Note – All correct answers are maked in bold

Q1. By implementing pipelined processor, we improve [2 points]

Time to execute individual instruction

Instruction Throughput

Reduce Cycles per Instruction

Increase clock frequency

Q2. Which of the following reduces performance of pipelined design [2 points]

1. Data dependence between instructions
2. Conditional Branch Operations
3. Unconditional Branch such as Jump
4. Register Writes

Q3. In the five stage MIPS processor design, __ and __ are latched on the IF/ID latch? [2 points]

PC + 4 and Instruction, 50% partial credit for 32 bit or word

Q4. Say Control Signal Propagation Delay = Latch Delay + Wire Delay, Assuming wire delay = ~0, and latch delay is 100 pico-seconds, which control signal will have the longest propagation delay? Calculate the delay in pico-seconds. [4 points]

MemtoReg or RegWrite

Both need to latch three times, so the total latch delay = 300 pico

Q5. We are using a register file design with two phase clock such that register read is performed in first clock cycle and register write is performed in second clock cycle. If we use this RF in our five-stage pipelined design. How many cycles will we need for the following program. Assume compile is stalling the pipeline. [5 points]

Answer: 14 total cycles

Instr / Cycle	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14
add $1, $2, $3	F	D	X	M	W										
add $4, $5, $6		F	D	X	M	W									
sub $7, $1, $4			N	N	N	F	D	X	M	W					
add $8, $7, $5							N	N	N	F	D	X	M	W	

1. [4 points] Consider the following MIPS assembly code:

```
L0.     sw      $s1, 100($s3)
L1.     add     $s3, $s1, $s0
L2.     lw      $s2, 100($s3)
L3.     lw      $s3, 0($s2)
L4.     beq     $s3, $s2, 20
```

In the table below, list all the data dependences (RAW, WAW, WAR) and indicate whether a data hazard may occur when executing on the five-stage pipeline as described in the textbook. If more rows are needed, just add them manually. Include instruction line numbers for both the instruction that produces the value and instruction that need to use that value (e.g., L0, L1).

Solution:

Register	Instruction Line #	Type of Dependency	Data Hazard?
$s3	L1, L2	RAW	Y
$s2	L2, L3	RAW	Y
$s2	L2, L4	RAW	Y
$s3	L3, L4	RAW	Y
$s3	L1, L3	WAW	N
$s3	L0, L1	WAR	N
$s3	L0, L3	WAR	N
$s3	L2, L3	WAR	N

2. [3 points] Consider two different machines. The first has a single cycle datapath (i.e., a single stage, non-pipelined machine) with a cycle time of 5 ns. The second is a pipelined machine with 5 pipeline stages and a cycle time of 1ns.

(a) [1 point] What is the speedup of the pipelined machine versus the single cycle machine assuming there are no stalls?

Solution:

The speedup is $5\ ns/1\ ns = 5$.

(b) [1 point] What is the speedup of the pipelined machine versus the single cycle machine if the pipeline stalls 1 cycle for 25% of the instructions?

Solution:

$$New\ CPI = 1 + .25 \times 1 = 1.25$$

Since the number of instructions is the same, the speedup is $\frac{CPI_{old} \times cycle\ time_{old}}{CPI_{new} \times cycle\ time_{new}} = \frac{1 \times 5\ ns}{1.25 \times 1\ ns} = 4.$

(c) [1 point] Now consider a 4-stage pipeline machine with a cycle time of 1.1 ns. Again, assuming no stalls, is this implementation faster or slower than the original 5 stage pipeline? Explain your answer.

Solution:

The 5-stage machine is faster. This is because it has a smaller cycle time, which results in a faster overall execution time (since there are no stalls, they both have the same CPI).