"COMPARATIVE STUDY OF ADVANCE COMPUTER ARCHITECTURE/PARALLEL PROCESSOR"

[1]AKASH THAKARE
H.V.P Mandal's, P.G. Department of Computer Science and Technology, Amravati, India
akashthakare007@gmail.com

[2]NIKHIL ROHANKAR
H.V.P Mandal's, P.G. Department of Computer Science and Technology, Amravati, India
rohankar.nikhil@gmail.com

[3]PROF V. S. BELSARE
H.V.P Mandal's, P.G. Department of Computer Science and Technology, Amravati, India
vsbelsare@gmail.com

ABSTRACT: *In this paper, various parallel architectures are discussed, which are based on the classification of parallel computers considered earlier. The two major parametric considerations in designing parallel computer architecture are: (i) executing multiple number of instructions in parallel, and (ii) increasing the efficiency of processors. There are various methods by which instructions can be executed in parallel and parallel architectures are based on these methods of executing instructions in parallel. Pipelining is one of the classical and effective methods to increase parallelism where different stages perform repeated functions on different operands. Vector processing is the arithmetic or logical computation applied on vectors whereas in scalar processing only one data item or a pair of data items is processed. Parallel architectures have also been developed based on associative memory organizations. Another idea of improving the processor's speed by having multiple instructions per cycle is known as Superscalar processing. Multithreading for increasing processor utilization has also been used in parallel computer architecture. All the architectures based on these parallel-processing types have been discussed in detail in this paper.*

1. INTRODUCTION

Computer architects have always strived to increase the performance of their computer architectures. High performance may come from fast dense circuitry, packaging technology, and parallelism. Single-processor supercomputers have achieved unheard of speeds and have been pushing hardware technology to the physical limit of chip manufacturing. However, this trend will soon come to an end, because there are physical and architectural bounds that limit the computational power that can be achieved with a single-processor system. Parallel processors are computer systems consisting of multiple processing units connected via some interconnection network plus the software needed to make the processing units work together. There are two major factors used to categorize such systems: the processing units themselves, and the interconnection network that ties them together. The processing units can communicate and interact with each other using either shared memory or message passing methods. The interconnection network for shared memory systems can be classified as bus-based versus switch-based. In message passing systems, the interconnection network is divided into static and dynamic. Static connections have a fixed topology that does not change while programs are running. Dynamic connections create links on the fly as the program executes. The main argument for using multiprocessors is to create powerful computers by simply connecting multiple processors. A multiprocessor is expected to reach faster speed than the fastest single-processor system. In addition, a multiprocessor consisting of a number of single processors is expected to be more cost-effective than building a high-performance single processor. Another advantage of a multiprocessor is fault tolerance. If a processor fails, the remaining processors should be able to provide continued service, albeit with degraded performance.

2. FLYNN'S TAXONOMY OF COMPUTER ARCHITECTURE

The most popular taxonomy of computer architecture was defined by Flynn in 1966. Flynn's classification scheme is based on the notion of a stream of information. Two types of information flow into a processor: instructions and data. The instruction stream is defined as the sequence of instructions performed by the processing unit. The data stream is defined as the data traffic exchanged between the memory and the processing unit. According to Flynn's classification, either of the instruction or data streams can be single or multiple. Computer architecture can be classified into the following four distinct categories:

. Single-instruction single-data streams (SISD);
. Single-instruction multiple-data streams (SIMD);
. Multiple-instruction single-data streams (MISD); and
. Multiple-instruction multiple-data streams (MIMD).

Conventional single-processor von Neumann computers are classified as SISD systems. Parallel computers are either

SIMD or MIMD. When there is only one control unit and all processors execute the same instruction in a synchronized fashion, the parallel machine is classified as SIMD. In a MIMD machine, each processor has its own control unit and can execute different instructions on different data. In the MISD category, the same stream of data flows through a linear array of processors executing different instruction streams. In practice, there is no viable MISD machine; however, some authors have considered pipelined machines (and perhaps systolic-array computers) as examples for MISD.

3. LOOSELY COUPLED AND TIGHTLY COUPLED MULTIPROCESSOR SYSTEM

3.1 Loosely Coupled

Multiprocessor is one which has more than two processors in the system. When the degree of coupling between these processors is very low, the system is called loosely coupled multiprocessor system. In loosely coupled system each processor has its own local memory, a set of input-output devices and a channel and arbiter switch (CAS). We refer to the processor with its local memory and set of input-output devices and CAS as a computer module.

Processes that execute on different computer modules communicate with each other by exchanging the messages through a physical segment of message transfer system (MTS). The loosely coupled system is also known as distributed system. The loosely coupled system is efficient when the processes running on different computer module require minimal interaction.

If the request for accessing MTS of two or more computer module collides, the CAS responsibly chooses one of the simultaneous requests and delays other requests until selected request is serviced completely. The CAS has a high-speed communication memory which can be accessed by all the processors in the system. The communication memory in CAS is used to buffer the transfers of messages.

3.2 Tightly Coupled Multiprocessor System

The throughput of the loosely coupled system may be too low for some of the applications that require fast access time. In this case, Tightly coupled microprocessor system must be used. The tightly coupled system has processors, shared memory modules, input-output channels. The above units of the tightly coupled system are connected through the set of three interconnection networks, processor-memory interconnection network (PMIN), I/O-processor interconnection network (IOPIN) and the interrupt-signal interconnection network (ISIN).The use of these three interconnection networks is as follow.

PMIN: It is a switch which connects each processor to every memory module. It can also be designed in a way that a processor can broadcast data to one or more memory module.
ISIN: It allows each processor to direct an interrupt to any other processor. IOPIN: It allows a processor to communicate with an I/O channel which is connected to input-output devices.

3.3 Pipeline Processor

A pipeline computer performs overlap computations to exploit temporal parallelism. Normally the process of executing an instruction in a digital computer involves four major steps: instruction fetch (IF) from the main memory; instruction decoding (ID), identifying the operation to be performed; operand fetch(OF), if needed in the execution; and then execution(EX) of the decoded arithmetic logic operation. In a non pipelined computer, these four step must be completed before the next instruction can be issued. In a pipelined computer, successive instructions are executed in an overlapped fashion. An instruction cycle consists of multiple pipeline cycles. A pipeline cycle can be set equal to the delay of the slowest stage. The flow of data (input operands, intermediate results, and output results) from stage to stage is triggered by a common clock of the pipeline. In other words, the operation of all stages is synchronized under a common clock control. For the non pipelined (non overlapped) computer, it takes four pipeline cycles to complete one instruction. Once a pipeline is tilled up, an output result is produced from the pipeline on each cycle. The instruction cycle has been effectively reduced to one-fourth of the original cycle time by such overlapped execution. Due to the overlapped instruction and arithmetic execution, it is obvious that pipe me machines are better tuned to perform the same operations repeatedly through the pipeline. Whenever there is a change of operation, say from add to multiply, the arithmetic pipeline must be drained and reconfigured, which will cause extra time delays. Therefore, pipeline computers are more attractive for vector processing, where component operations may be repeated many times. Most existing pipeline computers emphasize vector processing.

3.4 Vector Processor

A Vector processor is a processor that can operate on an entire vector in one instruction. The operand to the instructions is complete vectors instead of one element. Vector processors reduce the fetch and decode bandwidth as the number of instructions fetched is less. They also exploit data parallelism in large scientific and multimedia applications. Based on how the operands are fetched, vector processors can be divided into two categories in memory-memory architecture operands are directly streamed to the functional units from the memory and results are written back to memory as the vector operation proceeds. In vector-register architecture, operands are read into vector registers from which they are fed to the functional units and results of operations are written to vector registers. Many performance optimization schemes are used in vector processors. Memory banks are used to reduce load/store latency. Strip mining is

used to generate code so that vector operation is possible for vector operands whose size is less than or greater than the size of vector registers. Vector chaining the equivalent of forwarding in vector processors is used in case of data dependency among vector instructions. Special scatter and gather instructions are provided to efficiently operate on sparse matrices. Instruction set has been designed with the property that all vector arithmetic instructions only allow element N of one vector register to take part in operations with element N from other vector registers. This dramatically simplifies the construction of a highly parallel vector unit, which can be structured as multiple parallel lanes. As with a traffic highway, we can increase the peak throughput of a vector unit by adding more lanes. Adding multiple lanes is a popular technique to improve vector performance as it requires little increase in control complexity and does not require changes to existing machine code.

The reason behind the declining popularity of vector processors is their cost as compared to multiprocessors and superscalar processors. Vector processors do not use commodity parts. Since they sell very few copies, design cost dominates overall cost. Vector processors need high speed on-chip memory which is expensive.

3.5 Array Computers

Array processor is a synchronous parallel computer with multiple arithmetic logic units, called processing elements (PE) that can operate in parallel in a lockstep fashion. By replication of ALUs, one can achieve the spatial parallelism. The PEs are synchronized to perform the same function at the same time, An appropriate data-routing mechanism must be established among the PEs.

Scalar and control-type instructions are directly executed in the control unit (CU). Each PE consists of an ALU with registers and a local memory. The PEs are interconnected by a data-routing network. The interconnection pattern to be established for specific computation is under program control from the CU. Vector instructions are broadcast to the PBS for distributed execution over different component operands fetched directly from the local memories.

Array processors designed with associative memories are called associative processor. Different array processors may use different interconnection networks among the PEs. For example, Illiac-IV uses a mesh-structured network and Burroughs Scientific Processor (BSP) uses a crossbar network. In addition to Illiac-IV and BSP, we will study a bit-slice array processor called a massively parallel processor (MPP).

3.6 COMPARISON TABLE

Processor	Characteristics				
	Shared Memory	**Local Memory**	**Cache Memory**	**OS**	**Example**
Loosely Coupled		Available	Each process has its own cache.	It operates on single operating system	Beowulf cluster
Tightly Couple	Available		According to need of processing.	It operates on multiple operating system	Zeon
Pipeline Processor	Available	Available	Burst Cache	UNIX	Intel Pentium
Vector Processor	Available	Available	Available	UNIX (VUNIX, BSD UNIX)	Intel x86
Array Processor	Available	Available	Available	Parallel operating system(windows NT)	MPP

4. CONCLUSION

In this, paper we compare different computer architectures, and we conclude that, every processor have some different architecture. The internal organization of every processor is different. Different processors are developed or build for achieving some specific objective. Every architecture has its advantages and some disadvantages.

5. REFERENCES:

[1] Computer System Architecture By Morris Mano

[2] Advanced computer Architecture By Richard Y. Kain

[3] Advanced computer Architecture By Kai Hawang and Naresh Jotwani