Multithreading

Multithreading

- Thread : is a process with its own instructions and data
- It may be apart of a parallel program or represent an independent program on its own.
- Multithreading: is the execution of multiple threads simultaneously.
- ILP exploits implicit parallelism while TLP exploits explicit parallelism

Multithreading

- Multiple threads to share the functional units of a single processor in an overlapping fashion
- processor must duplicate the resources
  - Separate registers
  - PC
  - Page table
  - Memory is shared thru virtual memory mech.
  - H/W must support thread switching

Multithreading Classification

- Fine-grained multithreading
- Coarse-grained multithreading
- Simultaneous Multithreading

Fine grain Multithreading

- Switches between threads on each instruction
- Execution of multiples threads to be interleaved.
- Interleaving is done in a round-robin fashion
- CPU must be able to switch threads on every clock cycle

Fine grain Multithreading

- Advantage:
  - it can hide the throughput losses that arise from both short and long stalls.
- Disadvantage:
  - it slows down the execution of the individual threads.

Coarse-grained multithreading

- Switches threads only on costly stalls
  - Ex: level two cache misses
- Alternative to fine grained multithreading
- CPU with coarse-grained multithreading issues instructions from a single thread
- Advantage:
- Is less likely to slow the processor down .
  - since instructions from same thread will only be issued, when a thread encounters a costly stall

Coarse-grained multithreading

- Drawback:
  - limited in its ability to overcome throughput losses
  - especially from shorter stalls
- when a stall occurs, the pipeline must be emptied or frozen

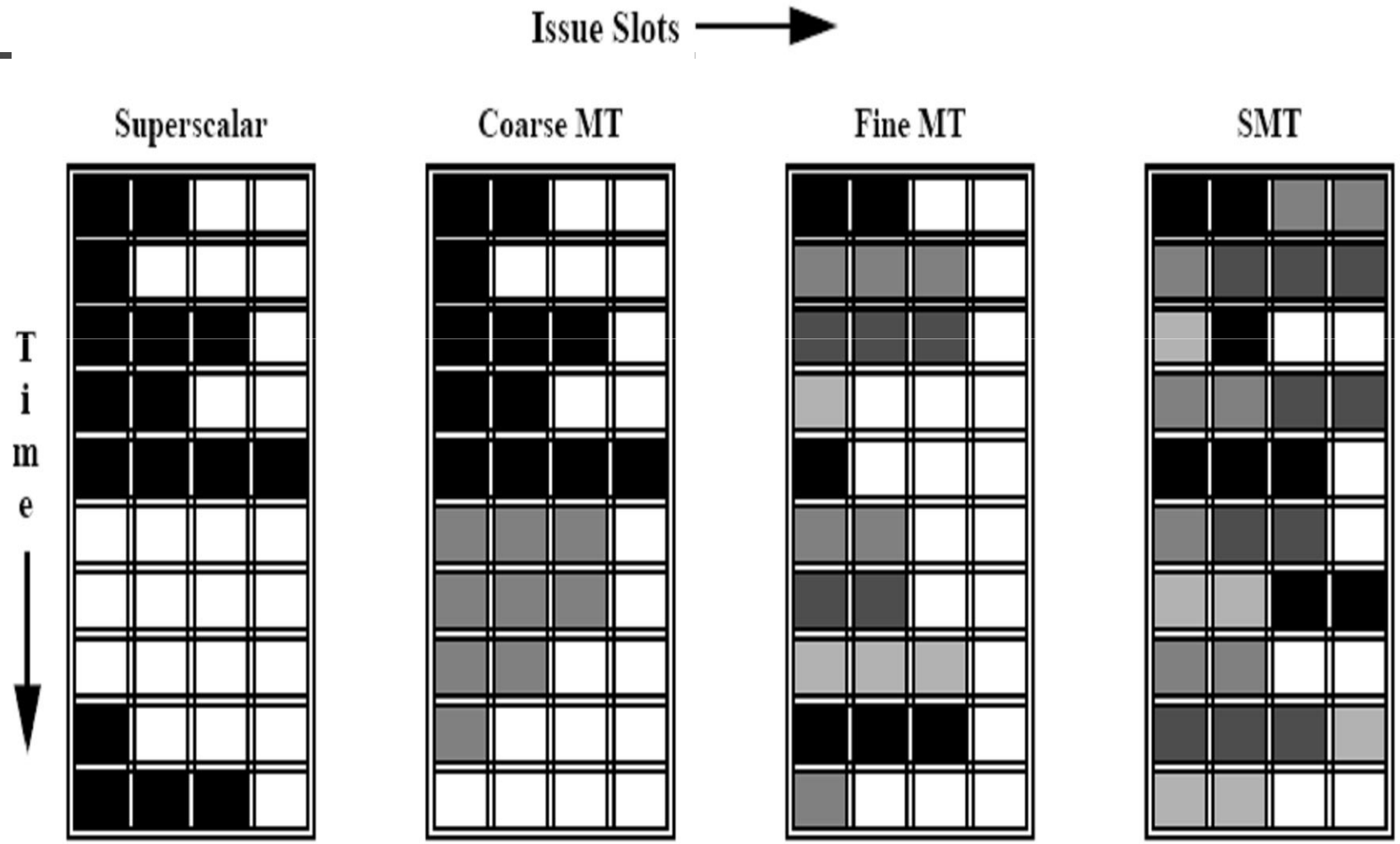
Simultaneous Multithreading

- It exploit TLP at the same time it exploits ILP
- SMT is multiple-issue processors often have more functional unit parallelism available
- SMT uses the concepts like
  - Multiple-issue
  - Register Renaming
  - Data forwarding
  - Static scheduling
  - Dynamic scheduling

Example

- superscalar with no multithreading support
- superscalar with coarse-grained multithreading
- superscalar with fine-grained multithreading
- superscalar with simultaneous multithreading.

Example



Example

- Horizontal dimension represents the instruction issue capability in each clock cycle.
- The vertical dimension represents a sequence of clock cycles.
- Empty box indicates that the corresponding issue slot is unused in that clock cycle

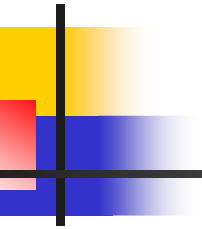
Example

- Superscalar without MT:
  - Exploits ILP
  - No Multithreading facility
  - Large no of processor idle cycles.
- Coarse Grain MT:
  - In the coarse-grained multithreading the long stalls are partially hidden by switching to another thread
  - since thread switching only occurs when there is a stall there are likely to be some fully idle cycles

Example

- fine-grained MT:
  - the interleaving of threads eliminates fully empty Slots
  - only one thread issues instructions in a given clock cycle
  - ILP limitations still lead to a significant number of idle slots within individual clock cycles.
  - SMT
  - ILP and TLP are exploited
  - multiple threads using the issue slots in a single clock cycle
  - No issue slot is idle

Multithreading



- Advantages:
  - If a thread gets lot of cache misses then the other threads can continue by using the computing resources.
  - If several threads work on the same set of data then better cache usage and sync can be achieved
  - If a thread can not use all the computing resources running other threads permit to use these resources.

Multithreading

- Disadvantages:
  - Multiple threads can interfere with each other when sharing h/w resources like cache ,TLB.
  - H/W support for Multithreading is more visible to S/W.
- Applications:
  - Used in server side applications