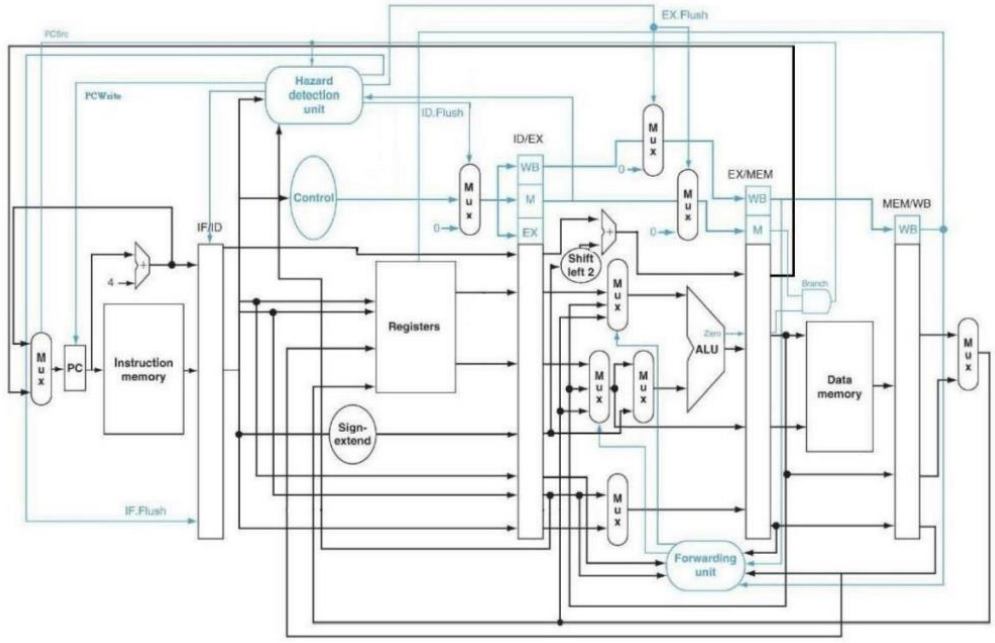
Computer Organization Lab5

Name:劉彥甫

ID:109550104

Architecture diagrams:

The architecture I used in this homework is same as the one in spec. The picture is below.



Hardware module analysis:

The difference between this module and the one in homework 4 is forwarding unit and hazard detection unit. If it detects data hazard, we can use forwarding wire to avoid error. And if the branch prediction is wrong, we can flush some register to get the correct result.

The upside of this design is it doesn't need to do unnecessary stall. When the data hazard occurs, it only needs to stall if the hazard is load-used hazard. It improves the CPI. However, the downside of this design is it's not efficiency enough. If we move the branch comparison from EX stage to ID stage, it can be more powerful.

Finished part:

The result of test1 and test2 are below. We can see the results are same as the one given by TA.

Now, we analyze test1. Follow the instruction set in test1, we can see some value changes. We can get r1 = 16, r2 = 256, r3 = 8, m1 = 16, r4 = 16, r5 = r4-r3 = 8, r6 = r3+r1 = 24, r7 = r1+10 = 26, r8 = r7&r3 = 8, r9 = 1 (r8<r7)

The answer is same as the one below.

```
################################ clk_count = 60#################################
=====================================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

m16=    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
```

```
################################ clk_count = 60#################################
=====================================Register=====================================
r0 =    0, r1 =    0, r2 =   12, r3 =    6, r4 =    0, r5 =   16, r6 =    0, r7 =    0

r8 =    2, r9 =    0, 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 =    4, m1 =    1, m2 =    0, m3 =    6, m4 =    0, m5 =    0, m6 =    0, m7 =    0

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

m16=    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
```

Problems you met and solutions:

The problem I met in this homework is we need to think about how to implement branch flush and which register need to be flush. It's an interesting task.

Summary:

After this homework, we finally made the best version of the single cycle CPU so far. But we all know it's not the best one. Maybe in the future course, we can improve our design.