第三章

資料流描述與組合電路

3-1 共時性與順序性

1. 組合邏輯電路：共時性（concurrent）

- 訊號從輸入端開始就會以平行方式傳送至輸出端，沒有順序產生。
- 於Verilog中的程式敘述大都具備這種特性，這些敘述稱為資料流(data flow)敘述。

2. 序向邏輯電路：順序性（sequential）

- 各種元件(正反器及時脈等)訊號存在著優先順序。
- 這些敘述稱之為行為(behavior)模式。

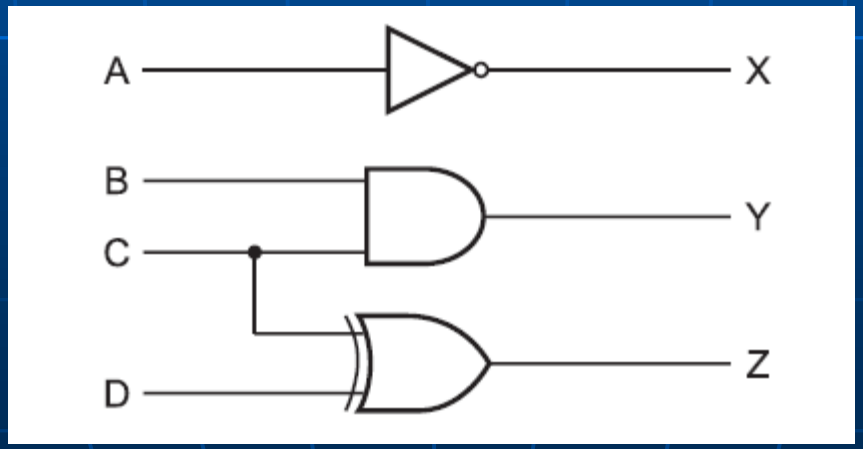
共時性 concurrent

敘述 A：

```
assign X = ~A;
assign Y = B & C;
assign Z = C ^ D;
```

敘述 B：

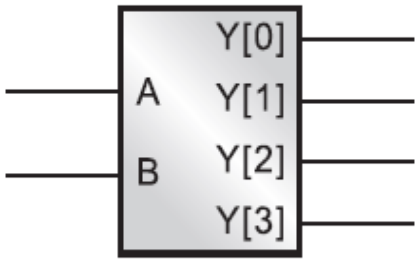
```
assign Y = B & C;
assign Z = C ^ D;
assign X = ~A;
```



3-2 持續指定與真值表描述

範例一	檔名：DECODER_2_4_HIGH_BOOLEAN
以布林代數持續指定方式，設計一個高態動作的 2 對 4 解碼器電路。	

1. 方塊圖：



2. 真值表：

A	B	Y[0]	Y[1]	Y[2]	Y[3]
0	0	1	0	0	0
0	1	0	1	0	0
1	0	0	0	1	0
1	1	0	0	0	1

3. 布林代數：

$Y[0] = \overline{A}\,\overline{B}$

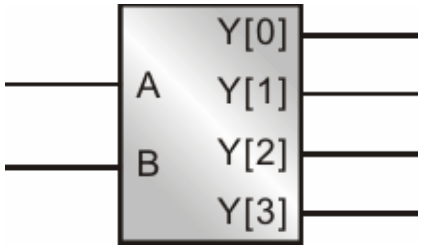
$Y[1] = \overline{A}B$

$Y[2] = A\overline{B}$

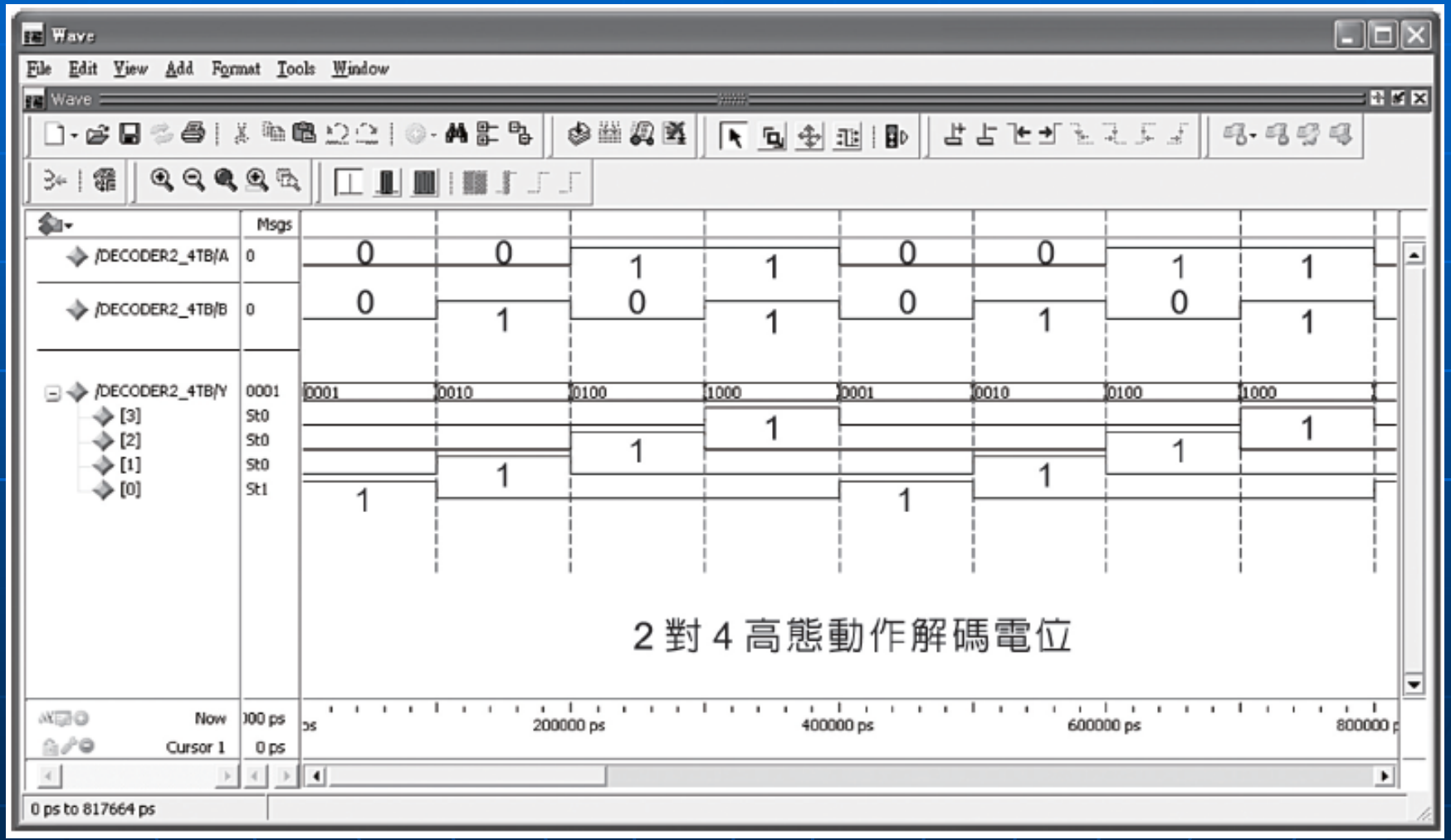
$Y[3] = AB$

原始程式 (source program) :

```
 1  /*********************************
 2   * 2 to 4 decoder active high   *
 3   *         using boolean        *
 4   *    Filename : DECODER2_4.v   *
 5   *********************************/
 6
 7  module DECODER2_4(A, B, Y);
 8
 9    input  A, B;
10    output [3:0] Y;
11
12    assign Y[0] = ~A & ~B;
13    assign Y[1] = ~A &  B;
14    assign Y[2] =  A & ~B;
15    assign Y[3] =  A &  B;
16
17  endmodule
```

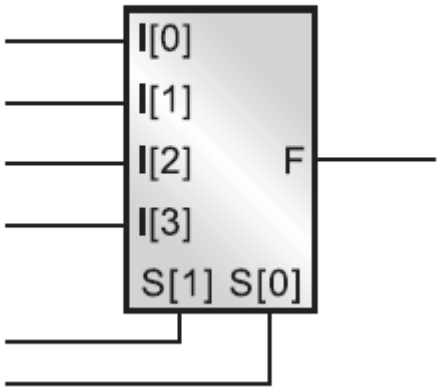


功能模擬 ：


Wave
File Edit View Add Format Tools Window
Wave
Msgs
/DECODER2_4TB/A
/DECODER2_4TB/B
/DECODER2_4TB/Y
[3]
[2]
[1]
[0]
0001
St0
St0
St0
St1
0001
0010
0100
1000
0001
0010
0100
1000
2對4高態動作解碼電位
Now
Cursor 1
0 ps
200000 ps
400000 ps
600000 ps
0 ps to 817664 ps

範例二	檔名：MUL4_1_BOOLEAN
以布林代數持續指定方式，設計一個 1 位元 4 對 1 的多工器電路。	

1. 方塊圖：



2. 真值表：

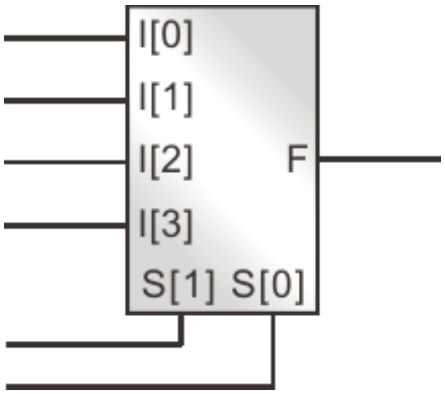
S[1]	S[0]	F
0	0	I[0]
0	1	I[1]
1	0	I[2]
1	1	I[3]

3. 布林代數：

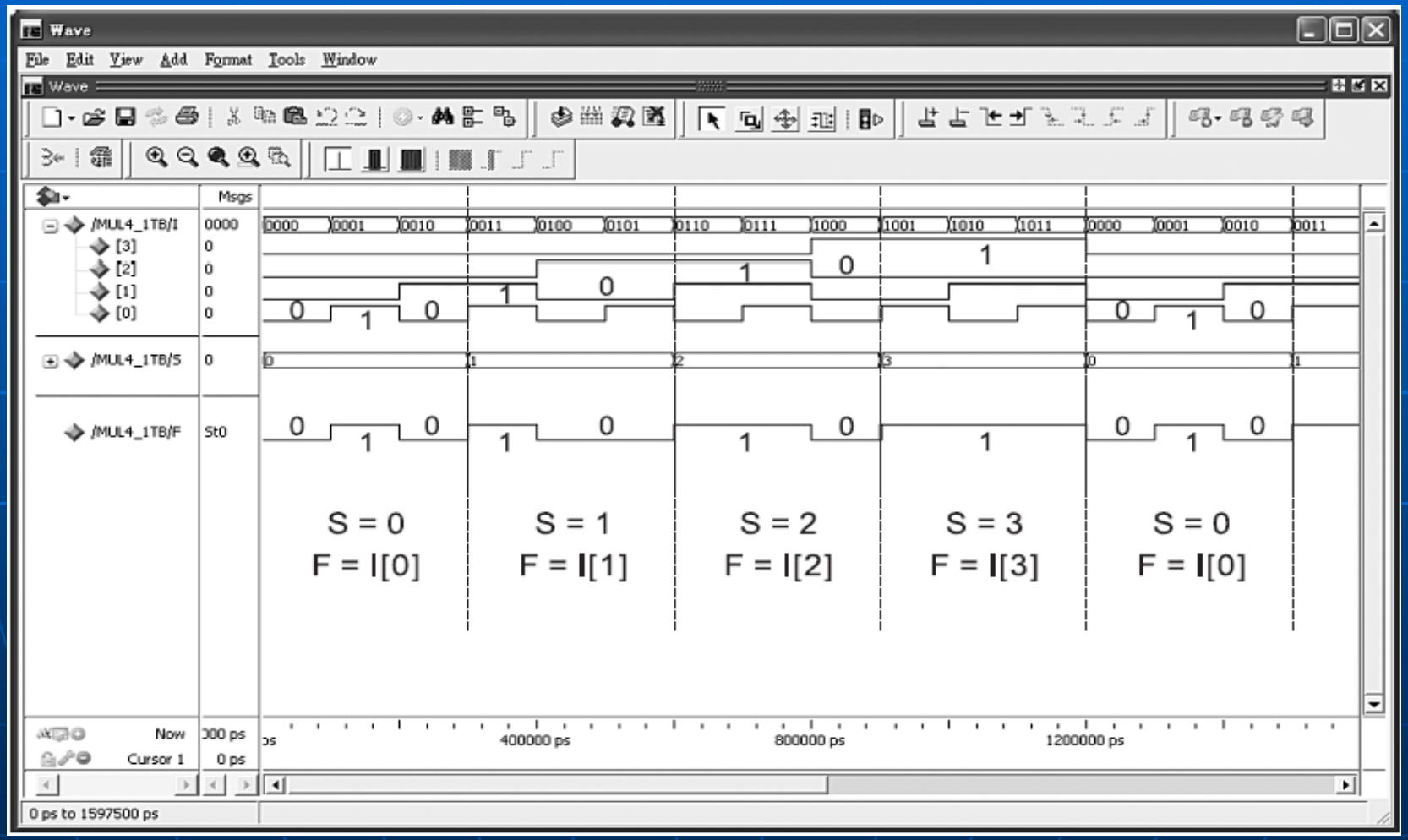
$$F = \overline{S[1]}\ \overline{S[0]}\ I[0] + \overline{S[1]}\ S[0]\ I[1] + S[1]\ \overline{S[0]}\ I[2] + S[1]\ S[0]\ I[3]$$

原始程式 (source program) :

```
 1  /*************************
 2   *    4 to 1 multiplexer   *
 3   *       using boolean     *
 4   *   Filename : MUL4_1.v   *
 5   *************************/
 6
 7  module MUL4_1(I, S, F);
 8
 9    input  [3:0] I;
10    input  [1:0] S;
11    output F;
12
13    assign F = ~S[1]  & ~S[0]  & I[0]|
14               ~S[1]  &  S[0]  & I[1]|
15                S[1]  & ~S[0]  & I[2]|
16                S[1]  &  S[0]  & I[3];
17
18  endmodule
```



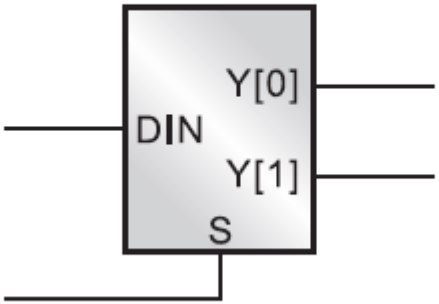
功能模擬 ：


Wave
File Edit View Add Format Tools Window
Wave
Msgs
/MUL4_1TB/I
[3]
[2]
[1]
[0]
/MUL4_1TB/S
/MUL4_1TB/F
St0
0000 0001 0010 0011 0100 0101 0110 0111 1000 1001 1010 1011 0000 0001 0010 0011
S = 0
F = I[0]
S = 1
F = I[1]
S = 2
F = I[2]
S = 3
F = I[3]
S = 0
F = I[0]
Now
Cursor 1
400000 ps
800000 ps
1200000 ps
0 ps to 1597500 ps

3-3 以條件指定實現組合邏輯

範例一	檔名：DEMUL1_2_CONDITION
以單行的條件敘述，設計一個 1 對 2 解多工器電路。	

1. 方塊圖：

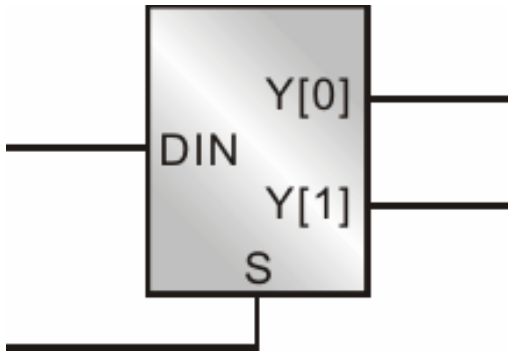


2. 真值表：

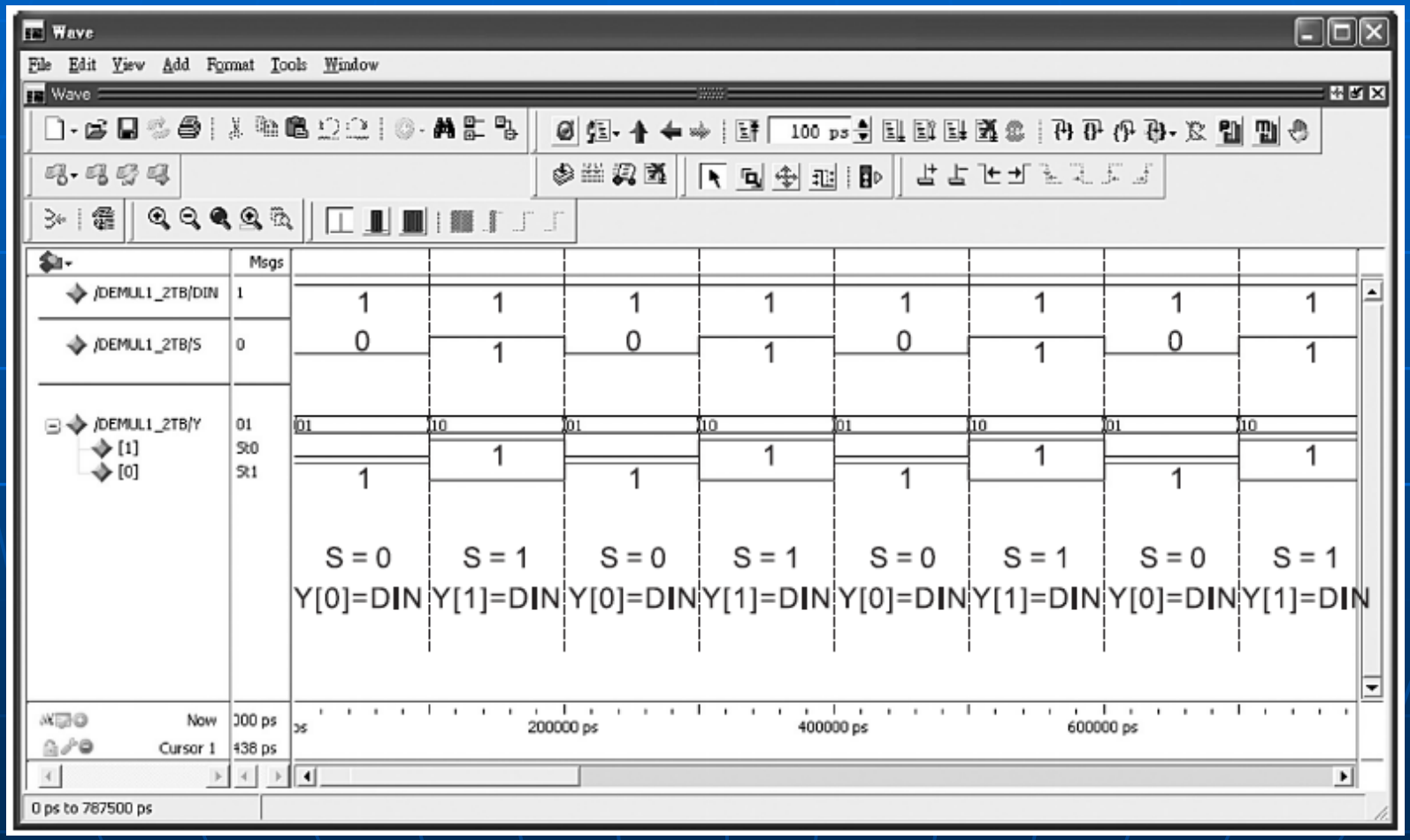
S	DIN	Y[0]	Y[1]
0	0	0	0
0	1	1	0
1	0	0	0
1	1	0	1

原始程式 (source program) :

```
 1  /***************************
 2   *  1 to 2 DEMULTIPLEXER   *
 3   *    using condition ?:   *
 4   *  Filename : DEMUL1_2.v  *
 5   ***************************/
 6
 7  module DEMUL1_2(DIN, S, Y);
 8
 9    input   DIN, S;
10    output [1:0] Y;
11
12    assign  Y[0] = (~S) ? DIN : 1'b0;
13    assign  Y[1] = ( S) ? DIN : 1'b0;
14
15  endmodule
```

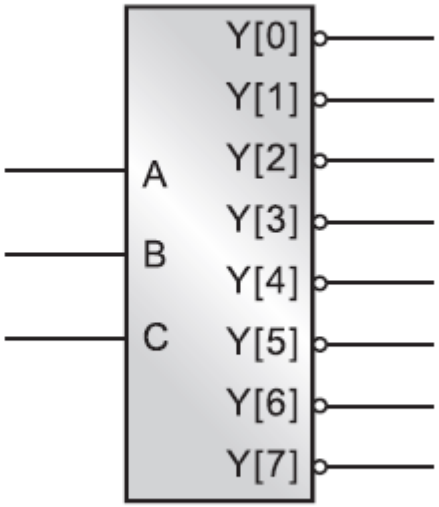


功能模擬 ：


Wave
File Edit View Add Format Tools Window
Wave
100 ps
Msgs
/DEMUL1_2TB/DIN
/DEMUL1_2TB/S
/DEMUL1_2TB/Y
[1]
[0]
S = 0
Y[0]=DIN
S = 1
Y[1]=DIN
Now
Cursor 1
200000 ps
400000 ps
600000 ps
0 ps to 787500 ps

範例二	檔名：DECODER3_8_LOW_CONDITION
以單行的條件敘述，設計一個低態動作的 3 對 8 解碼器電路。	

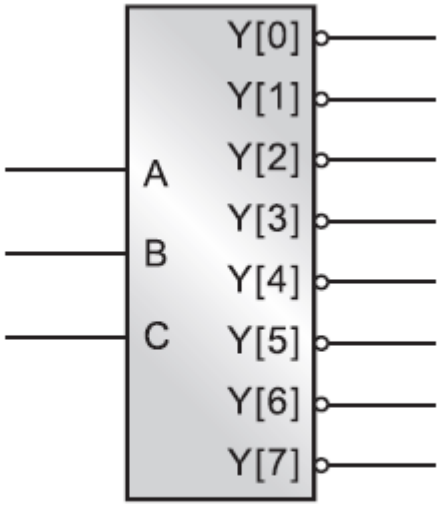
1. 方塊圖：



2. 真值表：

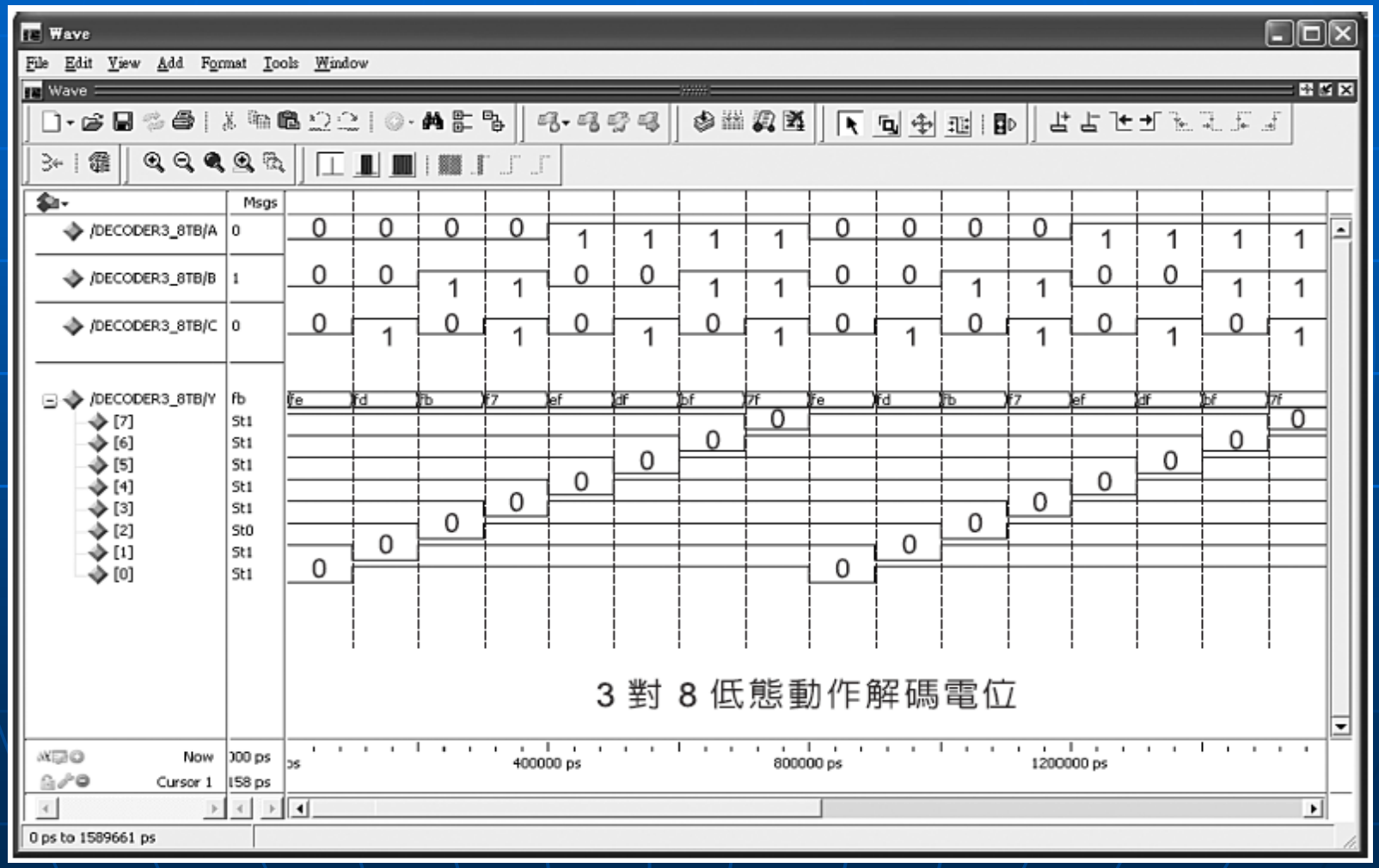
A	B	C	Y[0]	Y[1]	Y[2]	Y[3]	Y[4]	Y[5]	Y[6]	Y[7]
0	0	0	0	1	1	1	1	1	1	1
0	0	1	1	0	1	1	1	1	1	1
0	1	0	1	1	0	1	1	1	1	1
0	1	1	1	1	1	0	1	1	1	1
1	0	0	1	1	1	1	0	1	1	1
1	0	1	1	1	1	1	1	0	1	1
1	1	0	1	1	1	1	1	1	0	1
1	1	1	1	1	1	1	1	1	1	0

原始程式 (source program) :



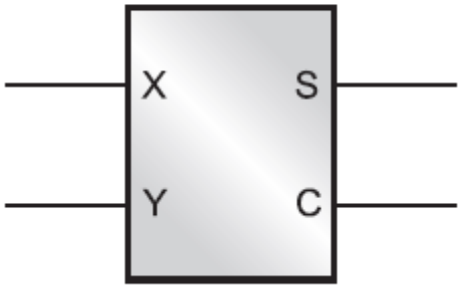

```
 1  /*******************************
 2   * 3 to 8 decoder active high   *
 3   *      using condition ?:      *
 4   *    Filename : DECODER3_8.v   *
 5   *******************************/
 6
 7  module DECODER3_8(A, B, C, Y);
 8
 9    input  A, B, C;
10    output [7:0] Y;
11
12    assign Y[0] = ({A, B, C}  == 3'o0) ? 1'b0 : 1'b1;
13    assign Y[1] = ({A, B, C}  == 3'o1) ? 1'b0 : 1'b1;
14    assign Y[2] = ({A, B, C}  == 3'o2) ? 1'b0 : 1'b1;
15    assign Y[3] = ({A, B, C}  == 3'o3) ? 1'b0 : 1'b1;
16    assign Y[4] = ({A, B, C}  == 3'o4) ? 1'b0 : 1'b1;
17    assign Y[5] = ({A, B, C}  == 3'o5) ? 1'b0 : 1'b1;
18    assign Y[6] = ({A, B, C}  == 3'o6) ? 1'b0 : 1'b1;
19    assign Y[7] = ({A, B, C}  == 3'o7) ? 1'b0 : 1'b1;
20
21  endmodule
```

功能模擬 ：


Wave
File Edit View Add Format Tools Window
Wave
Msgs
/DECODER3_8TB/A
/DECODER3_8TB/B
/DECODER3_8TB/C
/DECODER3_8TB/Y
[7]
[6]
[5]
[4]
[3]
[2]
[1]
[0]
fb
St1
St0
fe fd fb f7 ef df bf 7f
3 對 8 低態動作解碼電位
Now
Cursor 1
400000 ps
800000 ps
1200000 ps
0 ps to 1589661 ps

範例三	檔名：HALF_ADDER_CONDITION_NEST
以多行的條件敘述，設計一個半加器電路。	

1. 方塊圖：

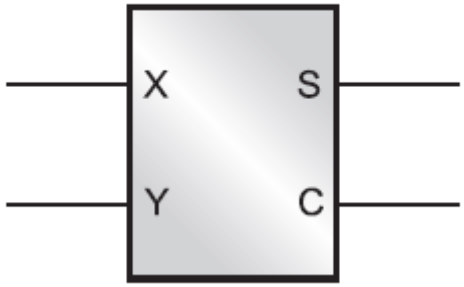


2. 真值表：

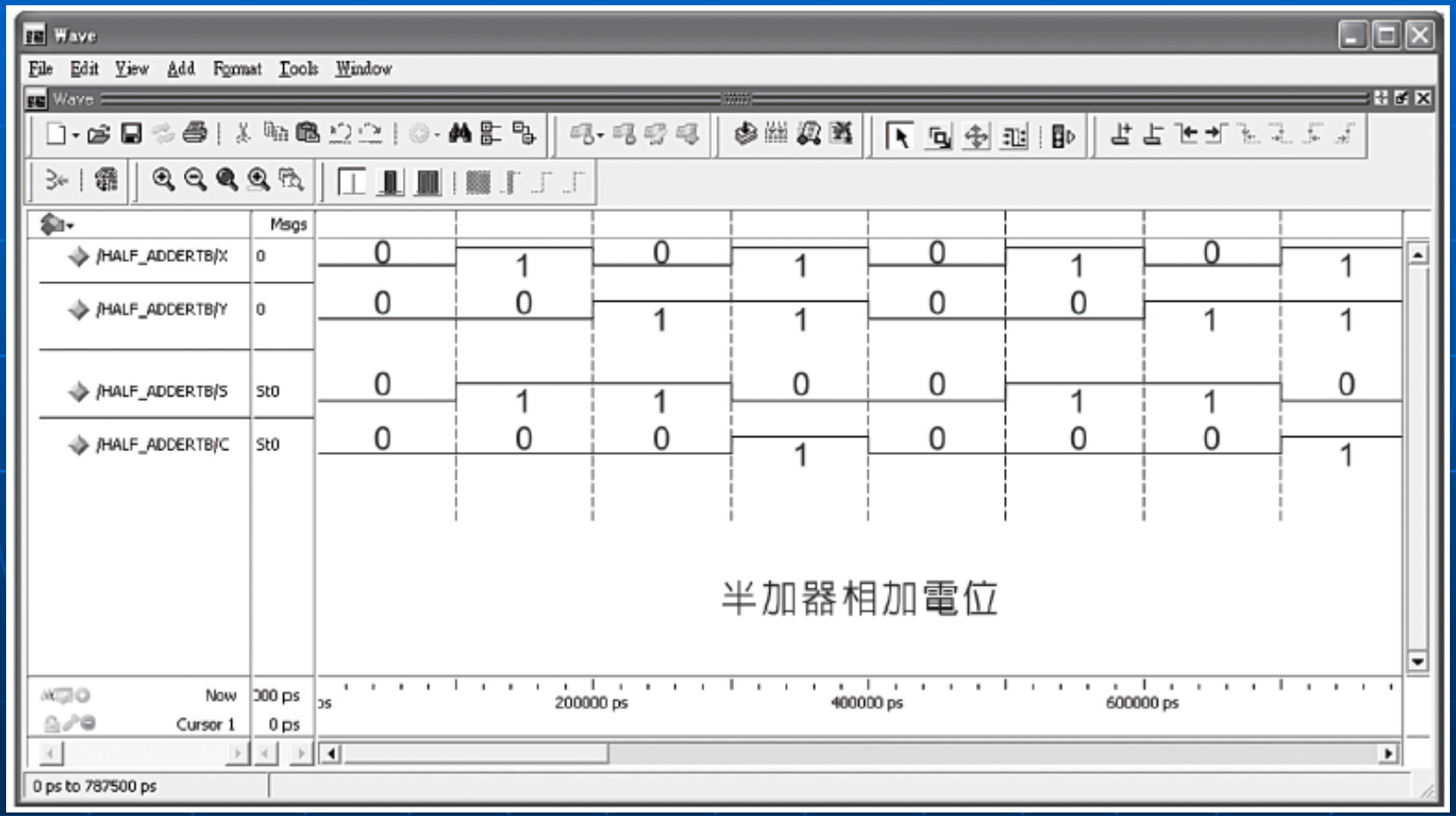
X	Y	S	C
0	0	0	0
0	1	1	0
1	0	1	0
1	1	0	1

原始程式 (source program) :

```
 1  /*********************************
 2   *           half adder          *
 3   *    using condition ?: nest    *
 4   *   Filename : HALF_ADDER.v     *
 5   *********************************/
 6
 7  module HALF_ADDER(X, Y, S, C);
 8
 9    input  X, Y;
10    output S, C;
11
12    assign S = (~X &  Y) ? 1'b1 :
13               ( X & ~Y) ? 1'b1 :
14                           1'b0;
15    assign C = ( X &  Y) ? 1'b1 : 1'b0;
16
17  endmodule
```

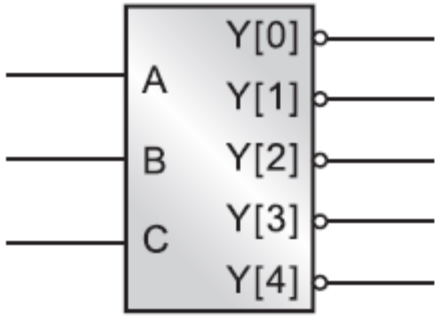


功能模擬 ：


Wave
File Edit View Add Format Tools Window
Msgs
/HALF_ADDERTB/X
/HALF_ADDERTB/Y
/HALF_ADDERTB/S
/HALF_ADDERTB/C
St0
Now
Cursor 1
0 ps
200000 ps
400000 ps
600000 ps
0 ps to 787500 ps
半加器相加電位

範例四	檔名：MULTIPLE_DECODER3_5_LOW_CONDITION_NEST
以多行的條件敘述，設計一個低態動作的 3 對 5 多重解碼電路。	

1. 方塊圖：

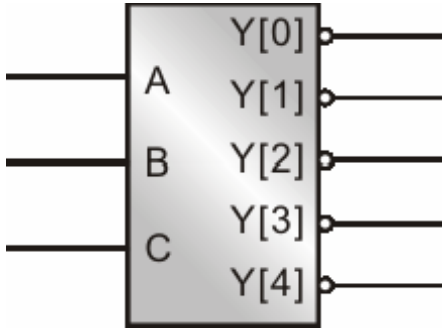


2. 真值表：

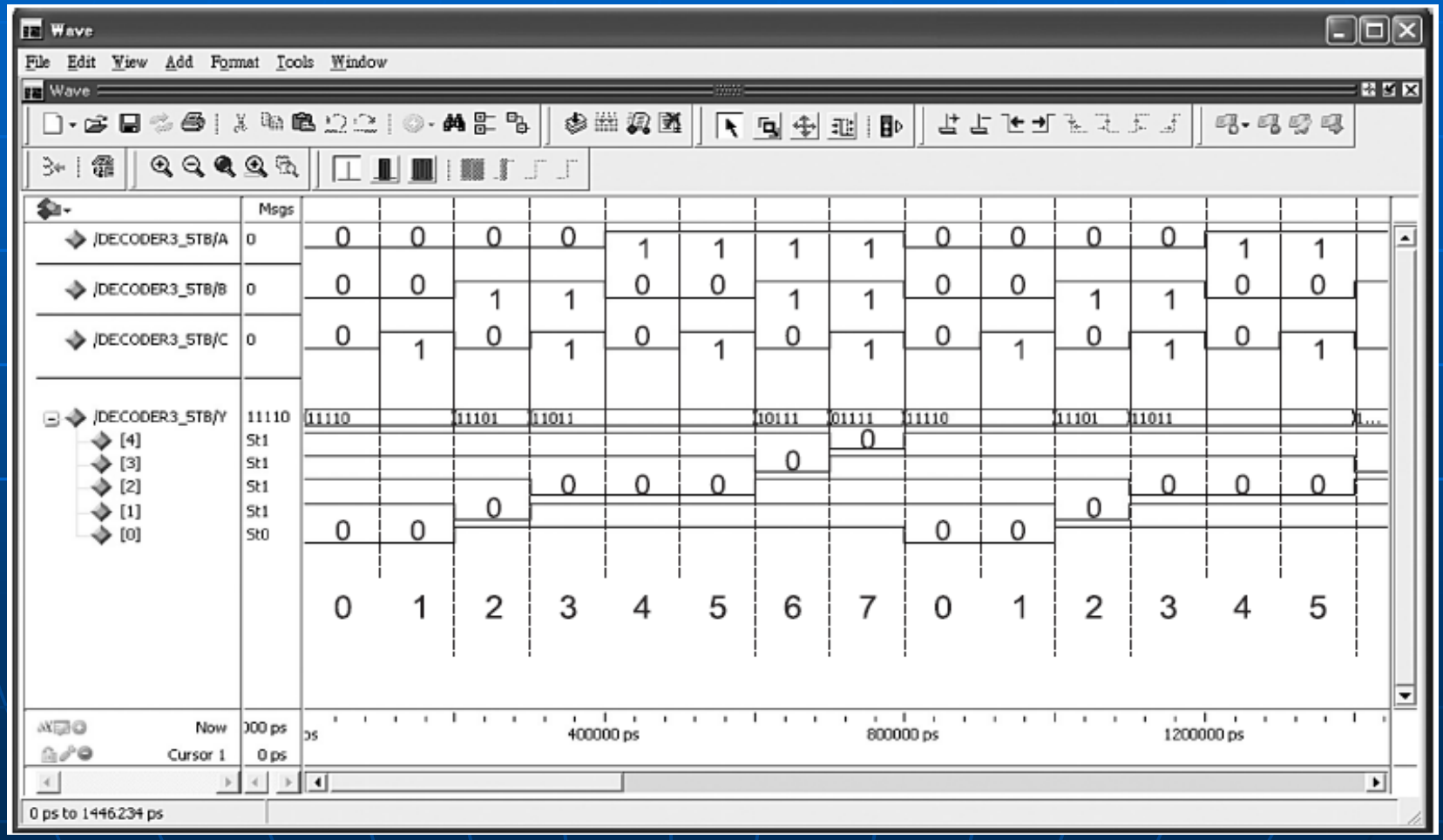
A	B	C	Y[0]	Y[1]	Y[2]	Y[3]	Y[4]
0	0	0	0	1	1	1	1
0	0	1	0	1	1	1	1
0	1	0	1	0	1	1	1
0	1	1	1	1	0	1	1
1	0	0	1	1	0	1	1
1	0	1	1	1	0	1	1
1	1	0	1	1	1	0	1
1	1	1	1	1	1	1	0

原始程式 (source program) :

```
 1  /*****************************
 2   *  3 to 5 multiple decoder  *
 3   *  using condition ?: nest  *
 4   *  Filename : DECODER3_5.v  *
 5   *****************************/
 6
 7  module DECODER3_5(A, B, C, Y);
 8
 9    input  A, B, C;
10    output [4:0] Y;
11
12    assign Y = (({A, B, C} == 3'o0)|
13                ({A, B, C} == 3'o1))? 5'b11110 :
14                ({A, B, C} == 3'o2) ? 5'b11101 :
15               (({A, B, C} == 3'o3)|
16                ({A, B, C} == 3'o4)|
17                ({A, B, C} == 3'o5))? 5'b11011 :
18                ({A, B, C} == 3'o6) ? 5'b10111 :
19                                      5'b01111 ;
20
21  endmodule
```



功能模擬 ：


Wave
File Edit View Add Format Tools Window
Wave
Msgs
/DECODER3_STB/A
/DECODER3_STB/B
/DECODER3_STB/C
/DECODER3_STB/Y
11110
[4]
[3]
[2]
[1]
[0]
St1
St0
11110
11101
11011
10111
01111
0 1 2 3 4 5 6 7 0 1 2 3 4 5
Now
Cursor 1
0 ps
400000 ps
800000 ps
1200000 ps
0 ps to 1446234 ps

3-4 結論

依真值表：

1. 直接以布林代數描述。
2. 以二選一的單行條件敘述描述。
3. 以多選一的多行條件敘述描述。