

COL216

Computer Architecture

Concluding Remarks:
What lies beyond the present course
31st March 2022

What have we studied

- Instruction set architecture and assembly language programming
- Arithmetic operations, how to build an ALU
- Constructing a processor to execute instructions – Micro architecture
- Performance issues
- Pipelining to improve performance
- Memory: caches and virtual memory
- Input / output

What we have not studied

- Advanced pipelining techniques
- Advanced concepts in memory hierarchy design
- Superscalars
- VLIW architectures
- Vector processors
- Multiprocessors
- GPUs

How to achieve high performance?

- Do things faster
- Do more things at the same time
- Achieve same thing by doing less
- Remove bottle necks – achieve balance
- What do N_{instr} , CPI_{avg} and T_{clock} depend on?
 - Technology
 - Circuit / logic design
 - Architecture
 - Compiler, OS
 - Algorithm

Technology trends (increase/year)

- Transistor density : 35%
- Die size : 10-20%
- Transistor count : 40-55% (double in 18-24 mo)
 - Moore's Law
- DRAM capacity : 25-40% (double in 2-3 yr)
- Flash chip capacity : 50-60%
- Disk capacity : 40% ($30 \Rightarrow 60 \Rightarrow 100 \Rightarrow 40$)

Forms of Parallelism

- Data Parallelism
- Functional Parallelism
- Fine Grain Parallelism
- Coarse Grain Parallelism

Data Parallel Architectures

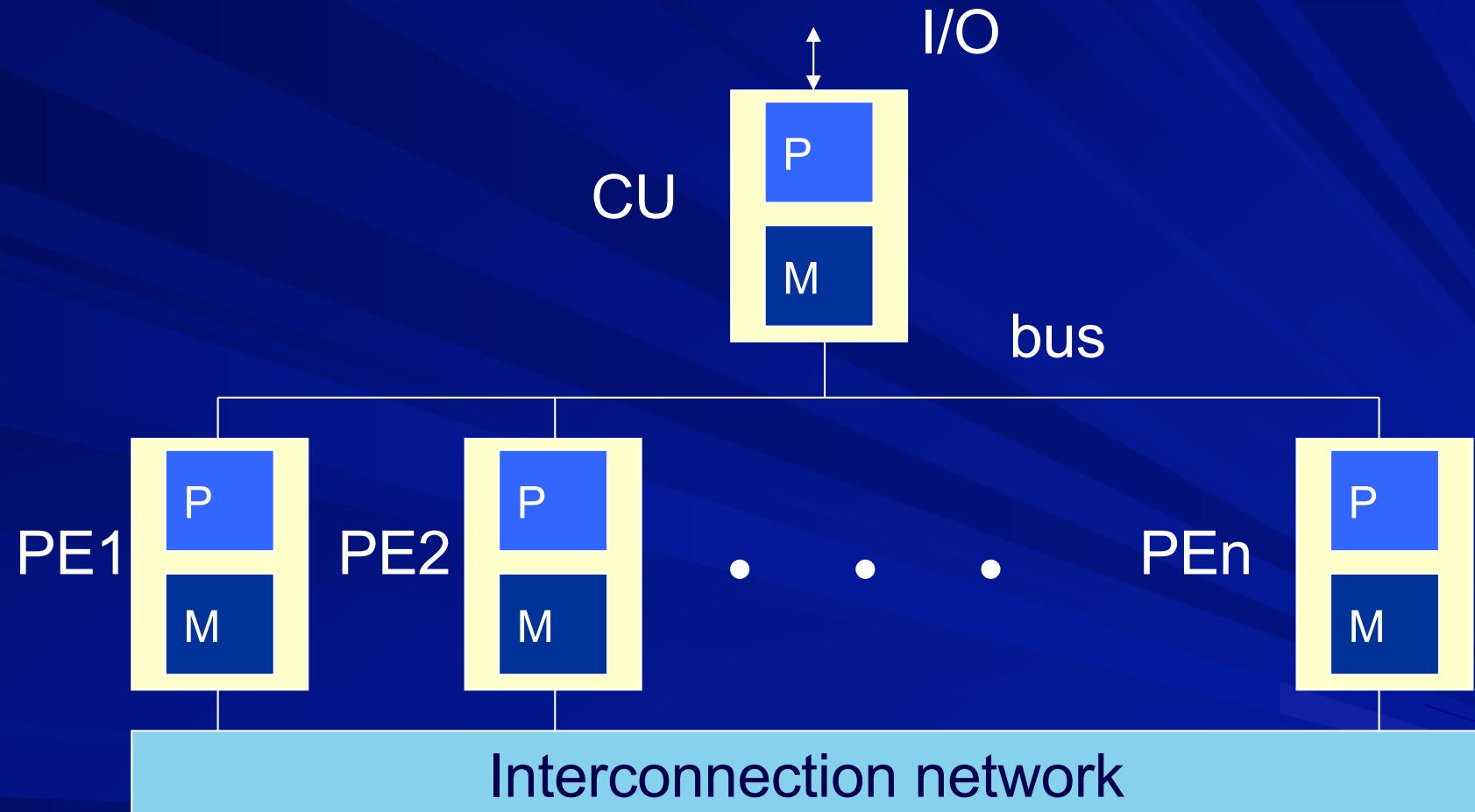
- SIMD Processors

- Multiple processing elements driven by a single instruction stream

- Vector Processors

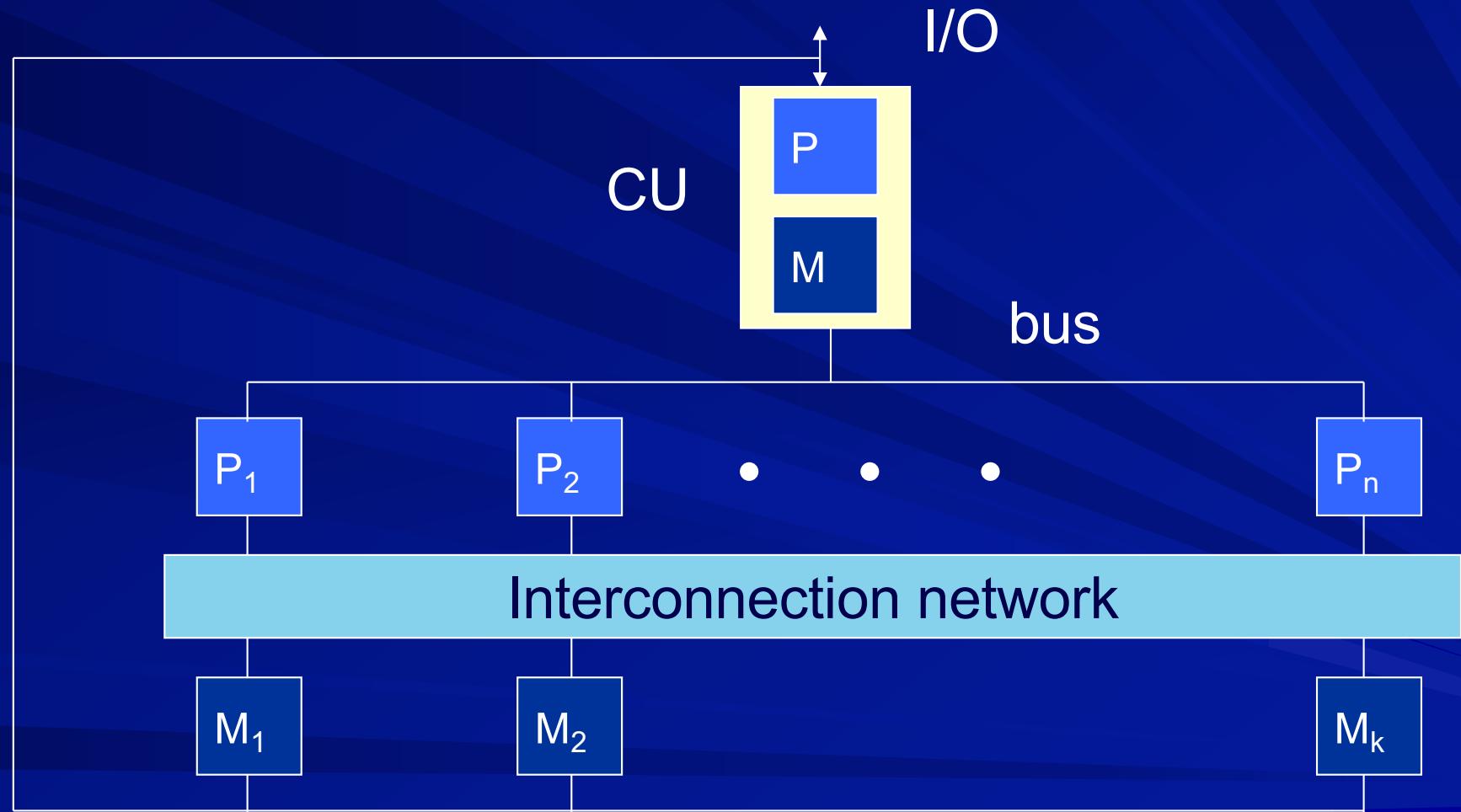
- Uni-processors with vector instructions

ILLIAC IV SIMD



Planned for 64 x 4 PEs, built only 64

Burroughs Scientific Processor



Rise and fall of SIMDs

- Introduced in 60' s (e.g. Illiac, BSP)
- Problems:
 - not cost effective
 - serial fraction and Amdahl's law
 - I/O bottle neck
- Overshadowed by Vector Processors
- Resurrected in 80' s
- Did not survive because of high cost
- Basic ideas used in some modern systems

Vector Processors

- Instructions to operate on vectors
- Vectors are streamed into and out of highly pipelined functional units
- Vector registers hold vector data
- Vectors are streamed from memory into vector registers and from vector registers into memory, no cache
- Sequences of vector operations are chained

Vector Processor Architecture

