# STM32 rosserial手冊

ROS(Robot Operation System)與STM32微控制器透過串列互相溝通教學手冊，此方式與ROS Arduino相同之方式，所以寫有關ROS之程式可參考Arduino的官方教學。

使用之硬體為: STM32F103C8

開發環境為: openSTM32

## Step 1: 使用STM32CubeMX建立專案

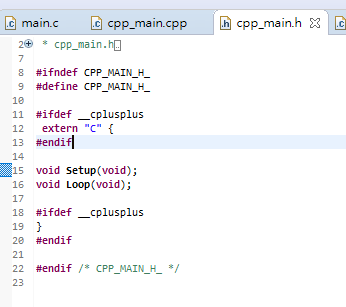
至STM32CubeMX建立一個給ROS USART之專案，需注意以下設定:

1. RCC: 至High Speed Clock選單選擇Crystal/Ceramic Resonator石英震盪器
2. SYS: Debug選單選擇Serial Wire
3. Connectivity → USARTx → Mode(Asynchronous)；Baud Rate: 預設為57600(可更改，但在作業系統中須更改相同鮑率)；NVIC Settings tab → 請將USARTx global interrupt使能；DMA Settings tab → Add新增兩個DMA Request(RX、TX)且RX之Mode選項請改成Circular模式，TX則為不更改(Normal模式)。x*為你設定之串列編號*
4. 至Clock Configuration中配置系統時間: 72MHz。

以上步驟完成即可生成專案，如有不清楚可參考檔案中之\*.ioc檔。

## Step 2: 搬移Library檔

1. 將參考檔案中之RosLibs資料夾移植專案資料夾當中，並將它Include至Ecilpse中(右鍵Project Explorer專案的Properties → C/C++ Build → Settings → GCC/G++ Compiler*兩個編譯器皆須設定* → includes做添加)。
2. 並將duration.cpp、time.cpp放進你的Src資料夾當中；round.h放進Inc資料夾中。
3. 將專案從C語言轉換成C++(右鍵Project Explorer中之專案可進行轉換)。
4. stm32f1xx\_it.c中找到SysTick\_Handler(void)函式庫添加一行Code為: HAL\_SYSTICK\_IRQHandler()。
5. 從Src資料夾中新增UserMain.cpp、UserMain.h檔(名稱可由你來設定)將此檔作為你的專案撰寫檔。並在UserMain.cpp中除了你所需的標頭檔外，並include “main.h”及include “stm32f1xx\_hal.h”其他添加你所需的Header檔。而在底下寫兩個副程式void Setup(void)、void Loop(void)並且記得在UserMain.h中做兩個副程式之宣告，且須添加一些指令如下



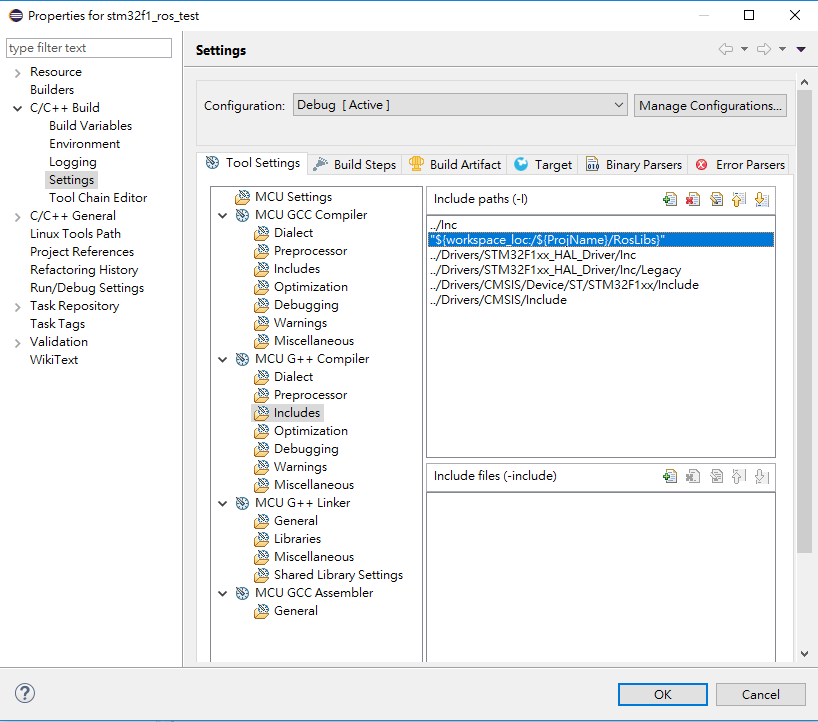
告知編譯器此檔為C++檔並可在C檔中執行。

而在UserMain.cpp中需添加傳輸設定兩個副程式(中斷向量函式):void HAL\_UART\_TxCpltCallback(UART\_HandleTypeDef \*huart)、void HAL\_UART\_RxCpltCallback(UART\_HandleTypeDef \*huart)，如下



完成這些設定後，請至main.c中include “UserMain.h”標頭檔及Setup()、Loop()。

補充: 吾人使用之硬體為STM32F103C8，如使用者使用非為F1系列須將有關xxxxxf1.xx中的1改成你使用之系列，如F4系列則將所有改成stm32f4xx\_hal.h。

下圖為Step 2 – (1)之說明圖:

1. 在RobLibs資料夾中STM32Hardware.h更改你的硬體系列與你所使用之串列編號(huartx，x則改成你所使用的編號，如吾人使用串列1則改成huart1)。
2. 以上皆完成後並寫完你的專案後即可進行編譯，透過Debug As → Ac6 STM32 C/C++ Application做燒寫(如有錯誤請至Debug Configurations中Debugger tab → Mode Setup →Reset Mode:Software system reset)。