Department of Computer Science and Information Engineering
National Cheng Kung University

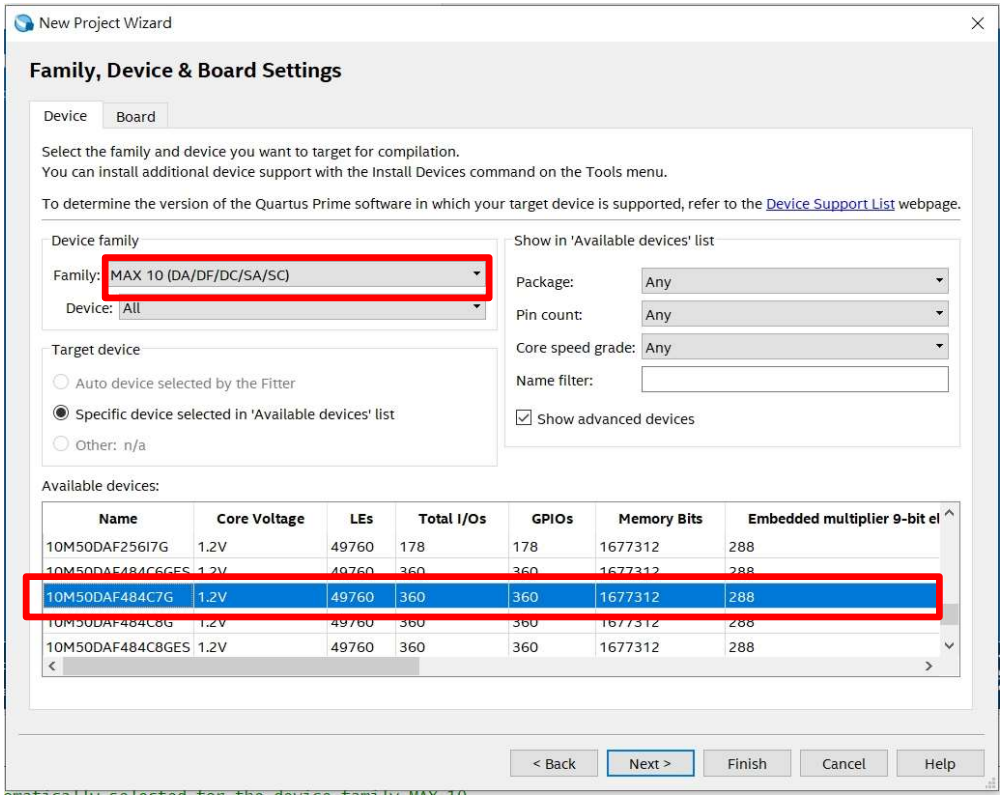
LAB - 06

陳培殷老師
國立成功大學　資訊工程系

- **Specify device settings - (DE10-Lite Device family are used). Click "Next."**

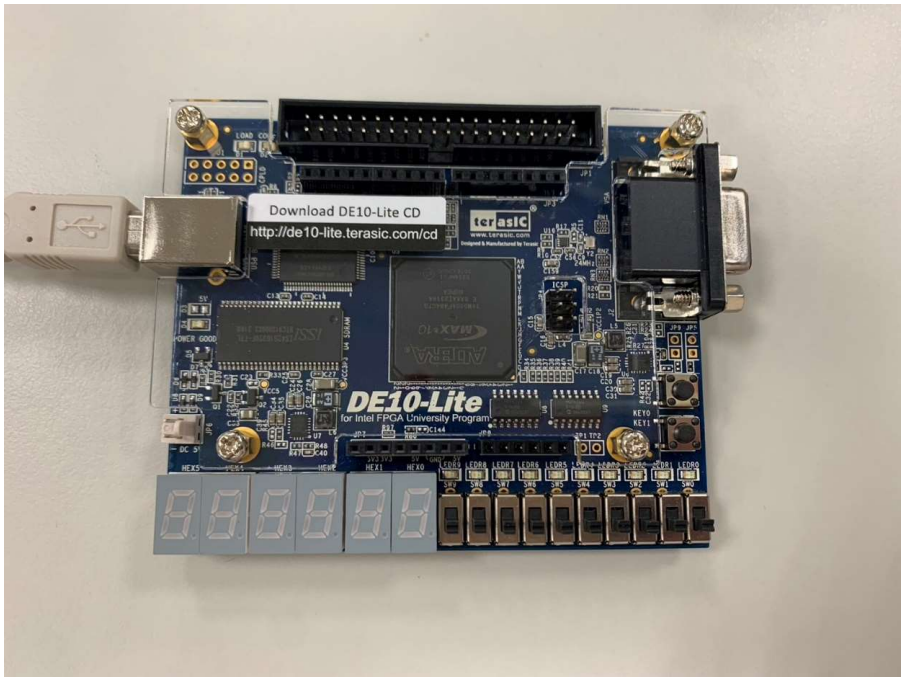
MAX 10(DA/DF/DC/SA/SC)

10M50DAF484C7G



Introduction to DE10-Lite

USB

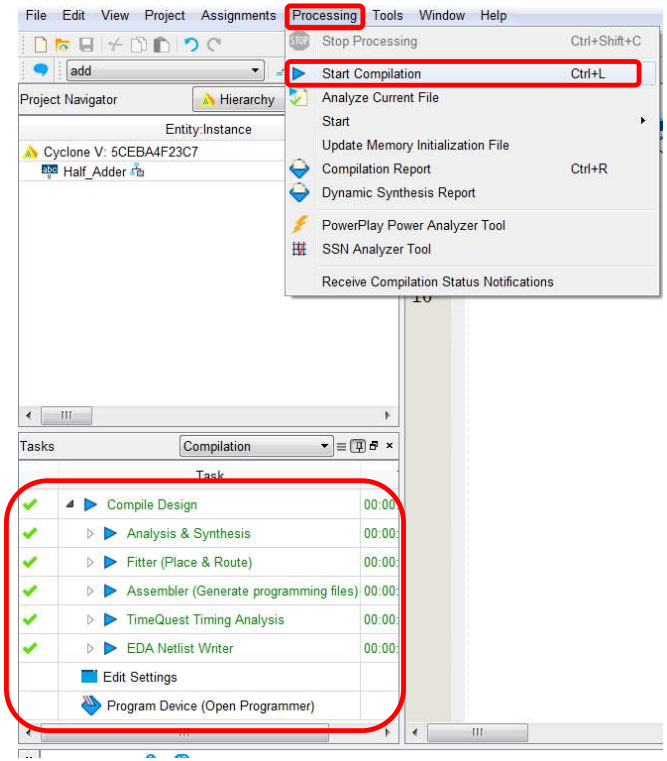


Programming DE10-Lite (1/13)

```
 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
```

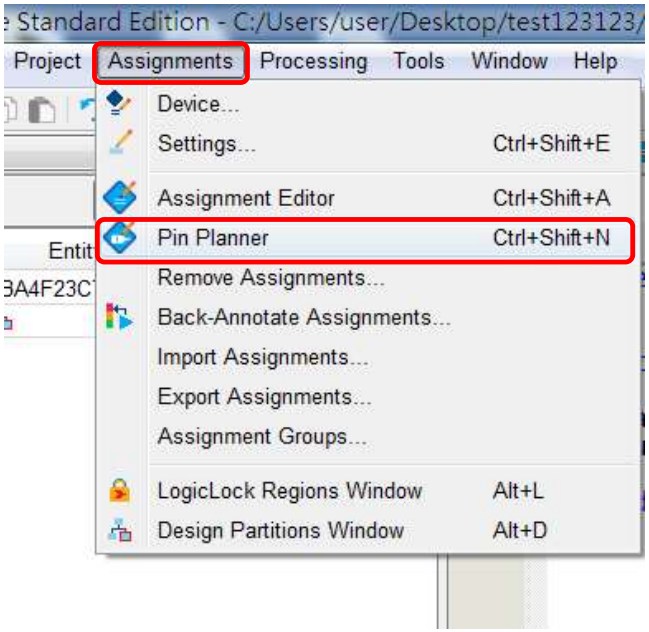
Programming DE10-Lite (2/13)

- Start compilation



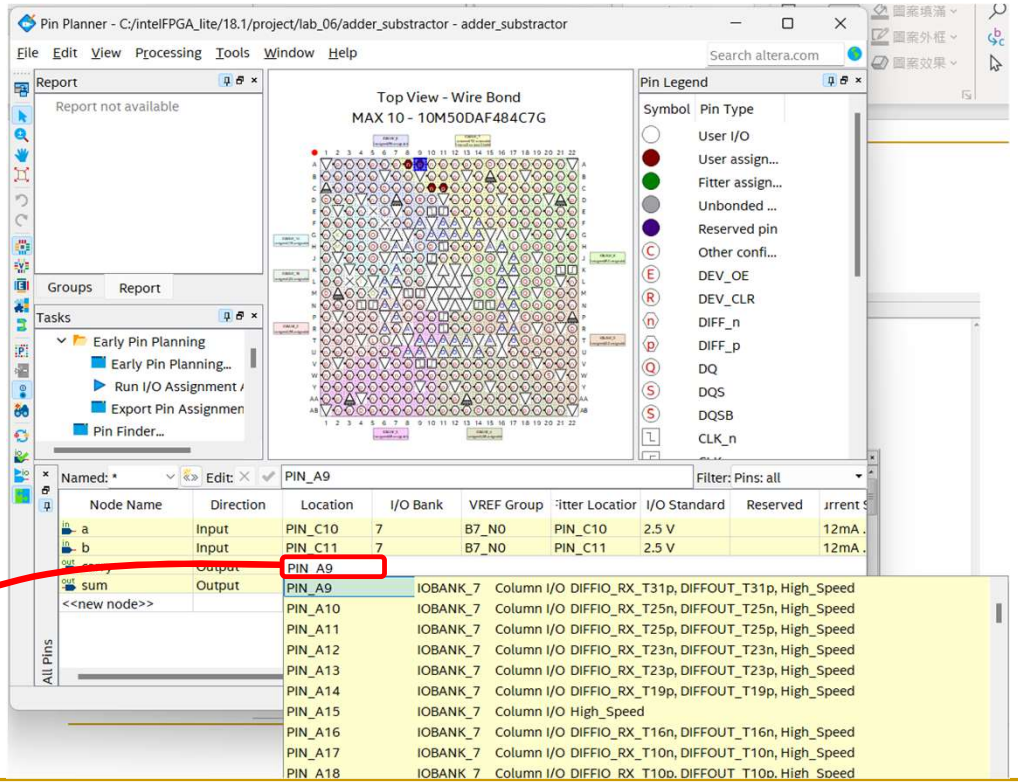
Programming DE10-Lite (3/13)

- Open Pin Planner



Programming DE10-Lite (4/13)

- Assign pin location to all inputs and outputs



Double click

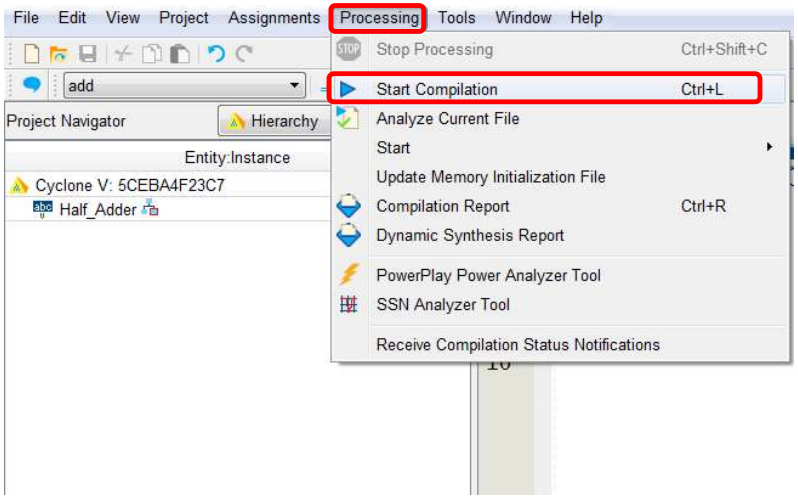
Programming DE10-Lite (5/13)

- 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_C10 SW0	7	B7_N0	PIN_C10	2.5 V
b	Input	PIN_C11 SW1	7	B7_N0	PIN_C11	2.5 V
carry	Output	PIN_A9 LED1	7	B7_N0	PIN_A9	2.5 V
sum	Output	PIN_A8 LED0	7	B7_N0	PIN_A8	2.5 V
<<new node>>						

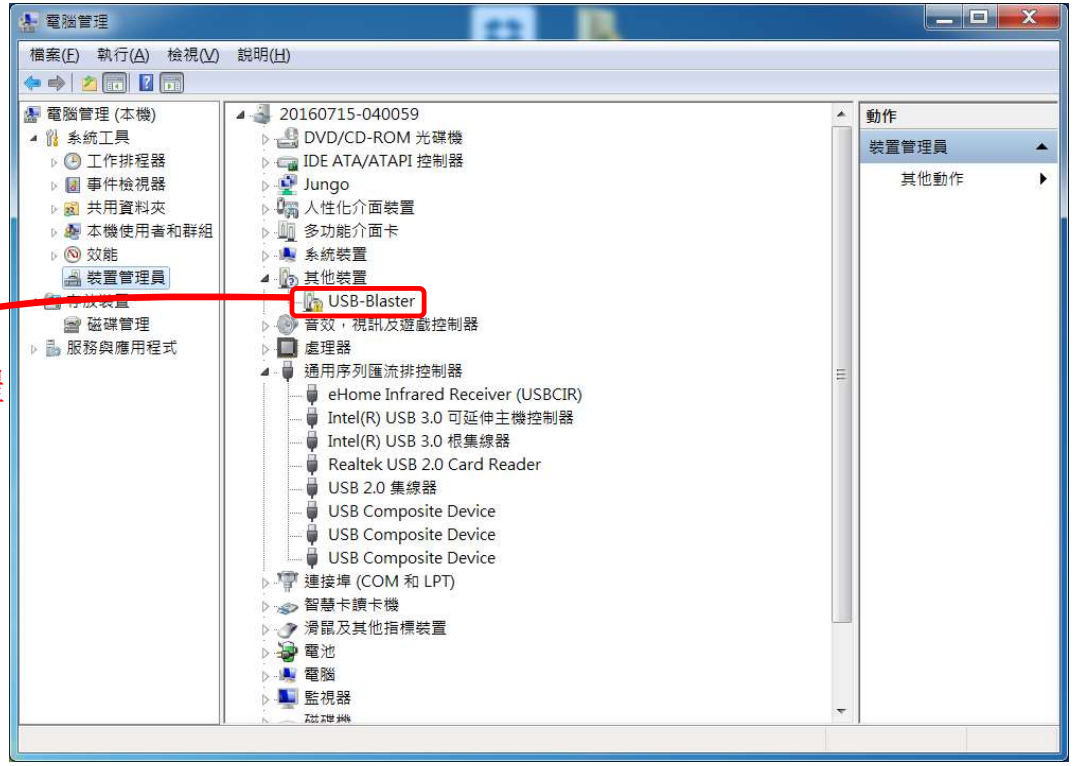
Programming DE10-Lite (6/13)

- Start compilation

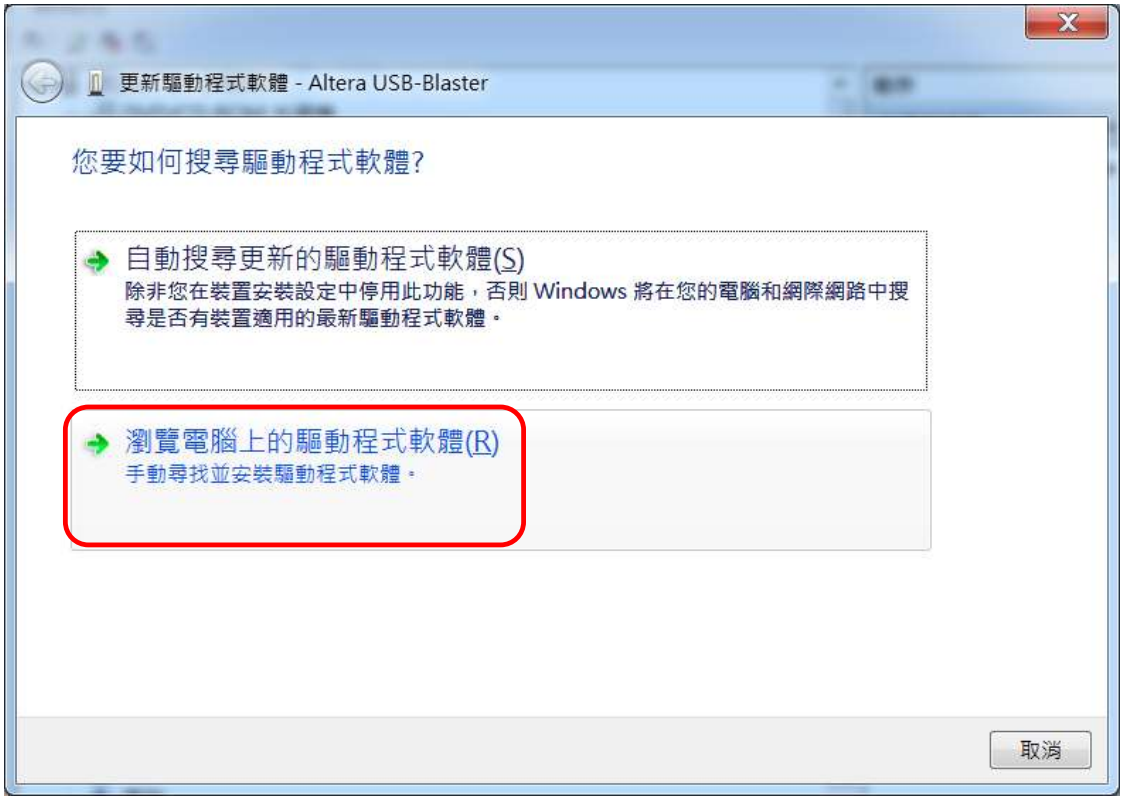


Programming DE10-Lite (7/13)

右鍵選更新驅動程式軟體

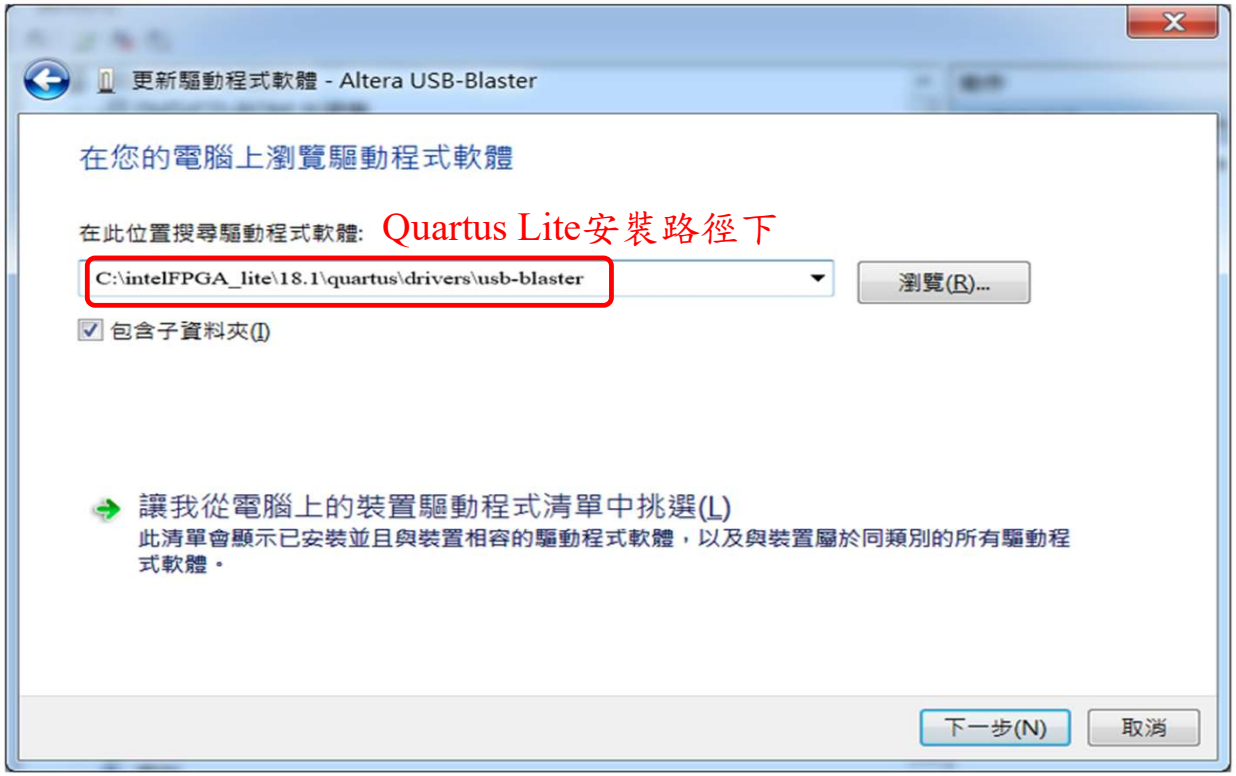


Programming DE10-Lite (8/13)

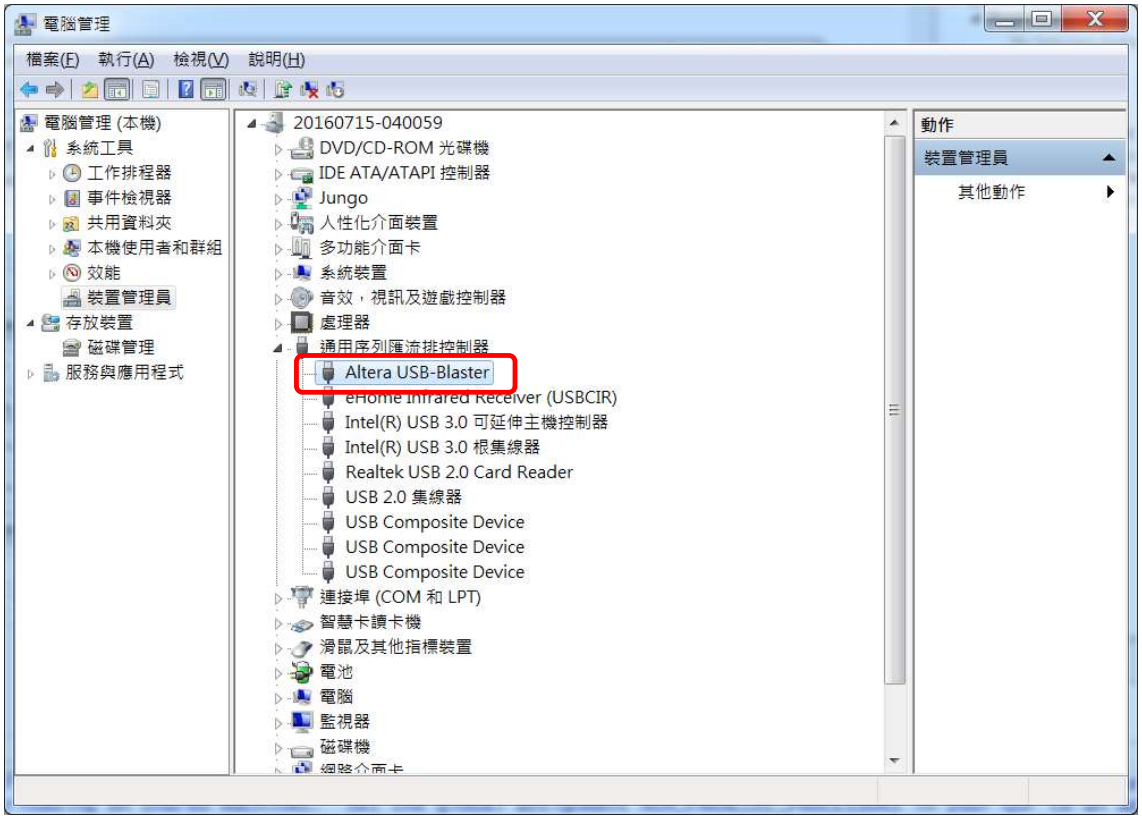


Programming DE10-Lite (9/13)

個人電腦 : C:\intelFPGA_lite\18.1\quartus\drivers\usb-blaster
實驗室電腦 : C:\intelFPGA_lite\16.1\quartus\drivers\usb-blaster

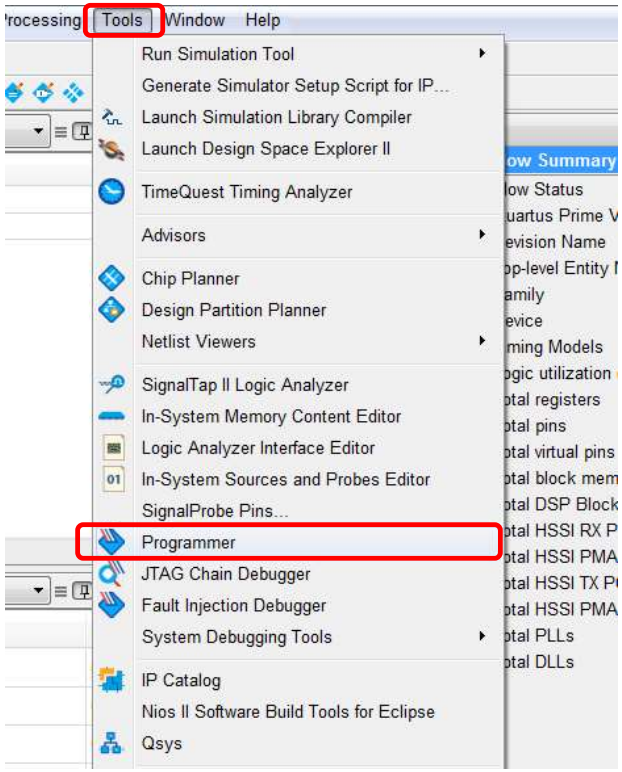


Programming DE10-Lite (10/13)



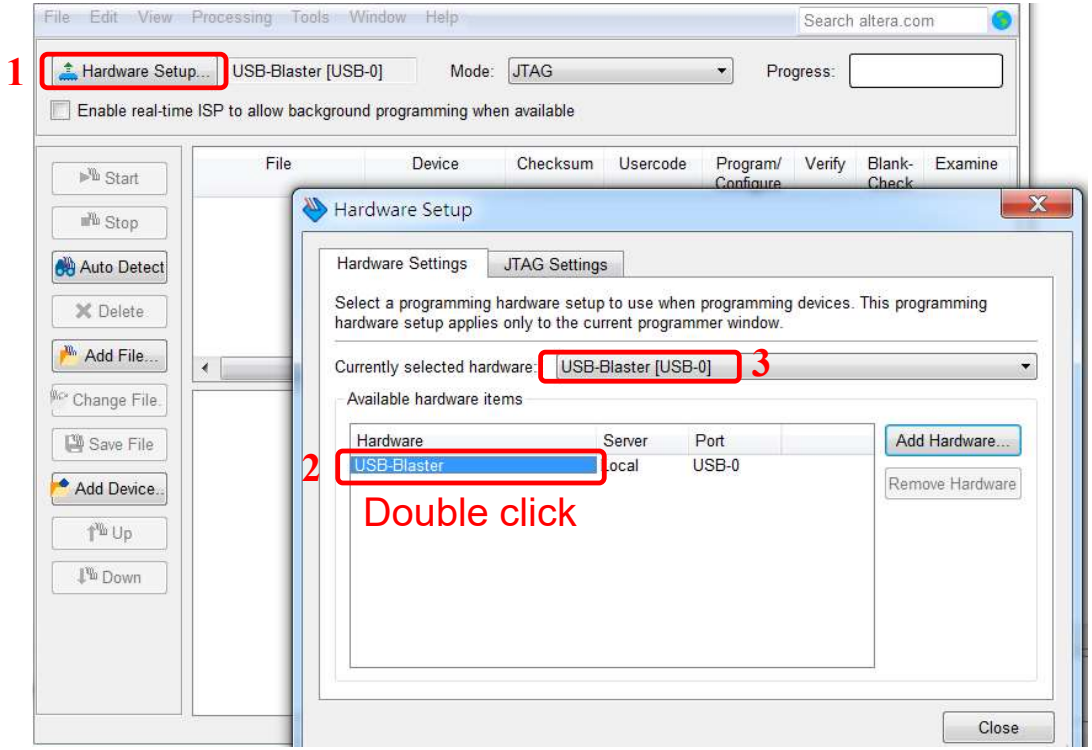
Programming DE10-Lite (11/13)

- Programming device



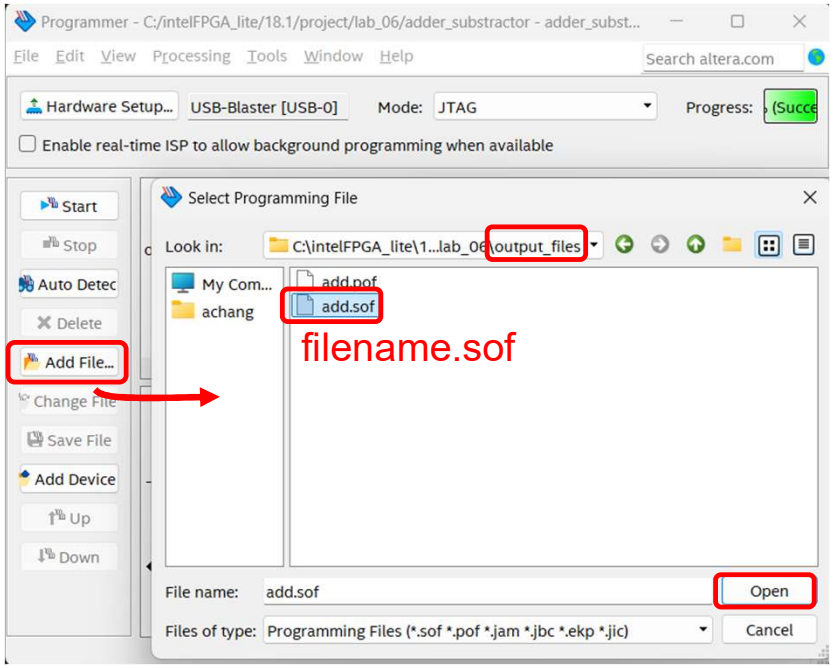
Programming DE10-Lite (12/13)

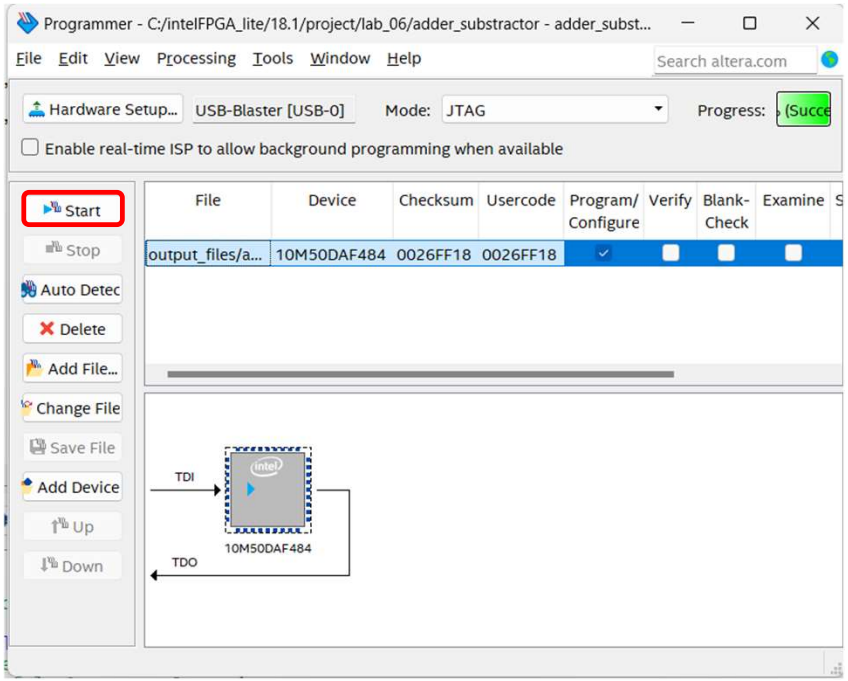
- Hardware setup: add USB-Blaster



Programming DE10-Lite (13/13)

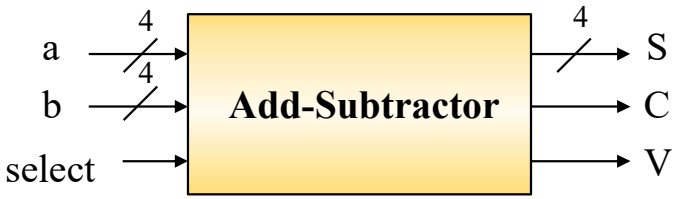
- Programming device

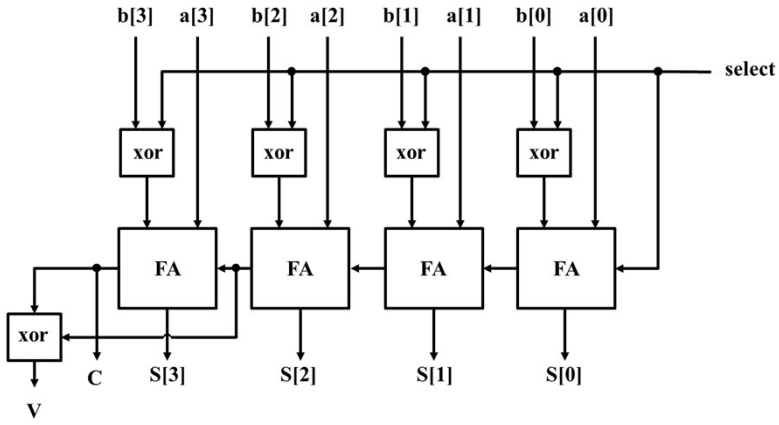




Lab I -- Adder-Subtractor to DE10-Lite

- 請設計一 4-bit 無號數加減法器，並燒錄至DE10-Lite開發板
  - Input: a(4 bits)、b(4 bits)、select(1 bit)
  - Output: S(4 bits)、C (1 bit) carry、V(1 bit) overflow
- 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 DE10-Lite

```
reg c;
always@(*) begin
    if (sel == 1'd1) begin
        c = a;
    end
    else begin
        c = b;
    end
end
```

```
wire c;
wire temp0, temp1;
wire sel_inv = ~sel;
and u0 (temp0, a, sel);
and u1 (temp1, b, sel_inv);
or u2 (c, temp0, temp1);
```

```
wire c;
assign c = (sel == 1'd1) ? a : b ;
```

```
reg carry;
reg sum;
always@(*) begin
    {carry, sum} = a + b;
end
```

HINT!

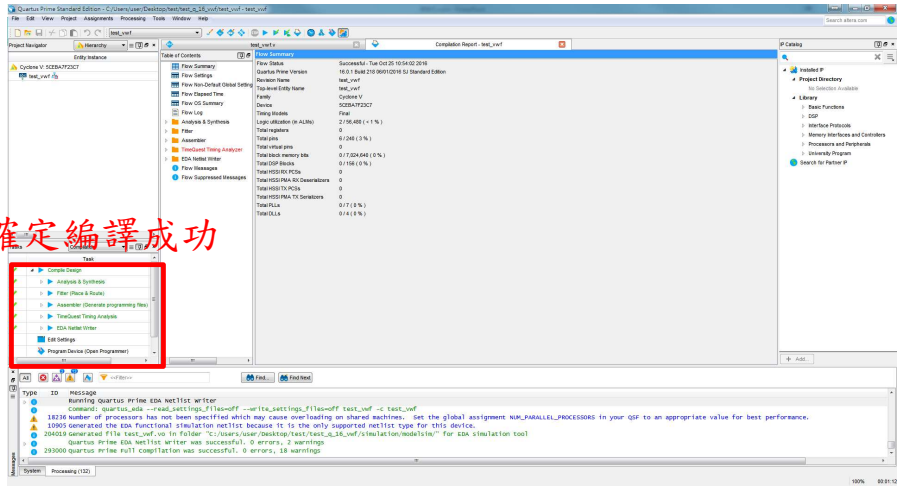
```
module test (a, b, sum, carry);
    input a;
    input b;
    output sum;
    output carry;

    reg sum_reg, carry_reg;
    assign sum = sum_reg;
    assign carry = carry_reg;

    always@(*) begin
        {carry_reg, sum_reg} = a+b;
    end
```

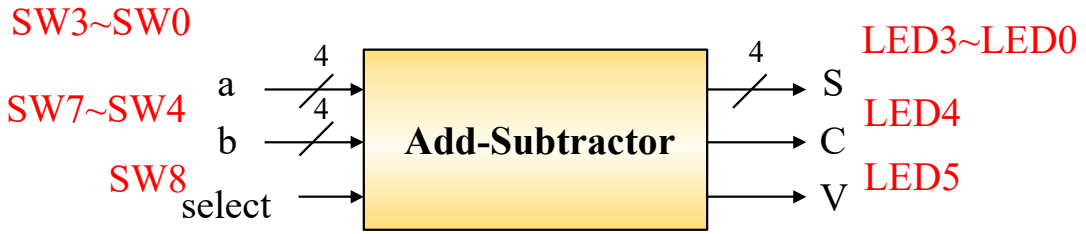
Lab I – Adder-Subtractor

- Example:
  - select=0
    - 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=1
    - 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 DE10-Lite

- 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DE10-Lite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

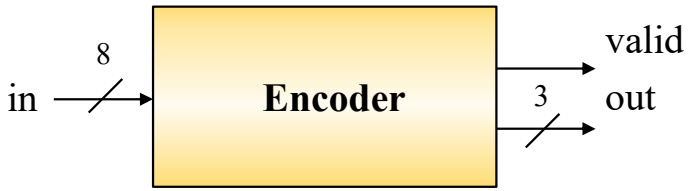




Lab II – encoder to DE10-Lite

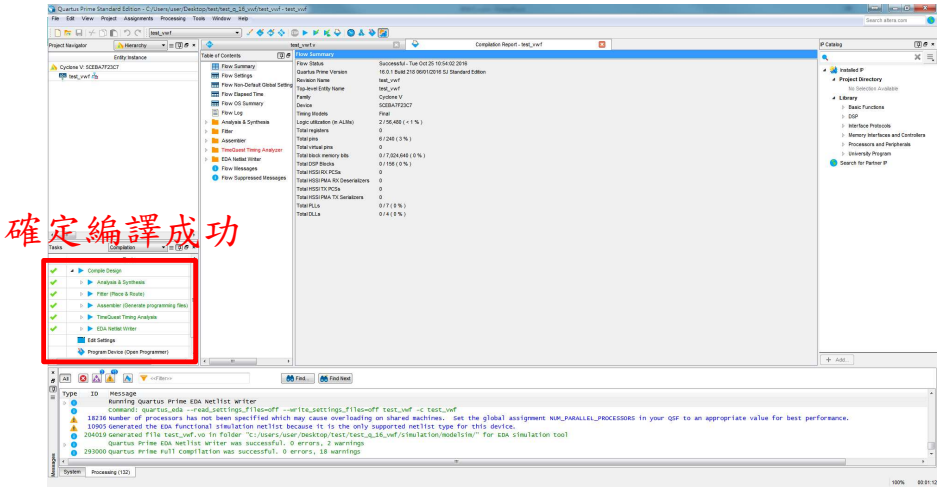
- 請設計一8對3編碼器(8 to 3 encoder)
- 編碼器可以將2^n個輸入訊號轉換成n位元輸出訊號，假設有m個輸入與n個輸出，則稱為m對n編碼器
- Hint: 可使用behavior description之case語法實作

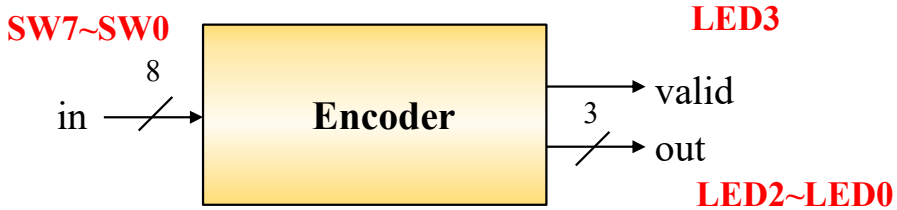
輸入 (input)								輸出 (output)			
in[7]	in[6]	in[5]	in[4]	in[3]	in[2]	in[1]	in[0]	valid	out[2]	out[1]	out[0]
0	0	0	0	0	0	0	1	1	0	0	0
0	0	0	0	0	0	1	0	1	0	0	1
0	0	0	0	0	1	0	0	1	0	1	0
0	0	0	0	1	0	0	0	1	0	1	1
0	0	0	1	0	0	0	0	1	1	0	0
0	0	1	0	0	0	0	0	1	1	0	1
0	1	0	0	0	0	0	0	1	1	1	0
1	0	0	0	0	0	0	0	1	1	1	1
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



Lab II – encoder to DE10-Lite

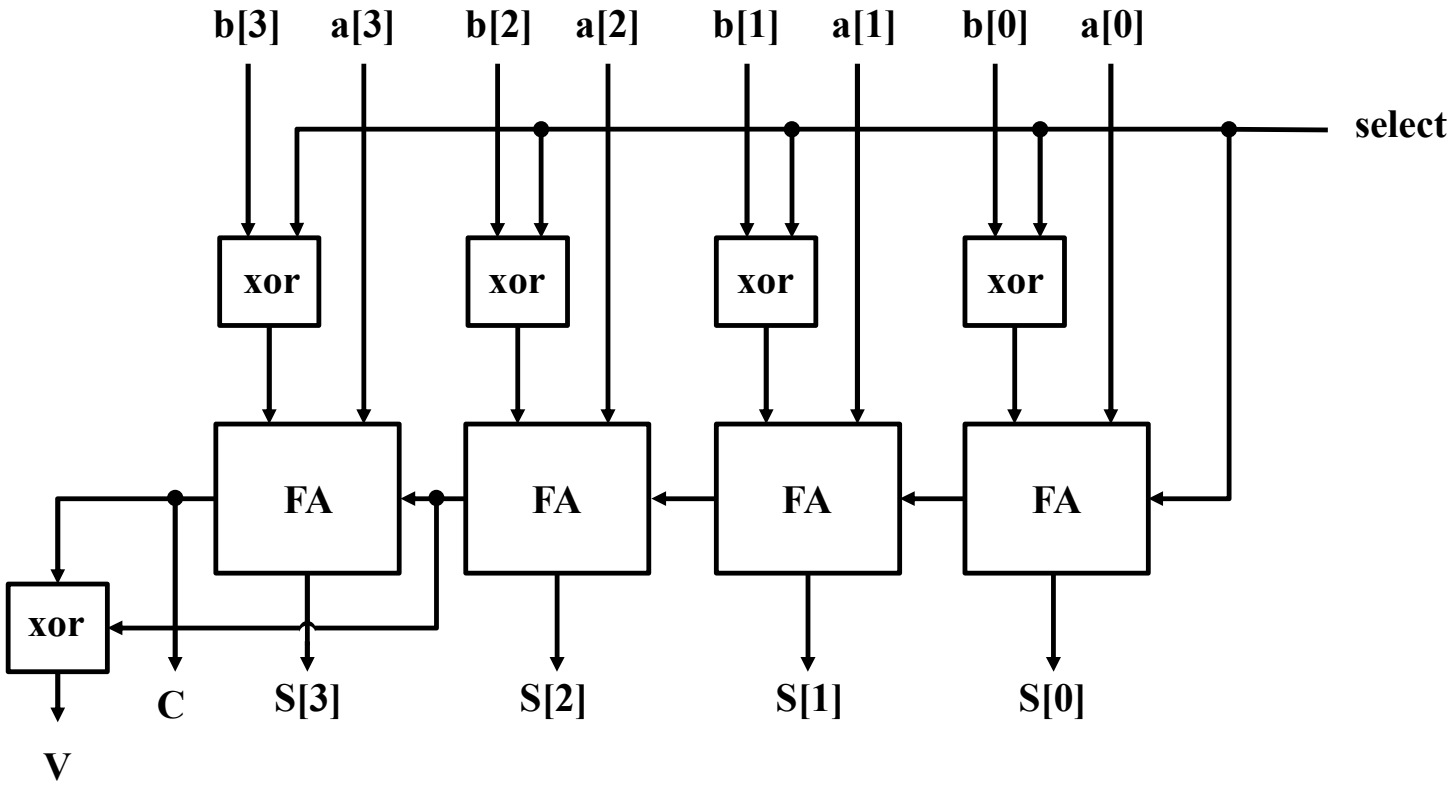
- 完成verilog電路設計後，需先確認其在Quartus可順利編譯，再將其燒錄至DE10-Lite開發板進行驗證
- 使用Switch(SW7~SW0)控制input訊號，使用LED(LED3~LED0)表示output





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 (10M50DAF484C7G)
- 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


b[3]
a[3]
b[2]
a[2]
b[1]
a[1]
b[0]
a[0]
select
xor
xor
xor
xor
FA
FA
FA
FA
xor
C
S[3]
S[2]
S[1]
S[0]
V