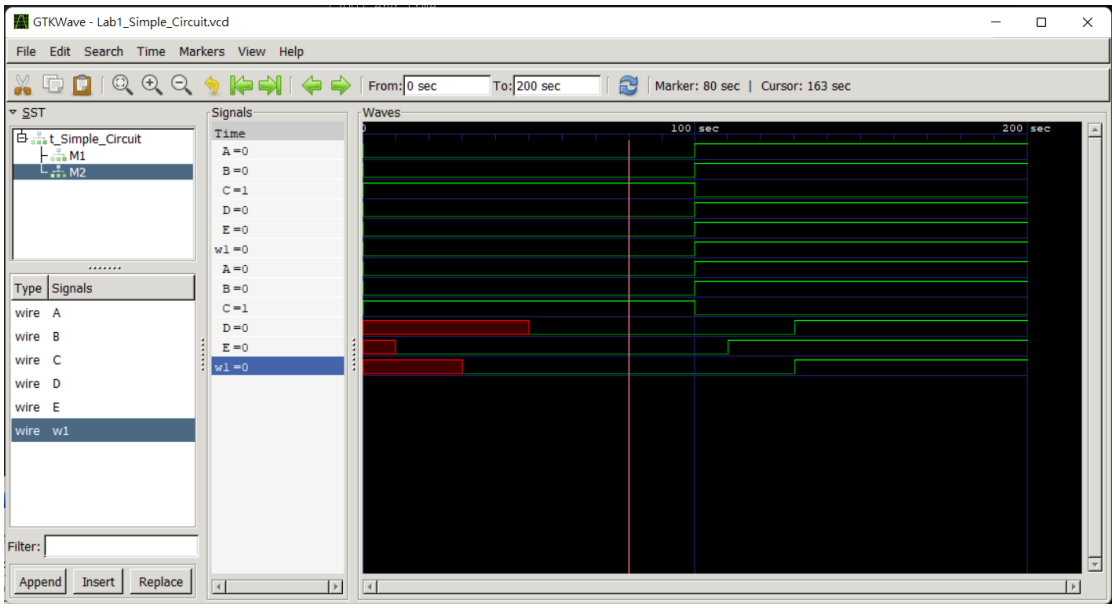
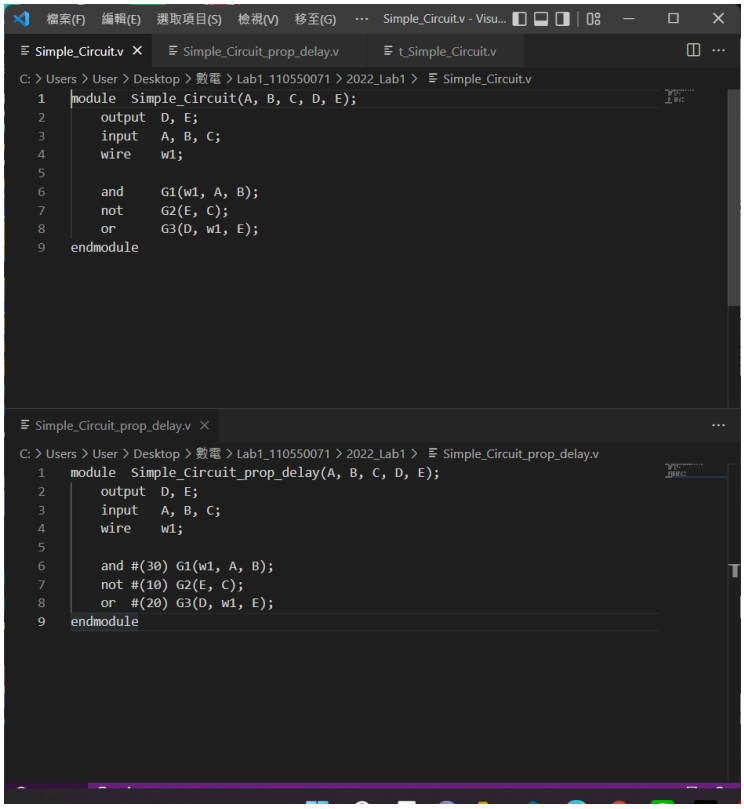
Lab 1

A-a 上半為 Simple_Circuit; 下半為 Simple_Circuit_Prop_Delay



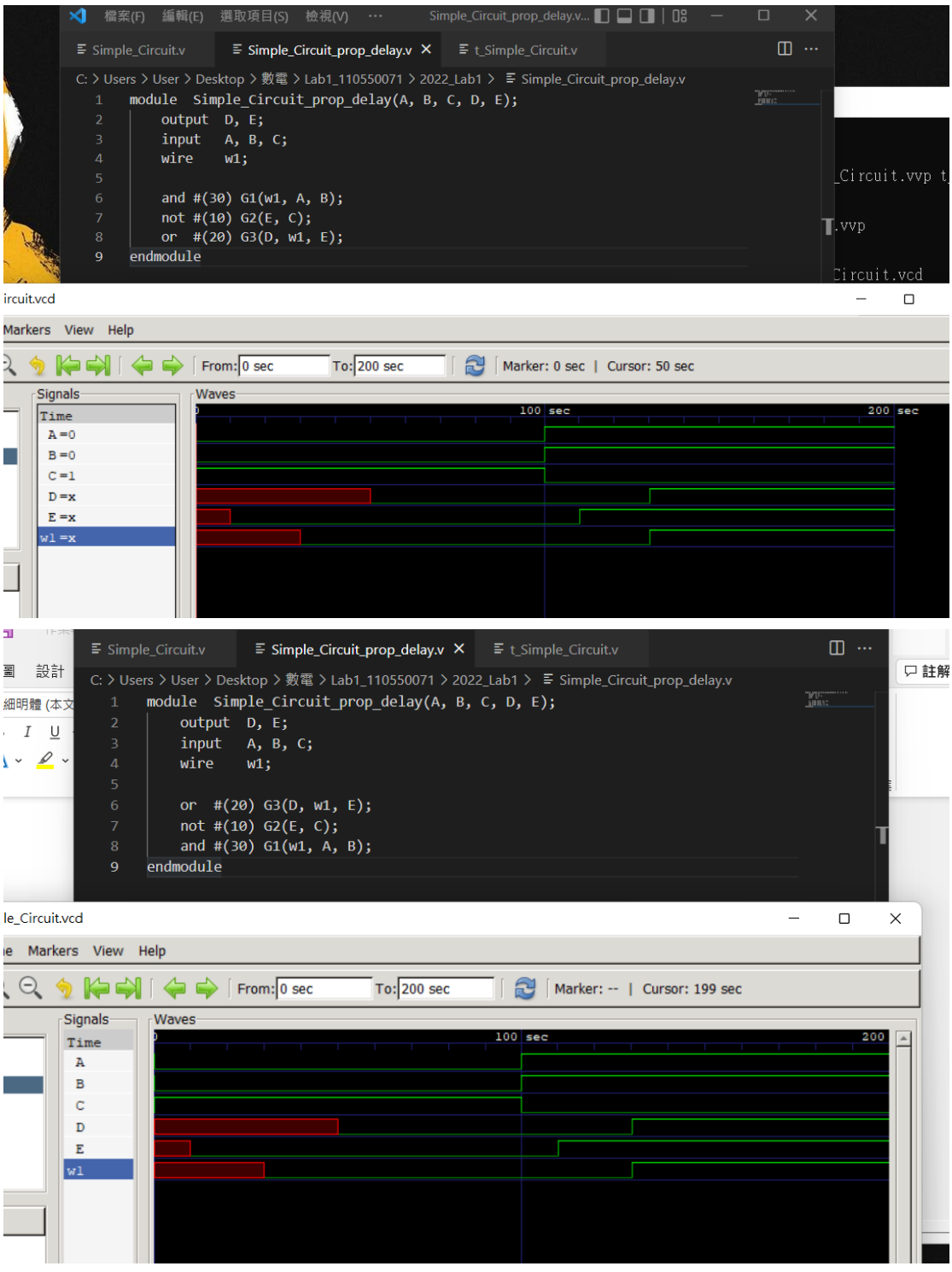


觀察:

兩組模組僅差有無延遲，上半為無下半為有延遲。

觀察下半有延遲的部分，從第六行開始輸出 w1 會延遲 30 秒，第七航輸出 E 會延遲 10 秒，第八行輸入為 w1 及 E 需要等待兩者皆出現才能開始運作，因此需等待 max(30,10)秒。而紅色 don't care 部分，w1 30 秒、E 10 秒、D 30+20 秒。

A-b　And / Or 互換　　結果相同



And / Or 互換後波形圖結果相同。

交換後 w1,E,D 延遲時間仍為 30,10,30 秒，紅色 don’t care 部分，一樣為 50,10,30 秒，與互換前相同。互換前後因為電路樣式不會改變，因此波形圖也不會改變。

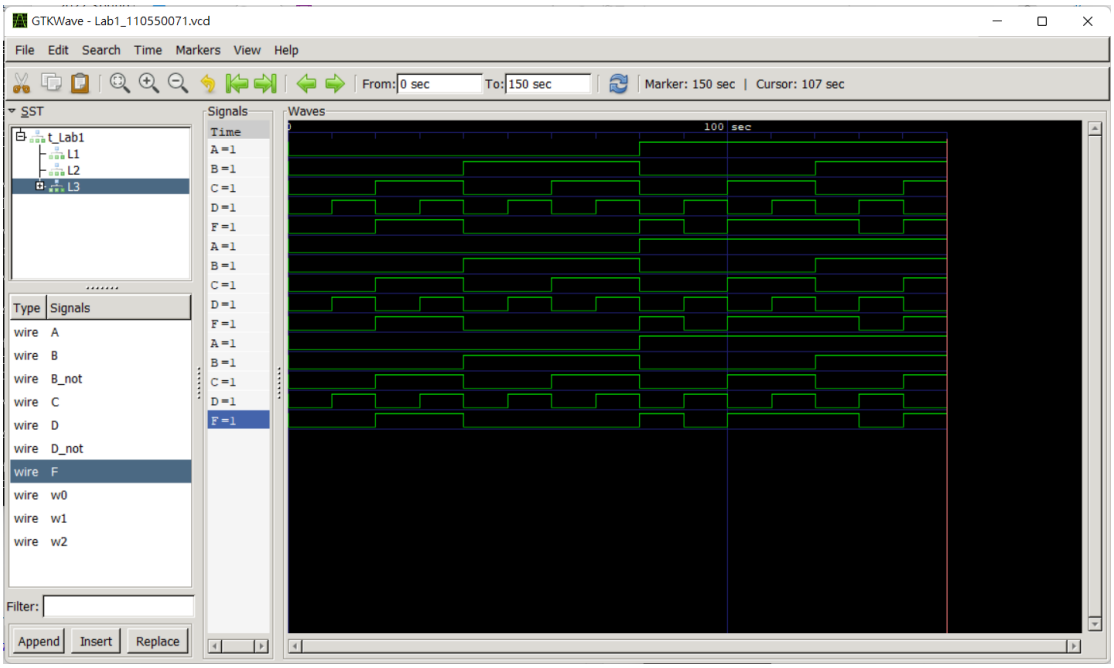
B

第一組是 gatelevel

第二組是 dataflow

第三組是 gatelevel_UDP

每組波形各有輸入源 ABCD 及輸出 F

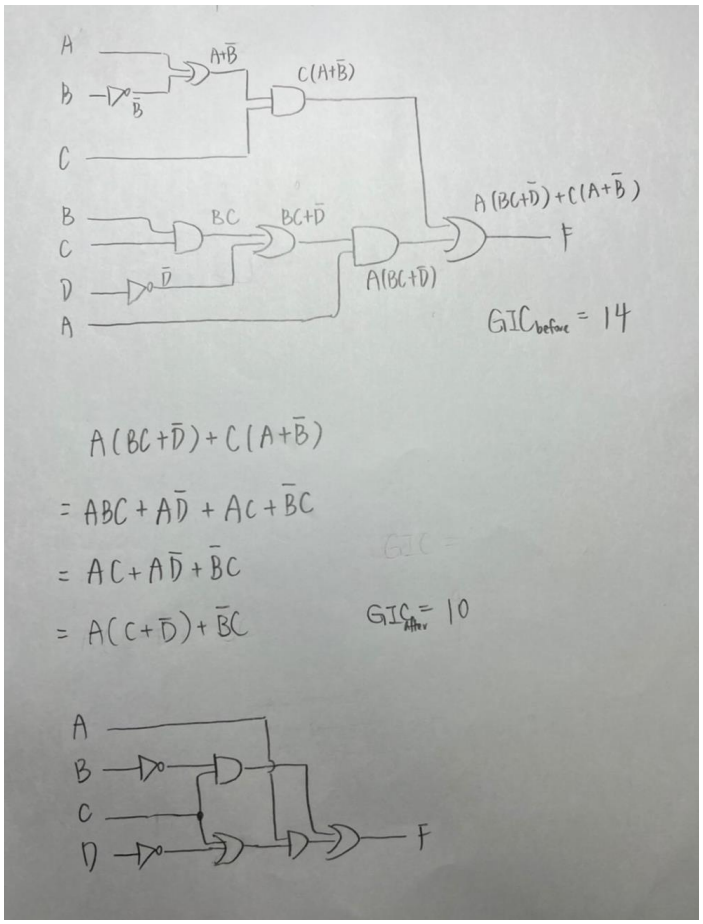


真值表

```
initial begin
    A = 0; B = 0; C = 0; D = 0;         // F = 0;
    #10 A = 0; B = 0; C = 0; D = 1;     // F = 0;
    #10 A = 0; B = 0; C = 1; D = 0;     // F = 1;
    #10 A = 0; B = 0; C = 1; D = 1;     // F = 1;
    #10 A = 0; B = 1; C = 0; D = 0;     // F = 0;
    #10 A = 0; B = 1; C = 0; D = 1;     // F = 0;
    #10 A = 0; B = 1; C = 1; D = 0;     // F = 0;
    #10 A = 0; B = 1; C = 1; D = 1;     // F = 0;
    #10 A = 1; B = 0; C = 0; D = 0;     // F = 1;
    #10 A = 1; B = 0; C = 0; D = 1;     // F = 0;
    #10 A = 1; B = 0; C = 1; D = 0;     // F = 1;
    #10 A = 1; B = 0; C = 1; D = 1;     // F = 1;
    #10 A = 1; B = 1; C = 0; D = 0;     // F = 1;
    #10 A = 1; B = 1; C = 0; D = 1;     // F = 0;
    #10 A = 1; B = 1; C = 1; D = 0;     // F = 1;
    #10 A = 1; B = 1; C = 1; D = 1;     // F = 1;
end
```

三種模組(gatelevel, dataflow, udp)得到的結果(波型)相同，真值表結果也與電路相同，所以結果應該是對的。

圖一之 GIC 為 11
經推導後得最小 GIC 為 10



心得感想困難問題:
從小沒有幾次碰到設計電路的機會，操作起來十分生疏，再來是把程式和電路結合在一起更是困難。花了一整個晚上來實作這次的 Lab，一直撞牆甚至還出現 don’t care，幸好最後都有順利解決，很多小細節都要格外小心。