Compare and contrast between Harvard and Von Neumann architecture.

Definition:

The Harvard architecture refers to the modern computer architecture that stores machine instructions and data in separate memory units that are connected by different busses.

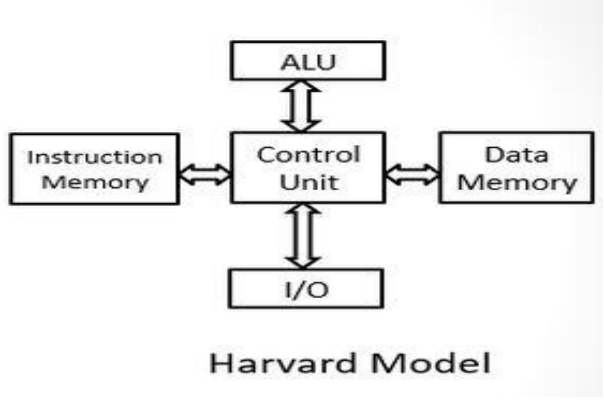
The Von Neumann Architecture is an ancient type of computer architecture where instruction data and program data are stored in the same memory. It consists of a single, shared memory for programs and data, a single bus for memory access, an arithmetic unit, and a program control unit.

Comparison

The following table highlights the major differences between Von Neumann and Harvard Architecture

Parameters	**Von Neumann Architecture**	**Harvard Architecture**
Physical Address	It uses one single physical address for accessing and storing both data and instructions.	It uses two separate physical addresses for storing and accessing both instructions and data.
Buses (Signal Paths)	One common bus helps in the transfer of both instruction and data.	It uses separate buses for the transfer of both data and instructions.
Number of Cycles	It requires two clock cycles for executing a single instruction.	An instruction is executed in a single cycle
Cost	It is cheaper in cost	It is more expensive
Access to CPU	The CPU is not able to read/write data and access instructions at the same time.	CPU can access instructions and read/write at the same time.

HARVARD ARCHITECTURE



Advantages:

- They generally offer high performance as data and buses are kept in separate memory and travel on different buses.
- Since data and instructions are stored in separate buses there are very few chances of corruption.
- Data that uses Read-Only mode and instructions which uses Read-Write mode are operated in the same way. They can also be accessed similarly.
- Parallel access to data and instructions can be maintained.
- Scheduling would no longer be required as there are separate buses for data and instructions.
- Control unit gets data and instructions from one memory. Thus, it simplifies the architecture of the control unit.

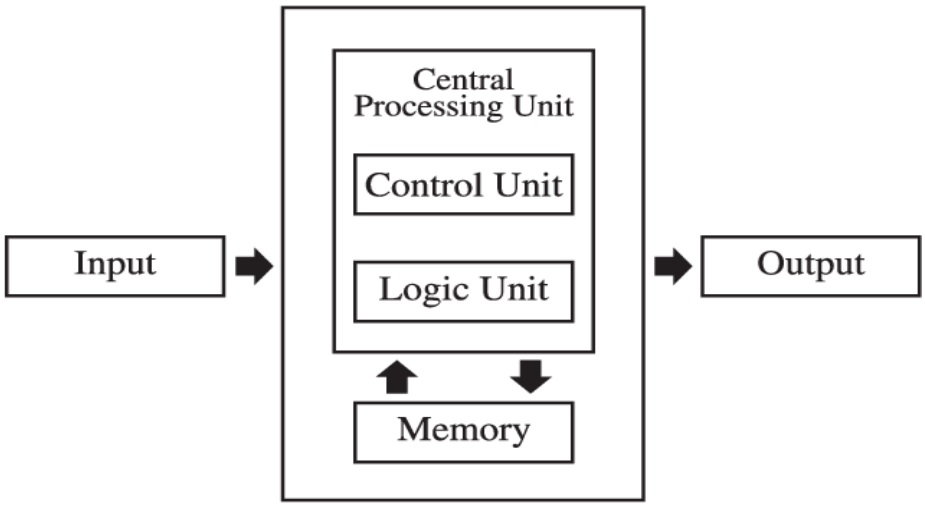
Disadvantages:

- The un-occupied data memory cannot be used by instructions and the free instruction memory cannot be used by data.
- The program cannot be written by the machine on its own as in Von Neumann Architecture.
- There are 2 buses on the architecture meaning that the motherboard would be more complex.
- It does not make most of the Central Processing Unit, always.

Examples of Harvard Architecture:

8051, ARM 9, AVR by Atmel Corporation and PIC microcontrollers by microchip Technology etc.

Von Neumann Architecture



Advantages

- Less physical space is required
- The process of controlling becomes comparatively simpler with this architecture because it fetches either instructions or data at any given time.
- Handling just one memory block is simpler and easier to achieve
- Cheaper to use than Harvard
- Requires lesser architecture because it only needs to reach one common memory.

Disadvantages

- A defective program can overwrite another in memory causing it to crash
- Memory leaks - some defective programs fail to release memory when they are finished with it, which could cause the computer to crash due to insufficient memory
- The CPU is much faster than the data bus, meaning it often sits idle (Von Neumann bottleneck)
- Data and instructions share the same data bus, even though the rate at which each needs to be fetched is often very different

Examples of Von – Neumann Architecture:

ARM 7 and Pentium Processors etc.

Compare and contrast between CISC RISC

Definition:

A Complex Instruction Set Computer (CISC) is a computer architecture in which individual instructions may perform many operations and take many cycles to execute several low-level operations (such as a load from memory, an arithmetic operation, and a memory store) or are capable of multi-step operations or addressing modes within single instructions

A Reduced Instruction Set Computer (RISC) is a processor architecture that shifts the analytical process of a computational task from the execution or runtime to the preparation or compile time. By using less hardware or logic, the system can operate at higher speeds

Comparison:

The following table highlights the major differences between CISC and RISC

CISC	RISC
Emphasis on hardware	Emphasis on software
Multi-clock complex instructions	Single-clock, reduced instruction only
Transistors used for storing complex instructions	Spends more transistors on memory register
Small code sizes	large code sizes
high cycles per second	Low cycles per second,

Complex instruction set computer (CISC)

Advantages

- Microprogramming is easy to implement and much less expensive than hard wiring a control unit.
- It is easy to add new commands into the chip without changing the structure of the instruction set as the architecture uses general-purpose hardware to carry out commands.
- micro program instruction sets can be written to match the constructs of high-level languages.

Disadvantages

- The number of general-purpose registers that can be fitted into the processor is less because decoding instructions require more transistors.

- The code requires several clock cycles to execute a single instruction despite having a minimal code size. This can decrease system efficiency.
- It consumes more power and dissipate more heat than RISC-based ones.
- It is more expensive to implement than RISC because of its complexity.

Examples of CISC processor

1. IBM 370/168
2. Intel 80486
3. VAX 11/780

Reduced Instruction Set Computer (RISC)

Advantages

- The performance of RISC processors is often two to four times than that of CISC processors because of simplified instruction set.
- This architecture uses less chip space due to reduced instruction set.
- The per-chip cost is reduced by this architecture that uses smaller chips consisting of more components on a single silicon wafer.
- RISC processors can be designed more quickly than CISC processors due to its simple architecture.
- High execution of instructions in RISC processors due to the use of many registers for holding and passing the instructions.

Disadvantage

- The performance of the processor depends on the programmer or software developer.
- Developing programs for this architecture requires more effort when compared to developing for CISC.
- Processors based on this architecture require large memory caches.

Examples of RISC processors

Alpha, AVR, ARM, PIC, PA-RISC, and power architecture.