

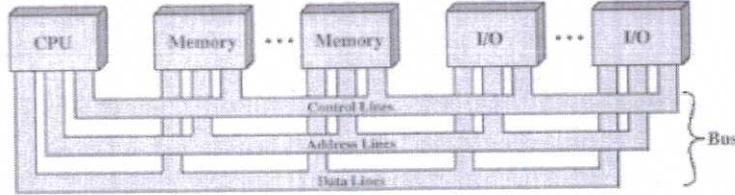
4501

Scheme of evaluation

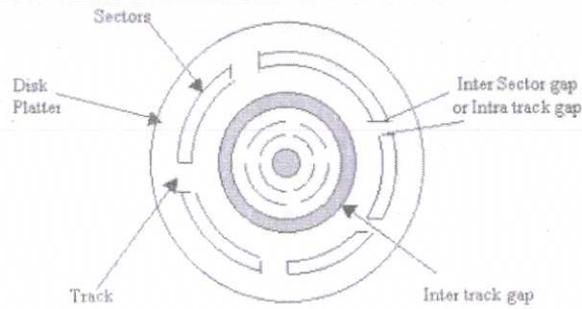
Course :
Version :
Code :

3B1(15)
Nov 2021

Scoring indicators

Qn No.	Scoring indicator	Split Score	Sub total	Total
I 1.	Pc holds address of the next instruction to be fetched	2		10
2	Time taken to position the head at track	2		
3	Redundant Array of independent disk....,	2		
4	Program status word	2		
5	One classification of Flynn's classification of parallel processing-single instruction single data	2		
II 1	<ul style="list-style-type: none"> A bus is a communication pathway connecting two or more devices It is a shared transmission medium A bus consists of multiple pathways or lines A bus that connects major components (CPU,Memory,I/O) is called System Bus. <p>BUS STRUCTURE</p> <ul style="list-style-type: none"> Each line is assigned a particular meaning or function. The lines can be classified into 3 functional groups  <ol style="list-style-type: none"> Data line Address line Control line <p>1. DATA LINE</p> <ul style="list-style-type: none"> Provide a path for moving data between system modules. These lines, collectively, are called the data bus Data bus may consist of from 32 to 100 separated line Each line can carry only one bit at a time Number of line in the bus determine the data rate and overall the system performance <p>2. ADDRESS LINE</p> <ul style="list-style-type: none"> Used to designate the source or destination of the data on the data bus The width of the address bus determines the maximum possible memory 		3 P3	

	<p>capacity of the system.</p> <p>3. Control line</p> <ul style="list-style-type: none"> • Used to control the access to and the use of the data and address lines • Since the data and the address line shared by all the components, there must be a means of controlling their use • Control signal transmit both commands and timing information between the modules <ul style="list-style-type: none"> • Typical control lines include <ol style="list-style-type: none"> 1. Memory write 2. Memory read 3. I/O write 4. I/O read 5. Clock 6. Reset 7. Bus request 8. Bus grant 9. Interrupt request 10. Interrupt ACK 11. Transfer ACK 			
II 2	<p>The three different types of mapping used for the purpose of cache memory are as follow, Associative mapping, Direct mapping and Set-Associative mapping.</p> <p>- Associative mapping: In this type of mapping the associative memory is used to store content and addresses both of the memory word. This enables the placement of the any word at any place in the cache memory. It is considered to be the fastest and the most flexible mapping form.</p> <p>- Direct mapping: In direct mapping the RAM is made use of to store data and some is stored in the cache. An address space is split into two parts index field and tag field. The cache is used to store the tag field whereas the rest is stored in the main memory. Direct mapping's performance is directly proportional to the Hit ratio.</p> <p>- Set-associative mapping: This form of mapping is a modified form of the direct mapping where the disadvantage of direct mapping is removed. Set-associative mapping allows that each word that is present in the cache can have two or more words in the main memory for the same index address.</p>	3* 2		6
II 3				



6

II 4

I/O MODULE FUNCTIONS

Control & Timing.

CPU Communication

Device Communication

Data Buffering

Error Detection

I/O steps

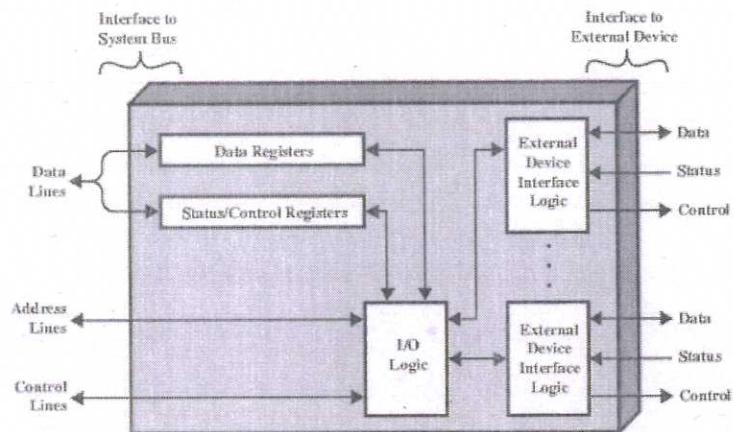
CPU checks I/O module device status

I/O module returns status

If ready, CPU requests data transfer

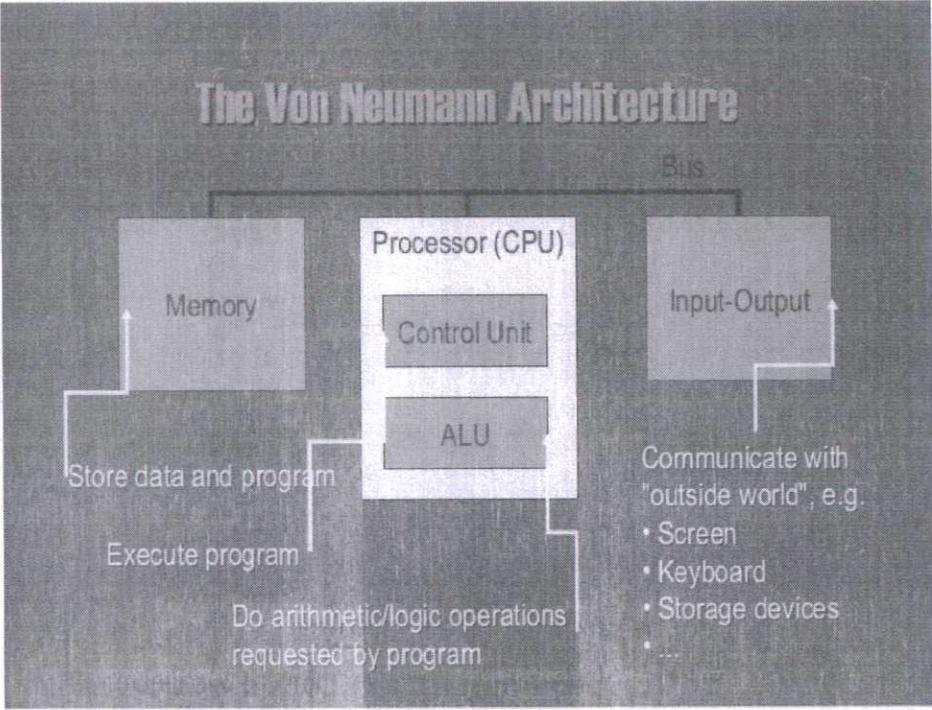
I/O module gets data from device

I/O module transfers data to CPU



6

Figure 7.4 Block Diagram of an I/O Module

II 5.	Resource hazard occur when two or more instruction in pipeline need the same resource Control hazard due to wrong decision about branch prediction..Short expln		3*2=6	
II 6.	<ul style="list-style-type: none"> • A standard process describes the steps needed for processing to take place. It is called the Fetch - Decode - Execute cycle or sometimes simply called the Fetch-Execute Cycle. • First of all, both the data and the program that acts upon that data are loaded into main memory (RAM) by the operating system. The CPU is now ready to do some work. • The first step the CPU carries out is to fetch some data and instructions (program) from main memory then store them in its own internal temporary memory areas. These memory areas are called 'registers'. • This is called the 'fetch' part of the cycle. • For this to happen, the CPU makes use of a vital hardware path called the 'address bus'. • The CPU places the address of the next item to be fetched on to the address bus. • Data from this address then moves from main memory into the CPU by travelling along another hardware path called the 'data bus'. 			30
II 7.	The decoders and sequencing logic unit of a micro-programmed control unit are very simple pieces of logic, compared to the hardwired control unit, which contains complex logic for sequencing through the many micro-operations of the instruction cycle. It simplifies the design of the control unit. Simpler design means the control unit is cheaper and less error-prone to implement - It is also flexible as changes could be easily made to the design Principal Disadvantage: - Slower than a hardwired unit of comparable technology Hardwired control unit is used for RISC Architecture because hardwired is faster and can improve the performance Microprogrammed control unit is used for CISC because it makes the design simpler and usually in CISC architecture, due to huge number of instructions in the instruction set, the control unit is quite complex. Hence it justifies using microprogrammed control unit.		3 points — 3*2=6	
III (a)				10 15
III (b)	Sequential,direct,random,----Any two with short exp		2.5* 2	
IV (a)	Any 4 charecterstics Location,Access methos,Performance. Physical types, Physical charecterstics,		4 *	8

		2	
IV (b)	,line size,number of cahes-, size,mapping function,replacement algorithm,write policy-- listname with short exp Cache address,cache		7*1 15
V (a)	Raid -redundent array of independent disk-contain seven layers. Raid0—No redundancy,Data striped across all disk, Raid1....Rdundancy isachieved by duplicating all the data RAID2. RAId3. RAID4 RAID5 RAID6 (what is raid(3)+each one-6)		3+6
V (b)	Programmed I/O (PIO) refers to data transfers initiated by a CPU under driver software control to access registers or memory on a device. The CPU issues a command then waits for I/O operations to be complete. As the CPU is faster than the I/O module, the problem with programmed I/O is that the CPU has to wait a long time for the I/O module of concern to be ready for either reception or transmission of data. The CPU, while waiting, must repeatedly check the status of the I/O module, and this process is known as Polling. As a result, the level of the performance of the entire system is severely degraded. Programmed I/O basically works in these ways: <ul style="list-style-type: none"> • CPU requests I/O operation • I/O module performs operation • I/O module sets status bits • CPU checks status bits periodically • I/O module does not inform CPU directly • I/O module does not interrupt CPU • CPU may wait or come back later 		6 15
VI (a)	this technique the processor do not wait for until the I/O operation is complete. Rather processor normally do the other task. When I/O is complete the I/O module interrupt into processor . Interrupt means that the operation is completed. * Primary disadvantage of programmed I/O is that CPU spends most of its time in a tight loop waiting for the device to become ready. This is called busy waiting. * With interrupt-driven I/O, the CPU starts the device and tells it to generate an interrupt when it is finished. * Done by setting interrupt-enable bit in status register. * Still requires an interrupt for every character read or written. * Interrupting a running process is an expensive business (requires saving context). * Requires extra hardware (DMA controller chip).		4pon ts*2 =8 15
VI (b)	<ul style="list-style-type: none"> • The data transfer between a fast storage media such as magnetic disk and memory unit is limited by the speed of the CPU. Thus we can allow the peripherals directly communicate with each other using the memory buses, removing the intervention of the CPU. This type of data transfer technique is known as DMA or direct memory access. During DMA the CPU is idle and it has no control over the memory buses. The DMA controller takes over the buses to manage the transfer directly between the I/O devices and the memory unit. 		2+5

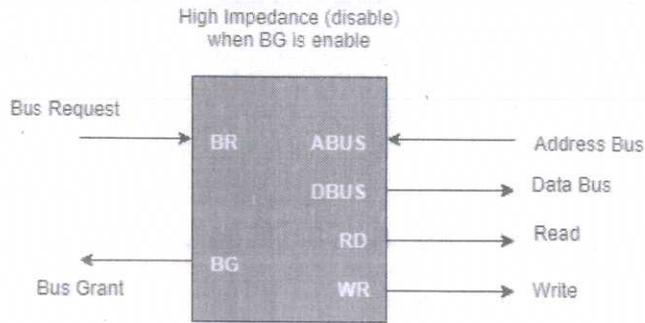


Figure - CPU Bus Signals for DMA Transfer

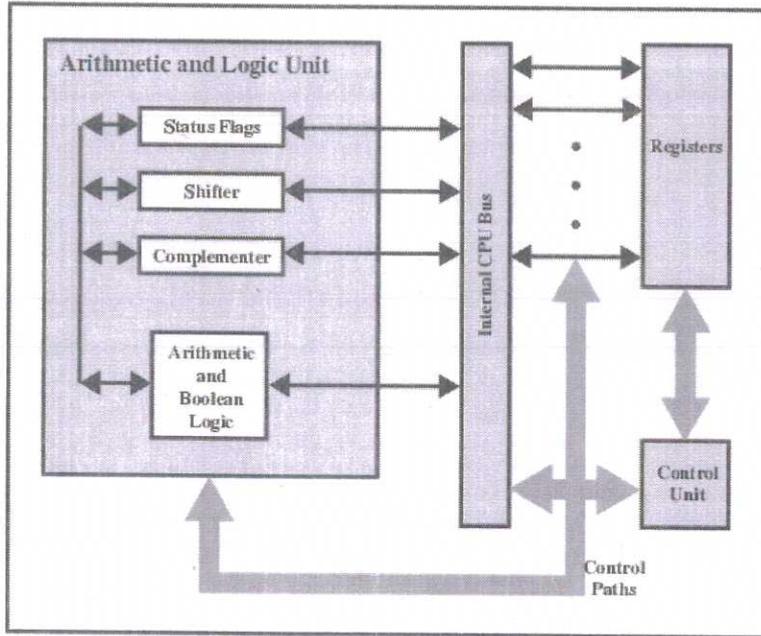
Bus Request : It is used by the DMA controller to request the CPU to relinquish the control of the buses.

Bus Grant : It is activated by the CPU to Inform the external DMA controller that the buses are in high impedance state and the requesting DMA can take control of the buses. Once the DMA has taken the control of the buses it transfers the data. This transfer can take place in many ways.

<p>VII (a)</p>	<p>Two types of registers....1.user visible registers2.control and status registers</p> <ul style="list-style-type: none"> • User visible registers--results of machine instructions, e.g. comparisons. It usually consists of several independent flags such as <u>carry</u>, <u>overflow</u> and zero. The CSR is chiefly used to determine the outcome of <u>conditional branch</u> instructions or other forms of conditional execution. A user visible register (UVR) are the registers visible to programmers. Basically this means that the programmer can only make use of these registers (UVA). • These registers includes general purpose or special purpose registers. <p>Example: Data Register, & Address Register. Control and status registers : A <u>register</u> in most CPUs which stores additional information about the results of machine instructions, e.g. comparisons. It usually consists of several independent flags such as <u>carry</u>, <u>overflow</u> and zero. The CSR is chiefly used to determine the outcome of <u>conditional branch</u> instructions or other forms of conditional execution</p>	<p>4*2= 8</p>	
<p>VII (b)</p>	<p>Parallel processing provides simultaneous data processing tasks for the purpose of increasing the computational speed of a computer system rather than each instruction is processed sequentially, a parallel processing system is able to perform concurrent data processing to achieve faster execution time and increase throughput.</p> <p>There are more advantages with parallel processing but it has some issues also. Due to parallel processing, the number of hardware increases and the cost of the system increases. Parallel processing is established by distributing the data among the multiple functional units.</p> <p>Flynn's Classification</p> <p>Flynn introduced the parallel processing classification. This classification considers the organization of a computer system by the number of instructions and data items that are manipulated simultaneously.</p>	<p>3+4 =7</p>	<p>15</p>

- The sequence of instructions read from the memory constitutes an instruction stream.
- The operations performed on the data in the processor constitutes a data stream.

VIII
(a)



Fig(4)+e
xp-A
=8

8

VIII
(b)

Timing Diagram for .Instruction Pipeline Operation

	Time →													
	1	2	3	4	5	6	7	8	9	10	11	12	13	14
Instruction 1	FI	DI	CO	FO	EI	WO								
Instruction 2		FI	DI	CO	FO	EI	WO							
Instruction 3			FI	DI	CO	FO	EI	WO						
Instruction 4				FI	DI	CO	FO	EI	WO					
Instruction 5					FI	DI	CO	FO	EI	WO				
Instruction 6						FI	DI	CO	FO	EI	WO			
Instruction 7							FI	DI	CO	FO	EI	WO		
Instruction 8								FI	DI	CO	FO	EI	WO	
Instruction 9									FI	DI	CO	FO	EI	WO

Comments

You do not have permission to add comments.

15

7

IX
(b)

Instruction fetch:

Transfer the address of PC to MAR. (Register Transfer)	MAR ← PC
MAR puts its contents on the address bus for main and issues a memory read signal. The word so read is placed on the data bus where it is accepted by the Data register. The PC is incremented by one memory word length to point to the next instruction in sequence.	DR ← (MAR) PC ← PC + 1
The instruction is transferred from data register to the Instruction register processing.	IR ← DR

Instruction Decode: The Control Unit determines the operation that is to be performed and the addressing mode of the data.

Operand Address Calculation: (In case of direct addressing)

Transfer the address portion of instruction to the MAR. (Register Transfer) is the direct address so no further calculation needed.	<i>IR (Address) and DR (Address) contain the Effective address.</i>
---	---

Operand Address Calculation: (In case of indirect addressing)

Transfer the address bits of instruction to the MAR. This transfer can be achieved using DR, as DR and IR at this point of time contain the same value. (Register Transfer)	MAR ← DR (Address)
Perform a memory read operation as done in fetch cycle and the desired address of the operand is obtained in the DR. (Memory Read)	DR ← (MAR)
Transfer the address part so obtained in DR as the address part of instruction. (Register Transfer) Thus, the indirect address is now converted to direct address or effective address.	IR (Address) ← DR (Address)

Execution:

Transfer the address portion of the instruction to the MAR. (MemAddress transfer)	MAR ← MemAddress
Store the AC register to Memory Address in MAR.	(MAR) ← AC

Interrupt Processing:

Transfer the contents of PC to DR	DR ← PC
Place the address of location, where the return address is to be saved, into MAR.	MAR ← Address of location for saving return address.
Store the contents of PC in the memory using DR and MAR. (Memory write) Transfer the address of the first instruction of interrupt servicing routine to the PC.	(MAR) ← DR PC ← address of the first instruction interrupt servicing programs

6

<p>IX (a)</p>	<ol style="list-style-type: none"> 1. An alternative scheme for hardwired control unit design is called micro programmed control design unit in which control signals are generated by a program similar to machine language programs. 2. A microprogrammed control unit is a relatively simple logic circuit that is capable of (1) sequencing through microinstructions and (2) generating control signals to execute each microinstruction. 3. The concept of micro program is similar to computer program. In computer program the complete instructions of the program is stored in main memory and during execution it fetches the instructions from main memory one after another. The sequence of instruction fetch is controlled by program counter (PC). 4. A control word (CW) is a word whose individual bits represent the various control signals. Each of the control steps in the control sequence of an instruction defines a unique combination of 1s and 0s in the CW. 5. Consider a control step example as follows: PCout, R=B, MARin, Read, IncPc. In this step, the contents of the PC are passed through the ALU, using the R=B control signal, and loaded into the MAR to start a memory read operation. At the same time the PC is incremented. 6. A sequence of CW's corresponding to the control sequence of a machine instruction constitutes the micro routine for that instruction, and the individual control words in this micro routine are referred to as microinstructions. The following Figure 8 shows the basic organization of a microprogrammed control unit <div data-bbox="478 795 1021 1265" data-label="Diagram"> </div> <p style="text-align: center;">Figure 9</p>	<p>Fig(3)+E xp6 9</p>	<p>15</p>
<p>X (a)</p>	<p>Flynn's classification is based on multiplicity of instruction streams and data streams observed by the CPU during program execution. Let Is and Ds are minimum number of streams flowing at any point in the execution, then the computer organisation can be categorized as follows:</p> <p>1) Single Instruction and Single Data stream (SISD) In this organisation, sequential execution of instructions is performed by one CPU containing a single processing element (PE), i.e., ALU under one control unit as shown in Figure 4. Therefore, SISD machines are conventional serial computers that process only one stream of instructions and one stream of data. This type of computer organisation is depicted in the diagram:</p> <p>2) Single Instruction and Multiple Data stream (SIMD)</p> <p>In this organisation, multiple processing elements work under the control of a single control unit. It has one instruction and multiple data stream. All the processing elements of this organization receive the same instruction broadcast from the CU. Main memory can also be divided into modules for generating multiple data streams acting as a distributed memory . Therefore, all the processing elements simultaneously execute the same instruction and are said to be 'lock-stepped' together.</p>	<p>Nam e- 4+E xp6= 10</p>	

	<p>Each processor takes the data from its own memory and hence it has on distinct data streams. (Some systems also provide a shared global memory for communications.) Every processor must be allowed to complete its instruction before the next instruction is taken for execution.</p> <p>3) Multiple Instruction and Single Data stream (MISD):</p> <p>In this organization, multiple processing elements are organised under the control of multiple control units. Each control unit is handling one instruction stream and processed through its corresponding processing element. But each processing element is processing only a single data stream at a time. Therefore, for handling multiple instruction streams and single data stream, multiple control units and multiple processing elements are organised in this classification. All processing elements are interacting with the common shared memory for the organisation of single data stream</p> <p>This classification is not popular in commercial machines as the concept of single data streams executing on multiple processors is rarely applied. But for the specialized applications, MISD organisation can be very helpful. For example, Real time computers need to be fault tolerant where several processors execute the same data for producing the redundant data. This is also known as N-version programming. All these redundant data 31 are compared as results which should be same; otherwise faulty unit is replaced. Thus MISD machines can be applied to fault tolerant real time computers.</p> <p>4) Multiple Instruction and Multiple Data stream (MIMD):-</p> <p>In this organization, multiple processing elements and multiple control units are organized as in MISD. But the difference is that now in this organization multiple instruction streams operate on multiple data streams . Therefore, for handling multiple instruction streams, multiple control units and multiple processing elements are organized such that multiple processing elements are handling multiple data streams from the Main memory as shown in Figure 7. The processors work on their own data with their own instructions. Tasks executed by different processors can start or finish at different times. They are not lock-stepped, as in SIMD computers, but run asynchronously. This classification actually recognizes the parallel computer. That means in the real sense MIMD organisation is said to be a Parallel computer. All multiprocessor systems fall under this classification.</p>			15
X (b)	<p>Data Hazards occur when there is data dependency between instructions.</p> <p>For example I1: Add R1,R2,R3 ; R1=R2+R3 I2: LD R5,10(R1) ; Load R5 with memory from 10+R1</p> <p>Now we need to wait for I1 to execute to know what R1 will be before I2 can move forward. This is called Read After Write Hazard (RAW) and also called true dependency. Other type of hazards in data hazards are Write after Write (WAW) and Write after read (WAR)</p>		5	