Name :Eslam Ahmed Ali

Day: 03 _08_08_2023

Group : ITI . Embedded Adv. Al-Azhar G4 23

Task : 2

a-Discuss the difference between harvard and von neumann (table)

	Harvard	Von neuman
Memory Organization	The Harvard architecture uses physically separate memory spaces for instructions and data. It has separate instruction memory (known as the Instruction Memory Unit or IMU) and data memory (known as the Data Memory Unit or DMU). This allows simultaneous access to instructions and data.	The Von Neumann architecture uses a unified memory space for both instructions and data. Instructions and data are stored together in a single memory unit, and they are accessed sequentially.
Data and Instruction Access	In the Harvard architecture, instructions and data can be fetched simultaneously from their respective memory units. This enables parallel fetching and execution of instructions, which can improve performance.	In the Von Neumann architecture, instructions and data share the same memory space. They are fetched sequentially, which means that fetching an instruction and accessing data cannot happen simultaneously.
Instruction Fetch and Execution	The Harvard architecture allows separate instruction fetch and data access paths, which can result in faster instruction execution. Instruction fetching can happen independently of data access, enabling efficient pipelining and reducing instruction fetch stalls.	In the Von Neumann architecture, the instruction fetch and data access share the same bus and memory, leading to potential bottlenecks. Instruction fetching and execution occur sequentially, which can limit the potential for parallelism.
Program and Data Memory Size	Since data and instruction memories are separate, the Harvard architecture can support different memory sizes for instructions and data. This can be advantageous in applications where the sizes of instruction and data memories differ significantly.	In the Von Neumann architecture, instruction and data memory share the same memory unit, so the total memory size is fixed and shared between instructions and data.
Program and Data Memory Protection	The separation of instruction and data memories in the Harvard architecture can provide inherent memory protection. It allows fine-grained control over the access permissions of instruction and data memory, enhancing system security.	The Von Neumann architecture does not have built-in memory protection between instructions and data. Protecting memory access requires additional mechanisms, such as memory management units (MMUs) or virtual memory.

b. Discuss the difference between RISC and CISC (table)

	RISC	CISC
Instruction Set	Simple and fixed instruction set	Large and variable instruction set
Instruction Format	Uniform and regular	Variable and irregular
Operand Addressing	Limited addressing modes	Rich and varied addressing modes
Instruction Execution	Simple instructions, executed in one clock cycle	Complex instructions, executed in multiple clock cycles
Pipelining	Well-suited for pipelining	More challenging for pipelining
Memory Access	Load/store architecture	Memory-to-memory operations
Complexity	Emphasizes simplicity and efficiency	Emphasizes instruction complexity and functionality
Code Size	Longer code due to simpler instructions	Shorter code due to complex instructions
Power Efficiency	Typically more power-efficient	Typically less power-efficient
Performance	High performance in pipelined implementations	Optimized for instruction-level parallelism and complex operations

c-Discuss the difference between HSE & HSI & PLL (table)

	HSE	HSI	PLL
Source	External crystal oscillator or clock input	Internal RC oscillator	Internal RC oscillator or external clock input
Accuracy	Higher accuracy	Moderate accuracy	Dependent on the accuracy of the input clock
Frequency Range	Typically higher (e.g., 4MHz - 26MHz)	Typically lower (e.g., 8MHz)	Dependent on external clock input or input frequency
System Clock	Can be used as the system clock source	Can be used as the system clock source	Can generate the system clock through multiplication
Power Consumption	Generally higher	Generally lower	Generally higher due to increased system clock frequency
Usage	Recommended for high-frequency operations	Suitable for general-purpose applications	Used to generate higher system clock frequencies