

微處理機 Lab1 Report

一、 實驗名稱：實驗環境建立與 Debugger 操作。

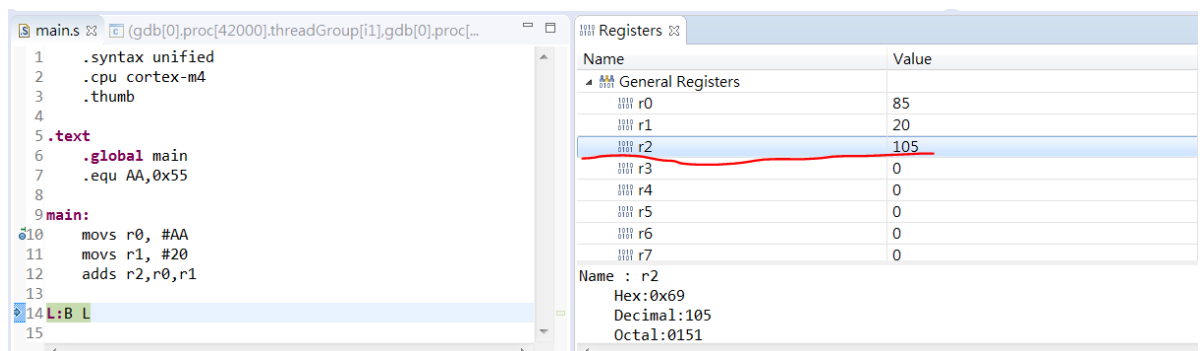
二、 實驗目的：測試實驗器材與熟悉開發環境。

三、 實驗步驟：

1. 專案建立與程式編譯。
2. 變數宣告與記憶體觀察。
3. 簡易算數與基本記憶體指令操作。

四、 實驗結果與分析：

1. 程式執行結束後 r2 的值為 105，觀察的方式為 build 完專案後開啟 debugger，當執行完 `adds r2, r0, r1` 後即可在 Watch 內的 register 欄位看到 r2 的值為 105，如下圖所示：

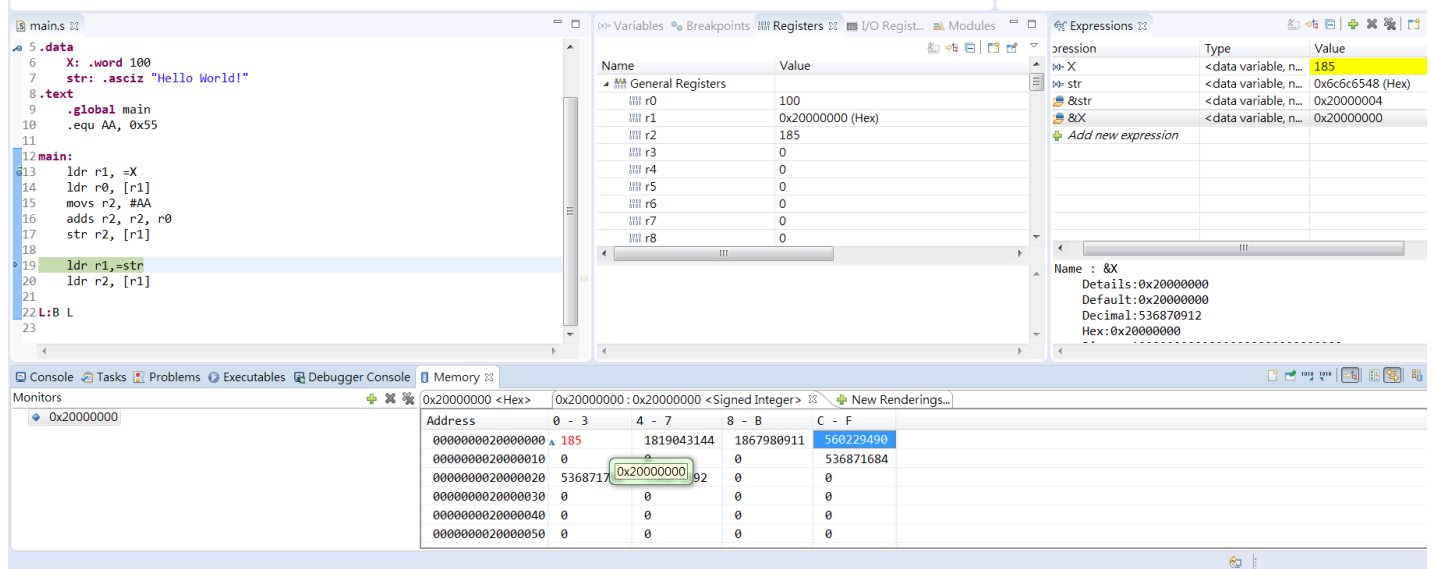


2. 變數宣告與記憶體觀察：

Q1：在 `startup_stm32.s` 這個檔案裡面的 `CopyDataInit` 去初始化變數的值。

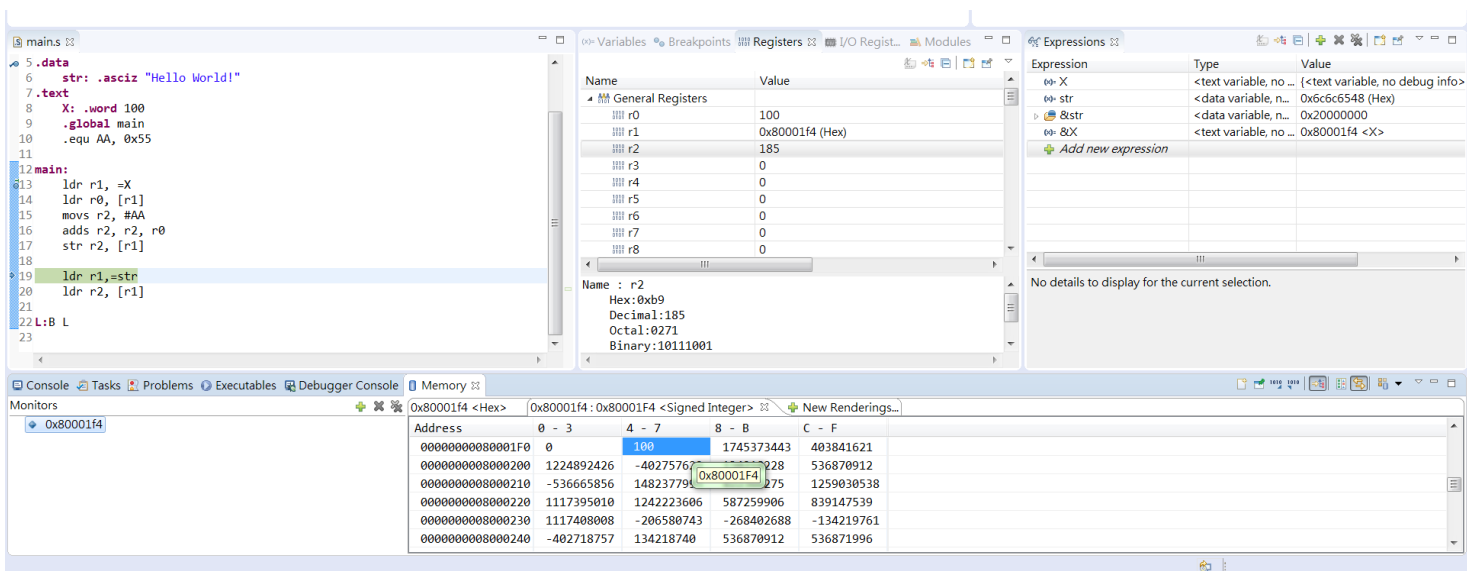
Q2：把 X 宣告改在 text section 後執行下去與宣告在 data section 幾乎一樣，唯獨不一樣的就是當 X 宣告在 data section，`str r2, [r1]` 這條指令

會將 X 在記憶體裡面的存值改為 r2 的值，但當 X 宣告在 test section 後，str r2, [r1]這條指令執行完後，X 在記憶體裡面的存值仍是原本的存



值，並沒有變為 r2 的值。如下圖所示：

↑ 當 X 宣告在 data section 時，可以明顯看出 str r2, [r1]執行完，X 在記憶體中的值變為 185。



↑ 當 X 宣告在 text section 時，可以明顯看出 str r2, [r1]執行完，X 在記憶體中的值仍是 100。

Q3 : r2 內容與 str 字串在 memory 前 4 個 byte 的內容是一模一樣的。

Q4 : str: .byte 'H','e','l','l','o',' ','W','o','r','l','d','!'

3. 簡易算數與基本記憶體指令操作：

```
rtup_stm32.s  x  main.s
1  »  .syntax unified
2  »  .cpu cortex-m4
3  »  .thumb
4  .data
5  »  X: .int 0
6  »  Y: .int 0
7  »  Z: .int 0
8  .text
9  »  .global main
10
11  main:
12  »  ldr r0,=X //r0=&X
13  »  ldr r1,=Y //r1=&Y
14  »  ldr r2,=Z //r2=&Z
15  »  ldr r3,=0x5
16  »  str r3,[r0] //X=5
17  »  ldr r3,=0xA //r3=10
18  »  ldr r4,[r0] //r4=X
19  »  str r3,[r1] //Y=10
20  »  ldr r5,[r1] //r5=Y
21  »  mul r4,r4,r5 //r4=r4*r5(X*10)
22  »  adds r4,r4,r5 //r4=r4+r5
23  »  str r4,[r0] //X=r4=X*10+Y
24  »  subs r4,r5,r4 //r4=r5-r4=Y-X
25  »  str r4,[r2] //Z=r4=Y-X
26  L:B L
27
```

I. 6~8 這三行是在宣告 X、Y、Z 這三個變數。

II. 12~14 是分別把這三個變數的地址讀進暫存器以方便後續的賦值。

III. 第 16 及第 19 行則是把 X 與 Y 初值設定好，[r0]的意思是指把某值存入 r0 內位址所指到的記憶體區塊。

IV. 第 21~24 行則是進行作業內要求的四則運算。

V. 第 23 及第 25 行則是更新 X 的值與將最終結果寫入 Z 變數。

五、心得討論與應用聯想：

這次的作業雖然大部分沒有太難，但教授上課還尚未提及組語相關的寫

法，所以導致在做作業時，其實很多是用猜的，並沒有辦法實證自己的寫法有沒有錯，另外，也遇到了一些未解的 IDE BUG，導致作業困難。

聯想的部分是希望未來將可以寫出自己的組語計算機，加速計算的執行。