International Journal of Research in Engineering, Science and Management
Volume-2, Issue-3, March-2019
www.ijresm.com | ISSN (Online): 2581-5792

A Survey on Hardware Multi-Threading and Multi-Processors

M. Siva Kumar Reddy[1], A. Gowtham Venkata Sai[2]
[1,2]*Student, Department of Electronics and Communications Engg., Saveetha School of Engg., Chennai, India*

***Abstract*: The goal of this survey is to present the topic of hardware multithreading in different fields and also the advancements in hardware multithreading concepts. The use of multithreading in different processors, IOT concepts using Multithreading, concept of multiprocessor for executing plurality of programs, pipelining concept, concurrent task executing Error protection by Redundant Multithreading and java multithreading and in VLIW Processors, Multithreading in XILINX Microkernel with hardware scheduler and also in reliable aerospace computers.**

***Keywords*: multithreading, multiprocessor, RISC, CISC, VLIW Processor, chip multiprocessor, simultaneous multithreading.**

1. Introduction

One of the main focuses of current work in computer architecture is the design of hardware organizations that support the parallel executions of instructions. Data flow architectures are becoming a great deal of attention as they execute instructions as soon as operands are given. A few topics of SAM architecture is also explained. Performance is the main difference of microprocessors. Before 2002, the microprocessors architecture focused on Instruction Level Parallelism(ILP) which was modified and improved when single-core superscalar processors were introduced. A few concepts of Thread Level Parallelism(TLP). Research in computer architecture aims at exploiting higher levels of parallelism in instruction processing. Simultaneous Multithreading(SMT) and Chip Multiprocessors(CMP) exploit instruction level and thread-level parallelism jointly to improve instruction throughput. The interfaces between the IoT network periphery and the sensors and actuators for instance are defined by systems -on-a-chip(SoC) devices. Chip multiprocessors (CMP) and Simultaneous Multithreading (SMT) allows effective resource utilization and thus potential to achieve higher levels of instruction throughput. This concept and its few challenges have been taken into consideration. A solution to some challenges regarding Multithread RISC Architecture based on Programmable Interleaved Pipelining is taken. The process of multithreading in *T processor to improve computation and long-latency memory operation is involved. In a processor core with hardware multithreading support (MTP), several threads can be executed concurrently. The threads that are not stalled compete for the processing resources such as ALU or memory interfaces. So, a brief idea about resolving this problem by selecting a thread whose instructions will be executed next is taken into consideration.

A new loop scheduling scheme called multithreaded self-scheduling (MY$) for distributed shared memory multiprocessor is explained. A simultaneous multithreading multicore processor which accelerate object recognition for 720p HD video streams is discussed. Resiliency is a first order design concern in modern microprocessor designing. Compiler-level Redundant Multithreading schemes are promising because of their compatibility to detect hardware transient and permanent faults. EXPERT, a compiler-level RMT scheme can detect manifestation of hardware faults in all hardware components is discussed.

For Java-Web computing and many Java applications, a Java processor called simultaneous multithreaded Java chip which enhances performance of previous Java Processor by hardware support of Java multithreading. SMT Java chip is an advanced architecture which has enhanced mechanism of stack cache, instruction cache, functional units etc. Simultaneous multithreading is a well-known technique that improves resource utilization by exploiting thread level parallelism at the instruction gain level. A method called Cluster-level Simultaneous Multithreading(CSMT) considers the set of operations that execute simultaneously in a given cluster as the assignment unit. So, the benefits of CSMT regarding VLIW (Very Long Instruction Word) is discussed. Thus it exposes architecture details to the compiler, so that ILP can be extracted at compile time. Thus most multithreading concepts, therefore are targeted to minimize vertical waste that arises due to memory latency cycles in a program.

2. Literature survey

Experiments with sorting hardware implementation showed many drawbacks and the delay was unacceptable. CPU depends on continues processing of information in the pipe. It allows the processor to do useful computation during long-latency memory operations. The primary transmitter instruction is sttx, store into transmit registers. The transmitter registers are a bank of 24 32-bit registers. The proposed method achieves the desired thread differentiation without hindering performance or increasing costs, as demonstrated by initial experimental results.

This delay is imposed by the increasing gap between the

processor cycle frequency and the memory and communication latency. Software multithreading can reduce the stalls but with coarser response times. With hardware multithreading and the appropriate scheduling, the average utilization of the computational units increases. The execution time requirements of threads may vary depending on several issues. It has lower priority.

A new loop scheduling scheme called multithreaded self-scheduling (MY$) for distributed shared memory multiprocessor is proposed. Attempts to hide the remote memory access latencies by switching be- tween multiple contexts of threads. A series of simulation results corresponding to various parameter changes arc presented which provides o measure of the effectiveness of MSS.

One of the characteristics of DSM multiprocessor is Non uniform memory Access(NUMA). Offers a favourable programming paradigm of a global address space for parallel programs such that concurrent executing pro- gram. In this paper, a new loop scheduling technique €or DSM multiprocessor is proposed. By multithreading the chunks in guided self-scheduling (GS) scheme, the remote memory access latencies, that frequently happen in DSM multiprocessor, may be effectively hidden by switching between multiple contexts of threads. It has capability to detect the manifestation of hardware transient and permanent faults. Thread-level error detection schemes. If there is no mismatch, the leading thread proceeds and submits the results, i.e., writing data to the shared memory. EXPERT, a compiler-level fault detection scheme which provides microprocessor-wide transient and permanent fault detection. EXPERT significantly improves the error coverage of state-of-the-art RMT schemes by providing full execution protection rather than just computational operations protection.

The disadvantage of fail to detect hardware permanent faults. Faults in the simulated processor while running original and protected versions of programs. An SMT Java chip is proposed to enhance hardware java multithreading. To enhance the performance of java processor considering java characteristics using hardware multithreading. Performance results show that execution is speeded up between 1.3 to 2.000 compared to single threaded java processor. The processor uses many clock cycles to speed up the execution. Combination of all of the multithreading paths comprises a total data processing system capable of handling more than minimum task for which highest reliability is required.

The inverse proportionalities between speed, capacity and reliabilities can be exploited in a multithreaded computer organization. Utilizes activity redundance to provide high capability for more extensive data processing tasks during no failures have occurred. Opportunities for tradeoffs is re-examined and a reliability model is designed. The rate at which the signals is processed and the number of arithmetic and logical operations which can be processed is multiplexed. An extra memory module and additional processors have been added to speed up the process. 20000 words have been processed easily even they are too complex. The computer system uses single threaded configuration. The processors used are more in number. It is dependent on probability rather than accuracy.

The processor is basically a multiple instruction, multiple data stream (MIMD) digital computer that utilizes pipelining for control and function units. This invention relates to data processing, and, more particularly, relates to concurrent task and instruction processing in a multiple instruction, multiple data stream computer. computers became faster, the utilization of other resources became the limiting factor. Resources included the adder, multiplier, and other function units which make up the central processing unit. Program instructions could be fetched, or pulled, from the memory unit much faster than the function units. This technique has been heretofore employed to speed up function units, such as an added. By the use of one central processing unit which is, in effect, multiplexed among the several data streams. The main problem with the use of separate control processing. If several tasks are to be processed concurrently, then the PSWs for each task must be distinguished from those for others as regards task memory allocation.

A data processing system having a plurality of data. The functional groups of dedicated registers when activated to share the common resource circuits, define a plurality of data processors. The processors execute programs, wherein each processor when active performs unique data processing functions operationally independent of the other processor. The resource allocation circuit includes a priority network that receives real time common resource utilization requests from the processors. The software design approach for I/O data processing systems is based on time sharing principles that allow individual data processing tasks to share a common memory. Multiple concurrent program executions can be performed at the expense of further hardware duplication. Response time is minimized in the true hardware I/O oriented data processing system.

The system also is not rendered inoperable because of the failure of a single processor to respond. Performance of individual data processing tasks is decreased by that amount of time required to read and execute the program interrupt instructions. Time sharing of common resource circuits under a software oriented approach requires program interrupt instructions and routines or polling to effect switching operations from one data processor to the next.

3. Conclusion

From this survey, we will be able to know the way in which hardware multithreading in different processors is successfully implemented. The overall objective of the present survey is to provide a method and apparatus for scheduling the execution of multiple tasks in a computer system which is greatly advancing. The ultimate aim to combine our approach with existing

performance-boosting and cost-reducing techniques. The next works are to combine our approach with existing performance-boosting and cost-reducing techniques. A dynamic power control method of hardware multithreading for real-time embedded system processors to improve energy efficiency and the performance of the active threads and eliminate energy wasted by idle threads by improving energy efficiency when the processor workload decreases is to be achieved in future works. The CSMT concept has become a landmark which helps to reduce error accumulation effect. All bundles belonging to a VLIW instruction from a given thread are simultaneously issued. The main part been implemented in Verilog, while the interleave controller and sequence patterns selection algorithms are coded in VHDL is to be developed further. A method for modelling chip multithreading processors architecture is introduced, and an instruction level simulation model is implemented as a set of extensions to the Simics machine simulator toolkit. EXPERT, a compiler-level RMT soft/hard error detection scheme. EXPERT significantly (~65x) improves the error coverage of state-of-the-art RMT schemes by providing full execution protection rather than just computational operations protection.

References

[1] R. P. Hassett and E. H. Miller, 1966, Supplement to IEEE Transactions on Aerospace and Electronic Systems.
[2] Gary L. Ferguson, Maxwell, C. Gilligand, Concern Task and Processor Method, 1971 United States Patent.
[3] Donald H. MALCOM, Multi-processor data processing system,1975, United States Patent.
[4] Alton B. Otis, Peter E Forsman, Multi- processor for simultaneous executing a plurality of program in a time-interlaced manner, 1980, United States Patent.
[5] Phillip H. Sherrod, Method and operation for scheduling by execution of multiple processing task in computer system, 1987, https://patentimages.storage.googleapis.com/ce/dd/d4/ff9bacd142cb61/US4642756.p
[6] Brian W. Case, Rod G Fleck, System for processing single cycle branch instruction in a pipeline having relative, absolute indirect, and trap address, https://patentimages.storage.googleapis.com/c6/52/2d/2f14dd31380ca8/US4777587
[7] Thomas E. Wheatley, Requirements for implementing real time control modules on a hierarchical parallel pipelining system,1989. https://ntrs.nasa.gov/archive/nasa/casi.ntrs.nasa.gov/19900020575.pdf
[8] Michel J. Beckerle, Overview of START [*T] multithreaded computer, 1993.
[9] K P Hung, N H C Yung, Multithreaded self-scheduling application of multithreading on loop scheduling for distributed memory multiprocessor, 1995.
[10] Bernard K. Gunther, Multi-threading with distributed functional units, 1997, IEEE Transactions On Computers.
[11] Chun mok Chung, Shin Dug Kim, Dual threading java multithreading, 1998, Korea Science Foundation Interdisciplinary Grant.
[12] Andreas Doring, Maria Gabnani, on networking multi -threading processor design: hardware threaded prioritization, 2004, IEEE Transactions on Computers.
[13] Lei Wang, Niral patel, Reducing common accumulation effect in multi-threaded memory system, 2005, ISCAS.
[14] Sheo Wang, Lei Wang, Thread associate memory for multi core and multi-threading computer, 2007, ISLPED.
[15] Andrei Pulka, Adam Milik, Multithreaded RISC architecture based on programable interleaved pipelining, 2009, Second International Conference On Computer Modelling and simulation.
[16] James Warnock, Verwin Behnan, Circuit design technology for 1G cell broadband engine processor,2006, IEEE Journal of Solid-State Circuits.
[17] Yasmin Syuhada Harmin, Razali Jidin, Enabling multithreading executions on the XILINX Microkernel with a hardware schedular, 2008, International Conference on Electronic Design.
[18] Mysore Satyanarayana, Bret Ronald Olszewski, Measuring Processor use in a Hardware Multithreading Environment, 2009, United States Patent.
[19] Zhifeng zhang, Geo chun Wan, Modeling and simulation of chip multi-threading processer architecture, 2010, Second International Conference on Computer Modeling and Simulation.
[20] Manoj Gupta, Joseph Kosa, CSMT: Simultaneous multithreading for cluster VLIW processor, 2011, IEEE Transactions on Computers.
[21] Janwook oh, Injaon Hong, "A simultaneous multithreading heterogenous object recognition processor with machine learning based dynamic resource management, 2012, IEEE Cool Chips XV.
[22] Luo Xing Qiang, Wang Lei, Dynamic power dissipation control method for real time processes based on hardware multi -threading,2013, China Communications.
[23] Peter M. Maurer, Mapping the data flow model of computation into an enhanced Von Neumann processor," 2015.
[24] Tobias Strauch, connecting things to IoT using virtual peripherals on dynamic multithreading cortex M3, IEEE transactions on circuits and systems, 2017.
[25] Hwisoo So, Moslem Didehban, EXPERT: Effective and Flexible Error Protection by Redundant Multithreading, Design, Automation and Test in Europe, 2018.