

ZYNQ 開發基本流程-硬體篇

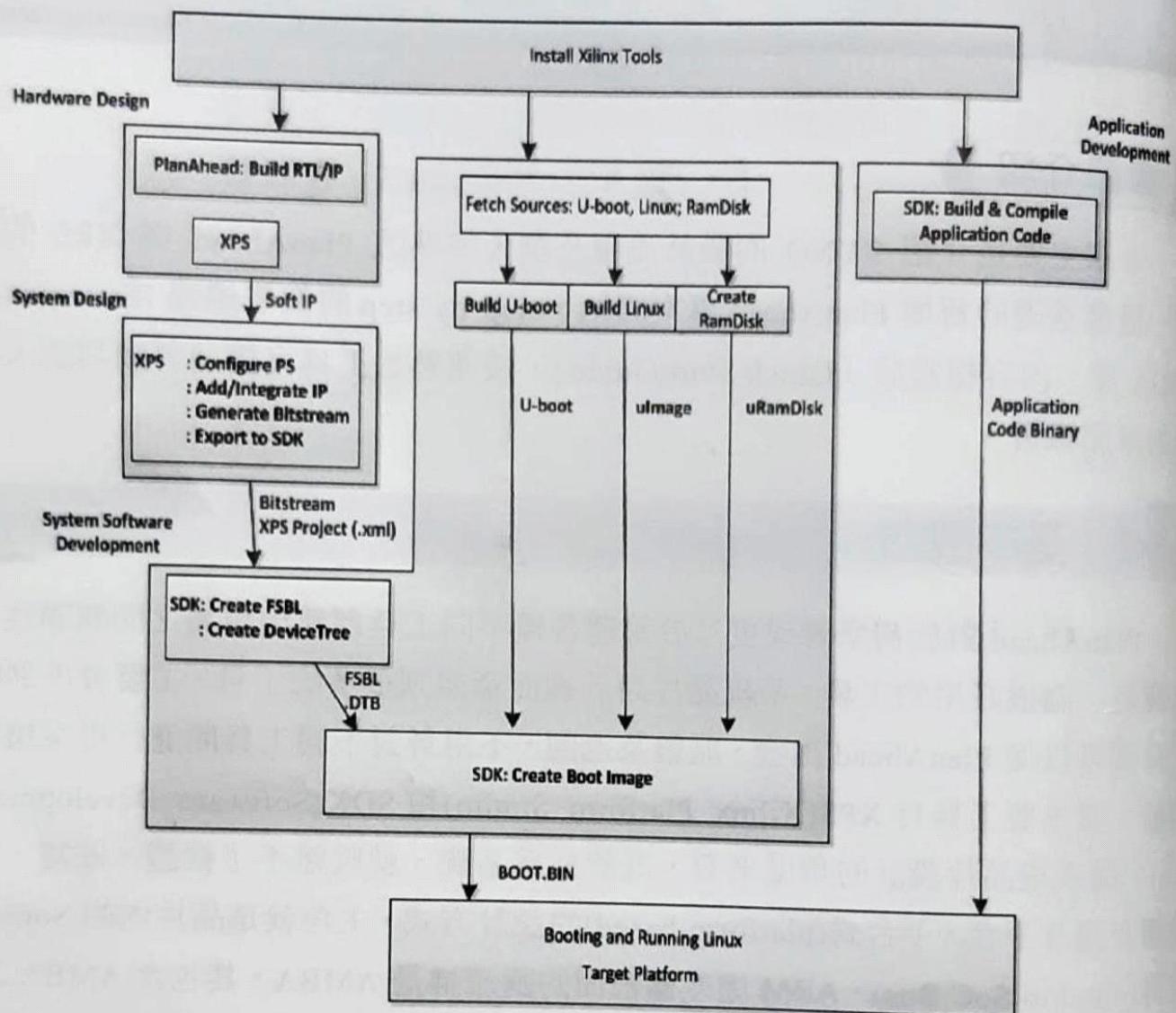
章節介紹

本章主要是介紹 ZYNQ 的設計流程及讓大家熟悉 PlanAhead 與 XPS 的操作，從最基礎的新增 PlanAhead 專案開始，step by step 的教學燒錄 Bitstream 到 FPGA 裡。內容相當於「Quick Start Guide」，快速熟悉工具可讓讀者順利進入後面的章節練習。

1-1 基本觀念

PlanAhead 對於初學者或更易於管理各種不同工具創建出專案之間關聯性，無疑是一個很好用的工具。系統晶片設計裡面牽涉到很多的工具，主要會用到的工具皆可以從 PlanAhead 出發、開啓及返回，不用針對不同工具間進行專案檔的連結。而主要工具有 XPS(Xilinx Platform Studio)和 SDK(Software Development Kit)，從系統晶片設計的角度來看，既然稱為系統，就脫離不了軟體、硬體、軟硬體介面及平台，平台式(platform-based)的設計方式，主角就是晶片內的 SoC 汇流排(on-chip SoC Bus)，ARM 處理器裡面的匯流排是 AMBA，其包含 AMBA 2.0 時期的 AHB、APB 及 ASB 三種匯流排，以及 3.0 之後更方便支援多核心(Multi-core)的 AXI 汇流排，目前 AHB、APB 和 AXI 是主流，因此 Zynq SoC 也包含了這三種匯流排，其中 APB 是連接慢速周邊與矽智財(Silicon Intellectual Property)的主要介面，AHB 和 AXI 是連接高速矽智財(包含 CPU)之間的介面，Zynq 採用 ARM Cortex A9 雙核心 CPU，開放 AXI bus 讓使用者的矽智財(邏輯電路設計)可以連接到 CPU，透過橋接也可以和其他 AHB 包裝的矽智財溝通。因

此，XPS 的任務就是讓使用者可以從匯流排的系統觀點來檢視自己的電路設計，並且協助使用者可以很簡單地用 AXI 標準協定包裝自己的矽智財(即 SIP 的 Wrapping)，掛載上 AXI 汇流排，整合到 SoC 裡面。電路整合完後，剩下的工作就是如何驅動自己設計的矽智財，此時就是啟動 SDK 的時候，讓軟體與硬體互相關聯與分工。如圖 1-1 所示，XPS 擔任的角色是系統觀點以及電路連接，SDK 擔任的角色是軟體驅動與上層軟體設計。



▲圖 1-1 PlanAhead, SDK, XPS 的關係圖與設計流程

在整個設計流程中，一共會使用到好幾個工具，分別是 PlanAhead、XPS 和 SDK。在 PlanAhead 中，主要是完成對硬體架構的設計，包含繪製 RTL 電路(RTL Schematic)、電路高階合成(Synthesis)及在 FPGA 內的實現(Implementation)。FPGA 實現包含將高階的 RTL 描述轉換成 FPGA 內部元件、電路最佳化及 Placement and Routing (PAR 或 P&R)，PlanAhead 最後可以產生 Bitstream 下載到 Zynq 的晶片。

中。XPS 可以用來設定 Processor System(PS 是 CPU)、新增(Add)或掛載智財(IP)到 SoC 匯流排上(Integrate IP)、產生 bitstream 及匯出(Export)SIP 內部的 port 暫存器資訊(暫存器的位址、位元寬度、個數及名稱等)給 SDK，以利在 SDK 環境中設計驅動程式。SDK 工具可以用來設計驅動程式模組、Linux 環境下的裝置樹(Device Tree)驅動程式及應用程式。如果要建立完整的 Linux 作業系統環境，SDK 當然還包含了建立開機程式(Boot Image)、設定如何使用 First Stage Boot Loader(FSBL)來載入 Boot Image 及編譯 Linux 作業系統核心(Kernel)與檔案系統等功能。

1-2 軟體工具的取得與安裝

1-2-1 ISE Design Tools 下載

前往 Xilinx 的官方網站 <http://www.xilinx.com/>，進入 Downloads 頁面，選擇 ISE Design Tools 下載，本書所用的版本是 14.7。

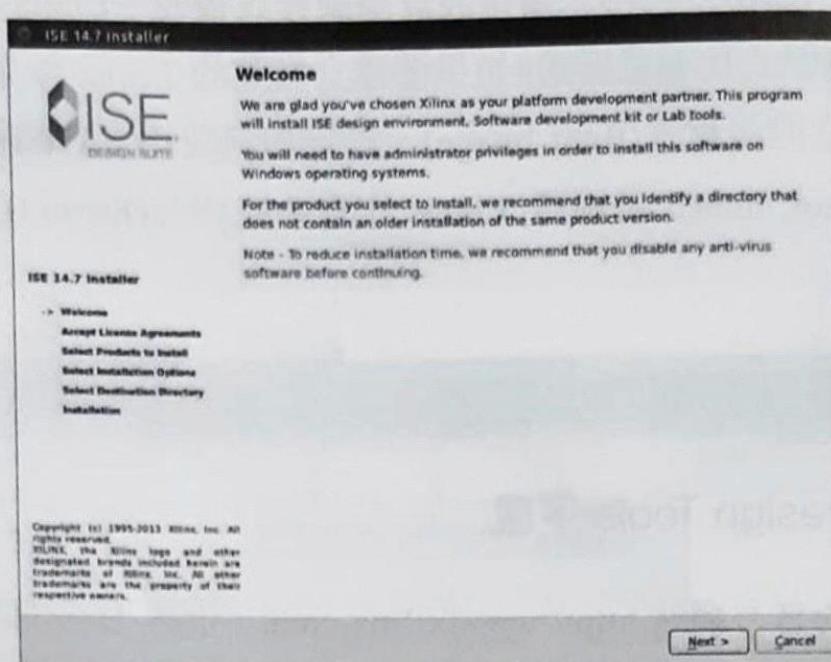
The screenshot shows the Xilinx download page for ISE Design Tools version 14.7. The top navigation bar includes links for Vivado Design Tools, ISE Design Tools, Device Models, CAE Vendor Libraries, and PetaLinux. A dropdown menu for 'Version' is open, showing options from 14.6 down to 10.1 SP3, with '14.7' selected. Below the dropdown is a section titled 'Multi-File Download: ISE Design - 14.7 Full Product Installation'. It contains a note: 'Last Updated October 2013' followed by a detailed text about the product's life cycle and support. Below this are four download links for 'All Platforms - Split Installer Base Image - File 1/4 (TAR/GZIP - 1.95 GB)' with MD5 SUM values: f0f8a08aba2b7110fa730c6b15067d6, c096203646476b772b20c032b2f954b, e6146a7eac7c026b4b507fd7549e4e, and 9094381327a083e89293e742416417. At the bottom of the page, there is an 'Important Information' section with a note about browser-based download managers.

▲圖 1-2 Xilinx 官網的 PlanAhead 下載頁面

1-2-2 ISE Design Tools 安裝步驟

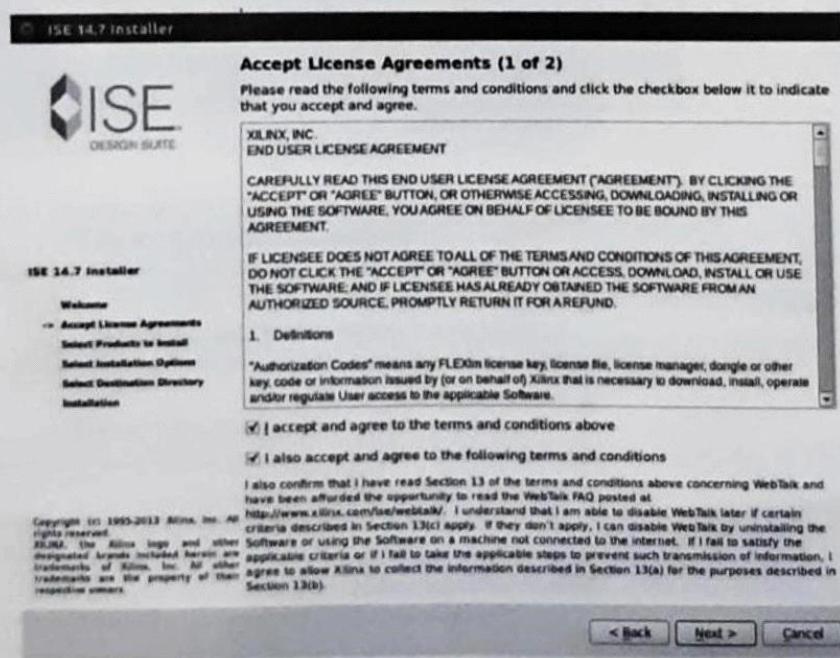
以下為安裝步驟畫面介紹。

1. 歡迎畫面，點擊 Next 繼續。



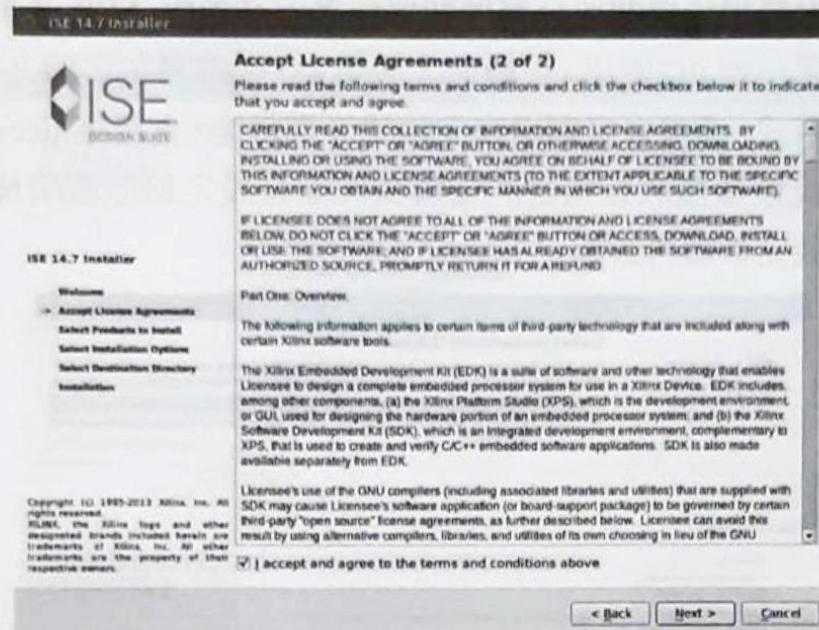
▲圖 1-3 安裝 ISE 14.7 軟體時的歡迎畫面。

2. 將兩個詢問許可的選項打勾，表示同意授權合約的內容：通常這類免費軟體安裝的合約，都有宣告授權使用範圍，例如：允許使用在系統晶片設計的教學上，不負宣告安裝及使用過程中造成電腦軟硬體損壞之責任。若使用者沒有意見，點擊 Next 繼續。



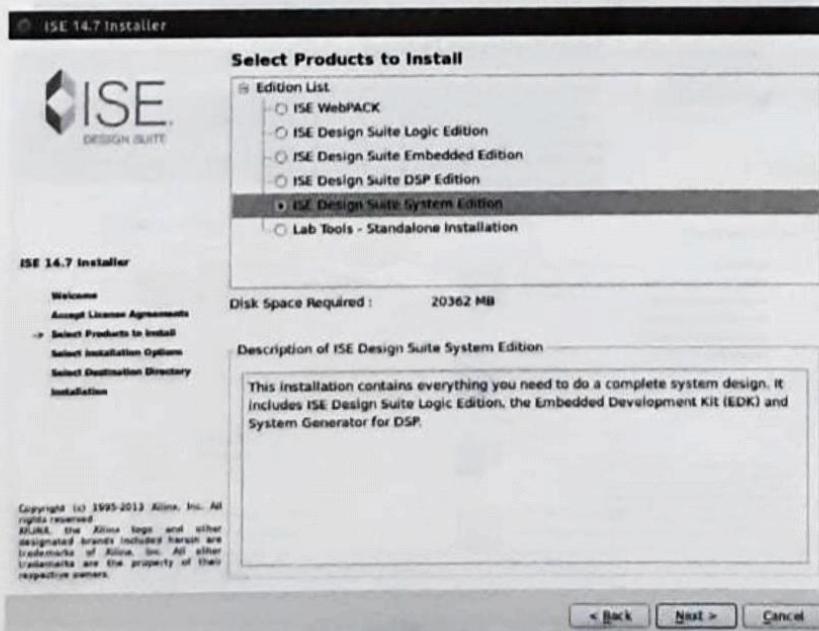
▲圖 1-4 軟體安裝授權合約第一頁。

3. 合約共有兩頁，若同意則將詢問許可的選項打勾，點擊 Next 繼續。



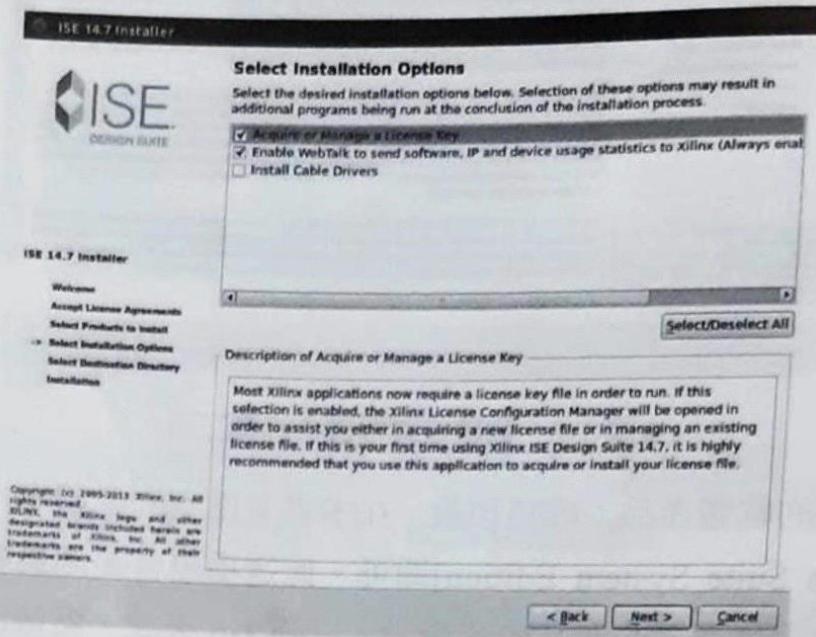
▲圖 1-5 軟體安裝授權合約第二頁。

4. 選擇要安裝的軟體產品：維持預設。在本書會用到的軟體中，維持預設的選擇 [ISE Design Suite System Edition]即可。此選項包含範圍最廣。點選不同選擇時，下方會出現簡單的產品內容說明。以圖 1-6 來說，點選此安裝選項會包含 Embedded Development Kit (EDK) 以及可以和 Matlab 軟體整合的 System Generator for DSP。



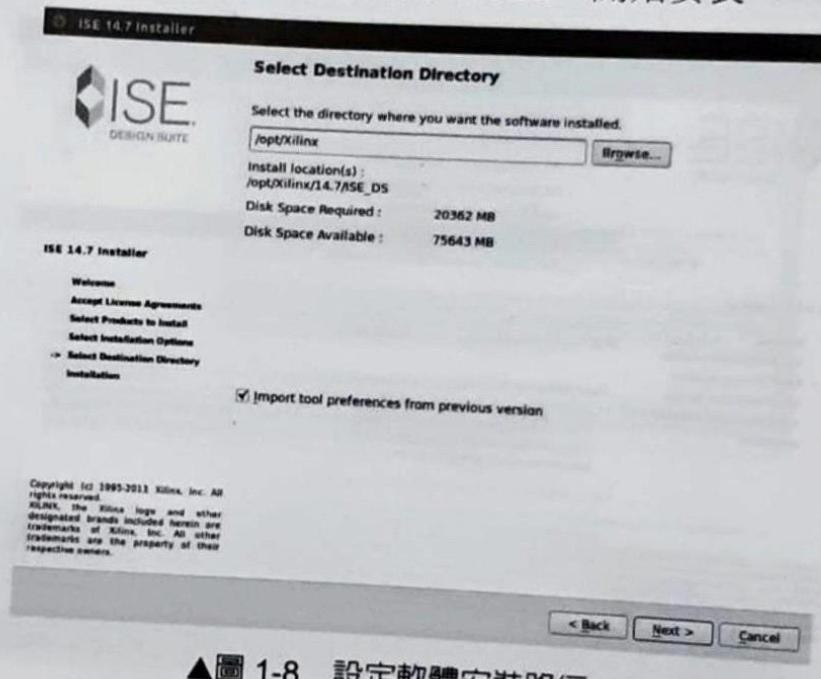
▲圖 1-6 選擇 ISE 14.7 軟體系列產品。

5. 安裝的選項：維持預設。第一個選項在安裝完之後會引導你上網管理授權，一些使用者是附屬學校機關或公務機關單位等，必須由 MIS 人員上網設定授權以及授權給哪一台 PC 使用。一般個人使用者，則需要在官網上建立一個帳號來設定授權方式。通常自官網填寫完帳號等資料後，會產生 license.dat 的檔案，依照指示將此授權檔放在指定的資料夾，並且設定路徑環境變數後就可以用，或者安裝完軟體再去指定 license.dat 的路徑。



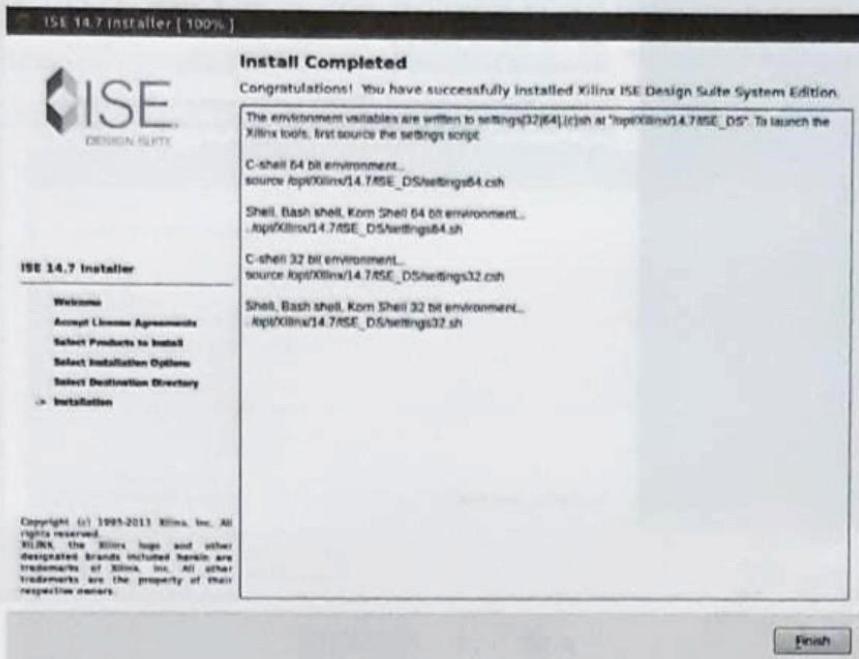
▲圖 1-7 選擇 ISE 14.7 軟體安裝選項。

6. 選擇要安裝的資料夾路徑，然後點擊 Install，開始安裝。



▲圖 1-8 設定軟體安裝路徑。

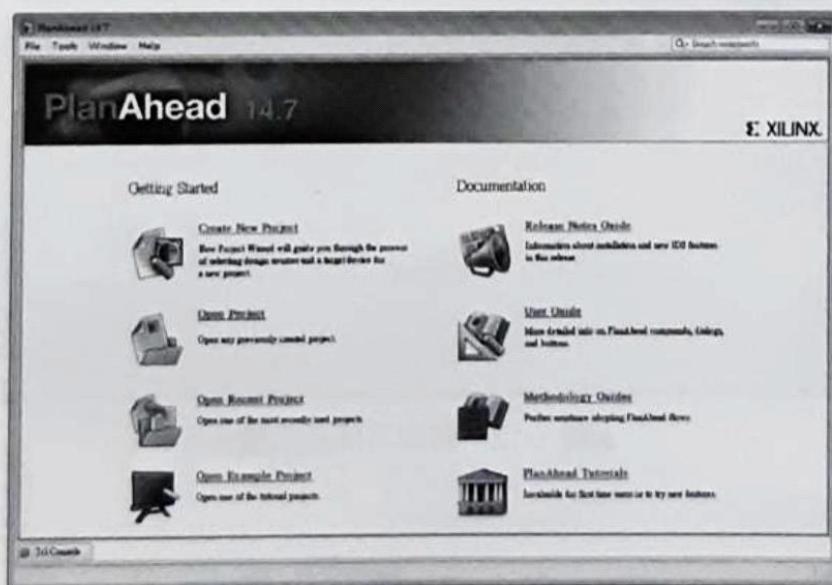
7. 安裝完成畫面如圖 1-9 所示，會簡單的整理安裝結果。主要說明相關環境設定的 C-shell script 檔案是那些。



▲圖 1-9 安裝完之畫面

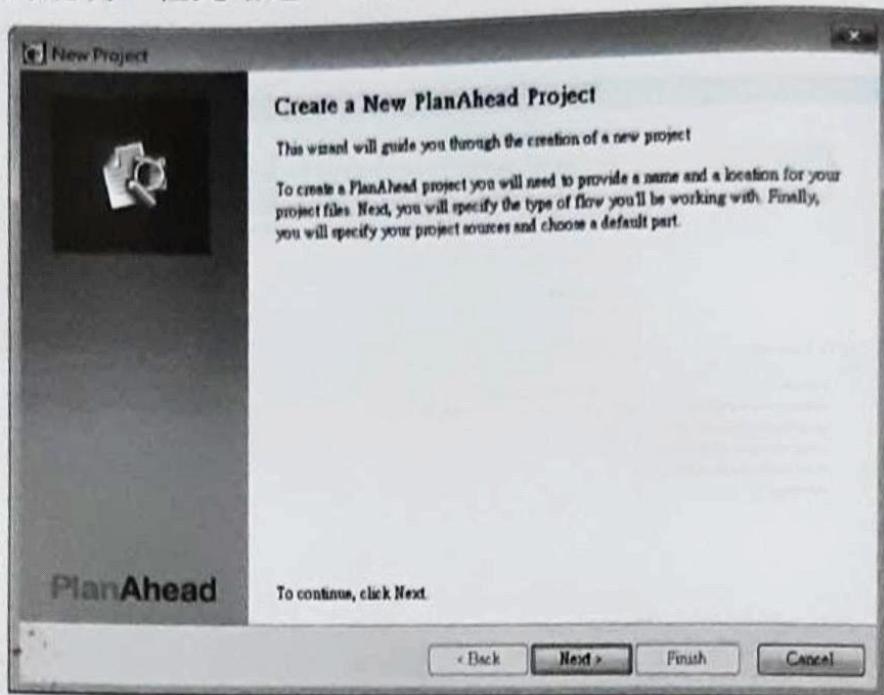
1-3 創建 PlanAhead 專案

1. PlanAhead 開啓後畫面如圖 1-10 所示，在畫面的左上方會顯示軟體的版本，在此使用的是 14.7 版本，點擊 Create New Project 來創建新專案。



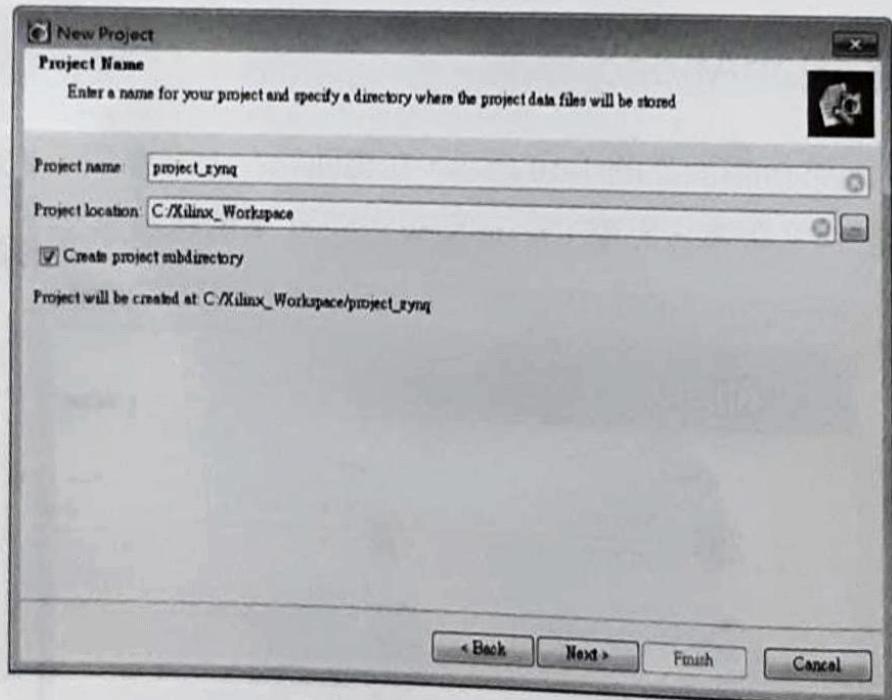
▲圖 1-10 PlanAhead 開始畫面

2. 專案精靈的說明，在此略過，直接點擊 Next 繼續。



▲圖 1-11 專案精靈

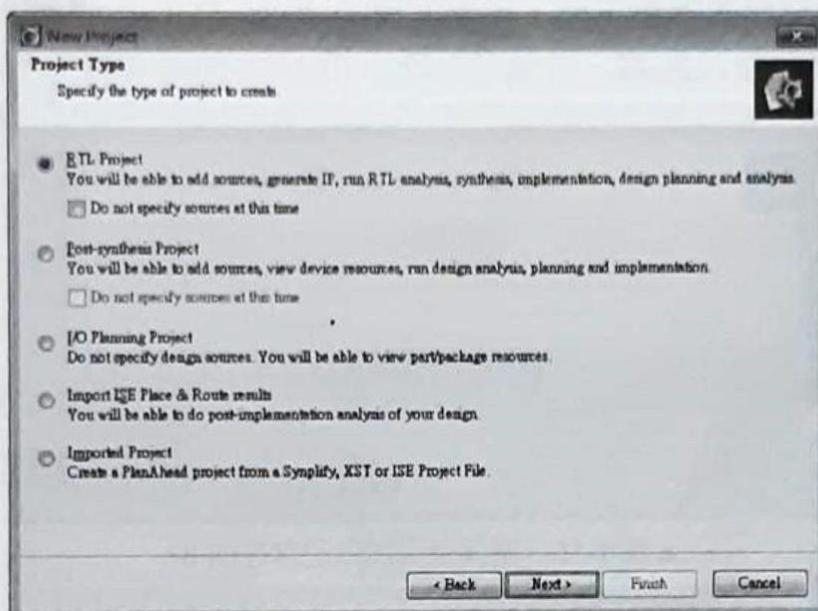
3 輸入專案名稱和專案所儲存的位置，完成後點擊 Next 繼續。



▲圖 1-12 專案名稱及位置

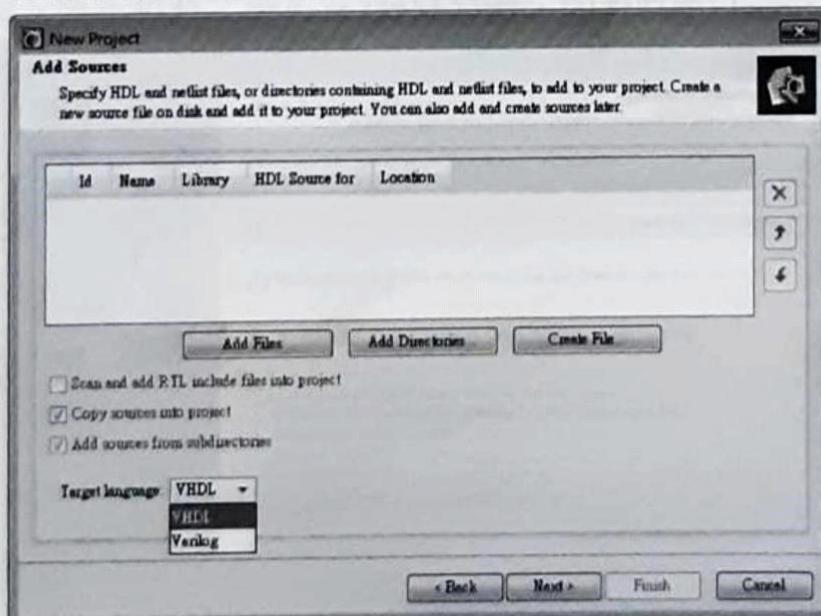
4. 選擇專案類型，選第一個 RTL Project，完成後點擊 Next 繼續

補充說明：PlanAhead 可以開啓其它種類的專案，例如：已在 ISE 其他軟體開啓過且完成 FPGA Placement and Routing (P&R)的專案、合成電路完畢的專案 (Post-synthesis Project)與設定 IO 接腳的專案等。在此 RTL 專案指的是將撰寫 Register Transfer Level (RTL)硬體描述語言來進行硬體電路的設計。



▲圖 1-13 專案類型

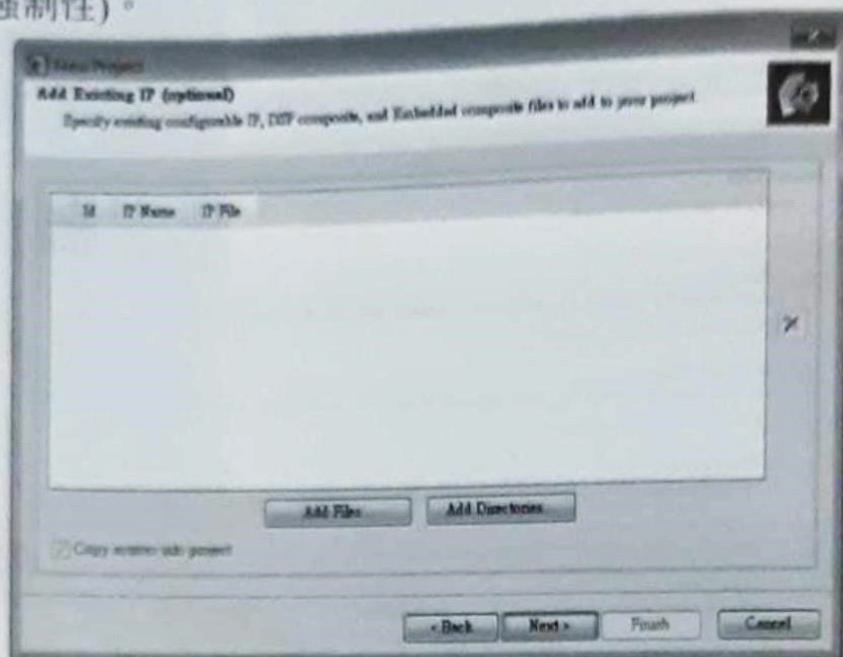
5. 左下方 Target language 選項影響後面 Top HDL 產生的硬體描述語言，在此選擇 VHDL，完成後點擊 Next 繼續。



▲圖 1-14 設定 Target language

6. Add Existing IP 畫面直接點擊 Next 繼續。

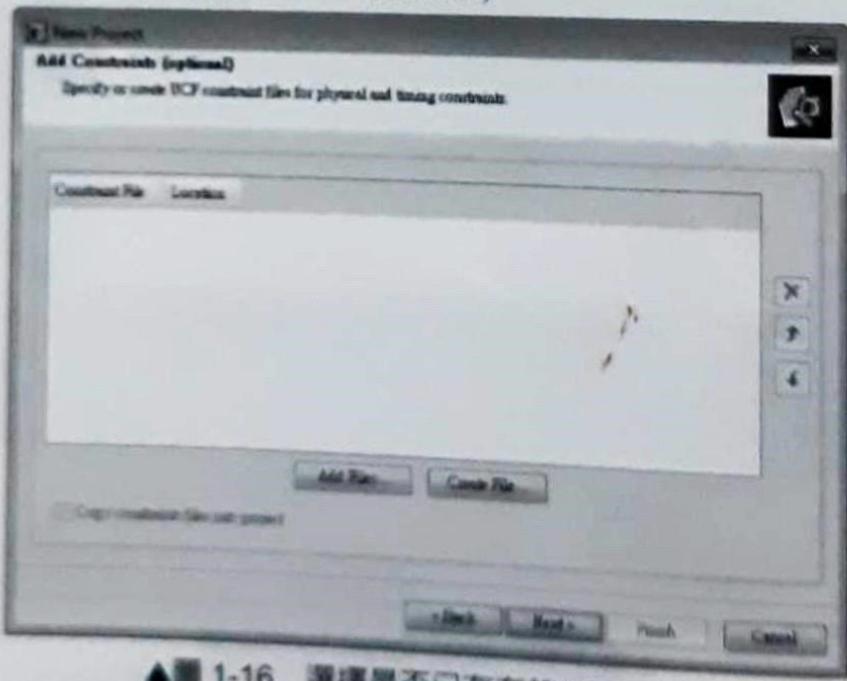
補充說明：由於要新建 IP，因此直接按 Next 繼續。本頁設定為 Optional (選擇性的，無強制性)。



▲圖 1-15 選擇是否加入已存在的 IP

7. Add Constraints 畫面直接點擊 Next 繼續。

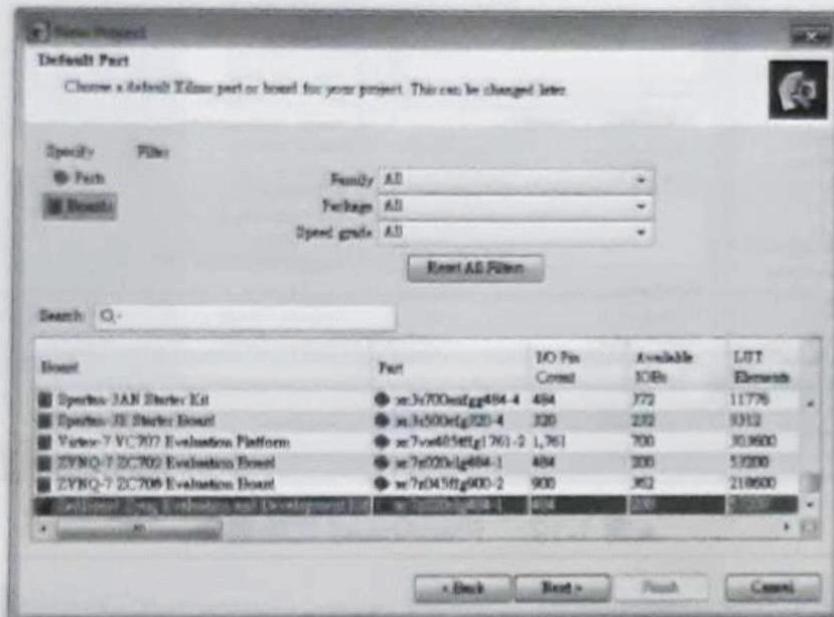
補充說明：Constraints 包含腳位設定以及電路合成與 P&R 的條件設定，在先使用預設的條件來合成與實作電路，腳位設定的部分，待後面再來描述。本頁設定為 Optional (選擇性的，無強制性)。



▲圖 1-16 選擇是否已存在的 UCF 檔案

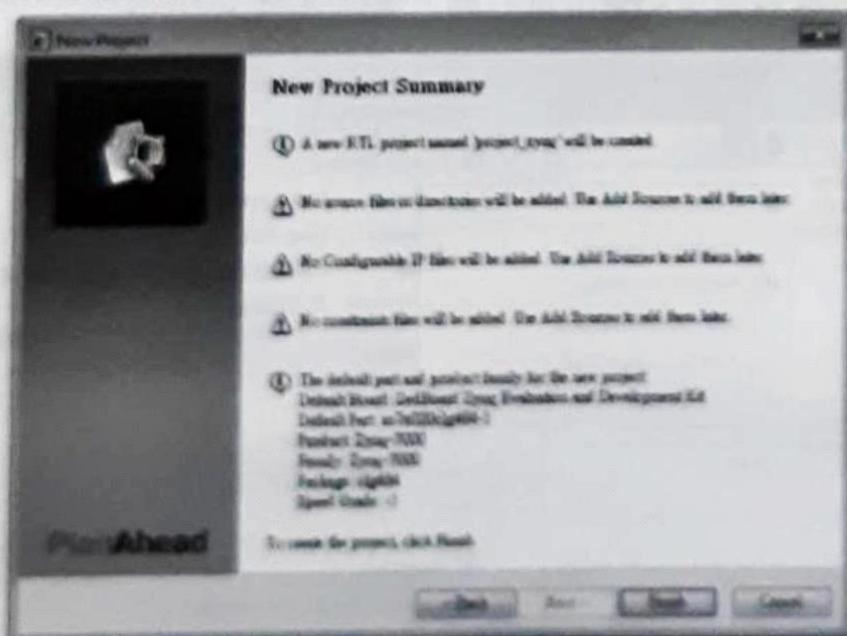
8. 選擇所用的開發板，左邊的 Specify 選擇 Boards，Family 選擇 Zynq，Board 選擇 Zedboard。

補充說明：每一塊電路板都可以把周邊裝置相關之設定存成檔案，這些設定包含使用到的 SoC 或 FPGA 型號、系統時脈頻率、周邊 IO、周邊使用到的中斷編號與周邊使用到的實體記憶體位置對應表等。透過設定檔，EDA 工具可以針對這塊電路板產生 RTL 或者 C 程式樣板，而不會和其他裝置衝突，並且方便開發人員直接修改這些 RTL 和 C 程式原始碼，跳過對電路板的設定過程。



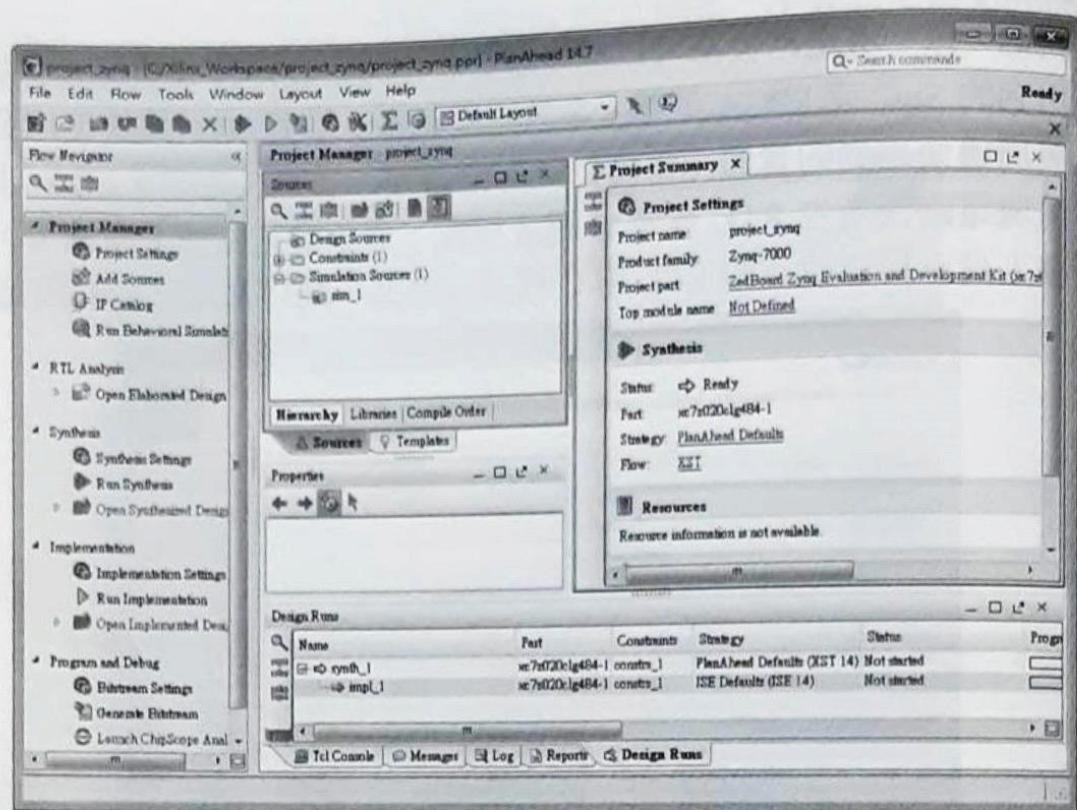
▲圖 1-17 選擇 Zedboard

9. Summy 在創立前重新看一遍是否有設定錯誤。



▲圖 1-18 專案創立前的總結

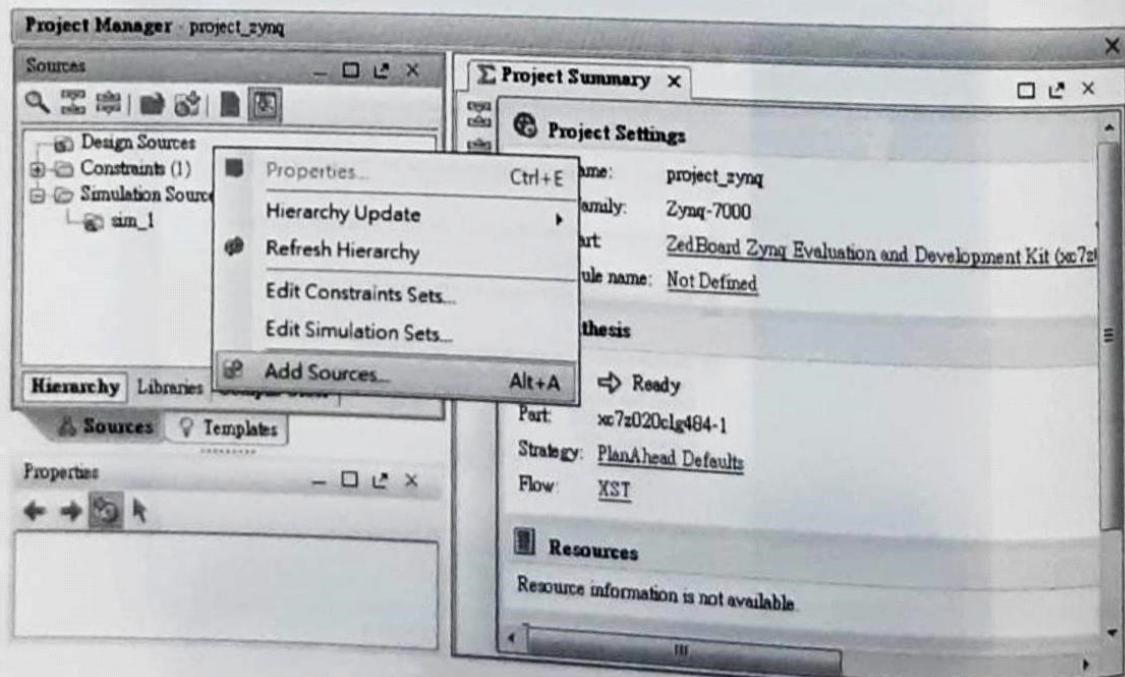
10. 完成專案創建後，如圖 1-19 之 PlanAhead 視窗所示。



▲圖 1-19 PlanAhead 起始畫面

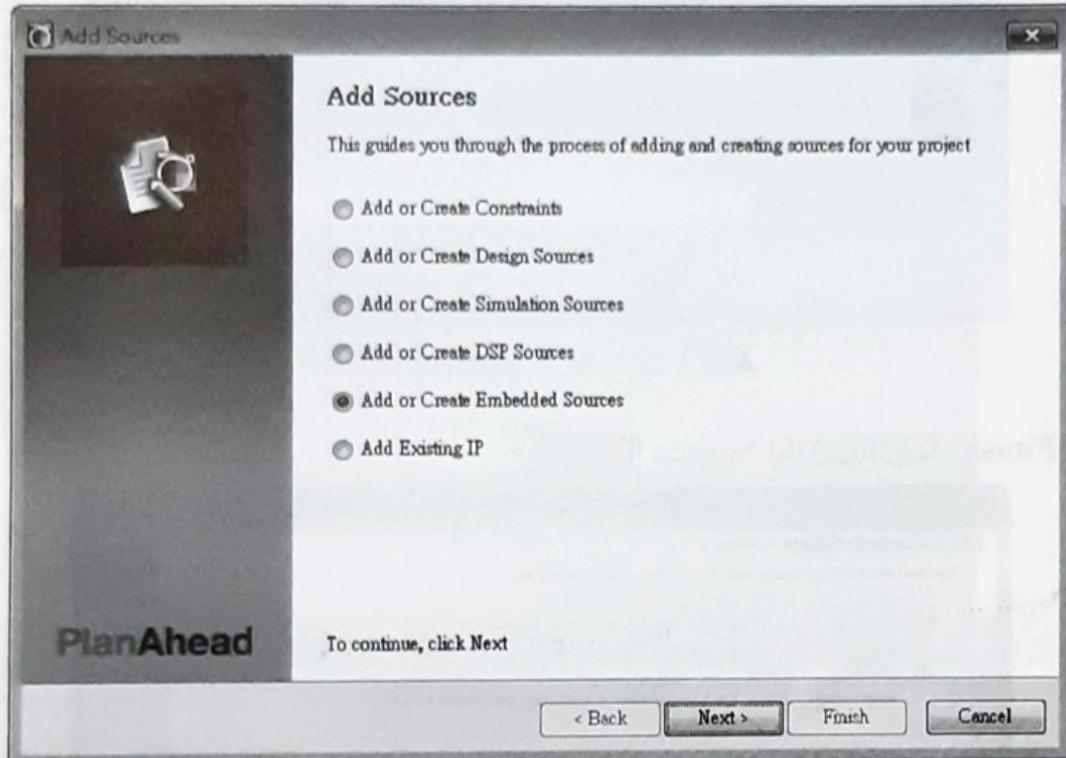
1-4 創立 XPS 專案

1. 首先在 Sources 視窗中點擊滑鼠右鍵，選擇 Add Sources。



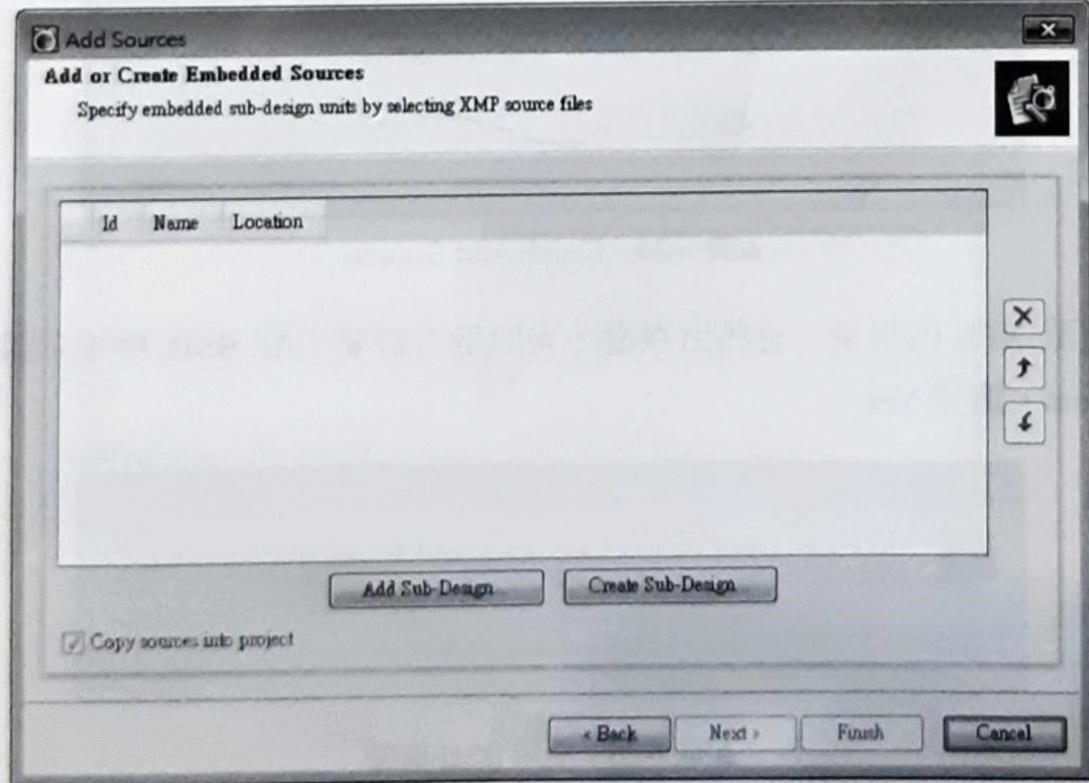
▲圖 1-20 Add Sources...

2. 接著選擇 Add or Create Embedded Sources，點擊 Next 繼續。



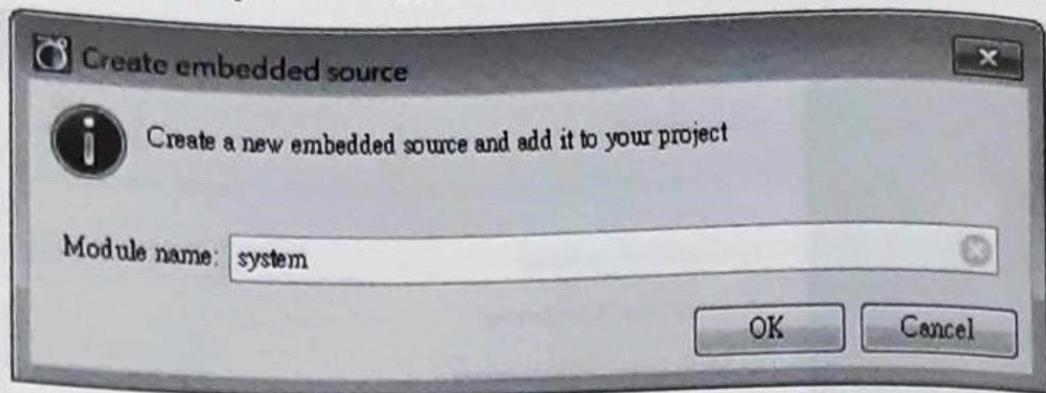
▲圖 1-21 Add or Create Embedded Sources

3. 點擊 Add Sub-Design。



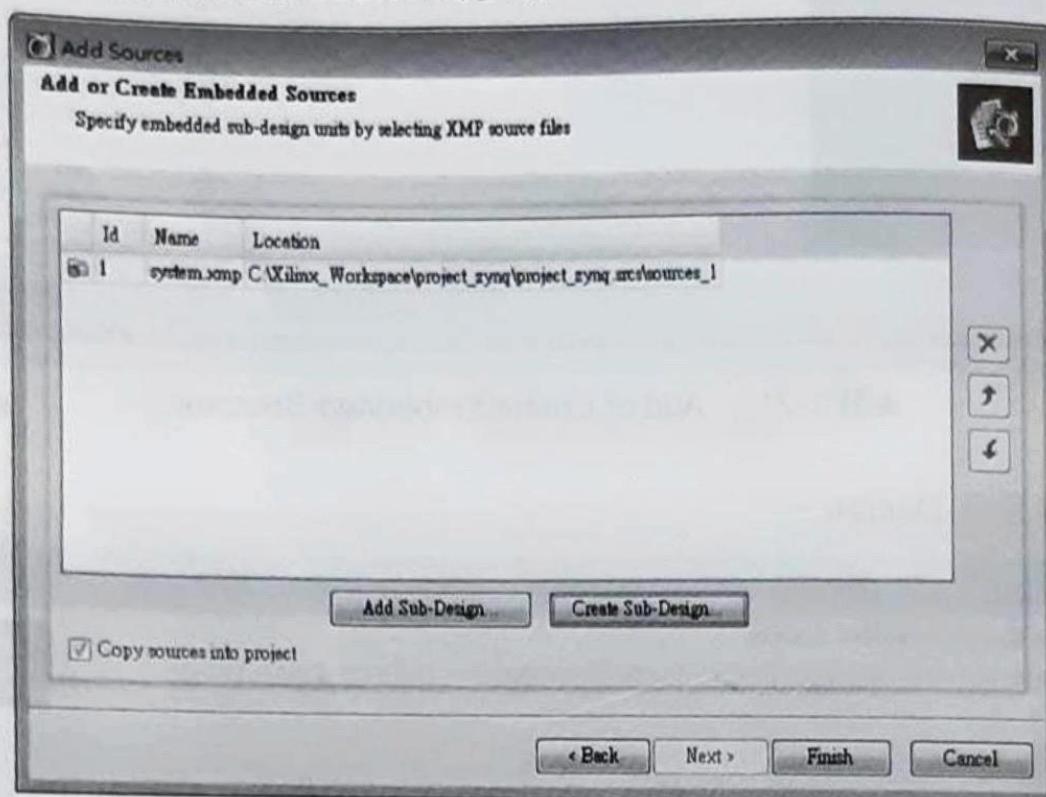
▲圖 1-22 Create Sub-Design

4. 輸入 Module name "system"，點擊 OK 完成建立。



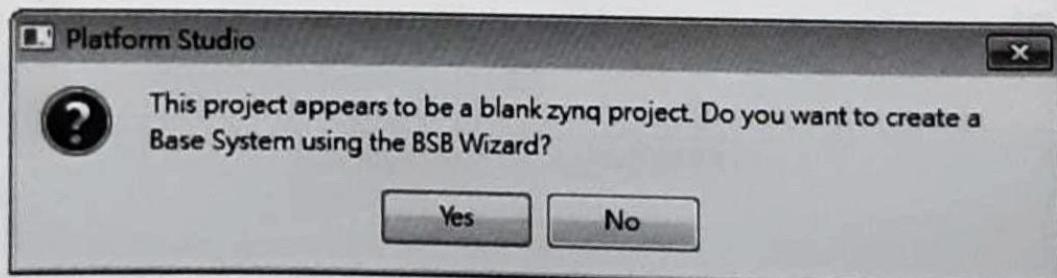
▲圖 1-23 輸入 Module name

5. 點擊 Finish 來完成 Add Source 的步驟。



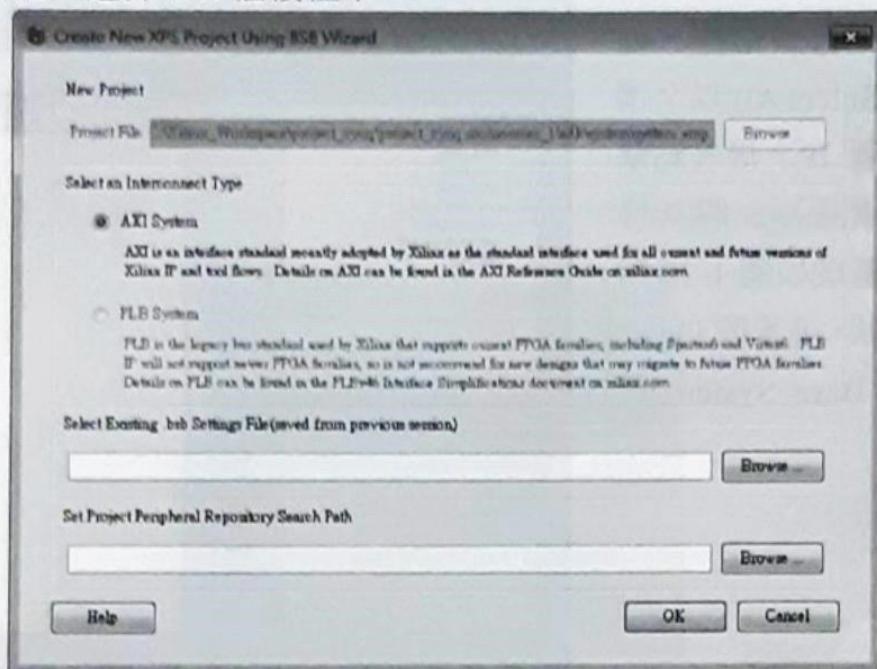
▲圖 1-24 Finish Add Source

6. 切換至 XPS 的視窗，會跳出視窗，詢問是否需要使用 BSB 精靈來創建 Base System，選擇 Yes。



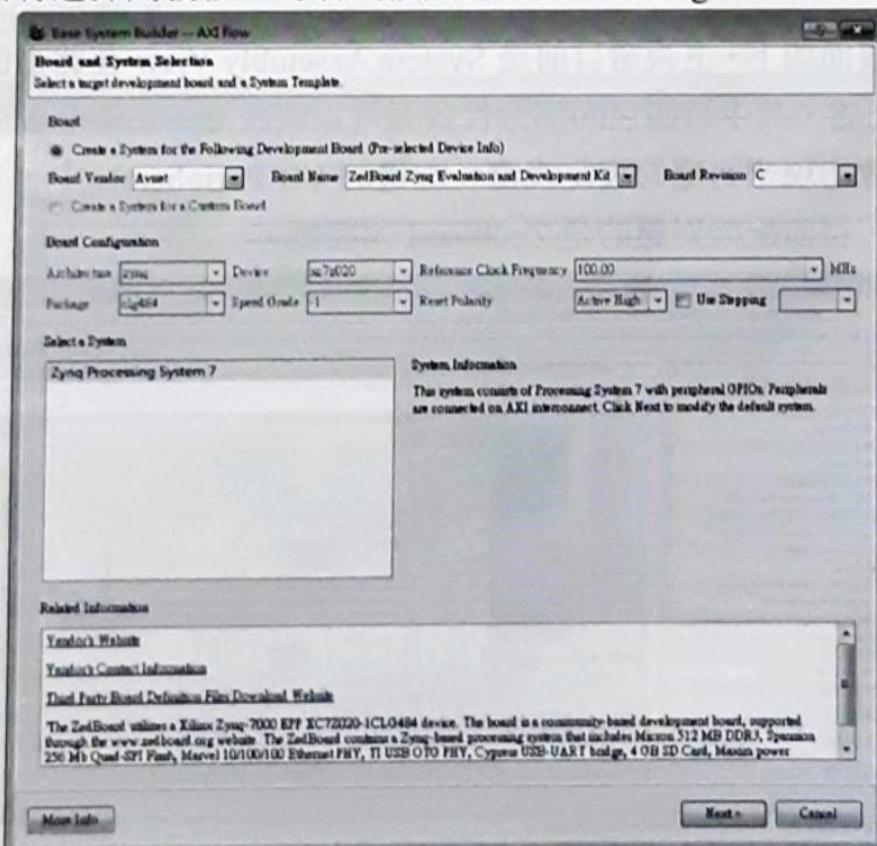
▲圖 1-25 使用 BSB 精靈

7. 維持預設值，選擇 OK 繼續往下。



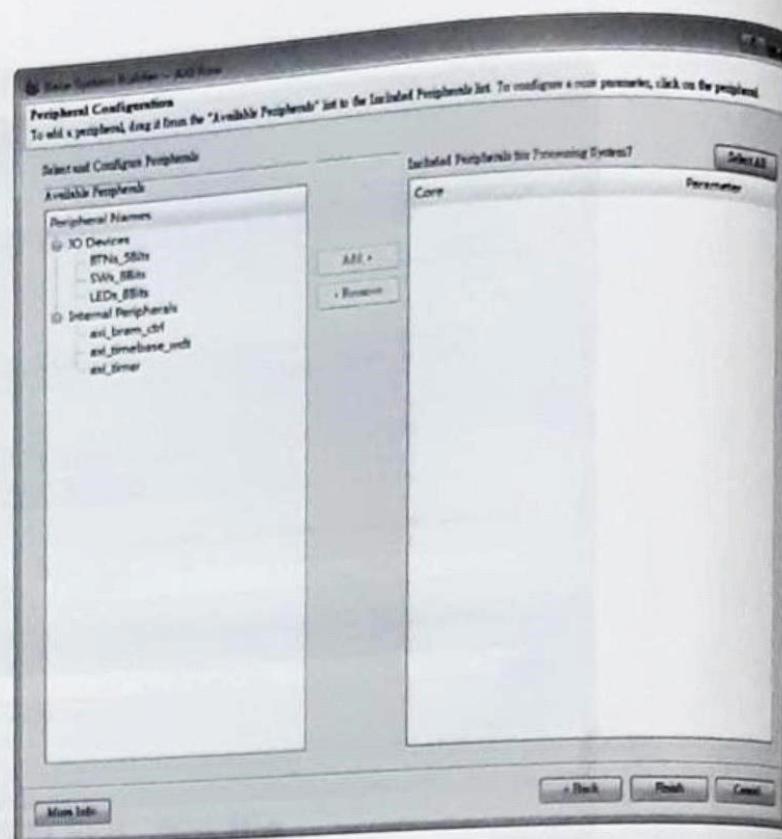
▲圖 1-26 BSB 精靈

8. 由於前面有選擇開發板，可看到預設好的值，Configuration 為灰色無法改變。



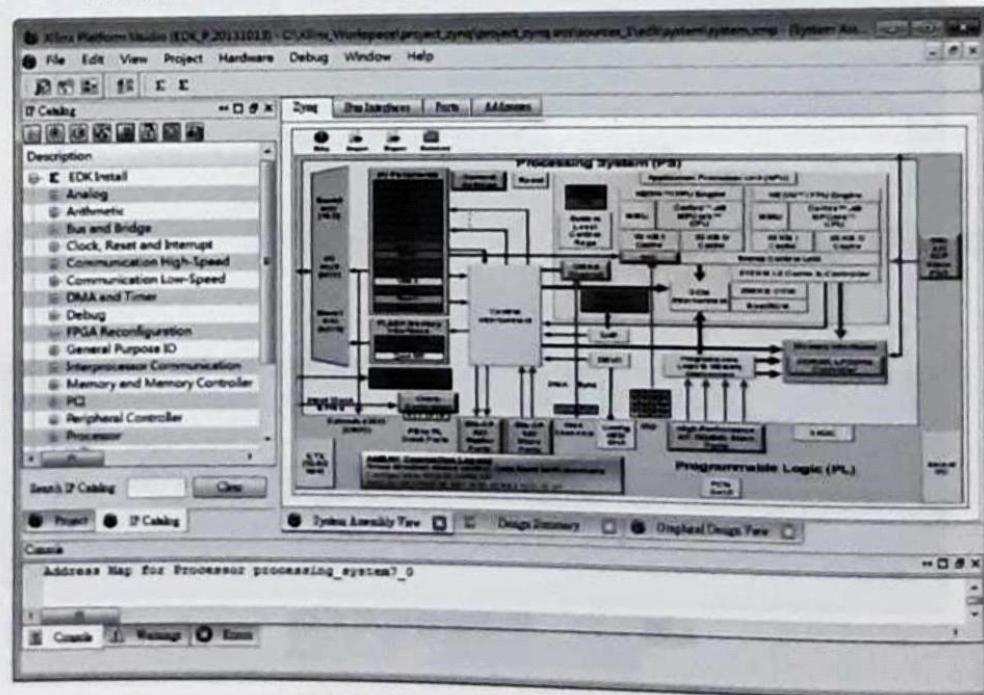
▲圖 1-27 選擇開發板與系統

9. 系統有預設的三個周邊裝置，在此不需要這些，點擊右上方 Select All 以全選預設的三個 IP，接著點擊右中間 < Remove 將其移除，右邊區域如圖 1-28 所示呈空白狀，接著按 Finish 完成整個 Base System 設定。



▲圖 1-28 移除預設的 IP

10. 完成後的畫面如下，主視窗目前是 System Assembly View 在此畫面可以看到有關 PS 的配置，其中有顏色的部分代表著可以做設定，點擊想要做設定的黃色方框，就會出現相對應的設定視窗，左邊的 I/O Peripherals 中，有顏色的代表有被啓用，目前配置對應的是 Zedboard 開發板。

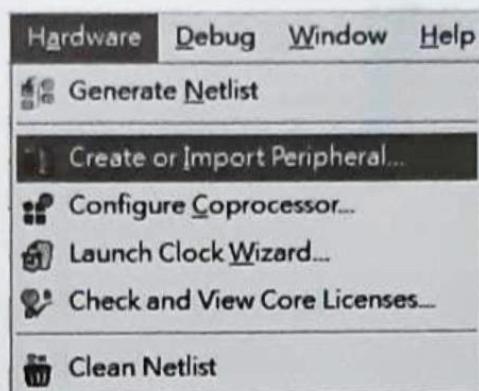


▲圖 1-29 XPS 的主畫面

1-5 IP 的創立與撰寫

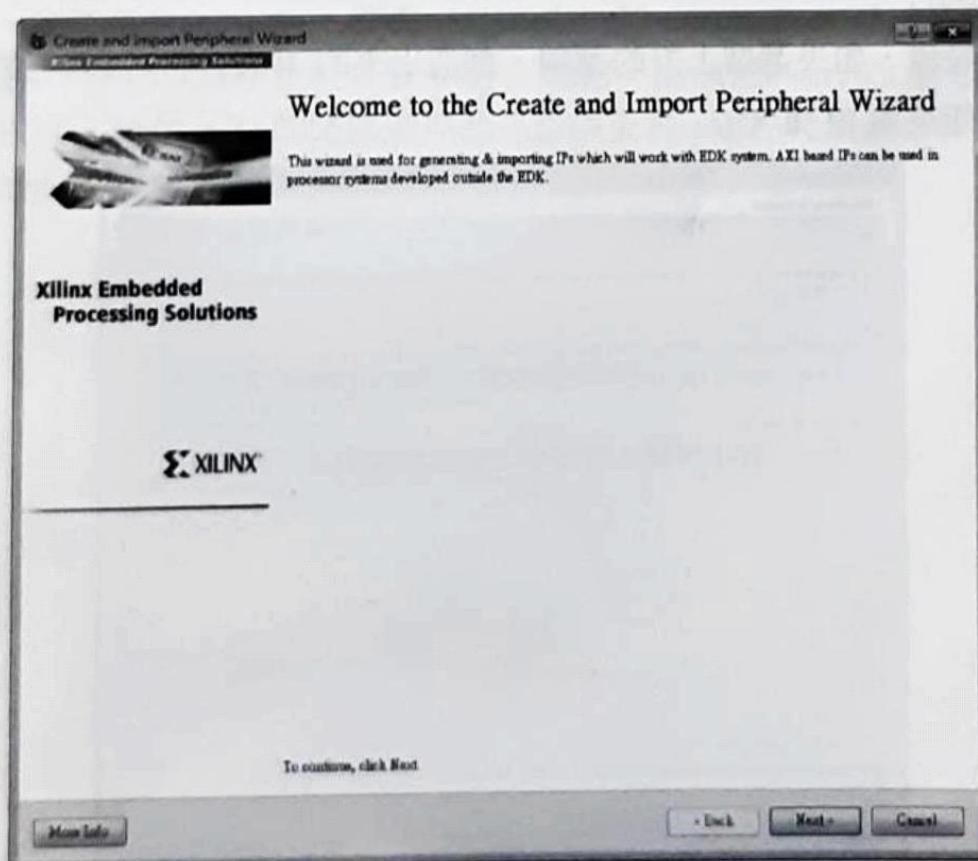
1-5-1 創立 IP

- 點選上方工具列的 Hardware → Create or Import Peripheral(CIPW)。



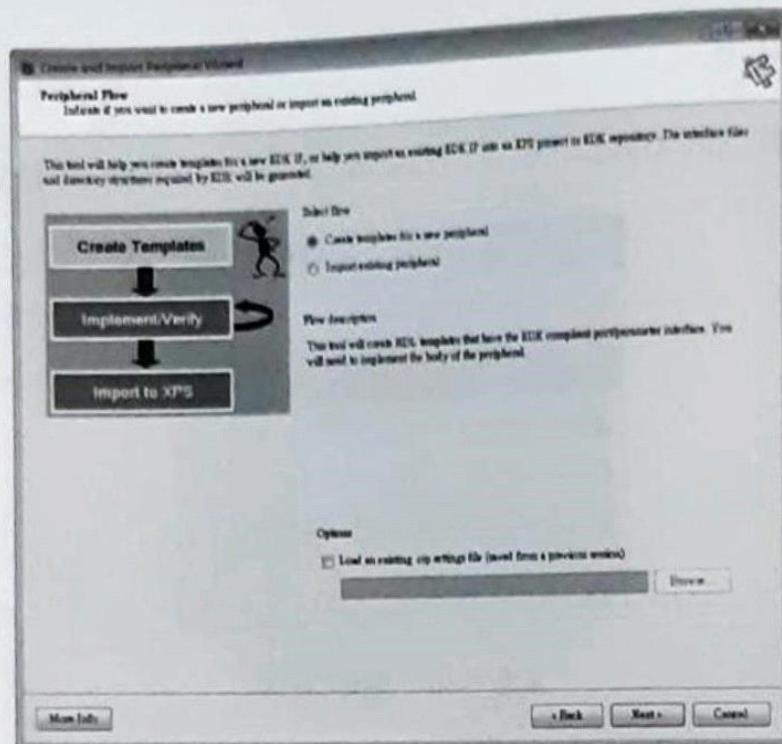
▲圖 1-30 選擇 CIPW

- 歡迎畫面直接跳過。



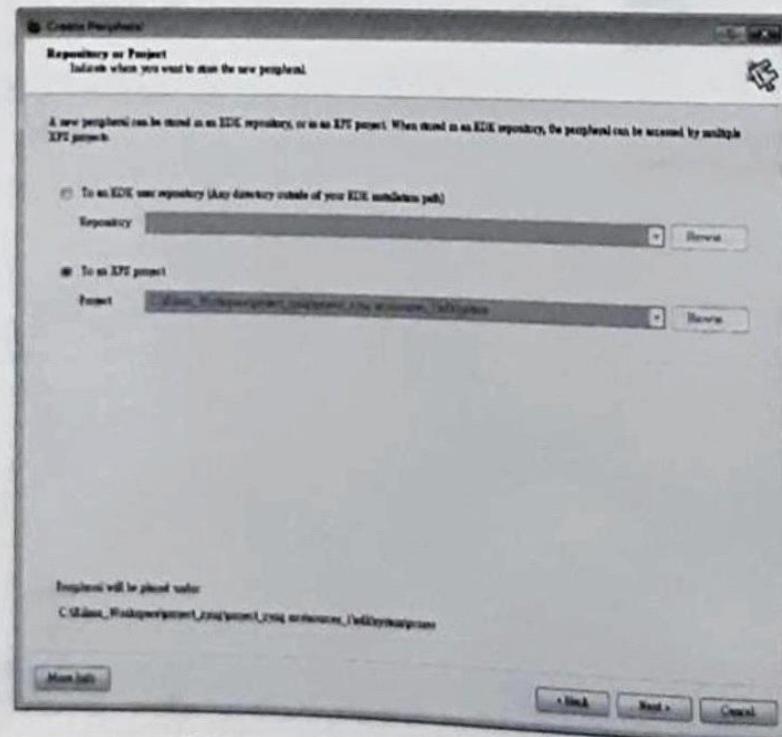
▲圖 1-31 CIPW 的歡迎畫面

3. 創造或是輸入現有的 IP，在此要創造一個新的 IP，所以直接點 Next >前往下一步。



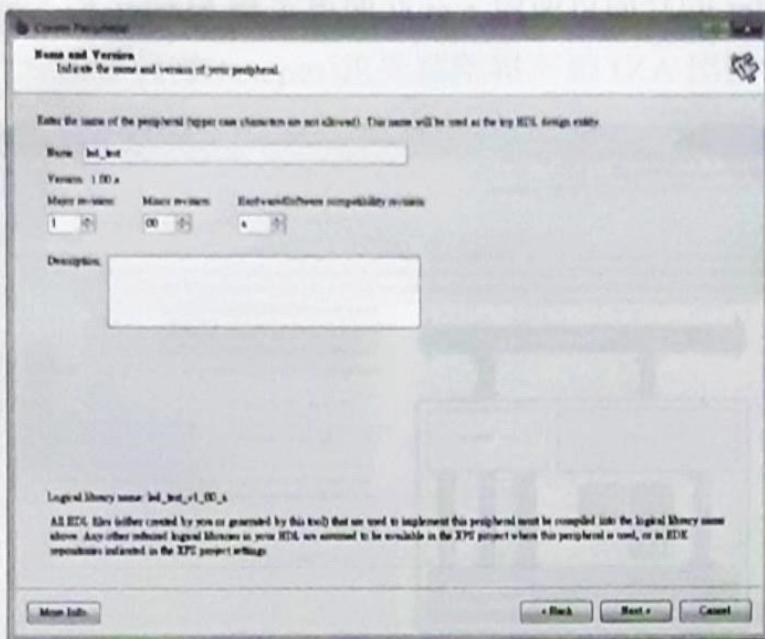
▲圖 1-32 新增或匯入現有的 IP

4. 維持預設值，如果選擇上方的選項，創造出來的 IP 將可以被其它的專案所使用，亦即視為 IP 庫。



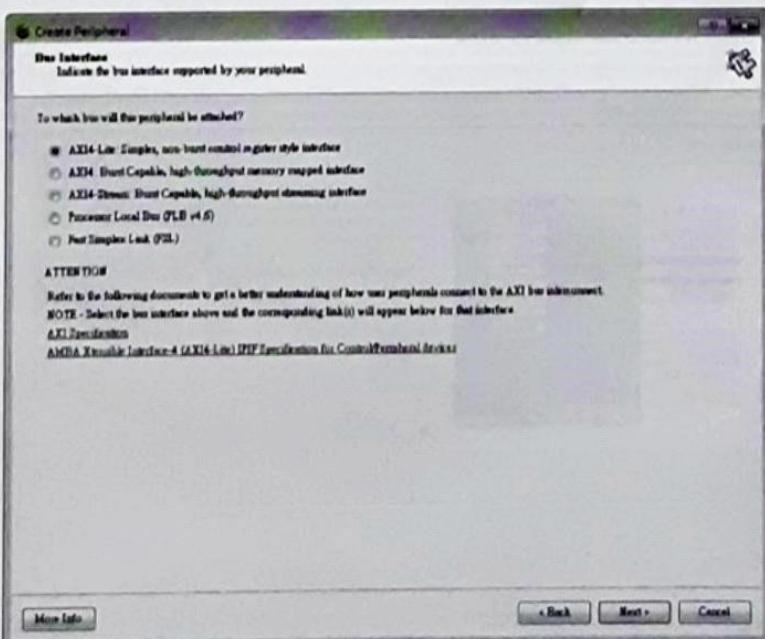
▲圖 1-33 選擇此 IP 是否成為資料庫

5. 輸入 IP 的名稱、版本和此 IP 的相關說明，主要是為了專案管理中，版本控管所設計的。



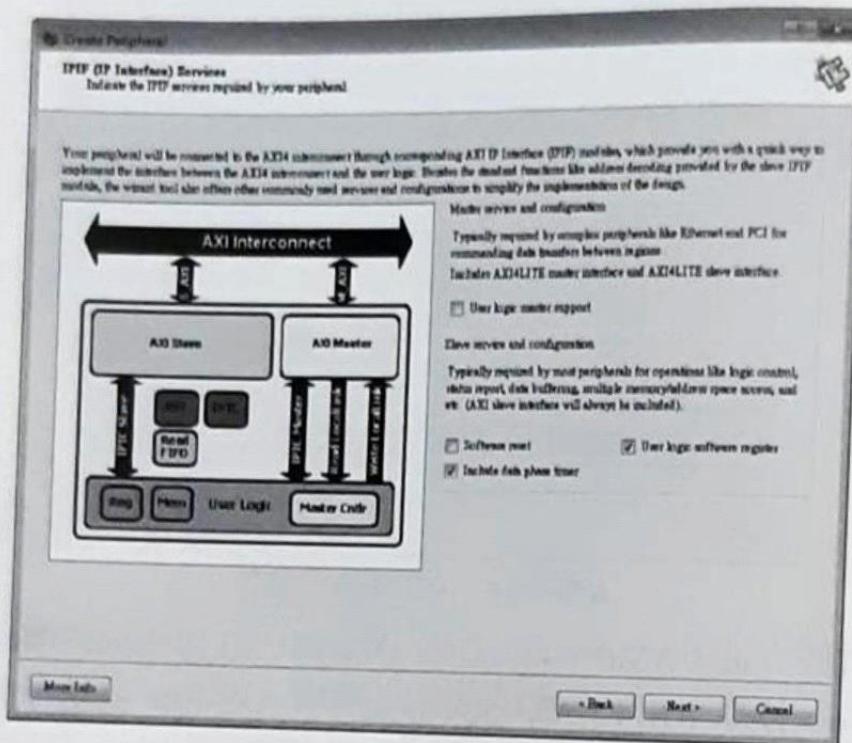
▲圖 1-34 IP 的名稱和版本

6. 選擇匯流排的介面：AMBA 4.0 以後，內部的 AXI 汇流排標準(AXI4)還可細分為 AXI-Lite、標準 AXI4、AXI-Stream，其中 AXI-Lite 就是每個 transaction 只包含一筆資料的 transfer；標準 AXI4 中一個 transaction 可以爆發(burst)多筆 transfers，例如：連續位址多筆 4-byte 傳輸。而 AXI4-stream 會包含讀與寫的緩衝 FIFO，每個 AXI 的時脈週期都可以傳資料，這點特別適用於網路傳輸(例如：Ethernet、無線網卡)中大量傳輸連續性的封包內容。



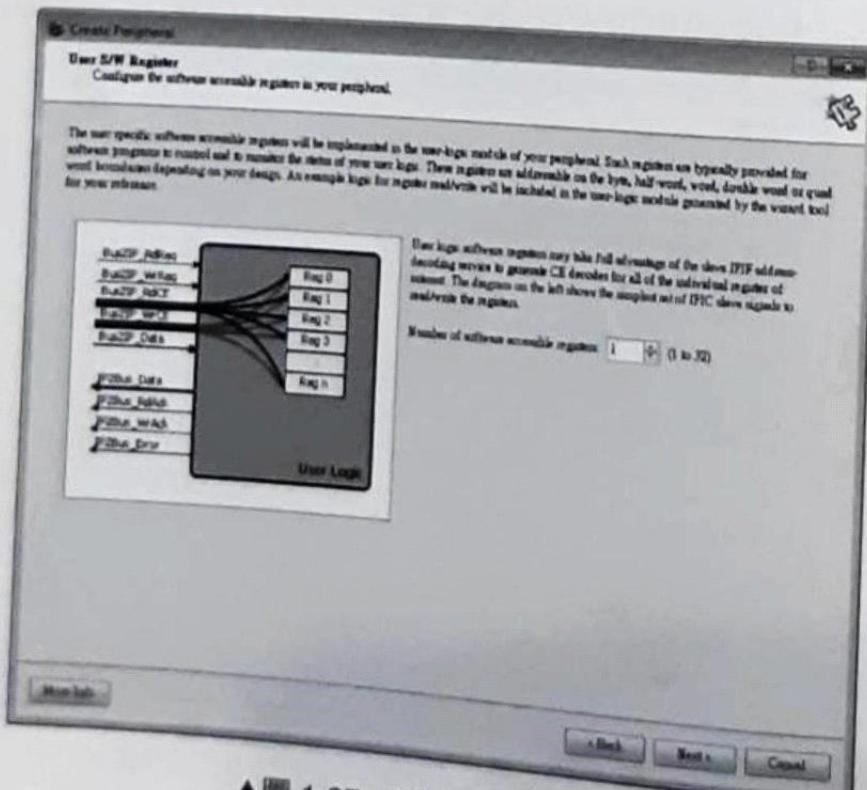
▲圖 1-35 選擇要使用的匯流排介面

7. 如果要創的 IP 是 master，則務必要勾選上方的 User logic master support，否
將不會有 master 的介面可使用。在此簡單定義 Master 和 Slave 的 IP，指的
主動和不主動發出 AXI 決流排讀寫要求(request)的矽智財。



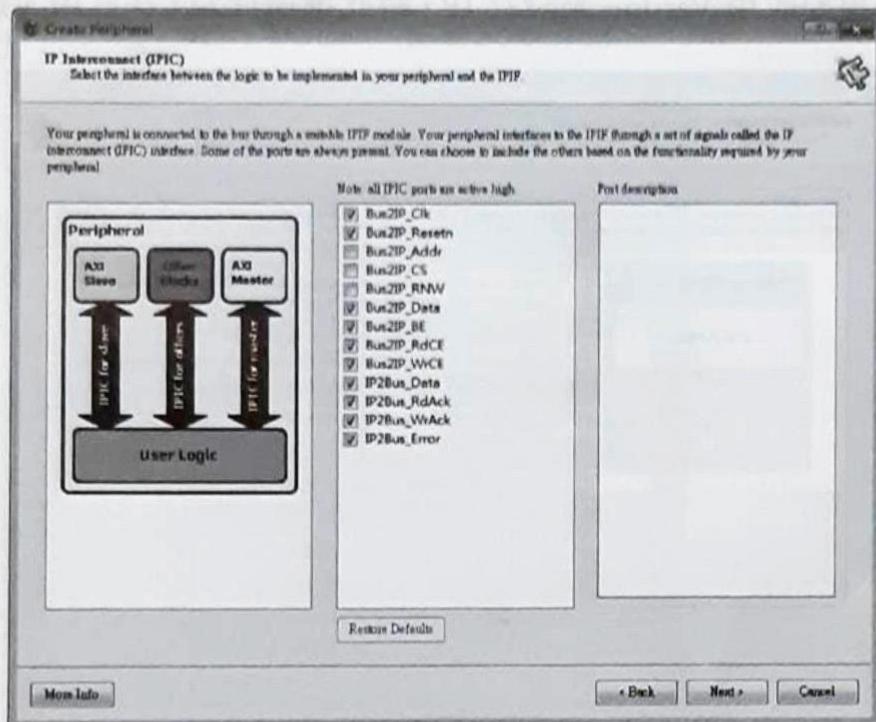
▲圖 1-36 IP 介面的指示

8. 設定暫存器的數量，每個暫存器都是 32 個 bit。



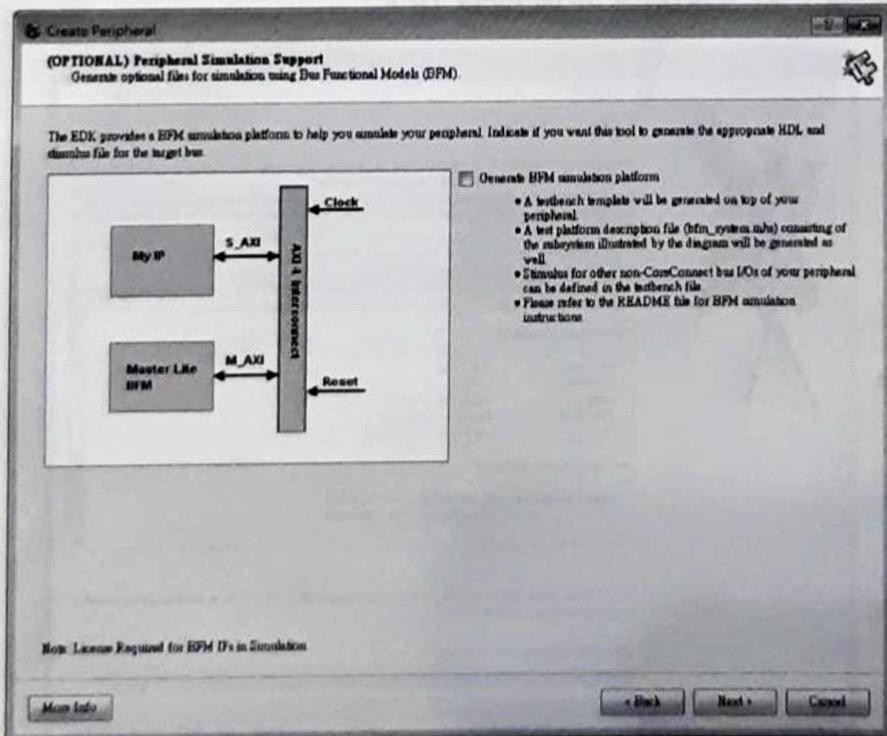
▲ 1-37 選擇暫存器數量

9. IP 的腳位。



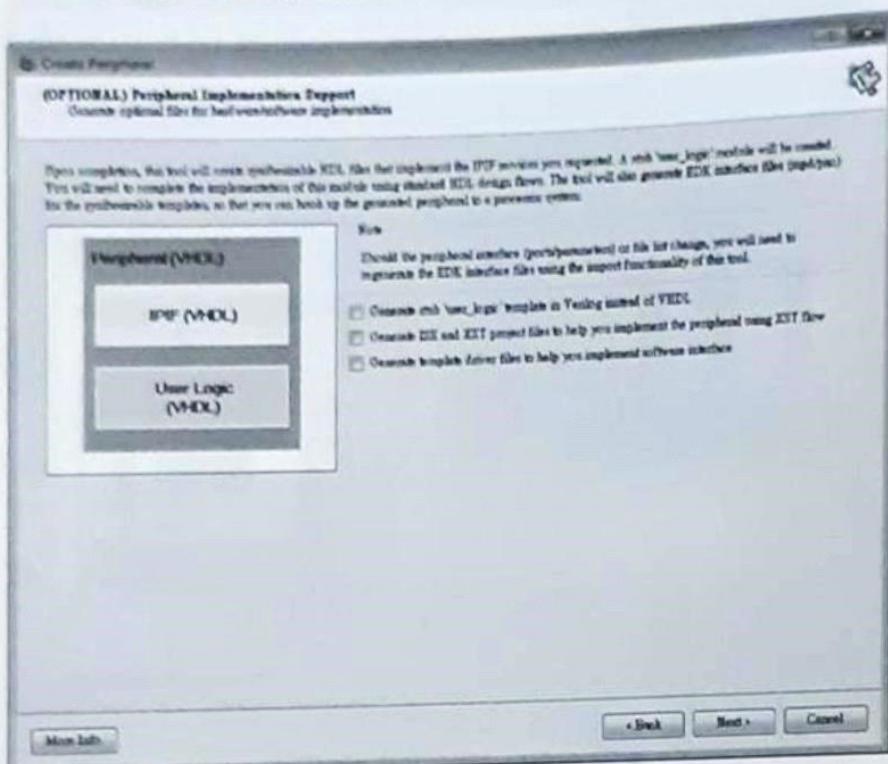
▲圖 1-38 IPIC 的設定畫面

10. 圖 1-39 視窗為選擇性內容，目前不需使用。



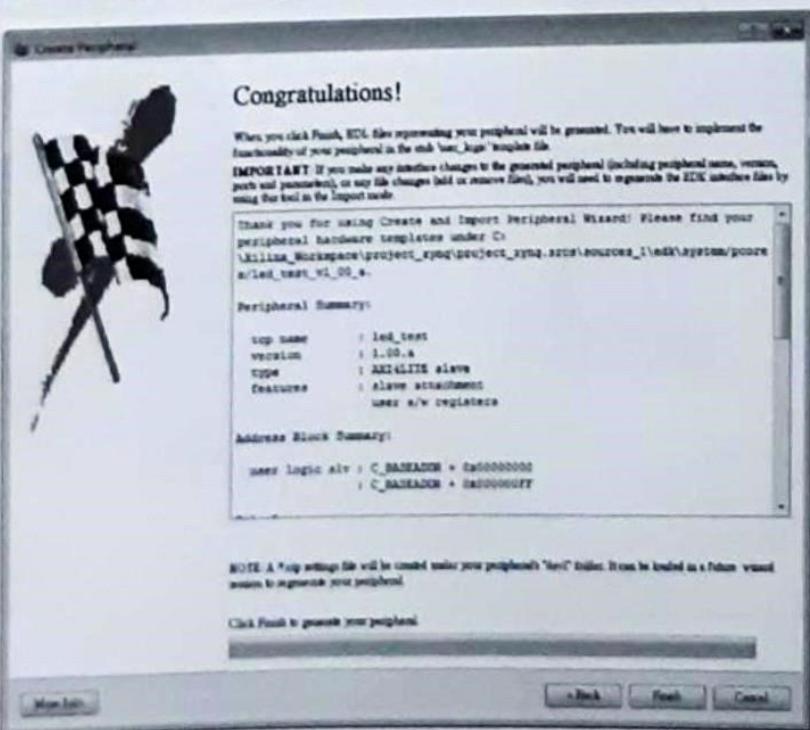
▲圖 1-39 設定模擬

11. 圖 1-40 的三個選項特別重要，第一個選項是預設使用的硬體描述語言是 VHDL，如果要改用 Verilog 來設計 IP，將此選項打勾，在此用 VHDL 做範例。



▲圖 1-40 周邊的支援

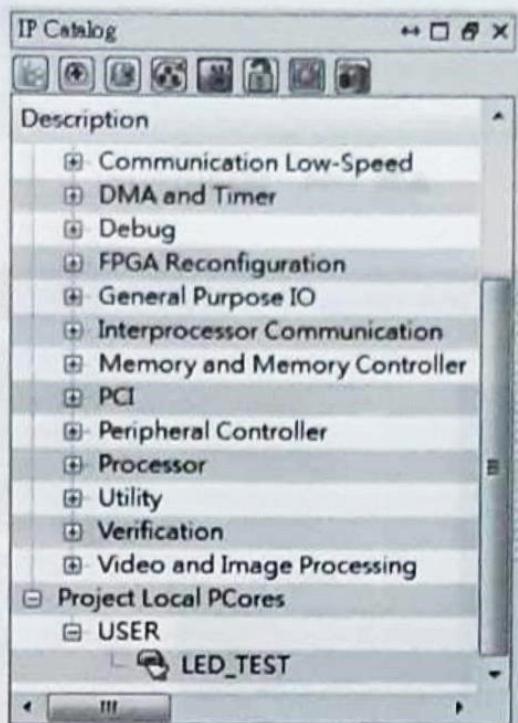
12. 檢查設定無誤，點選 Finish 完成創建 IP。



▲圖 1-41 創立前的回顧

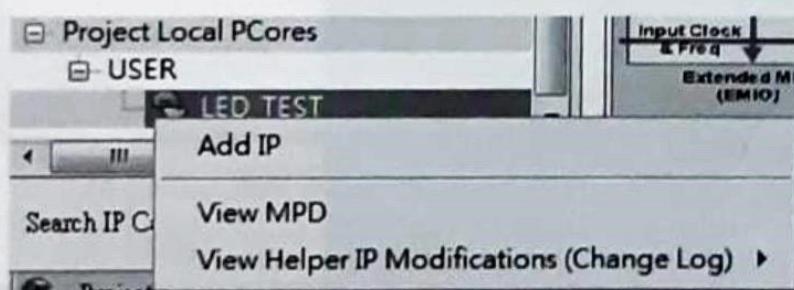
1-5-2 將 IP 加入系統中

- 點開 IP Catalog 最下方的 USER，會看到剛才建立的 IP。



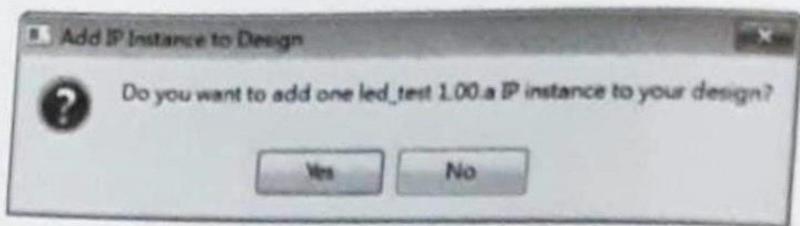
▲圖 1-42 IP Catalog

- 點擊兩下選擇 Add IP，將剛才創立的 IP 加入至系統中。



▲圖 1-43 將創立好的 IP 加入系統

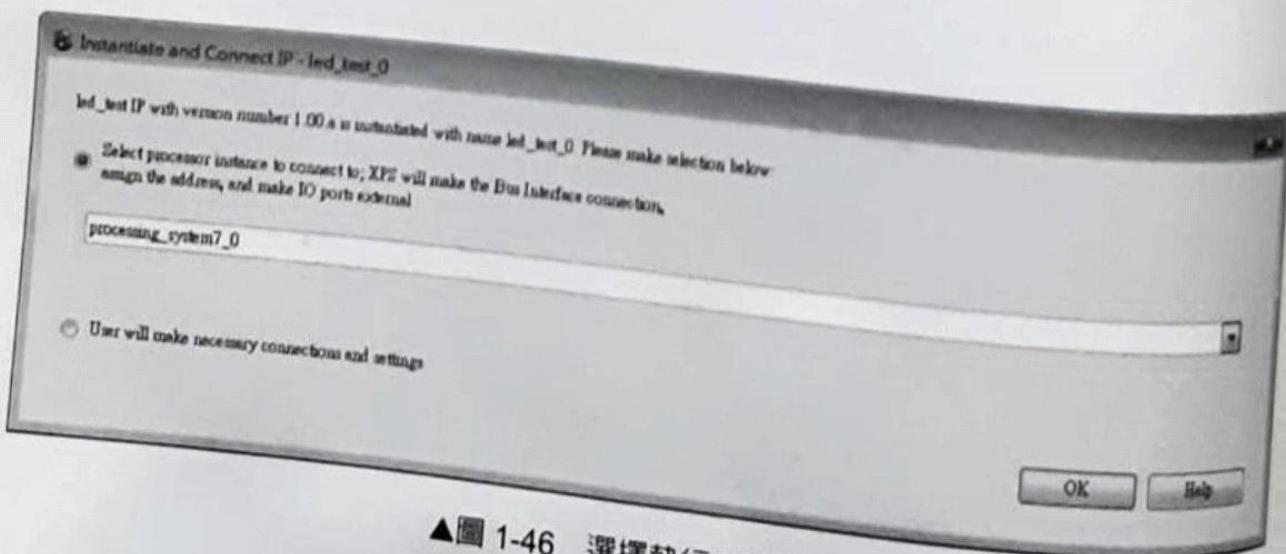
3. 圖 1-44 至圖 1-46，選擇 OK。



▲■ 1-44 確認加入 IP



▲■ 1-45 IP 的設定畫面



▲■ 1-46 選擇執行 IP 的核心

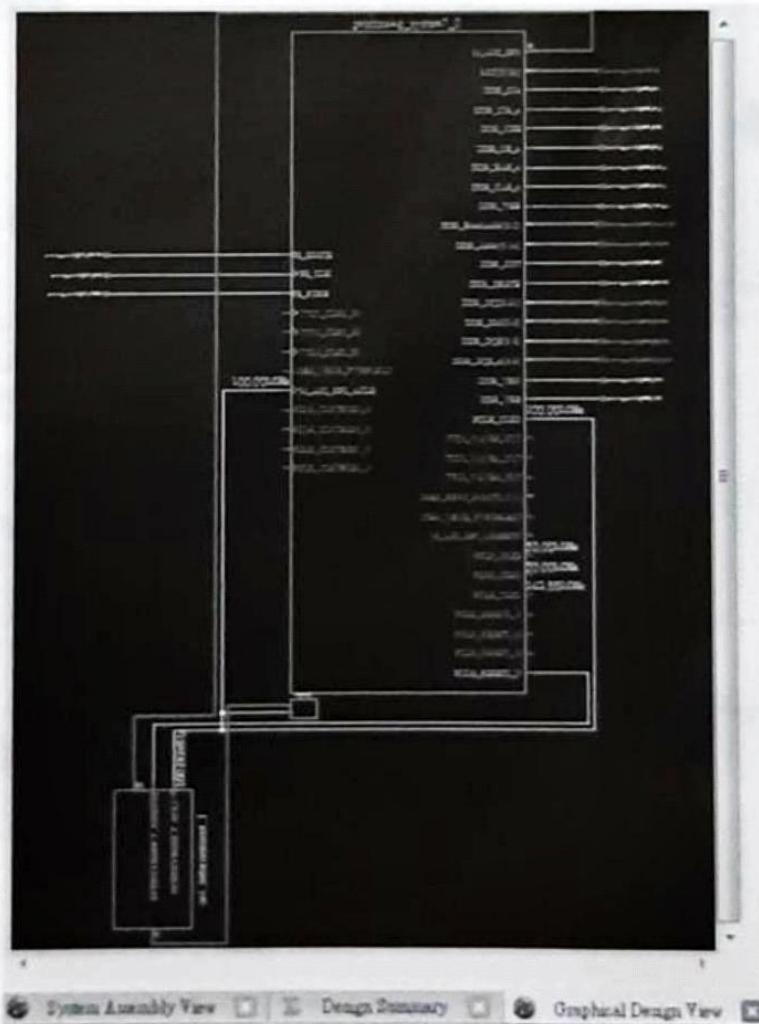
4. 在 Ports 確認是否有把 IP 加入。



▲圖 1-47 Ports 的畫面

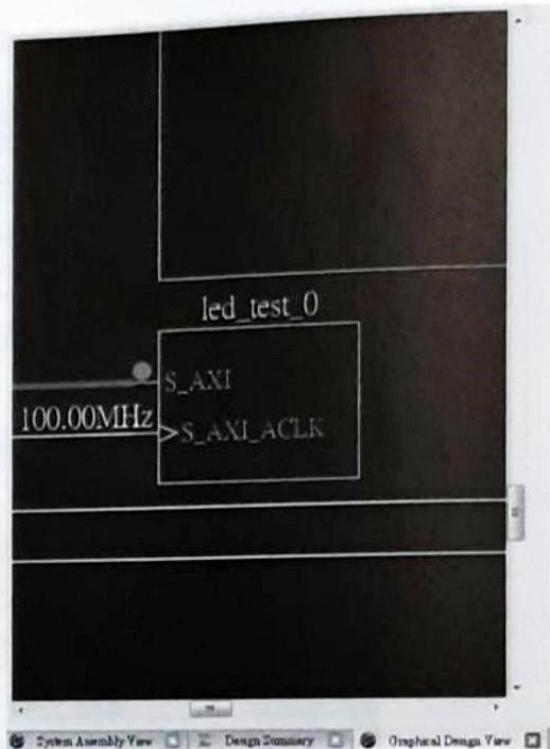
5. 在 Graph Design View 中，會多一個 AXI 的控制器與剛才加入的 IP。

- (1) 單點往右下移為局部放大。
- (2) 單點往左上移為全景觀看。
- (3) 單點往左下移為放大。
- (4) 單點往右上移為縮小。



▲圖 1-48 觀看系統設計

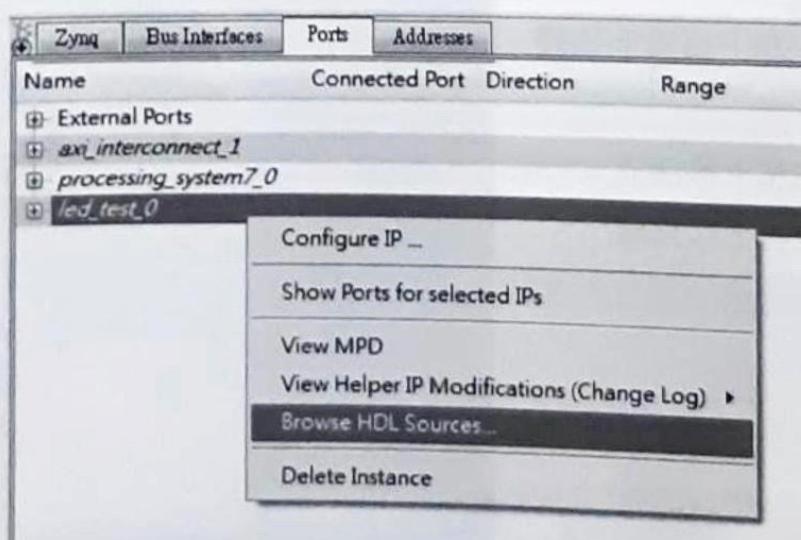
6. 放大觀看剛才建立的 IP，可看到此 IP 只有 AXI Bus 的控制線，沒有其它的輸入輸出。



▲圖 1-49 觀看加入系統的 IP

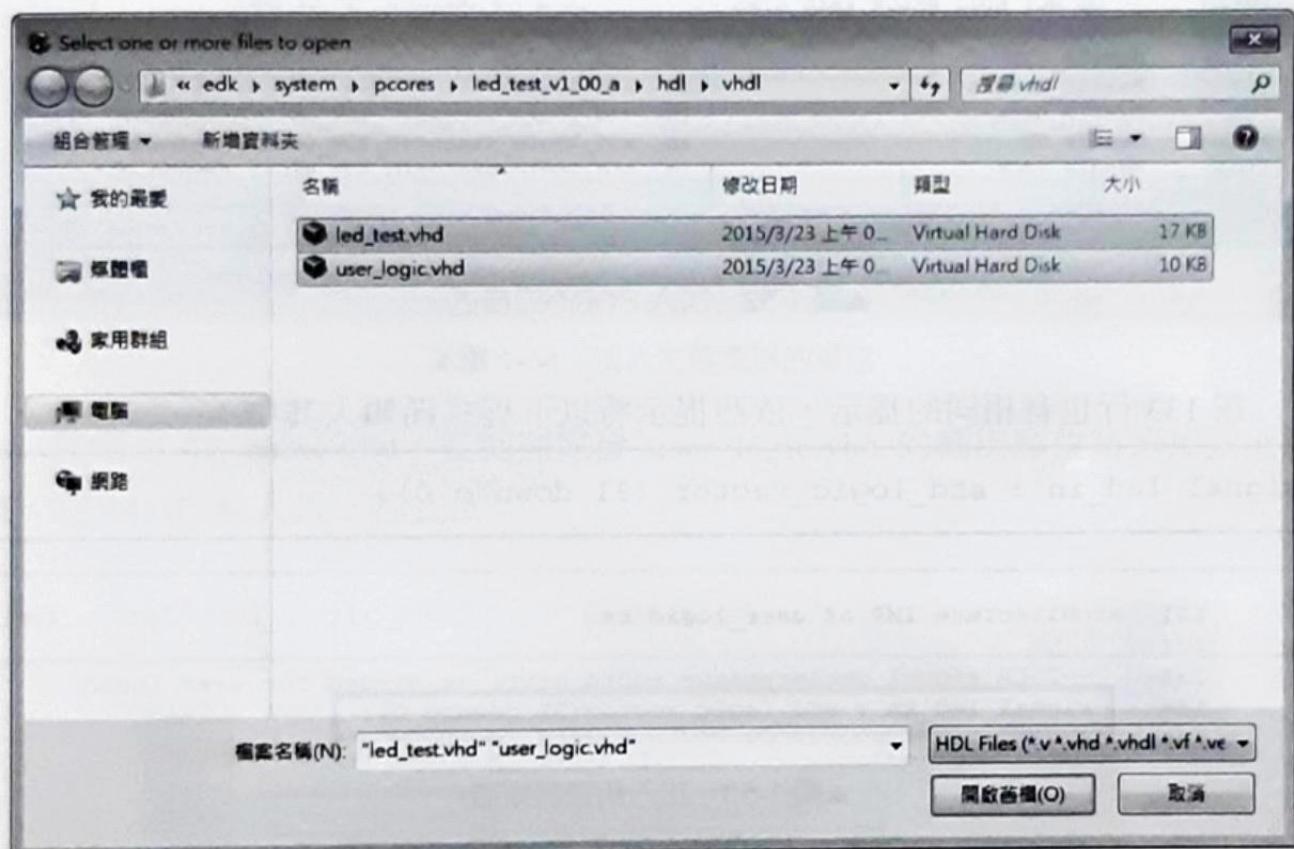
1-5-3 撰寫 IP

現在要開始來修改此 IP。回到 Ports，對 IP 點擊右鍵，選擇 Browse HDL Sources...



▲圖 1-50 開啓電路檔案

有兩個 vhd 檔案，把兩個檔案一起開啟，其中 user_logic.vhd 檔案是最底層的 module，硬體就是設計在裡面，而 IP_name.vhd 是上一層的 module，如果在 user_logic.vhd 有宣告輸入輸出 port，則在 IP_name.vhd 中也需要將其連接上來。



▲圖 1-51 選擇要開啓的 VHDL 檔案

在原始 99 行到 101 行的地方有提示，ADD USER PORTS BELOW THIS LINE 和 USER ports added here，依照提示將所需要的輸出腳宣告在其中：

```
led : out std_logic_vector( 7 downto 0);
```

```

97      port
98      (
99          -- ADD USER PORTS BELOW THIS LINE
100         --USER ports added here
101         led : out std logic vector(7 downto 0);
102         -- ADD USER PORTS ABOVE THIS LINE
103
104         -- DO NOT EDIT BELOW THIS LINE --
105         -- Bus protocol ports, do not add to or delete
106         Bus2IP_Clk           : in  std_logic;
107         Bus2IP_Resetn        : in  std_logic;
108         Bus2IP_Data          : in  std_logic_vector(C_SLV_DWIDTH-1 downto 0);
109         Bus2IP_BE             : in  std_logic_vector(C_SLV_DWIDTH/8-1 downto 0);
110         Bus2IP_RdCE           : in  std_logic_vector(C_NTM_REG-1 downto 0);

```

System Assembly View Design Summary Graphical Design View led_test.vhd user_logic.vhd

▲圖 1-52 加入 PORT 的描述

在 133 行也有相同的提示，依照提示將以下程式碼輸入其中：

```
signal led_in : std_logic_vector (31 downto 0);
```

```

131     architecture IMP of user_logic is
132
133     --USER signal declarations added here, as needed for user logic
134     signal led_in : std_logic_vector(31 downto 0);
135

```

▲圖 1-53 加入訊號的宣告

在 148 行開始主要的內容：

```

led <= led_in (31 downto 24);
process(Bus2IP_Clk)
begin
    if Bus2IP_Clk'event and Bus2IP_Clk= '1' then
        led_in <= led_in + '1';
    end if;
end process;

```

```

145 begin
146
147     led <= led_in(31 downto 24);
--USER logic implementation added here
148 process(Bus2IP_Clk)
149 begin
150     if Bus2IP_Clk'event and Bus2IP_Clk = '1' then
151         led_in <= led_in + '1';
152     end if;
153 end process;
154
155
156 -- Example code to read/write user logic slave model s/w accessible registers
157 --
158 -- Note:
159 -- The example code presented here is to show you one way of reading/writing
160 -- software accessible registers implemented in the user logic slave model.

```

System Assembly View Design Summary Graphical Design View led_test.vhd user_logic.vhd

▲圖 1-54 加入主要電路的描述

接著是 IP_name.vhd，要把在底層 user_logic.vhd 的輸出腳位給連接到上層來，在 141 行輸入以下程式碼：

```
led : out std_logic_vector(7 downto 0);
```

```

137 port
138 (
139     -- ADD USER PORTS BELOW THIS LINE -----
140     --USER ports added here
141     led : out std_logic_vector(7 downto 0);
142     -- ADD USER PORTS ABOVE THIS LINE -----
143
144     -- DO NOT EDIT BELOW THIS LINE -----
145     -- Bus protocol ports, do not add to or delete
146     S_AXI_ACLK           : in  std_logic;
147     S_AXI_ARESETN        : in  std_logic;
148     S_AXI_AWADDR         : in  std_logic_vector(C_S_AXI_I
149     S_AXI_AWVALID        : in  std_logic;
150     S_AXI_WDATA          : in  std_logic_vector(C_S_AXI_I
151     S_AXI_WSTRB          : in  std_logic_vector((C_S_AXI_
152     S_AXI_WVALID         : in  std_logic;
153     S_AXI_BREADY         : in  std_logic;
154     S_AXI_ARADDR         : in  std_logic_vector(C_S_AXI_I
155     S_AXI_ARVALID        : in  std_logic;
156     S_AXI_RREADY         : in  std_logic;
157     S_AXI_RREADY         : out std_logic;

```

System Assembly View Design Summary Graphical Design View led_test.vhd

▲圖 1-55 加入 PORT 的描述

在 305 行輸入以下程式碼：

```
led => led,
```

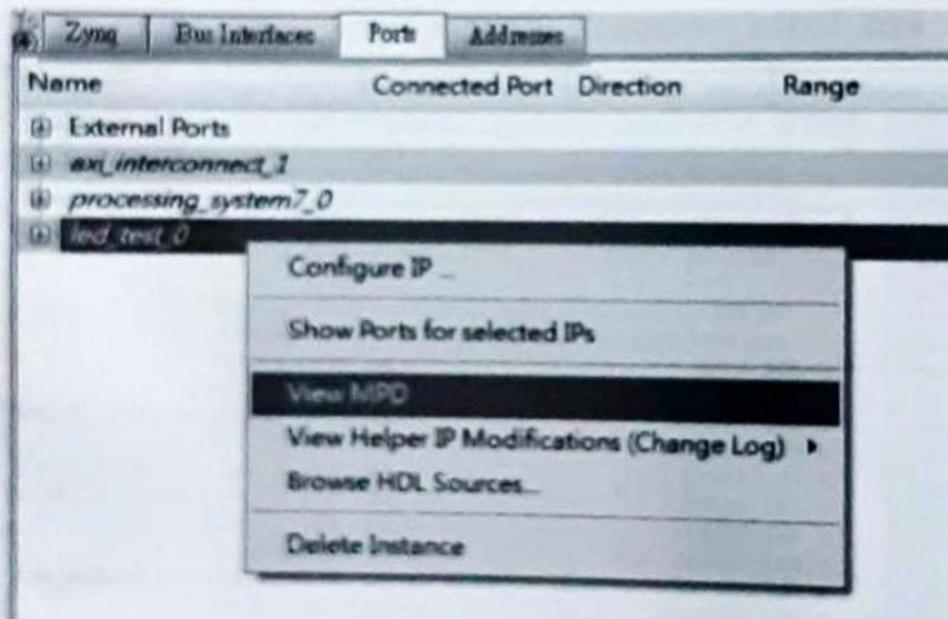
```

302      port map
303      (
304          -- MAP USER PORTS BELOW THIS LINE -----
305          --USER PORTS MAPPED HERE
306          led => led,
307          -- MAP USER PORTS ABOVE THIS LINE -----
308
309          Bus2IP_Clk           => ipif_Bus2IP_Clk,
310          Bus2IP_Resetn        => ipif_Bus2IP_Resetn,
311          Bus2IP_Data          => ipif_Bus2IP_Data,
312          Bus2IP_BE             => ipif_Bus2IP_BE,
313          Bus2IP_RdCE          => user_Bus2IP_RdCE,
314          Bus2IP_WrCE          => user_Bus2IP_WrCE.

```

▲圖 1-56 加入 PORT MAP 的描述

以上步驟都完成後存檔，回到 Ports 底下，右鍵點擊 IP 選擇 View MPD。



▲圖 1-57 開啓 MPD 檔案