微處理機 LAB 1

Due : 兩周後 上午8:00

PART 1. (50%)

1. 查閱programming manual，寫出MOV，STR，LDR用法與差異。(30%)

|  |  |  |  |
| --- | --- | --- | --- |
| Operator | Description | Example |  |
| MOV | Move. Copy the second operand’s value to the first operand (register). | MOV R0, R1; | Copy R1’s value and store it in R0. |
| STR | Store register. | STR R0, [R1]; | Store the value in R0 to the register which has address R1. |
| LDR | Load register. | LDR R0, [R1]; | Load the value from the register which has address R1 to R0. |

1. 舉一個暫存器間接定址法的程式碼並說明其運作過程。(20%)

MOV R0, #0x30; // 將0x30存入register R0

LDR A, [R0]; // 將位址為0x30的register中的word存入A

PART 2. (50%) 實作題 請完成實驗 截圖紀錄實驗結果並附上程式碼

1. 組內組員，一人一題 (50%)
2. 用組合語言寫出20H - 10H 並在register中追蹤其數值相加變化
3. 用組合語言寫出5H x 9H  並在register中追蹤其數值相加變化

(請分別擷取計算前register中的值及計算後之值的變化)

a.

.syntax unified

.cpu cortex-m4

.thumb

.text

.global main

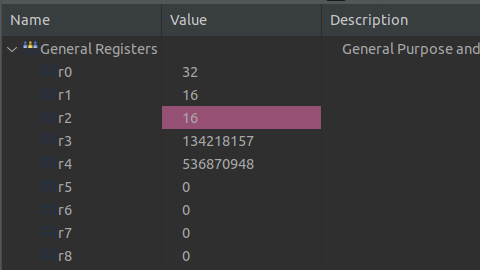
.equ AA, 0x55

main:

movs r0, #0x20

movs r1, #0x10

subs r2, r0, r1



b.

.syntax unified

.cpu cortex-m4

.thumb

**.text**

**.global** main

.equ AA, 0x55

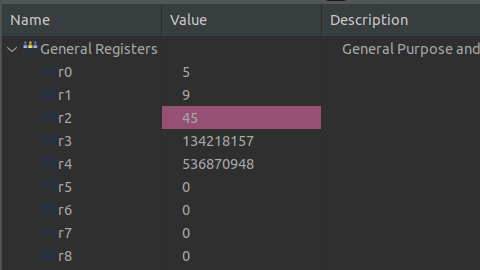
**main:**

movs r0, #0x5

movs r1, #0x9

mul r2, r0, r1

B main



PART 3. 加分練習，不計入平常成績

Fibonacci serial:  宣告一數值N (1≤N≤100)，計算Fib(N)並將回傳值存放至R4暫存器

Tips: Fib(0) = 0 ; Fib(1) = 1 ; Fib(N) = Fib(N-1) + Fib(N-2) for N>1

.syntax unified

.cpu cortex-m4

.thumb

**.text**

**.global** main

.equ AA, 0x55

**fib:**

cmp r1, r0

bge endfib

add r3, r4, r2

mov r4, r2

mov r2, r3

add r1, r1, #1

b fib

**endfib:**

**main:**

// N = &r0;

// i = &r1;

// a = &r4;

// b = &r2;

// c = &r3;

// a = 0;

// b = 1;

// for(int i = 0; i < N; i++)

// c = a + b;

// a = b;

// b = c;

mov r0, #9

mov r1, #0

mov r4, #0

mov r2, #1

b fib

b main