ECE-425: LECTURE 1 INTRODUCTION TO COMPUTER ARCHITECTURE

PROF: MOHAMED EL-HADEDY

EMAIL: MEALY@CPP.EDU

OFFICE: 909-869-2594

OBJECTS OF ECE-425

- IN-DEPTH UNDERSTANDING OF THE INNER-WORKINGS OF COMPUTING SYSTEMS, THEIR EVOLUTION, AND TRADE-OFFS PRESENT AT THE HARDWARE/SOFTWARE BOUNDARY.
  - INSIGHT INTO FAST/SLOW OPERATIONS THAT ARE EASY/HARD TO IMPLEMENTATION HARDWARE
- UNDERSTANDING OF THE *DESIGN PROCESS* IN THE CONTEXT OF A LARGE COMPLEX COMPUTING SYSTEMS DESIGN.
  - FUNCTIONAL SPEC --> CONTROL & DATA-PATH DESIGN--
    --> PHYSICAL IMPLEMENTATION

Computer Design

Instruction Set Deign

- **Machine Language**
- **Compiler View**
- **"Computer Architecture"**
- **"Instruction Set Processor"**

"Building Architect"

COMPUTER HARDWARE DESIGN

- MACHINE IMPLEMENTATION
- LOGIC DESIGNER'S VIEW
- "PROCESSOR ARCHITECTURE"
- "COMPUTER ORGANIZATION"

"CONSTRUCTION ENGINEER"

ORGANIZATION

ISA Level

FUs & Interconnect

Logic Designer's View

-- CAPABILITIES & PERFORMANCE CHARACTERISTICS OF PRINCIPAL FUNCTIONAL UNITS
(E.G., REGISTERS, ALUS, SHIFTERS, LOGIC UNITS, ETC.

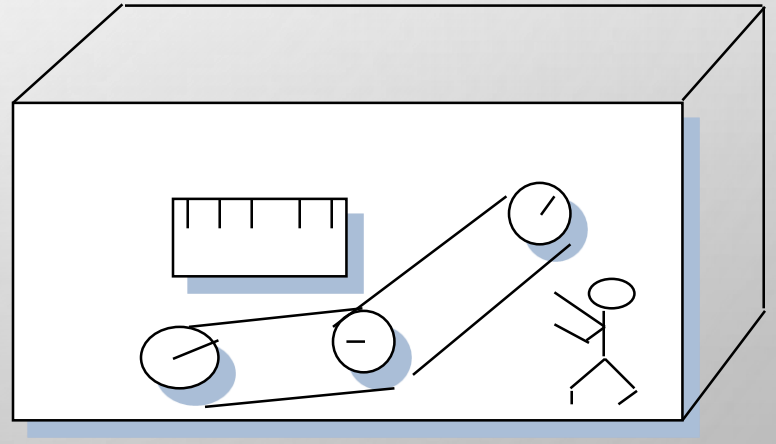
-- WAYS IN WHICH THESE COMPONENTS ARE INTERCONNECTED

-- NATURE OF INFORMATION FLOWS BETWEEN COMPONENTS

-- LOGIC AND MEANS BY WHICH
SUCH INFORMATION FLOW IS CONTROLLED.

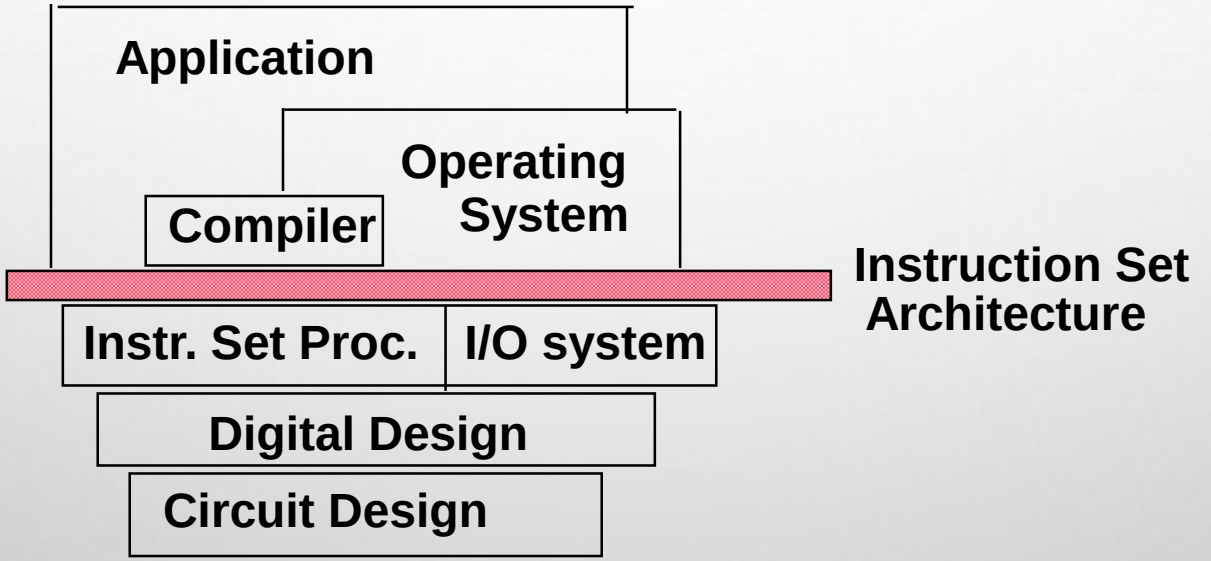
CHOREOGRAPHY OF FUS TO REALIZE THE ISA

REGISTER TRANSFER LEVEL DESCRIPTION



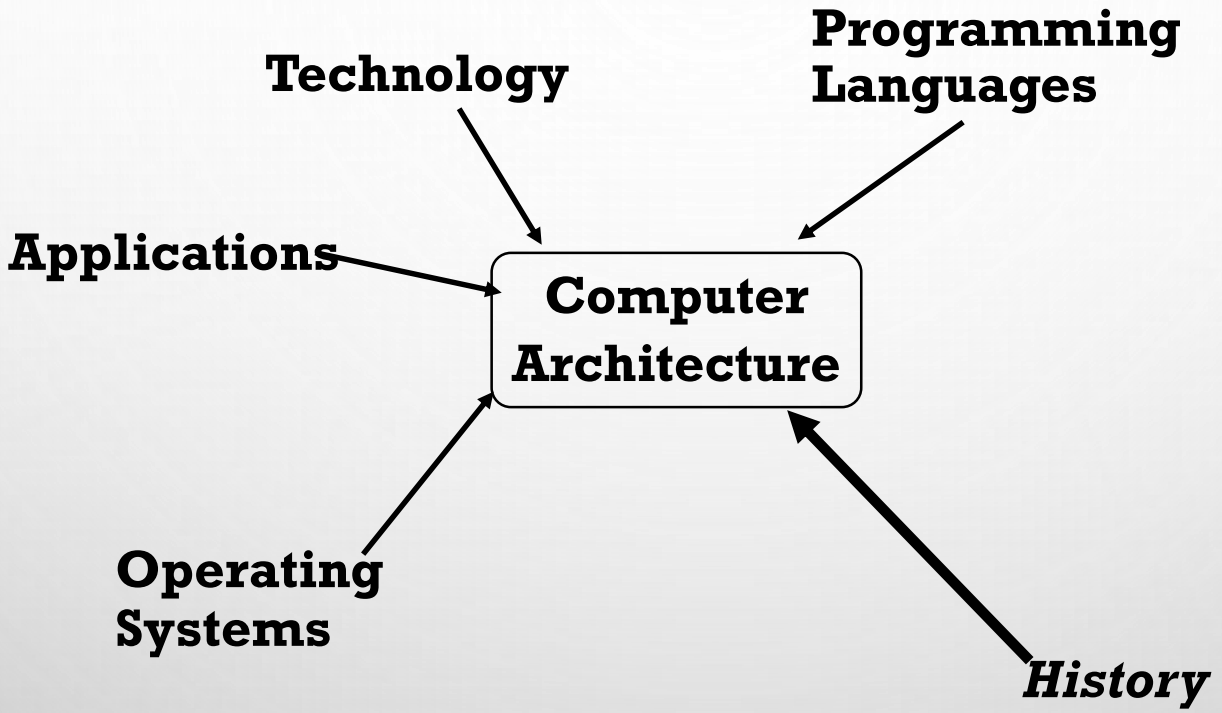
WHAT IS "COMPUTER ARCHITECTURE" ?
A SYSTEM CONCEPT INTEGRATING SOFTWARE, HARDWARE, AND FIRMWARE TO SPECIFY THE DESIGN OF COMPUTING SYSTEMS

° **Co-ordination of *levels of abstraction***

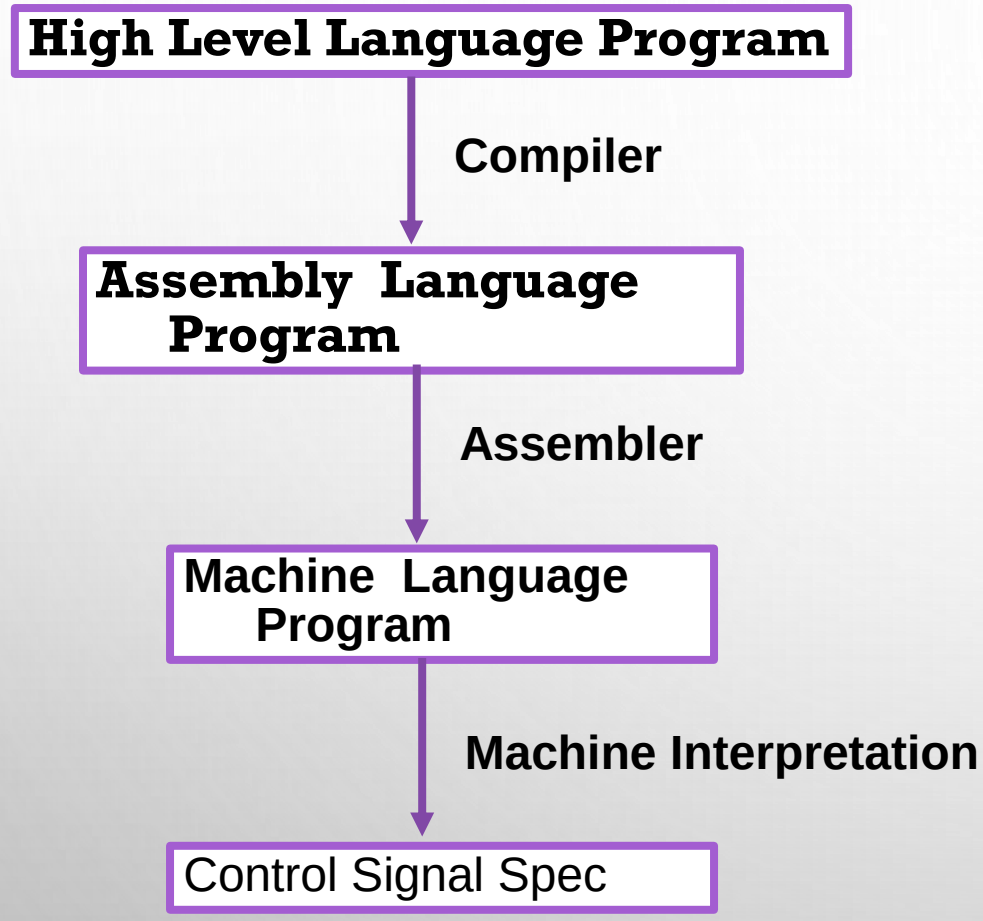


° **Under a set of rapidly changing *Forces***

FORCES ON COMPUTER ARCHITECTURE



LEVELS OF REPRESENTATION




```
temp = v[k];
v[k] = v[k+1];
v[k+1] = temp;
```

```
LW $15,   0($2)
LW $16,   4($2)
SW        $16,   0($2)
SW        $15,   4($2)
```

```
0000 1001 1100 0110 1010 1111 0101 1000
1010 1111 0101 1000 0000 1001 1100 0110
1100 0110 1010 1111 0101 1000 0000 1001
0101 1000 0000 1001 1100 0110 1010 1111
```

Technology used in computers

Year	Technology used in Computers	Relative performance/unit cost
1951	Vacuum tube	1
1965	Transistor	35
1975	Integrated circuit (IC)	900
1995	Very large-scale integrated circuit (VLSI)	2,400,000
2005	Ultra large-scale integrated circuit (ULSIC)	6,200,000,000

- **Vacuum tube:** mostly rely on thermionic emission of electrons from a hot filament or heated cathode.
- **Transistor:** is simply on/off switch controlled by electricity.
- **Integrated circuit (IC):** combined dozens to hundreds of transistors into a single chip
- **Very large-scale integrated circuit :** increasing number of transistors from hundreds to millions
- **Ultra large-scale integrated circuit:** (can be count as VLSI but with more transistors in one chip) (more than one million transistors is considered a ULSI implementation. Intel 486 and the Pentium series of processors were built on ULSI principles

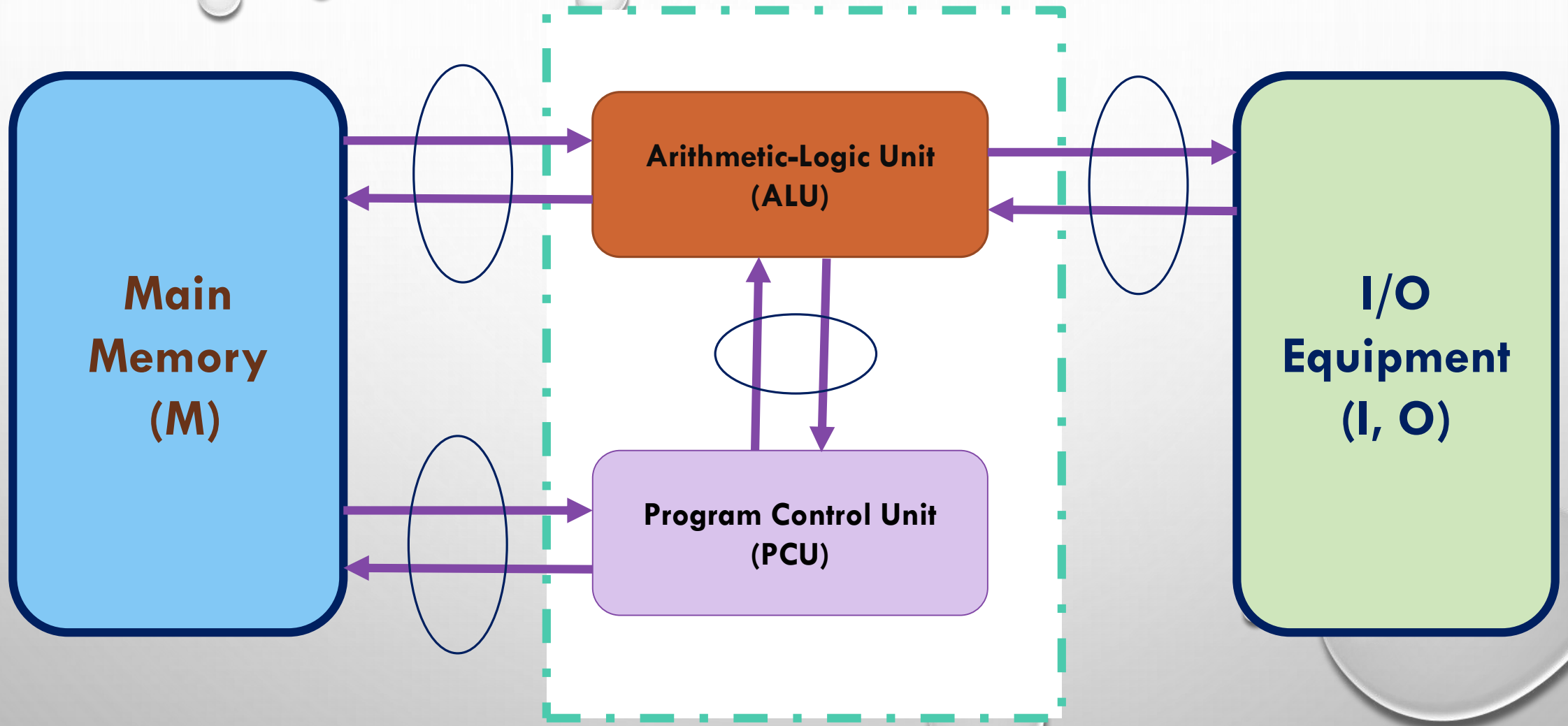
Computer History

- **Vacuum tube (1942-1963) ➔** most famous (ENIAC (1946), SEAC (1950), SWAC(1950), LEO 1(1951), IAS machine (1951), Harvard Mark IV (1952), FLAC (1953), IBM 702 (1953), Srela computer (1953, Soviet Union), BESK (1954, Sweden), PERM (1956), FUJIC (1956), RCA BIZMAC (1956, First commercial computer), DASK (1957), IBM 709 (1958), ORDA 1001 (1960), UMC-1 (1962, Poland), BRLESC (1962, 1727 tubes, and 853 transistors), OSAGE (1963).

History of Computers

- John von Neumann (Electronic Discrete Variable Computer)
  - ✓ First publication of the idea was in 1945
  - ✓ Stored program concept
    - ❑ Attributed to ENIAC designers, most notably the mathematician John von Neumann
    - ❑ Program represented in a form suitable for storing in memory alongside the data
  - ✓ Institute for Advanced Study (IAS) computer
    - ❑ Princeton Institute for Advanced Studies
    - ❑ Prototype of all subsequent general-purpose computers
    - ❑ Completed in 1952

Structure of Von Neumann Machine



Structure of IAS Computer

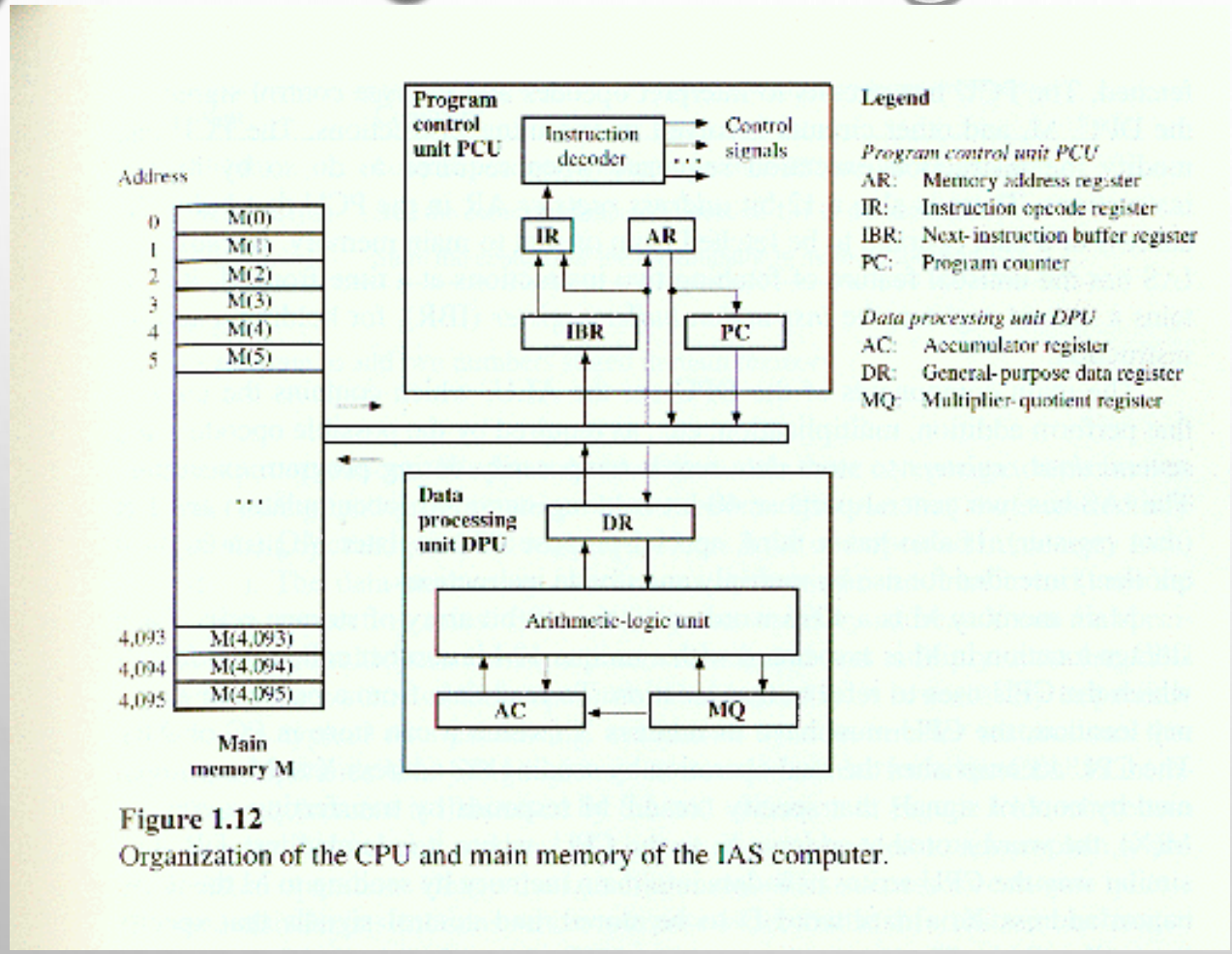
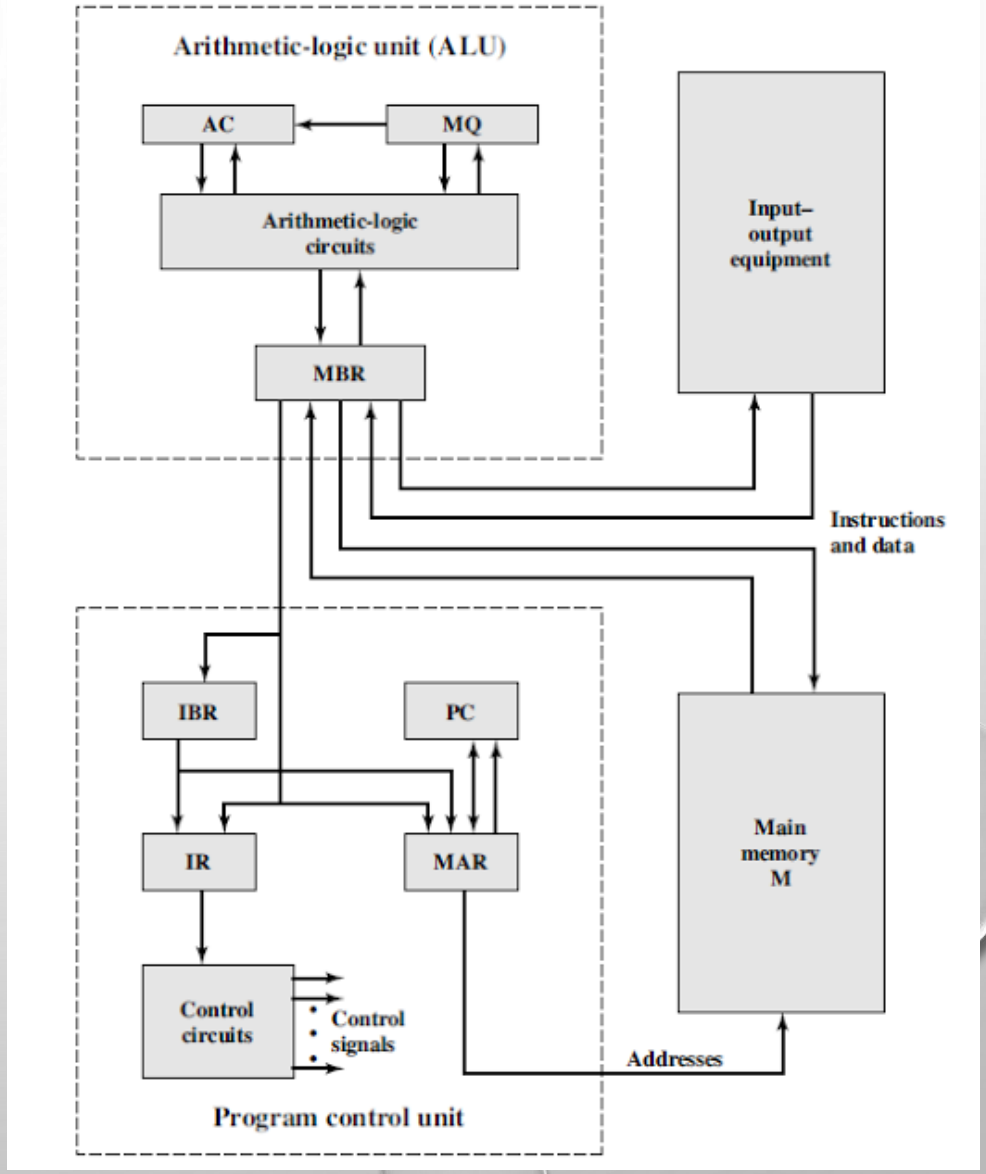


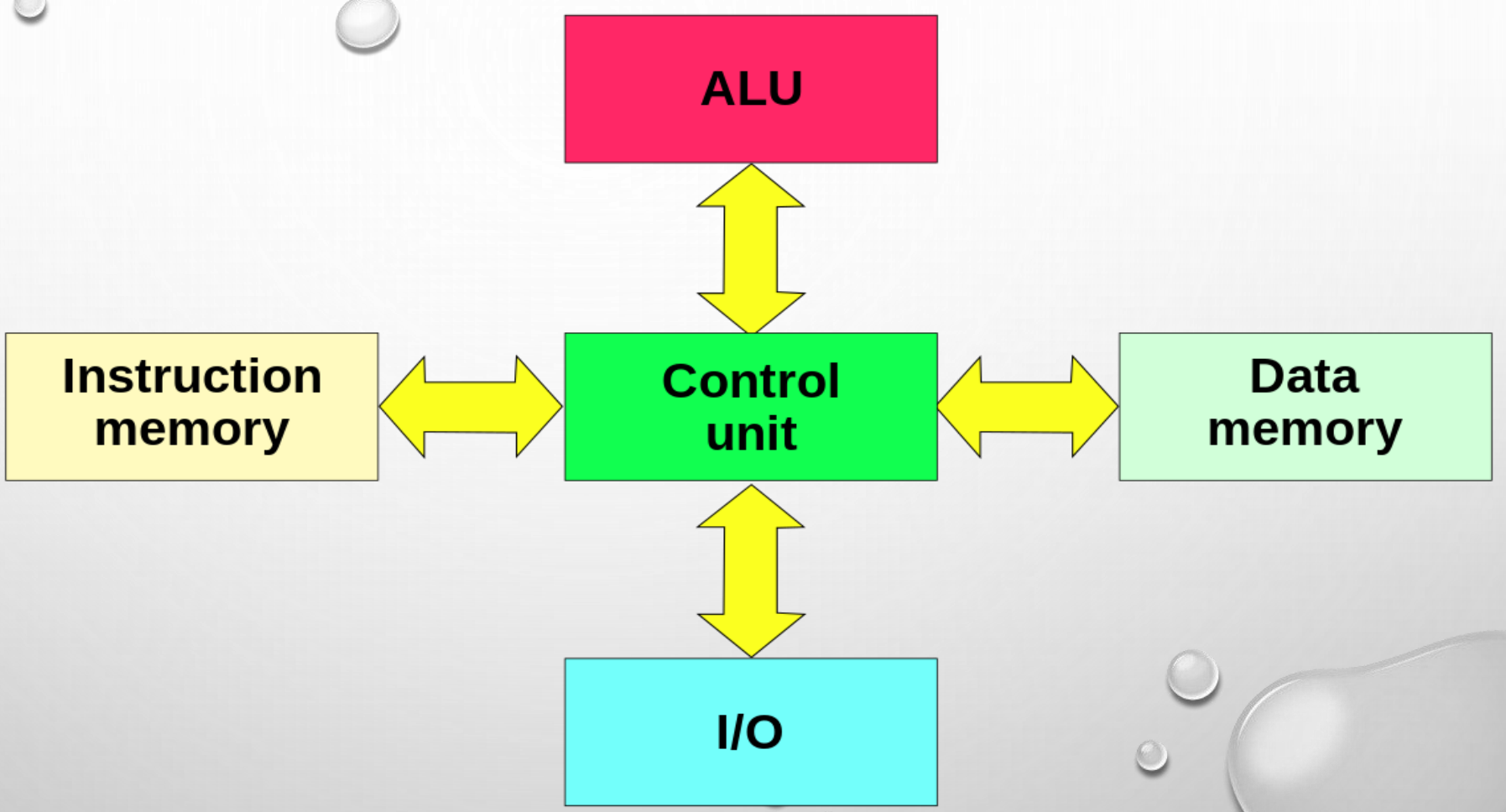
Figure 1.12
Organization of the CPU and main memory of the IAS computer.



REGISTERS

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 temporarily hold 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 temporarily hold operands and results of ALU operations

Harvard Architecture



In a Harvard Architecture machine, the computer system's memory is separated into two discrete parts: data and instructions. In a pure Harvard system, the two different memories occupy separate memory modules, and instructions can only be executed from the instruction memory.

Modern Computers

- Modern Computers, especially computers based on the intel X86 ISA are not Harvard Computers
- Both program instructions, and data are stored in the same RAM areas (**Von Neumann** Architecture)
- **Paging:** Most of the modern computers allows the physical memory to be segmented into large blocks of memory called “pages” each page of memory can either be instructions or data, but not both

RISC vs CISC (ISA architecture)

- **RISC (reduced instruction set computer) instructions**
  - ➢ Only load/store instructions access memory
  - ➢ Data (i.e., operands) must be in registers to perform operation
  - ➢ Each instruction roughly taking same amount of time
  - ➢ Simple addressing modes
  - ➢ Virtually all new instruction sets since 1982 have been RISC (M 68000 announced in year 1980)
- **CISC (Complex instruction set computer) instructions**
  - ✓ ALU/Logic instructions access memory to fetch operands
  - ✓ Load/store instructions access memory
  - ✓ Some instructions' execution time is much longer than other instructions
  - ✓ Complex addressing modes

RISC vs. CISC

RISC vs. CISC

Parameter	RISC	CISC
Instruction types	Simple	Complex
Number of instructions	Reduced (30-40)	Extended (100-200)
Duration of an instruction	One cycle	More cycles (4-120)
Instruction format	Fixed	Variable
Instruction execution	In parallel (pipeline)	Sequential
Addressing modes	Simple	Complex
Instructions accessing the memory	Two: Load and Store	Almost all from the set
Register set	multiple	unique
Complexity	In compiler	In CPU (micro-program)

RISC vs. CISC

Complex Instruction Set Computer	Reduced Instruction Set Computers
Many instructions • (e.g., 75-100)	Few instructions • (e.g., 30-40)
Many Instructions are macro-like	Smaller chip, smaller pin count, & very low-power consumption • Simple but fast instructions
Most microcontrollers are based on CISC concept • e.g., PDP-11, VAX, Motorola 68K • PIC is an eception	Harvard architecture, instruction pipelining
Von-Neumann architecture	Industry trend for microprocessor design • e.g., Intel Pentium, PIC • Harvard architecture