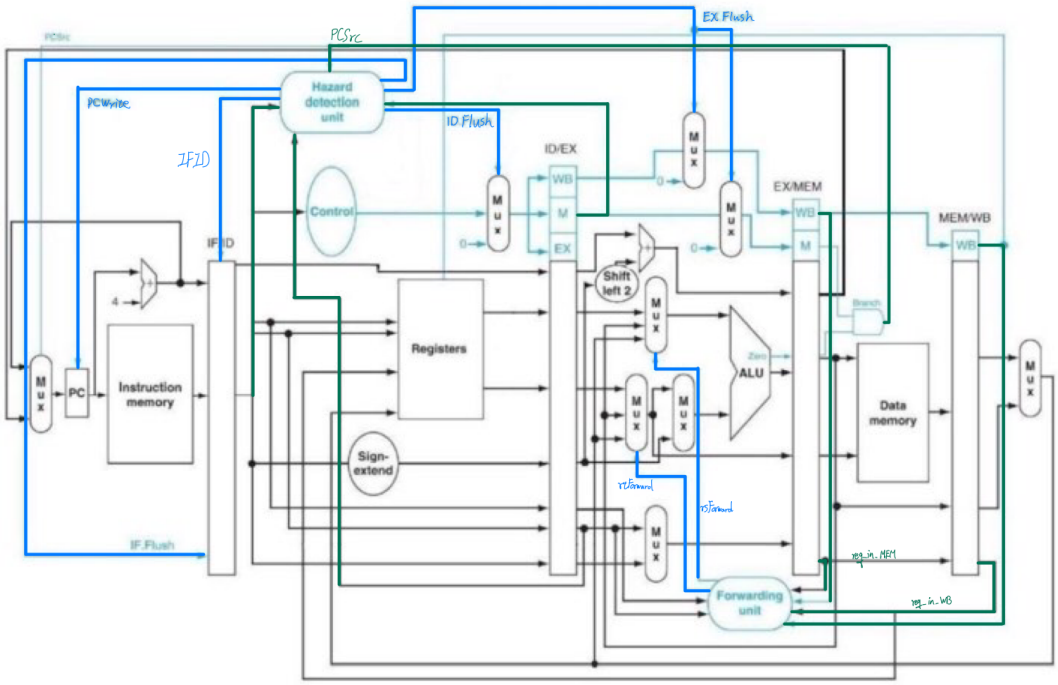
Computer Organization

Architecture diagrams:



上圖是此次作業的架構，因為大部分線路與Lab4差不多，所以只標了forwarding unit及hazard detection的線路。

Hardware module analysis:

這次Lab要用Forwarding解決Hazard的問題，首先是Rtype，如同下圖講義所述：

- EX hazard
  - if (EX/MEM.RegWrite and (EX/MEM.RegisterRd ≠ 0)
    and (EX/MEM.RegisterRd = ID/EX.RegisterRs))
    ForwardA = 10
  - if (EX/MEM.RegWrite and (EX/MEM.RegisterRd ≠ 0)
    and (EX/MEM.RegisterRd = ID/EX.RegisterRt))
    ForwardB = 10
- MEM hazard
  - if (MEM/WB.RegWrite and (MEM/WB.RegisterRd ≠ 0)
    and (MEM/WB.RegisterRd = ID/EX.RegisterRs))
    ForwardA = 01
  - if (MEM/WB.RegWrite and (MEM/WB.RegisterRd ≠ 0)
    and (MEM/WB.RegisterRd = ID/EX.RegisterRt))
    ForwardB = 01

程式方面則以下圖實現：

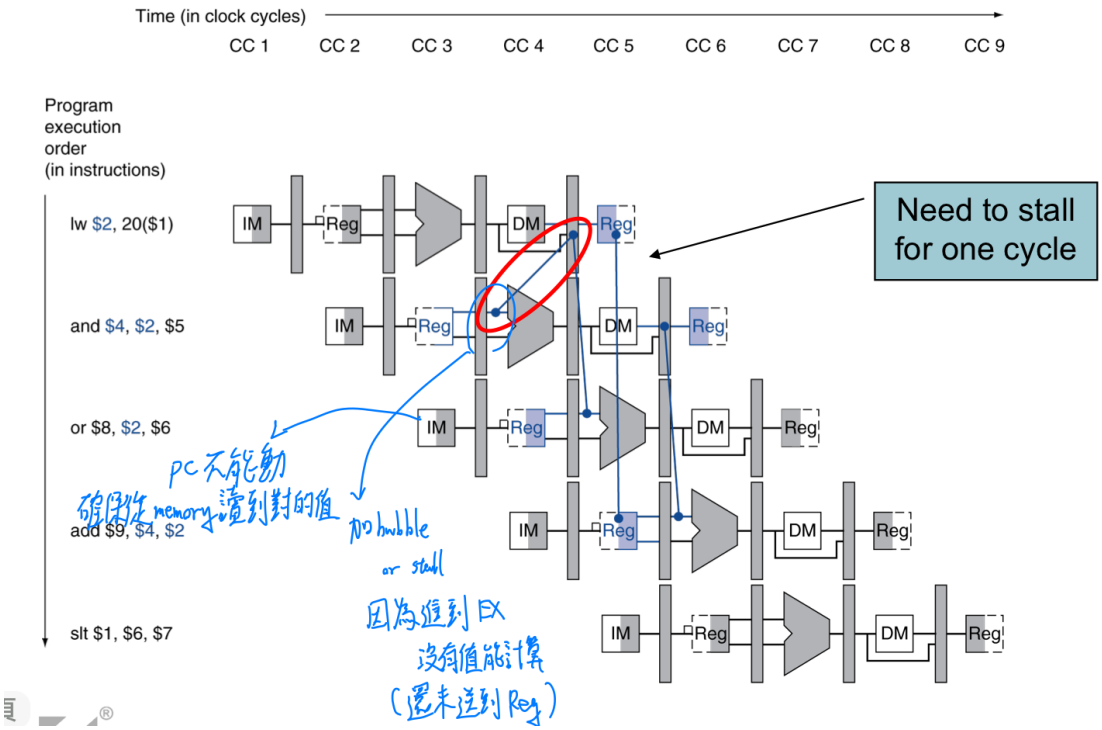
```
if (EXMEM_WB[1] && EXMEM_rd != 0 && (EXMEM_rd == IDEX_rs)) begin
    rsForward <= 2'b01;
end

if (EXMEM_WB[1] && EXMEM_rd != 0 && (EXMEM_rd == IDEX_rt)) begin
    rtForward <= 2'b01;
end

if (MEMWB_WB[1] && MEMWB_rd != 0 && (MEMWB_rd == IDEX_rs)) begin
    rsForward <= 2'b10;
end

if (MEMWB_WB[1] && MEMWB_rd != 0 && (MEMWB_rd == IDEX_rt)) begin
    rtForward <= 2'b10;
end
```

至於forwarding的部分則是用3to1MUX實作。再來是load-use hazard，如下圖所示，即使有forwarding，pipeline還是需要經過一個stall



而最後的branch hazard則是必須等到結果出來才會確定需不需要跳，所有hazard detection用下面方式實作：

```
if (PCSrc) begin
    IFFlush <= 1;       //branch
    IDFlush <= 1;
    EXFlush <= 1;
    PCWrite <= 1;
    IFIDWrite <= 1;
end

else if ((IFID_rs == IDEX_rt || IFID_rt == IDEX_rt) && MEM[1]) begin
    IFFlush <= 0;       //load-use
    IDFlush <= 0;
    EXFlush <= 0;
    PCWrite <= 0;       //stall
    IFIDWrite <= 0;
end

else begin
    IFFlush <= 0;       //Rtype
    IDFlush <= 0;
    EXFlush <= 0;
    PCWrite <= 1;       //no stall
    IFIDWrite <= 1;
end
```

Problem You Met and Solutions:

這次功課最大的問題是，在一開始接完線並且可以編譯後，我的r1到r9全部顯示x，看的我都傻了，於是叫出gtkwave看看哪錯，發現RegFile的輸出RSdata和RTdata都是x，於是從輸入下手，發現resultWB傳出來的也是x，反正就循線找了ALU和DM的輸入輸出，那波型圖的x實在有點搞我心態，我還一度懷疑我是不是手賤改到RF和DM，結果最後才發現是3to1MUX有時候輸出結果會變成x，而後加了default才終於解決，這花了我一整個晚上....

Result:

```
Register=======================================================

r0=         0, r1=        16, r2=       256, r3=         8, r4=        16, r5=         8, r6=        24, r7=        26

r8=         8, r9=         1, r10=          0, r11=          0, r12=          0, r13=          0, r14=          0, r15=          0

r16=          0, r17=          0, r18=          0, r19=          0, r20=          0, r21=          0, r22=          0, r23=          0

r24=          0, r25=          0, r26=          0, r27=          0, r28=          0, r29=          0, r30=          0, r31=          0


Memory=========================================================

m0=         0, m1=        16, m2=         0, m3=         0, m4=         0, m5=         0, m6=         0, m7=         0

m8=         0, m9=         0, m10=          0, m11=          0, m12=          0, m13=          0, m14=          0, m15=          0

r16=          0, m17=          0, m18=          0, m19=          0, m20=          0, m21=          0, m22=          0, m23=          0

m24=          0, m25=          0, m26=          0, m27=          0, m28=          0, m29=          0, m30=          0, m31=          0
** VVP Stop(0) **
** Flushing output streams.
** Current simulation time is 210000 ticks.
```

Summary:

終於寫完最後一次Lab，又爽又開心，可是期中期末實在慘不忍睹，雖然覺得在這堂課我蠻認真，也試著搞懂遇到的問題，但考卷擺在眼前就暈，真的希望不要再一年嗚嗚嗚嗚嗚