Advanced Computer Architecture (0630561)

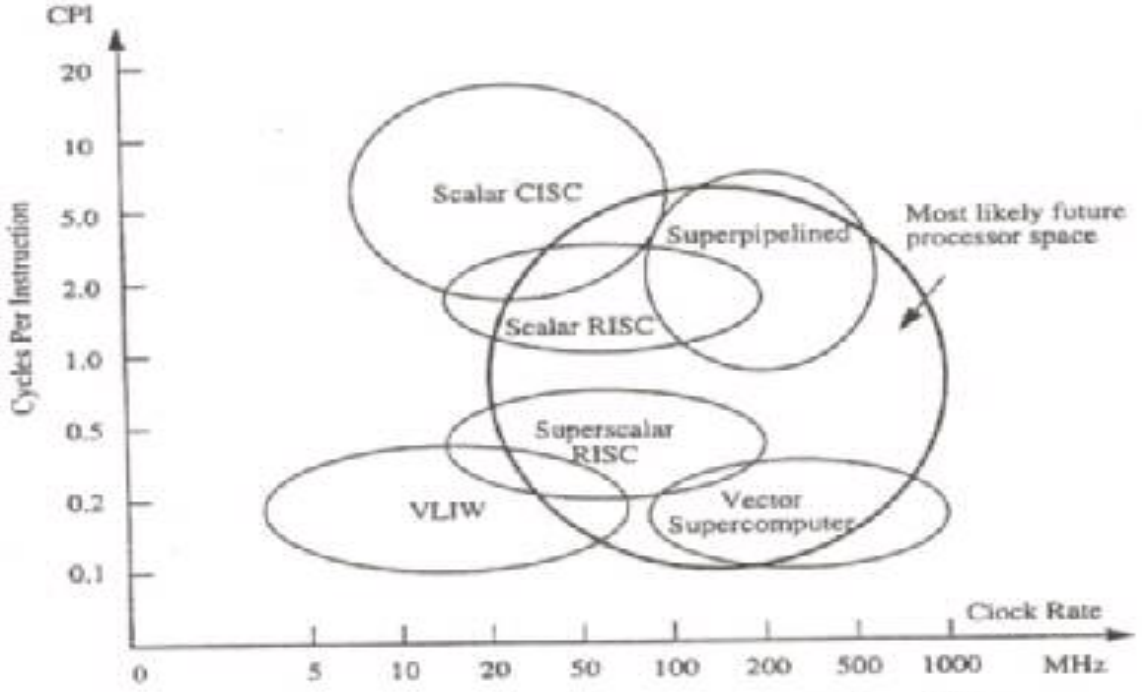
Lecture 13

Superscalar Architectures

Prof. Kasim M. Al-Aubidy
Computer Eng. Dept.

Design Space of Processor:

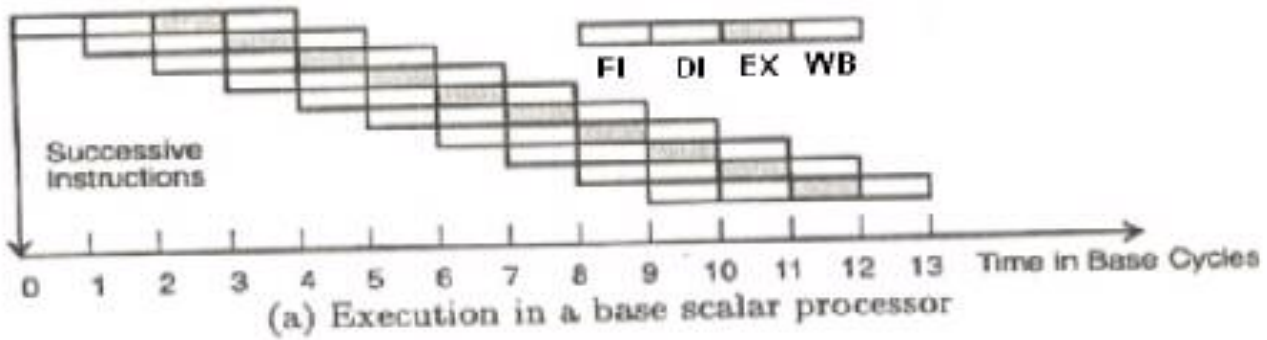
- Various processors families can be mapped onto a coordinated space of clock rate verses CPI.
- As implementation technology evolves rapidly, the clock rate are moving from low to higher speeds.
- Another trend is that processor designers are trying to lower the CPI rate using hardware and software approaches.



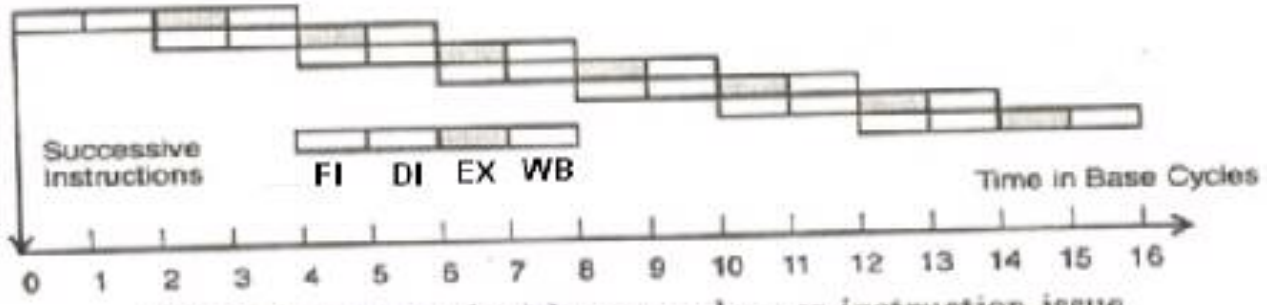
Design space of modern processor families.

Base Scalar Processor:

- It is defined as a machine with one instruction issued per cycle.



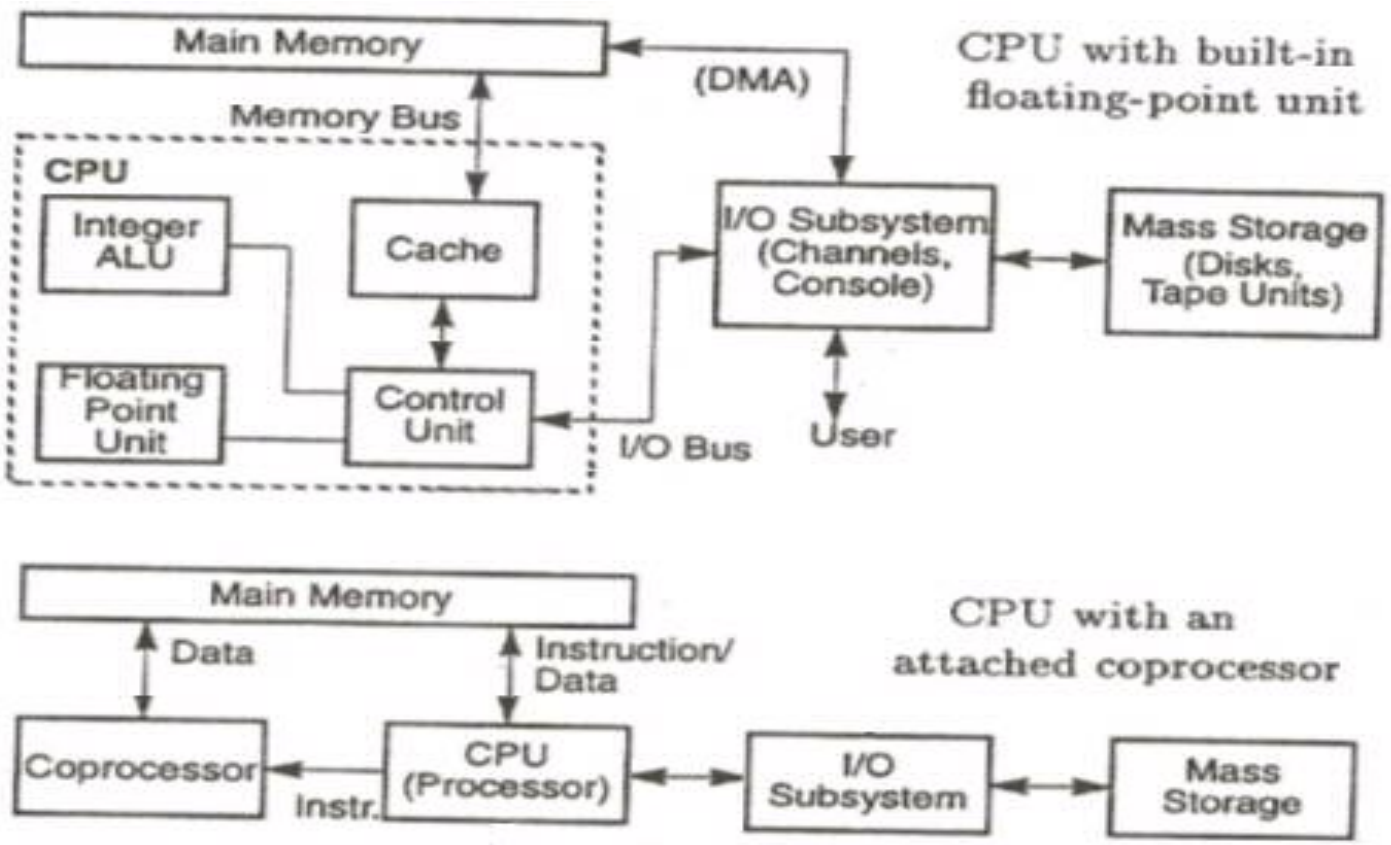
(a) Execution in a base scalar processor



(b) Underpipelined with two cycles per instruction issue

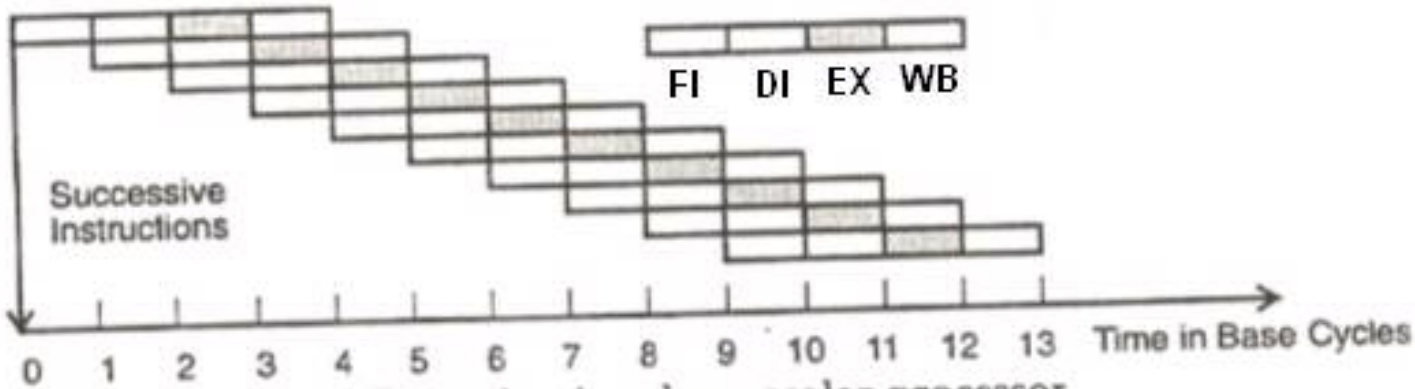
Base Scalar Computer Architecture:

- The CPU is essentially a scalar processor consists of multiple functional units.
- The floating-point unit can be built on a coprocessor attached to the CPU.



RISC Scalar Processors:

- Generic RISC processors are called scalar RISC because they are designed to issue one instruction per cycle, similar to the base scalar processor.



Execution in a base scalar processor

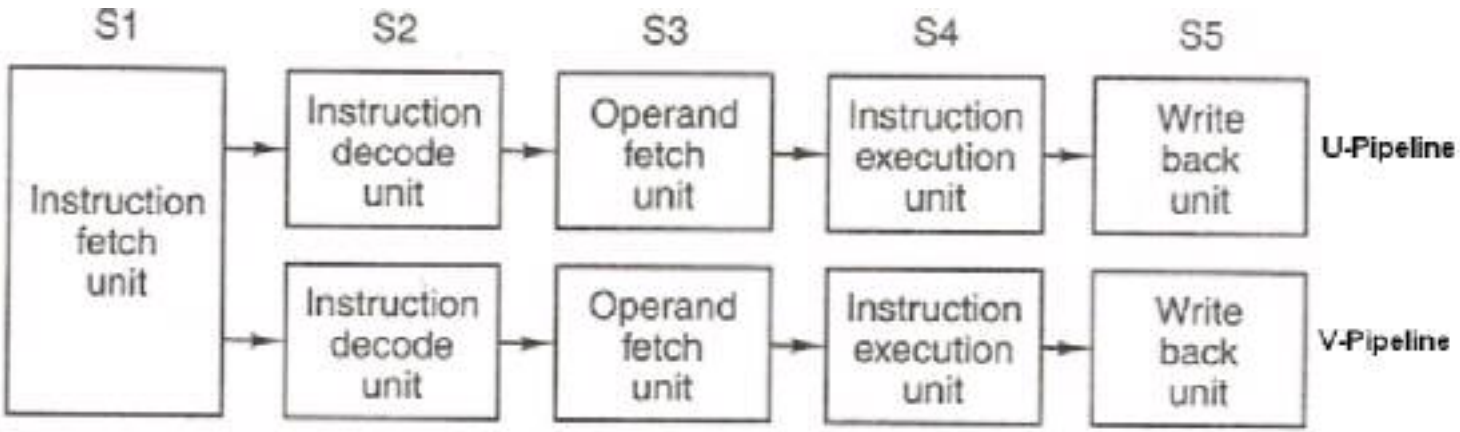
- In theory, both RISC and CISC scalar processors should perform about the same if they run with the same clock rate, and with equal program length.
- Without a high clock rate, a low CPI, and good compilation support, neither CISC nor RISC can perform well as designed.

Superscalar Architectures:

- If one pipeline is good, then two pipelines are better.

Example: Consider the following architecture (as in Pentium);

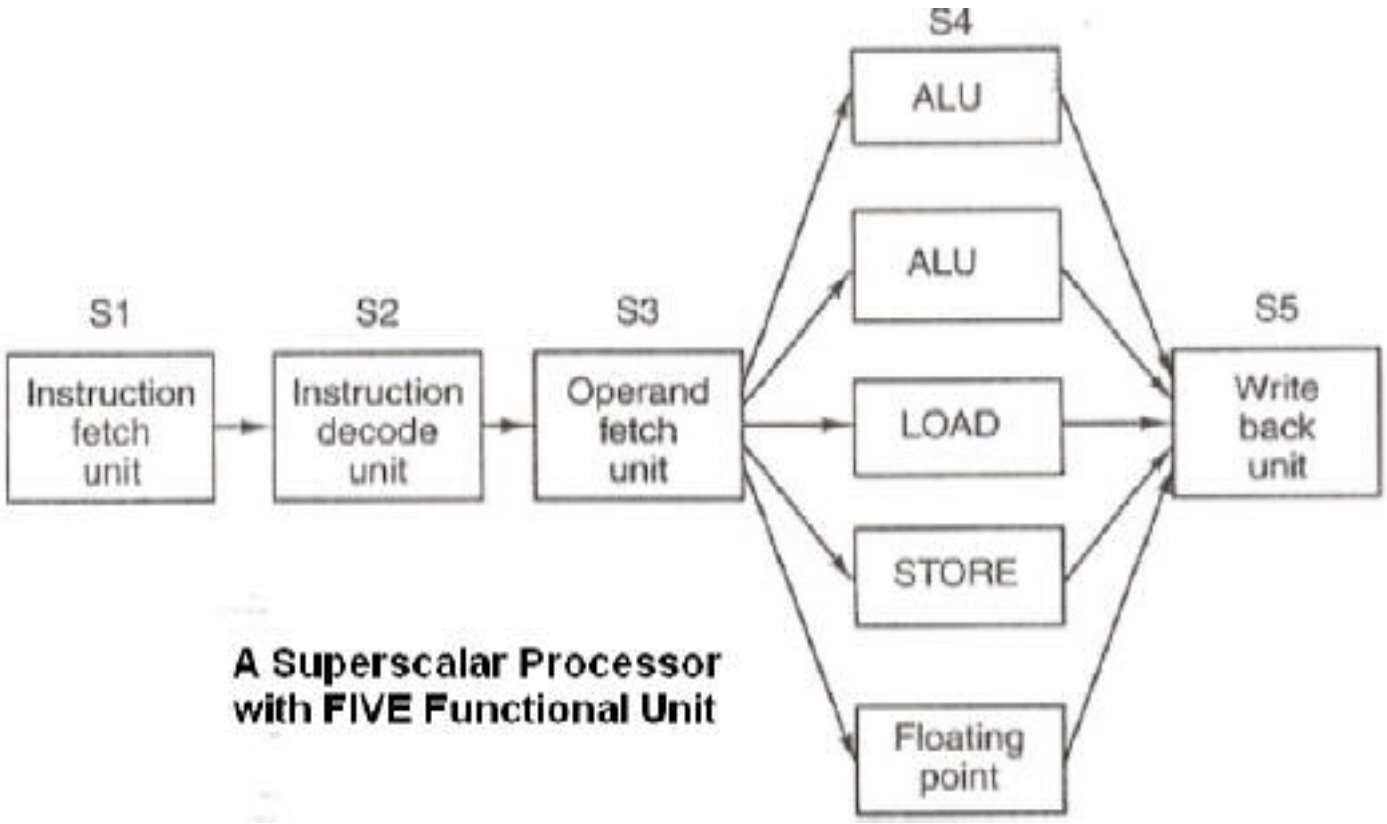
- Single instruction fetch unit fetches pairs of instructions together and puts each one into its own pipeline, complete with its own ALU for parallel operation.
- The main pipeline (U-Pipeline) could execute an arbitrary Pentium instruction.
- The V-Pipeline could execute only simple integer instructions (and also one simple floating-point instruction).
- If the instructions in a pair were not simple enough or incompatible, only the first one was executed (in U-pipeline). The second one was then held and paired with the instruction following it.



Dual five-stage pipelines with a common instruction fetch unit.

Example: A superscalar processor (Pentium II) with 5 functional units:

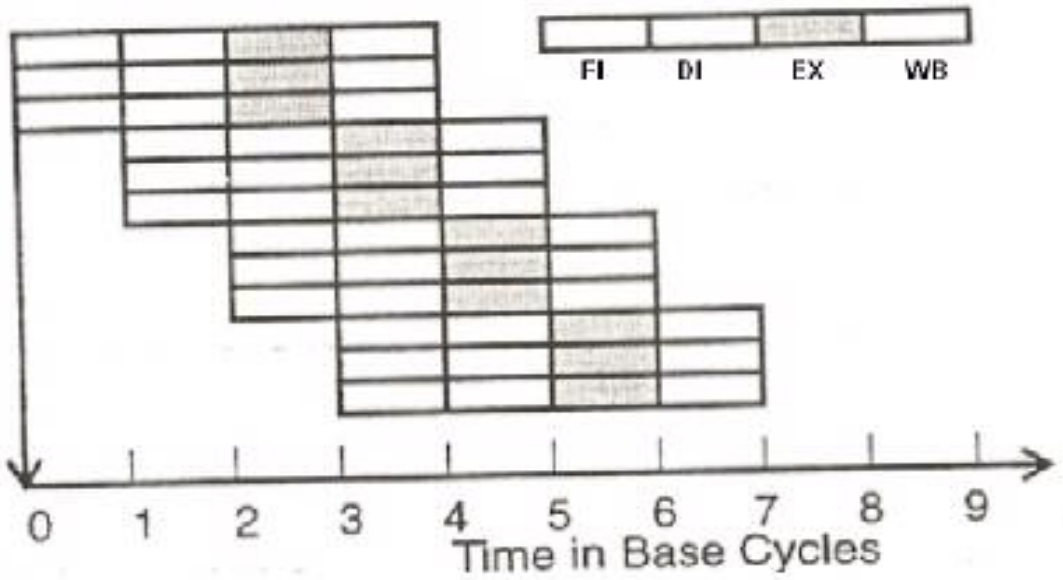
- It is possible to have multiple ALUs in stage (S4).
- Most of the functional units in stage (S4) take longer than one clock cycle to execute.
- Stage (S3) can issue instructions faster than the S4 stage.



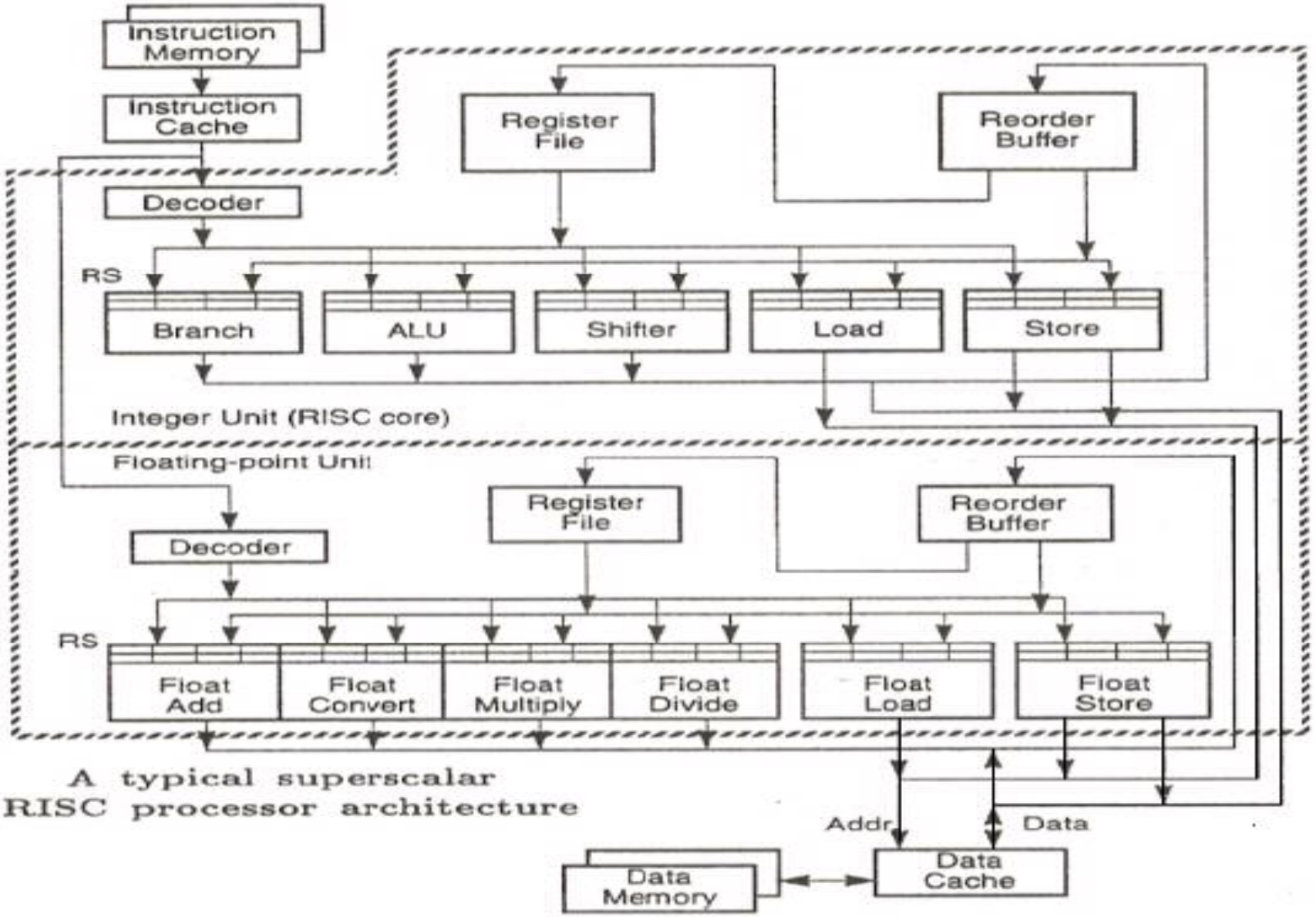
A Superscalar Processor with FIVE Functional Unit

Superscalar Processors:

- In a superscalar processor, multiple instruction pipelines are required. This implies that multiple instructions are issued per cycle and multiple results are generated per cycle.
- Superscalar processors are designed to exploit more instruction-level parallelism in user programs. Only independent instructions an be executed in parallel without causing a wait state.
- The instruction-issue degree in a superscalar processor is limited to 2-5 in practice.
- The effective CPI of a superscalar processor should be lower than that of a generic scalar RISC processors.



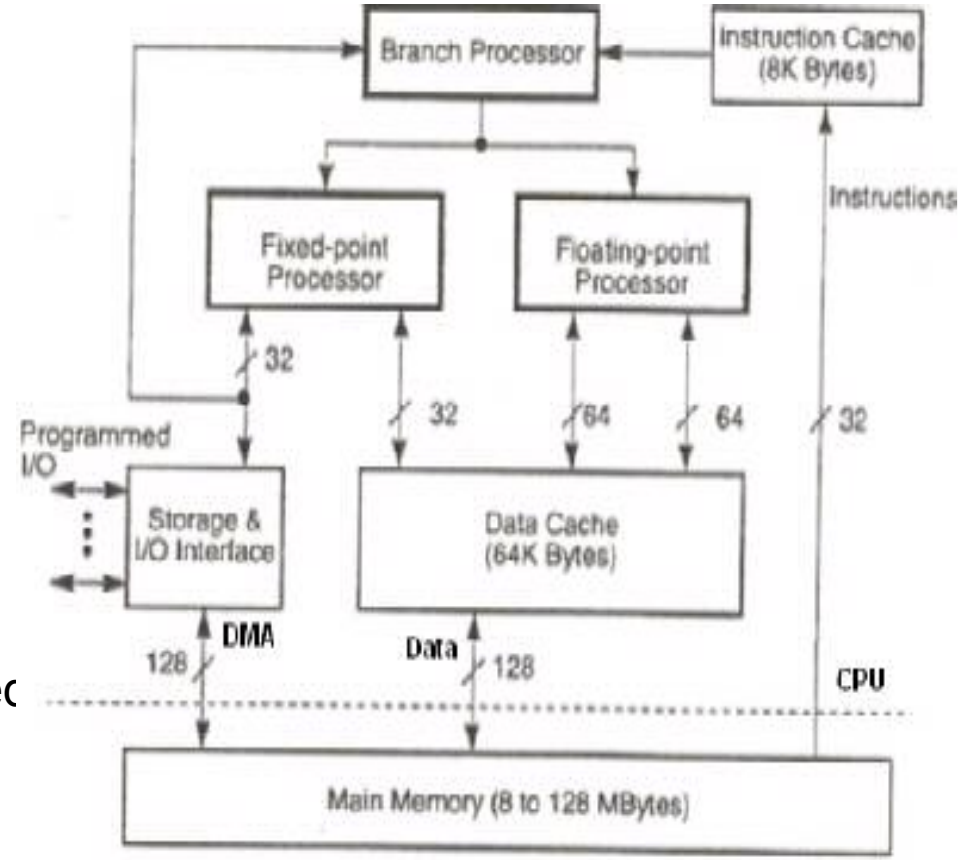
A superscalar processor of degree m = 3.



A typical superscalar RISC processor architecture

Example: IBM RS/6000

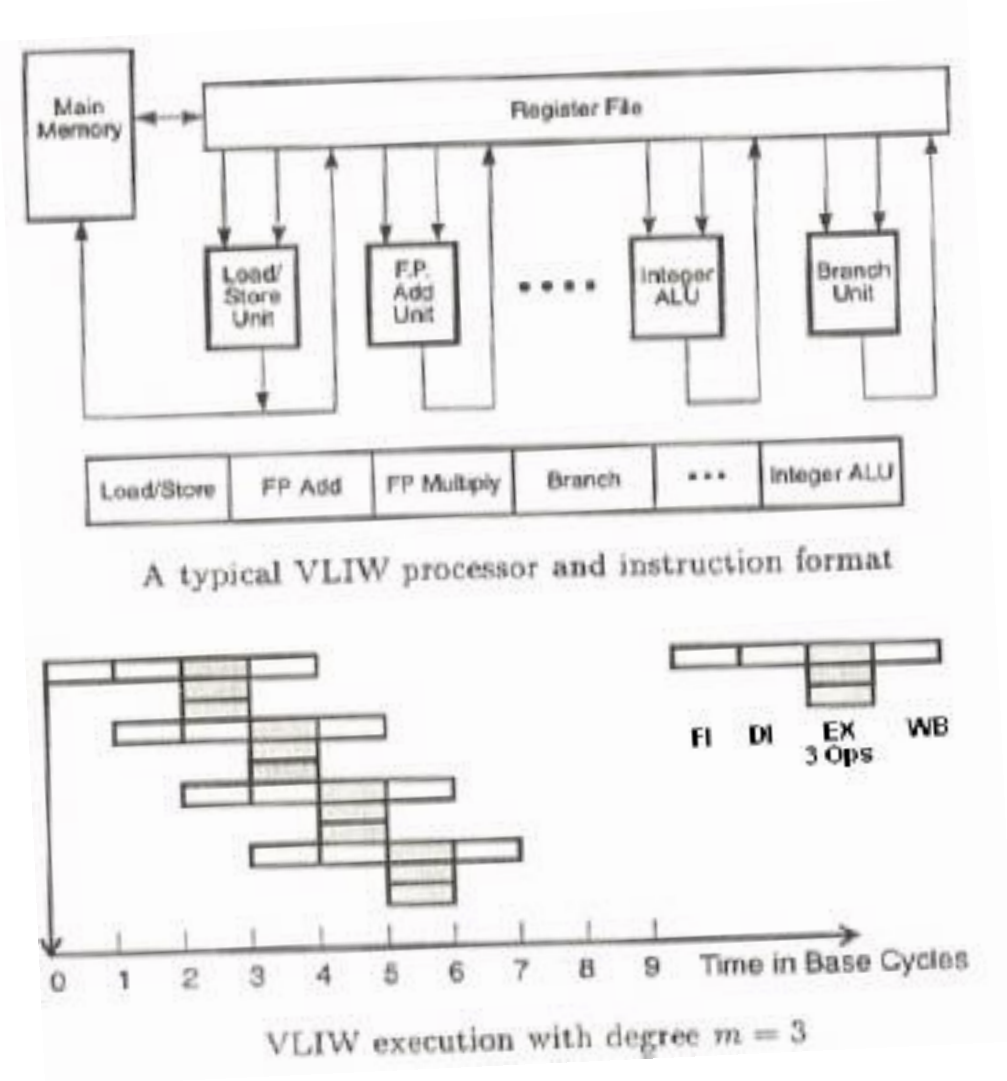
- IBM announced this superscalar RISC system in 1990.
- There are 3 parallel functional units; branch processor, fixed-point unit, and floating-point unit.
- The branch processor can arrange the execution of up to 5 IPC.
- It is hardwired rather than micro-programmed control unit.
- The system uses a number of wide buses. These will provide the high instruction and data bandwidths required for superscalar implementation.
- This system design is optimized to perform well in numerically intensive scientific and engineering applications.



The POWER architecture of the IBM RISC System/6000 superscalar processor.

The VLIW Architecture:

- Very Long Instruction Word (VLIW) architecture is generalized from two concepts; horizontal microcoding and superscalar processing.
- A typical VLIW machine has;
  - instruction words hundreds of bits in length.
  - Multiple functional units.
  - Common large register file shared by all functional units.



VLIW and Superscalar Processor:

VLIW machines behave much like superscalar machine with 3 differences:

1. The decoding of VLIW instruction is easier than that of superscalar instructions.
2. The code density of the superscalar machine is better than when the available instruction level parallelism is less than that exploitable by the VLIW machine.
3. A superscalar machine can be object-code compatible with a larger family of nonparallel machines. On the contrary, a VLIW machine exploiting different amount of parallelism would require different instruction sets.

VLIW: Advantages:

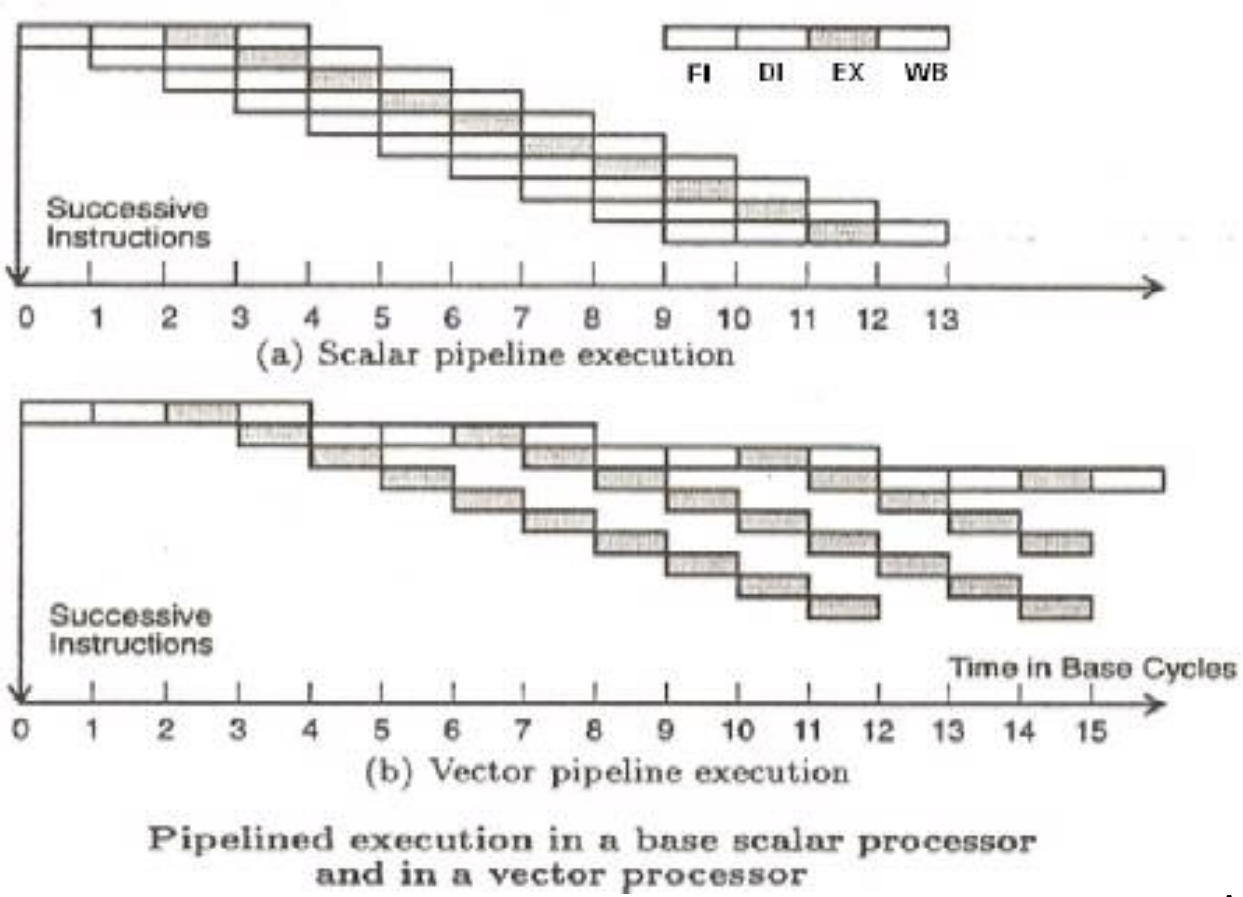
- The main advantage of VLIW architecture is its simplicity in hardware structure and instruction set.
- The VLIW processor can potentially perform well in scientific applications where the program behavior (branch predictions) is more predictable.

Vector Processors:

- A vector processor is a coprocessor specially designed to perform vector computations.
- A vector instruction involves a large array of operands. The same operation will be performed over a string of data.
- Vector processors are often used in a multipipelined supercomputer.
- Vector processors can assume either;
- A register-to-register architecture using shorter instructions and vector register files, or
- A memory-to-memory architecture using memory-based instructions.
- The vector pipelines can be attached to any scalar processor (whether it is superscalar, superpipelined, or both).

Vector Pipelines:

- In a scalar processor, each scalar instruction executes only one operation over one data element.
- Each vector instruction executes a string of operations, one for each element in the vector.



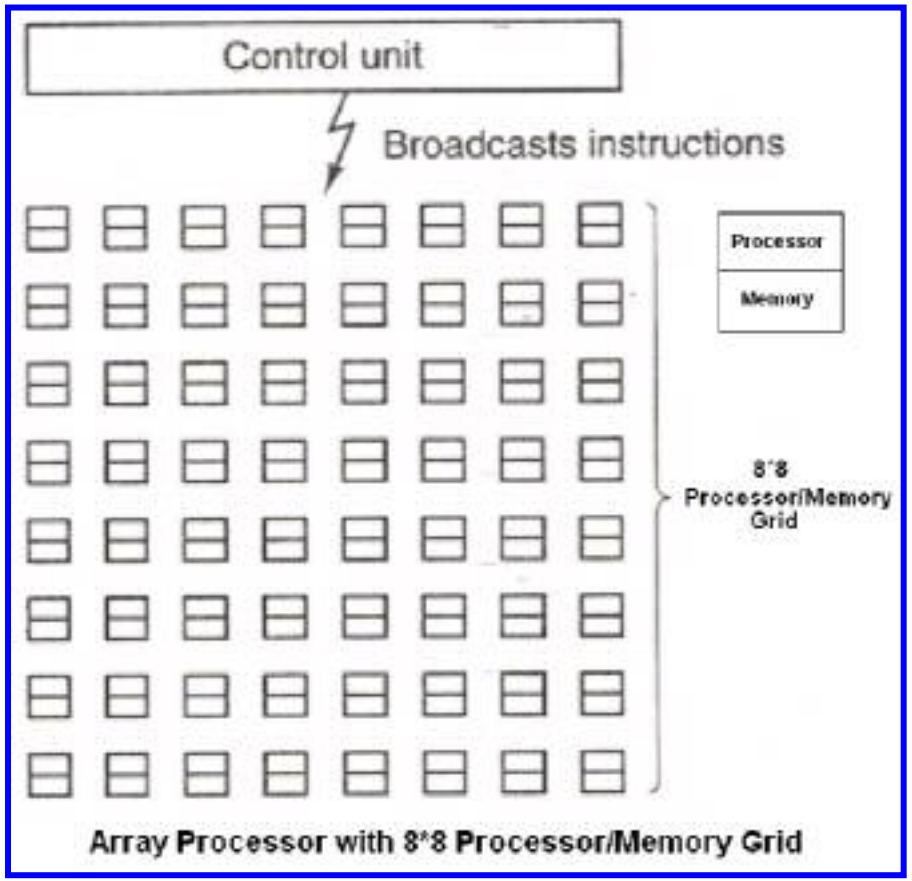
Pipelined execution in a base scalar processor and in a vector processor

Array Processor:

- An array processor consists of a large number of identical processors that perform the same sequence of instructions on different sets of data.

Example: ILLIAC IV

- Announced by University of Illinois in 1972.
- The original plan was to build a machine consisting of 4 quadrants, each having 8*8 square grid of processor/memory elements.
- Only one quadrant was built due to cost. It did achieve a performance of 50 megaflops.



Array Processor with 8*8 Processor/Memory Grid