A Comprehensive Survey of Various Processor types & Latest Architectures

Harisha M.S[1], Dr. D. Jayadevappa[2]

[1]*Research Scholar, Jain University, Bangalore, Karnataka, India*

[2]*Professor, Electronics Instrumentation, Dept., JSSATE, Bangalore, Karnataka, India*

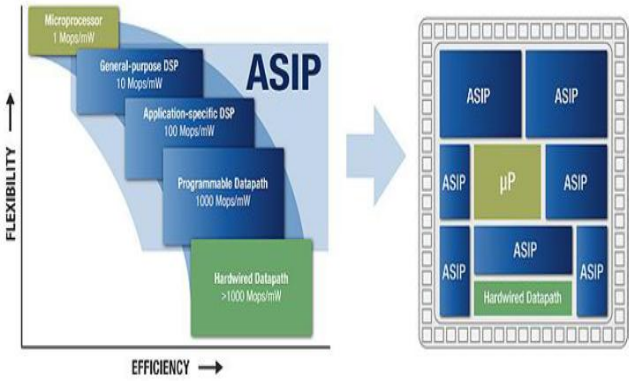
***Abstract*- This technology survey paper covers application based Processors- ASIP, Processor based on Flynn's classification which includes SISD, SIMD, MISD, MIMD, Special processors like Graphics processing unit (GPU), Physics processing unit (PPU), Digital signal processor (DSP), network processor, front end processor, co-processor and processor based on number of cores which includes- single core, multi core, multi-processor, hyper threading and multi core with shared cache processors.**

***Keywords*- ASIP, SISD, SIMD, MISD, MIMD, GPU, PPU, DSP, Network processor, coprocessor.**

I. INTRODUCTION

A. Types of processors- Processor based on application Application-specific instruction-set processor (ASIP)

Application-specific instruction set processor is a component used in system-on-a-chip design. The instruction set of an ASIP is designed to benefit a specific application [1]. This specialization of the core provides a tradeoff between the flexibility of a general purpose CPU and the performance of an ASIC.



. Fig.1 ASIP

B. Processor types based on Flynn's classification

In Flynn's taxonomy, the processors are classified based on the number of concurrent instruction and data streams available in the architecture: Single Instruction Single Data (SISD), Single Instruction Multiple Data (SIMD), Multiple Instruction Multiple Data (MIMD) and multiple instruction single data (MISD) [2].

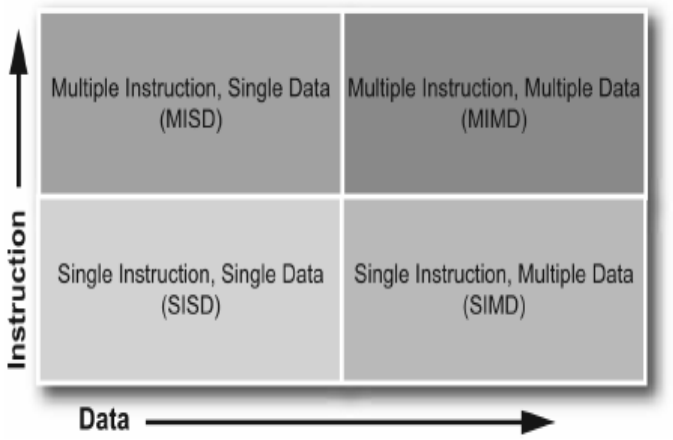


Fig.2 Flynn's classification

SISD & SIMD

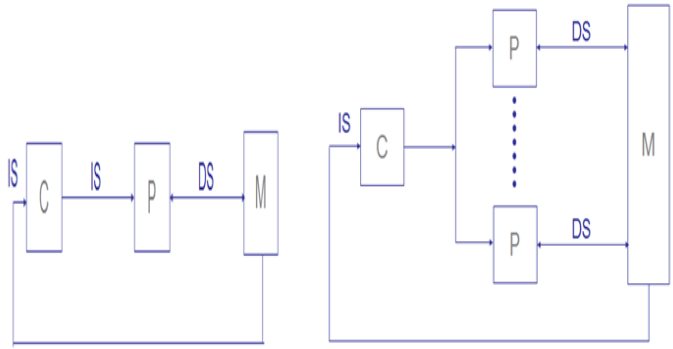


Fig.3 SISD and SIMD

SISD

- SISD is a computer architecture in which a single uni-core processor executes a single instruction stream, to operate on data stored in a single memory.
- A sequential computer which exploits no parallelism in either the instruction or data streams.
- Examples of SISD architecture are the traditional uniprocessor machines like PC or old mainframes [3].

SIMD

- SIMD is a computer with multiple processing elements that perform the same operation on multiple

data points simultaneously. It performs parallel computations, but only on a single instruction at a given moment.

- Example, an array processor [4].

MISD & MIMD

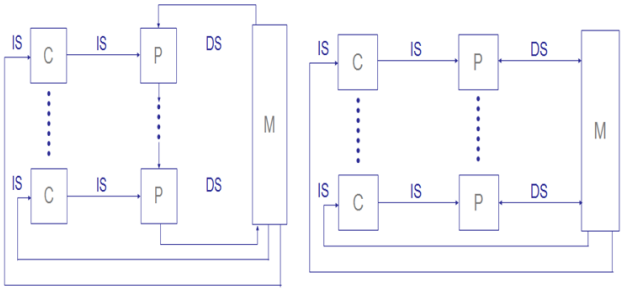


Fig.4 MISD and MIMD

MISD

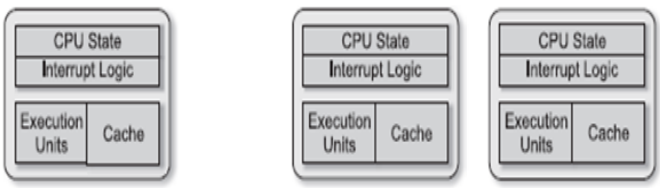
- MISD is a parallel computing architecture where many functional units perform different operations on the same data.
- Architecture which is generally used for fault tolerance.
- Heterogeneous systems operate on the same data stream and must agree on the result.
- Examples include the Space Shuttle flight control computer.

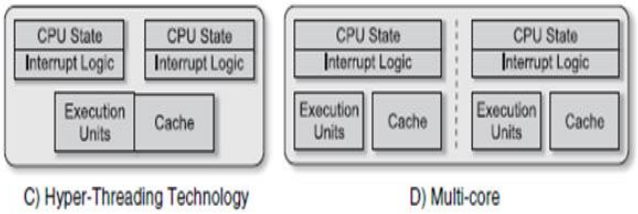
MIMD

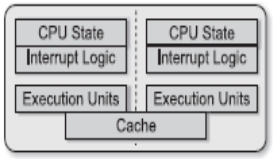
- MIMD machines include a number of processors that function asynchronously and independently. At any time, different processors execute different instructions on different pieces of data.
- Multiple autonomous processors simultaneously executing different instructions on different data.
- Distributed systems are generally recognized to be MIMD architectures; either exploiting a single shared memory space or a distributed memory space [8].

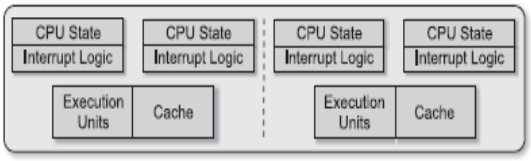
C. Processor classification based on no of cores:

- Single core
- Multiprocessor
- Hyper-Threading
- Multi-core
- Multi-core with shared cache









Single core processor

A single-core processor is a microprocessor with a single core on a chip, running a single thread at any one time.

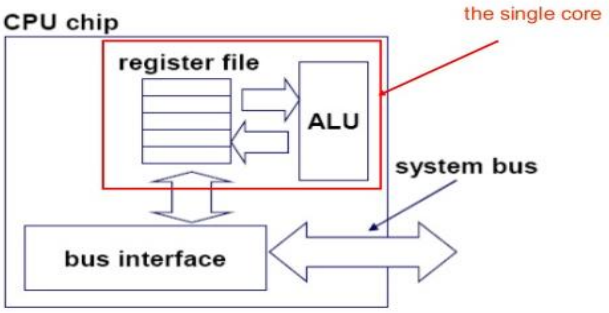


Fig.5 Single core processor

Multiprocessing is the use of two or more central processing units (CPUs) within a single computer system. Multiprocessor is a computer system having two or more processing units each sharing main memory and peripherals, in order to simultaneously process programs [5].

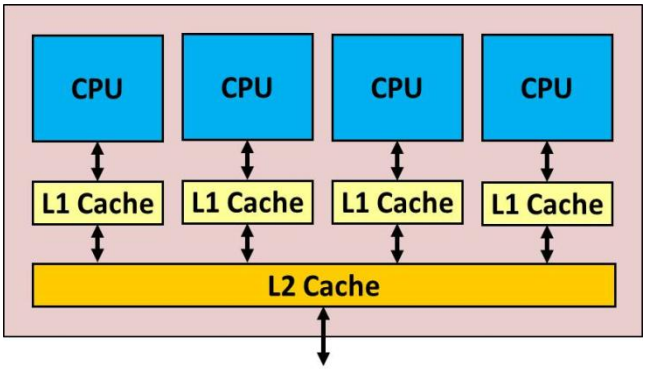


Fig.6 Multiprocessor

Hyper-Threading is a technology used by some Intel microprocessors that allow a single microprocessor to act like two separate processors to the operating system and the application programs that use it. Intel Hyper-Threading technology uses processor resources more efficiently, enabling multiple threads to run on each core. It increases processor throughput thus improving overall performance on threaded software [6].

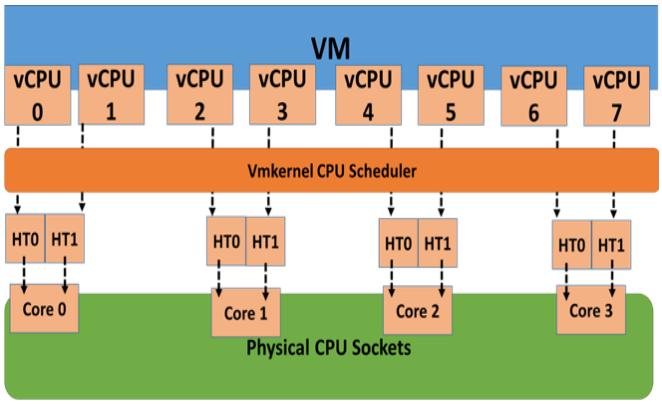


Fig.7 Hyper-threading processor

Multi-core processor is an integrated circuit (IC) which includes two or more processors which results in enhanced performance, reduced power consumption, and more efficient simultaneous processing of multiple tasks. A dual core processor is comparable to having multiple, separate processors installed in the same computer, but because the two processors are actually plugged into the same socket, the connection between them is faster. Practically, dual core processor is about one-and-a-half times as powerful as a single core processor [7].

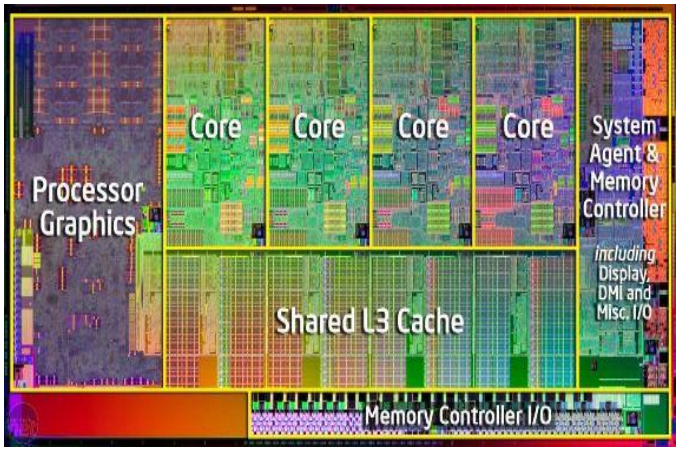


Fig.8 Multi core processor

Multi core processor with shared cache

A CPU cache is a hardware cache used by the central processing unit (CPU) of a computer to reduce the average cost (time or energy) to access data from the main memory. The L3 cache, and higher-level caches, are shared between the cores and are not split.

Advantages

- Reduces cache underutilization since, when one core is idle, the other core can have access to the whole shared resource.
- It offers faster data transfer between the cores.
- It simplifies the cache coherence logic and reduces the severe penalty caused by false sharing.
- It is well suited for facilitating multi-core application partitioning and pipelining.
- The shared cache architecture provides a better performance/cost ratio than dedicated cache.

D. Special Processors

Graphics processing unit (GPU)

Graphics processing unit (GPU) is a specialized electronic circuit designed to rapidly manipulate and alter memory to accelerate the creation of images in a frame buffer intended for output to a display device. GPUs are used in embedded systems, mobile phones, personal computers, workstations, and game consoles. Modern GPUs are very efficient at manipulating computer graphics and image processing, and their highly parallel structure makes them more efficient than general-purpose CPUs for algorithms where the processing of large blocks of data is done in parallel. In a personal computer, a GPU can be present on a video card, or it can be embedded on the motherboard [8].

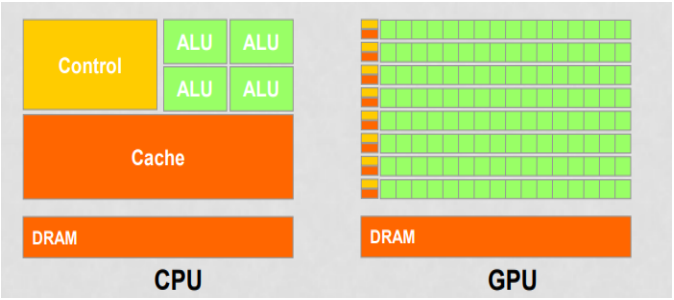


Fig.9 GPU

- A specialized circuit designed to rapidly manipulate and alter memory
- Accelerate the building of images in a frame buffer intended for output to a display

General Purpose Graphics Processing Unit (GPGPU)

- A general purpose graphics processing unit as a modified form of stream processor.
- Transforms the computational power of a modern graphics accelerator's shader pipeline into general-purpose computing power.

GPU architecture

- Generic many core GPU
- Less space devoted to control logic and caches
- Large register files to support multiple thread contexts

- Low latency hardware managed thread switching
- Large number of ALU per core with small user managed cache per core
- Memory bus optimized for bandwidth
- ~150 Gbps bandwidth allows us to service a large number of ALUs simultaneously. The GPU is specialized for high data parallel computation
- More transistors can be devoted to data processing rather than data caching and flow control

Physics processing unit (PPU) is a dedicated microprocessor designed to handle the calculations of physics, especially in the physics engine of video games. Examples of calculations involving a PPU might include rigid body dynamics, soft body dynamics, collision detection, fluid dynamics, hair and clothing simulation, finite element analysis, and fracturing of objects [9].

Digital signal processor (DSP) is a specialized microprocessor, with its architecture optimized for the operational needs of digital signal processing. The DSPs measure, filter or compress continuous real-world analog signals. Dedicated DSPs have better power efficiency and hence they are more suitable in portable devices such as mobile phones. DSPs often use special memory architectures that are able to fetch multiple data or instructions at the same time.

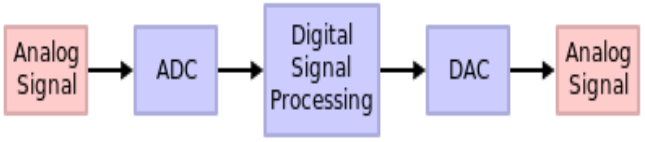


Fig.10 Digital signal processor

Digital signal processing algorithms typically require a large number of mathematical operations to be performed quickly and repeatedly on a series of data samples. Signals are constantly converted from analog to digital, manipulated digitally, and then converted back to analog form. A specialized digital signal processor provides a low-cost solution, with better performance, lower latency, and no requirements for specialized cooling or large batteries [10].

Network processor

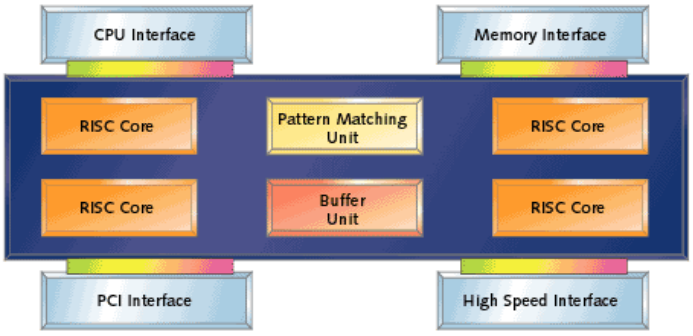


Fig.11 Network Processor

A network processor is a special-purpose, programmable hardware device that combines the low cost and flexibility of a RISC processor with the speed and scalability of custom silicon (i.e., ASIC chips). Network processors are building blocks used to construct network systems. It is specially designed for networking application.

Network processors are typically software programmable devices and have generic characteristics similar to general purpose central processing units that are commonly used in many different types of equipment and products [11].

Front end processor (FEP)

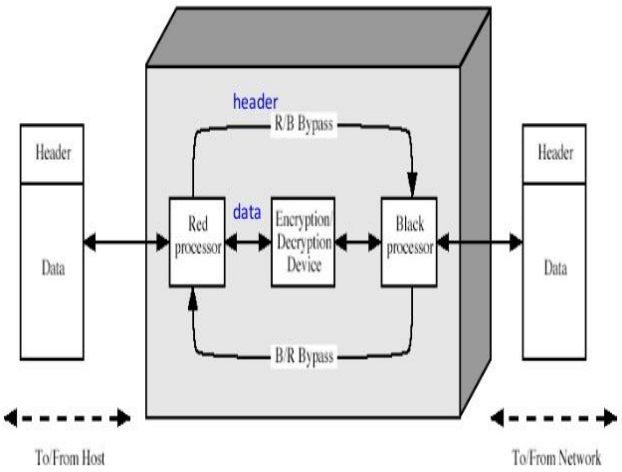


Fig.12 FEP

Front end processor is a small-sized computer which connects networks such as SNA, or peripheral devices, such as terminals, disk units, printers and tape units to the host computer. Data is transferred between the host computer and the front end processor through a high-speed parallel interface.

The front end processor communicates with peripheral devices using serial interfaces through communication networks. The purpose is to off-load from the host computer the work of managing the peripheral devices, transmitting and receiving messages, packet assembly and disassembly, error detection and error correction. Examples include IBM 3705 Communications Controller and the Burroughs Data Communications Processor. It performs tasks such as telemetry control, data collection, reduction of raw sensor data, analysis of keyboard input, etc [12].

Coprocessor is a computer processor that supplements the functions of the primary processor. Operations performed by the coprocessor includes floating point arithmetic, graphics, signal processing, string processing, encryption or I/O interfacing with peripheral devices. By offloading processor-intensive tasks from the main processor, coprocessors accelerate the system performance [13].

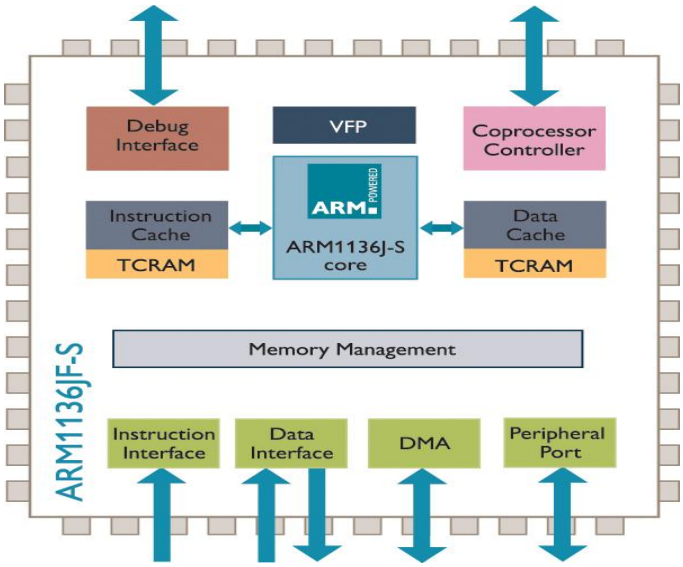
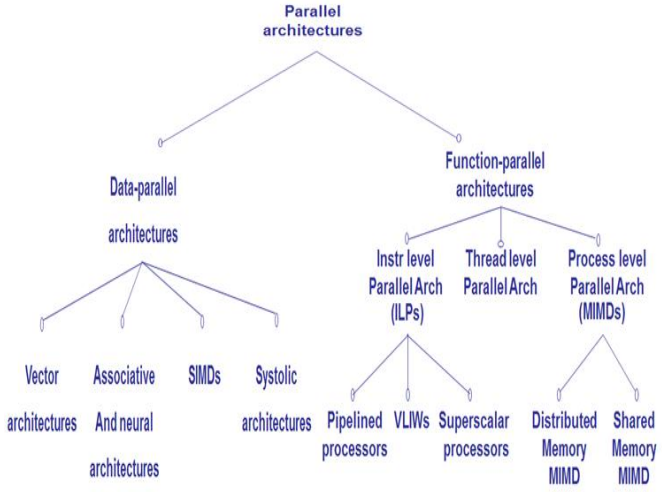


Fig.13 Coprocessor

E. Modern Processor Architecture classification

Data Parallel Architectures

- SIMD Processors
  - Multiple processing elements driven by a single instruction stream
- Vector Processors
  - Uni-processors with vector instructions
- Associative Processors
  - SIMD like processors with associative memory
- Systolic Arrays
  - Application specific VLSI structures



Systolic array processor

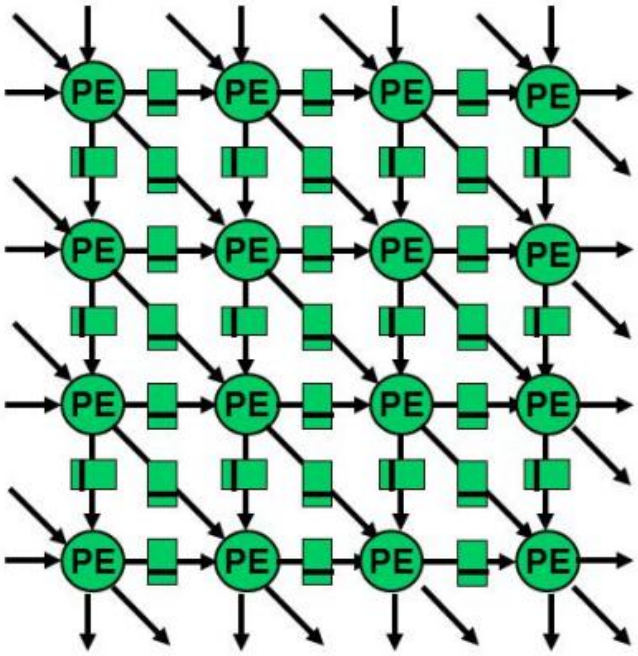


Fig.14 Systolic array processor

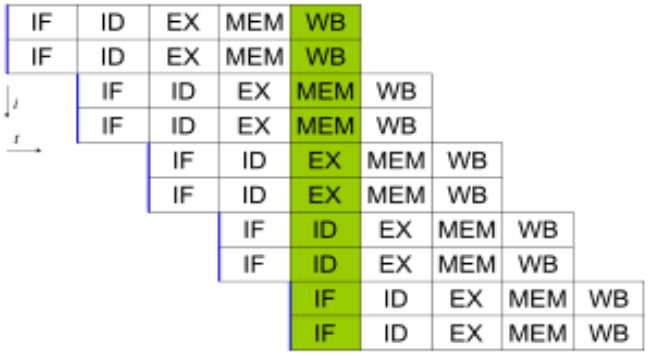
A systolic array is a homogeneous network of tightly coupled data processing units (DPUs) called cells or nodes. DPU independently computes a partial result as a function of the data received from its upstream neighbors, stores the result within itself and passes it downstream. DPU is similar to central processing units(CPU)s, but do not have a program counter, since operation is transport-triggered, i.e., by the arrival of a data object, in an array where data flows across the array between neighbors, usually with different data flowing in different directions [14].

Superscalar processor

Superscalar processor is a CPU that implements instruction-level parallelism within a single processor. A superscalar processor can execute more than one instruction during a clock cycle by simultaneously dispatching multiple instructions and results in high throughput. Characteristics of superscalar technique include:

- Instructions are issued from a sequential instruction stream.
- The CPU dynamically checks for data dependencies between instructions at run time.
- The CPU can execute multiple instructions per clock cycle.
- Superscalar machines issue a variable number of instructions each clock cycle, up to some maximum
- instructions must satisfy some criteria of independence
- simple choice is maximum of one fp and one integer instruction per clock
- need separate execution paths for each possible simultaneous instruction issue

- compiled code from non-superscalar implementation of same architecture runs unchanged, but slower
- Superscalar processing is the ability to initiate multiple instructions during the same clock cycle.
- A typical Superscalar processor fetches and decodes the incoming instruction stream several instructions at a time.
- Superscalar architecture exploits the potential of ILP (Instruction Level Parallelism) [15].



IF = Instruction Fetch, ID = Instruction Decode, EX = Execute, MEM = Memory access, WB = Register write back.

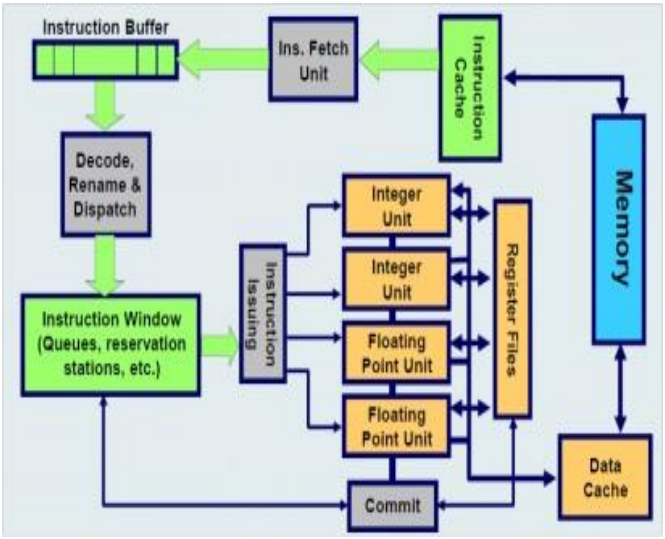


Fig.15 Super scalar architecture

Distributed memory MIMD

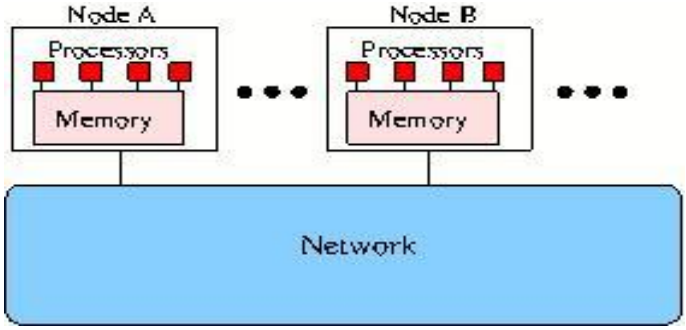


Fig.16 Distributed memory MIMD

- DM-MIMD machine represents the high-performance computers.
- Each processor has its own local memory.
- The processors are connected to each other.
- The demands imposed on the communication network are lower than in the case of a SM-MIMD the communication between processors may be slower than the communication between processor and memory.
- Distributed memory systems can be hugely expanded
- The processors can only access their own memory. If they require data from the memory of another processor, then these data have to be copied [16].

Advantages

- The bandwidth problem that haunts shared-memory systems is avoided because the bandwidth scales up automatically with the number of processors.
- The speed of the memory is less important for the DM-MIMD machines, because more processors can be configured without any bandwidth problems.

Disadvantages

- The communication between processors is slower and hence the synchronization overhead in case of communicating tasks is higher than in shared-memory machines.
- The accesses to data that are not in the local memory have to be obtained from non-local memory and hence it is slow compared to local data access.

Shared memory MIMD

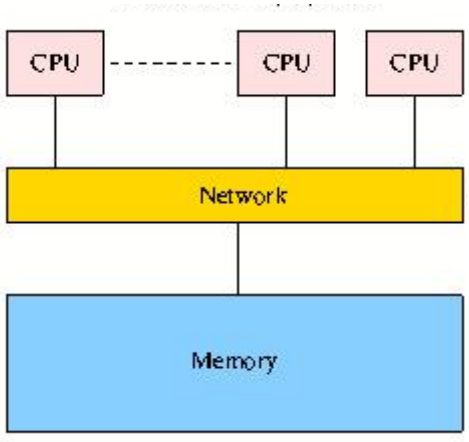


Fig.17 Shared Memory system

In shared-memory MIMD machines, several processors access a common memory from which they draw their instructions and data. Such system is used to perform a common task by executing parts of the programs in parallel there must be some way to coordinate and synchronies various parts of the program. This is done by a network that connects the CPU cores to each other and to the memory [17].

- All processors are connected to a common memory (RAM-Random Access Memory)
- All processors are identical and have equal memory access This is called symmetric multiprocessing (SMP).
- The connection between process and memory is of predominant importance.
- For example: a shared memory system with a bus connection.

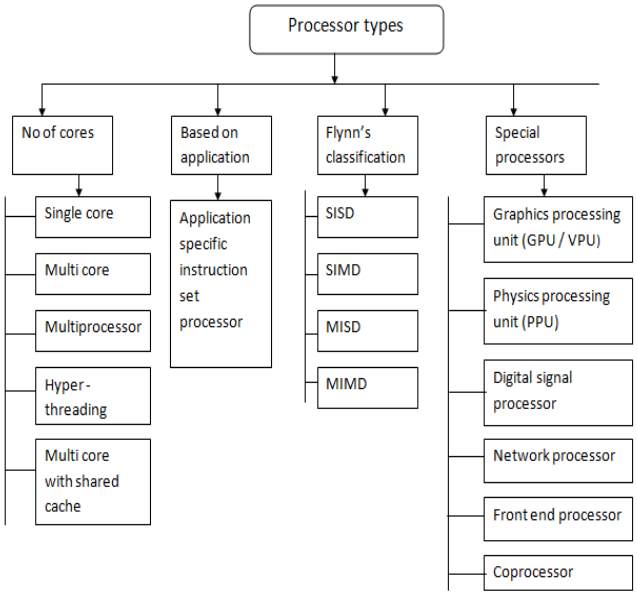
Disadvantage

All processors have to share the bandwidth provided by the bus. To circumvent the problem of limited memory bandwidth, direct connections from each CPU to each memory module are desired. This can be achieved by using a crossbar switch. The problem is their high complexity when many connections need to be made. This problem can be weakened by using multi-stage crossbar switches, which in turn leads to longer communication times. But the number of CPUs & memory modules than can be connected by crossbar switches is limited.

Advantages

- All processors make use of the whole memory. This makes them easy to program and efficient to use.
- The limiting factor to their performance is the number of processors and memory modules that can be connected to each other.
- Shared memory-systems usually consist of few processors.

F. Processor classification tree



II. CONCLUSION

As a part of my ongoing research work on design, development and implementation of multi core hybrid processor- it is necessary to study and understand all existing, contemporary and popular processors and their respective architectures, specifications, feature set and key performance matrices. Many of these parameters needs to be experimented implemented and tested on a common test bed. This survey has also thrown up many new varieties of application specific custom processors which perform with super efficiencies.

REFERENCES

[1]. https://en.wikipedia.org/wiki/Application-specific_instruction_set_processor
[2]. https://en.wikipedia.org/wiki/Flynn%27s_taxonomy
[3]. https://en.wikipedia.org/wiki/SISD
[4]. https://en.wikipedia.org/wiki/SIMD
[5]. https://en.wikipedia.org/wiki/Multiprocessing
[6]. https://en.wikipedia.org/wiki/Hyper-threading
[7]. "Multicore Applications in Real Time Systems" Vaidehi M, T.R.Gopalakrishnan Nair.
[8]. https://en.wikipedia.org/wiki/Graphics_processing_unit
[9]. https://en.wikipedia.org/wiki/Physics_processing_unit
[10]. https://en.wikipedia.org/wiki/Digital_signal_processor
[11]. "Network Processors" Douglas Comer
[12]. https://en.wikipedia.org/wiki/Front-end_processor
[13]. https://en.wikipedia.org/wiki/Coprocessor
[14]. https://www.revolvy.com/main/index.php?s=Systolic%20array&item_type=topic
[15]. https://en.wikipedia.org/wiki/Superscalar_processor
[16]. "Overview of recent supercomputers" Aad J. van der Steen
[17]. https://en.wikipedia.org/wiki/MIMD