**Computer Organization**

Student ID: 0516072 Name:洪立宇

HDL simulator you used:

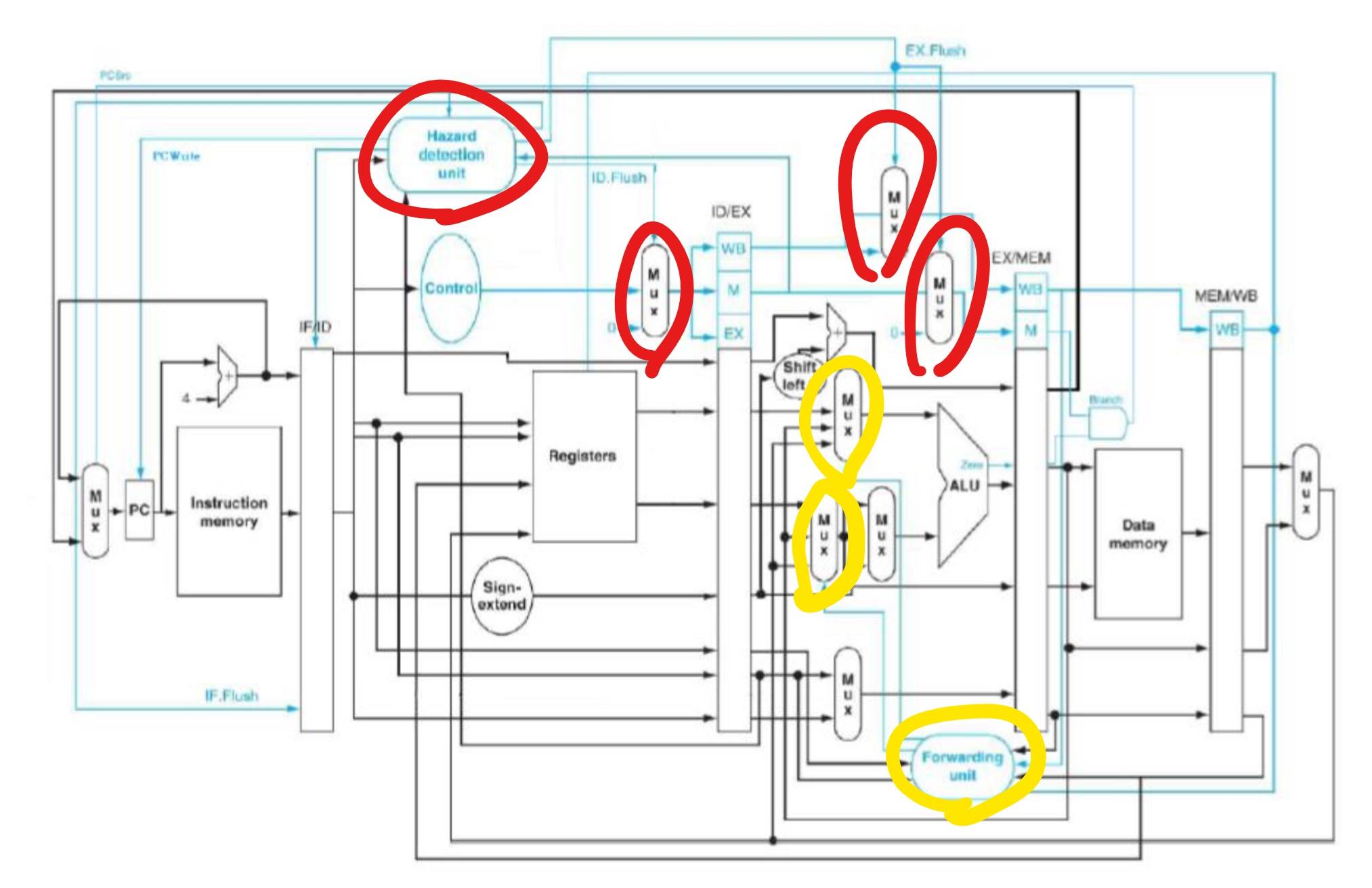
Modelsim

Finished part:

Finish the implementation of Hazard Detection and Forwarding and stall pipeline CPU if it detects load-use situation. Also I have done the bonus which is to flush the pipeline registers and give the correct instruction from the right PC value.

Architecture diagrams:

(Please write down or plot on the architecture diagram to show what did you do to improve in this lab.)



本圖示依照檔案給的圖下去標記，紅色地方為 Hazard，黃色地方為forwarding.

Hardware module analysis:

這次比上次還多的地方是

HazardDetectionUnit.v: 是用來兂測有沒有發生hazard ，然後發生hazard時要 stall 或是flush 的判斷。

Forwarding.v:用來決定當遇到forwarding的時候電路要跑去哪裡。

Explain how your **Forwarding.v** and **Hazard\_detection\_unit.v** to detect and set the signals to handle the hazard problems

(both test patterns CO\_P6\_test\_data1 and CO\_P6\_test\_data2 are needed)

Ex. The table is just an example format to answer the questions, and there is maybe more or less hazard parts.

CO\_P6\_test\_data1

|  |  |  |
| --- | --- | --- |
| Data/Control Hazard part | Forwarding | Hazard detection |
| lw $5, 4($0)  and $8, $5, $3  $5 發生data hazard | Forwarding A：10  Forwarding B：00 | IF\_flush: 1  ID\_flush: 0  EX\_flush: 0  PC\_write: 0  IF\_stall: 0 |

CO\_P6\_test\_data2

|  |  |  |
| --- | --- | --- |
| Data/Control Hazard part | Forwarding | Hazard detection |
| beq $1, $2, L1  control hazard | Forwarding A：x  Forwarding B：x | IF\_flush: 0  ID\_flush: 1  EX\_flush: 1  PC\_write: 1  IF\_stall: 1 |

Problems you met and solutions:

在這次的lab其實多要寫的東西並不多，只是要釐清data hazard 還有control hazard 時CPU 要進行 flush 或stall 的整個線路接法，還有線路forward之後會跑哪一條，其中也有很多小細節，讓在寫的時候一直出錯，要重新順過一次整個電路的邏輯才有辦法，慢慢找到錯的地方，真的很耗腦，一直在debug找腦洞。

Summary:

過了一個學期終於寫出了一個有pipeline又可以處理data hazard的CPU 感覺超猛的。