**Chapter 16 - Exercises**

**16.1. What is the essential characteristic of the superscalar approach to processor design?**

Can execute multiple independent instructions in parallel.

**16.2. What is the difference between the superscalar and super pipelined approaches?**

+ Superscalar machines can issue several instructions per cycle.

+ Superpipelined machines can issue only one instruction per cycle, but they have cycle times shorter than the time required for any operation.

+ Both of these techniques exploit instruction-level parallelism, which is often limited in many applications.

**16.3. What is instruction-level parallelism?**

Tells how many instructions in a program can be executed in

**16.4. Briefly define the following terms: • True data dependency • Procedural dependency • Resource conflicts • Output dependency • Antidependency**

● True data dependency Can fetch and decode second instruction in parallel with first but can NOT execute second instruction until first is finished

● Procedural dependency

Can not execute instructions after a branch in parallel with instructions before a branch

● Resource conflicts Two or more instructions waiting for the same resource

● Output dependency Write after write (WAW)

● Antidependency Write after Read (WAR).

**16.5. What is the distinction between instruction-level parallelism and machine parallelism?**

+ Instruction-level parallelism exists when instructions in a sequence are independent and thus can be executed in parallel by overlapping. Machine parallelism is a measure of the ability of the processor to take advantage of instruction-level parallelism.

+ Machine parallelism is determined by the number of instructions that can be fetched and executed at the same time (the number of parallel pipelines) and by the speed and sophistication of the mechanisms that the processor uses to find independent instructions.

**16.6. List and briefly define three types of superscalar instruction issue policies**.

+ In-order issue with in-order completion: Issue instructions in the exact order that would be achieved by sequential execution and to write results in that same order.

+ In-order issue with out-of-order completion: Issue instructions in the exact order that would be achieved by sequential execution but allow instructions to run to completion out of order.

+ Out-of-order issue with out-of-order completion: The processor has a look-ahead capability, allowing it to identify independent instructions that can be brought into the execute stage. Instructions are issued with little regard for their original program order. Instructions may also run to completion out of order.

**16.7. What is the purpose of an instruction window ?**

A buffer which allows out-of-order issue to decouple the Decoding stage and Execution stage

**16.8. What is register renaming and what is its purpose?**

Is a technique in pipelining concept which makes use of Data dependencies between the instructions by renaming operand registers. Eliminates WAR WAW dependencies.

**16.9. What are the key elements of a superscalar processor organization?**

Instructions fetch strategies which help in fetching several instructions simultaneously by calculating:

• Outcome

• Fetching out

• Instructions on conditional branch Multiple stages of pipeline fetch, Pipeline decode and pipeline branch prediction logic is used by above points.

**Answers to Questions**