

**UNIT-4**  
**PARALLELISM**  
**PART-A**

**1. What is Instruction Level Parallelism? (NOV/DEC 2011)**

Pipelining is used to overlap the execution of instructions and improve performance. This potential overlap among instructions is called instruction level parallelism (ILP).

**2. Explain various types of Dependences in ILP.**

- Data Dependences
- Name Dependences
- Control Dependences

**3. What is Multithreading?**

Multithreading allows multiple threads to share the functional units of a single processor in an overlapping fashion. To permit this sharing, the processor must duplicate the independent state of each thread.

**4. What are multiprocessors? Mention the categories of multiprocessors?**  
Multiprocessor are used to increase performance and improve availability. The different categories are SISD, SIMD, MIMD.

**5. What are two main approaches to multithreading?**

- Fine-grained multithreading
- Coarse-grained multithreading

**6. What is the need to use multiprocessors?**

1. Microprocessors as the fastest CPUs collecting several much easier than redesigning 1
2. Complexity of current microprocessors Do we have enough ideas to sustain 1.5X/yr? Can we deliver such complexity on schedule?
3. Slow (but steady) improvement in parallel software (scientific apps, databases, OS)
4. Emergence of embedded and server markets driving microprocessors in addition to desktops embedded functional parallelism, producer/consumer model
5. Server figure of merit is tasks per hour vs. latency.

**7. Write the advantages of Multithreading.**

If a thread gets a lot of cache misses, the other thread(s) can continue, taking advantage of the unused computing resources, which thus can lead to faster overall execution, as these resources would have been idle if only a single thread was executed. If a thread cannot use all the computing resources of the CPU (because instructions depend on each other's result), running another thread permits to not leave these idle. If several threads work on the same set of data, they can actually share their cache, leading to better cache usage or synchronization on its values.

**8. Write the disadvantages of Multithreading.**

Multiple threads can interfere with each other when sharing hardware resources such as caches or translation lookaside buffers (TLBs). Execution times of a single-thread are not improved but can be degraded, even when only one thread is executing. This is due to slower frequencies and/or additional pipeline stages that are necessary to accommodate thread-switching hardware. Hardware support for Multithreading is more visible to software, thus requiring more changes to both application programs and operating systems than Multi processing.

**9. What is CMT?**

Chip multiprocessors - also called multi-core microprocessors or CMPs for short - are now the only way to build high-performance microprocessors, for a variety of reasons. Large uniprocessors are no longer scaling in performance, because it is only possible to extract a limited amount of parallelism from a typical instruction stream using conventional superscalar instruction issue techniques. In addition, one cannot simply ratchet up the clock speed on today's processors, or the power dissipation will become prohibitive in all but water-cooled systems.

**10. What is SMT?**

Simultaneous multithreading, often abbreviated as SMT, is a technique for improving the overall efficiency of superscalar CPUs with hardware multithreading. SMT permits multiple independent threads of execution to better utilize the resources provided by modern processor architectures.

**11. Write the advantages of CMP?**

CMPs have several advantages over single processor solutions energy and silicon area efficiency i. By incorporating smaller less complex cores onto a single chip ii. Dynamically switching between cores and powering down unused cores iii. Increased throughput performance by exploiting parallelism iv. Multiple computing resources can take better advantage of instruction, thread, and process level

**12. What are the Disadvantages of SMT?**

Simultaneous multithreading cannot improve performance if any of the shared resources are limiting bottlenecks for the performance. In fact, some applications run slower when simultaneous multithreading is enabled. Critics argue that it is a considerable burden to put on software developers that they have to test whether simultaneous multithreading is good or bad for their application in various situations and insert extra logic to turn it off if it decreases performance.

**13. What are the types of Multithreading?**

- Block multi-threading
- Interleaved multi-threading

**14. What Thread-level parallelism (TLP)?**

- Explicit parallel programs already have TLP (inherent)
- Sequential programs that are hard to parallelize or ILP-limited can be speculatively parallelized in hardware.

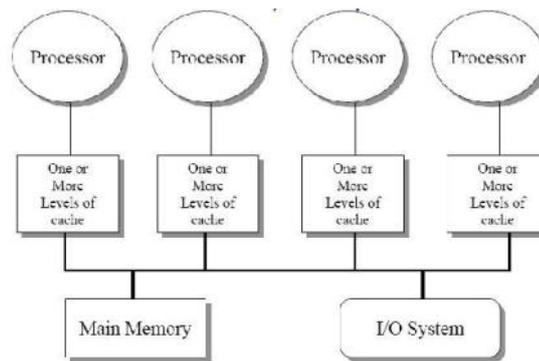
**15. List the major MIMD Styles**

Centralized shared memory ("Uniform Memory Access" time or "Shared Memory Processor") Decentralized memory (memory module CPU) get more memory bandwidth, lower memory Drawback: Longer communication latency Drawback: Software model more complex

**16. Distinguish between shared memory multiprocessor and message-passing multiprocessor.**

- A multiprocessor with a shared address space, that address space can be used to communicate data implicitly via load and store operations is shared memory multiprocessor.
- A multiprocessor with a multiple address space, communication of data is done by explicitly passing message among processor is message-passing multiprocessor.

**17. Draw the basic structure of Basic Structure of a Symmetric Shared Memory Multiprocessor.**



**18. What is multicore'?**

At its simplest, multi-core is a design in which a single physical processor contains the core logic of more than one processor. It's as if an Intel Xeon processor were opened up and inside were packaged all the circuitry and logic for two (or more) Intel Xeon processors. The multi-core design takes several such processor "cores" and packages them as a single physical processor. The goal of this design is to enable a system to run more tasks simultaneously and thereby achieve greater overall system performance.

**19. Write the software implications of a multicore processor?**

Multi-core systems will deliver benefits to all software, but especially multi-threaded programs. All code that supports HT Technology or multiple processors, for example, will benefit automatically from multi-core processors, without need for modification. Most server-side enterprise packages and many desktop productivity tools fall into this category.

**20. What is coarse grained multithreading?**

It switches threads only on costly stalls. Thus it is much less likely to slow down the execution an individual thread.

**21. What is multiple issue? Write any two approaches.**

Multiple issue is a scheme whereby multiple instructions are launched in one clock cycle. It is a method for increasing the potential amount of instruction-level parallelism. It is done by replicating the internal components of the computer so that it can launch multiple instructions in every pipeline stage. The two approaches are: 1. Static multiple issue (at compile time) 2. Dynamic multiple issue (at run time)

**PART-B**

1. **Explain Instruction level parallelism**
2. **Discuss the difficulties faced by parallel processing programs**
3. **Write about shared memory multiprocessor**
4. **Explain in detail Flynn's classification of parallel hardware**
5. **Discuss in detail hardware Multithreading**
6. **Explain SISD and MIMD**
7. **Discuss SIMD and MISD**
8. **Explain Multicore processors**
9. **Write about the different types of multithreading**