



MICROCHIP

Regional Training Centers

Section 4

Getting Started First Project

Lab0 – First Project



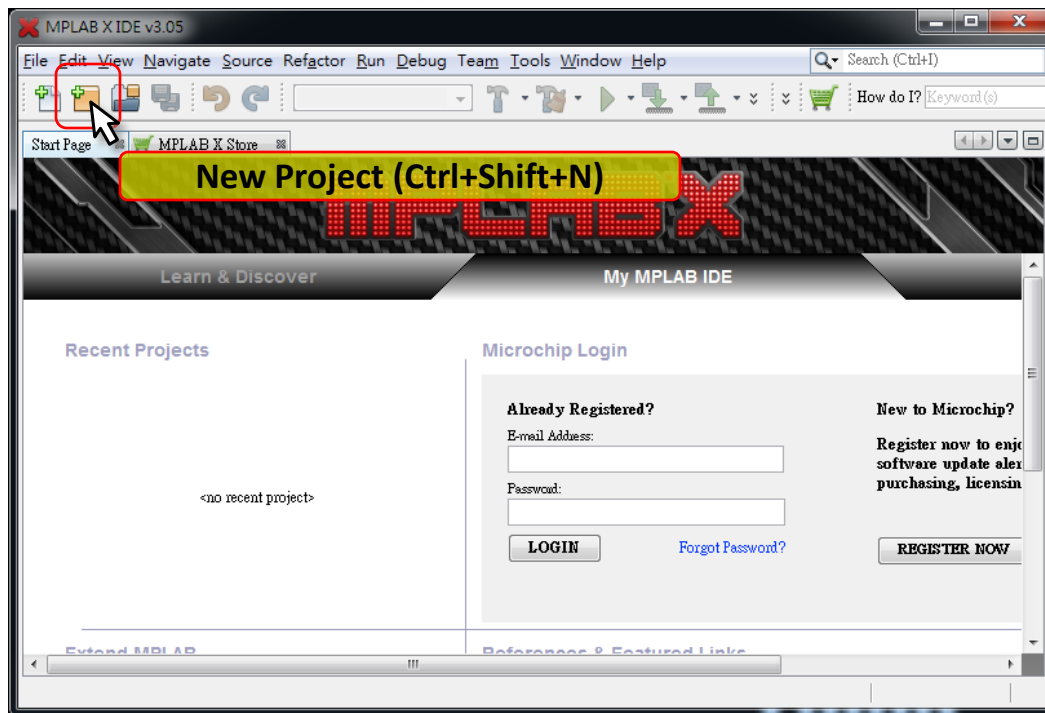
目標

- 嘗試建立自己的第一個MPLAB X IDE專案, 並以MPLAB XC16做為預設的編譯器。
 - 在專案中建立自己的第一個程式(Lab0 FirstProject.c)。
 - 透過建立專案的過程了解MPLAB X IDE的基本操作, 語言工具的設定, 以及各項基本功能。
-
- 該如何開始？

Lab0 - Create Project Step

步驟 A

- 開啟MPLAB X IDE, 啟動專案精靈

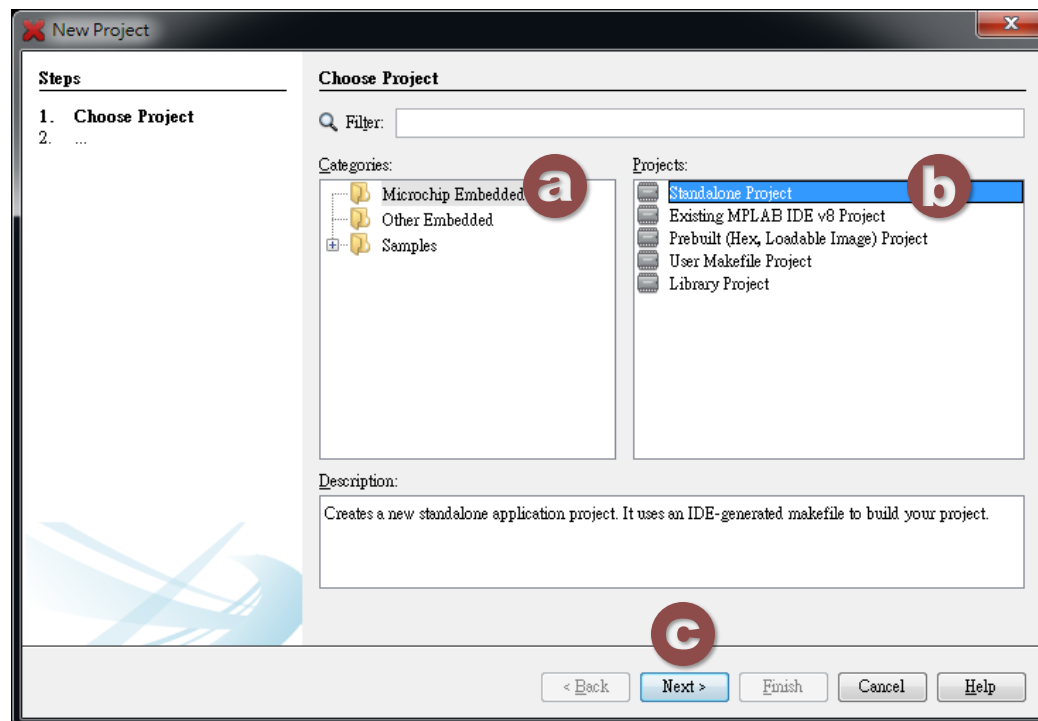


Lab0 - Create Project Step

步驟 B

- 選取專案型態 (Microchip Embedded ▶ Standalone Project)

- a** Categories 選擇
 - ▶ Microchip Embedded
- b** Project 選擇
 - ▶ Standalone Project
- c** 選擇
 - ▶ 

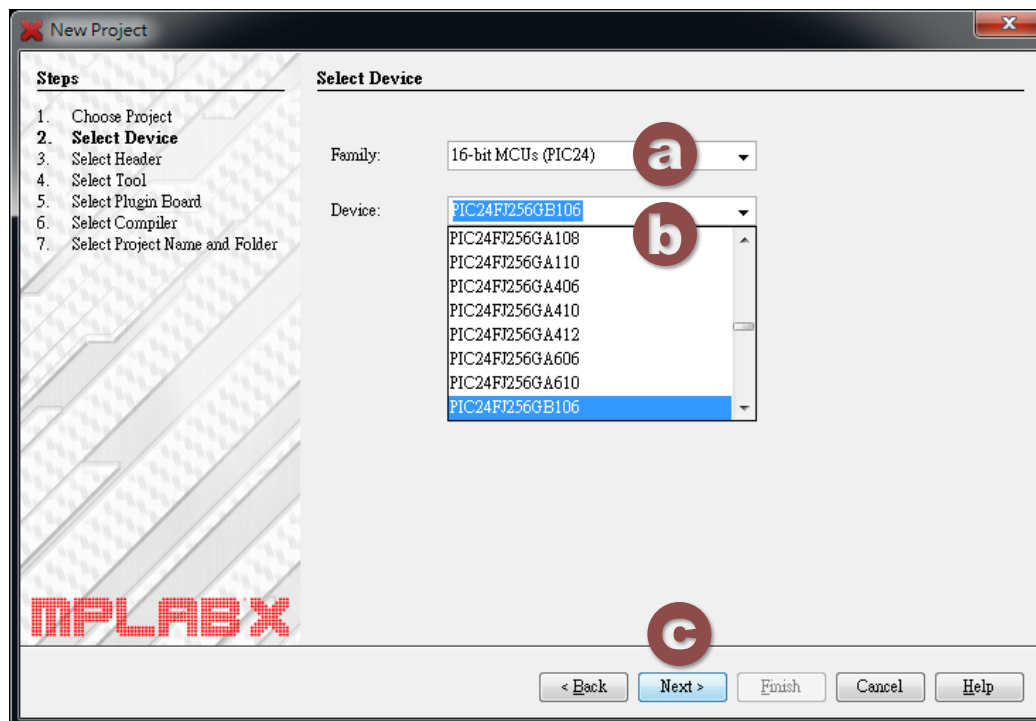


Lab0 - Create Project Step

步驟 C

- 選取目標元件 (16-bit MCUs (PIC24) ▶ PIC24FJ256GB106)

- a** Family 選擇
▶ 16-bit MCUs (PIC24)
- b** Device 選擇
▶ PIC24FJ256GB106
- c** 選擇
▶ [Next >](#)




Lab0 - Create Project Step

步驟 D

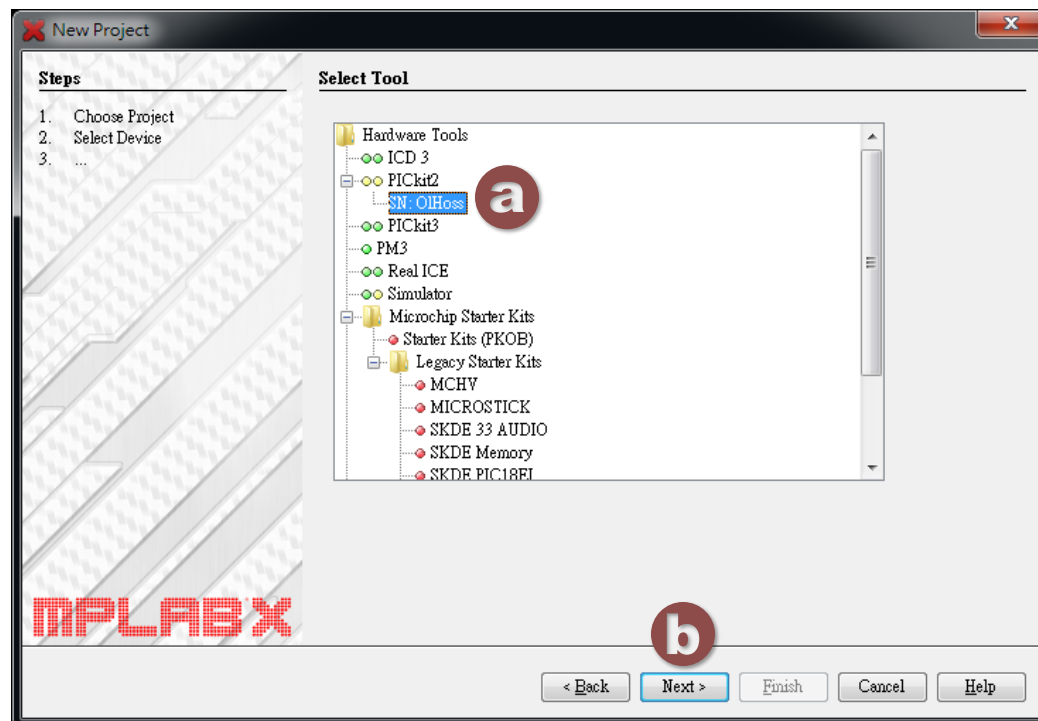
- 選取除錯/燒錄工具 (PICKit2 ▶ SN:OIHoss)

a Select Tools 選擇
▶ PICKit2 ▶ SN:OIHoss

 硬體工具必須先正確
連接上電腦。才能選
取。
PICKit2 並無序號, 選
取SN:OIHoss即可。

b 選擇

 Next >



Lab0 - Create Project Step

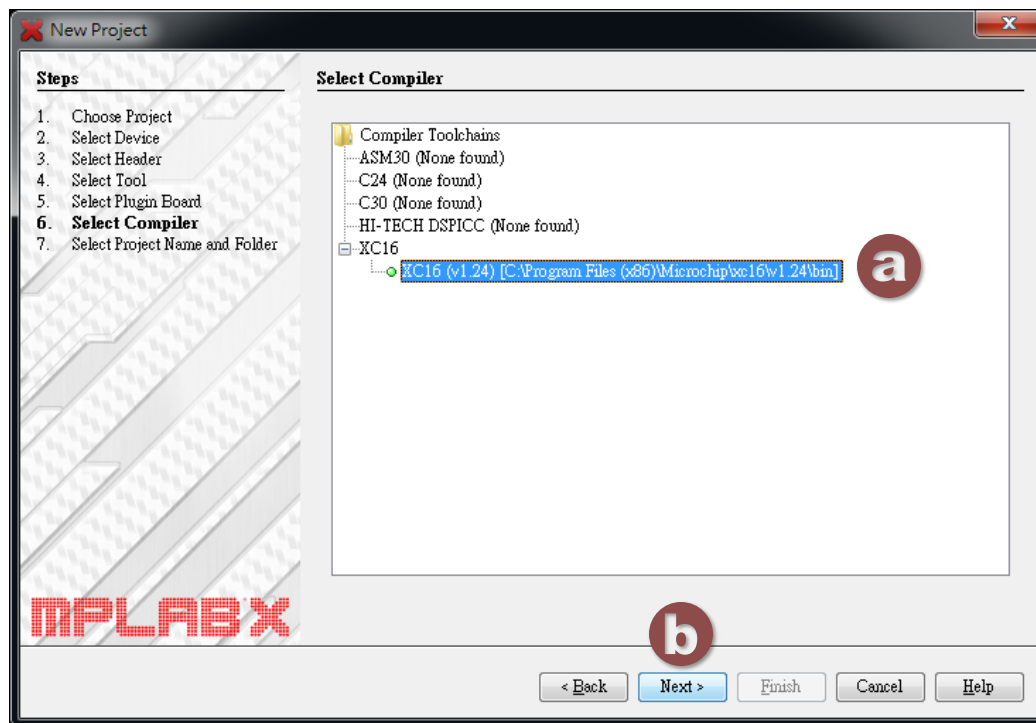
步驟 E

- 選取編譯工具 (XC16 (v1.24))

a Select Compiler 選擇
► **XC16 (v1.24)**

i 如果看不到編譯器，
則可能是編譯器未正
確安裝。

b 選擇
► **Next >**



Lab0 - Create Project Step

步驟 F


- 指定專案名稱與儲存路徑



a Project Name 輸入

► Lab0 First Project

b Project Location 輸入

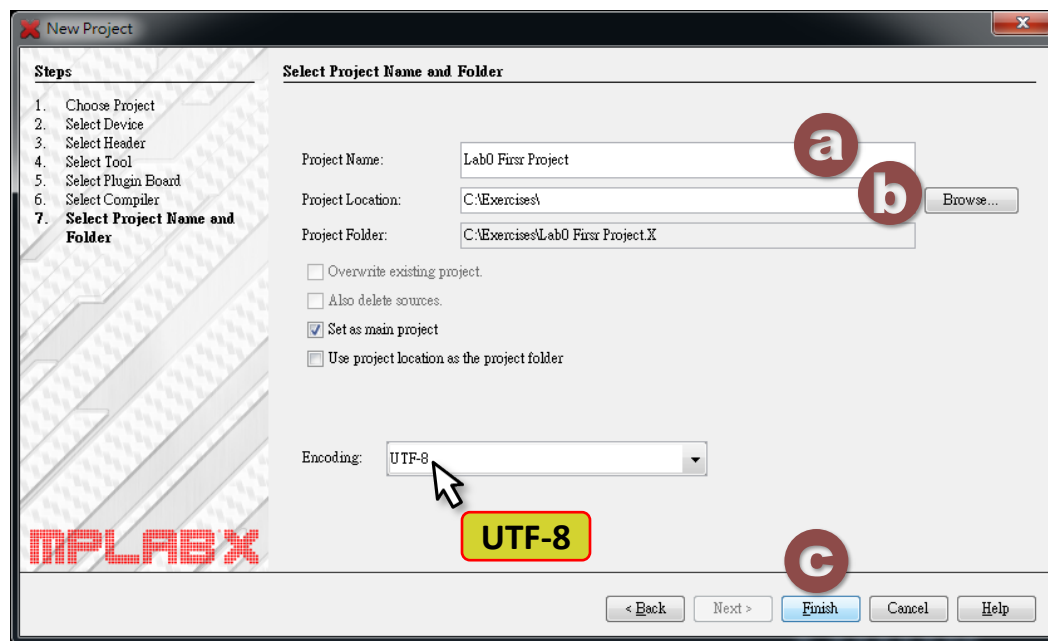
► C:\Exercises\Exams\

 精靈會在專案資料夾下建立一個以專案名命名的資料夾。

 Exercises
 Exams **Project Location**
 Lab0 First Project.x **Project.X**

c 選擇

► **Finish**

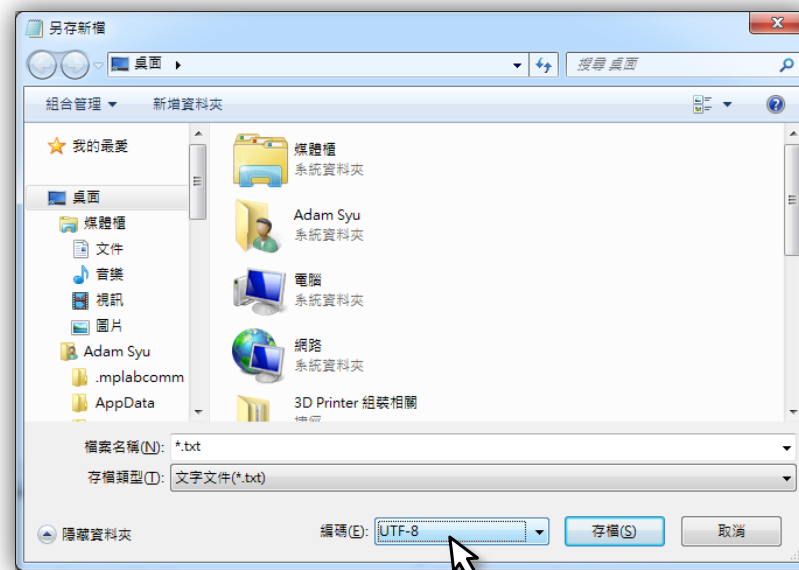


About Encoding

- Encode決定程式檔案的存檔編碼, 如果程式中會使用到非英語系的文字與符號來註解時, 建議修改編碼為UTF-8(Unicode)。
- MPLAB IDE v8.xx 的中文使用 Big-5 編碼方式, X IDE使用UTF-8。如果直接匯入v8.xx的程式檔案會造成中文亂碼。

解決方法:

將原先在 MPLAB v8.xx 的
程式檔案先以”記事本”開啟後,
再複製, 貼入MPLAB X中。
或使用 Save As選擇UTF-8 編碼
後存檔。



UTF-8


Lab0 - Create Project Step

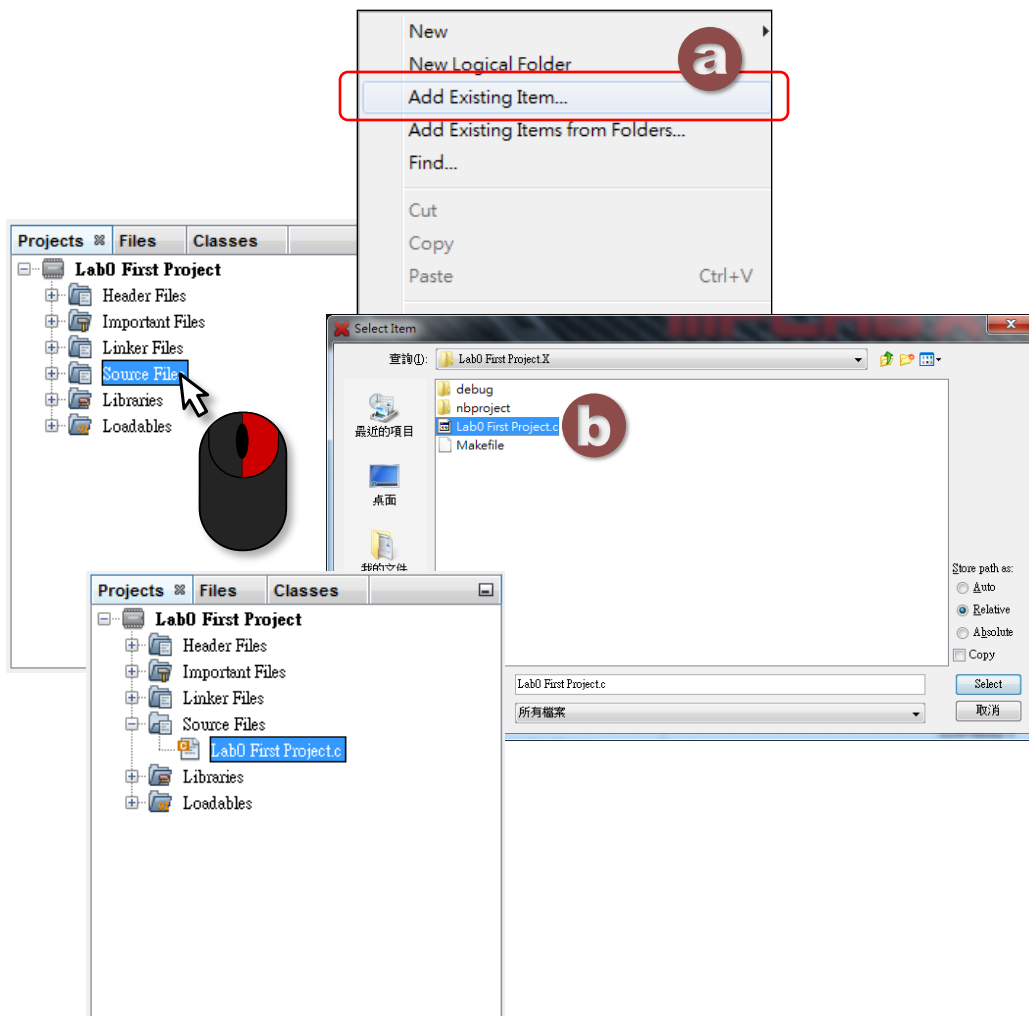
步驟 G

• 加入現有程式檔案

a Projects 視窗, 使用滑鼠右鍵, 點選**Source Files**。並從彈出視窗點選
► **Add Existing Item**

b 加入**Lab0 First Project.c**到專案的**Source Files**中。

 各類型的檔案, 可以在對應的資料夾上使用 Add Existing Item 加入。或用 New 新增空白檔案。



Lab0 - Create Project Step

步驟 H

• 編譯及除錯

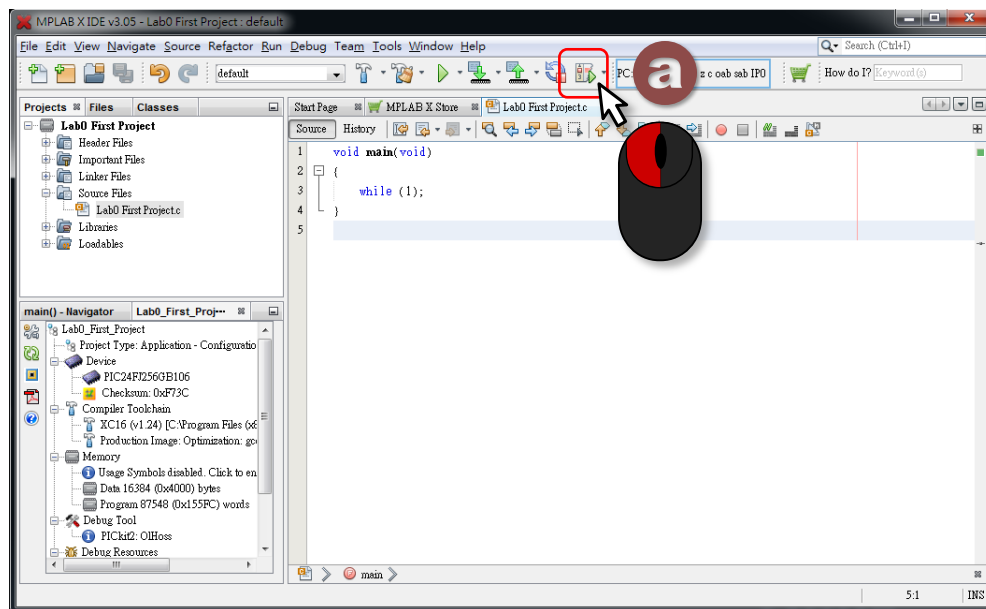
a 功能表點擊

► Debug Main Project



Debug Main Project:

在**除錯模式**下編譯 (Make), 成功後與 PICKit2連線將程式燒錄至MCU上, 完成後立即執行程式。



Build Failed !

- Build Failed表示程式有誤,Output視窗會提示錯誤訊息。可使用滑鼠雙擊錯誤訊息,會跳到程式中有錯誤的地方。接著觀察,冷靜的判斷,找出錯誤。
- 常見的錯誤:
 - **大小寫不一致:**
C語言中大小寫不同就代表不一樣的名稱。
 - **變數未宣告,或重複宣告:**
變數一定要先被宣告才能使用,且僅能宣告一次。若要使用在其他檔案所宣告的變數,則需加上`extern`修飾詞。
 - **敘述(Statement)區塊不完整:**
少了敘述結尾的分號(;)或Statement Block的大括弧({, })。
 - **引數或參數列不對稱:**
呼叫函式時的參數數量或型別不正確。
Ex: `void Func(int , int , int);` 呼叫時要給3個整數型態的參數。

Lab0 - Create Project Step



步驟 I

• 執行流程控制

a 嘗試除錯工具列的各項功能。



b 比較  與  的差異。



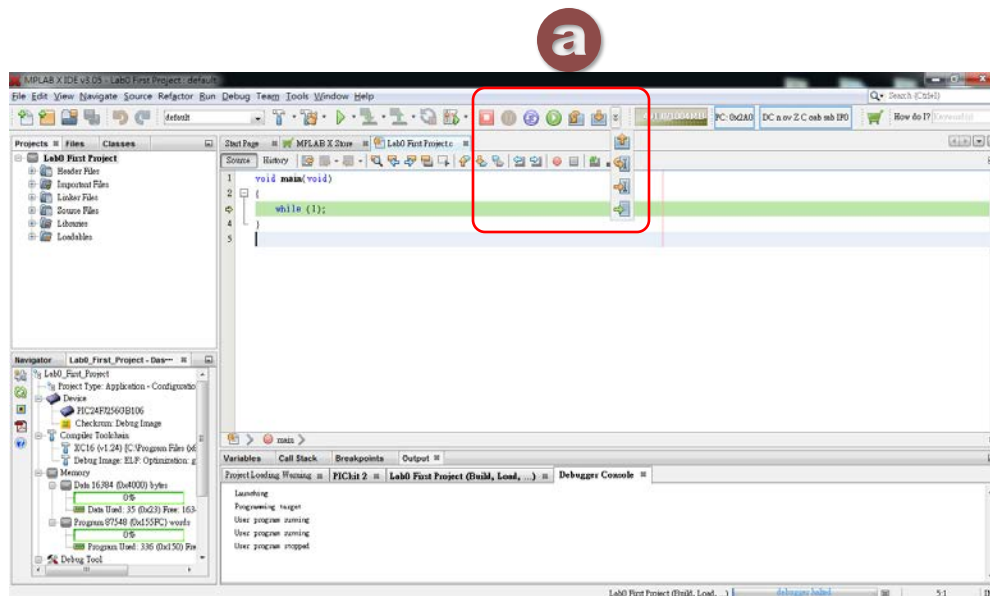
Pause

暫停程式執行。



End Debug Session

停止除錯模式。



Debug Mode Setting



如果您不想除錯模式下, 如果不要燒錄完成後, 自動執行。

Tools ▶ Options

▶ Embedded

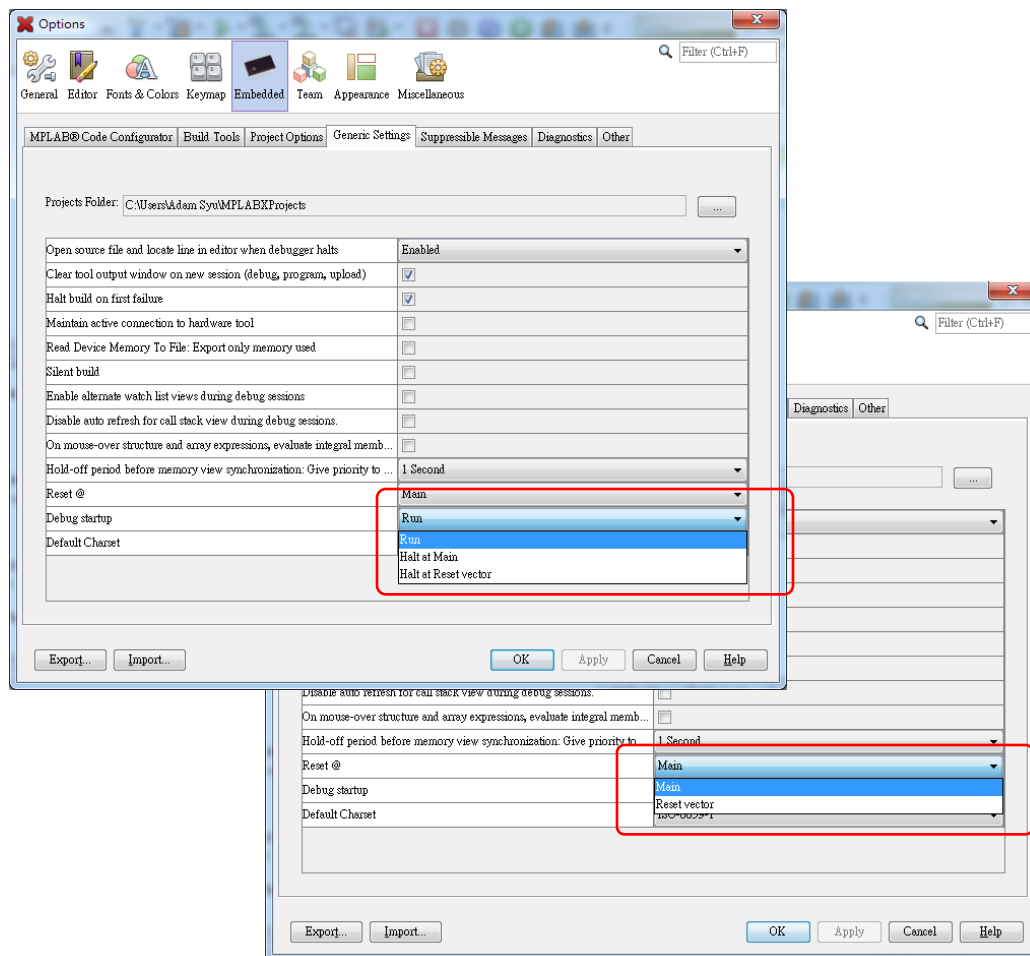
▶ Generic Settings

▶ Debug Startup



▶ Reset @

可以決定Reset後的程式起點。



Lab0 - Create Project Step

- 功能表 ▶ Window ▶ Debugging
功能表 ▶ Window ▶ PIC Memory View
中有許多可以用來觀看MCU內部暫存器, 程式記憶體, 資料記憶體及特殊功能暫存器的視窗。
- 視窗內的值會在於每次程式暫停時, 自動更新。
- 使用時建議不要持續開啟這些視窗。
否則會使得需更新的資料量過大,
導致軟體反應變慢。
- 因此強烈建議僅把要監視的數值,
放入**Watches Window**中觀看,
降低無謂的資料更新。

