Lab Assignment #4

EE 126: Computer Engineering

Tufts Univerisity, Fall 2019

Name: Ruoxi Ren Student ID: 1327867 E-mail: Ruoxi.Ren@tufts.edu

1 Waveforms of Pipelined CPU without Hazard Detection

The CPU runs a MIPS program as below:

```
add $t2, $t0, $t1
sw $t2, 0x0($zero)
sub $t3, $t0, $t1
sw $t2, 0x0($zero)
sw $t3, 0x4($t3)
sw $t3, 0x4($t3)
or $s2, $s0, $s1
nop
nop
sw $s2, 0xC($zero)
nop
nop
nop
```

Before running the program, some registers are set as below:

Registers	Value
$t0	8
$t1	4
$t2	2
$t3	0
$s0	0xCEA4126C
$s1	0x1009AC83
$s2	0
$s3	0

And the memory is set as below:

Address	Value
DMEM(0x0)	0x00 00 00 01
DMEM(0x4)	0x00 00 00 02
DMEM(0x8)	0x00 00 00 03
DMEM(0xC)	0x00 00 00 04

The waveform of cycle 1 to cycle 5 is shown in Fig. 1.

In cycle 1, the pipeline is reset. The first instruction is fetched. Since there is a rest signal at the beginning of the cycle, the instruction in ID stage is nop. Besides, all the registers and memory are set to the initial value.

In cycle 2, the second instruction is fetched. The first instruction enters the ID stage, and the control signal is set according to the instruction. It is an add instruction so the $RegWrite$ signal is 1. The nop instruction is in EX stage.

In cycle 3, the third instruction is fetched. The second instruction enters the ID stage, which is a store instruction so the $MemWrite$ is 1. Since the first instruction with the control signal is still in the EX stage, the value of $t2 has not been written back, so the value of $t2 read by the second instruction is the wrong value.

In cycle 4, the fourth instruction is fetched. The third instruction is in the ID stage while the second instruction is in the EX stage. The first instruction is in the MEM stage.

In the fifth cycle, the fifth instruction is fetched. The first instruction is in WB stage, so the value of $RegWrite$ in WB is 1, and the value of $t2 changes from 2 to 0xC in the second half of the cycle. The second instruction enters the MEM stage while it is storing the value of the wrong value of $t2 to MEM(0). The third instruction is in the EX stage while the fourth instruction is in ID stage. Note that because the first instruction makes the value of $t2 right, so the fourth instruction reads the right value of $t2.

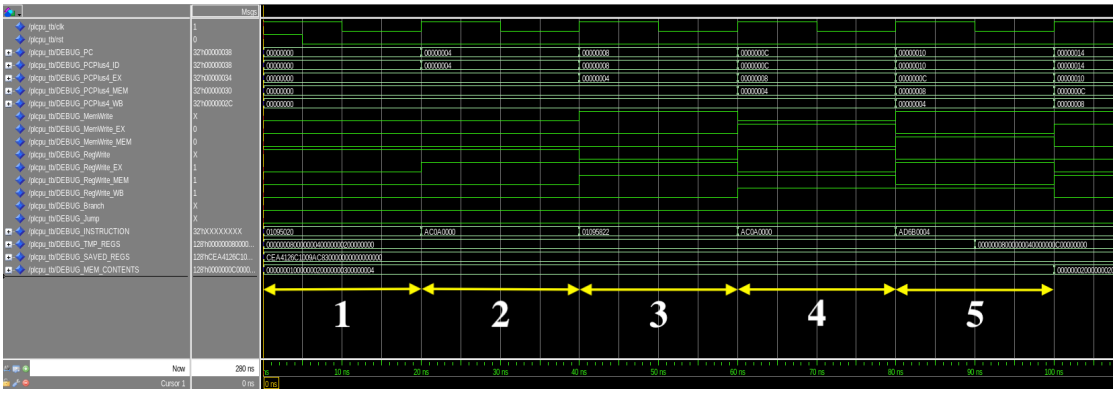

Figure 1: Simulation of pipelined CPU for cycle 1 to 5

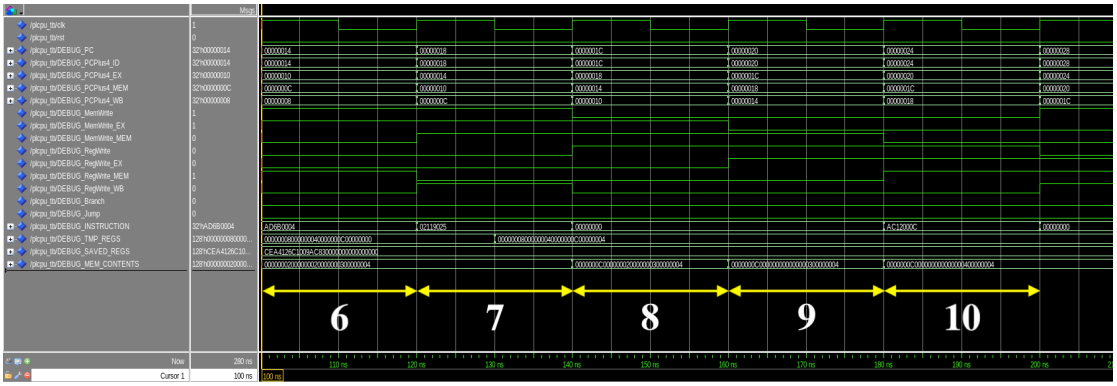

Figure 2: Simulation of pipelined CPU for cycle 6 to 10

The waveform of cycle 6 to cycle 10 is shown in Fig. 2.

In the sixth cycle, the sixth instruction is fetched.At the beginning of this cycle the value of MEM(0) is changed to the value of $t2 before the first instruction executes. The fifth instruction is in the ID stage while the fourth instruction is in the EX stage. The third instruction is in the MEM stage. Since the new value of $t3 is still in MEM stage, the fifth instruction get the old value of $t3 which is not correct.

In the seventh cycle, the seventh instruction is fetched. The sixth instruction is in the ID stage while the fifth instruction is in the EX stage. The third instruction is in the WB stage, which the right value of $t3 is written back to the register files. So in the second half of this cycle, the value of $t3 changes to 4. And also the value of $t3 read by the sixth instruction is right. The fourth instruction writes the right value to the data memory of $t2 after the first instruction executes.

In the eighth cycle, the eighth instruction which is nop is fetched. At the beginning of the cycle, the right value of $t2 is written to MEM(0). The seventh instruction is in ID stage while the sixth instruction is in EX stage. The fifth instruction with the old value of $s3 is in MEM stage. The fourth instruction in WB stage does not write to any registers, so $RegWrite$ is 0.

In the ninth cycle, the ninth instruction which is nop is fetched. At the beginning of the cycle, the old value of $t3 is written to MEM(4). The eighth instruction is in ID stage while the seventh instruction is in EX stage. The sixth instruction with the right value of $t3 is in the MEM stage, which writes 4 to DMEM(8).

In the tenth cycle, the tenth instruction is fetched. At the beginning of the cycle, the new value of $t3 is written to MEM(8). The nop instruction is in ID stage and EX stage. The seventh instruction is in MEM stage which the right value of $s2 has been calculated and ready to be written in next stage.

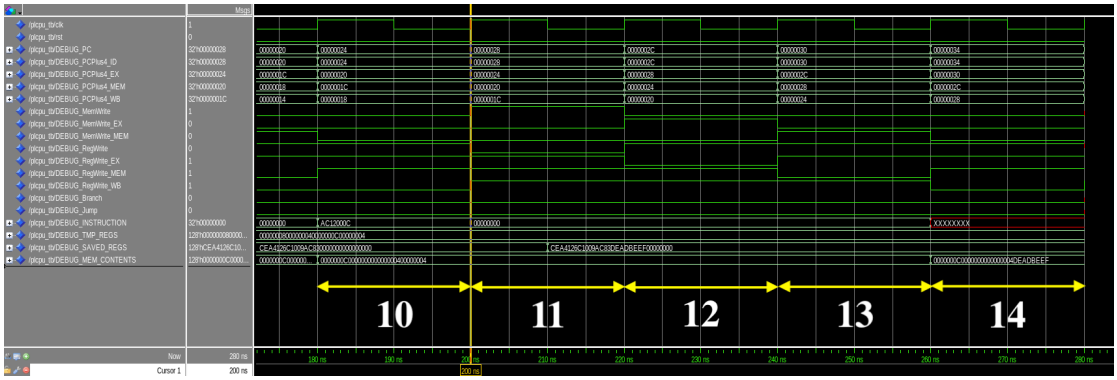

Figure 3: Simulation of pipelined CPU for cycle 10 to 14

The waveform of cycle 10 to cycle 14 is shown in Fig. 3.

In the 11th stage, the nop instruction is fetched. The nop instruction is in EX and MEM stage. The tenth instruction is in ID stage while the seventh instruction is in WB stage thus the new value of $s2 is written to register files in the second half of the cycle. What's more, the tenth instruction reads the right value of $s2 because it has been written back by the seventh instruction which is in WB stage. Note that 0xDEADBEEF = 3735928559. The two nop instructions help to eliminate the data hazard of $s2.

In the 12th stage, the nop instruction is fetched. The nop instruction is in ID, MEM and WB stage. The tenth instruction is in EX stage with the right value of $s2.

In the 13th stage, the last nop instruction is fetched. The nop instruction is in ID, EX and WB stage. The tenth instruction is in MEM stage with the right value of $s2 to write to MEM(0xC).

In the 14th stage, since we do not write anything to the instruction memory after the last nop, so there is no instruction fetched. At the beginning of the cycle, the new value of $s2 is written to MEM(0xC). The nop instruction occupies the ID, EX, MEM stage. The last three nop instructions help to continue the pipeline to write the right value of $s2 to the right position of the memory.