計算機組織

實驗四

結果報告

第十二組

組員:

|  |  |
| --- | --- |
| 學號 | 姓名 |
| E24099059 | 陳旭祺 |
| F74044046 | 蕭佑永 |

日期:2020/10/26

一、實驗內容

(1)實作一

1.題目:實現RISC-V CPU的Interrupt Handler與ISR(Interrupt Service Routine)

2.實現方式(組合語言):

流程圖:

參考指令:

|  |  |  |
| --- | --- | --- |
| 指令 | csrr | csrw |
| 意義 | 讀取CSR(control and status register) | 寫入CSR |
| 範例 | 將mepc(CSR mepc地址=833)的值讀入x31 register中 | mepc的值寫入x31 register  一張含有 儀錶, 時鐘 的圖片  自動產生的描述 |

第1部分Interrupt Handler

addi sp, sp, -4 //空出一個4-byte stack的空間

sw ra, 0(sp) //將ra存起來

addi sp, sp, -12 //再空出一個12-byte stack的空間

csrr x31, 833 //mepc=833(0x341)

sw x31, 0(sp) //儲存mepc的值(=833)

csrr x31, 768 //mstatus=768

sw x31, 4(sp) //儲存mstatus的值(=768)

csrr x31, 834 //mcause=834

sw x31, 8(sp) //儲存mcause的值(=834)

andi x31, x31, 3 //弄一11的mask與mcause進行AND，保留右邊兩個bit(unchange)，其他變為0

addi x31, x31, 1197

jalr x31 //依據 mcause算出要跳入的ISR地址

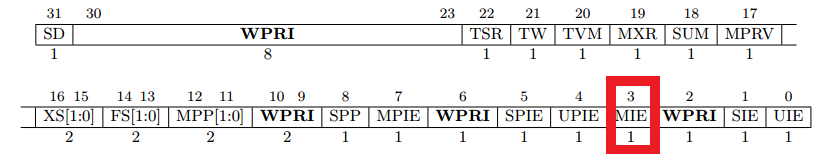
//========================ISR==============================

lw x31, 8(sp) //mcause從memory還原到暫存器中

csrw 834, x31

lw x31, 4(sp) //mstatus從memory還原到暫存器中

ori x31, x31, 8 //弄一1000的mask與mstatus進行OR，在從右數來第四個bit無論是多少都會變成1，因此變成enable interrupt



csrw 768, x31

lw x31, 0(sp)

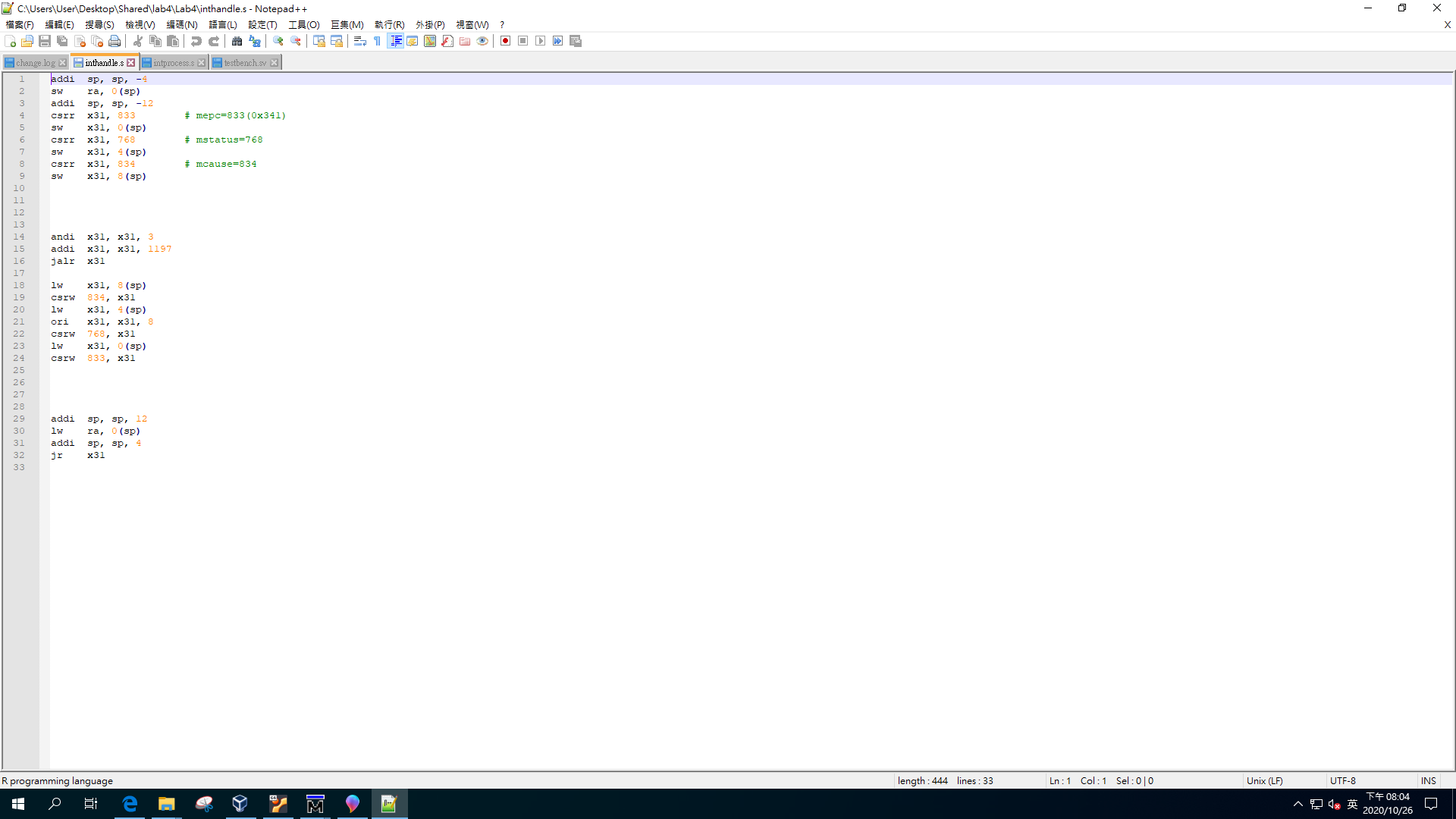
csrw 833, x31 // mepc從memory還原到暫存器中

addi sp, sp, 12 //釋出一個12-byte stack的空間

lw ra, 0(sp)

addi sp, sp, 4 //釋出一個4-byte stack的空間

jr x31 //最後跳回mepc位置



第2部分ISR(Interrupt Service Routine)

addi sp, sp, -128 //空出一個128-byte stack的空間

//stack push(last in first out)

sw x1, 4(sp)

sw x2, 8(sp)

sw x3, 12(sp)

sw x4, 16(sp)

sw x5, 20(sp)

sw x6, 24(sp)

sw x7, 28(sp)

sw x8, 32(sp)

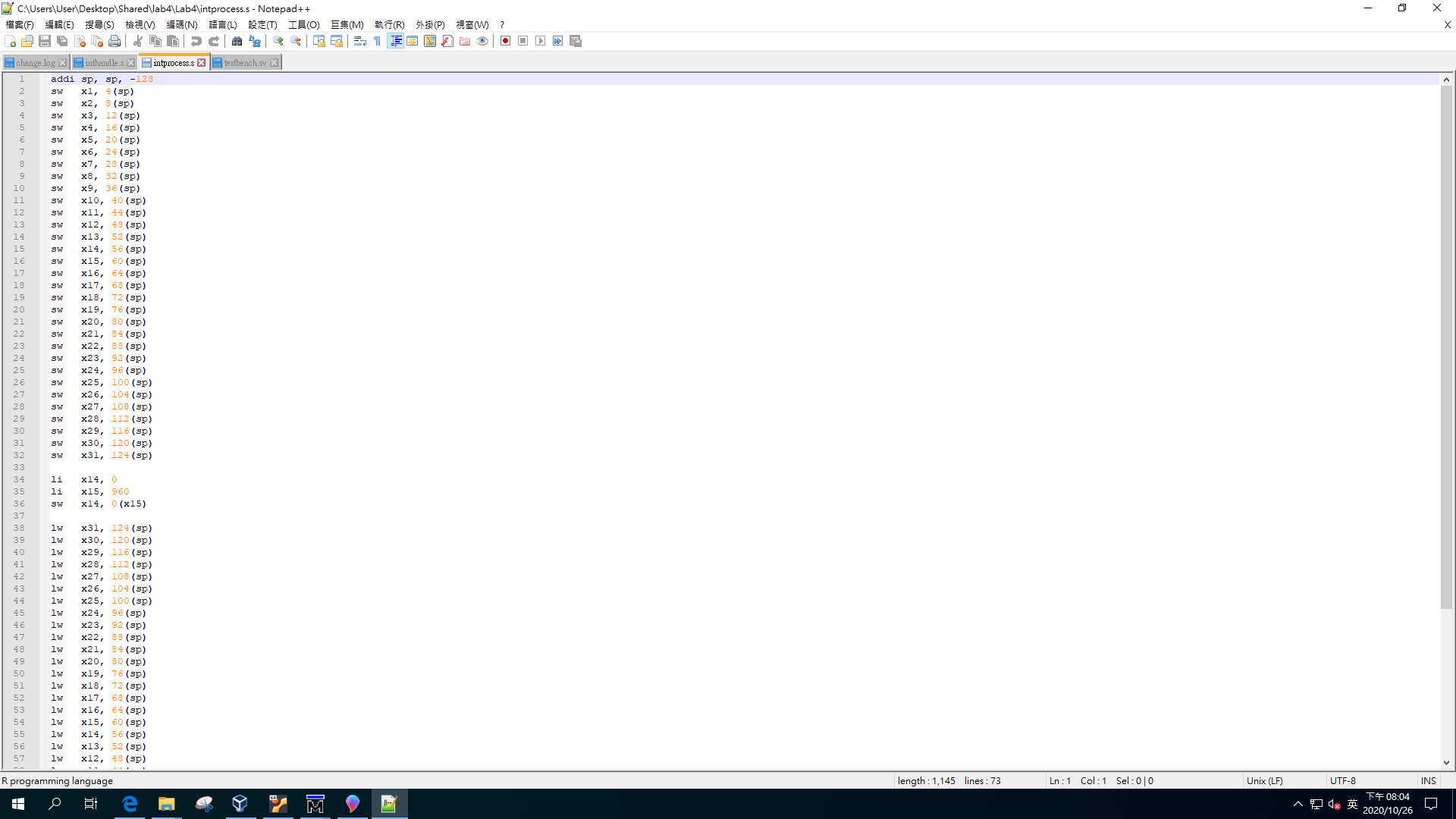
sw x9, 36(sp)

sw x10, 40(sp)

sw x11, 44(sp)

sw x12, 48(sp)

sw x13, 52(sp)

sw x14, 56(sp)

sw x15, 60(sp)

sw x16, 64(sp)

sw x17, 68(sp)

sw x18, 72(sp)

sw x19, 76(sp)

sw x20, 80(sp)

sw x21, 84(sp)

sw x22, 88(sp)

sw x23, 92(sp)

sw x24, 96(sp)

sw x25, 100(sp)

sw x26, 104(sp)

sw x27, 108(sp)

sw x28, 112(sp)

sw x29, 116(sp)

sw x30, 120(sp)

sw x31, 124(sp)

li x14, 0

li x15, 960

sw x14, 0(x15)

//把0存到記憶體(位置960)

//stack pop

(last in first out)

lw x31, 124(sp)

lw x30, 120(sp)

lw x29, 116(sp)

lw x28, 112(sp)

一張含有 文字 的圖片

自動產生的描述lw x27, 108(sp)

lw x26, 104(sp)

lw x25, 100(sp)

lw x24, 96(sp)

lw x23, 92(sp)

lw x22, 88(sp)

lw x21, 84(sp)

lw x20, 80(sp)

lw x19, 76(sp)

lw x18, 72(sp)

lw x17, 68(sp)

lw x16, 64(sp)

lw x15, 60(sp)

lw x14, 56(sp)

lw x13, 52(sp)

lw x12, 48(sp)

lw x11, 44(sp)

lw x10, 40(sp)

lw x9, 36(sp)

lw x8, 32(sp)

lw x7, 28(sp)

lw x6, 24(sp)

lw x5, 20(sp)

lw x4, 16(sp)

lw x3, 12(sp)

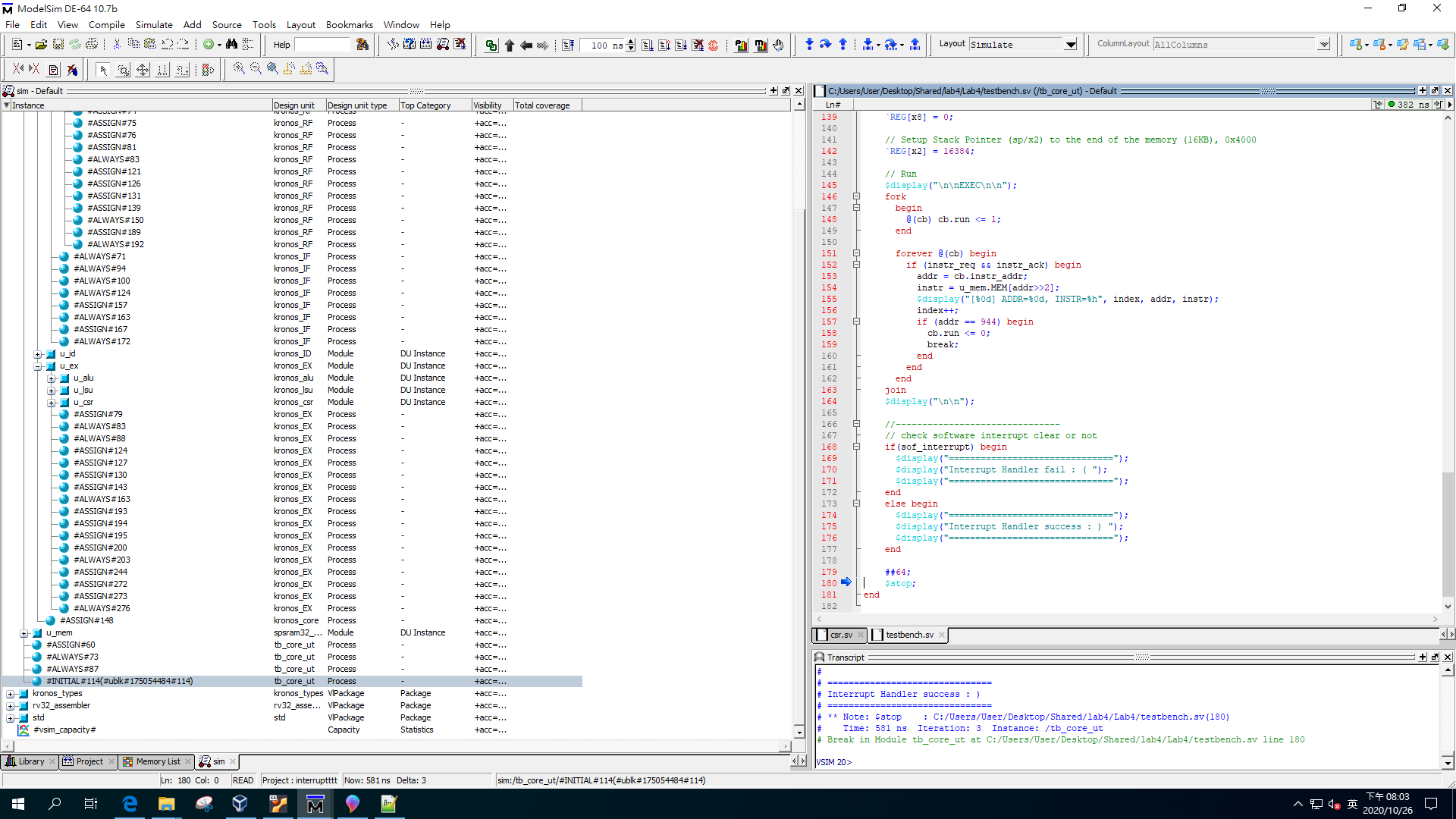
lw x2, 8(sp)

lw x1, 4(sp)

addi sp, sp, 128 //釋出一個128-byte stack的空間

jr ra //跳回 Handler

3.結果分析(modelsim):



下圖為中斷前

一張含有 坐, 監視器, 電腦, 桌 的圖片

自動產生的描述

下圖為中斷後

一張含有 坐, 監視器, 電腦, 大 的圖片

自動產生的描述

下圖之黃線為區隔中斷前後

一張含有 電子用品, 電腦, 監視器, 膝上型電腦 的圖片

自動產生的描述

二、實驗心得(心得可隨意發揮，但是內容會做工程認證保留檔案)

陳旭祺:

本次實驗主要是讓我們認識當RISC-V遇到Interrupt(or Internal (Exception), External)時，Interrupt Handler與Interrupt Service Routine執行、處理的步驟。計組期中考後我們這組有2個成員退選，這次實驗簡單，難保下次做得出來，因此希望能趁現在比較沒考試，閒暇之餘多精進自己，在之後的實習課和期末Project能呈現自己最好的一面。

蕭佑永:

這次實驗讓我更加了解當RISC-V碰到中斷時是如何處理的，也稍微運用到了mask，最後也期許自己更加精進以面對project!