Computer Organization & Architecture

Pipeline Processing

DPP: 2

Q1 Consider the given set of instructions, which are having instruction format as follows:

Opcode, Destination operand, Source1, Source2

1. ADD R2, R1, R0 R2 ← R1 + R0
2. MUL R4, R3, R2 R4 ← R3 * R2
3. SUB R6, R5, R4 R6 ← R5 - R4
4. ADD R6, R1, R8 R6 ← R1 + R8
5. MUL R4, R9, R2 R4 ← R9 * R2
6. SUB R9, R3, R4 R9 ← R3 - R4

The number of RAW, WAW and WAR dependencies are respectively ?

(A) 4, 2, 2 (B) 5, 2, 2

(C) 4, 1, 2 (D) 5, 1, 2

Q2 Consider the given set of instructions, which are having instruction format as follows:

Opcode, Destination operand, Source1, Source2

ADD R2, R1, R0 R2 ← R1 + R0

MUL R4, R3, R2 R4 ← R3 * R2

SUB R6, R5, R4 R6 ← R5 - R4

ADD R6, R7, R8 R6 ← R7 + R8

MUL R7, R1, R2 R7 ← R1 * R2

SUB R1, R3, R4 R1 ← R3 - R4

The given instructions are executed in a 5 segment instruction pipeline which has segments as: Instruction Fetch, Instruction Decode, Operand Fetch, Execution and Write Back.

The speed up is calculated as follows :

$$\text{Speed up} = \frac{\text{Number of cycles needed without operand forwarding}}{\text{Number of cycles needed with operand forwarding}}$$

Speed of the pipeline (correct up to 1 decimal place) for execution of above instructions is _____ ?

Q3 The instruction pipeline of a RISC processor has the following stages: Instruction Fetch (IF), Instruction Decode (ID), Operand Fetch (OF), Perform Operation (PO) and Writeback (WB), The IF, ID, OF and WB stages take 1 clock cycle each for every instruction. Consider a sequence of 200 instructions. In the PO stage, 40 instructions take 4 clock cycles each, 65 instructions take 2 clock cycles each, and the remaining instructions take 1 clock cycle each. Assume that there are 20 instructions which cause 2 stalls each due to data hazards and there are no control hazards in program.

The number of clock cycles required for completion of execution of the sequence of instruction is _____?

Q4 Consider a 5 - stage instruction pipeline, where the stages take delays of 6 nanoseconds, 5 nanoseconds, 8 nanoseconds, 6 nanoseconds and 7 nanoseconds respectively. When an application is executing on this 5-stage pipeline, consider 20% of the instructions incur 3 pipeline stall cycles. The speed up (correct up to 2 decimal places) of the pipeline as compared to its corresponding non-pipeline system is _____ ?

Q5 Consider a non-pipelined processor operating at 10 GHz. It takes 4 clock cycles to complete an instruction. The same processor is upgraded to pipelined processor with 5 stages but clock rate is reduced to 8GHz. A program is executed on these processors which has 8% load/store

instructions, 12% branch instructions and remaining ALU instructions. The memory access instructions cause 1 clock cycles each if there is no any cache miss but cause 40 cycles with cache miss. The 40% of all branch instructions cause 2 stalls each. There is no any stall associated with ALU instructions. Assume that cache has 94% hit. The speed up (round of up to 2 decimal place) achieved by pipeline over the non-pipeline processor for this program is _____?

Q6 Which of the following statements is/are false regarding pipelining ?

(A) For single input non-pipeline can perform better than equivalent pipeline system

(B) Pipeline can be fruitful when multiple different processing is applied over multiple inputs

(C) Pipeline with single segment cannot provide parallel processing

(D) Non-pipeline system does not require intermediate registers/buffer for synchronization

Q7 Consider a 6 segment pipeline in which all segments take 1 cycle for each instructions except the execution phase. Execution phase takes 3 cycle for multiply instruction, 6 cycles for division instruction and 1 cycle for addition and 1 cycle subtraction instruction. Suppose a code segment is executed on this pipeline in which total 8 instructions are executed and total 29 cycles needed for execution. Which of the following option(s) is/are correct regarding number of instructions of each type ?

(A)

ADD	SUB	MUL	DIV
2	2	2	2

(B)

ADD	SUB	MUL	DIV
1	1	3	3

(C)

ADD	SUB	MUL	DIV
1	2	3	2

(D)

ADD	SUB	MUL	DIV
2	1	2	3

Answer Key

Q1	(A)
Q2	1.4~1.4
Q3	429~429
Q4	2.5~2.5
Q5	2.35~2.35
Q6	(B)
Q7	(C)

Hints & Solutions

Q1 Text Solution:

RAW dependencies are from instruction 1 to 2 for R2, from instruction 1 to 5 for R2, from instruction 2 to 3 for R4 and from instruction 5 to 6 for R4. There is no any RAW dependency for R4 from instruction 2 to 6 because R4 of instruction 6 depends on R4 of instruction 5 and not of R4 of instruction 2.

WAW dependencies from instruction 2 to 5 for R4 and from instruction 3 to 4 for R6.

WAR dependencies from instruction 3 to 5 for R4 and from instruction 5 to 6 for R9

Q2 Text Solution:

Solution: Number of data dependencies which can cause stall are: 2; which are from instruction 1 to 2 for R2, and from instruction 2 to 3 for R4.

Number of stalls due to each data dependency = stage number of write back - stage number of operand fetch

= 5 – 3

= 2

Number of stages (k) = 5

Number of instructions executed (n) = 6

Without operand forwarding total number of cycles = k + n – 1 + stall cycles for 2 data dependencies

= 5 + 6 – 1 + 2 * 2

= 14

With operand forwarding, the stalls due to dependencies are eliminated here, hence

With operand forwarding total number of cycles

= k + n – 1

= 5 + 6 – 1

= 10

Hence speed up = 14/10 = 1.4

Q3 Text Solution:

Stalls due to 40 instructions which take 4 cycles in PO stage = 40 * (4 – 1) = 120

Stalls due to 65 instructions which take 2 cycles in PO stage = 65 * (2 – 1) = 65

Stall due to data hazards = 20* 2 = 40

Total stall cycles = 120 + 65 + 40 = 225

Number of stages (k) = 5

Number of instructions (n) = 200

Total number of cycles = k + n – 1 + total stall cycles

= 5 + 200 – 1 + 225

= 429

Q4 Text Solution:

One instruction execution time in non-pipeline system (tn) = 6 + 5 + 8 + 6 + 7 = 32ns

CPI of pipeline system = 1 + 0.2 * 3 = 1.6

Pipeline cycle time (tp) = max (6,5,8,6,7) = 8 nanoseconds

Speed up of pipeline = 32 / (1.6*8)

= 2.5

Q5 Text Solution:

Cycle time in non-pipeline system = 1/10GHz = 0.1nanoseconds

One instruction execution time in non-pipeline system = 4 * 0.1 = 0.4 nanoseconds

Average Cycles per Instruction (CPI) in pipeline = 1 + 0.08 * 0.94 * 1 + 0.08 * 0.06 * 40 + 0.12 * 0.4 * 2

= 1 + 0.2672 + 0.096

= 1.3632

Pipeline cycle time (tp) = 1/8GHz = 0.125 nanoseconds

Speed up of pipeline = 0.4 / (1.3632 * 0.125) = 2.347 = 2.35

Q6 Text Solution:

(A) Option A is true because for 1 input pipeline time is k*tp, which is practically greater than one operation time in non-pipeline

(B) Option B is false because pipeline is useful when same processing is to be applied over multiple inputs not different processing.

(C) Option C is true because for parallel processing pipeline must have multiple segments or stages

(D) Option D is true because intermediate buffer or register are used only in pipeline.

Q7 Text Solution:

Number of stall cycles due to structural hazard for each ADD instruction: 1 – 1 = 0

Number of stall cycles due to structural hazard for each SUB instruction: 1 – 1 = 0

Number of stall cycles due to structural hazard for each MUL instruction: 3 – 1 = 2

Number of stall cycles due to structural hazard for each DIV instruction: 6 – 1 = 5

Without stalls number of cycles = k + n – 1 = 6 + 8 – 1 = 13

For option A, number of stalls = 2 * 2 + 2 * 5 = 14, hence total cycles = 13 + 14 = 27

For option B, number of stalls = 3 * 2 + 3 * 5 = 21, hence total cycles = 13 + 21 = 34

For option C, number of stalls = 3 * 2 + 2 * 5 = 16, hence total cycles = 13 + 16 = 29

For option A, number of stalls = 2 * 2 + 23 * 5 = 19, hence total cycles = 13 + 19 = 32.