

數位系統設計作業

HW5

學號: 01257027 | 姓名: 林承羿

基本題: 用 2 對 1 多工器組成 4 對 1 多工器(使用 port map)

程式碼

```
1  library IEEE;
2  use IEEE.STD_LOGIC_1164.ALL;
3  use IEEE.STD_LOGIC_ARITH.ALL;
4  use IEEE.STD_LOGIC_UNSIGNED.ALL;
5
6  entity DE2to1 is
7      port(
8          I:in std_logic_vector(1 downto 0);
9          S:in std_logic;
10         X:out std_logic
11     );
12 end DE2to1;
13
14 architecture DE2to1 of DE2to1 is
15 begin
16     process(S, I)
17     begin
18         if(S = '0') then
19             X <= I(0);
20         elsif(S = '1') then
21             X <= I(1);
22         end if;
23     end process;
24 end DE2to1;
```

-- 此項作業的基本component，兩項輸入
-- select 線，選擇高低位輸出
-- 選擇後的輸出
-- 偵測所有輸入、選擇線
-- select 0 表示 選擇 I(0)
-- select 1 表示 選擇 I(1)

基本元件，用此組成所有問題的最小單元

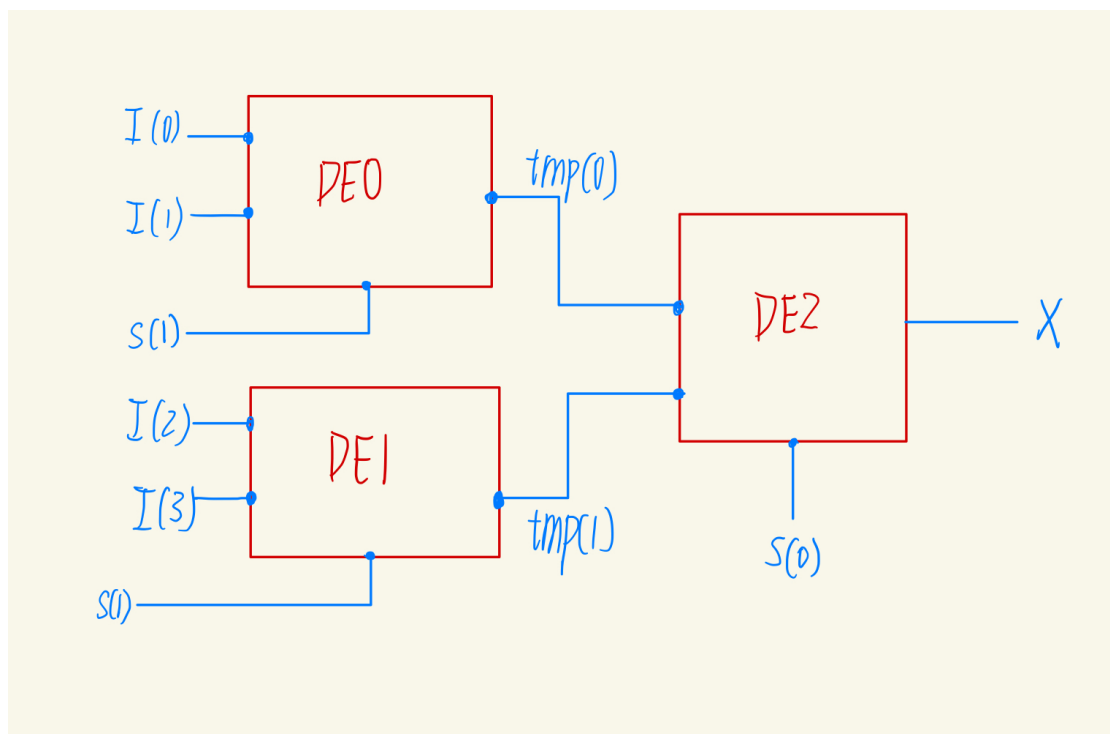
```

2  use IEEE.STD_LOGIC_1164.ALL;
3  use IEEE.STD_LOGIC_ARITH.ALL;
4  use IEEE.STD_LOGIC_UNSIGNED.ALL;
5
6  entity DE4to1 is
7      port(
8          I:in std_logic_vector(3 downto 0);           -- 四項輸入
9          S:in std_logic_vector(1 downto 0);           -- 兩層電路需要兩個select
10         X:out std_logic                                -- 最後輸出
11     );
12 end DE4to1;
13
14 architecture DE4to1 of DE4to1 is
15     signal tmp:std_logic_vector(1 downto 0);         -- 連接兩層電路間的signal
16     component DE2to1                                 -- 組成單元
17     port(
18         I:in std_logic_vector(1 downto 0);
19         S:in std_logic;
20         X:out std_logic
21     );
22 end component;
23 begin
24     DE0:DE2to1 port map(I(1 downto 0), S(1), tmp(0)); -- 引用元件，輸出放進signal
25     DE1:DE2to1 port map(I(3 downto 2), S(1), tmp(1)); -- 引用元件，輸出放進signal
26     DE2:DE2to1 port map(tmp, S(0), X);               -- 第二層電路，接上輸出
27 end DE4to1;

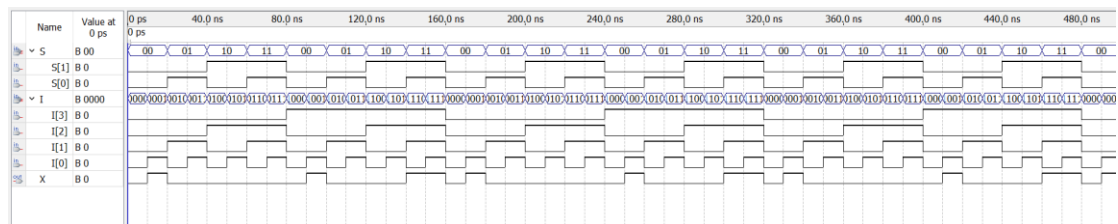
```

以上程式說明

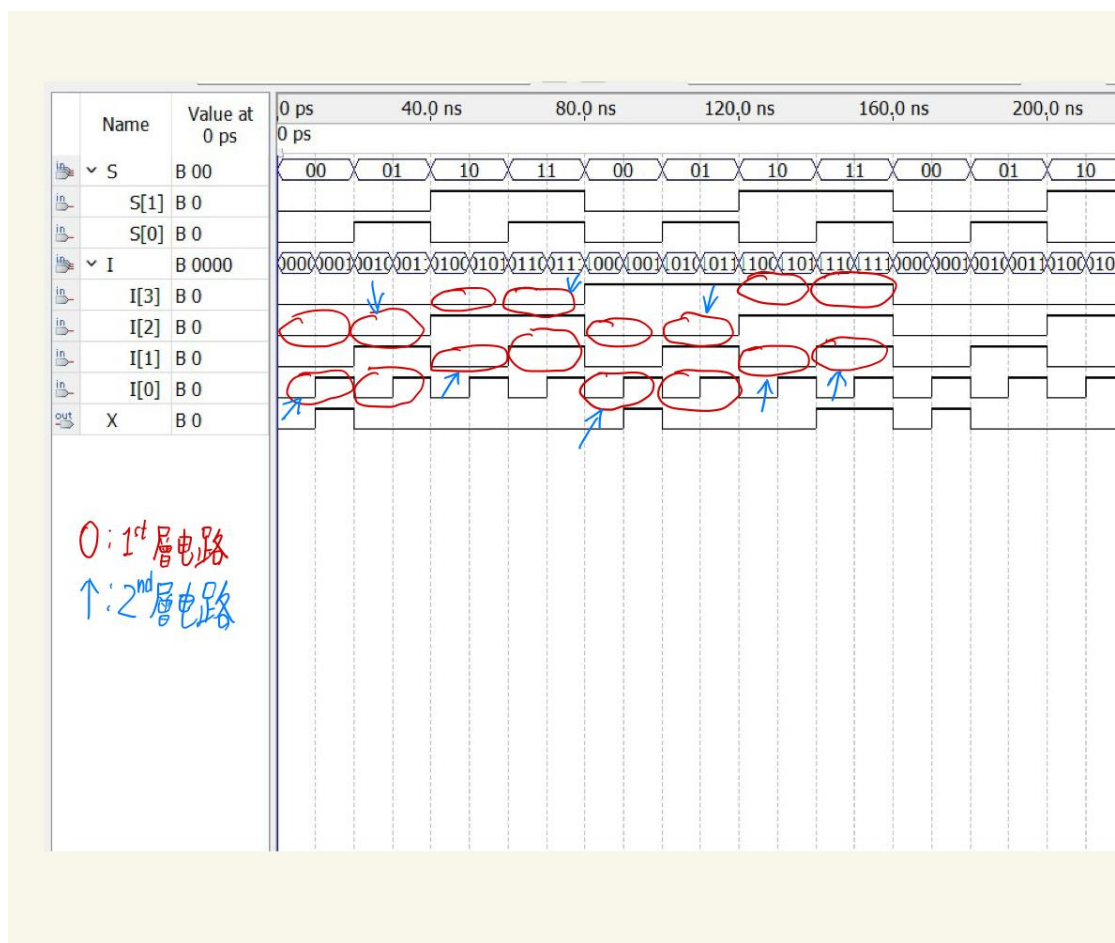
以下為程式表示圖



波形圖



純輸出



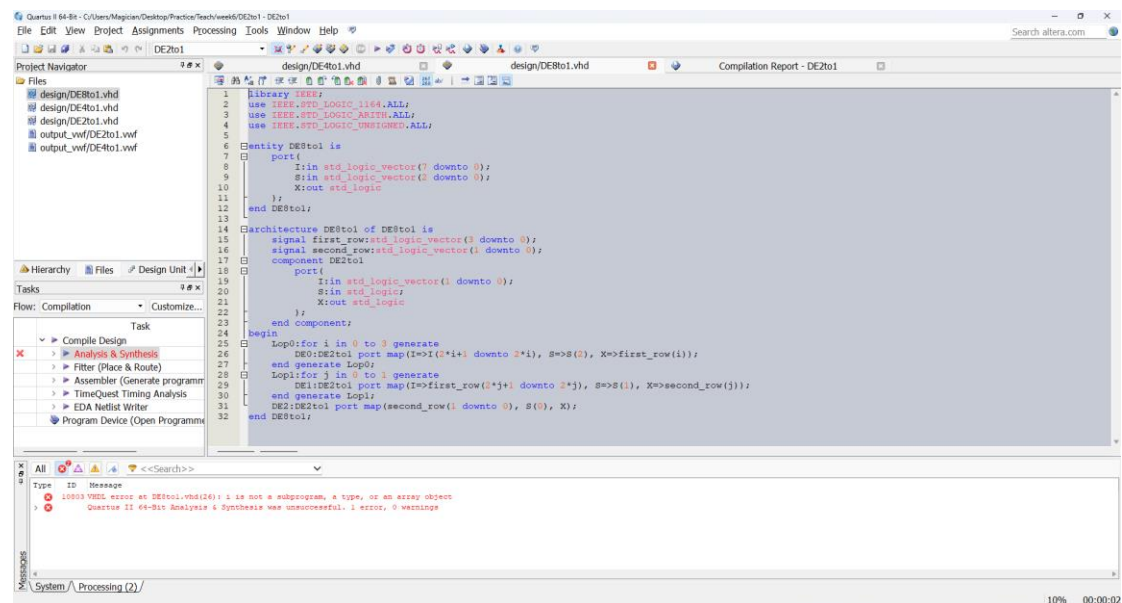
藍色箭頭應與 X 相同

加分題: 用 2 對 1 多工器組成 8 對 1 多工器

a. 使用 port map

問題：小丑...來自 PPT 的詐欺（寫漂亮迴圈根本編譯就不會過），如 i, j 根本不能拿來當

index 遍歷。考慮過是否因為 generate 更換 loop，但網路上資料皆提示 loop 更偏向 assign value，如果行為只是接線，應當使用 generate。那拿 i 先以 generic 的形式存在跳過整數問題，但不可以編譯：)



程式碼

```

1  library IEEE;
2  use IEEE.STD_LOGIC_1164.ALL;
3  use IEEE.STD_LOGIC_ARITH.ALL;
4  use IEEE.STD_LOGIC_UNSIGNED.ALL;
5
6  entity DE8to1 is
7      port(
8          I:in std_logic_vector(7 downto 0);
9          S:in std_logic_vector(2 downto 0);
10         X:out std_logic
11     );
12 end DE8to1;
13
14 architecture DE8to1 of DE8to1 is
15     signal first_row:std_logic_vector(3 downto 0);
16     signal second_row:std_logic_vector(1 downto 0);
17     component DE2to1
18     port(
19         I:in std_logic_vector(1 downto 0);
20         S:in std_logic;
21         X:out std_logic
22     );
23     end component;
24 begin
25     DE0:DE2to1 port map(I=>I(1 downto 0), S=>S(2), X=>first_row(0));
26     DE1:DE2to1 port map(I=>I(3 downto 2), S=>S(2), X=>first_row(1));
27     DE2:DE2to1 port map(I=>I(5 downto 4), S=>S(2), X=>first_row(2));
28     DE3:DE2to1 port map(I=>I(7 downto 6), S=>S(2), X=>first_row(3));
29
30     DE4:DE2to1 port map(I=>first_row(1 downto 0), S=>S(1), X=>second_row(0));
31     DE5:DE2to1 port map(I=>first_row(3 downto 2), S=>S(1), X=>second_row(1));
32
33     DE6:DE2to1 port map(second_row(1 downto 0), S(0), X);
34 end DE8to1;

```

-- 最左方八項輸出
-- 三層電路需要三個select
-- 最後輸出

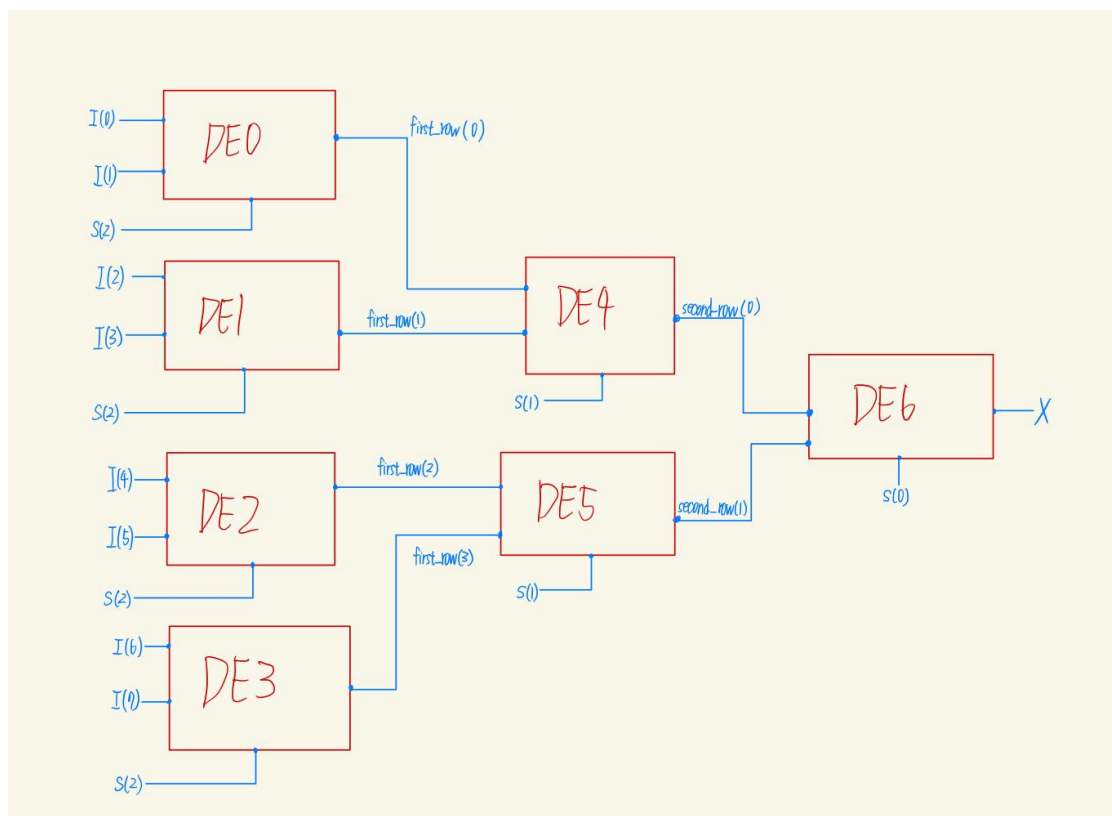
-- 一二層連接signal
-- 二三層連接signal
-- 引用元件

-- 暴力不能解決問題，就是不夠暴力
-- 第一層電路連接

-- 第二層電路連接

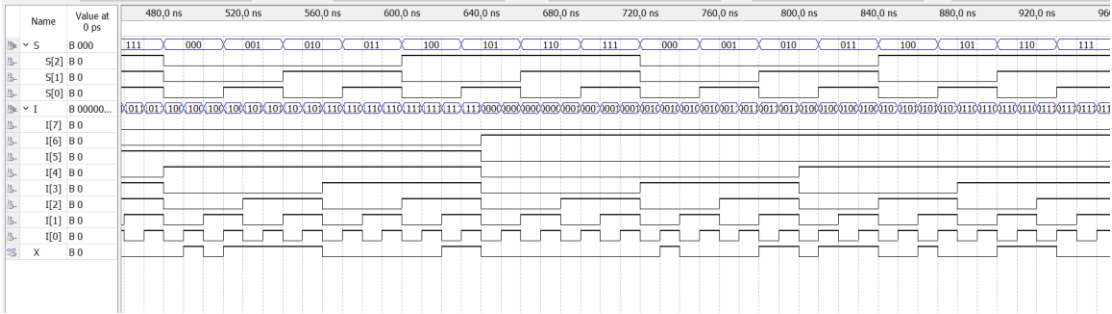
-- 第三層電路連接

以下圖片表示此題及加分題所有程式圖像化示意圖

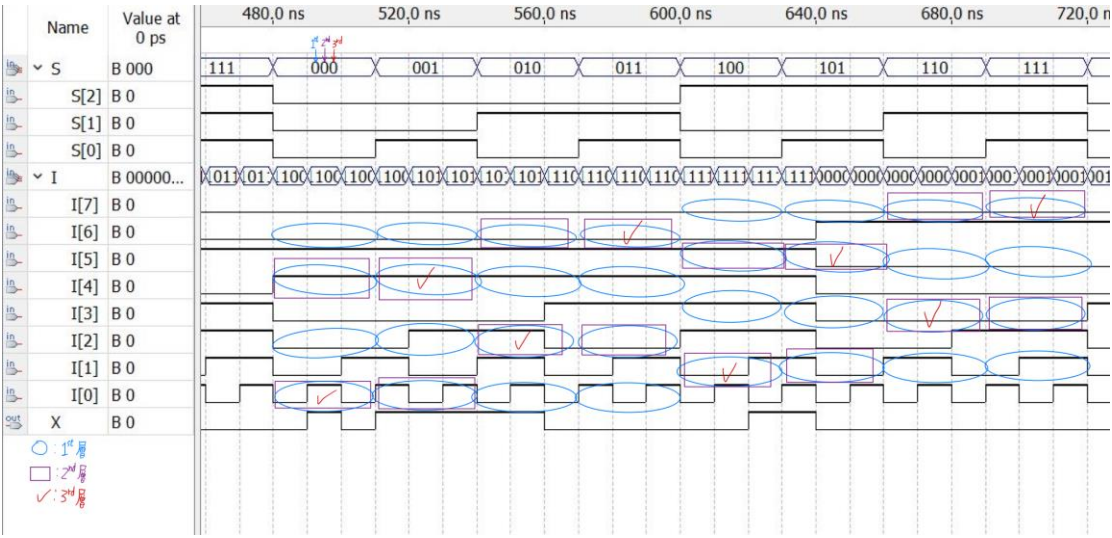


波形圖

純波形輸出



以下對波形圖的解釋



b.使用 function

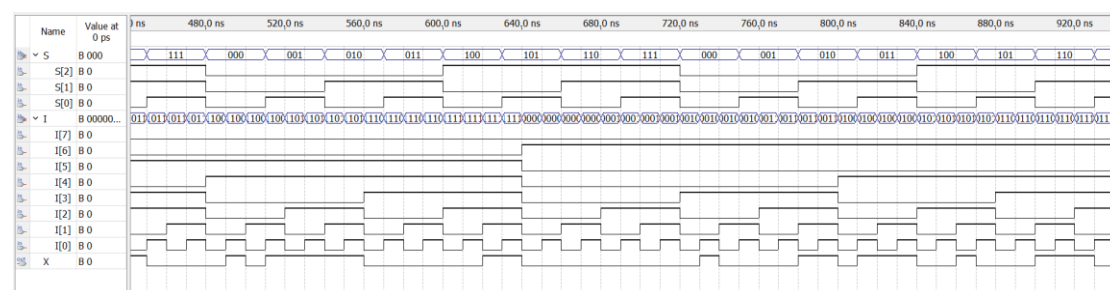
程式碼

```
1  library IEEE;
2  use IEEE.STD_LOGIC_1164.ALL;
3  use IEEE.STD_LOGIC_ARITH.ALL;
4  use IEEE.STD_LOGIC_UNSIGNED.ALL;
5
6  entity DE8to1_func is
7  port(
8      I:in std_logic_vector(7 downto 0);
9      S:in std_logic_vector(2 downto 0);
10     X:out std_logic
11 );
12     function DE2to1 (I:std_logic_vector(1 downto 0); S:std_logic) return std_logic is
13     begin
14         if(S = '1') then -- 依照規定，1 輸出下位
15             return I(1);
16         elsif (S = '0') then -- 依照規定，0 輸出上位
17             return I(0);
18         else -- 為滿足一定有return value
19             return '0';
20         end if;
21     end DE2to1;
22 end DE8to1_func;
23
24 architecture DE8to1_func of DE8to1_func is
25     signal first_row:std_logic_vector(3 downto 0); -- 一二層連接signal
26     signal second_row:std_logic_vector(1 downto 0); -- 二三層連接signal
27 begin
28     first_row(0) <= DE2to1(I(1 downto 0), S(2)); -- 將return value 以 function 方式得到並正確連接輸出
29     first_row(1) <= DE2to1(I(3 downto 2), S(2));
30     first_row(2) <= DE2to1(I(5 downto 4), S(2));
31     first_row(3) <= DE2to1(I(7 downto 6), S(2));
32
33     second_row(0) <= DE2to1(first_row(1 downto 0), S(1)); -- 第二層電路
34     second_row(1) <= DE2to1(first_row(3 downto 2), S(1));
35
36     X <= DE2to1(second_row(1 downto 0), S(0)); -- 第三層電路，接上輸出
37 end DE8to1_func;
```

以上程式碼區塊與 a 有很大相似，電路圖可參考 a 部分

波形圖

以下波型圖與 a 相同（不同的檔案，取相同 ns 的結果），顯示結果相同，符合預想，解釋可參考 a 部分



c.使用 procedure

程式碼

小插曲：procedure 如果參數不輸入 type，默認以 variable 接收，需以 := 接受 assign

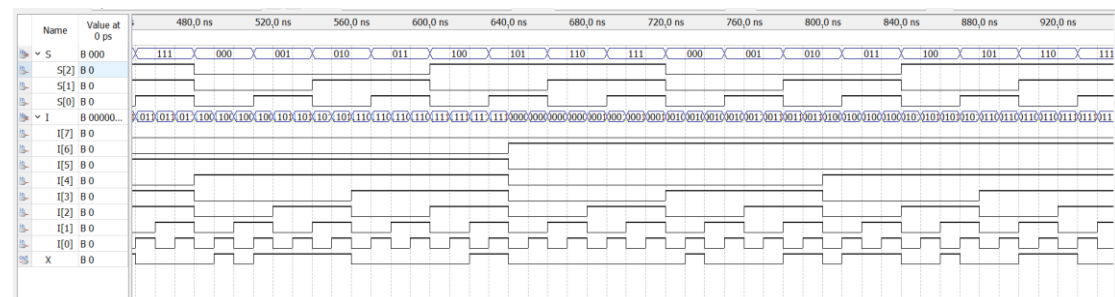
Value，故 signal 在 <= 表 assign 的情況下為必寫。

```
1  library IEEE;
2  use IEEE.STD_LOGIC_1164.ALL;
3  use IEEE.STD_LOGIC_ARITH.ALL;
4  use IEEE.STD_LOGIC_UNSIGNED.ALL;
5
6  entity DE8to1_proc is
7  port(
8      I:in std_logic_vector(7 downto 0);
9      S:in std_logic_vector(2 downto 0);
10     X:out std_logic
11 );
12 procedure DE2to1 (signal I:in std_logic_vector(1 downto 0); signal S:in std_logic; signal X:out std_logic) is
13 begin
14     if(S = '1') then -- 依照規定，1 輸出下位
15         X <= I(1); -- 直接對內容更改，而非回傳
16     elsif (S = '0') then -- 依照規定，0 輸出上位
17         X <= I(0);
18     end if;
19 end DE2to1;
20 end DE8to1_proc;
21
22 architecture DE8to1_proc of DE8to1_proc is
23     signal first_row:std_logic_vector(3 downto 0); -- 一二層連接signal
24     signal second_row:std_logic_vector(1 downto 0); -- 二三層連接signal
25 begin
26     DE2to1 (I(1 downto 0), S(2), first_row(0)); -- 第一層電路，不須接收return value，已經將輸出接至first_row
27     DE2to1 (I(3 downto 2), S(2), first_row(1));
28     DE2to1 (I(5 downto 4), S(2), first_row(2));
29     DE2to1 (I(7 downto 6), S(2), first_row(3));
30
31     DE2to1 (first_row(1 downto 0), S(1), second_row(0)); -- 第二層電路，不須接收return value，已經將輸出接至second_row
32     DE2to1 (first_row(3 downto 2), S(1), second_row(1));
33
34     DE2to1 (second_row(1 downto 0), S(0), X); -- 第三層電路，接上輸出
35 end DE8to1_proc;
```

以上程式碼區塊與 a 有很大相似，電路圖可參考 a 部分

波形圖

以下波型圖與 a 相同（不同的檔案，取相同 ns 的結果），顯示結果相同，符合預想，解釋可參考 a 部分



心得

在此次實驗中，本人更認知到錯誤只有自己拿頭去撞才會知道原資料上寫得有多少充滿美好的幻想。更了解到自己上網查閱更多資料的重要性。

且在經歷此次過程後，我深刻體驗到學過都是有用的，證實了資料結構中提及的——凡是有意義的程式碼，接必須滿足可使用紙筆追蹤且為有限狀態的事實。慢慢一步步照著程式的腳步，我學到了更多指令中想傳達的意思。

雖然如同教授於題目講解所述，“希望”可以使用 `for generate`、`if generate` 描述電路，但經歷三小時嘗試的我了解到這...似乎是過於美好的理想，種種現實表示要碼我得換種寫法，要碼放棄高階描述使用暴力，希望有機會可以更深入的了解如何撰寫高階描述而非全靠蠻力硬幹。