數位邏輯設計-HW4　　鄭凱文　104062223

HW4_1：

程式內容：

Case:

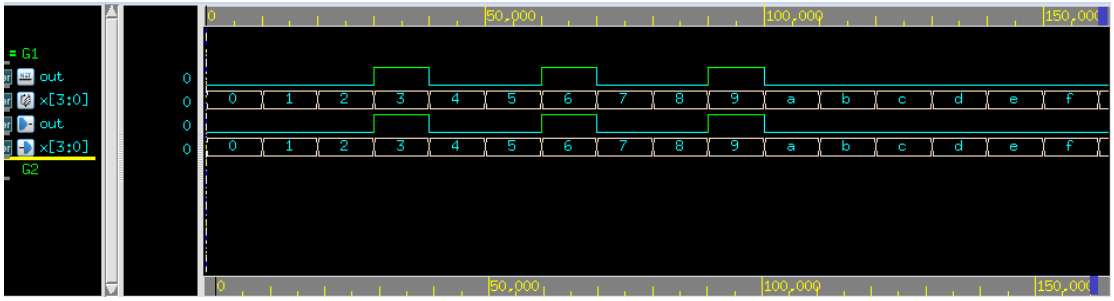
X = 3, 6, 9: Output = 1

Default: Output = 0

Simulation:

```
in = 0000, out = 0
in = 0001, out = 0
in = 0010, out = 0
in = 0011, out = 1
in = 0100, out = 0
in = 0101, out = 0
in = 0110, out = 1
in = 0111, out = 0
in = 1000, out = 0
in = 1001, out = 1
in = 1010, out = 0
in = 1011, out = 0
in = 1100, out = 0
in = 1101, out = 0
in = 1110, out = 0
in = 1111, out = 0
```

波形圖：



問題：無

HW4_2：

程式內容：

兩個 4-bit 的數相加，最多可達 5-bit。

Input: A[3:0], B[3:0], cin(配合之後的規律性而存在，cin 恆=0)

中間連接: c1, c2, c3

Output: cout, sum[3:0]

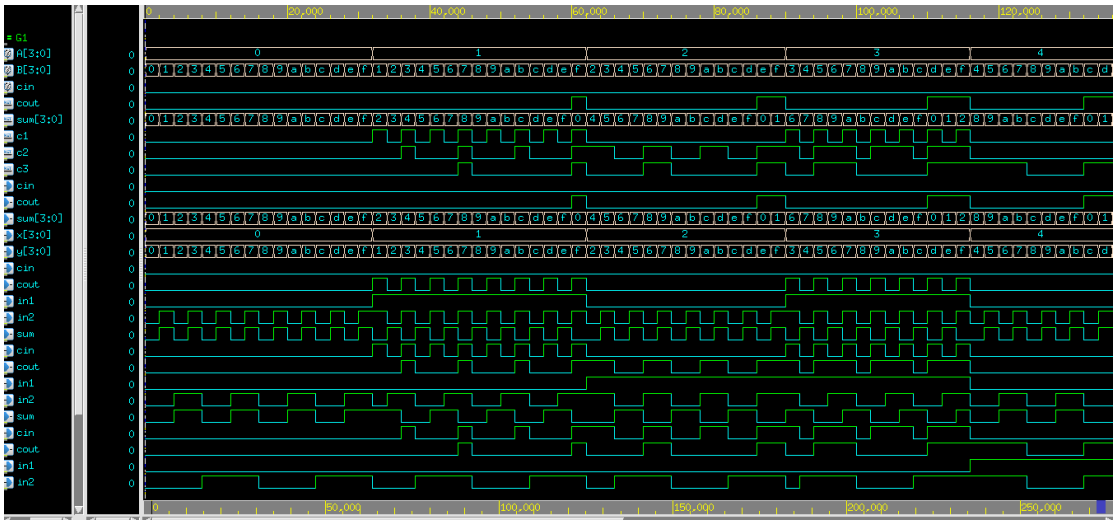
藉由 1-bit 的 full-adder，從低到高每次傳一個 bit 進去，每次得到的進位值則藉由 c1, c2, c3 傳遞給下一個 full-adder，重複 4 遍。最後的進位值給 cout。

```
always @* begin
    {cout,sum} = in1 + in2 +cin;
end
```

Simulation(部分):

```
A=0000 B=0000 cin=0 | cout=0 sum=0000
A=0000 B=0001 cin=0 | cout=0 sum=0001
A=0000 B=0010 cin=0 | cout=0 sum=0010
A=0000 B=0011 cin=0 | cout=0 sum=0011
A=0000 B=0100 cin=0 | cout=0 sum=0100
A=0000 B=0101 cin=0 | cout=0 sum=0101
A=0000 B=0110 cin=0 | cout=0 sum=0110
A=0000 B=0111 cin=0 | cout=0 sum=0111
A=0000 B=1000 cin=0 | cout=0 sum=1000
A=0000 B=1001 cin=0 | cout=0 sum=1001
A=0000 B=1010 cin=0 | cout=0 sum=1010
A=0000 B=1011 cin=0 | cout=0 sum=1011
A=0000 B=1100 cin=0 | cout=0 sum=1100
A=0000 B=1101 cin=0 | cout=0 sum=1101
A=0000 B=1110 cin=0 | cout=0 sum=1110
A=0000 B=1111 cin=0 | cout=0 sum=1111
A=0001 B=0001 cin=0 | cout=0 sum=0010
A=0001 B=0010 cin=0 | cout=0 sum=0011
A=0001 B=0011 cin=0 | cout=0 sum=0100
A=0001 B=0100 cin=0 | cout=0 sum=0101
A=0001 B=0101 cin=0 | cout=0 sum=0110
A=0001 B=0111 cin=0 | cout=0 sum=0111
A=0001 B=1000 cin=0 | cout=0 sum=1000
A=0001 B=1001 cin=0 | cout=0 sum=1001
A=0001 B=1001 cin=0 | cout=0 sum=1010
```

波形圖(部分)：



關於 test pattern 的選擇：

因為 A 和 B 皆為 4 位數，而 A+B=B+A，即前後兩數對調後相加情形一樣。

因此為避免重複，每次 A=A+1 時，B 的起始值從 A 開始。

```
repeat(16) begin
    B = A;      //避免重複，B從A的數值開始
    while(B!=4'b1111) begin
        #20
        $display("A=%b B=%b cin=%b | cout=%b sum=%b", A, B, cin, cout, sum);
        B = B + 4'b0001;
    end
    //以下為B=4'b1111時的情況
    $display("A=%b B=%b cin=%b | cout=%b sum=%b", A, B, cin, cout, sum);
    A = A +4'b0001;
end
```