Introduction to Processor Architecture

Tassadaq Hussain
Riphah International University
Barcelona Supercomputing Center
Universitat Politècnica de Catalunya

Types of Computers

- Desktop Computer
- Servers
- Embedded Computer

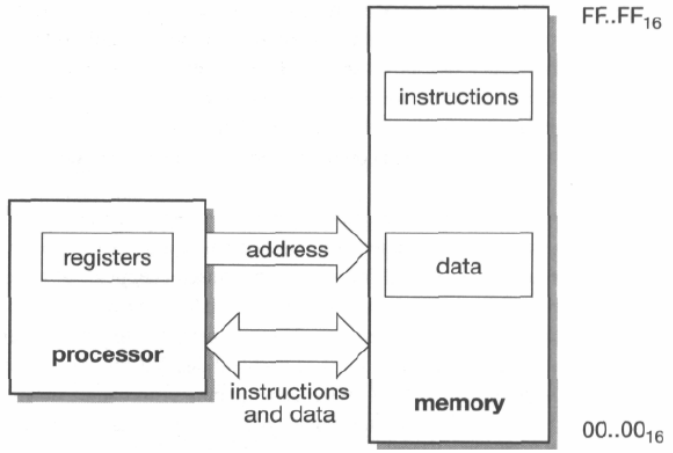
Feature	Desktop	Server	Embedded
Price of system	$500-$5000	$5000-S5,000,000	$10-$100,000 (including network routers at the high end)
Price of microprocessor module	$50-$500 (per processor)	$200-$ 10,000 (per processor)	$0.01-$100 (per processor)
Critical system design issues	Price-performance, graphics performance	Throughput, availability, scalability	Price, power consumption, application-specific performance

Defining Computer Architecture

- In the past, the term computer architecture often referred only to instruction set design.
- Now System Architecture is much more than instruction set design, and the technical hurdles in the other aspects.

Definition Processor

A general-purpose processor is a finite-state automaton that executes instructions held
in a memory.
The state of the system is defined by the values held in the memory locations together with the values held in certain registers within the processor itself Each instruction defines a particular way the total state should change and it also defines which instruction should be executed next.

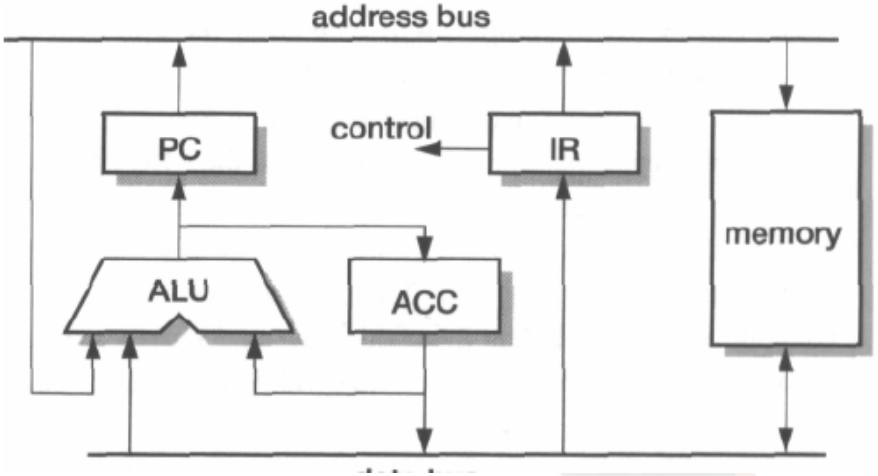


Computer System Design Hierarchy

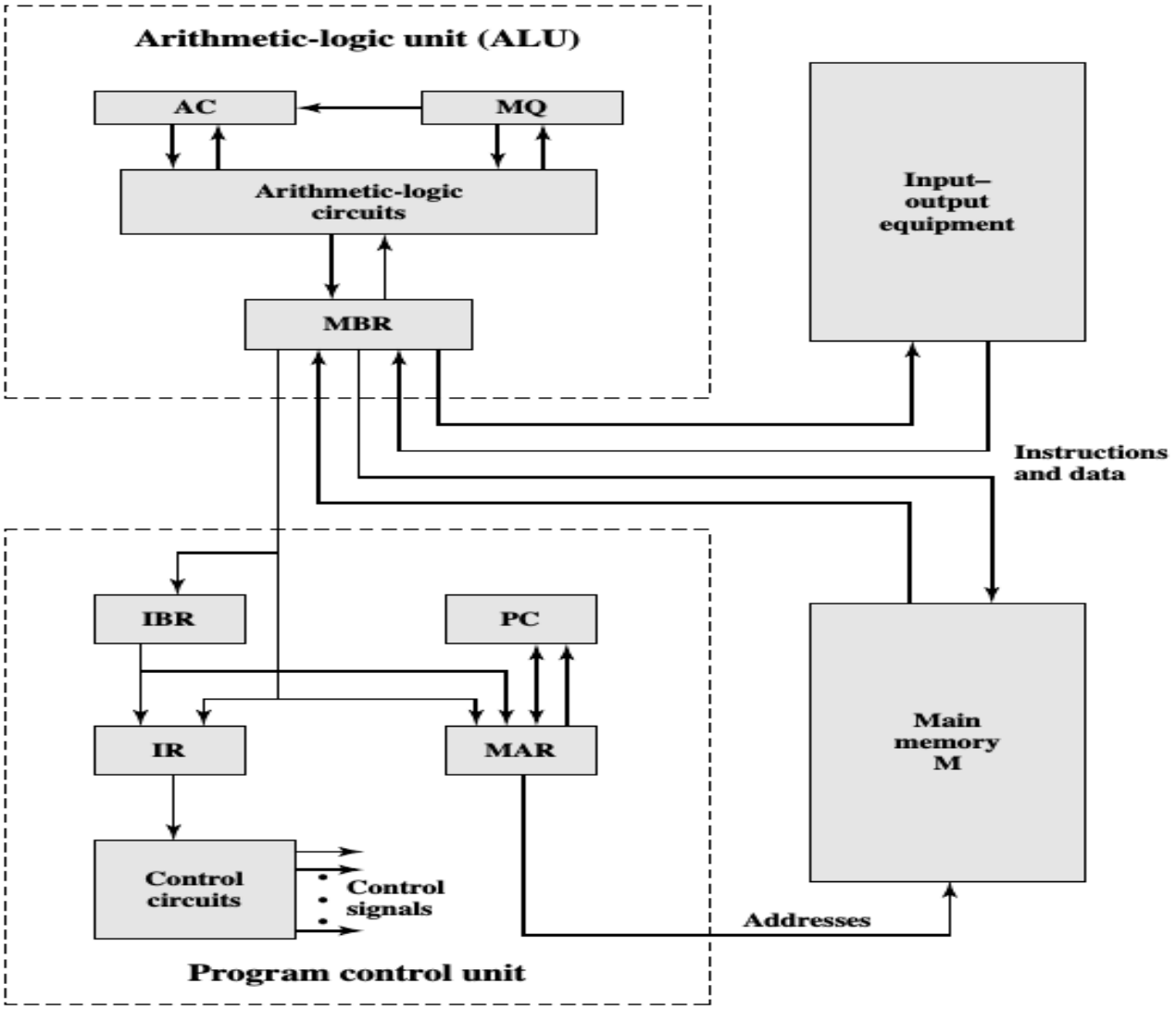
1. Transistors;
2. Logic gates, memory cells, special circuits;
3. Single-bit adders, multiplexers, decoders, flip-flops;
4. Word-wide adders, multiplexers, decoders, registers, buses;
5. ALUs (Arithmetic-Logic Units), barrel shifters, register banks, memory blocks;
6. Processor, cache and memory management organizations;
7. Processors, peripheral cells, cache memories, memory management units;
8. Integrated system chips;
9. Printed circuit boards;
10. Mobile telephones, PCs, engine controllers.

A Simple Processor

- A program counter (PC) register that is used to hold the address of the current instruction.
- A single register called an accumulator (ACC) that holds a data value while it is worked upon.
- An arithmetic-logic unit (ALU) that can perform a number of operations on binary operands, such as add, subtract, increment, and so on.
- An instruction register (IR) that holds the current instruction while it is executed; instruction decode and control logic that employs the above components to achieve the desired results from each instruction.



Basic Computer Architecture

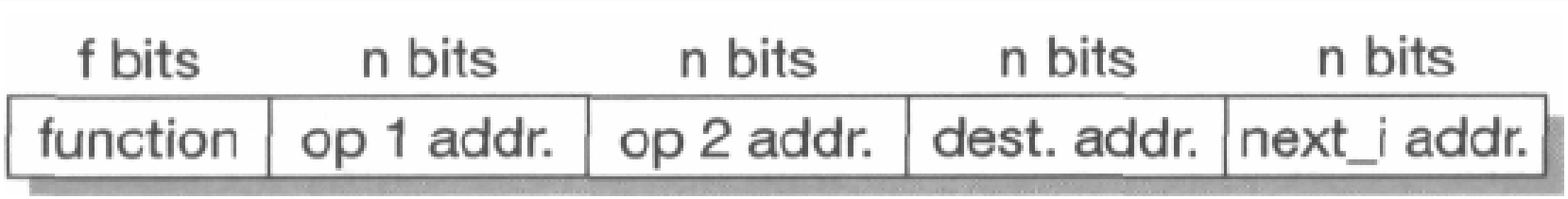


RIPHAH INTERNATIONAL UNIVERSITY

Basic Computer Register

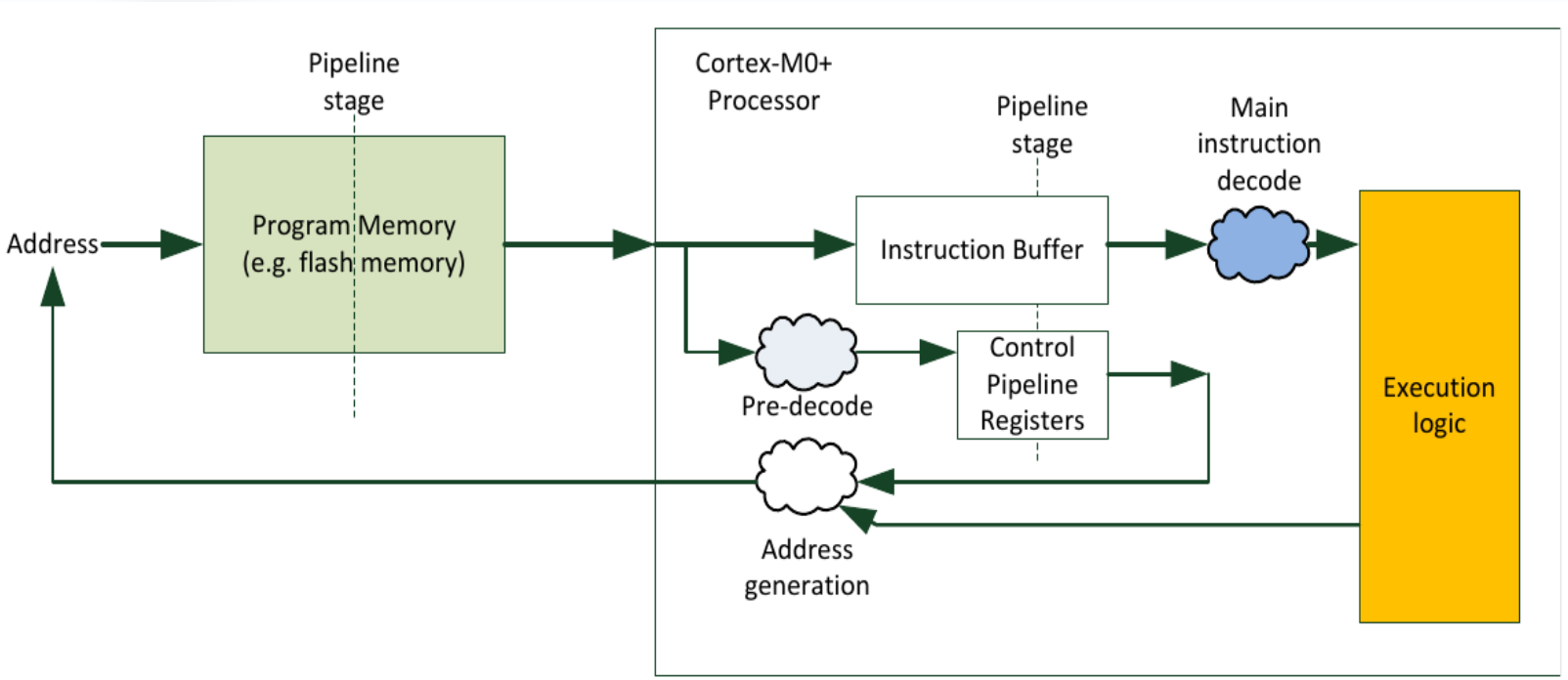
- **Memory buffer register (MBR)**: Contains a word to be stored in memory or sent to the I/O unit, or is used to receive a word from memory or from the I/O unit.
- **Memory address register (MAR)**: Specifies the address in memory of the word to be written from or read into the MBR.
- **Instruction register (IR)**: Contains the 8-bit opcode instruction being executed.
- **Instruction buffer register (IBR)**: Employed to hold temporarily the right-hand instruction from a word in memory.
- **Program counter (PC):** Contains the address of the next instruction-pair to be fetched from memory.
- **Accumulator (AC): and multiplier quotient (MQ)**: Employed to hold temporarily operands and results of ALU operations. For example, the result of multiplying two 40-bit numbers is an 80-bit number; the most significant 40 bits are stored in the AC and the least significant in the MQ.

Instruction set design



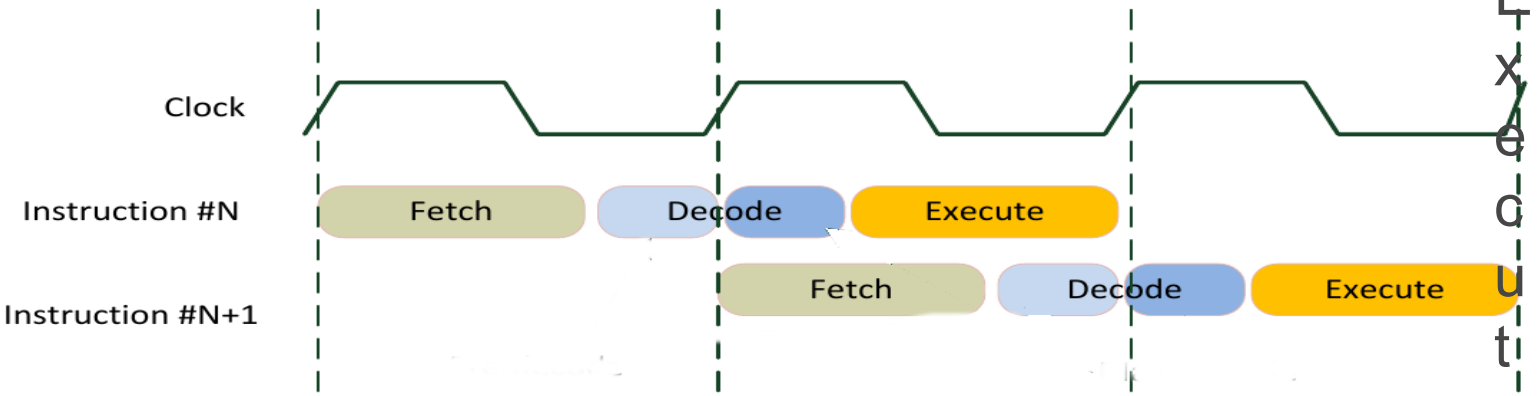
4-address instruction format
3-address instruction format
2-address instruction format
1-address instruction format
0-address instruction formate

Processor: Performance Improvement



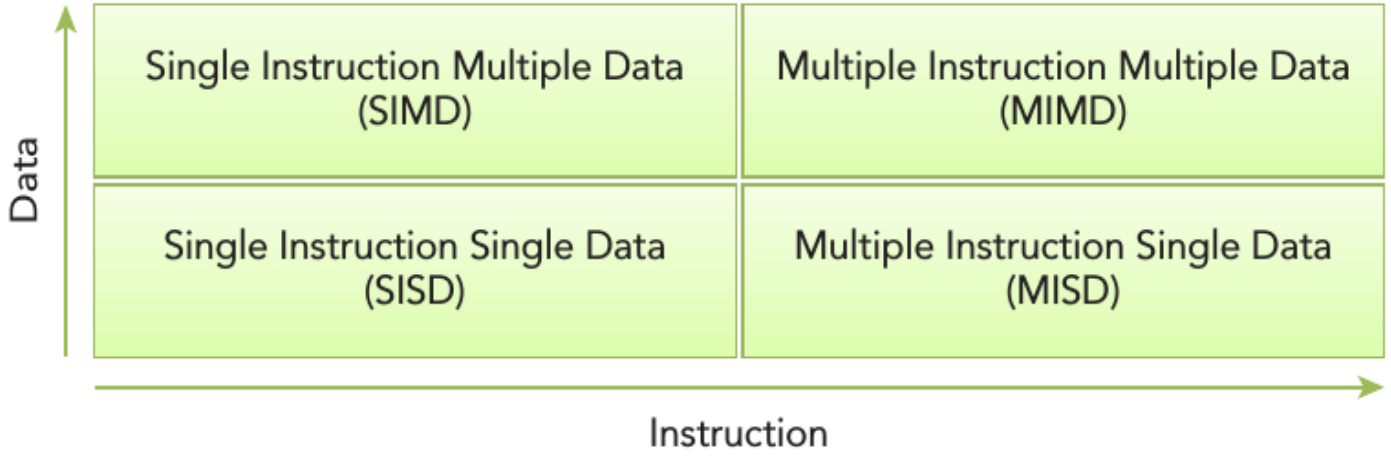
Speed and Performance

- Fetch
- Decode
- Execute
- Fetch
- Decode
- Execute



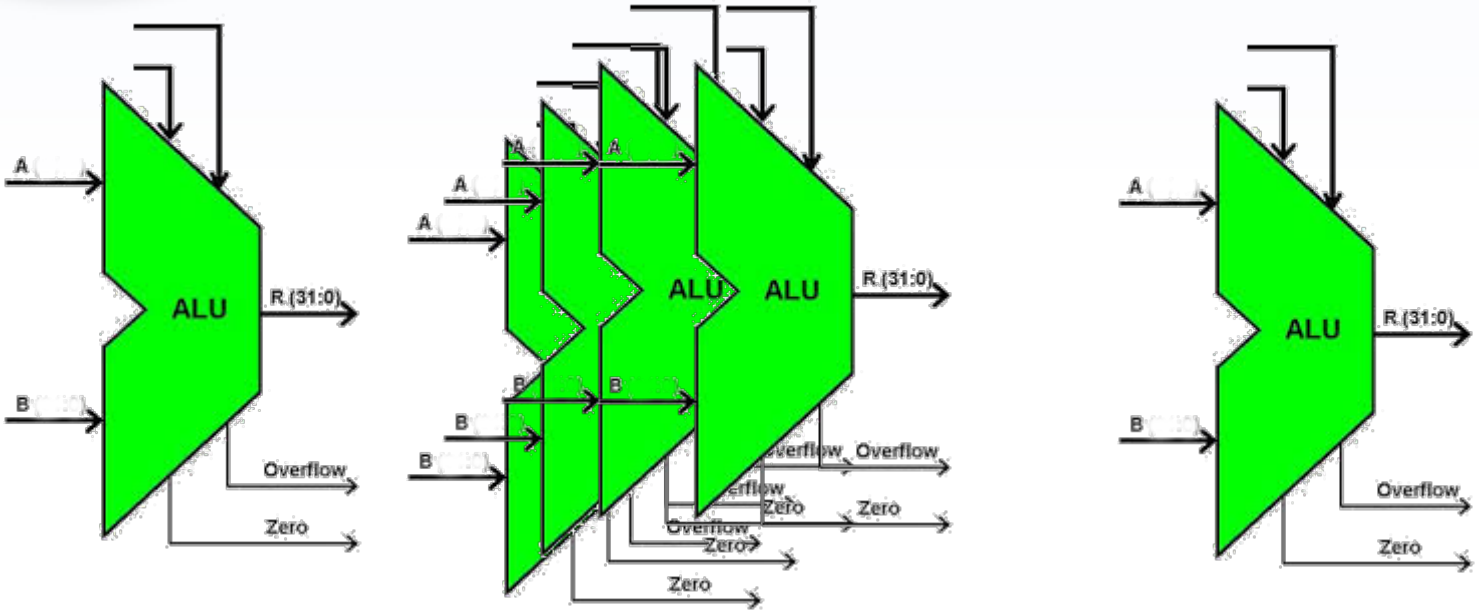
Processor Architectures

- Single Instruction Single Data (SISD)
- Single Instruction Multiple Data (SIMD)
- Multiple Instruction Single Data (MISD)
- Multiple Instruction Multiple Data (MIMD)



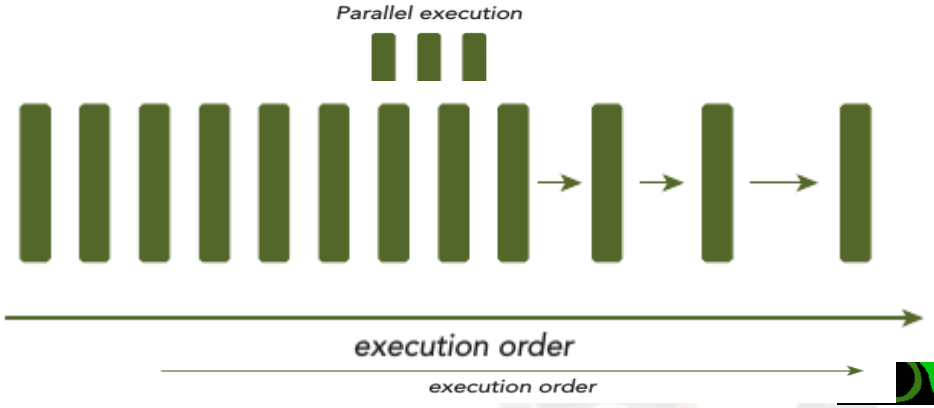
RIPHAH INTERNATIONAL UNIVERSITY

Performance Improvement

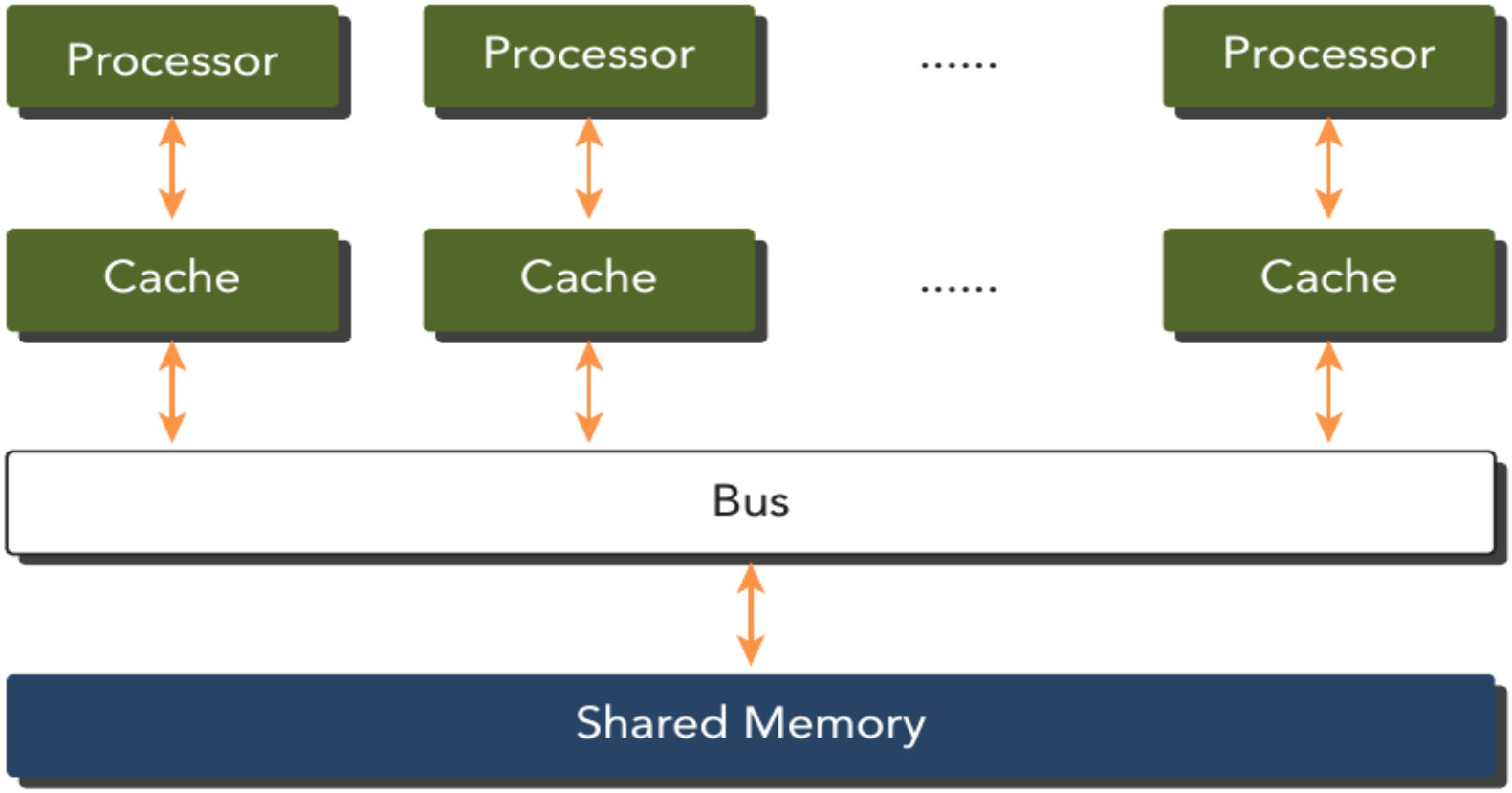


Instruction Level Parallelism

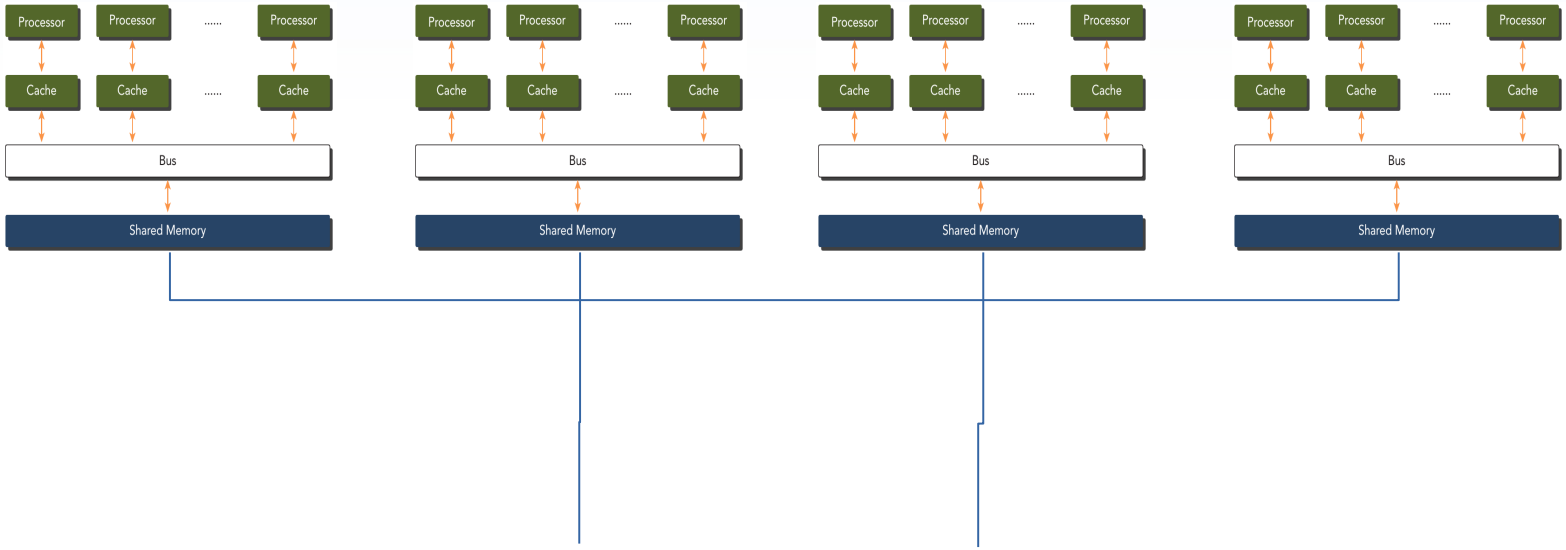
Parallel execution



Uni-core, Multi-core CPUs, Clusters, and Grid Computing



Grid Computing



Instruction set architecture (ISA)

- ISA is an abstract model of a computer also referred as computer architecture.
- An ISA permits multiple implementations that may vary in performance, physical size, and monetary cost.
- The ISA serves as the interface between software and hardware. Software that has been written for an ISA can run on different implementations of the same ISA.
- The ISA defines everything a machine language programmer needs to know in order to program a computer.

- The ISAs define the supported data types, what state there is (such as the main memory and registers) and their semantics (such as the memory consistency and addressing modes), the instruction set (the set of machine instructions that comprises a computer's machine language), and the input/output model.

Classification of ISAs

- A common classification is by architectural complexity.
- A **complex instruction set computer (CISC)** has many specialized instructions, some of which may only be rarely used in practical programs.
- A **reduced instruction set computer (RISC)** simplifies the processor by efficiently implementing only the instructions that are frequently used in programs, while the less common operations are implemented as subroutines, having their resulting additional processor execution time offset by infrequent use.
- Other types include **very long instruction word (VLIW)** architectures, and the closely related long instruction word (LIW) and explicitly parallel instruction computing (EPIC) architectures.
  - These architectures seek to exploit instruction-level parallelism with less hardware than RISC and CISC by making the compiler responsible for instruction issue and scheduling.

Instruction types

- Data handling and memory operations
- Arithmetic and logic operations
- Control flow operations
- Coprocessor instructions
- Complex instructions

ARM Instruction Set

- All instructions are 32 bits long
  - Most instructions execute in a single cycle.
  - Most instructions can be conditionally executed.
  - A load/store architecture
  - Data processing instructions act only on registers
  - Three operand format
  - Combined ALU and shifter for high speed bit manipulation
  - Specific memory access instructions with powerful auto-indexing addressing modes.
- 32 bit and 8 bit data types
  - and also 16 bit data types on ARM Architecture v4.
- Flexible multiple register load and store instructions
- Instruction set extension via coprocessors
- Very dense 16-bit compressed instruction set (Thumb)