

Classification	XPAC SDK FAQ					No.	6-010-00
Author	Sean	Version	1.0.1	Date	2012/10/18	Page	1/13

## 如何將XPacSDK(CE6)函式庫更新為PACSDK函式庫

適用於:

平台	OS 版本	XPAC utility 版本
XPAC 系列 (CE6)	所有版本	所有版本

將XPacSDK函式庫更新為PACSDK函式庫之前，首先需於PC安裝XPAC(CE6)平台開發SDK的安裝套件。請確認提供 PACSDK 函式庫的 SDK 安裝套件版本，其版本號碼需晚於或為 1.4.0。例如，XPacSDK\_CE\_1.4.0\_VS2008.msi or XPacSDK\_CE\_1.4.0\_VS2005.msi

最新版本的 XPAC(CE6)平台開發 SDK 安裝套件，其出貨 CD 及 FTP 下載路徑為

CD:

CD: \XP-8000-CE6\SDK\PlatformSDK

CD: \XPAC-ATOM-CE6\SDK\PlatformSDK

FTP:

<http://ftp.icpdas.com/pub/cd/xp-8000-ce6/sdk/platformsdk/>

<http://ftp.icpdas.com/pub/cd/xpac-atom-ce6/sdk/platformsdk/>

File name: xpacsdk\_ce\_n.n.n\_vsxxxx.msi (n.n.n : 平台開發 sdk 版本號碼，xxxx: 2005 代表 VS2005, 2008 代表 VS2008)

安裝步驟為:

步驟 1: 執行“XPacSDK\_CE.msi

步驟 2: 依著設定畫面顯示按下 Next 鍵，直到 xpacsdk\_ce\_n.n.n\_vsxxxx.msi 安裝程序完成。



Classification	XPac SDK FAQ				No.	6-010-00	
Author	Sean	Version	1.0.1	Date	2012/10/18	Page	2/13

我們提供 2 種方法將 XPacSDK 函式庫更新為 PACSDK 函式庫。  
 第一種方式是提供詳細的更新步驟給使用者手動更新，而另一種是提供更新工具自動更新為 PACSDK 函式庫。

## 手動更新 SDK

### VC 程式

依以下步驟更新 PACSDK 函式庫至你的程式

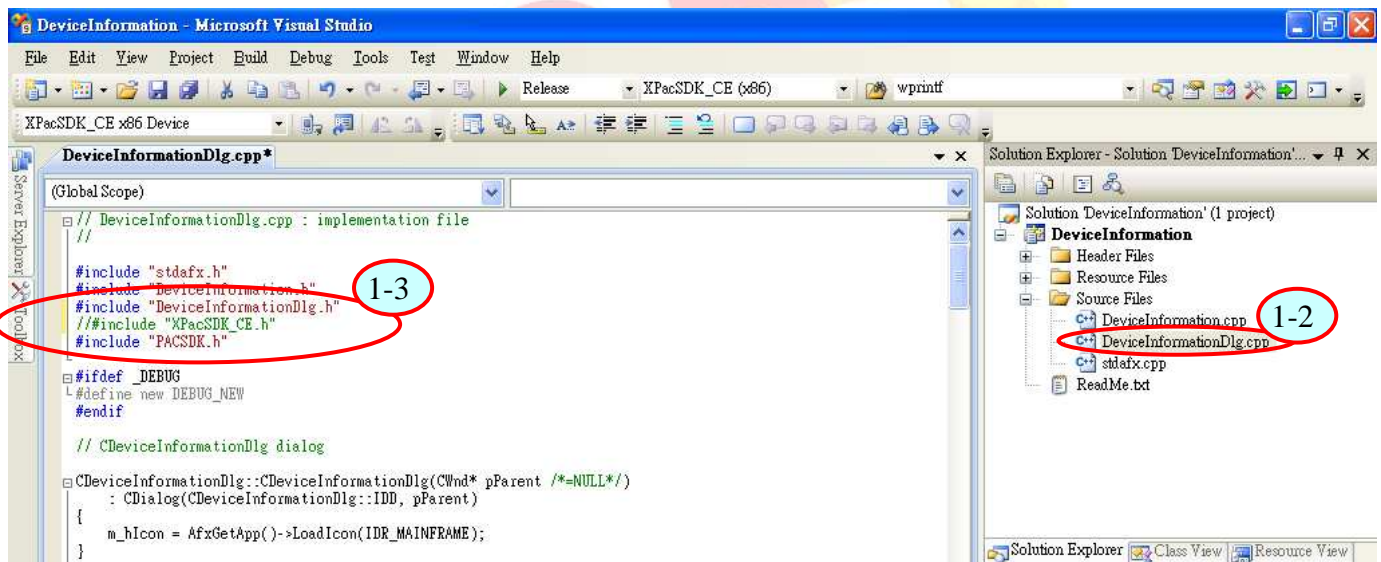
**步驟 1:** 更改 XPacSDK\_CE.h 為 PACSDK.h

- 1-1 開啟你的 XPac 系列模組使用的 **Visual Studio 2005 or 2008** 專案 (\*.sln)
- 1-2 於 Solution explorer 上點選要修改的\*.cpp 檔將此檔開啟
- 1-3 於 Editor 視窗將程式修改成以下的內容 (參考下圖)

```

// #include "XPacSDK_CE.h"
#include "PACSDK.h"

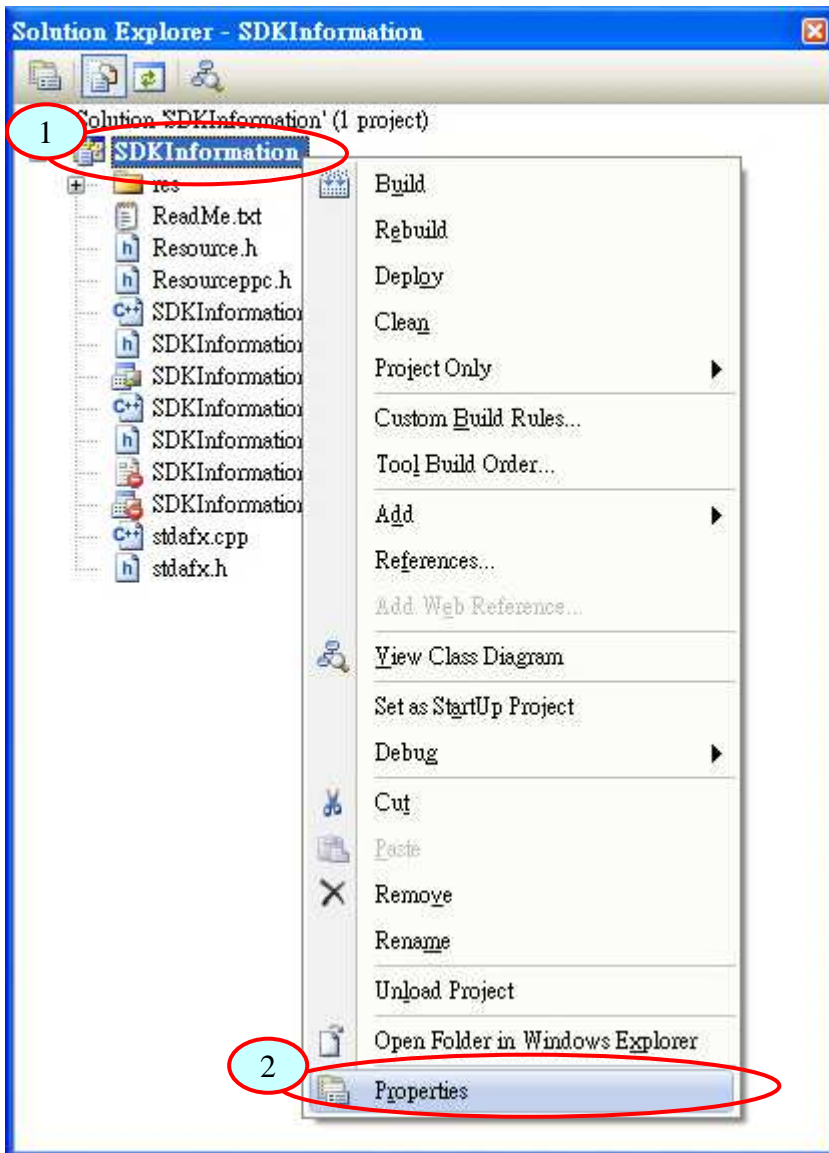
```



Classification	XPAC SDK FAQ				No.	6-010-00	
Author	Sean	Version	1.0.1	Date	2012/10/18	Page	3/13

步驟 2: 更改 XPacSDK\_CE.lib 為 PACSDK.lib

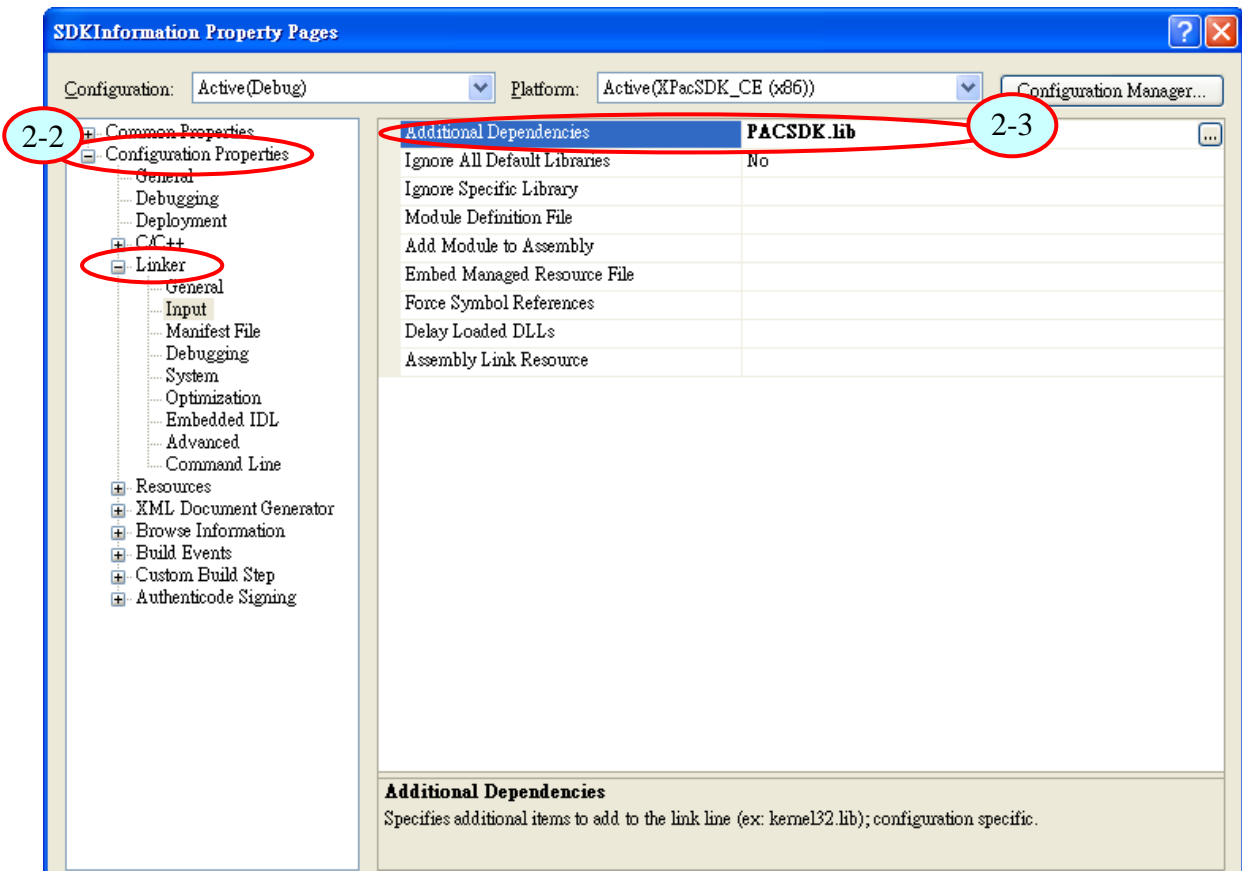
2-1 右鍵點擊 Solution explorer 視窗上的專案名稱，接著點選“Properties”項目(如下圖，此例子專案名稱為 SDKInformation)



2-2 出現的 Property Pages 視窗的左邊窗格(pane)上，展開 Configuration Properties 及 Linker 項目，然後點選 Input 選項。

2-3 視窗的右邊窗格上，於 Additional Dependencies 文字框上輸入 PACSDK.lib (如下圖)。

Classification	XPAC SDK FAQ					No.	6-010-00
Author	Sean	Version	1.0.1	Date	2012/10/18	Page	4/13



**步驟 3:** 參考以下說明並確認你程式碼是否有以下的項目，如果有請修改你的程式碼

**1. 錯誤碼修改**

```
//if(pac_GetLastError() == PAC_ERR_INTR_CRATE_EVENT_FAILURE)
if(pac_GetLastError() == PAC_ERR_INTR_CREATE_EVENT_FAILURE)
{
    //...
}
```

PAC\_ERR\_INTR\_CRATE\_EVENT\_FAILURE 錯誤碼是定義在 XPacSDK\_CE.h，使用 PACSDK.h，需將它們修改為 PAC\_ERR\_INTR\_CREATE\_EVENT\_FAILURE。

**2. API 函式修改**

```
//pac_EnableLED(pin, bFlag);
pac_EnableLEDs(pin, bFlag);
```

XPacSDK\_CE.lib 內的 pac\_EnableLED(pin, bFlag)函式在 PACSDK.lib 被更名為 pac\_EnableLEDs(pin, bFlag)。

**3. 增加記憶體系統保留區段**

為了保留 EEPROM 及 SRAM 部份區段記憶體給系統使用，PACSDK 函式庫的 pac\_ReadMemory

Classification	XPAC SDK FAQ					No.	6-010-00
Author	Sean	Version	1.0.1	Date	2012/10/18	Page	5/13

及 pac\_WriteMemory 函式有更改過。

保留區段的定義如下：

EEPROM

0 ~0x1FFF (8KB) 使用者使用

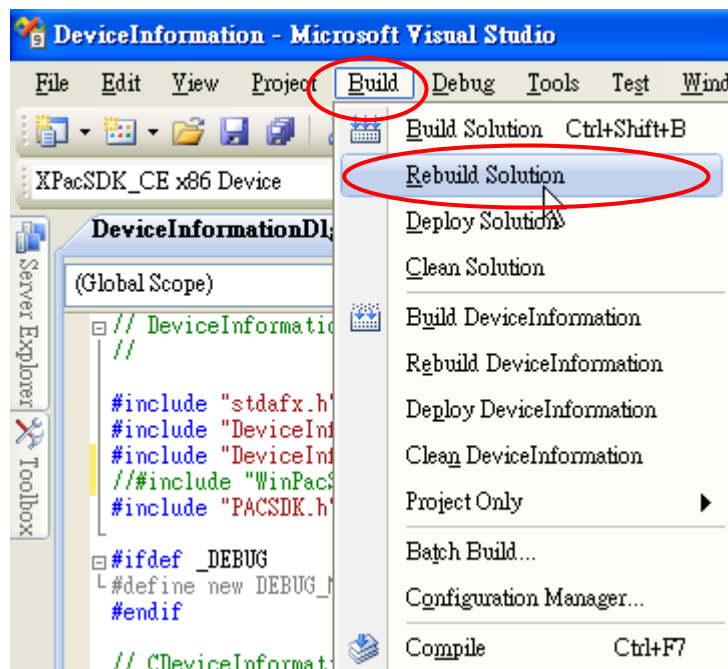
0x2000~0x3FFF (8KB) 保留給系統使用

SRAM

使用者可用的範圍為0 ~0x6FFFF (448KB)，最後一個64KB空間是保留給系統使用

請確認原來的程式在 pac\_ReadMemory 及 pac\_WriteMemory 內有使用到保留的空間，如果有請將位址移至使用者使用空間。

**步驟 4:** 在 Visual studio IDE 視窗上，點選 Build 選單的“Rebuild Solution”項目，重新編譯你的專案。



Classification	XPAC SDK FAQ				No.	6-010-00	
Author	Sean	Version	1.0.1	Date	2012/10/18	Page	6/13

## .NET Compact Framework 程式(C#, VB.net)

依以下步驟更新 PACNET 函式庫至你的程式.

### 步驟 1: 更改 XPacNet.dll 為 PACSDK.dll

1-1 取得 PACNET.dll 檔案及將它置於.NET CF 專案目錄內。

PACNET.dll 可以由以下連結路徑得到

出貨 CD 內:

CD:\XPAC-ATOM-CE6\SDK\XPacNET

CD:\XP-8000-CE6\SDK\XPacNET

ICP DAS 網站:

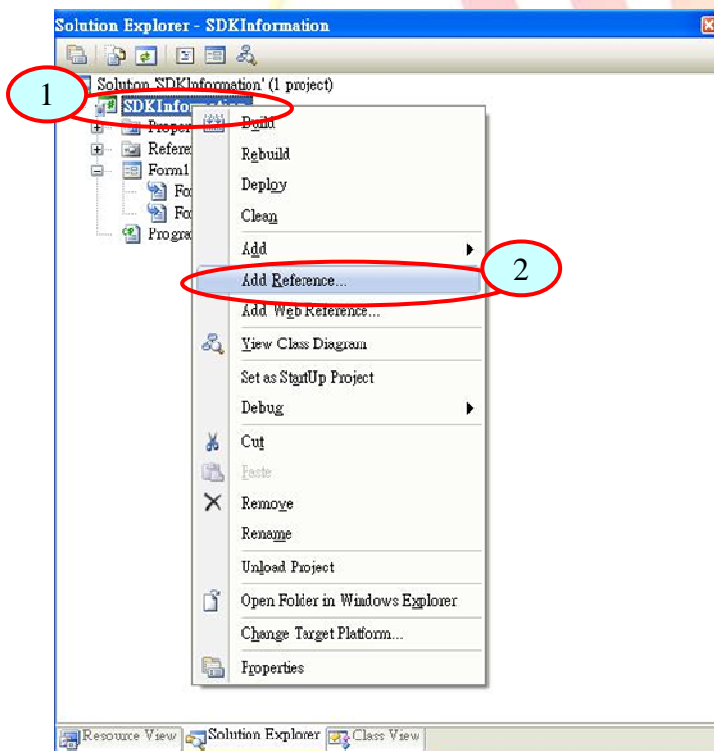
<http://ftp.icpdas.com/pub/cd/xpac-atom-ce6/sdk/xpacnet/>

<http://ftp.icpdas.com/pub/cd/xp-8000-ce6/sdk/xpacnet/>



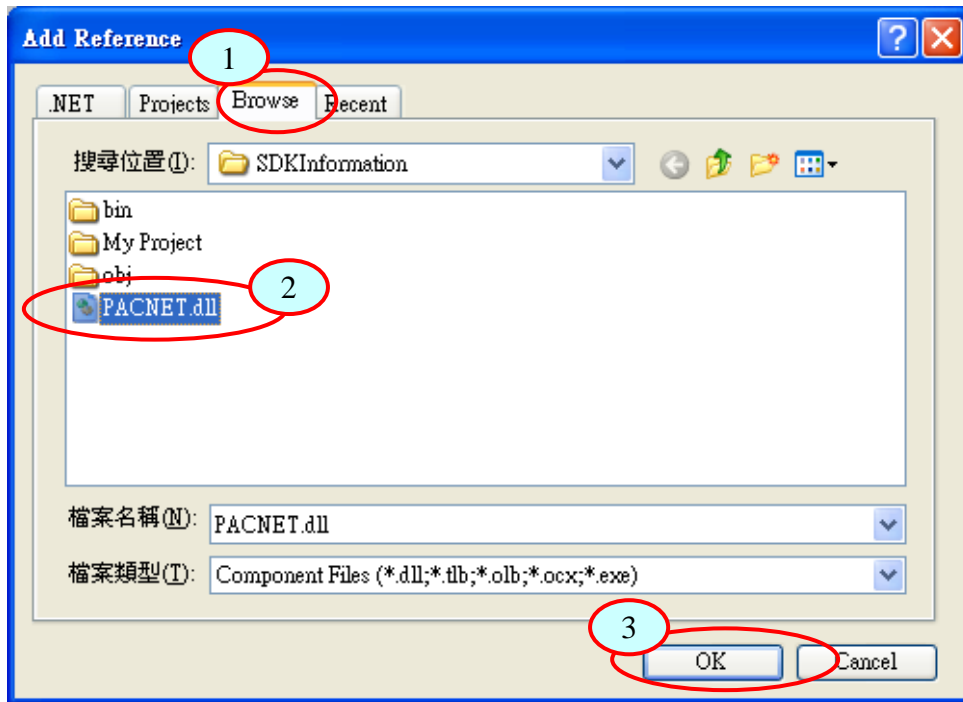
1-2 開啟你的 XPAC 系列模組使用的 **Visual Studio 2005** or **2008 C#/VB.net** 專案 (\*.sln)。

1-3 右鍵點擊 Solution explorer 視窗上的專案名稱，接著點選“Add Reference...” 項目 (如下圖，此例子專案名稱為 SDKInformation)



Classification	XPAC SDK FAQ				No.	6-010-00	
Author	Sean	Version	1.0.1	Date	2012/10/18	Page	7/13

1-4 於“Add Reference”視窗上，選“Browse”標籤及點選 PACNET.dll 加入此專案。



步驟 2: 參考以下說明並確認你程式碼是否有以下的項目，如果有請修改你的程式碼。

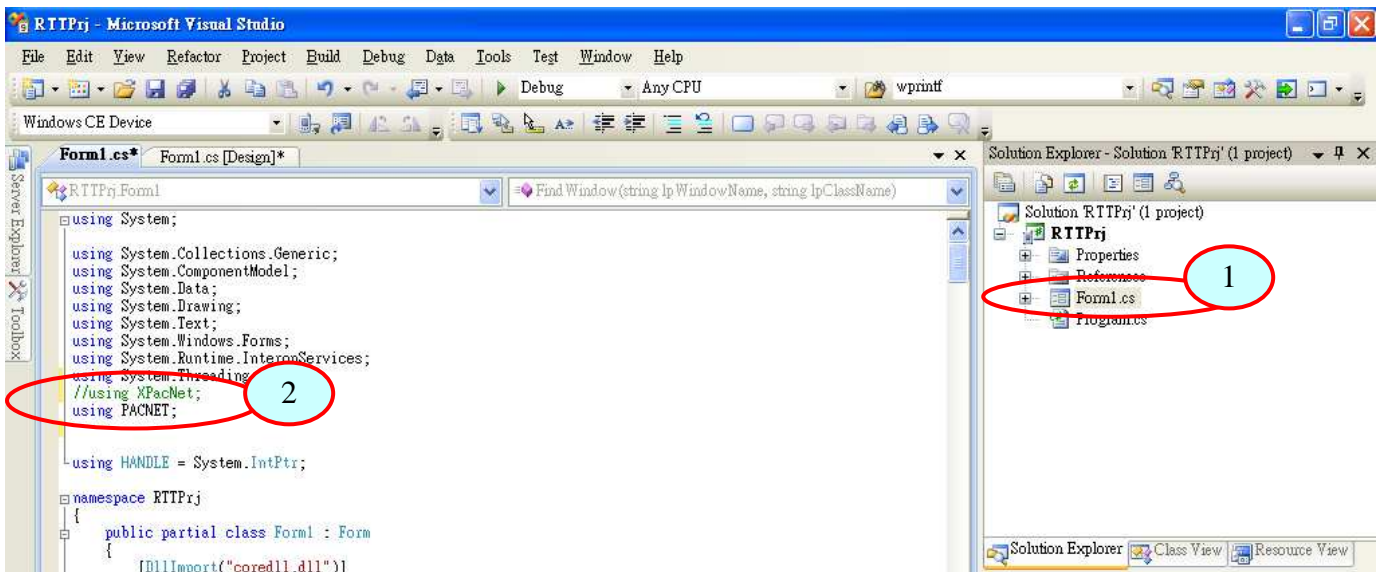
1. 修改程式碼。

於 C# 程式將“using XPacNET”改為“using PACNET”

```
//using XPacNet;
using PACNET;
```

於 VB 程式將“Imports XPacNET”改為“Imports PACNET”。

```
//Imports XPacNet
Imports PACNET
```



Classification	XPAC SDK FAQ					No.	6-010-00
Author	Sean	Version	1.0.1	Date	2012/10/18	Page	8/13

## 2. API 函式修改

```
//XPacNet.XPac.pac_EnableLED(pin, bFlag);
PACNET.SysInfo.pac_EnableLEDs(pin, bFlag);
```

XPacNet.dll 內的 XPacNet.XPac.pac\_EnableLED(pin, bFlag) 函式在 PACNET.dll 被更名為 PACNET.SysInfo.pac\_EnableLEDs(pin, bFlag)。

## 3. DllImport 修改

於 C# or VB program 使用 XPAC SDK 的 API 函式，需使用 DllImport 命令。

於 C# 程式

```
//[DllImport("XPacSDK_CE.dll")]
[DllImport("PACSDK.dll")]
```

於 C# 程中，使用 `DllImport("PACSDK.dll")` 取代 `DllImport("XPacSDK_CE.dll")`

於 VB 程式

```
<<DllImport("XPacSDK_CE.dll")>
<DllImport("PACSDK.dll")>
```

於 VB.net 程中，使用 `DllImport<"PACSDK.dll">` 取代 `DllImport<"XPacSDK_CE.dll">`

## 4. 類別(Class)名稱修改

在 XPacNet.dll 的所有 API 函式，都只被分類在一個類別(Class)內(XPacNet.XPac.xxx class)，在新的 PACNET.dll 中，API 函式被分類在不同的類別(Class)內，如 PACNET.Sys，PACNET.Memory 和 PACNET.Interrupt 等。

在 PACNET.dll 所有 API 函式 Class 分類，在 API 手冊有詳細定義說明，而主要分類對照表如下

API 手冊的分類名稱	PACNET.dll Class 分類名稱
2.1 System Information API	Sys
2.1 Backplane API	Sys
2.1 Buzzer API	Sys.Buzzer
2.2 Interrupt API	Interrupt
2.3 Memory Access API	Memory
2.4 Watchdog API	Sys.WDT
2.5 Registry API	PAC_Reg
2.6 UART API	UART
2.7 PAC_IO API	PAC_IO
2.8. PWM API	PWM
2.9. Backplane Timer API	BPTimer
2.10. Error Handling API	ErrHandling
2.11 MISC API	MISC



Classification	XPAC SDK FAQ					No.	6-010-00
Author	Sean	Version	1.0.1	Date	2012/10/18	Page	9/13

### 5. 增加記憶體系統保留區段

為了保留 EEPROM 及 SRAM 部份區段記憶體給系統使用，PACNET 函式庫的 PACNET.Memory.ReadMemory 及 PACNET.Memory.WriteMemory 函式有更改過。保留區段的定義如下：

#### EEPROM

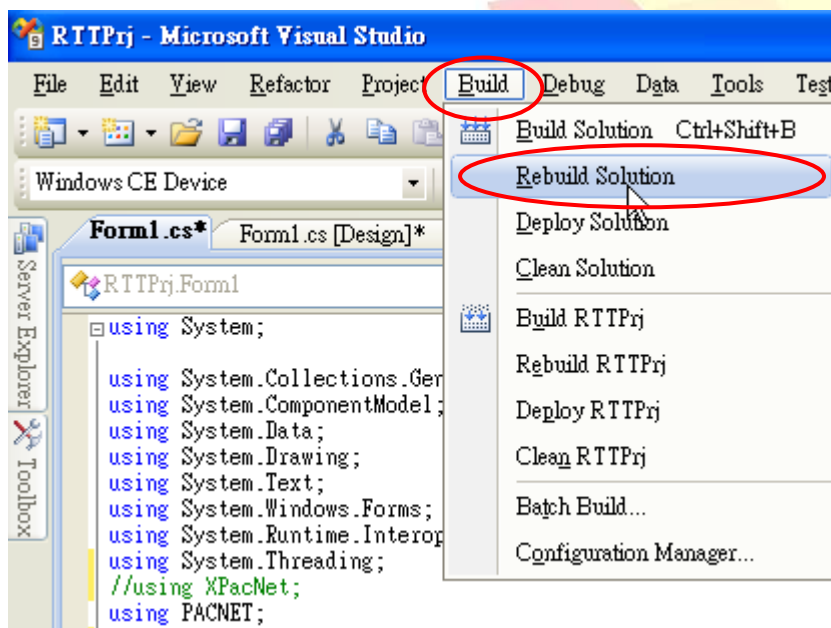
- 0 ~0x1FFF (8KB) 使用者使用
- 0x2000~0x3FFF (8KB) 保留給系統使用

#### SRAM

使用者可用的範圍為0 ~0x6FFFF (448KB)，最後一個64KB空間是保留給系統使用

請確認原來的程式在PACNET.Memory.ReadMemory及PACNET.Memory.WriteMemor內有使用到保留的空間，如果有請將位址移至使用者使用空間。

**步驟 4:**在 Visual studio IDE 視窗上，點選 Build 選單的“Rebuild Solution”項目，重新編譯你的專案。



Classification	XPAC SDK FAQ				No.	6-010-00	
Author	Sean	Version	1.0.1	Date	2012/10/18	Page	10/13

## 使用更新工具來更新 SDK

### EasyUpgradeSDK.exe

EasyUpgradeSDK 是一支程式轉換工具，最主要的功能是将連結 XPacSDK 函式庫及 WinPacSDK 函式庫的 VC 程式轉換使用 PACSDK 函式庫。以及將連結 XPacNet 函式庫及 WinPacNet 函式庫的 C#.VB.net 程式轉換使用 PACNET 函式庫  
此應用程式必須運作於一般 Windows PC 上。

**EasyUpgradeSDK.exe 最新版本是置於 ICPDAS 網站:**

[http://ftp.icpdas.com/pub/cd/xp-8000-ce6/pc\\_tools/EasyUpgradeSDK](http://ftp.icpdas.com/pub/cd/xp-8000-ce6/pc_tools/EasyUpgradeSDK)

[http://ftp.icpdas.com/pub/cd/xpac-atom-ce6/pc\\_tools/EasyUpgradeSDK](http://ftp.icpdas.com/pub/cd/xpac-atom-ce6/pc_tools/EasyUpgradeSDK)

EasyUpgradeSDK 只允許轉換檔案的副檔名為:

- \*.cpp
- \*.cs
- \*.vb
- \*.vcproj
- \*.csproj
- \*.vbproj
- \*.vcp

這個工具的運作規則為:在使用者選取轉換的檔案後，程式會以關鍵字搜尋整個檔案內容，搜尋完成後並以新的關鍵字取代舊的。而且程式會自動尋找使用者選的目錄內是否有 Winpacsdk.h/Xpacsdk.h, Winpacsdk.lib/Xpacsdk.lib 或 WinPacNet.dll/XPacNet.dll 檔案存在。如果有會將 PACSDK.h, PACSDK.lib 或 PACNET.dll 拷貝至此目錄。(原來的.h/.lib/.dll 會保留不會刪除)

依以下步驟使用 EasyUpgradeSDK.exe 去更新 SDK

**步驟 1:** 確認檔案目錄架構是否如以下所示，將這個目錄拷貝至 Windows PC 上

```

EasyUpgradeSDK
├── EasyUpgradeSDK.exe
├── PACNET.dll
├── release_arm
│   ├── PACSDK.H
│   └── PACSDK.lib
├── release_x86
│   ├── PACSDK.H
│   └── PACSDK.lib

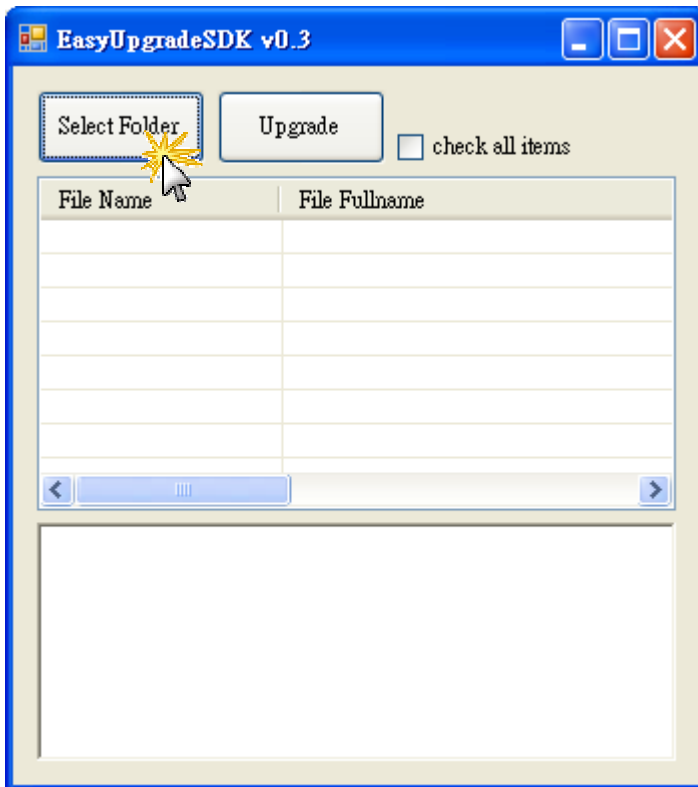
```

**步驟 2:** 執行 EasyUpgradeSDK.exe

**步驟 3:** 選擇一個目錄，專案的所有相關檔案都需在此目錄內。(這個專案必需是 WinPAC/XPAC 系列模組使用，且這個專案必須為 C#, VB.net, VC or eVC 所建立)

Classification	XPAC SDK FAQ				No.	6-010-00	
Author	Sean	Version	1.0.1	Date	2012/10/18	Page	11/13

3-1: 點選 “Select Folder” 鍵(如下圖)

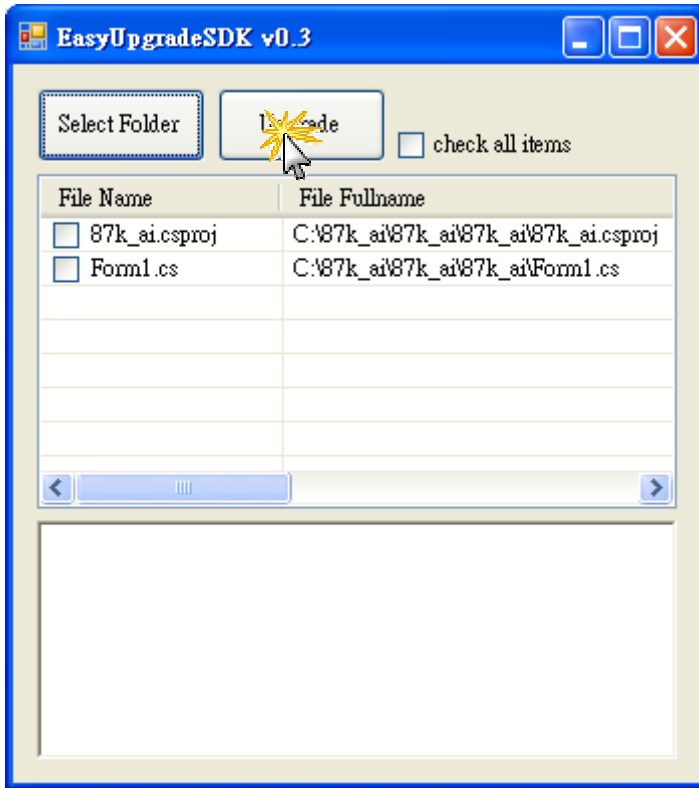


3-2: 點選一個目錄並按下“OK” 鈕。  
(此例子，我們是選 C# 87k\_dio demo)

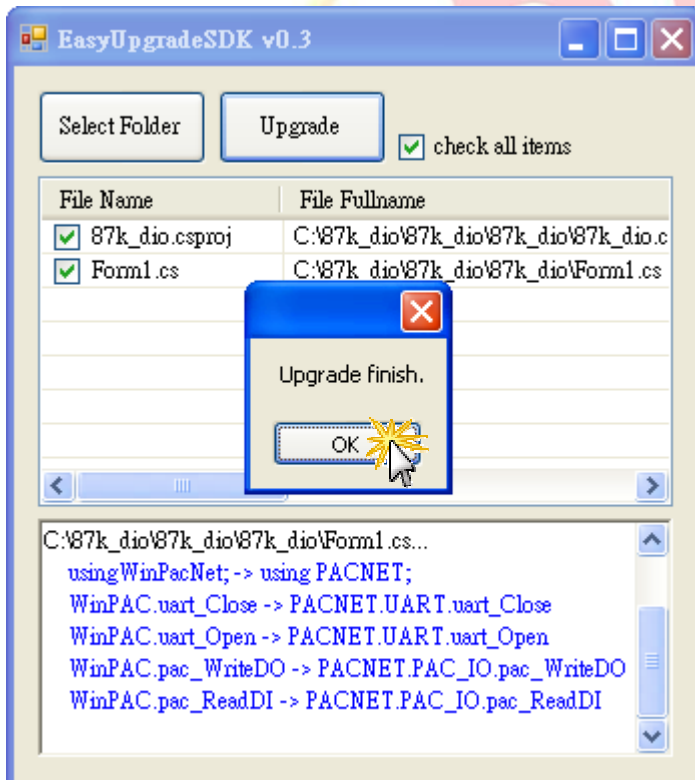


Classification	XPAC SDK FAQ				No.	6-010-00	
Author	Sean	Version	1.0.1	Date	2012/10/18	Page	12/13

3-3: 點選“check all items” checkbox，接著按下“Upgrade” 鈕。



3-4: 等待，直到“Upgrade finish” messagebox 顯示在畫面(如下圖)



Classification	XPAC SDK FAQ				No.	6-010-00	
Author	Sean	Version	1.0.1	Date	2012/10/18	Page	13/13

**步驟 4:** 完成轉換工作後，此轉換的程式碼仍需由開發程式重新編輯後才能使用。(參考手動更新 SDK 來重新編輯程式)

