International Journal of Computer Engineering and Information Technology

VOL. 9, NO. 8, August 2017, 181–185
Available online at: www.ijceit.org
E-ISSN 2412-8856 (Online)

Parallel-Processing: A Comprehensive Overview of Modern Parallel Processing Architectures

Ibtisam Rauf[1] and Abdul Majeed[2]

[1] Department of computer science and information technlogy, Virtual University of Pakistan,Islamabad, Pakistan

[2] Department of information and electronics engineering, Korea Aerospace University, Goyang, South Korea

[1]ms130400028@vu.edu.pk, [2]abdulmajid09398@kau.kr

ABSTRACT

This paper presents the core concepts and different architectures of the parallel processing in computer systems. Parallel processing is most commonly used technology in today's computer systems to speed up the computation and data processing power. We present the detailed overview of the different parallel processing architectures and their working in detail. In this paper, we discussed about seven parallel processing architectures, such as, SIMD, MISD, MIMD, vector processing, SMP, NUMA and cluster systems which can be used in our computer systems to strengthen the computation and processing power. We also discuss the key factors of each architecture and situations when to use the given architectures. The one-unit discussion of all architectures provides the concise overview of them and their role in computer organization processing.

Keywords: *Parrallel-Processing, Computer-Systems, Architecture, Computation, Technology.*

1. INTRODUCTION

Parallel processing [1] is a technique in which data and instructions are manipulated simultaneously by the computer machine. Parallel processing reduces the total turnaround time of these instructions but it increases the CPU time due to increase in demand of the memory due to duplication of instructions. Similarly coding the parallel instructions are more complicated than writing instruction in sequential manner. There can be different architectures or different models which can be useful for different applications. Now it depends on the characteristics of the application that which architecture is more suitable for the making of application and which parallel hardware should be used for application. A stream is a instruction set to be executed by the CPU. SISD: It is a computer architecture [2] in which there is only single instruction cycle and instruction are fetched in sequential manner into single central processing unit before execution. SIMD: It is multiprocessor architecture in which there is only one instruction cycle but multiple sets of instructions are fetched to the multiple central processing units and are executed simultaneously in a single instruction cycle. MISD: it is parallel processing architecture in which many functional components perform different functions on the same data. MIMD: It is also a multiprocessor architecture in which multiple instruction sets are active at the same time and each cycle fetch instructions independently at the same time and perform operation on the instructions concurrently on multiple central processing units.

2. CATEGORIZATION OF PARALLEL PROCESSOR ARCHITECTURE

2.1 Single Instruction Single Data (SISD) Architecture

Single instruction single data is computer architecture [3] in which there is only one processor which executes a single instruction set, data is stored in a single memory chip. SISD can be used for parallel processing; it has the characteristics of concurrent processing. Pipelined execution of instructions is an example of SISD computers. An SISD modern computer has:

- A uniprocessor (A type of serial computer)
- CPU act on a single instruction stream in any give instruction cycle
- Only single data stream can be used as input in one clock cycle
- Deterministic implementation
- Found in oldest type of computers although can be available in today's computers

- Today's personal computers and old mainframe computers are example of the SISD computer architecture.

2.2 Single Instruction Multiple Data (SIMD) Architecture

Single instruction multiple data is second computer architecture. It has many multiple processing elements that perform same operation on multiple data elements simultaneously. SIMD are useful for multimedia applications. Modern CPUs have SIMD instructions to improve the performance of multimedia applications. SIMD architecture provides the parallelism but not concurrency because in SIMD on single instruction stream is active at the same time not multiple instructions streams are active. CPU executes the different data items of a single instruction stream.

- A type of parallel computer
- All processing units have the same instruction set working on the different data elements of that instruction set at a given clock cycle
- Can be useful for the application where graphics are included such as image processing
- Synchronous execution
- It has two varieties i.e. Array Processor and Vector Processor
- IBM 9000, connection machine CM-2, Cray X-MP are some examples of SIMD architecture
- Graphics processor units used the SIMD architecture and instruction units

While using SIMD architecture the data is executed in form of blocks and multiple values can be loaded into the main memory at once. For example, while working with graphics we can load n number of pixels into the main memory at the same time instead of loading one pixel in main memory and then the next pixel will be loaded into the main memory. Similarly, SIMD include only those data points from an instruction set on the operation can be perform at once.

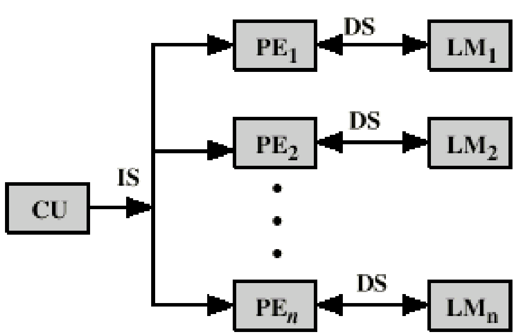

Fig 1. Single Instruction Multiple Data (SIMD) architecture

2.3 Multiple Instruction Single Data (MISD) Architecture

Multiple instruction Single data architecture [4] uses various functional components to perform different operations on same data. Pipelining is an example of this architecture where at each stage of the pipeline data is different after processing. Computers having this type of architecture have high fault tolerance where the tasks are replicated i.e. Same instructions are executed redundantly to detect errors. Example of MISD is space shuttle flight control computers.

- Use for parallel computing
- Every CPU use separate instruction streams to operate on data independently
- Single data stream given to multiple CPUs
- Example of parallel computer Carnegie Mellon computer 1971
- Several frequency filters working on single signal stream
- Several cryptography algorithms trying to break a single coded message

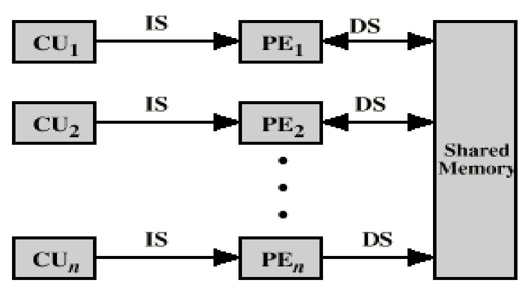

Fig 2. Multiple Instruction Single Data (MISD) architecture

2.4 Multiple Instruction Multiple Data (MIMD) Architecture

Multiple Instruction Multiple data architecture uses many processors to work asynchronously and independently. At same time many processors executes many instructions on different data elements. Computer Aided Design (CAD), [5] Simulation and modeling etc are some applications of MIMD architecture. MIMD processors either used shared memory or distributed memory types. MIMD processors which used shared memory type may be hierarchical type or bus based or extended. Distributed memory MIMD processors use mesh interconnection systems. A multi core processor is an example of MIMD processor.

- A kind of parallel computer
- Each CPU executes the different instructions
- Every CPU works on different data elements
- Execution can be in synchronous or asynchronous mode, deterministic or non-deterministic mode
- Mostly supercomputers use the MIMD processors

- Supercomputers, networked computers, SMP computers or multi core computers are examples of MIMD architecture

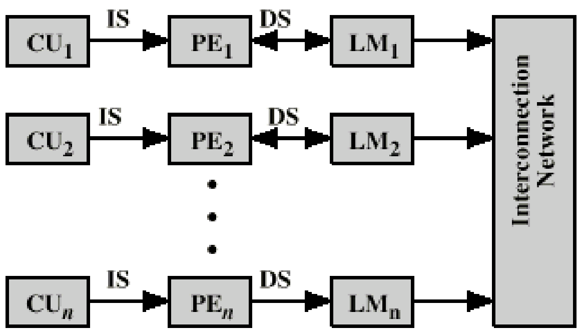

Fig 3. Multiple Instruction Multiple Data (MIMD) architecture

Difference between SIMD and MIMD

- The amount of hardware used in SIMD is less than the amount of hardware used in MIMD architecture because in SIMD there is only one control unit while in MIMD there are multiple controls units
- SIMD computers used less amount of memory as compared to the MIMD computers because exactly one copy of data stored in memory. while in MIMD computers multiple copies of data are store at different processors
- SIMD computers required short time and effort to be built from inexpensive of the shelf components. While MIMD computers required larger effort and more time because of their specialized hardware architecture, design control and application characteristics.

2.5 Vector Processor/Array Processor

In vector processor or array processor [5], [6] architecture processors work with the array of data. Array of data has two indices. Vector processors work with vector instructions. A single vector processor works on the multiple instructions with the use of pipelining.

- The result of the next pipeline is independent of the result of previous pipeline. It allows very deep pipeline without data hazards.
- With the single instruction, a single vector process can do tremendous amount of work just like executing the entire loop at the same time. It reduces the requirement of instruction bandwidth.
- The behavior of the vector instruction is predetermining so there is no possibility if control hazards. Here we represent while loop in the form of vector instruction.
- Vector processor includes two types of architectures: vector register processor and memory-memory vector processor. Vector register processors include all the operations expect load and store among vector registers. Memory-memory vector processor includes all operations memory to memory.

  Vector processor provides very high clock rates as in vector processor each result is independent pf the previous result in pipeline. Vector processor accesses the data from memory the form of block at a time so memory latency amortized over multiple data elements. Vector instructions use specific patterns to access the memory blocks simultaneously. Vector processor takes less time in accessing memory block which means that it provides the faster processing time.

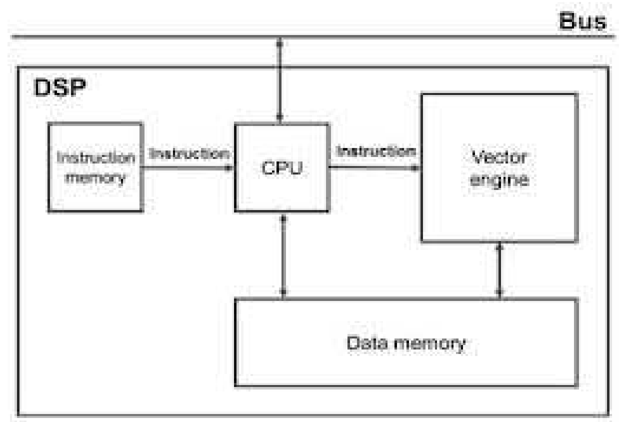

Fig. 4. Vector processing architecture

2.6 Symmetric Multiprocessing architecture

Symmetric multiprocessing architecture has the hardware which supports the multiple processors to be used in a single computer. In SMP two or more central processing units are connected together and share a single main memory. Processors are controlled by a single operating system instance. Today's most commonly used multi core CPUs use SMP architecture which treats them as a separate processor. These multi core processors can be interconnected with the help of a single bus or on chip mesh networks. Use of single bus architecture cannot handle the issues of power consumption and bandwidth. Mesh architecture solve the issues of power consumption and bandwidth and provide linear scalability.

In SMP architecture tasks are performed in parallel. All the processor can perform the same function so in case of failure of a processor system will not be halted or stopped. User can easily enhance the performance of their systems by attaching more processors with their system as SMP support scalability.

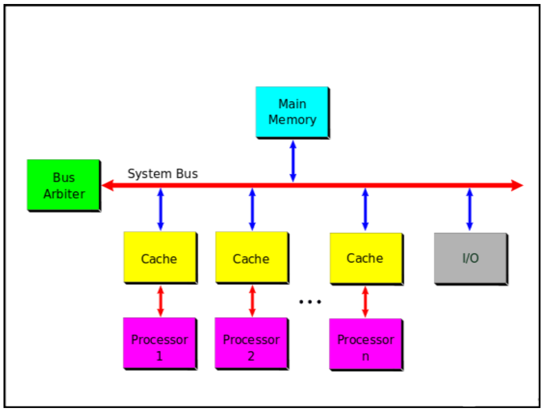

Fig. 5. Symmetric Multiprocessing (SMP) architecture

2.7 Non-Uniform Memory Access architecture

Non-uniform memory access architecture also used in multiprocessor systems. It is just like SMP architecture but instead of providing the shared memory to all processors NUMA provides separate memory for each of the processor. NUMA architecture uses some additional hardware or software to transfer data elements among memory blocks. Processors are loosely coupled in NUMA architecture as there is a separate memory area for each of the processor and each processor work with its local memory area instead of non-local memory which enhance the overall speed of the systems.

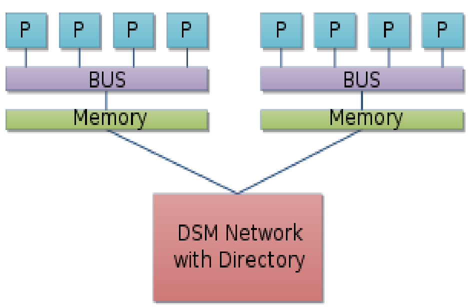

Fig. 6. Non-Uniform Memory Access (NUMA) Architecture

2.8 Cluster System Architecture

Cluster system architecture [6] provides distributed environment in which multiple stand-alone computers are connected together and work together as a single unit, as a integrated processing resource. Cluster provides fastercommunication links and has lower latency communication protocols. It is responsible for the availability of a CPU and helps to improve the performance of the system. Cluster systems used the centralized management approach to makes the node available for processing. Cluster systems are cost effective and eliminate complexity of procuring. We can easily attach and remove hardware or software resources in our cluster system to meet the requirements of processing. We can run task anytime anywhere by launching tasks using APIs (application program interface) tools. We can access our computer resources within some minutes instead of wasting time in waiting in queues.

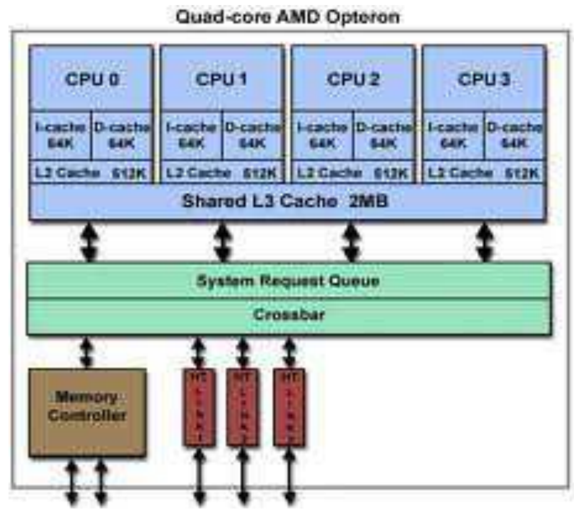

Fig. 7. Cluster system Architecture

3. CONCLUSION

In our research paper, we have discussed about different computing architecture we can use in our computer systems. We discussed that how these architectures can be useful in parallel processing. We discussed about the SIMD (single instruction multiple data) and MIMD (Multiple instruction multiple data) and also draw a comparison between these two architectures. We also talked about the vector processor, SMP Symmetric multiprocessing, NUMA (non-uniform memory access) and cluster systems. The detailed discussion of the above-mentioned architectures would assist designers and system users in selecting the most appropriate architectures according to their needs.

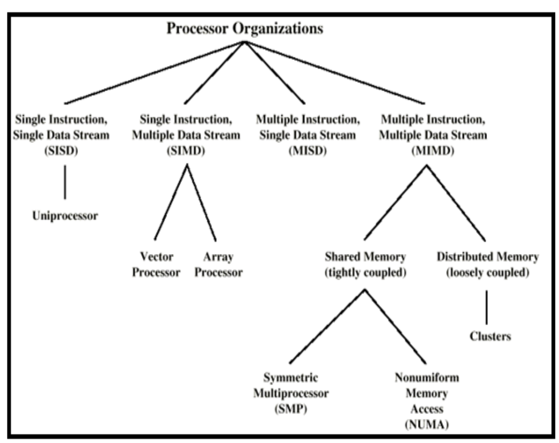

Fig. 8. Parallel Processors Architectures

REFERENCES

[1] Introduction to Parallel Processing, Cornell Theory Center Workshop, 1997.

[2] O. Serlin, The Serlin Report On Parallel Processing, No.54, pp. 8-13, November 1991.

[3] Advances in Parallel Computing from the Past to the Future, Dr.P Sammulal, Volume 1, Issue 4, September 2013

[4] http://www.cse.unsw.edu.au/~cs4211/04s1/seminars/rompo.pdf

[5] http://groups.engin.umd.umich.edu/vi/w3_workshops/VI_wi nter04_ganesan.pdf

[6] https://docs.oracle.com/cd/A91202_01/901_doc/rac.901/a89867/pshwarch.htm.

AUTHOR PROFILES:

I. Rauf did his BS (2013) in Information Technology from University Institute of Information Technology, PMAS AAUR Rawalpindi, Pakistan and his MS (2016) in Database from Virtual university of Pakistan. He is currently working as SSE (CS) in Government Education Department.

A. Majeed did his BSc (2013) in Information technology from University institute of information technology, Rawalpindi, Pakistan and his MS (2016) in Information Security from COMSATS Institute of Information Technology, Islamabad, Pakistan. He worked as Security Analyst at Trillium Information Security Systems Organization, Pakistan from 2015 to 2016. He is currently pursuing his PhD from Korea Aerospace University, South Korea. His research includes data privacy, information security, data mining and machine learning.