Name : M. N. KHAN

ID : 2216080041

Subject: Computer Architecture.

Code : CSE 313.

Ans: to the: 3: No: 01 (a)

Ans: IBM z EC12 I/O Channel subsystem:- The z enterprise EC12 is IBM's latest mainframe computer offering (At the time of this writing). This system is based on the use of the z EC12 Prossesor chip, which is a 5.5-GHz Multicore chip with six cores. The zEC12 architecture can have maximum of 101 Prossesor chip with 6 cores.

Channel structure:- ≤15 Partition per channel subsystem.

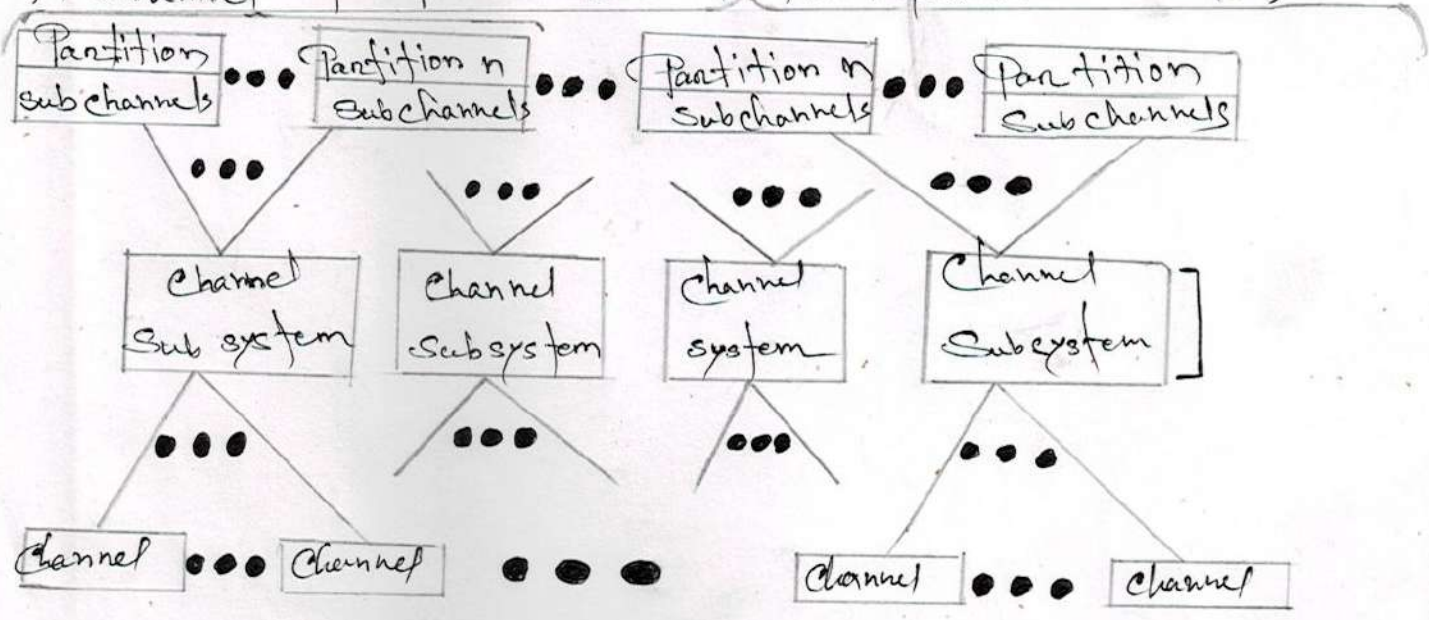



Fig: IBM zEC12 I/O channel sub system structure.

Ans: to the Q. NO: 01 (b)

Ans: I/O device completes an I/O operation's Sequence of Hardware:

The occurence of an interrupt triggers a number of events, Both in the prosseser Hardwere and in softwere. figure 7.6 Shows a typicle sequencel. when an I/O device completes an I/O operation. The following sequence of Hardwere events occurs.

* → The device issues an interrupt signal to the prosseser.
** → The prosseser finishs execution of the current instruction before responding to the interrupt, as indicated in.

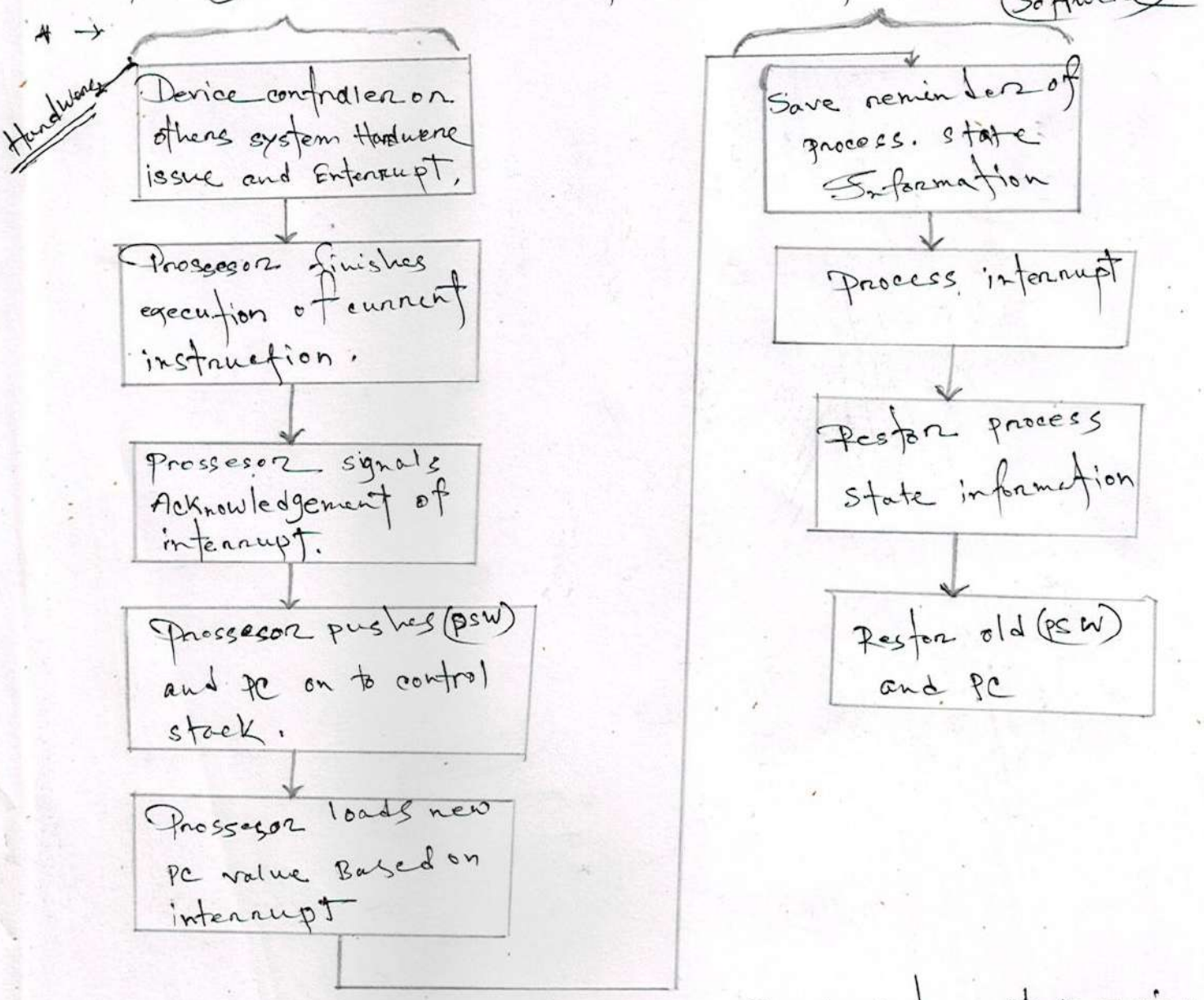



figure: 7.6 Simple interrupt processing.

→ The processor now needs to prepare to transfer to transfer control to the interrupt routine. To begain, it needs to save information needed to Resum the current program at the point of interrupt. The menimum information required is (a) the status of the processor which contained in a register called the program status word (PWS) and (b) the location of the next instruction to be executed which is contained in the program Counter. These can be pushed on to the says tem control stock 2.

→ The processor now loads the program counter with the entry location of the interrupt-handling program that will respond to this interrupt. Depanding on the computer Architecture and operating system design. There maybe a signal program on program foreach type of interrupt; On on one program for each device and each type of interrupt. If there is more then on interrupt handilling routine. The processor must determind which one of invoke. The information may have been include in the original interrupt signal or the Processor. May have to issue a request to the device that issued the interrupt to get a responce that contains the needed information.

At this point, the program counter and (PSW) Relating to the interrupted Program have been saved on the system stack. However, there is other information that is considered part of the "state" of the executing Programs. In particular, the contents of the processor registers need to be saved, because these registers maybe used by interrupt handle. So all of these values plus any other stat information need to be saved. Typically, the interrupt handler.

P.T.O

- The interrupt handeler next processes the interrupt. This include an examination of status information relating to the I/O operation or other event that caused an interrupt. It may Also involve Sending Additional commands or acknowledgement to the I/O Device.

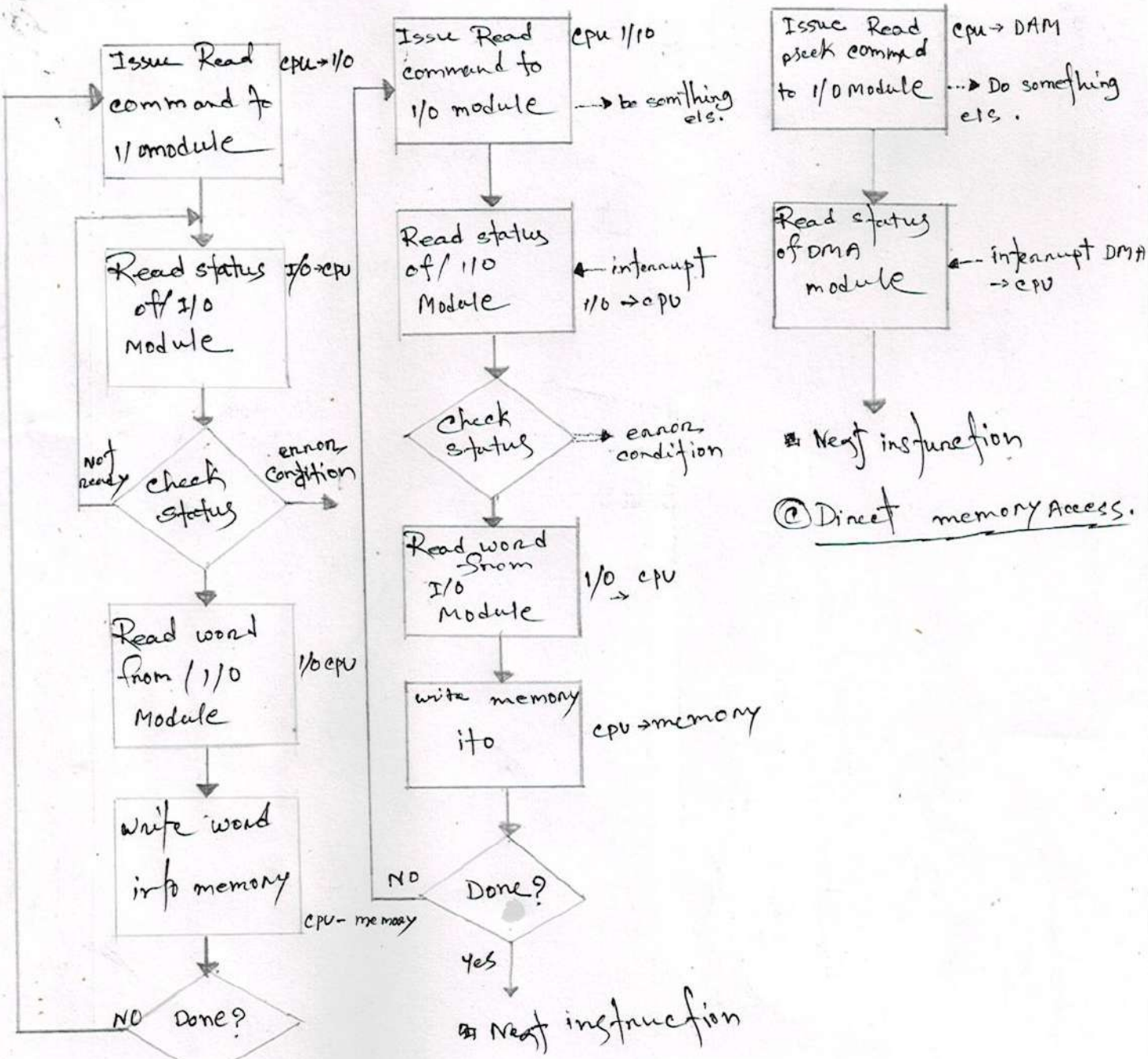
- When interrupt Processing in completed the save register value are retrieved from the stack and Rostor to the register.

- The final act is to restor (psw) and program counter value from the stack. As result, the next instruction to be executed will be from the previously interrupted Program.

- Note that it is important to save all the state information about the interrupted program for later resumption. This is because that interrupt is not a routine called from the Program. Its occurrence is unpredictable.
Indeed as will see in the next chapter the two Programs not may have anythings in common & may belong to two different user.

Q.no.to:The:3:No:01 (c)

Ans: Describing three techniques for input of a Block diagram:




(a) Program I/O

(b) Interrupt-Driven I/O.

(c) Direct memory Access.

→ Three techniques for input of Block data.

I/O - Instruction: Processor views I/O operation in a similer manner as memory operation each device is given a unique indentifier or address. Processor issue commands containting device address - I/O module must check address line to see if the command if for itself.

→ I/O mapping

→ memory – mapped I/O
→ Isolated I/O.

→ Interrupt - Driven I/O

→ Overcomes the processor having to wait long periods of time for I/O models.

→ The processor does not have to repetedly check the I/O Model status.

* Direct Memory Access: Drawbacks of programed and Interrupt - Driven I/O.

→ I/O transfer rate limited to speed that processor can test and service devices.

→ Processor tied up managing I/O transfer.

** DMA function:

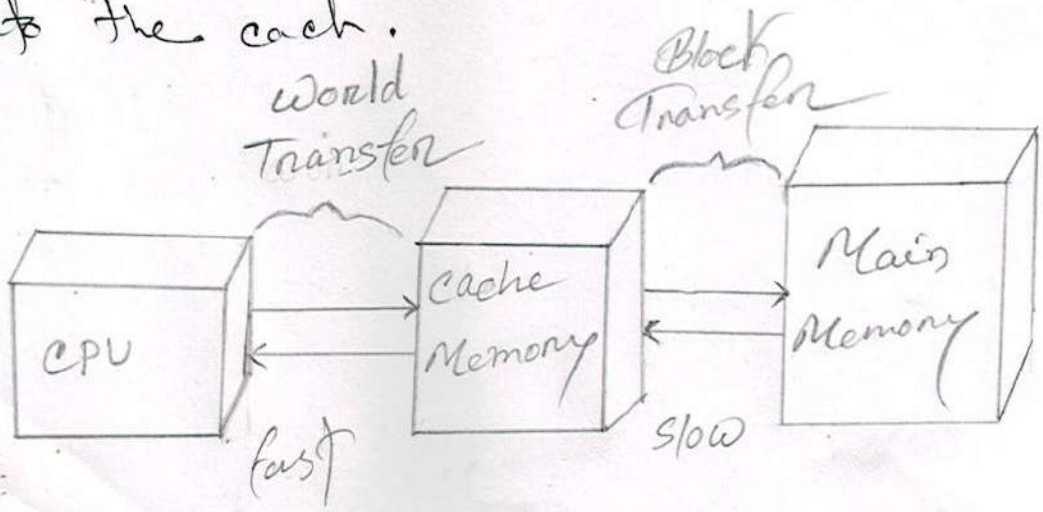
→ DMA Module on system bus used to mimic the Processor.

→ DMA Module may temporarily force processor to suspend operation cycle stealing.

→ DMA Module only used system bus when processor doesn't need it.

Ans: to the Q: No: 02 (a)

Ans: Describing the cache memory & main memory: Cache memory is designed to combine the memory access of expensive, high-speed memory combined with the large memory size of by expensive, lower-speed memory. The concept is illustrated. There is a relatively large & slow main memory together with a smaller, faster, cache memory. The cache contain a copy of portions of main memory. When the processor attempts to read a word of memory check is made to determine if the word is in the cache. If so the word is delivered to the processor. If not a block of main memory. consisting of some fixed number of word is read into the cach.




The cache is much faster and more expensive the main memory. However, both of the computer memories are directly accessible by the Processor.

## Basic of Comparison:

| | Cache memory | Main Memory |
|---|---|---|
| Purpose | It is used to store frequently used Data | It holds the data that is currently being processed. |
| Access | Comparatively faster than memory. | It is also the faster accessing memory. |
| Cost | More expensive than main memory. | Expensive memory. |
| Size | Comparatively smaller than main memory. | Large than cach memory. |
| Types | L1, L2, L3 | SRAM & DRAM |

Ans: the Q: No. 02: (b)

Ans: logical and physical caches describing:

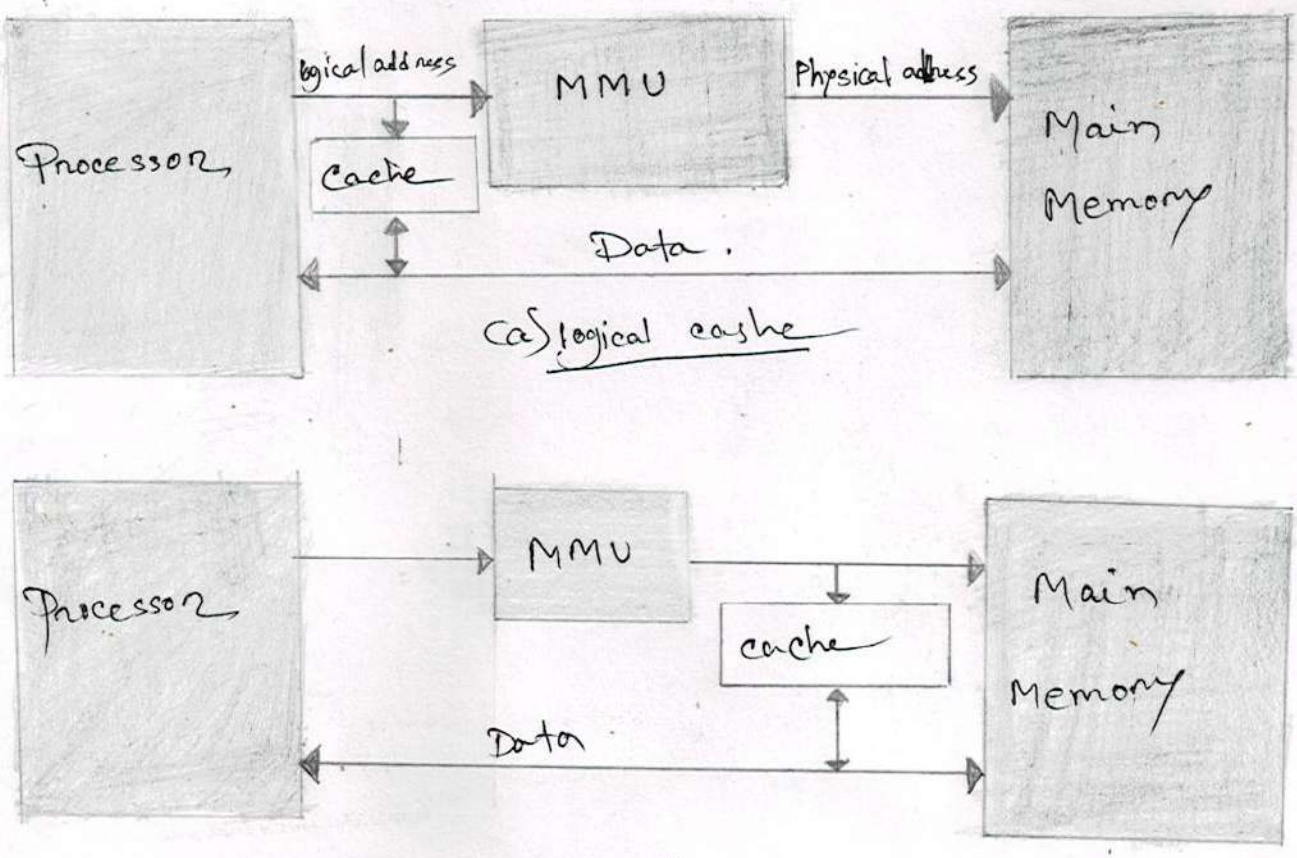



Fig: logical cache & physical cache:

logical cache:

→ It is also known as a virtual cache.

→ It is store data using virtual addresses.

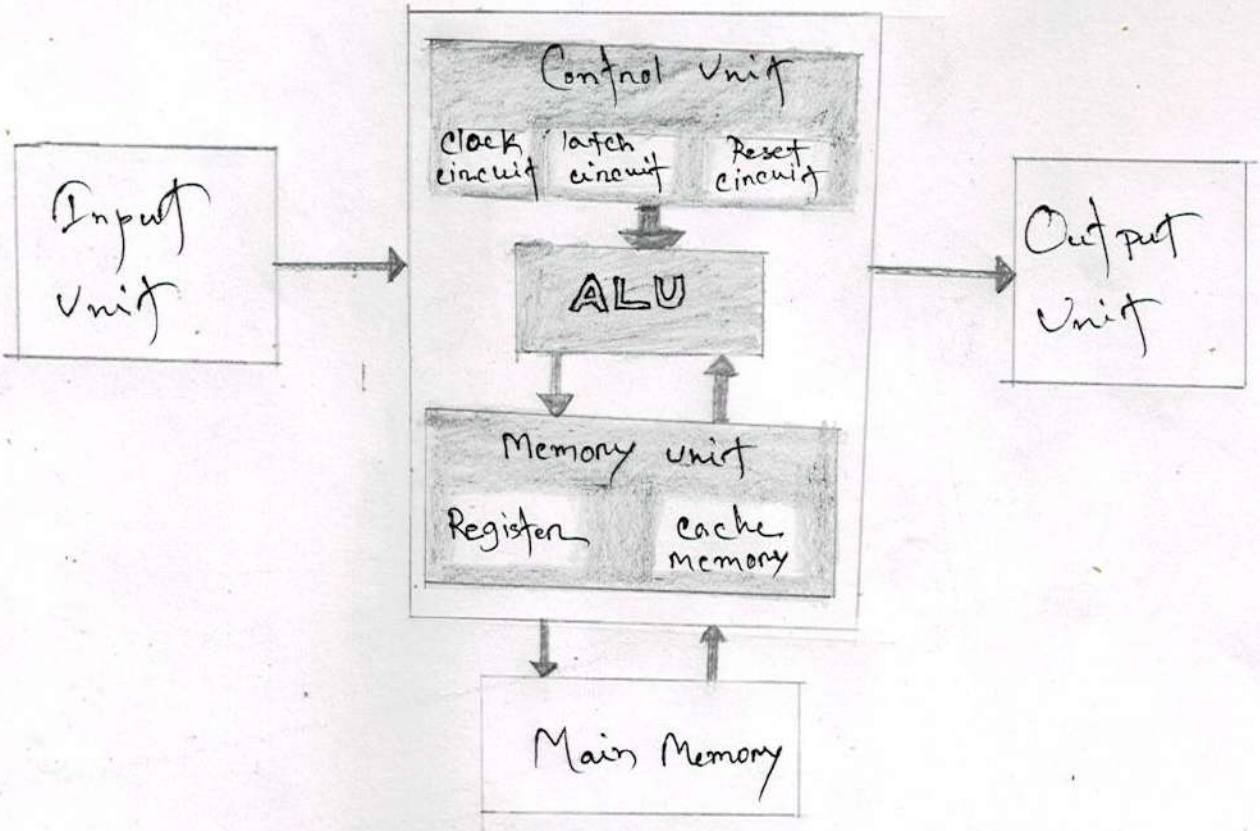
→ The processor access caches directly without going through MMU.

→ Advantage: It is faster then physical cache. Because the cache can respond before the MMU perform an address translation.

Physical caches: The processon access the cache directly without going through the MMU. A physical caches stones data using main memory physical adress. One obvious advantage of the logical cache it that cache access speed is faster then for a physical cache. Because the cache can respons Before the MMU Panforms an adress translation. The disadvantage his to do with the fact that most virtual memory system supply caches application with the same virtual memory Adress space. That is each application sees a virtual memory that stand at adress 0. Thus the same virtual adress in two Different application refers to two different physical adress. The cache memory must there for be completly flushed with cache application context.

Ans: to the Q. NO: 03 (a)

Ans: Central Processing Unit (CPU)




→ Central Processing main Unit:

→ Control Unit → click circuit
→ latch circuit
→ Reset circuit

→ Arithmatic & logical Unit (ALU)

→ Memory Unit → Register
→ cache memory

~~Ans: to the Q. No: 03 (b)~~

Ans: ⇒ ~~In~~ of 8086 Addressing indirect defining:

→ In register indirect addressing mode, the address of the memory location where the operand resides is held by a Register.

→ The register used for this respond purpose are SI, DI, & BP.

→ If these three register are used as pointer, they must be combined with ~~d~~sign Order to generate the 20 bit physical Adress.

Physical address,

| Processor Register | | |
|---|---|---|
| AX | AH | AL |
| BX | BH | BL |
| CX | CH | CL |
| DX | DH | DL |
| SI | | |
| DI | | |
| BP | | |
| SP | | |
| CS | | |
| DS | | |
| SS | | |
| ES | | |

| Memory | |
|---|---|
| memory Address (in Hex) | Data |
| 10000 | AA H |
| 10001 | BB H |
| 10002 | CC H |
| 10003 | EE H |
| 10004 | DD H |
| 10005 | EE H |
| 10006 | FF H |
| 10007 | 11 H |
| 10008 | 22 H |
| 10009 | 33 H |
| 1000A | 44 H |