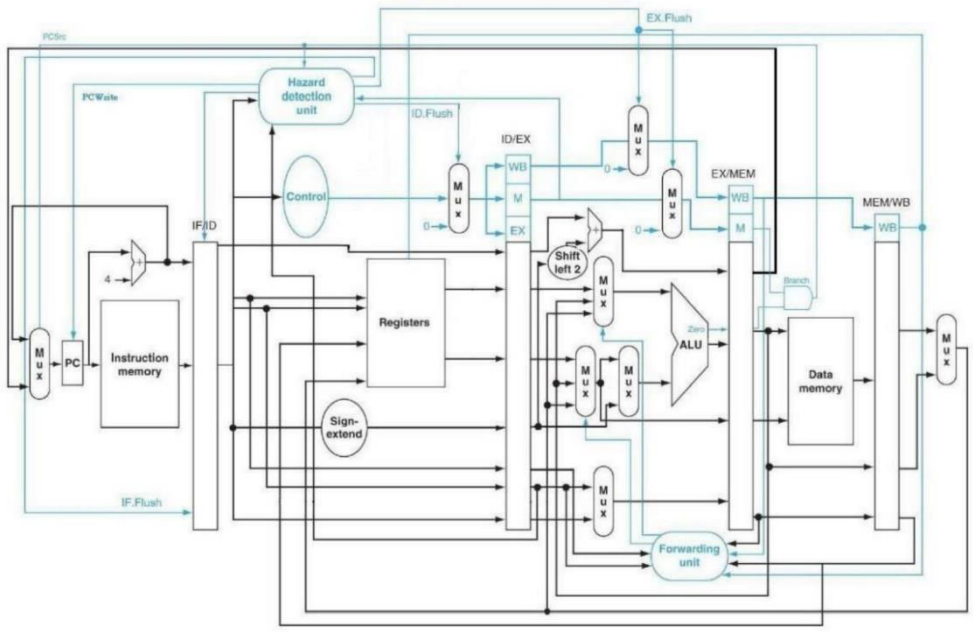
Computer Organization Lab5

Name: 房天越

ID: 110550085

Architecture diagrams:



Compared to Lab4, hazard detection unit and forwarding unit are added to handle the possible hazards.

Hardware module analysis:

Two modules are added in this Lab, which are Hazard.v and Forward.v:

Hazard.v:

The hazard detection unit that could flush the registers when branch or hazard happens, also it can determine whether to keep the PC or let it be the next command.

Forward.v:

When the situations that need forwarding generate, it can send ForwardA and ForwardB to do forwarding, with this module we do not need too many NOPs and

can reduce the number of clocks.

Also, since bgt, bge, and bne are needed to be used in this Lab, Branchtype is added back to determine when branch should be executed.

And, flush and write are added to Pipe_Reg, they can determine when to keep the value or clear all the values when hazards are generated.

Mux_3to1.v and Mux_4to1.v are also included to fulfill the requirements.

Finished part:

Data1:

```
################################## clk_count = 16####################################
=====================================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
```

Data2:

```
################################## clk_count = 64####################################
====================================Register====================================
r0 =    0, r1 =    0, r2 =   16, 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
```

Both results correspond to the given answer, and both get the correct results with clock usage that are below the limit.

Problems you met and solutions:

The debugging part for this Lab is also exhausting, there are a lot of values that are required to determine, I must be very concentrated to avoid mistakes. I failed, so I searched for like three hours just to find that I had a typo of writing 1 into 0.
Also, since many types of branches are required to be implemented in this Lab, I spent some time realizing how I should deal with them.

Summary:

After this Lab, I now have a better comprehension of the ways to deal with the hazards. Also, I learned when to do forwarding and how to do it.
After the labs in this semester, I think I now understand better about how CPU works and how to implement it, I hope that I can use this knowledge to deal with the following challenges in my life.