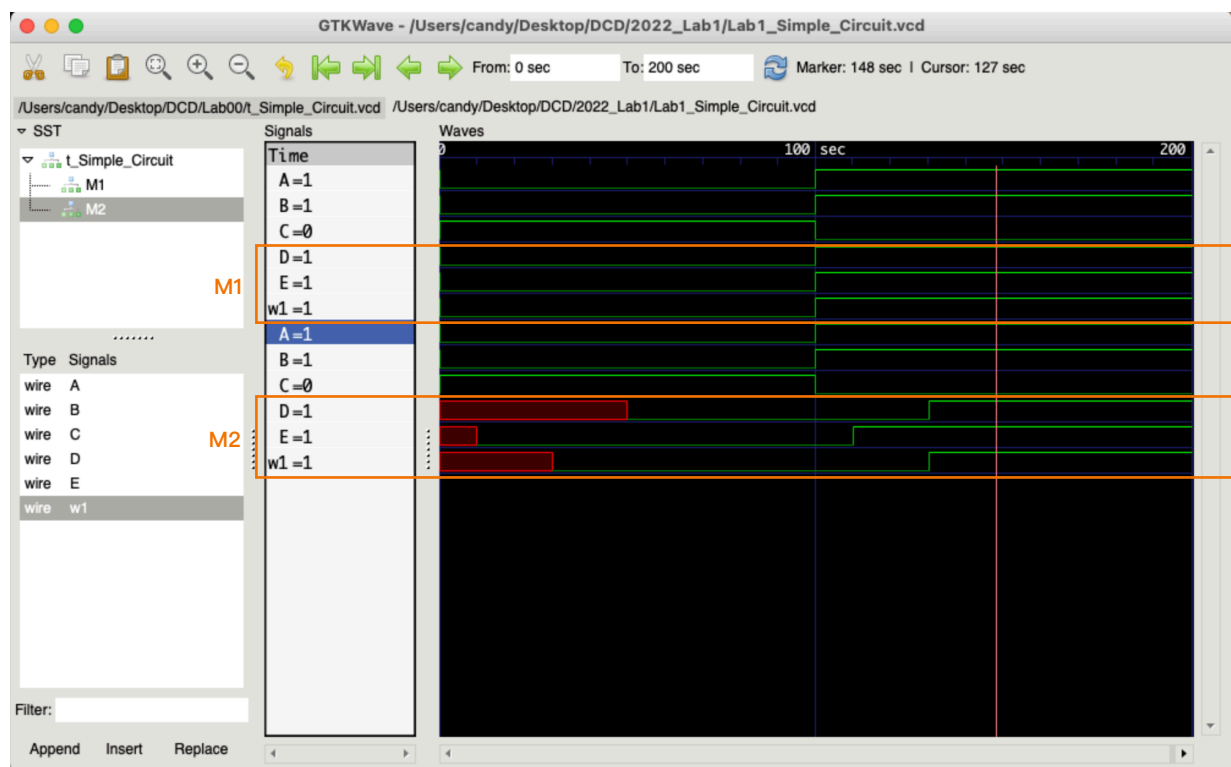


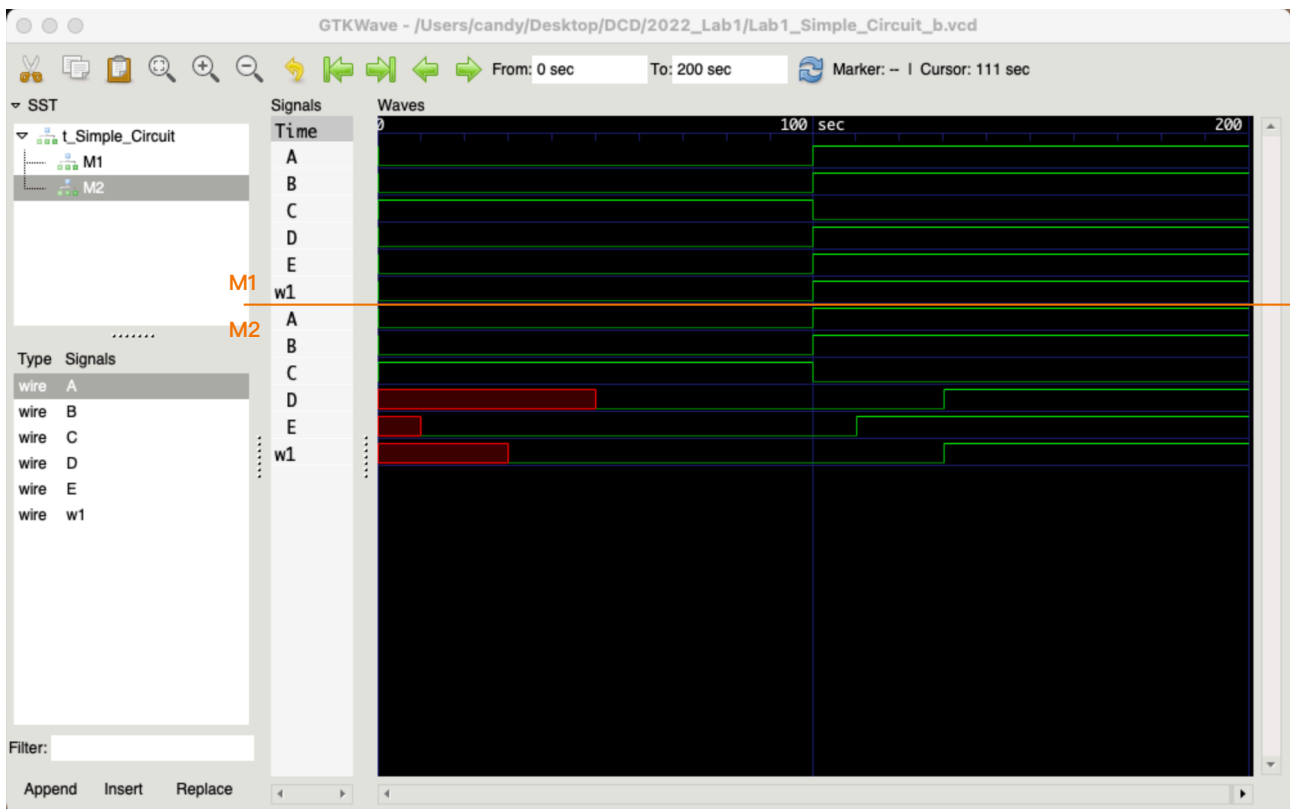
1. 2A(a)之模擬結果波形圖，並說明 Simple_Circuit.v 與 Simple_Circuit_prop_delay.v 之波形圖之間的差異。



因為有設定延遲時間，所以在一開始的時候D、E、w1沒有訊息的產生。又因為G3在輸出D時會需要w1及E的 true or false，所以就算在設定時G3的延遲只有設定為20秒，因為G1(生成w1的 or gate)的延遲有30秒，使得G3的延遲延長為30秒，因此相較於 Simple_Circuit.v 的圖形w1的產生延遲了30秒。(G3並不會受G2影響，因為G3的延遲時間較G2長) 而w1、E則維持原本的延遲秒數，w1(G1)延遲30秒，E(G2)延遲10秒產生。其他沒有設定延遲時間的A、B、C則維持原樣，與原本圖形相同。

```
1 module Simple_Circuit_prop_delay(A, B, C, D, E);
2     output D, E;
3     input A, B, C;
4     wire w1;
5
6     and #(30) G1(w1, A, B);
7     not #(10) G2(E, C);
8     or #(20) G3(D, w1, E);
9 endmodule
```

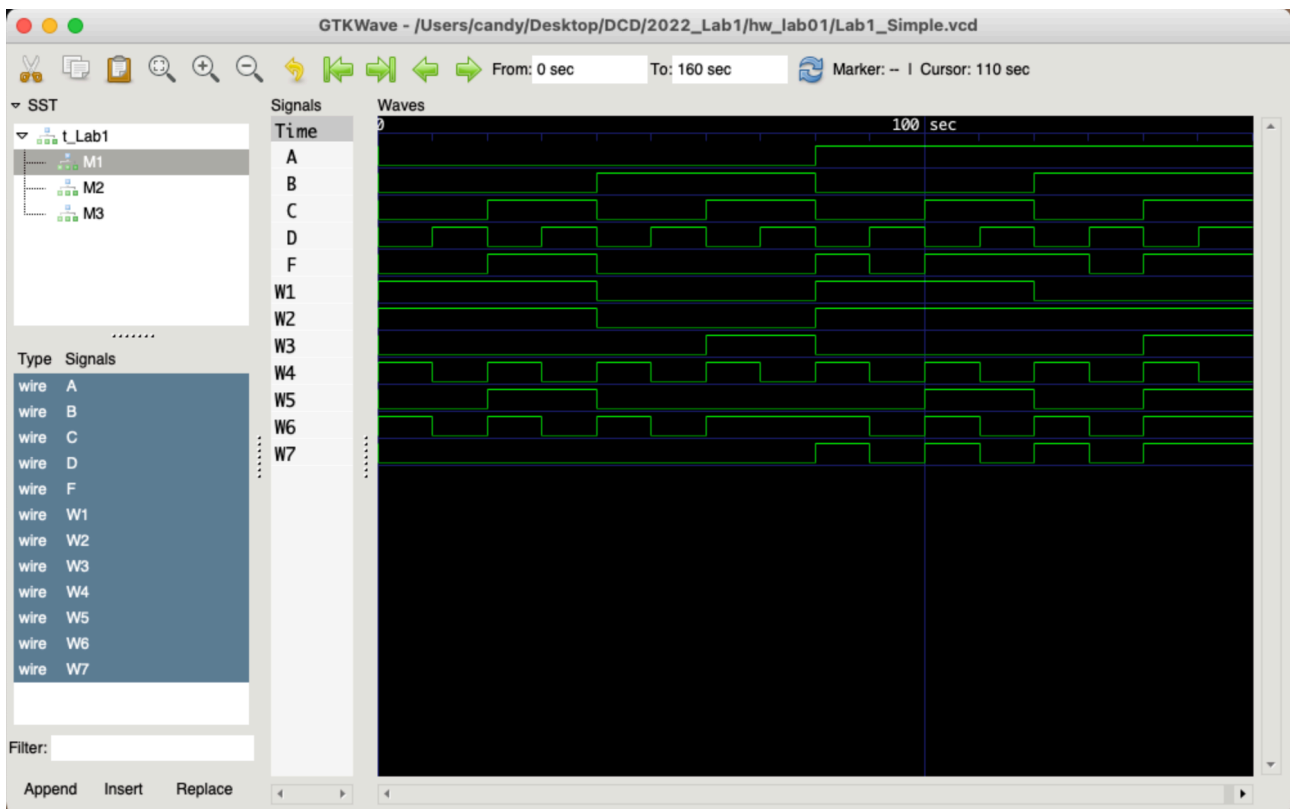
2. 2A(b)之模擬結果波形圖，並說明與2A(a)之波形圖是否有差異及原因。



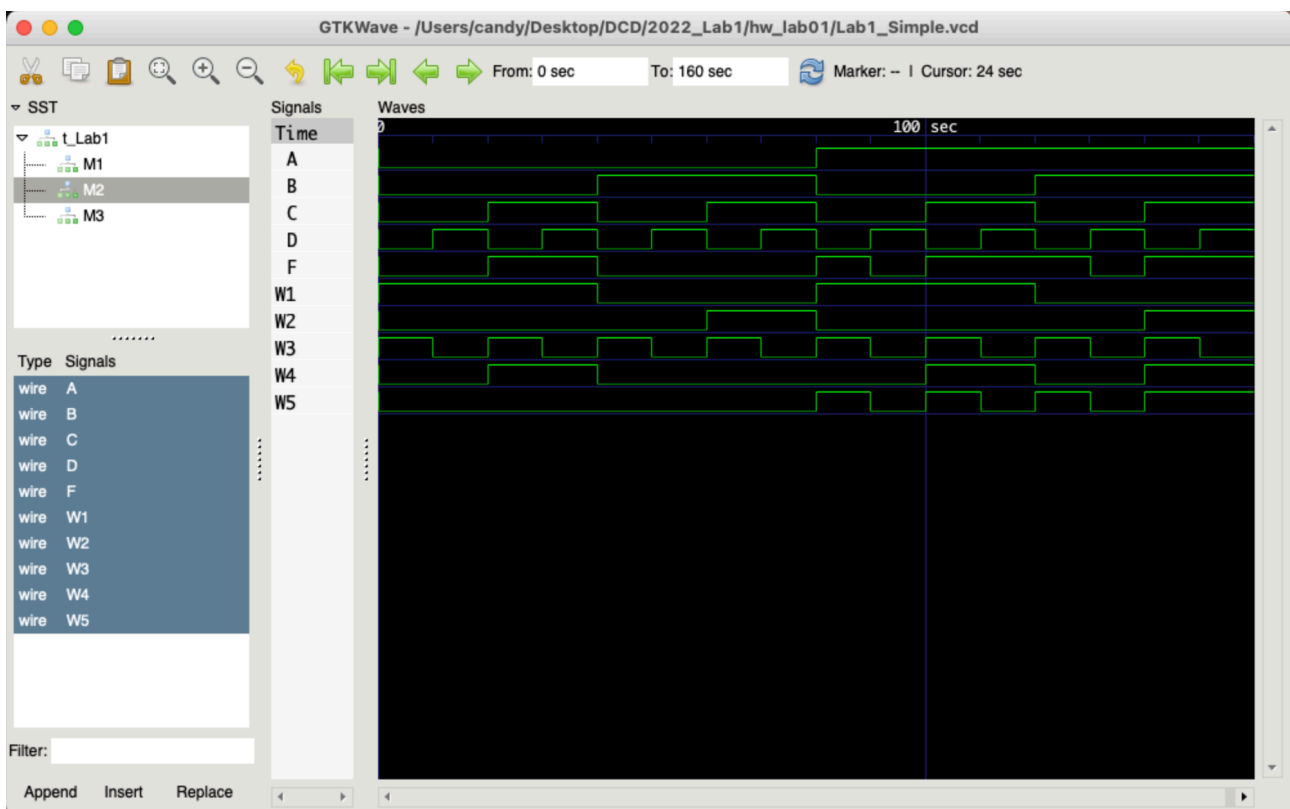
兩者圖形相同，因為只是將宣告順序改變，並沒有更改gate的形式 (ex. or 改成 and)，所以結果圖形相同。

3. 2B(d)之電路模擬結果波形圖，並說明三個電路模組是否正確。

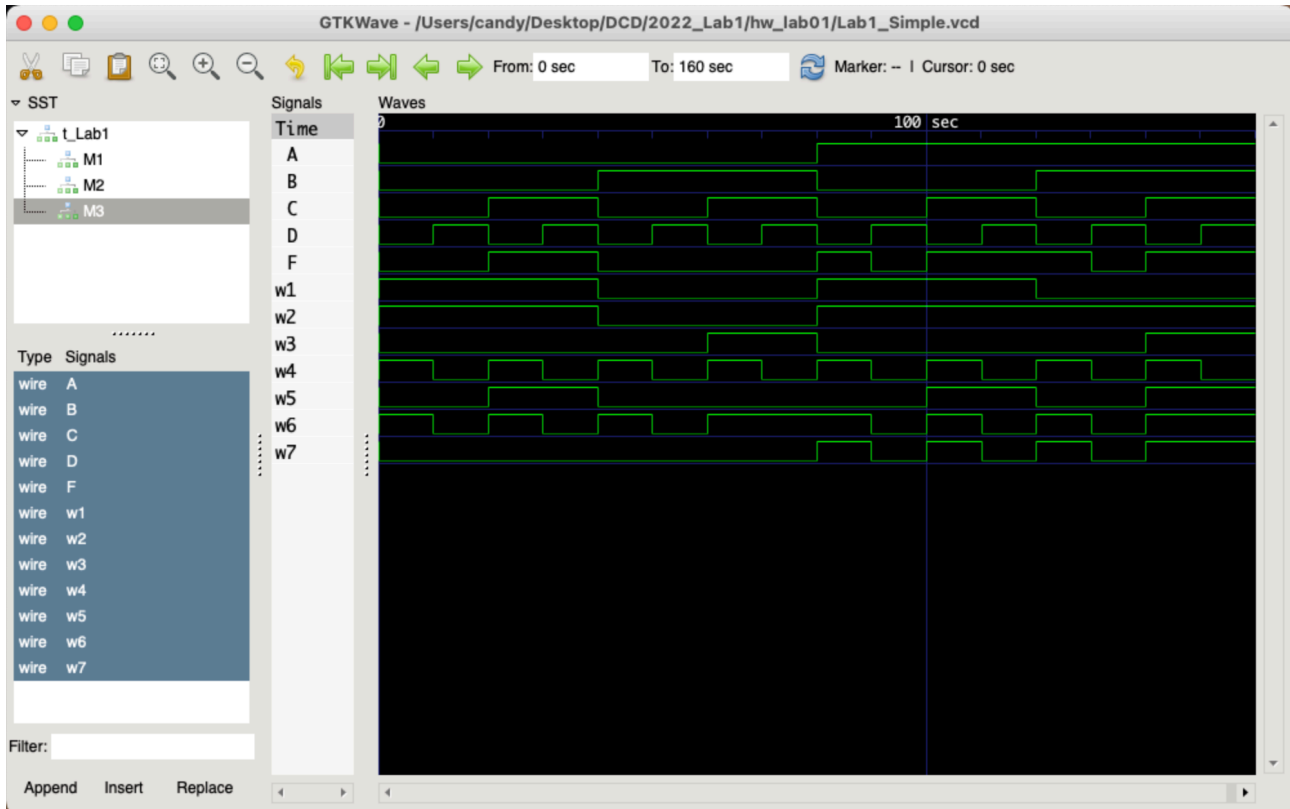
Lab1_dataflow M1



Lab1_gatelevel_UDP M2



Lab1_gatelevel M3

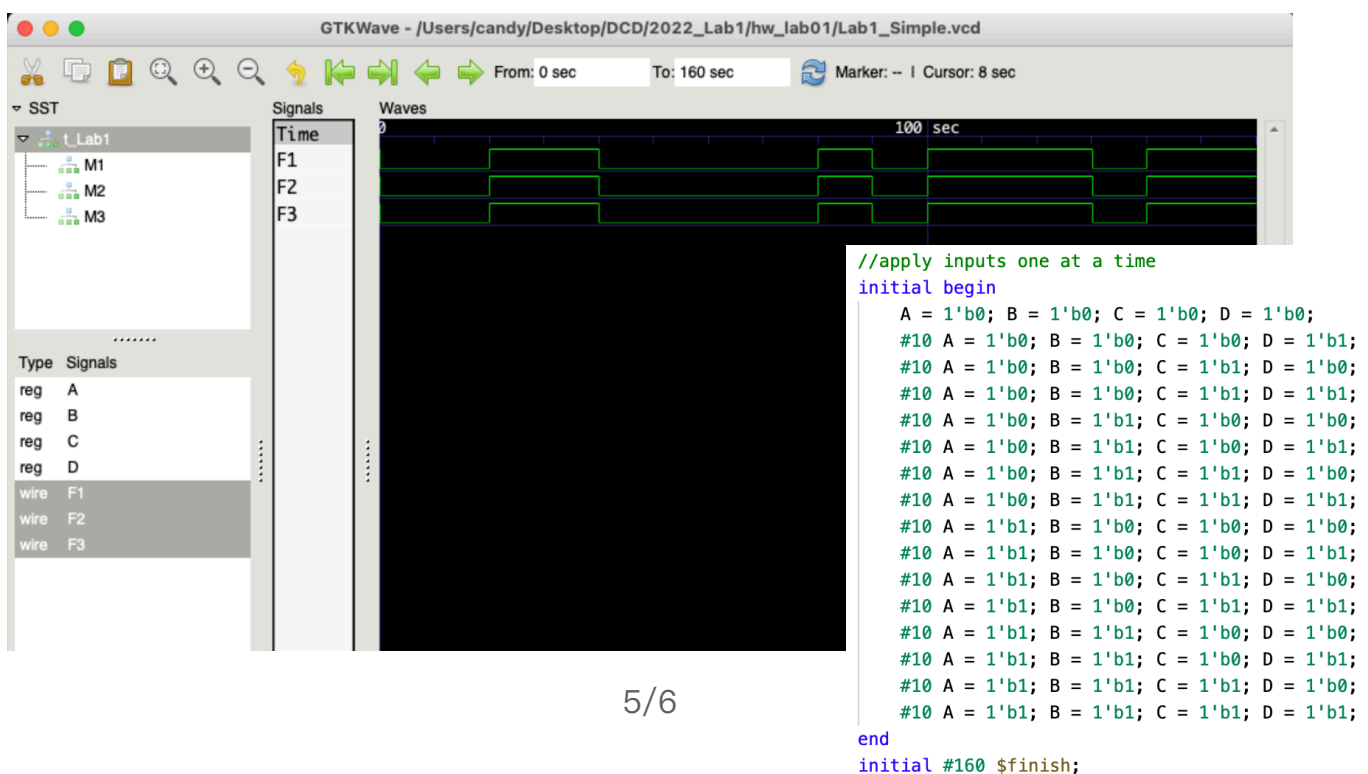


三個電路模組皆為正確。

比較M1、M2、M3三者的圖形在 (A, B, C, D) 初始值相同時 F 是否相同，並與真值表進行對照，確認三者圖形相同。

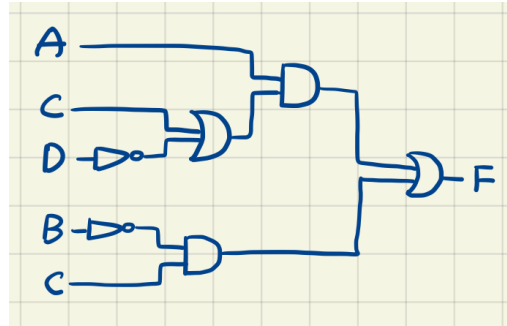
$$F = C(A+B') + A(D'+BC)$$

A	B	C	D	B'	A+B'	C(A+B')	D'	BC	A(D'+BC)	F
0	0	0	0	1	1	0	1	0	0	0
0	0	0	1	1	1	0	0	0	0	0
0	0	1	0	1	1	1	1	0	0	1
0	0	1	1	1	1	1	0	0	0	1
0	1	0	0	0	0	0	1	0	0	0
0	1	0	1	0	0	0	0	0	0	0
0	1	1	0	0	0	0	1	1	0	0
0	1	1	1	0	0	0	0	1	0	0
1	0	0	0	1	1	0	1	0	1	1
1	0	0	1	1	1	0	0	0	0	0
1	0	1	0	1	1	1	1	0	1	1
1	0	1	1	1	1	1	0	0	0	1
1	1	0	0	0	1	0	1	0	1	1
1	1	0	1	0	1	0	0	0	0	0
1	1	1	0	0	1	1	1	1	1	1
1	1	1	1	0	1	1	0	1	1	1



4. 請判斷圖1之電路是否為該函式 gate input counts 最少之實作？若是，請說明之；若否，則請推導出此函式 gate input counts 最少的布林代數式，寫出 gate input count 數值，並以 AND、OR、NOT 邏輯閘畫出其電路圖。

$$\begin{aligned} F &= C(A+B') + A(D'+BC) \\ &= AC + B'C + AD' + ABC \\ &= AC(1+B) + B'C + AD' \\ &= A(C+D') + B'C \\ \text{GIC} &= 10 \end{aligned}$$



5. 心得與感想、及遭遇到的問題或困難。

因為自己本身使用的是Mac，所以從最一開始的安裝iverilog及gtkwave就和大家不一樣，要運用到平常不熟悉的terminal並在其中輸入指令，後面的檔案生成方式似乎也不盡相同。因此在面對到不會的問題時，需要花更多的心力去解決、研究它，與以往詢問同學便能解決的問題處理方式相差甚遠。但也因為如此，經由這次的作業我對於iverilog的使用方式及gtkwave的運作有了更深入的了解，也有信心在日後同學詢問時能夠協助他們解決問題。此外，正因為途中遇到了許多問題，在圖形跑出來得那一剎那更有成就感。