Lab Assignment #5

Teo Patrosio

EE-126/COMP-46: Computer Engineering w/lab

Professor: Mark Hempstead TA:Parnian Mokri

Tufts University, Fall 2021

**Introduction + Purpose**

In this lab, the hazard detection and forwarding modules were built on top of the last lab which was the pipelined cpu.

**Components built**

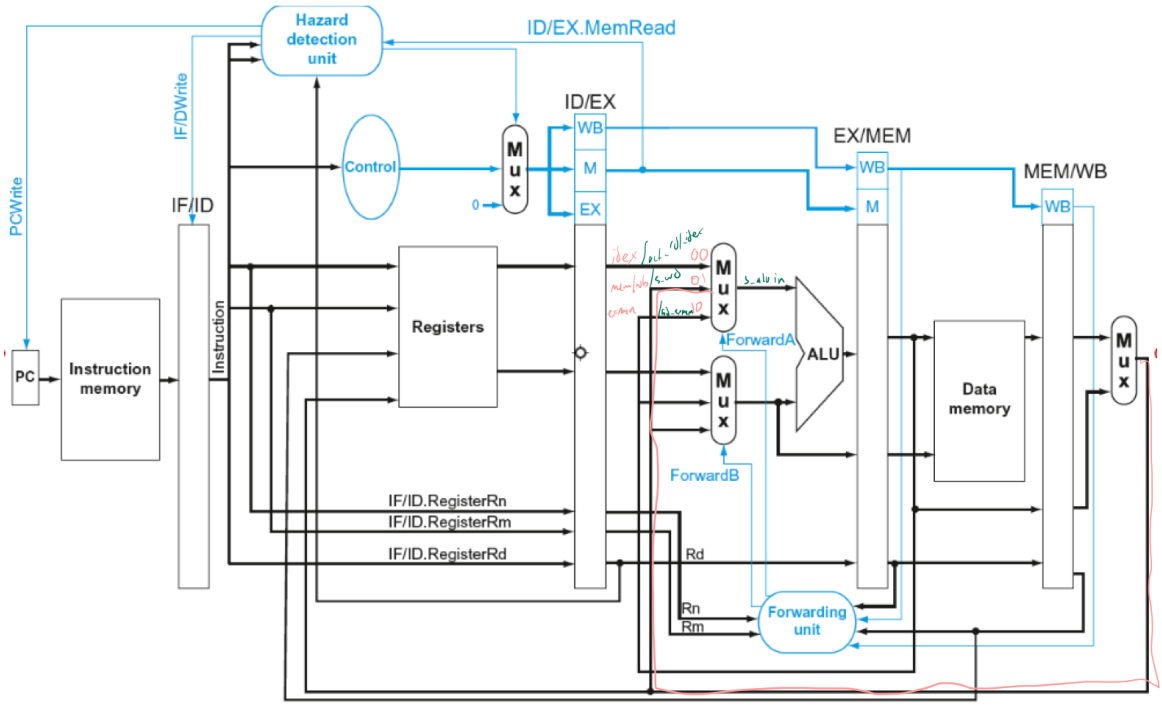


***Figure 1:*** *Overall system diagram of the Hazard detection and Forwarding Units*

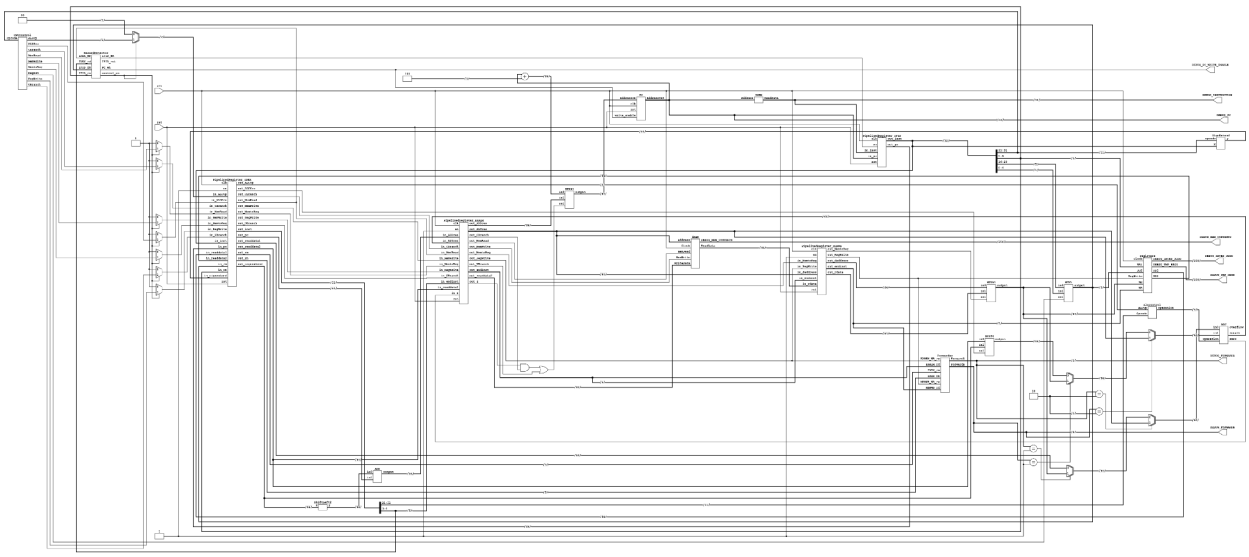

***Figure 2:*** *System block diagram* (Created by https://blog.eowyn.net/netlistsvg/)

```
vcom -quiet -2008 *.vhd tb/*.vhd imem/*.vhd
vsim -quiet -c work.pipelinedcpu1_tb -do "log -r /*; run 500ns; exit" -wlf
pipelinedcpu1.wlf &&  wlf2vcd pipelinedcpu1.wlf -o pipelinedcpu1.vcd
```

***Figure 3:*** *Commands to analyze and run the code*

| Cycle | PC | | | |
|---|---|---|---|---|
| 0 | 0 | LDUR X9, [XZR, 0] | 11111000010000000000001111101001 | f8 40 03 e9 |
| 1 | 4 | ADD X9, X9, X9 | 10001011000001001000000100101001 | 8b 09 01 29 |
| 2 | 8 | ADD X10, X9, X9 | 10001011000001001000000100101010 | 8b 09 01 2a |
| 3 | 8 | | | |
| 4 | 12 | SUB X11, X10, X9 | 11001011000001001000000101001011 | cb 09 01 4b |
| 5 | 16 | STUR X11, [XZR, 8] | 11111000000000001000001111101011 | f8 00 83 eb |
| 6 | 20 | STUR X11, [XZR, 12] | 11111000000000001100001111101011 | f8 00 c3 eb |
| 7 | 24 | | | |
| 8 | 28 | | | |
| | 32 | | | |

***Figure 4:*** *Test program in Assembly, Binary, and Hex along with the associated cycles and PC counter*

| Cycle | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | | |
|---|---|---|---|---|---|---|---|---|---|---|---|
| PC | 0 | 4 | 8 | 8 | 12 | 16 | 20 | 24 | 10 | 32 | 36 |
| LDUR X9, [XZR, 0] | IF | ID | EX | MEM | WB | | | | | | |
| ADD X9, X9, X9 | | IF | ID | ID* | EX | MEM | WB | | | | |
| ADD X10, X9, X9 | | | | IF | ID | EX | MEM | WB | | | |
| SUB X11, X10, X9 | | | | | IF | ID | EX | MEM | WB | | |
| STUR X11, [XZR, 8] | | | | | | IF | ID | EX | MEM | WB | |
| STUR X11, [XZR, 12] | | | | | | | IF | ID | EX | MEM | WB |

***Figure 5:*** *Test program and instructions with their associated pipeline phase for each clock cycle.*

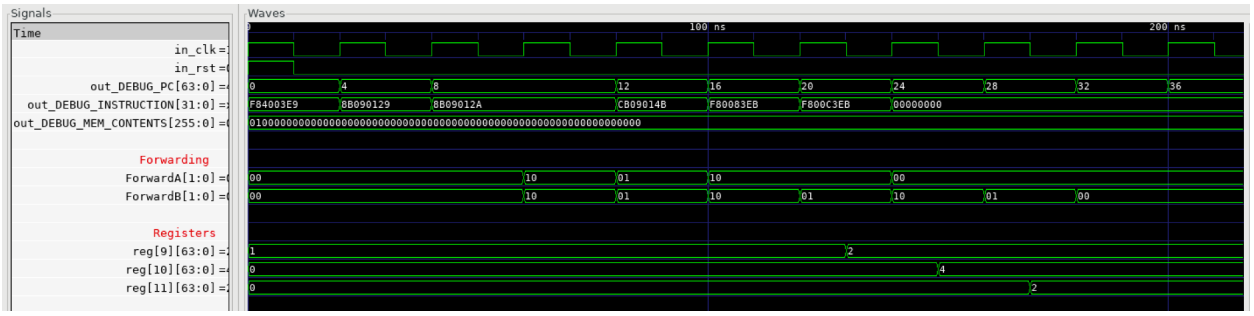


***Figure 6:*** *Output waveforms from test program on pipelinedcpu1.vhd*

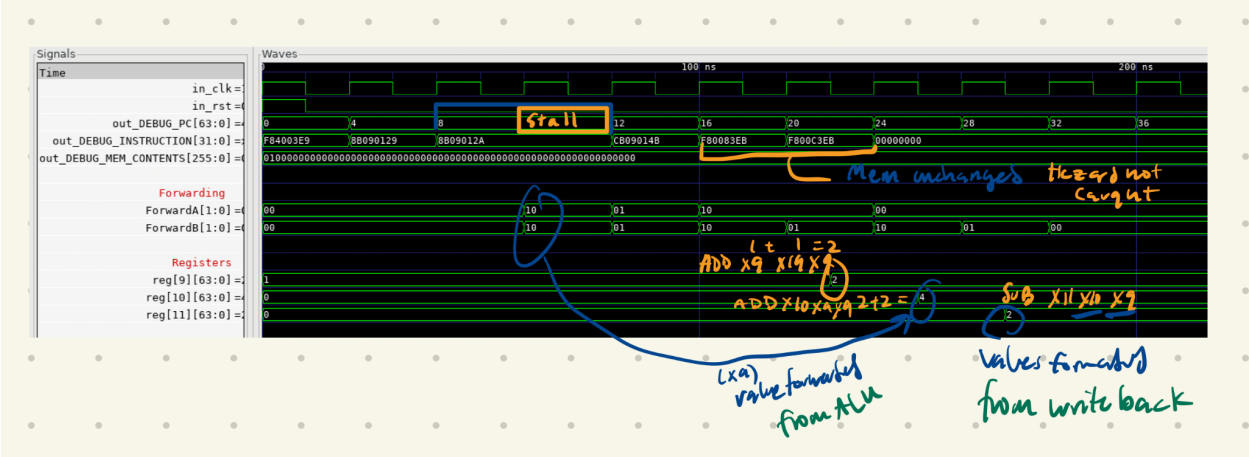


***Figure 7:*** *Annotated Output waveforms from test program on pipelinedcpu1.vhd*

As seen in Figure 7, there several main events;

- The stall at PC 8 at cycle 8 on the ADD X10, X9, X9/`8b 09 01 2a`
    - This stall occurs because X9 is being calculated by the instruction before.
- Otherwise events such as the Add and Sub with forwarding results directly from the output of ALU and the output of the MEM phase.
- A final thing to note is that data memory does not get updated fully as expected because there is a hazard that occurs that cannot be remedied by forwarding and the instructed hazard detection.