

圖 1-1 ALTERA 網站首頁

2. 點選 Download 按鈕之後便會出現 Download Center 的網頁畫面，此處我們必需選擇 Quartus® II Web Edition v7.2 Service Pack 2 來下載。

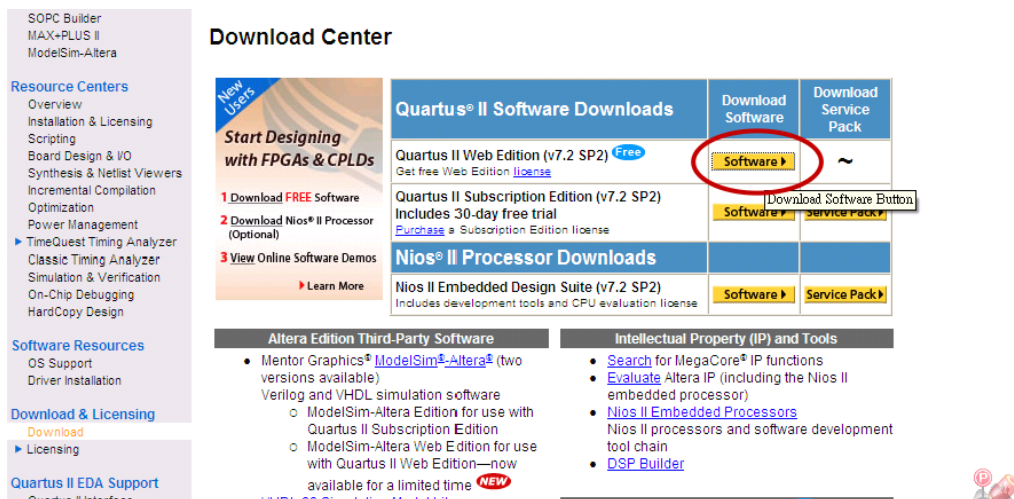
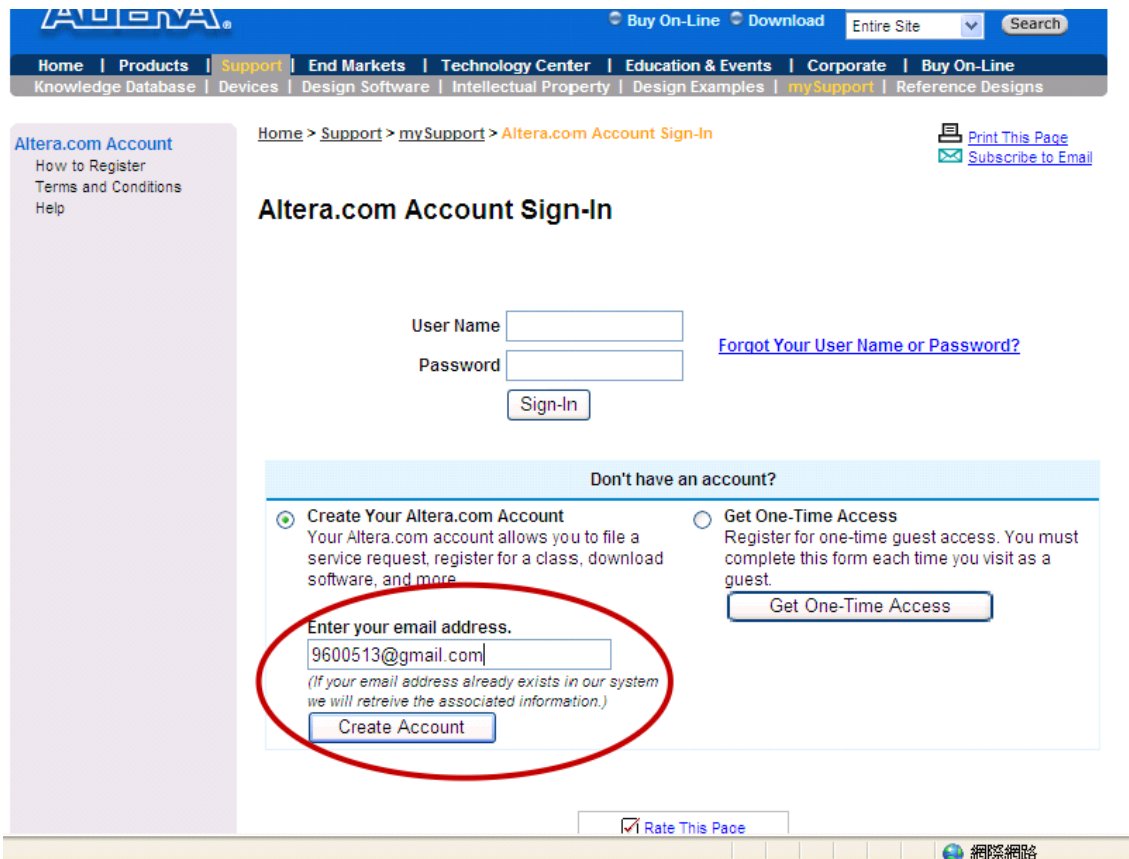


圖 1-2 Quartus II 下載中心網頁

3. 進入建立使用者帳號的訊息畫面。第一次使用的使用者在輸入正確的電子郵件信箱之後，隨即會產生一個要求輸入使用者基本資訊的帳號註冊視窗，在依要求逐項輸入欄位資料(有星號標示的欄位必填)，並自行建立使用者名稱(Create User Name)和密碼(Create Password)之後，按下【Create Account】。



Altera.com Account Sign-In

User Name

Password

[Forgot Your User Name or Password?](#)

Don't have an account?

**Create Your Altera.com Account**  
Your Altera.com account allows you to file a service request, register for a class, download software, and more.

**Get One-Time Access**  
Register for one-time guest access. You must complete this form each time you visit as a guest.

Enter your email address.  
  
(If your email address already exists in our system we will retrieve the associated information.)

Rate This Page

網際網路

圖 1-3 建立自己在 Altera.com 的帳號



Home | Products | **Support** | End Markets | Technology Center | Education & Events | Corporate | Buy On-Line  
 Knowledge Database | Devices | Design Software | Intellectual Property | Design Examples | **mySupport** | Reference Designs

Altera.com Account  
 How to Register  
 Terms and Conditions  
 Help

Home > Support > mySupport > Altera.com Account Registration [Print This Page](#)  
[Subscribe to Email](#)

### Altera.com Account Registration

Create Your Altera.com Account

(Note: Data must be entered in English)

\* First Name   
 \* Last Name   
 Job Title   
 \* Company Name   
 \* Address   
 Address (Line 2)   
 \* City   
 \* Country   
 State / Province (Outside of USA)   
 Zip / Postal Code   
 Email Address  ([Edit Email](#))  
 \* Telephone Number   
 Example: +6046366100 x123

圖 1-4 輸入個人資料

**ALTERA** Literature Licensing Buy On-Line Download  
 Entire Site Search

Home | Products | **Support** | End Markets | Technology Center | Education & Events | Corporate | Buy On-Line  
 Knowledge Database | Devices | Design Software | Intellectual Property | Design Examples | **mySupport** | Reference Designs

Altera.com Account  
 How to Register  
 Terms and Conditions  
 Help

Home > Support > mySupport > Your Altera.com Account has been Created [Print This Page](#)  
[Subscribe to Email](#)

### Your Altera.com Account has been Created

Thank you for registering with Altera! You now have an Altera.com account. We are sending you an Altera.com account confirmation email for your reference.

You can use this account to register for classes, download software, file a service request and much more.

[Rate This Page](#)

[Please Give Us Feedback](#)

圖 1-5 帳號建立完成

- 我們可點選新彈出 Download Manager 視窗下方的“[download the file without using the Download Manager](#)”，便能開始下載 Quartus II Software Web Edition Version 7.2 Service Pack 2。

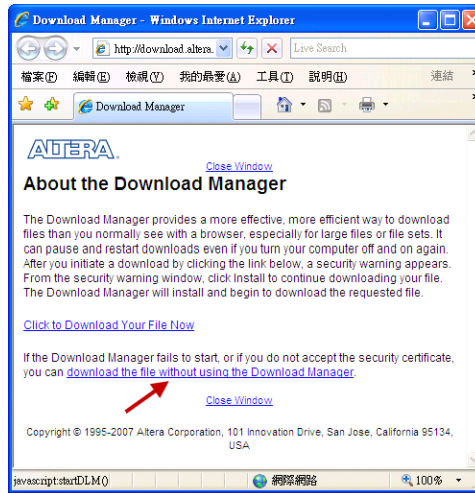


圖 1-6 所產生之 Download Manager 視窗

## 1-2-2 Quartus II 軟體取得授權

- 在執行下載 Quartus II 軟體後，我們也可同時為 Quartus II 軟體取得授權檔(\*.dat)。首先，我們得在完成下載程序的最後網頁中點選“License your software”。

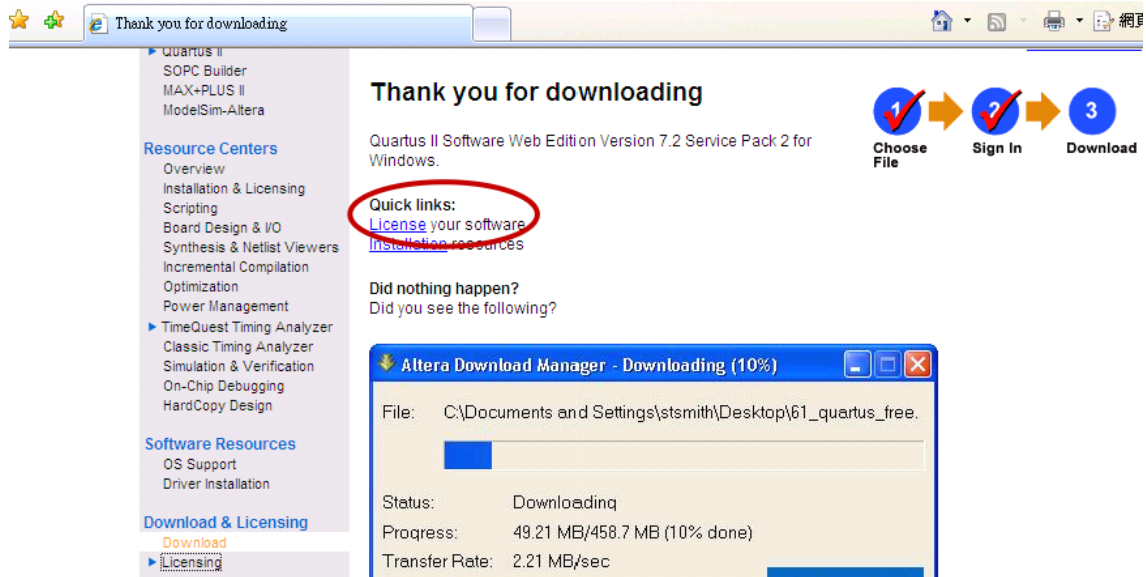


圖 1-7 License your software

2. 選擇【 Get licenses】之後，便會出現 Quartus II Web Edition Software Licenses 的進一步選擇頁面，此處我們點選 “Get a license for the Quartus II Web Edition Software Licenses and the ModelSim-Altera Web Edition Software”。

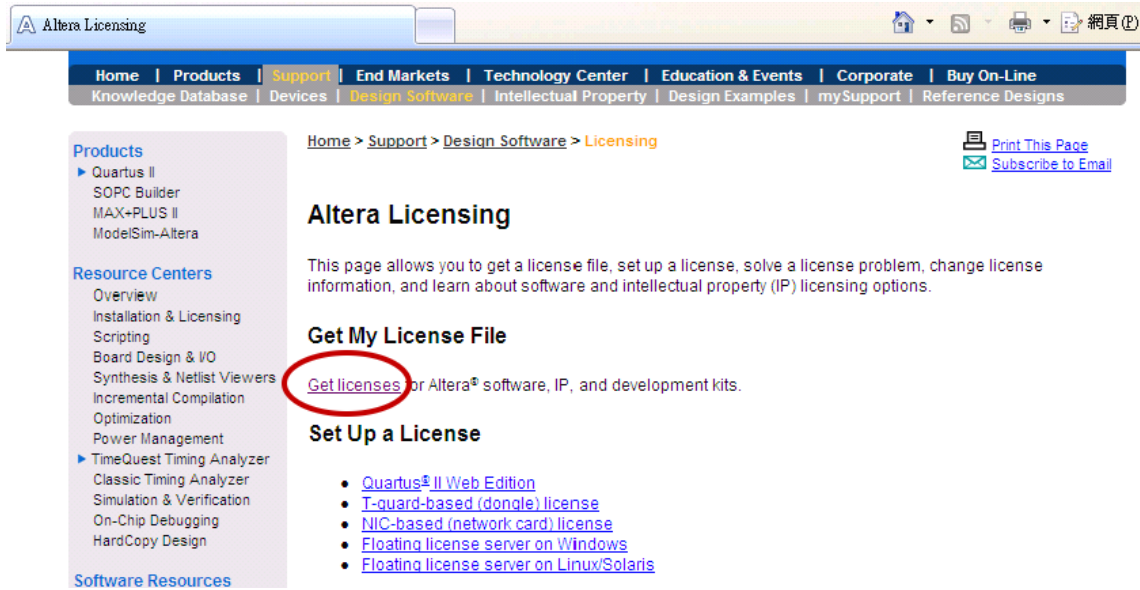


圖 1-8 點選【Get My License File】/【Get licenses】

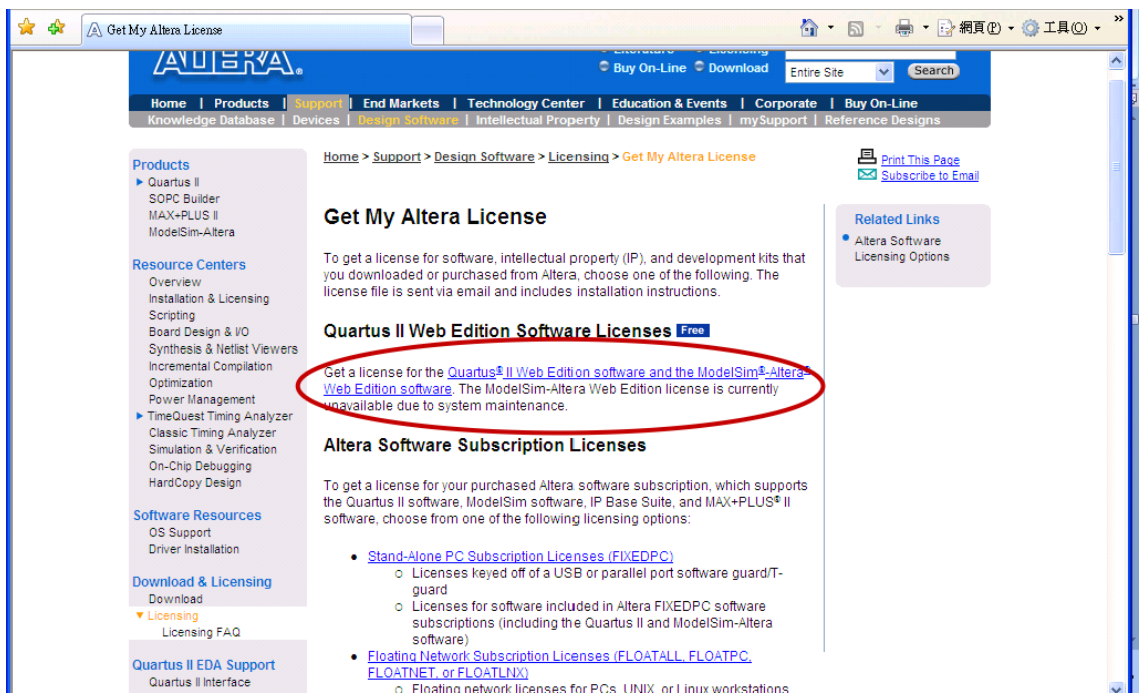


圖 1-9 Quartus II Web Edition Software Licenses



3. 出現 Quartus II Web Edition and ModelSim-Altera Web Edition Licensing 對話畫面，由於我們使用的瀏覽器之前才剛建立個人帳號並執行軟體下載工作，因此此處關於個人資料的部份已無需要再次輸入，但我們仍需輸入註冊資料中的網路卡號碼(network interface card number)。

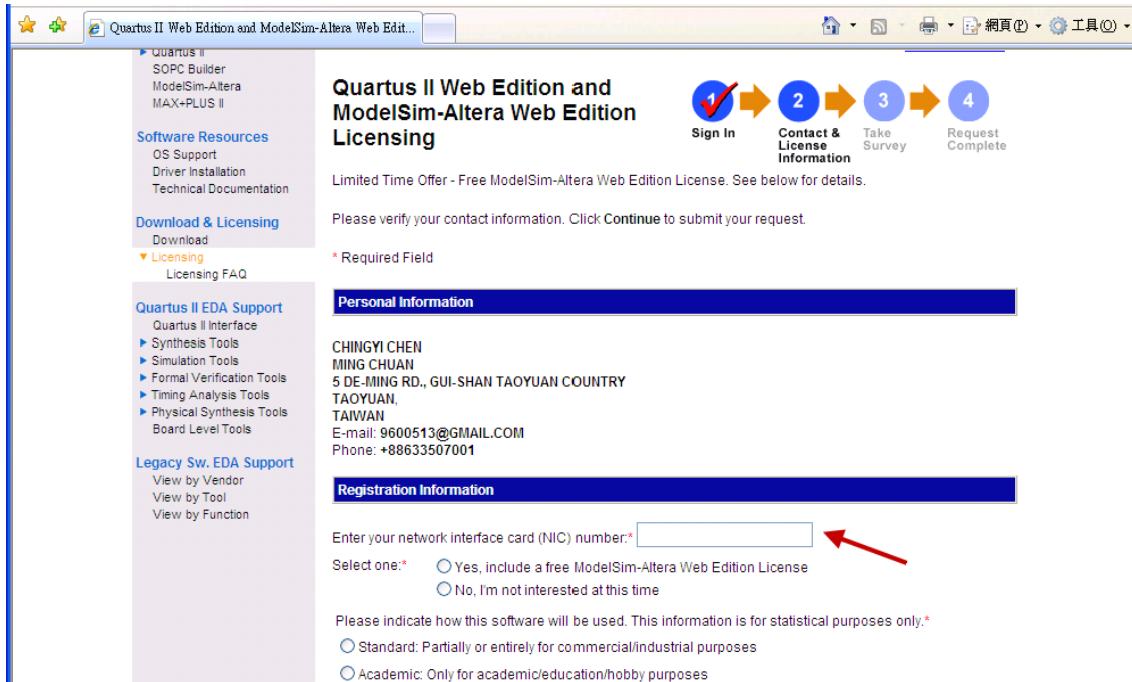


圖 1-10 輸入 network interface card (NIC) number

4. 欲取得自己電腦之 network interface card (NIC) number 有二種方式。第一種方式是在 win XP 下直接查看電腦的網路連線狀態，並在連線狀態視窗下點選【支援】/【詳細資料】，便可看到實體位址。第二種方式則是在作業系統【開始】/【執行】下輸入 cmd 指令以進入 DOS 模式，再輸入 ipconfig -all 指令即可看到 Physical address。



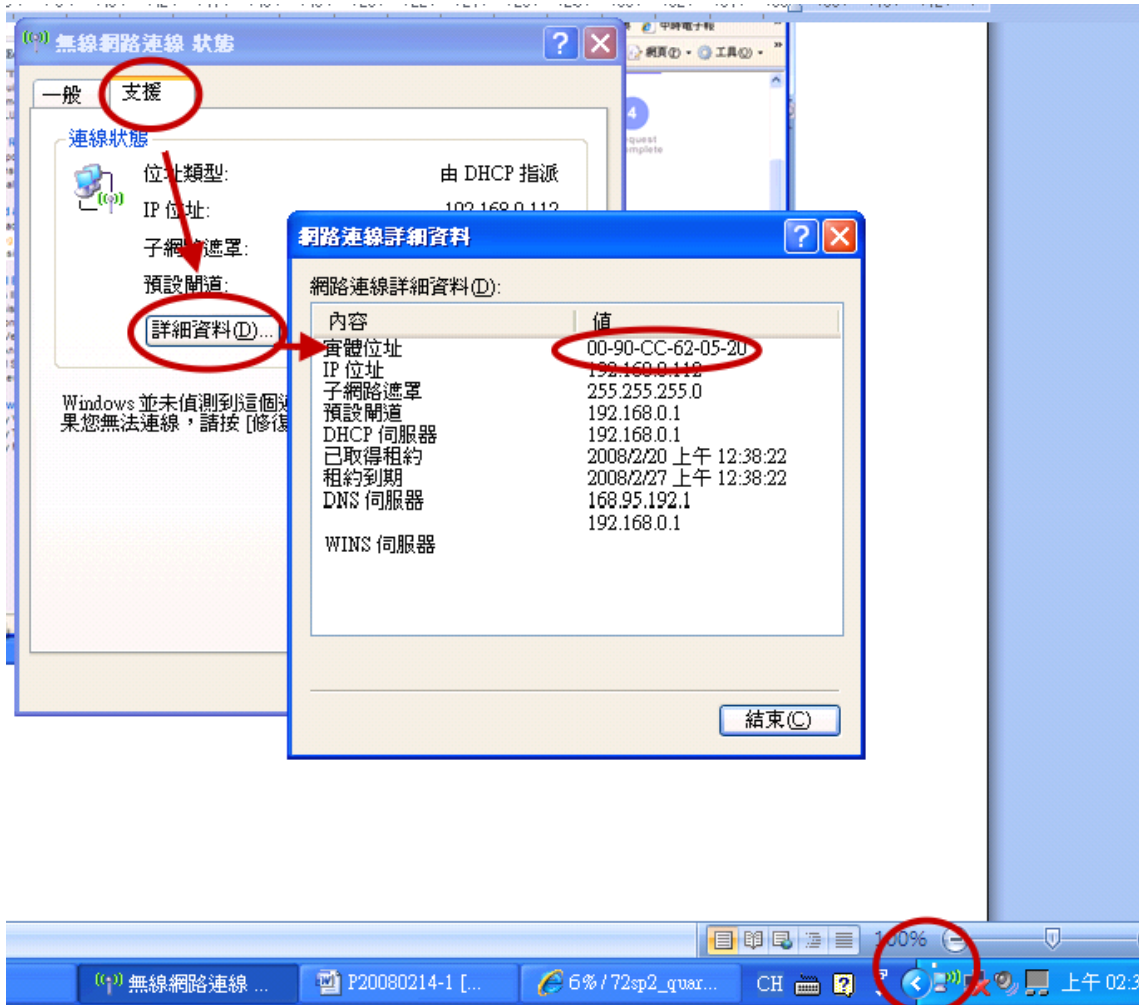


圖 1-11 在視窗作業環境下取得 network interface card (NIC) number





```
C:\WINDOWS\system32\cmd.exe
Ethernet adapter 無線網路連線:

    Connection-specific DNS Suffix  . : 
    Description . . . . . : PLANEX GW-US11H 11Mbps Wireless LAN
USB Adapter
    Physical Address. . . . . : 00-90-CC-62-05-20
    Dhcp Enabled. . . . . : Yes
    Autoconfiguration Enabled . . . . : Yes
    IP Address. . . . . : 192.168.0.112
    Subnet Mask . . . . . : 255.255.255.0
    Default Gateway . . . . . : 192.168.0.1
    DHCP Server . . . . . : 192.168.0.1
    DNS Servers . . . . . : 168.95.192.1
                          192.168.0.1
    Lease Obtained. . . . . : 2008年2月20日 上午 12:38:22
    Lease Expires . . . . . : 2008年2月27日 上午 12:38:22

C:\Documents and Settings\chen>
```

圖 1-12 執行 cmd 指令進入 DOS 模式查看網路連線的實體位址

5. 輸入網路 network interface card (NIC) number 後，自行決定是否要包含免費 ModelSim-Altera Web Edition License 的授權，並回答軟體的使用用途等問題，按【Continue】繼續。

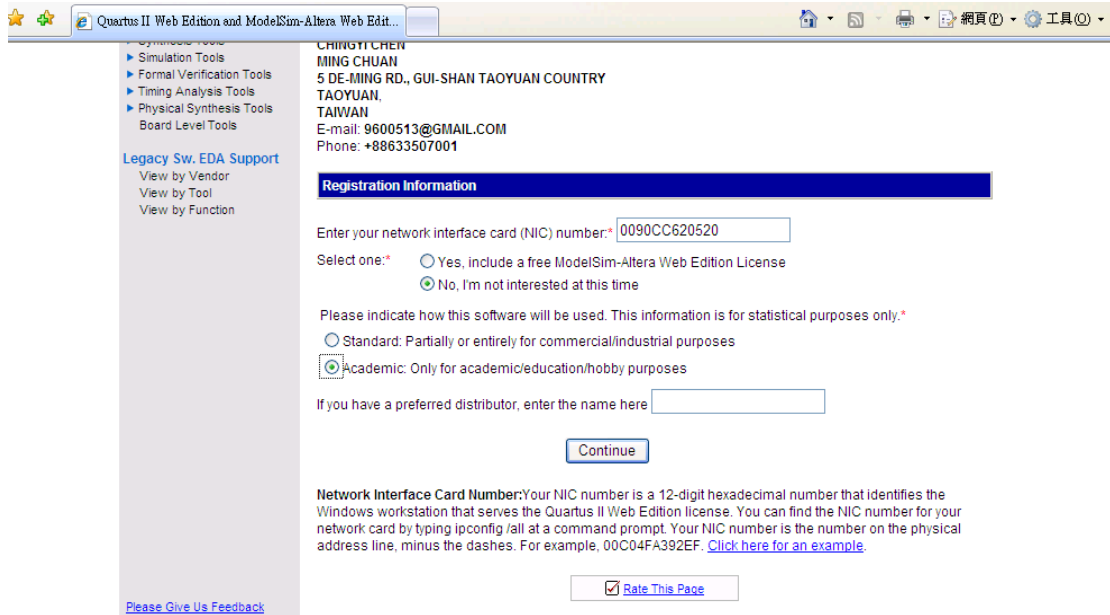


圖 1-13 輸入 NIC 號碼並回答軟體的使用用途

6. 填寫 Quartus II Web Edition Licensing Survey 的問卷調查內容，所有標示“\*”的問題都必須回答才能按【Finish】送出。

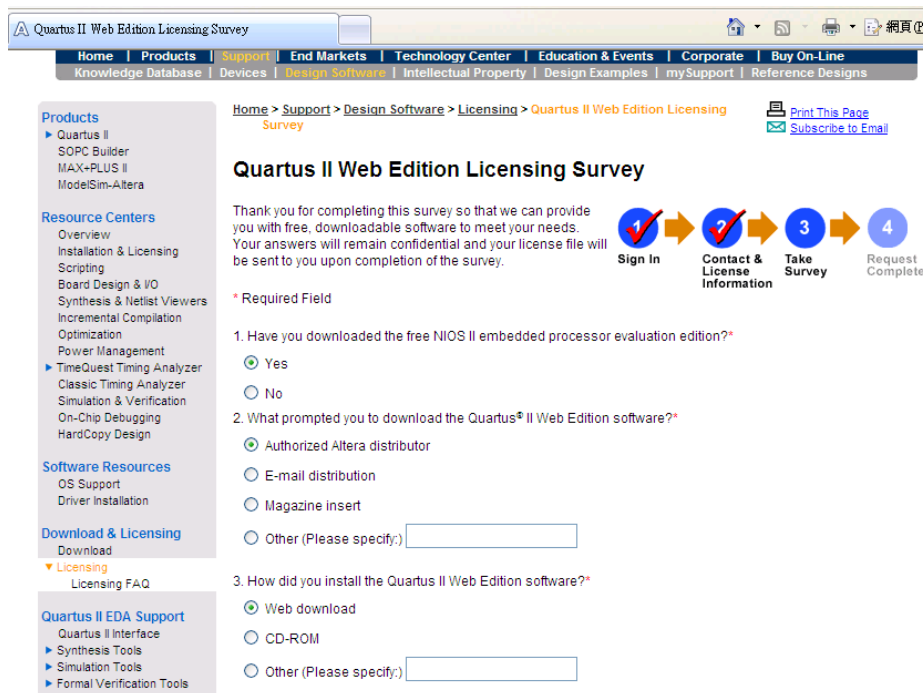




圖 1-14 填寫 Quartus II Web Edition Licensing Survey 問卷調查

7. 完成取得軟體授權流程，並進入當初註冊帳號時所填寫的電子郵件信箱收信，我們很快就會收到一封由 Altera 網站所寄出的郵件。此時我們可將附件中 \*.dat 的授權檔儲存至自己指定的資料夾中（例如 c:\altera\72sp2\quartus），以供 Quartus II 軟體進行 License Setup 之用。

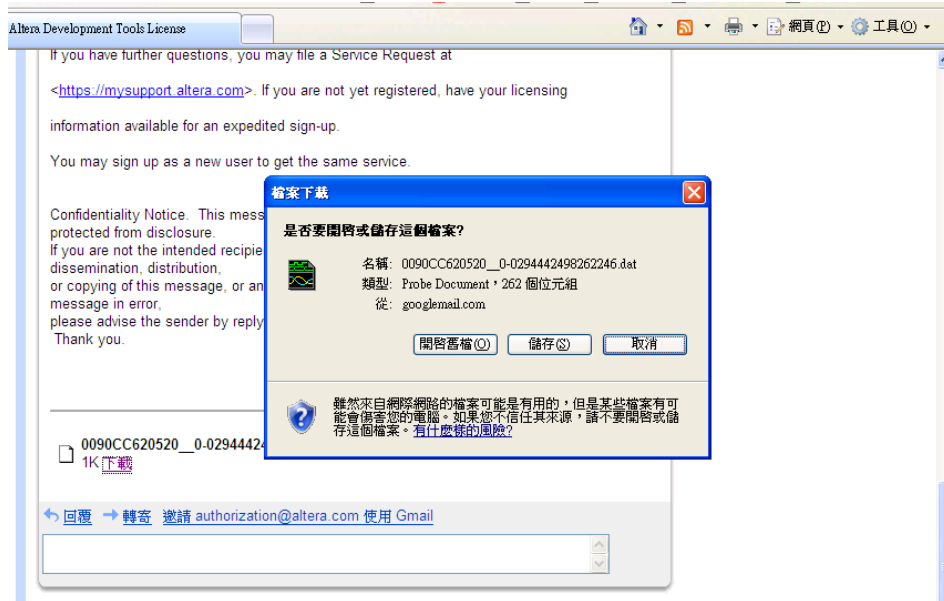


圖 1-15 進入電子郵件信箱收信並將授權檔 (\*.dat) 儲存至指定的資料夾中

8. 若是我們並非在完成下載之後直接進行註冊，那麼也可以在日後再從主網頁登入註冊。首先，我們可在 Altera 首頁按下【Licensing】選項，接下來同樣的進入到 Get My License File，按下 Get licenses，然後再選擇“Get a license for the [Quartus® II Web Edition software and the ModelSim®-Altera® Web Edition software.](#)”。

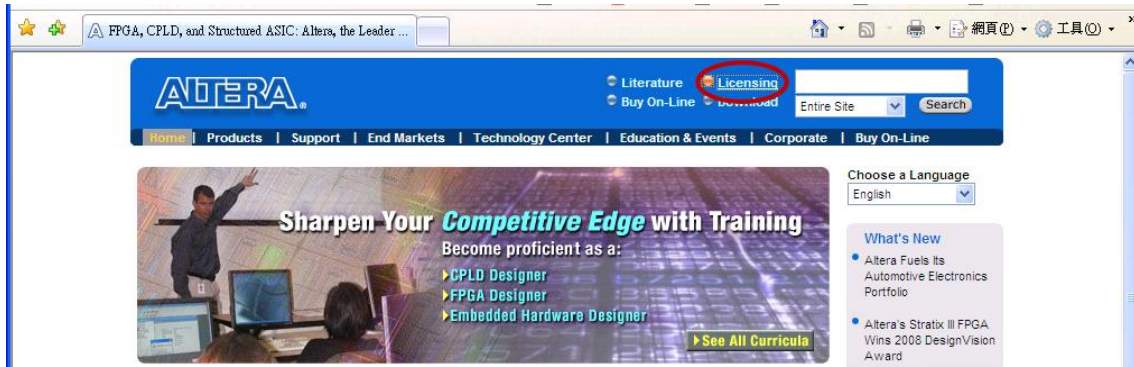


圖 1-16 直接從首頁選擇【Licensing】取得軟體授權

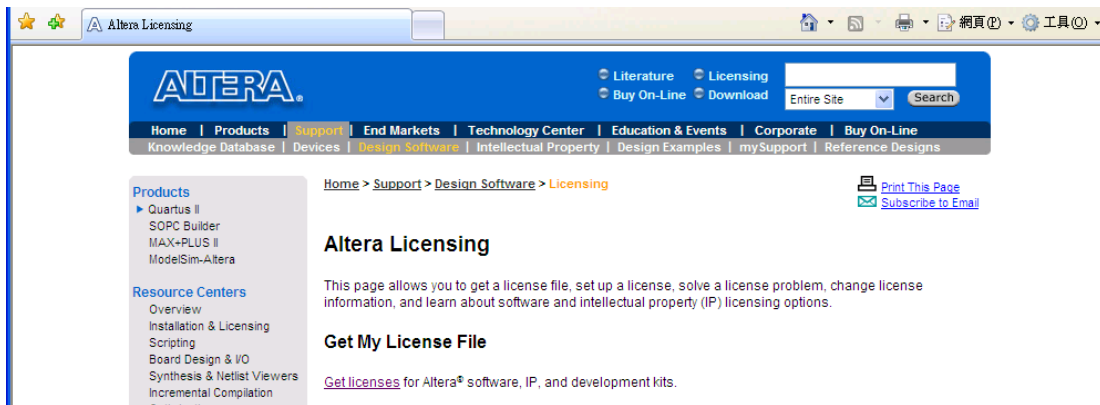


圖 1-17 點選【Get My License File】 / 【Get licenses】

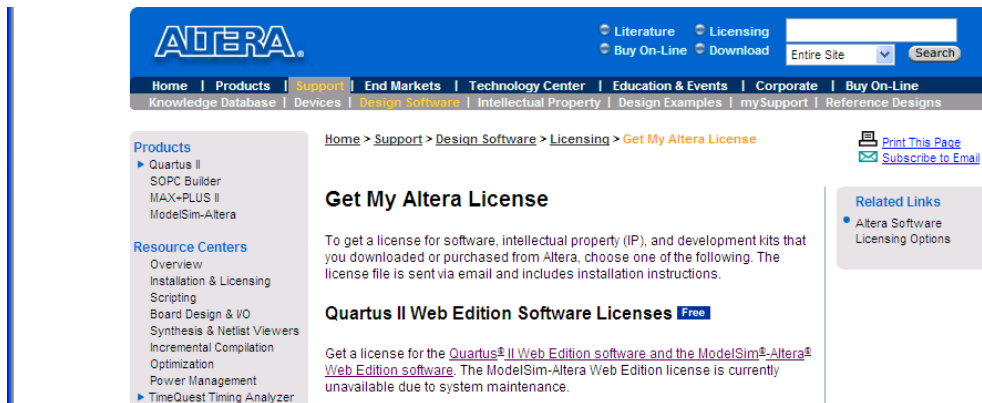


圖 1-18 選擇 Quartus II Web Edition Software Licenses 以取得授權



9. 此處我們選擇在視窗右側輸入所曾建立的帳號和密碼,便可避免一些重新輸入個人基本資料的麻煩。

圖 1-19 直接登入帳號可避免一些重新輸入個人資料的麻煩

10. 出現之前所註冊的個人資訊內容, 確認有標註星號的欄位均已填寫。接下來的執行步驟與前面所述的內容相同。

圖 1-20 出現之前所曾註冊的個人資訊內容

### 1-2-3 安裝 Quartus II 軟體

1. 執行 72sp2\_quartus\_free.exe , 開始安裝 Quartus II。

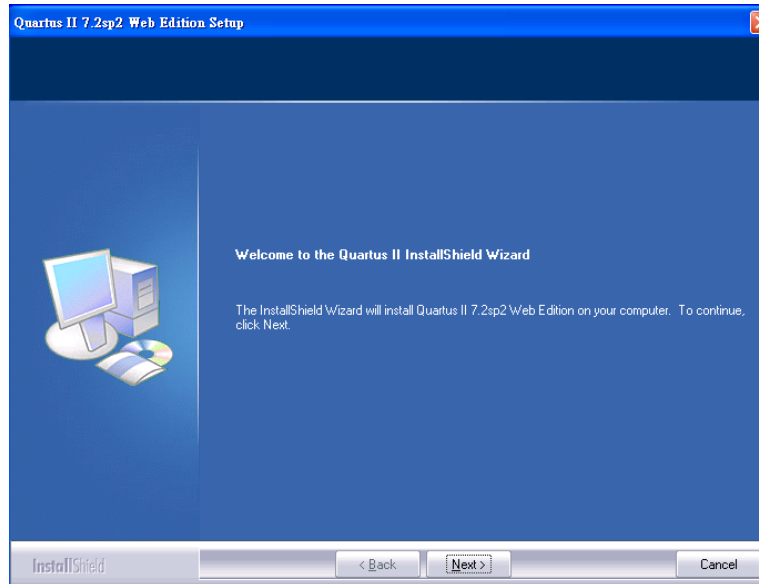


圖 1-21 執行 72sp2\_quartus\_free.exe 安裝畫面

2. 同意授權同意書上之內容。

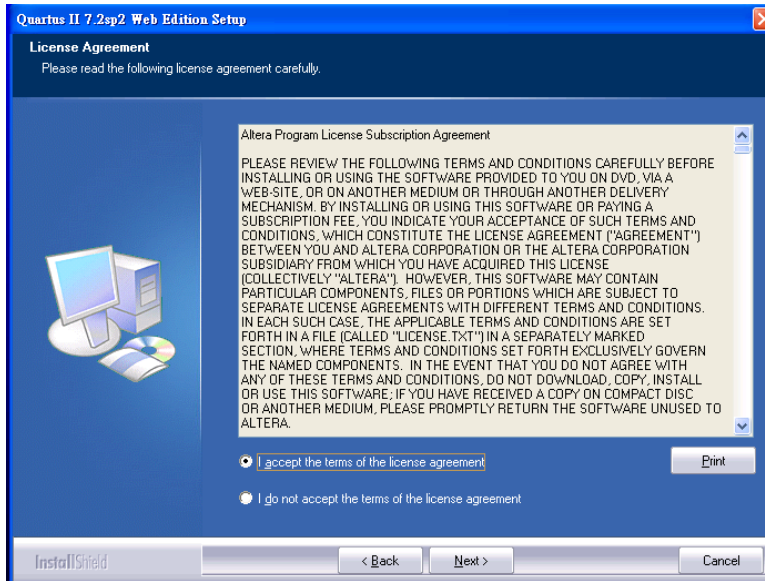


圖 1-22 點選“I accept the terms of the license agreement”

3. 在 Customer Information 視窗中輸入 User Name 和 Company Name 等資訊。

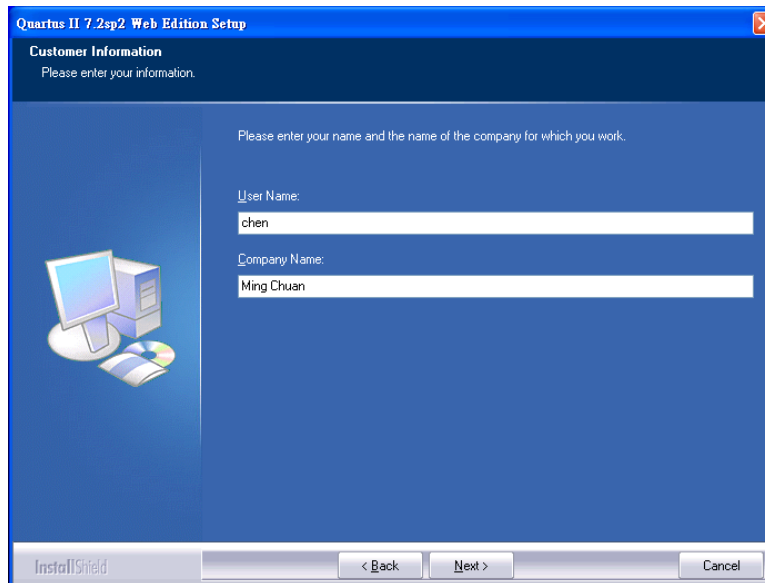


圖 1-23 輸入個人資訊

4. 指定安裝路徑；此處我們直接以預設路徑作為軟體安裝的路徑。



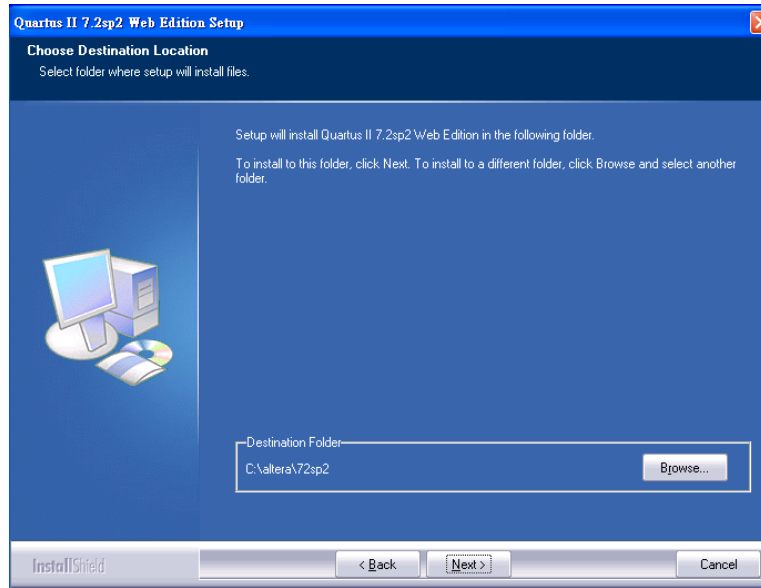


圖 1-24 指定安裝路徑

5. 選擇程式的資料夾之後按【Next】繼續安裝。

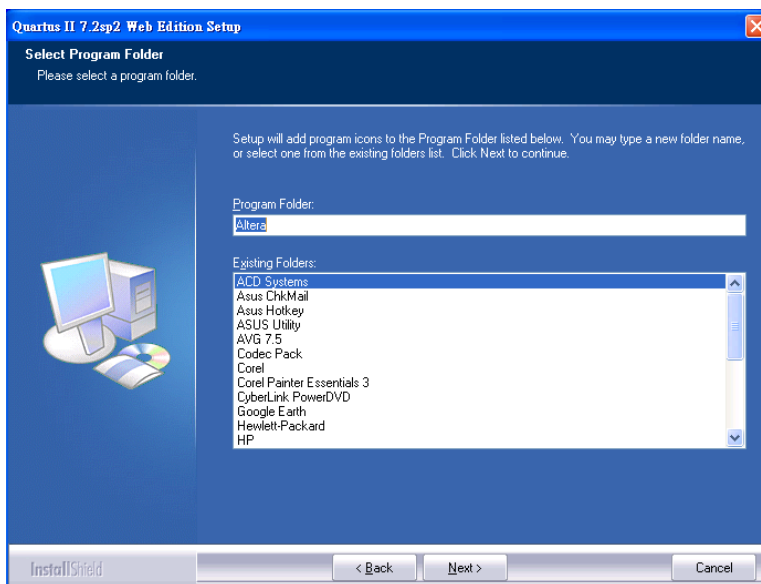


圖 1-25 選擇程式資料夾



6. 接著出現選擇安裝型式的視窗，我們選擇“Complete 後”按“Next”繼續安裝。

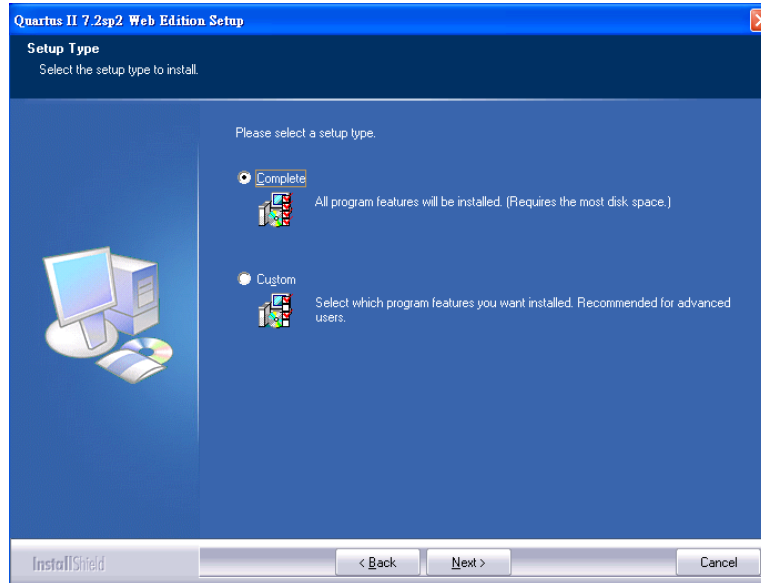


圖 1-26 選擇安裝型式

7. 安裝程式開始執行前會顯示所有的安裝設定內容讓我們作最後的確認。

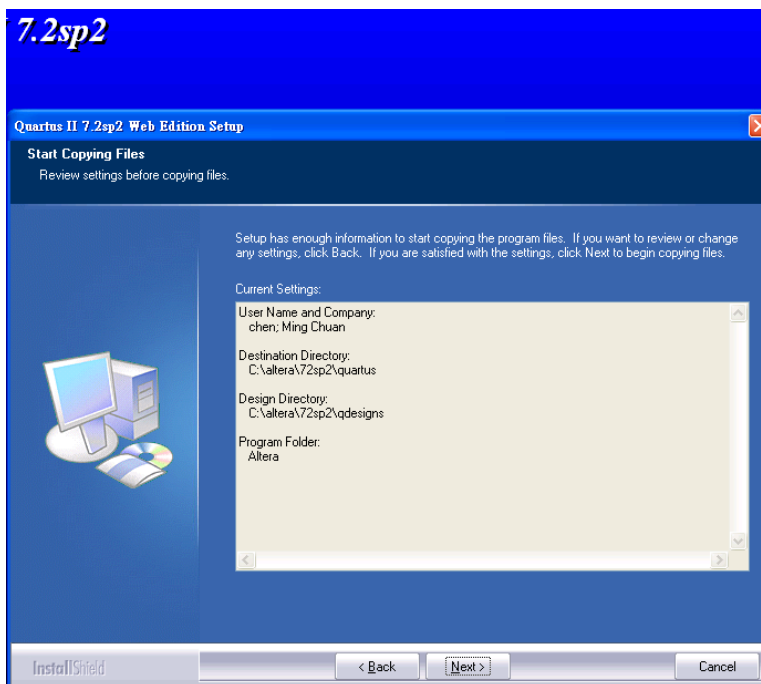


圖 1-27 顯示安裝設定

8. 開始執行安裝程式。

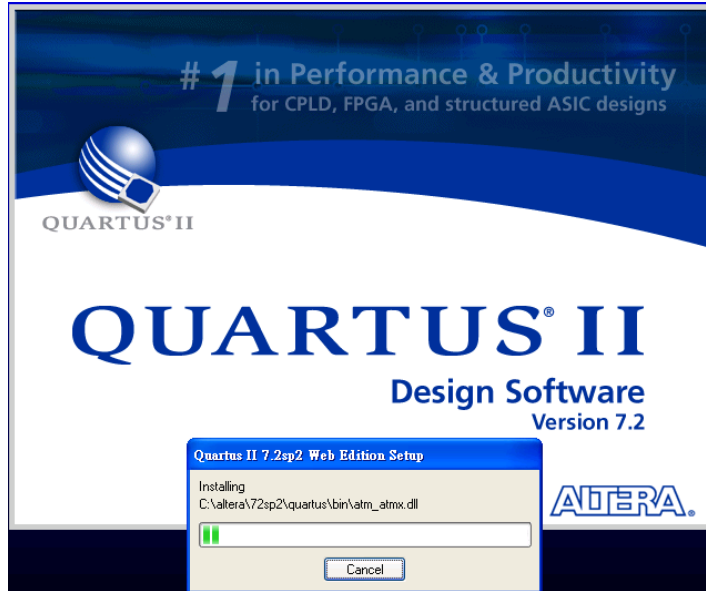


圖 1-28 開始執行安裝程式

9. 安裝結束，可自行選擇是否要查看“readme.txt”。

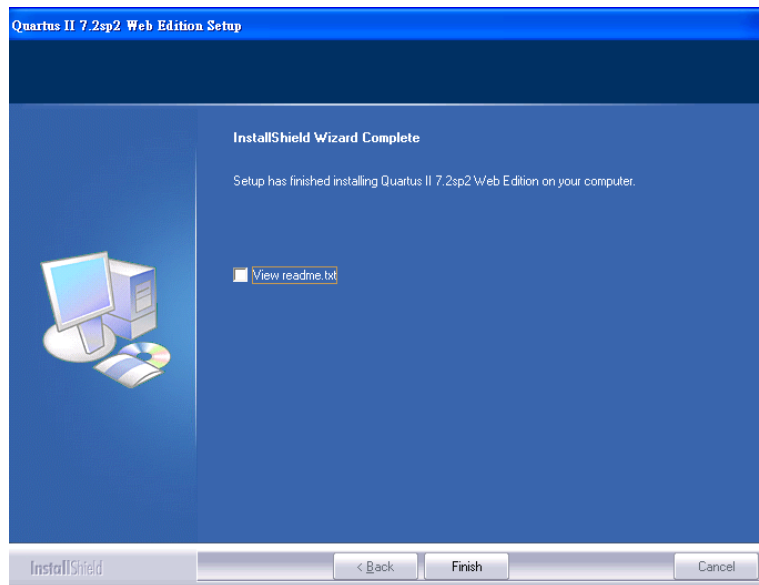




圖 1-29 安裝完成

10. 執行 Quartus II。第一次操作軟體時，會出現詢問使用者有關操作介面選擇的對話視窗，使用者可依操作習慣選擇 Quartus II 或傳統 MAX+PLUS II 的操作環境。

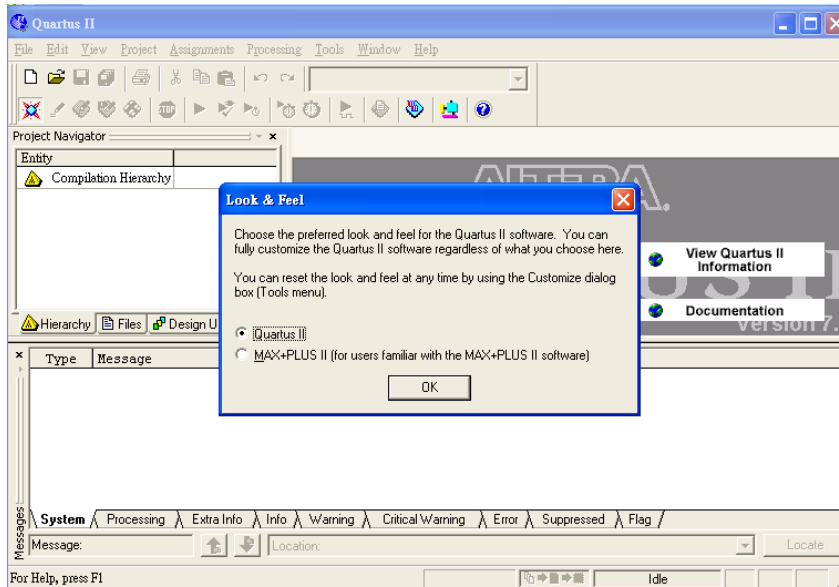


圖 1-30 開啟 Quartus II 畫面

11. 接著出現 License Setup Required 對話視窗，由於我們之前已經上網取得授權檔(\*.dat)，因此我們可以指定該授權檔之正確路徑。

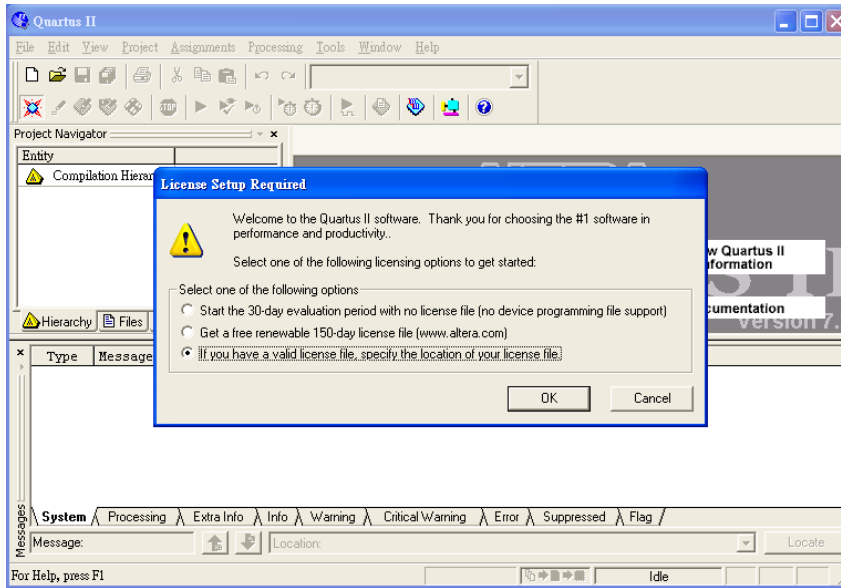


圖 1-31 選擇指定正確的軟體授權檔

12. 指定正確授權檔儲放的路徑。本例中，我們已先行將之前 e-mail 所收到的授權檔改存至 C:\altera\72sp2\quartus 路徑下，所以我們必須將 License file 的欄位指定到這個位置才能找到授權檔。

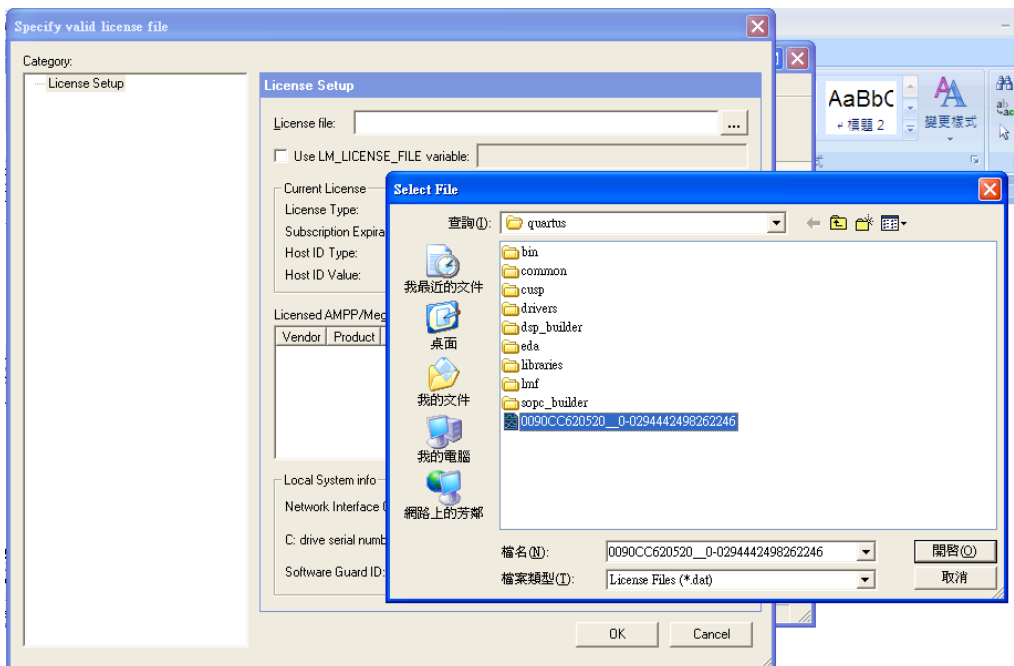




圖 1-32 指定正確授權的路徑

### 13. 安裝完成。

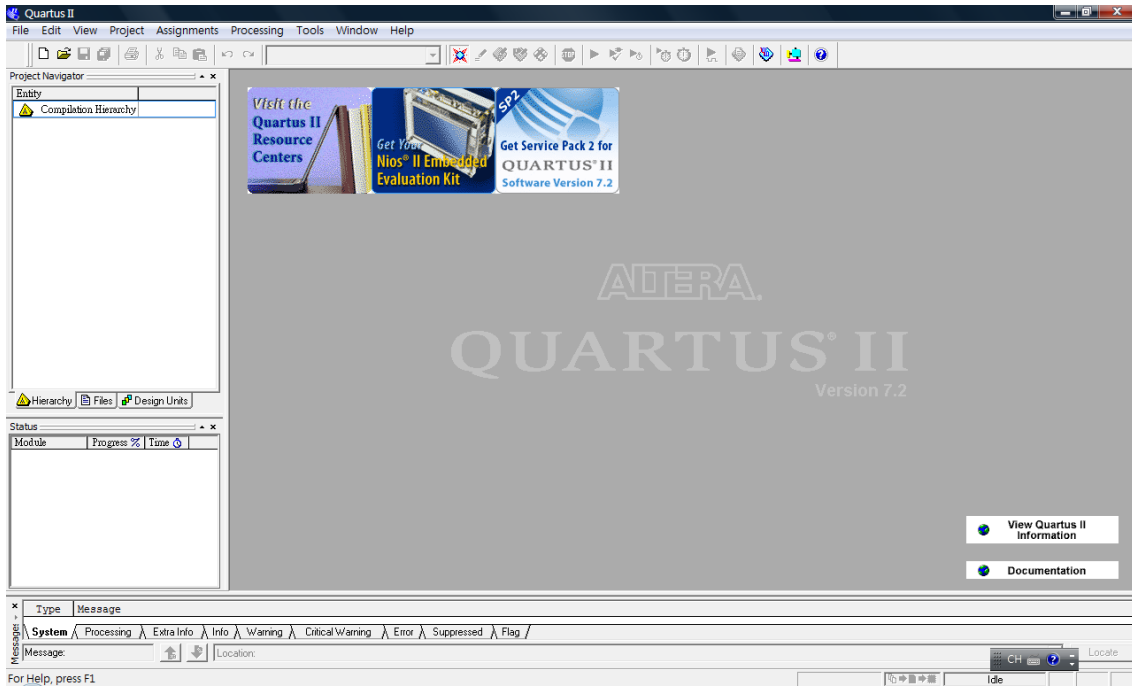


圖 1-33 Altera Quartus II Version 7.2 操作畫面

## 1-3 在 XP 和 Win2000 下安裝 Byteblaster

在 Windows98 作業系統下，我們並不需另外安裝 ByteBlaster 驅動程式即可使用 Parallel Port 下載所編譯完成的檔案；但在 WinXP 或 Win2000 作業系統下，我們在安裝好 Altera 開發軟體之後，預設環境並不支援 ByteBlaster (LPT1) 下載，此時我們必須另外再安裝驅動程式才能使用 LPT1 介面進行實驗板之晶片燒錄，其驅動程式可在以下的目錄中找到： ..\altera\72sp2\quartus\drivers\win2000\win2000.inf。

### 1-3-1 Windows XP 下安裝 ByteBlaster 驅動程式的作法

1. 選擇【開始】/【設定】/【控制台】/【印表機和其他硬體】，按下【新增硬體】，出現新增硬體精靈，選取【下一步】。

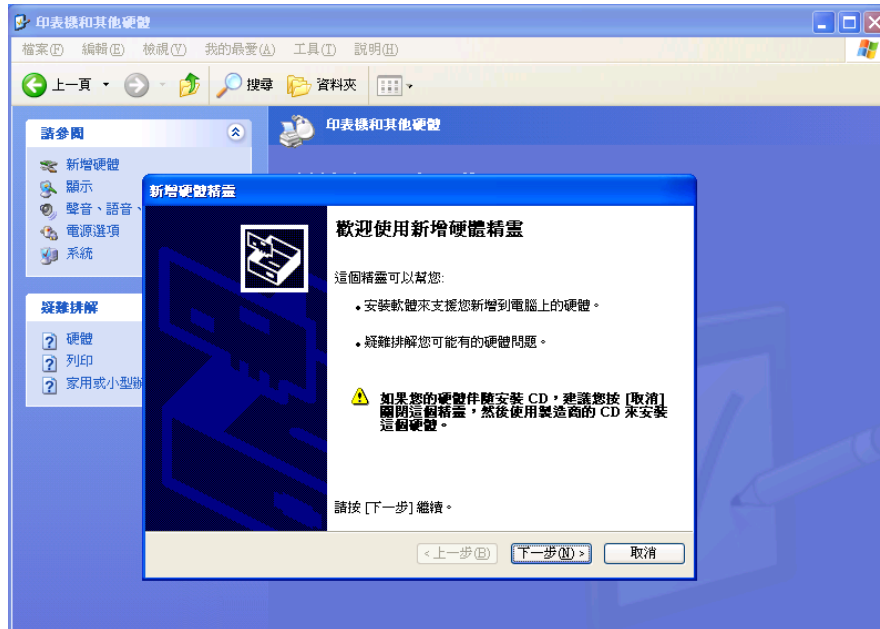


圖 1-34 使用新增硬體精靈

2. 選擇【是，我已連接這個硬體】，並按【下一步】。



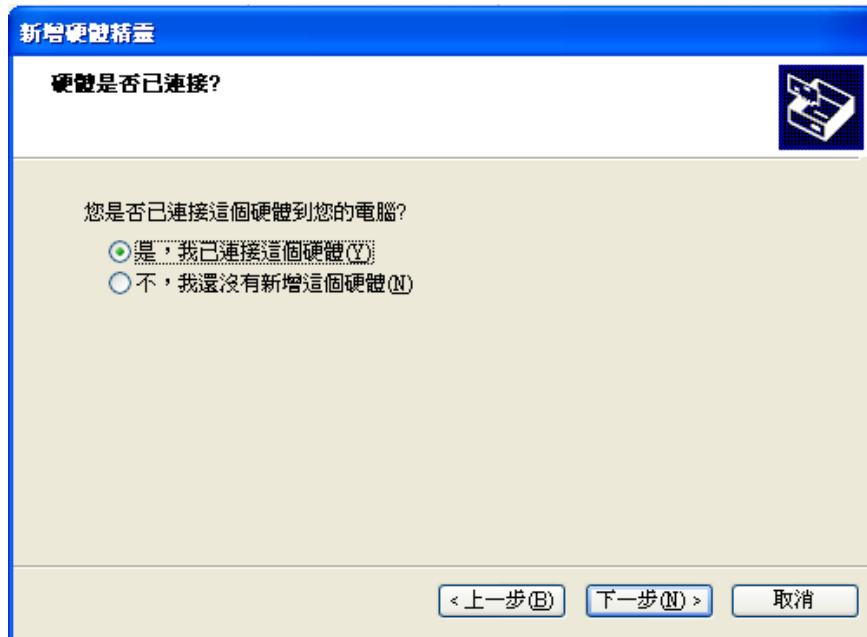


圖 1-35 選擇【是, 我已連接這個硬體】

3. 在新增硬體精靈視窗中選擇【新增硬體裝置】，並按【下一步】。



圖 1-36 選擇【新增硬體裝置】

4. 選擇【安裝我從清單中手動選取的硬體(進階選項)】，並按【下一步】。

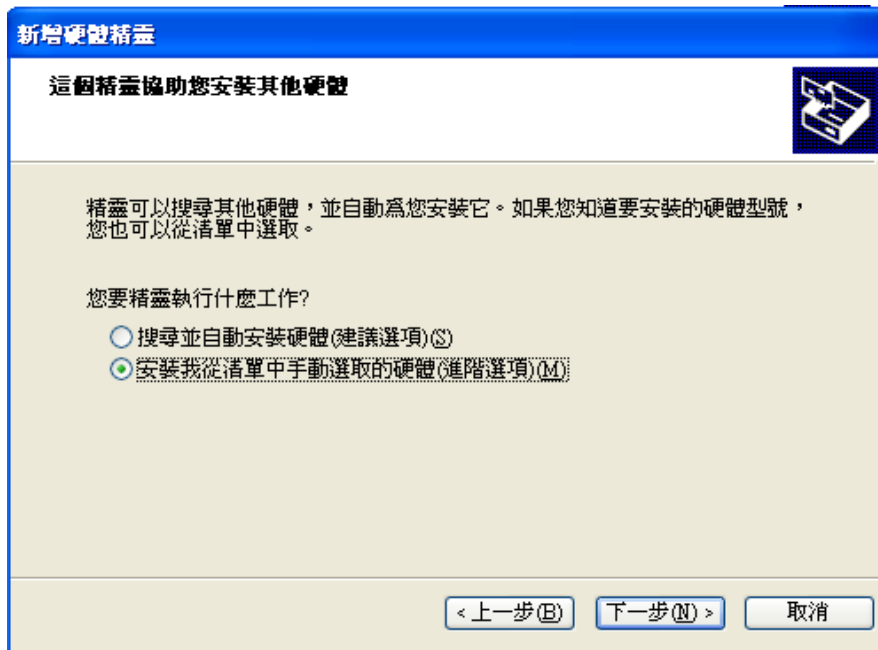


圖 1-37 選擇【安裝我從清單中手動選取的硬體(進階選項)】

5. 選擇【音效，視訊及遊戲控制器】，並按【下一步】。





圖 1-38 選擇【音效，視訊及遊戲控制器】

6. 選擇【從磁片安裝】。

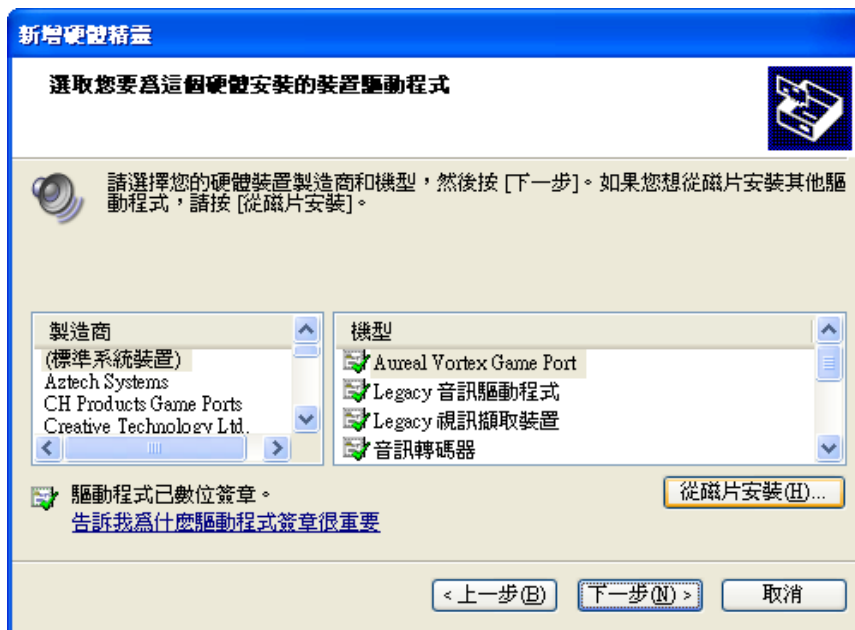


圖 1-39 選擇裝置驅動程式

7. 按【瀏覽】，選擇驅動程式所在位置的資料夾。

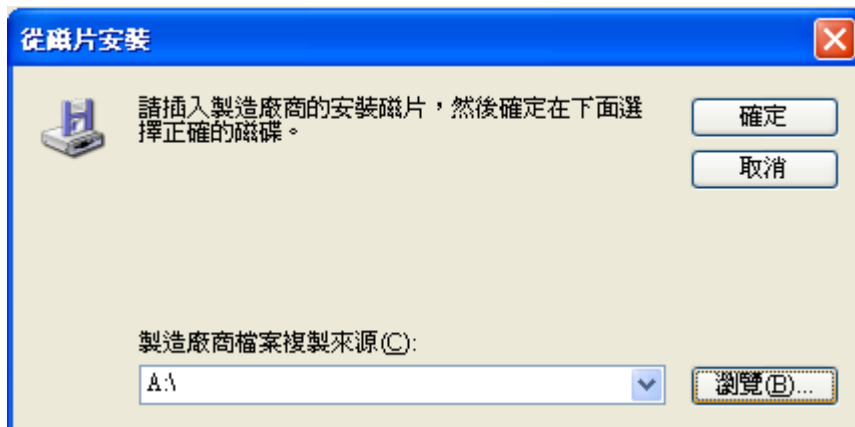


圖 1-40 選擇驅動程式所在位置的資料夾

8. 至『...\altera\72sp2\quartus\drivers\win2000』目錄中找到 Win2000.inf 檔案後，選取【確定】。

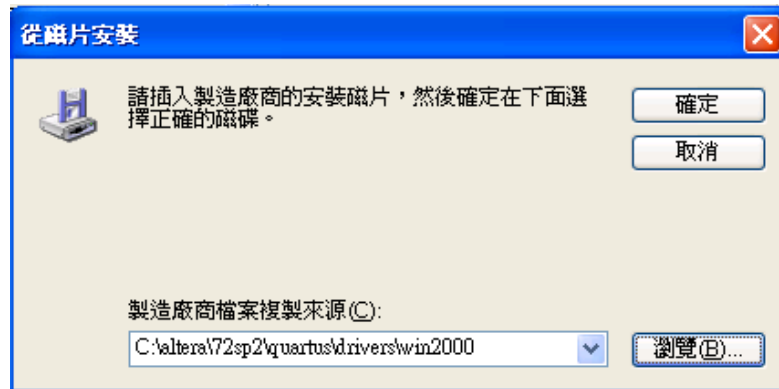


圖 1-41 找出 Win2000.inf 檔案所在位置

9. 按下【繼續安裝】按鈕。

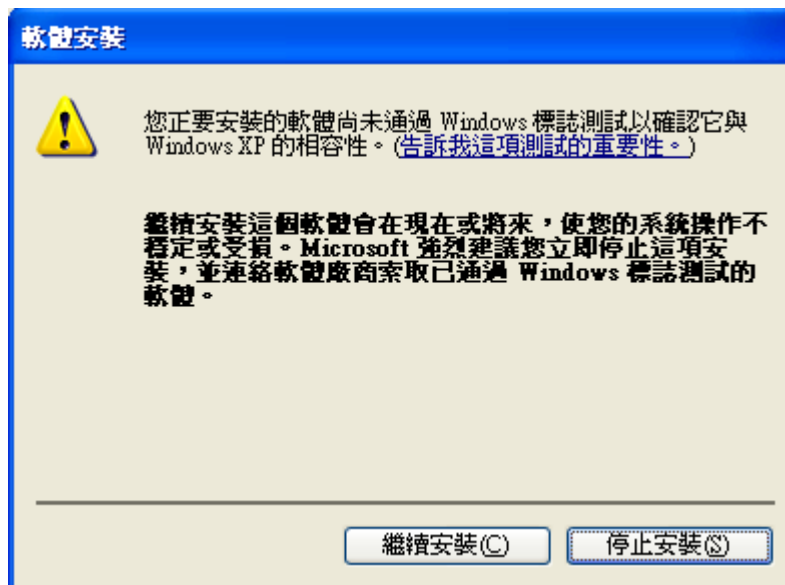


圖 1-42 繼續安裝

10. 選取要為這個硬體安裝的裝置驅動程式：【Altera ByteBlaster】。



圖 1-43 選取要為這個硬體安裝的裝置驅動程式

## 11. 按【下一步】開始安裝新硬體。

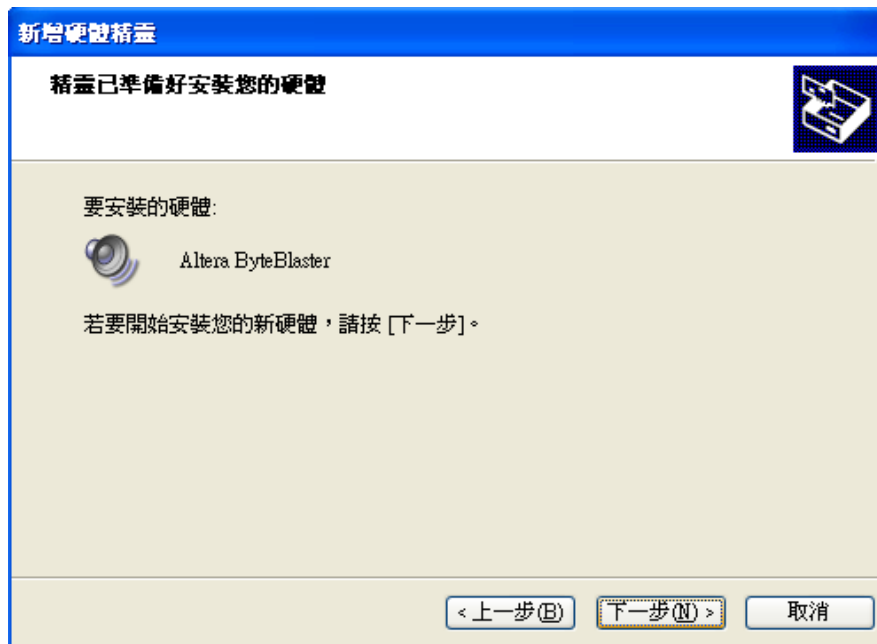


圖 1-44 按【下一步】開始安裝新硬體

## 12. 選擇【繼續安裝】。

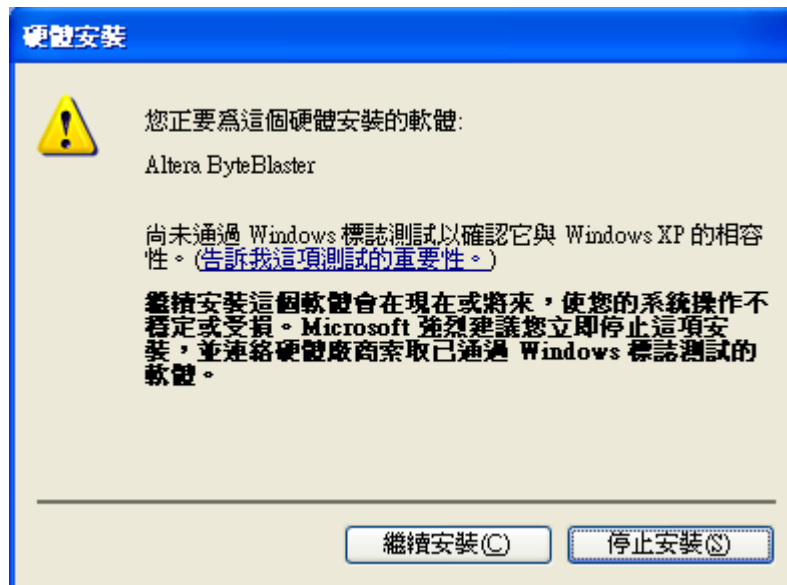


圖 1-45 繼續安裝

## 13. 按【完成】以關閉新增硬體精靈。



圖 1-46 完成新增硬體精靈



## 1-3-2 Win2000 下安裝 ByteBlaster 驅動程式的作法

1. 在【控制台】/【新增/移除硬體】項目中新增一項裝置。

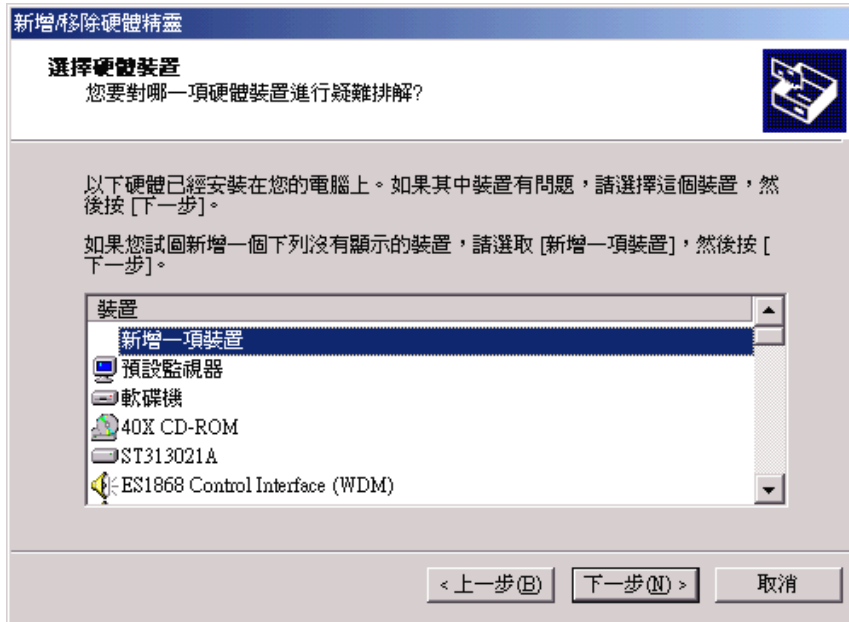


圖 1-47 新增硬體

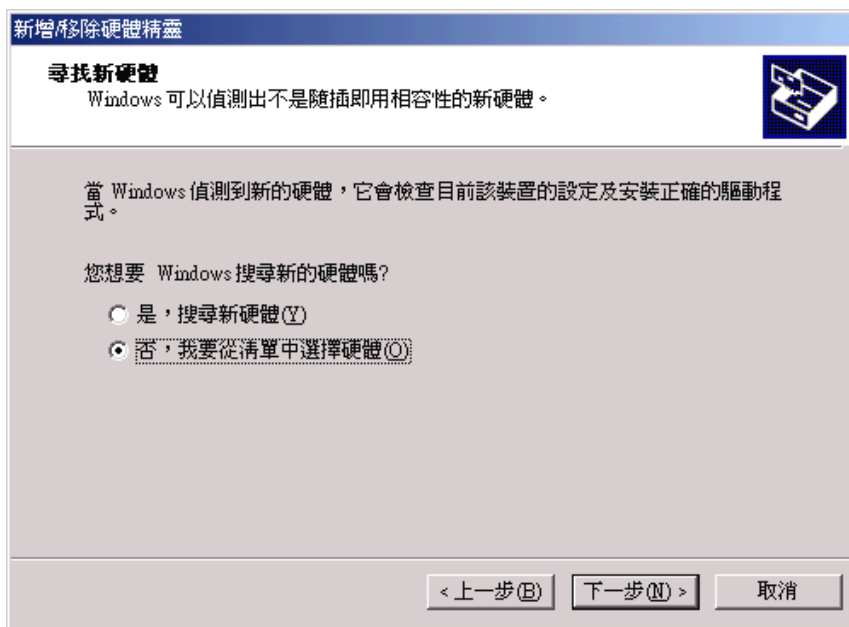




圖 1-48 從清單中自行選擇硬體

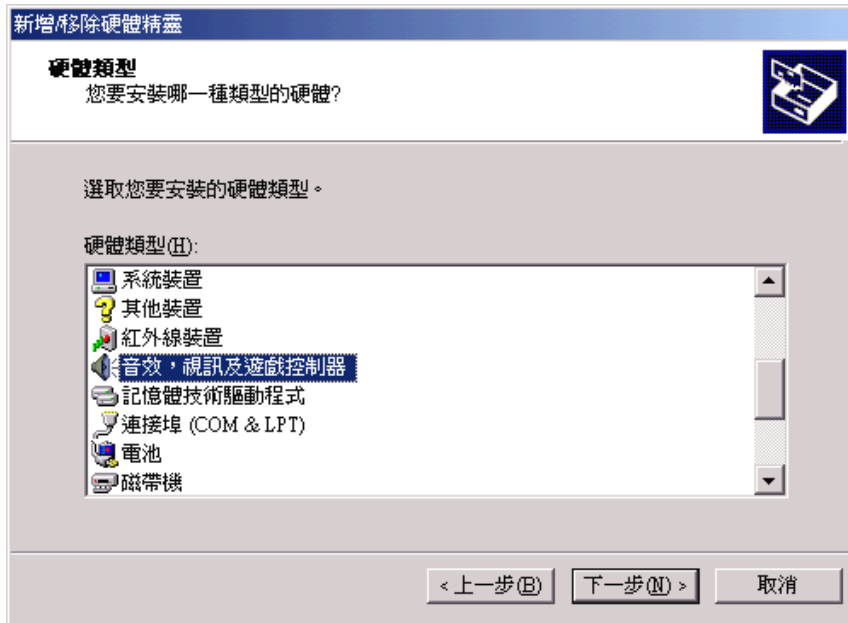


圖 1-49 選擇硬體類型

2. 選擇【從磁片安裝】，並提供 win2000.inf 的正確路徑。

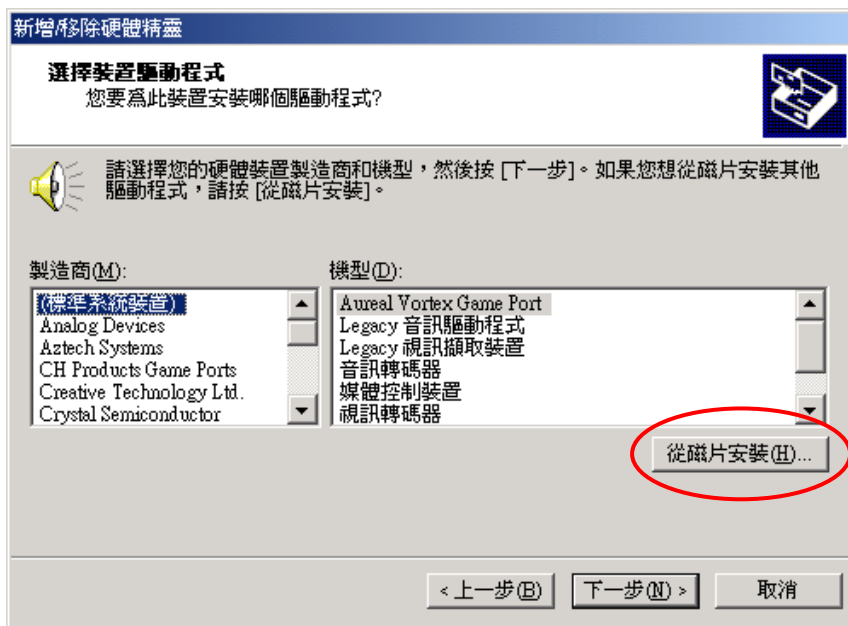


圖 1-50 選擇從磁片安裝

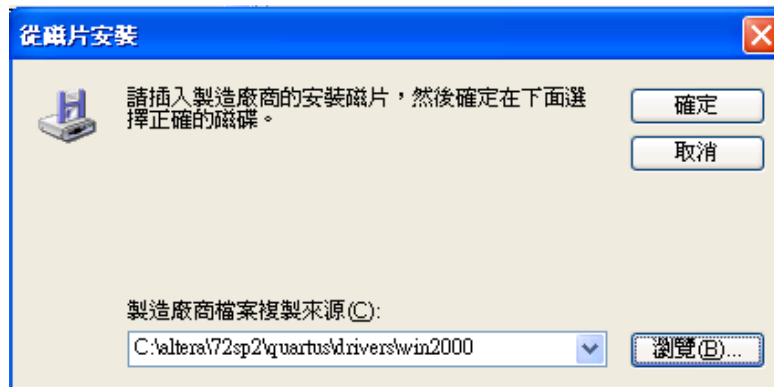
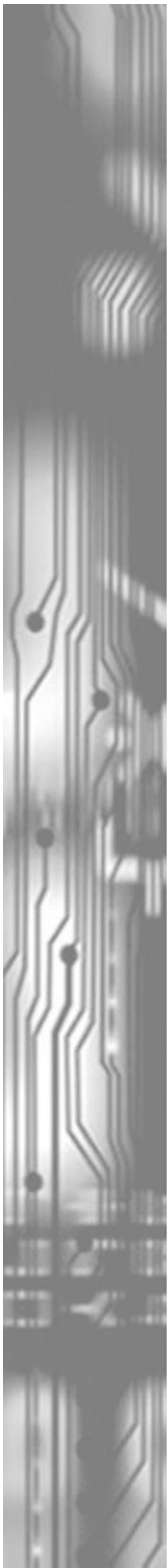


圖 1-51 選擇正確的檔案路徑

3. 完成安裝程序後選擇重新開機即可。

# 第三章

## VHDL 語言的 基本概念





## 3-1 什麼是 VHDL

1980 年代初期，IC 設計產業面臨新情勢的挑戰，由於半導體技術呈現爆炸性的成長，如何縮短產品設計時間，如何將更複雜龐大的系統功能設計在晶片中，傳統的邏輯設計方法顯然在面對這些課題時，已經顯出明顯的侷限性。產業界和學術界都了解到必須有一種新的電路設計的型態和工具，以便協助電子工程師設計出更複雜、更多邏輯閘數目的晶片。就在此一時機，美國國防部（The United States Department of Defense）提出一個名為極高速積體電路（VHSIC, Very High Speed Integrated Circuit）的計畫，其目的在於規劃一個標準化的電路描述語言，使所有軍事零件承包商的電子電路文件記錄便於管理和重複使用。

計畫初期以邏輯閘層次（Gate Level）的方式描述電路，然而當設計者面對高達數萬個甚至數十萬個邏輯閘以上的電路時，整個設計工作就變成了一項極其複雜而且非常具挑戰性的艱鉅任務；事實上，這甚至可能比直接徒手繪製電路還要困難，因此有必要發展新的描述方法來降低設計的困難度。於是，改良的電路描述方式 - “VHSIC 硬體描述語言”（VHSIC Hardware Description Language）便在 1982 年順應而生，名稱再次的簡化就成了 VHDL 的稱呼。其後，在 1987 年 VHDL 成為國際電機電子工程協會（International Electrical & Electronic Engineering, IEEE）的一種標準語言，叫做 IEEE 1076。到了 1988 年時，美國國防部規定所有官方的 ASIC 設計均要以 VHDL 為其硬體描述語言，自此之後 VHDL 也漸漸成為業界間流通的一種標準。現在 VHDL 已經成為「電子設計自動化」（EDA）工程的共通語言；透過它，不同工程師、公司、軟體工具可以精準地傳遞數位設計理念，以高階電路描述語言的方式輕易而快速的實現複雜電路的設計規格，現今電子工業的數位「矽智財」（SIP, Silicon Intellectual Properties）幾乎都是以這類 HDL 語言設計出來的。

## 3-2 VHDL 的基本架構

目前常用的數位硬體描述語言主要有兩大主流：VHDL 和 Verilog-HDL，而 Altera 公司另外也自行提供了 AHDL 硬體描述語言，我們將在本章簡單的介紹 VHDL 架構和語法。

一個基本的 VHDL 程式除了 Library 外，還包括單體 ( Entity ) 和架構 ( Architecture )。Library 是一個目錄區，它可以解譯我們程式中所宣告使用的各單位。電路單體 ( Entity ) 在 VHDL 語言中只是用來描述階層式方塊的界面 ( Interface )，即描述電路設計中的輸出和輸入訊號的宣告；架構描述 ( Architecture description ) 則是用來描述電路單體的內部結構或行為，它有三種主要的描述方式，分別是結構性描述 ( Structure Description )、資料流型式 ( Data flow ) 和行為性描述 ( Behavior Description )。

一個 VHDL 程式的基本架構如下：

```
Library ieee;
Use ieee.std_logic_1164.all

entity 晶片名稱 is
  port
  (
    宣告輸入和輸出接腳
  );
end 晶片名稱 ;

architecture 架構名稱 of 晶片名稱 is
  begin
    ...
    VHDL並行敘述 (concurrent) 指令
    ....
end 架構名稱 ;
```



### ◆ 單體 ( Entity ) :

在較為傳統的電腦輔助設計軟體中，使用者必須以基本的電路元件(如 AND 閘、OR 閘或正反器)來組織建構所需的邏輯功能，而這些最基本的電路元件或暫存器在 CAD 軟體中均以具有輸入、輸出腳位的電路符號方塊圖來表示，使用者透過各電路元件之間介面腳位的連線佈局來實現電路的功能設計。

在 VHDL 語言中，我們以 ENTITY 的描述來表示一個傳統線路中的電路符號 (SYMBOL)，單體(Entity)用來宣告電路的外部包裝，也就是設計體的介面(Interface)，我們在這裡定義出所有的外接腳。單體(Entity)所描述的「介面」(Interface)是外界和設計實體連接的單元，外部系統只認得它；所以，它具有「設計什麼？」的目的。ENTITY 在 VHDL 中是構成所有設計的基礎，設計者在 ENTITY 中必須完整定義一個設計物中所有的輸入和輸出信號。在一個階層式的設計中，最頂層的設計考量就是整體電路的 ENTITY，之後往下延伸的每一階層線路的描述，則各自包含在不同底層電路的 ENTITY 之中。

VHDL 語法中 ENTITY 描述的「介面」是以「Entity...IS...End」來表示。下圖是一個 D 型正反器的電路元件以及其 ENTITY 的宣告內容，CLK 和 D 是輸入腳位 (PORTS)，Q 是輸出；這幾個信號都是屬於 STD\_LOGIC 的位元型式。

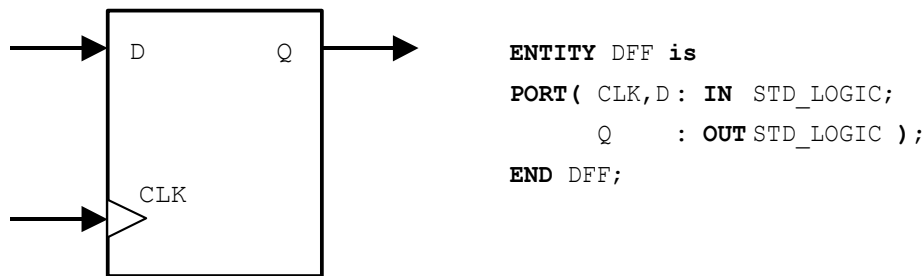


圖 3-1 D 型正反器與其單體(Entity)宣告

同樣的，我們再以一個二輸入的 NAND 閘為例，在這個電路的單體 ( ENTITY ) 中，我們宣告 A 和 B 是輸入腳位 ( PORT )，C 是輸出；而且這幾個信號都是屬於最基本的邏輯位元(bit)型式。

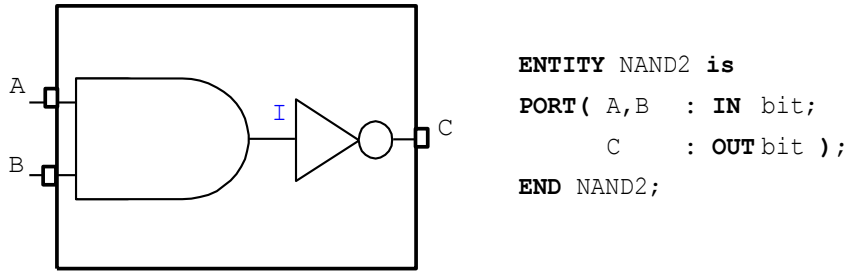


圖 3-2 二輸入的 NAND 閘與其單體(Entity)宣告

【註】在 VHDL 程式的單體(ENTITY)中，可以使用的輸出入腳位 (埠，PORT)共可分為四種模式：

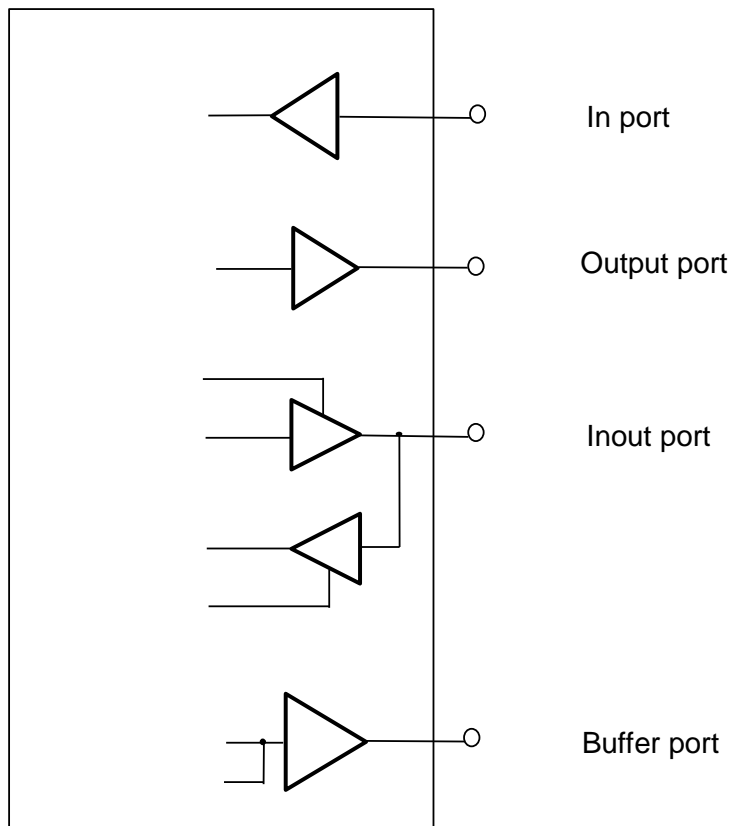


圖 3-3 Entity 周圍腳位的四種模式

1. **In**：表示該腳位要從外界接收信號，即輸入模式的 Port 訊號。
2. **Out**：該腳位將傳送信號到外界，即輸出模式的 Port 訊號。





3. **Inout** : 可收送雙向 ( Bi-direction ) 模式的 Port 信號，但使用 Inout 來宣告訊號會造成程式的可讀性變差。
4. **Buffer** : 表示緩衝模式的 Port 訊號；該腳位是輸出，但它允許其回授到其內的電路。

### ◆ 架構 ( Architecture ) :

架構(Architecture)是用來描述 ENTITY 所涵蓋所有電路行為特性的主要部份，其內容就是設計實體內的數位運算單元，外部系統看不到它；因此，相對於 ENTITY 的功能而言，它則是具有「設計如何運作？」( How a Design Operates ) 的目的。就系統分析 ( System Analysis ) 的角度來看，ENTITY 所描述的「介面」是電路的「規格」( Specification )，而架構(Architecture)則是「實現」( Implementation )。在 VHDL 語言中，架構(Architecture)是以「Architecture..of..IS...Begin...End」來表示；它在起頭處必須先宣告它是屬於那一個 ENTITY，並以 Begin 和 End 作為程式的起始和結束。關於 Architecture 的描述風格一般可以歸納成三種型式：

#### 1、結構性描述(Structure Description) :

這種描述方式相當於在描述電路中各元件之間的連線關係，故又可稱為連線關係描述風格 ( Netlist Description Style )。以前面所述的 NAND 閘為例，該電路是由一個 AND 元件(Component)以及一個 NOT 元件所共同組成(其電路的內部連接節點是 I)，因此其架構若以結構性風格描述可撰寫如下：

```
ARCHITECTURE structure OF NAND2 IS
Signal I:BIT;
  component AND_2                                --二輸入的NAND元件與其腳位宣告
  port ( I1,I2 : in bit;
        O1   : out bit );
  end component;
  component INVERTER                             --NOT元件與其腳位宣告
  port ( I1  : in bit;
        O1  : out bit );
  end component;
BEGIN
```

```
Cell1:AND_2 port map(I1=>A, I2=>B, O1=>I); --NAND元件腳位之對應連線關係
Cell2:INVERTER port map(I1=>I, O1=>C); --NOT元件腳位之對應連線關係
END structure;
```

## 2、資料流描述 ( Dataflow Descriptions ) :

資料流型式的 VHDL 電路架構描述風格主要是利用布林方程式來表現各信號之間的布林代數關係，使用者在不使用順序性(Sequential)敘述的條件下，定義出資料在信號間或資料在輸入、輸出間的傳遞關係。Behavioral Description 和 Dataflow Description 之間最大的差異就在於前者會使用到 Process 敘述指令，而後者不會。在 dataflow 的描述方法中，通常我們會以“選擇”的方式來描述電路，包含 Case-When、With-Select-When 等並行(Current)敘述指令，而非直接描述整個電路的演算法。

```
architecture Dataflow of NAND2 is
begin
    C<=A nand B; --布林方程式表示式 :C=(AB)'
end Dataflow;
```

## 3、行為性描述 ( Behavioral Description ) :

VHDL 語言中的行為性描述通常是使用 Process 敘述的方式來完成( Sequential 執行模式 )；使用者可以在不需要了解實際電路結構或連線方式的情況下，只要依序列出執行的順序，或是定義出電路的行為功能，便可以達到系統設計的目的；這種方式比較接近高階程式語言的用法，較適合大型且複雜的電路模組設計，一般我們也將之稱為高階描述方式(High-level Description)。它的優點在於使用者只需將精神專注於正確地定義出電路的功能上，無需虛擲力量在閘級 ( Gate Level ) 層次的設計工作，而且系統的可讀性和可維護性也較佳，不過它也有電路合成結果較難控制的缺點。

```
architecture behavior of NAND2 is
begin
    process (A,B) --宣告為順序(Sequential)執行模式
    begin
        if (A='1') and (B='1') then
            C<='0';
        else
```



```
        C<='1';  
    end if ;  
end process;  
end behavior;
```

## 3-3 簡單邏輯處理範例

### 3-3-1 一個簡單的反相器範例

下圖為反相器的晶片上視簡圖，下面則是其對應的 VHDL 程式寫法（檔名為 abc.vhd）：

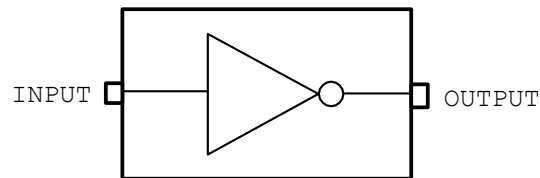


圖 3-4 反相器電路圖

```

--inverter gate design

library IEEE;
use IEEE.STD_LOGIC_1164.all;
entity abc is
port (input : in STD_LOGIC;
      output: out STD_LOGIC);
end abc;
architecture a of abc is
begin
    output <= not input;
end a;

```

本行是程式的註釋說明

宣告欲使用的程式庫與套件

檔名

單體宣告區：  
input宣告為Std\_logic型態的輸入位元；  
output宣告為Std\_logic型態的輸出位元。

架構名稱

架構宣告區：  
描述電路abc的動作是將input信號經過反相  
後傳遞到output信號。

【註】：在 VHDL 中標準邏輯 ( STD\_LOGIC ) 訊號型式定義如下：

```

Type STD_LOGIC is (
    'X', --Forcing Unknown;浮接不定
    '0', --Forcing 0;低電位
    '1', --Forcing 1;高電位
    'Z', --High Impedance;高阻抗
    'W', --Weak Unknown;弱浮接
    'L', --Weak 0;弱低電位
    'H', --Weak 1;弱高電位
    '-', --Don't care;不必理會
);

```



上面的 VHDL 程式在 Quartus II 中編譯程式的作法與電路輸入設計法幾乎完全相同，唯一有差異的一點是 VHDL 程式的副檔名是\*.vhd，整個編譯和分析的過程我們重述如下。

#### 1. 程式輸入與編譯：

- (1) 選擇【File】/【New Project Wizard】開啟新專案精靈，並依對話視窗要求選擇專案工作目錄、指定專案之名稱及最上層電路設計單體名稱後按【Finish】按鈕結束專案精靈。

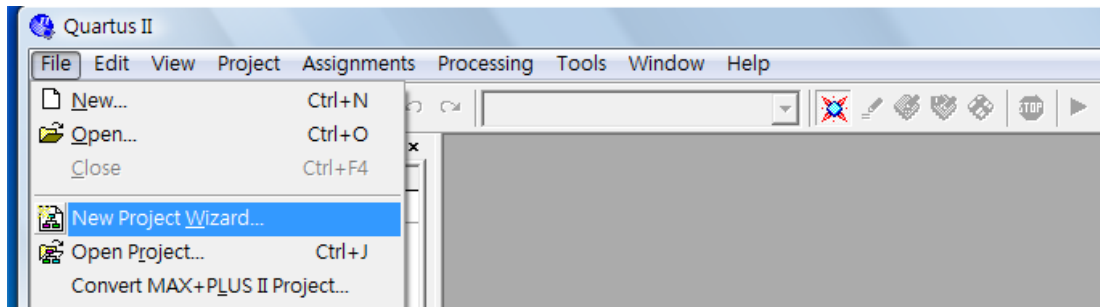


圖 3-5 開新專案精靈

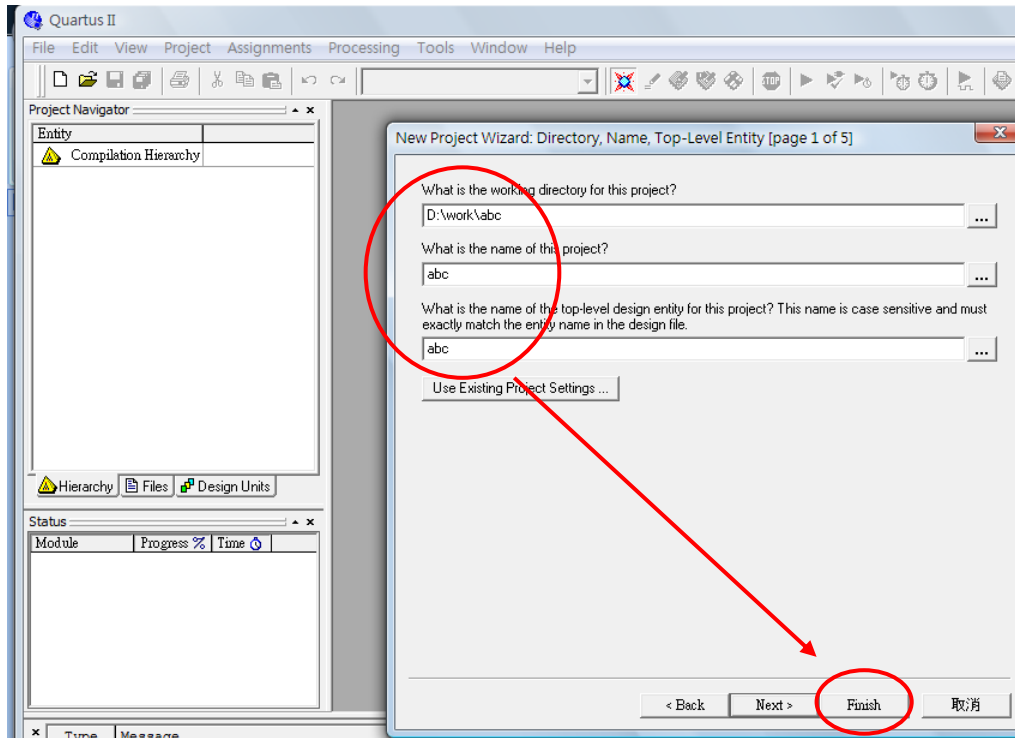


圖 3-6 選擇專案工作目錄、指定專案之名稱及最上層電路設計單體名稱

- (2) 選擇【File】/【New】開啟一個新的檔案，並於“Device Design Files”標籤下選定所欲開啟的檔案格式為“VHDL File”，輸入程式後儲存檔名為 abc.vhd(程式名稱可由文字、數字和下標線構成)，並保留“Add file to current project”選項為被勾選的狀態。完成上述程序後，此時我們若點選視窗左邊專案導覽視窗之“Files”按鈕，便可看到 abc.vhd 檔案出現。

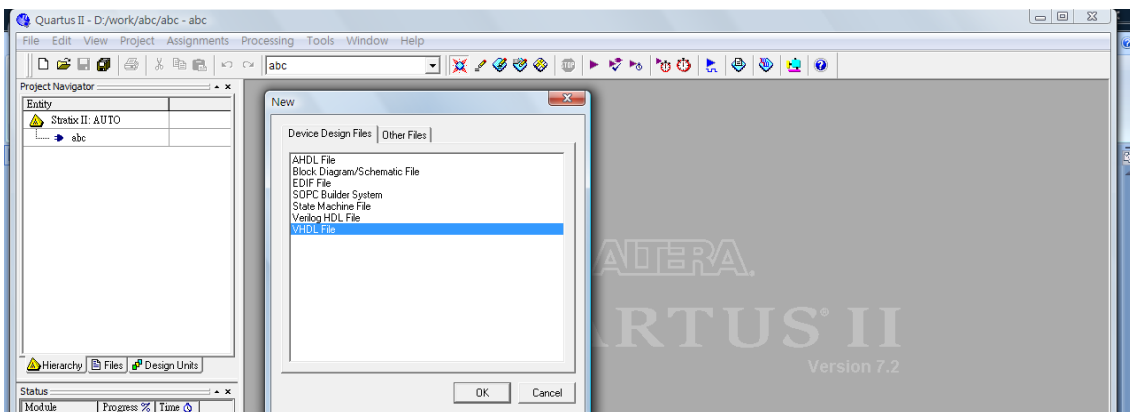




圖 3-7 開新檔案，並選擇檔案格式為 VHDL File

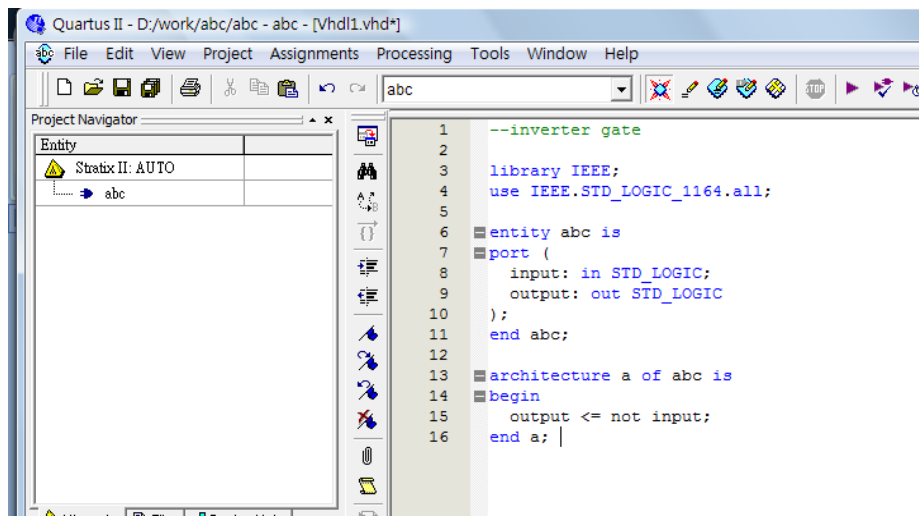


圖 3-8 輸入程式內容

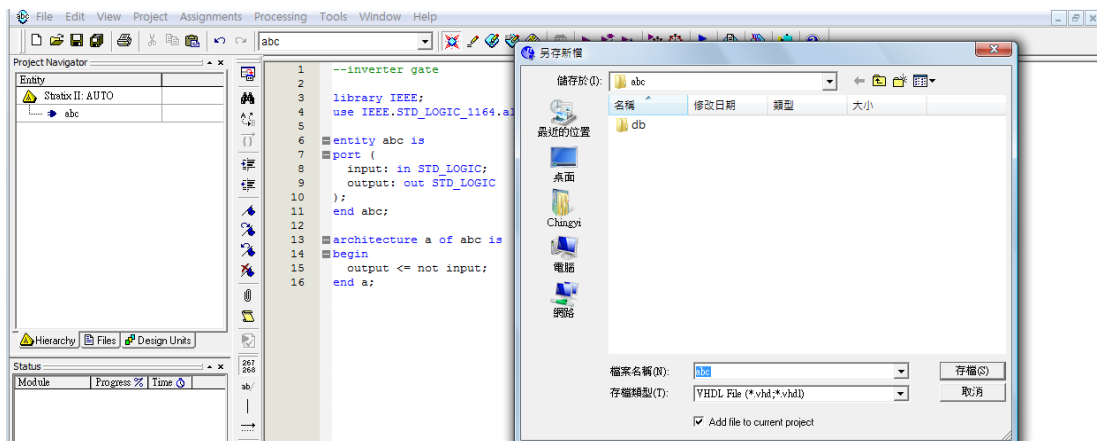


圖 3-9 儲存程式，並勾選“Add file to current project”

●關於 VHDL 的命名法則可整理如下：

- (a) 第一個字元必須是英文字母。
- (b) 最後一個位元不能是底線。
- (c) 不能有兩個底線連續出現。

- (3) 按下快捷工具列的 ► 圖示按鈕來執行程式編譯工作(或選擇【Processing】 / 【Start Compilation】選項亦可)。

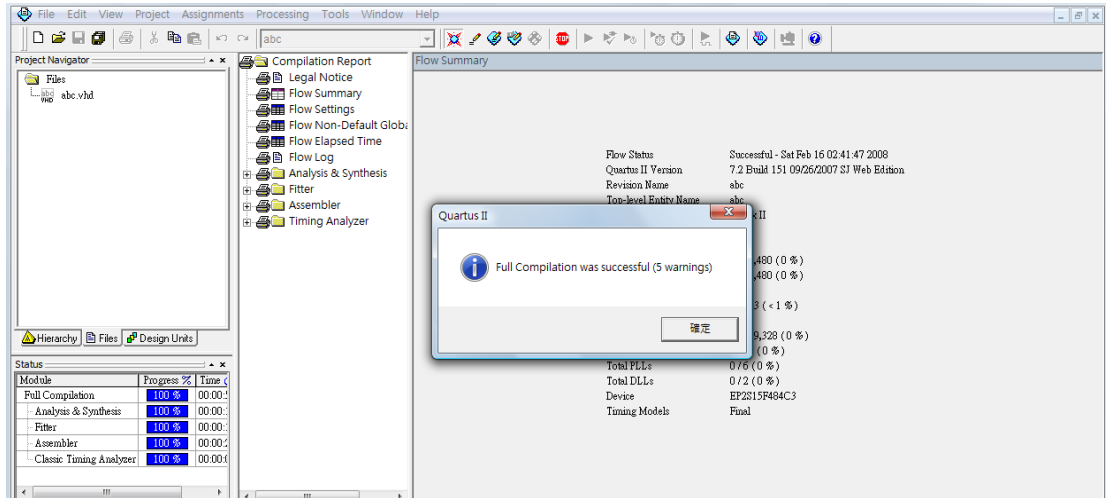


圖 3-10 執行程式編譯程序

## 2. 電路功能模擬：

- (1) 點選【File】 / 【New】開啟一個新的檔案，在對話視窗中選擇【Other Files】 / 【Vector Waveform File】選項以開啟 Waveform Editor 編輯一個新的波形檔 Vector Waveform File (\*.vwf)。。

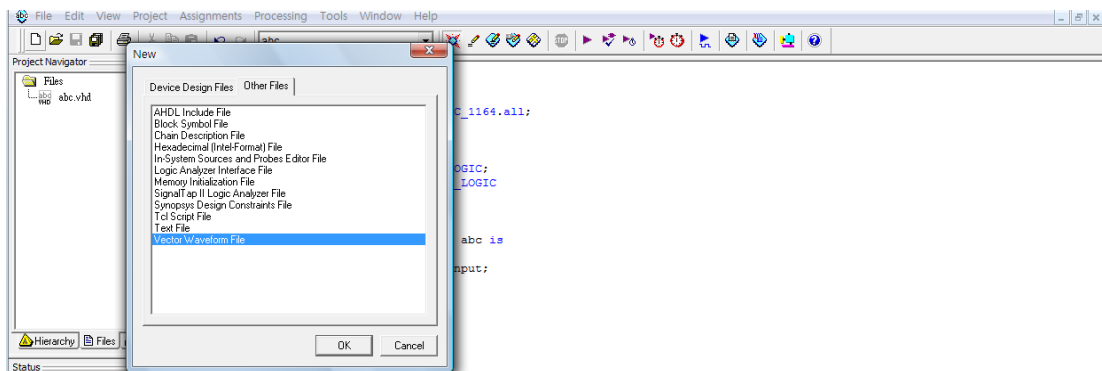


圖 3-11 開新波形檔





- (2) 選擇【Edit】/【Insert】/【Insert Node or Bus】以輸入欲分析的節點，並在【Insert Node or Bus】對話視窗中按【Node Finder】挑選電路節點。

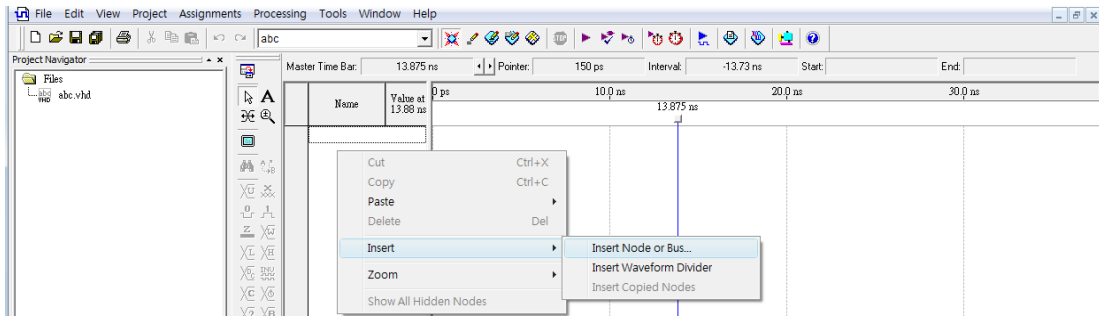


圖 3-12 【Insert】/【Insert Node or Bus】

- (3) 按下【List】鍵以顯示所有的 pin 腳，再將左邊框中所欲分析的節點挑選到右邊框內。

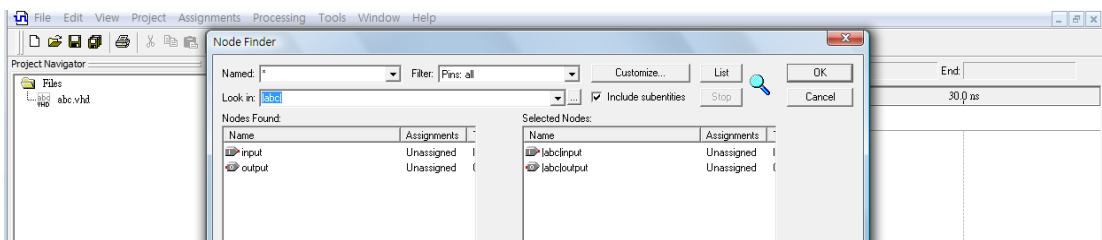





圖 3-13 點選所欲分析的節點

- (4) 利用視窗左側工具列圖示(如 clock 、數位脈衝 High  或 Low  等)來定義輸入腳位的波形條件，並儲存檔案為 abc.vwf。

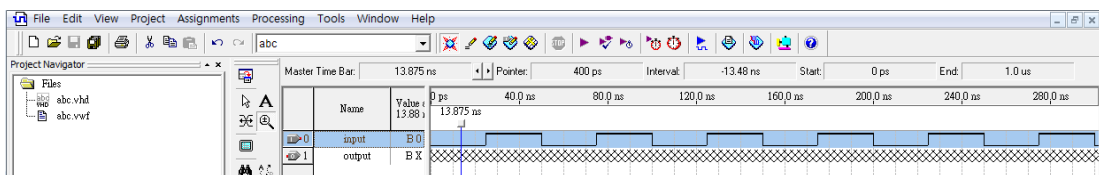



圖 3-14 輸入模擬信號

- (5) 選擇【Processing】/【Start Simulation】(或直接由快捷工具列的  圖示來執行)執行波形的模擬，以得到模擬報告(Simulation Report)視窗來查閱波形模擬的結果。

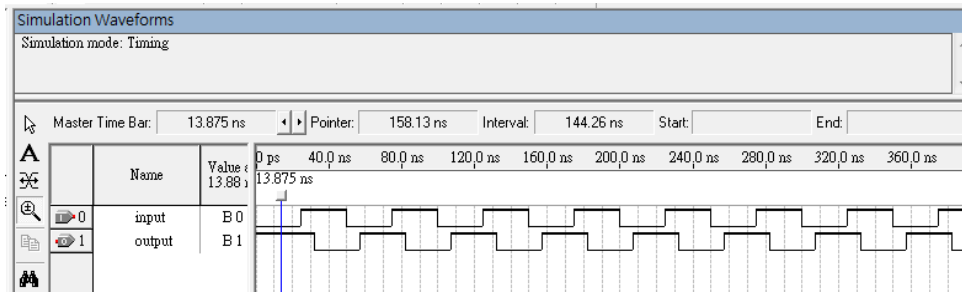


圖 3-15 產生的波形模擬視窗(考慮時間延遲因素)

- (6) 若我們選擇【Processing】/【Simulator Tool】選項，將 Simulation mode 設定為【Functional】，再按下【Generate Functional Simulation Netlist】選項，便可重新執行模擬得到理想狀態的波形(不考慮時間延遲因素)。

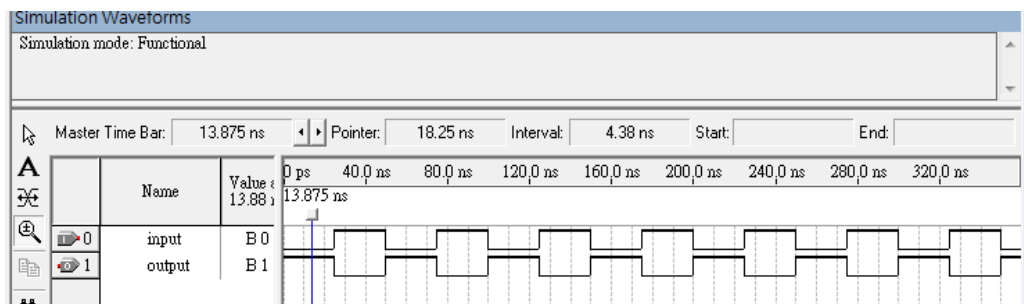


圖 3-16 不考慮時間延遲因素的理想狀態波形模擬

### 3. 將 VHDL 程式轉換成 RTL :

點選【Tool】/【Netlist Viewers】/【RTL Viewers】便可得到由暫存器和邏輯元件所構成的 RTL (Register Transfer Level) 電路圖。但由於本例子過於簡單，因此我們從電路圖中看到 VHDL 程式所描述的線路僅是一個反相關而已(圓形圈圈)。

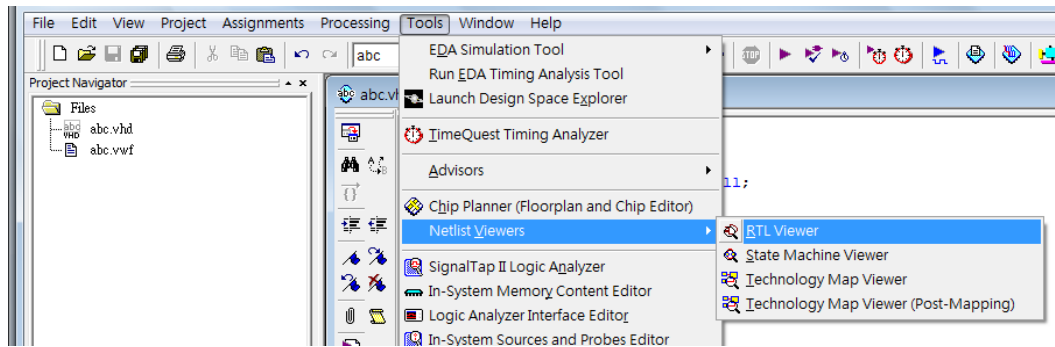


圖 3-17 使用 RTL Viewer

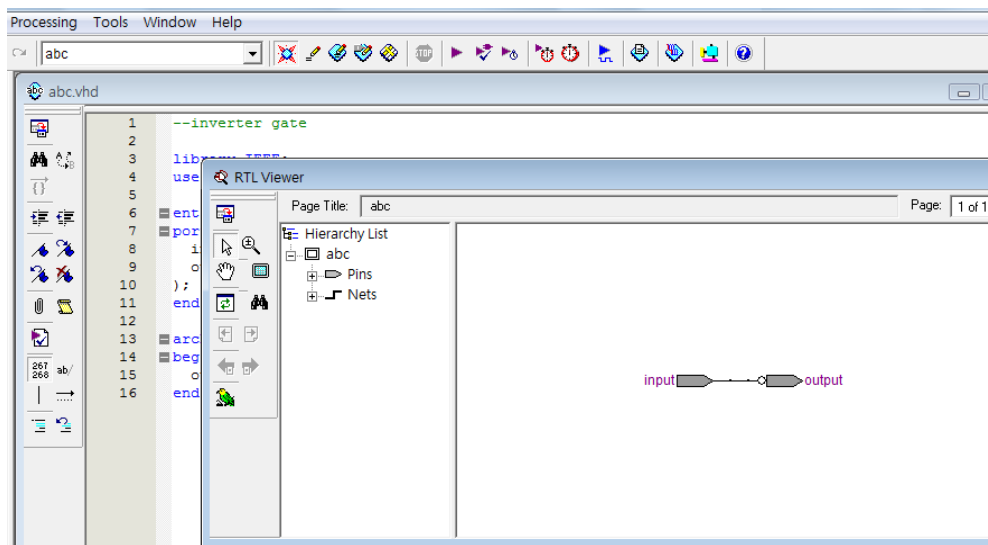


圖 3-18 VHDL 程式所描述的 RTL 線路圖

### 3-3-2 其他邏輯電路描述

在數位電路中，我們常用的一些基本邏輯閘有 AND 閘、OR 閘、NOT 閘、XOR 閘等元件，其電路符號如下：

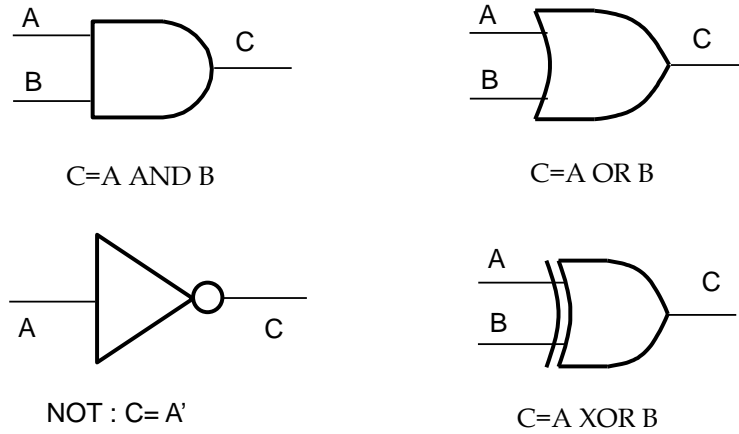


圖 3-19 基本邏輯閘介紹

再來我們將探討工業上常使用的 NOR 閘和 NAND 閘，我們可以由其電路邏輯閘之間等效轉換關係了解到笛摩根定理(De Morgan's Theorem)：

1. NOR 閘：其邏輯意義即為在 OR 閘輸出端再加上一個反相器。

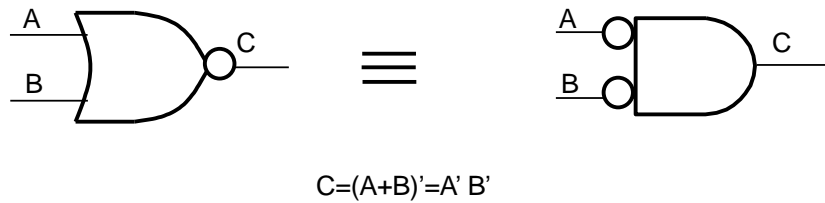


圖 3-20 笛摩根第一定理  $(A+B)'=A' B'$

2. NAND 閘：其邏輯意義即為在 AND 閘輸出端再加上一個反相器。

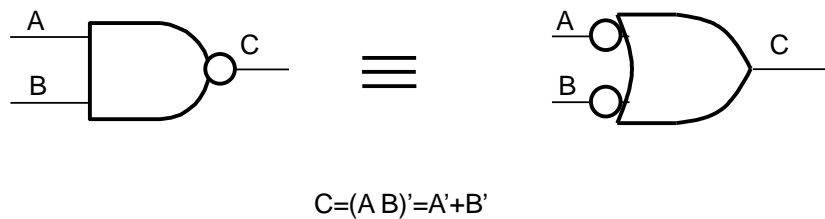


圖 3-21 笛摩根第二定理  $(AB)'=A'+B'$



而在 VHDL 語法中，關於 BIT 型式的邏輯我們可以使用的運算如下：

邏輯 運算	NOT	AND	OR	XOR	NAND	XNOR
比較 運算	= equal	/= not equal	< less than	<= less than or equal	> greater than	>= greater than or equal



試以 VHDL 邏輯運算元實現 NOR 閘電路。

```

library IEEE;
use IEEE.STD_LOGIC_1164.all;

entity nor_gate is                                --檔名為nor_gate.vhd
port ( A,B: in STD_LOGIC;
      C : out STD_LOGIC );
end nor_gate;

architecture a of nor_gate is                    --將架構名稱命名為 a
begin
  C <= not (A or B);
end a;

```

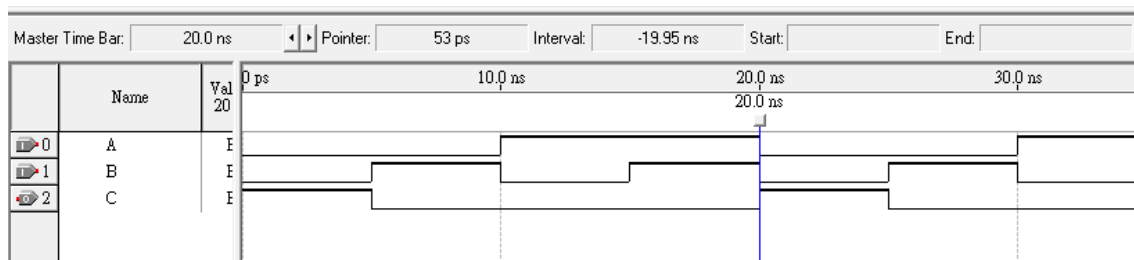


圖 3-22 時序波形模擬圖

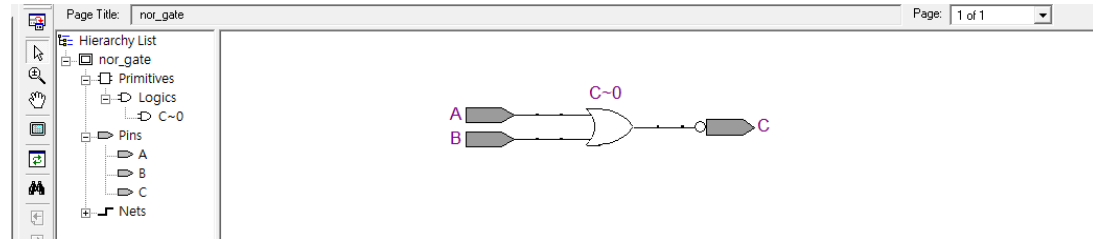


圖 3-23 使用 RTL Viewer 查看 VHDL 程式所描述的 RTL 線路圖



練習：

請完成下面邏輯電路的程式設計：

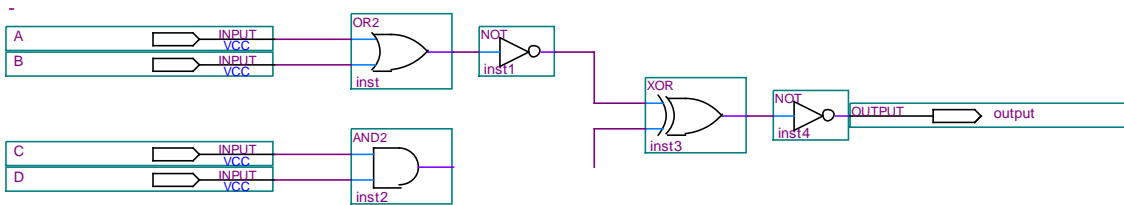


圖 3-24 邏輯電路實例

從前面範例中，我們可以很清楚直接設定敘述“<=”的作用，VHDL 的描述語法與 C 語言等高階程式語言最大的差異是硬體設計必須讓電路具有同時並行處理的能力，而非如一般高階語言般以一次一個命令的順序來執行動作。下面的範例中，三組電路在繪圖上雖有上下左右之分，但事實上彼此之間卻是完全獨立的電路，故三組電路的輸入和輸出會在同一個時間發生，即使顛倒重畫也不致於影響功能。

範例

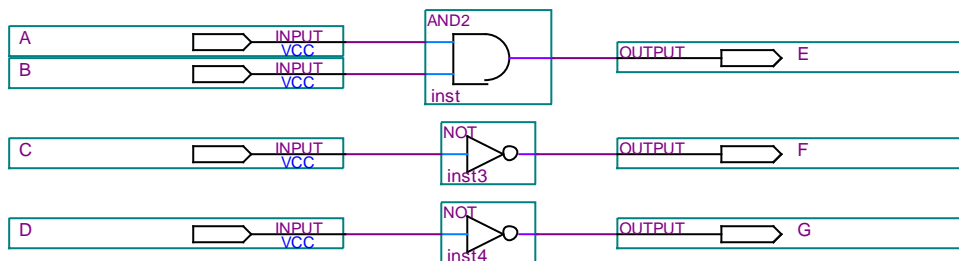




圖 3-25 一個具有四個輸入、三個输出的邏輯電路

```

library IEEE;
use IEEE.STD_LOGIC_1164.all;

entity conex is
    port (A,B,C,D: in STD_LOGIC;
          E,F,G  : out STD_LOGIC);
end conex;

architecture a of conex is
begin
    E <= A and B;
    F <= not C;
    G <= not D;
end a;

```

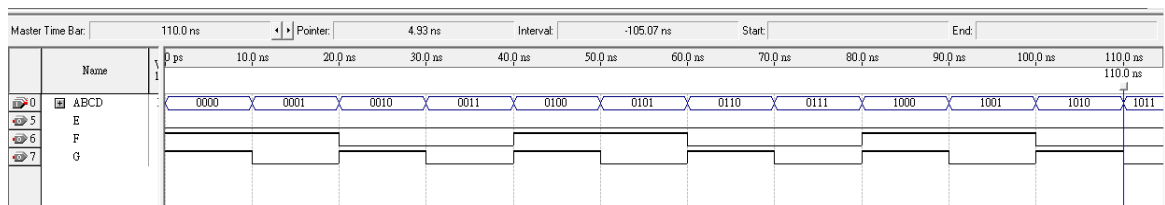


圖 3-26 功能模擬圖

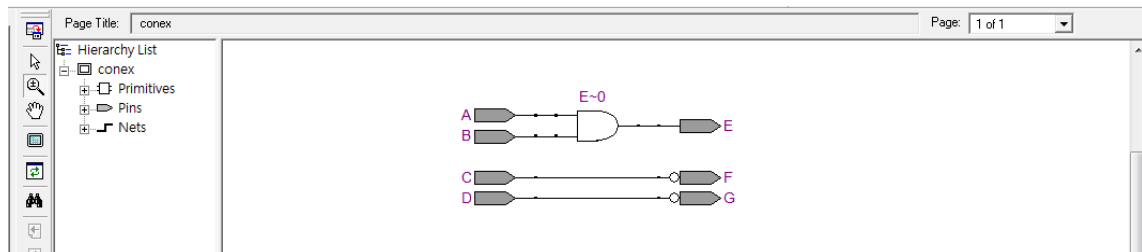


圖 3-27 使用 RTL Viewer 查看 VHDL 程式所描述的 RTL 線路圖



上面的 VHDL 程式在 Quartus II 中編譯程式的作法與電路輸入設計法幾乎完全相同，唯一有差異的一點是 VHDL 程式的副檔名是\*.vhd，整個編譯和分析的過程我們重述如下。

#### 1. 程式輸入與編譯：

- (1) 選擇【File】/【New Project Wizard】開啟新專案精靈，並依對話視窗要求選擇專案工作目錄、指定專案之名稱及最上層電路設計單體名稱後按【Finish】按鈕結束專案精靈。

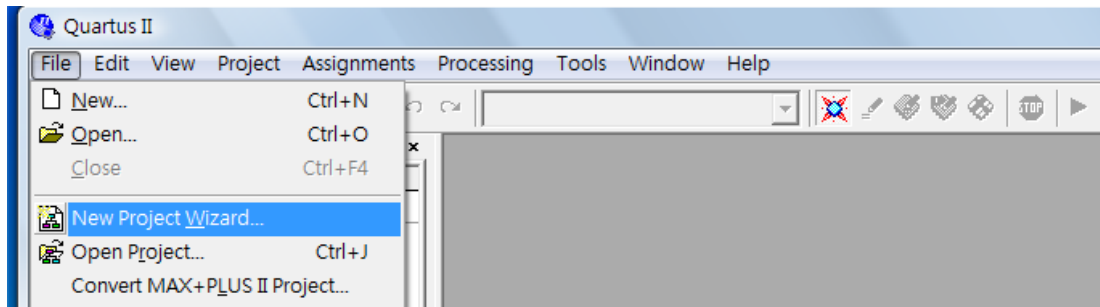


圖 3-5 開新專案精靈



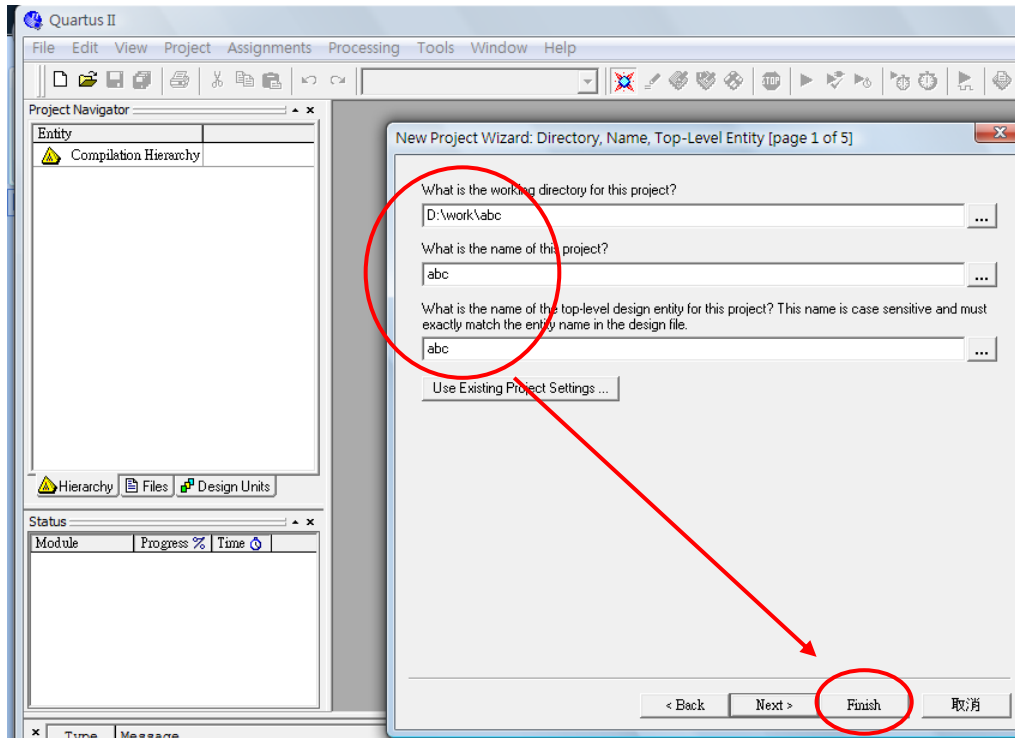


圖 3-6 選擇專案工作目錄、指定專案之名稱及最上層電路設計單體名稱

- (2) 選擇【File】/【New】開啟一個新的檔案，並於“Device Design Files”標籤下選定所欲開啟的檔案格式為“VHDL File”，輸入程式後儲存檔名為 abc.vhd(程式名稱可由文字、數字和下標線構成)，並保留“Add file to current project”選項為被勾選的狀態。完成上述程序後，此時我們若點選視窗左邊專案導覽視窗之“Files”按鈕，便可看到 abc.vhd 檔案出現。

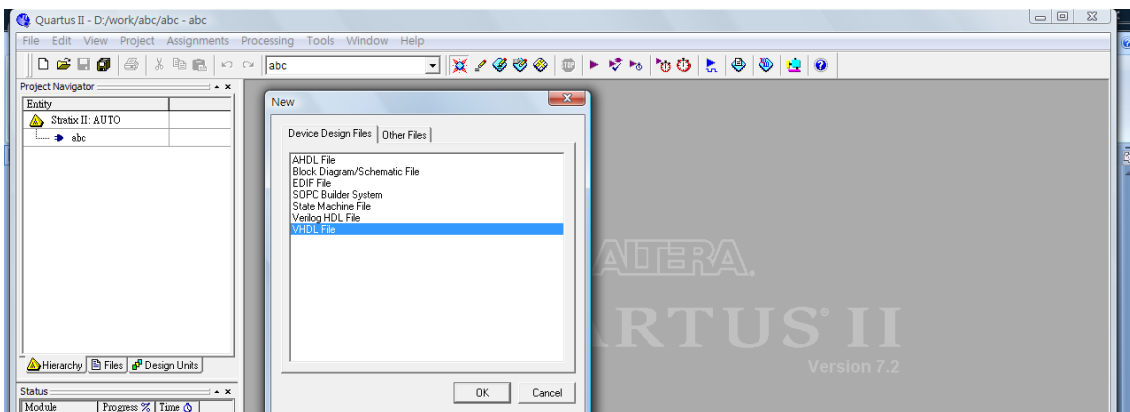




圖 3-7 開新檔案，並選擇檔案格式為 VHDL File

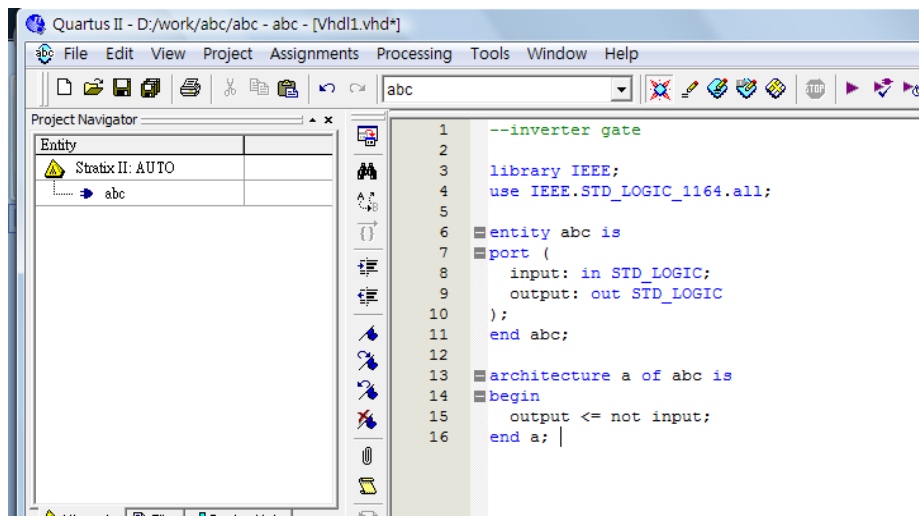


圖 3-8 輸入程式內容

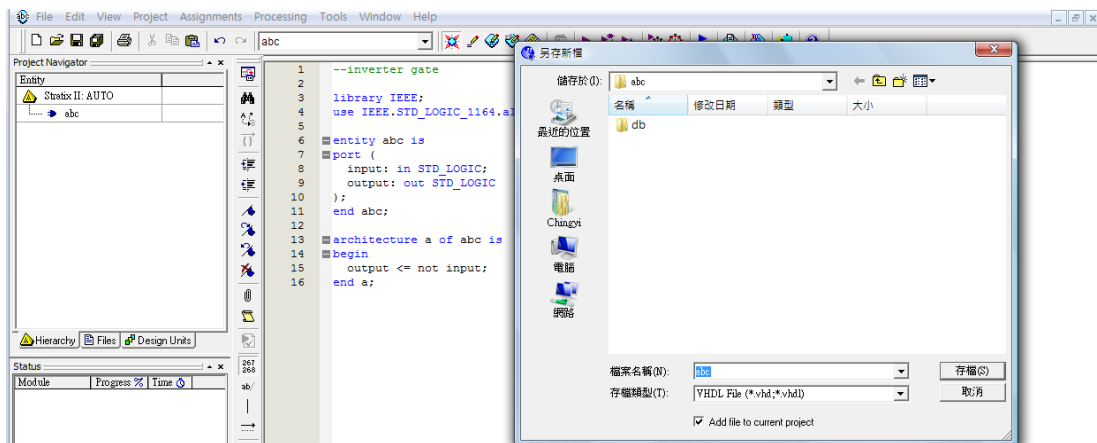


圖 3-9 儲存程式，並勾選“Add file to current project”

●關於 VHDL 的命名法則可整理如下：

- (a) 第一個字元必須是英文字母。
- (b) 最後一個位元不能是底線。
- (c) 不能有兩個底線連續出現。

- (3) 按下快捷工具列的 ► 圖示按鈕來執行程式編譯工作(或選擇【Processing】 / 【Start Compilation】選項亦可)。

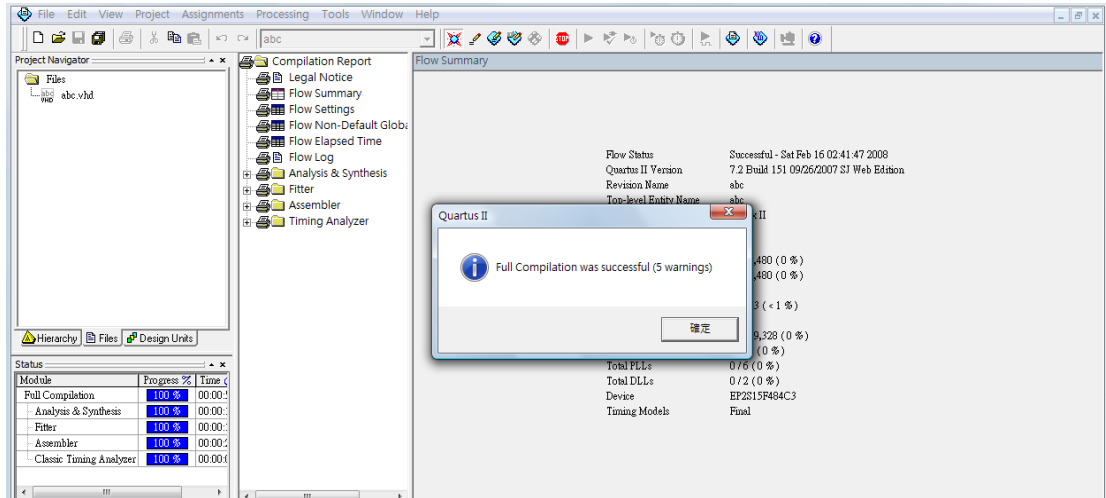


圖 3-10 執行程式編譯程序

## 2. 電路功能模擬：

- (1) 點選【File】 / 【New】開啟一個新的檔案，在對話視窗中選擇【Other Files】 / 【Vector Waveform File】選項以開啟 Waveform Editor 編輯一個新的波形檔 Vector Waveform File (\*.vwf)。。

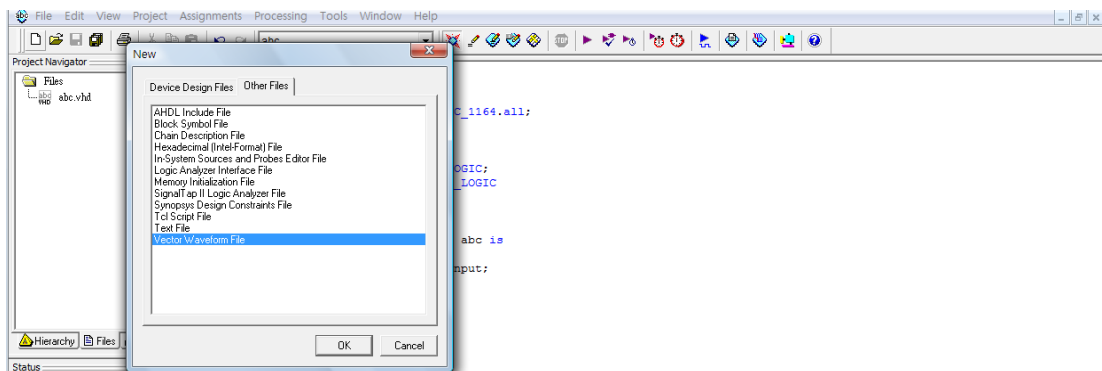


圖 3-11 開新波形檔



- (2) 選擇【Edit】/【Insert】/【Insert Node or Bus】以輸入欲分析的節點，並在【Insert Node or Bus】對話視窗中按【Node Finder】挑選電路節點。

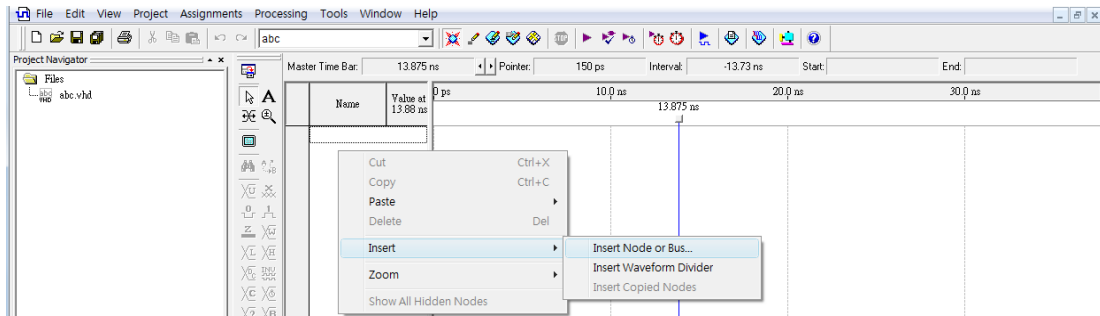


圖 3-12 【Insert】/【Insert Node or Bus】

- (3) 按下【List】鍵以顯示所有的 pin 腳，再將左邊框中所欲分析的節點挑選到右邊框內。

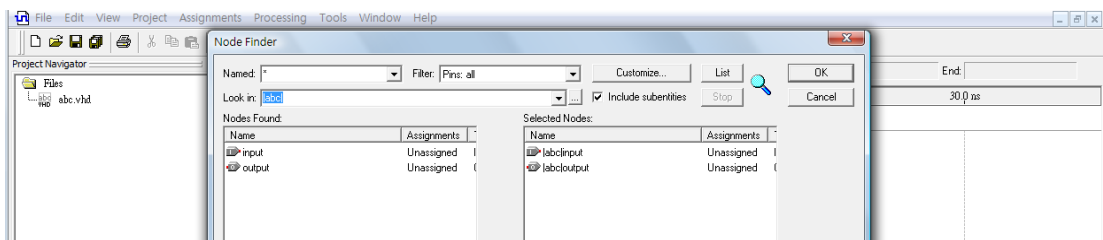





圖 3-13 點選所欲分析的節點

- (4) 利用視窗左側工具列圖示(如 clock 、數位脈衝 High  或 Low  等)來定義輸入腳位的波形條件，並儲存檔案為 abc.vwf。

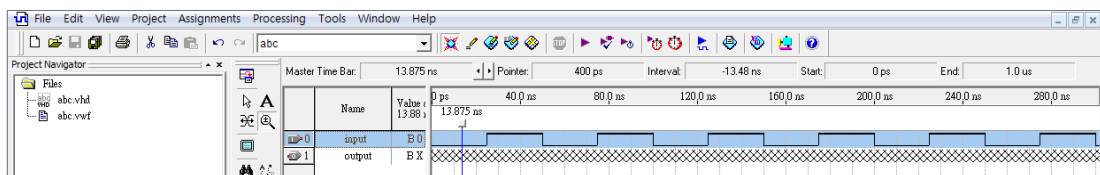



圖 3-14 輸入模擬信號

- (5) 選擇【Processing】/【Start Simulation】(或直接由快捷工具列的  圖示來執行)執行波形的模擬，以得到模擬報告(Simulation Report)視窗來查閱波形模擬的結果。

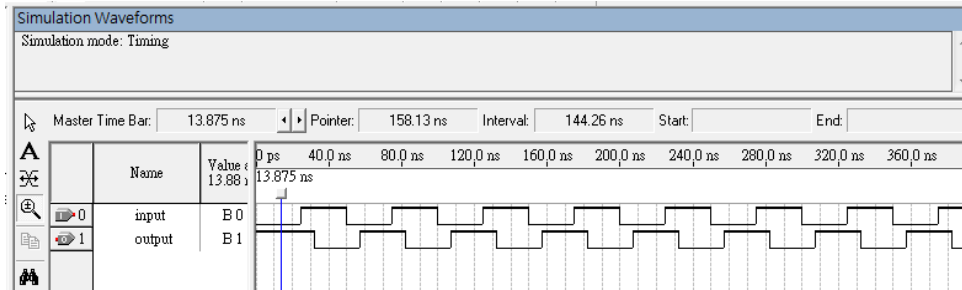


圖 3-15 產生的波形模擬視窗(考慮時間延遲因素)

- (6) 若我們選擇【Processing】/【Simulator Tool】選項，將 Simulation mode 設定為【Functional】，再按下【Generate Functional Simulation Netlist】選項，便可重新執行模擬得到理想狀態的波形(不考慮時間延遲因素)。

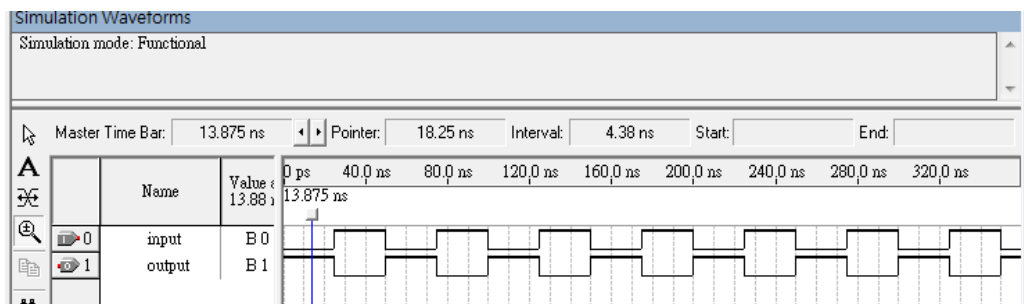


圖 3-16 不考慮時間延遲因素的理想狀態波形模擬

### 3. 將 VHDL 程式轉換成 RTL :

點選【Tool】/【Netlist Viewers】/【RTL Viewers】便可得到由暫存器和邏輯元件所構成的 RTL (Register Transfer Level) 電路圖。但由於本例子過於簡單，因此我們從電路圖中看到 VHDL 程式所描述的線路僅是一個反相關而已(圓形圈圈)。

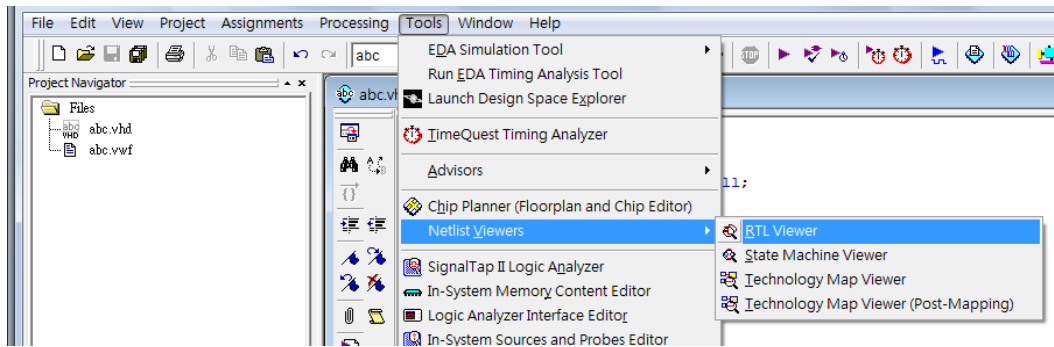


圖 3-17 使用 RTL Viewer

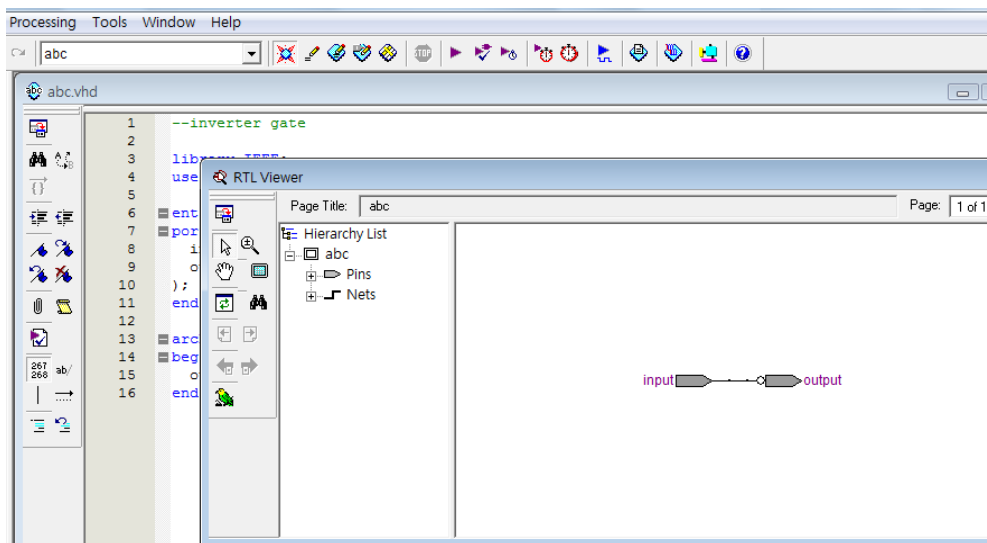


圖 3-18 VHDL 程式所描述的 RTL 線路圖

### 3-3-2 其他邏輯電路描述

在數位電路中，我們常用的一些基本邏輯閘有 AND 閘、OR 閘、NOT 閘、XOR 閘等元件，其電路符號如下：