

DNA

Introduction to Processors

Alexandre David

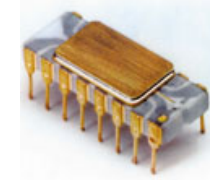
1.2.05

adavid@cs.aau.dk



First Computers & Processors

Busicom 141-PF printing
calculator, Intel 4004
1st microprocessor



IBM PC 1981
4.77MHz, Intel 8088
16 bits, 16kB RAM



Commodore 64 1982
1MHz, 8 bits, 64kB RAM

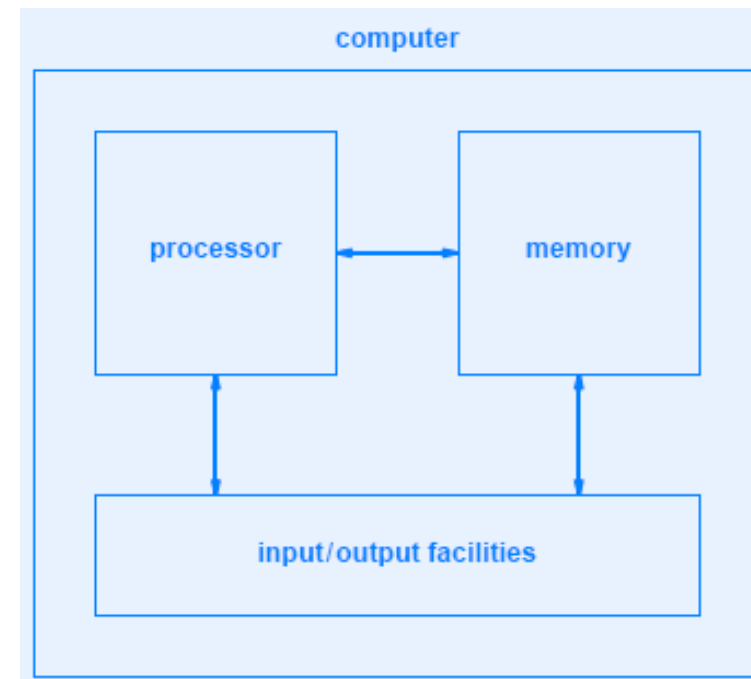


Apple Macintosh 1984
8MHz, Motorola 68000
32 bits, 128kB RAM

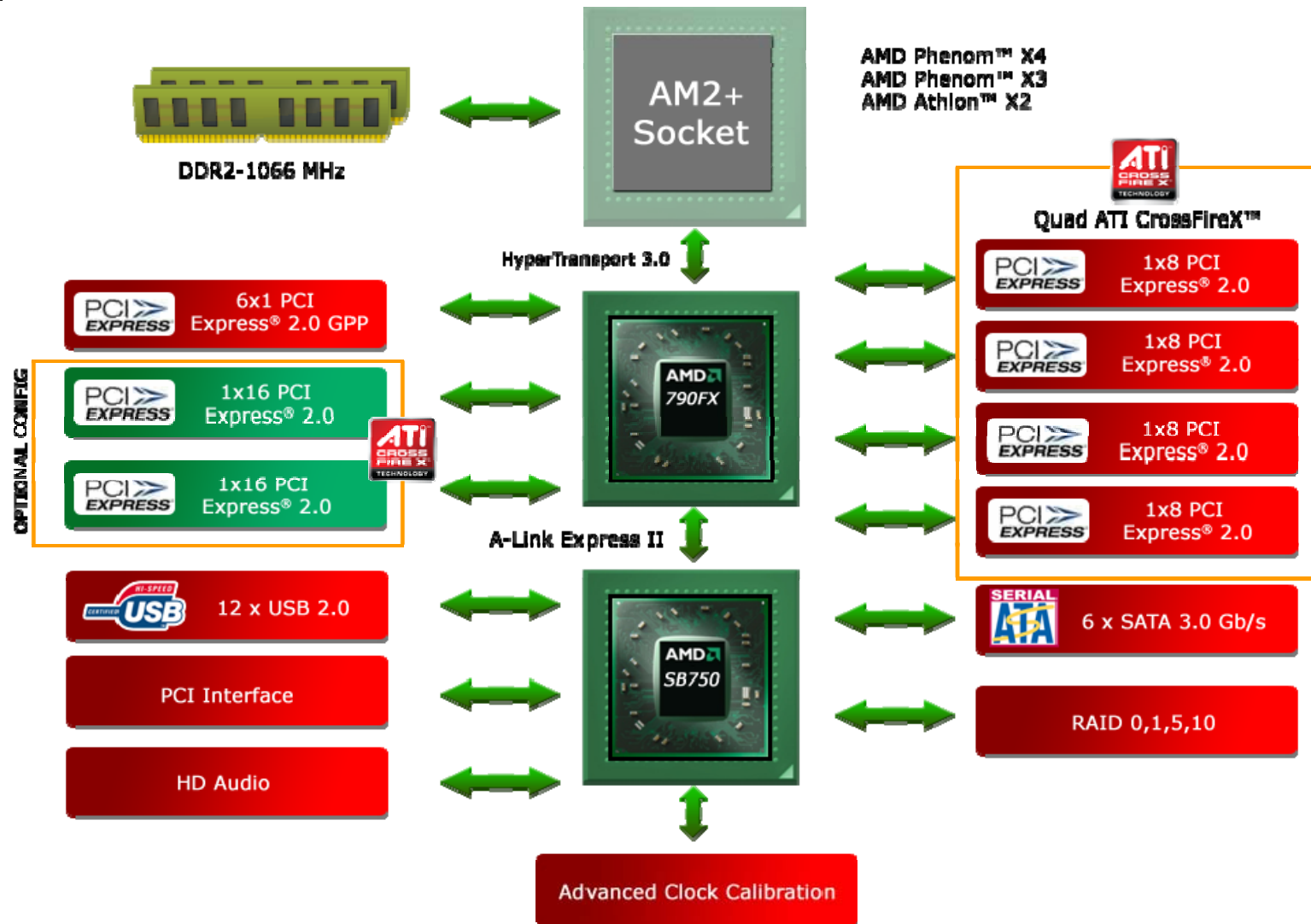


Von Neumann Architecture

- Fundamental concept – *stored program*.
 - (John Von Neumann – mathematician)
 - The model fits most modern processors.
- Basic components:
 - processor
 - memory
 - I/O

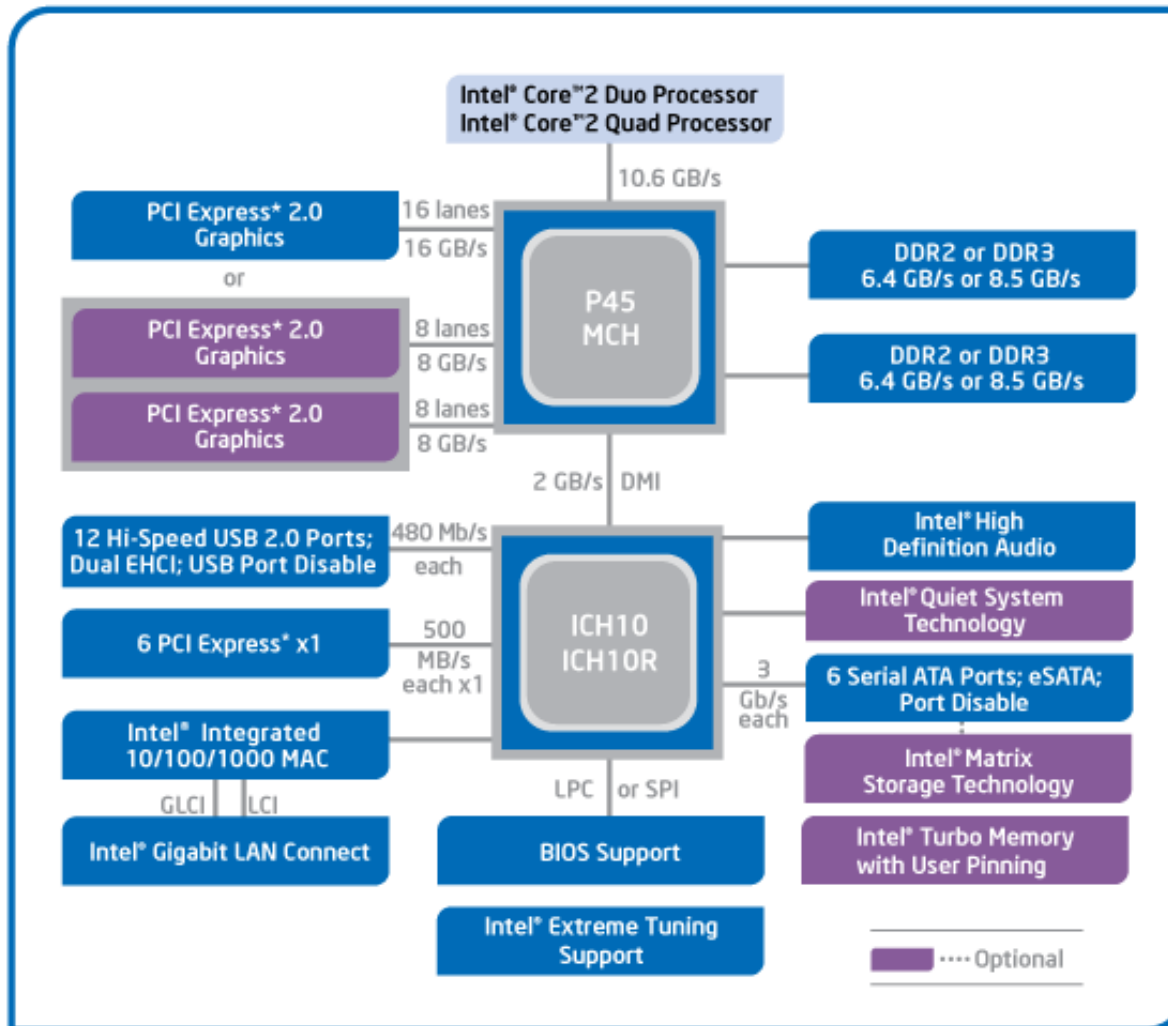


790FX





P45



Intel® P45 Express Chipset Block Diagram



Processors

- Perform computations involving multiple *steps*. Always doing something.
- Variety of capabilities
 - logical/arithmetic operations, int, float, double, mmx, cryptography...
- Types:
 - fixed logic - single operations
 - selectable logic – selectable operations
 - parameterized logic – control operations via parameters
 - programmable logic – program to run

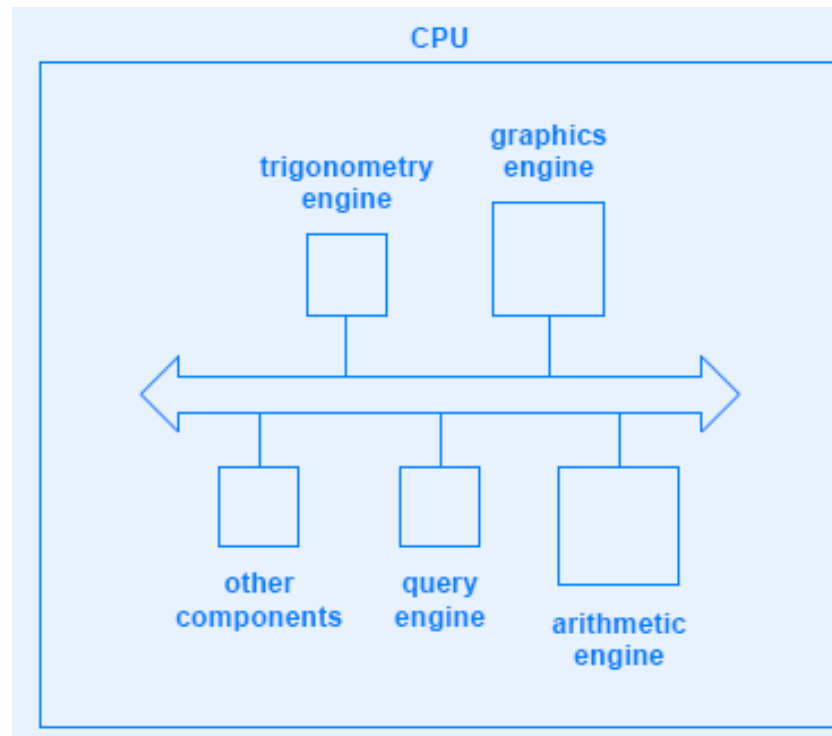


FPGA

- Special processors.
- Field Programmable Gate Array
 - programmable gates
 - “morph” the logic to be anything, on-the-fly
 - can be a processor A, change to be processor B, can execute programs, can do anything.

Structure of Processors

- Hierarchical approach
 - “computational engines”
computational units
 - idea:



→ CPUs
→ GPUs

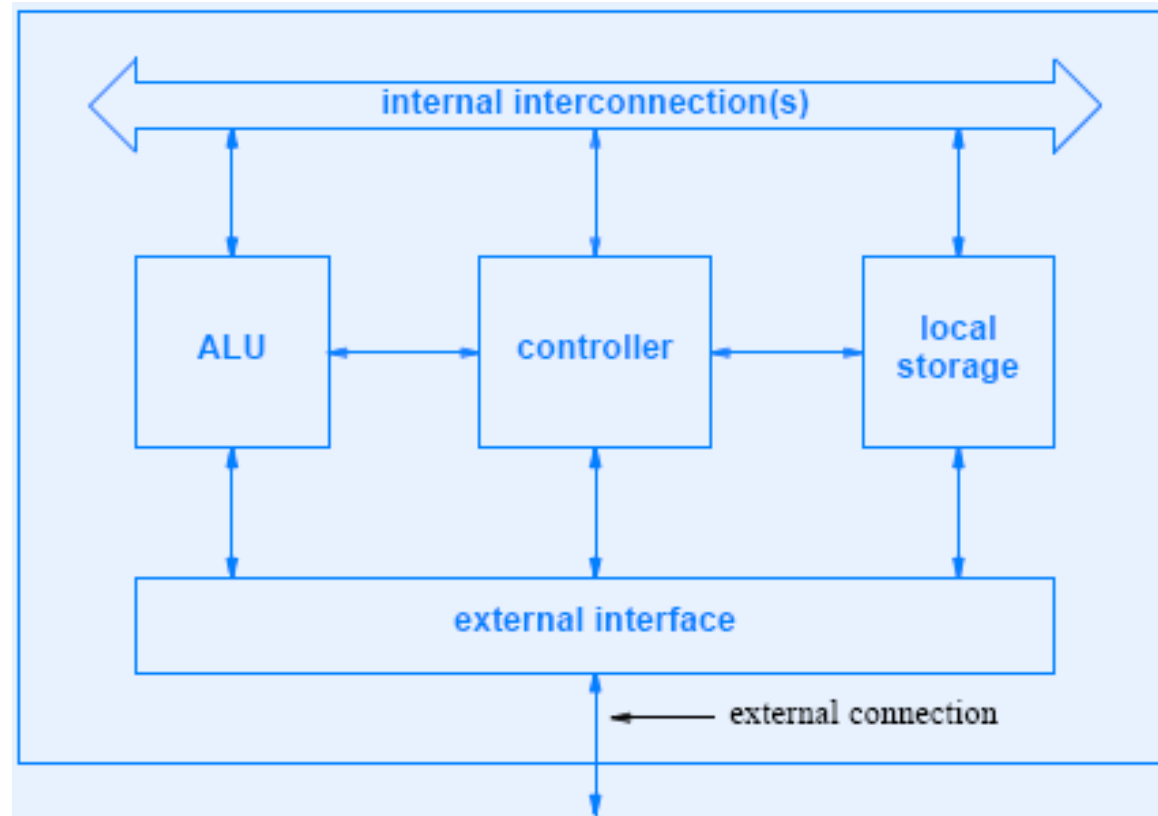


Major Components

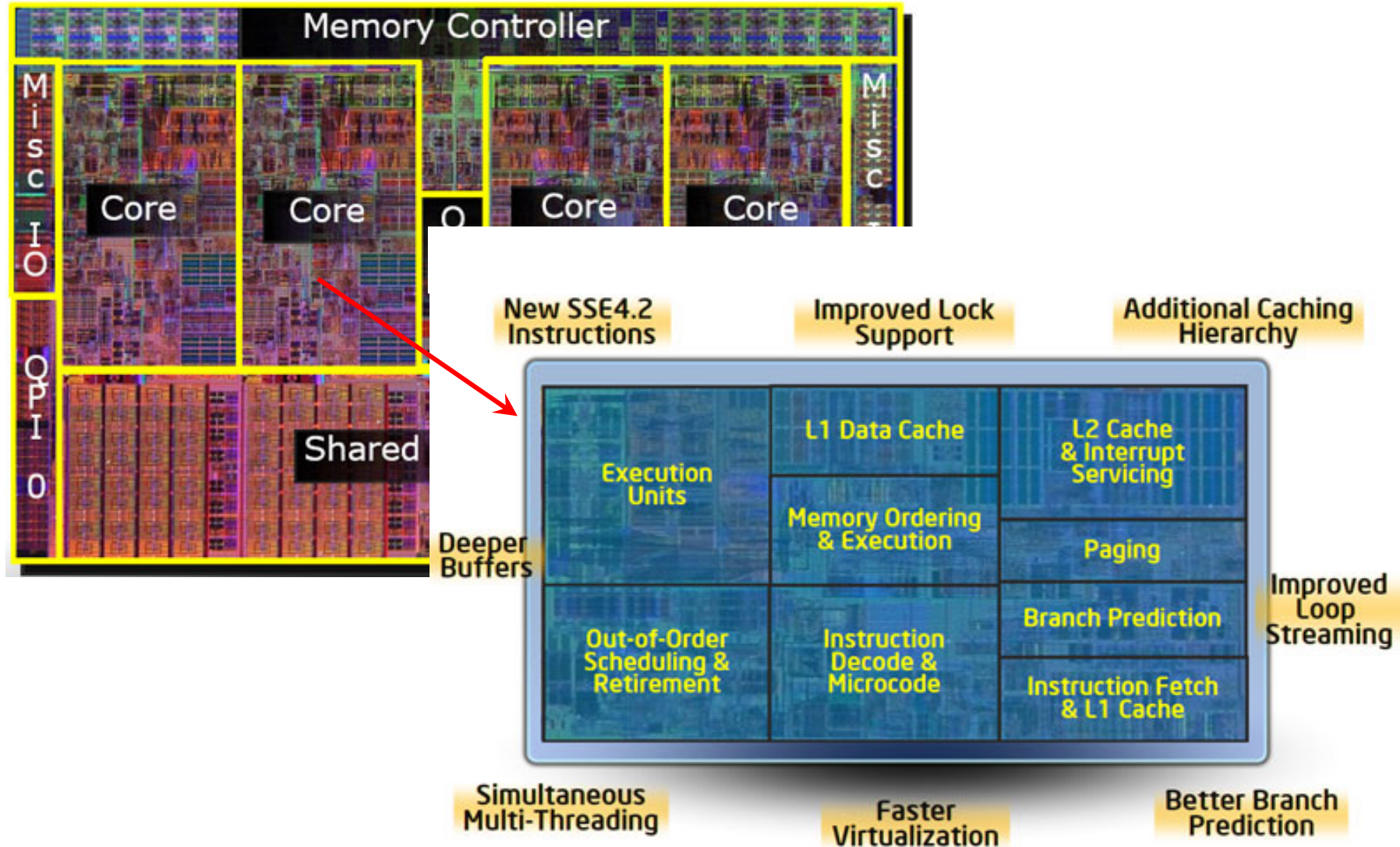
Concepts

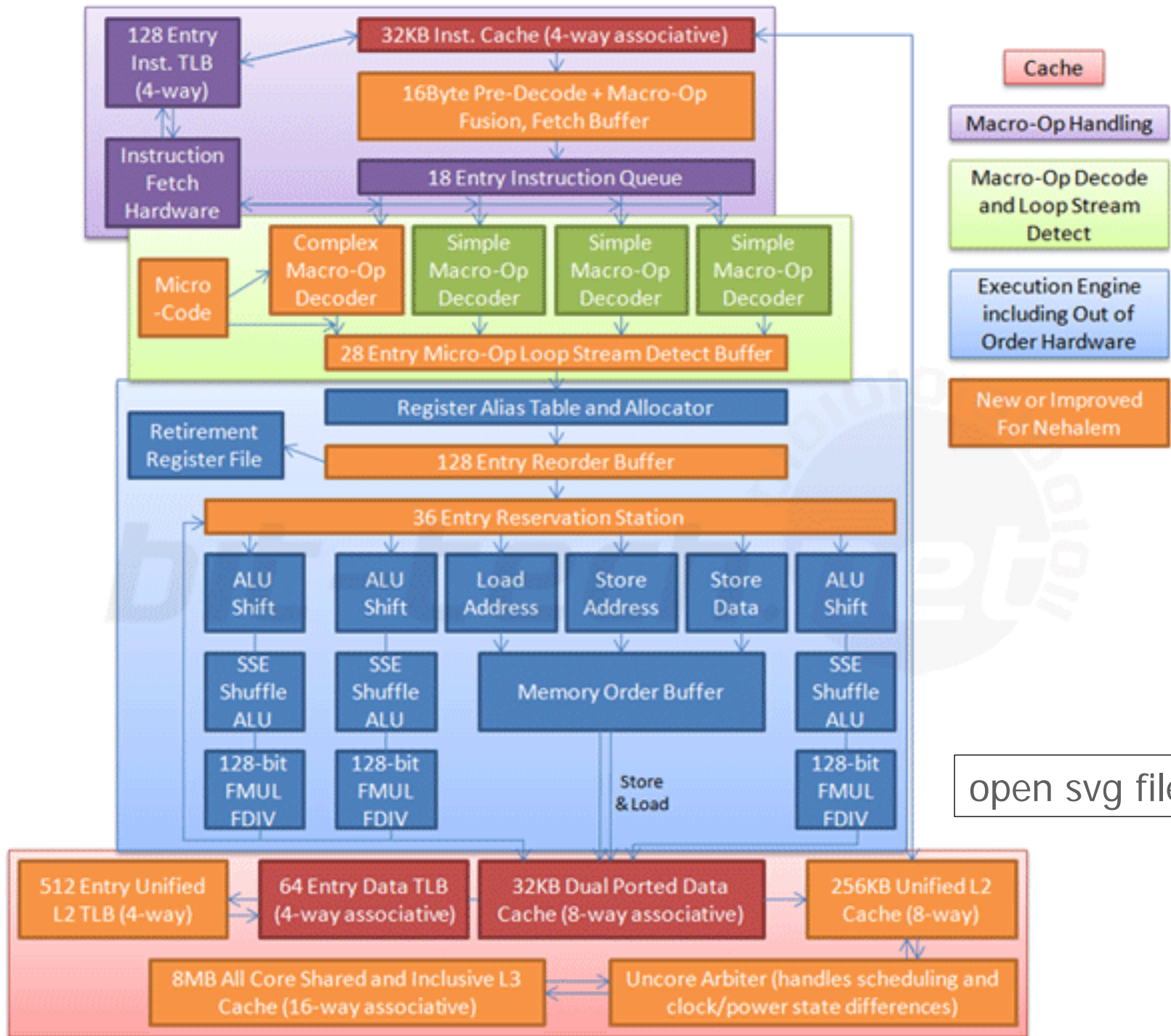
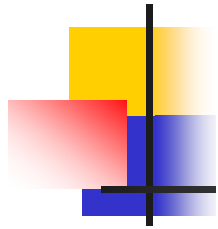
- Controller
 - control execution order
- Computational units
 - ALU, FPU, (MMU)
- Local data storage
 - registers, cache
- Internal interconnections
 - data paths, buses
- External interfaces
 - (memory controller), I/O

Concept



Nehalem





open svg file



Computational Units

- ALU
 - arithmetic (int add, sub, div, mul)
 - shifts (left, right, circular)
 - logic (and, or, xor, not)
- FPU
 - arithmetics ... + complex functions
 - MMX, SSE
- MMU
 - load/store out-of-order
- Decoders

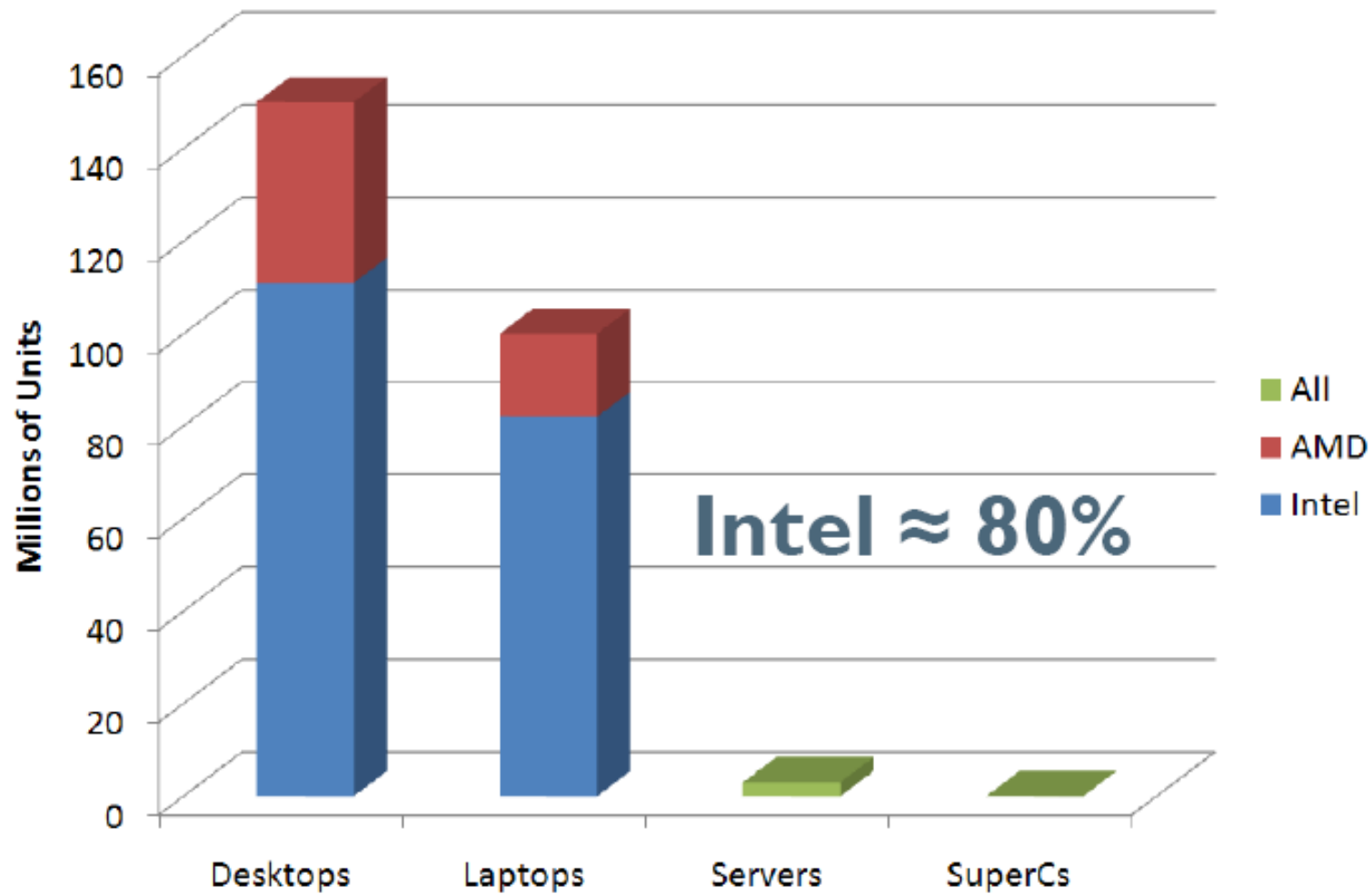


Processor Categories

- Coprocessors – accelerate some operations
 - SPUs inside Cell
- Microcontrollers
 - control physical systems – no real computations
- Microsequencers
 - control coprocessors & other units
- *Embedded system processors*
 - specialized CPUs – mobiles, mp3 players...
- General purpose processors
- System-on-chip
- FPGA

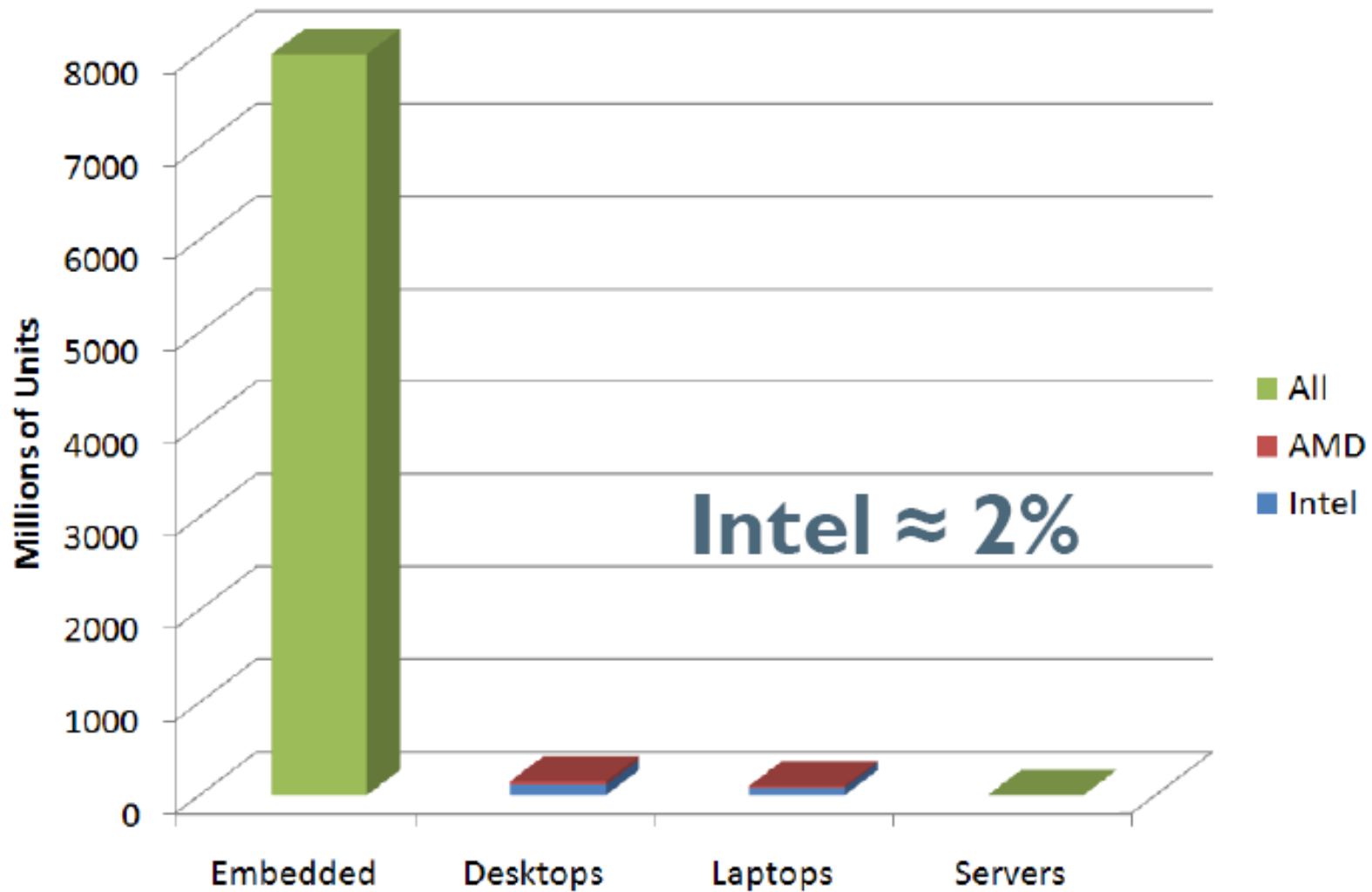
Microprocessor Market Shares

Illusion

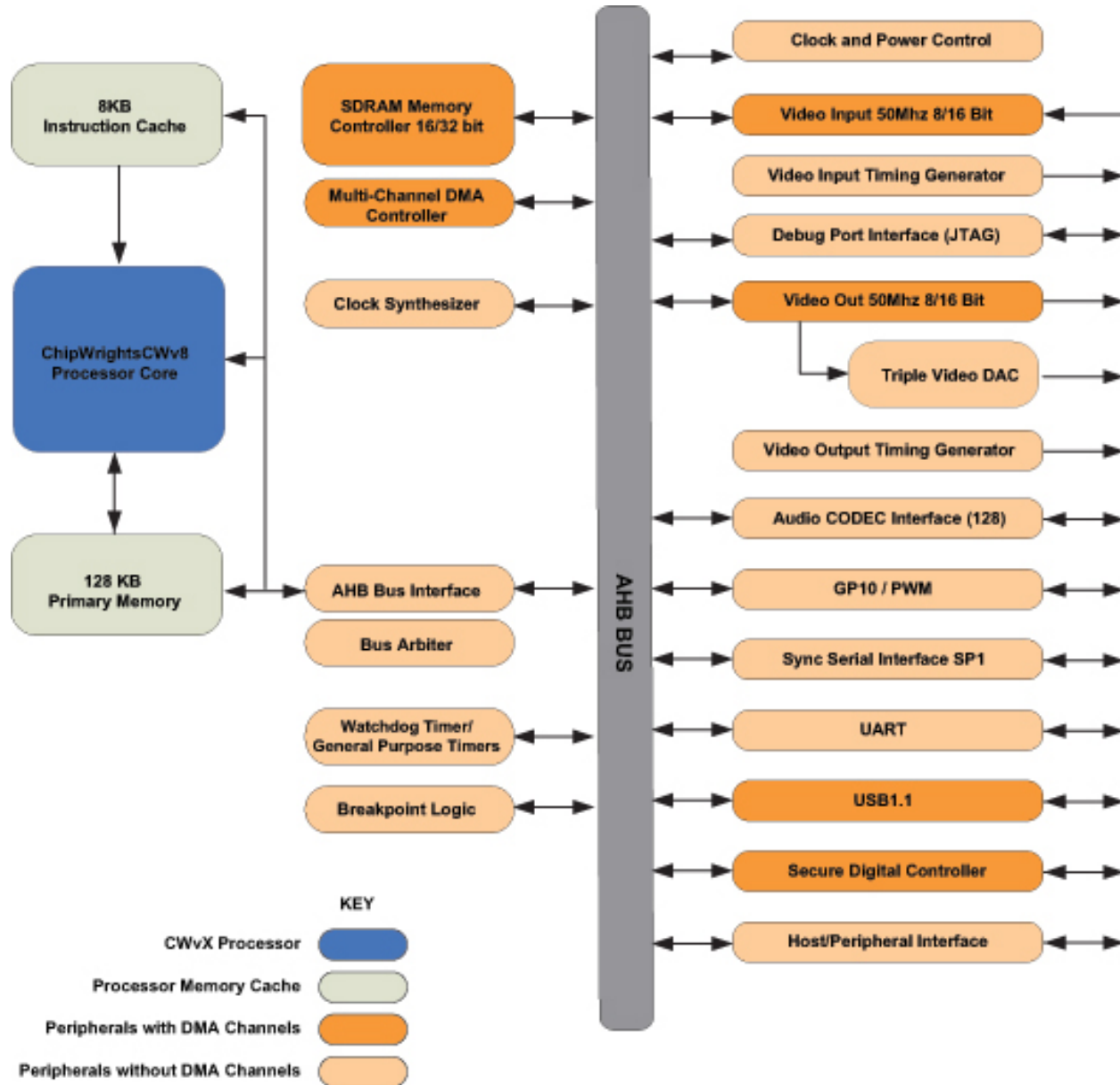


Microprocessor Market Shares

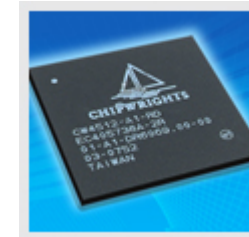
Reality



System-on-chip



CW4512



FPGA



Array of gates.
Configure the interconnect.



Stored Program

- Concept: programmable device if it is possible to change the program *independently* from the processor.
- Program stored in ROM/RAM.
 - How does a computer start? Bootstrap.
- Fetch-and-execute cycle: basis.
 - Move through program automatically.
 - repeat forever {
 fetch next instruction
 execute
}



Fetch-and-Execute

- When do you stop?
 - You don't – unless you shutdown the system.
 - OSs → execute infinite loop to schedule processes.
 - Embedded systems → infinite loop to probe inputs.
 - Sometimes possible to “pause” – standby.



Clock & Instruction Rate

- Clock rate:
 - clock for the gates of the processor
 - hint on speed
- Instruction rate:
 - IPC – how much work per cycle
 - depends on instructions & processors
 - linked to pipeline
- Which kind of instructions are available?
- How to execute them?