VIT-AP UNIVERSITY

MODULE-I : LECTURE-5

Introduction to Microprocessors and Microcontrollers

Dr. Subhasish Mahapatra

Sr. Assistant Professor

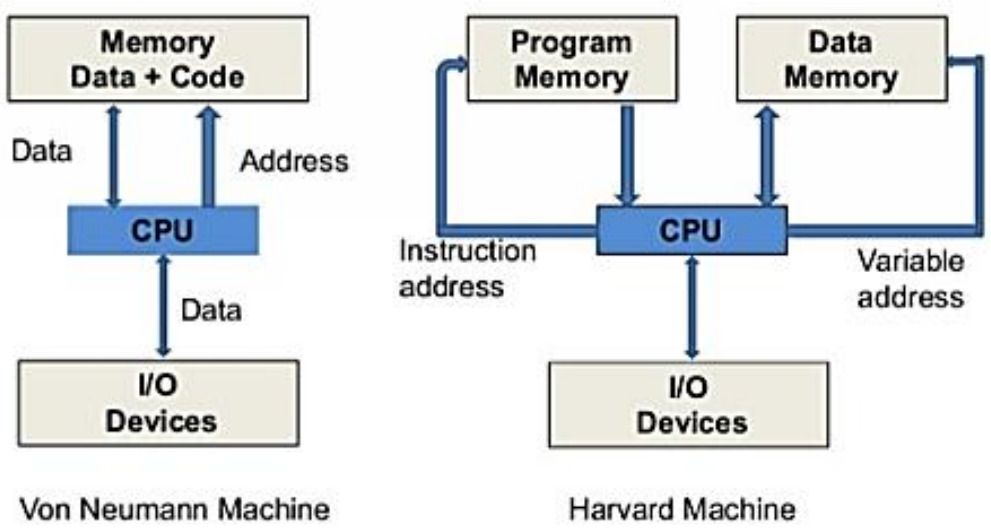
School of Electronics Engineering (SENSE)

VIT-AP University

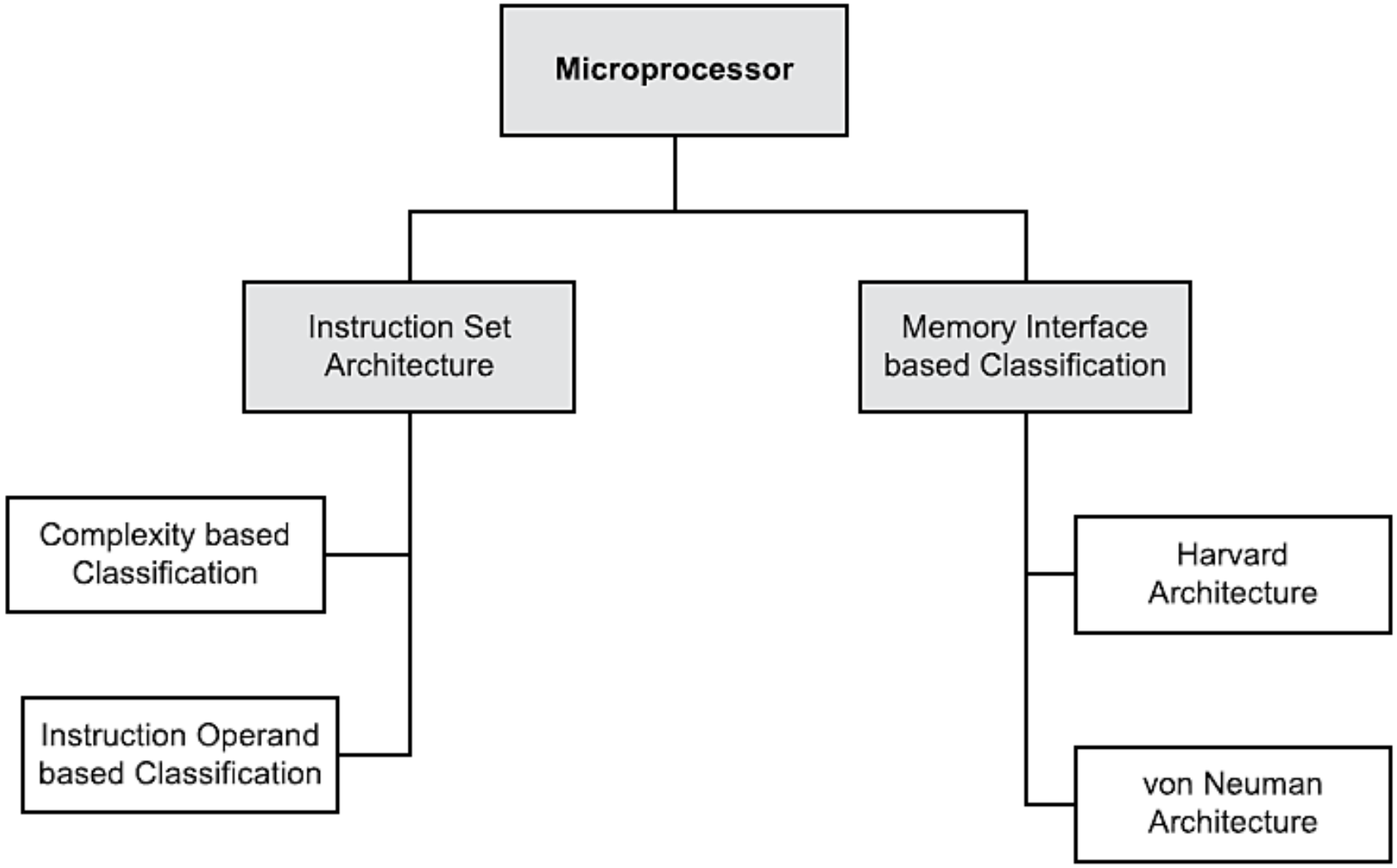
E-Mail: subhasish.m@vitap.ac.in

CONTENT

- ✓ **Classification of Microprocessor Architecture**



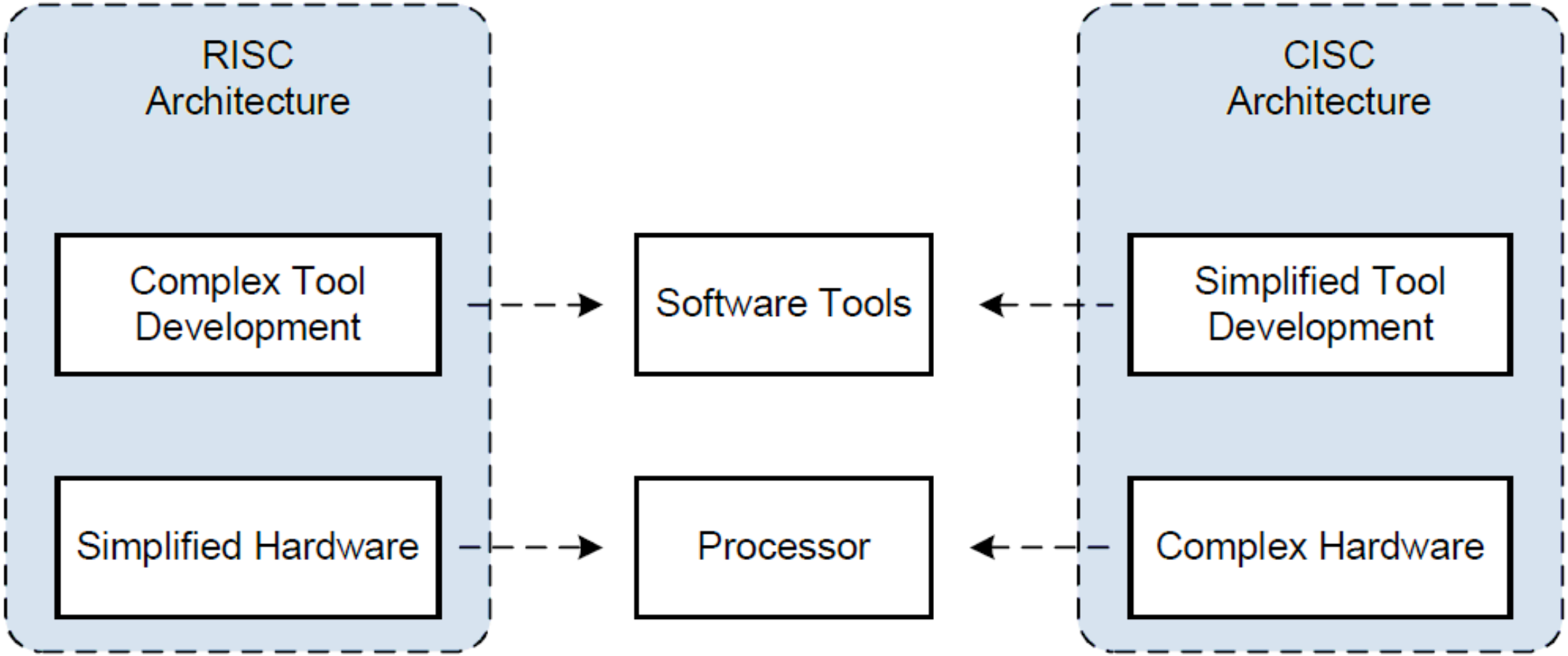
MICROPROCESSOR ARCHITECTURE CLASSIFICATION



Complexity - based ISA Classification

- ✓ Using the complexity of instructions, it can be again categorised into two types
  - Complex Instruction Set Computer (CISC)
  - Reduced Instruction Set Computer (RISC)

COMPLEXITY-BASED ISA CLASSIFICATION



COMPLEXITY-BASED ISA CLASSIFICATION

Reduced Instruction Set Computer (RISC)	Complex Instruction Set Computer (CISC)
▪ RISC processors have simple instructions taking about one clock cycle. The average clock cycle per instruction (CPI) is 1.5.	▪ CSIC processor has complex instructions that take up multiple clocks for execution. The average clock cycle per instruction (CPI) is in the range of 2 and 15.
▪ Performance is optimized with more focus on software	▪ Performance is optimized with more focus on hardware
▪ It has no memory unit and uses separate hardware to implement instructions	▪ It has a memory unit to implement complex instructions
▪ It has a hard-wired unit of programming	▪ It has a microprogramming unit
▪ The instruction set is reduced i.e. it has only a few instructions in the instruction set. Many of these instructions are very primitive.	▪ The instruction set has a variety of different instructions that can be used for complex operations.

COMPLEXITY-BASED ISA CLASSIFICATION

Reduced Instruction Set Computer (RISC)	Complex Instruction Set Computer (CISC)
▪ The instruction set has simple addressing modes.	▪ CISC has many different addressing modes and can thus be used to represent higher-level programming language statements more efficiently.
▪ Complex addressing modes are synthesized using the software.	▪ CISC already supports complex addressing modes
▪ Multiple register sets are present	▪ Only has a single register set
▪ RISC processors are highly pipelined	▪ They are normally not pipelined or less pipelined
▪ The complexity of RISC lies with the compiler that executes the program	▪ The complexity lies in the microprogram
▪ Execution time is very less	▪ Execution time is very high

COMPLEXITY-BASED ISA CLASSIFICATION

Reduced Instruction Set Computer (RISC)	Complex Instruction Set Computer (CISC)
▪ Code expansion can be a problem	▪ Code expansion is not a problem
▪ The decoding of instructions is simple	▪ Decoding of instructions is complex
▪ It does not require external memory for calculations	▪ It requires external memory for calculations
▪ The most common RISC microprocessors are Alpha, ARC, ARM, AVR, MIPS, PA-RISC, PIC, Power Architecture, and SPARC.	▪ Examples of CISC processors are the System/360, VAX, PDP-11, Motorola 68000 family, AMD, and Intel x86 CPUs.
▪ RISC architecture is used in high-end applications such as video processing, telecommunications, and image processing.	▪ CISC architecture is used in low-end applications such as security systems, home automation, etc.

Instruction Operand-Based ISA Classification

- ✓ Instructions in an assembly language program in general have multiple operands.
- ✓ The operands for an instruction can be specified either using memory or registers or combination of both.
- ✓ The ISA classification based on how the instruction operands are specified can be categorized in the following groups
    1. Memory-memory
    2. Register-memory
    3. Register-register

INSTRUCTION OPERAND-BASED ISA CLASSIFICATION

Memory-memory

- ✓ This type of ISA allows more than one operand of most instructions to be specified in memory.
- ✓ Examples: VAX and PDP series

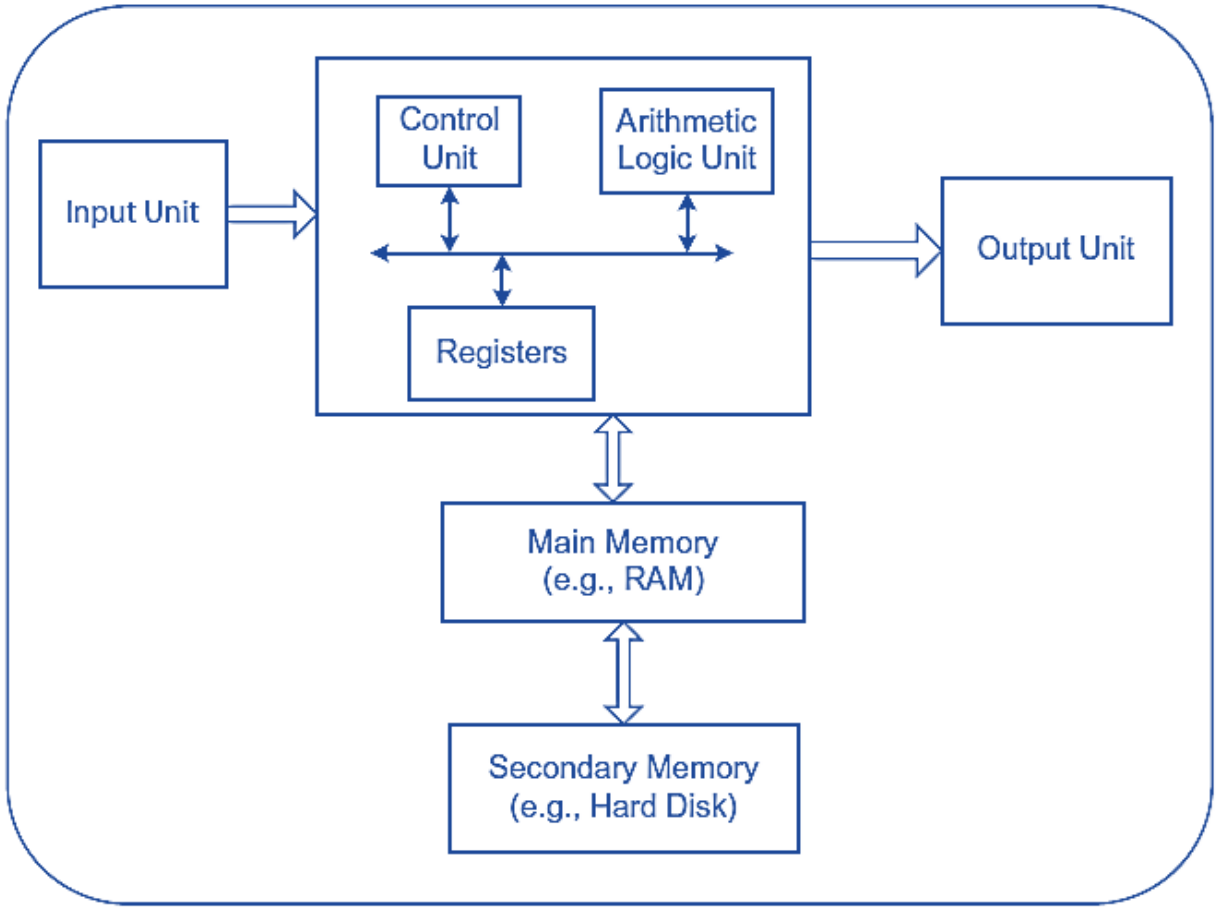
Register-memory

- ✓ These architectures allow one operand of an instruction to be specified in memory, while the other operand is in CPU register.
- ✓ Examples: x86 and Motorola 68k

Register-register

- ✓ This ISA classification is also called load-store architecture. Only load and store instruction can access the memory.
- ✓ All instructions other than load and store instructions get their operands from and store their results to registers.
- ✓ Examples: ARM and MIPS

INSTRUCTION OPERAND-BASED ISA CLASSIFICATION



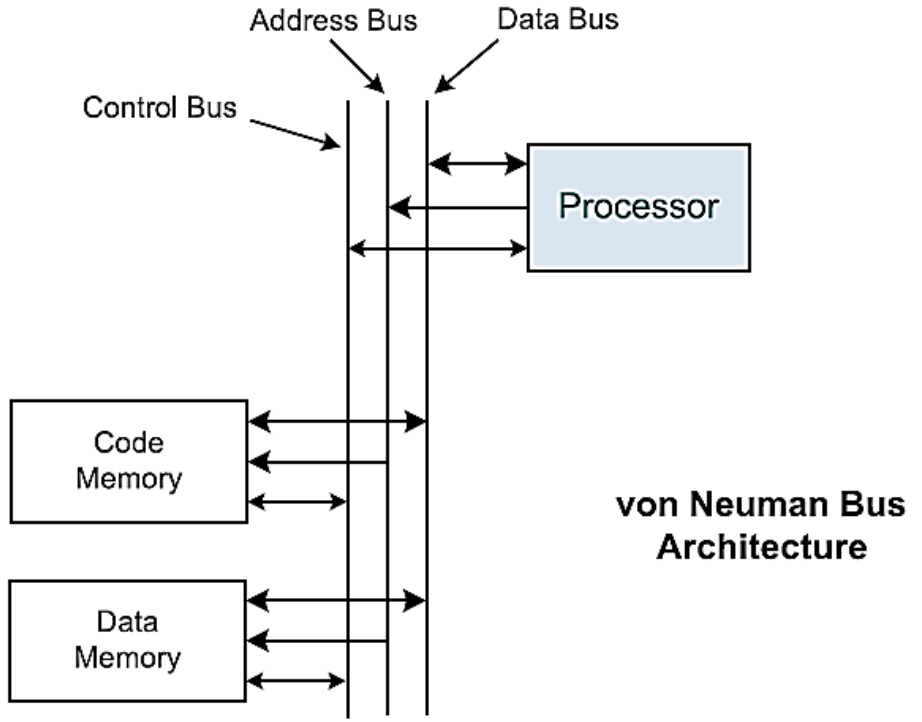
Memory-memory
MOV [1000H], [3000H]
ADD [R1], [2000H]
	Slower Speed

Register-memory
MOV R1, [1050H]
SUB [2020H], R2
	Medium Speed

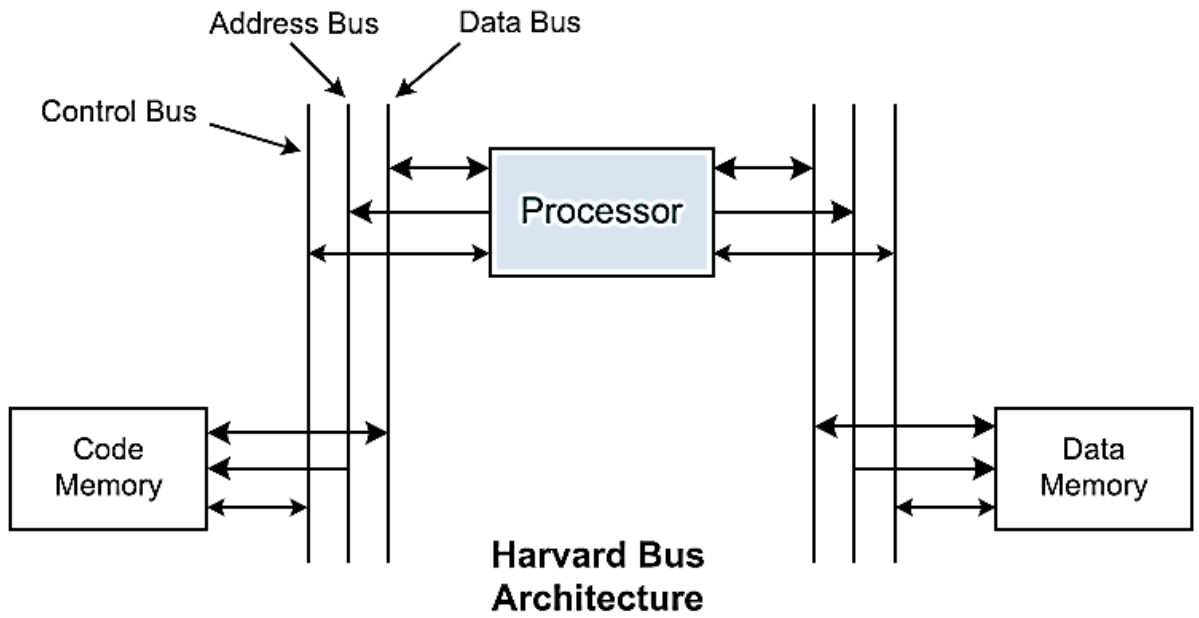
Register-register
MOV R1, R4
ADD R2, R3, R1
	Faster Speed

MEMORY INTERFACE-BASED ARCHITECTURE CLASSICATION

- ✓ There are two widely used memory interface architectures
  1. von Neumann architecture
  2. Harvard architecture



von Neuman Bus Architecture



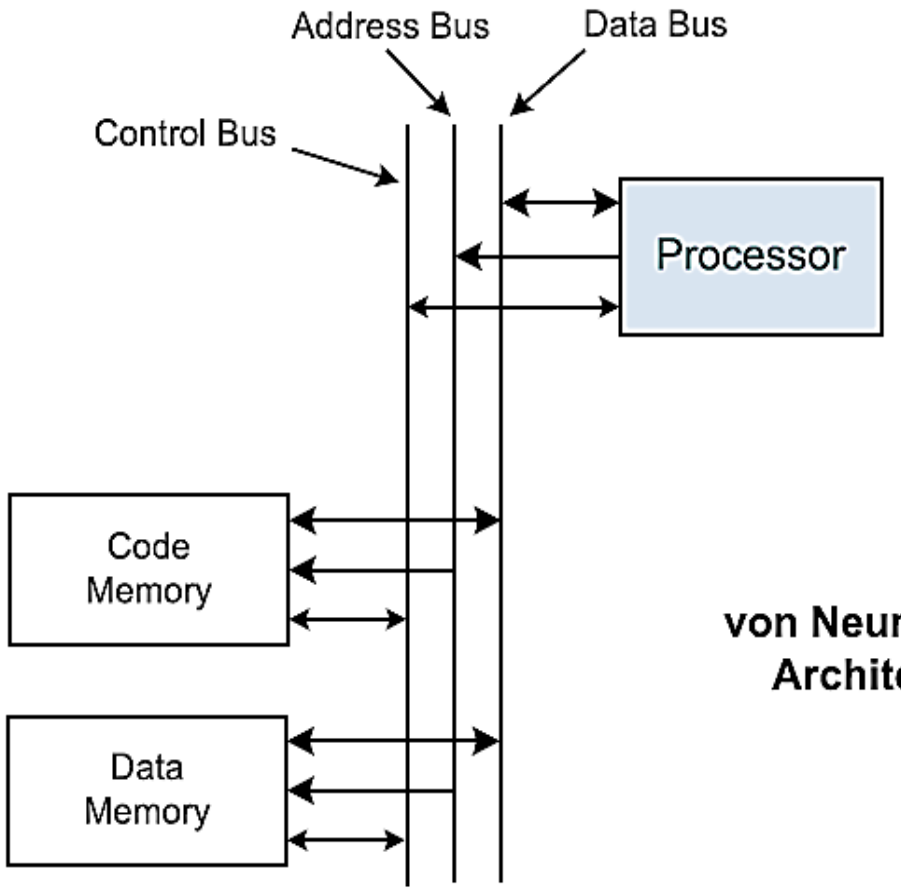
Harvard Bus Architecture

MEMORY INTERFACE-BASED ARCHITECTURE CLASSICATION

- ✓ Mostly, microprocessors are implemented using von Neumann architecture, while microcontrollers use Harvard architecture.
- ✓ ARM based microcontrollers use Harvard architecture.

VON-NEUMAN BUS ARCHITECTURE

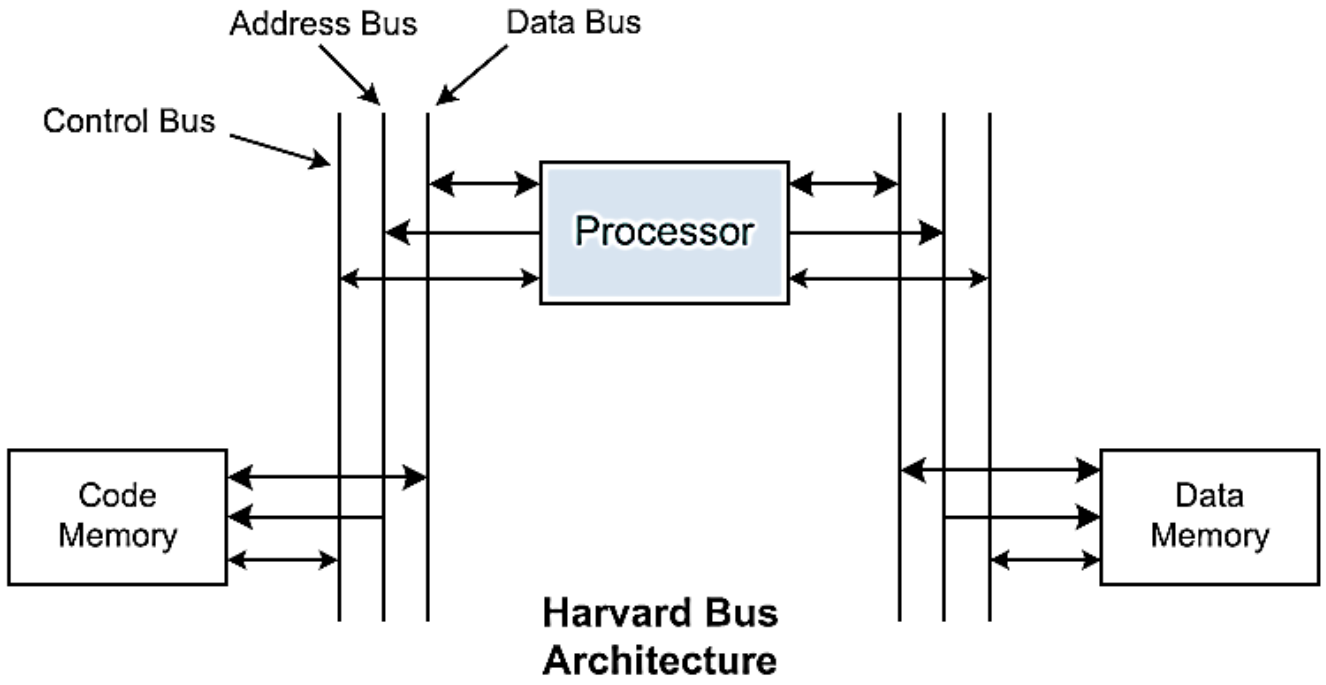
- ✓ The von-Neumann architecture uses a common bus for both data as well as code memory.
- ✓ As a result either an instruction can be fetched from memory or data can be read/written from/to memory during each memory access cycle.
- ✓ Instructions and data are stored in the same memory subsystem and share a common bus to the processor.



von Neuman Bus Architecture

HARVARD BUS ARCHITECTURE

- ✓ The Harvard architecture utilizes separate buses for accessing code and data memories.
- ✓ This allows instructions and data to be accessed simultaneously.
- ✓ In addition, the next instruction may be fetched from memory at the time when the previous instruction is about to finish its execution, allowing for a primitive form of pipelining.
- ✓ Pipelining decreases the per instruction execution time; however, main memory access time is a major bottleneck in the overall performance of the system.



Harvard Bus Architecture

Thank You

Please follow the instructions of the government and stay safe