SPARC : US7209996B2:Multi-core multi-thread processor

This invention is very vital for current and future sparc development in oracle . Sparc has greatly benefited by this invention in delivering next generation of server processors to market.The innovation has been motivated by a need for hiding memory latency of operations by an processor architecture configured to efficiently process server applications using multi-threading.

It should be appreciated although the present invention can be implemented in numerous ways, here a device and method have been described to accomplish it.

Some of the workloads like OLTP(TPC-C),SAP and SPEC2000/2006 rate type of workloads saw huge throughput improvements over single threaded processing which existed before and attained a leadership position in market.Sparc Processor like T4,T5,M5 and M6 all use this innovation for multithreading till date to enhance the performance.As can be seen T4 with64 threads and T5 with 128 threads more than doubled the performance for following workloads:-

SPECint_rate2006:-179(T4 at 3.0Ghz),
464(T5 at 3.6Ghz) with ratio of T5/T4 = 2.59
SPECfp_rate2006:-144(T4 at 3.0Ghz),
381(T5 at 3.6Ghz) with ratio of T5/T4=2.15
TPC-C (K) :- 570(T4 at 3.0Ghz) to
1228(T5 at 3.6Ghz) with ratio of T5/T4 = 2.15
SAP-SD EHP4 (Users):- 2380(T4 at 3.0Ghz) to
5400(T5 at 3.6Ghz) with ratio of T5/T4 = 2.27

and above behavior is observed in many other java related applications as well.

The invention benefits more to processors which includes two or more cores, where each of the cores include a first level cache memory. Each of the cores are multi-threaded capable of keeping context of one to N threads active . A crossbar is included to connect second level cache bank memories in communication with the cores. Each of the level 2 cache bank memories also communicate with a main dram memory interface.

The Method of Multi-threading operation has been described which initiates operation of accessing a processor core resources through a first thread instruction stream. As soon as first thread encounters a long latency operation in its stream it is suspended and second thread operation ready to access the processor core is identified and selected to execute. Meanwhile first thread performs and completes the long latency operation in the background.

In essence the method described above for multithreading can keep many contexts alive and make progress in their execution within a single core and similarly for many cores within a processor chip.It can hide long latencies of dram memory as seen by any thread for load misses by taking

advantage of scheduling another thread and keeping high utilization of resources of core all the time.

This nature of multithreading in hardware enables very high throughput from a single core or a single cpu chip.