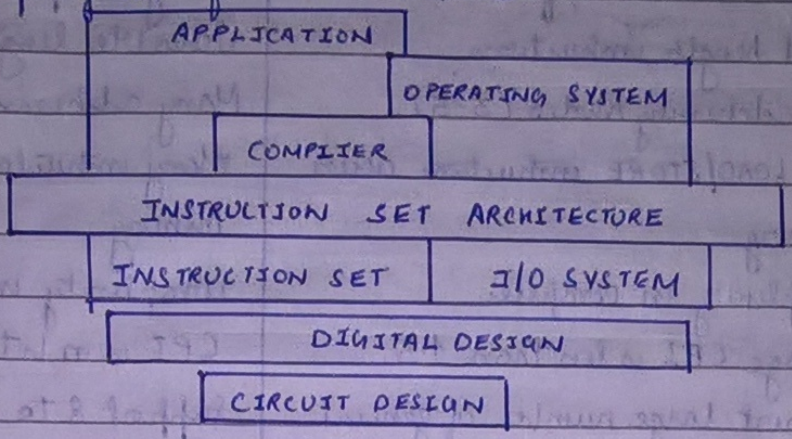
UNIT- II

INSTRUCTION SET ARCHITECTURE

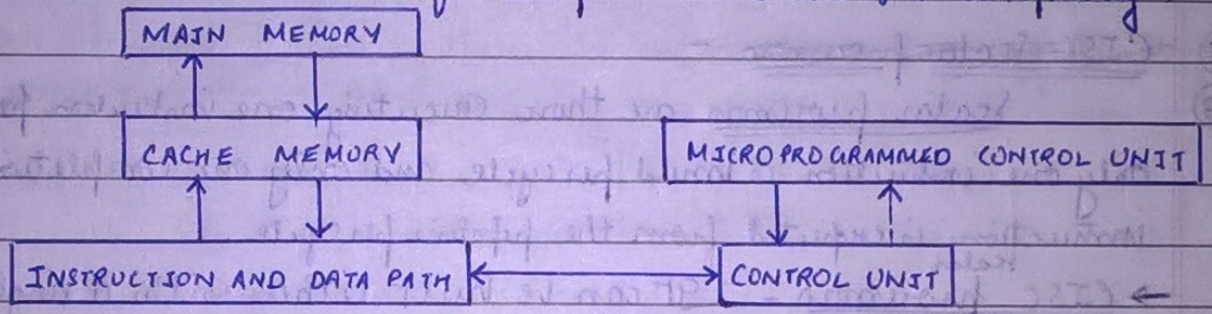
① (ISA) Instruction set architecture is the structure of a computer that a machine language programme (or a compiler) must understand to write a correct program for that machine.



Portion of ISA

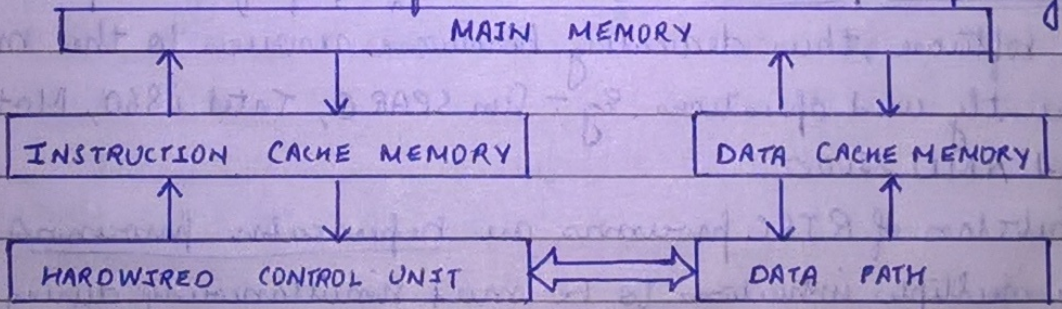
→ CISC architecture with microprogrammed control unit -

CISC stands for complex instruction set computing.

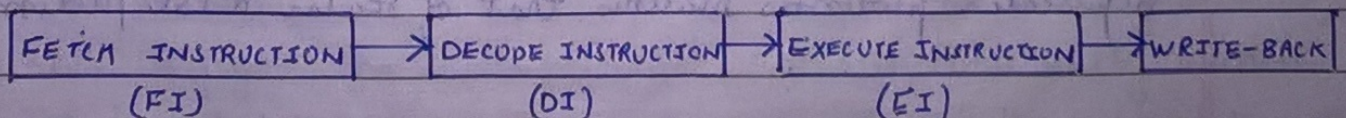


→ RISC architecture with hardware control unit -

RISC stands for reduced instruction computing.



→ Stages in instruction pipeline -



② Comparison between RISC and CISC –

S.No.	RISC	CISC
1.	Simple instructions, few in numbers	Many complex instructions
2.	Fixed length instructions	Variable length instructions
3.	Few addressing modes (3-5).	Many addressing modes (12-24)
4.	Only LOAD/STORE instructions access memory	Many instructions can access memory
5.	Complexity in compiler	Complexity in microcode
6.	Average CPI is less than 1.5	CPI is in between 2 and 15
7.	Support large number of general purpose registers	Support 8 to 24 number of general purpose registers
8.	Highly pipelined	less pipelined

③ ~~CISC scalar processor~~ –

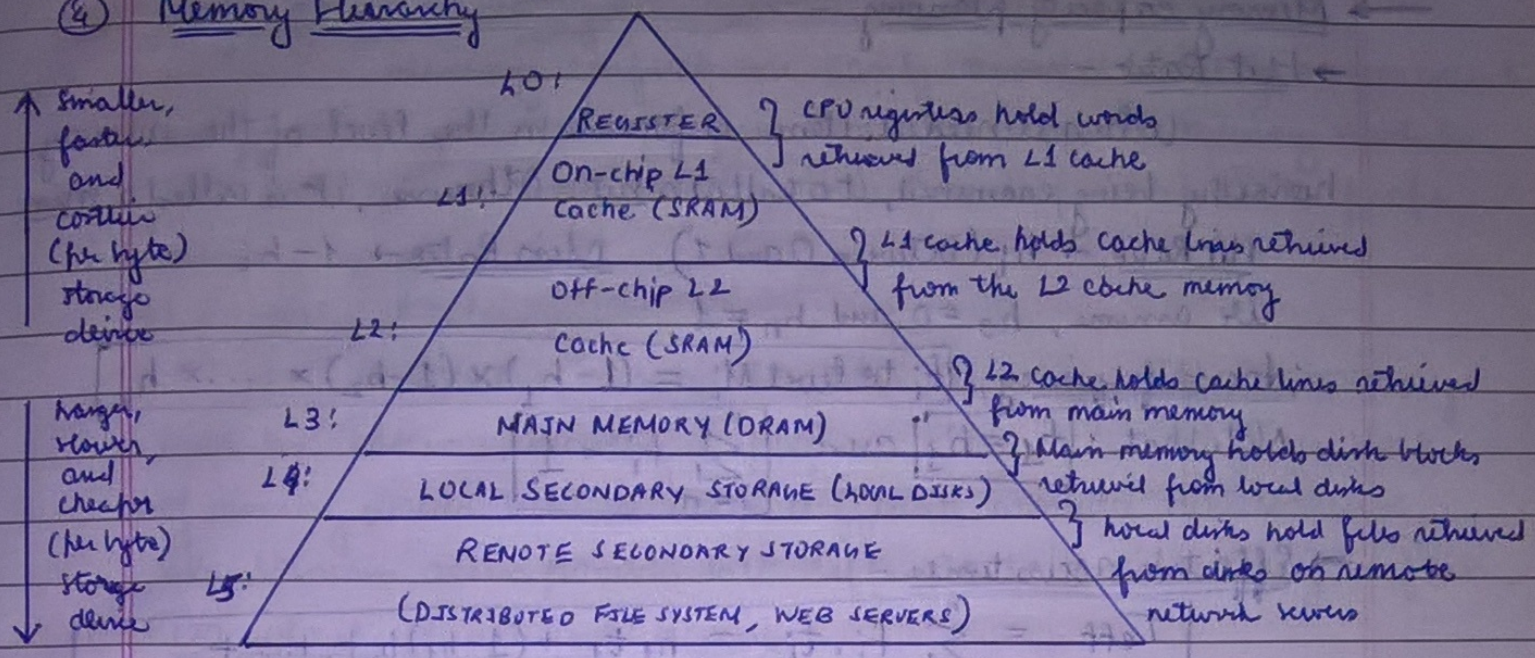
③ Scalar processors are those executing one instruction per cycle, only one instruction is issued per cycle and only one completion of instruction is expected from the pipeline per cycle.

→ CISC scalar processors – It can be built either with a single chip or with multiple chips mounted on a processor board.
Eg – Intel 386 and 486, M68040, VAX 8600 etc.

→ RISC scalar processors – It moves least frequent operations into software, thus dedicating hardware resources to the most frequently used operations. Eg – Sun SPARC, Intel i860, Motorola M8810 and AMD 29000.

*****→ Subclass of RISC processors are superscalar processors, which allow multiple instructions to be issued simultaneously during each cycle. Thus the effective CPI of a superscalar processor should be lower than that of a scalar RISC processor. The clock rate of superscalar processors matches that of scalar RISC processors

④ Memory Hierarchy –



Organizing memory and storage systems is known as memory hierarchy.

→ Memory hierarchy stores the information it satisfy three properties-

(1) Inclusion - It is stated by the following set inclusion relations among n memory levels.

$$M_1 \subset M_2 \subset M_3 \subset \cdots \subset M_n$$

M_0 → cache memory, (lowest level)

M_n → contains all of the information words stored (highest level).

(2) Coherence - Copies of the same information item at higher levels of the memory hierarchy be consistent. Two strategies to maintain -

Write-Through (WF) - Update data immediately to all higher levels

Write-Back (WB) - Delay the update of copies at higher levels until the data being modified in the lower level is replaced or removed from that level.

(3) Locality - Three dimension of the locality principle are -

→ Temporal locality - If a program accesses one memory address, there is a good chance that it will access the same address again.

→ Spatial locality - If a program accesses one memory address, there is a good chance that it will also access other nearby addresses.

→ Sequential locality - Execution of program that follows certain sequential order.

→ Memory capacity planning -

→ Hit Ratio -

When a needed item is found in the level of the memory hierarchy being examined, it a called a hit. Otherwise, it is called a miss.

Hit Ratio → h_i (between 0 and 1) Miss Ratio → $1 - h_i$

We assume, $h_0 = 0$ and $h_n = 1$.

Access frequency, f_i to level $M_i = (1-h_1) \times (1-h_2) \times \ldots \times h_i$

Note that $f_1 = h_1$ and $\sum_{i=n}^{n} f_i = 1$

→ Effective access times -

$$T_{eff} = \sum_{i=1}^{n} f_i \cdot t_i = h_1 t_1 + (1-h_1) h_2 t_2 + \ldots + (1-h_1)(1-h_2)\ldots(1-h_{n-1}) h_n t_n$$

→ Hierarchy optimization

Total cost of a memory hierarchy is estimated as follows -

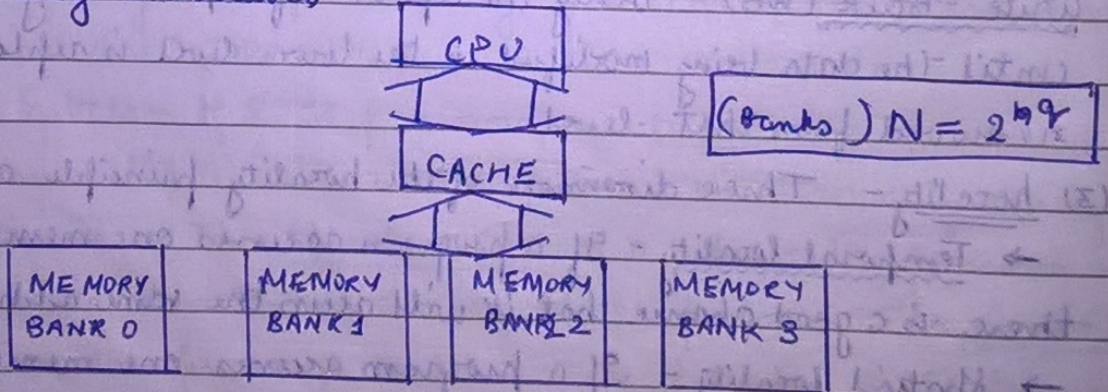
$$C_{total} = \sum_{i=1}^{n} c_i s_i$$

$c \rightarrow$ cost, $s \rightarrow$ capacity

Since $c_1 > c_2 > c_3 > \ldots > c_n$, we have to choose $s_1 < s_2 < s_3 < \ldots < s_n$

⑤ Interleaved memory organization -

In an interleaved memory, the memory is divided into a set of banks. An interleaved memory with n banks is said to be n-way interleaved



(Banks) $N = 2^{hq}$

There are two types of interleaving -

(1) High-order (2) low-order. (q-bits are lower)
(q-bits are higher)

→ High-order interleaving -

Memory conflicts are easily avoided. Each processor executes a different program and program stored in separate memory modules. Interconnection network is set to connect each processor to its proper memory module.

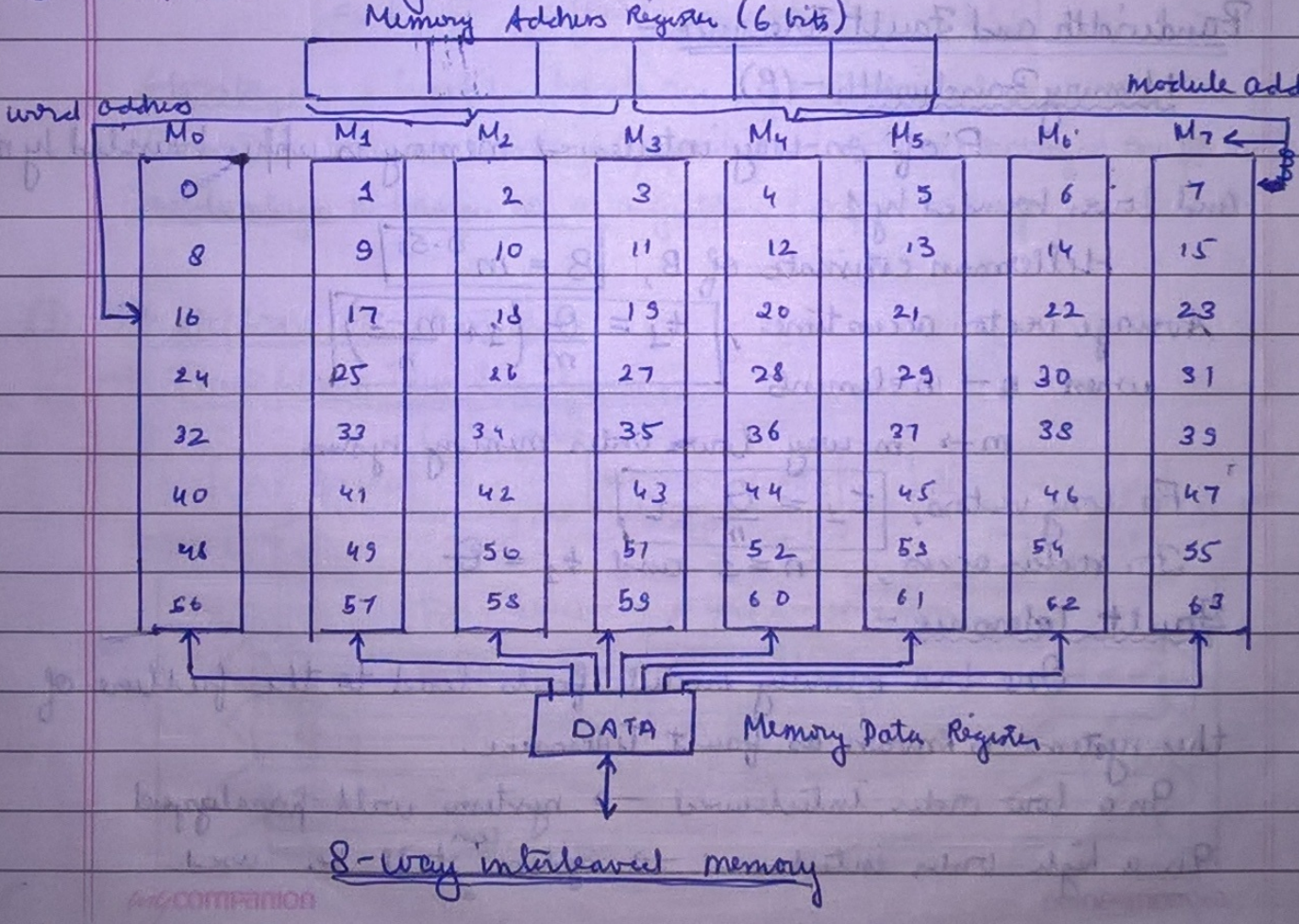
→ Low-order interleaving -

Consecutive memory locations reside in different memory modules. Process executing a program stored in a contiguous block of memory would need to access different modules simultaneously. Simultaneous access possible but difficult to avoid memory conflicts.

→ Drawbacks of interleaved memory -

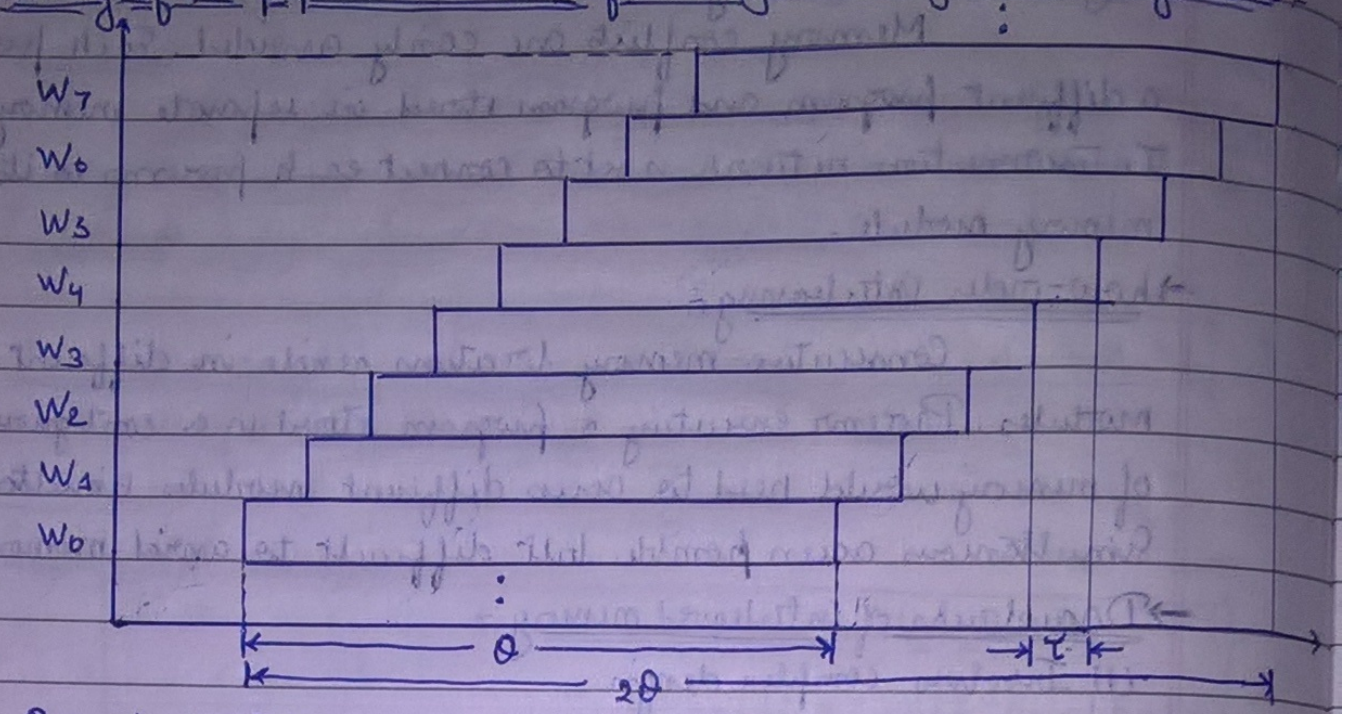
(1) Involves complex design
(2) Reduced fault-tolerance
(3) Cannot be expanded incrementally.

⑥ Pipelined Memory Access -



8-way interleaved memory

Timing of the pipelined access of the eight contiguous memory words -



θ = Major Cycle
τ = Minor Cycle
n = Degree of interleaving

$$\tau = \frac{\theta}{m}$$

⑦ Bandwidth and Fault Tolerance -

Memory Bandwidth - (B)

B of on-way interleaved memory is upper bounded by m and lower bounded by 1

Hellerman estimate of B, $B = m^{0.56}$

Average vector access time, $t_1 = \frac{\theta}{m}\left(1 + \frac{m-1}{n}\right)$

where n → n elements

m → m-way lower order memory system

For long vectors, $t_1 = \frac{\theta}{m} = \tau$

For scalar accesses, $n = 1$ and $t_1 = \theta$

Fault Tolerance -

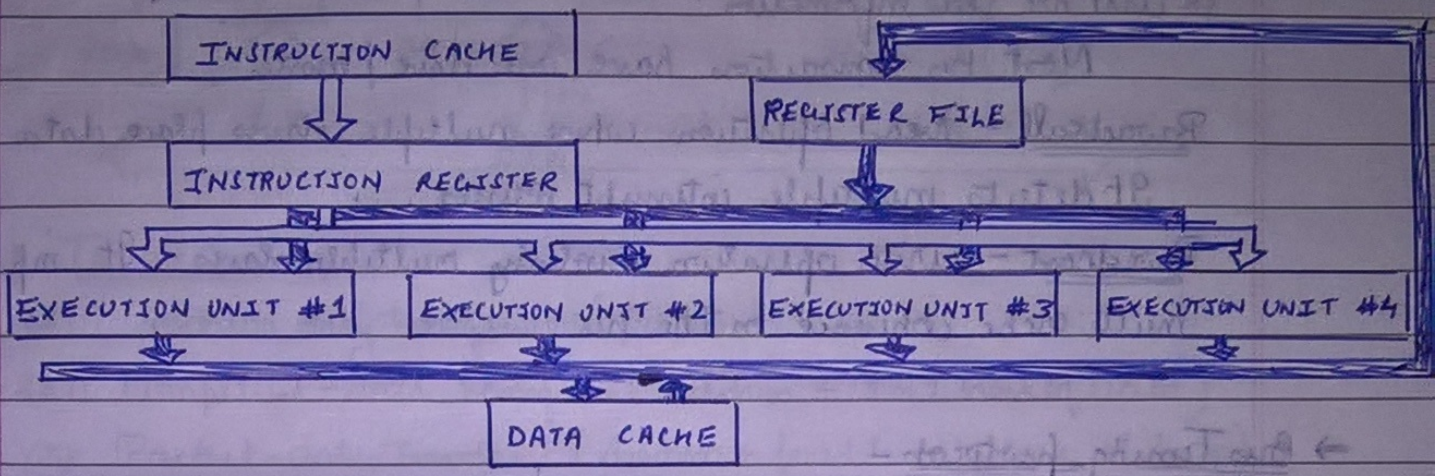
One ~~ten~~ memory module fails lead to the failure of the system is known as fault Tolerance.

In a low order interleaved → system will paralyzed

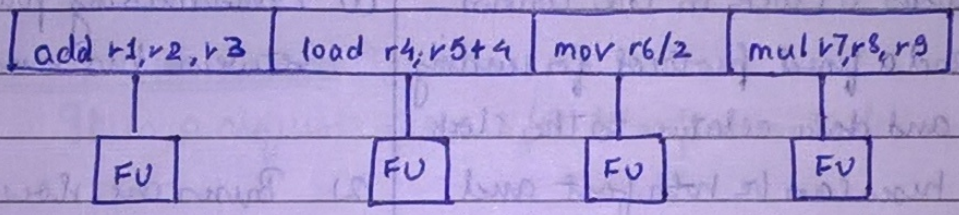
In a high order interleaved → system can still be used.

⑧ VLIW architecture - (Very long Instruction word)

In a VLIW machine, many operations (instructions in a normal machine) are encoded in a single instruction. The word of the instruction is very long and can contain multiple operations independent



VLIW instruction execution -

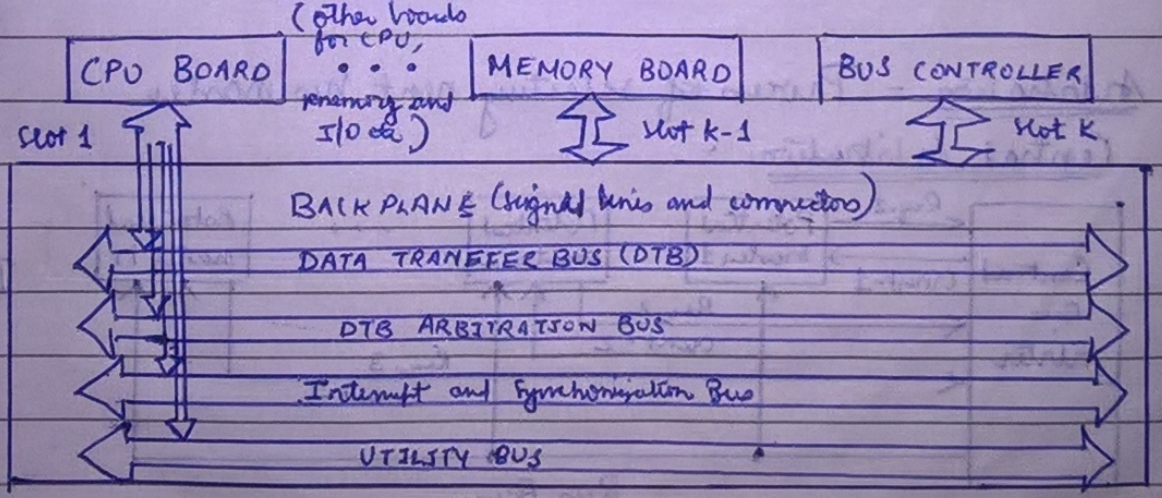


Advantages - Simpler hardware, good compilers, compiler friendly, Run time behaviour is highly predictable, Easily managed exceptions & interrupts

Disadvantages - Large no. of registers, large code size

⑨ Backplane Bus system -

→ Backplane Bus Specification -



→ Addressing and Timing Protocols -

Bus addressing -

The backplane bus is driven by a digital clock with a fixed cycle time called bus cycle. Backplane has limited physical size, so will not skew information.

Most bus transactions have one slave / master.

Broadcall - Read operation where multiple slaves place data on bus. It detects multiple interrupt sources.

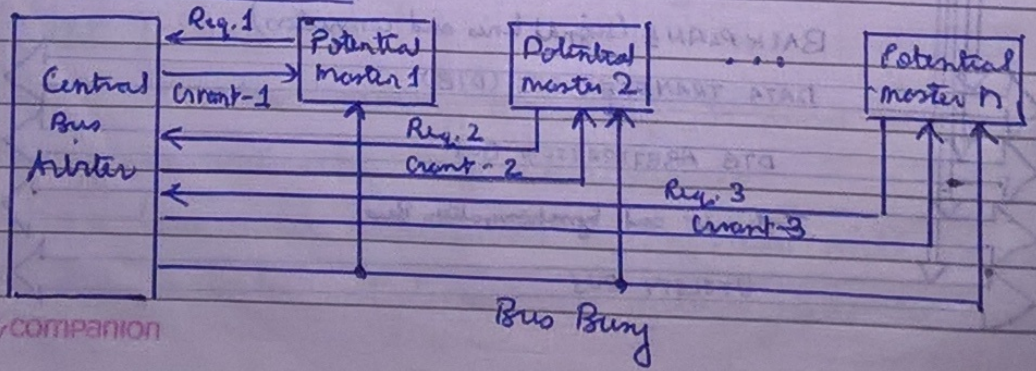
Broadcast - Write operation involving multiple slaves. It implements multicache coherence on the bus.

→ Bus Timing protocols -

Synchronous Bus	Asynchronous Bus
(1) It includes a clock in the control lines and a fixed protocol for sending address and data relative to the clock	(1) Handshaking protocols are used between bus sender and receiver
(2) These buses can be both fast and inexpensive	(2) Buses are slow
(3) Buses cannot be long	(3) Buses can be long enough
(4) Everything on a bus must run at the same clock rate	(4) Same clock is not used.
(5) CPU-memory buses are typically synchronous	(5) I/O buses are more likely to be asynchronous.

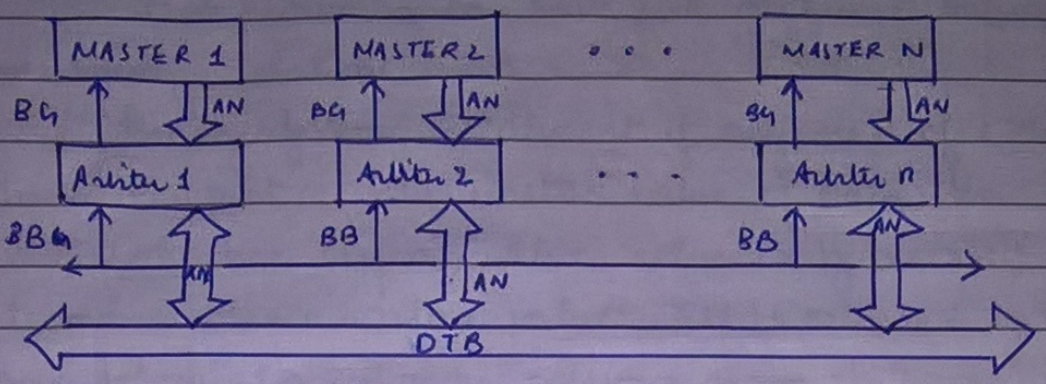
→ Arbitration - Process of selecting next bus master.

Central Arbitration -



Daisy Chaining

Distributed (Decentralized) arbitration - priority based arbitration is used



→ Transaction modes -

(1) Address only transfer (Address)

(2) Compelled-data transfer (Address + 1 or 2 word of Data)

(3) Packet-data transfer (Address + fixed length block of data)

A bus transaction consists of a request followed by a response

→ Interrupt Mechanism -

It is a request from I/O to a processor for service or attention.