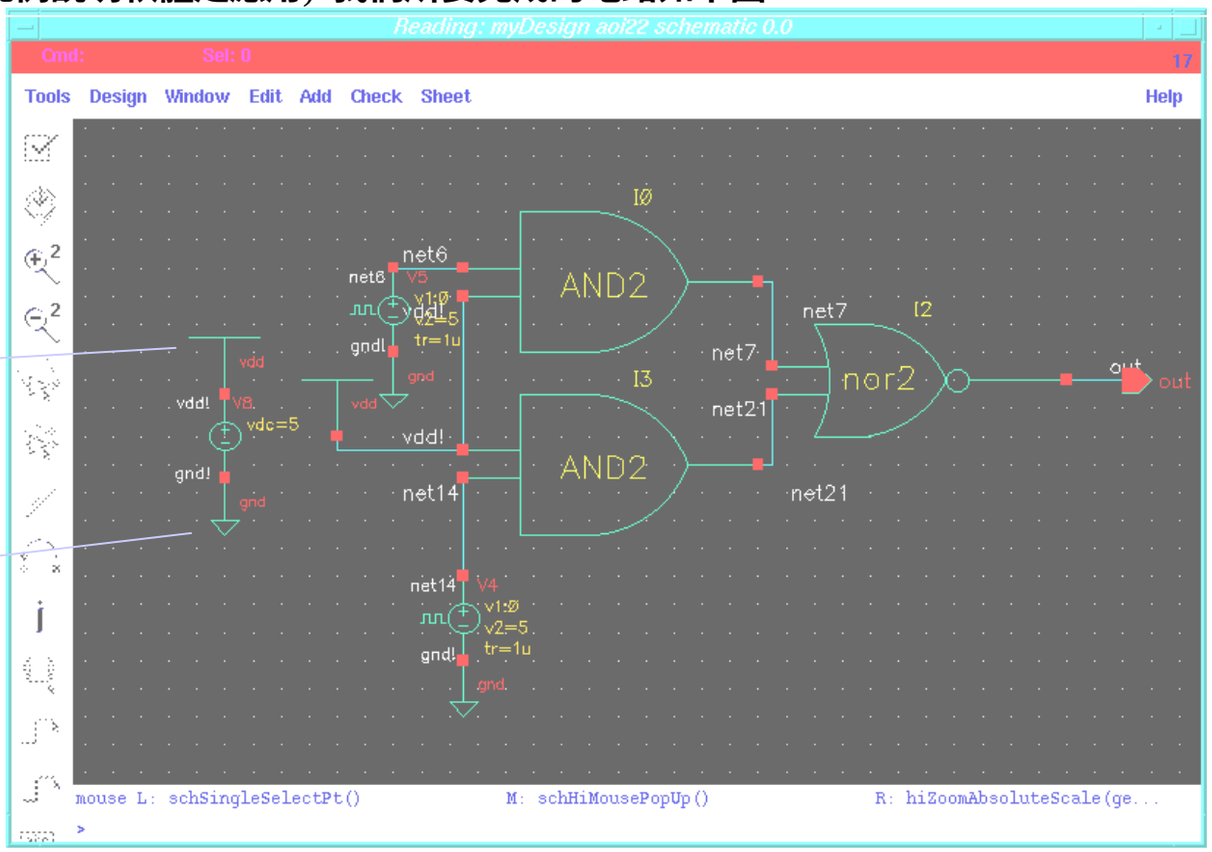
  Worst case delay : VDD_{min}, T_{max}, SS corner

Schematic Entry & Netlist Creation

除了用hspice netlist 建立input ckt,或用verilog in import ckt外,可用composer建立schematic view, 我們將舉例建 2 input and 與 nor 的 schematic view,symbol view, 並將 and 的兩個 transistor w值設為變數, 而利用 symbol view 完成 AOI ckt 的schematic view, 再以 hspice 作模擬,(實際 AOI ckt 並非如此設計,只是藉此例說明軟體之應用). 我們所要完成的電路如下圖 .

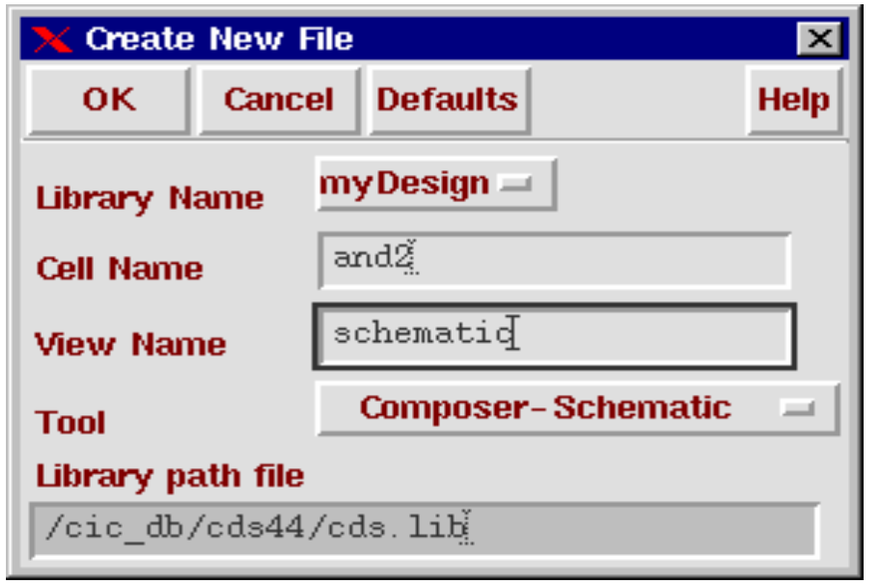
vdd!(global vdd)

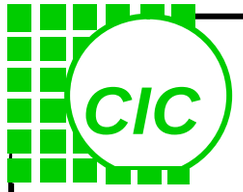
global DC power



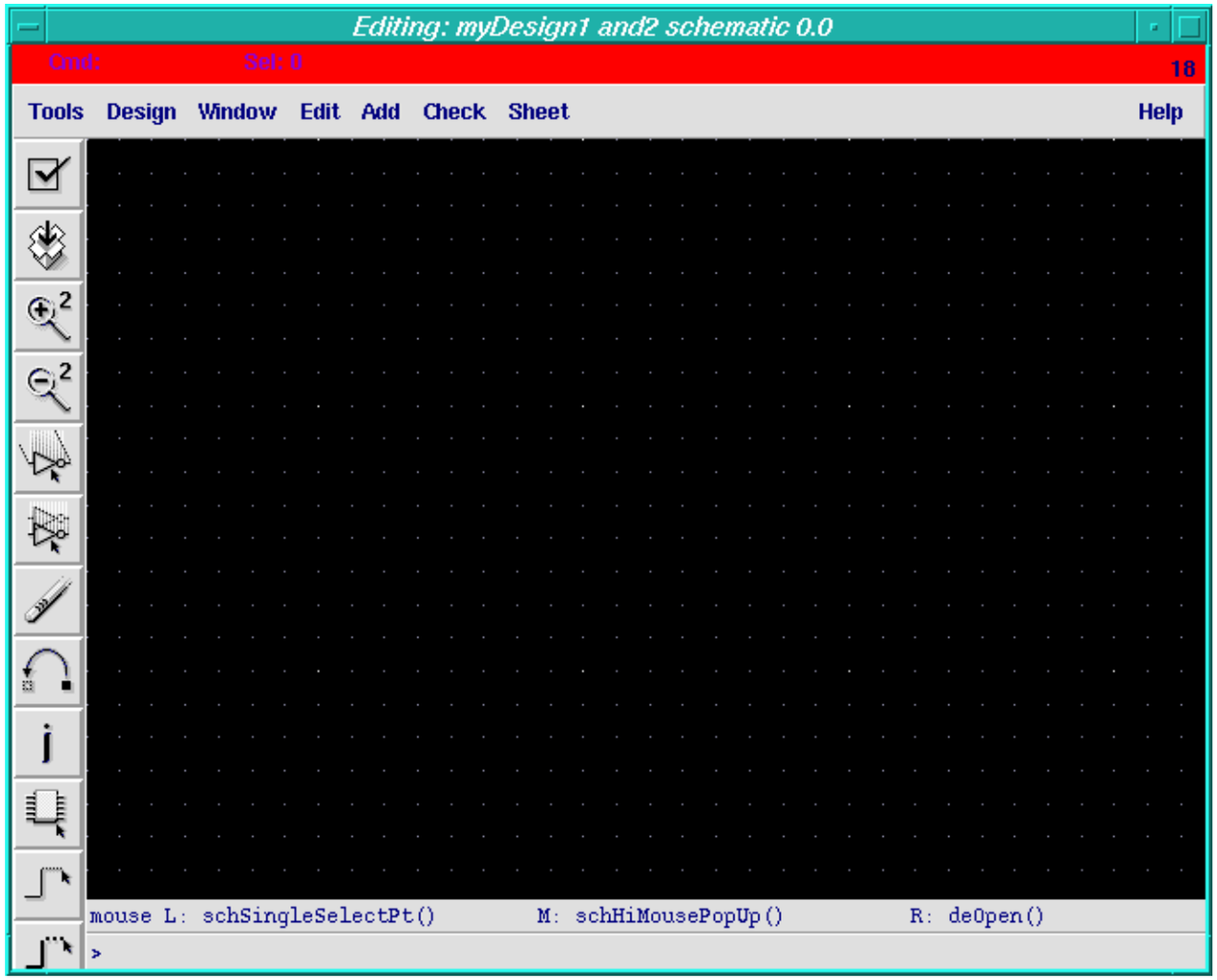
Schematic entry - Create design

File -> New -> Cellview

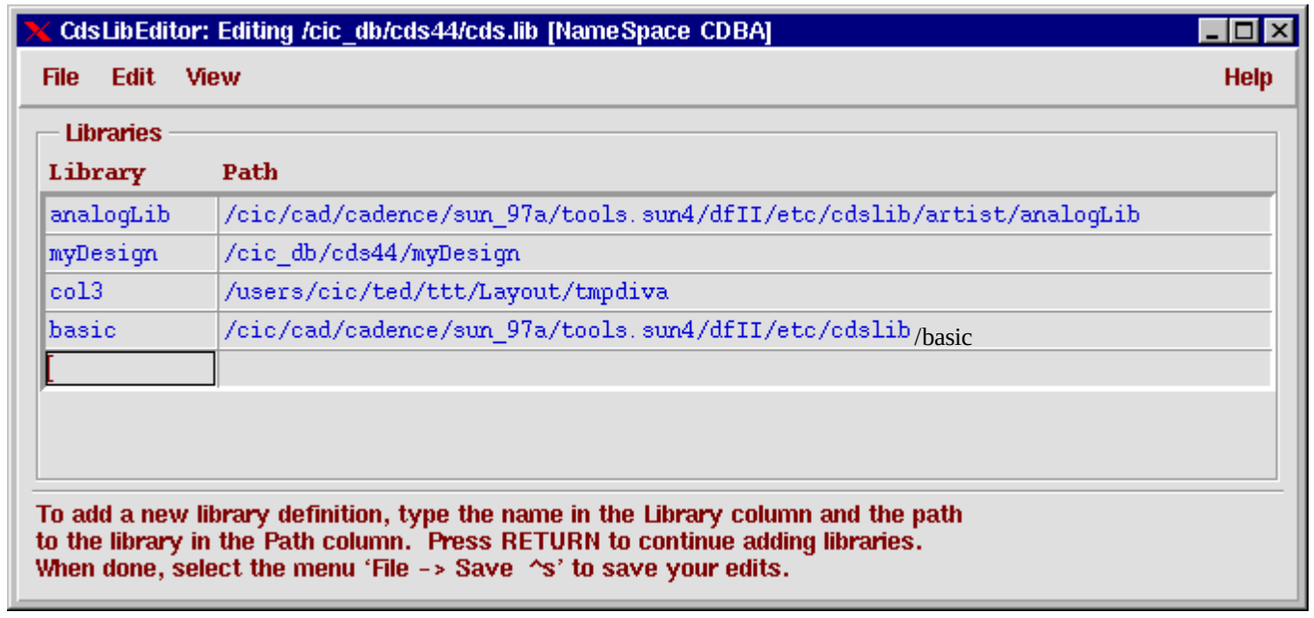




將出現 空白的 schematic window **如下**, **此** window **讓** user**畫電路圖** (schematic), **以作**on-line LVS or circuit netlisting **等**.

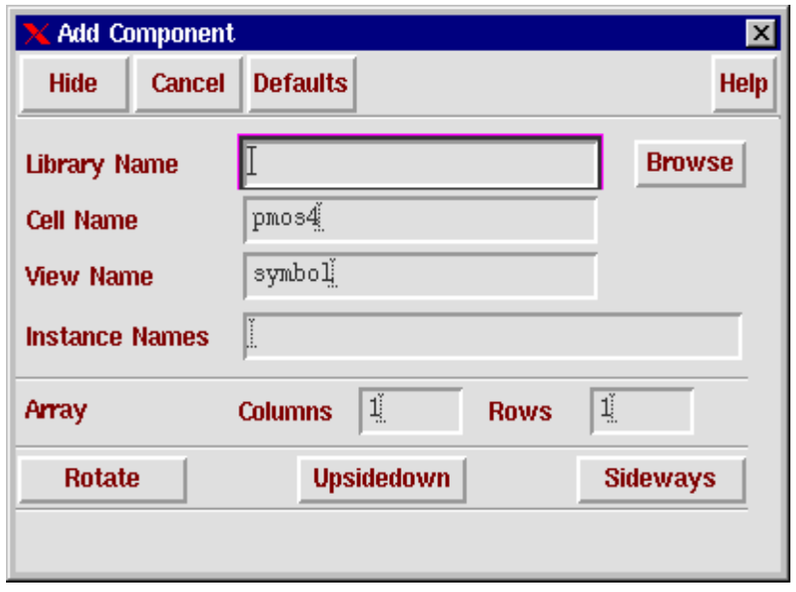


利用 2個現成的 library:analogLib & basic (show 於 Library Manager中) ,若未出現 在 Library Manager中, 則CIW上Click
Tools -> Library Path Editor (或在Library Manager中選 Edit -> Library Path...)
呼叫出 CdsLibEditor window, 將analogLib & basic 之Path(<install_dir>/etc/cdslib & <install_dir>/etc/cdslib/artist/analogLib)加進,再select 'File -> Save'存到cds.lib以使顯示在 Library Manager 的window中

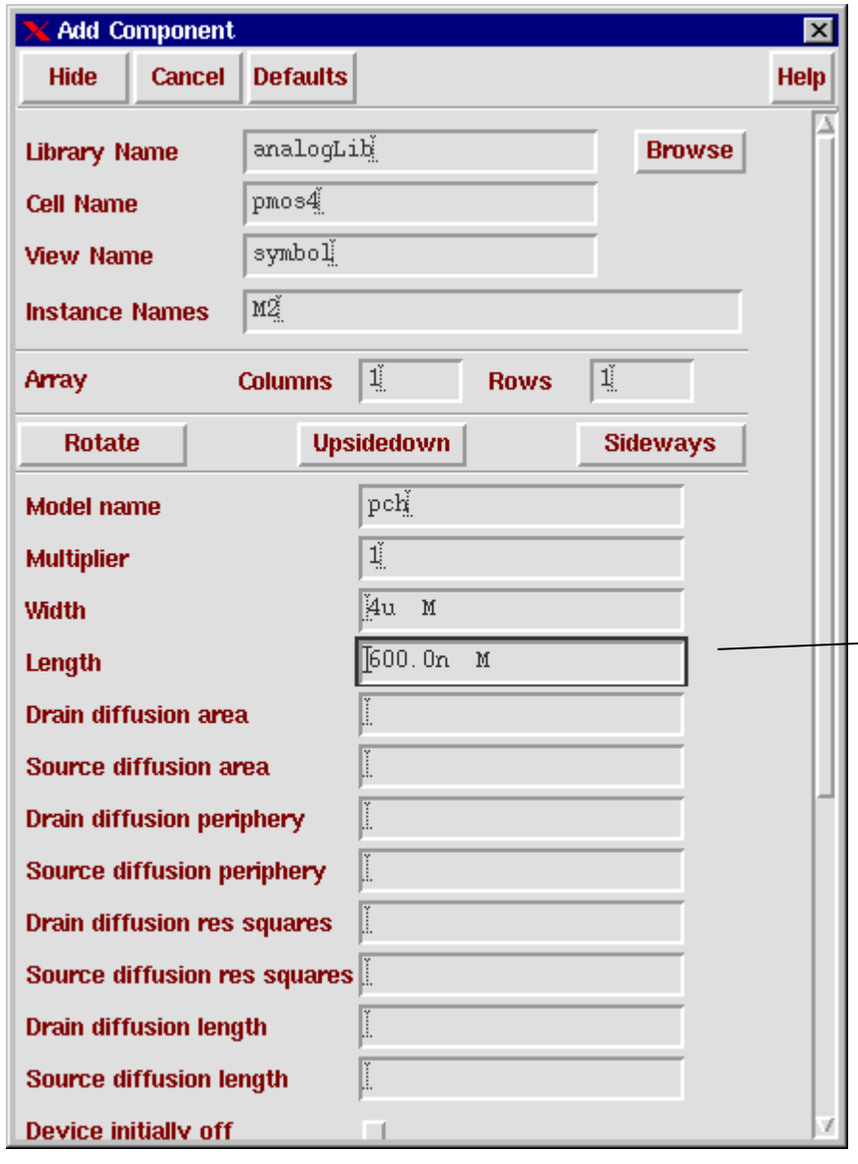


利用DFII軟體中的 library: analogLib 與 basic 已定義好的元件完成schematic view

Add -> Component...



Click 此 form 之 Browse, 則出現Library Browser window (for Add component)並click Library Browser 中 analogLib 之 pmos4 處, 則form 中 自動填 Library Name 與 Cell Name .



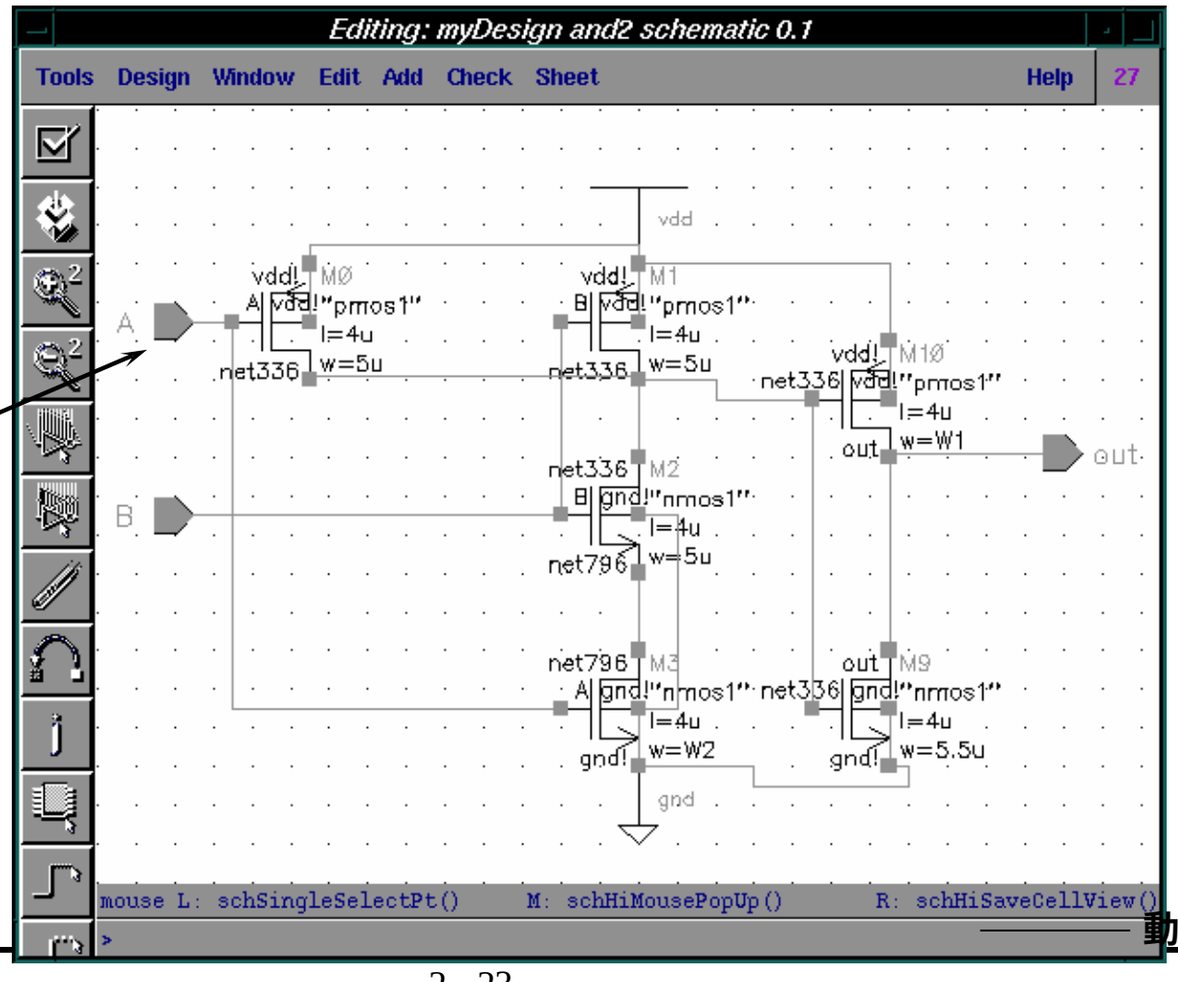
在 form**中可 填** w ,l **值等** ; **在此只填** w ,l **值** , **移** mouse **於** schematic view **中**, **可 看到** pmos**的** symbol , **移** mouse **到欲置放之處** click

同理 ,Add Component nmos4 ,vdd **與** gnd

再 Add ->Pin... ,define input terminal A,B**與** output terminal out(pins represent connection points between different cellviews. Using named pins identifies equivalent input, output, and I/O ports. Pin name, pin type, and pin direction should be consistent throughout the whole design.)

再用 Add ->Wire(narrow) **作接線**

完成 2 input and **之** schematic**如右圖**

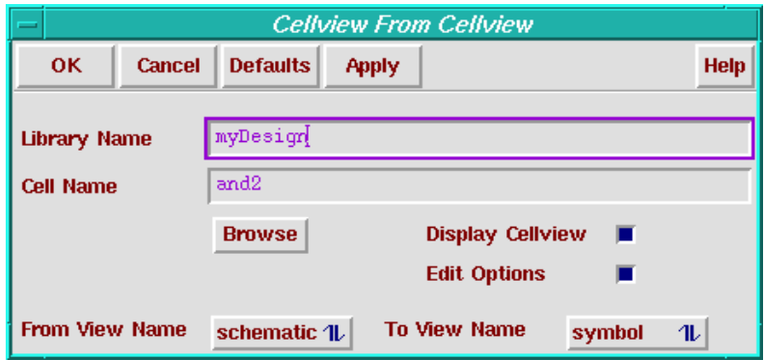


Pin

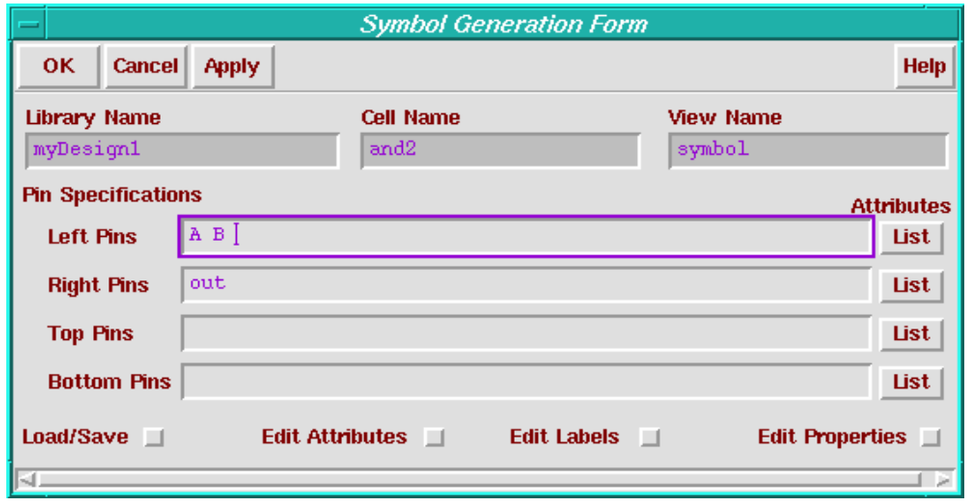
改w,l**等值可用**
bindkey q

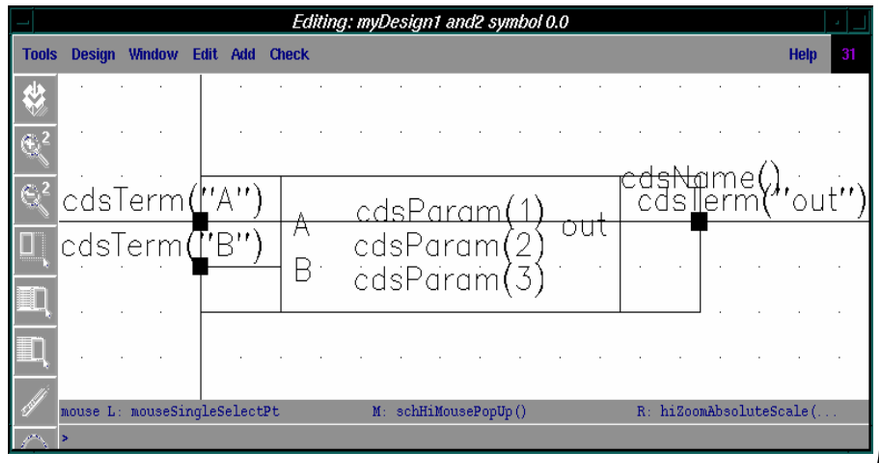
動作提示

Design->Check and Save, **如有錯誤，需改正到**CIW show no error
Design->Create Cellview->From Cellview… **從**schematic **來產生**symbol view



Click OK, then define the location of I/O pins Click OK ——— A symbol is generated.





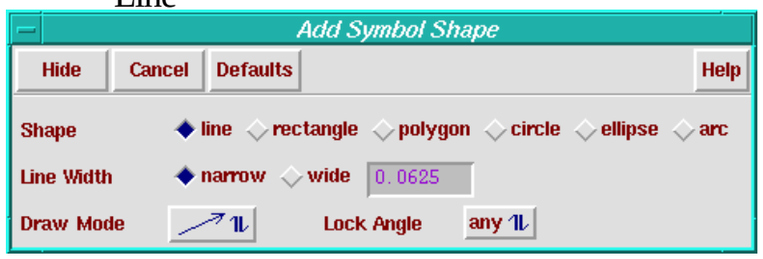
Create Symbol View Manually

File -> New -> Cellview

Select the Composer - symbol as the tool name for editting symbol view

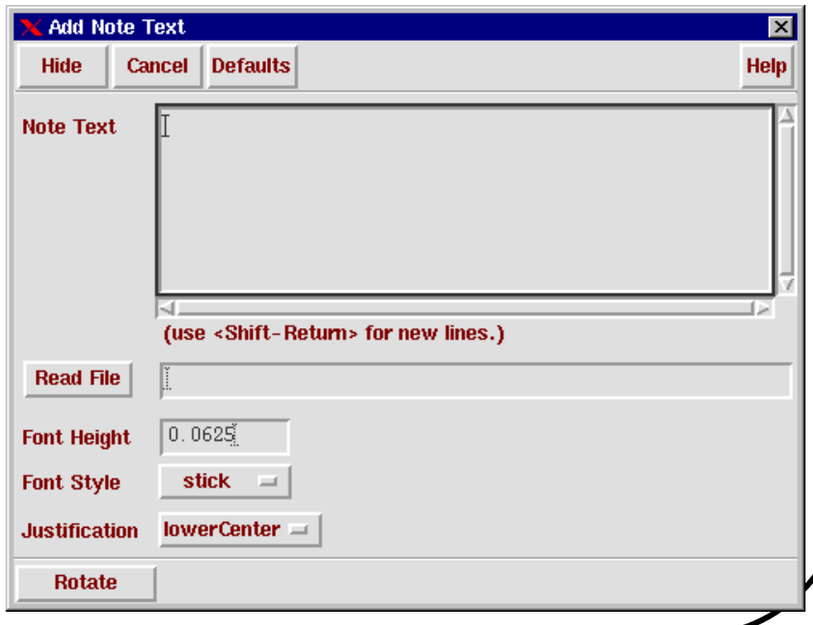
使用 Add->Shape->Arc

Line

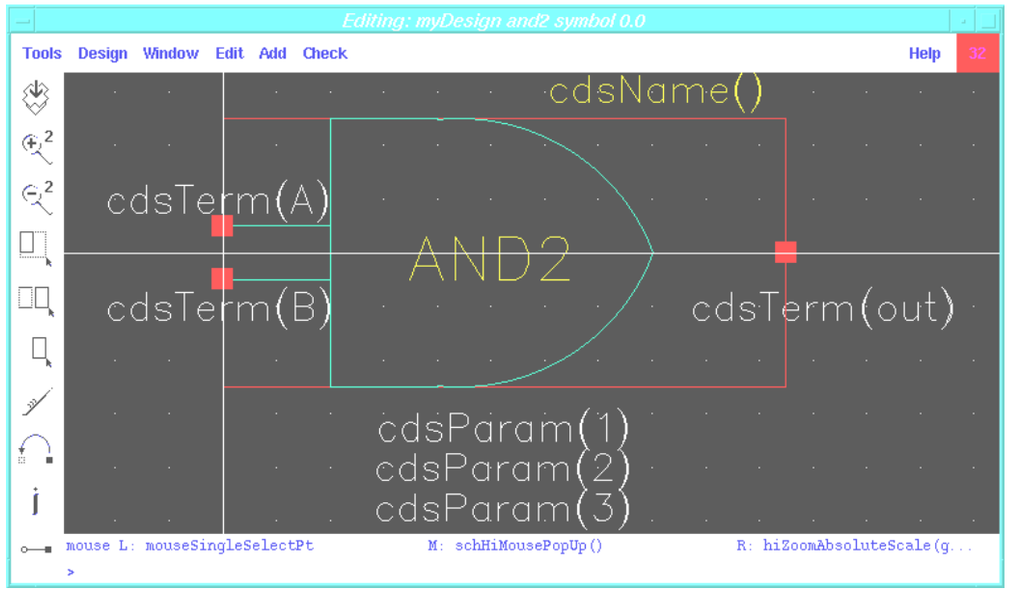


Edit->Move

Add->Note->Note Text...

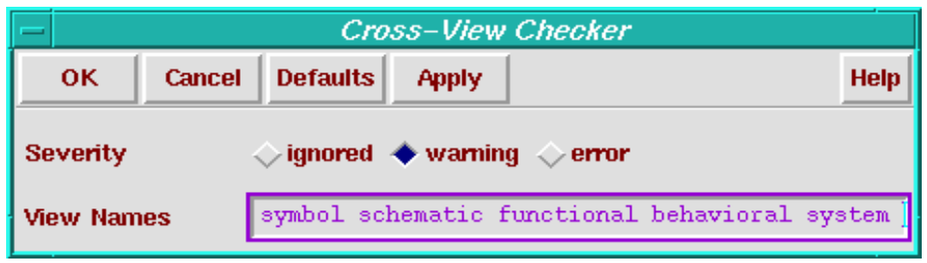


得如下之symbol view



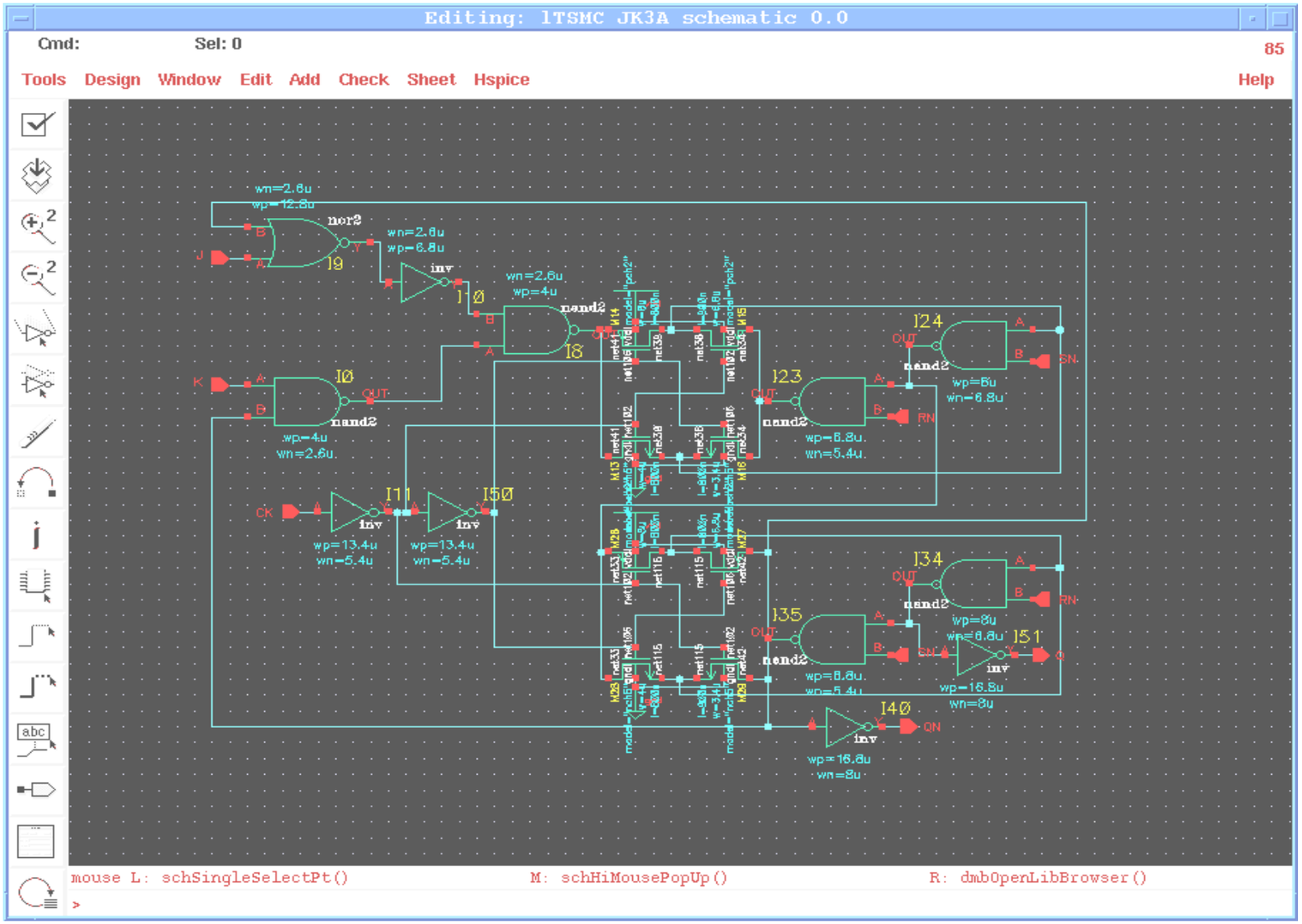
cdsTerm() labels display pin names or the net name.
cdsParam() labels display parameters of an instance.
cdsName() labels display the instance or cell names.

Check->Cross View Check...



Click OK, Design->Save

我們舉例建 JK Master-Slave flip flop Schematic view **如下**:



變數設定方法

為了方便使用者設計電路從事模擬 (hspiceS simulation), 避免電路組件多重建立，
故採用變數設定增加使用者做電路模擬的彈性． 使用變數設定參數值，
具備有階層傳遞參數值的功能(兩層)，可以在插入元件時順便設定參數值，並使用Edit -> Label Display
顯示出目前的參數值。(類同定義sub-circuit, 而sub-circuit中設變數以便呼叫時給定不同變數值)

1. 直接以變數設定：

   在編輯 Schematic cellview 插入元件時，以變數名稱代替數字。
   如下例所示：

   pmos width -> wp
   length -> lp

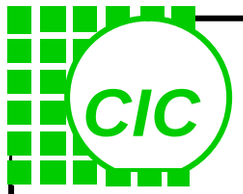
   做電路模擬時在 simulation window 做 Setup -> Design Variables -> Edit 的步驟
   設定及更改參數值，以便進行模擬.

2. 用pPar變數設定：

   使用 pPar 方式設定變數, 在 create symbol 之後, 可以在CDF中設定預設值
   (在 CIW 視窗上 -> Design -> CDF -> Edit), 並顯示出參數是否為預設值.
   如下例所示：

   nmos width -> pPar("wn")
   length -> pPar("ln")

   做電路編輯時，可以在插入元件時順便設定參數值，做電路模擬時便不用做
   Setup -> Design Variables -> Edit 的步驟，只須更改上層電路元件的 Objects
   Properties, 便可進行模擬。

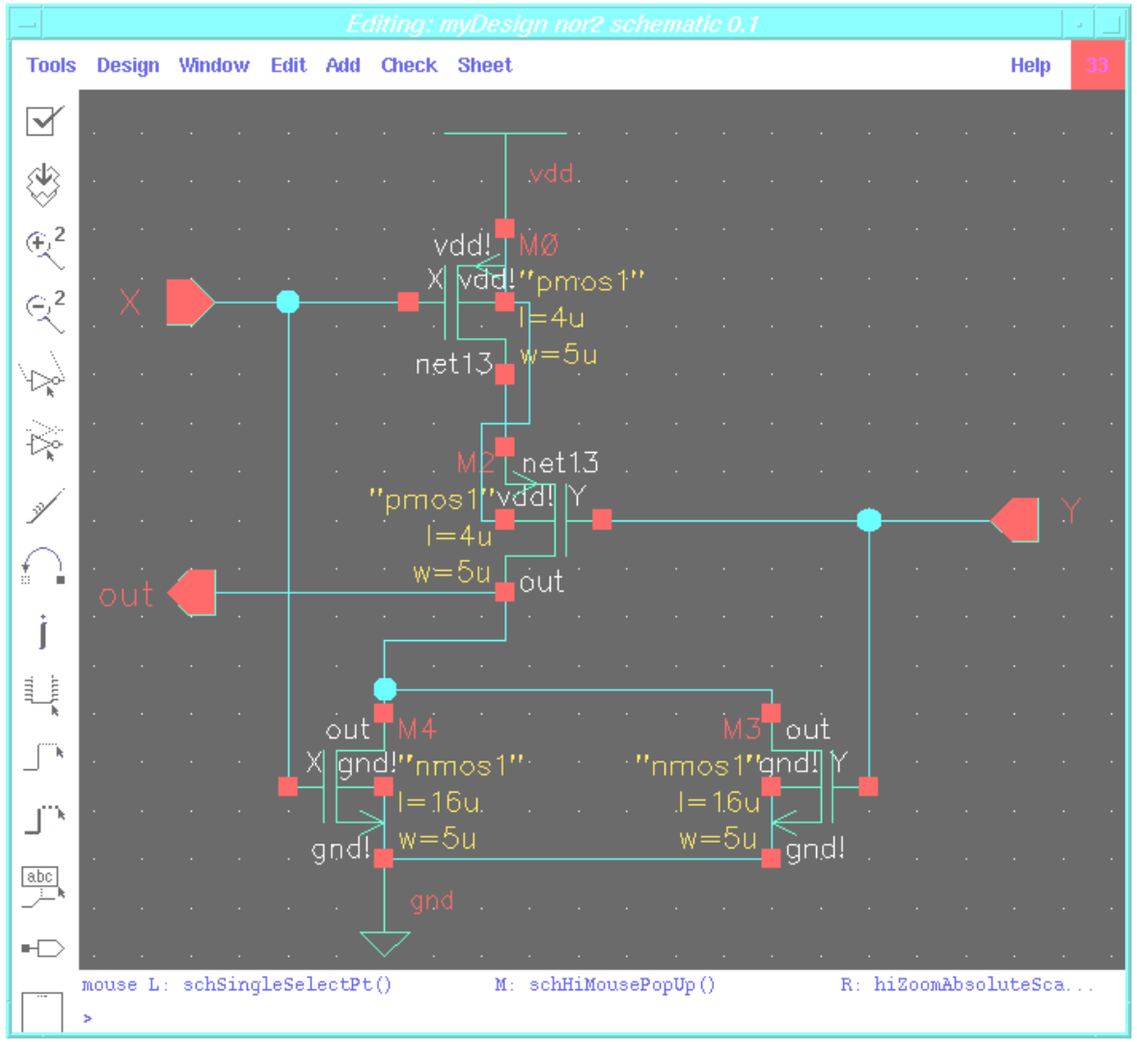


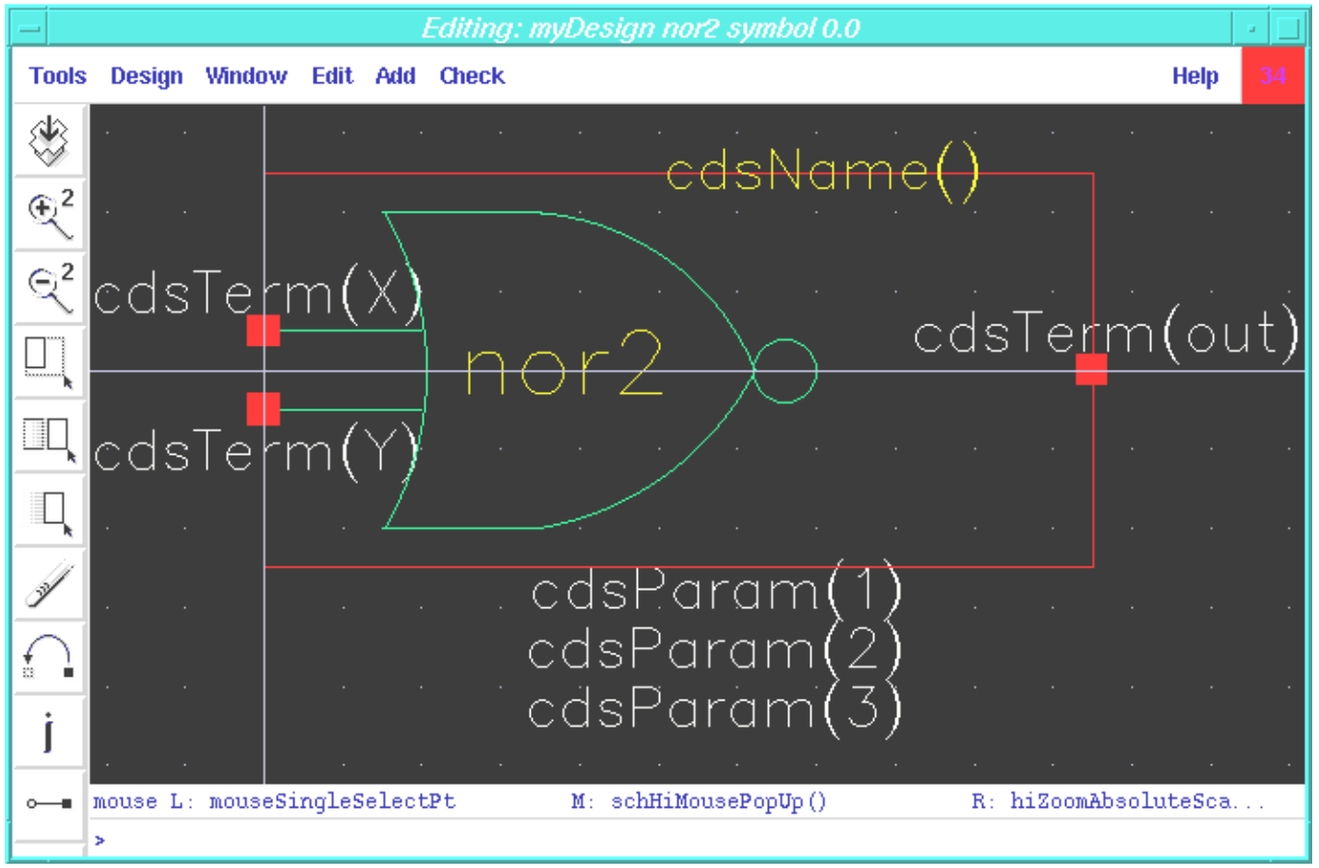
上述兩種方式在進行 hspiceS simulation **時**, **都無須重新做** Schematic Editing component **修改參數值 和** check and save・

Note: 1. **欲用** Edit Label Display **顯示參數值時**, **在已經建好之** symbol view**必須有** cdsParameter **存在**, **若沒有建** cdsParameter **則須用** (Add -> Label, **選擇** art device annotate) , **否則無法顯示**・

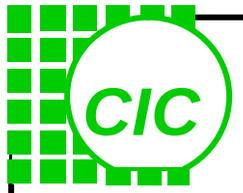
2. **若** user **有增加或減少** cellview pPar **變數**, **則必須將舊有** CDF parameter**刪除** (**在** CIW **視窗上** Design Manager -> CDF -> Delete), **重新** create symbol view **否則在進行模擬時** netlisting **會有** error message **產生**, **如** macroArguments **的錯誤**・

同法, **在** library myDesign **建** cell nor2,**並** create schematic view **與** symbol view **如下圖**, check **無誤後**, Design -> Save





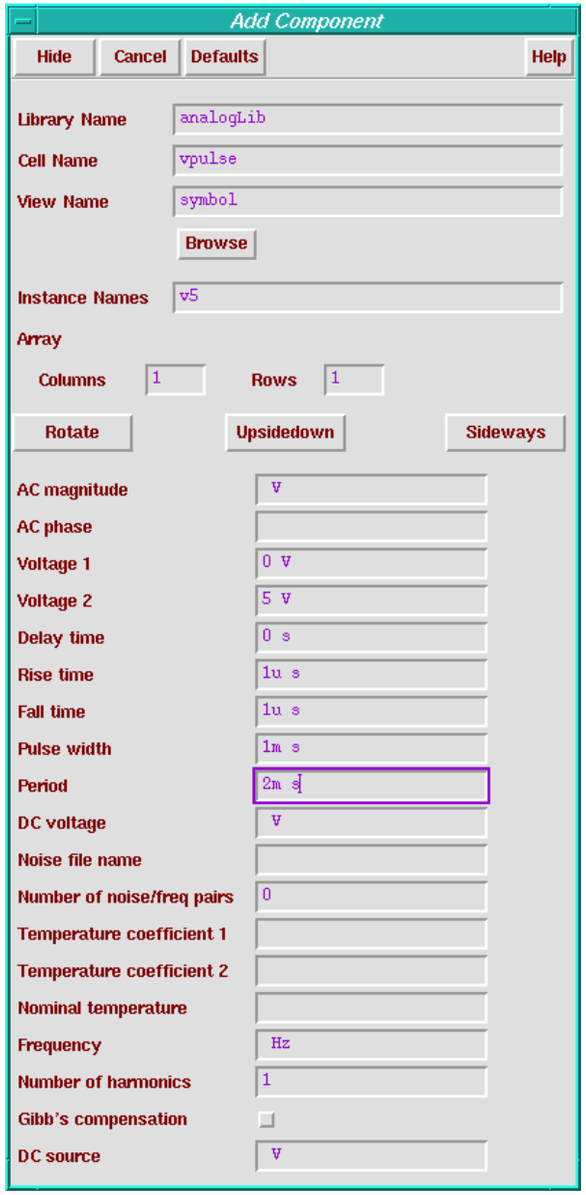
現 在, **我們完成了**cell and2 **與** nor2**的** schematic view**和**symbol view.
類同 5-1), open design: cell aoi22 schematic view, **利用** Add Component**等** edit aoi22 **之** schematic view **如下圖**

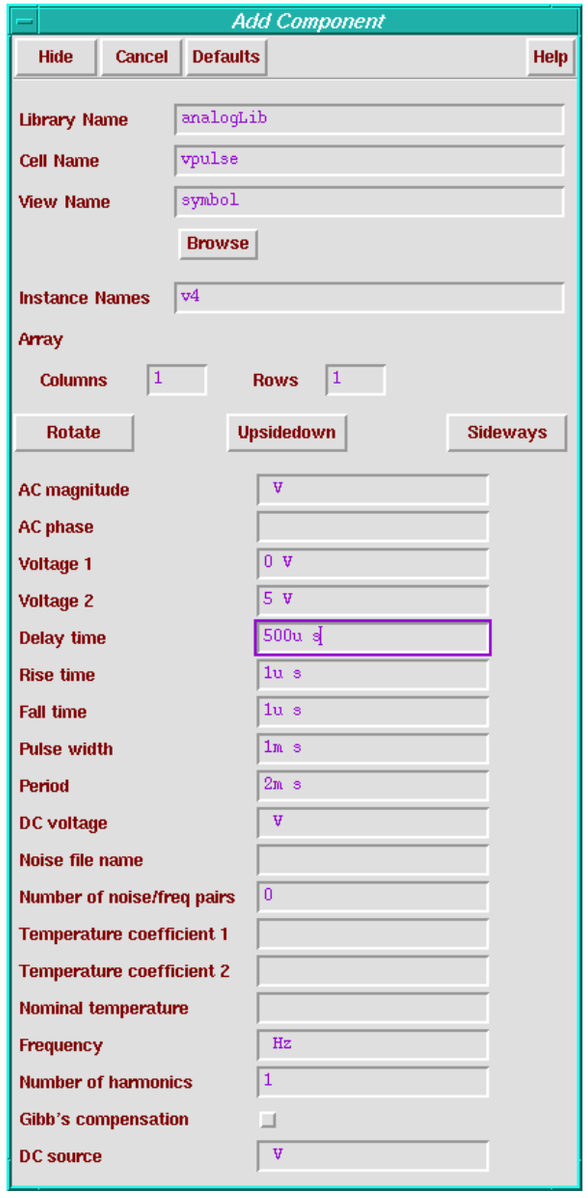


加入電壓源等元件，以便進行電路的模擬

Vdc, vpulse, vpwl

為 create 輸入電源V4, V5 , Add Component 如右, vpulse之參數表格乃於 Tools->CDF->Edit... 中設定

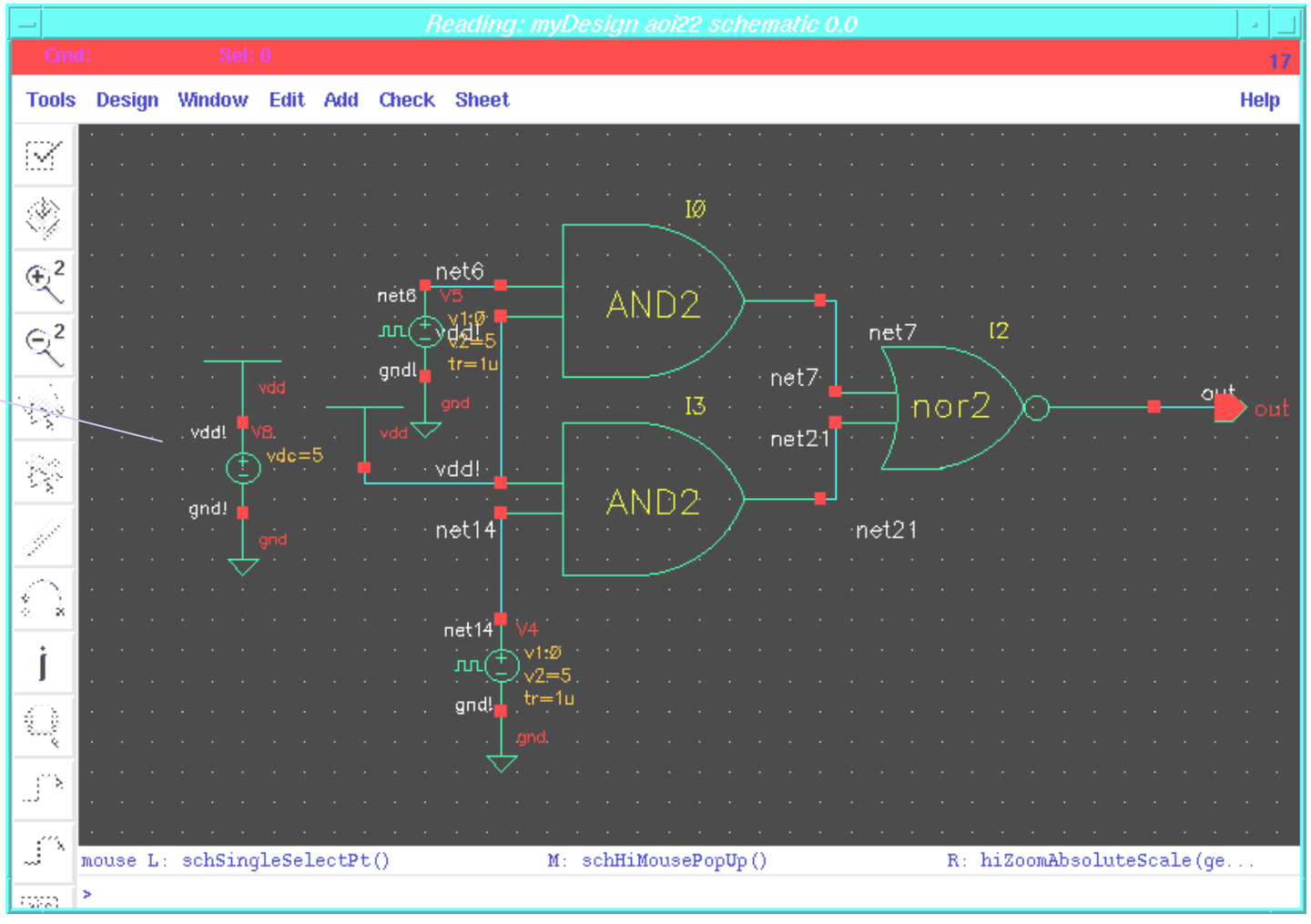




Any net or pin name that ends in “!” will be part of a global net. Global nets are automatically connected through the hierarchy without the use of wire

global DC power

Cell “gnd” is required for DC convergence.



Creating Netlist

- Creating netlist in Composer :
  - File->Export-> CDL　: CDL format netlist file for LVS verification
  - Analog Artist : Window based simulation environment, integrated form for entering other simulation information
- The same approach can be applied for extracted layout view parasitic devices can be reported if divaEXT.rul provide parasitic extract commands.

Circuit Simulation Techniques

Analysis types of SPICE

DC Analysis Statements:

(Each of the following statements uses the DC equivalent model of the circuit for its analysis functions.)

.DC Statement--DC Sweep
.OP Statement--Operating Point
.PZ Statement--Pole/Zero analysis
.SENS Statement--DC Small-Signal sensitivities
.TF Statement--DC Small-Signal Value of Transfer Function

AC Analysis Statements:

(The program first solves the DC operating point of the circuit and determines linearized, small-signal models for all nonlinear devices in the circuit.)

.AC Statement--AC Sweep(Small-Signal Analysis)
.NOISE Statement--used in conjunction with the .AC Statement, controls the noise analysis of the circuit.
.DISTO Statement--compute the distortion characteristics of the circuit in an AC small-signal sinusoidal steady-state analysis.
.Network Statement--Small-Signal Network Analysis
(Computes the Impedance matrix, Z, the Admittance matrix, Y, the Hybrid matrix, H, and the Scattering matrix, S, parameters. The input, output impedance and admittance are also computed. used in conjunction with the .AC Statement)

Temperature Analysis:

.Temp Statement--setting the temperature of a circuit for an HSPICE run.

Transient Analysis:

.TRAN Statement--compuates the circuit solution as a function of time over a time range specified in the .TRAN statement.

Fourier Analysis:

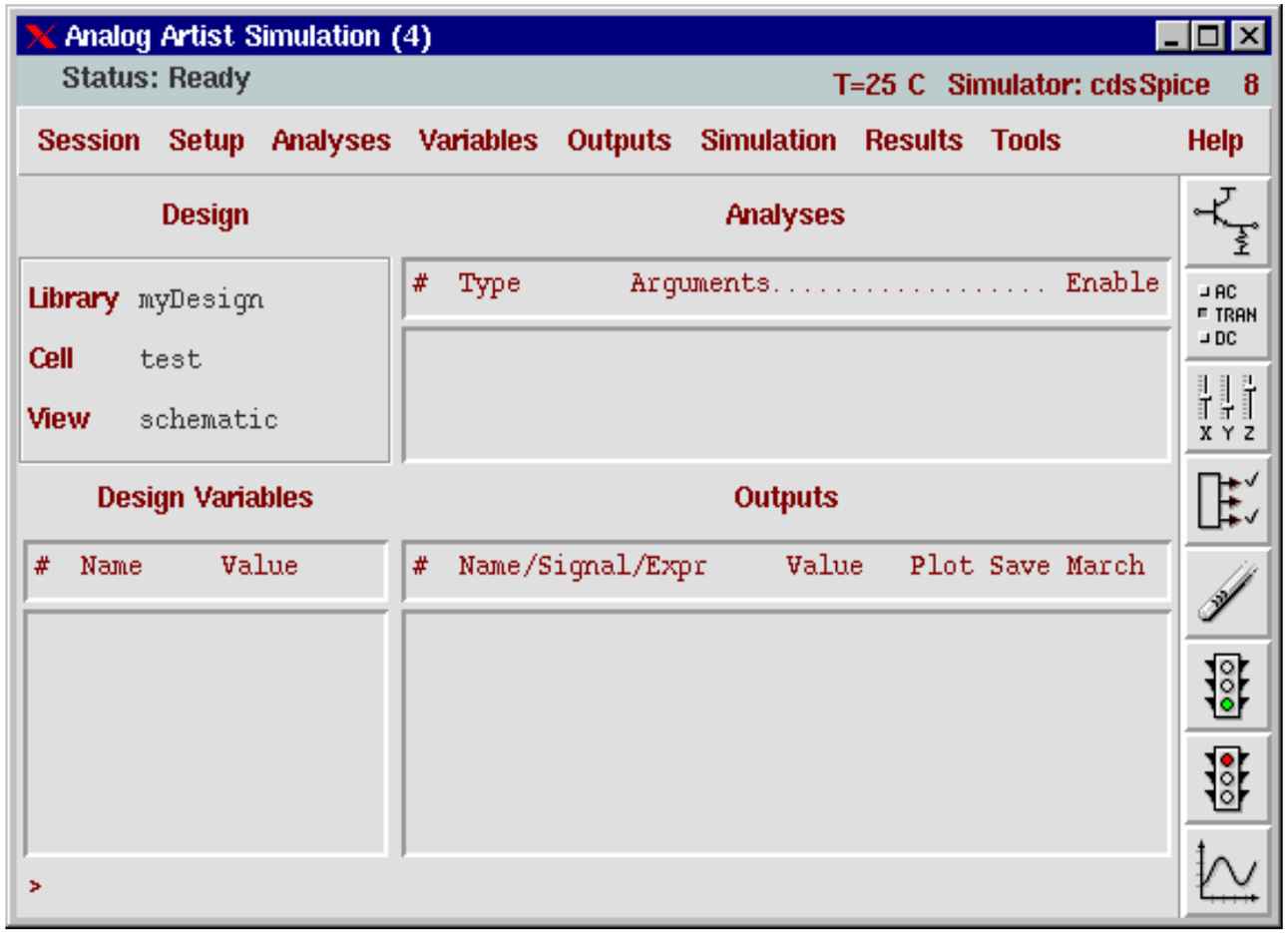
.FOUR Statement--perform a Fourier analysis as a part of the transient analysis.

Output Statements:

.PRINT---------prints numeric analysis results
.PLOT ---------plots generates low resolution ‘ptinter’ plots in the output listing file.
.GRAPH-------generates high resolution plots for supported devices such as HP LaserJet and Postscript printers without using HSPLOT or GSI.
.PROBE-------allows output variables to be saved in all the interface files with no additional output in the listing file.
.MEASURE---prints numeric results of measured electrical specifications for specific analyses.

Circuit simulation by SPICE Tools(spectre, hspice, sbtspice)

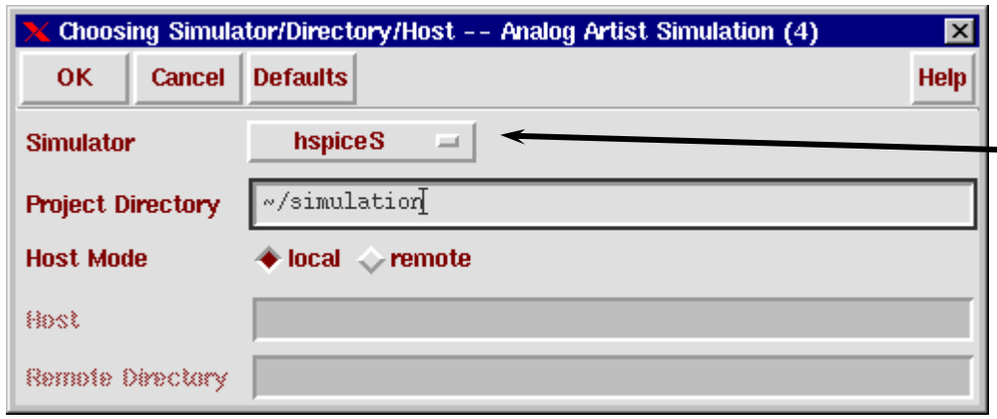
完成了 schematic view, **接著用**SPICE**作模擬**, **在**schematic view**中選**Tools -> Analog Artist



再用此window**選項**: Setup -> Simulator/Directory/Host ...

Initialize simulator

再用此window**選項**: Setup -> Simulator/Directory/Host ...



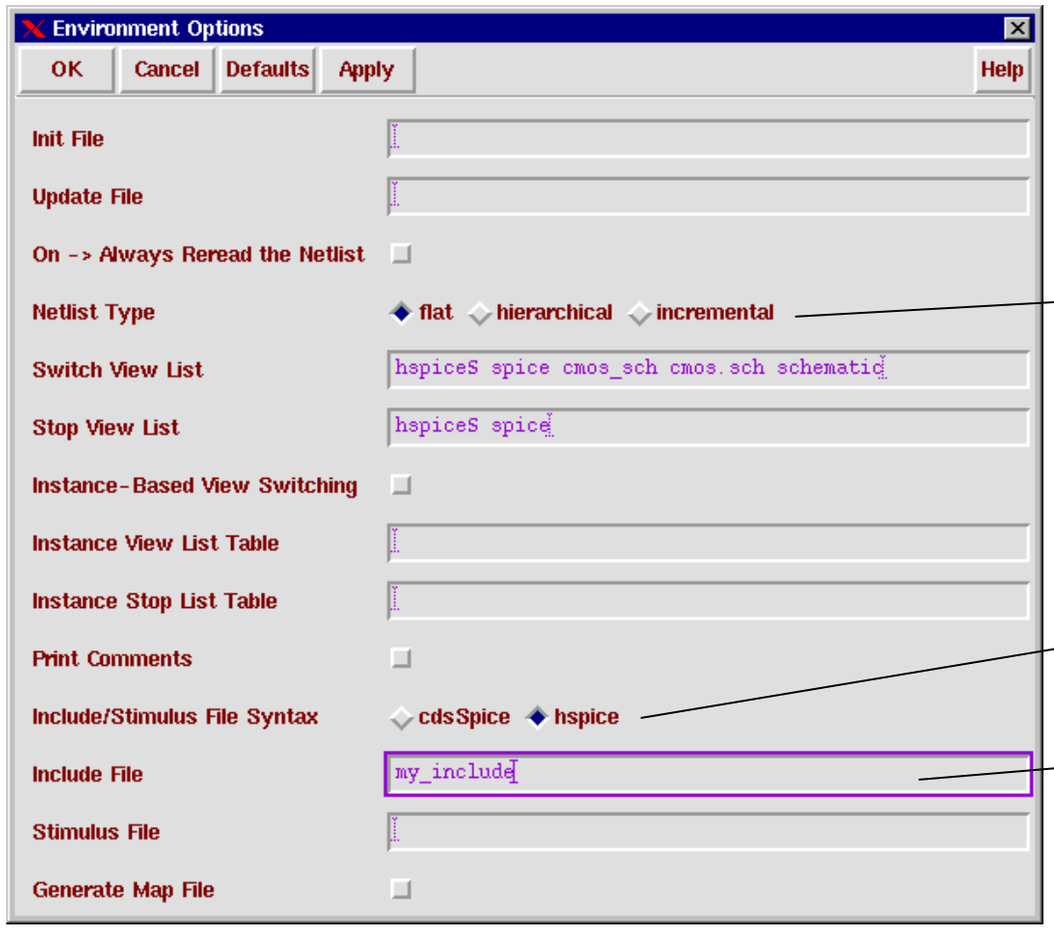
Select Simulator

Then Click “OK”

使用spectre **及** hspice **兩種模擬軟體時，可以直接在**analog artist **環境中指定觀察點、並執行模擬，及顯示模擬結果，若要在**unix **環境下使用** hspice/sbtspice**，並使用**awaves**或**sbtplot **來觀察模擬結果，亦可利用**analog artist **來產生**netlist**，再到指定的目錄中進行模擬的動作。**

為了得到所作schematic view**之**netlist,**可在** hspiceS window**上選擇** Simulation->Netlist->Create Final ,**則產生**schematic view**之**netlist “hspiceFinal” **在**netlist file**中指定分析種類及相關指令**, **以便在**UNIX**下以**HSPICE**或** SBTSPICE**作模擬**.

Simulation Setup



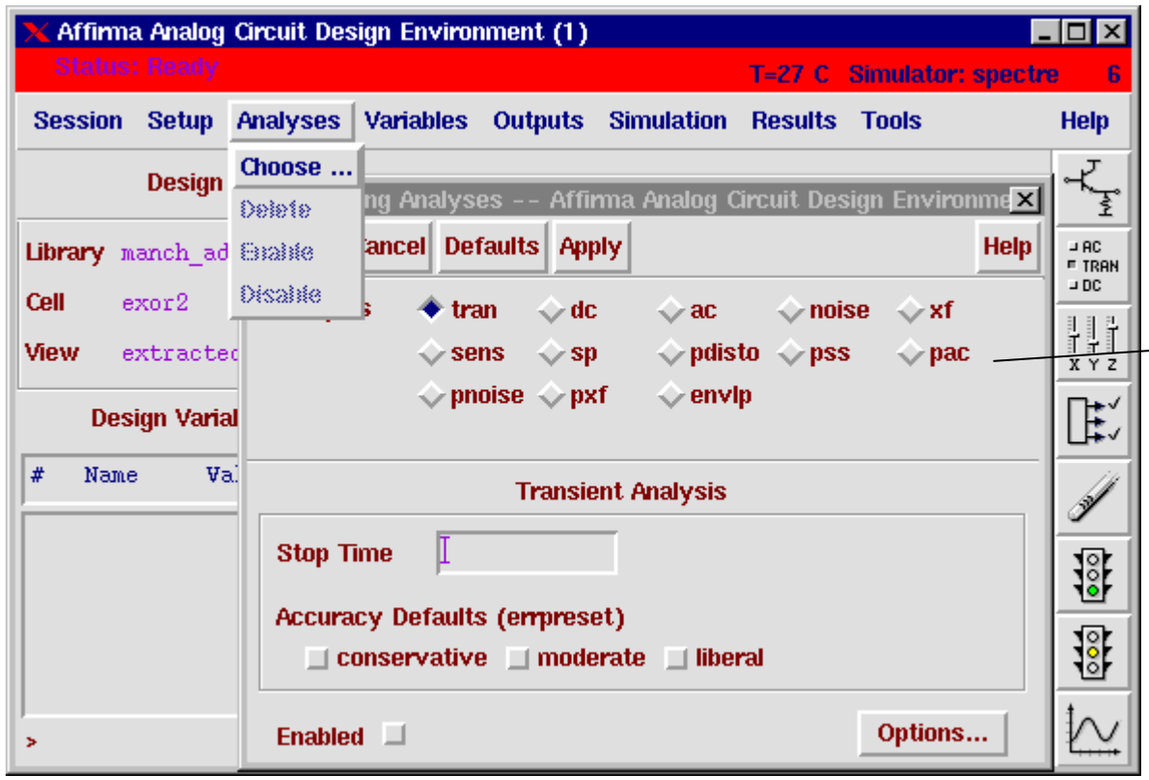
Select mode of netlist

Content of include file
.lib "model_lib" Corner

Select format of include file

Specify path and name of include file

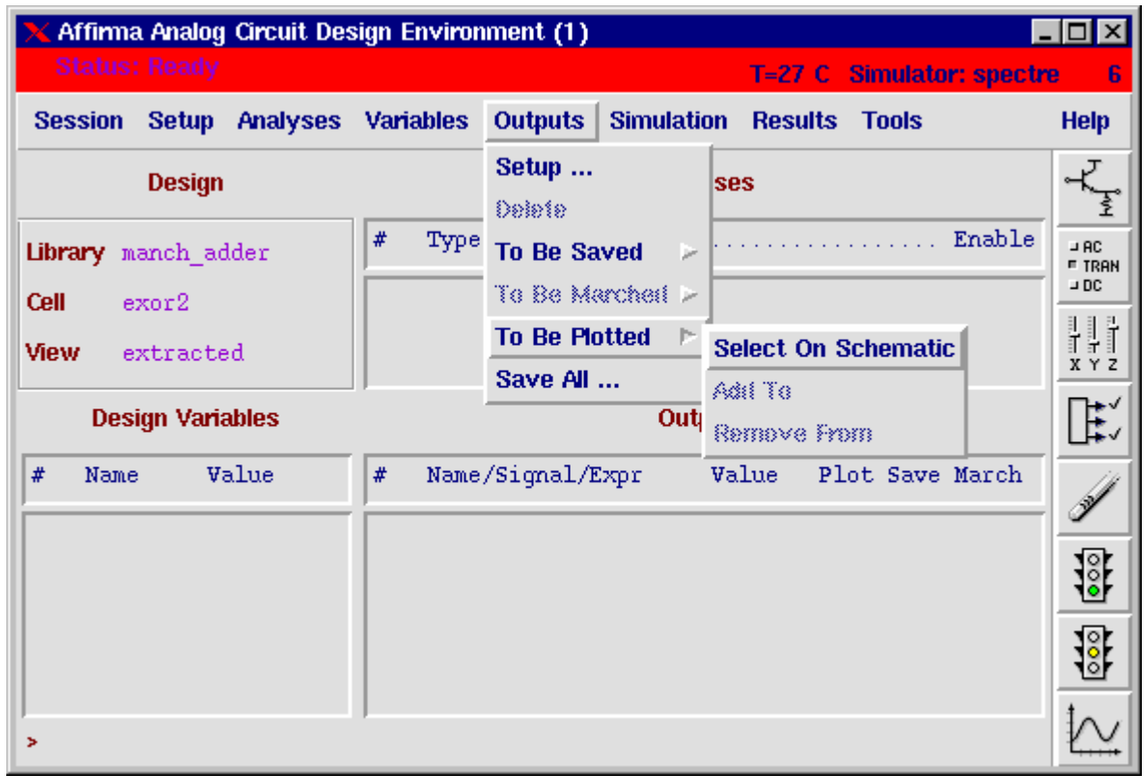
Selecting Analysis Type



Choose the analysis type for this simulation

For SpectreRF

Selecting Output Node

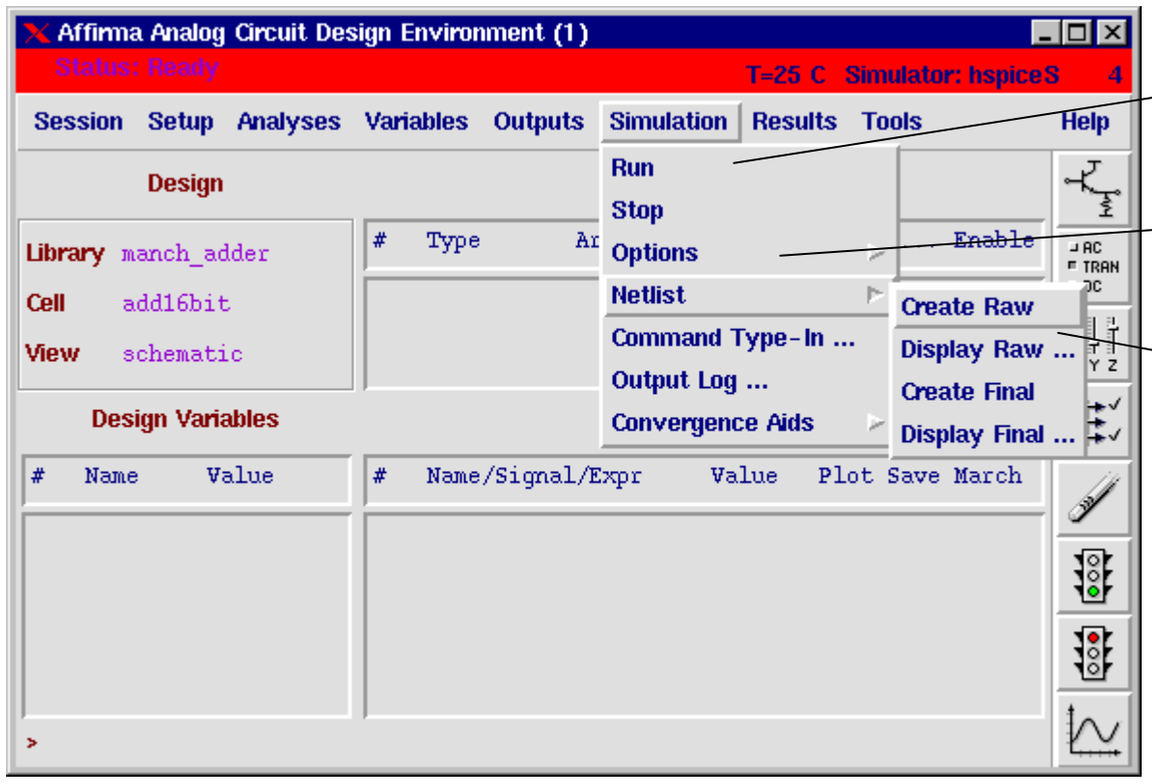


Saved : storing waveform data

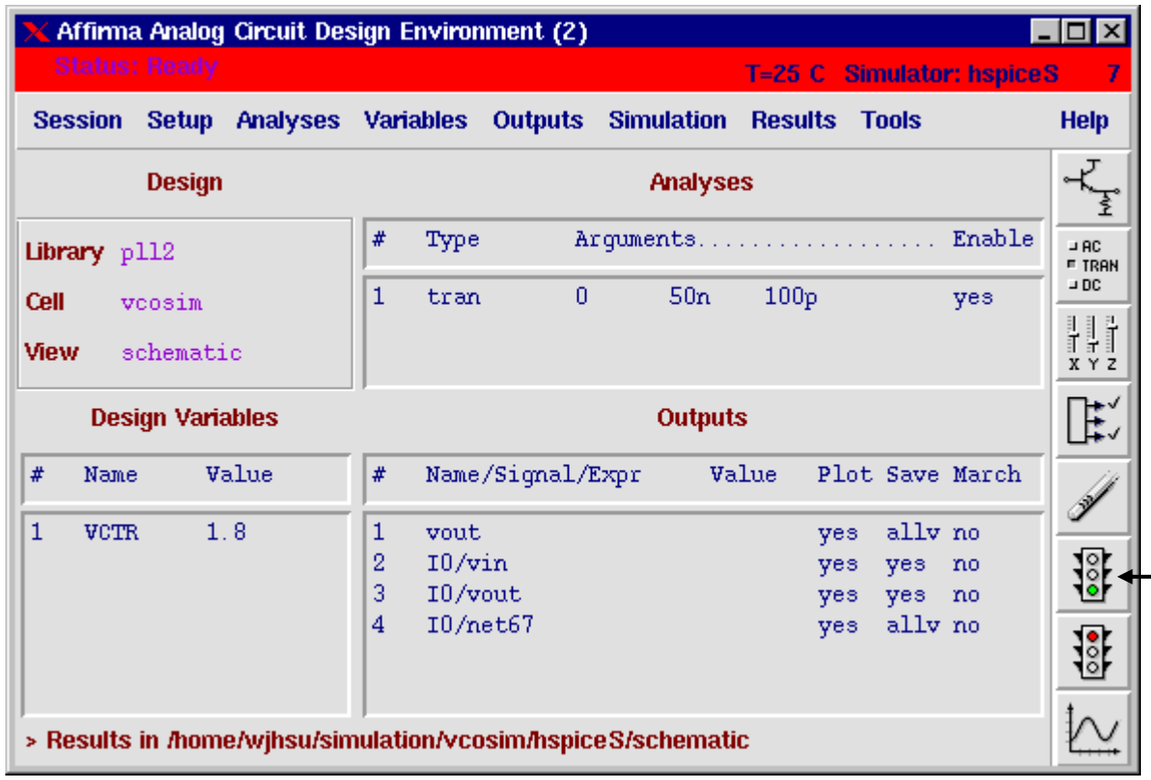
Marched : showing wave during simulation

Plotted : on waveform window after simulation

Netlisting and Simulation



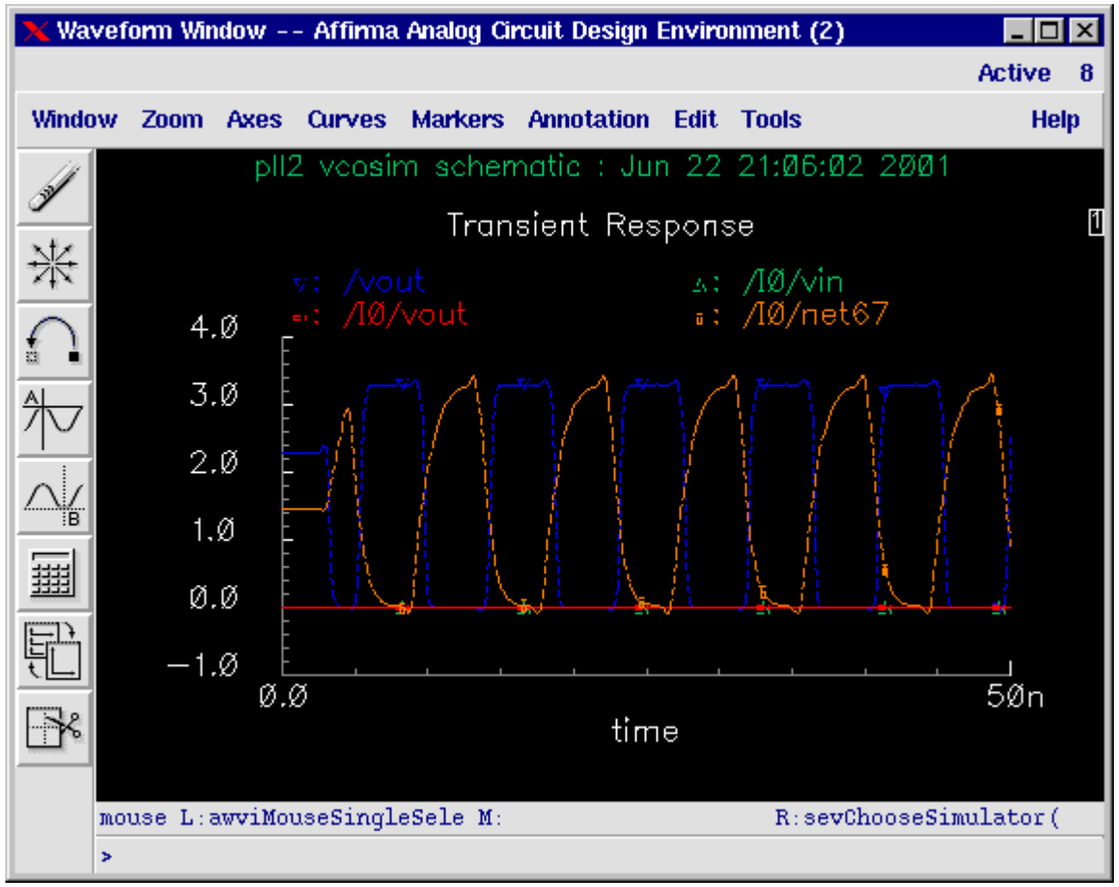
Simulation Window



在設定各項模擬參數後，可按 RUN 以開始進行模擬。

Run button

Result Waveform



模擬完成後，analog artist **自行啟動** waveform viewer **顯示模擬結果。**

Preparing SPICE Netlist - Simulation outside analog artist(in unix environment)

SPICE 模擬所需的netlist file 可利用之前所得的"hspiceFinal", 再將輸入信號, 所欲做的分析, 輸出命令等命令列加入後來模擬電路, SPICE的分析著重點在精確, 所以當元件太多時模擬會有困難

利用HSPICE的automatic model selector,即MOS之model name均給為一樣 ,如 nch & pch.
netlist file中改 .OPTIONS為
(ex) .options post acct probe (enable HSPLOT interface, binary output format)
並利用 .LIB call, 如 .lib 'ls35_4_1.l' fs (意即用本目錄下的 ls35_4_1.l model file(for 0.35um 1P4M Silicide process),以fast NMOS, slow PMOS case作模擬)
在抓用model file前該先檢查model file中是否有使用說明, 適用的HSPICE版本等.
若用0.5um 2P2M process則可用 .lib '9905spice.model' mos_sf, 意即用本目錄下的 9905spice.model model file, 以slow NMOS, fast PMOS case作模擬.
再用SPICE作模擬.例如netlist file name為myckt.sp,則用
%hspice myckt.sp > myckt.lis(以SBTSPICE模擬, 則用sbtspice command)

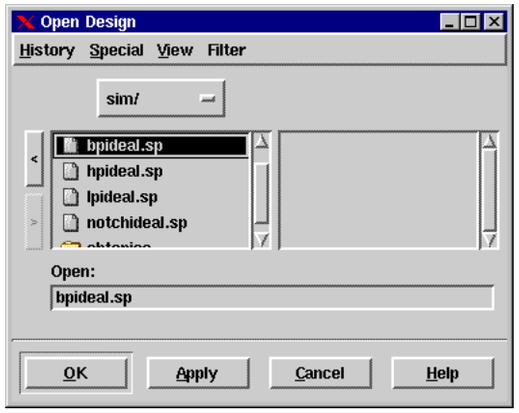
由 myckt.lis可看到執行成功訊息或失敗原因, 須檢查是否有error, warning report.
,執行成功後在netlist中所指定的每個分析會產生
圖形資料檔,副檔名為 "XX#" , 例如 myckt.tr0 <---第一個暫態分析的圖形資料檔
myckt.sw0 <---第一個直流分析的圖形資料檔
myckt.ac1 <---第二個交流分析的圖形資料檔

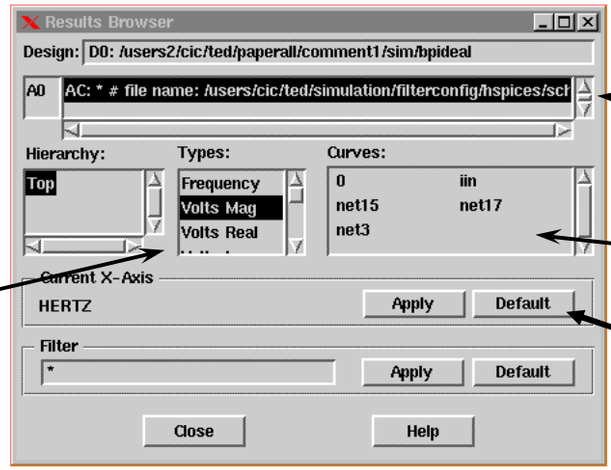
再利用AvanWaves(SBTSPICE 用sbtplot)顯示模擬後的波形

使用HSPICE的AvanWaves

%awaves

Design->Open（選模擬之電路檔）



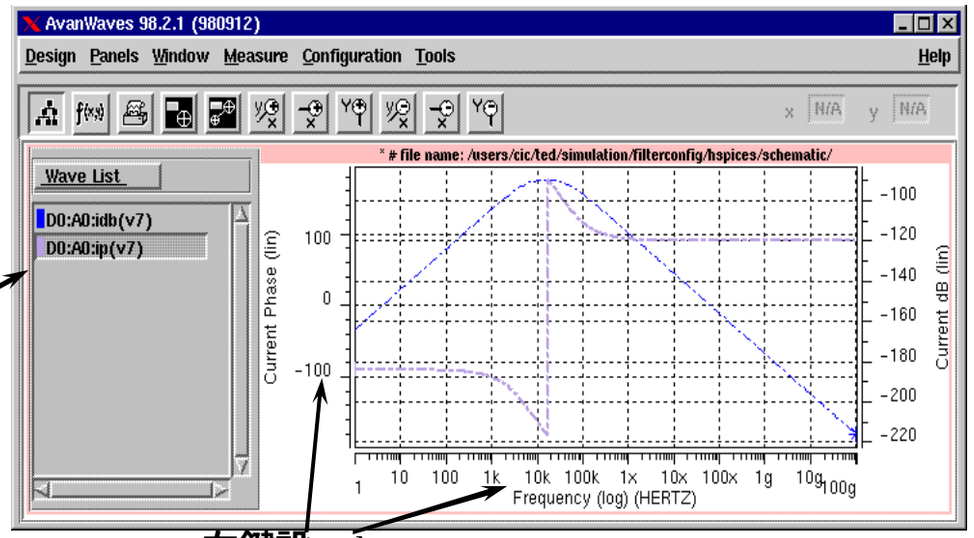


選看電流或電壓資料

選定要看的圖形資料檔

Double click 欲看波形的節點

設定X軸項目



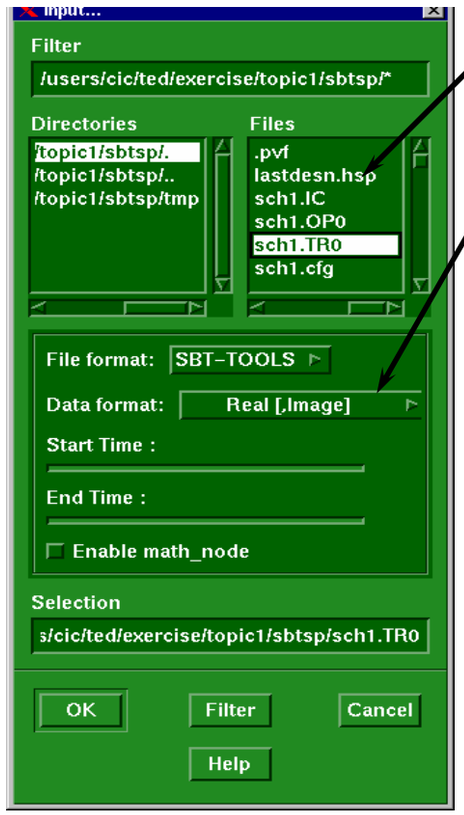
選node再按Del鍵可移除波形

右鍵設scale

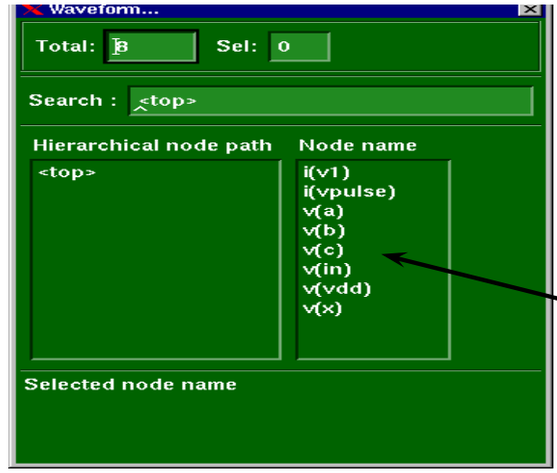
使用SBT-SPICE的SBTPLOT

%sbtplot

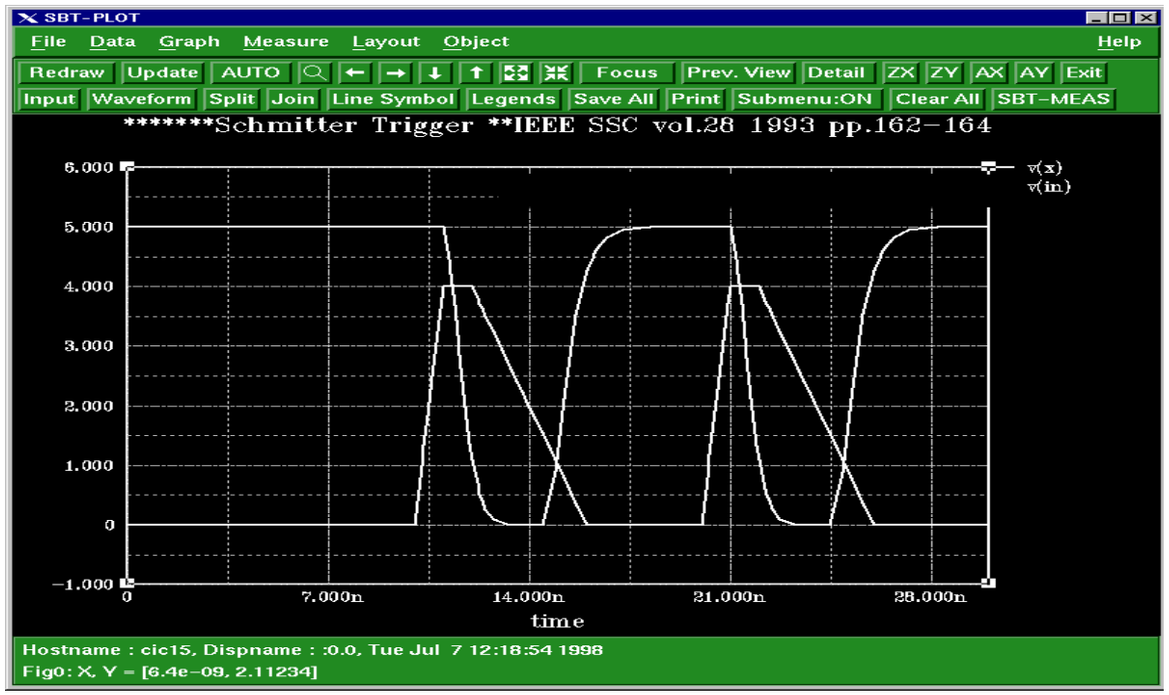
選定要看的圖形資料檔: File -> Input...



選定波形資料格式



選定欲看電流或電壓波形的節點



電路佈局及驗證

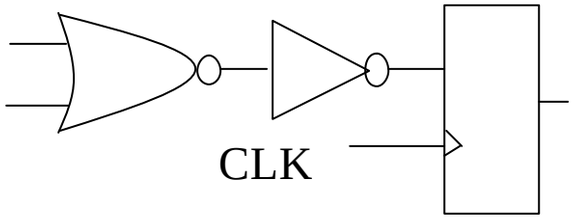
- 製程與佈局
- 佈局設計基本觀念
- 佈局設計環境
- 線上佈局驗證環境
- 批次模式佈局驗證
- I/O PAD使用簡介

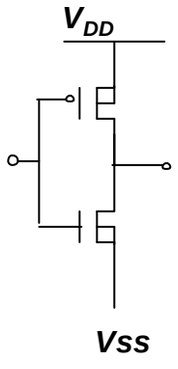
Design Abstraction

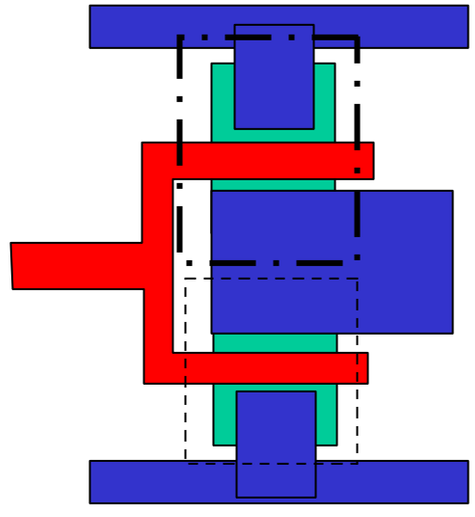
Gate and Connect
- Functionality
- Complexity

Device and Interconnect
- Performance
- Characteristic

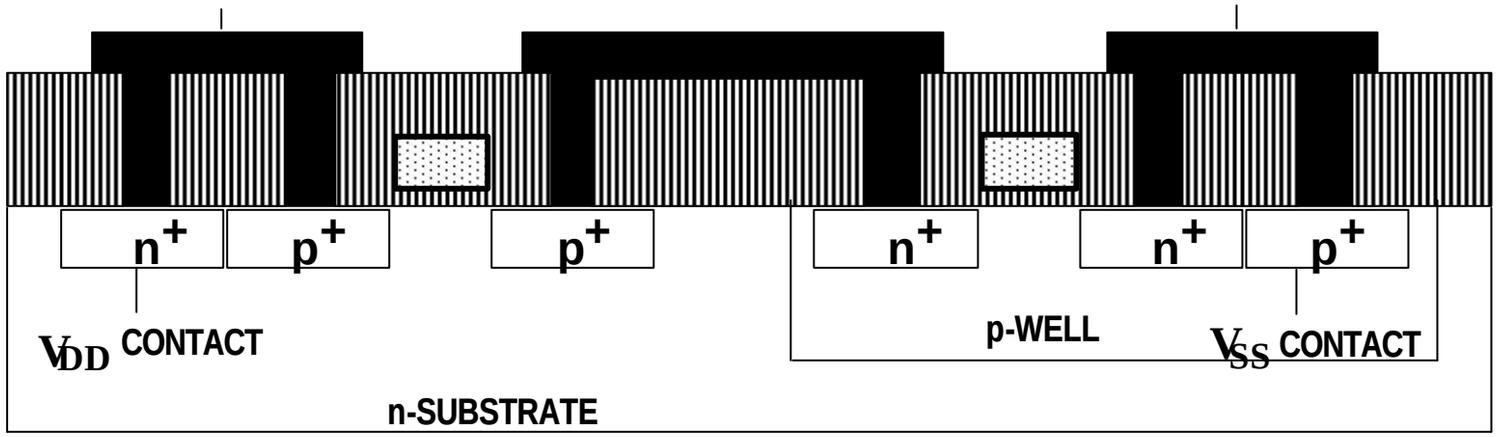
Geometry Object
- Fabrication
- Area/Cost
- Performance

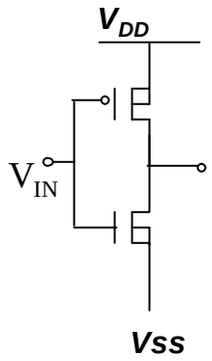


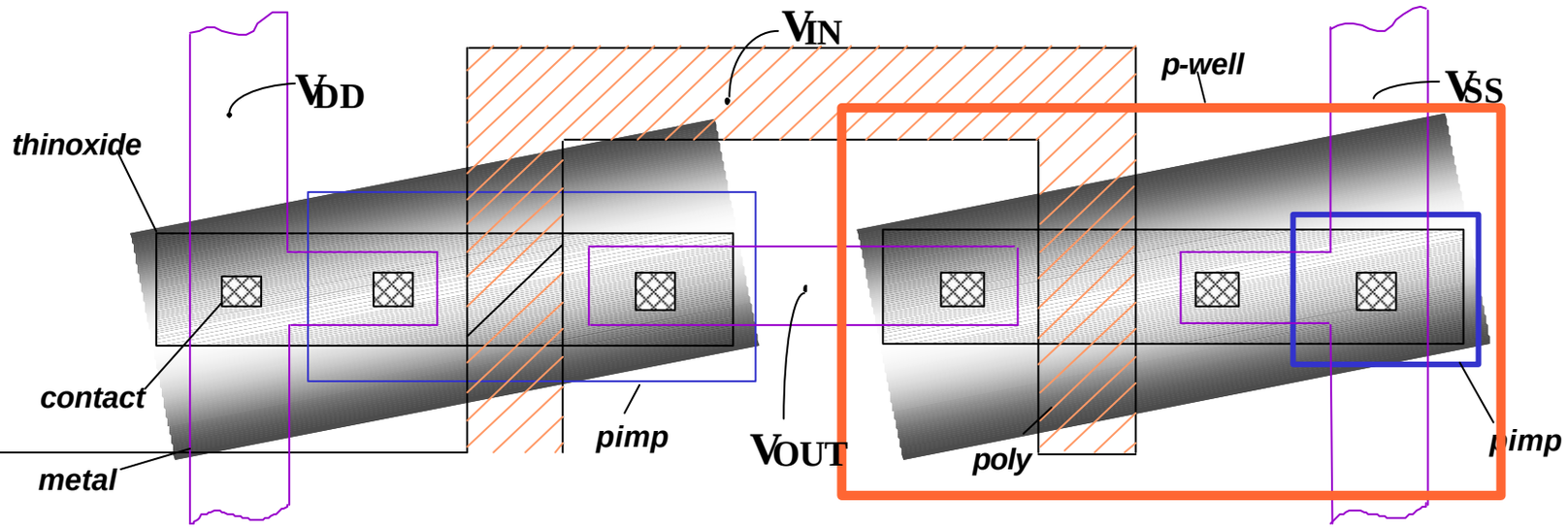




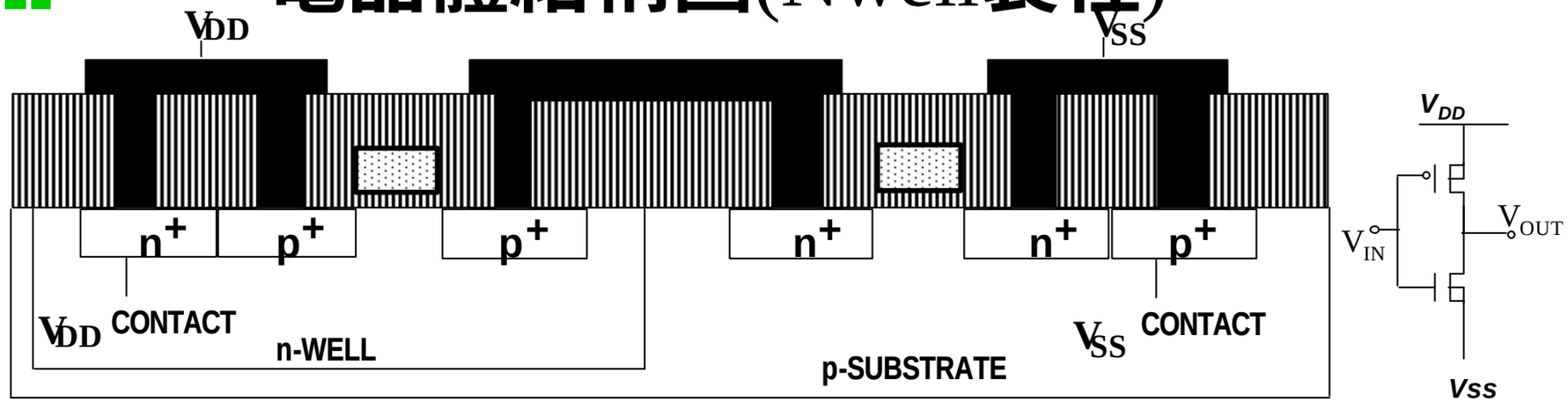
電晶體結構圖(Pwell製程)

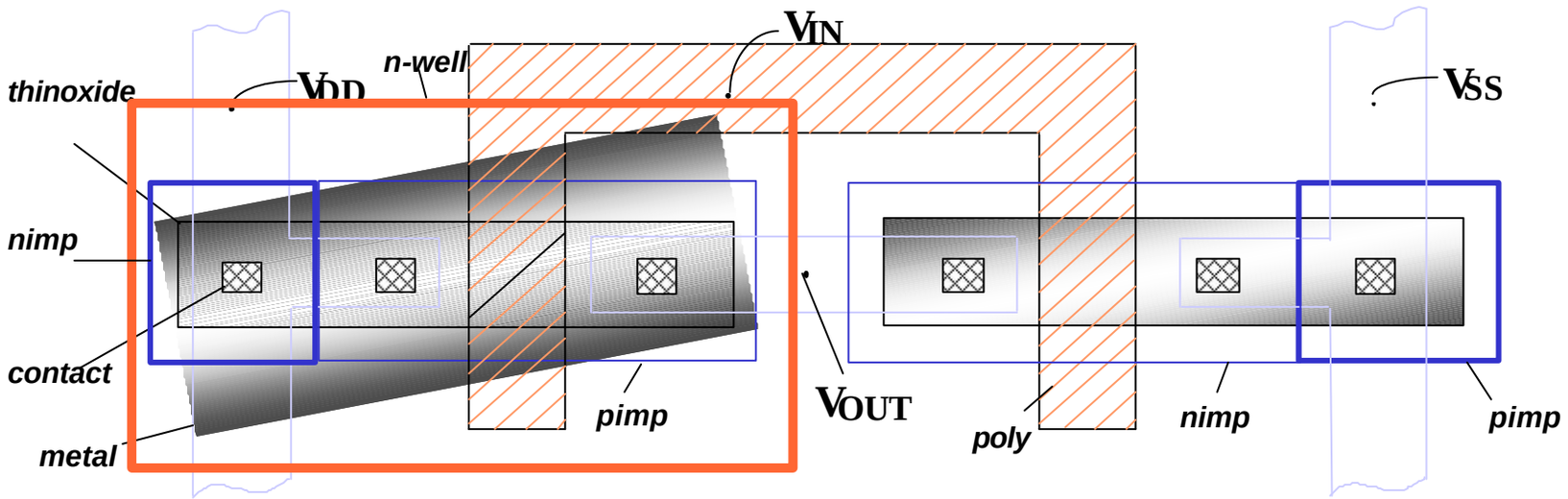






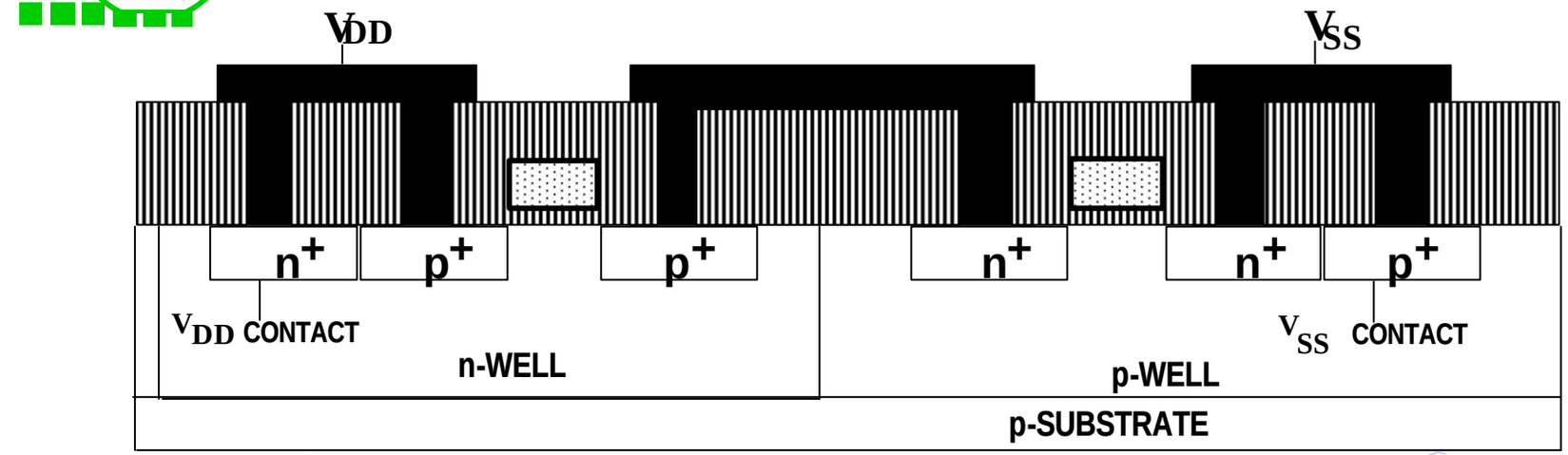
電晶體結構圖(Nwell製程)

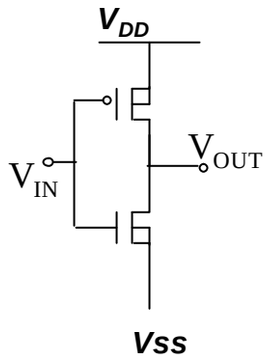


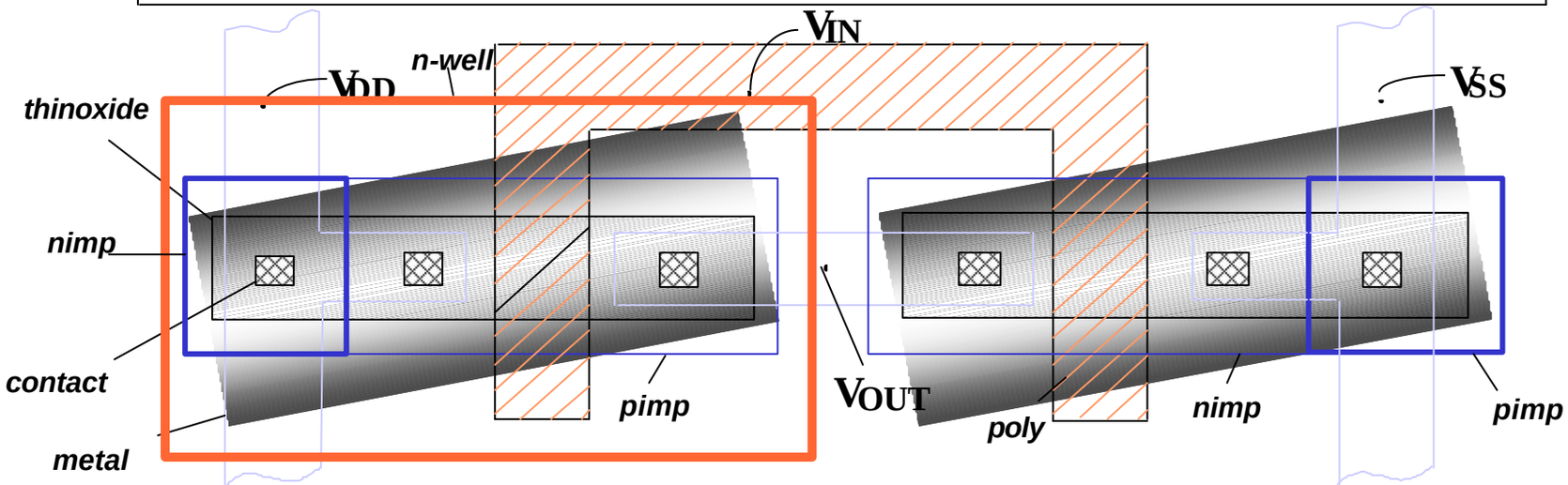


The cross-section view and layout of a CMOS(n-well) inverter

電晶體結構圖(Twinwell製程)

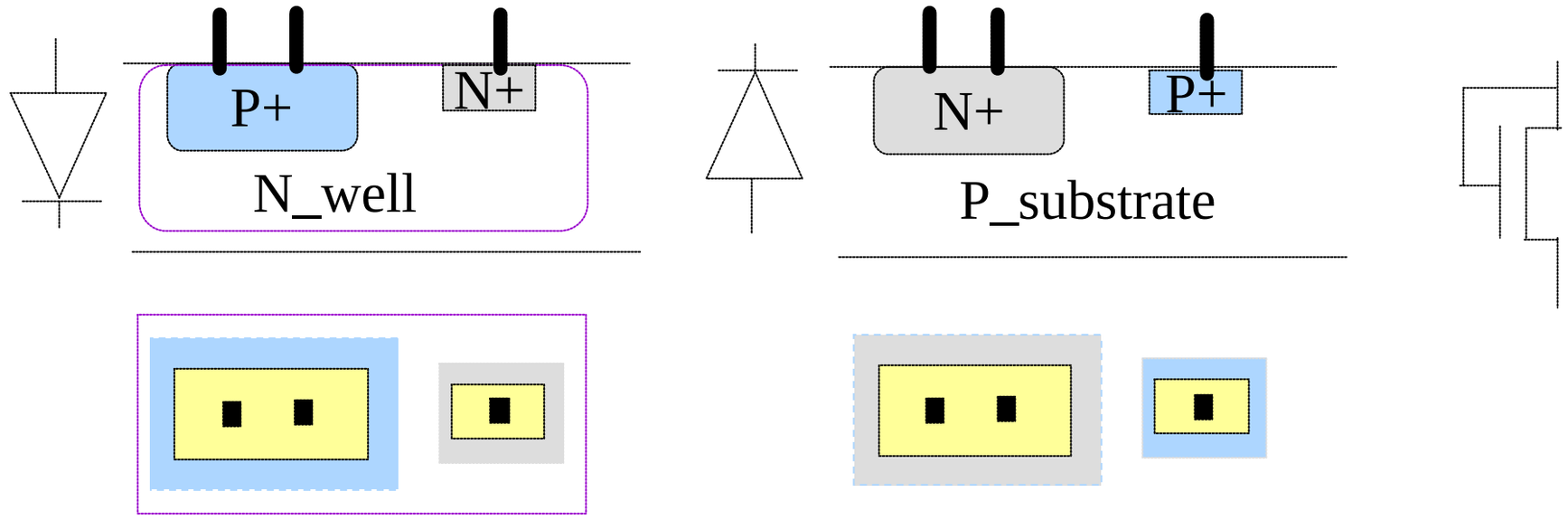






The cross-section view and layout of a CMOS(twin-well) inverter
(for 0.5um, 0.35um process of CIC)

CMOS製程中的二極體



在CMOS製程中，二極體係利用Source/Drain,Well/Substrate之接面，因此在使用上必需確保其他接面的逆向偏壓。
如使用MOS當二極體則可避免逆向偏壓的限制。

被動元件-電阻

電阻的實現須參考製程資料之單位方塊電阻值來設計,並參考其對製成之變異度

一般單位電阻值 ： Well > Diffusion > Poly > metal

電阻值的計算： ρ * L / W

其中 單位電阻值ρ : Ω/

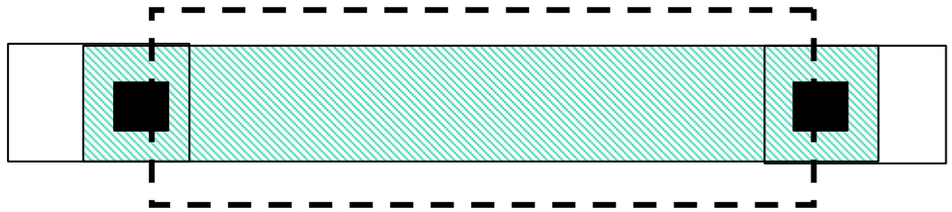
電阻應用考量：

1. **無法製作精密電阻**
2. **小的電阻值會有較大的誤差值**
3. **面積和電阻誤差值的取捨**
4. **使用電阻標示層供佈局驗證時萃取電阻**
5. **考量Contact 電阻值及 Bending 的電阻值**

電阻的佈局

Poly, Diffusion 電阻

電阻標示層的畫法需和command file **配合**



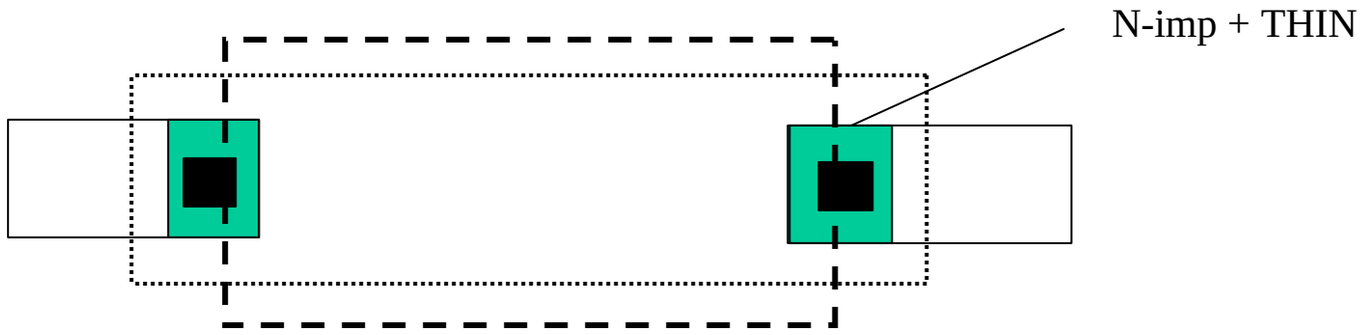
Select poly cut Resis polyR

電阻標示層

電阻值：Rs * L/W = Rs * L/(Area/ Length)

注意：0.35um**製程 之** Diffusion **電阻需加** RPO layer**以避免因**Silicide**的製作造成低電阻值**

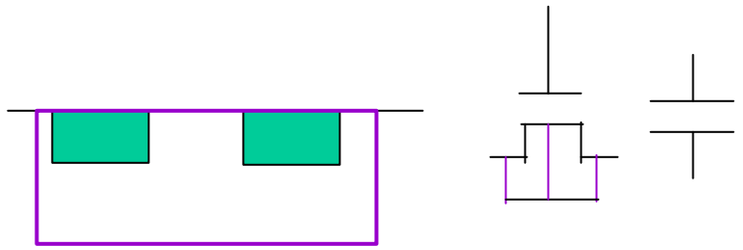
Well 電阻

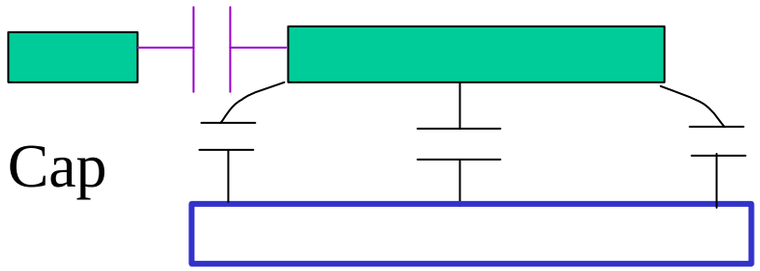


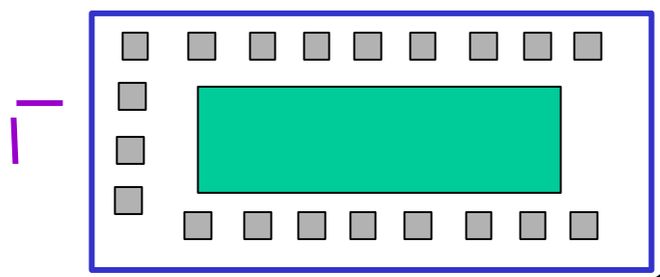
被動元件-電容

MOS**製程的電容**:

1. **接面電容**(Junction Capacitor)
   - **非線性電容**
   - **面積效益較高**(C/area)
2. **平板電容**(Plate Capacitor)
   - **電容值較固定**
   - **構造簡單**
   - **電容值包含** Overlap Cap + Fringing Cap
   - **額外光罩層可增加面積效益**
     Double poly **製程**
     MiM（Metal-insulator-Metal)**製程**
   - 1P**製程** LVS**無法處理電容**
     加 Dummy layer，**修改**commamd file





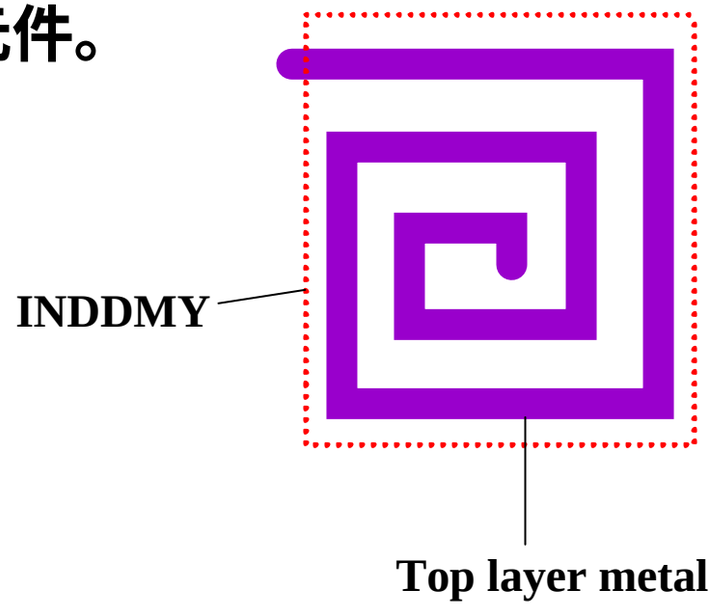


被動元件-電感

在MOS製程中電感並非可良好控制之元件。

CMOS電感之主要問題：

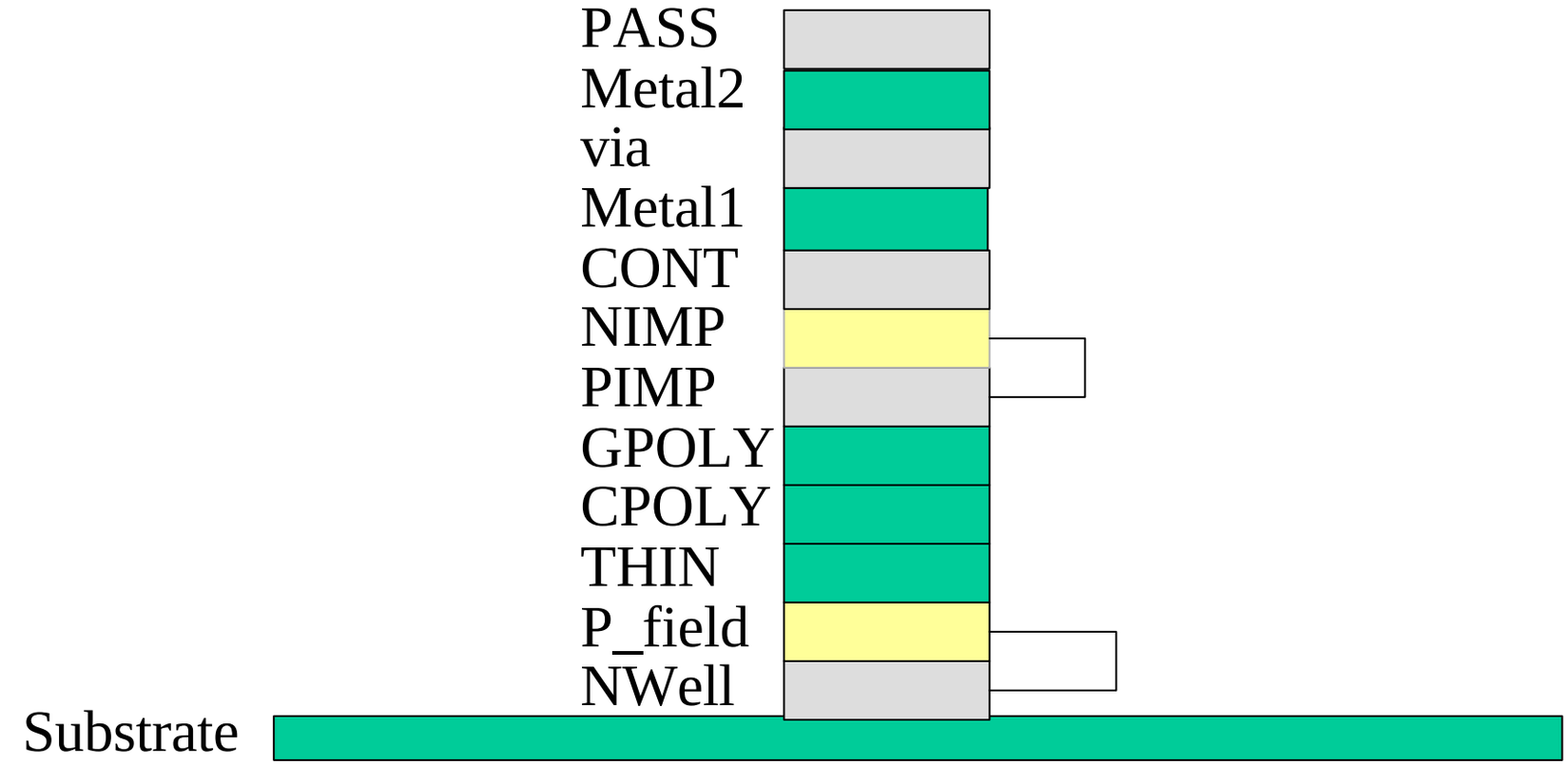
1. 電感值小
2. Q值小
3. 元件模型未充分建立，現階段需自行建立等效模型，RLC model



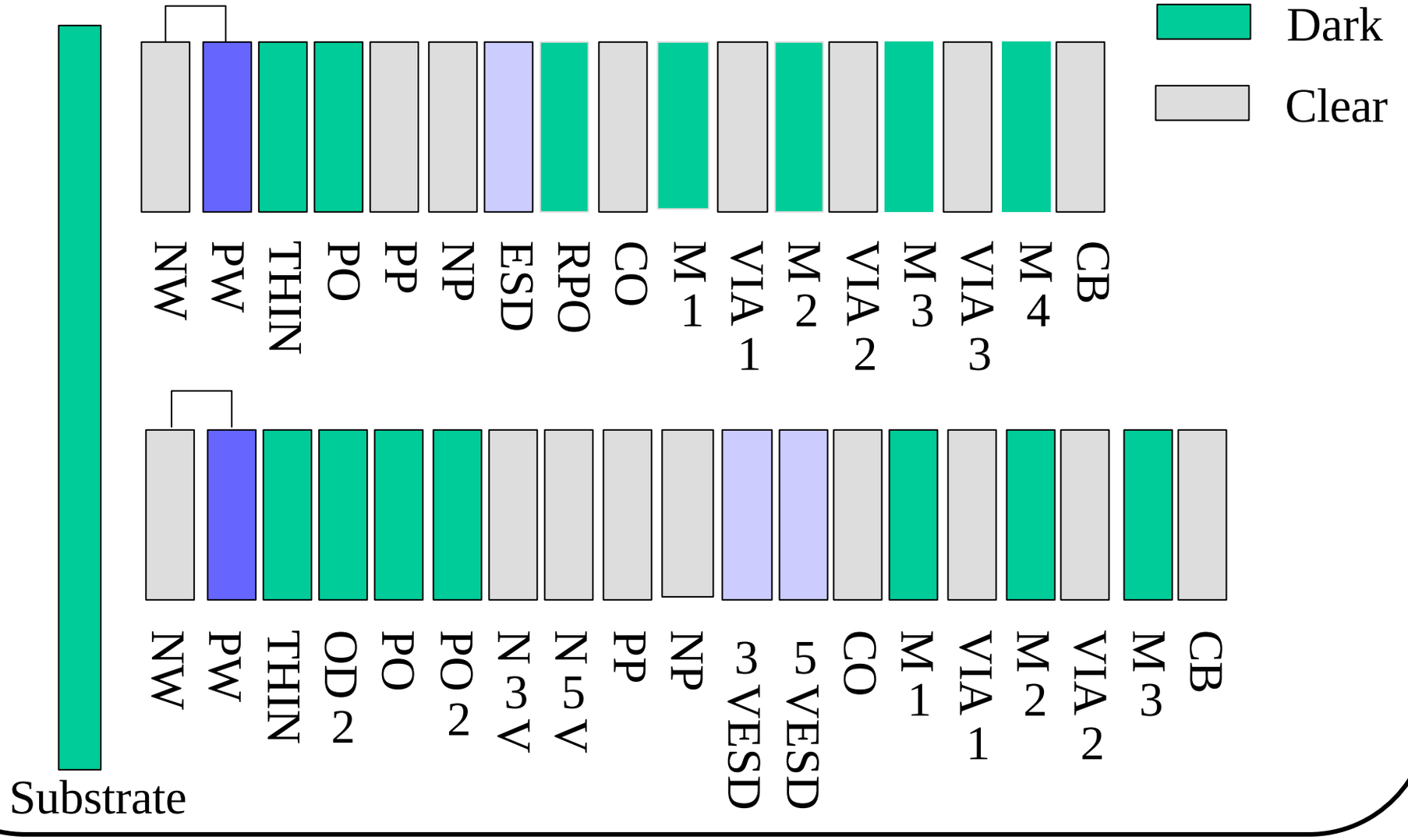
電感之佈局：

1. 使用較小片電阻之連接層來設計電感(Thick Top metal)
2. 運用多層並聯增電感值
3. 目前LVS/LPE 需特殊處理

佈局層次與製作對應(0.5um製程)



佈局層次與製作對應(0.35u)



佈局規劃與步驟

- 佈局係定義各元件的位置大小及相關的連線
- 佈局的設計應考量製程的變異對電路特性的影響
- 為提昇佈局的效率，事先規劃水平及垂直走向的線層
- 為確保電路工作的獨立性及正確性，適當加入隔離及遮蔽功能之電路或架構

佈局設計通常包含：

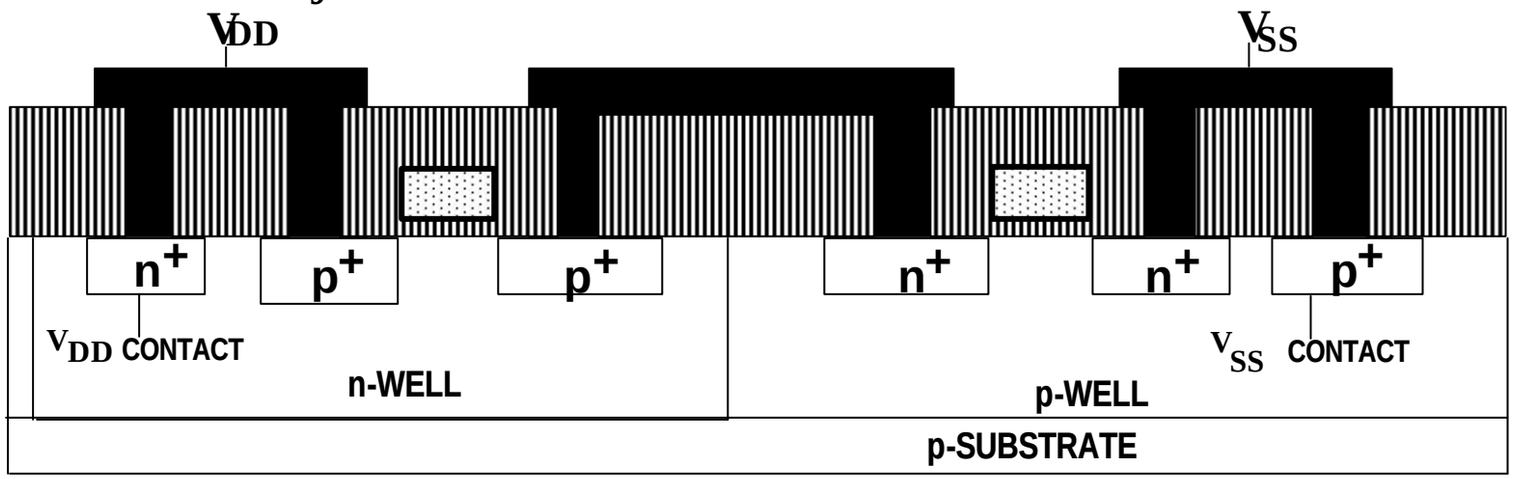
1. Block partition
2. Block placement(Pin location and orientation)
3. Device placement and connection
4. Block connection
5. I/O Placement and Connection

佈局設計次序

- Device floor-planning and placement
  - Symbolic draw the transistor placement and routing channel
- Device definition and connection
  - Draw THIN + POLY for transistor definition
  - Draw Metal + Contact for device connection
  - Draw P-Implant / N-Implant for NMOS/PMOS Source/Drain
  - Draw Well for Completeness
- Isolation and protection
  - Add enough Well contact/ Substrate contact
  - Add guard ring

Well/Substrate Contact

Connecting well/substrate to appropriate potential for correct source/drain junction bias

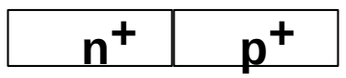


Keep each well to be connected

Keep each well in same potential

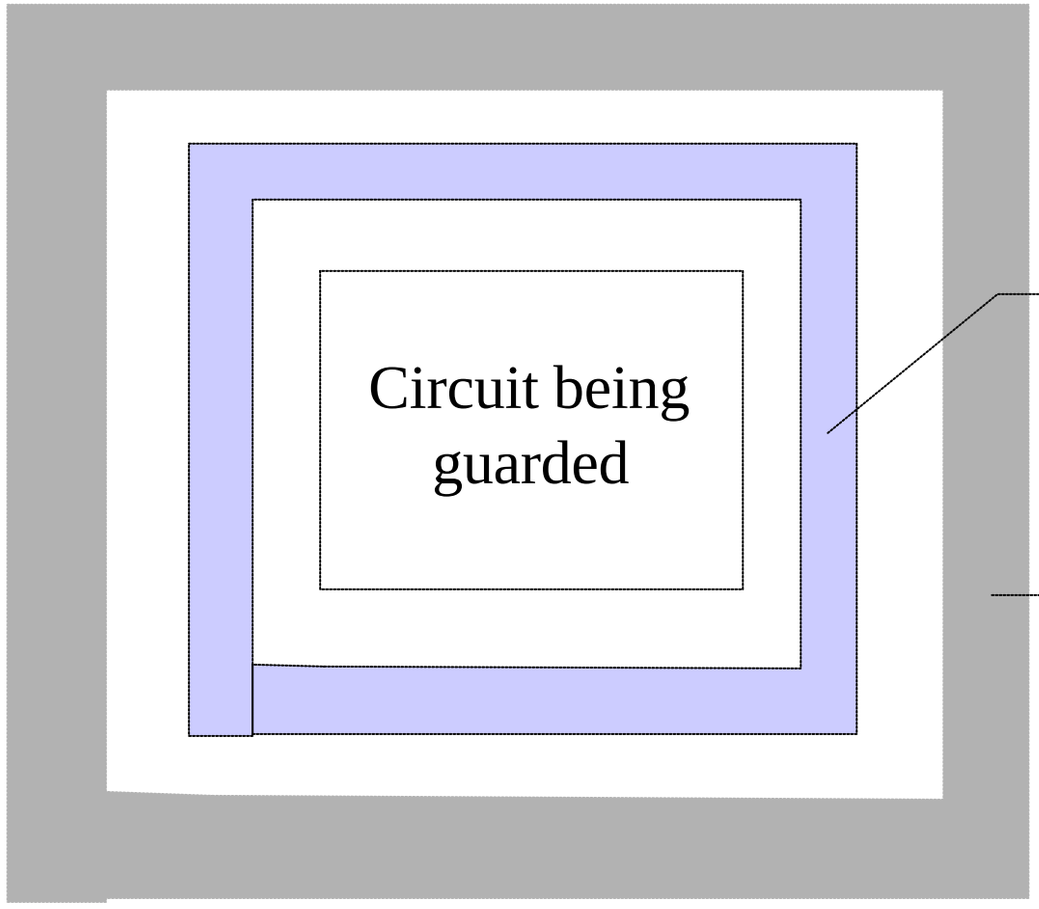
Keep well/substrate contact as many as possible

Some process might not allow butting contact



Butting contact

Guard Ring



Collecting injected current for reducing noise into internal circuits
If possible, keep the gnd and vdd clean

P+ guard ring
to ground

Layers include:
thin, p-plus,contact, metal

N-well guard ring

To dc potential
Layers include:
N-well, thin, n-plus,contact, metal

I/O Placement

- **為提供封裝接腳到晶片內部的連線，需加上**PAD
- **為提供足夠的驅動能力在**output pad**前通常加上驅動電路**
- **為提供內部電路的保護，在**input pad**後加上保護電路。**
- **為提供晶片外部及晶片本身訊號位準的相容性，**input pad **後可加上**level shifting **的電路。**
- **為提供驅動電路及保護電路的電源，**I/O pad**處需有電源，為避免內部電路受**I/O **訊號的干擾，**I/O **電源及** CORE **電源最好分開。**
- Output pad **因需提供較大的驅動能力，因此一組**power**最好只供應不超過** 8 **個會同時動作的輸出。**
- **為求偵錯的便利性，可在需要觀察的訊號上加上** probing window(**即加上** PASS **層)。**

佈局設計考量事項

- 佈局設計之考量因素
  - •降低元件特性變異
    - 保持元件參數值的變化量在適當範圍
  - •保持元件間之匹配
    - 使相關元件具同樣之變化趨勢(如溫度、幾何環境相似，隔離，接觸點等)
  - •降低雜散效應值
    - 縮短信號線長度，減少耦合電容
  - •提供充裕之current density margin
  - •注意雜訊之隔離
    - Signal shielding, device isolation
  - •減少整體晶片之面積

佈局設計驗證環境

佈局軟體 - Virtuoso

- Layout Editor in Design Framework II
- Shape based layout editing - Use polygon, rectangle, path, circle to define the device and connection.
- 255 definable layers for use, only part of them are meaningful to the fabrication.
- Hierarchical layout editing with edit-in-place
- Definable I/O pins for online layout verification
- Import/export layout data file to/from layout database
- Similar tools include Laker(思源) and IC Station(Mentor-Graphic)

線上佈局驗證- diva

- **為提昇佈局驗證的效率，不要在完成整個設計後才進行** DRC/LVS **等佈局驗證。**
- Incremental / hierarchical layout verification
- **利用線上佈局驗證軟體以早期去除問題。**
- **搭配線上佈局驗證軟體所提供的**probing **功能可顯示錯誤發生的地點及原因。**
- **需提供製程之規則檔**

  divaDRC.rul : rule file for design rule checking

  divaERC.rul : rule file for electrical rule checking(short/open…)

  divaEXT.rul : rule file for device definition

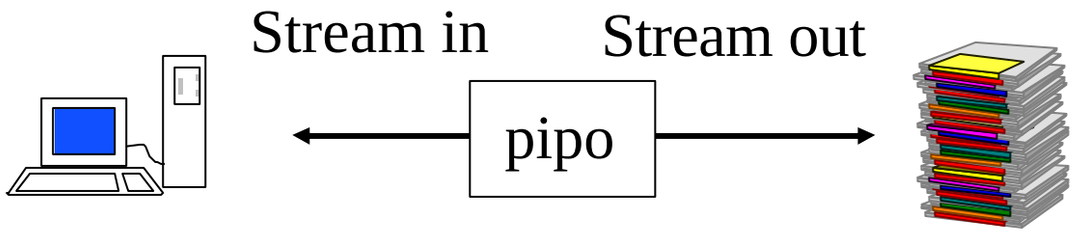
  divaLVS.rul : rule file for layout/schematic cross checking

佈局檔格式與佈局輸出

- **為提供共通之佈局交換環境，因此存在幾種佈局檔格式：如**GDSII, CIF,Applicon**等**
- **業界採用最廣之共通格式為** GDSII (Stream Format)
- **在**Cadence**環境中係透過** Straeam in/Stream out **來讀入/寫出** GDSII Stream data
- **而實際執行之指令為** pipo **程式**

  pipo strmin *template_file_name*

  pipo strmout *template_file_name*



批次模式佈局驗證

- GDSII檔為設計者與晶圓廠之唯一媒介，為確保佈局檔的正確性，必須有完整的驗證步驟。
- 目前採用Dracula 或 Calibre進行佈局驗證。
- Dracula包含一系列指令群，透過前處理器的處理將命令檔內容轉換成一程序檔，佈局驗證時依程序檔之內容依序執行各項指令。
- 為求簡化複雜度，各項製程一般提供三種命令檔

  DRC/ERC command file

  LVS command file

  LPE/PRE command file
- Dracula 的驗證結果可透過Dracula Interactive環境進行偵錯。

Cadence Layout 設計環境

啟動layout editor **的指令有**:
icfb : Full IC design environment
layoutPlus : layout editor + diva
layout : layout editor

Memory usage

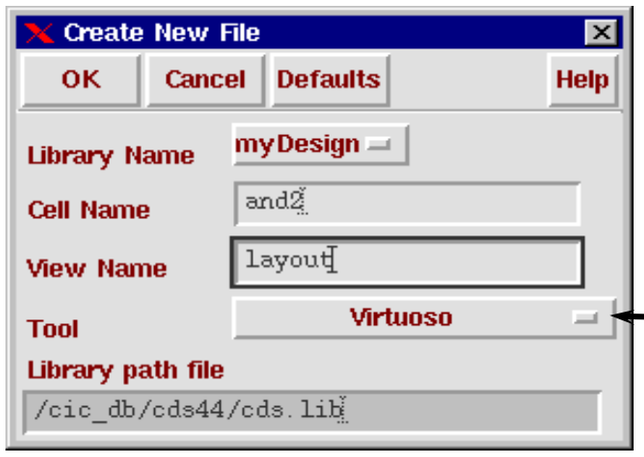
開啟 layout view **時，系統需要有** display.drf **的定義，若系統找不到** display.drf **檔，或在該檔內沒有所用到的**layer **定義時，則系統會提示要求**merge display.drf **選擇** Tools->Conversion Tool box **項下的** Merge Display Resource Files

Create Cell Layout View

Layout Editor

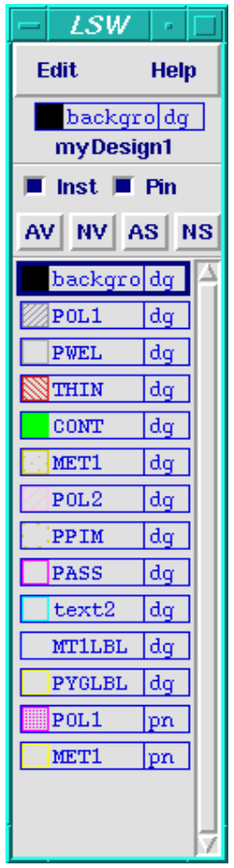
在這裏將介紹 Layout Editor **的使用**, **並將** layout **好的** 2 input and**用** on-line interactive verification tool (DIVA) **作**DRC, LVS

File -> New -> Cellview



指定 cellview **的編輯程式**

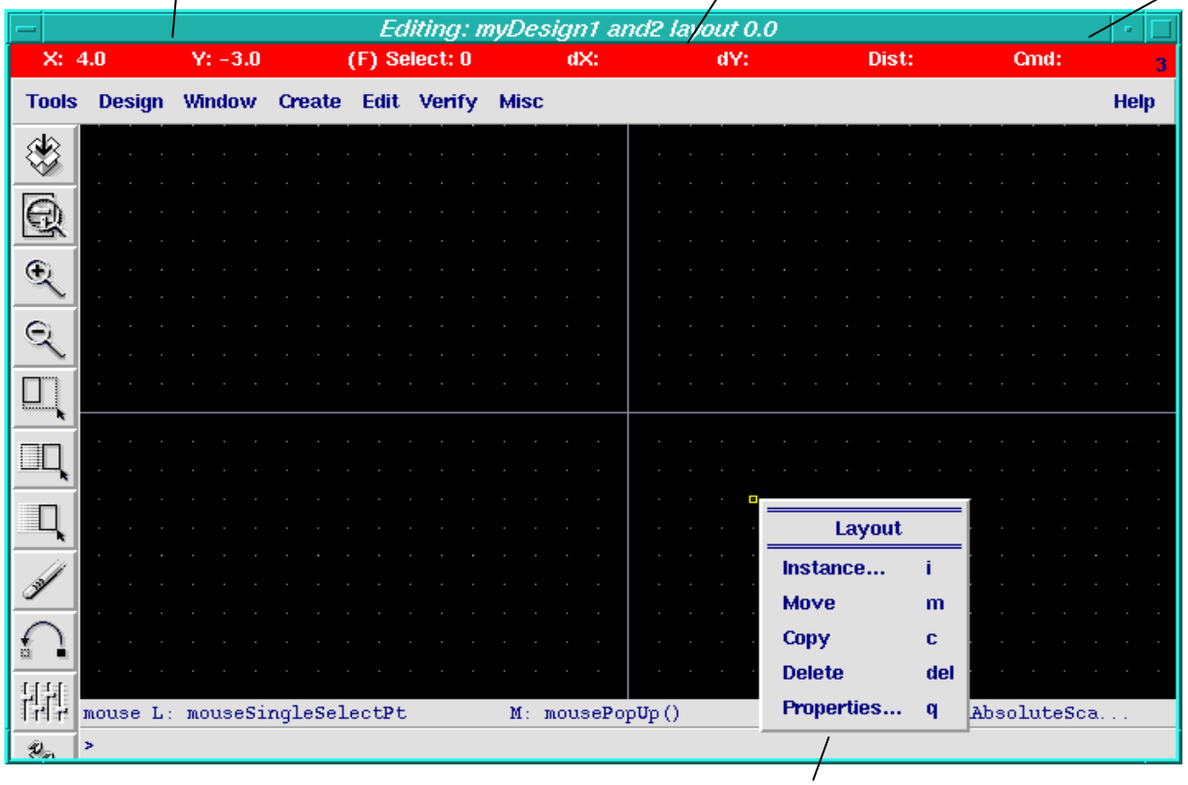
Layout Editor Window



mouse **位置**

與上一點的相對位置

目前command



按 mouse **中鍵切換此** window**以方便選用**

加選或少選 (Shift or Ctrl)

Layout Editor Menus

keyin z**功 能同**Zoom in(z**為其**bindkey)

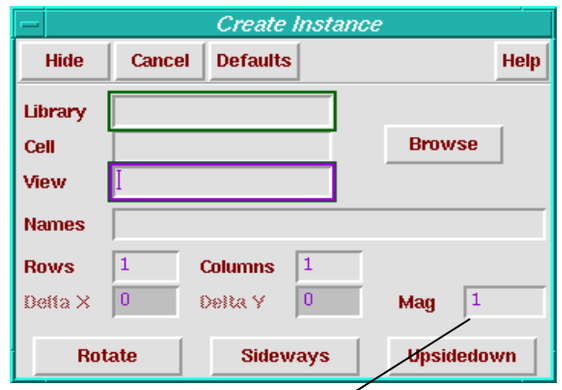
layout editor **的主要選項與其相對** bindkey

Create and Design Menu

放大 mouse **所選區域**
放大兩倍

縮小兩倍
以 mouse **點 處為中心顯示** layout view
使整個 layout view **均** show **於** layout window **中**
fit **到前一個** edit **處**
重show
create **一個包 含** current window**圖 形之** new window
存取某些特定的image (**在**layout view **中**)

畫矩形
畫多邊形(最後一點 double click)
畫長條接線
畫文字 標示
將已畫好之 cell**引用到此**layout view
畫 tech. file **中已定義好之**contact
畫 pin(**用以給定**node name)
畫圓, **橢圓**,**中空 圓**

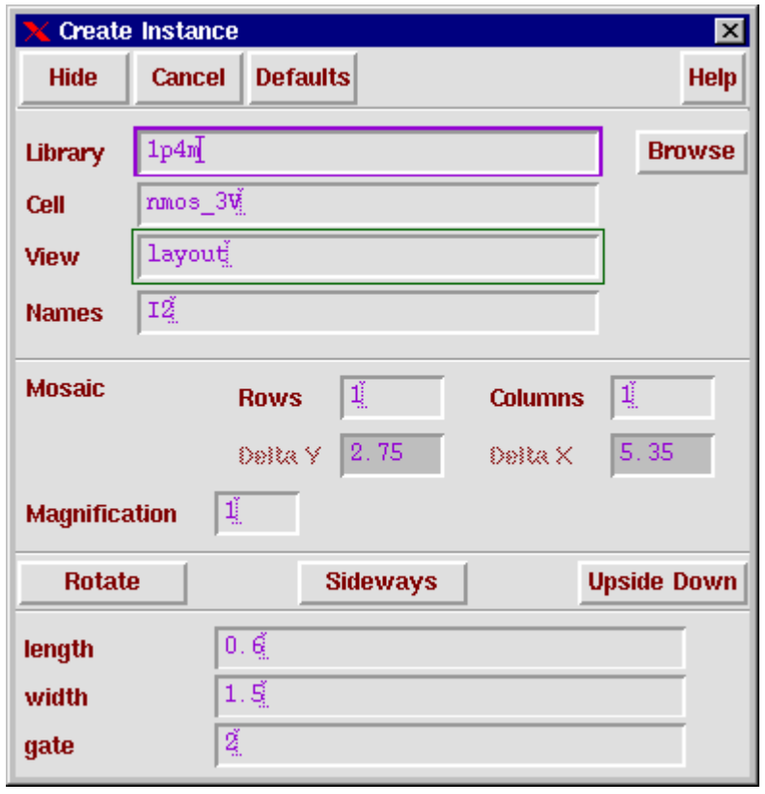


放大倍數(須正數) **>1 放大**
<1 縮小

Using Pcell in layout editor

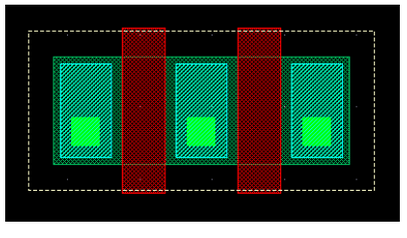
Pcell - parameterized layout cell

Use **create instance** menu
Specify length/width/number of gates

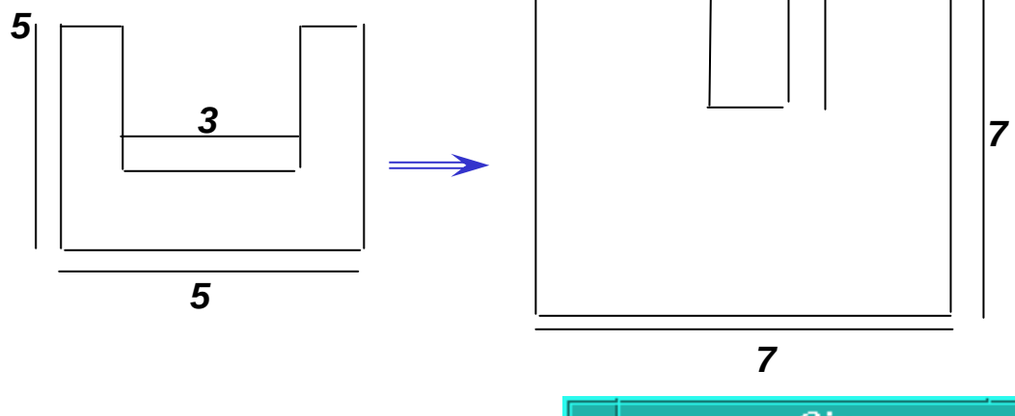


Available pcells in 0.35um process

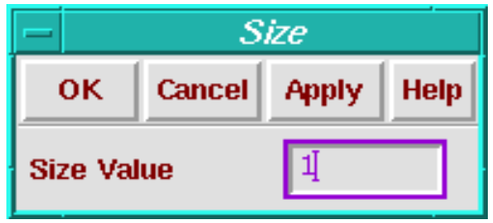
nmos_3V , pmos_3V



Edit Menu



邊緣或角落擴展
調整 object **邊緣處**



{ + **擴大**
- **縮小**

砍去 object **部份圖形以調整形狀**

合成一體(**須同** layer)

將mouse **所選部份存成一個** cell

flatten instance, array **使與目前同** level,**而能** edit **之**

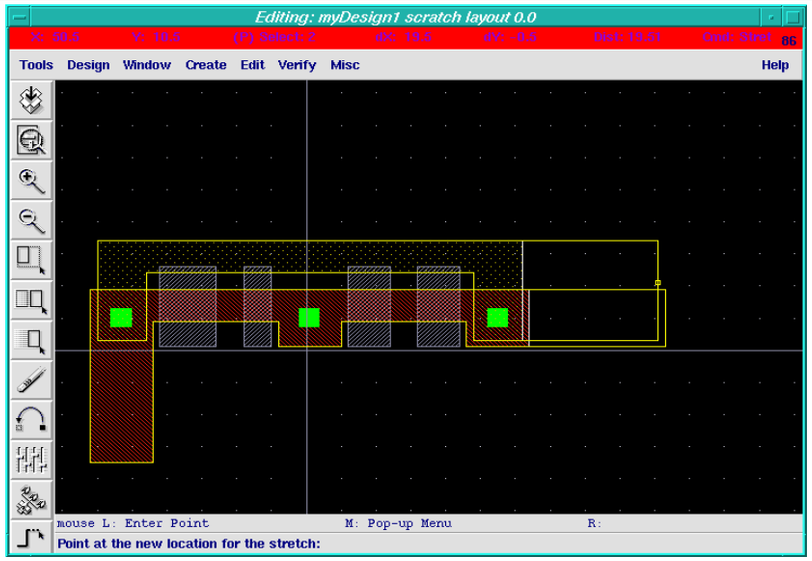
corner **調變**

擴大或縮小某 object

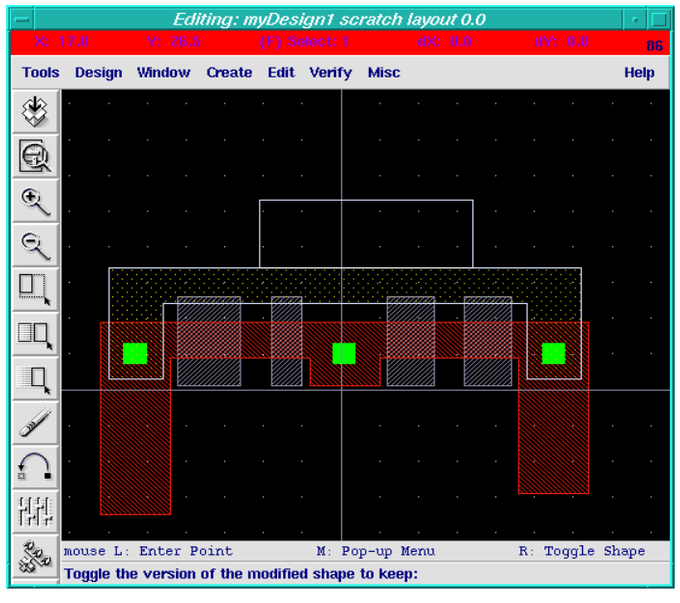
定分割線使object partial stretch

使一object **附屬到另一**object,**當一**object move, copy, delete **時**,**其附屬** object **同時動作**

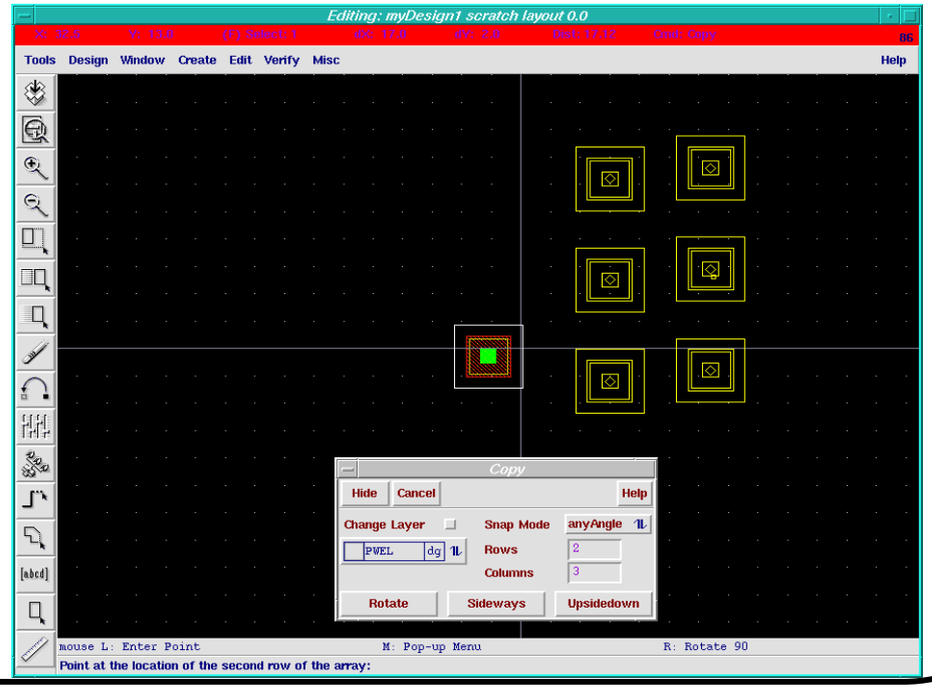
將rectangle, path, circle, ellipse,donut **轉成多邊形**



Stretch



Reshape



Display Control Wndow

Design-> Options-> Display

對cellView存目前設定情況

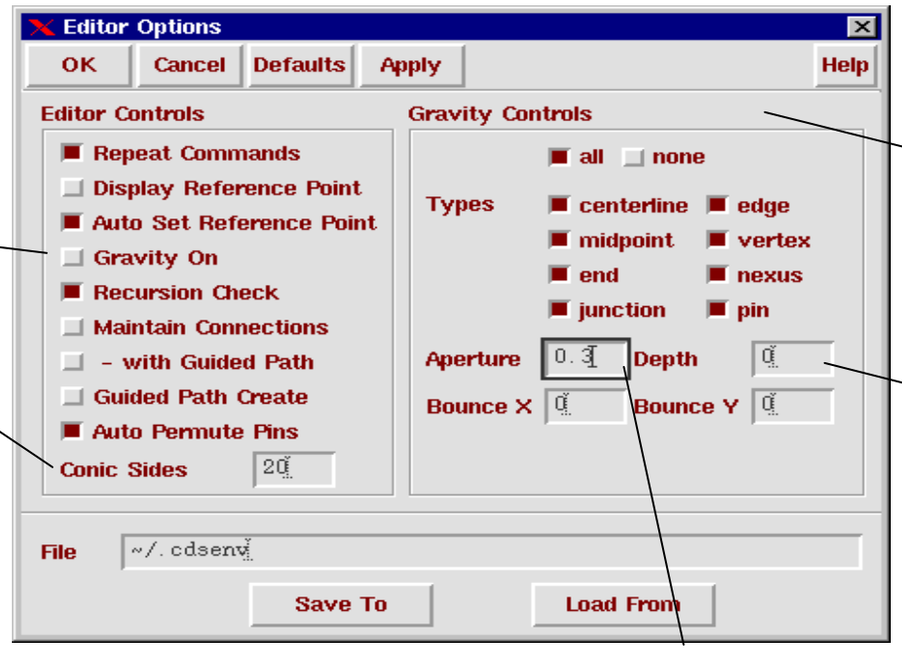
set grid **顯示方式**

set minor grid **間距多少** user unit
set major grid **間距幾倍於** minor grid
set X **軸移動之** min. **間距**
set Y **軸移動之** min. **間距**
(**以上二值之設定須為**design rule **之公因數**)

set **畫線時** cursor **之 跳動方式**(snap)
set **畫線時之限制方式**

Editor Option Control Window

Design-> Options-> Editor



set cursor **靠近** object **時即被吸到** object **邊緣**

set conic **經過** Convert To Polygon **或** Merge **後變成幾邊形**

set gravity on **時所作用之** object

gravity**作用深度**

set gravity on **時能影響之範圍為幾** user unit **以內**

Adding Label and Pin

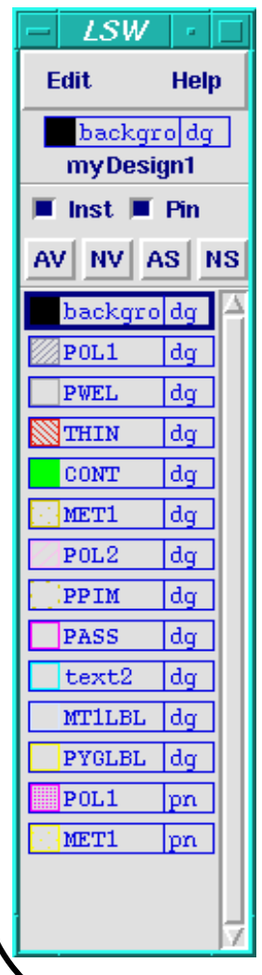
為使 DIVA LVS check **時認得** layout **之** terminal(node) name, **在用** MET1 dg layout **之** terminal **上,選用** MET1 dg ,

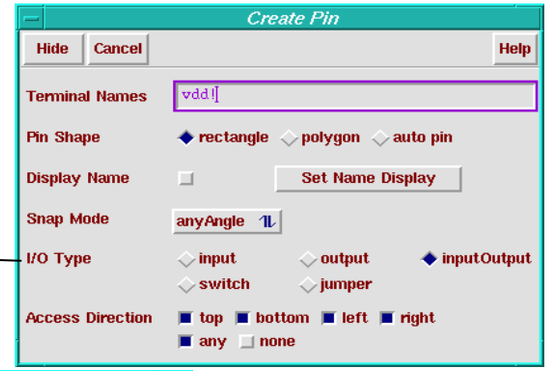
Create-> Pin...

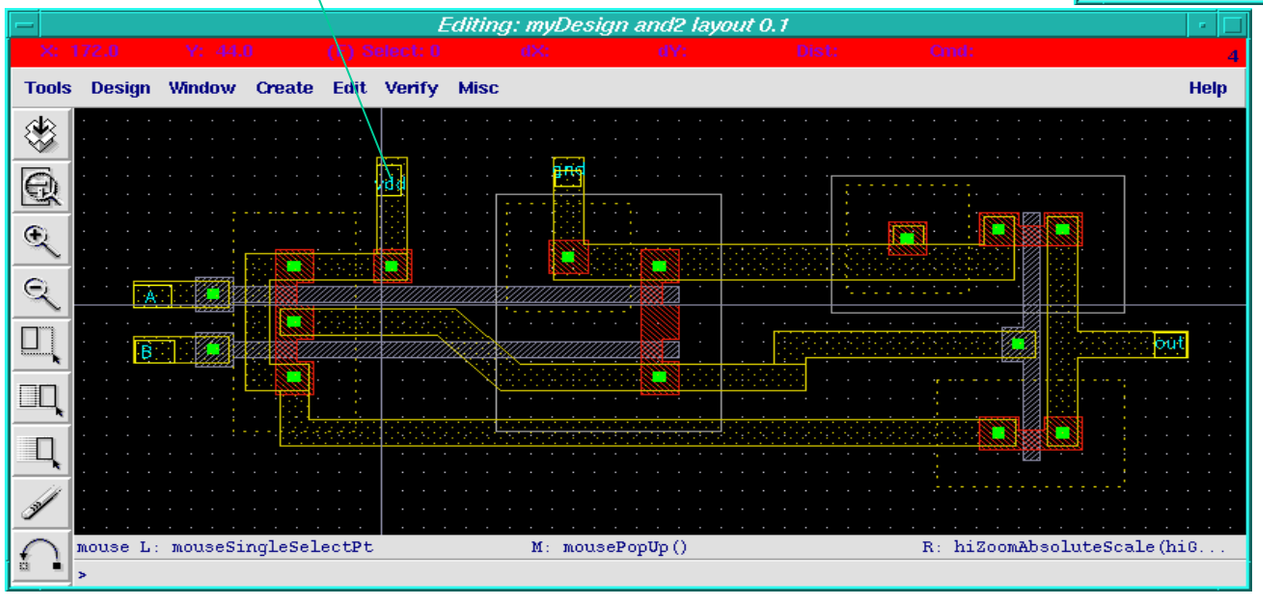
(note:**此層將視同**layout**製作**)

layout**一層**

(**選 適當的** I/O type)



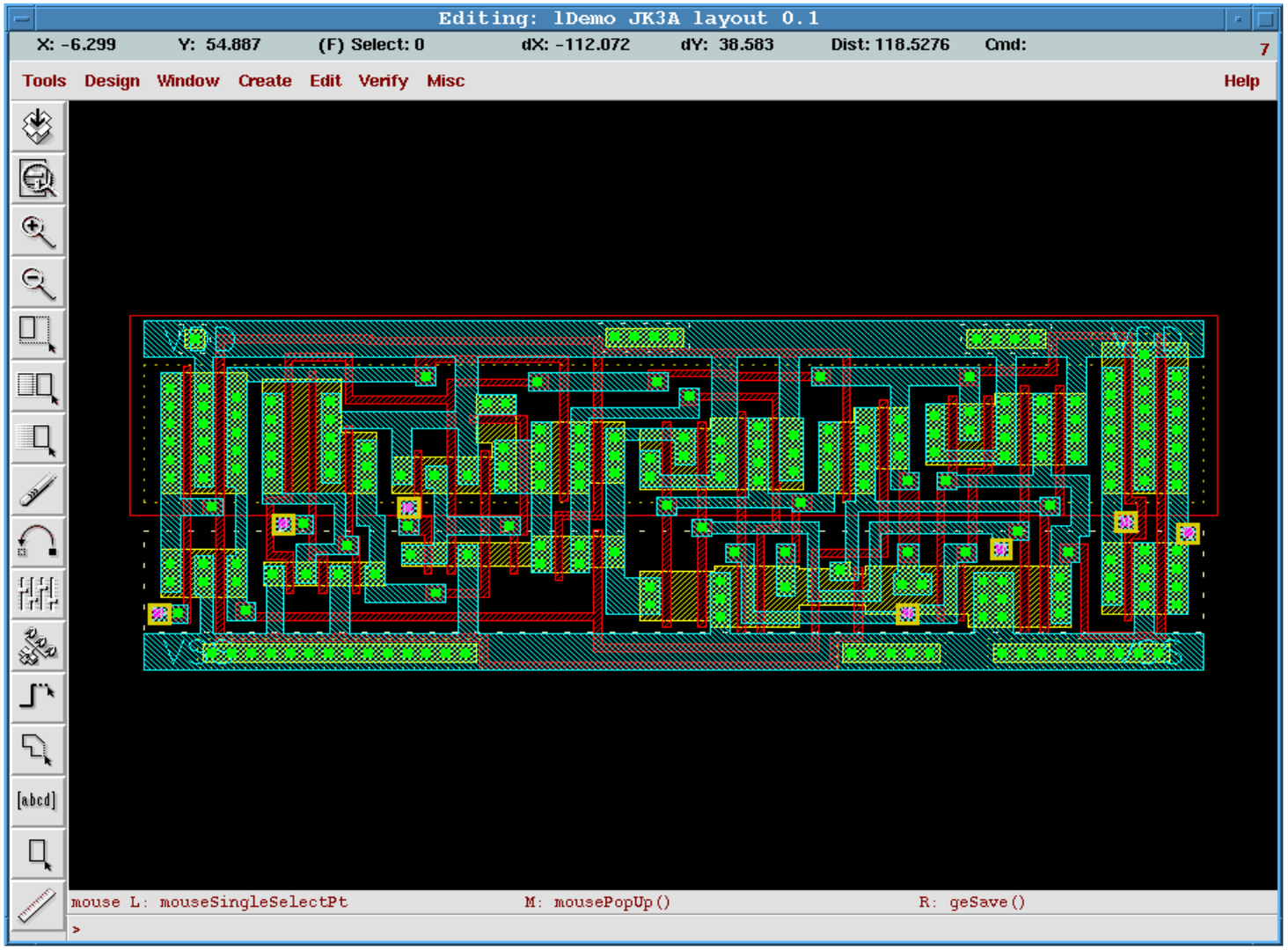




Pin : Diva
Label : Dracula

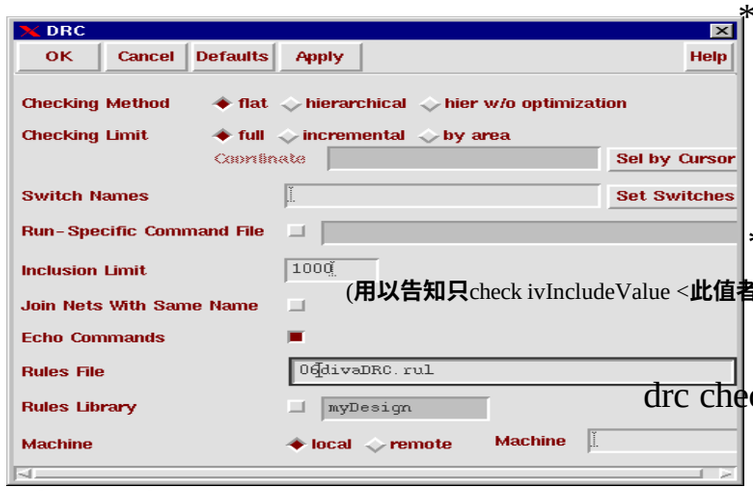
A, B, out, vdd, gnd **為選用** text2 dg Create-> Label...**標示於** terminal **上**,
除為 user **本身認知用外**, **亦使作**Dracula check **時認知此** label

舉例 JK Master-Slave flip flop Layout view **如下**:



On-line 佈局驗證器 (DIVA)

在 technology file 完備的情況下, Diva 可 作DRC, ERC, LVS check 與 LPE, 在此僅作 DRC, LVS check design rule check:
Verifty-> DRC...



**未設定時default用drc?

**同net name者視同相接

**指定tech. file library

drc checking rule

**尋找違反rule(相對error message 'cont')處:

```
Edit->Search
Zoom To Figure
Search for any shape in current cellview
property  drcWhy  cont
```

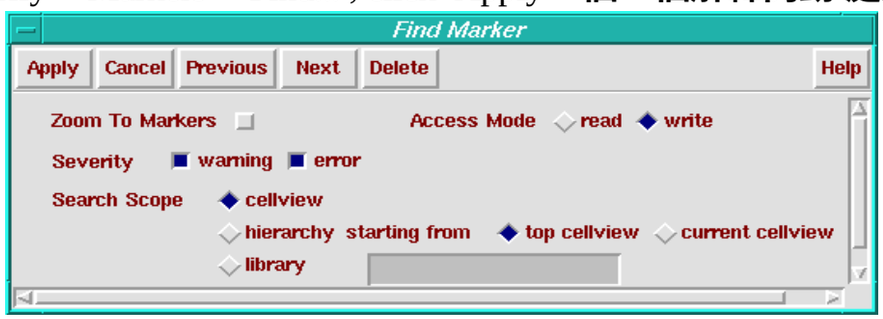
**設定DRC時Inclusion Limit:

(用以告知只check ivIncludeValue <此值者,default為0)

Design->Design properties=>property
Add Name: ivIncludeValue
Value: 2
Type: Int

software check結果show 於 CIW中

software check 後, 會在 cell 之 layout view 上用閃 動的方式顯示違犯 design rule 之處,
可用 Verify->Markers->Explain , click 於閃動處以得悉所違犯之 rule,
或用 Verify-> Markers-> Find... , click Apply一個一個解釋閃動 處所違之 rule

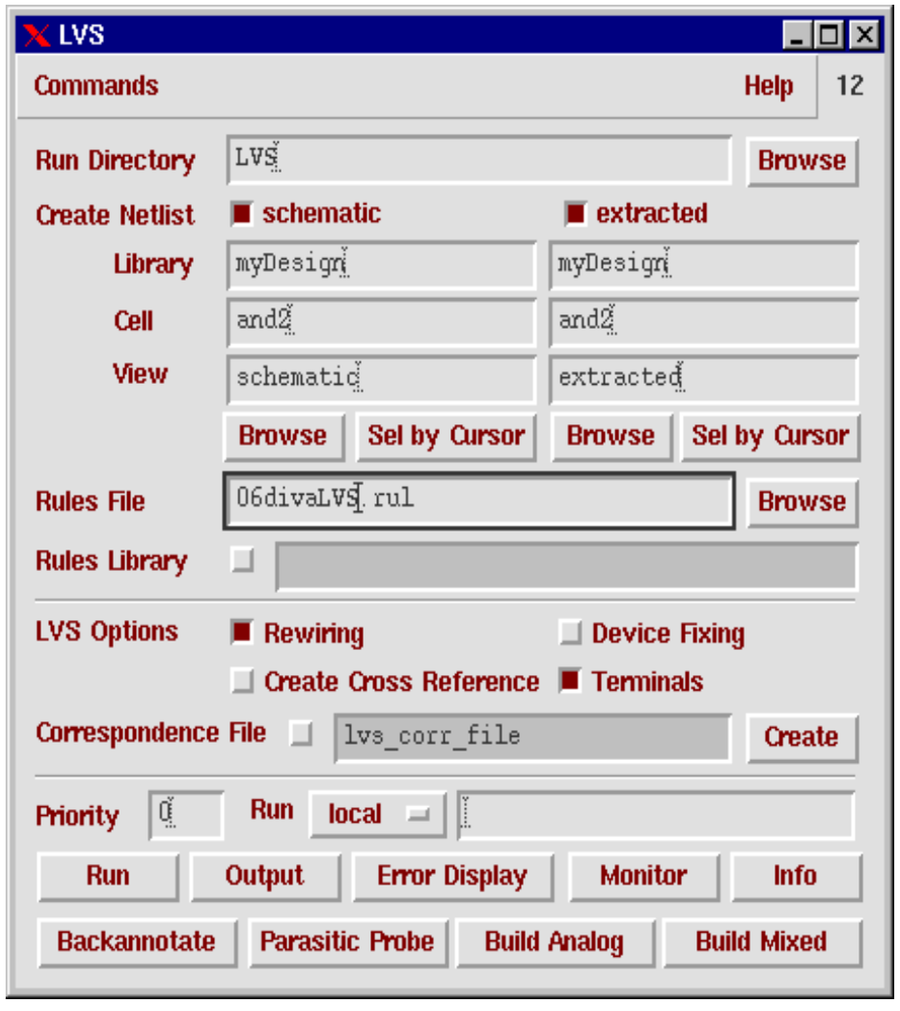


根 據 CIW上之 message 修改 layout 圖, 並作 DRC no error

在extracted view 中, Verify-> LVS...

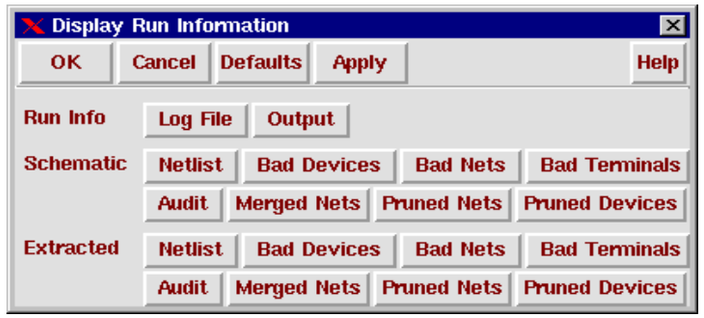
指定LVS rule file

指定tech. file library



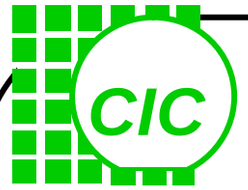
Click Run, software 作on-line LVS check, 比對所指定的schematic view與由layout得的extracted view之 netlist

在on-line LVS run 結束後, 會自動顯示 job has failed or succeeded
LVS window 上, click Info



Click Run Info的Log file可 看到 software run的 訊息
Output 可 看到 LVS run的 結 果, 以下為 Output file(si.out)內容

```
LVS version 4.4.1 Mon May 2 23:37:50 PDT 1998 (cmrd1)
Library is myDesign
Path is . ~ /cds44_sun/cadence/tools.sun4/dfII/etc/cdslib/artist
Like matching is enabled.
Net swapping is enabled.
Using terminal names as correspondence points.

  Net-list summary for /user/usr6/hywang/Labs/06sptm/LVS/layout/netlist
        count
            7                   nets
            5                   terminals
            3                   pmos
            3                   nmos

  Net-list summary for /user/usr6/hywang/Labs/06sptm/LVS/schematic/netlist
        count
            7                   nets
            5                   terminals
            3                   pmos
            3                   nmos
  Terminal correspondence points
```



```
    1              A
    2              B
    3              out
    4              vdd!
    5              gnd!
The net-lists match.
```

比對 匹配

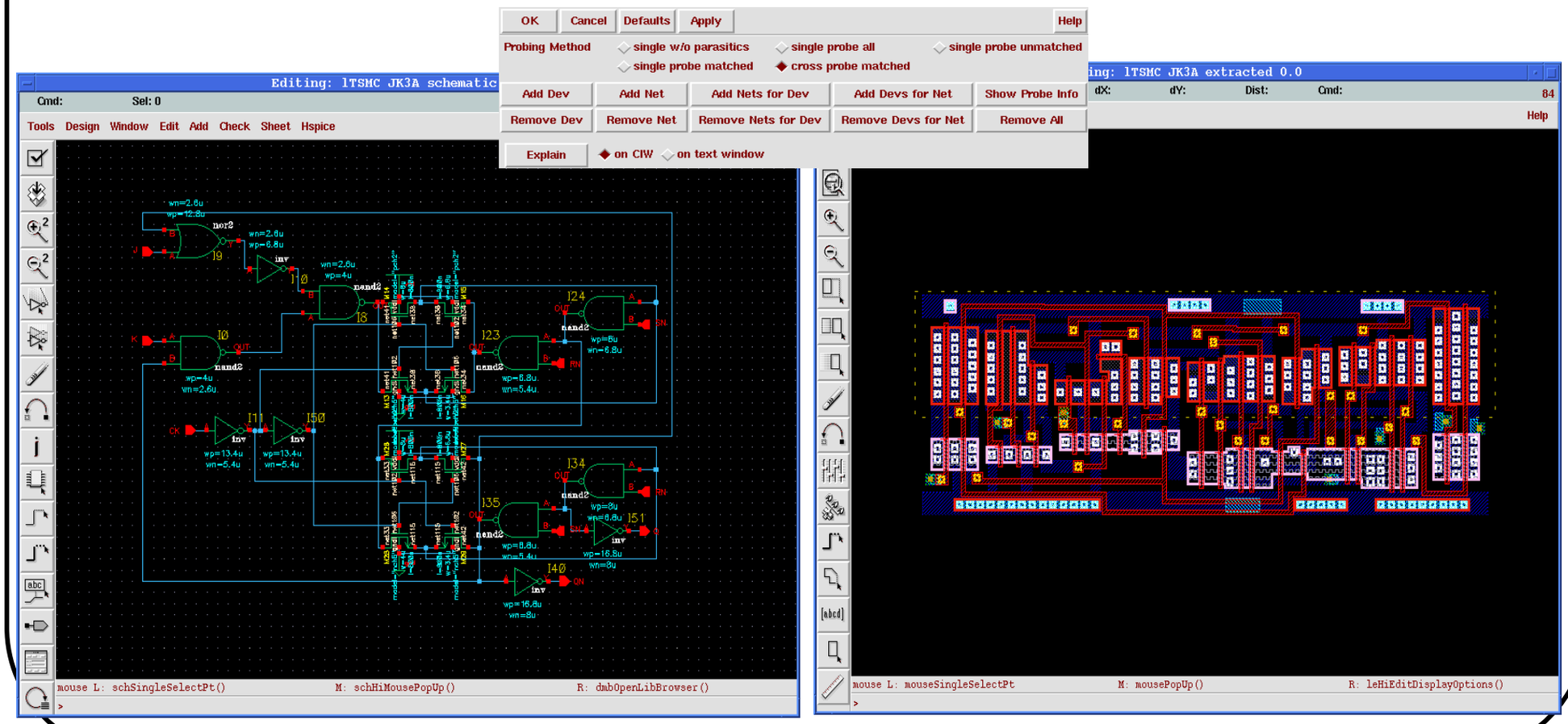
	layout	schematic
	instances	
un-matched	0	0
rewired	0	0
size errors	0	0
pruned	0	0
active	6	6
total	6	6
	nets	
un-matched	0	0
merged	0	0
pruned	0	0
active	7	7
total	7	7
	terminals	
un-matched	0	0
total	5	5

```
Probe files from /user/usr6/hywang/Labs/06sptm/LVS/schematic
devbad.out:
netbad.out:
mergenet.out:
termbad.out:
prunenet.out:
prunedev.out:
audit.out:
Probe files from /user/usr6/hywang/Labs/06sptm/LVS/layout
devbad.out:
netbad.out:
mergenet.out:
termbad.out:
prunenet.out:
prunedev.out:
audit.out:
```

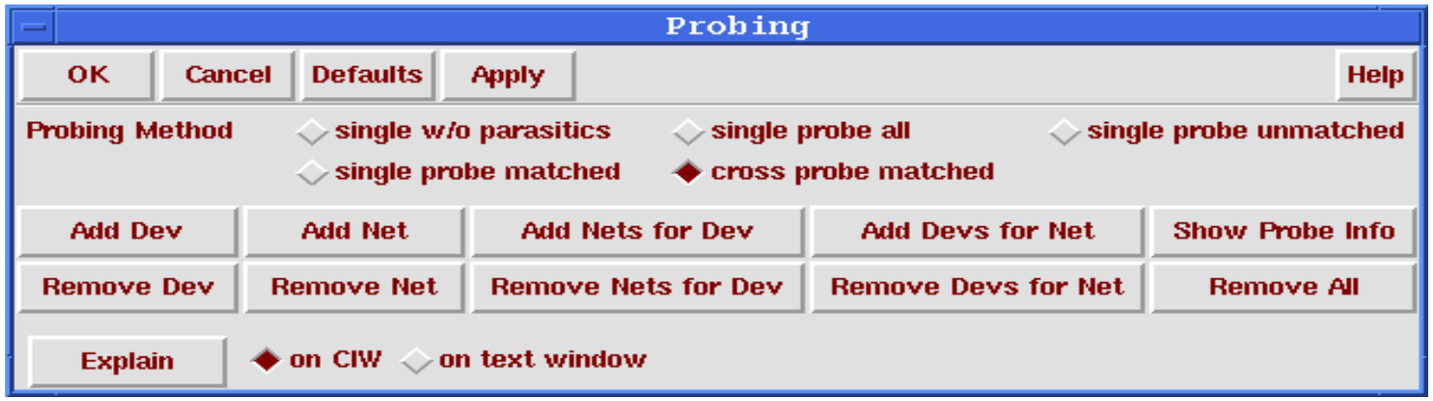
Probing in Diva

Probing in Diva **可以用來偵察** LVS **的錯誤, 是故在執行** probing**之前 必須先執行** LVS , **否則 無法正常運作** , **除了在選擇** single w/o parasitics **時**, **不須先執行** LVS.

尤其當我們選擇 cross probe matched 時, LVS schematic and Layout extracted cellview**可叫出下圖所示**



當進行 on-line Diva cross probing 時, Probing Method 選擇 cross probe matched(如下圖所示)



各個選項功能解釋如下:

Add Dev: 將所選 Device 加入到 probing data, 此時相對應的 extracted 或 schematic Device 若有 matched 時, 均會閃動變色.

Add Net: 將所選 Net 加入到 probing data, 此時相對應的 extracted 或 schematic Device 若有 matched 時, 均會閃動變色.

Add Nets for Dev: 將所選 Device 相關之 Nets 加入到 probing data, 此時相對應的 extracted 或 schematic Device 若有 matched 時, 均會閃動變色.

Add Devs for Net: 將所選 Net 相關之 Devices 加入到 probing data, 此時相對應的 extracted 或 schematic Device 若有 matched 時, 均會閃動變色.

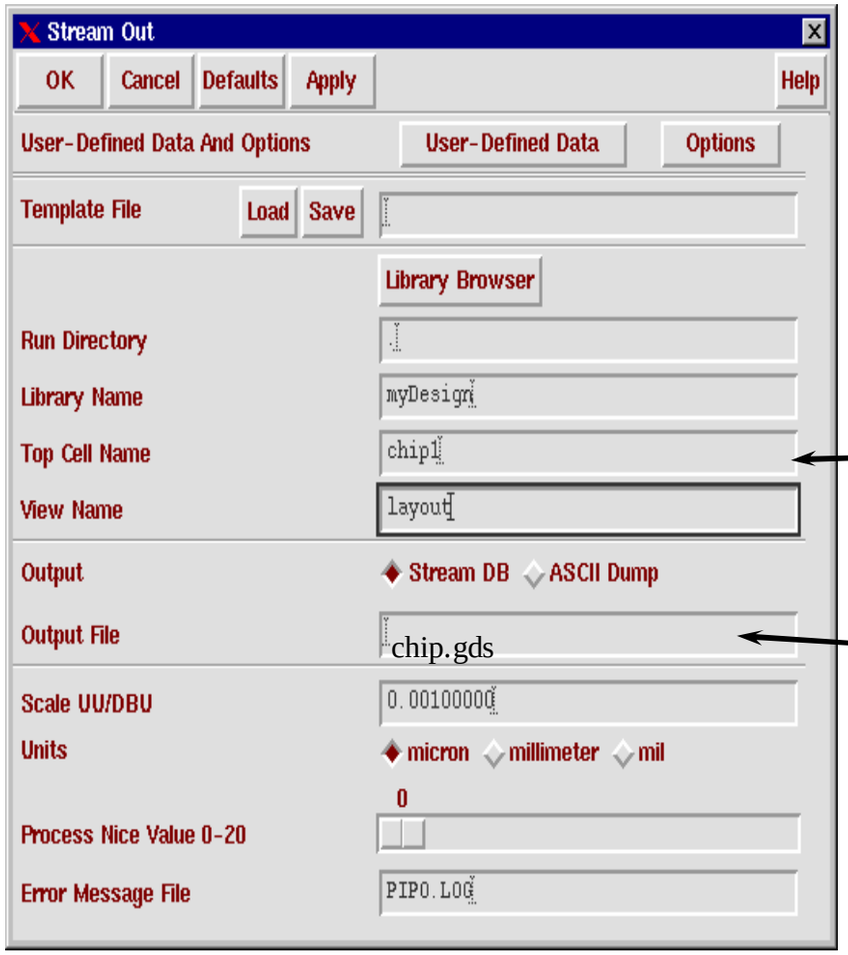
Show Probe Info: 將所有選取之 Devices 和 Nets 的 probing data 資料顯示出來.

Explain: 顯示所選取之 Devices 和 Nets 的 probing data 可顯示在 CIW 或是在新的 Text Window上.

Remove All: 將所有選取之 Devices 和 Nets 的 probing data 資料和顯示移去.

Preparing Stream(GDSII) format layout

In CIW, select File -> Export -> Stream ...



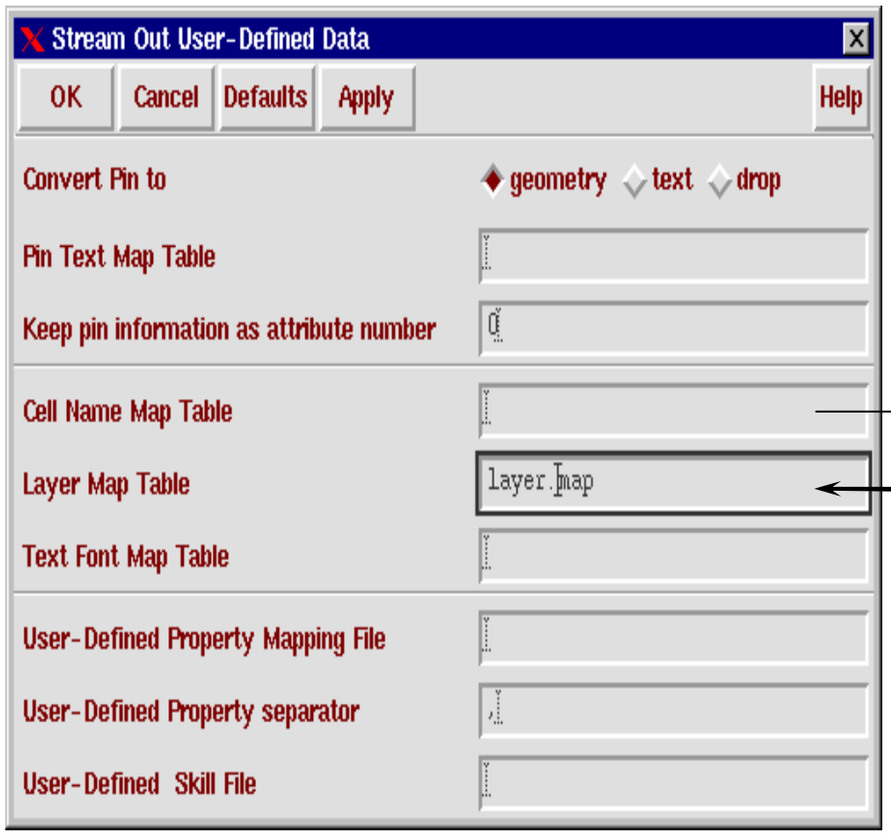
assign which layout to stream out

assign the stream-out layout file name

translation information file

Stream Out User defined data

In Stream Out form, select “User-Defined Data” button



assign the file of cell name mapping

assign the file of mapping number

Stream Layer Mapping Table

GDSII file 也稱作Stream-format, 當你在CIW選擇Translators ->Physical ->Stream Out 時, 則出現如同上圖的表格, 其中經常使用的選項是:

(1).Library Name = (myDesign)是你在Opus系統中的Library名稱
(2).Top Cell Name = (top)是你在Library中最上層cell,top cell 在Dracula 亦稱primary cell
(3).Output File = (top.db) 是寫到disk中GDSII file名稱

.Layer Map Table = (layer.map)通常是空白不使用, 若須轉出非本 technology file 定義的 layerNubmer時,則編輯layer.map檔案格式如下:

```
;
;Opus_layer_name  layer_purpose  Stream_layer_no  Stream_data_type
;
  POLY              drawing          6                0
  MET1              drawing         12                0
;
```

以上二列的作用, 對Stream Out而言,是告知OPUS將POLY層轉出為layer 6, 將MET1層轉出為 layer 12.(在Stream In時也可用此mapping file,以告知OPUS POLY 取layer 6, MET1取 layer 12).

Stream Out實際上是呼叫PIPO(Physical In,Physical Out)執行,所以當你執行完 Stream Out, 程式後會自動產生PIPO.LOG檔案,PIPO.LOG是執行過程的摘要,統計資料包括(1)Top Cells ,(2)List Hierarchy,(3) Individual Cell 內容(4) 各Layer統計.

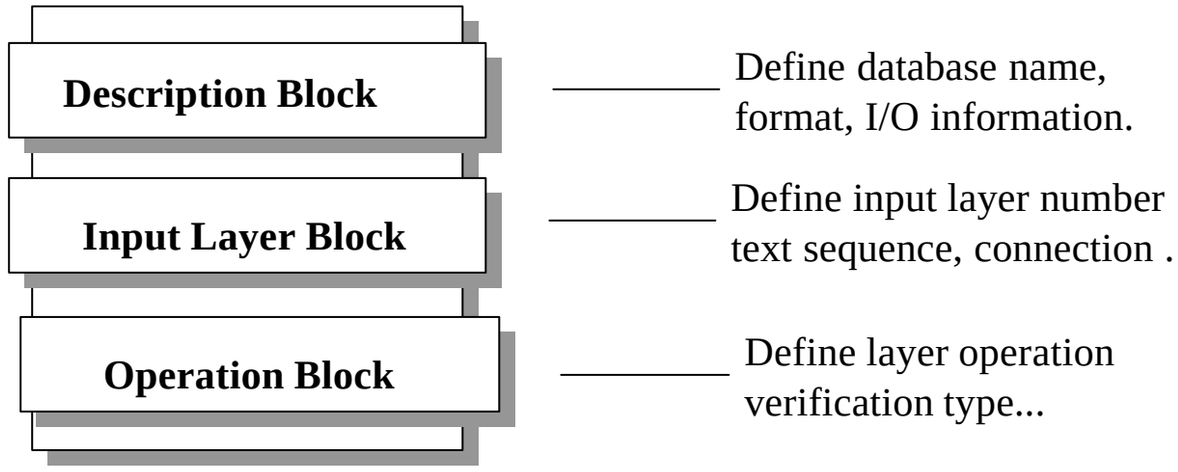
在Opus technology file中有定義layerName及layerNumber,layerColor,layerPattern,但是GDSII(top.db)中只有layerNumer(0-63),確定Opus technology file中的 layerName與GDSII layerNumer的對照, 以便在用Dracula讀layerNumber及送交光罩公司 MT-form時不致發生layer 不相吻合的錯誤.

批次模式佈局驗證

How to Use Dracula

- Create/Obtain a command file
- Fill in the database information
- Compile the command file : **PDRACULA**
- Submit the run file : **jxrun.com** or **jxsub.com**

Structure of Command File



Running procedure of DRC/ERC

After the assignment of the file name and top-cell name of layout in command file, compile command file by

```
% PDRACULA
 ********************************************************************************
 */N*    DRACULA3 ( REV. 4.51.0398    /     SUN-4     /GENDATE: 6-MAR-98/14  )
                *** ( Copyright 1995, Cadence ) ***
 */N*           EXEC TIME =16:07:10       DATE = 4-JUL-98
 ********************************************************************************
  :/g  9707drc06.com  n
  :/f

  ** NOTE : PARTIAL DELETIONS OF FILES WILL BE PERFORMED

  ** CREATING : COMMAND FILE : jxrun.com

  ** NOTE : THIS JOB HAS   122 STAGES

 END OF   DRACULA   COMPILATIONS
 *      .086 Mbytes allocated to the current process.
 *      .045 Mbytes is still in use.
 *  THE END OF PROGRAM                  TIME = 15:57:48    DATE = 4-JUL-98  *
```

%jxrun.com > drc.log & (Submit thr run file)

順利run **結束後會產生** *.sum file, **若** abort , **則須由**log file(drc.log)**找其原因** .
由 log file**亦可 得知** DRACULA**是否確實抓讀到** text(**因**Dracula**由**text**獲知**node name)

Label Information in Run Log

在log file(top.log) 中 IDLABL stage(search character ‘ IDLABL‘) 檢查是否所有要讓 DRACULA抓讀的 text都有被抓讀

```
*******************************************************************************
*/N*   IDLABL   ( REV. 4.51      /     SUN-4    /GENDATE: 6-AUG-98/10  )
              *** ( Copyright 1998, Cadence ) ***
*/N*          EXEC TIME =09:33:41      DATE =10-SEP-98
*******************************************************************************
*/L* Q1          X=    -25.70 Y=    -14.90
*/L* Q3          X=    -92.70 Y=    -14.10
*/L* VSS         X=     19.30 Y=     7.50
*/L* CLK         X=   -195.10 Y=     60.40
*/L* TC          X=     30.10 Y=     83.10
*/L* Q2          X=   -194.20 Y=     121.30
*/L* Q0          X=     32.50 Y=     181.60
*/L* RESET       X=   -194.20 Y=     270.90
*/L* VDD         X=   -173.10 Y=     356.70
*/L* COUNTEN  X=   -133.60 Y=     369.90
```

在log file(top.log) 中 POSATT stage(search character ‘ POSATT ‘) 檢查是否不同的 text均在不同的 node

```
*******************************************************************************
*/N*   POSATT   ( REV. 4.51      /     SUN-4    /GENDATE: 6-AUG-93/10  )
              *** ( Copyright 1998, Cadence ) ***
*/N*          EXEC TIME =09:33:51      DATE =10-SEP-98
*******************************************************************************
   Q1          X    -25.70    Y    -14.90      NODE        1   ATTACH MT2
   Q3          X    -92.70    Y    -14.10      NODE        2   ATTACH MT2
   CLK         X    -195.10   Y     60.40      NODE       22  ATTACH MT2
   TC          X     30.10    Y     83.10     NODE        43  ATTACH MT2
   Q2          X    -194.20   Y    121.30     NODE        41  ATTACH MT2
```

```
Q0          X    32.50     Y   181.60   NODE     32   ATTACH MT2
RESET       X  -194.20     Y   270.90   NODE     53   ATTACH MT2
COUNTEN     X  -133.60     Y   369.90   NODE     74   ATTACH MT2
.
.
.
.
.
```

node #

GENERATE SHORTED NODES DIRECTORY : MULTILAB WITH 0 ENTRIES

無short ckt **發生**

GENERATE OPENED NODES DIRECTORY : SAMELAB WITH 0 ENTRIES

無open ckt **發生**

.. LIST OF EXTRACTED PAD NAMES AND NODE NUMBERS :

```
CLK          22
COUNTEN      74
Q0           32
Q1            1
Q2           41
Q3            2
RESET        53
TC           43
VDD          23   P
VSS           6   G
```

*****唯有在無short ckt與open ckt之情況下ERC 才會確實執行,故須先解決short,open ckt後重新作ERC**

DRC Report Summary

由summary file(top.sum)中之 OUTPUT CELL SUMMARY 部份可見一些因違犯 design rule而得的 CELL-NAME, 如下例

因違犯design rule
而得的CELL-NAME

```
---------- OUTPUT CELL SUMMARY ----------

CELL-NAME  LAYER #  ----------  W I N D O W  ----------
      DATATYPE                                  # OF POLYGONS TEXTS
                                  (LINE SEGMENTS)

D11M50     50/ 0   -154.50   -138.50    126.00    137.50    1     0
E349       50/ 0    -54.50   -123.00     75.50   -102.50    2     0
E449       49/ 0    -75.50    100.50     54.50    120.00    2     0
E749       49/ 0     52.50   4773.50   4925.50   4946.00   92     0
E849       49/ 0     52.50   4773.50   4925.50   4946.00   92     0

OUTDISK PRIMARY CELL :  OUTTOP
   WINDOW :       -1.00     -1.00    5001.00    5001.00

      ENDED   AT  TIME =16:26:13     DATE = 4-OCT-98
******     PROBLEM  GEOMETRY  ERROR  LISTING     ******

*****   END  OF  PROBLEM  GEOMETRY  LISTING   *****
```

D p receding 為DRC error
E p receding 為ERC error

all error flag region

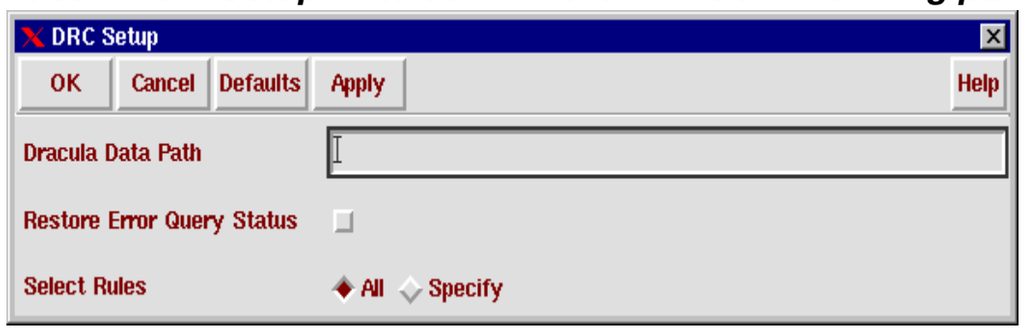
此處列出有問題之object(非design rule所造成,command file所指定check,為免犯此類error,layout object用45,90度之rectangle,polygon,path)

**** layout上標 text注意事項
(1)text origin 在layout object上
(2)text所有I/O pad,power&gnd pad,important internal node
(3)以A-Z,a-z,0-9 text較佳, 且text以字母起始
(4)Appending a colon(:) to text will define “virtual wire” to Dracula.
Note:ERC時電阻的兩端視為同node,故勿標不同text於電阻兩端 (for CIC DRC/ERC command file)

Debugging DRC with Dracula Interactive

Using Dracula Interactive **to identify and analyze Dracula verification errors**

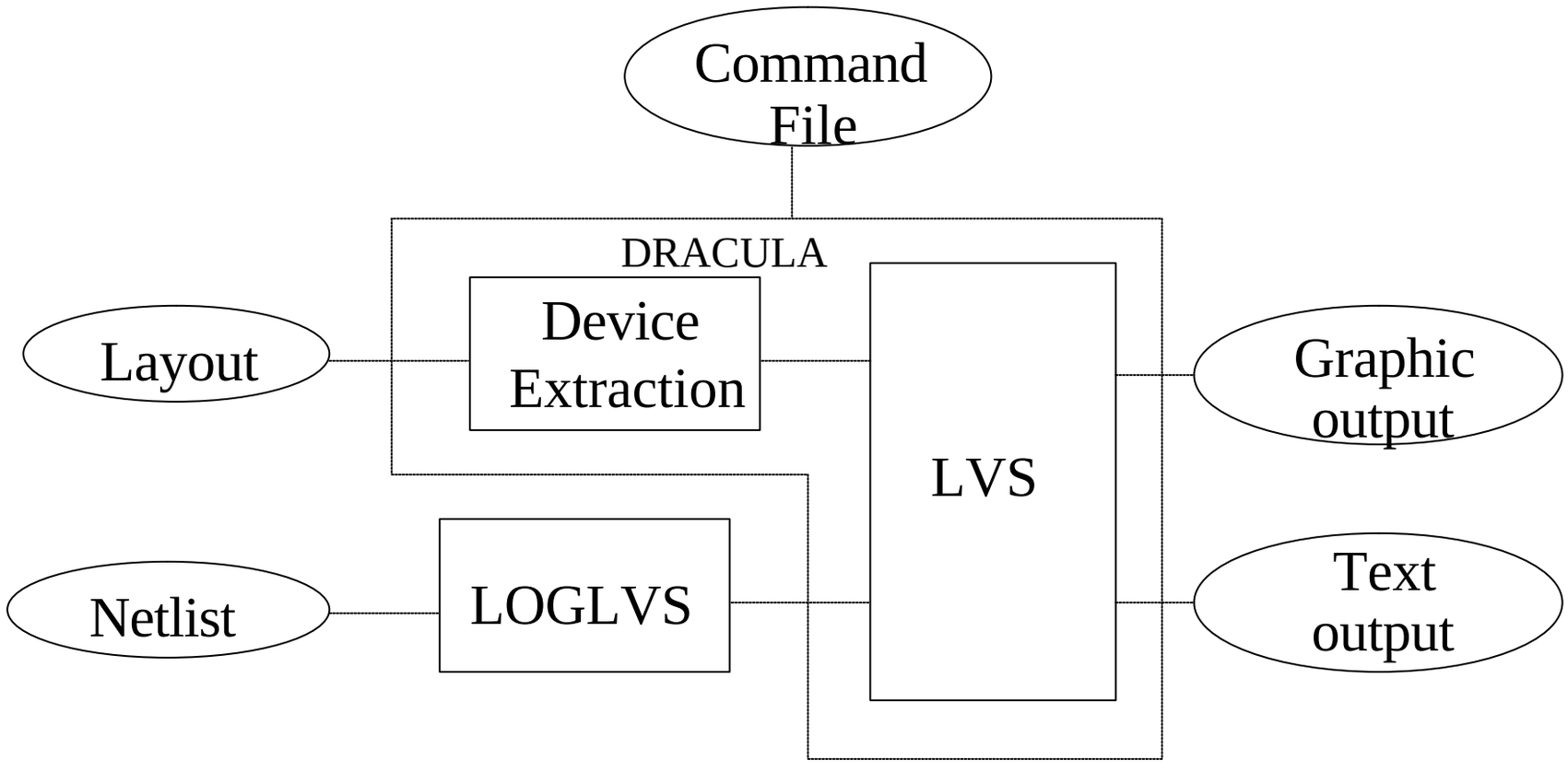
在 layout(InQuery) view 上, Tools-> Dracula Interactive
use DRC-> Setup... to confirm the DRACULA running path



The KEEPDATA must assign to be InQuery in DRACULA command file for the using of InQuery. The using of InQuery can refer to “InQuery Tutorial” manual.

The running procedure of LVS

LVS Flow



Preparing Netlist

進行LVS之前需先準備 netlist 檔
若已建好相對的schematic view, 用File->Export->CDL... (由schematic view轉出netlist)
原始netlist檔可為SPICE檔，CDL檔或verilog 格式
進行LVS之前，netlist 檔需先經LOGLVS轉換成Dracula format以便讀取

Netlist檔(for Dracula LOGLVS)之control statement

以‘ * ’ 開頭的statement表此列為給Dracula讀取
以‘ $ ’開頭的statement表此列為comment,Dracula不讀取

.GLOBAL (宣告global node) (ex).GLOBAL VDD,VSS
*.BIPOLAR (有此statement LOGLVS 將保留netlist中的 R,C,diode以作比對用,當然此時配合用的
command file須有相對的元件定義)
搭配*.BIPOLAR 可加
*.RESVAL *.RESSIZE (作R value或size check)
*.CAPVAL *.CAPAREA (作C value或area check)
*.DIOAREA *.DIOPERI (作diode area或perimeter check)
*.UNSPEC_MOS (有此statement,則當netlist中無指定W,L時,將show message於‘PRINT.OUT’中
*.REVERSE(LOGLVS讀取MOS順序為W,L,若MOS順序為L,W時可用此statement)

Compiling Netlist

LOGLVS program is used to compile netlists and create a flat transistor level netlist(LVSLOGIC.DAT) in Dracula format.

%LOGLVS

:**htv**	generate information for inquery
:**genpad**	generate a 6gpads.dat file for LVS device reduction
:**case**	specify case sensitivity
:**cir artchip.cdl**	read the netlist
:**con artchip**	convert from the top cell name
:**exit**	

Error file: PRINT.OUT
Translation table: IMAGE.LIS

LVS Running Steps

After the assignment of the file name and top-cell name of layout in command file, use

```
% PDRACULA
*******************************************************************************
*/N*    DRACULA3 ( REV. 5.1       /     SUN-4     /GENDATE: 6-AUG-93/11  )
                *** ( Copyright 1997, Cadence ) ***
*/N*          EXEC TIME =16:07:10       DATE = 4-OCT-94
*******************************************************************************
 :/g  9708lvs06.com  n
 :/f

 ** NOTE : PARTIAL DELETIONS OF FILES WILL BE PERFORMED

 ** CREATING : COMMAND FILE : jxrun.com

 ** NOTE : THIS JOB HAS   42 STAGES

END OF   DRACULA   COMPILATIONS
*      .086 Mbytes allocated to the current process.
*      .045 Mbytes is still in use.
*  THE END OF PROGRAM                TIME = 15:57:48    DATE = 4-JUL-98  *
```

% jxrun.com >lvs.log & (**如此使之** background run, run **的 訊息** show **於** top.log**中**)

順利run **結束後會產生** *.lvs file, **若** abort , **則須由**log file(lvs.log)**找其原因** .
由 log file**亦可 得知** DRACULA**是否確實抓讀到** text(text**用處在於給**node name, **務必確定無**short ckt**與**open ckt**發生**)

LVS Running Report *.lvs

HEADER{Including Device number before/after reduce in layout and schematic}
CORRESPONDENCE NODE PAIR
LVS DEVICE MATCH SUMMARY
DISCREPANCY POINTS LISTING
DEVICE MATCHING SUMMARY BY TYPE
LVS SUMMARY(REPEATED)

The following message will be reported in *.lvs file if the layout database match the schematic in connectivity.

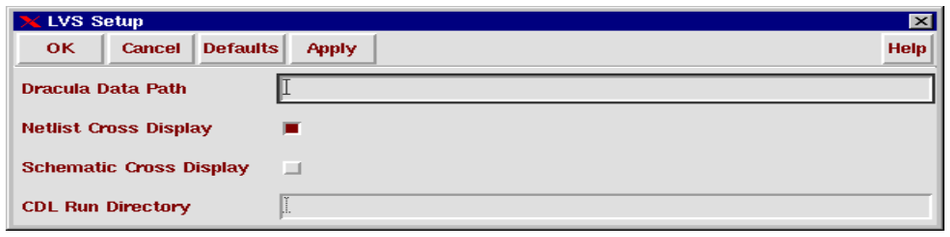
```
**********************************************************
*********  DISCREPANCY POINTS LISTING  ************
**********************************************************

                NO  DISCREPANCIES
```

LVS Debugging

1. Open Schematic Window & Layout Window, **以便進行** Dracula LVS cross probing.

2. **在** Layout Window **上端的功能選項** Menu **，選擇**
   Tools --> InQuery(**此時** Layout Menu **將會改變成** InQuery Menu)

Setup: **設定** Dracula **及** CDL Run Directory **和** Display Options.



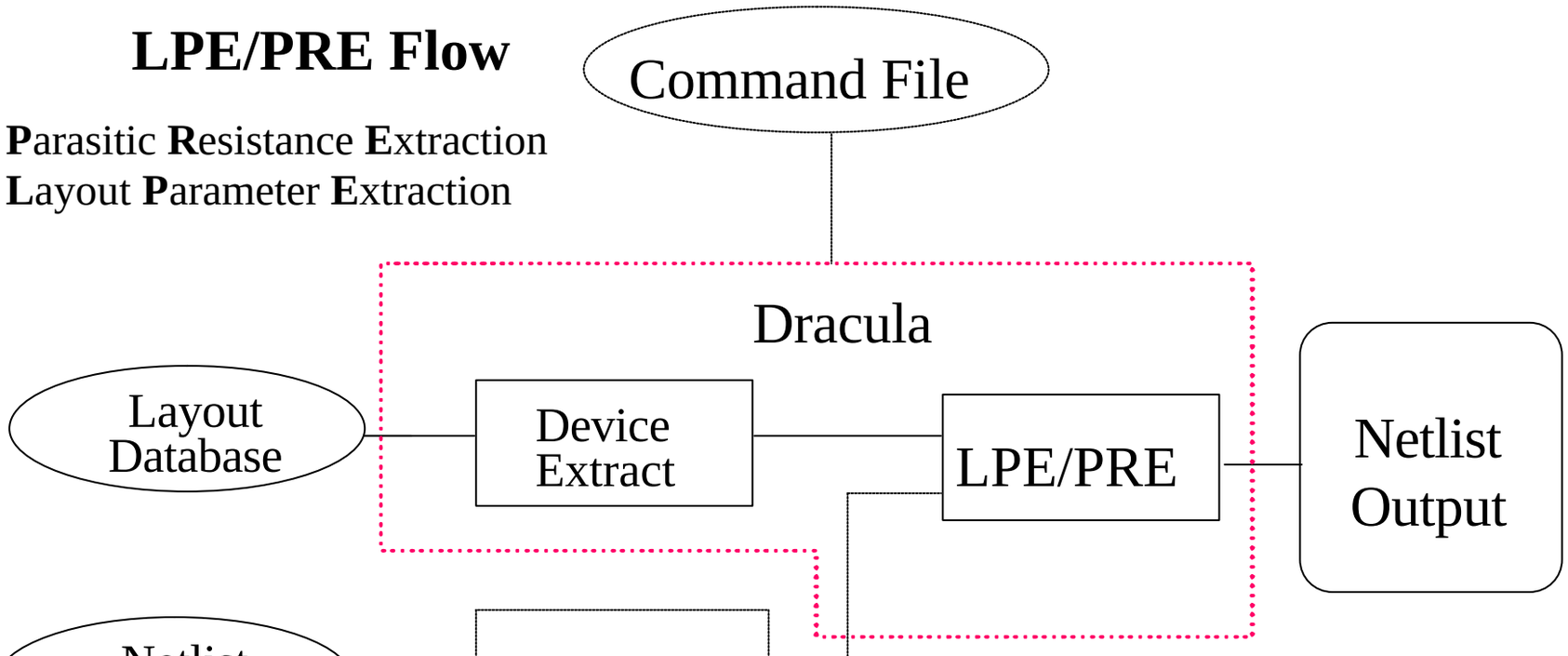
Set Visibility: **設定要顯示的選項**?
Open Netlist Window**：打開** InQuery Netlist Window, **使用者可直接在視窗上選擇所要進行** cross probing **的** net **或** device **以及**subcircuit?
Show Network Hierarchy...:Opens a browser window that displays the hierarchy of the circuit netlist so you can select the cell name for the netlist you want.
Show Discrepancy Report**：顯示** LVS Discrepancy Report?
Display Errors**：選擇欲顯示的**Discrepancy error#, error type?
Clear Error Display: Removes the highlghting from errors
Highlight...:Highlights nets and devices in the layout window.
Unhighlight....: Removes the highlghting from nets or devices in the layout window
Show Net or Device Information...:Display the #, layer, and name of the selected net or device.
Fit Last Display: Zooms in the last displayed net, device, or error.
Clear Last Display：Removes the last displayed net, device, or error.
Explain:Displays the message associated with a highlighted net or device.

Running procedure of LPE/PRE

Layout Parasitic Extraction includes LPE/PRE of Dracula

LPE/PRE Flow

Parasitic **R**esistance **E**xtraction
Layout **P**arameter **E**xtraction



Backannotation of schematic node index
LPECHK
LPESELECT[S]

LPE Example Output

The running procedure of LPE/PRE is similar to LVS except the using of different command file.
After the successful running of the run file, LPE/PRE will output netlist in SPICE-like format.

LPESELECT[N] MOS OUTPUT **SPICE 01** ——— + **.DAT** = Output File name

Example of LPE output

```
*.GLOBAL   vdd!  gnd!  VCC
*
.SUBCKT ARTCHIP   i_inject  input  out
*
******    CORNER ADJUSTMENT FACTOR =   0.5600000
******
MM16-XI23 XI23-net320 net24 vdd! vdd! P L=1.00U W=16.00U
MM1-XI23 out XI23-net320 vdd! vdd! P L=1.00U W=16.00U
MM17-XI23 XI23-net320 net24 gnd! gnd! N L=1.00U W=11.00U
MM0-XI23 out XI23-net320 gnd! gnd! N L=1.00U W=11.00U
*
CC17-XI12 gnd! XI12-net10637  6.31350E-12PF
CC10 gnd! out  2.76975E-12PF
*
RR20-XI12 VCC XI12-net10651 2.30000E02 $.MODEL=P
RR8-XI12 gnd! XI12-net10657 1.00000E03 $.MODEL=P
.ENDS
```

Post-Layout Simulation

The netlist of SPICE/CDL format that includes parasitic element can be obtained after the Dracula LPE/PRE. It can be used for post simulation by circuit simulator, such as SPICE or TimeMill,PowerMill and PathMill.

Add simulation control and input stimulus for final simulation

- **透過** Post-layout simulation, **可以評估電路在** interconnect parasitics **的影響程度。**
- Signal Coupling **的效應亦可由**Post-layout simulation **來模擬。**
- Power/Ground bounce **的程度亦可由**extracting power/ground line **的**parasitics **來模擬評估。**

The I/O Supported by CIC

先由CIC**取得**0.5um 2P2M**製程**PAD**的** library, **使顯示在**library manager**以便使用**, **在很多**cell**中都搭配有**schematic view**以便作**whole-chip LVS, CIC**提供的**PAD library**包括** “y2kpad052p2m” pad library**以適用於**CIC**提供的**0.5um**製程**.

一般我們將chip**的**internal circuit**部份稱為**core, **而**pad**部份(包括**power, ground, input, output)**稱為** I/O, **無論是**core**或** I/O**部份的電路**, **都得注意** latch-up**問題**, I/O**部份的電路須作**ESD protection. **在**process data**中一般還包括這些避免**latch-up**與**ESD**的**design rule,**若有已驗證過且合本身**design**需求的**I/O pad, **最好直接引用**.

PAD cells

library “y2kpad052p2m**” 中各 cell 之說明如下:**

(針對 digital circuit design)

*****I/O cells:**

1. power pad: **cic_pwr--> 此 cell 用於 internal ckt 之 vdd or vss pad**
cic_gnd--> i/o(external) & internal ckt 共同的 vss pad
cic_vss--> i/o(external) ckt 之 vss pad
cic_vdd--> i/o(external) ckt 之 vdd pad, 亦可作internal與external 共用之 vdd pad

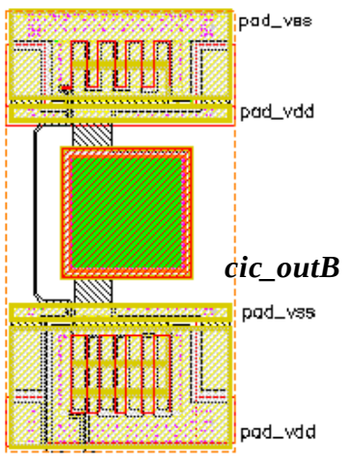
2. i/o pad: **cic_aout--> 作input pad 或無 driving 能力之 output pad 用**
cic_in--> 作input pad
cic_outX--> output pad(規格參考下頁圖示)
cic_biX--> bi-direction in/out pad(規格參考下頁圖示)

3. 其它： **cic_corner--> pad 角落連接用**
cic_dummy--> pad 與pad 連接用

*****predriver cells:**
pre8--> cic_outX,cic_biX 之 predriver
pre8h--> cic_outX,cic_biX 之 predriver (speed 比 pre8 快, 但須注意noise 的問題)

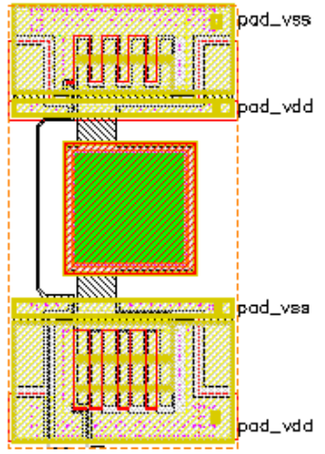
*****input TTL buffer cell:**
ttl--> cic_in or cic_aout 之 input buffer, 其 D.C. spec. : V_{IL}=0.8, V_{IH}=2.0 volt (5V supply)

cic_outA



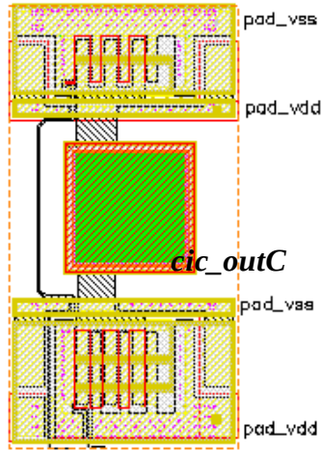
IOH(@ Vout=4.5)=16.5mA
IOL(@ Vout=0.18)=-8.6mA

cic_outB



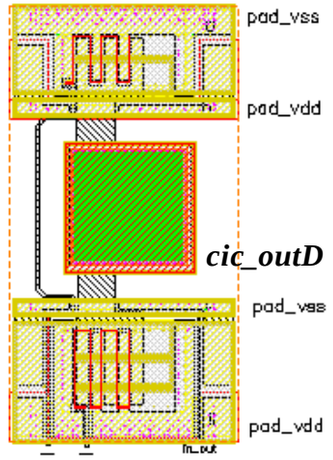
IOH(@ Vout=4.5)=14.5mA
IOL(@ Vout=0.18)=-7.5mA

cic_outC



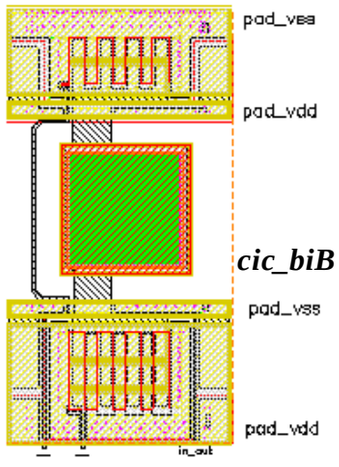
IOH(@ Vout=4.5)=12.5mA
IOL(@ Vout=0.18)=-6.5mA

cic_outD



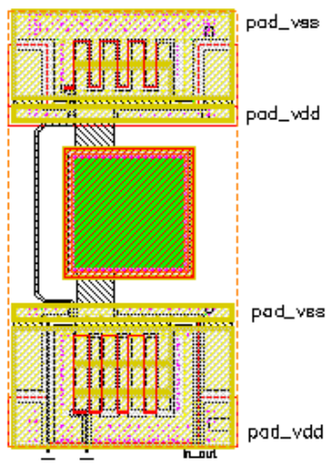
IOH(@ Vout=4.5)=10.5mA
IOL(@ Vout=0.18)=-5.5mA

cic_biA



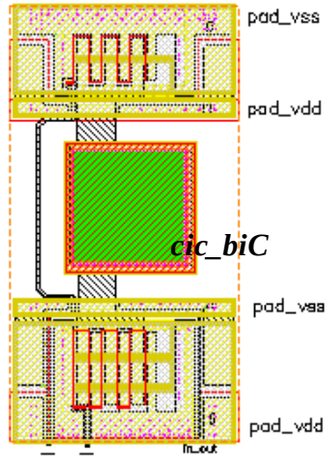
IOH(@ Vout=4.5)=16.5mA
IOL(@ Vout=0.18)=-8.6mA

cic_biB



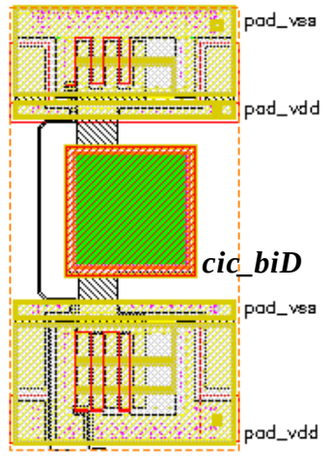
IOH(@ Vout=4.5)=14.5mA
IOL(@ Vout=0.18)=-7.5mA

cic_biC



IOH(@ Vout=4.5)=12.5mA
IOL(@ Vout=0.18)=-6.5mA

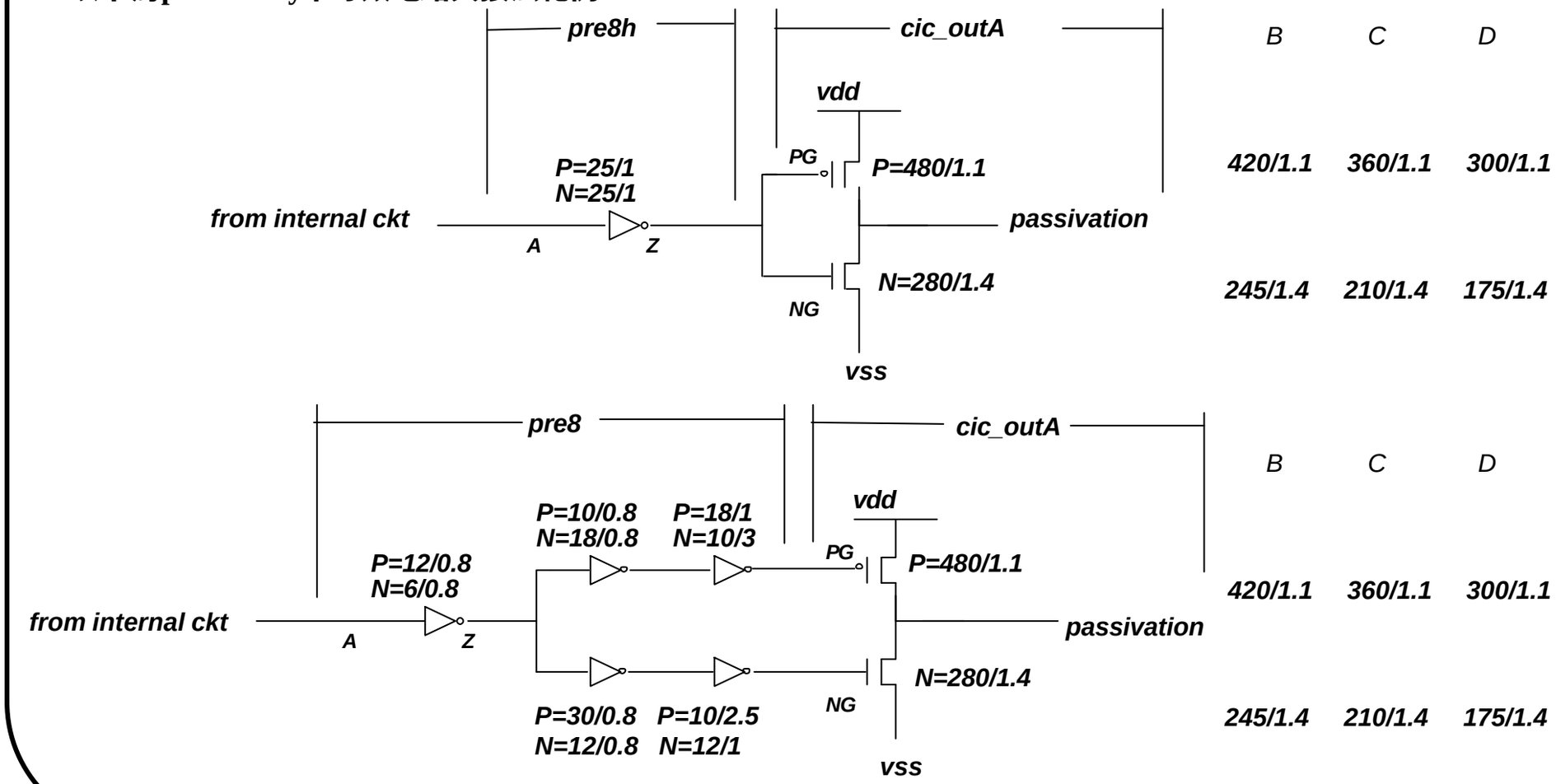
cic_biD



IOH(@ Vout=4.5)=10.5mA
IOL(@ Vout=0.18)=-5.5mA

Output PAD

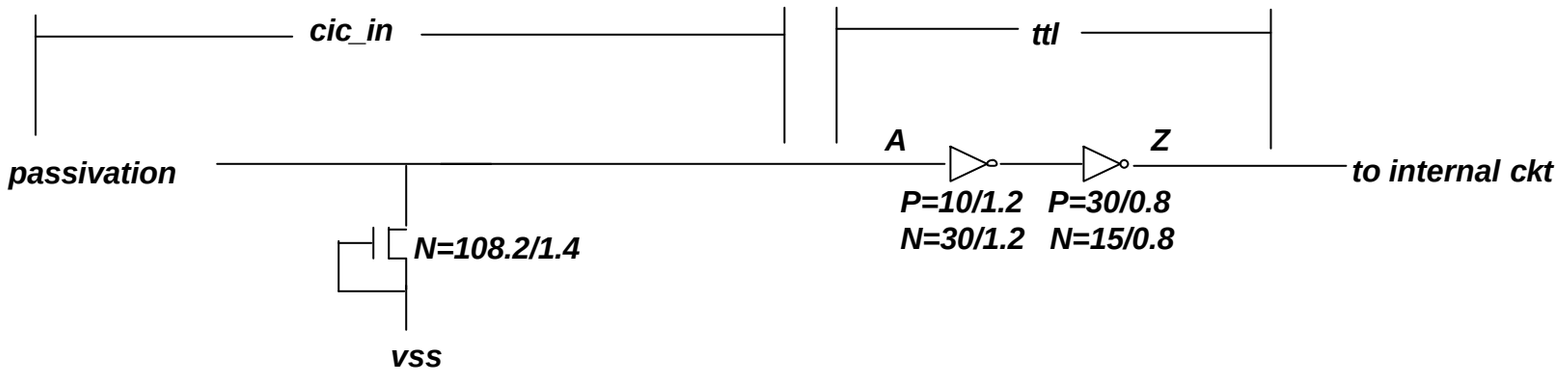
以下為pad library中等效電路與接法範例

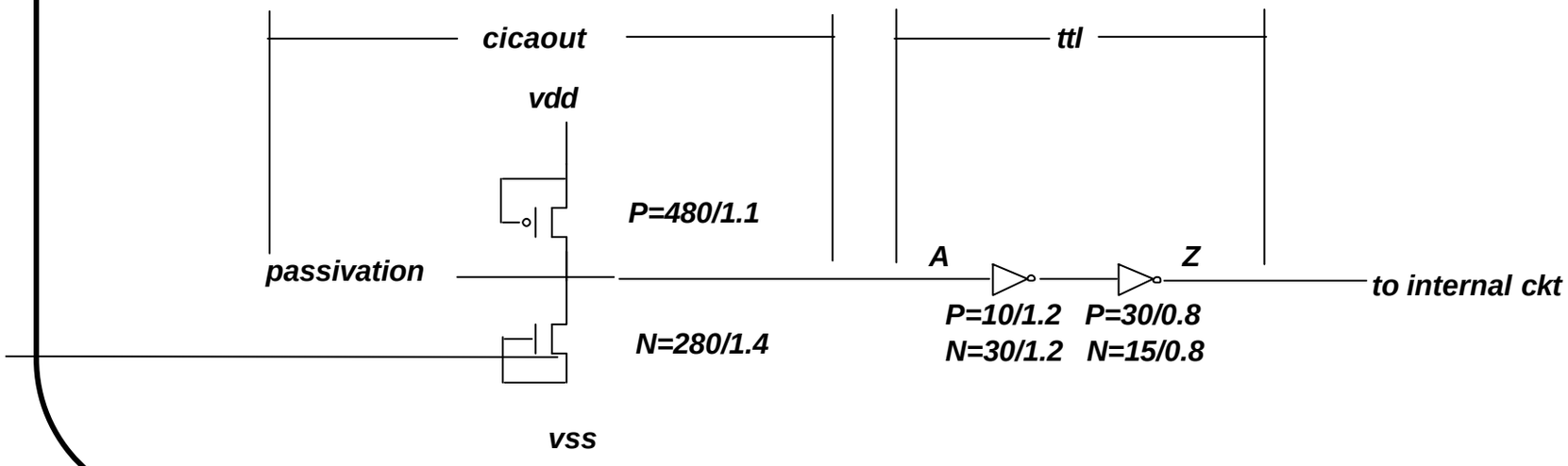


	B	C	D
	420/1.1	360/1.1	300/1.1
	245/1.4	210/1.4	175/1.4

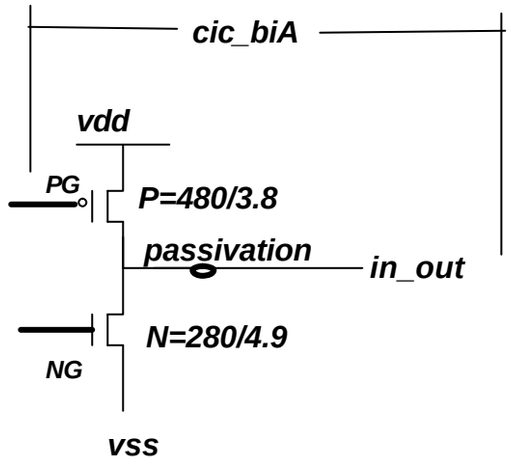
	B	C	D
	420/1.1	360/1.1	300/1.1
	245/1.4	210/1.4	175/1.4

Input PAD


cic_in
ttl
passivation
A
Z
to internal ckt
P=10/1.2
P=30/0.8
N=30/1.2
N=15/0.8
N=108.2/1.4
vss

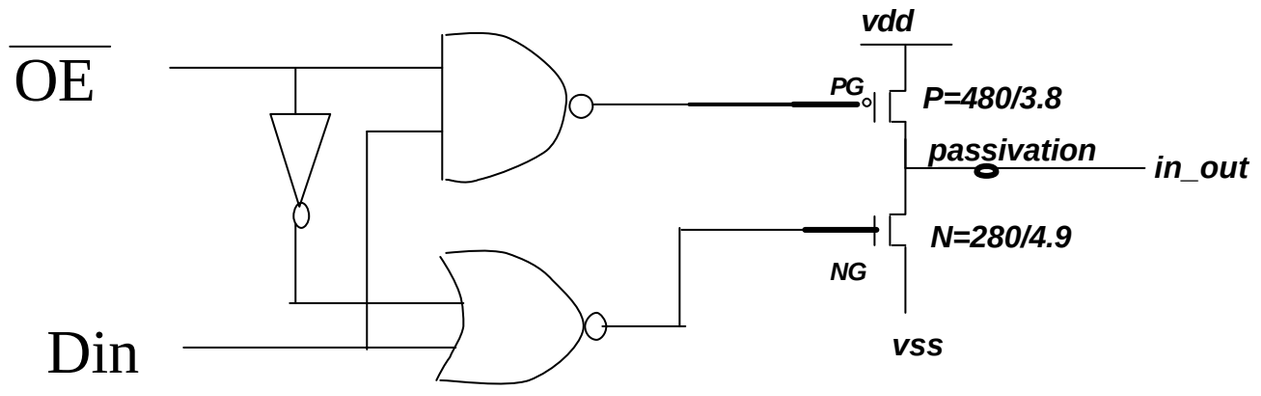

cicaout
ttl
vdd
P=480/1.1
passivation
A
Z
to internal ckt
P=10/1.2
P=30/0.8
N=30/1.2
N=15/0.8
N=280/1.4
vss

Bidirectional PAD

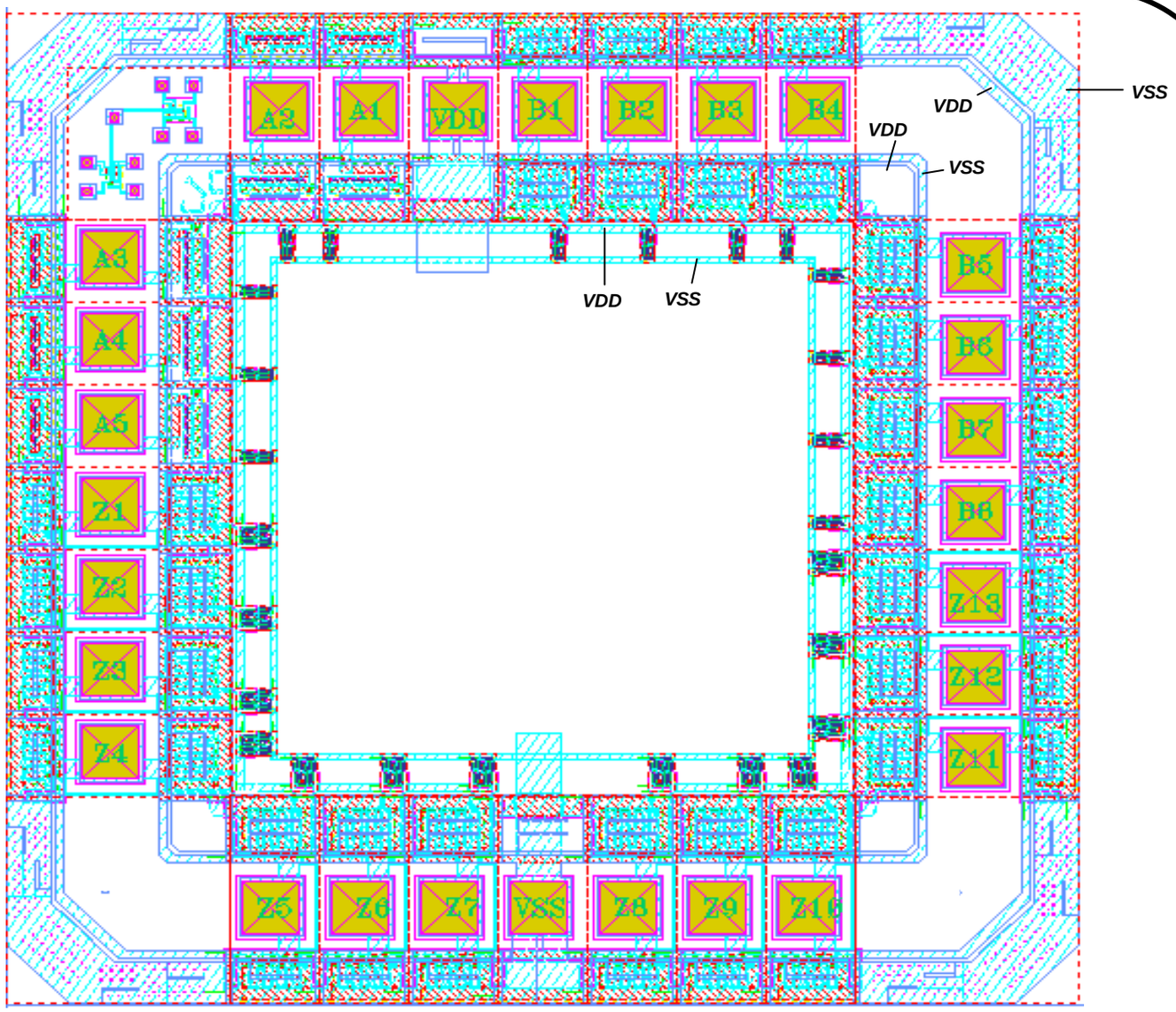


	B	C	D
P=480/3.8	420/1.1	360/1.1	300/1.1
N=280/4.9	245/1.4	210/1.4	175/1.4

Bidirectional Pin **的用法**

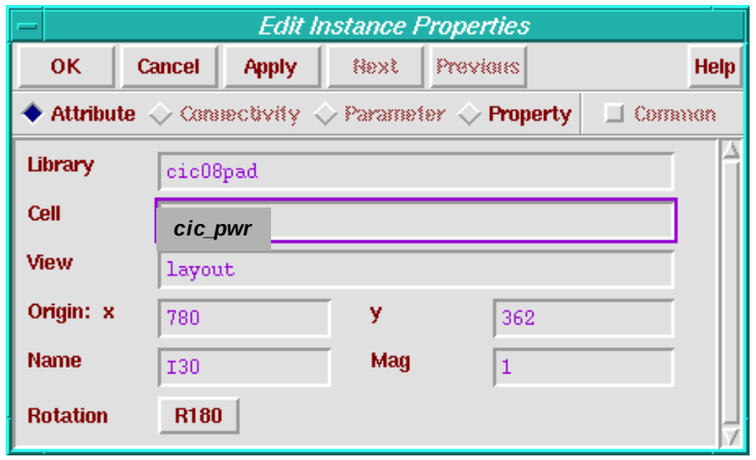


右圖為 cell “edu”, 是digital circuit 28pins I/O pads的規劃範例,其中input pad(用cell “cic_in”和“cic_aout”)已連接cell “ttl”, output pad(用cell “cic_outA”)則已接cell “pre8”



若要更換成不同之 pad, 則在 layout view上, 用mouse選欲更換之pad.再Edit-> Properties...

PAD Replacement



Click Apply

以上在 digital circuit design output部份的buffer(在output pad與predriver)是為提高output current driving capability與增加noise margin, input部份則多是為了作ESD protection, 或使chip外的TTL high-low level與 chip內CMOS的 high-low level 一致.

若是作analog circuit design, 也是得考慮 input 與 output部份的ESD protection, 可使用cell "cic_aout"(當然在pre-simulation時同時要將 "cic_aout"電路包括進去), 或直接output(可用cell "cic_pwr")

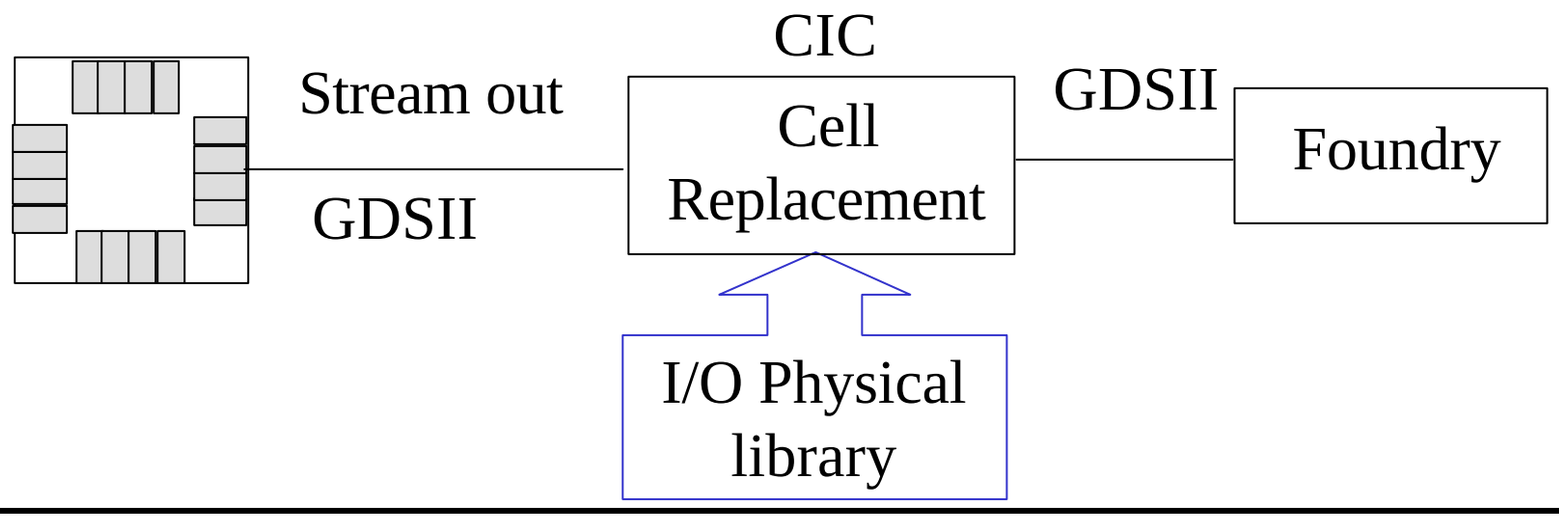
0.35um PAD Usage

- **由於**0.35um **製程在**ESD **的保護方面需求較為嚴謹，因此，**0.35um **製程的**I/O PAD**係委請**TSMC**提供，但因智慧財產權考量，無法提供完整**Layout View**及** netlist **給**End User**。申請者請使用** abstract view(**僅包含**metal,via**及**boundary) **來進行** I/O PAD **的**placement.

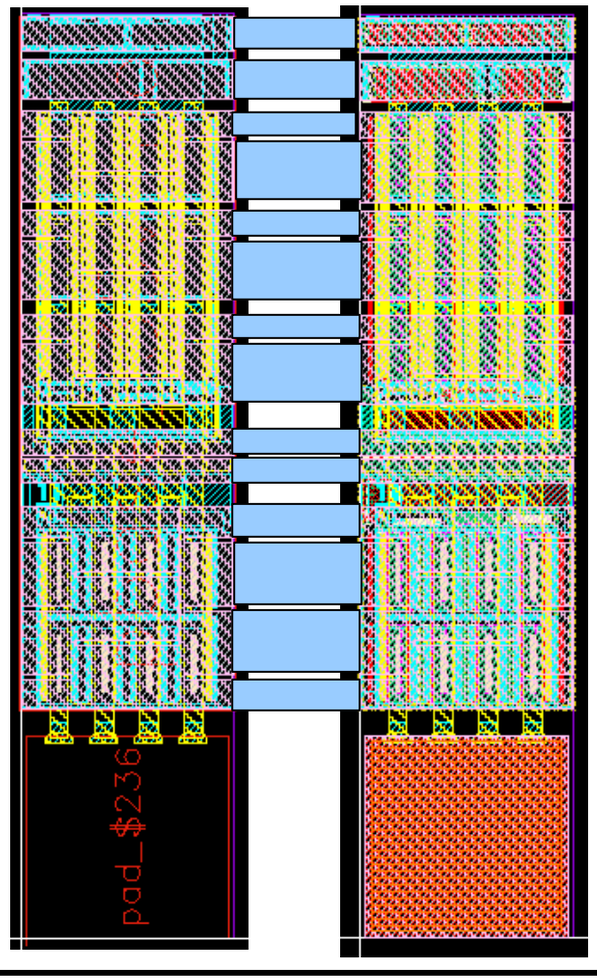
  PAD library : tpz773sn_200A for 0.35um 1P4M silicide process

  tpz773sb_analog_100a for 0.35um analog pads

- User **必須自**pad library **中自行選擇所需的**pad cell **並完成** PAD **的**wire routing
- User **的**cell name **不能和** I/O cell **的**cell name**相同**
- **當使用上述之** library **時**, **切記在申請表上勾選 "申請使用**TSMC I/O Pad"



0.35um PAD Usage(Con't)



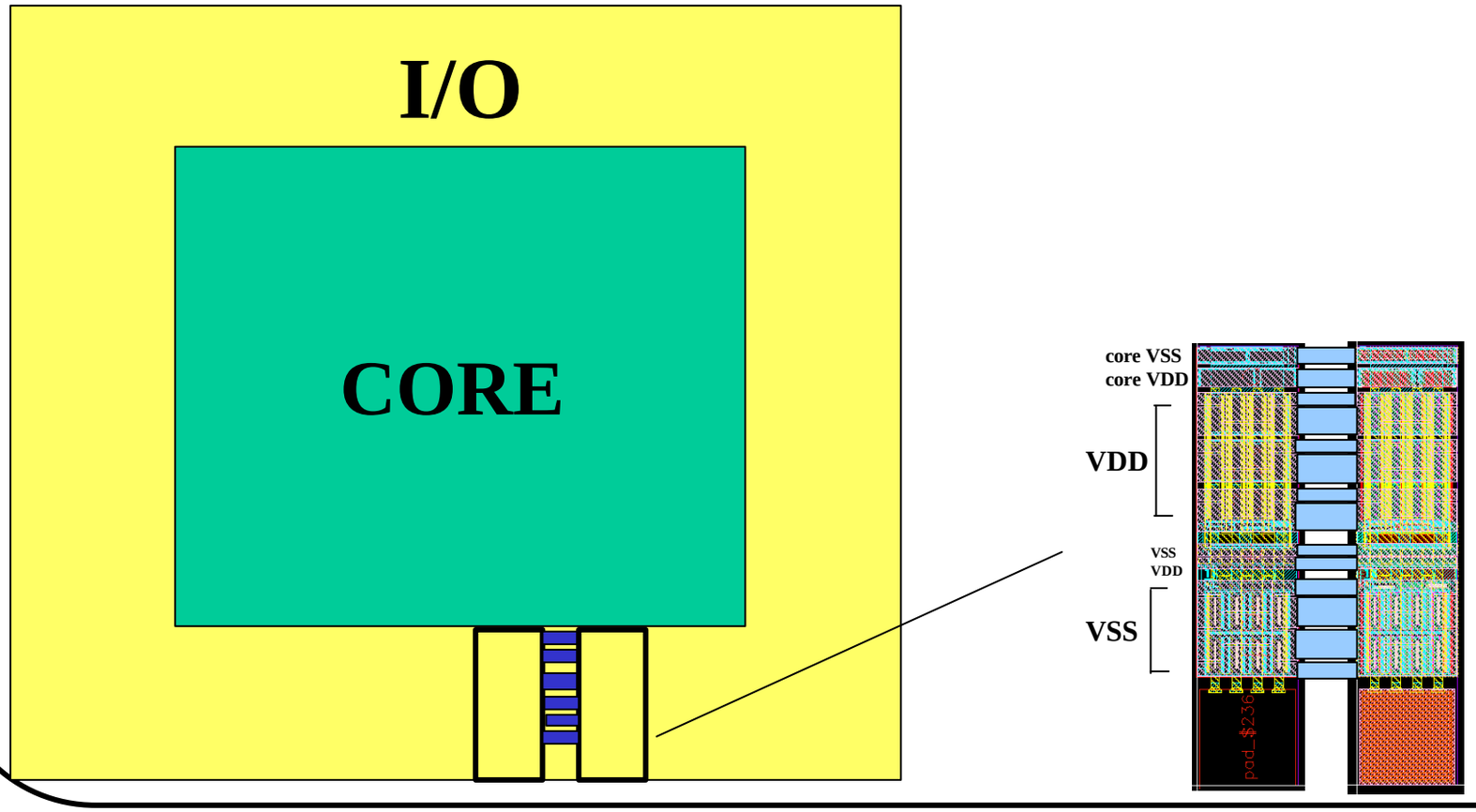
1. Connect by wire
2. Connect by feeder cell
3. Connect by butting - prBoundary layer

0.35um PAD Usage(Con't)

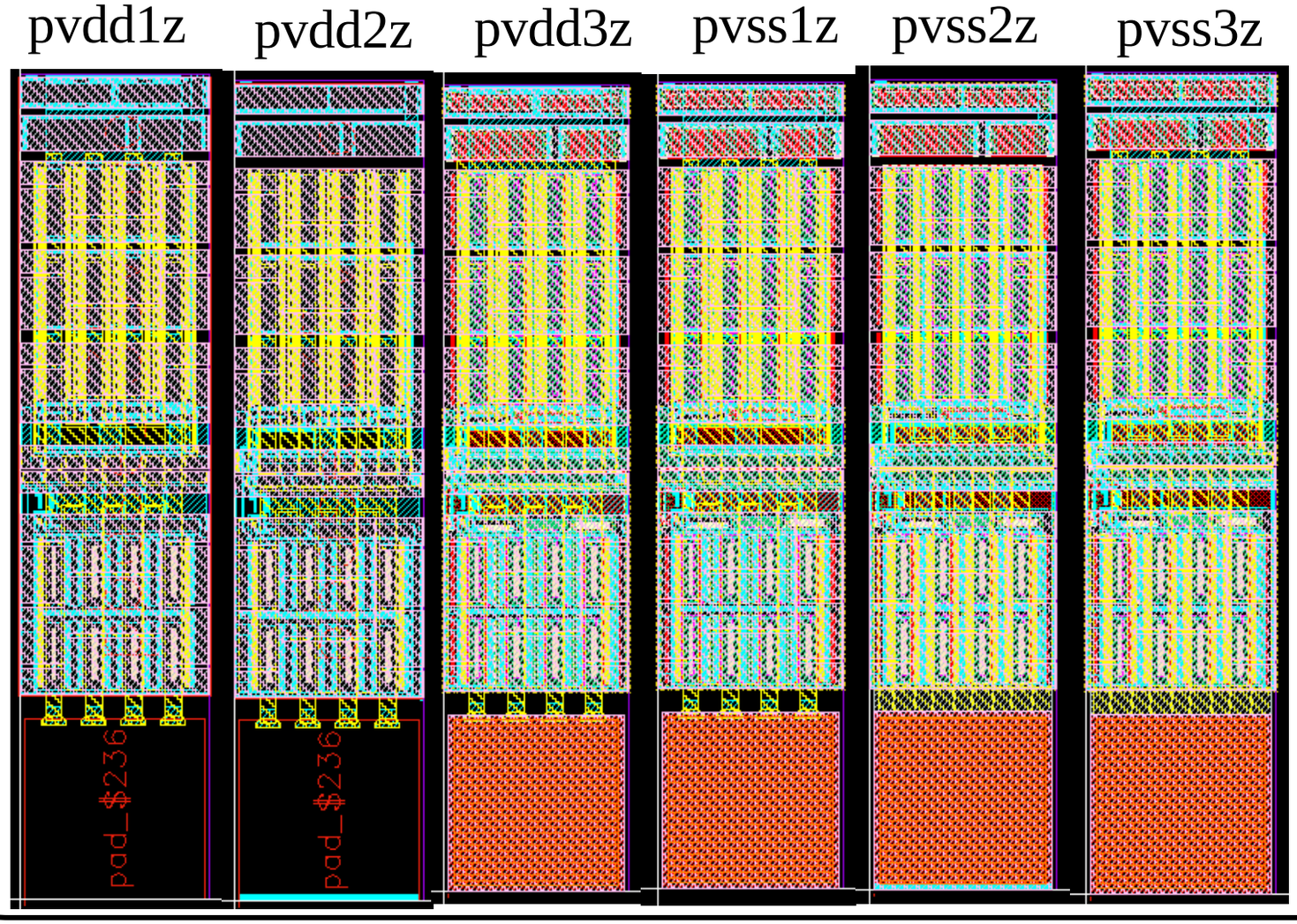
1.**取得**0.35um 1p4m**製程之**I/O pad**的**digital library**(tpz773snV200A)與** analog library**(tpz773sn_analogV100A)**與相關資料**, **了解各** pad**的規格使用方法**, **安排本身設計**chip**的**I/O pad,power ground pad**位置**

2. **將**I/O pad**的**library**設到**Library Manager**中以便叫用**

3. **安排各**cell**位置並將**pad**間的拉線完成**



0.35um PAD Usage(Voltage/Source)



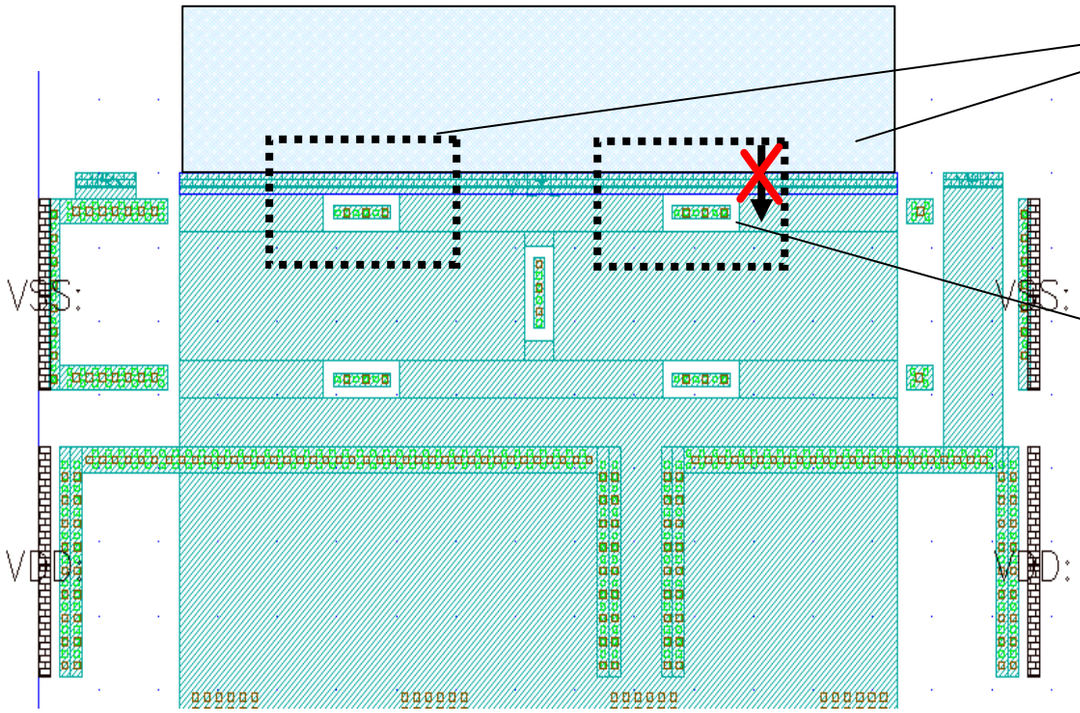
1: for core

2: for io

3: for both

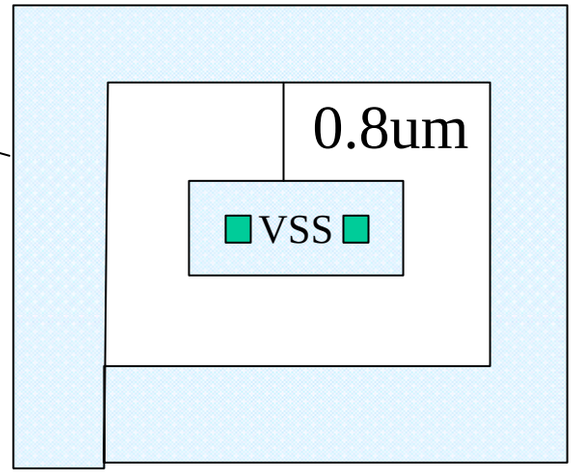
0.35um PAD使用注意事項

pvdd1z

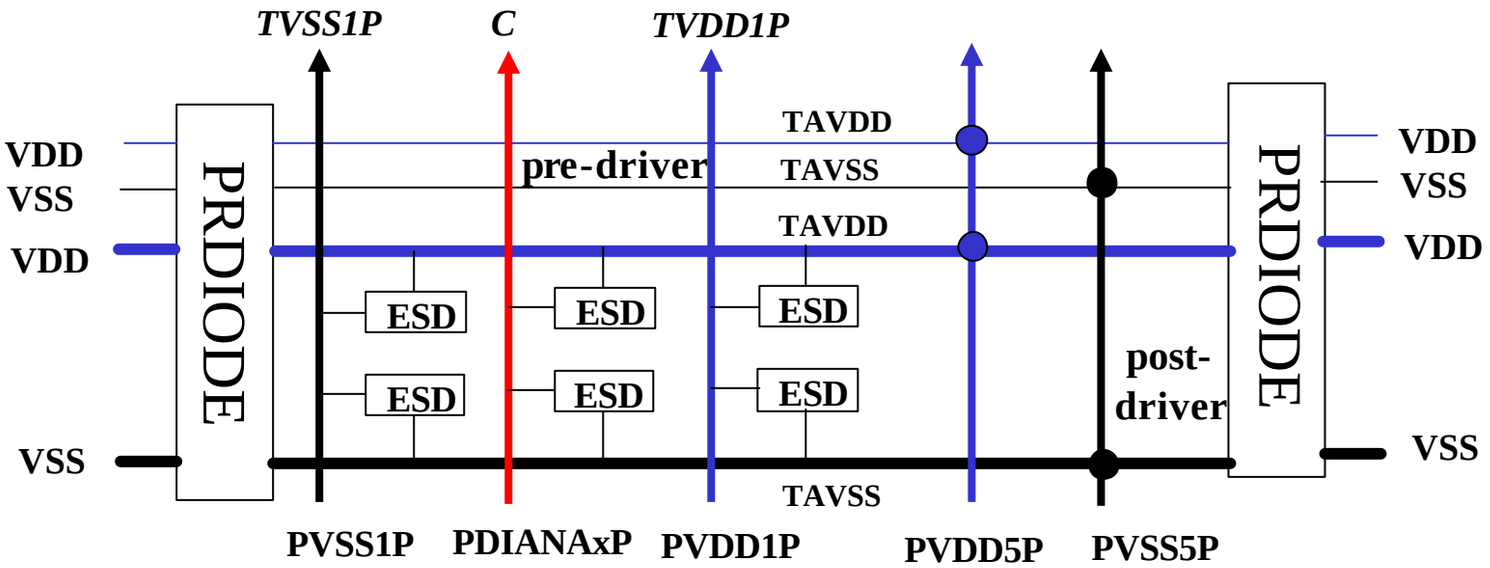


Warning !!

Metal line can’t extend into the pad, otherwise the metal will short with the VSS signal



Analog I/O Pad Structure



PRDIODE : provide isolation of digital/analog I/O power rings
PVDD3P : VDD for core/ pre-driver/ post-driver
PVDD1P : VDD for core ; PVDD1P1 : VDD for core using with PVDD2P
PVDD5P : VDD for pre-driver/post-driver
PVDD4P: VDD for pre-driver
PVDD2P: VDD for post-driver

0.35um 製程Whole Chip Verification

由於 TSMC 0.35um **製程之** I/O pad **不提供** netlist **及** detailed layout, **因此無法進行** whole chip **之** LVS **及** LPE

1. Whole Chip DRC
2. Whole Chip ERC - Can add labels in I/O pads for checking OPEN/SHORT Connection
3. Core LVS

Post-layout simulation

1. Extracting the core circuit
2. Look in the data sheet for I/O cell loading, add to the extracted circuit
3. Simulate the circuit
4. Add I/O cell delay for final timing