Computer Architecture (**CSL3020**)

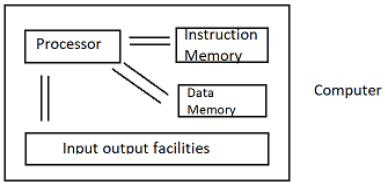
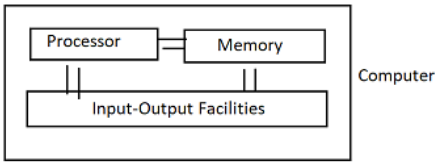
Assignment-1

Solution-1

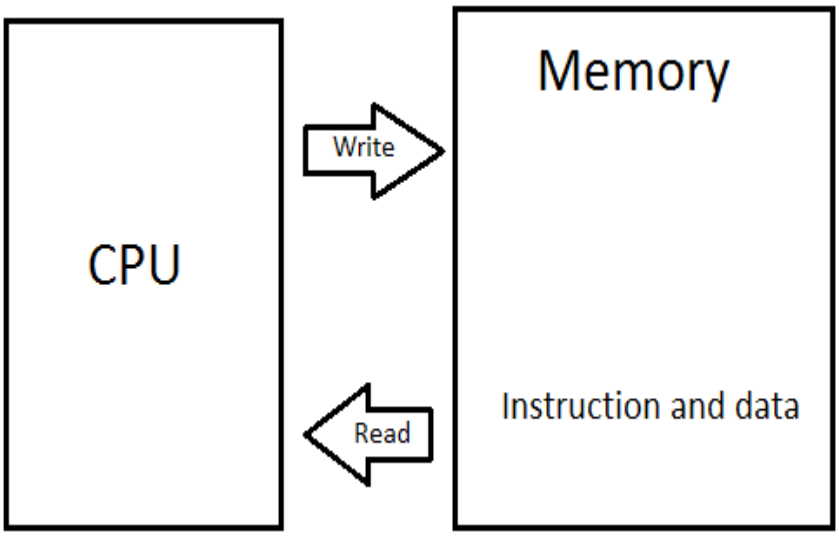
Serial and Parallel bus relative advantages and disadvantages:-

Parameters / Categories	Serial Bus	Parallel Bus
Definition	Bus that delivers data in the form of bits across a single channel pathway.	Bus that delivers data in the form of bits across several channels at the same time.
Cost	Only a single channel pathway would be required to transmit data cost effectively.	Multi-Channels would be required to transmit data, thus requiring more pathway constructs, leading to more expenses.
Synchronization	No procedure for Sync would be required in a single channel pathway.	Sync between multiple channel pathways would be required, thus leading to bits intermixing and information mis-match.
Length of the Bus	A set of whole data bits would be required to be sent through a single channel pathway.	Due to availability of multiple channel pathways, single bits would be sent via each channel, leading to decreased cumulative distance.
Transmission Rate	As the whole set of data bits would be passing, the transmission rate would be slower.	As multiple channel pathways would be providing data bits simultaneously at a given stance of time, the transmission rate would be higher.

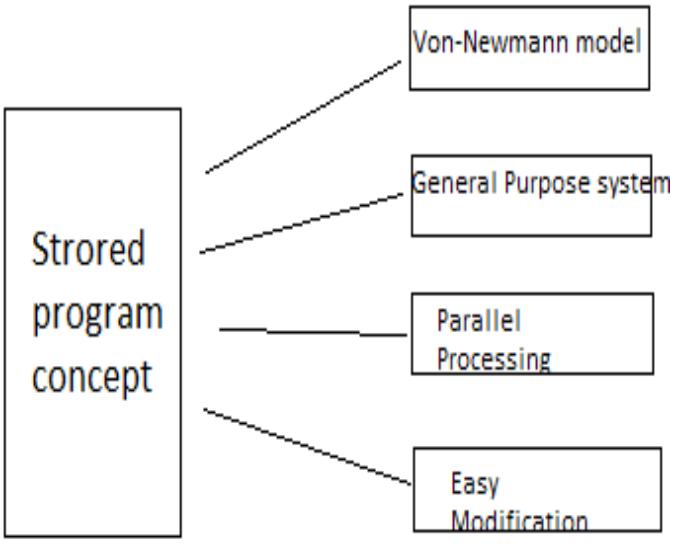
Major differences between Von-Newmann and Harvard architecture:-

Parameters / Categories	Von-Neumann Architecture	Harvard Architecture
Diagram		
Application Area	In Embedded Systems like Microcontrollers.	In Personal Computers and Hand-held Devices like Mobiles.
Extent of Execution via Cycles	Require a Single Cycle to completely execute the instruction.	Require 2 cycles to completely execute the instruction.
Memory usage and Financial Expenses	Separate memory would be required for program as well as data, leading to more cost expense.	Program and Data work on the same memory, leading to cost-effective implementation.
Write and Read Operations	CPU is multi-tasking at the same time, i.e Accessing Instructions, reading them and then writing.	The CPU is single-tasking and cannot handle multiple operations at the same time.

Solution-2



Approach-1



Approach-2

Approach-1

The computer's memory, according to this concept, is employed to store both the instructions and the data that goes with them. The von Neumann architecture is an architectural design concept for a stored-program digital computer that uses the same memory and a single, independent storage structure to hold both instructions and data. It was created by Hungarian mathematician John Von Neumann. The advantage of using saved programmes is that they may be changed while they are running.

Approach-2

- People no longer need to execute instructions without the help of a computer thanks to the stored programme concept.
- It is possible to store instructions in memory and execute them in a sequential manner, referencing the data values needed to function.
- With the stored programme concept, it is possible to store multiple programmes in memory at the same time, allowing for the execution of multiple processes at the same time.
- As a result, there will be no need to save instructions and data separately. It is simple to change the apps as an additional feature.
- Also, it may be used for general-purpose computing, with the added benefit of simultaneous task processing and the ability to make changes while the programme is running.

Solution-3

An instruction set architecture determines how programmes should be encoded for a range of different devices that share the same instruction set architecture. If adequate memory and I/O resources are available, a programme can be executed after it has been coded in an ISA. As a result, ISAs are essential in that they provide a hardware-software interface while encouraging both technologies to develop independently (like other technical abstractions).

If the ISA has capabilities that target high-level languages (or make the compiler's job easier), it can make optimising the implementation much easier (for example, long pipelines, or wide problems)

Categories	System Software	Application Software

Programming complexity	Due to the need for hardware compatibility and the arising complexity, little to no user input is given.	As it is user friendly and intended to operate at higher-language level, it would be easier than system software programming.
Essential Functioning	They would be essential for each and every possible functioning of the system.	Not so important for effective functioning of the system.
Dependency on Run Command	Independently run on its own.	Although self-contained, it requires the existence of System Software to function.
Classification	Divided into 2 categories,i.e Packed and Customised Software.	Classified into Time Sharing and Resource Sharing Softwares.
Language preference	Low-level languages like assembly and machine codes would be used, enabling greater inter-operable characteristics with hardware and software.	High-Level Languages like Java, C++, etc. would be used, leading to user-friendly environment work.
Interface structure	It acts as an interface between application software and system.	It operates on the basis of user-defined instructions.
Examples	Compiler, Debug Interface	Media Player, VS-Code

Solution-4

No. It won't be necessary to achieve faster execution by the CPU, while increasing only and only clock rate of the computer, because :-

- For the scenario of parallel bus driven systems, increasing the clock rate without any synchronization and delays would obliterate the potential usage of instruction transmittance.

- Ideally, the CPU Execution Time is inversely proportional to Clock Rate, but in Real Life, the Rate of Signal-based Attenuations and Noise Capturing would be paced up due to surrounding environmental disturbances. Although the quality of Channel Pathways are

significantly improved, these things still ensure that CPU Execution Time won't be drastically improved in extreme configurational changes.

- If the Clock Rate would be increased for certain instructions that depend on each other's synchronization and timings without suitable considerations and limitations, there might be chances that some of the instructions got neglected / intermixed, leading to incomplete execution of different component operations, thus ruining the overall purpose of Fast-Paced CPU Working.

Solution-5

Moore's Law says that the number of transistors on a microchip doubles every two years. From a formal viewpoint, the speed of a computer rises as the number of transistors on a microchip grows. Moore's law still holds true today, albeit its importance is diminishing as new ways for assessing processing capacity develop.
Moore's law is expected to become obsolete in the early 2020s as the number of transistors doubles every two years, according to experts at Intel.

As a result, transistors will no longer be designed to manage in smaller circuits at higher temperatures as cooling them consumes more energy than the energy flowing through the transistor. Consequently, two-dimensional chip layouts have a limited lifespan attributed to the industry's investment in 3D technology.

Quantum computing, DNA computing, and neuromorphic computing are among some of the alternatives that could be used in the upcoming future.

Solution-6

The Program Code would be attached in the respective zip file, with certain key notes :-

1. It is generalizable for each type of Roll Number digits of Campus Students to be passed through.

2. Using bias, approximation for length of Exponent Bits was made, via logarithmic method with 2 as a base and ceiling function, in order to avoid the issue of overflow.

3. One Major Shortcoming is that it approximates the Floating value into an integer via internal functions, making the operation quite less accurate.

Solution-7

Instruction Type	Frequency	Clock Cycles
ALU Instructions	50%	4
Load Instructions	30%	5
Store Instructions	10%	4
Branch Instructions	10%	2

According to the above-mentioned Table for Instructions with their corresponding Frequencies and Clock Cycles :-

(a) Cycles per Instruction (CPI) = No. of Clock Cycles * Frequency of a certain Instruction.

But for overall set of Instructions, the resultant CPI can be calculated as following :-

$CPI = \sum_{1}^{i} Num(i) * Freq(i)$

So for corresponding Instruction types :-

ALU Instruction → 50% * 4 = 0.5*4 = 2
Load Instruction → 30% * 5 = 0.3*5 = 1.5
Store Instruction → 10% * 4 = 0.1*4 = 0.4
Branch Instruction → 10% * 2 = 0.1*2 = 0.2

After Calculating all these :-

CPI = 2+1.5+0.4+0.2 = 4.1

(b) CPU execution time = Total Number of Instructions * CPI * (1/Frequency)
= (3 * (10**6)*4.1) / (2*(10**6)) milli-seconds
= 6.15 milli-seconds

(c) As the Formula for Amdahl's Law would be :-

S0 = 1/((1-p) + (p/s))

S0 → Overall Speedup = 1.5
P → Proportion of Execution Time = **CPI of an instruction / CPI of All Instructions = (2/4.1)**
S → Speedup of the Enhanced Instruction Type = ?

After simplifying the formula for the value of s :-

$S = 1/(((1/p)*((1-S0)/S0)) +1)$

Putting up all the values, $S = 60/19$ which is nearly equal to 3.15.