Basic SYNOPSYS User Guide

CCU EE VLSI Lab

Outline

- Setup system environment
- Default Synthesis
- Setting Design SPEC.
- Setting Operating Environment
- Setting Design Constraint
  - Combinational circuits
    - -- Timing constraints
  - Sequential Circuits
    - -- Specify the clock
    - -- Setting input delay
    - -- Setting output delay
- Simulation of Synthesized Circuit

CCU EE VLSI Lab

Setup a New Synopsys User

- Add the contents of */usr/synopsys/cicSynop/synopsys.cshrc* to your .cshrc

```
hsieh> cat /usr/synopsys/cicSynop/synopsys.cshrc  >> .cshrc
hsieh> source .cshrc
hsieh> design_analyzer&
```

完成系統環境設定

啓動 Synopsys Design Analyzer

- Use *online documentation*

```
hsieh> cd
hsieh> cp /usr/synopsys/cicSynop/lview .
hsieh> iview&
```

CCU EE VLSI Lab

.synopsys_dc.setup File

- : 在.synopsys_dc.setup定義了啓動 design compiler 時的設定以及使用的 cell Library .
- : .synopsys_dc.setup 檔案使用者可洽系統管理者. 此檔案可複製到 user 的目錄下方便修改 .
- : 在 .synopsys_dc.setup 中所使用的 Library 相關設定如下：

```
search_path = {. /vlsi-a/Librarys/LIB06_V2/Synopsys/usr/synopsys/libraries/syn};
```

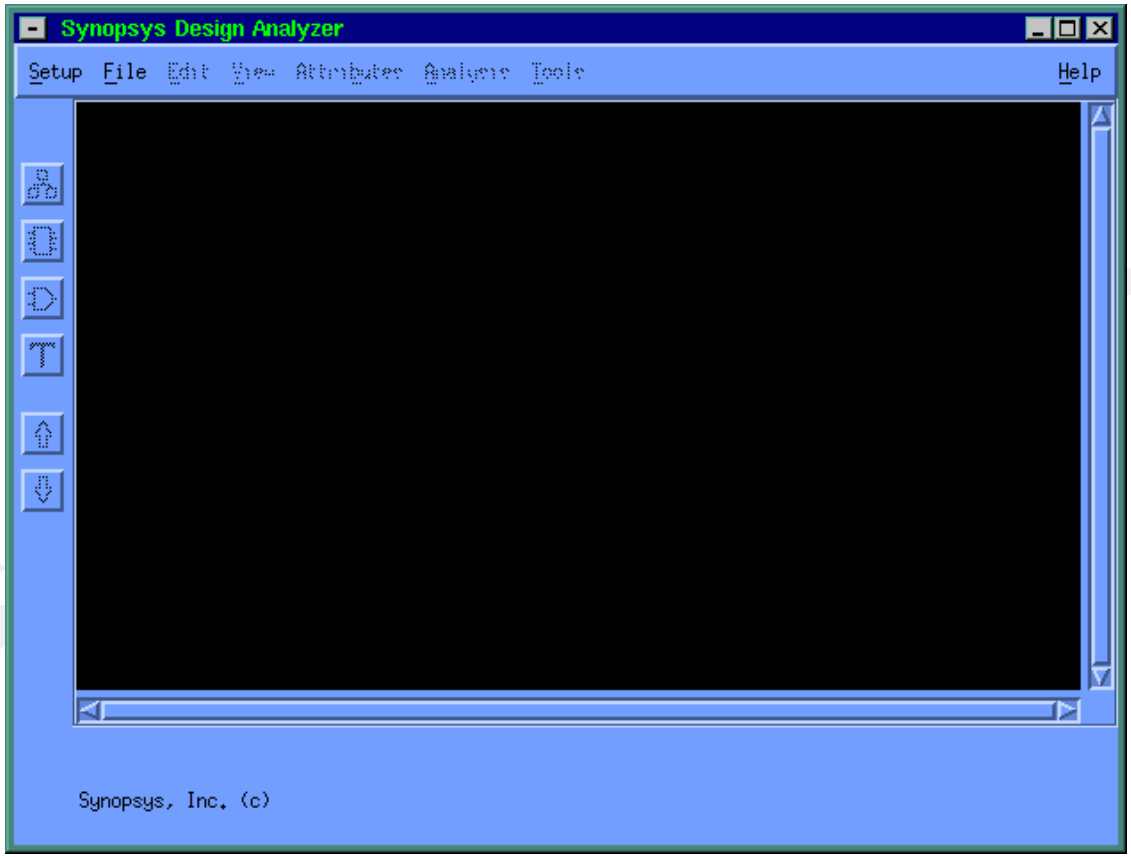
➢路徑要配合系統管理者的規畫設定 , 以上路徑只是以VLSI-CAD 實驗室的設定爲例說明

```
target_library = {cb60hp231d.db};
link_library = {cb60hp231d.db};
symbol_library = {cb60hp231d.sdb};
verilogout_no_tri = “true” ;
```

可以自行定義自己的 cell library(verilog model) 使用 synopsys 的轉換程式轉換成所需的 .db file(binary file)

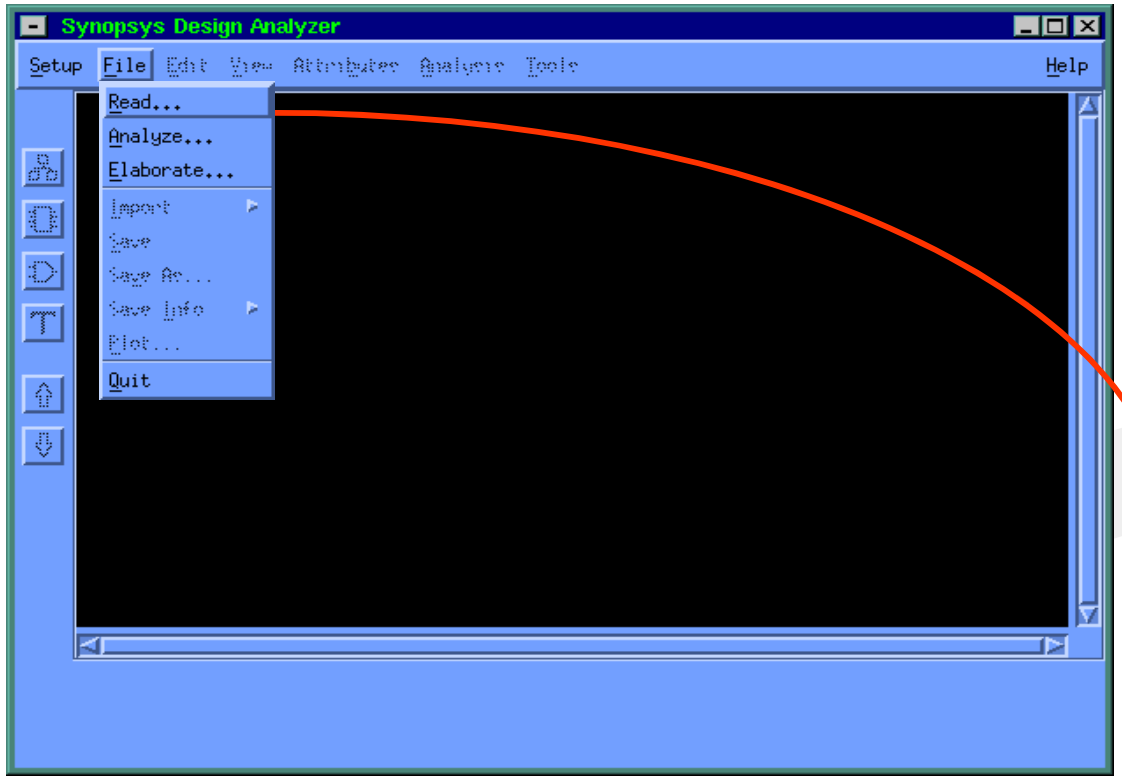
命令 SYNOPSYS 輸出的 verilog file 中不要有 tri_state 的敘述. (在 CADENCE 中使用 verilog IN 時不可有 tri_state 的敘述)

Hsieh > design_analyzer &



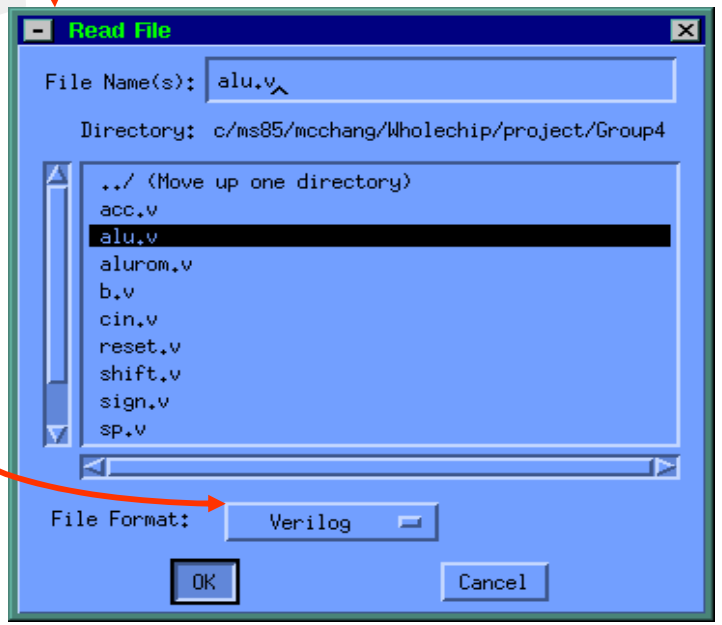
啓動 **Synopsys Design Analyzer** 後的畫面:

Default Synthesis



由 **File** → **Read** 出現 Read File window 。

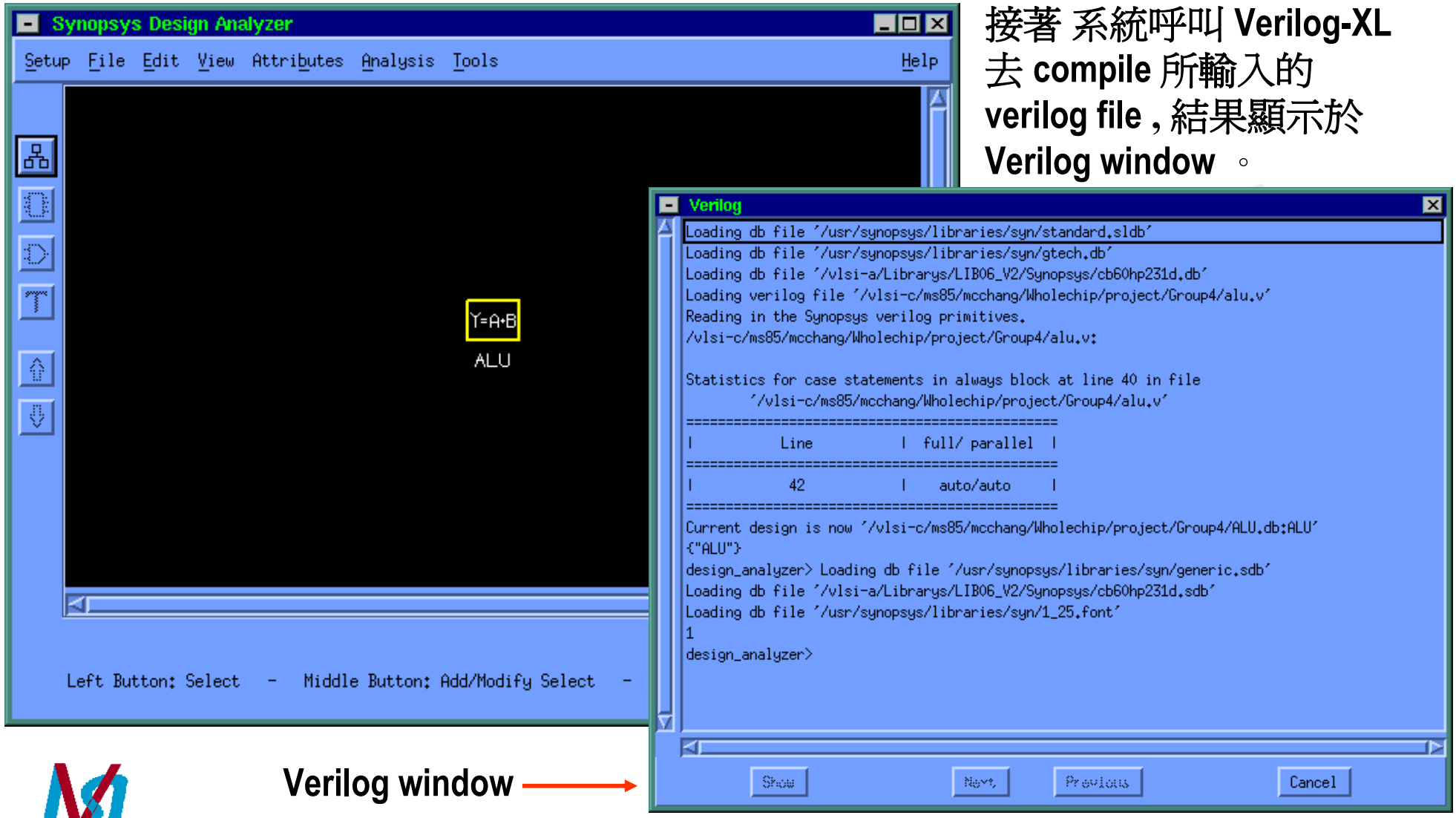
在 Read File window 中選擇要合成的 verilog file ，然後按 OK。(此時系統會開出一個 Verilog window , 見下頁圖)



Read Formats button
-- Synopsys formats
DB (binary) : .db
equation : .eq
state table : .st
-- Verilog : .v
-- PLA(Berkeley Espresso) : .pla
-- EDIF

Default Synthesis (cont.)

接著 系統呼叫 Verilog-XL 去 compile 所輸入的 verilog file , 結果顯示於 Verilog window 。



Verilog window →

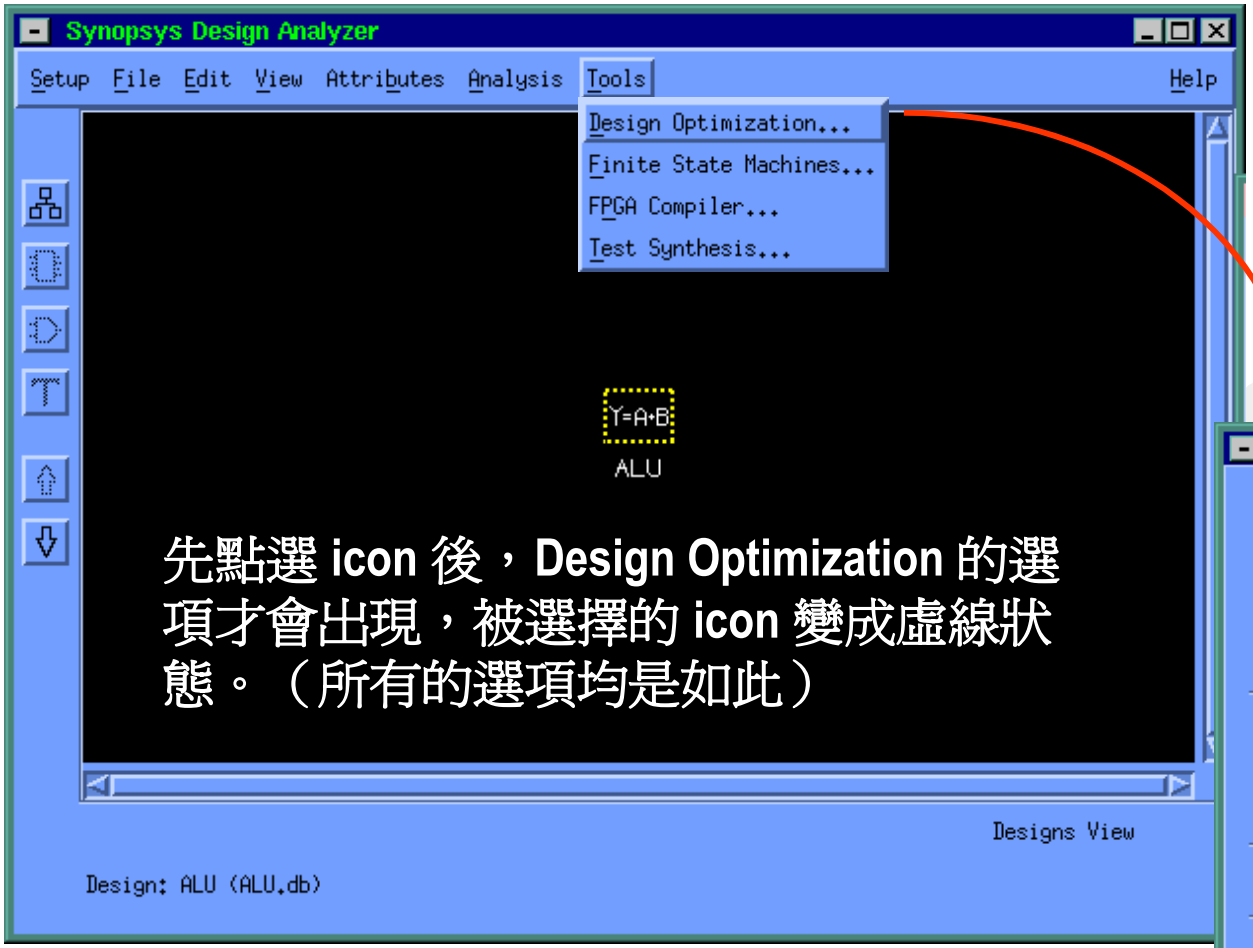
Default Synthesis (cont.)

常見的錯誤如下 :

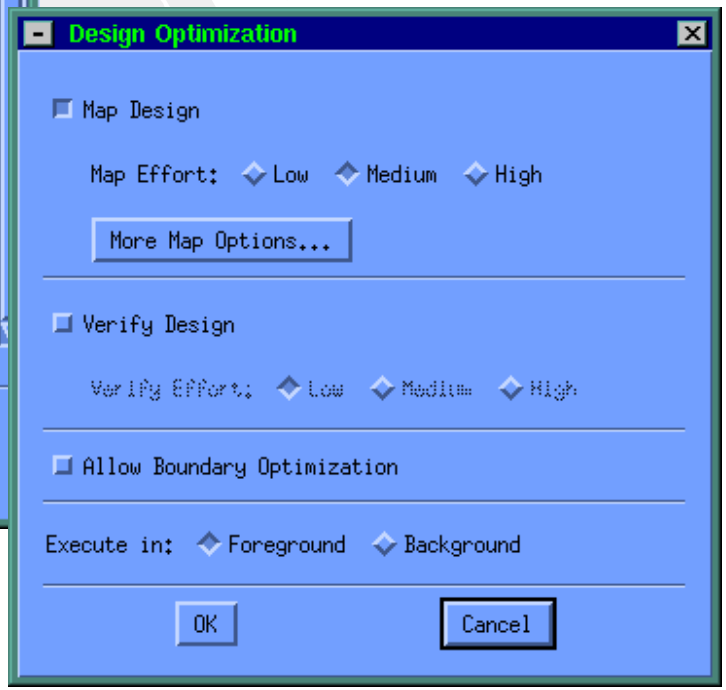
- Syntax error
  - *verilog 語法錯誤*
- Unsupported statements
  - *delay, initial, repeat, wait, fork, event, deassign, force, release*
- Unsupported definitions and declarations
  - *primitive, time, event, trand, trior, tri0, trireg*
- Unsupported operators
  - *=== and !==*
  - *Division operator (/)*
  - *Modulus operator (%)*
- Unsupported gate-level constructs
  - *nmos, pmos, cmos, rnmos, rpmos, rcmos*
  - *pullup, pulldown*
  - *rtran, tranif0, tranif1, rtranif0, rtranif1*

CCU EE VLSI Lab

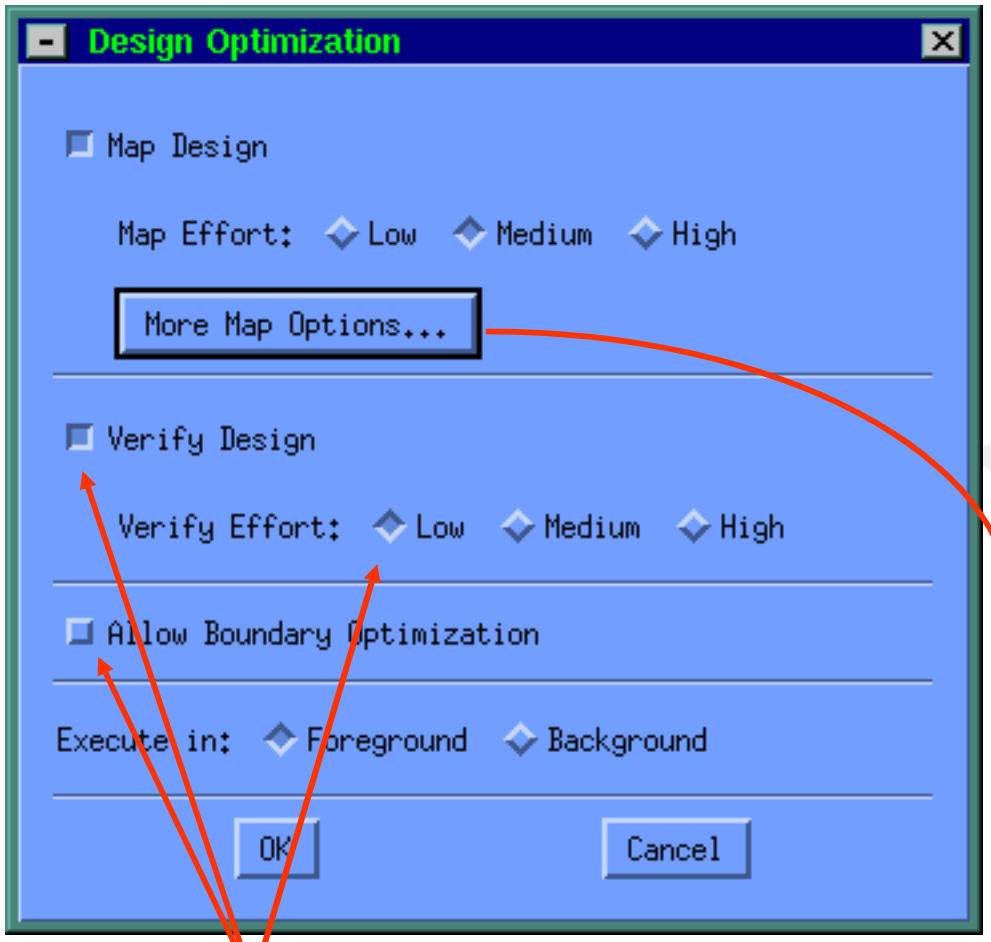
Default Synthesis (cont.)



由 Tools → Design Optimization 出現 Design Optimization window。



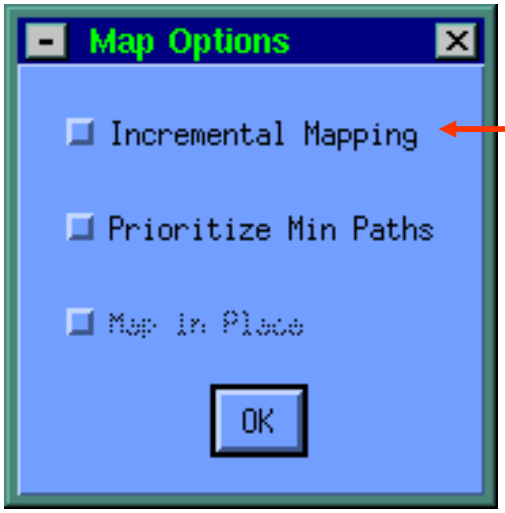
Default Synthesis (cont.)



可依照需要點選小 icon 選擇合成的方式

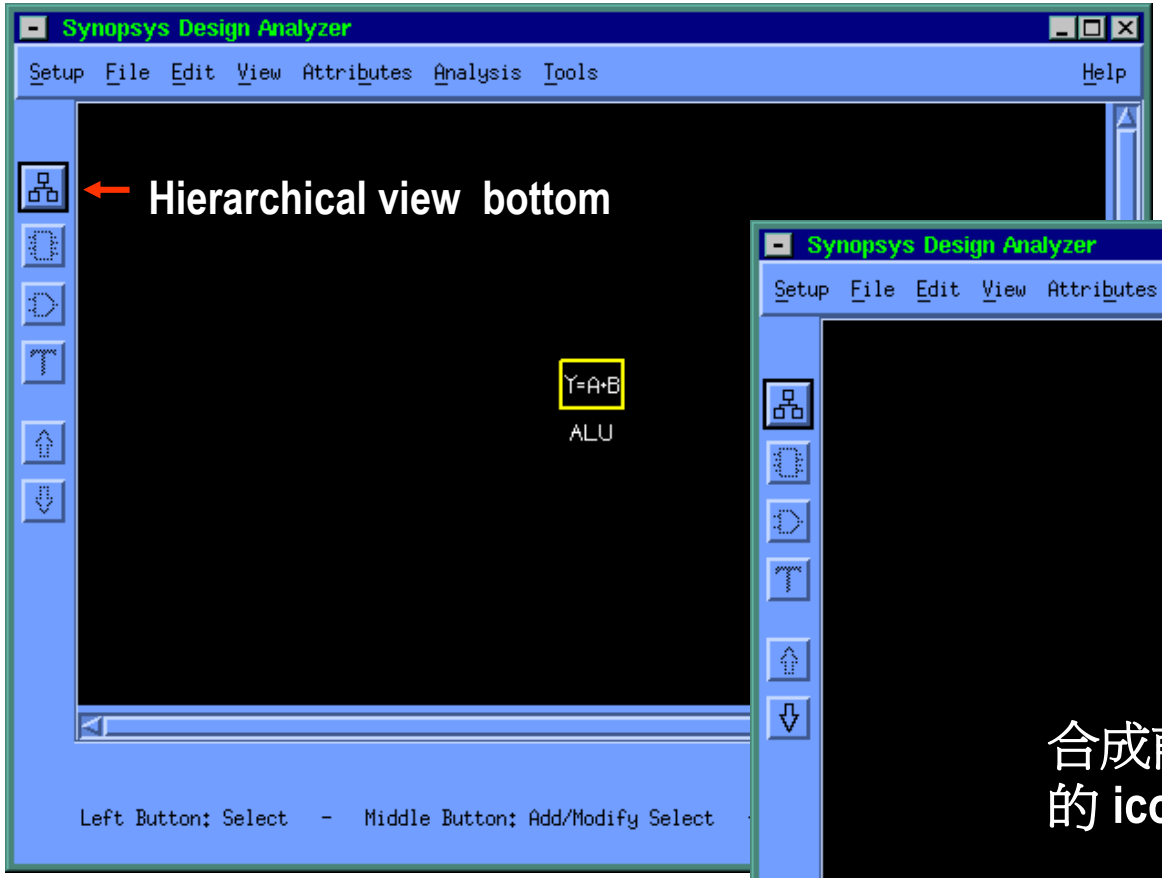
: 最原始的合成不需要點選任何設定，使用 **defaults** 的值即可。

: 選擇愈多的方法，合成的結果可能更好．但是，相對的是必須花費更長的時間.



命令 **SYNOPSYS** 在 **mapping** 時嚐試所有的 **cells** .

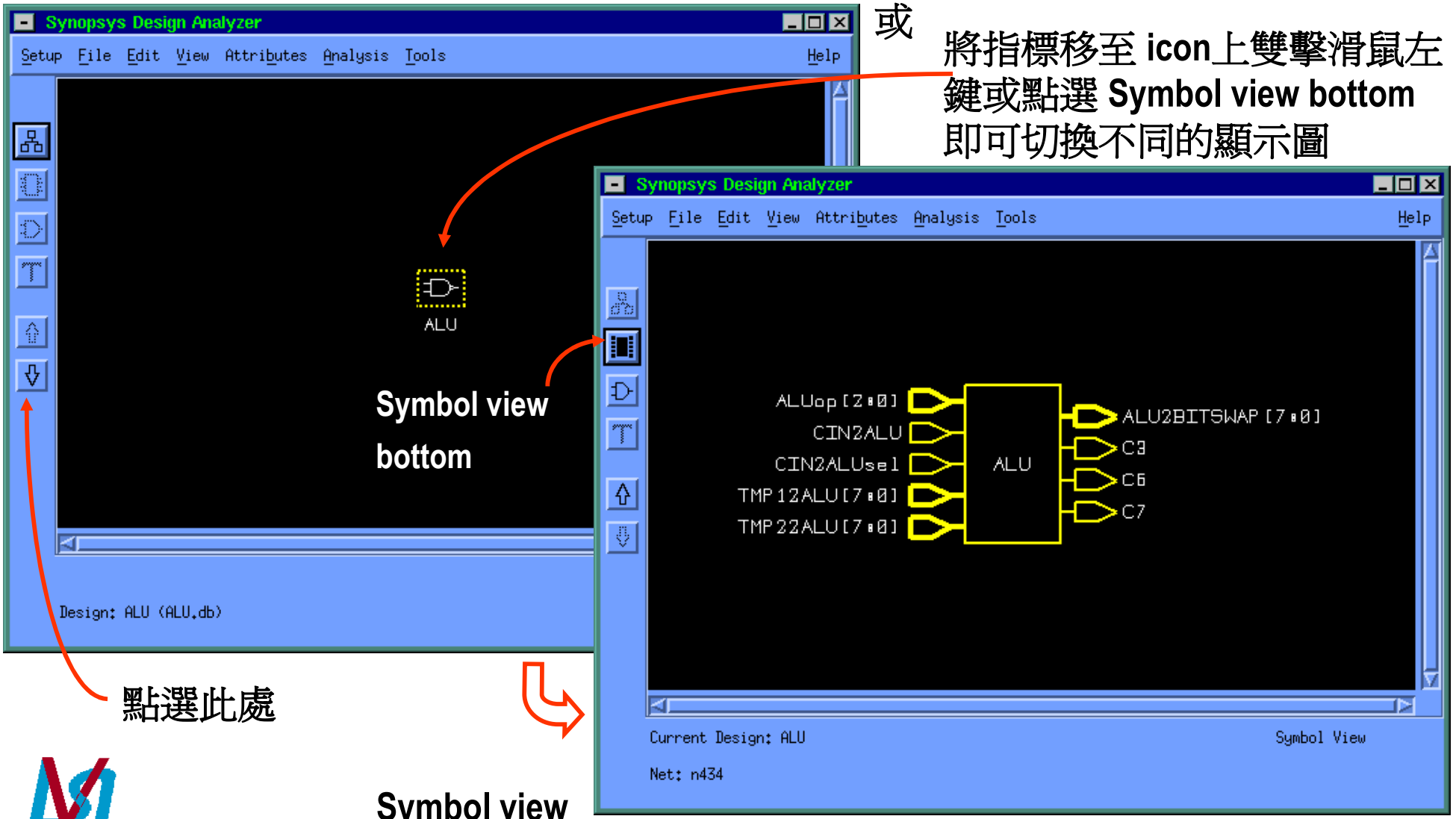
Default Synthesis (cont.)



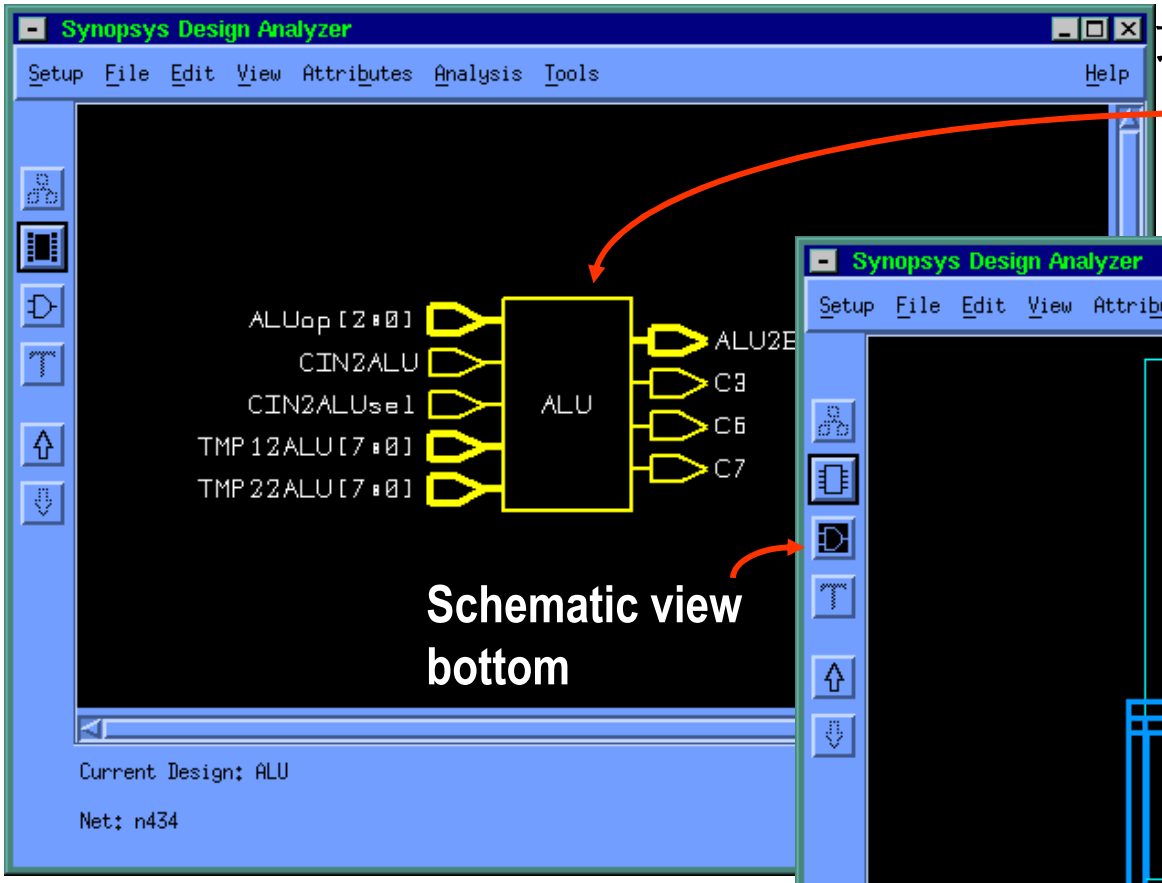
合成前

合成後(Hierarchical view)

Default Synthesis (cont.)

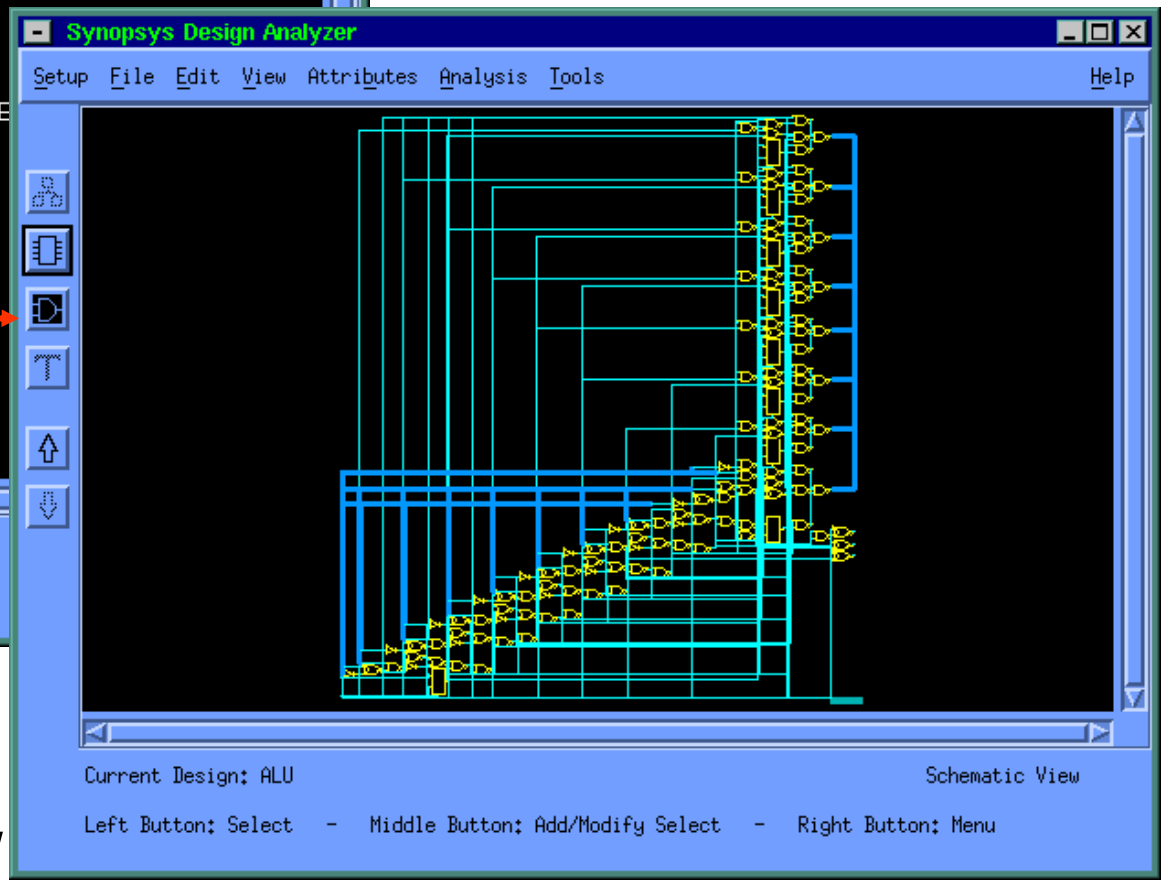


Default Synthesis (cont.)



或

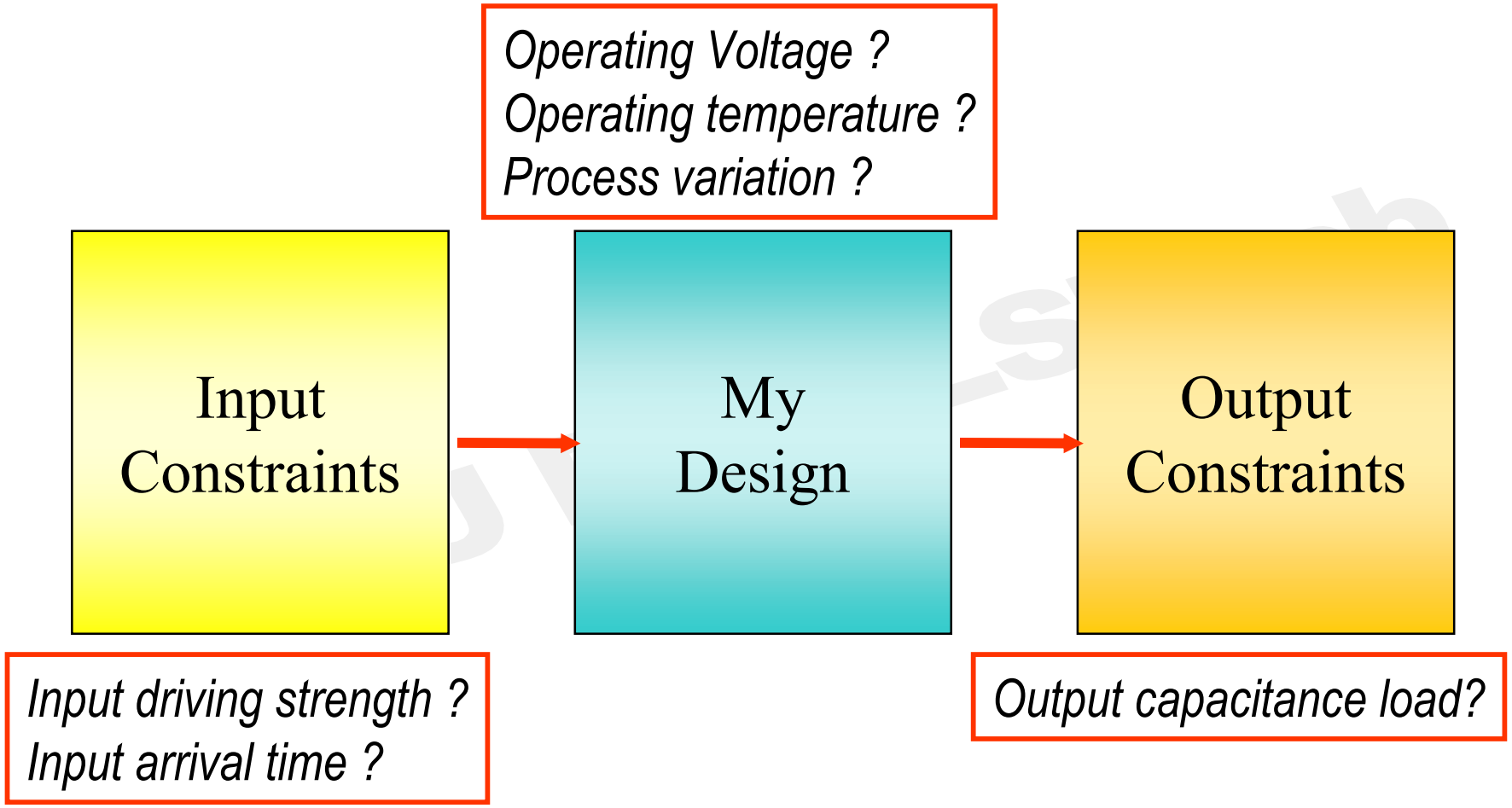
將指標移至 icon上雙擊滑鼠左鍵或點選 Schematic view bottom 即可切換不同的顯示圖



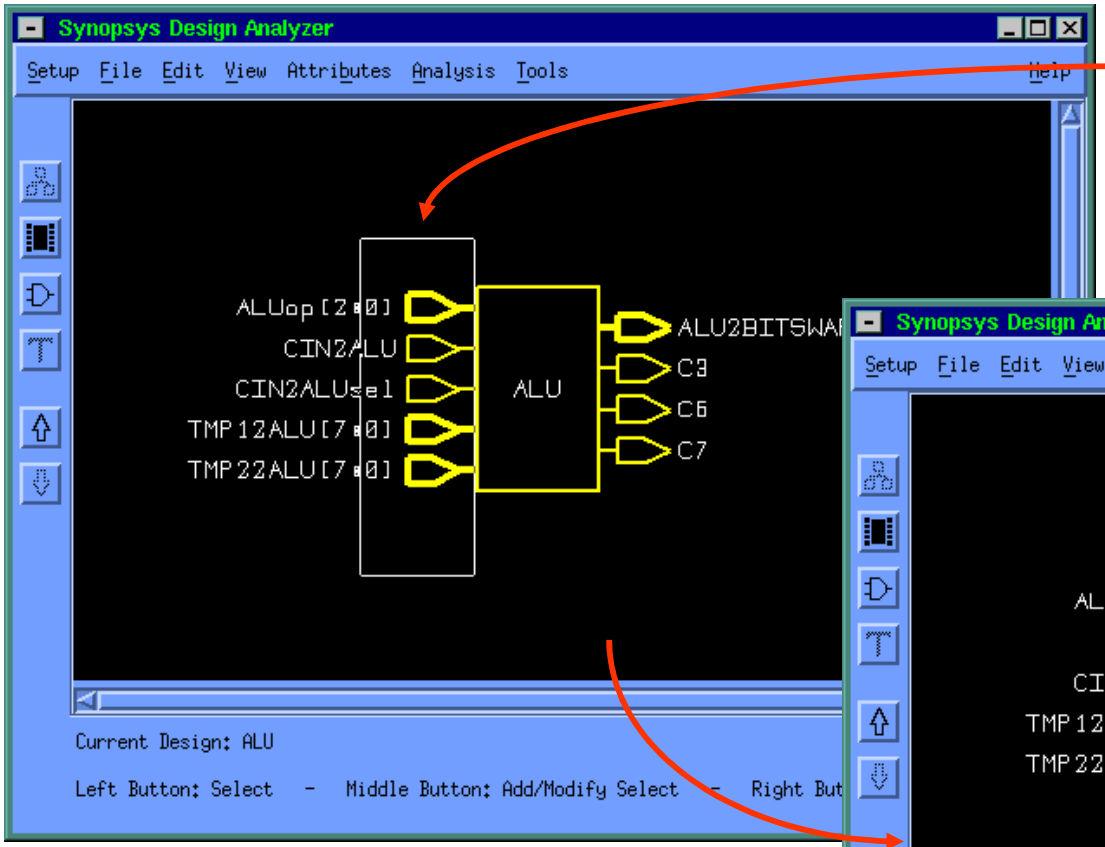
Schematic view

- 到此，Synopsys 已經爲我們的 verilog code 合成電路 (如上圖), 但是這電路是由 Synopsys 根據原始的設定合成的 . Synopsys 是假設 input drive is infinite , output capacitance is zero , 也沒有相關的 Operating conditions, Timing, Area 等資訊 (因爲在 .synopsys_dc_setup 中並沒有設定)。這些都不是真實的情況 , 所以我們需要針對我們的 Spec. 設定我們設計上的 Constraints , 使 Synopsys 能依據我們的需要將 Design 做最佳化 (Optimization) .

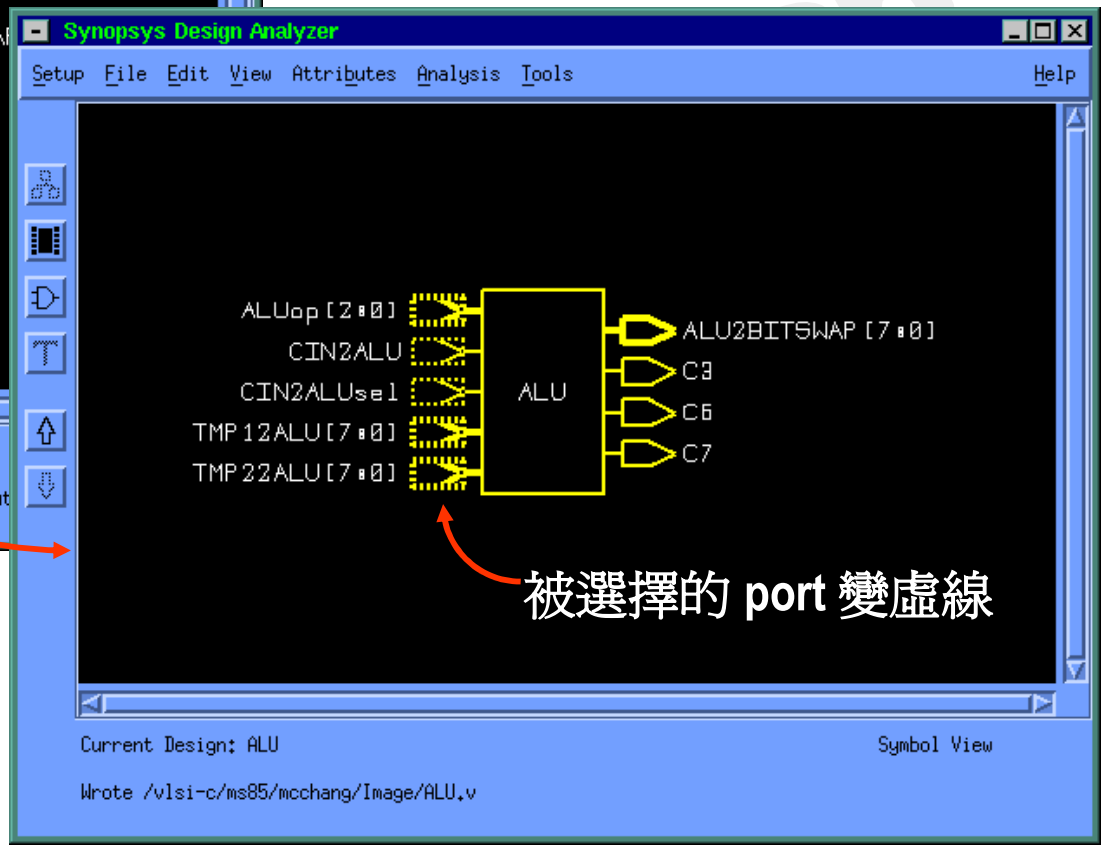
Setting Design SPEC.



Setting Operating Environment

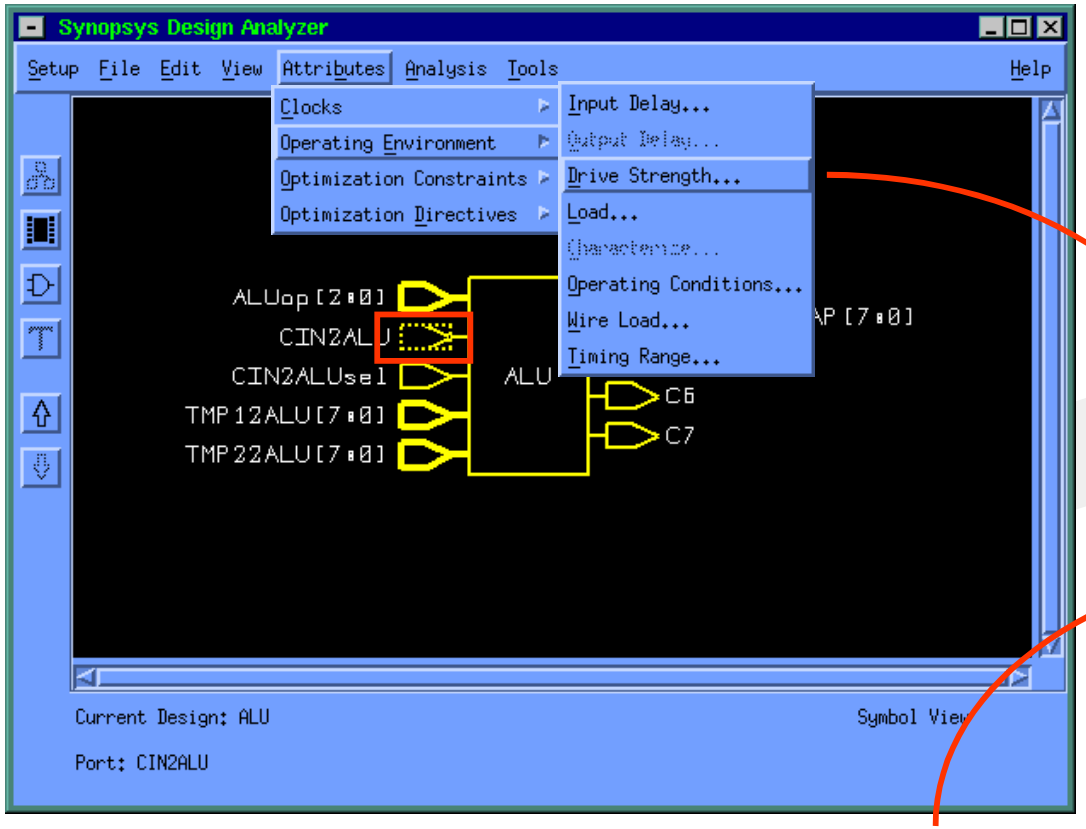


將游標移至視窗中一點使用滑鼠左鍵按住後拖曳滑鼠即出現範圍框，然後放開滑鼠，就可圈選物件。被選擇之物件以虛線狀態顯示。（如下圖）



：將游標移至欲點選的物件上按住 **Shift** 鍵後，再用滑鼠左鍵點選也可以點選物件或取消選擇（單用滑鼠中鍵亦可）。

Setting Operating Environment (cont.)

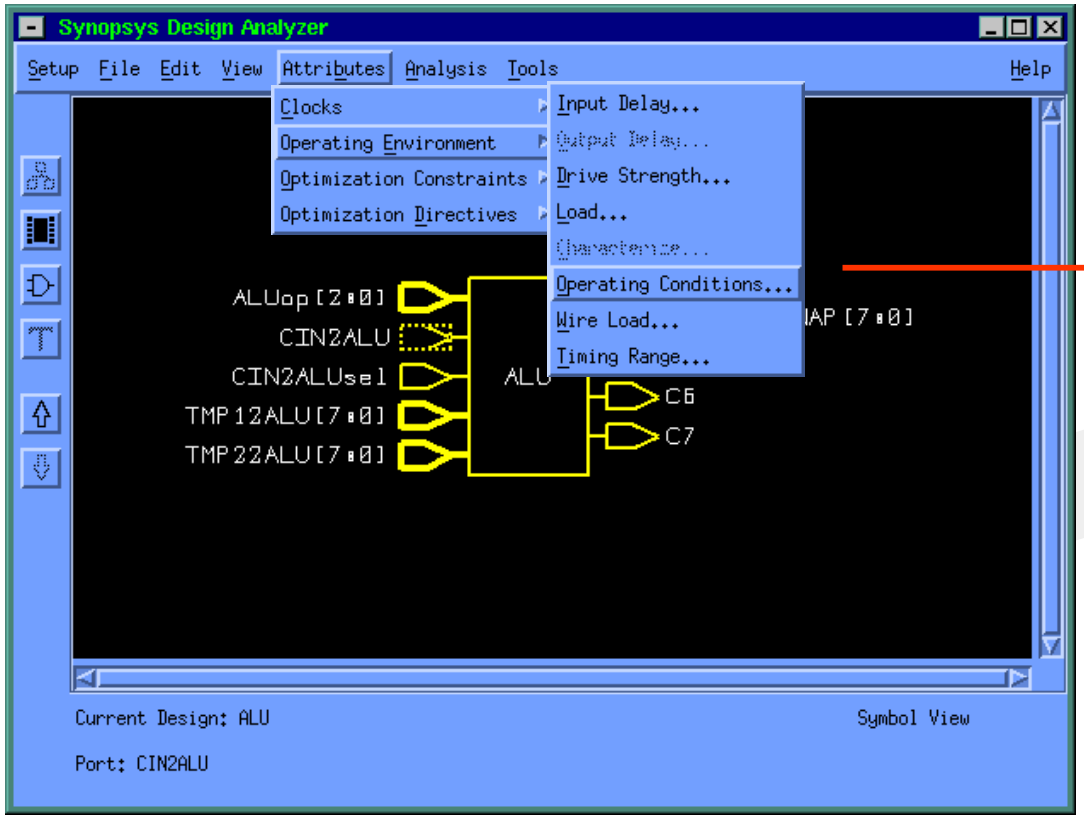


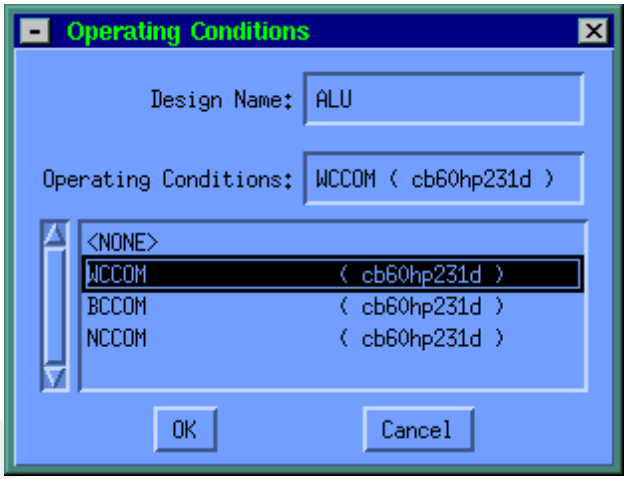
由
Attributes → Operating Environment → Drive Strength 即出現設定 Drive Strength 的視窗。（如下圖）

若只選擇一個 port 時，此處會出現 port name 及原始的設定值。
填入欲設定的值後按 Apply

：
其他設定均如上圖所述。

Setting Operating Environment (cont.)



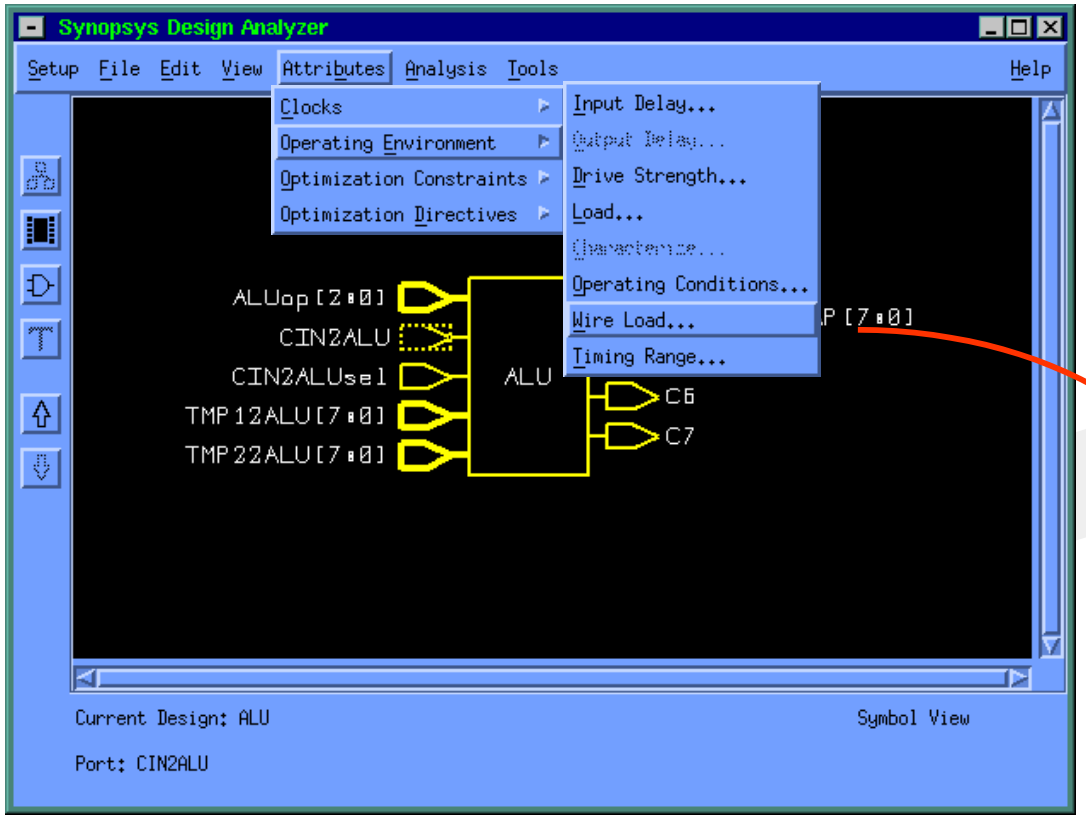


✍：選擇 Operating condition model. (delay model)

☺ ： *Cell Library 中有定義才有效*

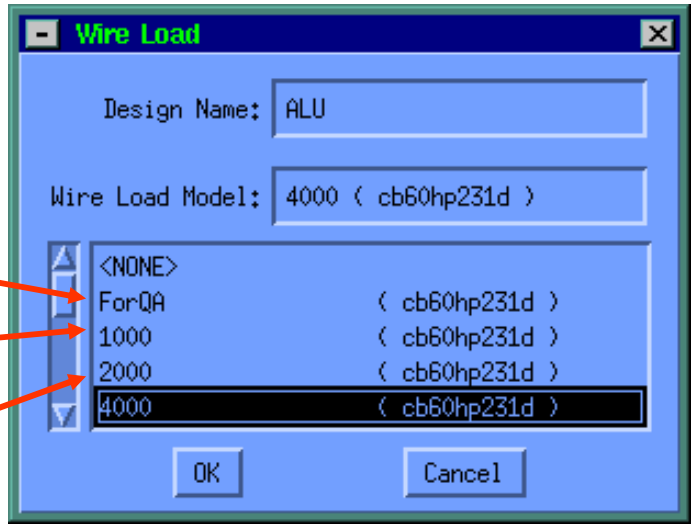
Name	Process	Temp	volt	Interconnection Model
WCCOM	1.33	70.00	4.75	worst_case_tree
BCCOM	0.92	0.00	5.25	best_case_tree
NCCOM	1.00	25.00	5.00	balance_tree

Setting Operating Environment (cont.)



✍：選擇 設計電路的格局大小。*最好選大一點*。

☺ ： *Cell Library 中有定義才有效*

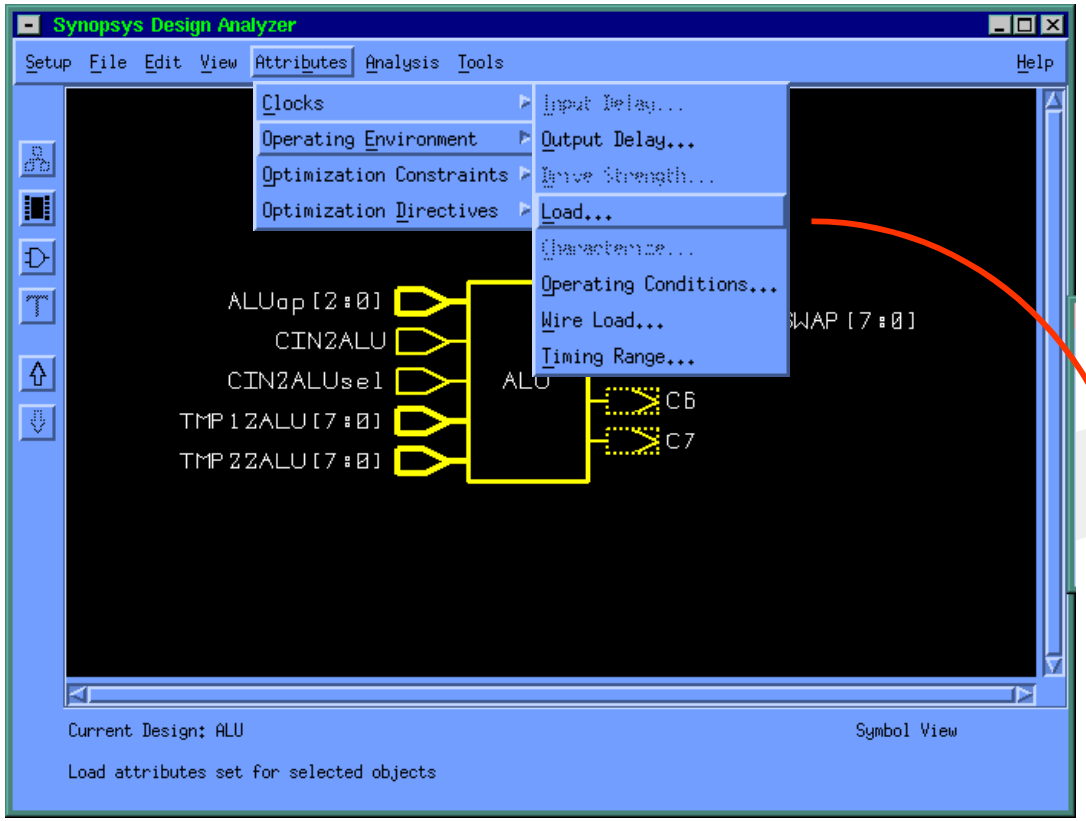


小於 1000 gates 的設計

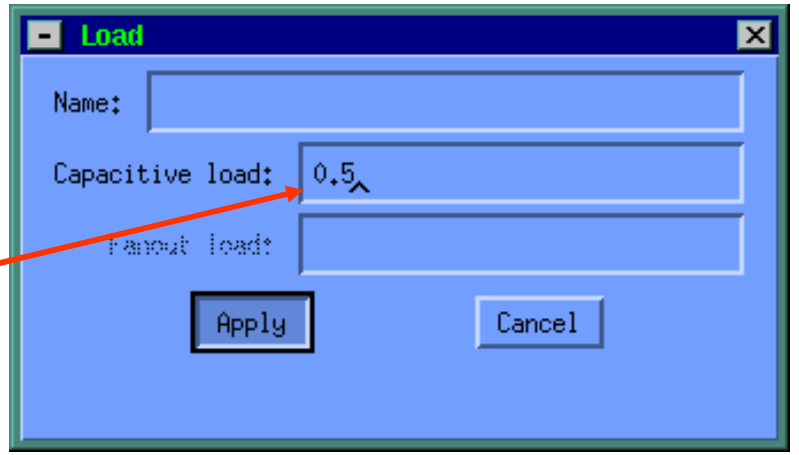
1000 gates 的設計

2000 gates 的設計

Setting Operating Environment (cont.)



由
Attributes → Operating Environment → Load 即出現設定 loading 的視窗。（如下圖）



填入欲設定的值後按 **Apply**

：若 output loading 太大, Synopsys 會根據 cell library中的定義適當的加入 buffer (cell library 有定義才有效)

Setting Design Constraint

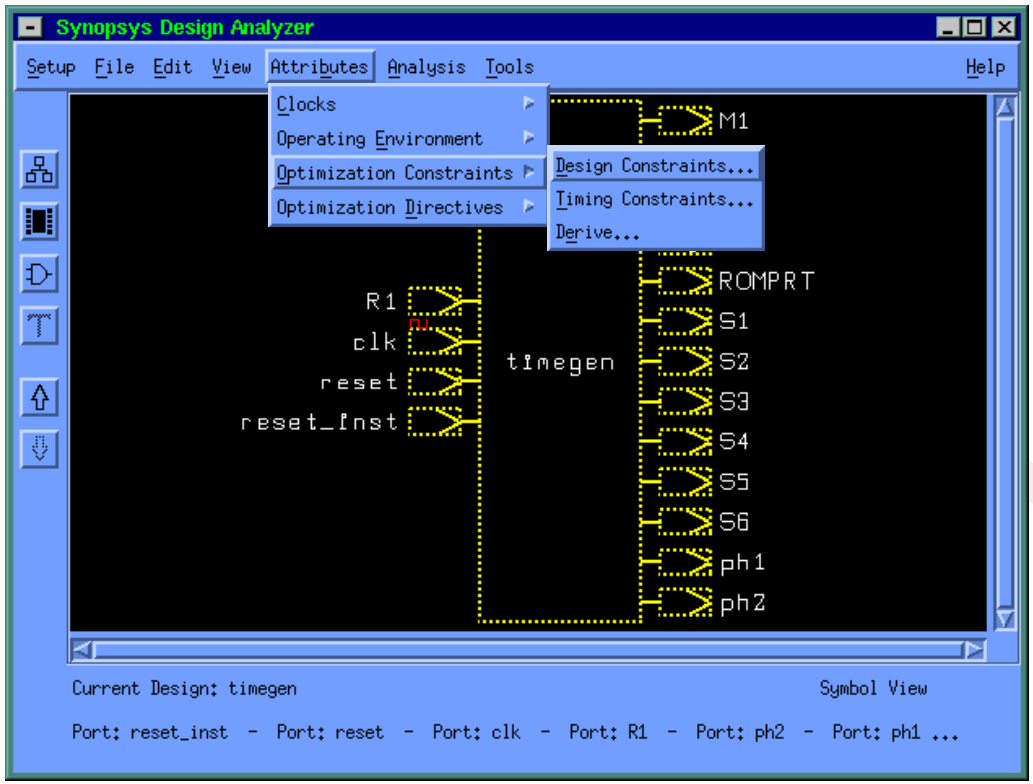
- Optimization constraints
  - Maximum delay , Minimum delay
    - For combinational circuits
      - Select the start and end points of the concerned paths
    - For sequential circuits
      - Specify the clock
      - Setting input delay
      - Setting output delay
  - Maximum area

CCU EE VLSI Lab

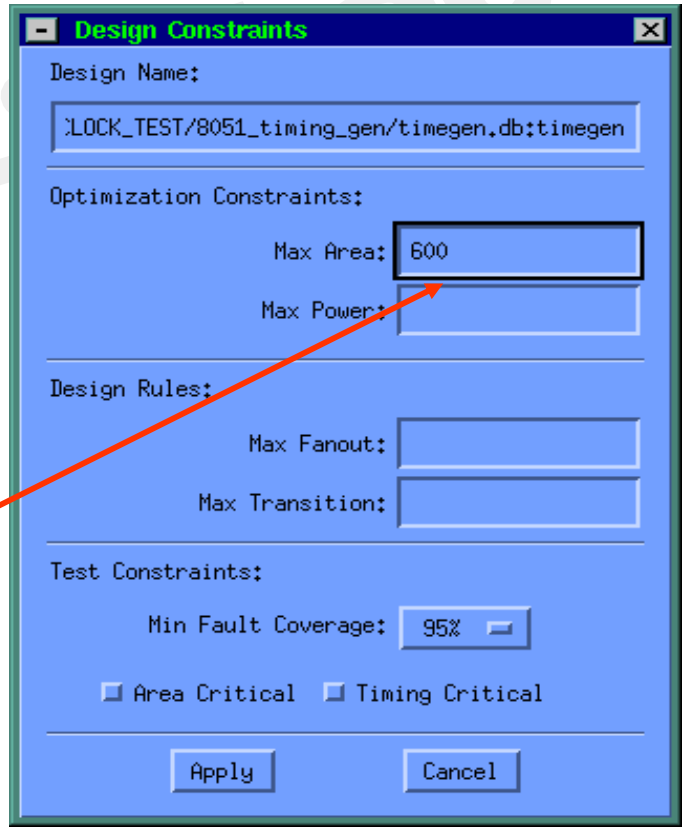
Combinational circuits

CCU EE VLSI Lab

Setting Constraints (cont.)



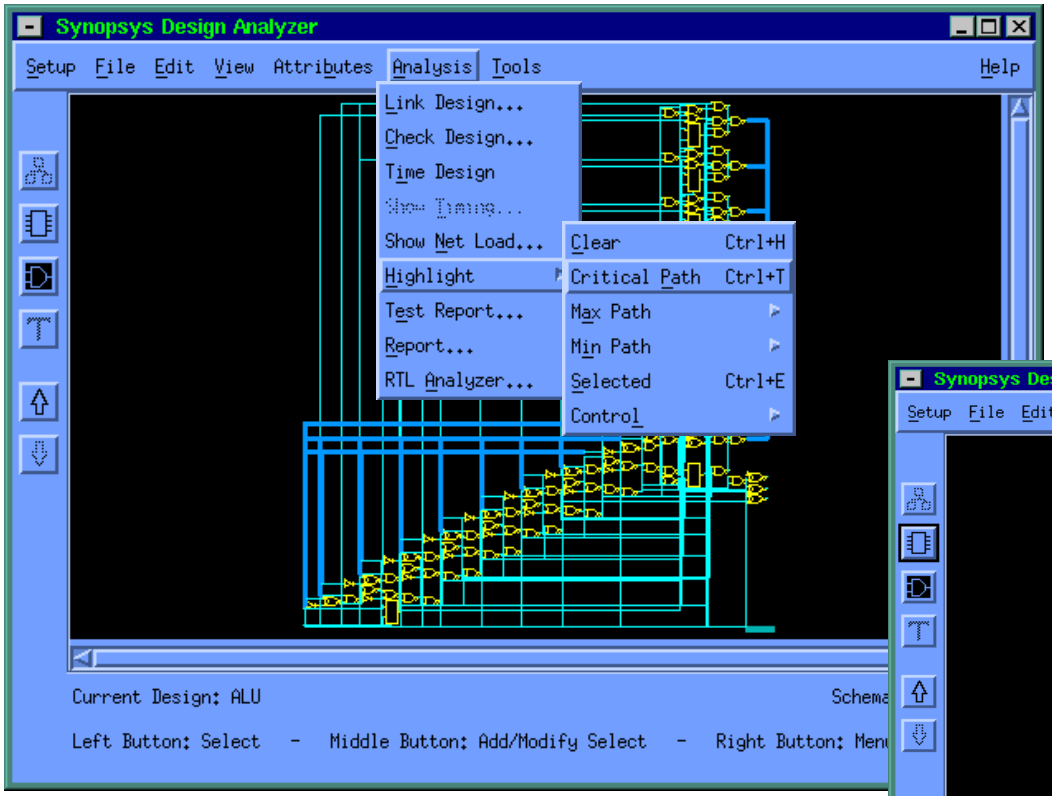
由 **Attributes → Optimization Constraints → Design Constraints** 即可進入設定 **Design constraints window**



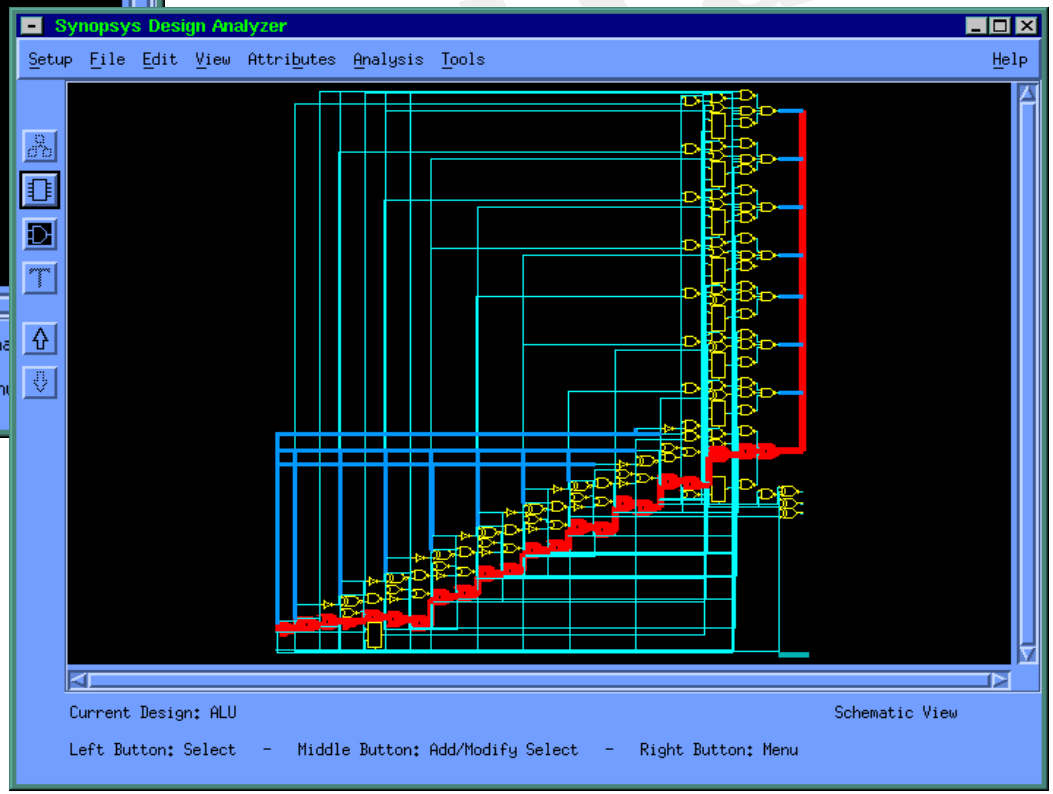
在 **Max Area** 欄中 **Area** 大小 (單位 : **gates**). 因爲沒有 **license** , 所以無法對 **power** 設定 **constraint** .

∽ : 視情況而定, **default** 是不需設定

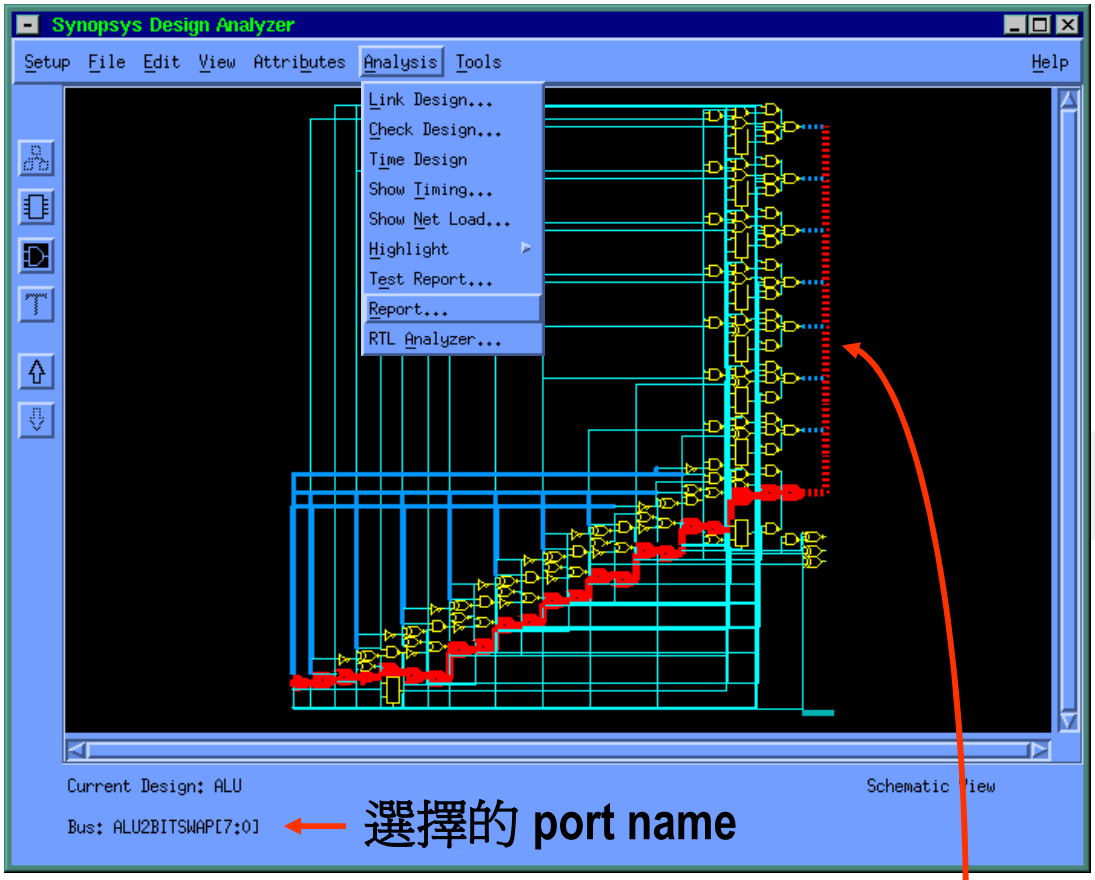
Combinational circuits



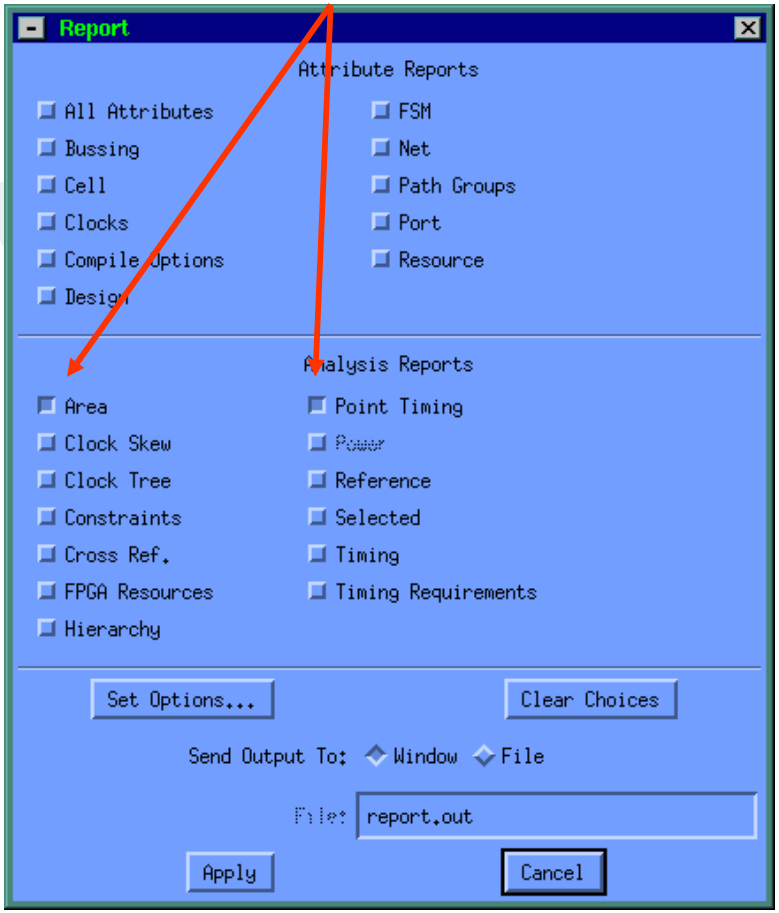
所有環境變數設定完後, 由 **Analysis → Highlight → Critical Path ,** Synopsys 會將電路的 Critical Path 顯示 (如圖)



Combinational circuits (cont.)

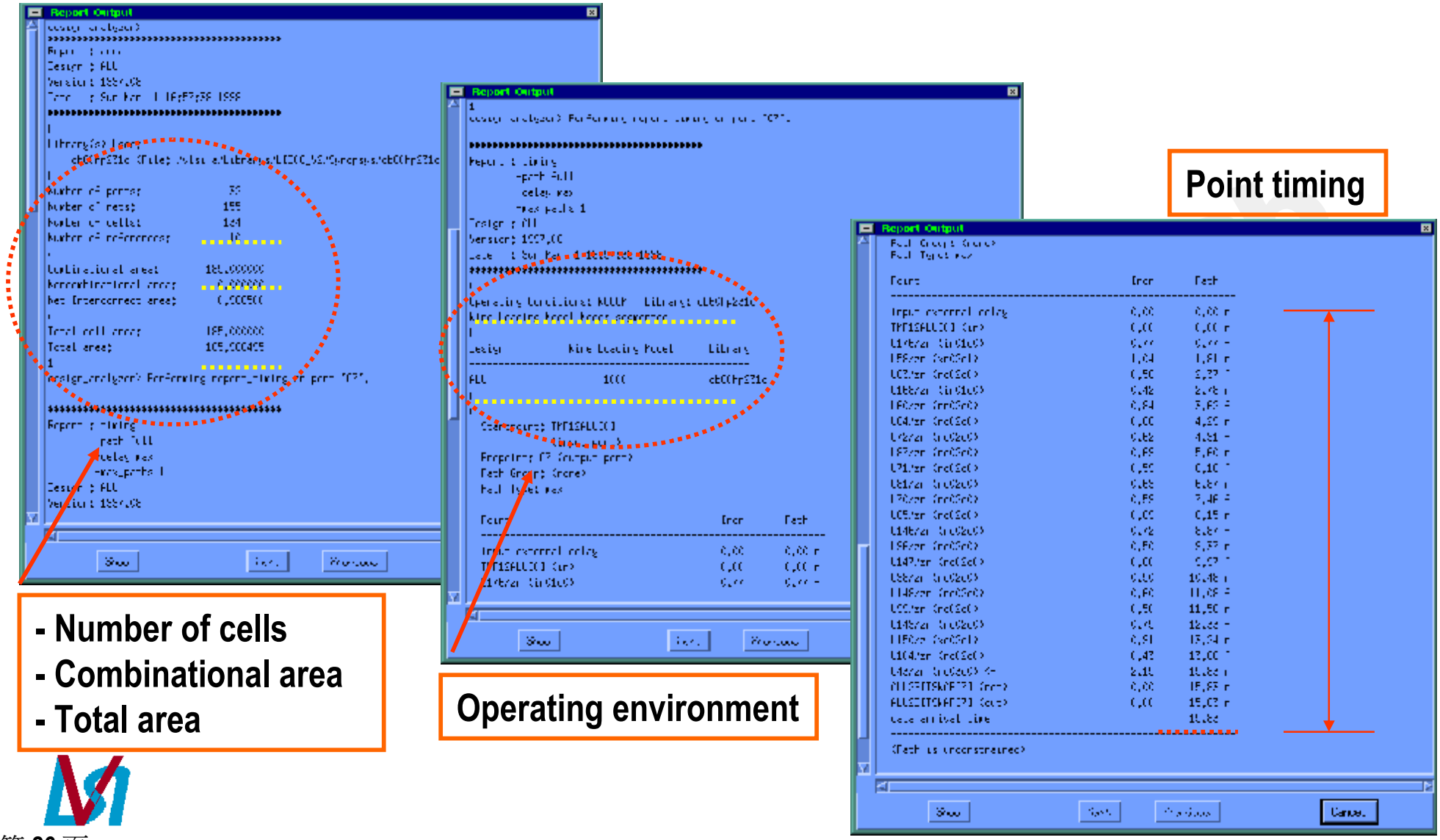


再由
Analysis → Report 即出現 **Report** 視窗(如圖), 點選需要的項目後按 **Apply**

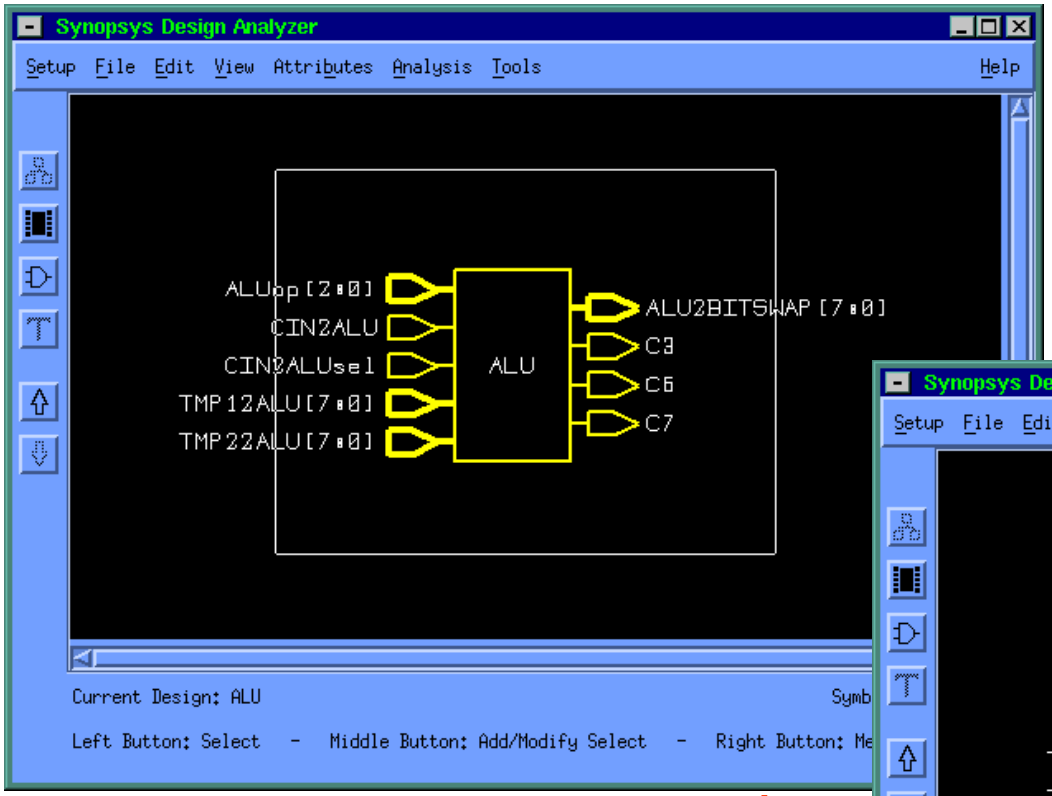


先點選 **Critical path output port** (點選的 **port** 呈虛線狀,如上圖)

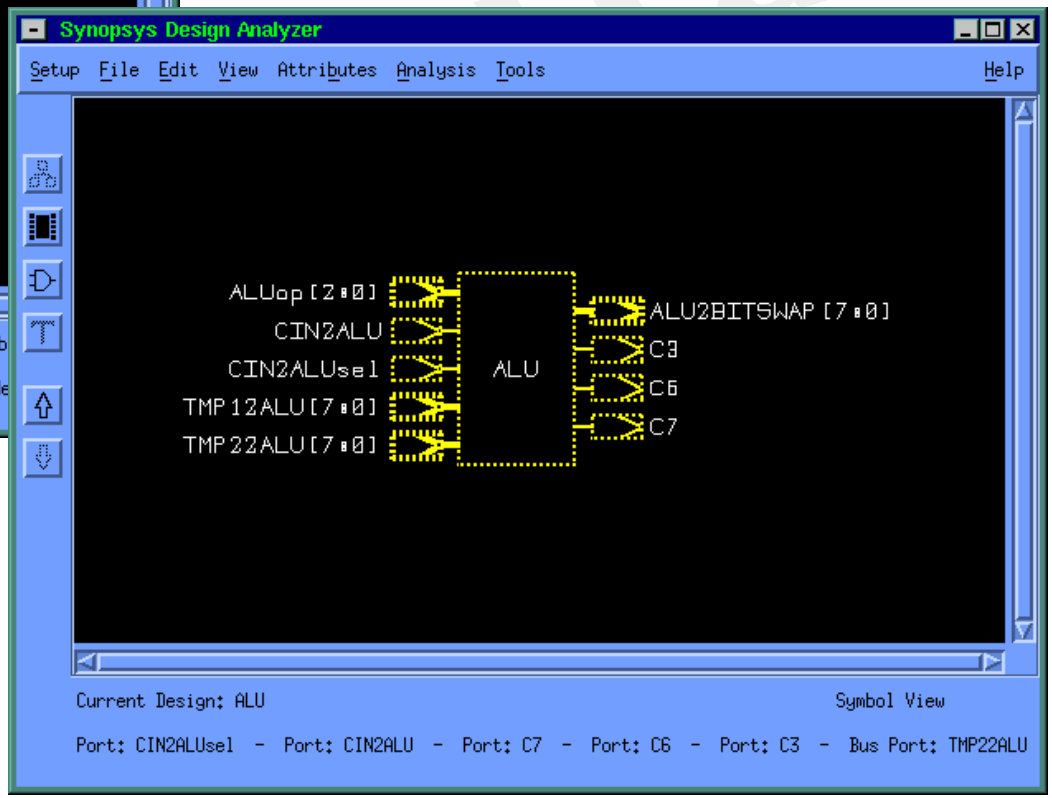
Combinational circuits (cont.)



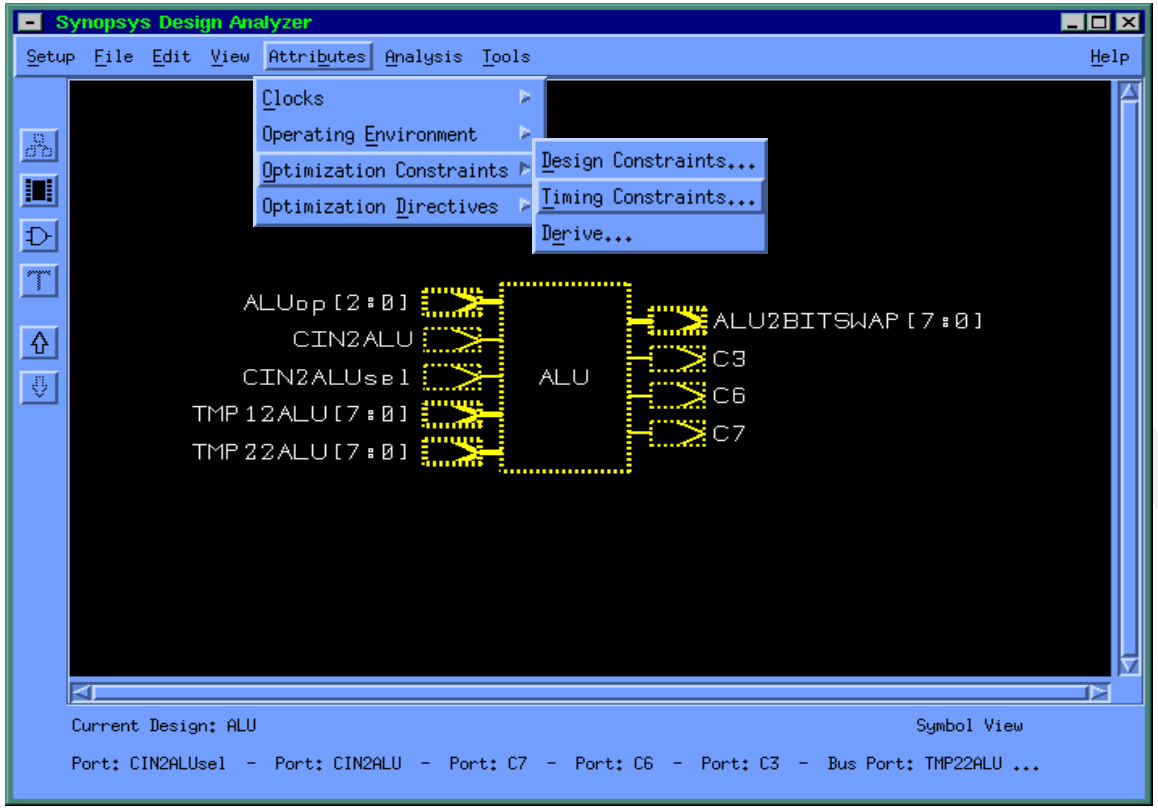
Combinational circuits (cont.)



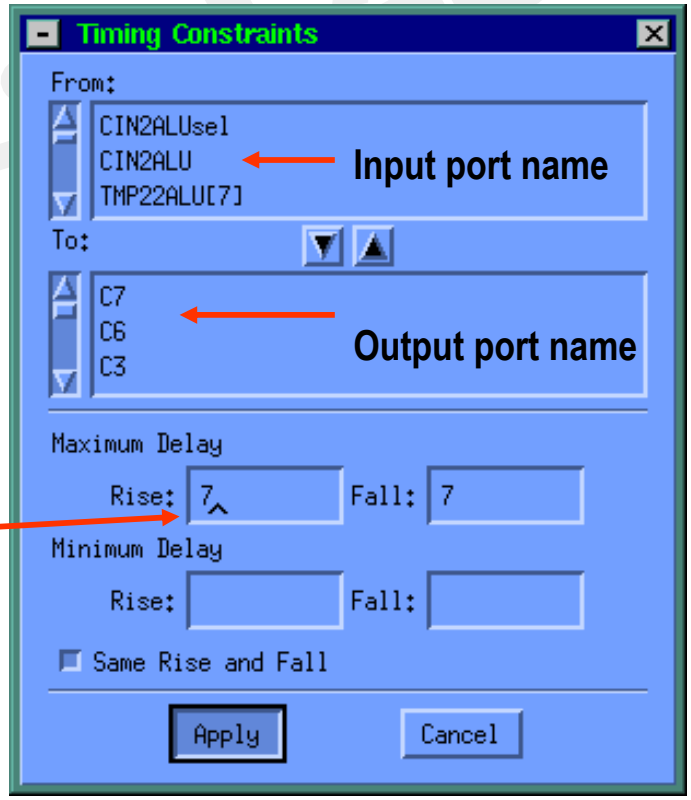
設定 **constraints** 時, 必須先點選欲設定的 **port**, 圖例是圈選所有的 **ports**



Combinational circuits (cont.)



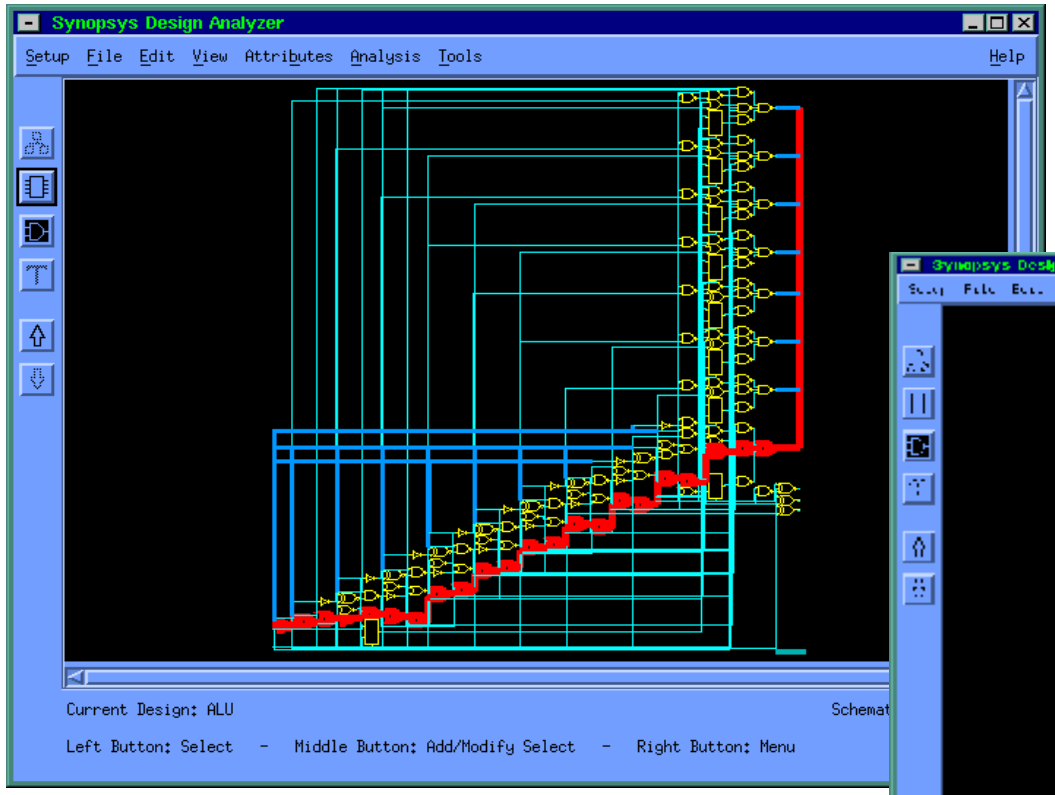
由 Attributes → Optimization Constraints → Timing Constraints 即可進入設定 timing constraints window



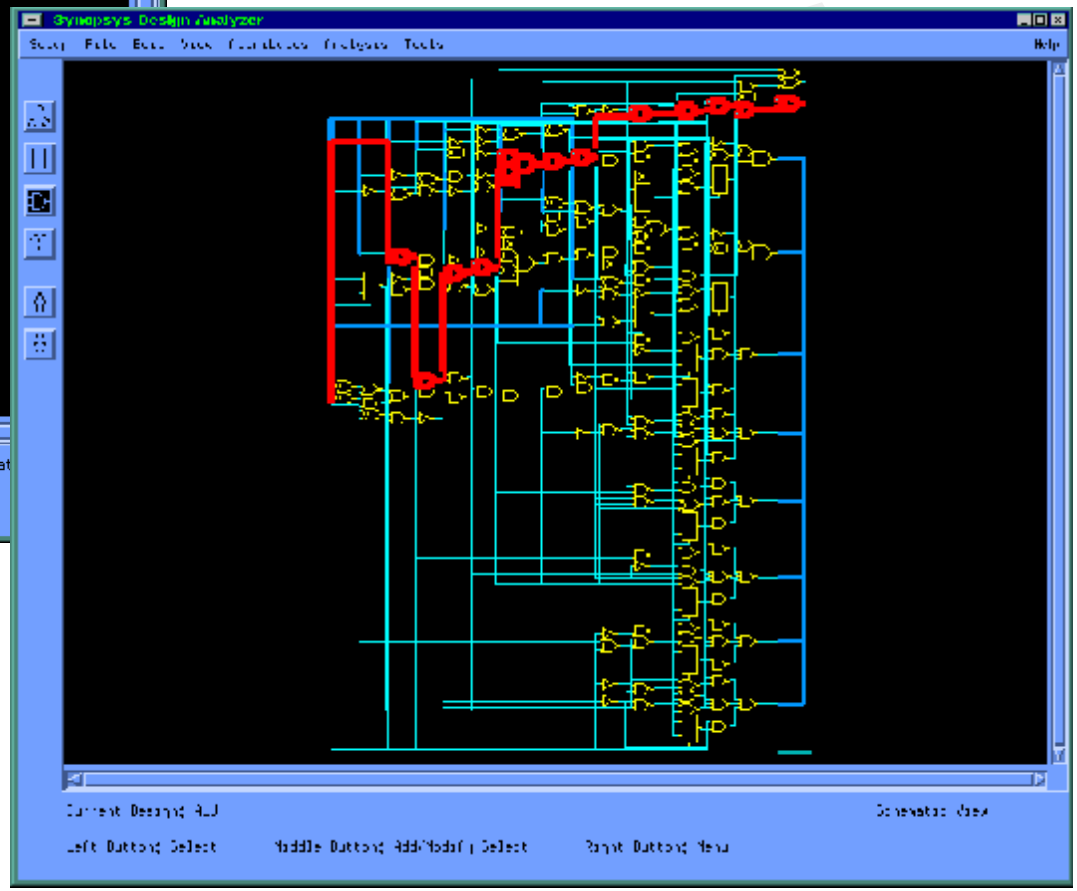
在 Maximum Delay 欄中填入 Input port 到 Output port 的 timing constraints

: 可針對單一 input 及 output 設定

Combinational circuits (cont.)



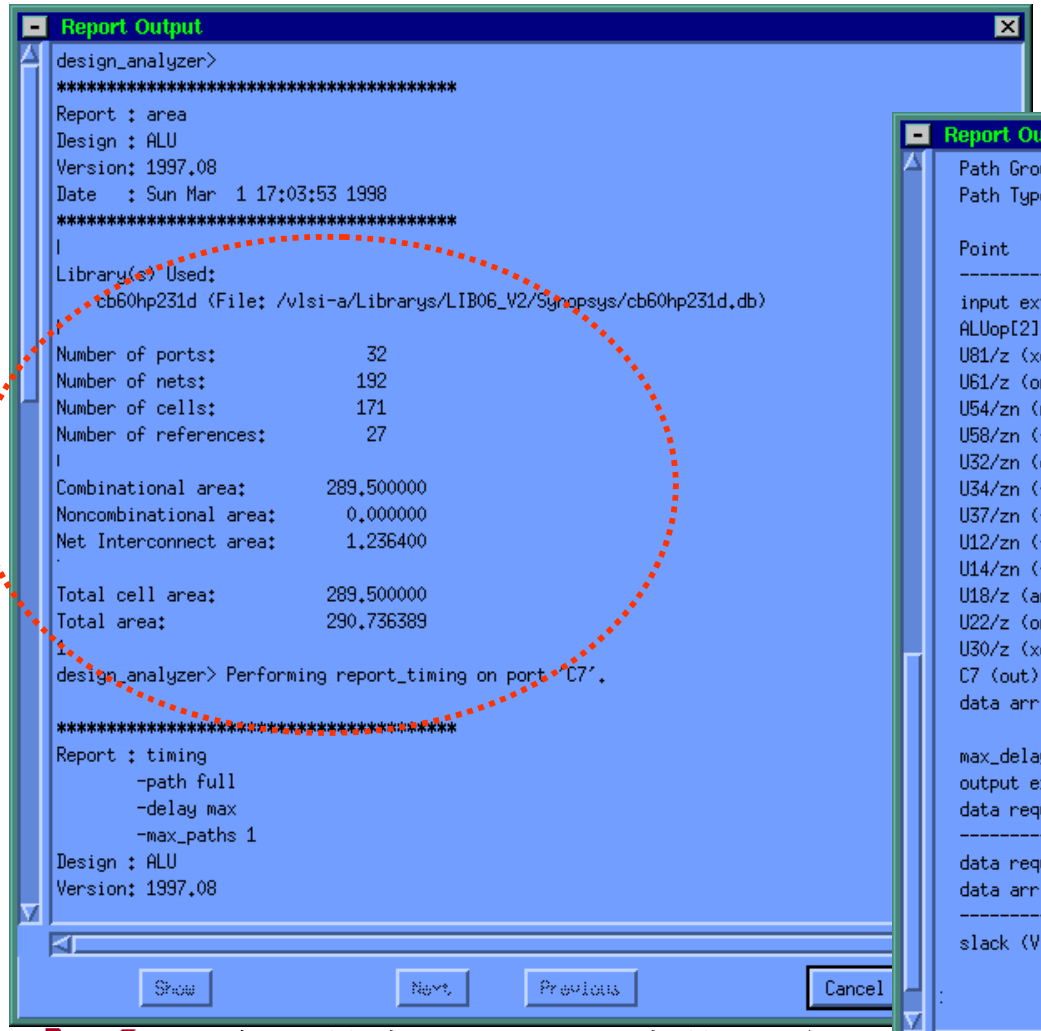
重新合成前

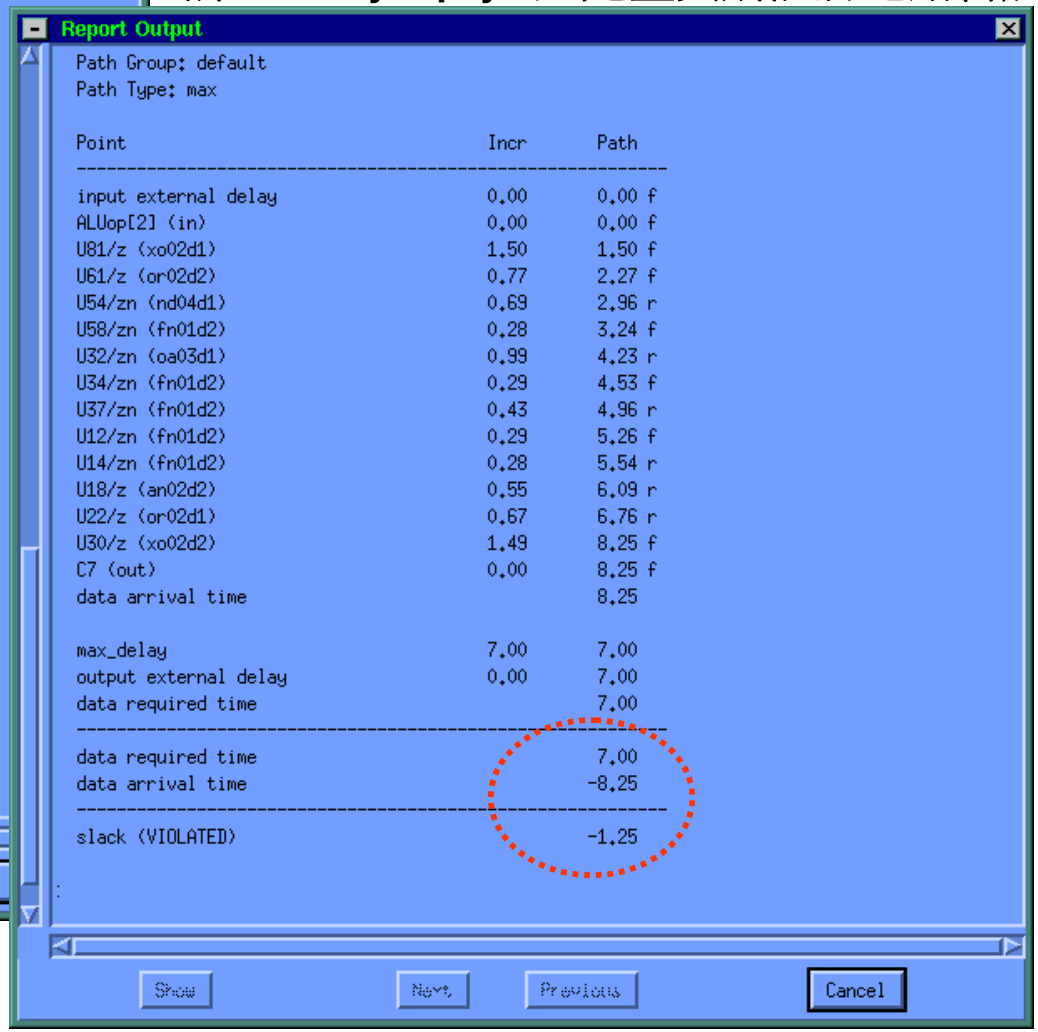


重新合成後

Combinational circuits (cont.)

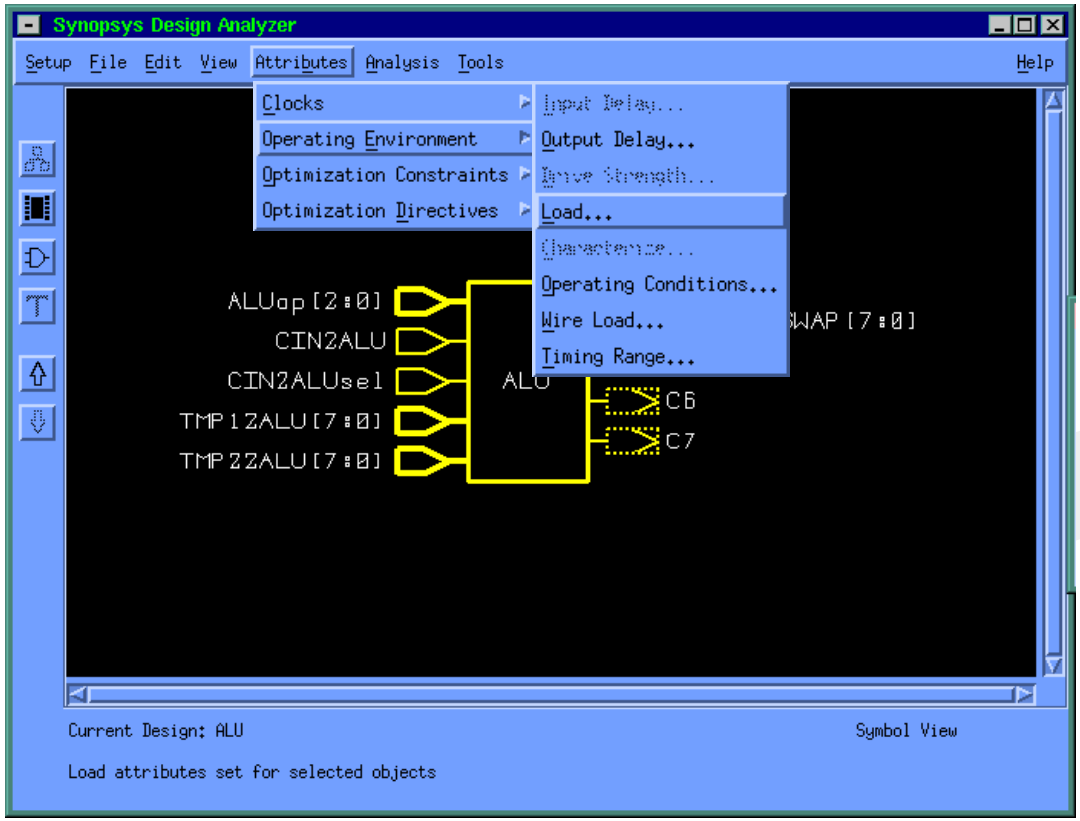
Timing 不符 Spec. 但是 Synopsys 並不會告訴 user. Synopsys 只是盡其所能將電路合成.



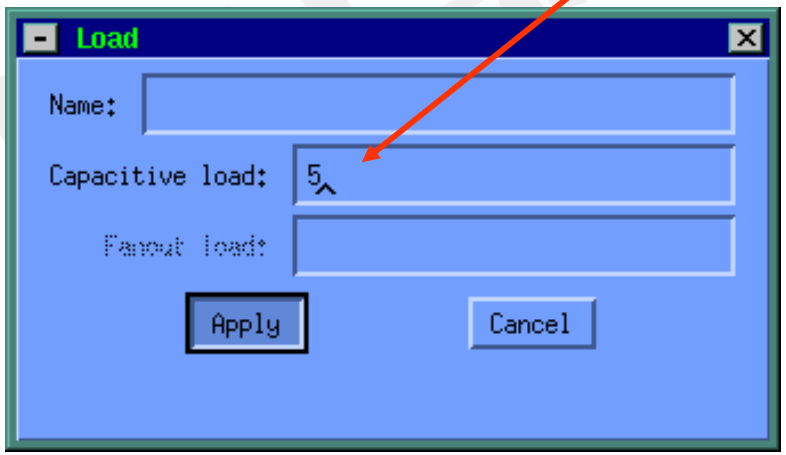


若無法符合 Spec. 就只有修正電路的設計或更換 codeing style .

Combinational circuits (cont.)



output loading 設定爲 5 pF 後, 再重新合成一次

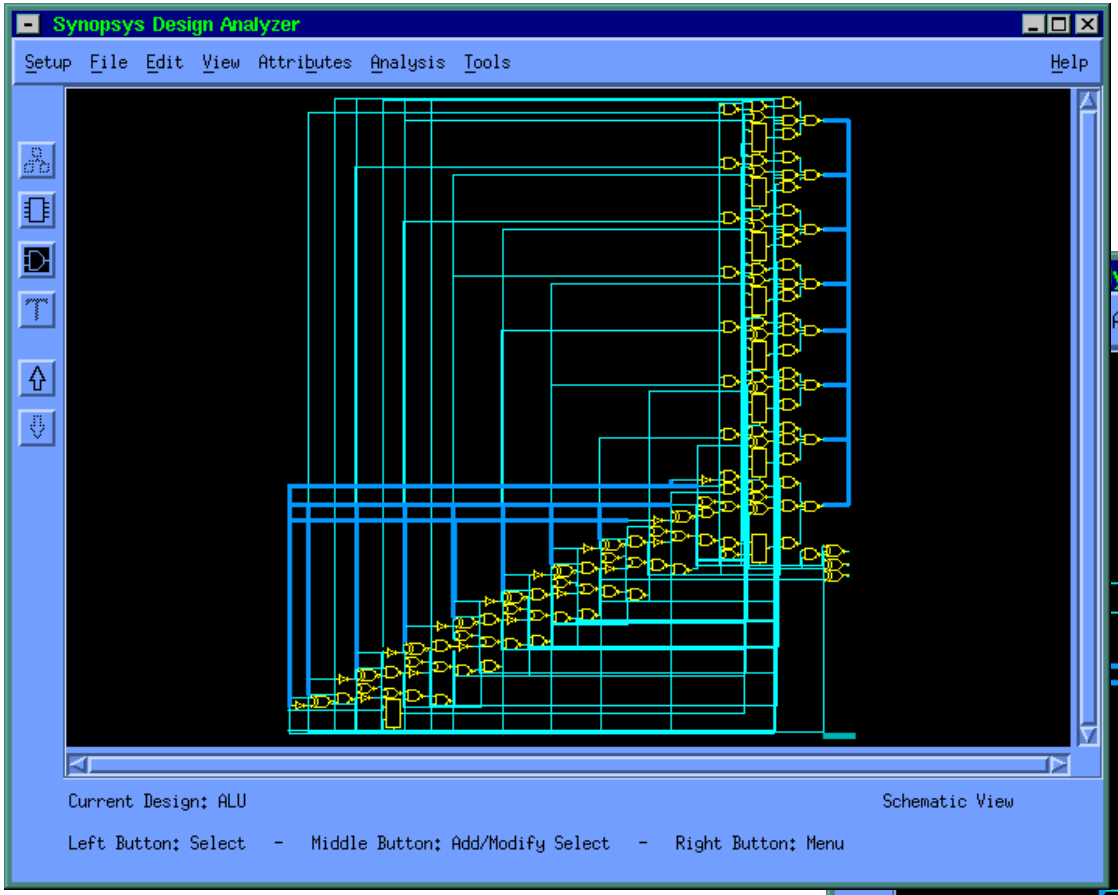


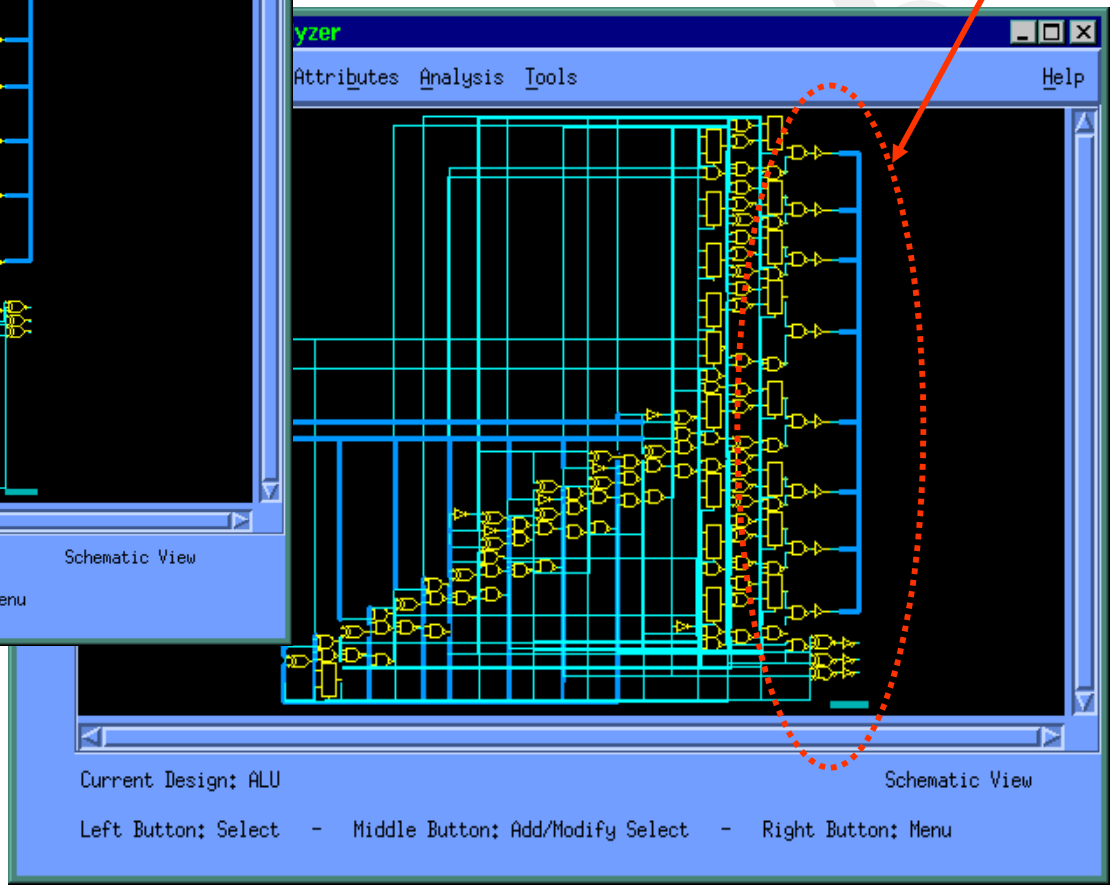
☍ : cell library 有定義才有效

Combinational circuits (cont.)

重新合成前

Output 增加了 buffer



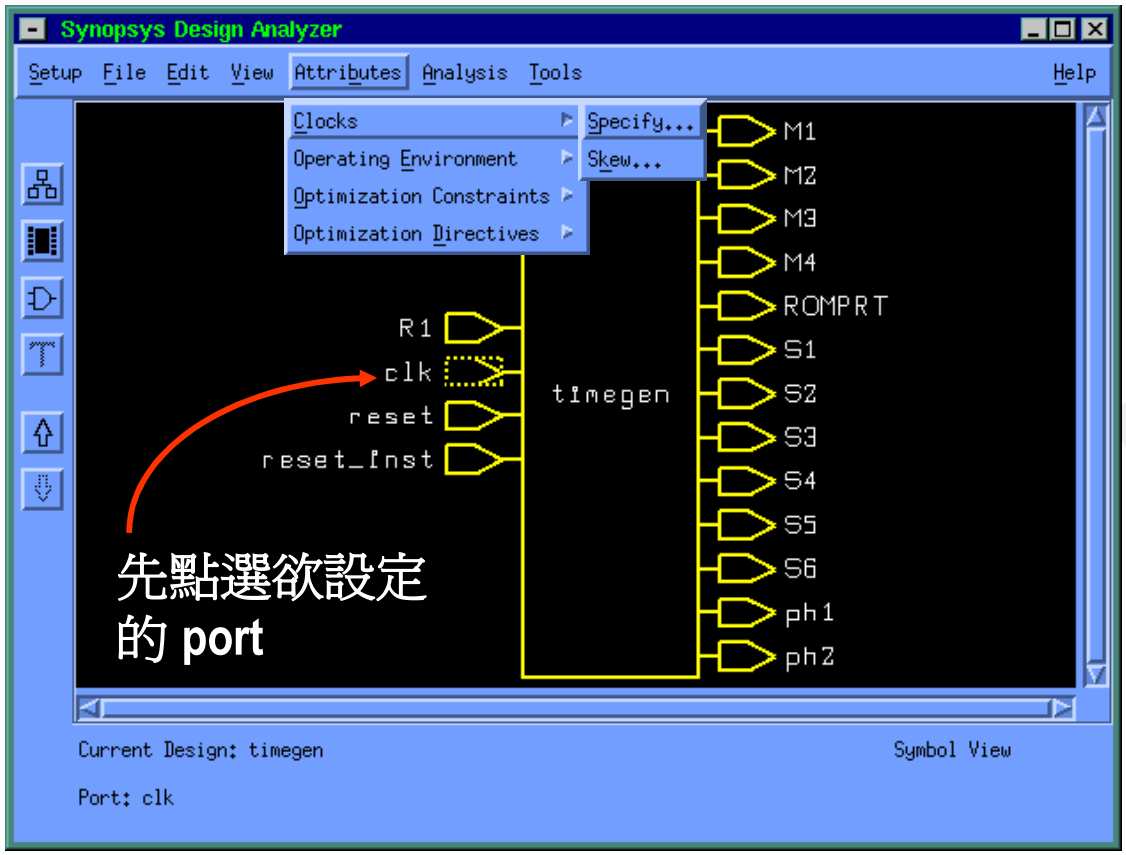


重新合成後

Sequential Circuits

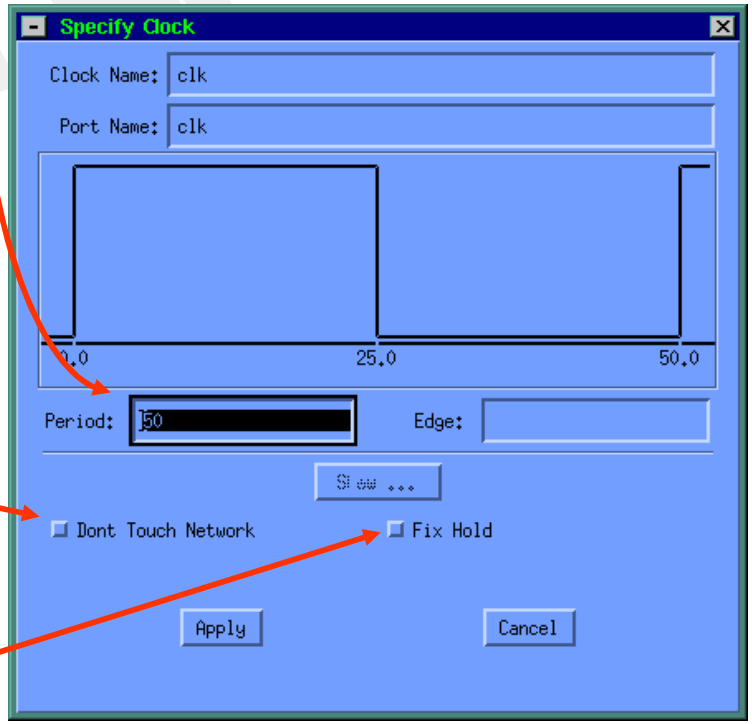
CCU EE VLSI Lab

Specify Clock



由 Attributes → Clocks → Specify
即出現設定視窗

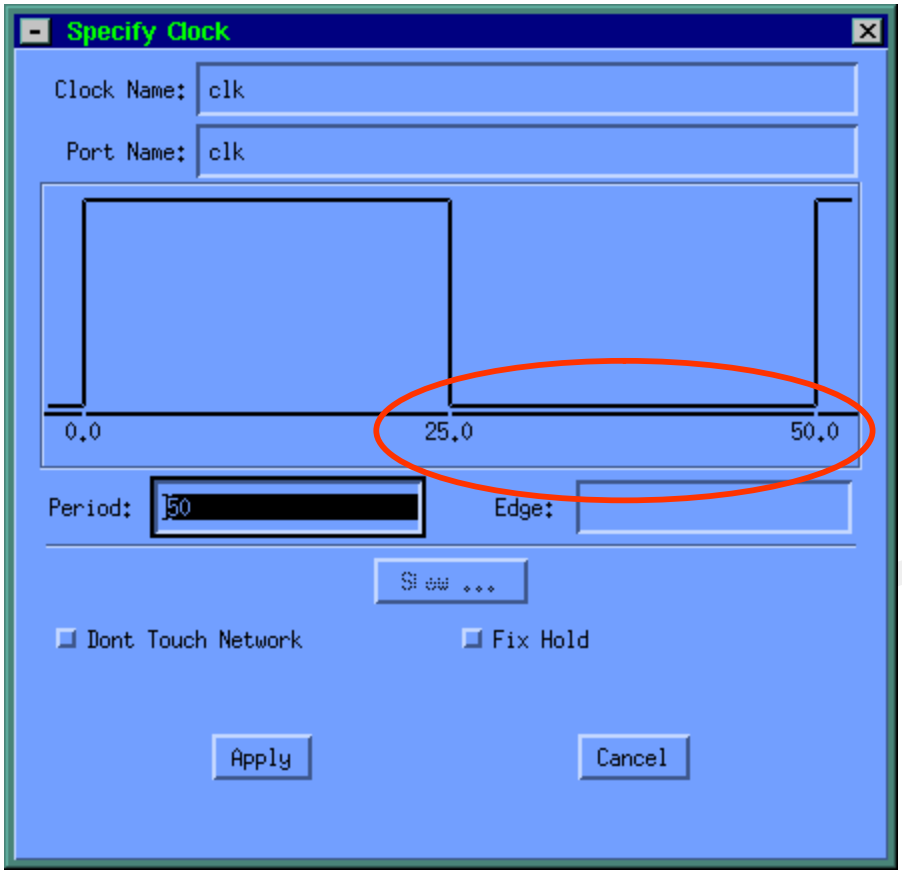
在 Period 欄中填入時間(單位：ns)

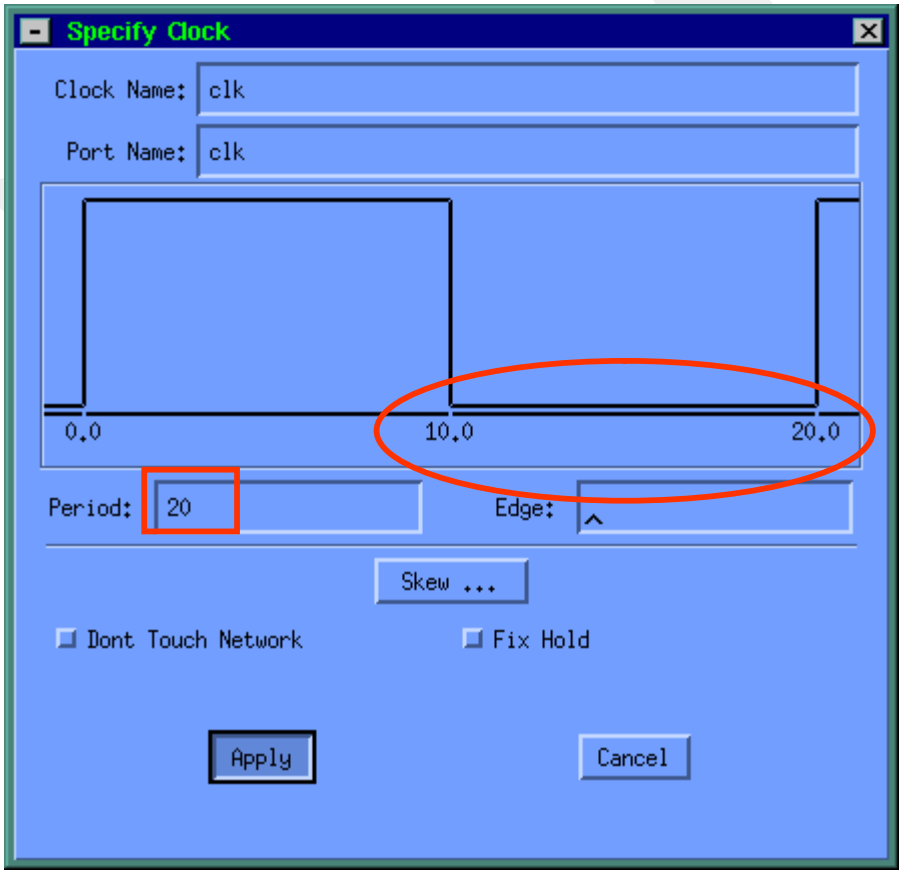


Don't Touch Network : ***Do not re-buffer the clock network***

Fix hold : ***Fix your hold time requirement in compile, create hold constraint for the clock***

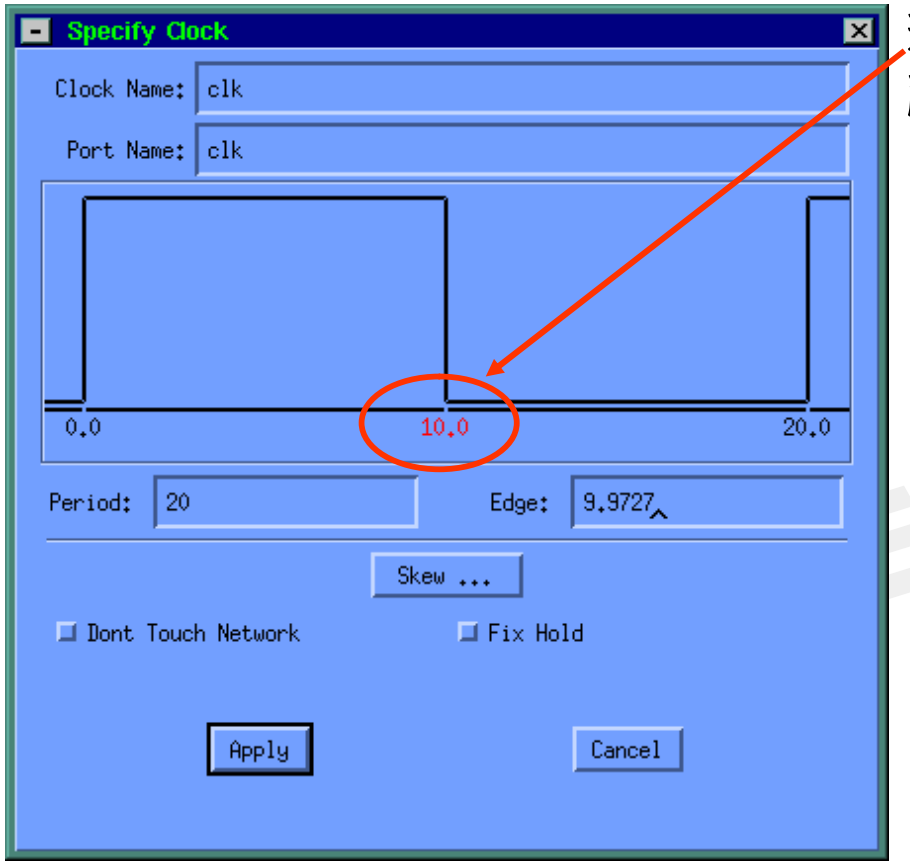
Specify Clock (cont.)





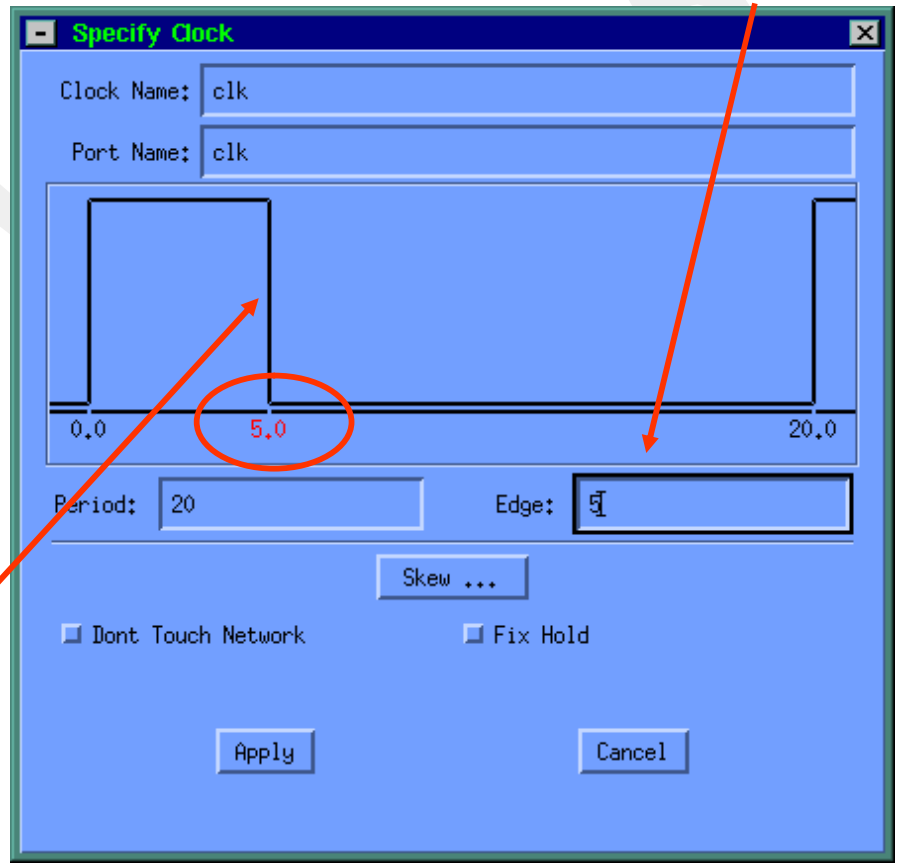
在 Period 欄中填入時間後按 Apply，
(單位：ns)即可改變 clock period

Specify Clock (cont.)



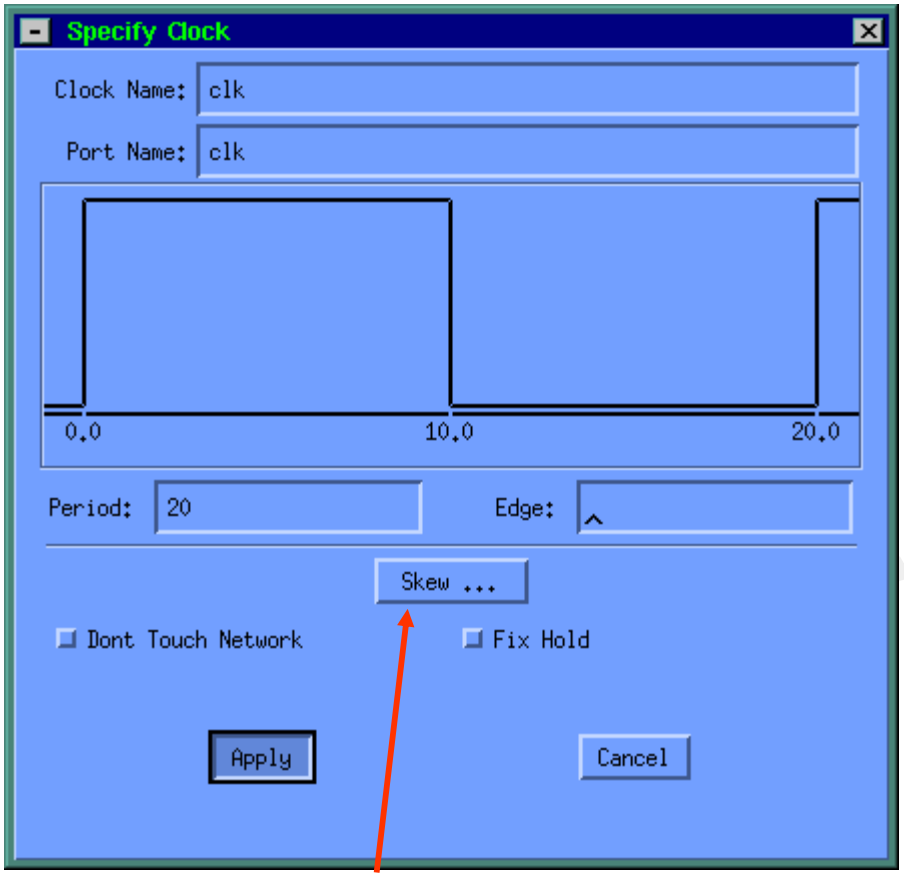
若欲調整 duty cycle , 先點選 clock edge (數字顏色變紅 , Edge 欄中會出現原設定時間

然後在 Edge 欄中填入設定時間後按 ***enter*** 鍵即可調整 duty cycle , 最後按 Apply 確定



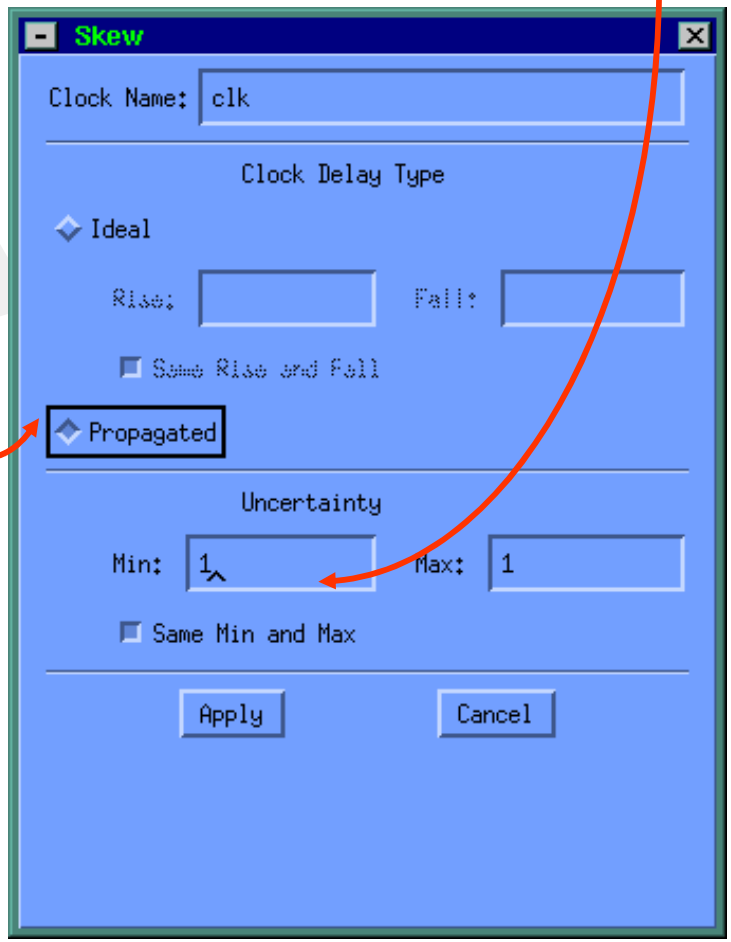
: 使用滑鼠左鍵點選 edge 按住不放後拖曳滑鼠也可調整 duty cycle , 最後按 Apply 確定

Specify Clock (cont.)

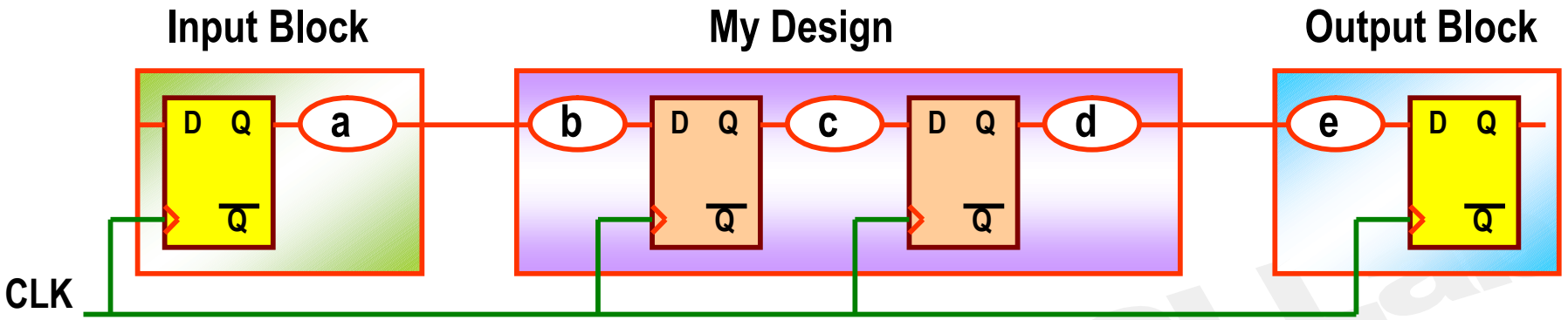


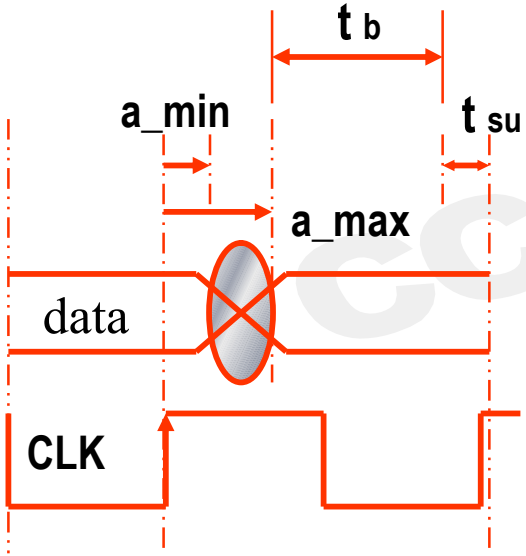
點選 Skew bottom進入設定
clock skew window

點選 Propagated bottom 並設定
skew time (ns) 後按 Apply



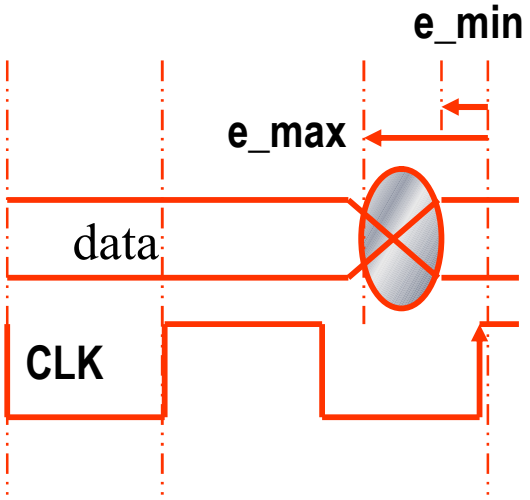
MAX. Delay & MIN. DELAY





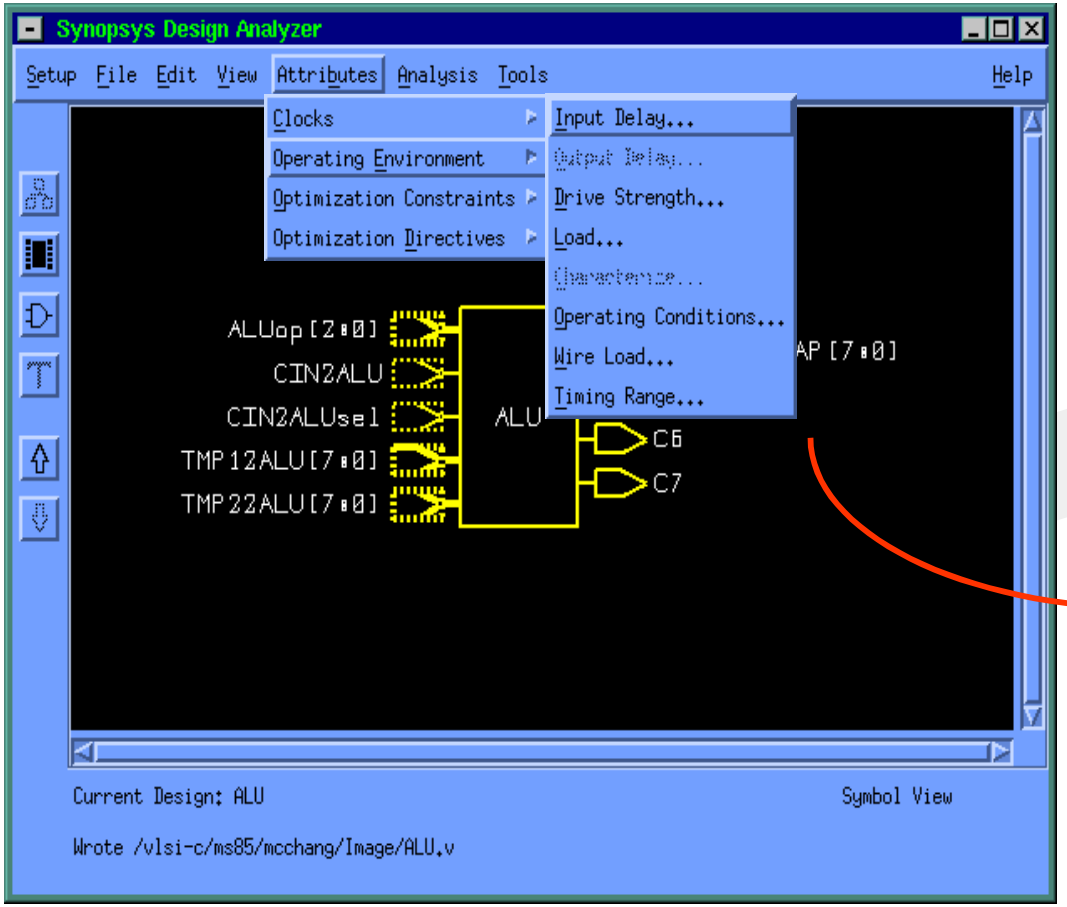
a : Input Delay

我們拿已知的時脈作爲參考，扣除週邊電路所花的時間，即是 My design所能用於合成最佳化的時間。其中，以左圖爲例， a 是我們的 input delay time可以視爲 input block 在時脈到達後的 data ready time，b 則包含了Data 在 My design 前的 delay time t_b 加上 Data 的 setup time t_{su}。

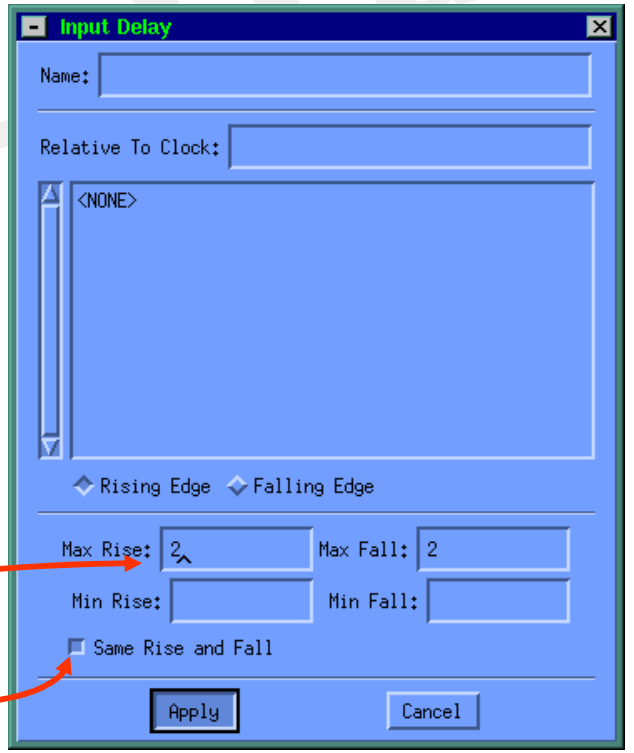


e : output Delay

Setting Input Delay



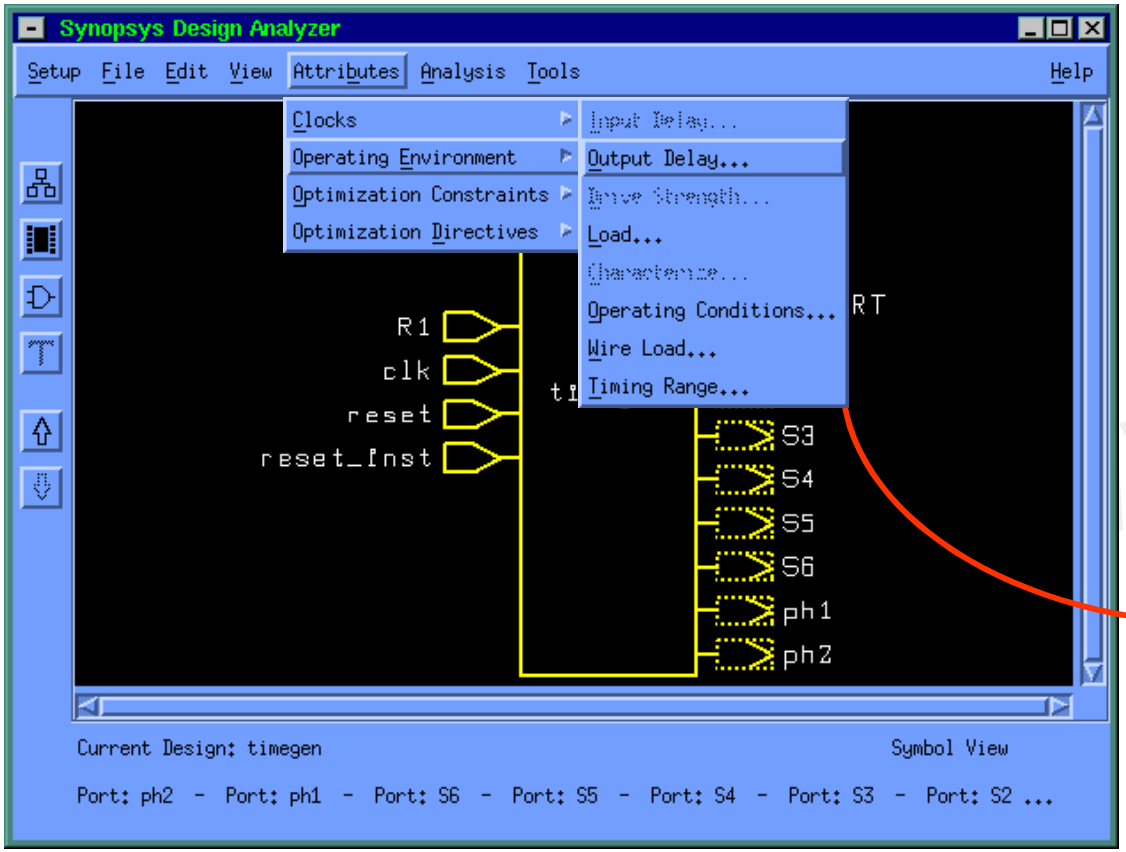
選擇物件後由
Attributes → Operating Environment → Input Delay 即出現設定延遲的視窗。（如下圖）



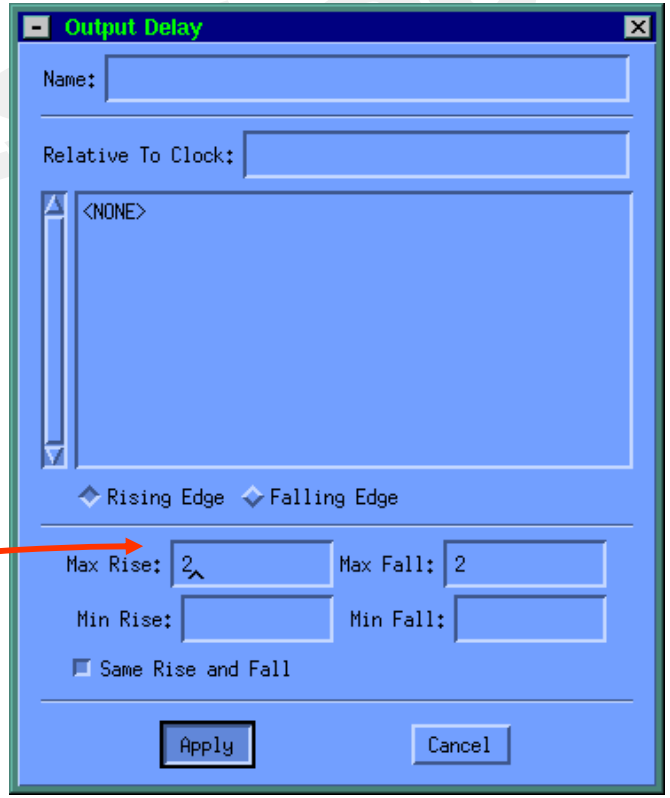
在此填入設定值 (單位：ns) 後按Apply

若要設定不同的 rise，fall delay ，點選小 icon 取消。(其他設定均如此)

Setting Output Delay

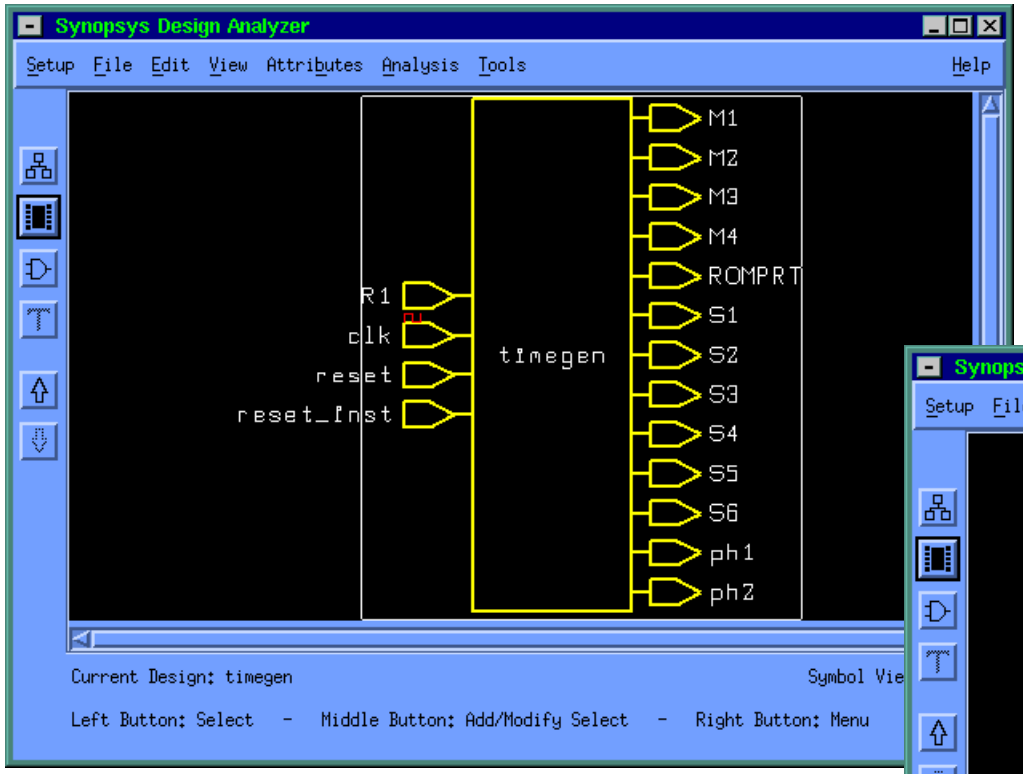


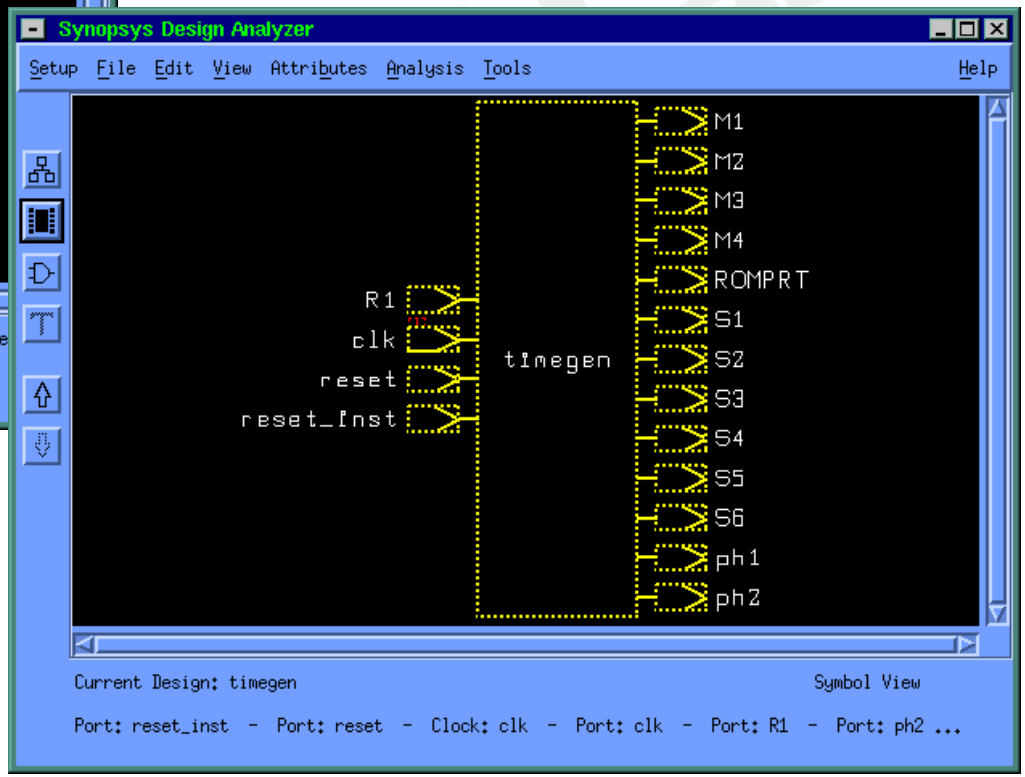
選擇物件後由
Attributes → Operating Environment → Output Delay 即出現設定延遲的視窗。（如下圖）



在此填入設定值 (單位：ns) 後按Apply

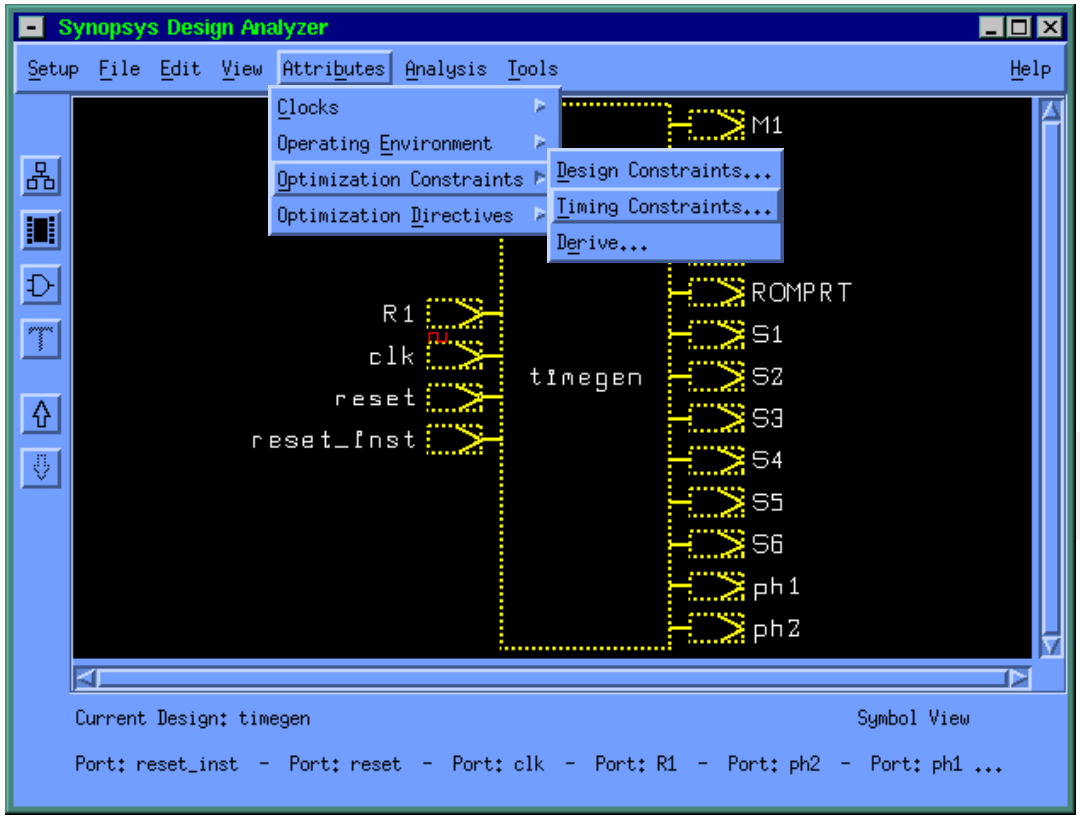
Setting Constraints -- sequential circuit



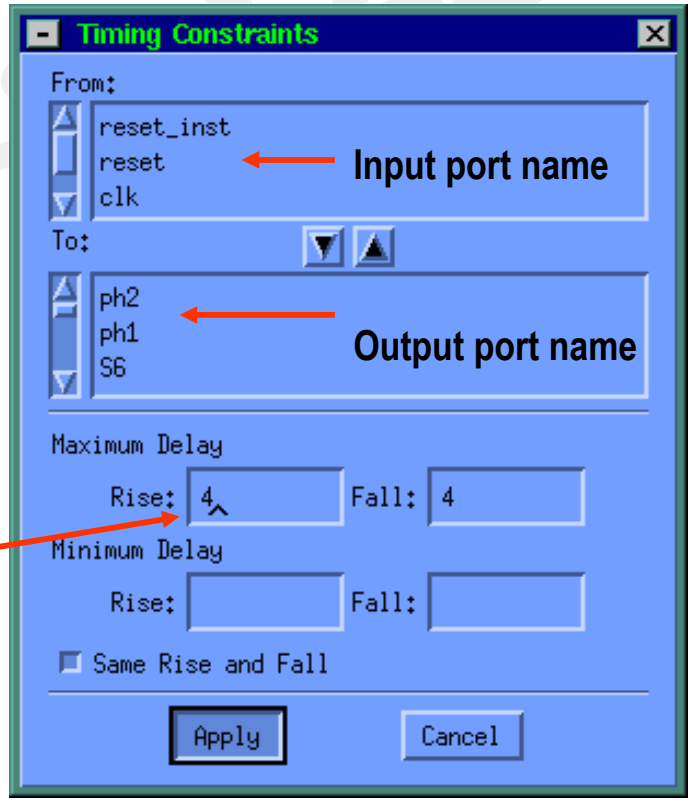


設定 constraints 時, 必須先點選欲設定的 port, 圖例是圈選所有的 ports

Setting Constraints (cont.) -- sequential circuit



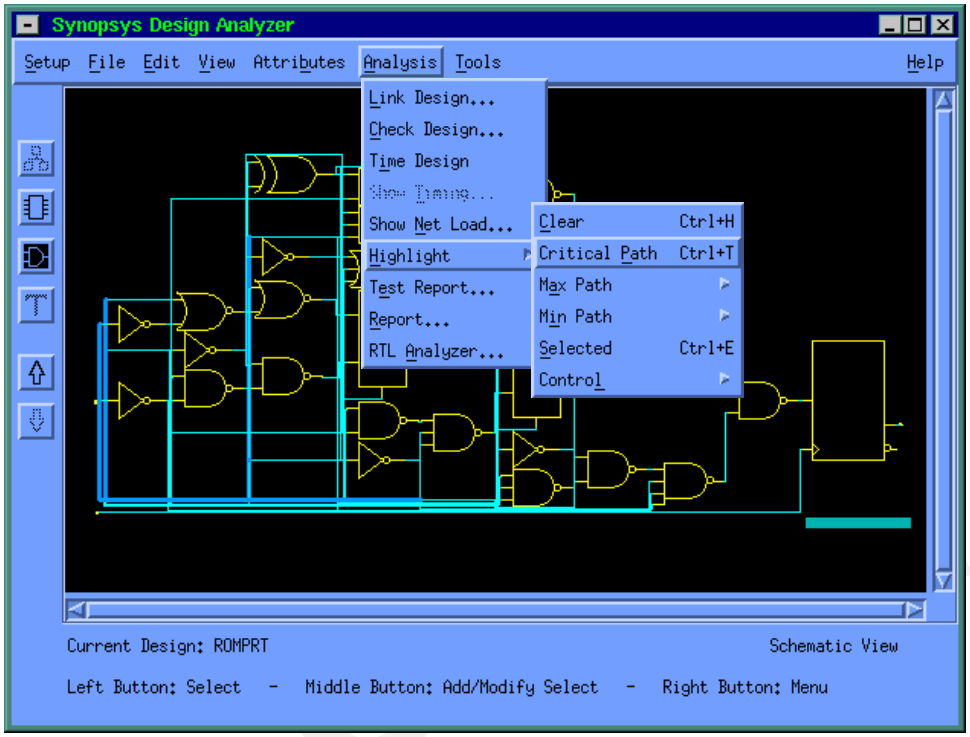
由 Attributes → Optimization Constraints → Timing Constraints 即可進入設定 timing constraints window



在 Maximum Delay 欄中填入 Input port 到 Output port 的 timing constraints

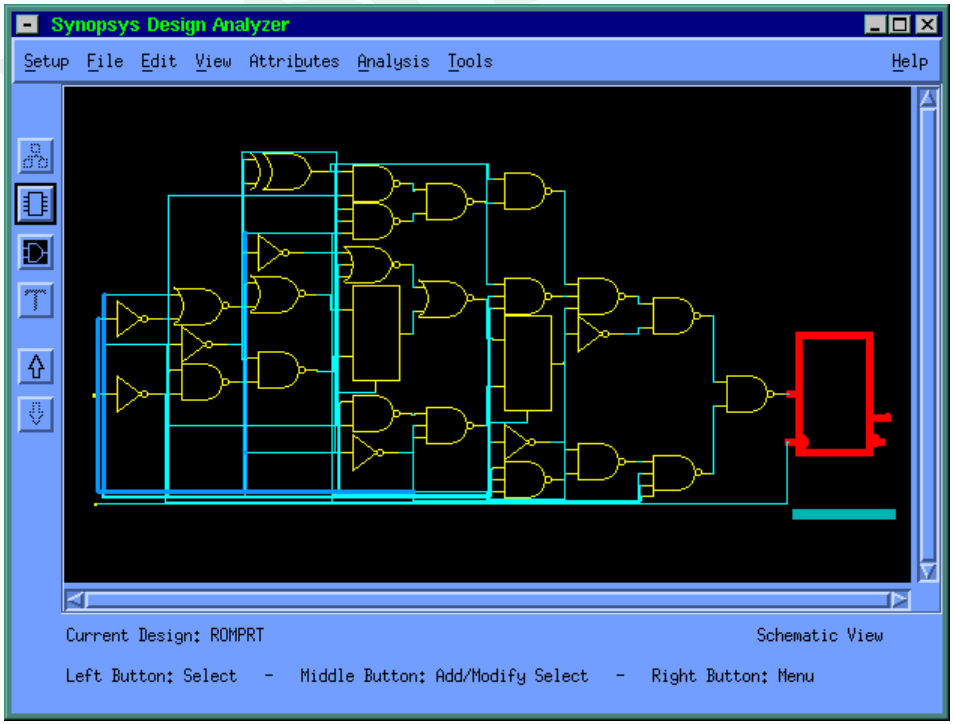
：可針對單一 input 及 output 設定

Setting Constraints (cont.) -- sequential circuit

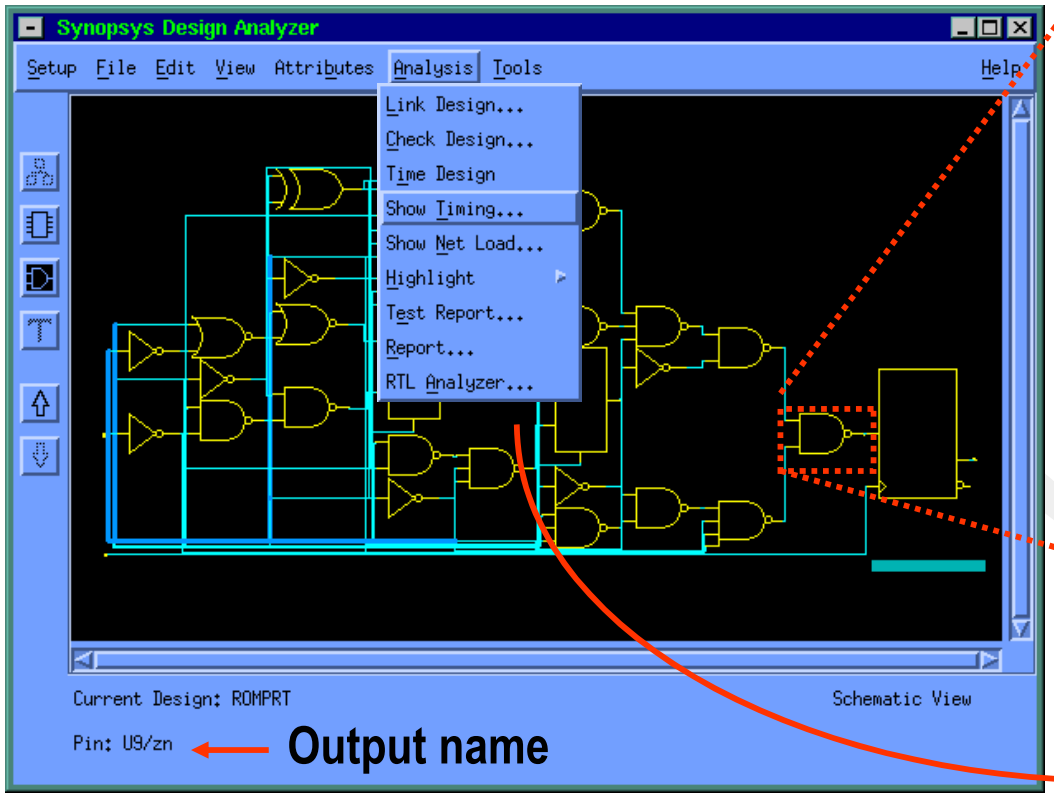


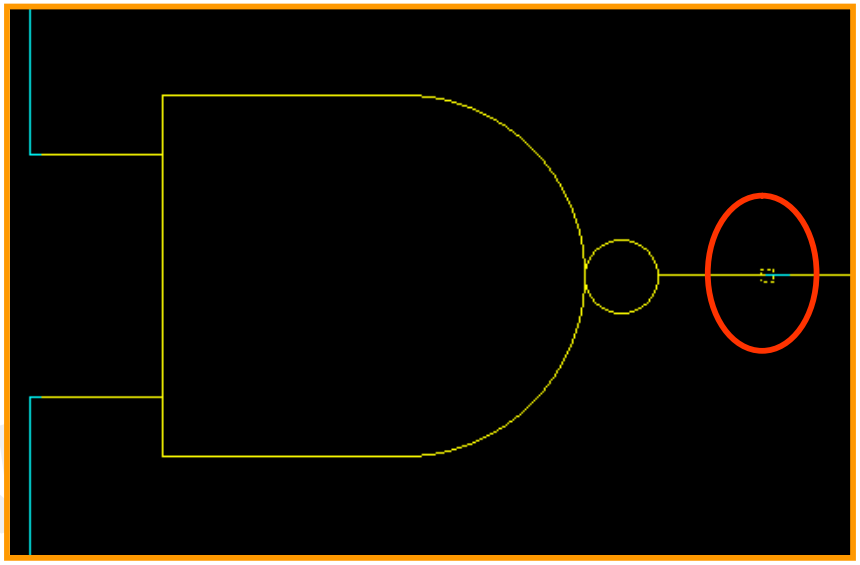
所有環境變數設定完後, 由 **Analysis → Highlight → Critical Path ,** Synopsys 會將電路的 Critical Path 顯示 (如圖)

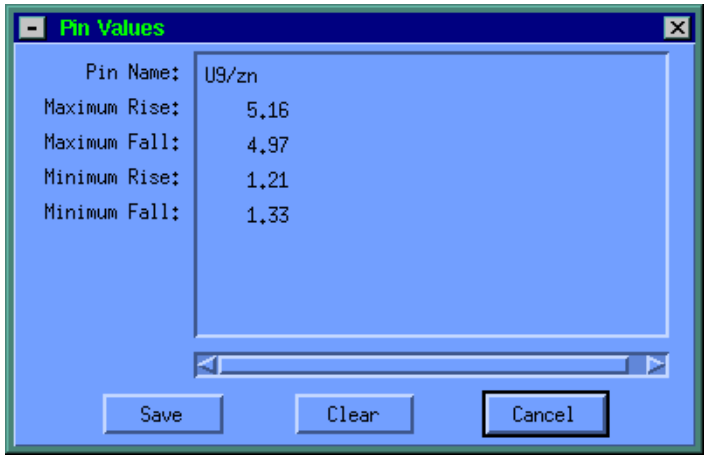
: 由此可看出 Synopsys 在 combinational circuit 與 sequential circuit 上對 Critical Path定義的差異 . 因此不能使用這樣的方法分析 Critical Path.



Setting Constraints (cont.) -- sequential circuit

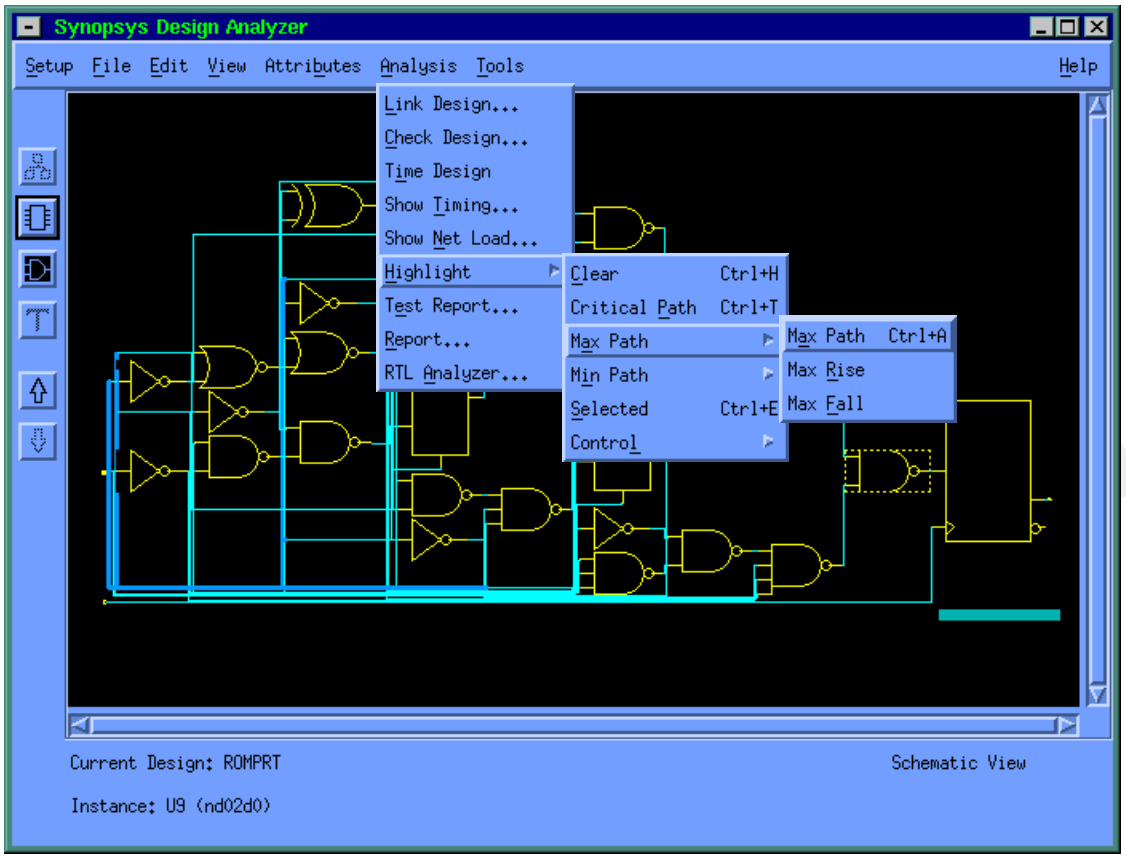






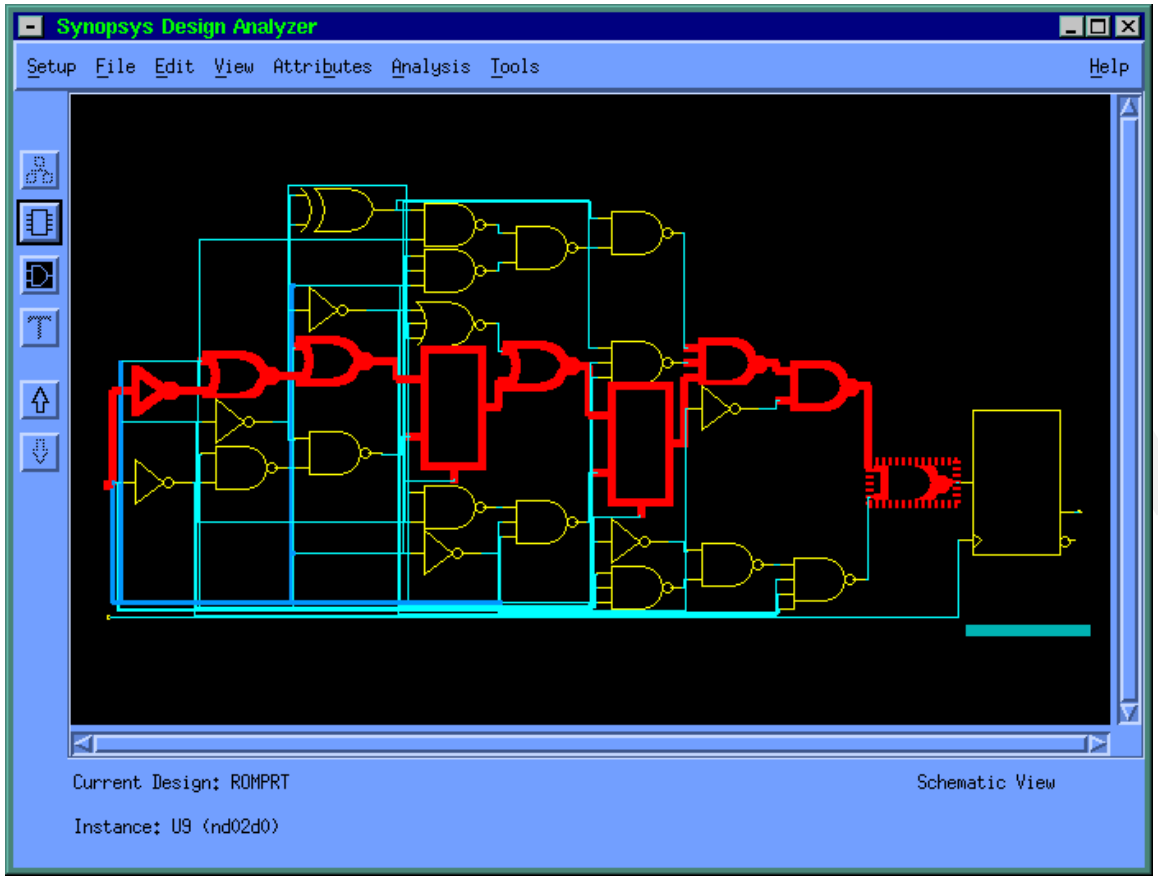
點選欲觀察的 gate output point (如右上圖所示), 由 Analysis → Show Timing即可觀察該點的 timing . 使用 Report 亦可.

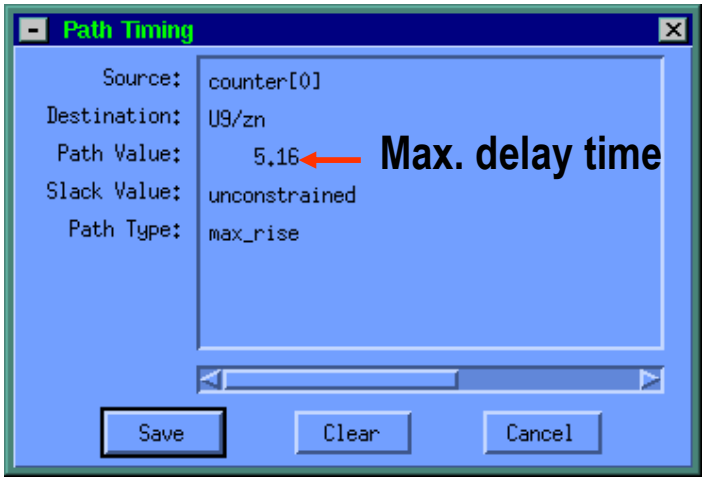
Setting Constraints (cont.) -- sequential circuit



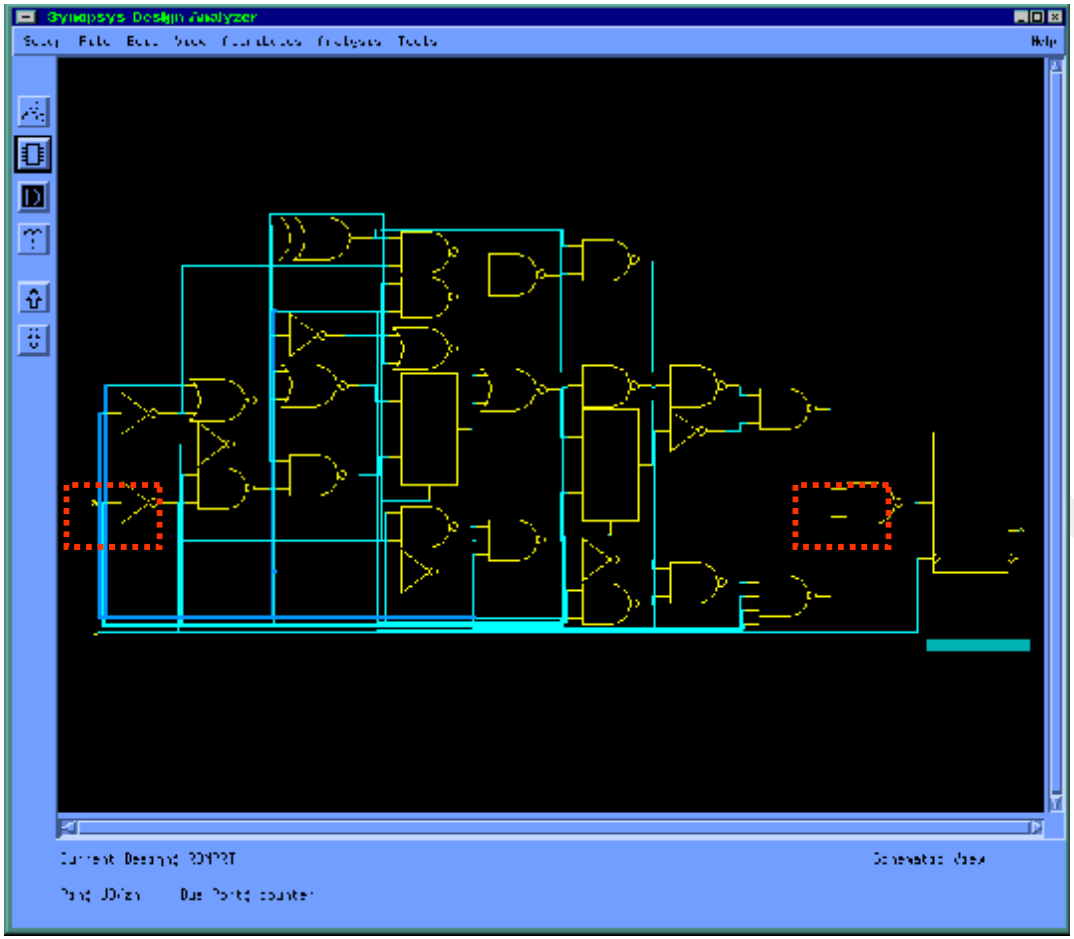
如同上頁(只要點選該 gate) 由 **Analysis → Highlight → Max Path → Max Path** 即可觀察該點的 timing path .(如同下頁所示)

Setting Constraints (cont.) -- sequential circuit

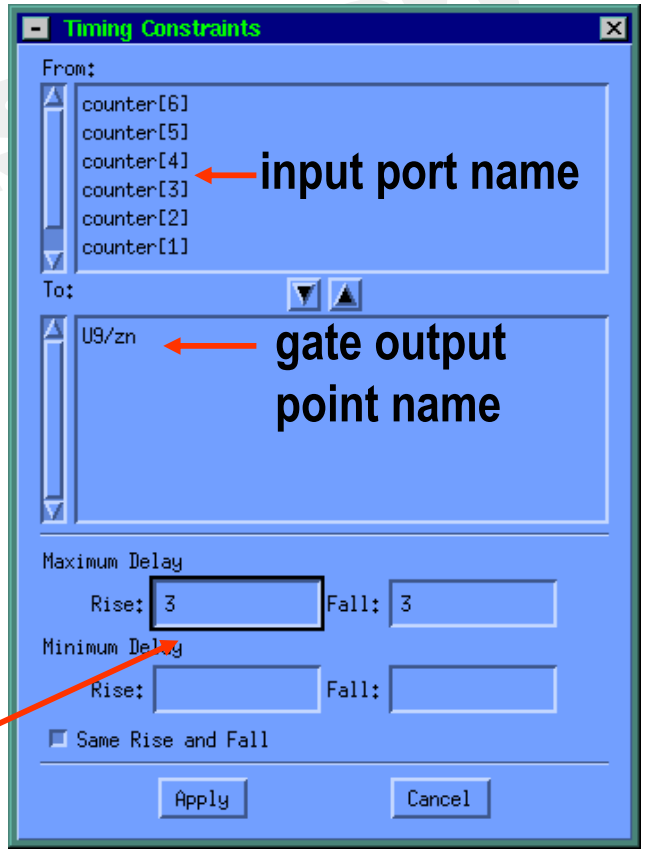




Setting Constraints (cont.) -- sequential circuit

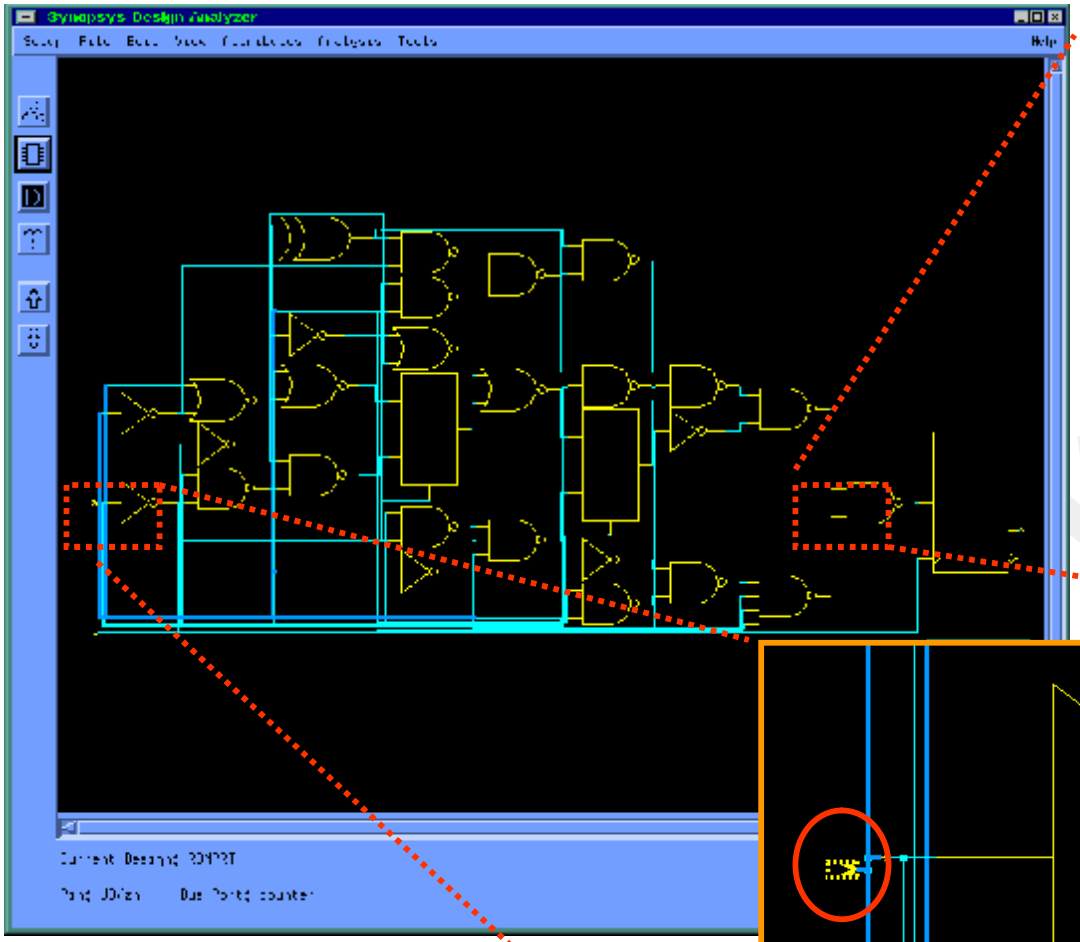


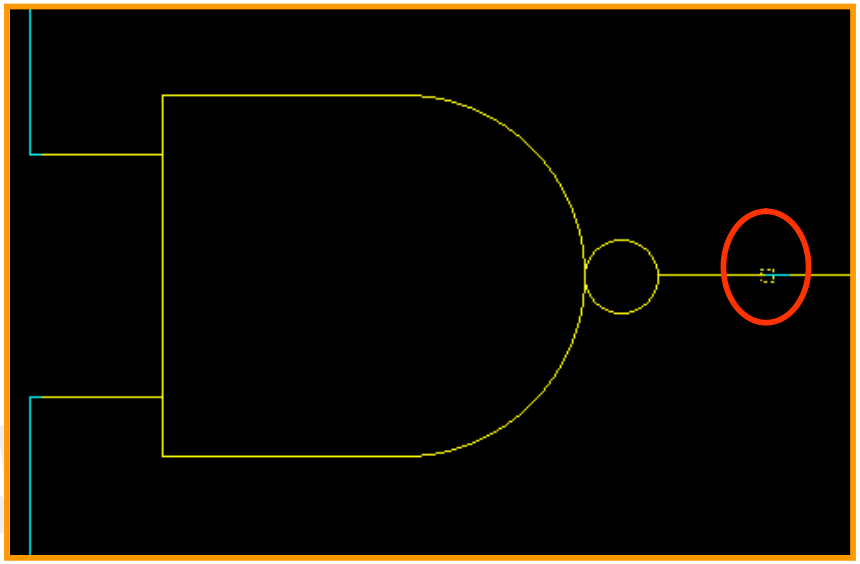
點選欲設定的 input port 及 gate output point (如左圖所示). 放大圖如下頁所示.

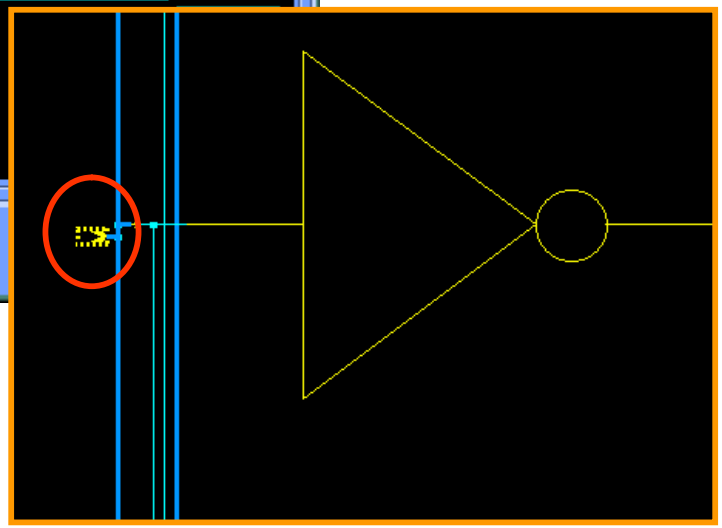


在此填入設定值 (單位：ns) 後按Apply

Setting Constraints (cont.) -- sequential circuit

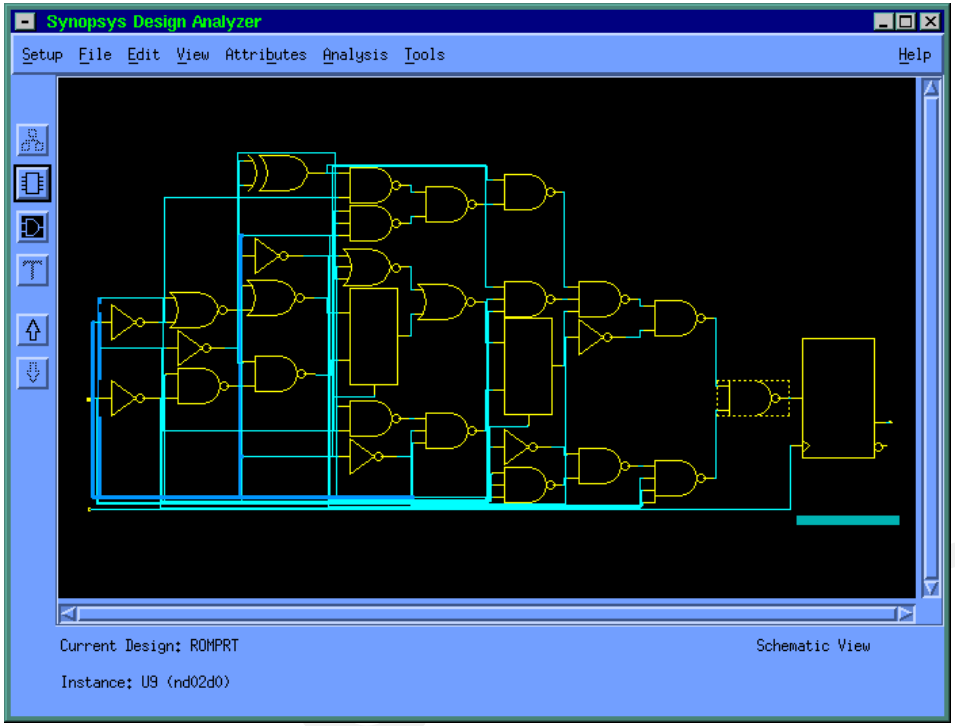




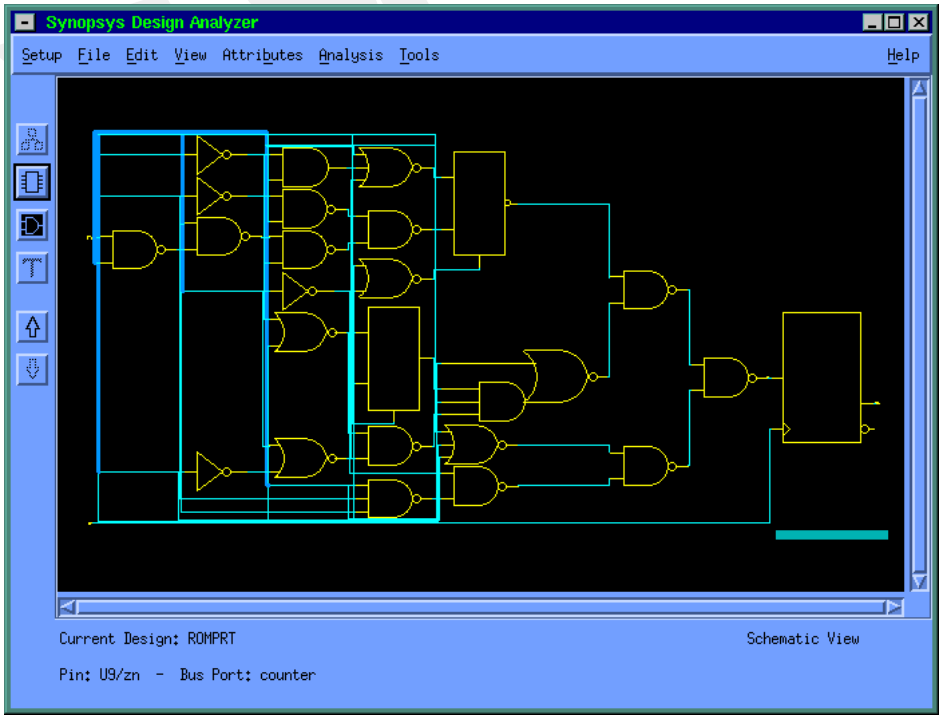


使用滑鼠左鍵點選 input port 再使用中鍵點選任一 gate 的 output point(如圖)

Setting Constraints (cont.) -- sequential circuit

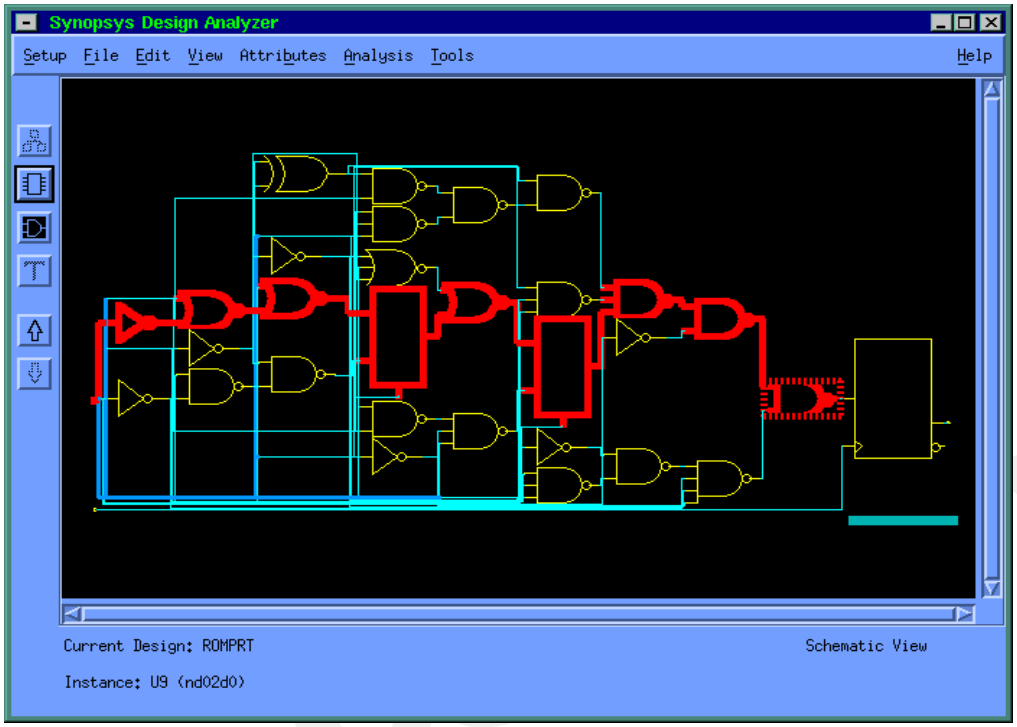


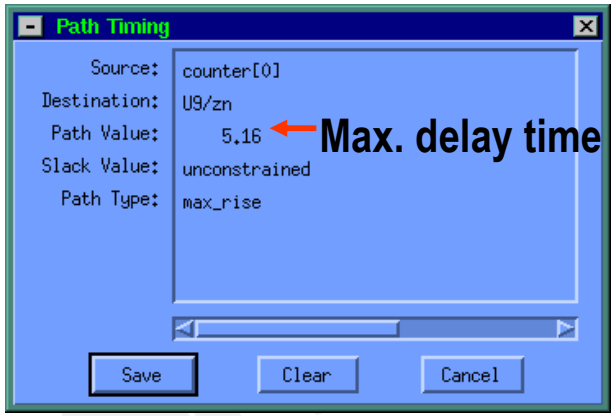
重新合成前

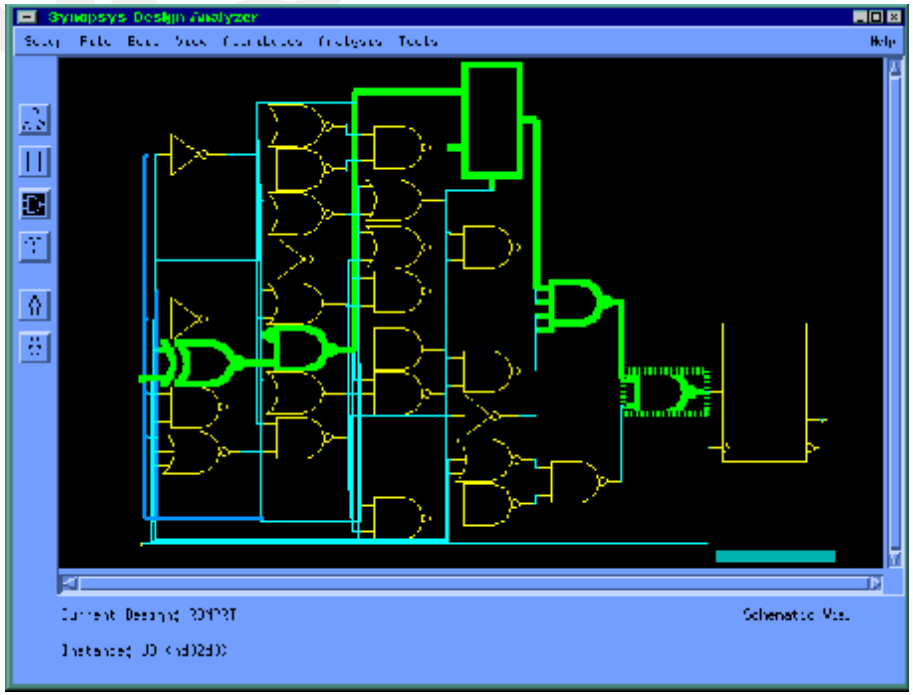


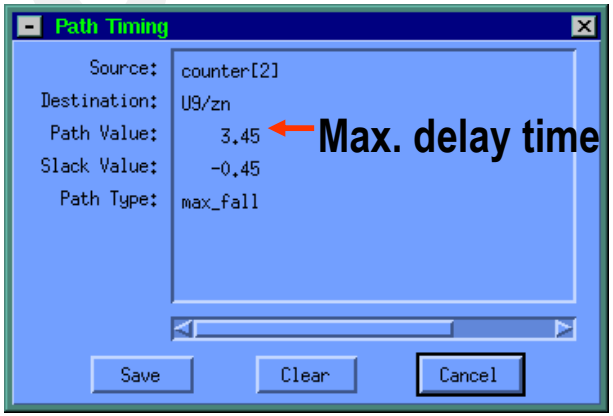
重新合成後

Setting Constraints (cont.) -- sequential circuit

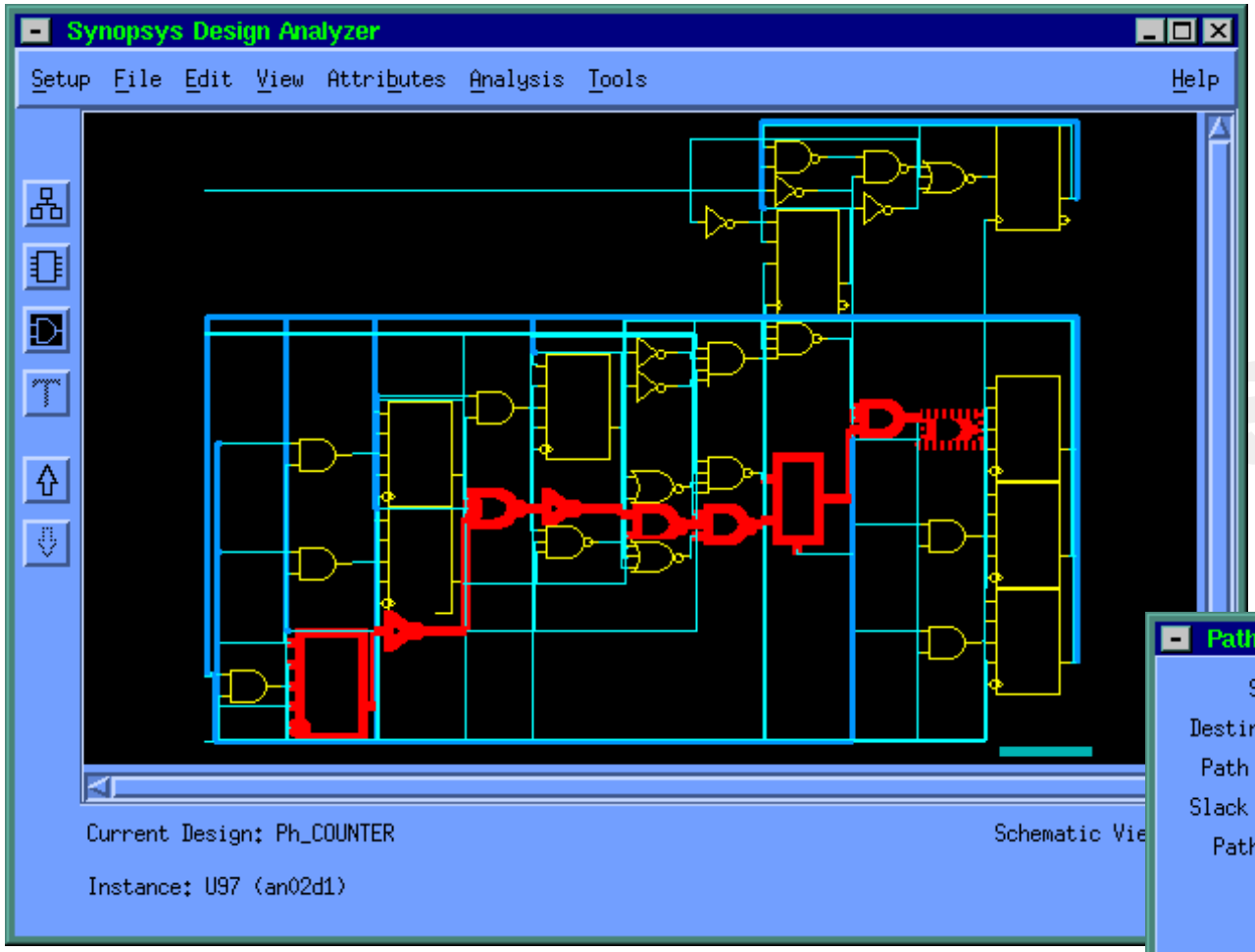








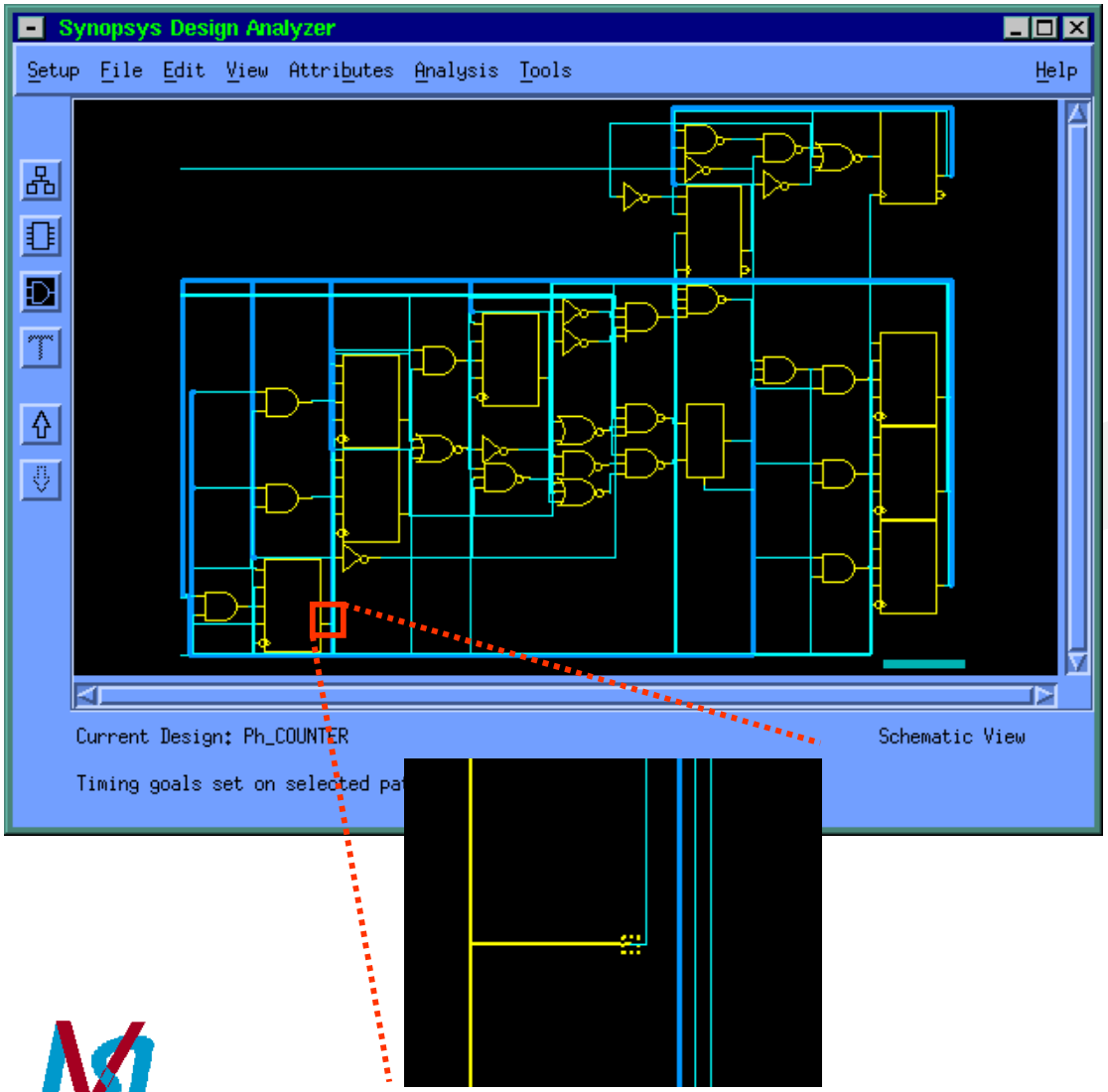
Setting Constraints (cont.) -- sequential circuit



: 兩個 latch or flip flop 間如何設 constraints ?

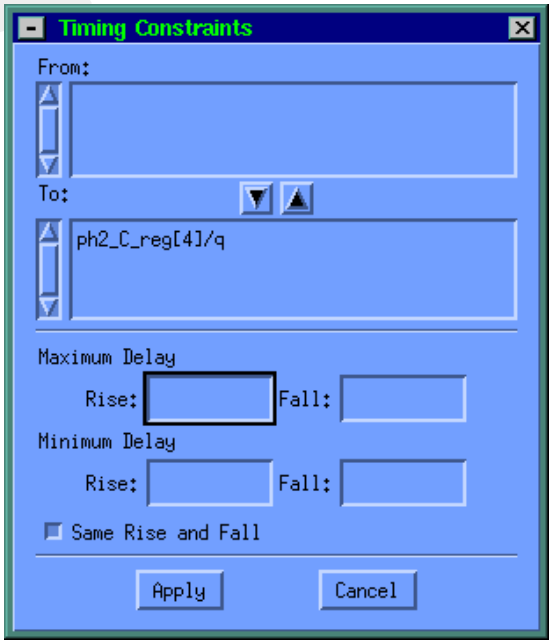
如圖所示, 如果 delay time 不符 spec.或是覺得結果不佳.可依照 52~54 頁所示設定.

Setting Constraints (cont.) -- sequential circuit

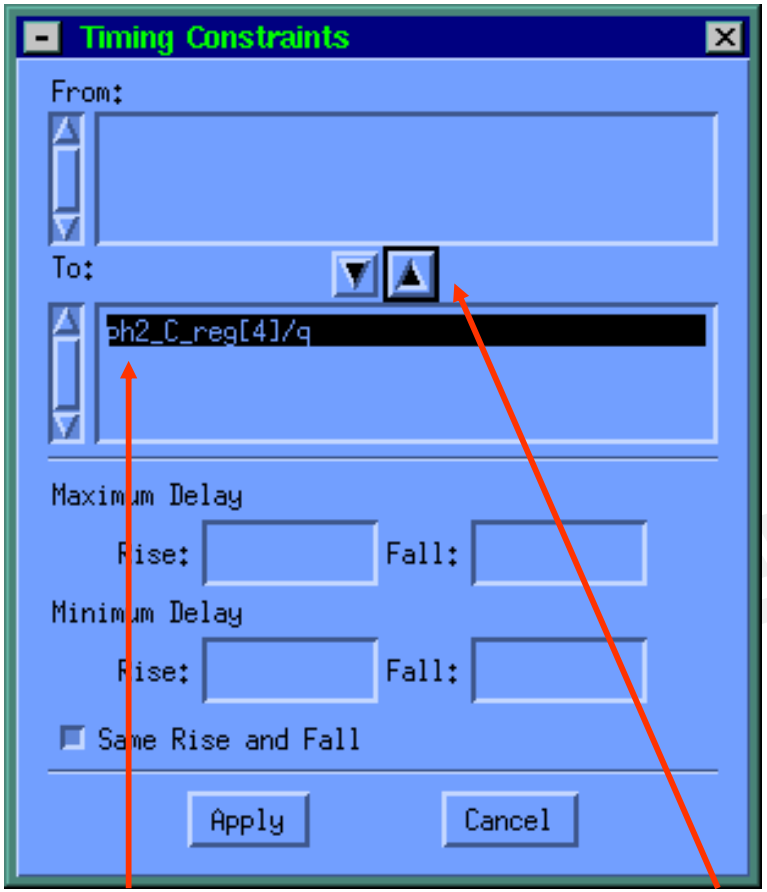


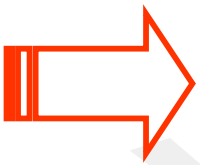
：

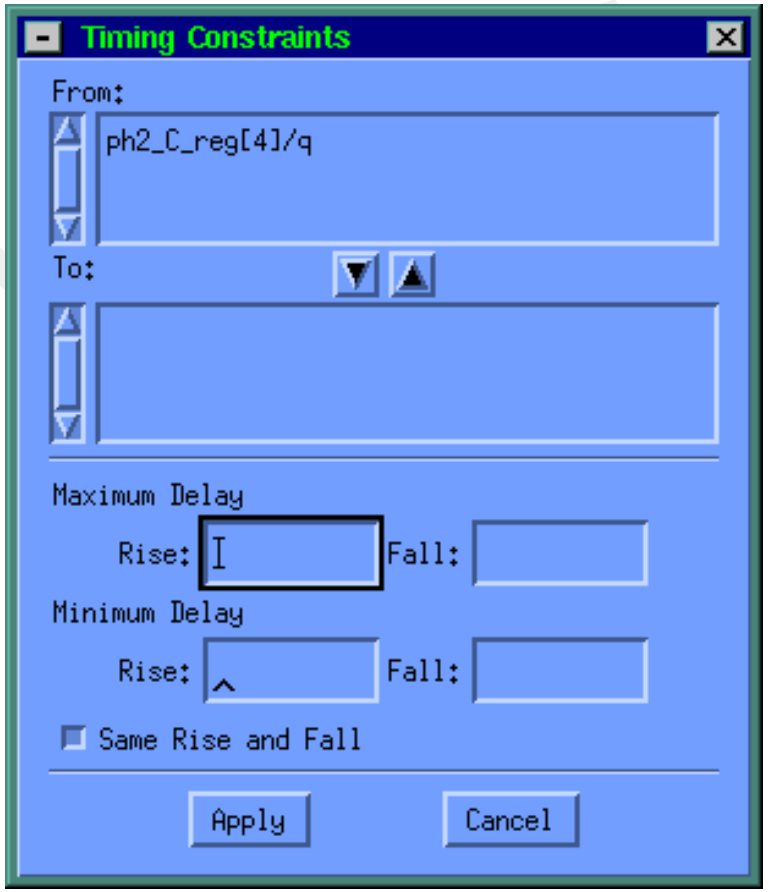
使用滑鼠中鍵選取或取消



Setting Constraints (cont.) -- sequential circuit

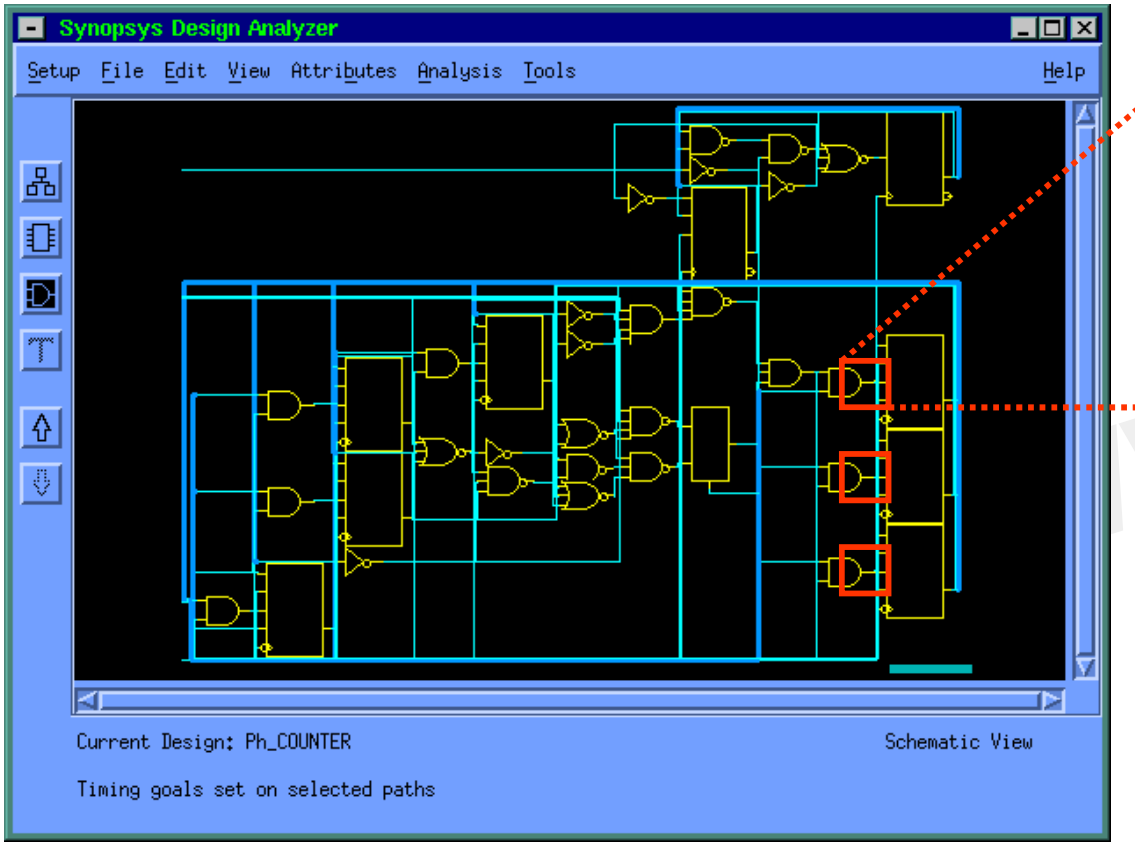


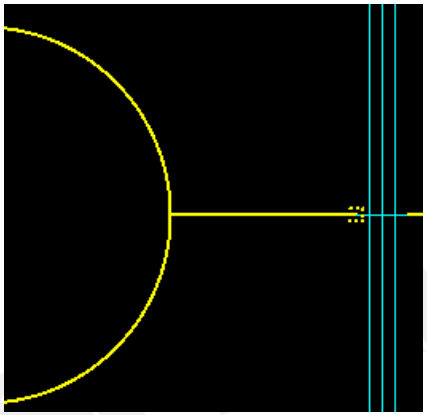


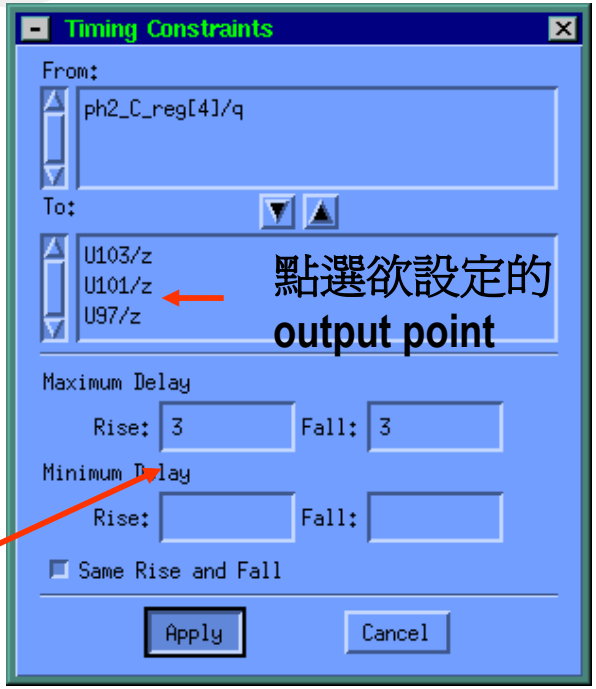


點選此處使之反白後,按方向 **icon** .
即可設定 **from point**

Setting Constraints (cont.) -- sequential circuit

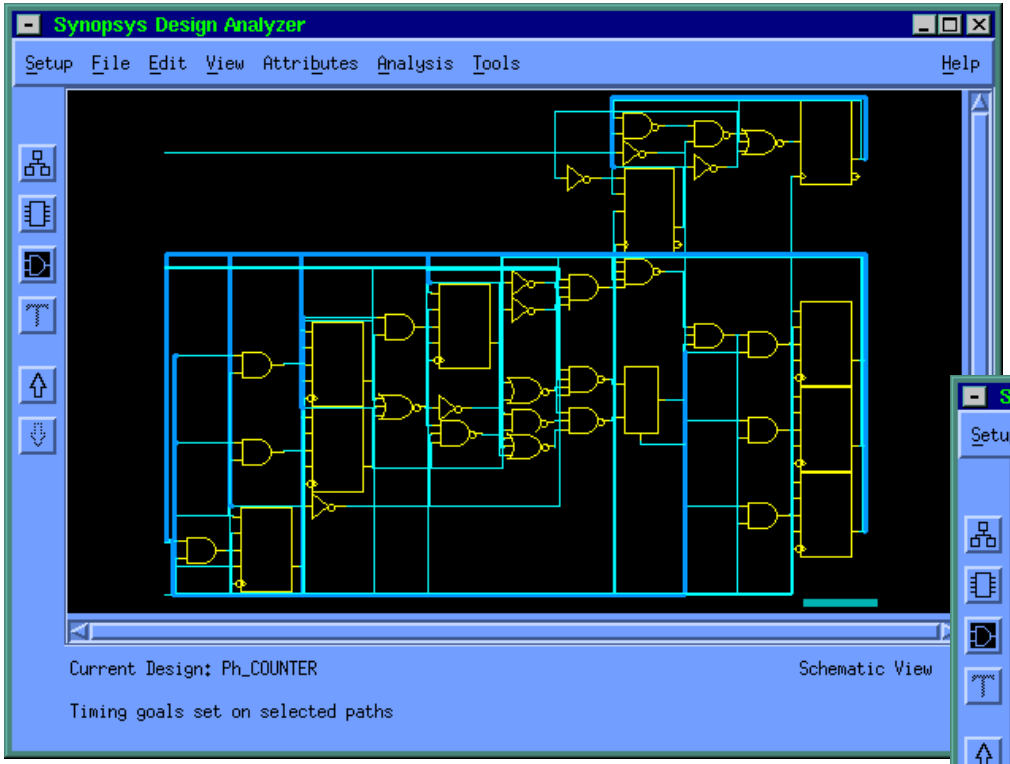




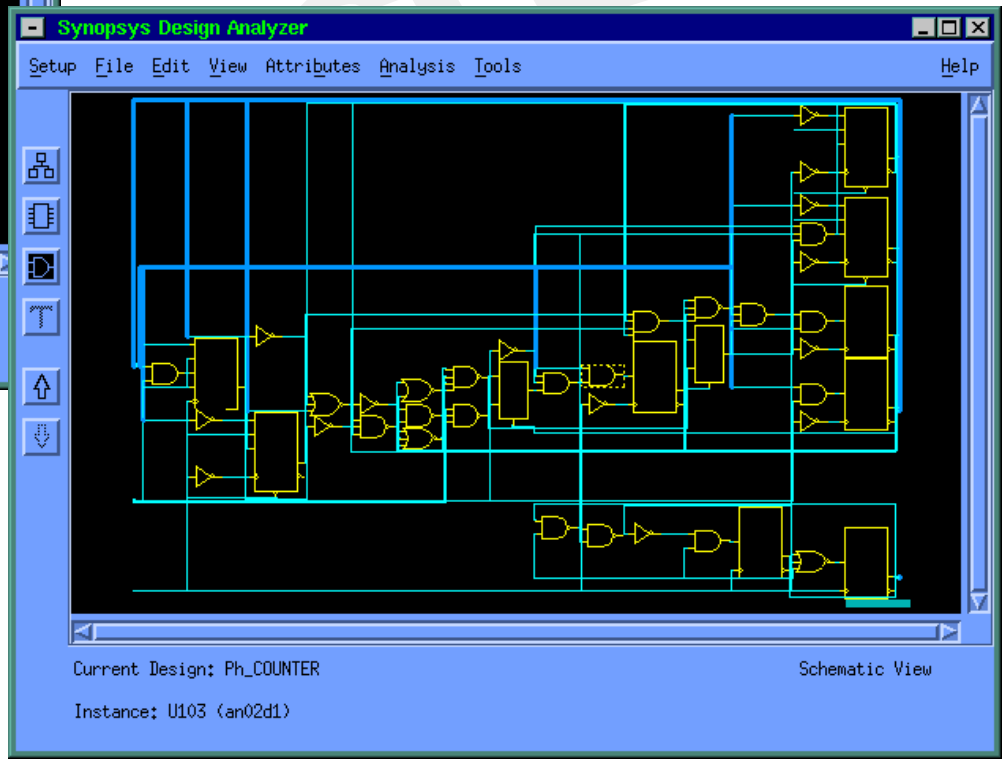


在此填入設定值 (單位：ns) 後按Apply

Setting Constraints (cont.) -- sequential circuit

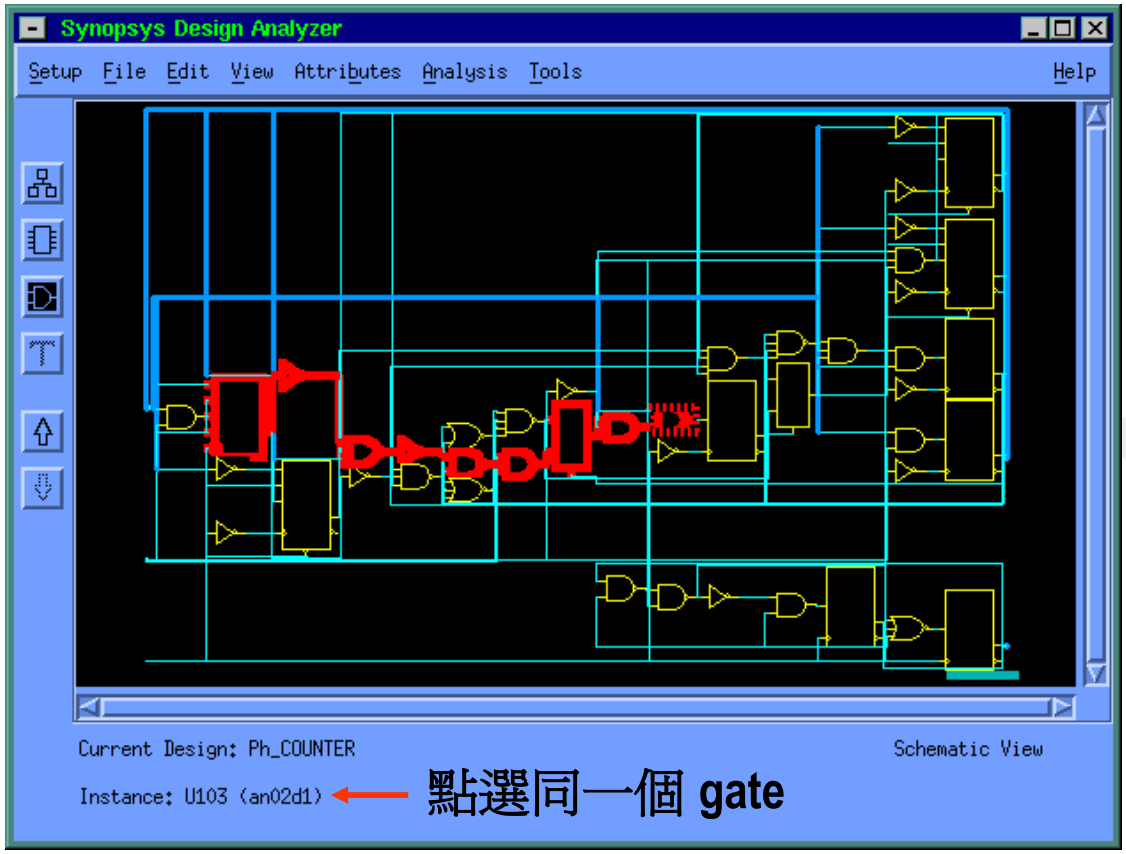


重新合成前

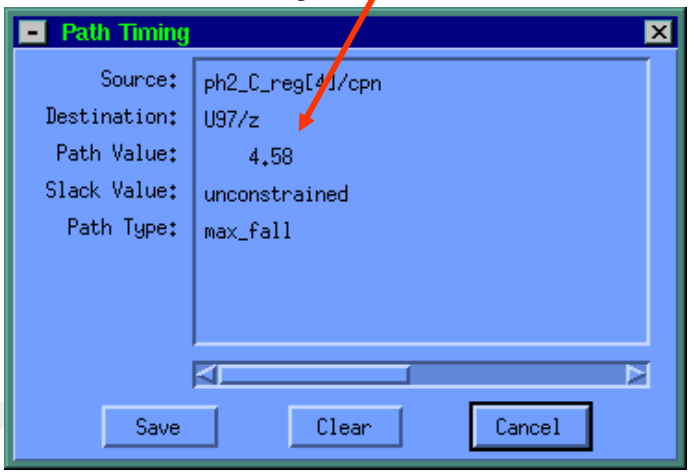


重新合成後

Setting Constraints (cont.) -- sequential circuit



合成前 Delay time



合成後 Delay time

符合 timing SPEC.

如同 43 ~ 44 頁所示方法顯示 Max. Path

Differences of coding

- structure
- key word --- //synopsys

CCU EE VLSI Lab

細節部分請參考 SYNOPSYS on line document
--- HDL Compiler for Verilog Reference

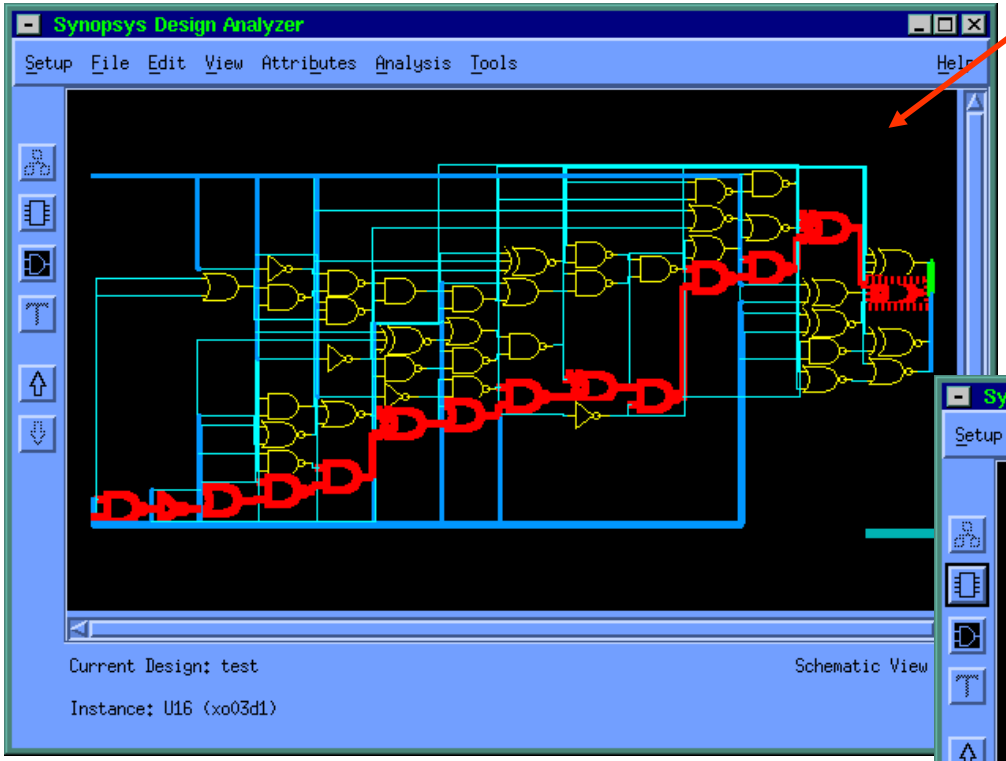
Structure

Coding 上的差異可以影響合成的結果, synopsys 可以讓 user自行對 Coding 做不同的組合, 以達成不同的目的. 詳細的說明請參考 HDL Compiler for Verilog Reference --- Chapter 8 (SYNOPSYS on line document). 底下是一個例子.

```
module coding_1(a, b, c, d, out);
input [3:0] a, b, c, d;
reg [3:0] outtmp;
output [3:0] out;
always @(a or b or c or d) begin
outtmp = a + b + c + d;
end
assign out = outtmp;
endmodule
```

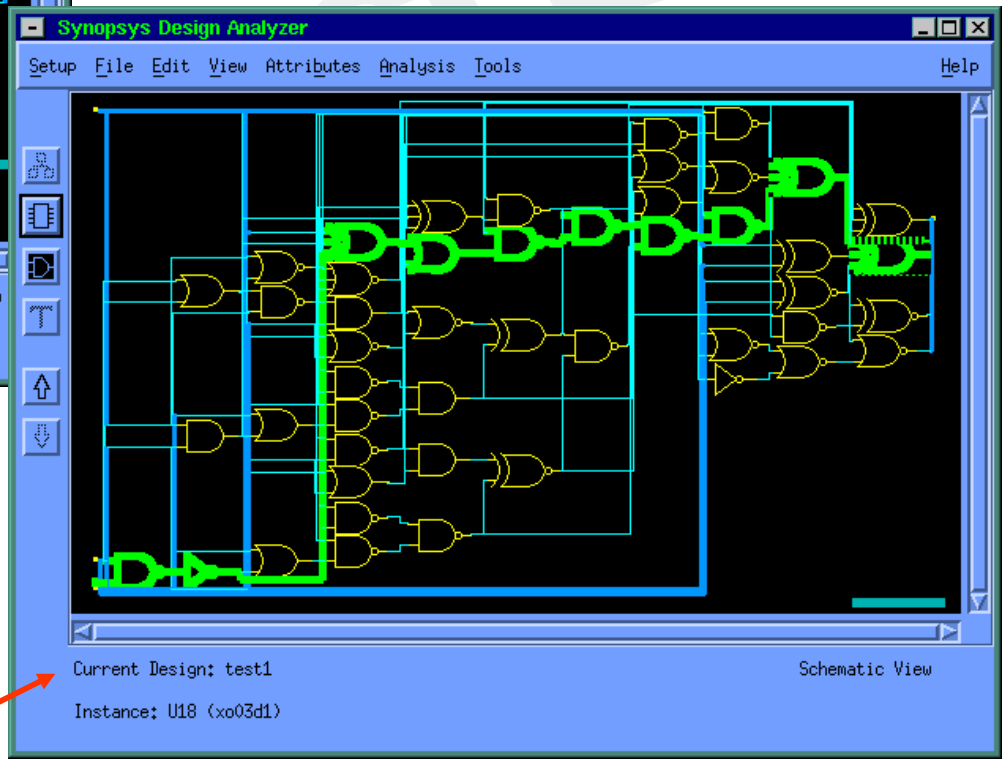
difference

```
module coding_2(a, b, c, d, out);
input [3:0] a, b, c, d;
reg [3:0] outtmp;
output [3:0] out;
always @(a or b or c or d) begin
outtmp = ((a + b) + (c + d));
end
assign out = outtmp;
endmodule
```



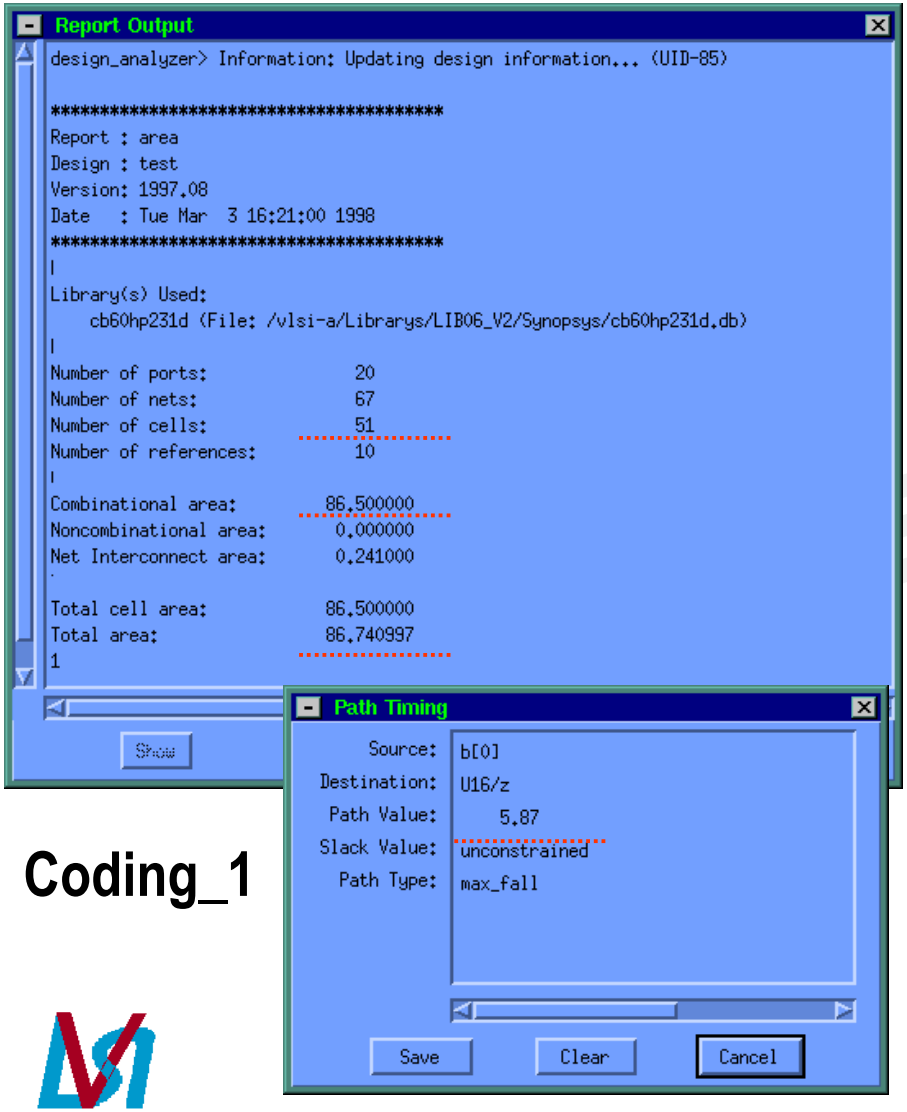
Coding_1

out = a + b + c + d ;

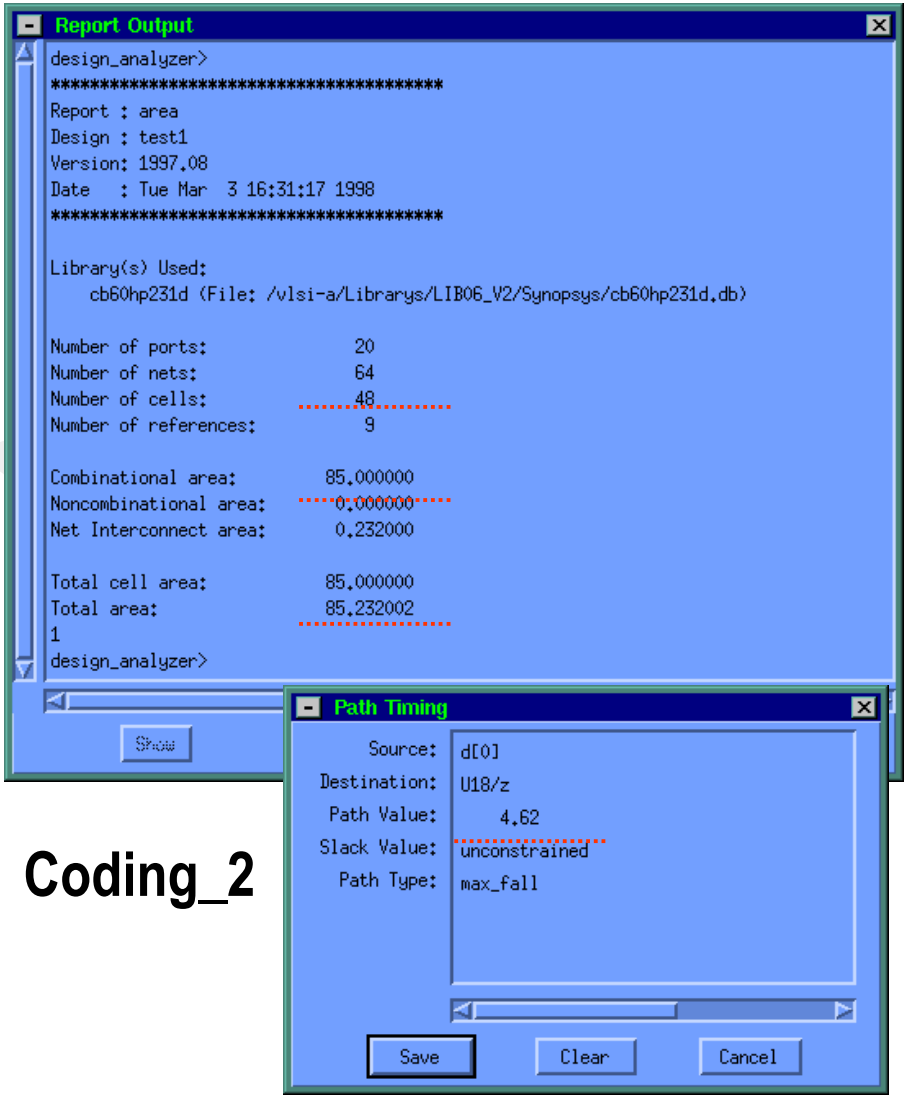


Coding_2

out = ((a + b) + (c + d));



Coding_1



Coding_2

key word --- //synopsys

```
module SP1 ( reset,SP2IB1,IB12SP,SPen,
                    SP2IB1en ) ;
input SPen, SP2IB1en ,reset;
input [2:0] IB12SP ;
output [2:0] SP2IB1;
reg [2:0] sp;


always @(SPen or IB12SP or reset)
begin
      if (reset)
            sp = 07;
     else if (SPen)
            sp = IB12SP;
end

assign SP2IB1 = (SP2IB1en ? sp : 'bz);
endmodule
```

Coding_1

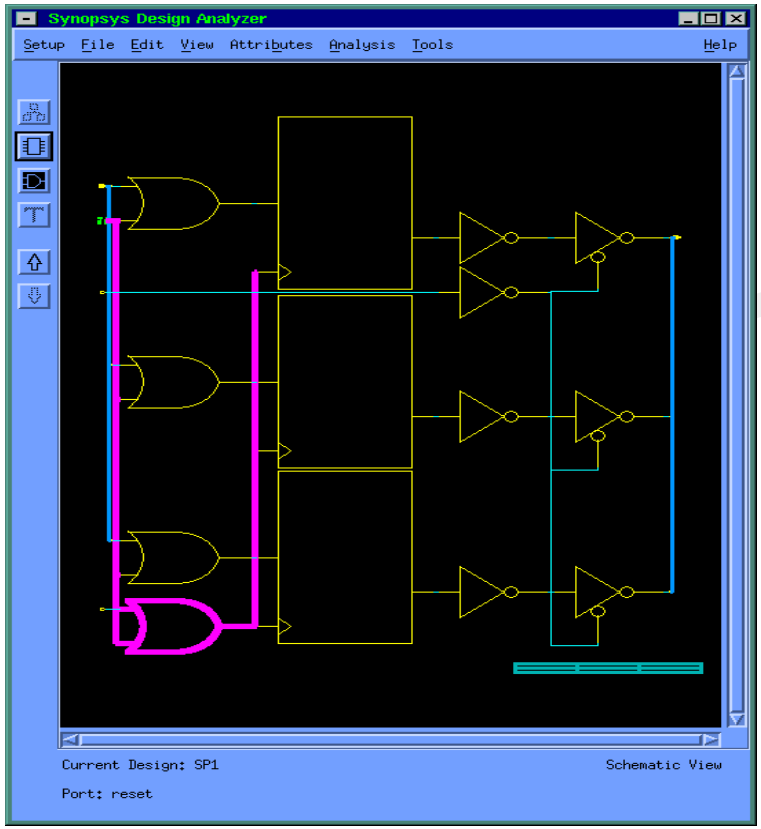
difference

```
module SP2 ( reset,SP2IB1,IB12SP,SPen,
                    SP2IB1en ) ;
input SPen, SP2IB1en ,reset;
input [2:0] IB12SP ;
output [2:0] SP2IB1;
reg [2:0] sp;

//synopsys async_set_reset "reset"
always @(SPen or IB12SP or reset)
begin
      if (reset)
            sp = 07;
     else if (SPen)
            sp = IB12SP;
end

assign SP2IB1 = (SP2IB1en ? sp : 'bz);
endmodule
```

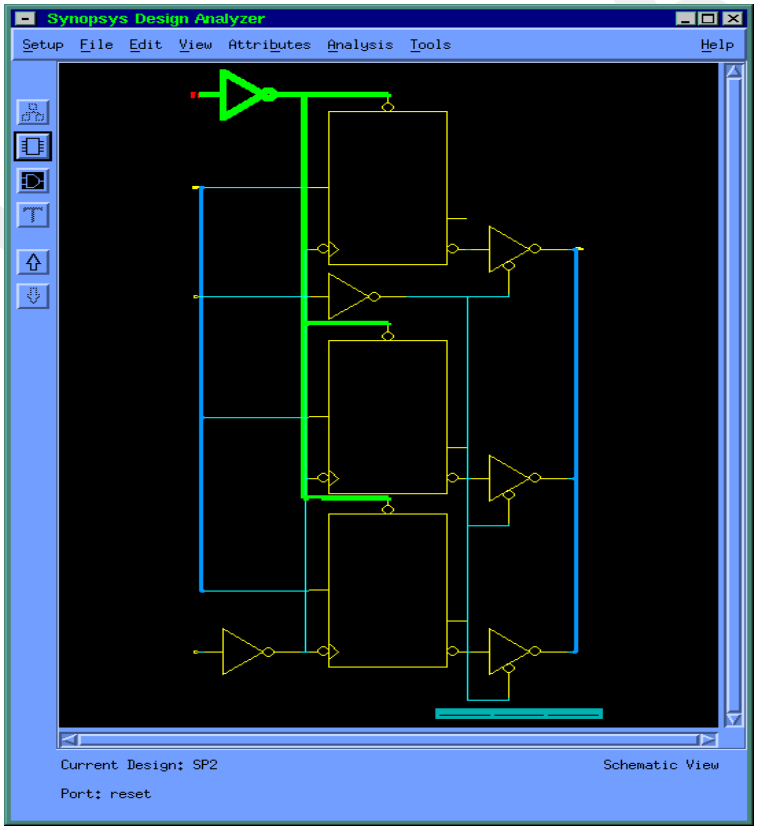
Coding_2

CCU EE VLSI Lab

☞ : 若不定義 //synopsys async_set_reset 時, SYNOPSYS並不會選取含 “set” or “reset” 的 latch. 因此 SYNOPSYS 會將 reset 及 enable signal 做 gated control 輸入 latch 的 enable input, 這樣的方式可能使得電路不正常運作.



Coding_1



Coding_2

key word --- //synopsys

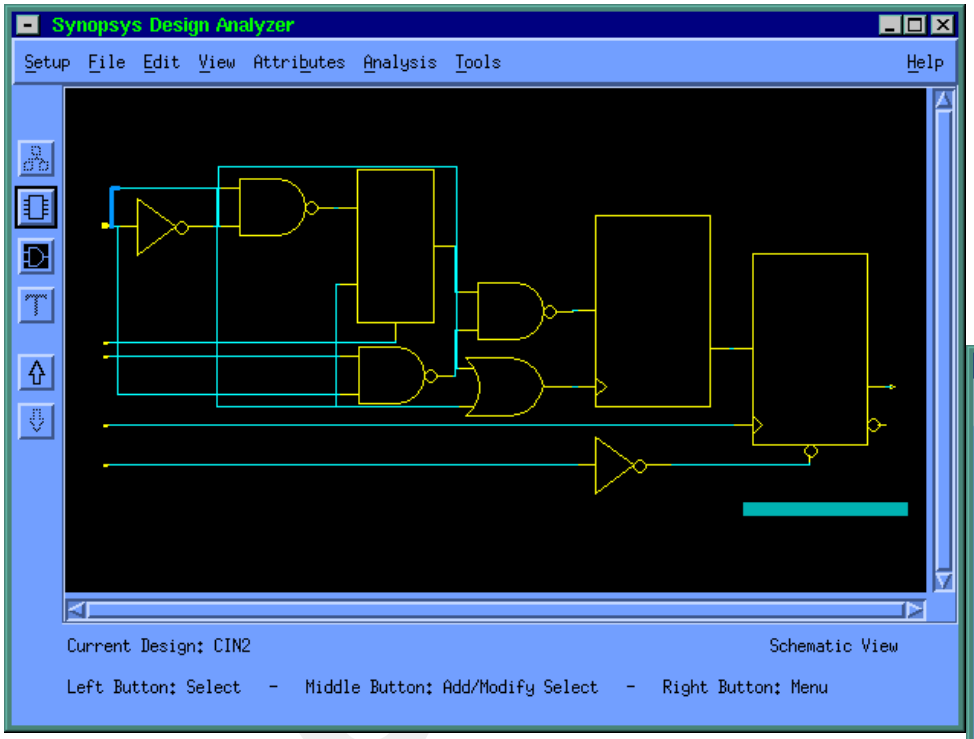
```
module CIN1(C2CIN ,IB172CIN ,CINout ,CINsel ,CINen
,CINclr);
input C2CIN, IB172CIN, CINen, CINclr;
input [1:0] CINsel;
output CINout;
reg muxans,cin;

always @(CINsel or C2CIN or IB172CIN) begin
    case(CINsel) //synopsys full_case parallel_case
            2'b00 : muxans = C2CIN;
            2'b01 : muxans = ~C2CIN;
            2'b11 : muxans = IB172CIN;
            //default: muxans = 1'bx;
    endcase
end

//synopsys async_set_reset "CINclr"
always @(muxans or CINen or CINclr) begin
    if (CINclr)
            cin = 0;
    else if (CINen)
      cin = muxans;
end
assign CINout = cin;
endmodule
```

若不用 full_case 的方式, 可將 case 補足.

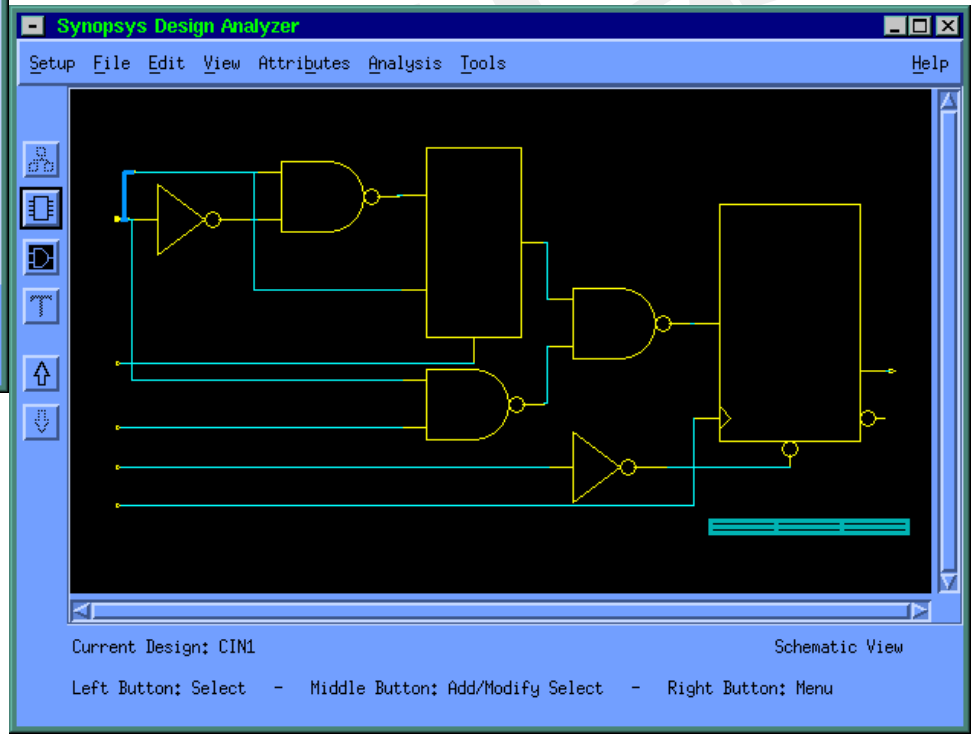
Coding_1

```
module CIN2(C2CIN ,IB172CIN ,CINout ,CINsel
,CINen ,CINclr);
input C2CIN, IB172CIN, CINen, CINclr;
input [1:0] CINsel;
output CINout;
reg muxans,cin;

always @(CINsel or C2CIN or IB172CIN) begin
    case(CINsel)
            2'b00 : muxans = C2CIN;
            2'b01 : muxans = ~C2CIN;
            2'b11 : muxans = IB172CIN;
            //default: muxans = 1'bx;
    endcase
end

//synopsys async_set_reset "CINclr"
always @(muxans or CINen or CINclr) begin
    if (CINclr)
            cin = 0;
    else if (CINen)
      cin = muxans;
end
assign CINout = cin;
endmodule
```

Coding_2



Coding_1



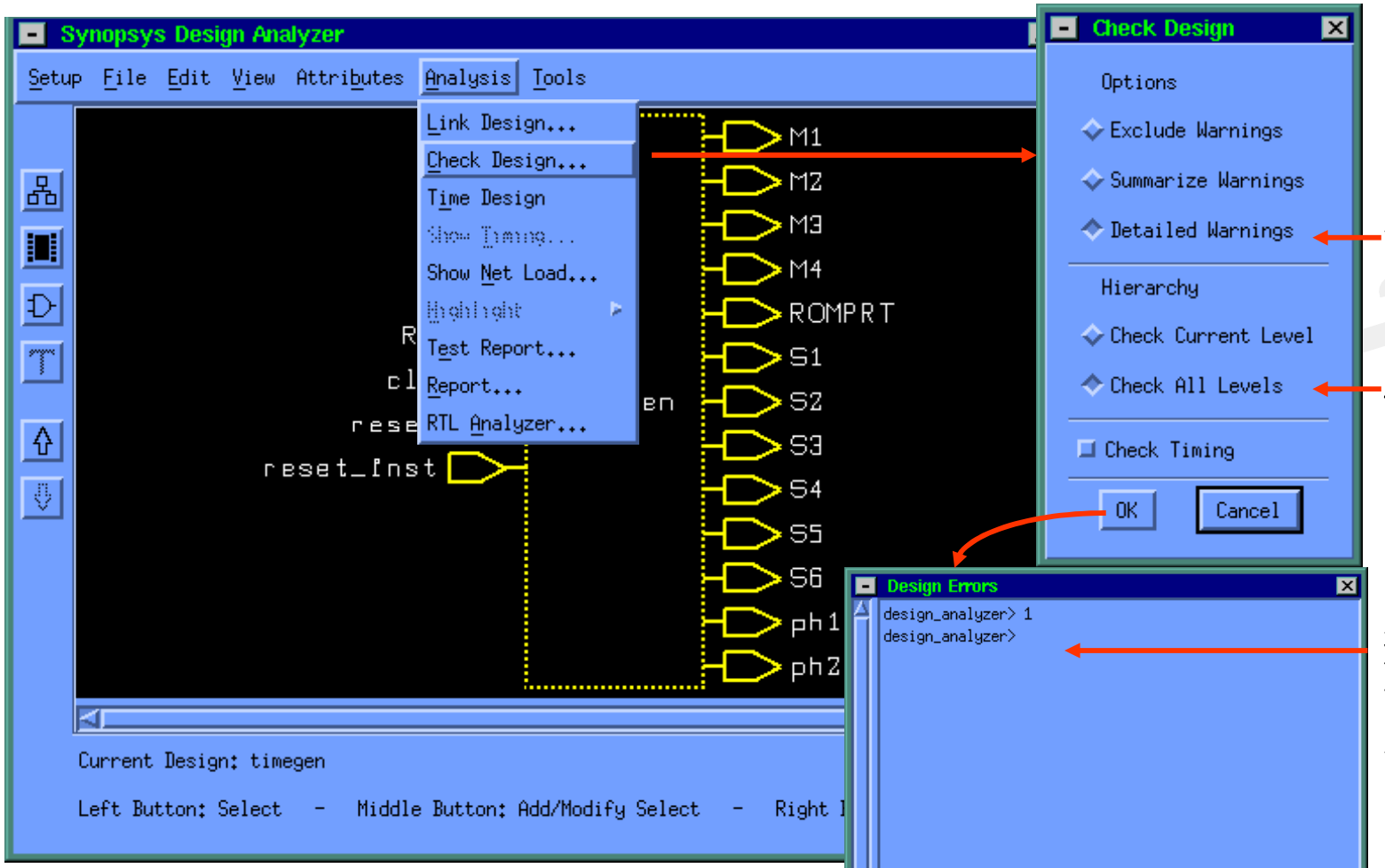
Coding_2

Additional Setting

CCU EE VLSI Lab

細節部分請參考 SYNOPSYS on line document
--- Design Compiler Family Reference Manual

Check Design



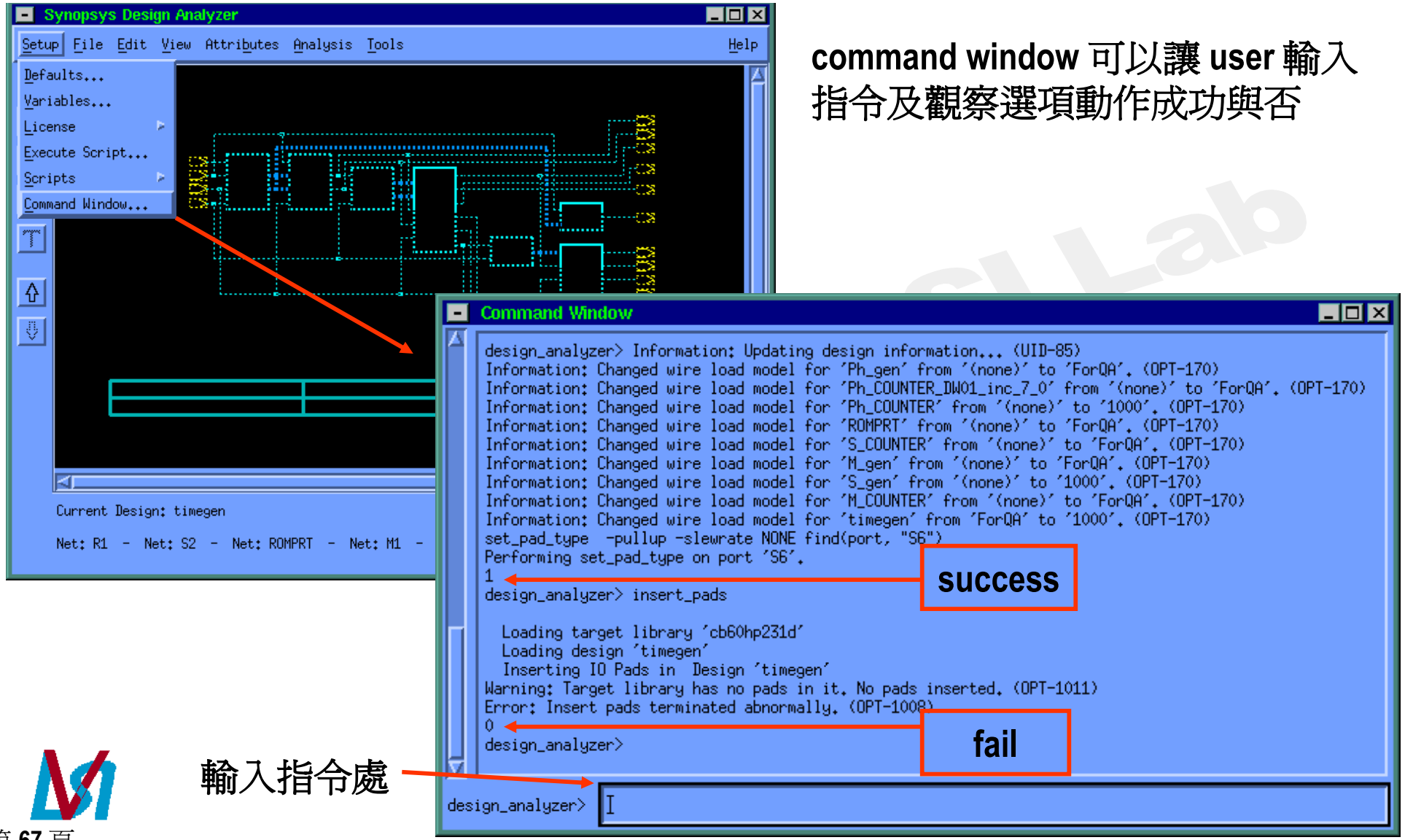
最詳細的 Report

檢查所有的電路

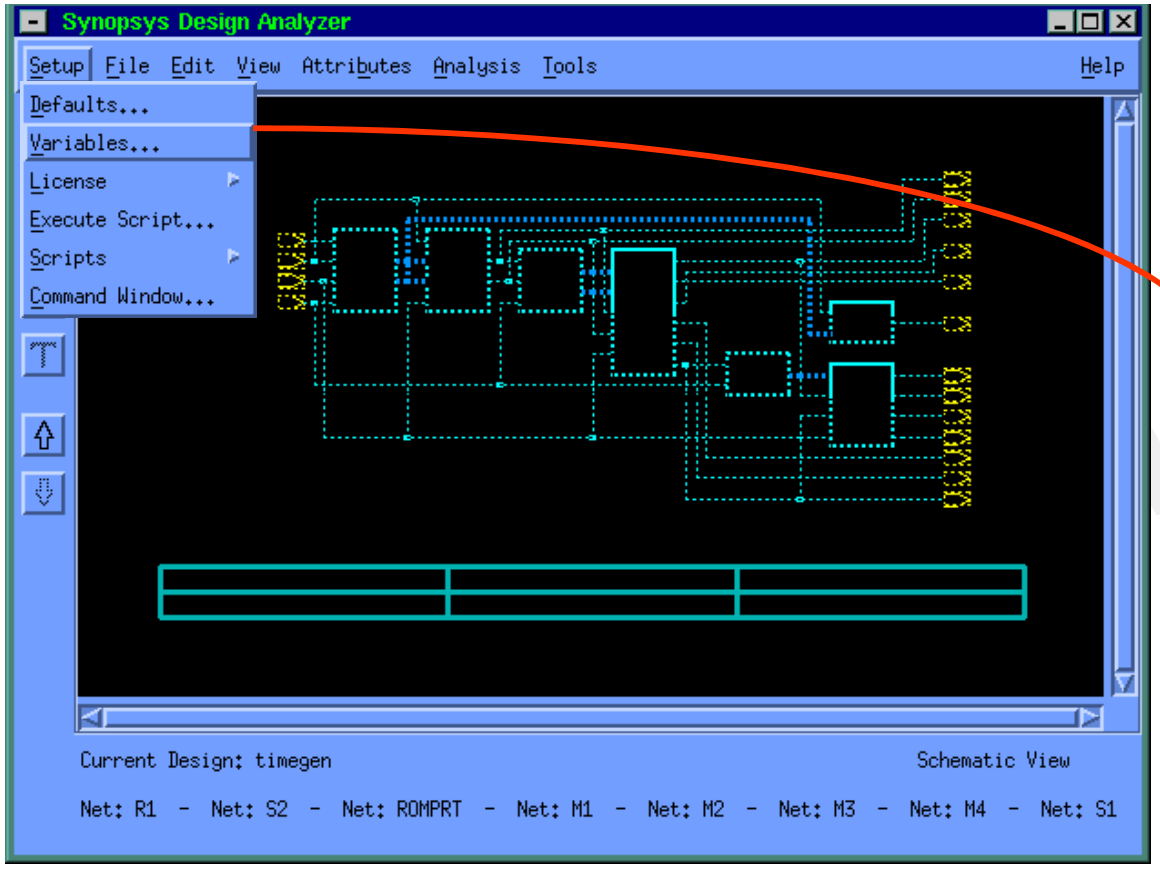
若電路有 errors 或 warnings 或顯示於此視窗, 若完全無錯則如圖所示

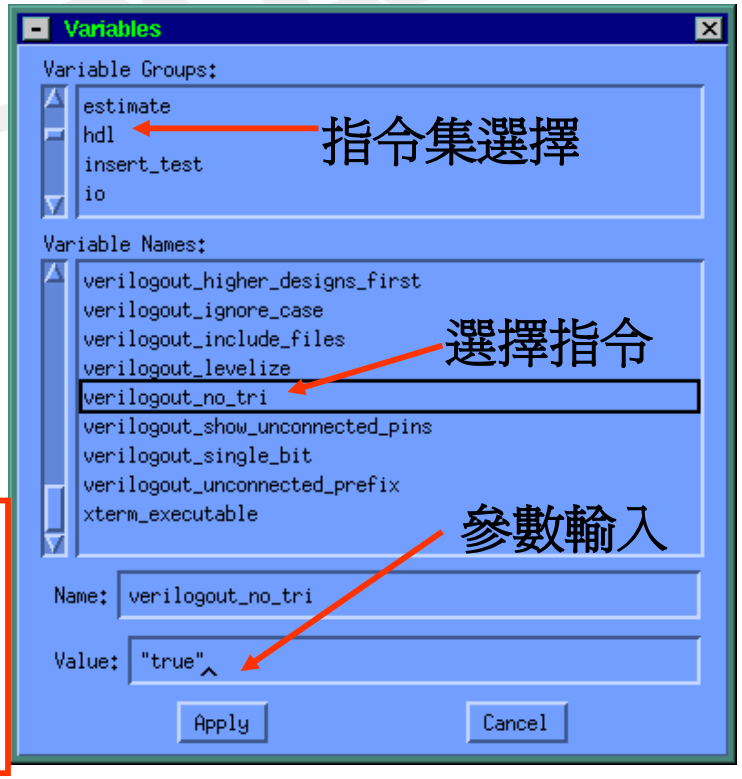
利用 Check Design 檢查設計是否有問題 .

Command Window



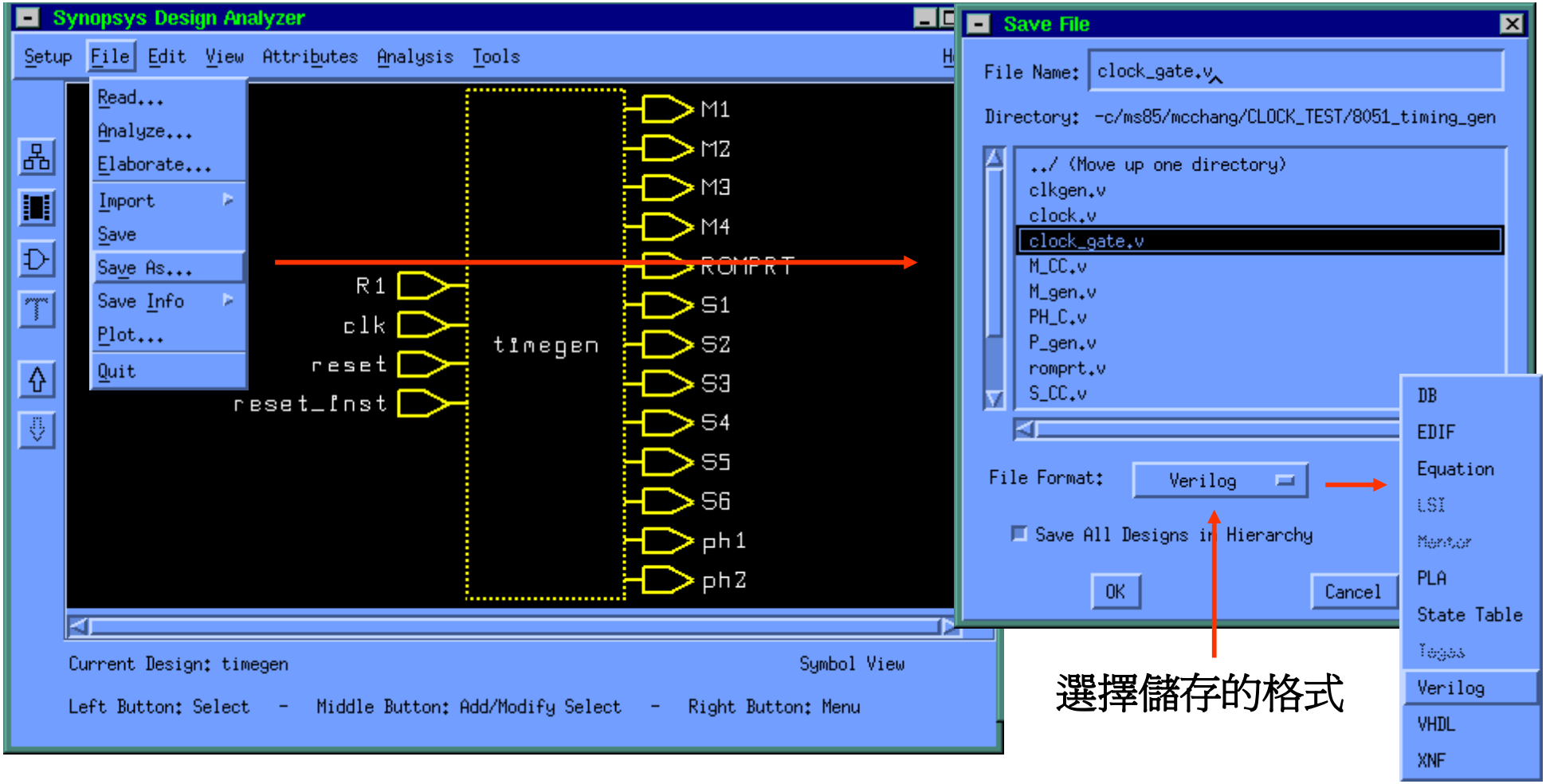
command window 可以讓 user 輸入指令及觀察選項動作成功與否





: Variables window 可以讓 user 輸入修改設定. 圖例是針對verilogout_no_tri 的設定由 false 更改爲 true (如同修改 .synopsys_dc_setup 一般)

Simulate the Synthesized Circuit



(1) Save the synthesized circuit as verilog format (*.v)

Simulate the Synthesized Circuit(cont.)

(2) Simulation

方法一 : Verilog XL command line simulation method

ex) 寫一個含測試 pattern 的 test.v

hsieh> verilog incr.v test.v -f ~/simopt.f

其中 , incr.v 是我們由 Synopsys 轉出的 verilog file

simopt.f 是有關 library 的路徑設定 (User 自建 , 如下頁)

方法二 : Verilog In

" Verilog In"是在 Cadence 環境下 , 將 netlist 轉成 schematic 的工具 . 由於 Verilog In 只接受 netlist , 而我們由 Synopsys 轉出來的 verilog file 可能含有一些 behavior level 的描述 , 因此不能直接用 Verilog In 去得到 schematic 去做模擬 。但是 , 只要適當的修改設定檔案，我們由 Synopsys 轉出的 verilog file , 就可以直接用 Verilog In去做模擬 。方法如下：

在 .synopsys_dc.setup 檔案中，尋找 verilogout_no_tri = " false";

將他改爲verilogout_no_tri="true"; 就可以了。

Variables of Verilog and Simopt.f file

Variables of Verilog :

```
-s                : interactive mode
-f < filename >: read host command arguments from file
+venv             : invoke the verilog control window and LSE
-h                : help
```

simopt.f 檔案內容如下 :

```
+ism
-v /vlsi-a/Librarys/LIB06_V2/Verilog/cb60hp231d.ismvmd
-v /vlsi-a/Librarys/LIB06_V2/Verilog/cb60hd231d.ismvmd
-v /vlsi-a/Librarys/LIB06_V2/Verilog/cb60io420d.ismvmd
-v /vlsia/Librarys/LIB06_V2/Verilog/cb60hp231d/cells/support/udps.vmd
-v /vlsi-a/Librarys/LIB06_V2/Verilog/cb60hd231d/cells/support/udps.vmd
```

nonlinear model

linear model

CCU EE VLSI Lab

Reference :

- SYNOPSYS on line document
  --- Design Compiler Family Reference Manual
  --- HDL Compiler for Verilog Reference
- CIC 使用手冊
  --- Login Synthesis Training Manual

CCU EE VLSI Lab