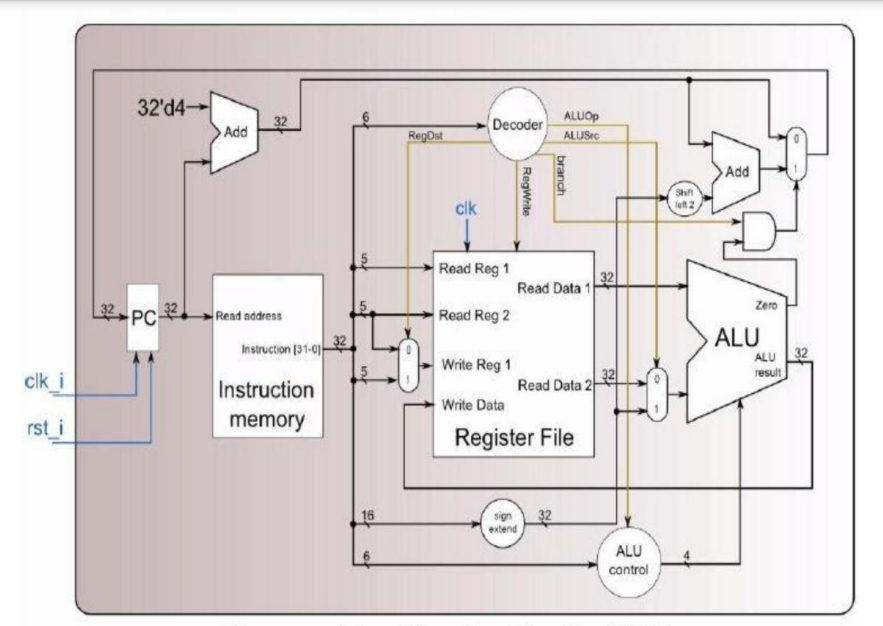
**Computer Organization Lab2**

**Name:林秉**

**ID:109550112**

**Architecture diagrams:**



**Hardware module analysis:**

**Adder:**

一張含有 文字 的圖片

自動產生的描述

**We simply add result and return to sum.**

**ALU\_ctrl:**

一張含有 文字 的圖片

自動產生的描述

**We followed the function field in pdf to give 6 instructions.**

一張含有 文字 的圖片

自動產生的描述

**Then we checked the table of ALU\_OP to assign the ALU\_ctrl output by the following graph. When ALU\_op[2] = 0, means it’s R\_format(Which you’ll see in decoder), thus we assign value for R\_format instructions here. For rest I\_format instructions, we assign values one-by-one. The ALUCtrl\_o[3] would remain 0 since there’s no need in this lab.**

一張含有 桌 的圖片

自動產生的描述

**ALU:**

一張含有 文字 的圖片

自動產生的描述

**Zero = 1 iff result is 0. Then again, we follow the ALUctrl instructions with given values by the above graph.**

**Decoder:**

一張含有 文字 的圖片

自動產生的描述

一張含有 時鐘, 美工圖案 的圖片

自動產生的描述

**We assign four wires for specified instructions. When r = 1, means the instruction is R-format, else it’s I-format. And with checking the instruction table, we can assign correct value for the decoder, especially for ALU\_op, since it explains the specific condition segment for ALU\_ctrl.**

**MUX:**

一張含有 文字 的圖片

自動產生的描述

**When select\_i is 1 then output data 1, likewise for select\_i = 0.**

**Shift\_Left\_Two\_32:**一張含有 文字 的圖片

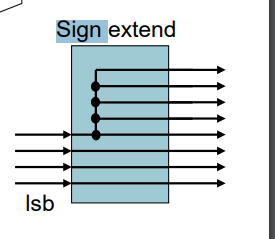
自動產生的描述

**We simply shift 2 bits.**

**Sign\_Extend:**

一張含有 文字 的圖片

自動產生的描述



**We extend the output bits by the above diagram.**

**Simple\_Single\_CPU:**

一張含有 文字 的圖片

自動產生的描述

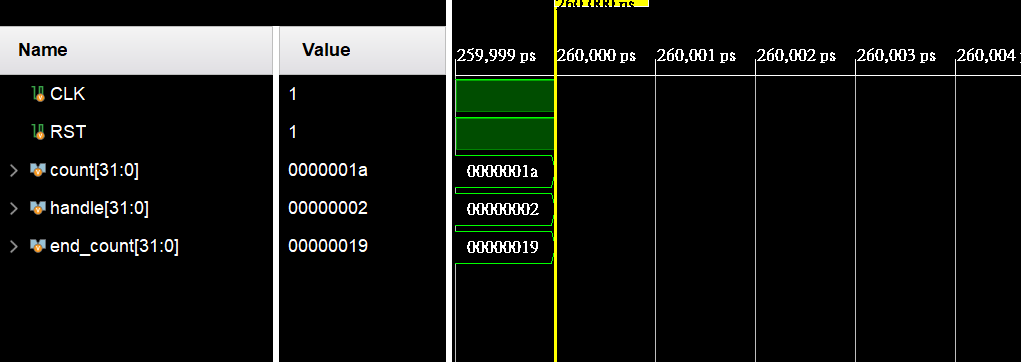
**We assign enough different wires with different bits count to allocate to different modules. And the result would be assigned by the given architecture diagram above.**

**Finished part:**

**Case1:**

一張含有 桌 的圖片

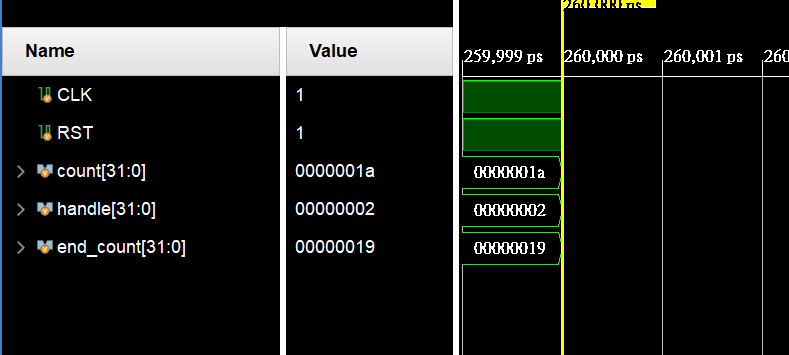
自動產生的描述



**Case2:**

一張含有 桌 的圖片

自動產生的描述



**Problems you met and solutions:**

**Since this lab includes lots of variables calculations, when I was debugging, I find myself often using variables with typo, and the program can still run with undeclared variables. This has confused me a lot. And the ALU\_operations are a little hard to understand at first. But with lots of careful double checks, I eventually solved the problems.**

**Summary:**

**This lab is more messy comparing to lab1, there are a lot of details needed to pay attention, I think I need to be careful on every tasks and focus myself for the next challenge.**