

此說明檔以題目中的 example design 為範例以及題目中的 true path 1 來做說明。

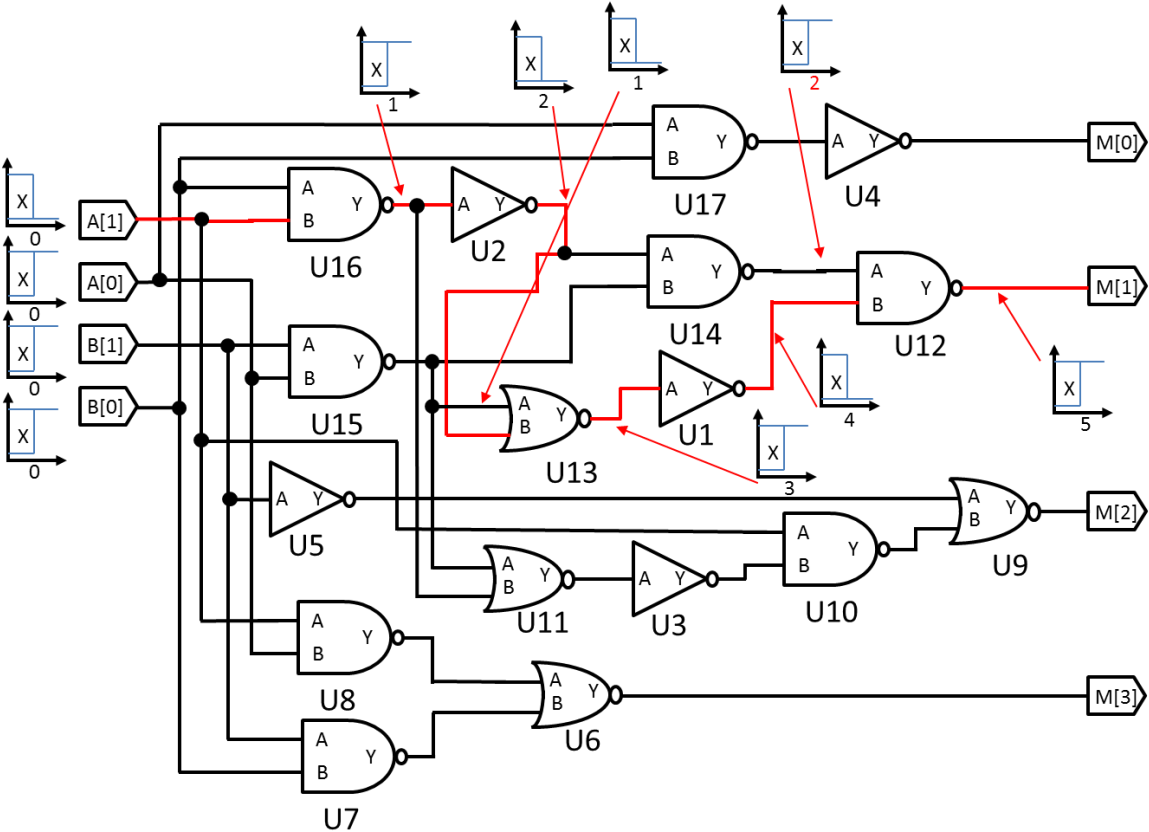


Figure 1: Example design.

Header { A True Path Set }

Benchmark { case1 }

Path { 1 }

A True Path List

{

Pin	type	Incr	Path delay
A[1] (in)		0	0 f
U16/B (NAND2)		0	0 f
U16/Y (NAND2)		1	1 r
U2/A (NOT1)		0	1 r
U2/Y (NOT1)		1	2 f
U13/B (NOR2)		0	2 f
U13/Y (NOR2)		1	3 r
U1/A (NOT1)		0	3 r
U1/Y (NOT1)		1	4 f

U12/B (NAND2)	0	4 f
U12/Y (NAND2)	1	5 r
M[1] (out)	0	5 r

---

Data Required Time	10
Data Arrival Time	5

---

Slack	5
-------	---

}

Input Vector

{

A[0] = 1

A[1] = f

B[0] = 1

B[1] = 1

}

## Verify 程式

驗證 path 格式是否正確和存在，以及驗證是否有相同重複的 path，此程式會利用 prime time 做驗證。

### Step1 :

將參賽者的 true path set file 轉換成 prime time 的 command。

此檔案放置於 PD\_case/caseX/tmp/parallel\_execute\_file

```
report_timing -input_pins -signifi 0 -slack_lesser_than 5 \
-fall_from A[1] \
-fall_through U16/B \
-rise_through U16/Y \
-rise_through U2/A \
-fall_through U2/Y \
-fall_through U13/B \
-rise_through U13/Y \
-rise_through U1/A \
-fall_through U1/Y \
-fall_through U12/B \
-rise_through U12/Y \
-rise_to M[1] \
```

### Step2 :

利用上一個步驟產生的 parallel\_execute\_file 放入 prime time 執行  
執行結果放置於 PD\_case/caseX/tmp/parallel\_execute\_file.rpt

Startpoint: A[1] (input port)  
Endpoint: M[1] (output port)  
Path Group: \*\*default\*\*  
Path Type: max

Point	Incr	Path
-----		
input external delay	0	0 f
A[1] (in)	0	0 f
U16/B (NAND2) <-	0	0 f
U16/Y (NAND2) <-	1	1 r
U2/A (NOT1) <-	0	1 r
U2/Y (NOT1) <-	1	2 f
U13/B (NOR2) <-	0	2 f
U13/Y (NOR2) <-	1	3 r
U1/A (NOT1) <-	0	3 r
U1/Y (NOT1) <-	1	4 f
U12/B (NAND2) <-	0	4 f
U12/Y (NAND2) <-	1	5 r
M[1] (out)	0	5 r
data arrival time		5
max_delay	10	10
output external delay	0	10
data required time		10
-----		
data required time		10
data arrival time		-5
-----		
slack (MET)		5

### Step3 :

利用程式截取 case1\_true\_path\_set 中的 path 資訊，結果放置於  
PD\_case/caseX/tmp/tmp1 中。

A[1](in)00f  
U16/B(NAND2)00f  
U16/Y(NAND2)11r  
U2/A(NOT1)01r  
U2/Y(NOT1)12f  
U13/B(NOR2)02f  
U13/Y(NOR2)13r  
U1/A(NOT1)03r  
U1/Y(NOT1)14f  
U12/B(NAND2)04f  
U12/Y(NAND2)15r  
M[1](out)05r  
slack5

**Step4 :**

利用程式截取 parallel\_execute\_file.rpt 中的 path 資訊，結果放置於 PD\_case/caseX/tmp/tmp2 中。

1  
A[1](in)00f  
U16/B(NAND2)00f  
U16/Y(NAND2)11r  
U2/A(NOT1)01r  
U2/Y(NOT1)12f  
U13/B(NOR2)02f  
U13/Y(NOR2)13r  
U1/A(NOT1)03r  
U1/Y(NOT1)14f  
U12/B(NAND2)04f  
U12/Y(NAND2)15r  
M[1](out)05r  
slack5

**Step5 :**

利用 diff 比較 tmp1 與 tmp2 是否相同，若不相同代表 true path set 中的 path 有誤或格式有誤，比較結果放置於 PD\_case/caseX/output/diff.log。

**Step6 :**

利用程式檢查 true path set 中 path 是否有重複。

以上六個步驟中若有任一個步驟發生錯誤都會讓最後的結果為 fail。

## Justify 程式

Justify 程式會將參賽者的 true path set file 轉換成 simulation testbench，作 floating mode simulation，驗證是否為 true path。

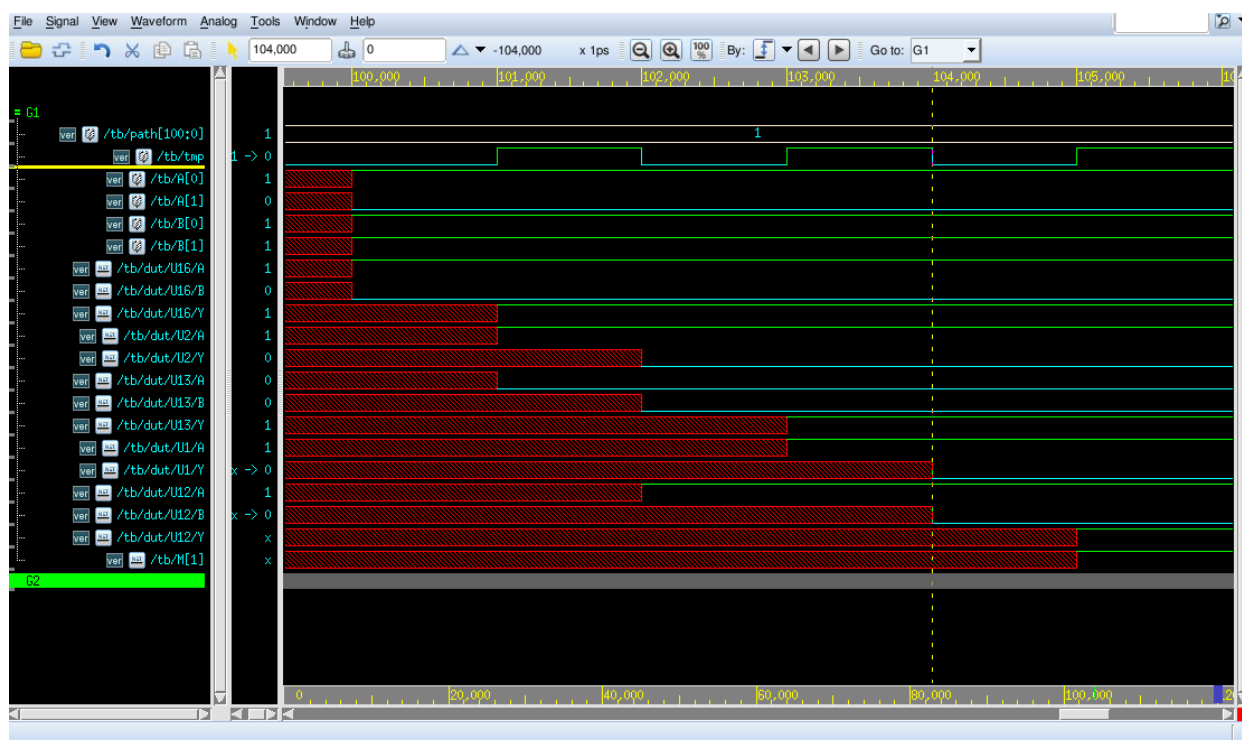


Figure2: Floating mode simulation example for true path 1.

Figure2 為 simulation 的結果，其中 path[100:0] 訊號表示目前正在驗證的 path，tmp 為其相對應時間點上訊號的正確值。根據 input vector 為  $A[0] = 1, A[1] = f, B[0] = 1, B[1] = 1$ ，此為一條 falling path，依據 floating mode simulation 的定義 A[0], B[0], B[1] 三個 input 的訊號皆為 unknow 到邏輯 1，而 A[1] 則為 unknow 到邏輯 0 的訊號，接著在 U16 的輸入端 A 為 rising 訊號，而 B 為 falling 的訊號如題目中的 Figure 9 所示，依據題目中 Table1-4 的定義以及題目中第 3 頁的定義說明當兩個訊號同時到達 NAND2 gate 時，Falling 訊號為 true path，因此 U16/B 到 U16/Y 為 **true path**，接著訊號由 U2/A 傳遞到 U2/Y 會經過 1 個 ns 的 delay，U2 為 NOT1 gate 故 U2/A 到 U2/Y 必定為 true path 且訊號會反向，所以 U2/Y 的訊號為 falling 訊號。接下來訊號到達 U13/B，依照定義 NOR2 gate，兩個訊號都是 falling 時，晚到的是 true path，故 U13/B 到 U13/Y 為 true path，以此類推，

U1/A 到 U1/Y 是 true path，最後訊號到達 U12/B 時因 U12/B 為 controlling value 且 U12/A 為 non-controlling value 且 U12/A 較 U12/B 早到達，所以依據題目中 Table1-2 的定義以及說明，U12/B 到 U12/Y 為 true path。經過每一個 gate 的推論我們可以證明此組 input vector 可證明此條 path 為 true path。

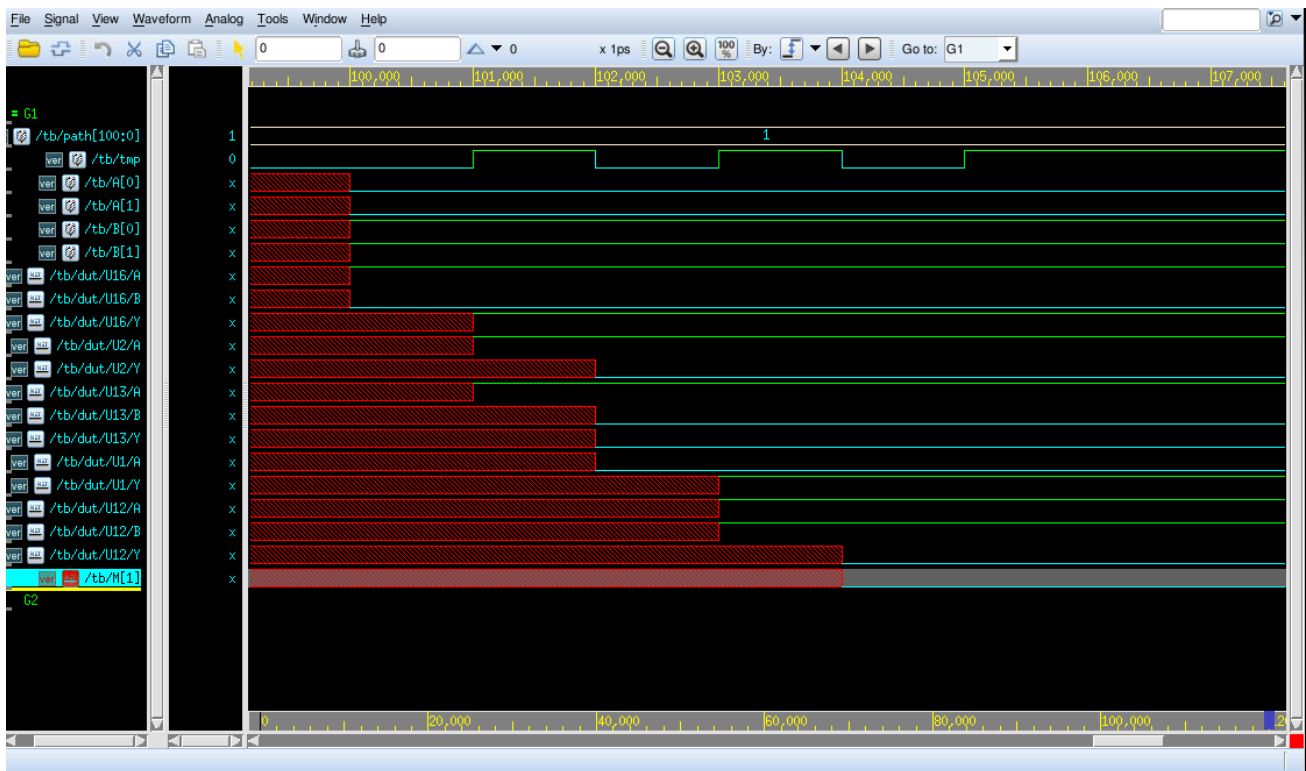


Figure 3: Floating mode simulation example for true path 1.

如 Figure3 所示，假設我們將 input vector 改為  $A[0] = 0$ ,  $A[1] = f$ ,  $B[0] = 1$ ,  $B[1] = 1$ ，此時 U16/B 到 U16/Y 依然是 true path，U2/A 到 U2/Y 依然也是 true path，但 U13/A 為 rising 訊號以及 U13/B 為 falling 訊號，依照 Table1-1 的定義 U13/A 到 U13/Y 為 true path，U13/B 到 U13/Y 為 false path，故此 input vector 不能證明此條 path 為 true path。若要知道哪一個 gate 發生問題可參考 PD\_case/caseX/tmp/vcs.log.caseX

以下為一個 vcs.log.caseX 的範例，其中的 U13, U1, U12 在 simulation 時發現了此組 input vector 不能證明此條 path 為 true path。

```
path=1
error 1 .U13.Y.Cx0
error 1 .U13.Y.tmp
error 1 .U1.Y.Cx1
error 1 .U1.Y.tmp
error 1 .U12.Y.Cx0
error 1 .U12.Y.tmp
path=end
```

根據以上推論，我們可驗證此組 input vector 是否可以證明這條 path 為 true path。