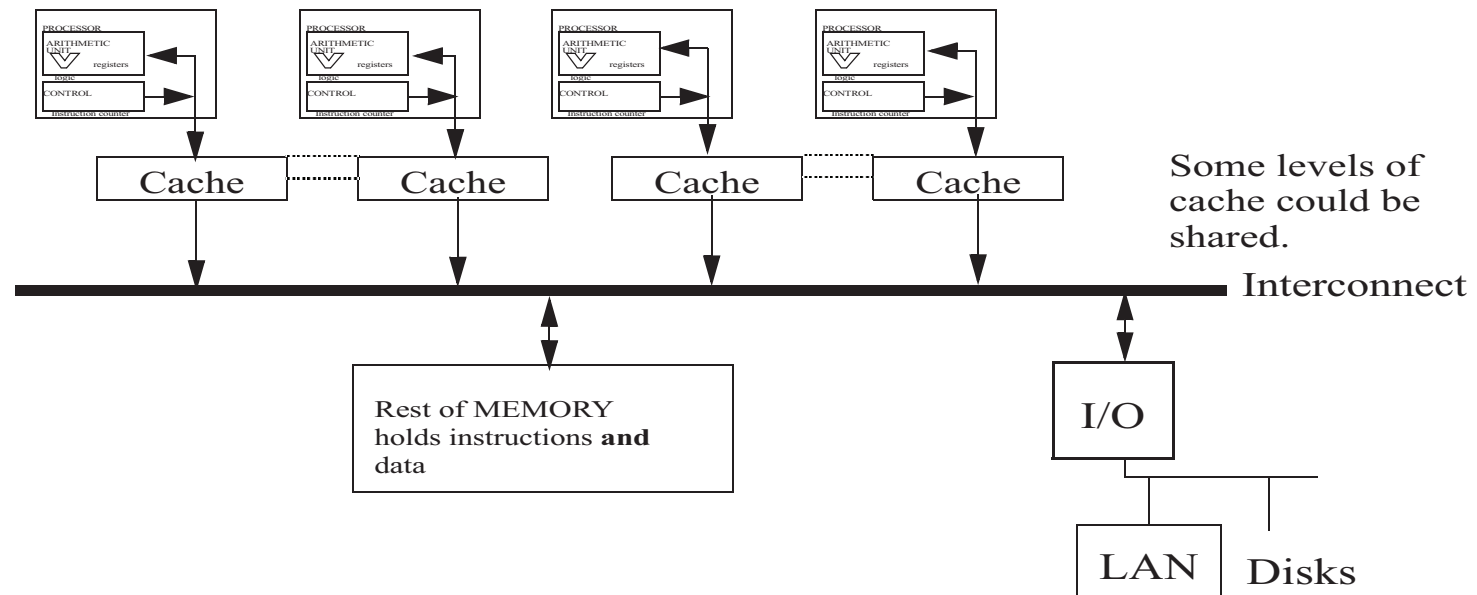


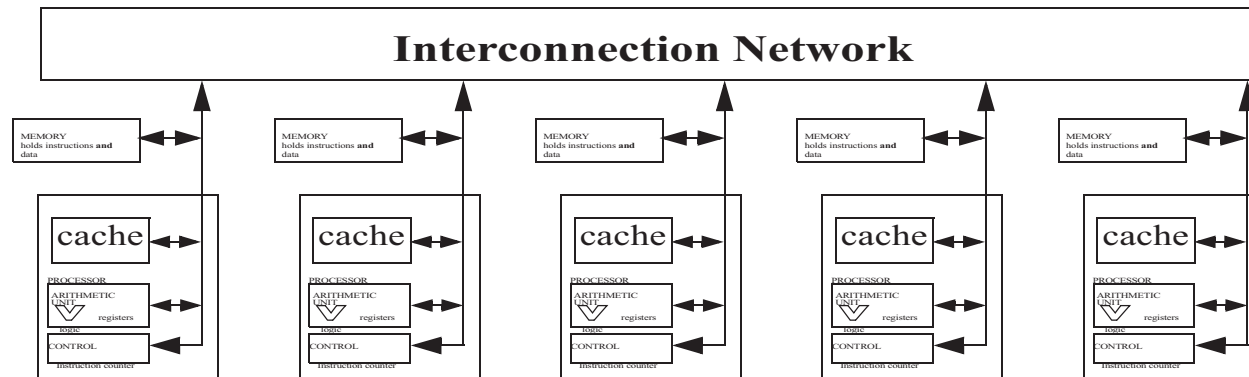
Shared-Memory Multiprocessors

- In shared-memory multiprocessors, there is an address space that can be accessed by all processors in the system. An address within that space represents the same location in all processors.
- The shared address space does not require a single, centralized memory module.

- The simplest form of a shared-memory multiprocessor is the symmetric multiprocessor (SMP). By symmetric we mean that each of the processors has exactly the same abilities. Therefore any processor can do anything: they all have equal access to every location in memory; they all can control every I/O device equally well, etc. In effect, from the point of view of each processor the rest of the machine looks the same, hence the term symmetric



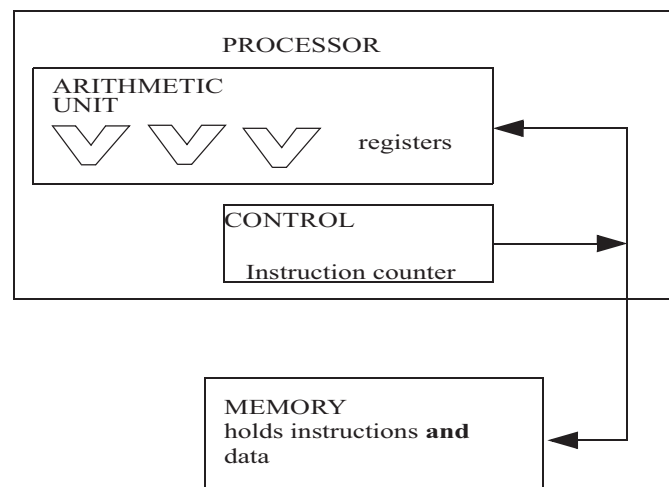
- An alternative design are distributed shared-memory. These are also called NUMA (nonuniform memory accesses) machines. These designs reduce the cost of access to local memory and are a cost-effective way of scaling the memory bandwidth if most of the accesses are to local memory.



- In shared-memory multiprocessors communication and synchronization is typically implemented exclusively by write and read operations on the shared address space.

Other Forms of Parallelism

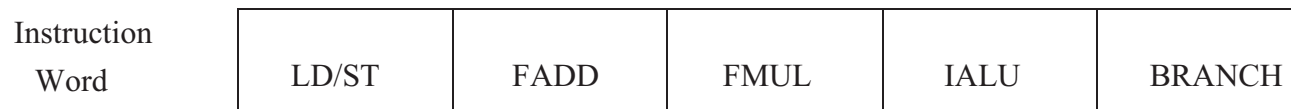
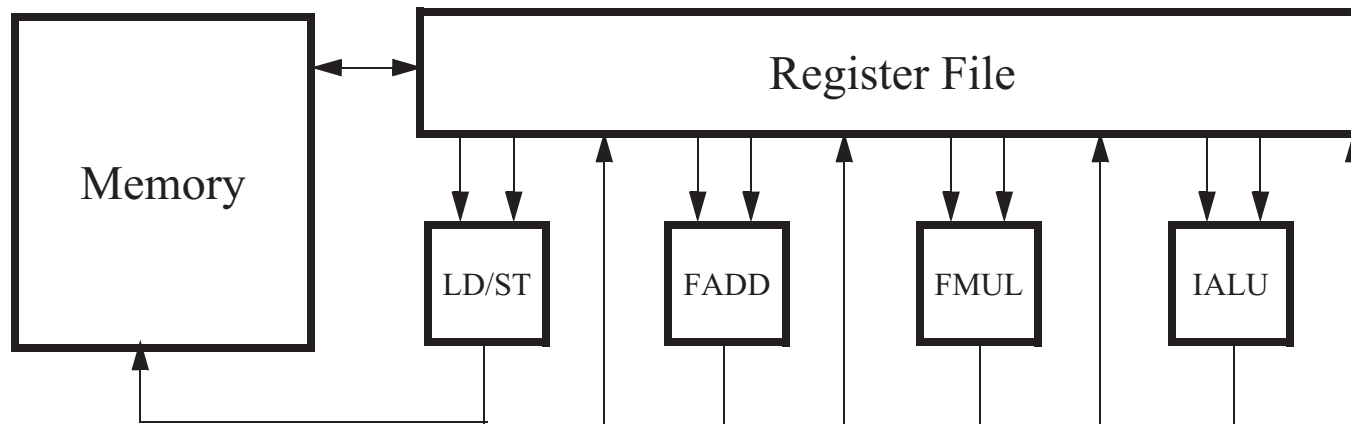
- As discussed above, there are other forms of parallelism that are widely used today. These usually coexist with the coarse grain parallelism of multicomputers and multiprocessors.
- Pipelining of the control unit and/or arithmetic unit.
- Multiple functional units



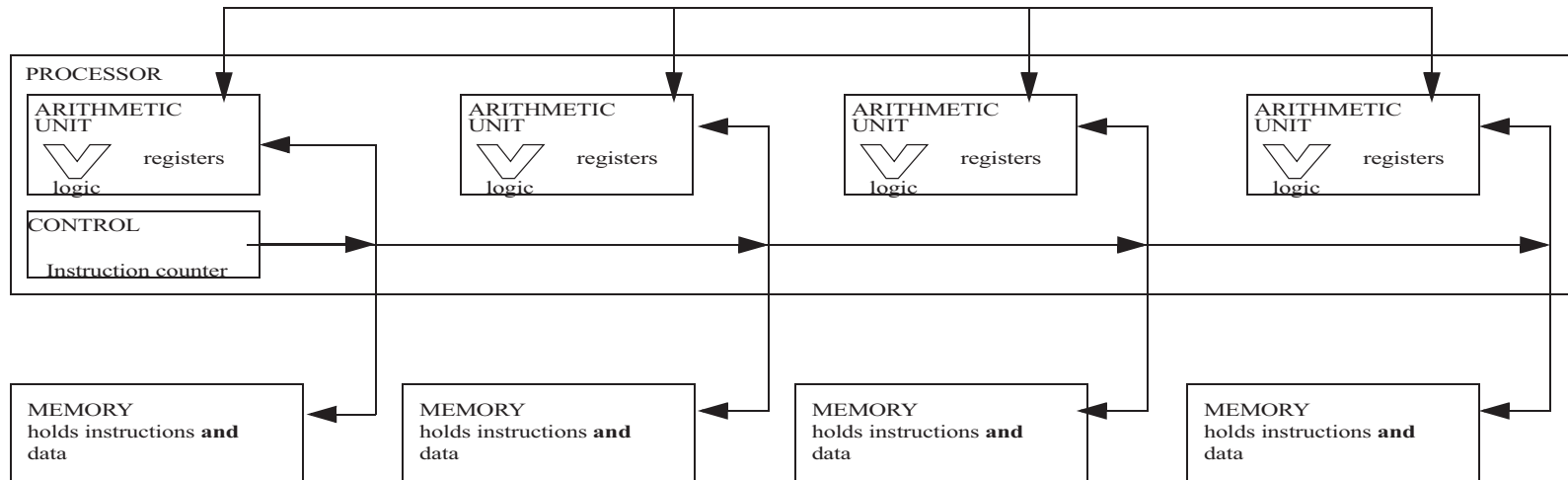
- Most microprocessors today take advantage of this type of parallelism.

- **VLIW (Very Long Instruction Word) processors are an important class of multifunctional processors. The idea is that each instruction may involve several operations that are performed simultaneously. This parallelism is usually exploited by the compiler and not accessible to the high-level language programmer. However, the programmer can control this type of parallelism in assembly language.**

Multifunction Processor (VLIW)



- **Array processors. Multiple arithmetic units**



- **Illiack IV is the earliest example of this type of machine. Each processing element (containing an arithmetic unit) of the Illiac IV was connected to four others to form a two-dimensional array (torus).**
- **A modern example is the NVIDIA GPU.**

Flynn's Taxonomy

- Michael Flynn published a paper in 1972 in which he picked two characteristics of computers and tried all four possible combinations. Two stuck in everybody's mind, and the others didn't:
- **SISD**: Single Instruction, Single Data. Conventional Von Neumann computers.
- **MIMD**: Multiple Instruction, Multiple Data. Multicomputers and multiprocessors.
- **SIMD**: Single Instruction, Multiple Data. Array processors.
- **MISD**: Multiple Instruction, Single Data. Not used and perhaps not meaningful.



Microarchitectures of Today

Three main categories:

- Conventional shared-memory MIMD machines (multicores)
- (Multi) SIMD devices (MEA Multiple Execution Arrays) (manycores)
- Hybrid systems

Shared-Memory MIMD micro architectures (Multicores)

Conventional processors and often (but not always)

- Shared caches
- Multithreaded
- Vector extensions of increasing power such as IBM's VMX, Intel's SSE4 and AVX (Advanced Vector Extensions)
- SMP today, NUMA in the future (?)