classmate
Date 5.2.2020
Page

MULTITHREADING

use of multithreading to increase uniprocessor throughput by using multiple threads to hide pipeline and memory latencies

multithreading :- primary technique for exposing more parallelism to hardware.

multiple threads to share the functional units of a single processor in an overlapping fashion. ILP has a great advantage is it is reasonable transparent to programmer.

multithreading does not duplicate the entire processor multithreading shares most of the processor core among a set of threads duplicating only private state such as registers and PC.

very recent processors will have multiple cores on a single chip and provide multithreading on within each core.

hardware must support the ability to change to a different thread relatively quickly. (thread switching should be faster than process switching), and it should be more efficient than process switching.

3 main hardware approaches to multi-threading.

1. fine grained multithreading.
2. coarse grained multithreading.
3. simultaneous multithreading.

FINE GRAINED MULTITHREADING

it switches threads on each cycle. this interleaving will be carried out in a round-robin fashion.

advantage →
- can hide throughput losses that arise from both short and long stalls.

disadvantage →
- it slows down the execution of individual threads.

COARSE GRAINED MULTITHREADING

thread switch takes place only on costly stalls, like L2-cache / L3-cache miss. there is much less likely to slow down the execution of any thread.

it is limited in its ability to overcome throughput losses especially from shorter stalls.

the most common implementation of multithreading is called simultaneous multithreading SMT.

SIMULTANEOUS MULTITHREADING

THREAD LEVEL PARALELLISM

increased importance of multiprocessing
- the dramatically lower efficiencies in silicon and energy use that were encountered in 2002 - 2005. designers attempted to find and exploit more ILP.

- a growing interest in high-end servers as cloud computing and ~~SAS~~ SaaS became more important.

- growth in data intensive applications is driven by massive amounts of data on the internet.

- the insight that increasing performance on the desktop is less important. highly compute data intensive applications are being run on the clouds.

- an improved understanding of how to use multiprocessors effectively.

- advantage of leveraging a design investment by replication rather than by unique design.

exploit TLP through multiple instructions multiple data MIMD

extensive use of TLP for a wide range of general purpose applications.

MULTIPROCESSORS.

they are tightly coupled processors and there would be a single OS and they will have a shared memory.

the tightly coupled set of threads collaborating on an single task - parallel processing.

a form of request level parallelism can be executed on them.
(in general they will run on multicomputer systems).

multicomputers are less tightly coupled than multiprocessors.