STM32 bootloadet 跳轉學習

Bootloader 根據開關不同跳轉不同APP

APP1 LED BLINK + UART PRINT”APP1”

APP2 LED BLINK + UART PRINT”APP2”

Bootloader 更新APP

Bootloader 更新APP1文字

APP1 更新 bootloader

檢查機制

Bootloader

Boot 對boot

Boot 對 MAIN

MAIN 對 Boot

MAIN 對 MAIN

首先先了解晶片的記憶體位置，我們使用STM32F103C8T6進行學習

軟體的話，使用keil IDE及STM32CubeIDE

那首先使用STM32cubeIDE建立環境

依據官方教材依序學習

**第一章**

目標

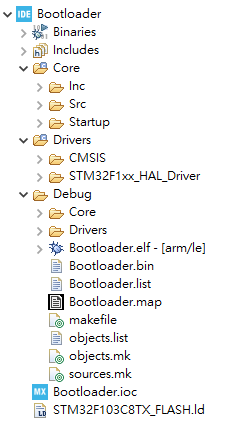
1. 在連結檔案中建立一個部分
2. 在建立的部分中建立一個可變的區段

步驟

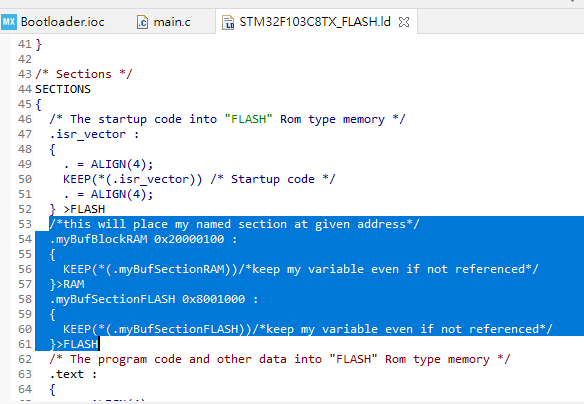
1. 透過STM32 Cube IDE編輯linker file並增加區段

學習紀錄

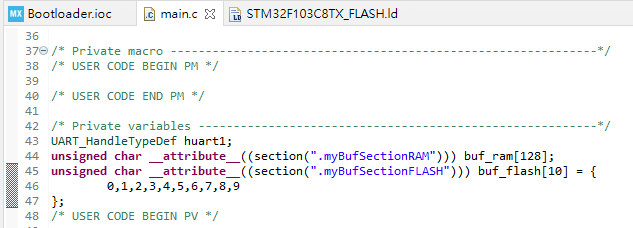
空專案建立好後，開啟專案中的.ld的檔案如下圖



先照著影片中，找到要加入的部分



Main檔案中也要新增對應的變數



過程中呢有一個有趣的小實驗

如果我們在自己建立的區域時後面加註(NOLOAD)時，bin檔會變小

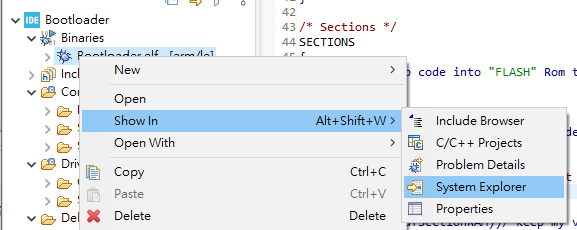
.myBufBlockRAM 0x20000100 (NOLOAD):

{

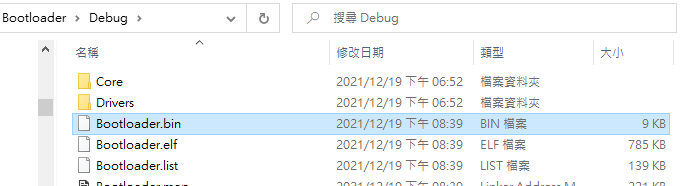
KEEP(\*(.myBufSectionRAM))/\*keep my variable even if not referenced\*/

}>RAM

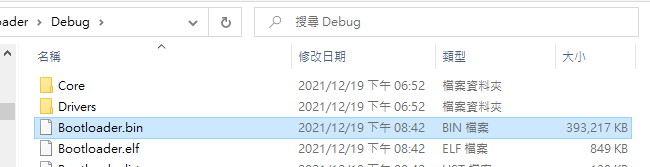
看的方式如下



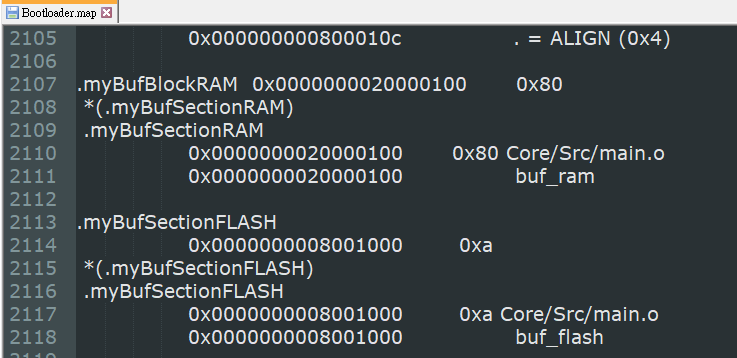
有加NOLOAD



沒加NOLOAD



.map檔產生的樣子



今天來學習第二章