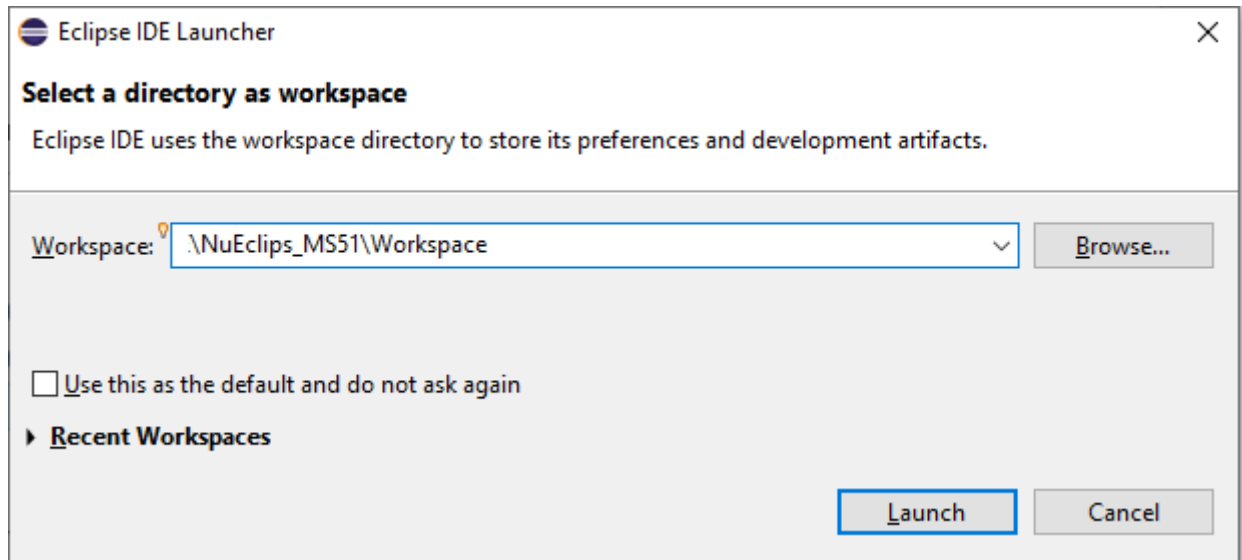


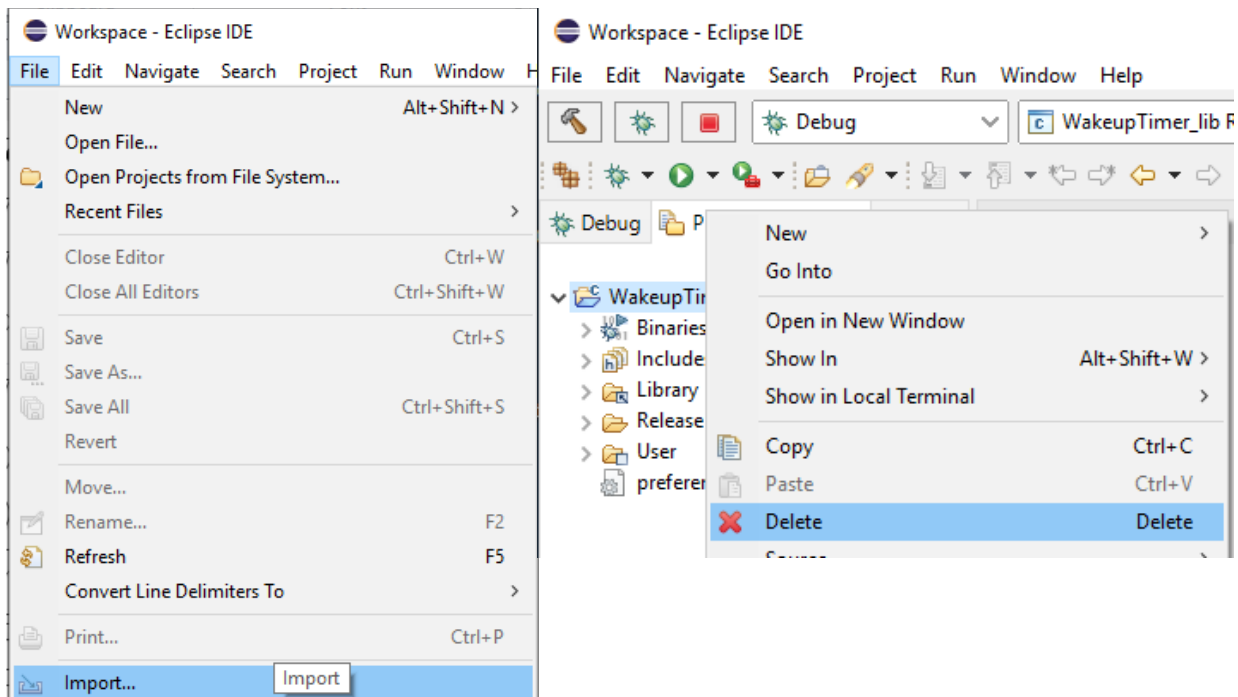
# Nu Eclipse SDCC Quick Start

## 開啟現有 BSP project

1. Eclipse 採用 workspace import project 的方式，  
BSP 有建立一個 workspace folder 可直接開啟。直接在 launcher 的 workspace 內點選對應路徑

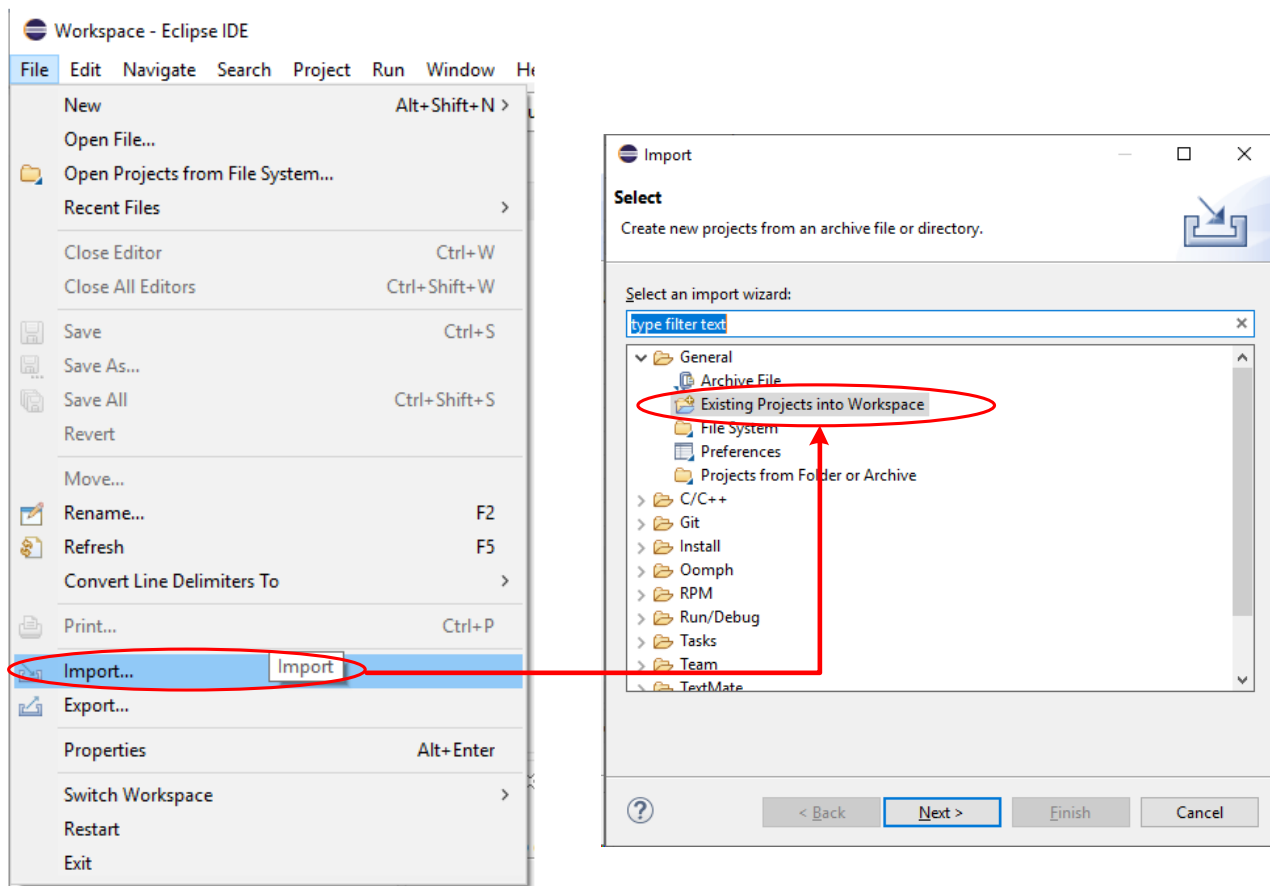


當開啟一個 project 採用 import ，關閉採用 workspace 內 delete project

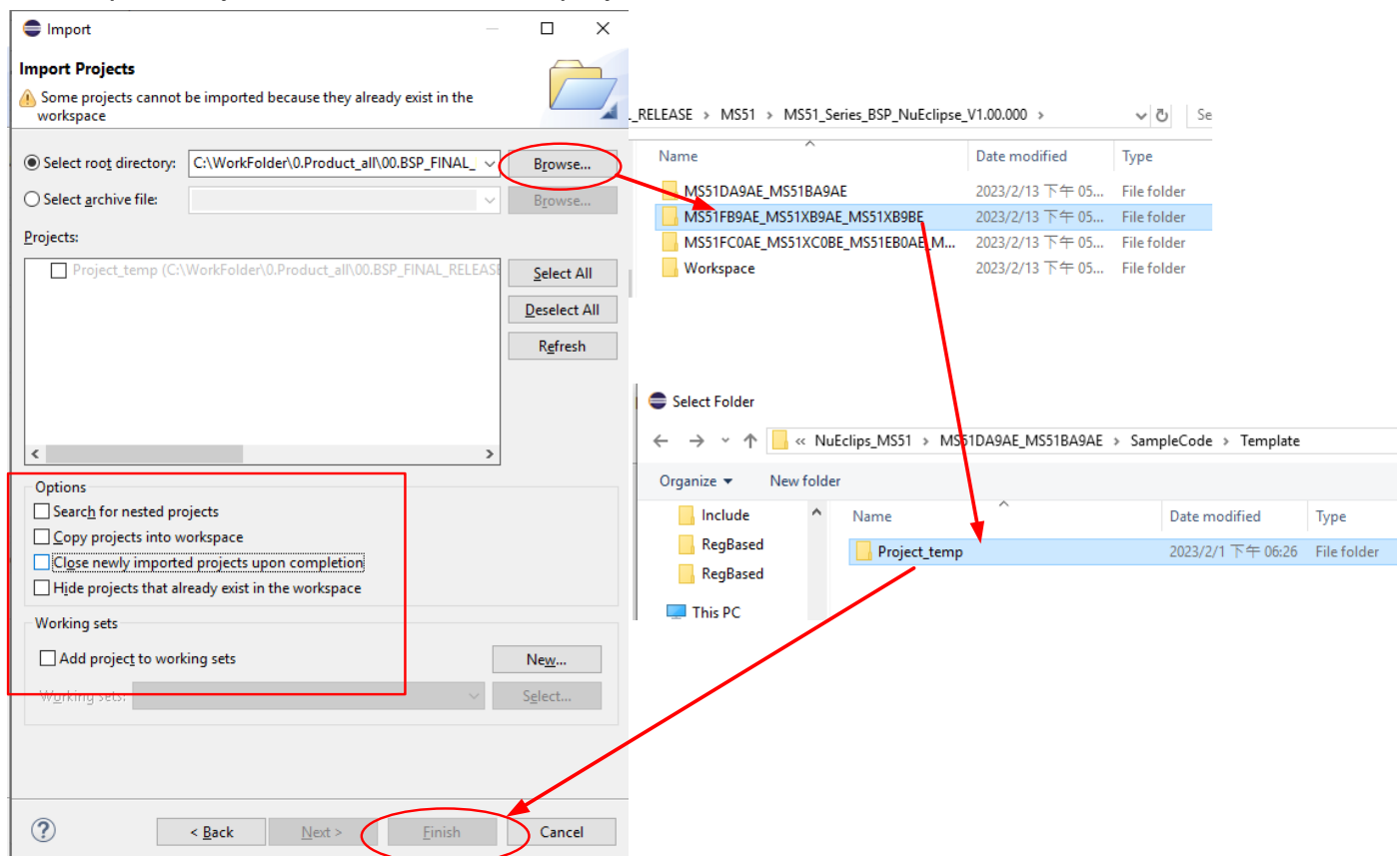


## 2. 開啟 project

點選 file / Import “Existing Projects into Workspace.”



啟在 import Projects 中點選 BSP 對應 project folder，並點選 Finish.



### 3. Compiler / Build

右鍵點選 Project 名稱並選擇"Build Project"，正常完成後，會顯示 0 error，0 warning 以及 checksum 相關資訊。

The screenshot shows the Eclipse IDE interface. The Project Explorer on the left displays a project named 'Project\_temp (in SDCC)' with subfolders: Includes, Library, User, and preferences.ini. A red arrow points from the 'Build Project' option in the context menu to the 'Project\_temp' project. The context menu also shows options like 'New', 'Go Into', 'Open in New Window', 'Show In', 'Show in Local Terminal', 'Copy', 'Paste', 'Delete', 'Source', 'Move...', 'Rename...', 'Import...', 'Export...', and 'Clean Project'.

The main editor shows the source code for 'main.c' with the following content:

```
7
8- /*****
9  /* File Function: MS51 DEMO project
10 /*****
11
12 #include "MS51_8K_SDCC.h"
13
14
15 void main (void)
16 {
17     /* UART0 initial setting
18     * include sys.c in Library for modify HIRC value to 24MHz
19     * include uart.c in Library for UART initial setting
20     */
21     MODIFY_HIRC(HIRC_24);
22     Enable_UART0_VCOM_printf_24M_115200();
23
24     printf("\n Hello world!");
25     while(1);
26
27
28 }
```

The Console window at the bottom shows the build output for 'Project\_temp'.

```
CDT Build Console [Project_temp]
Finished building target: Project_temp.elf

make --no-print-directory post-build
arm-none-eabi-objcopy -R REG_BANK_0 -R REG_BANK_1 -R REG_BANK_2 -R
>>> Project_temp.bin Checksum = 0xF7C3 <<<
>>> Project_temp.bin CRC-8 Checksum = 0xED <<<

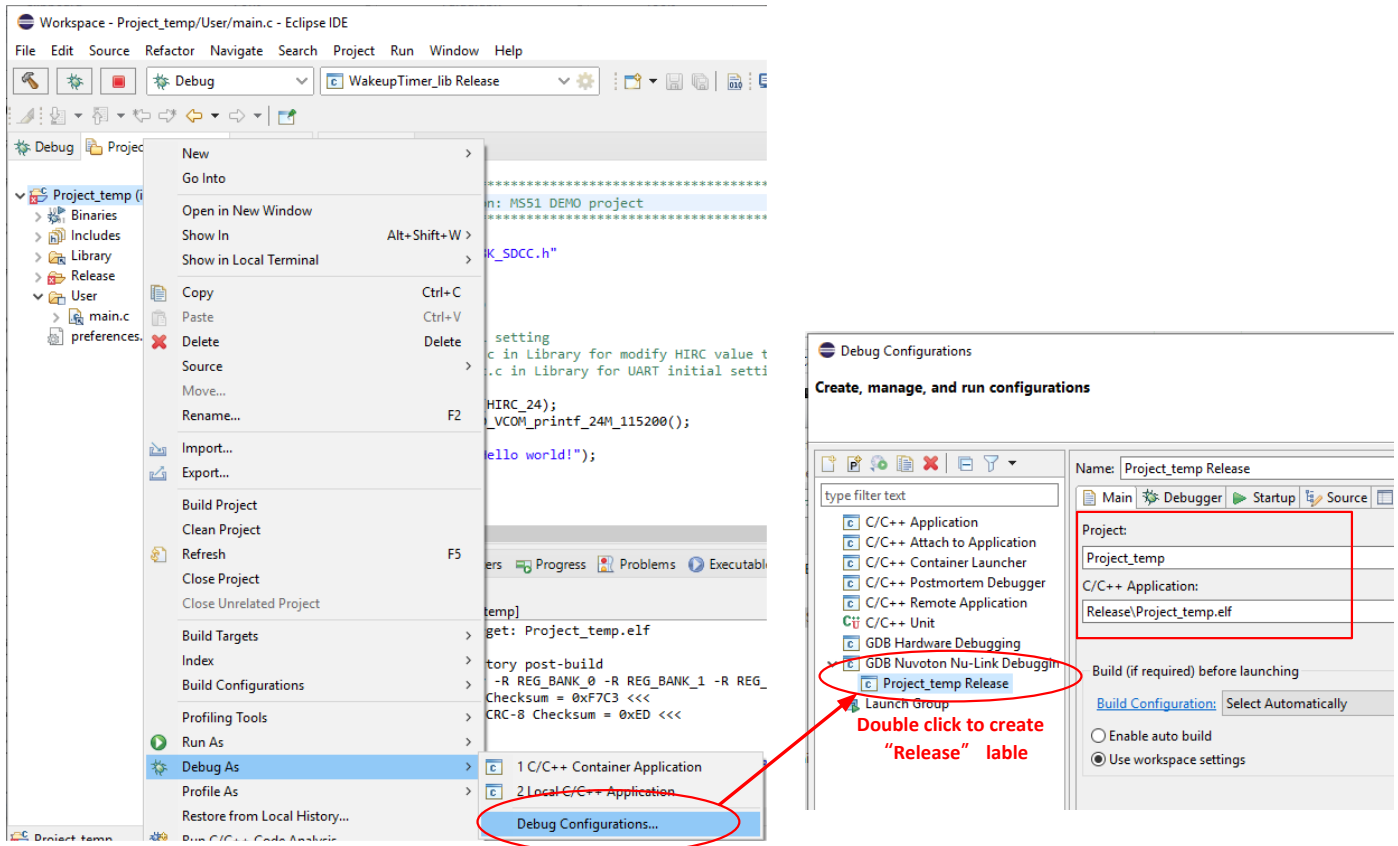
19:02:05 Build Finished. 0 errors, 0 warnings. (took 3s.845ms)
```

The console output is circled in red, highlighting the checksum information.

## 4. Debug

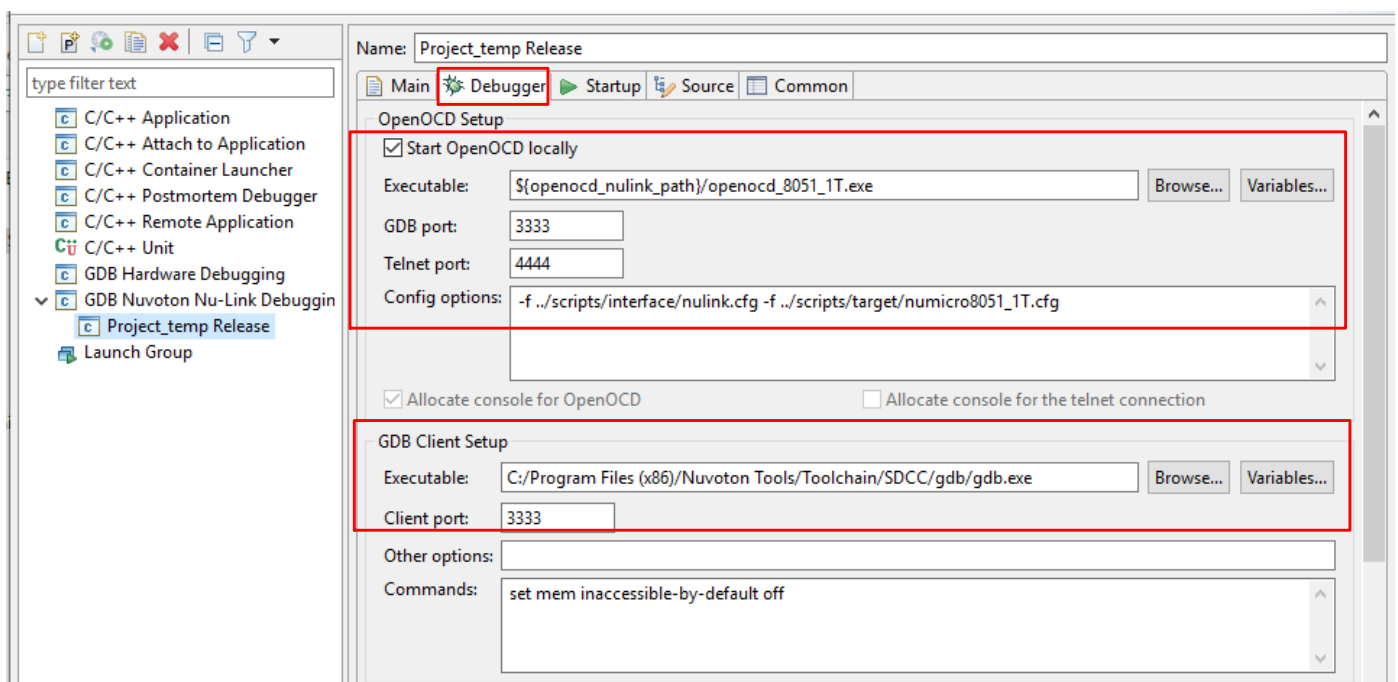
右鍵點選 Project 名稱並選擇"Debug As" Debug Configurations.

在 Debug Configurations 頁面內，雙擊 GDB Nuvoton Nulink Debugging. 系統會自動產生 Project Release 標籤。檢查標籤內 C/C++ Application 是否自動出現 Release project 的.elf 內容。



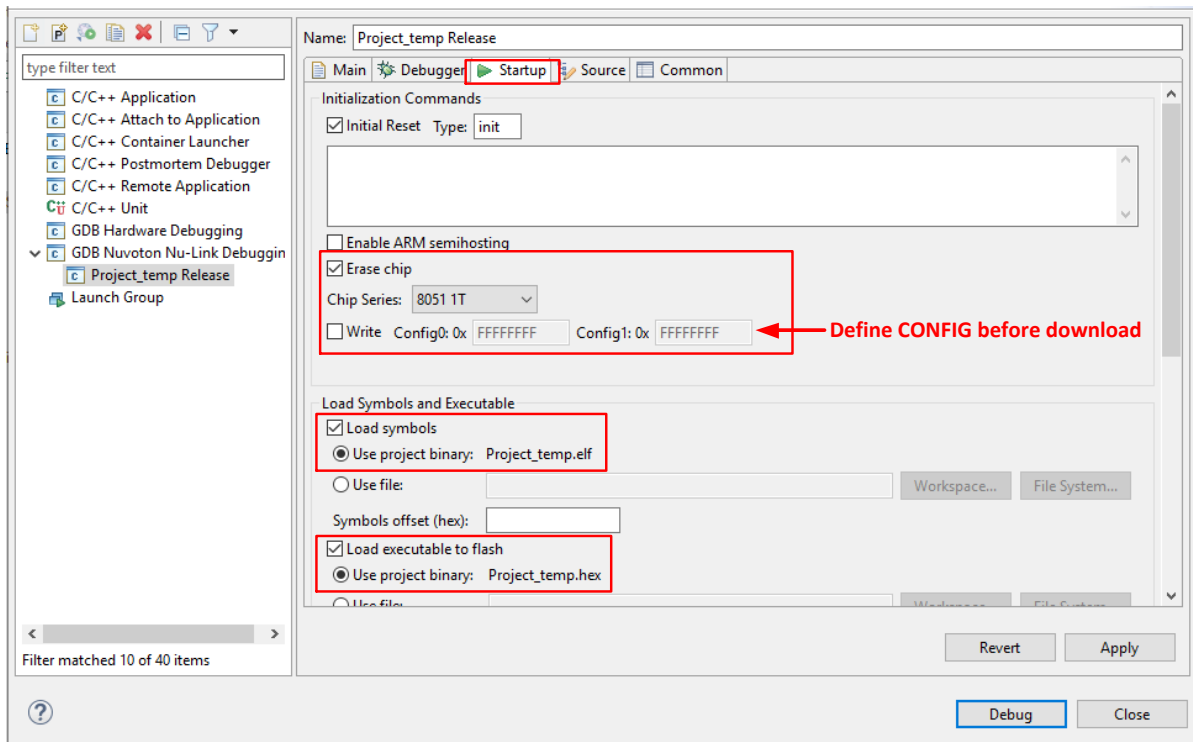
確認 Debugger 頁面內參數

GDB Client Setup 第一次可能需要手動填入參數，找到對應 Toolchain 下 gdb.exe 安裝路徑，初始值為 **C:/Program Files (x86)/Nuvoton Tools/Toolchain/SDCC/gdb/gdb.exe**

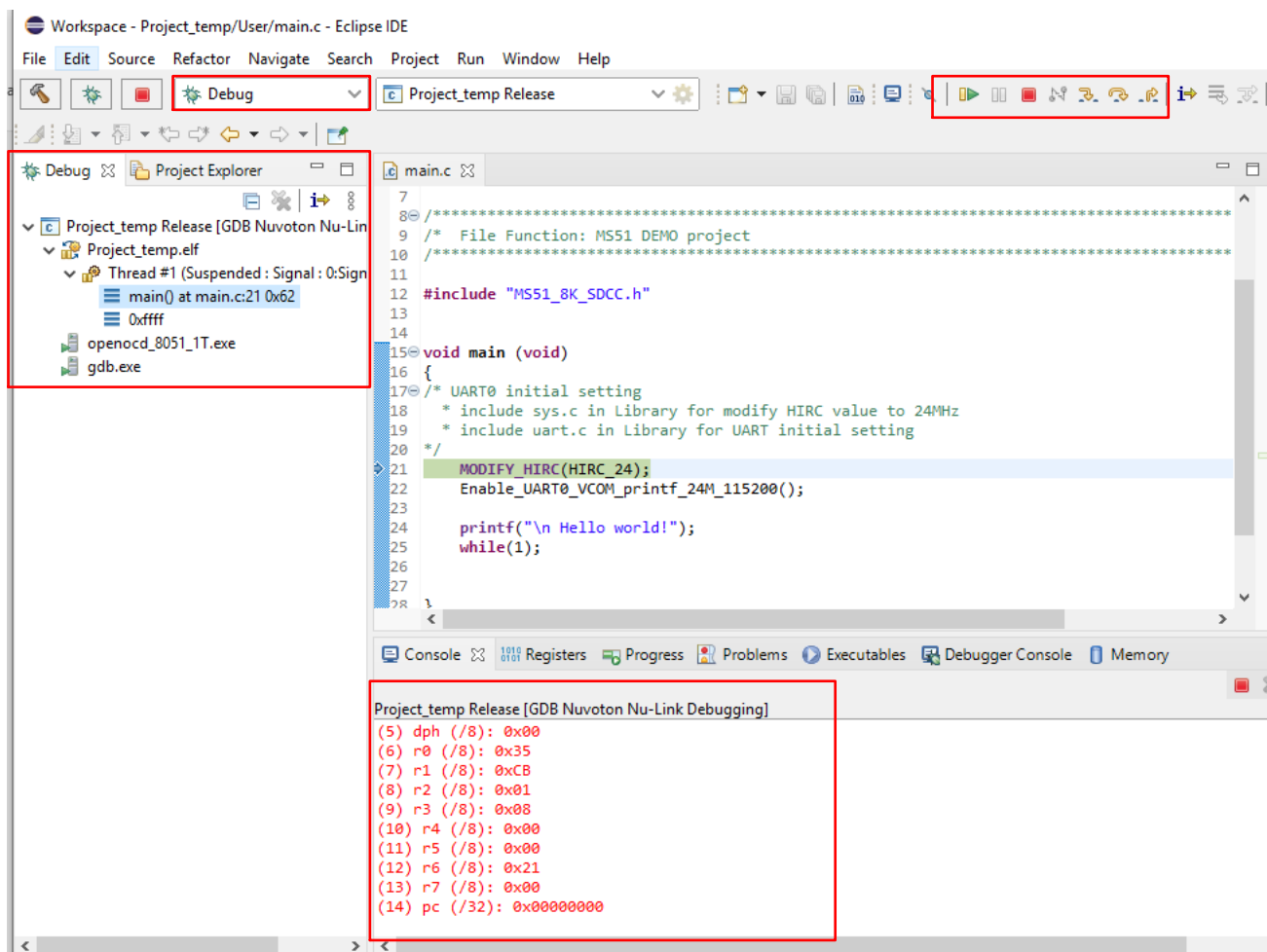


確認 Startup 頁面內參數, CONFIG 如需更改, 手動填入。

確認 load .elf / .hex 名稱與 Project 一致, 然後點選 Debug

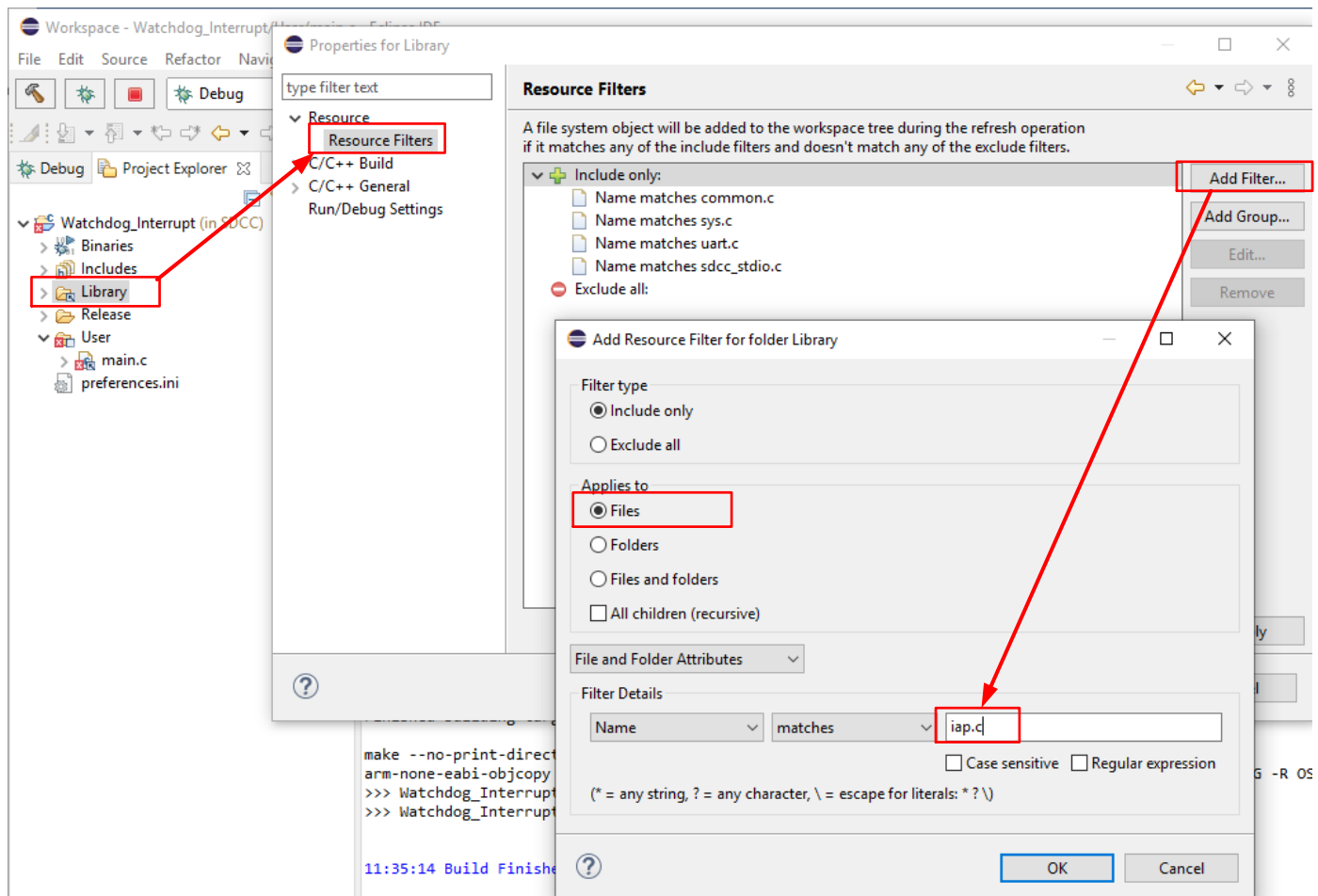


顯示如下視窗及說明進入 debug mode

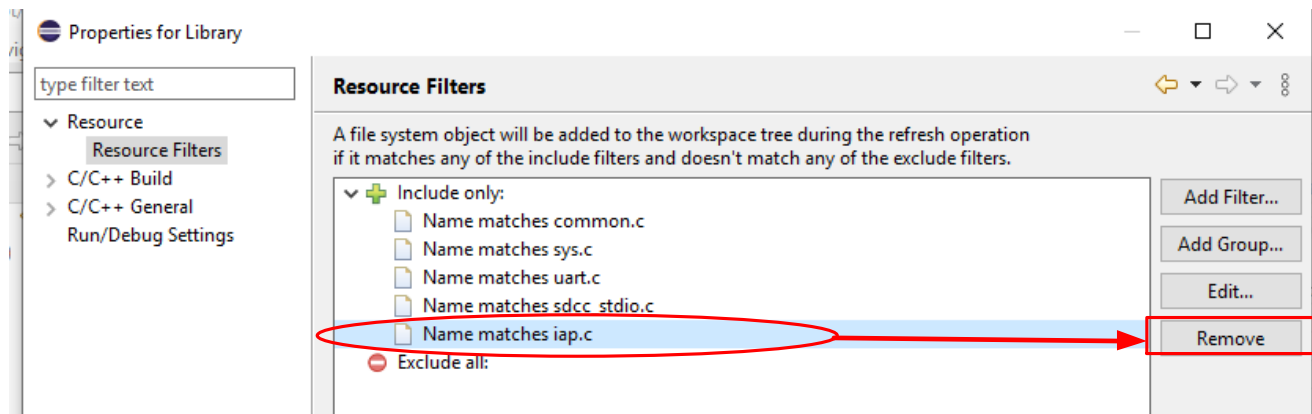


## 5. 增減 Library .c file

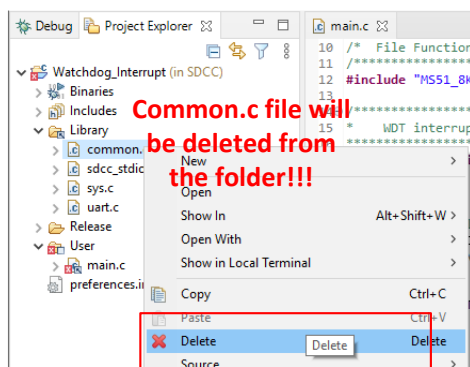
Project 內加入 lib .c file 進行 build ， 是採用 Add Filter 來增加



移除 lib ， 是 Remove Filter



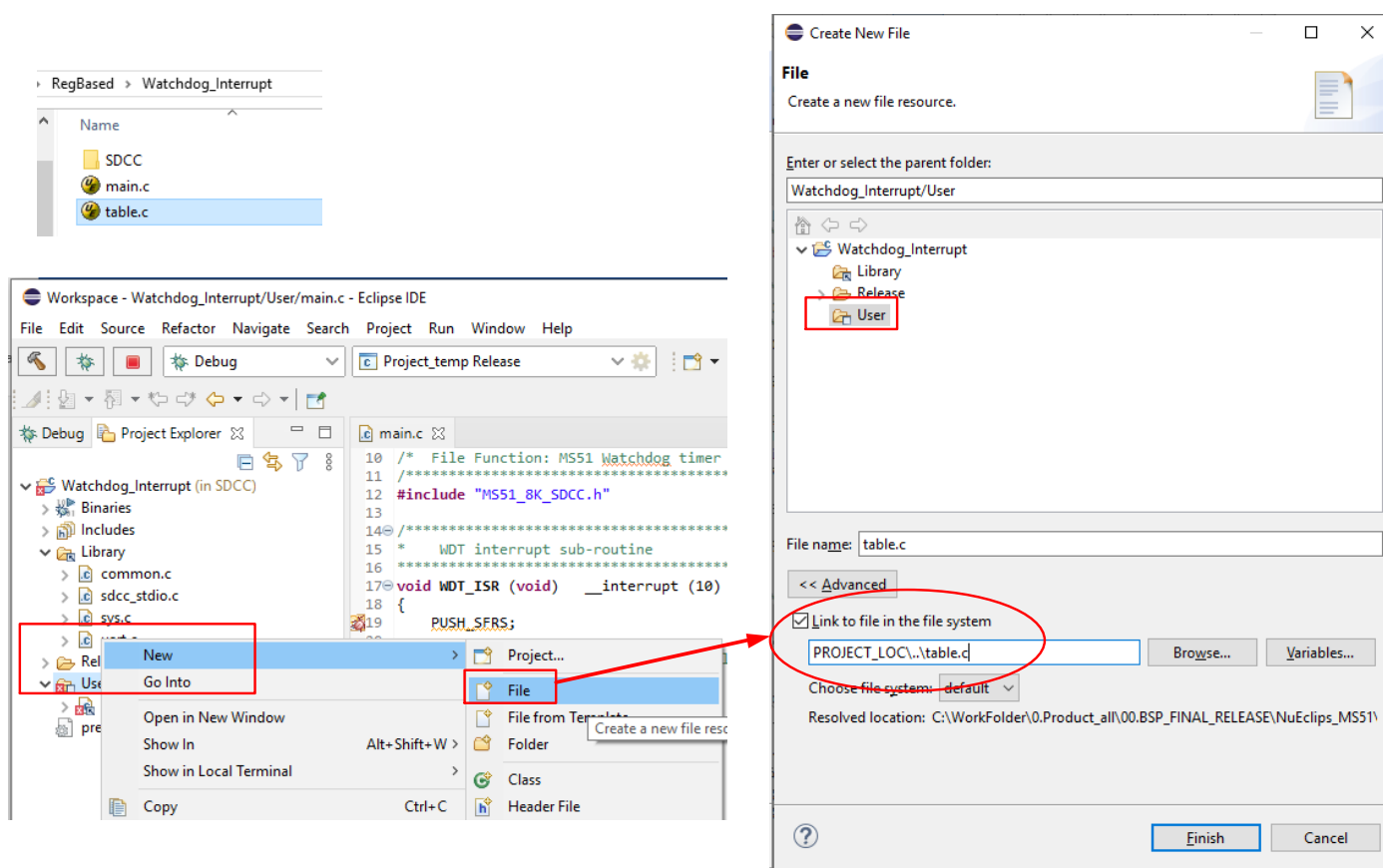
移除動作不能 Library 內選擇 delete ， 會直接在資料夾內刪除 source .c file



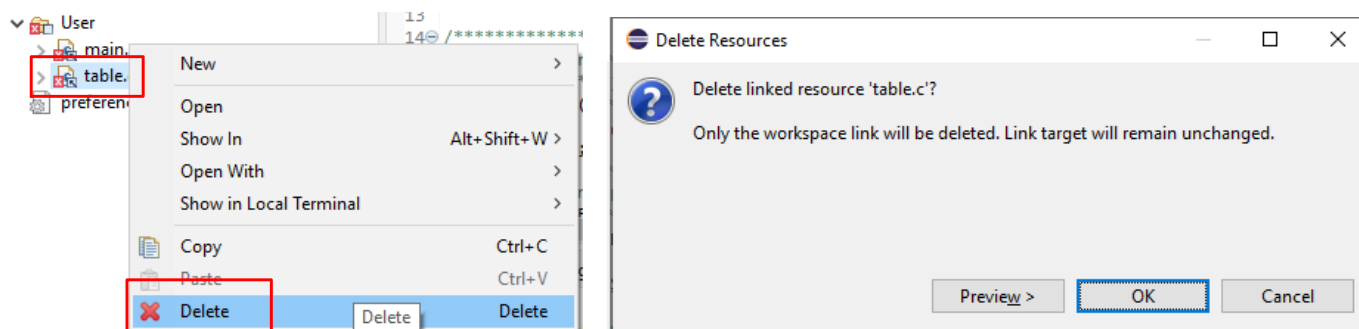
## 6. 增減 user code

新增一個 table.c ，右鍵點選 user / New / File 在 Create New File 頁面內點選 Link to file in the file system.建議填入相對路徑方式，絕對路徑會造成 Project 位置移動無法找到對應 file 問題。

**PROJECT\_LOC\.\<file name>**

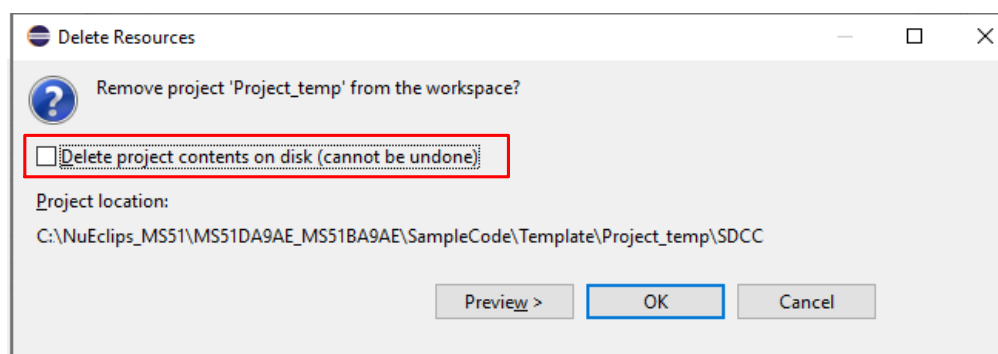


移除 user .c file



## 7. 關閉 Project

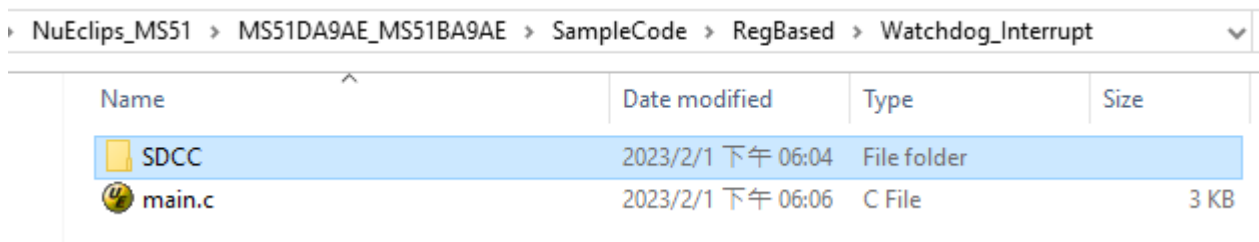
請不要勾選 delete project contents on disk, SDCC project folder 會被刪除。





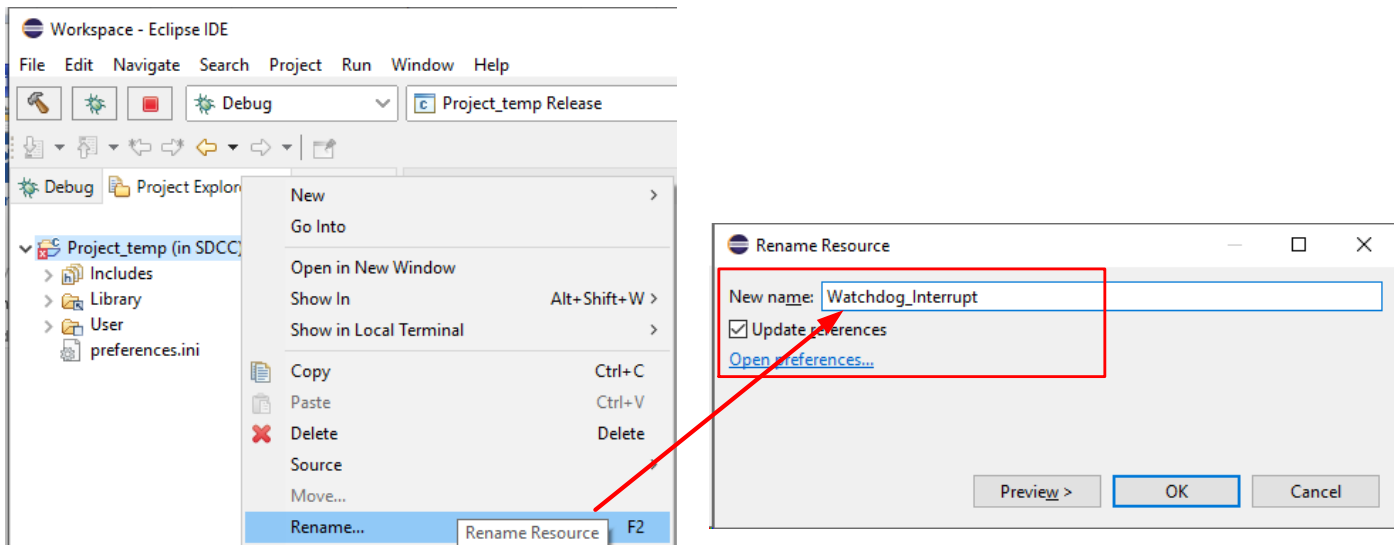
## 由現有 BSP project 創建新的 Project

### 1. 複製現有的 SDCC project folder 到新的 Project folder 中

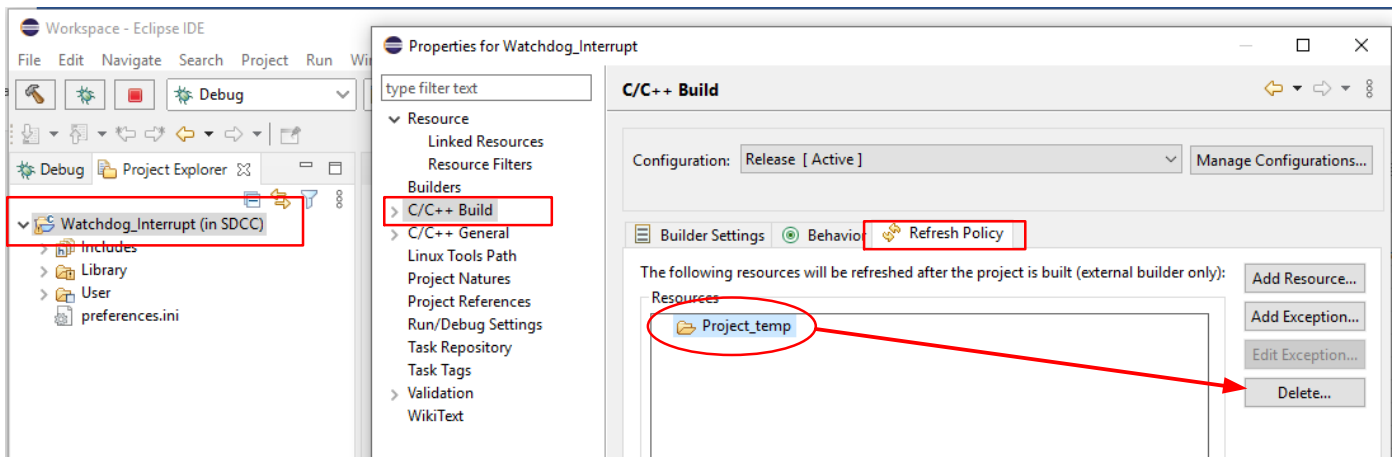


### 2. 在 Workspace 中 import 這個 project

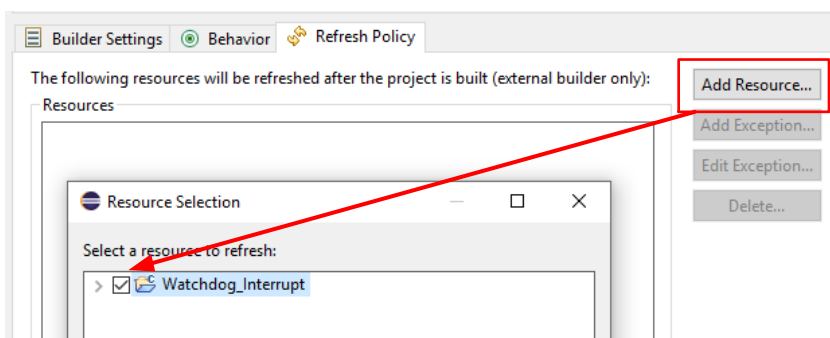
### 3. Rename Project



確認 Project 名稱已更改後，右鍵點選 Properties，修改 Refresh Policy，刪除舊 Resource

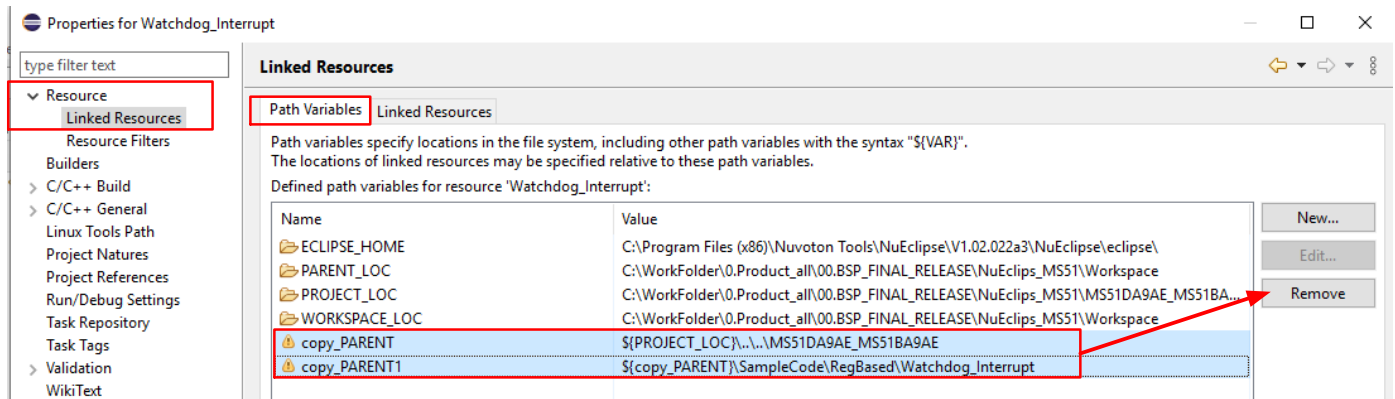


刪除後新增，系統會自動帶入 Rename 後的 Project Resource，手動勾選。





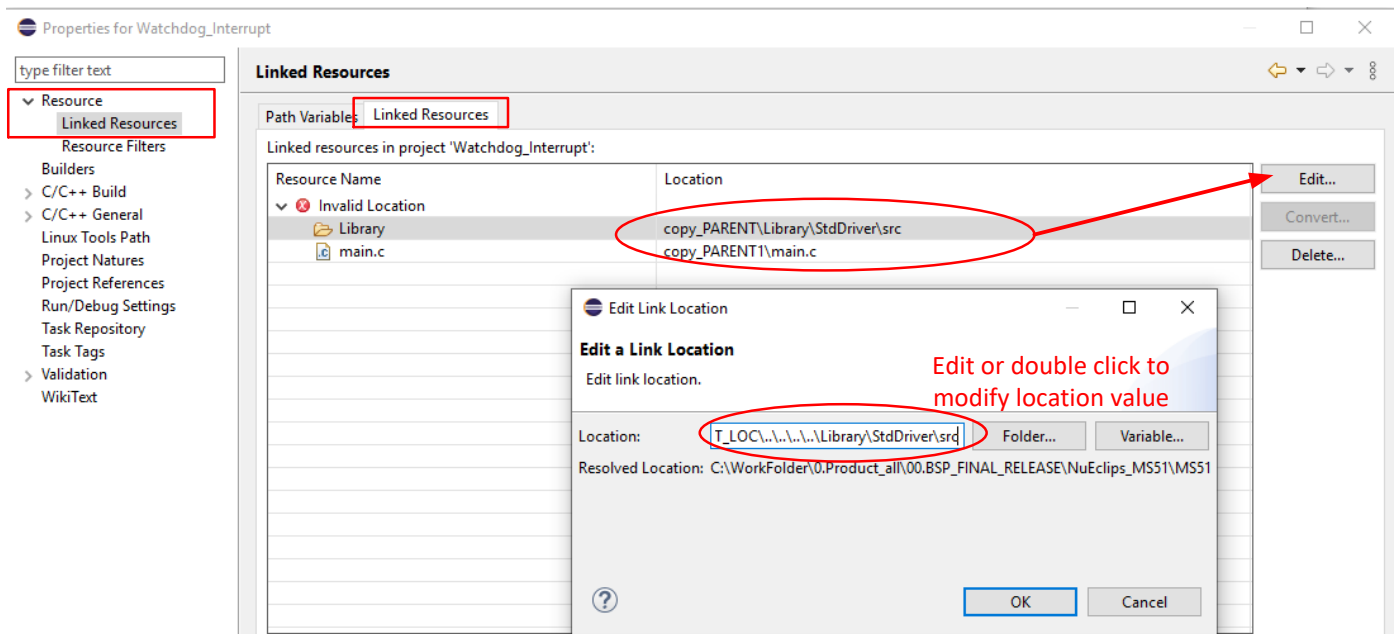
## 移除因 Rename 產生的錯誤 Resource



更正 Resource linker 設置，直接貼下列內容

**PROJECT\_LOC\..\..\Library\StdDriver\src**

**PROJECT\_LOC\..\main.c**

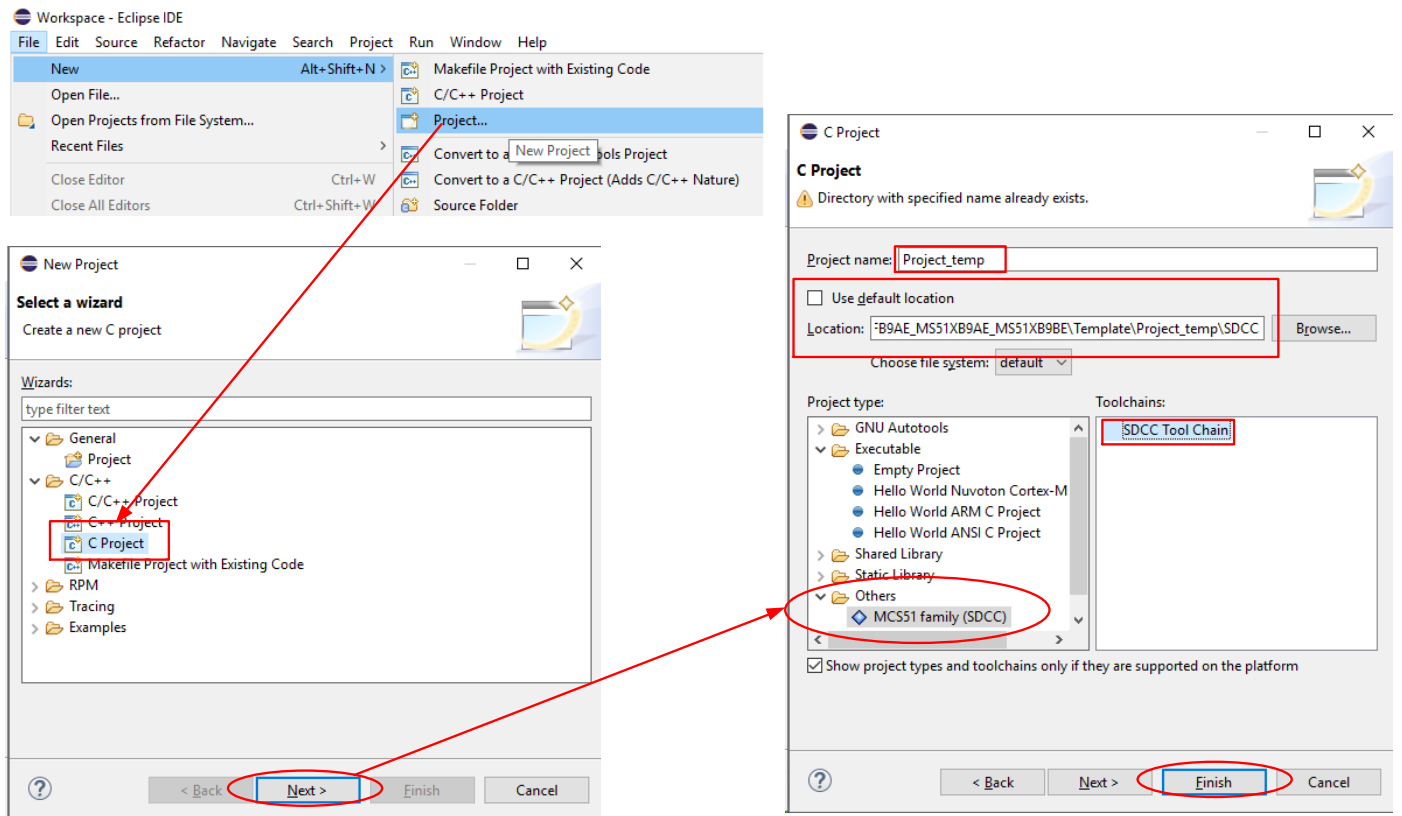


4. Build 更改後的 Project

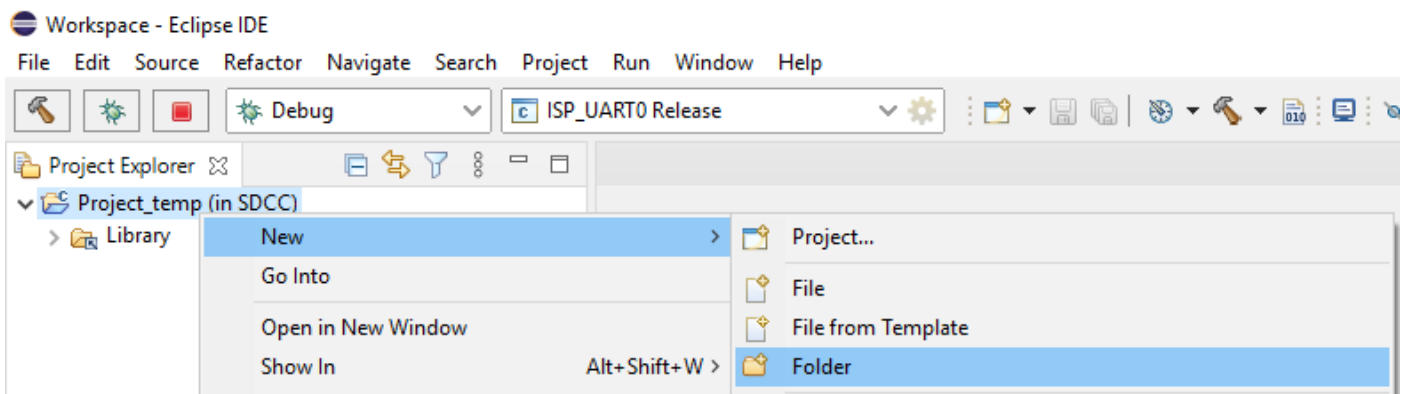
5. Debug

## 新建 Project （不建議）

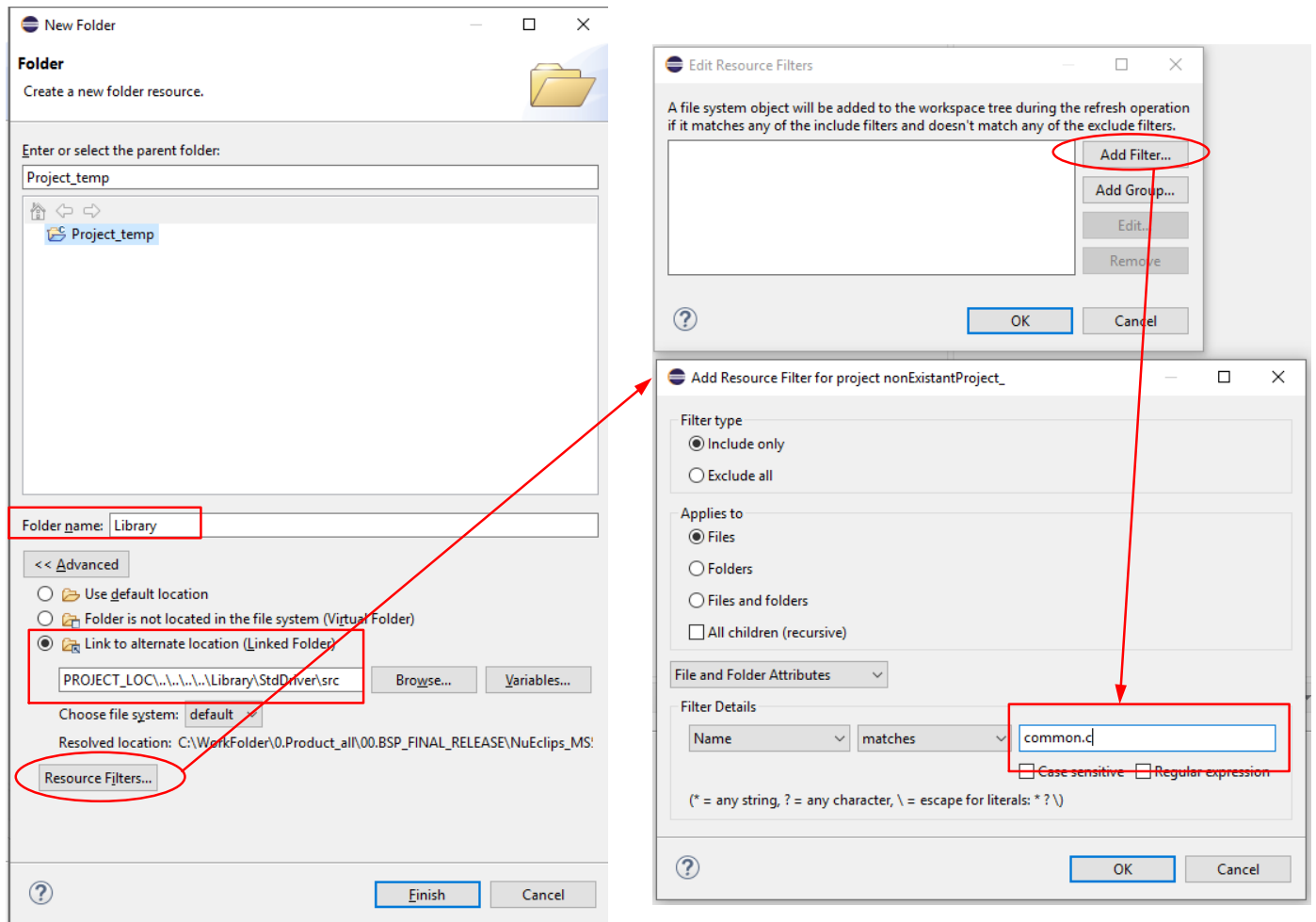
1. 選擇 Others SDCC Tool Chain, 選擇正確的 Location.



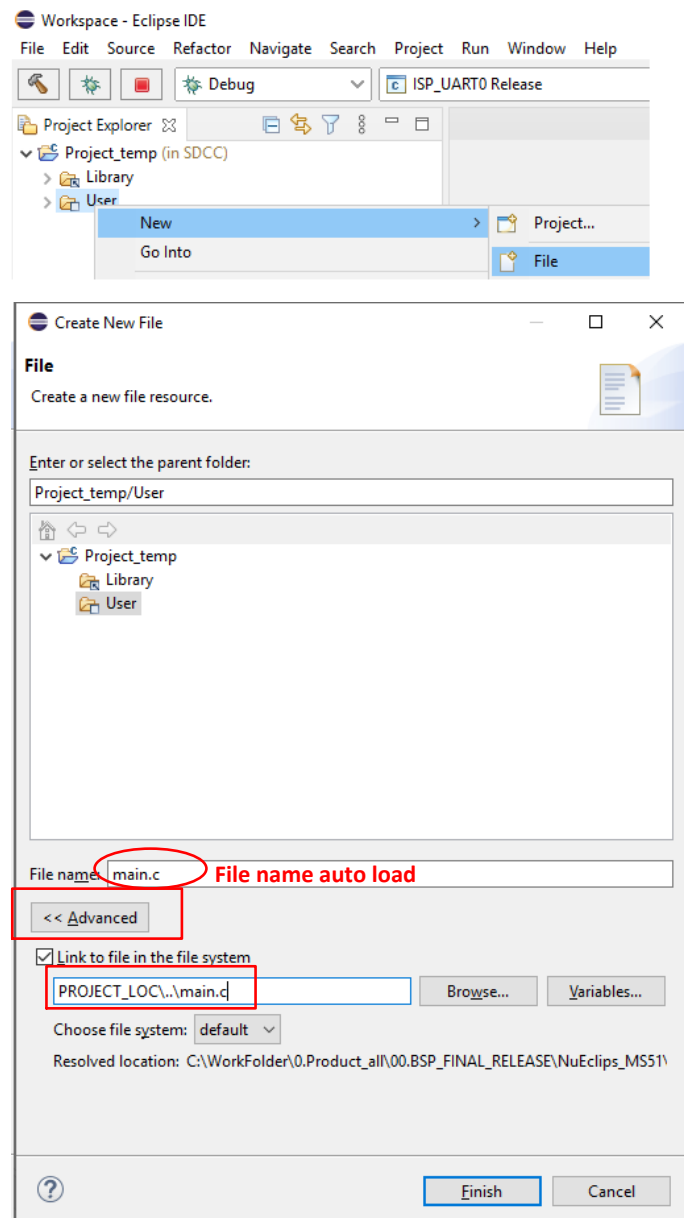
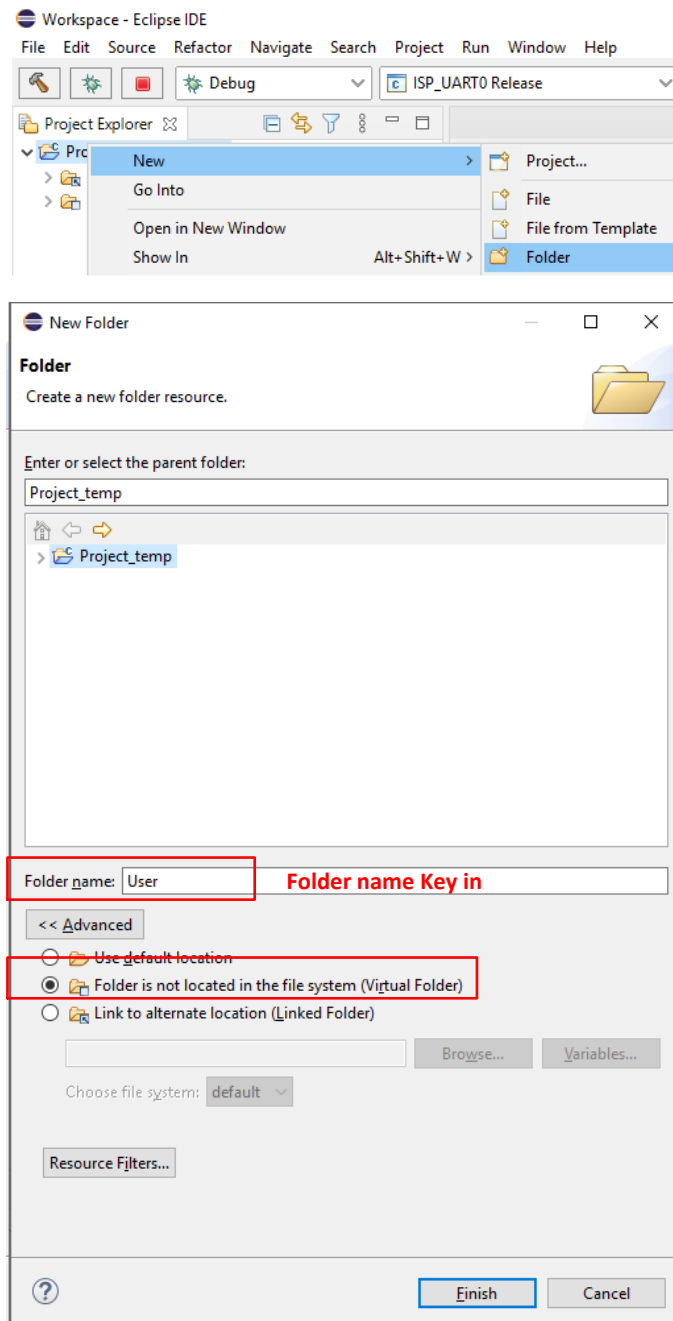
2. 新增 library 及 user source code



Link folder 方式. 可直接 link 已有的整個 folder 路徑, 不加入 filter 就自動加入 folder 內所有文件。  
新增或減少文件都是用 Filter 方式, 不能採用 delete。



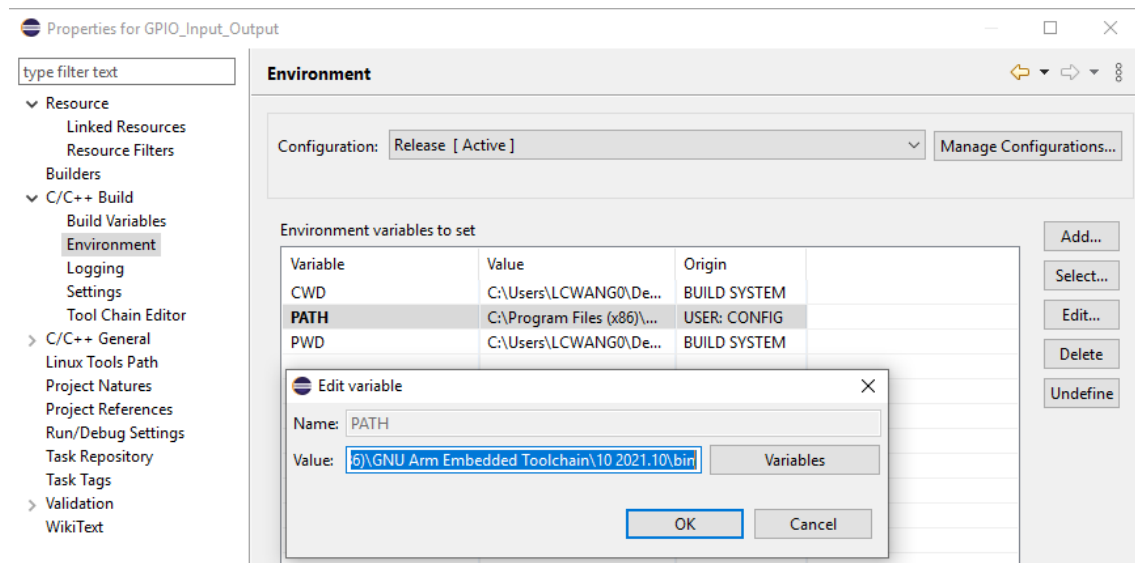
Link file 方式，先創建 Virtual Folder ，再加入文件 link ，每個文件為單一 link，可用 delete 刪除 link



### 3. 設定 project properties

#### a. Build / Environment / PATH

**C:\Program Files (x86)\Nuvoton Tools\Toolchain\SDCC\bin;C:\Program Files (x86)\GNU ARM Eclipse\Build Tools\2.8-201611221915\bin;C:\Program Files (x86)\GNU Arm Embedded Toolchain\10 2021.10\bin**

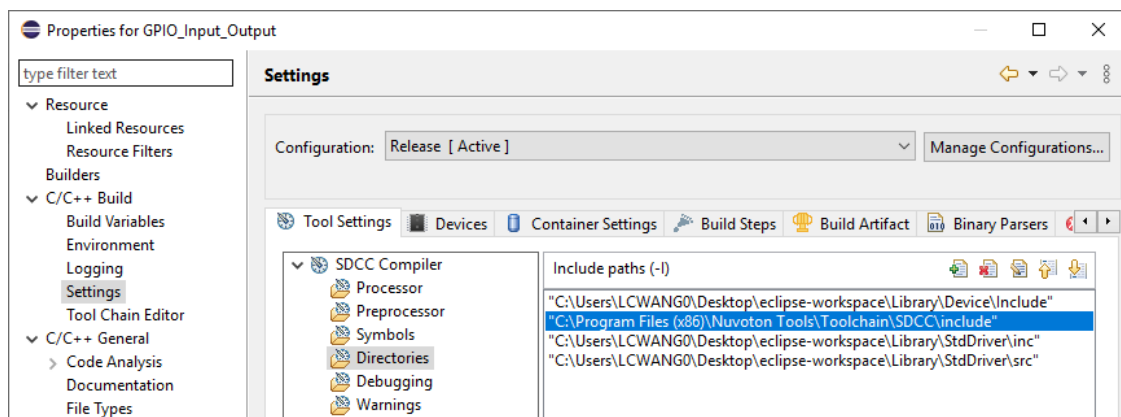


#### b. Build setting

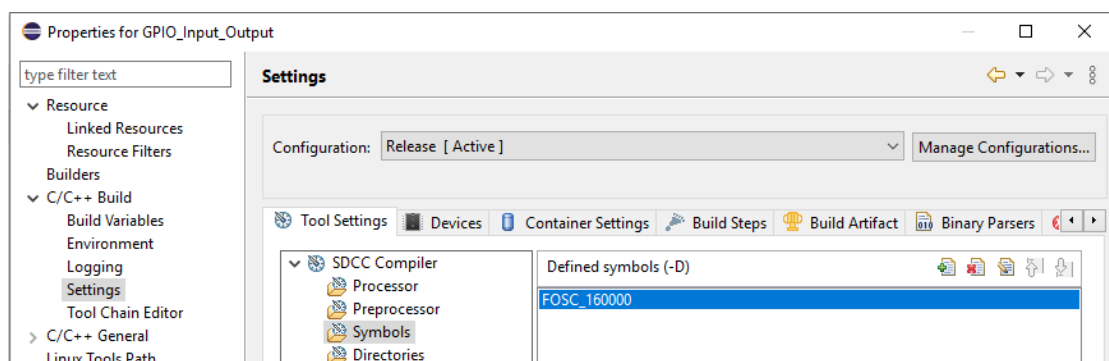
Symbols 請依 sample code 需要自行增加

Settings / Tool Settings / SDCC Compiler /Directories 要加入 SDCC include 路徑

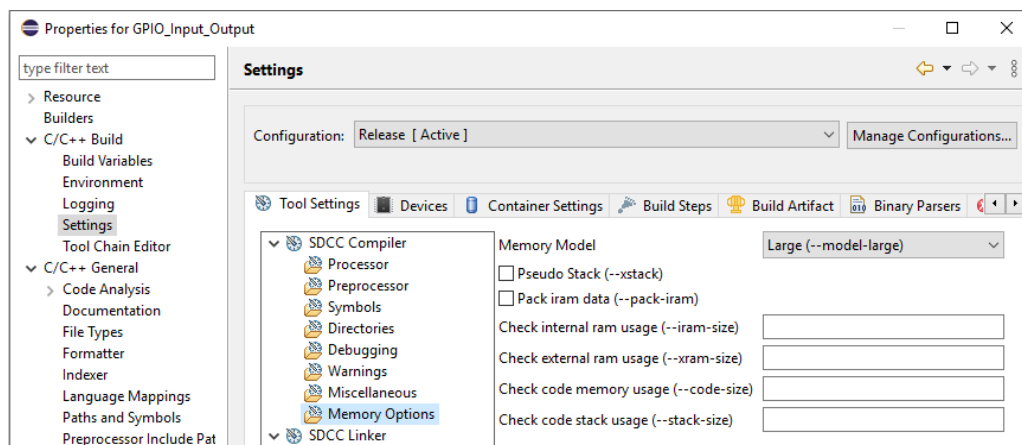
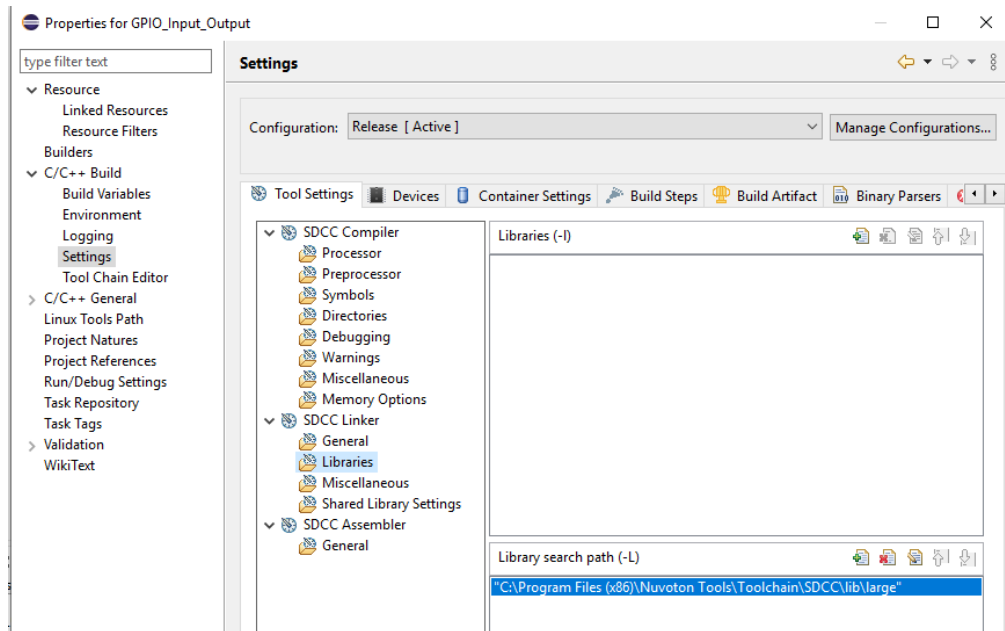
**C:\Program Files (x86)\Nuvoton Tools\Toolchain\SDCC\include**



#### c. Directories/ Tool Settings / SDCC Compiler / Symbols 寫 Define

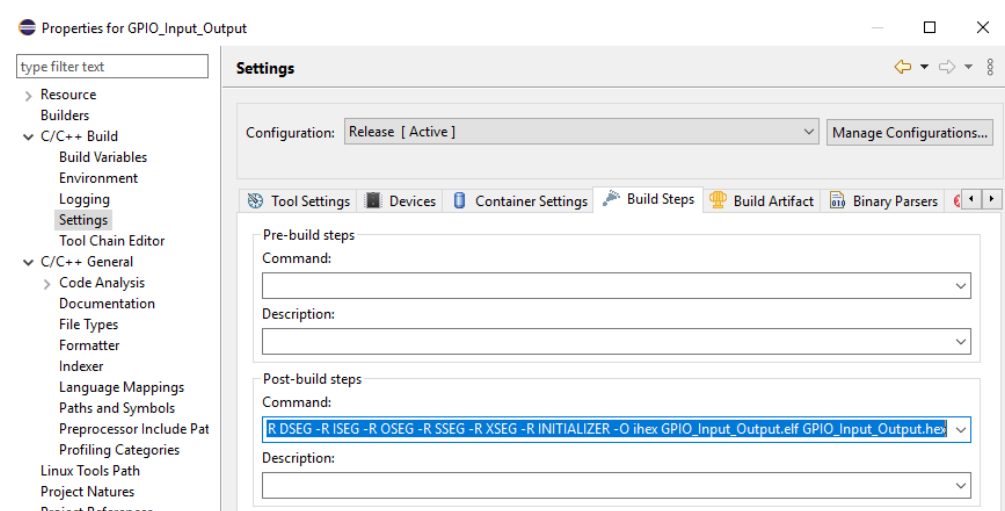


#### d. Directories/ Tool Settings / SDCC Linker / Library 需要依照 Directories/ Tool Settings / SDCC Compiler / Memory Options 設置 SMALL OR LARGE



e. Build Steps 要加入產生 hex 檔的 command, elf 和 hex 檔名請依自己的 project 名更改

**arm-none-eabi-objcopy -R REG\_BANK\_0 -R REG\_BANK\_1 -R REG\_BANK\_2 -R REG\_BANK\_3 -R BSEG -R DSEG -R ISEG -R OSEG -R SSEG -R XSEG -R XISEG -R INITIALIZER -O ihex \${ProjName}.elf \${ProjName}.hex;Hex2bin \${ProjName}.hex**



f. Code Analysis 取消紅框的兩個選項

type filter text

- > Resource
- Builders
- ▼ C/C++ Build
  - Build Variables
  - Environment
  - Logging
  - Settings
  - Tool Chain Editor
- ▼ C/C++ General
  - > Code Analysis
  - Documentation
  - File Types
  - Formatter
  - Indexer
  - Language Mappin
  - Paths and Symbols
  - Preprocessor Inclu
  - Profiling Categor
- Linux Tools Path
- Project Natures
- Project References
- Run/Debug Settings
- Task Repository
- Task Tags
- > Validation
- WikiText

## Code Analysis

- ☐ Use workspace settings  
☒ Use project settings

[Configure Workspace Settings...](#)

## Problems

type filter text

Name	Severity
<input checked="" type="checkbox"/> Suspicious semicolon	Warning
<input type="checkbox"/> Symbol shadowing	Warning
<input checked="" type="checkbox"/> Unused function declaration	Warning
<input checked="" type="checkbox"/> Unused return value	Error
<input checked="" type="checkbox"/> Unused static function	Warning
<input checked="" type="checkbox"/> Unused variable declaration in file :	Warning
<input type="checkbox"/> Virtual method call in constructor/c	Error
▼ <input type="checkbox"/> Security Vulnerabilities	
<input type="checkbox"/> Format String Vulnerability	Warning
▼ <input checked="" type="checkbox"/> Syntax and Semantic Errors	
<input checked="" type="checkbox"/> Abstract class cannot be instantiate	Error
<input checked="" type="checkbox"/> Ambiguous problem	Error
<input checked="" type="checkbox"/> Circular inheritance	Error
<input checked="" type="checkbox"/> Field cannot be resolved	Error
<input checked="" type="checkbox"/> Function cannot be resolved	Error
<input checked="" type="checkbox"/> Invalid arguments	Error
<input checked="" type="checkbox"/> Invalid overload	Error
<input checked="" type="checkbox"/> Invalid redeclaration	Error
<input checked="" type="checkbox"/> Invalid redefinition	Error
<input checked="" type="checkbox"/> Invalid template argument	Error
<input checked="" type="checkbox"/> Label statement not found	Error
<input checked="" type="checkbox"/> Member declaration not found	Error
<input checked="" type="checkbox"/> Method cannot be resolved	Error
<input type="checkbox"/> Symbol is not resolved	Error
<input type="checkbox"/> Type cannot be resolved	Error



### 3. Debug Environment

設定不同區域位置

memory space of SDCC declaration	Data access address
__code	(CODE address) + 0
__data	(DATA address) + 0xF0000000
__idata	(IDATA address) + 0xF1000000
__xdata	(XDATA address) + 0xF2000000

#### a. 使用 Expression window 查看變數

Expression	Type	Value
(x)= *(&SBUF+0xF0000000)	volatile unsigned char	0 '\0'
(x)= *(&trimvalue16bit+0xF2000000)	unsigned int	12562
+ Add new expression		

#### b. 使用 memory window 查看

Monitors

0x00 + 0xF2000000

Monitor Memory

Enter address or expression to monitor:

0x00 + 0xF2000000

OK Cancel

Address	0 - 3	4 - 7	8 - B	C - F
F2000000	34CB0100	00790E80	00000000	00000000
F2000010	00000000	00000000	00000000	00000000
F2000020	08000000	00000000	00000000	000060CB
F2000030	01006B13	807A0000	2E09510A	0000004A
F2000040	0937BD08	00000000	00000000	00000000
F2000050	00000000	00000000	00000000	00000000
F2000060	00000000	00000000	00000000	00000000
F2000070	00000000	00000000	00000000	00000000
F2000080	00000000	00000000	00000000	00000000
F2000090	00000000	00000000	00000000	00000000
F20000A0	00000000	00000000	00000000	00000000

變更顯示格式

Expression

Type

Default

Hex

Decimal

Octal

Binary

String

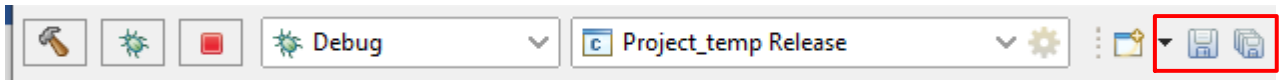
Number Format

Layout

Expression	Type
(x)= *(&itmp+0xF1000000)	volatile unsigned int
(x)= *(&xtemp+0xF2000000)	volatile unsigned int
(x)= *(&dtemp+0xF0000000)	volatile unsigned int
(x)= xtemp + 0xF2000000	unsigned int
+ Add new expression	

# TIPS

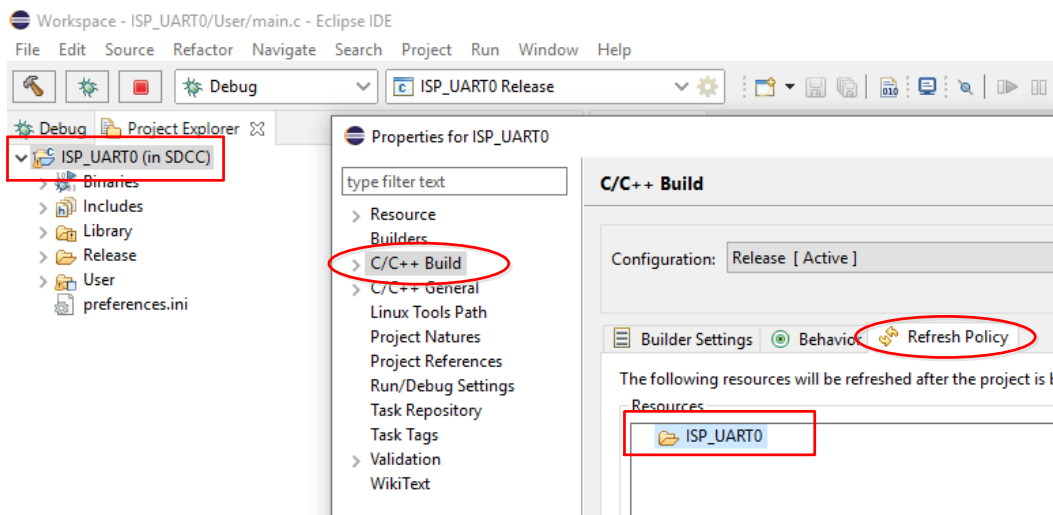
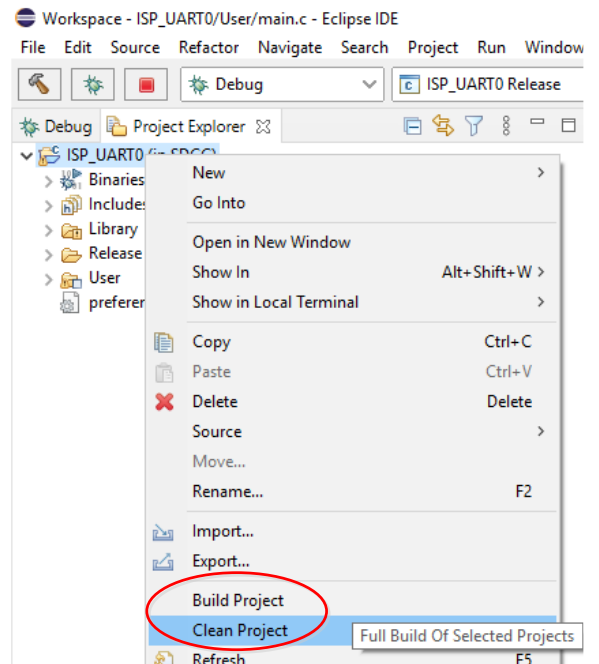
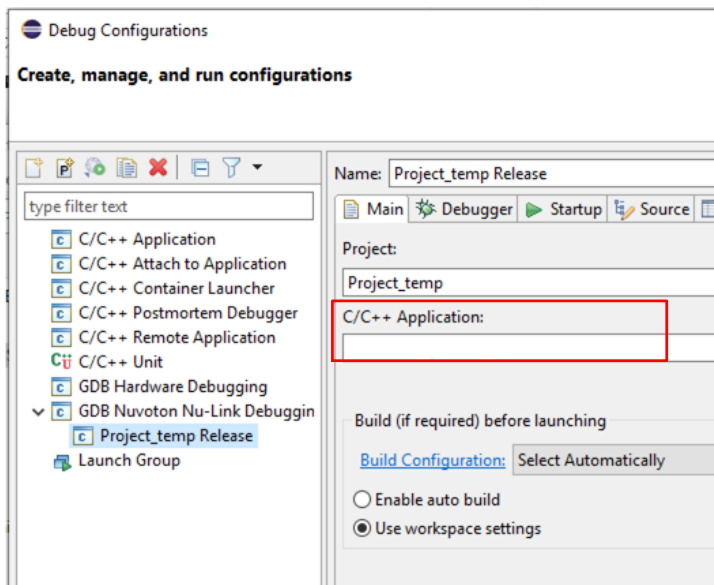
1. 所有文件附檔名都要用小寫。
2. Interrupt Vector 要放在 main.c 中。
3. 更改過 code 內容後，build 之前一定要重新存檔。



4. Debug mode 未顯示正確的.elf

Clean Project 並再次 Build Project 直至沒有 error。

確認 project Refresh Policy 建立正確，名字與 Project 一致。



- 5.