



*Department of Computer Science and Information Engineering*

*National Cheng Kung University*

# **LAB - 05**

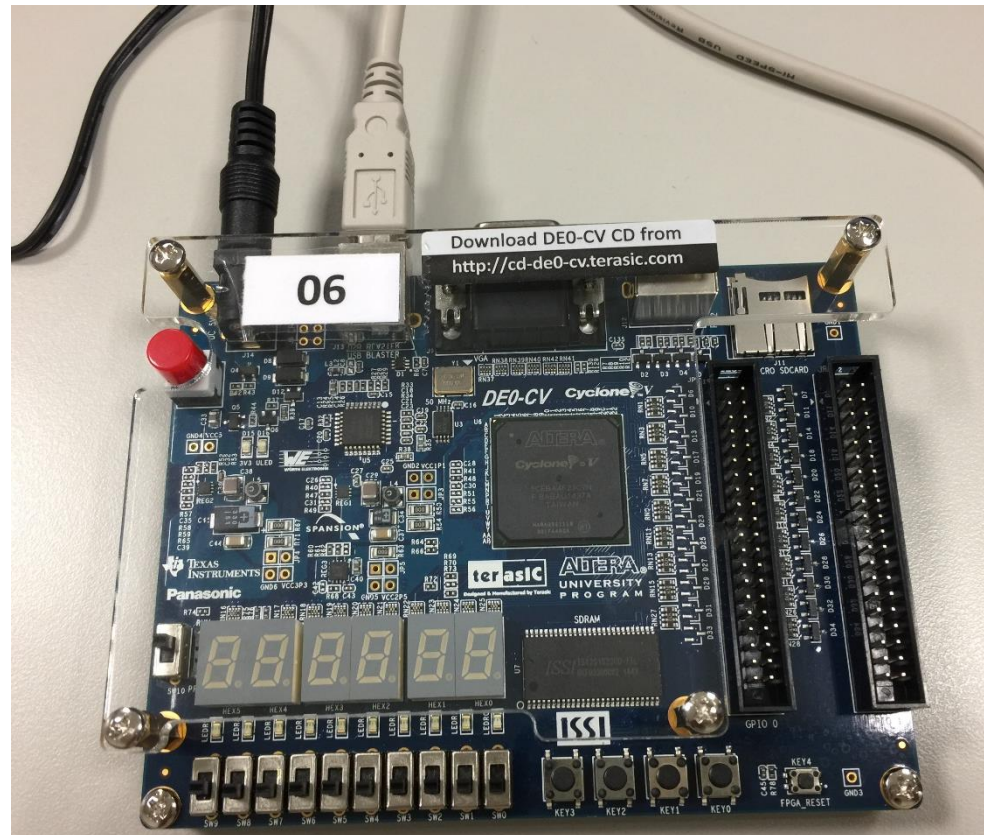
陳培殷老師

國立成功大學 資訊工程系



# Introduction to DE0-CV

電源 USB

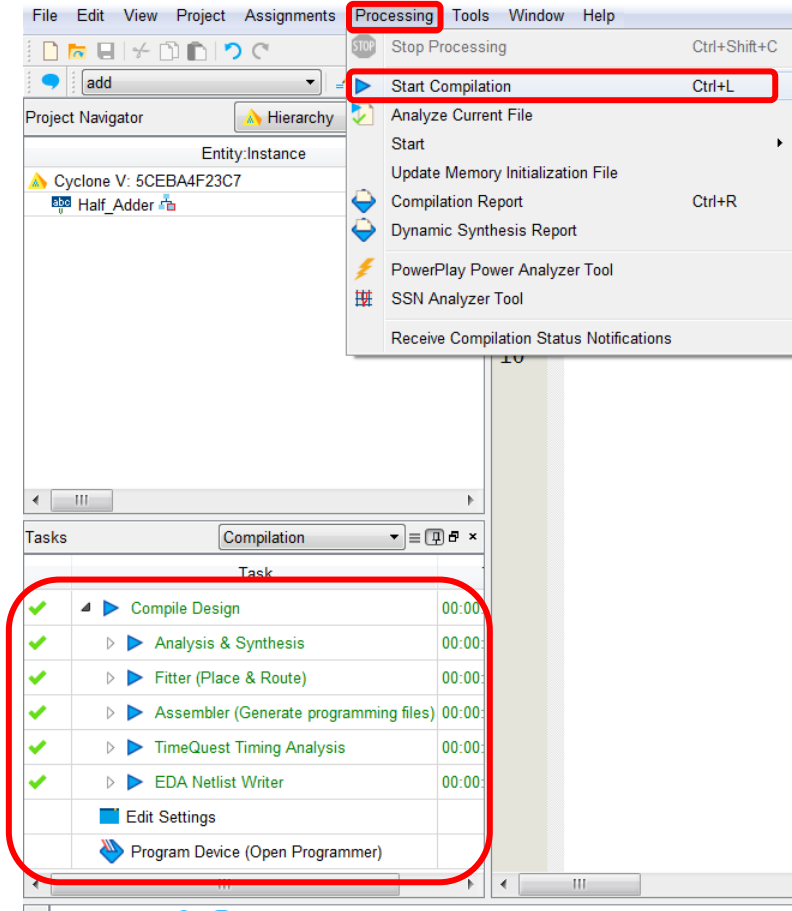


# Programming DE0-CV (1/13)

```
1 module Half_Adder(a, b, sum, carry);  
2  
3 input a,b;  
4 output sum, carry;  
5  
6 and(carry,a,b);  
7 xor(sum,a,b);  
8  
9 endmodule  
10
```

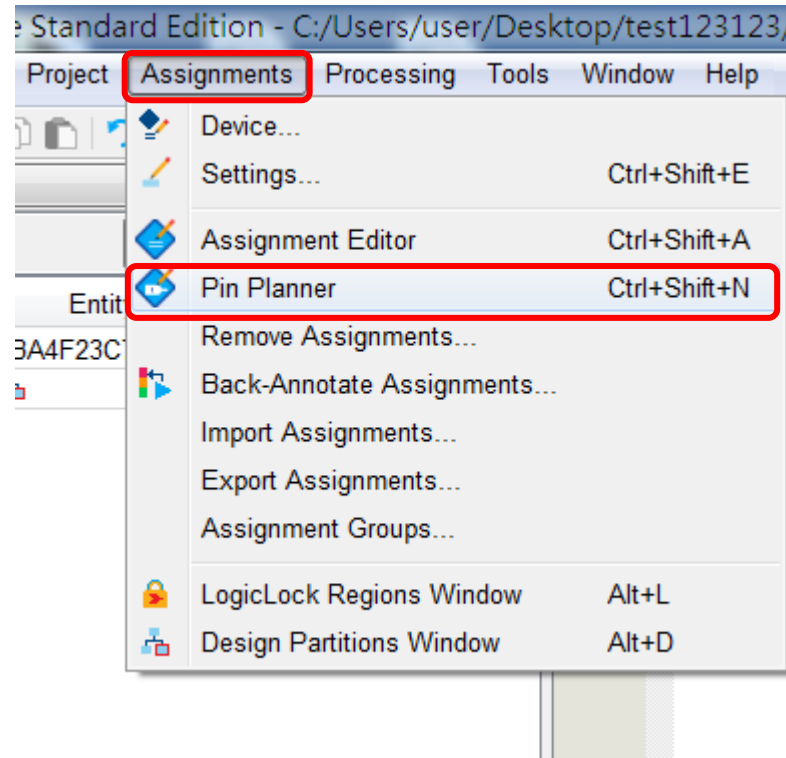
# Programming DE0-CV (2/13)

## ■ Start compilation



# Programming DE0-CV (3/13)

- Open Pin Planner



# Programming DE0-CV (4/13)

- Assign pin location to all inputs and outputs

Pin Planner - C:/Users/user/Desktop/test123123/add - add

Report not available

Top View - Wire Bond  
Cyclone V - 5CEBA4F23C7

Groups Report

Tasks

- Early Pin Planning
- Early Pin Planning...
- Run I/O Assignment Ana...

Named: \* Edit: PIN\_AA2 Filter: Pins: all

Node Name	Direction	Location	I/O Bank	VREF Group	Fitter Location	I/O Standard
a	Input	PIN U13	4A	B4A N0	PIN M16	2.5 V (default)
b	Input	PIN V13	4A	B4A N0	PIN N21	2.5 V (default)
carry	Output	PIN AA1	2A	B2A N0	PIN N16	2.5 V (default)
sum	Output	PIN AA2				
<<new node>>						

Pin Legend





Symbol	Pin Type
○	User I/O
●	User assigne...
●	Fitter assigne...
○	Unbonded pad
○	Reserved pin
E	DEV_OE
n	DIFF_n
p	DIFF_p
n	DIFF_n output
p	DIFF_p output

Double click

IOBANK\_2A Row I/O DIFFIO\_TX\_L16p, DIFFOUT\_L16p, DQ1L  
IOBANK\_3B Column I/O DIFFIO\_TX\_B12n, DIFFOUT\_B12n, DQ2B  
IOBANK\_3B Column I/O DIFFIO\_TX\_B13n, DIFFOUT\_B13n, DQ2B  
IOBANK\_3B Column I/O DIFFIO\_TX\_B16p, DIFFOUT\_B16p, DQ2B  
IOBANK\_3B Column I/O DIFFIO\_TX\_B16n, DIFFOUT\_B16n, DQ2B  
IOBANK\_3B Column I/O DIFFIO\_TX\_B24p, DIFFOUT\_B24p, DQ3B  
IOBANK\_4A Column I/O DIFFIO\_TX\_B28p, DIFFOUT\_B28p  
IOBANK\_4A Column I/O DIFFIO\_TX\_B28n, DIFFOUT\_B28n, DQ4B  
IOBANK\_4A Column I/O DIFFIO\_TX\_B29p, DIFFOUT\_B29p, DQ4B  
IOBANK\_4A Column I/O DIFFIO\_TX\_B36n, DIFFOUT\_B36n, DQ5B

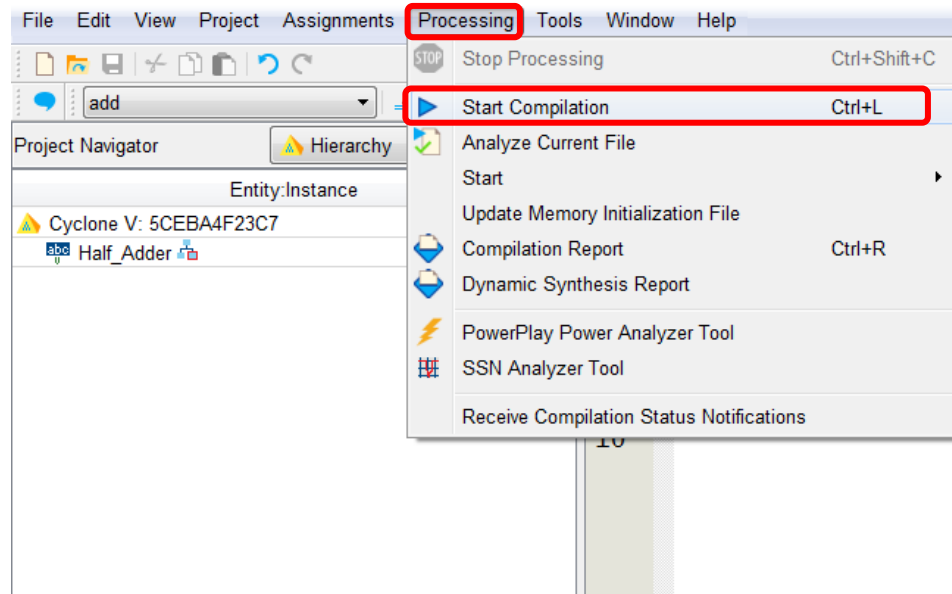
# Programming DE0-CV (5/13)

- 詳細的pin腳位資料請參考moodle檔案 “FPGA\_pin腳位對照.xlsx”

Node Name	Direction	Location	I/O Bank	VREF Group	Fitter Location	I/O Standard
 a	Input	PIN U13 SW0	4A	B4A N0	PIN M16	2.5 V (default)
 b	Input	PIN V13 SW1	4A	B4A N0	PIN N21	2.5 V (default)
 carry	Output	PIN AA1 LED1	2A	B2A N0	PIN N16	2.5 V (default)
 sum	Output	PIN AA2 LED0				

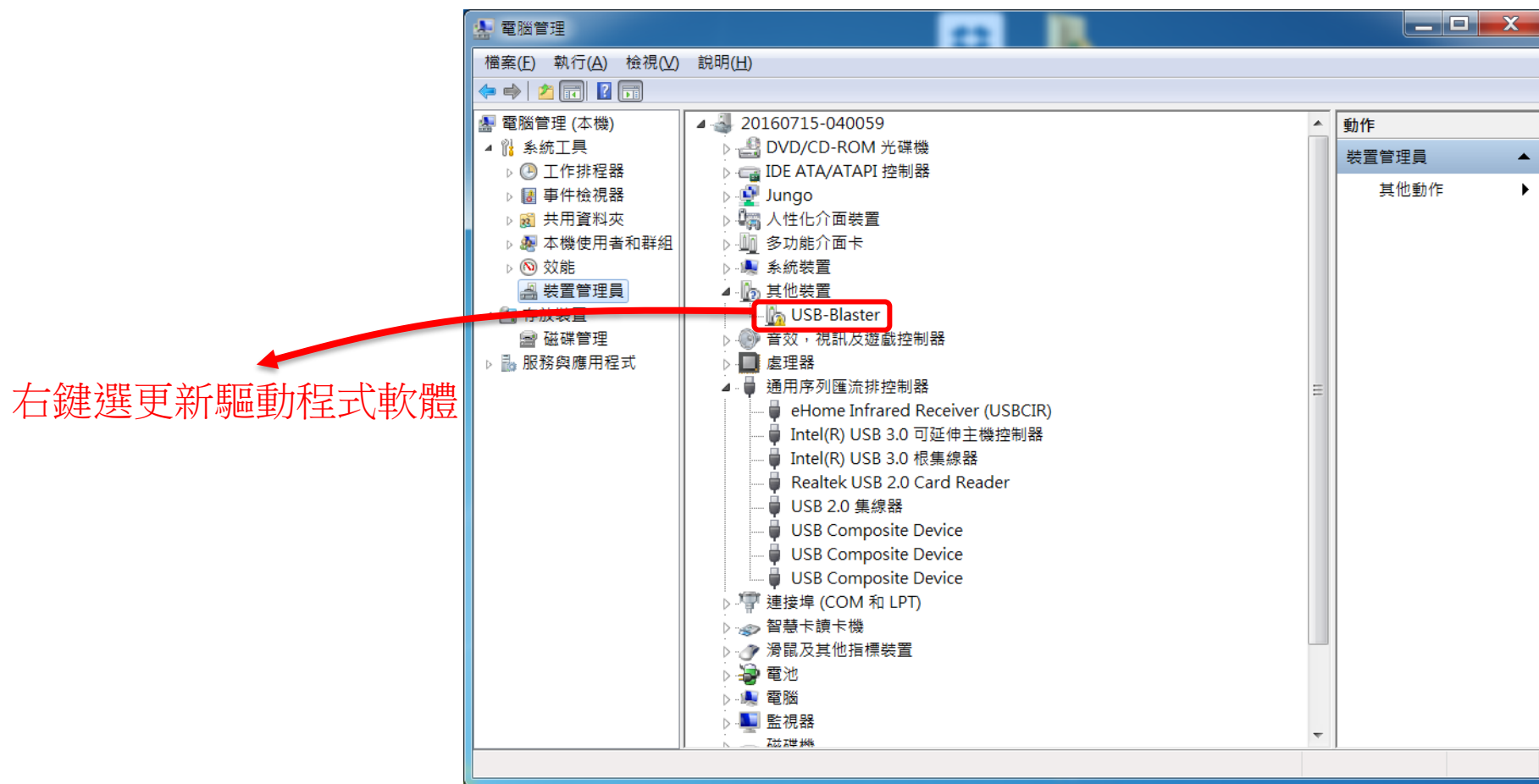
# Programming DE0-CV (6/13)

- Start compilation

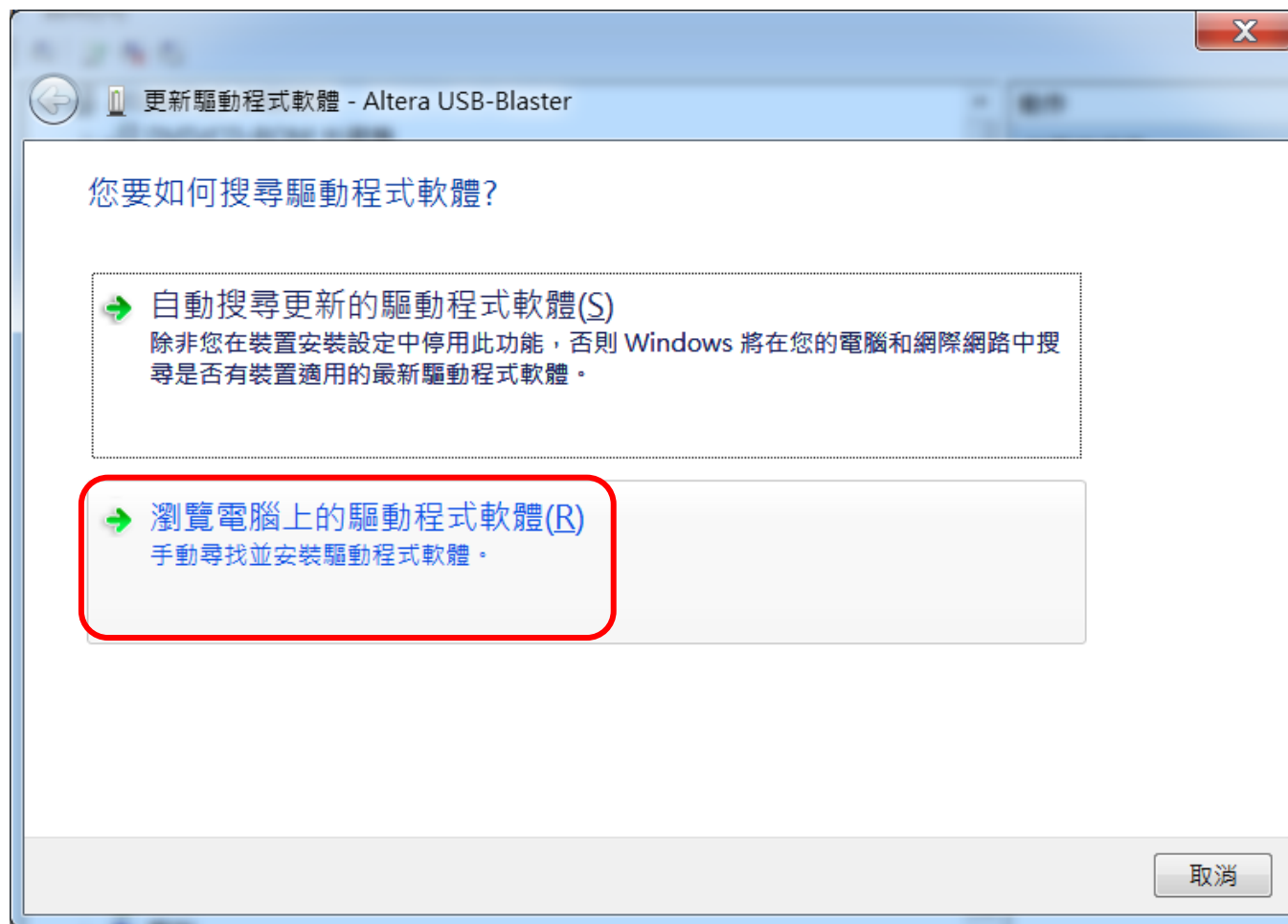




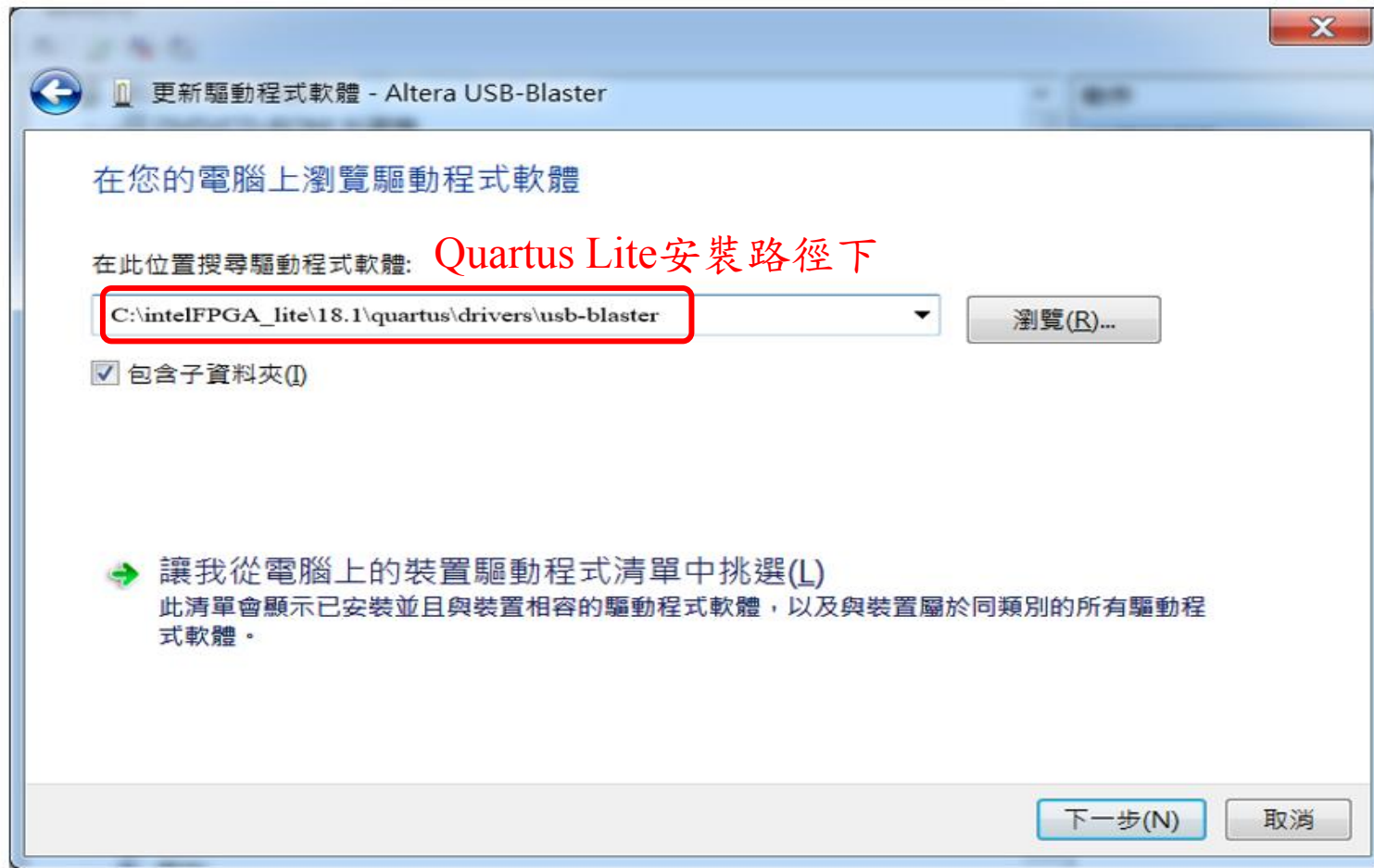
# Programming DE0-CV (7/13)



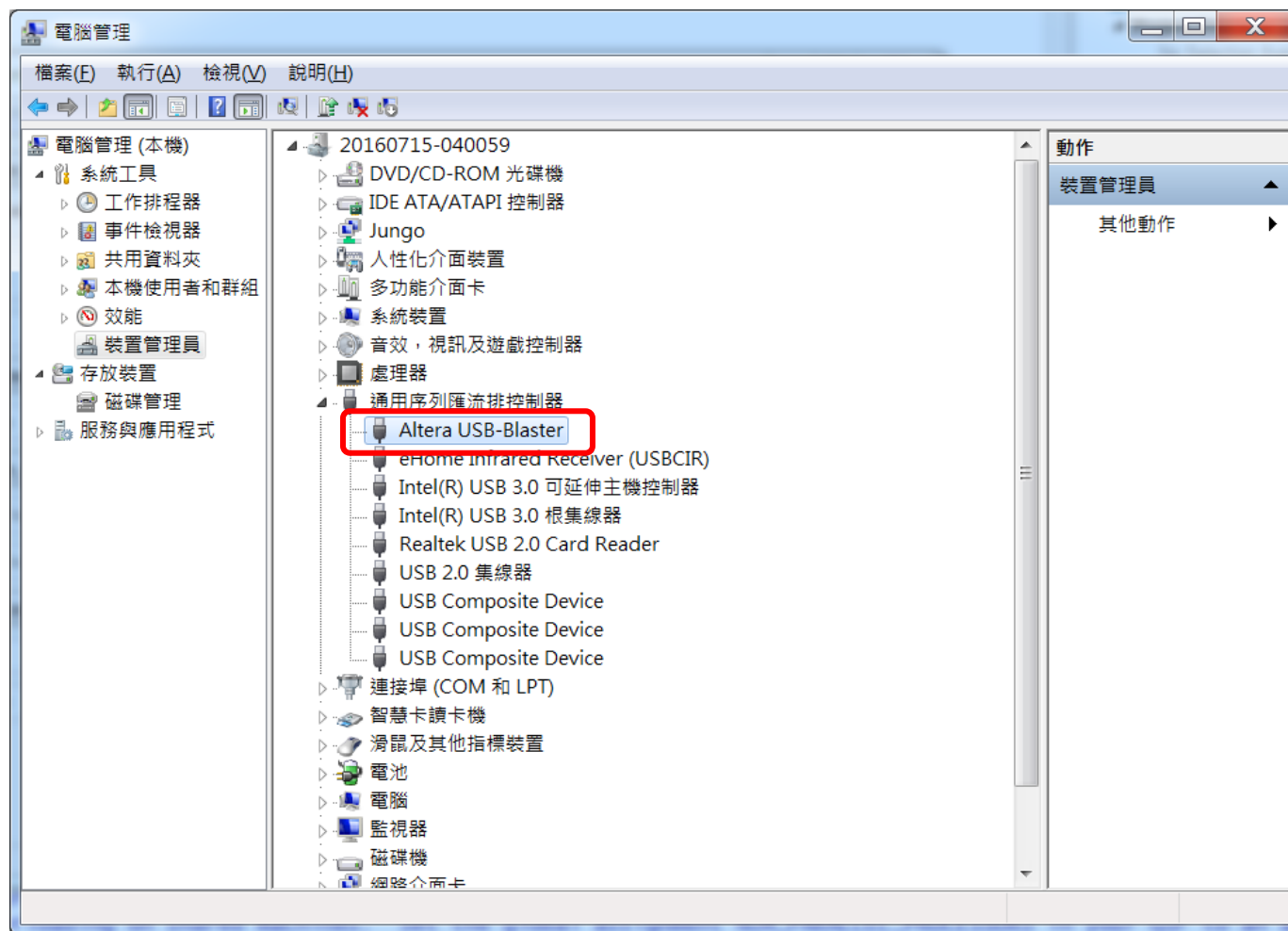
# Programming DE0-CV (8/13)



# Programming DE0-CV (9/13)

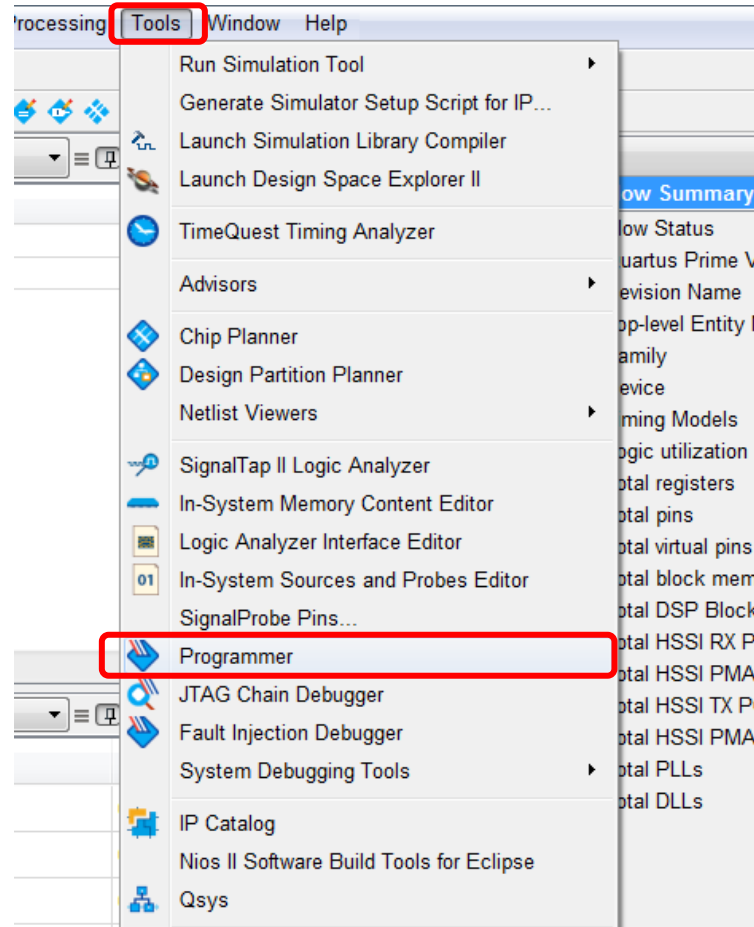


# Programming DE0-CV (10/13)



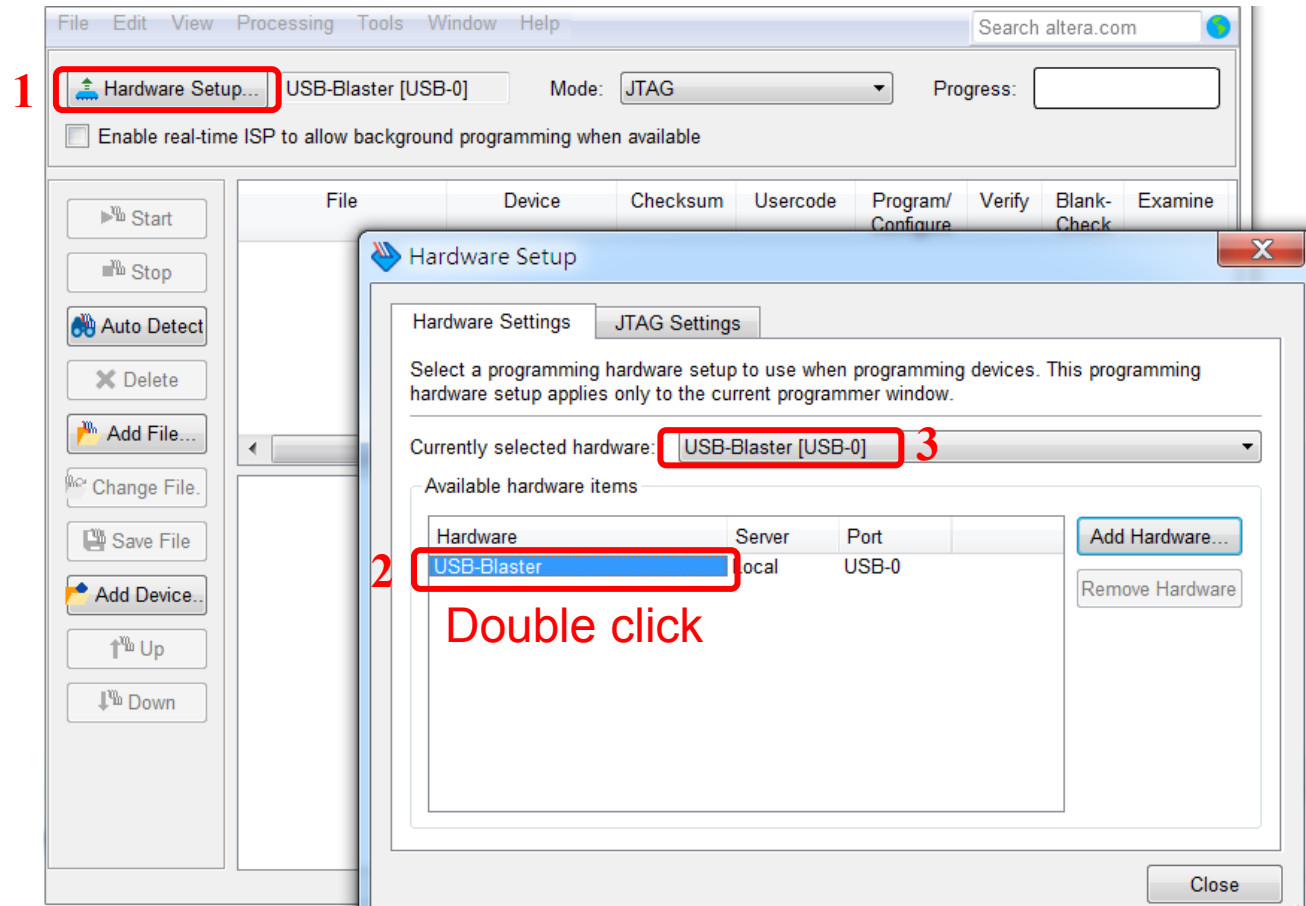
# Programming DE0-CV (11/13)

## ■ Programming device



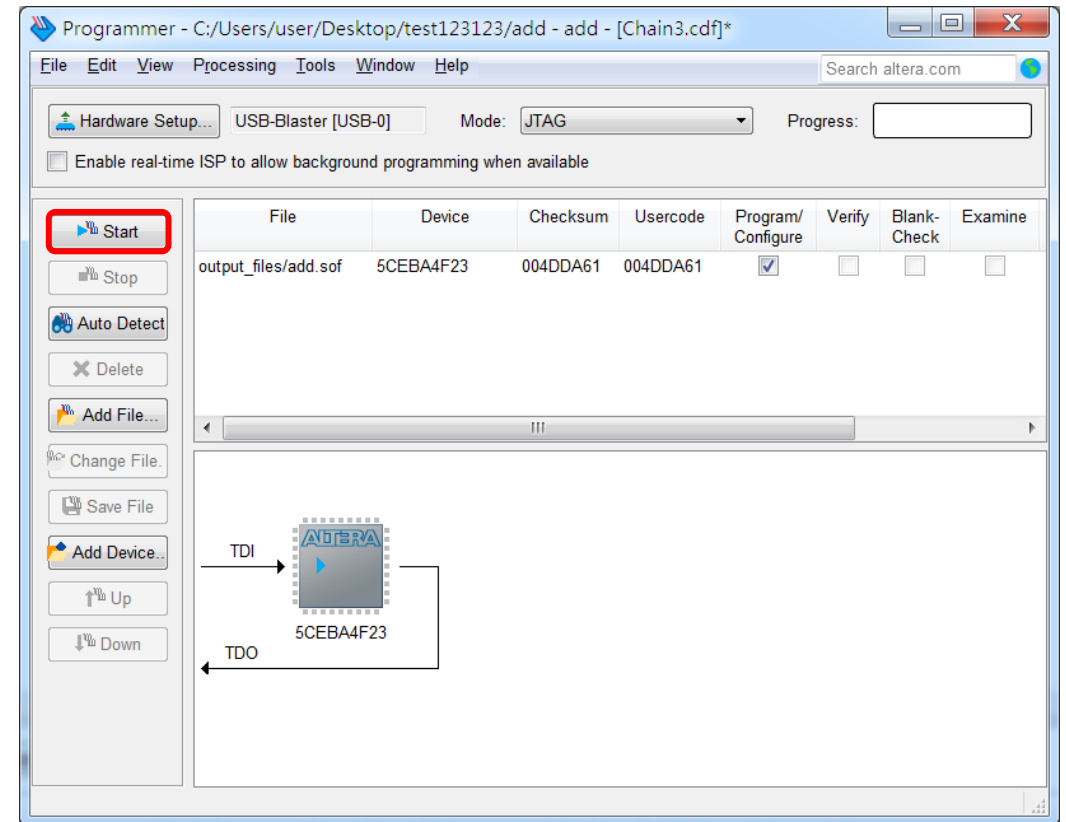
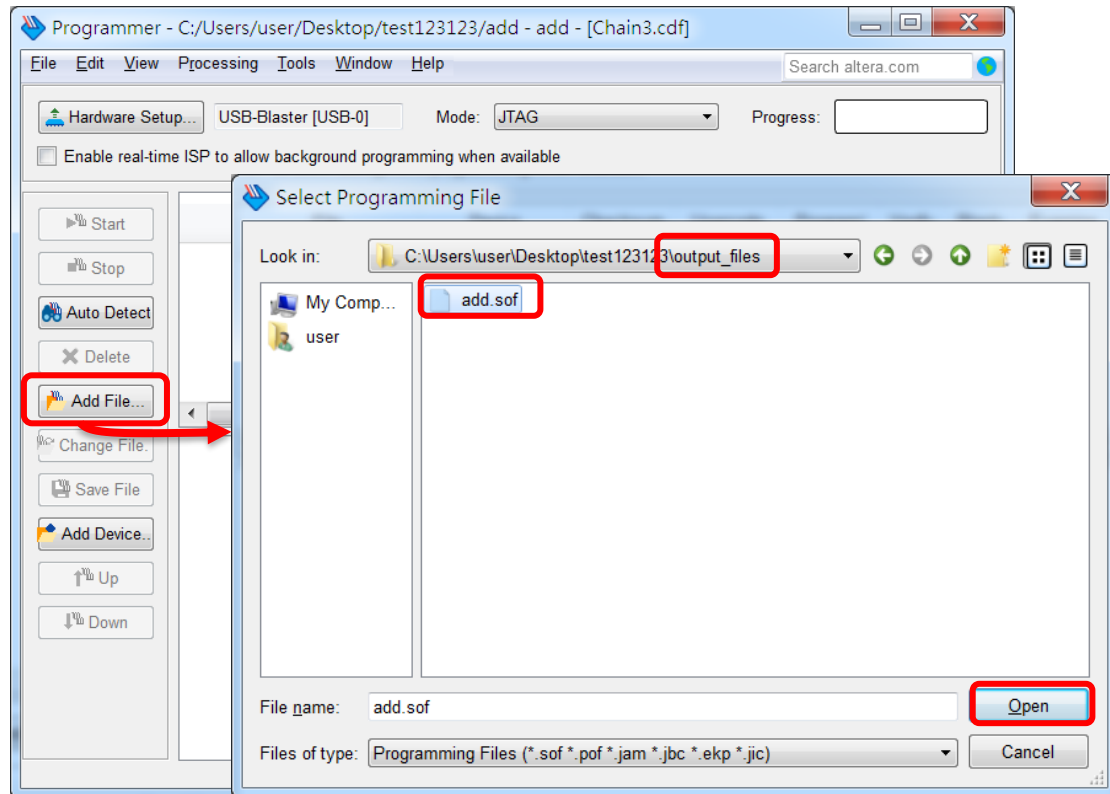
# Programming DE0-CV (12/13)

- Hardware setup: add **USB-Blaster**



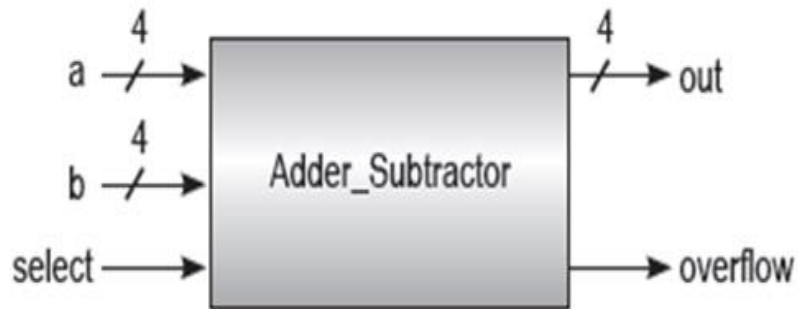
# Programming DE0-CV (13/13)

## ■ Programming device



# Lab I -- Adder-Subtractor to DE0-CV

- 請設計一 4-bit 無號數加減法器，並燒錄至DE0-CV開發板
  - Input: a(4 bits)、b(4 bits)、select(1 bit)
  - Output: out(4 bits)、overflow(1 bit)
- 無號數加減法器藉由選擇(select)訊號決定進行加法或減法運算
  - select訊號為1時，out輸出 $a + b$
  - select訊號為0時，out輸出 $a - b$
  - 溢位(overflow)訊號用來表示有無進位或借位





# Lab I -- Adder-Subtractor to DE0-CV

## ■ Hint:

- 可利用三元運算子(?:)或behavior description中的if-else語法來依照select訊號完成電路
- 可使用concatenation來簡化運算
- 電路運作模式參考1-bit 加減法器之真值表

輸入 (input)			輸出 (output)	
被加減數 (a)	加減數 (b)	選擇 (select)	和 / 差 (out)	溢位 (overflow)
0	0	0	0	0
0	0	1	0	0
0	1	0	1	1
0	1	1	1	0
1	0	0	1	0
1	0	1	1	0
1	1	0	0	0
1	1	1	0	1

# Lab I – Adder-Subtractor

## ■ Example:

### □ select=1

■ out輸出  $a + b$

■ Ex1：輸入  $a=0010$ ,  $b=1101$ ,  $a+b=0010+1101=1111$ , 因為沒有進位,  $out=1111$ ,  $overflow = 0$

■ Ex2：輸入  $a=0100$ ,  $b=1101$ ,  $a+b=0100+1101=10001$ , 因為有進位,  $out=0001$ ,  $overflow = 1$

### □ select=0

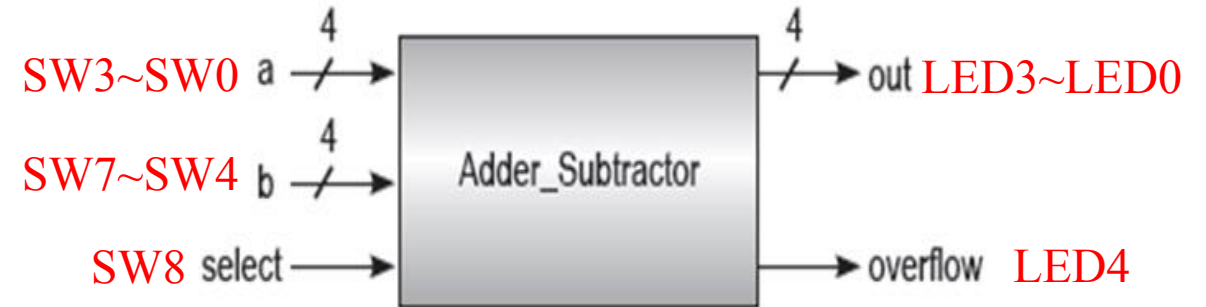
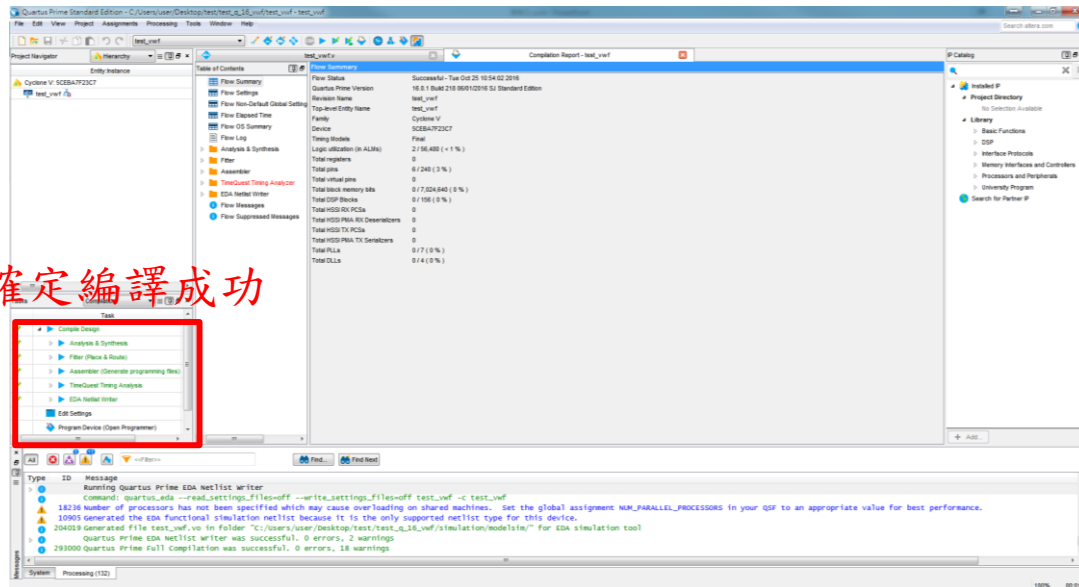
■ out輸出  $a - b$

■ Ex1：輸入  $a=1010$ ,  $b=0011$ ,  $a-b=1010-0011=0111$ , 因為沒有借位,  $out=0111$ ,  $overflow = 0$

■ Ex2：輸入  $a=0010$ ,  $b=1101$ , 因為0010不夠減1101, 所以需要借位, 所以 $a-b=10010-1101=0101$ , 因為有借位,  $out=0101$ ,  $overflow = 1$

# Lab I – Adder-Subtractor to DE0-CV

- 完成verilog電路設計後，需先確認其在Quartus可順利編譯，再將其燒錄至DE0-CV開發板進行驗證
- 使用Switch(SW8~SW0)控制input訊號，使用LED(LED4~LED0)表示output



# Lab II – encoder to DE0-CV

- 請設計一8對3編碼器(8 to 3 encoder)
- 編碼器可以將 $2^n$ 個輸入訊號轉換成n位元輸出訊號，假設有m個輸入與n個輸出，則稱為m對n編碼器
- Hint: 可使用behavior description之case語法實作

輸入 (input)								輸出 (output)			
in[7]	in[6]	in[5]	in[4]	in[3]	in[2]	in[1]	in[0]	valid	out[2]	out[1]	out[0]
0	0	0	0	0	0	0	1	1	0	0	0
0	0	0	0	0	0	1	0	1	0	0	1
0	0	0	0	0	1	0	0	1	0	1	0
0	0	0	0	1	0	0	0	1	0	1	1
0	0	0	1	0	0	0	0	1	1	0	0
0	0	1	0	0	0	0	0	1	1	0	1
0	1	0	0	0	0	0	0	1	1	1	0
1	0	0	0	0	0	0	0	1	1	1	1
其餘的輸入情況								0	0	0	0

```

input [7:0] i;
output valid, [2:0] out;
reg valid;
always @(*)
    if (i[0])
        out = 3'b000;
    else if (i[1])
        out = 3'b001;
    else if (i[2])
        out = 3'b010;
    else if (i[3])
        out = 3'b011;
    else if (i[4])
        out = 3'b100;
    else if (i[5])
        out = 3'b101;
    else if (i[6])
        out = 3'b110;
    else if (i[7])
        out = 3'b111;
    else
        out = 3'b000;
    valid = 1;
end

```

begin

★ case (i)

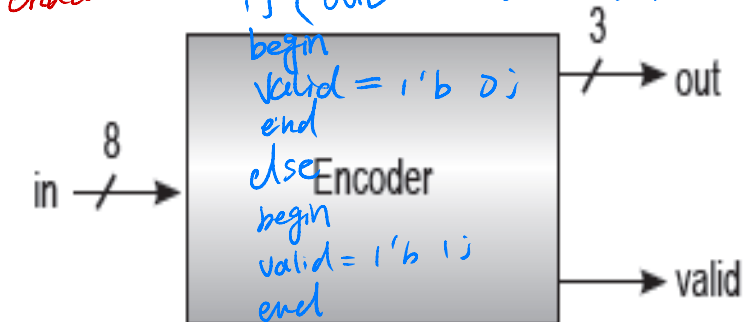
8'b00000001 : out = 3'b000;

8'b10000000 : out = 3'b111;

default : out = 3'b000;

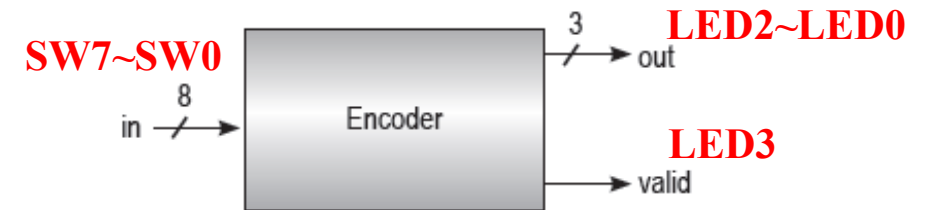
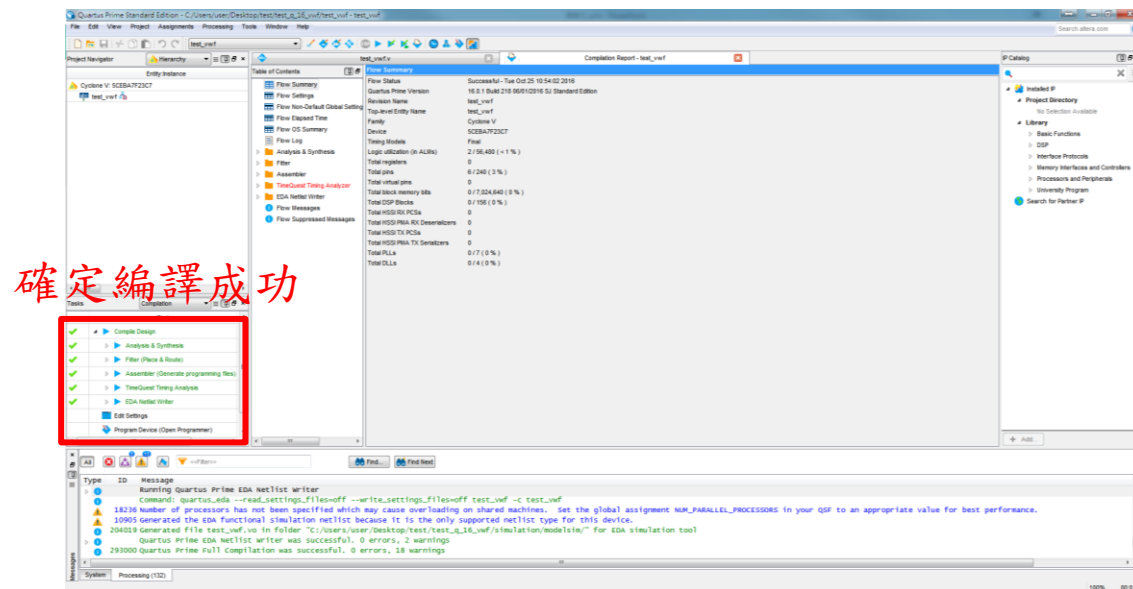
if (out == 3'b000 && i != 8'b00000001)

endcase



# Lab II – encoder to DE0-CV

- 完成verilog電路設計後，需先確認其在Quartus可順利編譯，再將其燒錄至DE0-CV開發板進行驗證
- 使用Switch(SW7~SW0)控制input訊號，使用LED(LED3~LED0)表示output



# Notice

- 請勿命名中文或數字開頭的資料夾
- Device family 請確認與 FPGA Chip 符合 (5CEFA4F23C7)
- Top module name & Project name 需要一致
- 在組合電路中，case、if...else...若沒有寫滿，合成後會產生latch