CMR INSTITUTE OF TECHNOLOGY

USN | | | | | | | | | | |

Internal Assessment Test-I									
Sub:	Computer Organization and Architecture							Code:	BEC306C
Date:	16/12/2024	Duration:	90 mins	Max Marks:	50	Sem:	3rd	Branch:	ECE(A,B,C,D)
Answer any **FIVE FULL** Questions									

		Marks	OBE CO	OBE RBT
1.	Explain the concept of memory-mapped I/O with a neat diagram of the I/O interface with a program example.	[10]	CO3	L2

Memory-mapped I/O: When I/O devices and the memory share the same address space, the arrangement is called *memory-mapped I/O.*

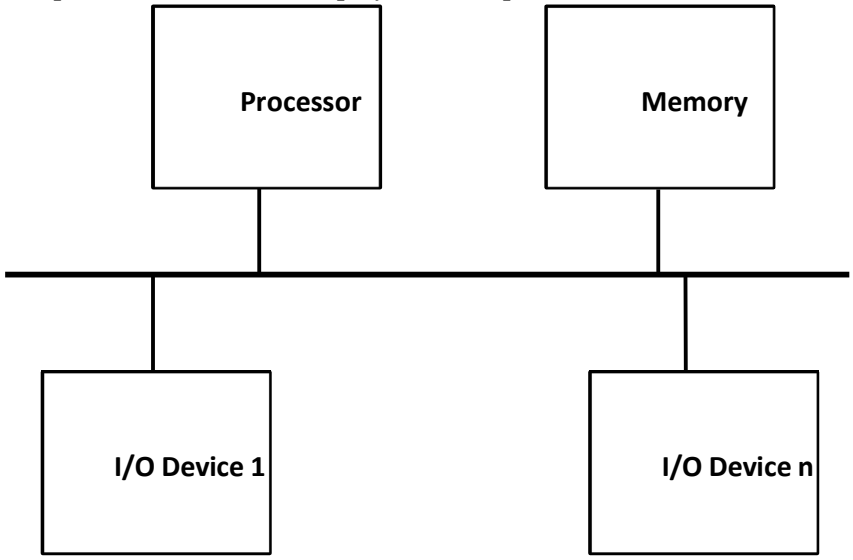
With the memory mapped I/O any machine instruction that can access memory can be used to transfer data to or from an I/O device.

Move DATAIN, R0

Reads data from the DATAIN and stores into processor register

R0. Similarly Move R0, DATAOUT

Sends the contents of register R0 to location DATAOUT which is the output data buffer of a display unit or a printer.



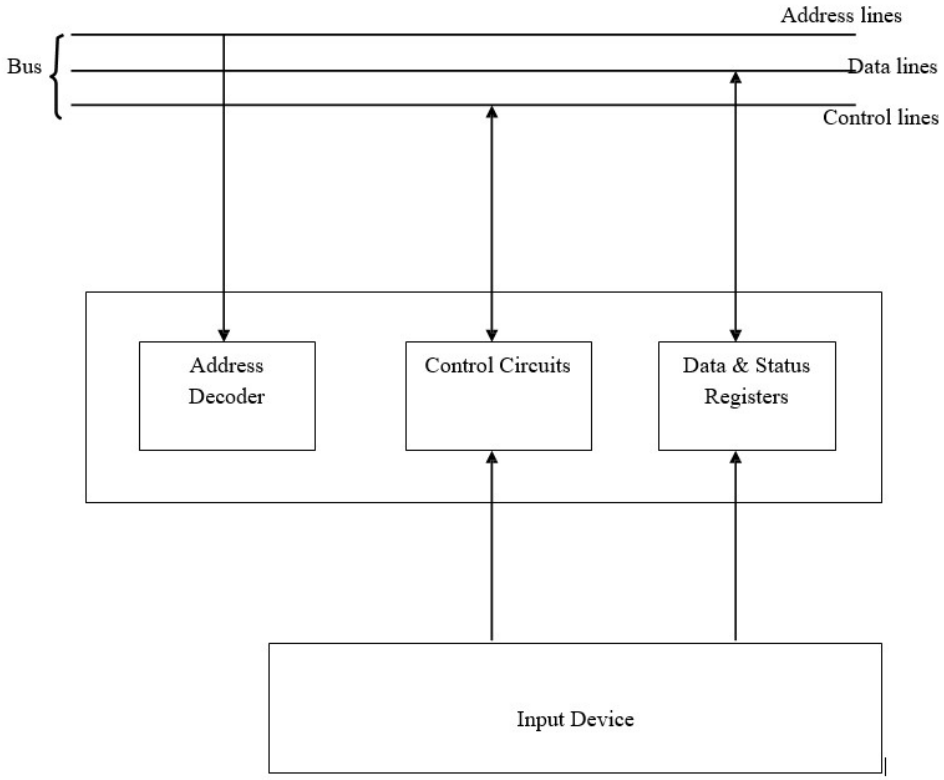


Fig 2 illustrates the hardware required to connect the I/O device to the bus. The address decoder enables the device to recognize its address when its address appears on the address lines. The data register holds the data being transferred to or from the processor. The status register contains the information relevant to the operation of I/O device. Both status and data registers are connected to the data bus and assigned unique addresses.

Interface circuit: The address decoder, data & status registers and the control circuitry required to coordinate I/O transfers constitute the device interface circuit.

Let us consider a simple example of I/O operation involving a keyboard and a display device in the computer system. The four registers shown are used for data transfer operations. Register STATUS contains two control flags SIN and SOUT which provide status information for keyboard and display unit.

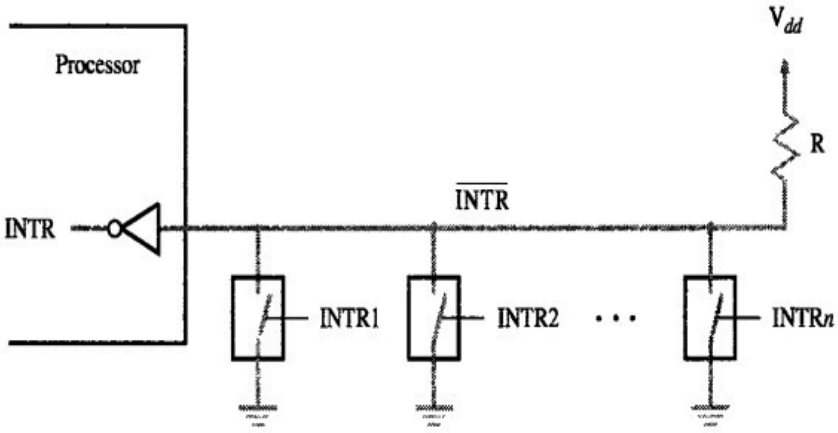
```
            Move      #LINE, R0        Initialize memory pointer
WAITK       TestBit   #0, STATUS       Test SIN
            Branch=0  WAITK            Wait for the character to be entered
            Move      DATAIN, R1       Read character
WAITD       TestBit   #1, STATUS       Test SOUT
            Branch=0  WAITD            Wait for display unit to be ready
            Move      R1, DATAOUT      Send character to be displayed
            Move      R1, (R0) +       Store character and advance pointer
            Compare   #$0D, R1         Check if carriage return
            Branch ≠ 0 WAITK           if not get another character
            Move      #$0A, DATAOUT    otherwise send line feed.
            Call      PROCESS          call the subroutine to process the input line
```

2. Write a short note on (i) interrupt hardware and (ii) interrupt nesting. [10] CO3 L2

Interrupt Hardware

I/O device requests an interrupt by activating a bus line called interrupt request. A single interrupt may be used to serve 'n' devices.

(i)



- To request an interrupt device closes its associated switch.
- If all the interrupt request signals INTR1 to INTRn are inactive, the voltage on the interrupt request line is V_{dd}.
- This is the inactive state of the line.
- When the device requests the interrupt by closing its switch, the voltage line drops to zero causing the interrupt request line INTR received by the processor to go to 1.
- The value of INTR is the logical OR of the requests from individual devices, that is

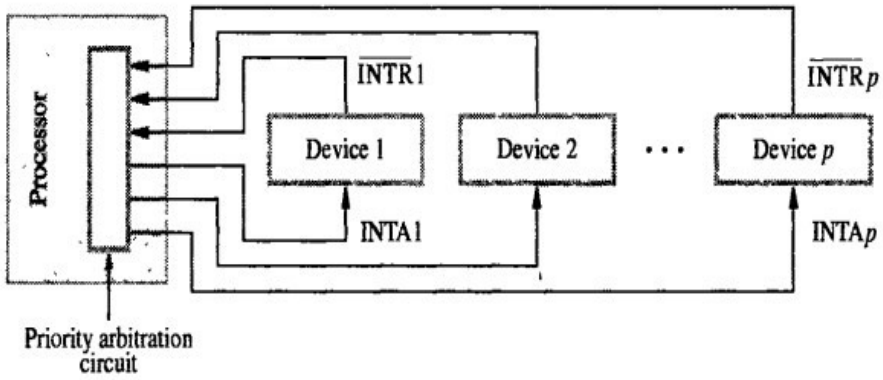
- $INTR=INTR_1 + ...+ INTR_n$

- R is the pull up register because it pulls line voltage up to high voltage when the switches are open.

(ii)

I/O devices should be organized in a priority structure. An interrupt request from a high priority should be accepted while the processor is serving another request from the lower priority device.

A multiple-level priority organization means that during the execution of an interrupt service routine, interrupt requests will be accepted from some devices but not from others, depending upon the device's priority. We can assign priority level to the processor that can be changed under program control. The priority level of the processor is the priority of the program currently being executed. The processor accepts interrupts from devices that have priorities higher than its own.



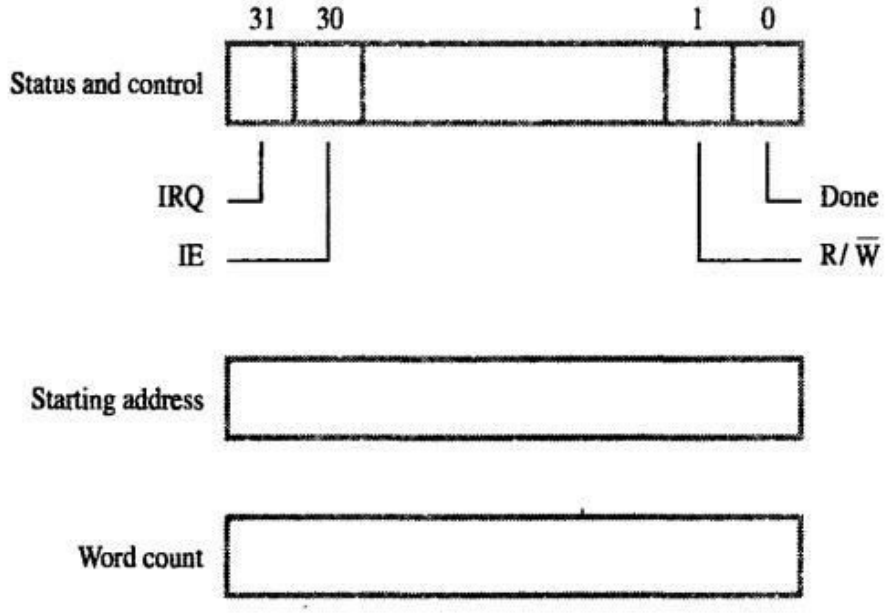
Here, each of the interrupt-request lines is assigned a different priority level. Interrupt requests received over these lines are sent to a priority arbitration circuit in the processor. A request is accepted only if it has a higher priority level than that currently assigned to the processor.

3. a) Explain the working of Vectored Interrupts. [04] CO3 L2

A device requesting an interrupt can identify itself by sending special code to the processor over the bus. The code supplied by the device represents the starting address of the interrupt service routine. The code length is 4 to 8 bits. The processor reads this address called the interrupt vector and stores it into the PC. This arrangement implies that the interrupt-service routine for a given device must always start at the same location. The interrupt vector may also include a new value for a processor status register.

The interrupted device must wait to put data on the bus only when the processor is ready to receive it. When the processor is ready to receive the vector interrupt code, it activates the interrupt acknowledge line INTA. The I/O device responds by sending its interrupt vector code and turning off the INTR signal.

3. b) Explain the operation of DMA with a neat diagram. [06] CO3 L2



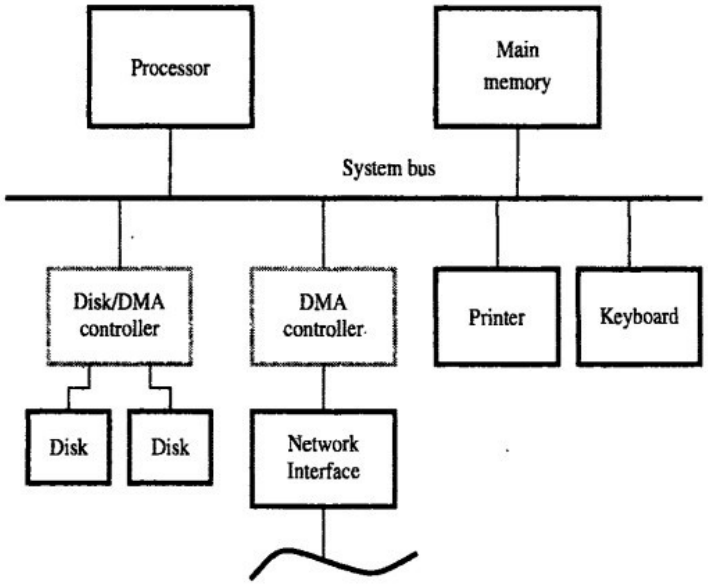
To transfer large blocks of data at high speeds, an alternate approach is used. A special control unit may be provided to allow transfer of block of data directly between external device and main memory without intervention by processor. This approach is called direct memory access or DMA.

DMA transfers are performed by control circuits that are part of I/O interface called DMA controller. The DMA controller performs functions that would normally be carried out by processor when accessing main memory.

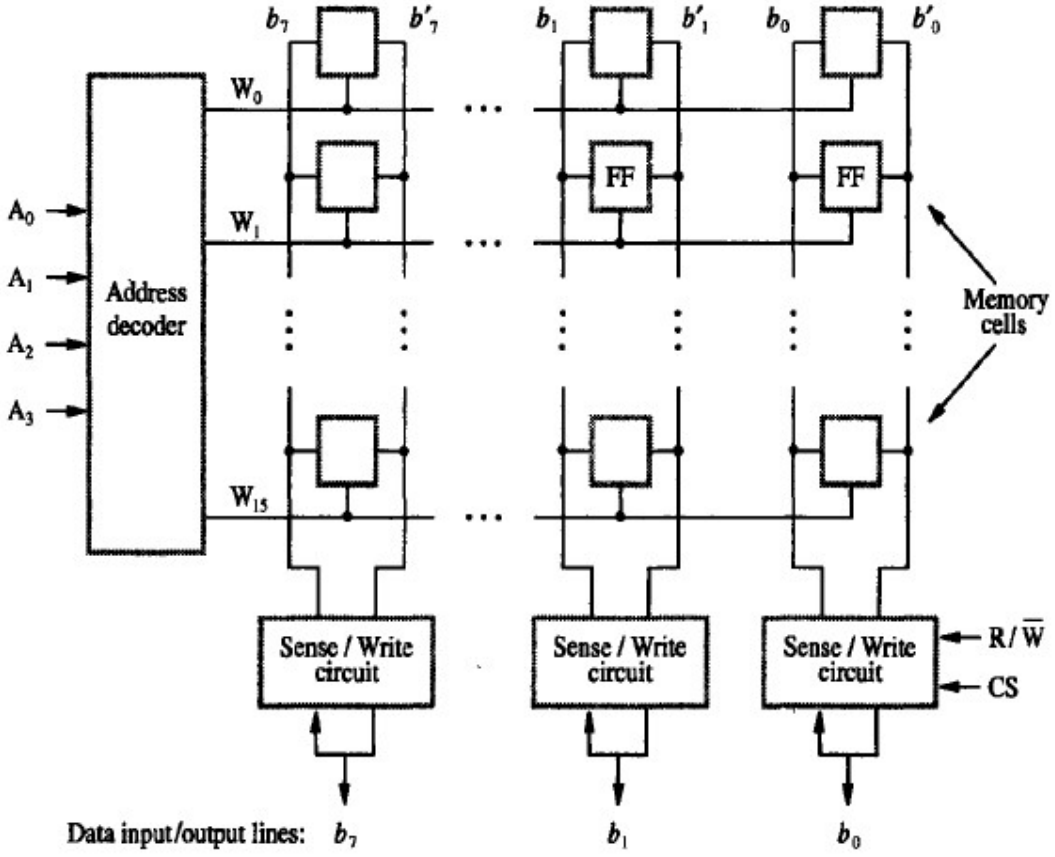
The R/$\overline{W}$ bit determines the direction of transfer. When this bit is set to 1 by a program instruction, the controller performs a read operation that is it transfers data from memory to I/O device. When transfer is complete, it sets **Done flag** to 1. When IE is1, it causes the controller to raise an interrupt after it has completed transferring a block of data. Finally, IRQ bit is set to 1 when it has requested an interrupt.

Requests from DMA devices are given high priority than processor requests. Among different DMA devices, high priority is given to high speed peripherals such as disks, high-speed network interfaces or graphic display devices.

The processor originates most memory cycles, the DMA controller is said to steal memory cycles from processor. This technique is called cycle stealing. DMA controller is given access to main memory to transfer a block of data without interruption. This is called as block or burst mode.



4. Explain the internal organization of a 16 X 8 memory chip. [10] CO4 L2



- Memory cells are organized in the form of array (Fig. 2).
- Each cell is capable of storing 1-bit of information.
- Each row of cells forms a memory-word.
- All cells of a row are connected to a common line called as **Word-Line**.

- The cells in each column are connected to **Sense/Write** circuit by 2-bit-lines.
- The Sense/Write circuits are connected to data-input or output lines of the chip.
- During a write-operation, the sense/write circuit receive input information & store input info in the cells of the selected word.
- The data-input and data output of each Sense/Write circuit are connected to a single bidirectional data-line. Data-line can be connected to a data-bus of the computer.
- Following 2 control lines are also used:
  - R/$\overline{W}$' : specifies the required operation.
  - CS': Chip Select input selects a given chip in the multi-chip memory system.
- Data bus 8 bit, address bus 4 bits, R/$\overline{W}$ 1 bit, CS 1 bit = total 14 bits for address, data and control
- It also needs two lines for power supply and ground connections.

5. Explain the concept of cache memory in detail [10] CO4 L2

Cache memory is an architectural arrangement which makes the main memory appear faster to the processor than it really is.
Cache memory is based on the property of computer programs known as "locality of reference".

Analysis of programs indicates that many instructions in localized areas of a program are executed repeatedly during some period of time, while the others are accessed relatively less frequently.

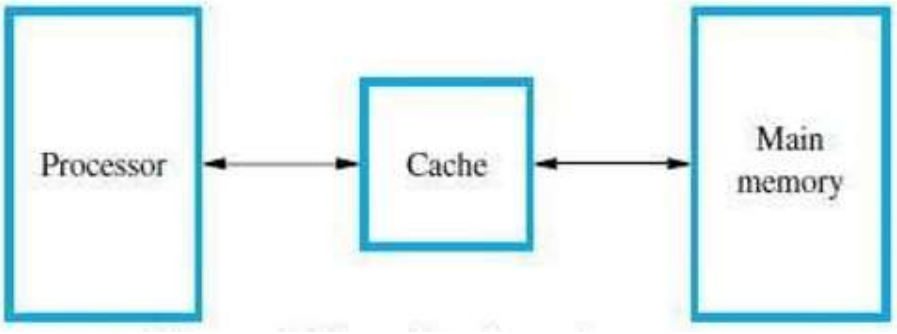
These instructions may be the ones in a loop, nested loop or few procedures calling each other repeatedly. This is called "locality of reference".

Temporal locality of reference:

Recently executed instruction is likely to be executed again very soon.

Spatial locality of reference:

Instructions with addresses close to a recently executed instruction are likely to be executed soon. **Many instructions in the localized areas of the program are executed repeatedly during some time period.** Remainder of the program is accessed relatively infrequently (Figure 8.15).



Block refers to the set of contiguous address locations of some size.

Correspondence b/w main-memory-block & cache-memory-block is specified by mapping-function. Cache control hardware decides which block should be removed to create space for the new block. The collection of rule for making

this decision is called the **Replacement Algorithm.** The cache control-circuit determines whether the requested-word currently exists in the cache.
Processor issues a Read request, a block of words is transferred from the main memory to the cache, one word at a time.
Subsequent references to the data in this block of words are found in the cache.

- Existence of a cache is transparent to the processor. The processor issues Read and Write requests in the same manner.
- If the data is in the cache it is called a **Read or Write hit**.
- **Read hit:**
  - The data is obtained from the cache.
  - Main memory is not involved
- ***Write hit:***
  - Cache has a replica of the contents of the main memory.
  - Contents of the cache and the main memory may be updated simultaneously. This is the **write-through protocol.**
  - Update the contents of the cache, and mark it as updated by setting a bit known as the dirty bit or modified bit. The contents of the main memory are updated when this block is replaced. This is **write-back or copy-back protocol**

If the data is not present in the cache, then a Read miss or Write miss occurs.

- **Read miss:**
  - Block of words containing this requested word is transferred from the memory.
  - After the block is transferred, the desired word is forwarded to the processor.
  - The desired word may also be forwarded to the processor as soon as it is transferred without waiting for the entire block to be transferred. This is called **load-through or early-restart**.
- **Write-miss:**
  - Write-through protocol is used, then the contents of the main memory are updated directly.
  - If write-back protocol is used, the block containing the addressed word is first brought into the cache. The desired word is overwritten with new information.

6. a) Explain a static RAM cell with a neat diagram. [05] CO4 L2

Memories consist of circuits capable of retaining their state as long as power is applied are known **as static memories.**

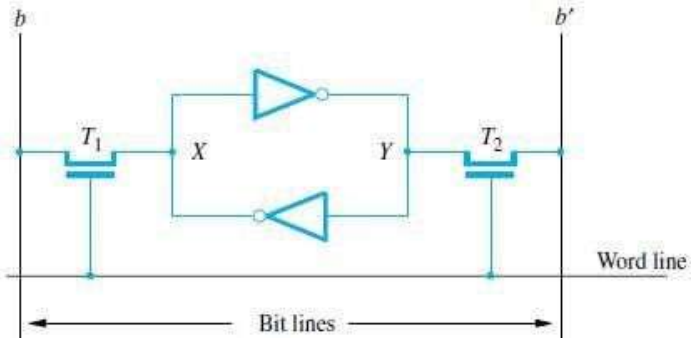


Fig. 4: A Static RAM cell

- Two inverters are cross connected to form a latch (Figure 4).
- The latch is connected to 2-bit-lines by transistors T1 and T2.
- The transistors act as switches that can be opened/closed under the control of the word-line.
- When the word-line is at ground level, the transistors are turned off and the latch retain its state.

Read Operation

- To read the state of the cell, the word-line is activated to close switches T1 and T2.
- If the cell is in state 1, the signal on bit-line b is high and the signal on the bit-line b' is low.
- Thus, b and b' are complement of each other.
- Sense/Write circuit
  → monitors the state of b & b' and
  → sets the output accordingly.

Write Operation

The state of the cell is set by
→ placing the appropriate value on bit-line b and its complement on b' and
→ then activating the word-line. This forces the cell into the corresponding state.

The required signal on the bit-lines is generated by Sense/Write circuit

- Continuous power is needed for the cell to retain its state.
- lf power is interrupted, the cell's contents will be lost,
- When power is restored, the latch will settle into a stable state, but it will not necessarily be the same state the cell was in before the interruption.
- SRAMs are said to be volatile memories

6. b) Briefly discuss the concept of virtual memory with a diagram [05] CO4 L2

- Virtual memory is an architectural solution to increase the effective size of the memory system.
- The addressable memory space depends on the number of address bits in a computer.
- For example, if a computer issues 32-bit addresses, the addressable memory space is 4G bytes.
- Physical memory typically ranges from a few hundred megabytes to 1G bytes.
- Large programs that cannot fit completely into the main memory have their parts stored on secondary storage devices such as magnetic disks.
- Pieces of programs must be transferred to the main memory from secondary storage before they can be executed.
- When a new piece of a program is to be transferred to the main memory, and the main memory is full, then some other piece in the main memory must be replaced.
- Operating system automatically transfers data between the main memory and secondary storage.
- Techniques that automatically move program and data between main memory and secondary storage when they are required for execution are called virtual-memory techniques.
- Programs and processors reference an instruction or data independent of the size of the main memory.
- Processor issues binary addresses for instructions and data.
- These binary addresses are called **logical or virtual addresses.**
- Virtual addresses are translated into physical addresses by a combination of hardware and software subsystems.
- If virtual address refers to a part of the program that is currently in the main memory, it is accessed immediately.
- If the address refers to a part of the program that is not currently in the main memory, it is first transferred to the main memory before it can be used.
- Memory management unit (MMU) translates virtual addresses into physical addresses.
- If the desired data or instructions are in the main memory they are fetched as described previously.
- If the desired data or instructions are not in the main memory, they must

be transferred from secondary storage to the main memory.

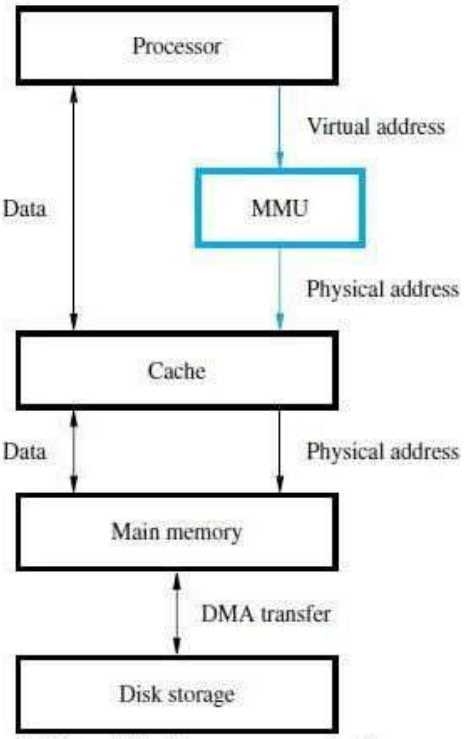
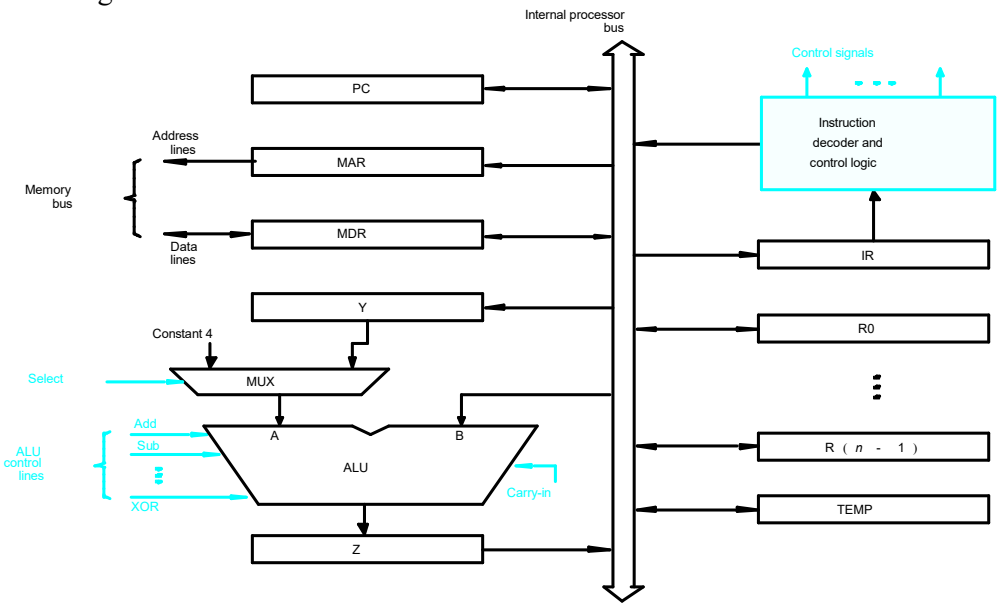


Figure 8.24 Virtual memory organization.

7. Explain the single bus organization of the datapath inside a processor with neat diagram. [10] CO5 L2



- Data Path:
- The registers, ALU, and interconnecting bus are collectively called the data path
- The arithmetic and logic unit (ALU) and all the registers are interconnected via a single common bus.
- This bus is internal to the processor
- Registers for temporary storage
- Various digital circuits for executing different micro-operations (gates, MUX, decoders, counters)

- Internal path for movement of data between ALU and registers
- Driver circuits for transmitting signals to external units
- Receiver circuits for incoming signals from external units
- **PC:** Keeps track of execution of a program. Contains the memory address of the next instruction to be fetched and executed.
- **MAR:** Holds the address of the location to be accessed.
- I/P of MAR is connected to Internal bus and an O/p to external bus.
  MDR: Contains data to be written into or read out of the addressed location. It has 2 inputs and 2 Outputs. Data can be loaded into MDR either from memory bus or from internal processor bus.The data and address lines are connected to the internal bus via MDR and MAR.
- **Registers**: The processor registers R_0 to R_{n-1} vary considerably from one processor to another.
- Registers are provided for general-purpose use by the programmer.
- Special purpose registers-index & stack registers.
- Registers Y, Z & TEMP are temporary registers - used by the processor during the execution of some instruction.
- **Multiplexer**:Select either the output of the register Y or a constant value 4 to be provided as input A of the ALU.
- Constant 4 is used by the processor to increment the contents of PC.

8. Develop the complete control sequence for the execution of instruction Add (R3), R1 [10] CO5 L3

1. Fetch the instruction: Fetching an instruction is the process of reading an operation code from memory and storing it in the instruction register (IR).
2. Fetch the first operand (the contents of the memory location pointed to by R3)
3. Perform the addition
4. Load the result into R1

Control Sequence for execution:

Step	Action
1	PC_{out} , MAR_{in} , Read, Select4,Add, Z_{in}
2	Z_{out} , PC_{in} , Y_{in} , WMF C
3	MDR_{out} , IR_{in}
4	$R3_{out}$, MAR_{in} , Read
5	$R1_{out}$, Y_{in} , WMF C
6	MDR_{out} , SelectY, Add, Z_{in}
7	Z_{out} , $R1_{in}$, End