

Lab 3 : Requirement Description

- 連結

- [說明文件](#)
- [影片](#)

- 基礎題 (70%)

- 題目敘述

請利用 **bitwise 指令**(NOT、AND、OR、XOR、ROTATE)實作算術移位和邏輯移位，對 **TRISA** 先進行一次邏輯左移後再進行一次算術右移

- 範例與測資

 邏輯左移 算術右移
01011111 → 10111110 → 11011111
 0x5F 0xBE 0xDF

- 評分標準

1. 會檢查是否使用至少一個 **bitwise 指令**
2. 請勿使用 **branch** 相關指令
3. 違反評分標準將斟酌扣分

- 進階題 (30%)

- 題目敘述

請實作 **16-bits*16-bits 的乘法運算** (測資計算結果必為正數且不會 overflow)

- 範例與測資

12CB * 0935 = AD0707 (hex)

address	00	01	02	03
000	12	CB	--	--
010	09	35	--	--
020	00	AD	07	07

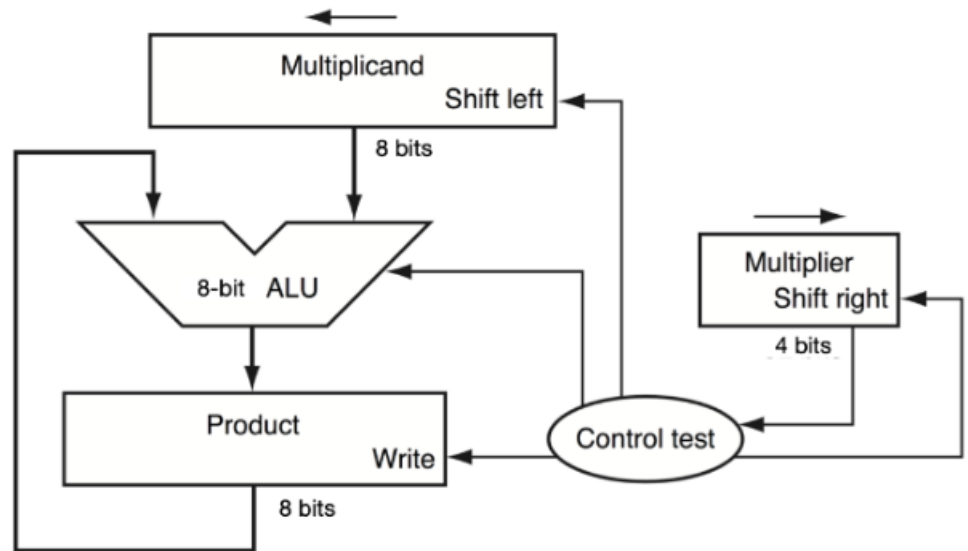
- 評分標準

1. 第一個數放在 0x000-0x001，第二個數放在 0x010-0x011，結果放在 0x020-0x023

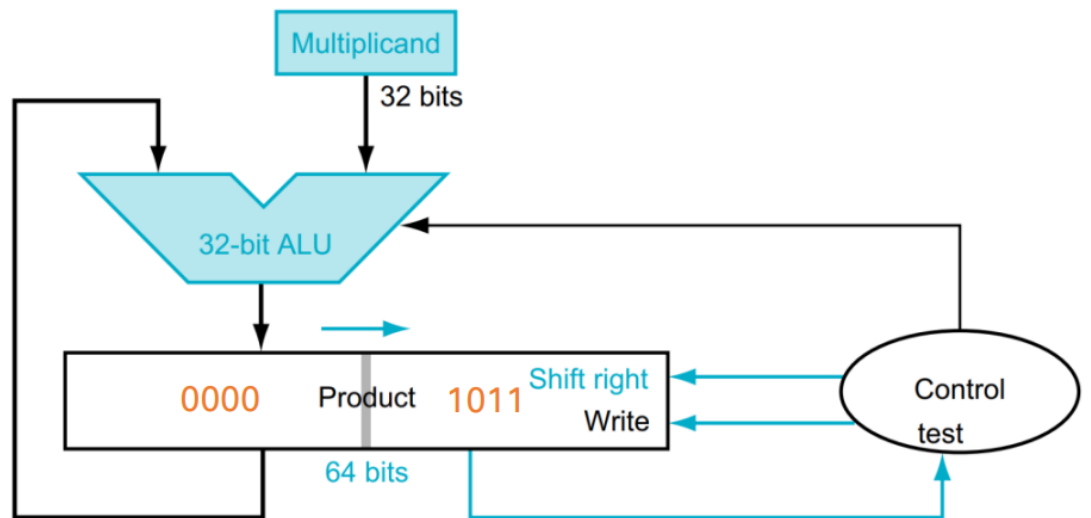
2. 無任何的指令使用限制，但推薦各位同學練習乘法器的使用(或自行額外練習)
3. 違反評分標準將斟酌扣分

■ 提示

1. Simple multiplier



2. optimized multiplier



3. 詳細步驟可以參考計算機組織的內容

● 加分題 (20%)

■ 題目敘述

給一個 16-bits 的數 x ， $x \geq 2$ ，計算 $\lceil \log_2(x) \rceil$ ($\log_2(x)$ 上取整數)

■ 範例與測資

1. 0040(hex) => 6
2. 0041(hex) => 7
3. 2A41(hex) => E

■ 評分標準

1. x 放在 0x000-0x001，結果放在 0x002
2. 請勿使用 **MUL** 相關指令
3. 請勿使用連續加法來做乘法
4. 違反以上評分標準將斟酌扣分
5. 請勿以直接對應等違規方式解題（範例如下），否則**此題沒分**

Ex: 禁止使用連續 if else

```
int x = 65;
int ans;
if (x <= 2) {
    ans = 1;
} else if (x <= 4) {
    ans = 2;
} else if (x <= 8) {
    ans = 3;
} else if (x <= 16) {
    ...
} ...
```

也就是說，不應該直接讀取一個數字後，直接將他存入答案位置

```
MOVLW 0x05
MOVWF 0x02
```

Lab 3 : Requirement Description

- Link

- [Document](#)

- [Video](#)

- Basic (70%)

- Description

Use **bitwise instructions** (NOT, AND, OR, XOR, and ROTATE) to implement both arithmetic and logical shifts. Begin with a logical left shift on the **TRISA** register, then perform an arithmetic right shift.

- Example

logical left shift arithmetic right shift

01011111 → 10111110 → 11011111

0x5F 0xBE 0xDF

- Standard of grading

1. Use **at least one bitwise instruction** to implement it.
2. DO NOT use the branch instruction.
3. Points will be deducted for any violation.

- Advance (30%)

- Description

Please implement a **16-bit by 16-bit multiplication operation** (the result must be positive and will not cause overflow).

- Example

12CB * 0935 = AD0707 (hex)

address	00	01	02	03
000	12	CB	--	--
010	09	35	--	--
020	00	AD	07	07

- Standard of grading

1. Store the first number in the memory addresses 0x000 to 0x001, second number in 0x010 to 0x011, and the result in the 0x020 to 0x021.
2. There are no restrictions on the use of instructions, but students are encouraged to use the multiplier (or practice additionally).

■ **Hint: multiplier architecture.**

● **Bonus (20%)**

■ **Description**

Give a 16-bits number x , $x \geq 2$, calculate $\lceil \log_2(x) \rceil$ (ceil of $\log_2(x)$)

■ **Example**

1. 0040(hex) => 6
2. 0041(hex) => 7
3. 2A41(hex) => E

■ **Standard of grading**

1. Store x in the memory addresses 0x000 to 0x001, and store the result in 0x002.
2. DO NOT use any **MULXX** instruction.
3. DO NOT use multiple additions to simulate multiplication.
4. Points will be deducted for any violation.
5. DO NOT solve the problem by hard-coding (as shown in the examples below); otherwise, you will not receive any points for this question.

Ex: Do not use if-elif-else.

In other words, you should not read a number and store it directly in 0x002.

```
int x = 65;
int ans;
if (x <= 2) {
    ans = 1;
} else if (x <= 4) {
    ans = 2;
} else if (x <= 8) {
    ans = 3;
} else if (x <= 16) {
    ...
} ...
```

MOVLW 0x05
MOVWF 0x02