

# 邏輯系統實驗

## Lab 6 Verilog – Sequential Design

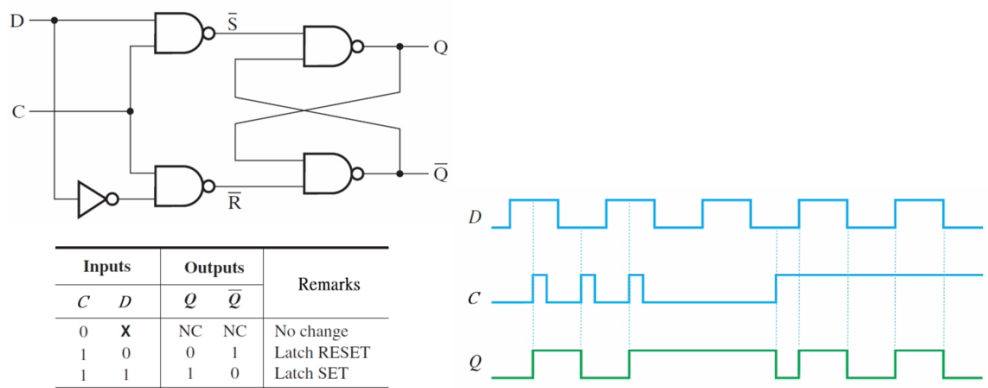
2022/04/07

第 5 組	
組員姓名	學號
林珮玉	E24084096
廖本恩	E24102179
蘇冠誠	E24084143

# 實作題(一): Gated D Latch

## 1. 簡述題目：

利用 Gate Level 撰寫 Gated D Latch，並查看波形圖是否正確。



## 2. 實現方式：

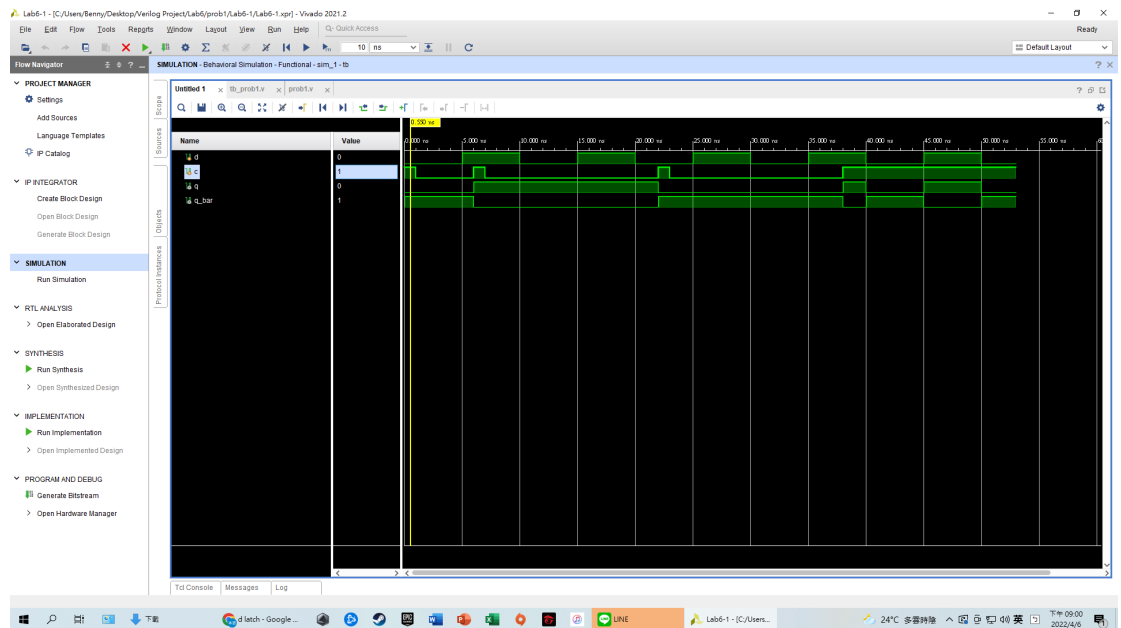
```
prob1.v
module problem1 (input d, input c, output q, output q_bar);

wire s_bar,r_bar,d_bar;

not(d_bar,d);
nand(s_bar,d,c);
nand(r_bar,c,d_bar);
nand(q,s_bar,q_bar);
nand(q_bar,r_bar,q);

endmodule
```

## 3. 結果呈現

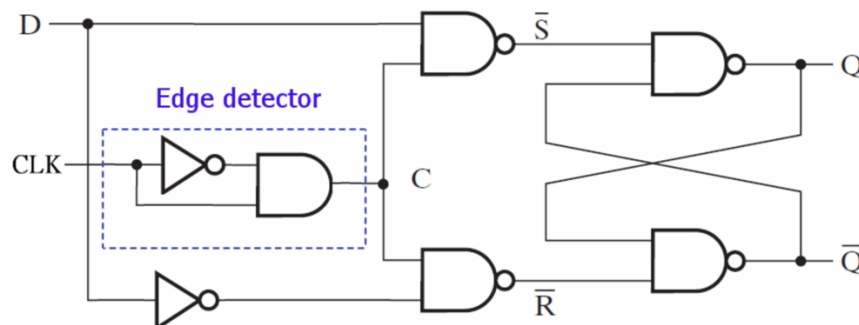


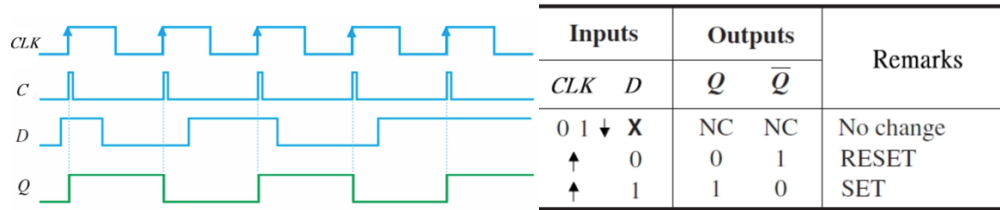
1. 當訊號 c 為 1 時，訊號 q 會和 d 同步；反之，當訊號 c 為 0，則訊號 d 不會影響訊號 q。
2. q 和 q\_bar 兩訊號永遠不會相同，即其中一個為 0，則另一個為 1。

## 實作題(二): D Flip-Flop – using Edge Detector

### 1. 簡述題目：

利用 Edge Detector 的方式實作 Rising Edge trigger 的 D Flip-Flop ( 需使用 Gate Level ) 並查看波形圖確認行為是否正確。





## 2. 實現方式：

### prob2.v

```

`timescale 1ns/1ps
module problem2 (input d, input clk, output q, output q_bar);

wire c,s_bar,r_bar,d_bar,clk_bar;

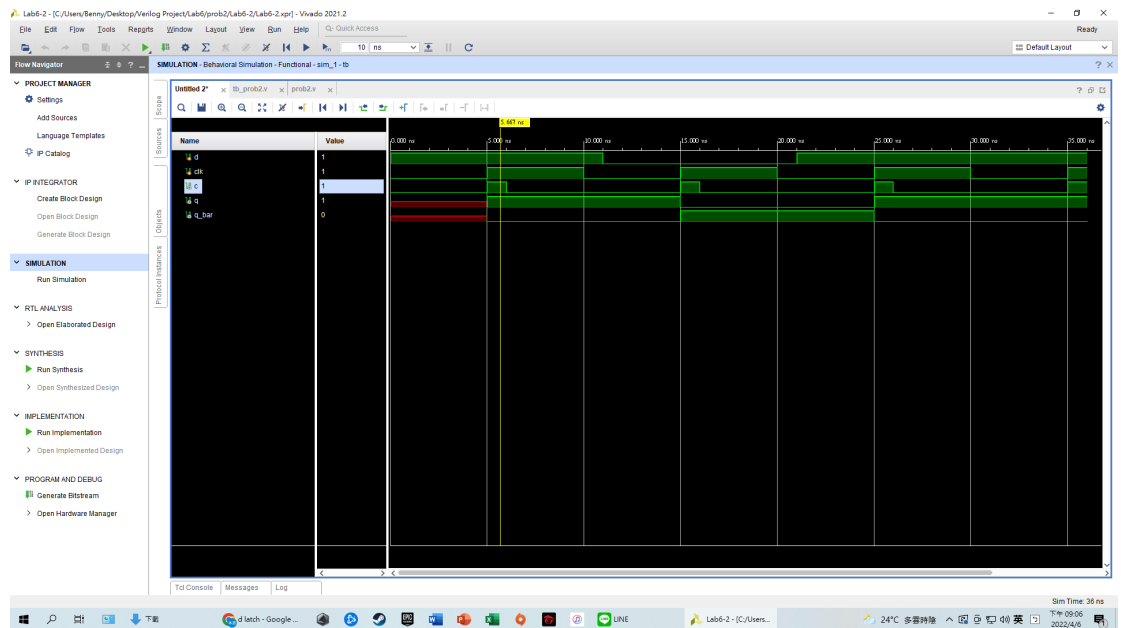
not #(0,1) (clk_bar,clk);
and(c,clk,clk_bar);

not(d_bar,d);
nand(s_bar,d,c);
nand(r_bar,c,d_bar);

nand(q,s_bar,q_bar);
nand(q_bar,r_bar,q);
endmodule

```

## 3. 結果分析：

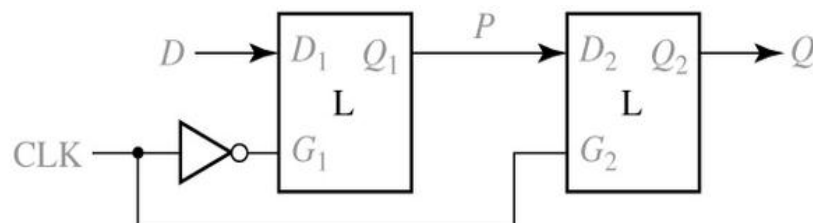


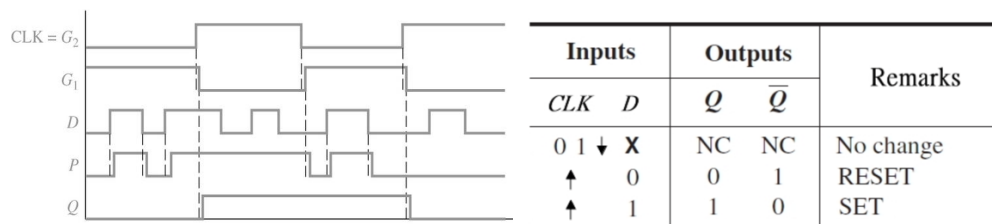
接續實作一，把訊號  $c$  改由邊緣觸發的  $clk$  驅動，因為  $clk\_bar$  需要經過 NOT 閘，所以  $clk\_bar$  會比較晚從 1 變為 0，因此有短暫的時間訊號數值同為 1。因此，由上圖驗證可知，兩訊號經過 AND 閘後的輸出訊號  $c$  在該延遲期間為 1，而其餘期間則因為輸入的兩訊號之一為 0 導致輸出的訊號  $c$  為 0。

## 實作題(三): D Flip-Flop – using Master Slave D Latch

### 1. 簡述題目：

利用 Master Slave D Latch 的方式實作 Rising Edge trigger 的 D Flip-Flop (需使用 Gate Level) 並查看波形圖確認行為是否正確。





## 2. 實現方式：

### **latch.v**

```
module latch (input d, input g, output q, output q_bar);

wire s_bar, r_bar, d_bar;

not(d_bar, d);
nand(s_bar, d, g);
nand(r_bar, g, d_bar);
nand(q, s_bar, q_bar);
nand(q_bar, r_bar, q);

endmodule
```

### **prob3.v**

```
module problem3 (input d, input clk, output q, output q_bar);

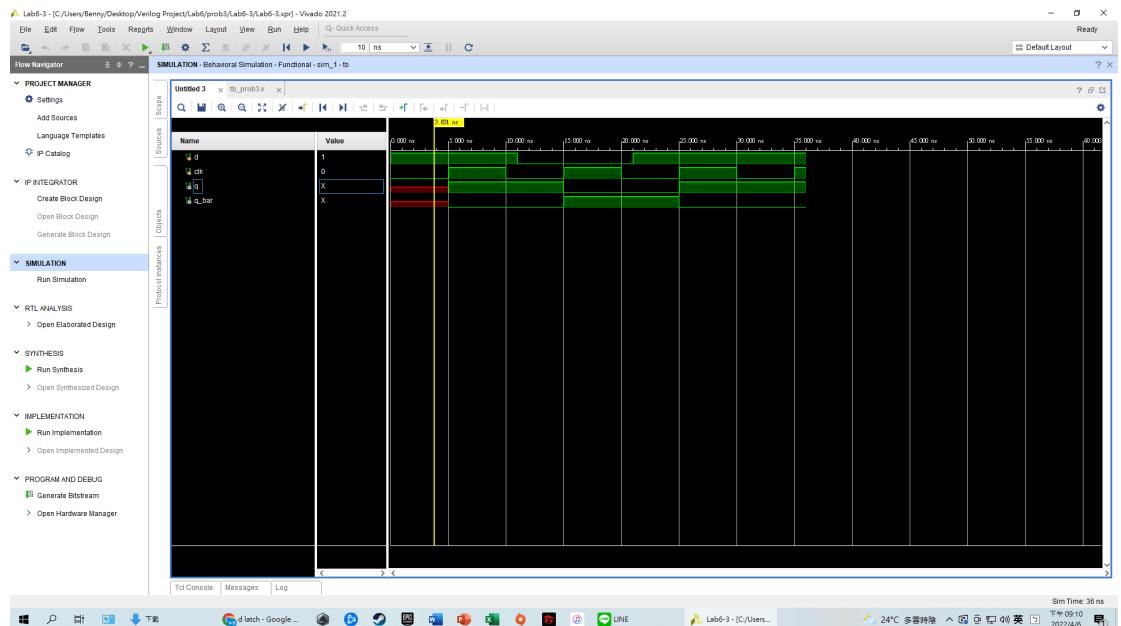
wire clk_bar, p, p_bar;

not(clk_bar, clk);

latch a(d, clk_bar, p, p_bar);
latch b(p, clk, q, q_bar);

endmodule
```

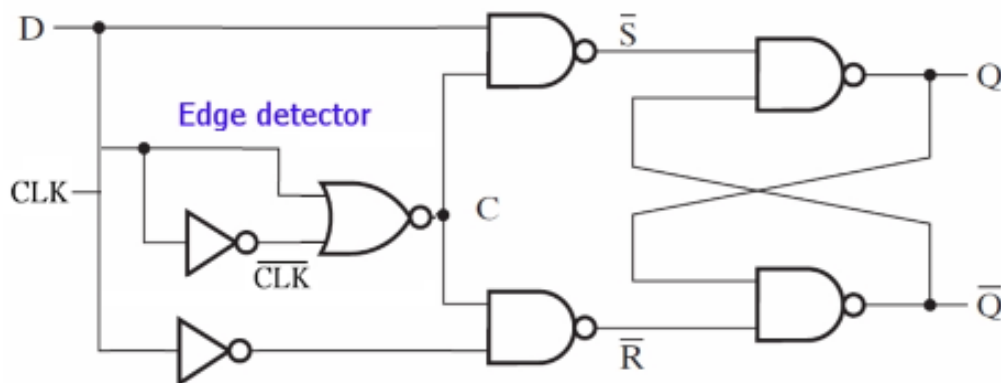
## 3. 結果分析：

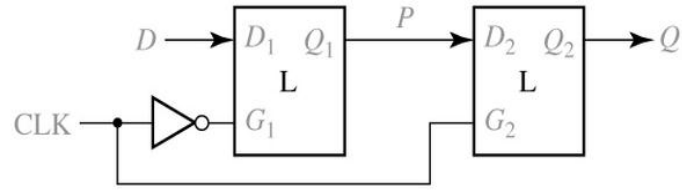


這裡使用的兩個 latch 來自實作一。在訊號  $clk$  為 0 時，訊號  $d$  會持續被存進第一個 latch (即訊號  $p$ )；第二個 latch 為了使訊號  $q$  不被訊號  $p$  的變動影響，input 為值為 0 的  $clk$  以及第一個 latch 的輸出  $p$ ，可將 input 訊號  $d$  隔絕，進而將訊號  $q$  和訊號  $p$  同步。最終得出的結果是，訊號  $q$  僅會記錄最近一次訊號  $clk$  上升瞬間 (前) 訊號  $d$  的值，達到 Master Slave D Latch 的目的。

## 思考：

思考如何使用兩個 D Latch 實作 Falling Edge trigger 的 D Flip-Flop 並畫出電路圖 (如投影片 24 頁的圖) 及解釋其原理





#### prob4.v

```
`timescale 1ns/1ps
module problem2 (input d, input clk, output q, output q_bar);

wire c,s_bar,r_bar,d_bar,clk_bar;

not #(0,1) (clk_bar,clk);
xor(c,clk,clk_bar);

not(d_bar,d);
nand(s_bar,d,c);
nand(r_bar,c,d_bar);

nand(q,s_bar,q_bar);
nand(q_bar,r_bar,q);
endmodule
```

跟實作二不一樣的地方是使用 Falling Edge Detector，因此把原本的 AND 改成使用 XOR。當第一個 Latch 在 CLK 輸入為 1 時，將資料存入  $Q_1$ ，當 CLK 輸入轉為 0 時 (Falling Edge)， $Q_1$  會進入第二個 Latch 並改變 output  $Q$ 。

## 心得

### 組員一 林珮玉 E24084096

這週因為期中考比較多，沒有時間預習本週實驗，再加上早上有期中考比較晚進實驗室，整體來說大概一小時後才進入狀況。當我還在理解的時候，我的夥伴已經把三個實作題寫完了，超厲害！所以這次我基本上沒什麼貢獻QQ不過值得慶幸的是原本不了 Vivado 的筆電後來成功安裝 Modelsim 了，為了把背後



的原理搞懂，我有好好地將組員打的 code 再用 Modelsim 自己跑一次，最後結報也是由我負責。果然收穫超多，尤其在寫結果分析的時候，不僅讓我更了解 Sequential Design、Latch 與 Flip-Flop 的關係，也更清楚知道為什麼要用邊緣偵測。還有一點覺得這堂課很厲害的是，大一讀邏輯系統很多不懂的概念竟然可以透過一次實驗完全解惑！很期待下次實驗，希望跟前兩次實驗一樣可以提早下課、一樣順利！

## **組員二 廖本恩 E24102179**

本次實驗做的是新的 sequential circuit，在上課時原本以為 sequantial circuit 只是把前一次的 output 當成 input 一起考慮這麼簡單，沒想到還可以完成如 latch 跟 filp-flop 這種類似控制開關的效果。而這次的程式碼雖然簡單，但是無法像之前一樣一次看出結果會是什麼，體認到了模擬的重要性。

## **組員三 蘇冠誠 E24084143**

這次實驗在經過預習後做起來簡單了很多，同時我也想起了 verilog 的模組定義與部分語法，而組中也有人成功把模擬用的應用程式載下來了，讓我有種下次上課會順利的預感。