Assignment 2

Qinyun Song

1 Pipelining Performance

1. As we know, $CPUtime = IC \times CPI \times CT$. Here, IC is 1000, CPI is 1 since it can execute one instruction per cycle. And CT is 10ns. So we can have
$$CPUtime = 1000 \times 1 \times 10ns = 10000ns$$

2. First of all, 2 bubbles are added every 10 cycles. So we have
$$newCPI = (10 + 2)/10 = 1.2$$
Since it is converted into 10 stage pipeline and 1ns additional delay is added, we know that
$$newCT = 10ns/10 + 1ns = 2ns$$
As for the IC, it is not changed. So we can get the new CPU time as
$$newCPUtime = 1000 \times 1.2 \times 2ns = 2400ns$$
Now we can calculate the speedup:
$$10000ns/2400ns \approx 4.17$$

2 Control Hazards

First of all, we can see that, 80% instructions are not branches. For the remaining 20%, 60% of them are taken. Also, we know that we can move two instructions from the taken side into the branch delay slot. So there will be no penalty when we take the branches. For the remaning 40% part, two delays are generated each time. After all, there are only $20\% \times 40\% = 8\%$ instructions need two more cycles. So the expected CPI is
$$1 + 8\% \times 2 = 1.16$$

3 Multi-cycle Instructions

1. First of all, index those instructions from 1 to 7. Then we can generate the following table.

Time	1	2	3	4	5	6	7
1	IF						
2	IS	IF					
3	RR	IS	IF				
4	LOAD	RR	IS	IF			
5	WRITE	LOAD	NOP	NOP			
6	*	WRITE	RR	IS	IF		
7	*	*	MUL	RR	IS	IF	
8	*	*	MUL	SAVE	NOP	NOP	
9	*	*	MUL	WRITE	NOP	NOP	
10	*	*	WRITE	*	RR	IS	IF
11	*	*	*	*	DIV	RR	IS
12	*	*	*	*	DIV	ADD	RR
13	*	*	*	*	DIV	WRITE	SAVE
14	*	*	*	*	DIV	*	WRITE
15	*	*	*	*	WRITE	*	*

2. (a) RAW between instruction 2 and 3 when they both need F2.
   (b) RAW between instruction 3 and 5 when they both need F0.
   (c) RAW between instruction 1 and 4 when they both need F6.
   (d) RAW between instruction 1 and 5 when they both need F6.
   (e) RAW between instruction 4 and 6 when they both need F8.
   (f) RAW between instruction 6 and 7 when they both need F8.
   (g) RAW between instruction 2 and 6 when they both need F6.
   (h) WAR between instruction 4 and 6 when they both need F6.
   (i) WAR between instruction 5 and 6 when they both need F6.
   (j) WAW between instruction 1 and 6 when they both need F6.
   (k) Structual hazard on cycle 4 when both LOAD and IF perform at the same time. (Can be solved by seperating the memory and register reading.)
   (l) Structual hazard on cycle 6 when both WRITE and READ at the same time. (Can be solved if half of the cycle is used to write and the remaining half is used to read.)
   (m) Structual hazard on cycle 10 when both WRITE and READ at the same time. (Solution is the same as above).

4 Points of Production and Consumption

1. For un-pipelined processor, the cycle time is $36 + 0.5 = 36.5ns$. The IPC is 1. So the thoughput is $1/36.5 \approx 0.03$.

2. For the pipelined processor, since it has 12 stages, each stage needs 3ns. And there is a latch between 2 stages. So one stage will take $3 + 0.5 = 3.5ns$. If there is no hazard, the thoughput would be $1/3.5$. But if data hazard exists, the thoughput would be $1/(3.5 \times 4)$. From the problem we know that only half instruction lead to data hazard. So the total thoughput would be $1/(0.5 \times (3.5 + 3.5 \times 4)) \approx 0.11$.