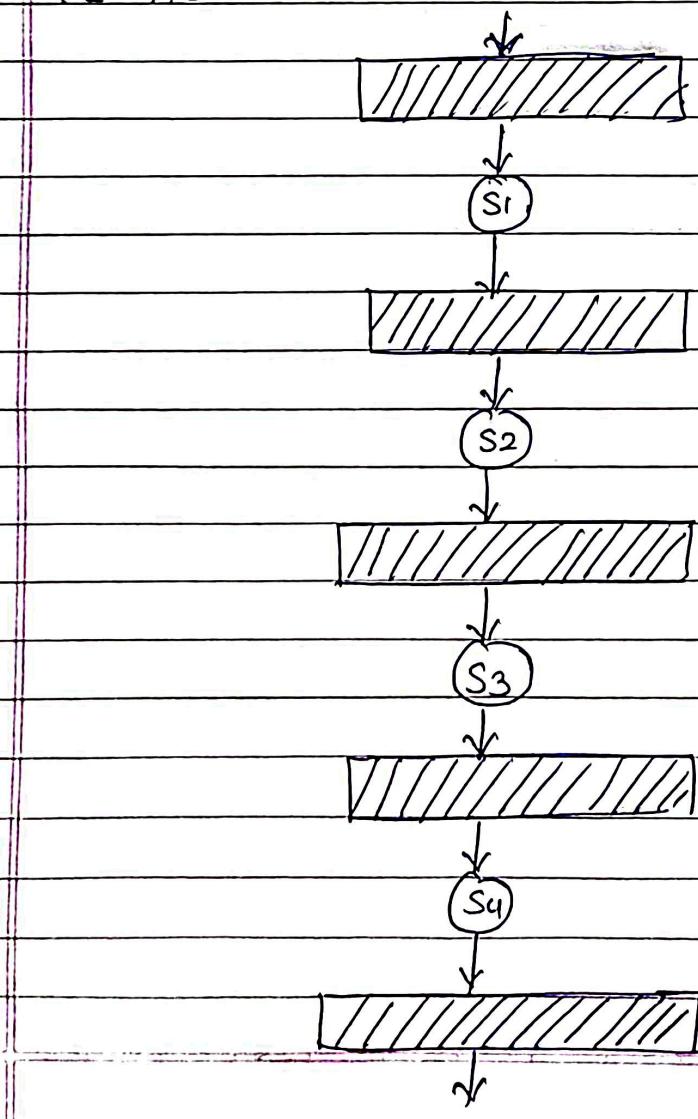


Q8.1. How pipeline architectures are classified?

→ According to levels of processing, Handler has proposed following classification scheme for pipeline processors -

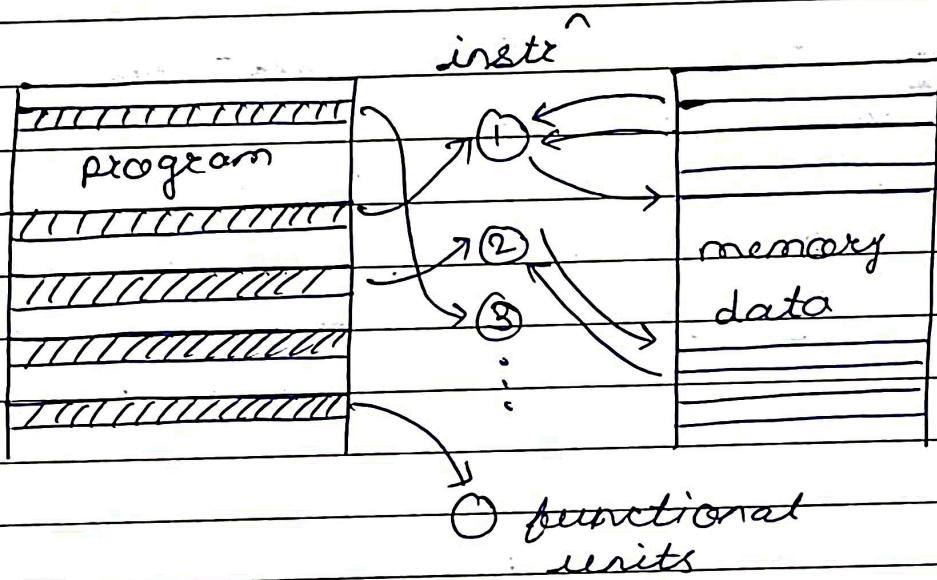
i) arithmetic pipeline -

The arithmetic logic units of a computer can be segmented for pipeline operations in various data formats. Examples are four-stage pipes used in stat-100, eight-stage pipes used in TI-ASC.



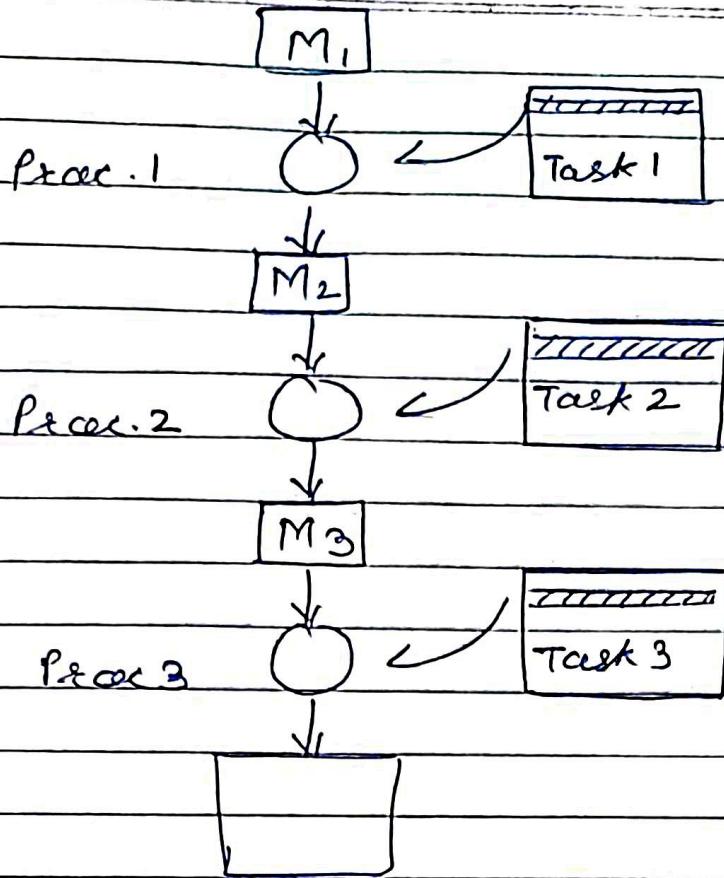
## 2) Inst<sup>n</sup> pipeline -

The execution of stream of inst<sup>n</sup> can be pipelined by overlapping the execution of current inst<sup>n</sup> with the fetch, decode & operand fetch of subsequent inst<sup>n</sup>. This technique is also known as inst<sup>n</sup> lookahead.



## 3) processor pipelining -

The data processor stream passes the first processor with results stored in memory block which is also accessible by second processor. The second processor passes the refined results to third & so on.



Ramamoorthy & Li have proposed following three pipeline classification schemes.

1] Unifunction vs. multifunction pipelines-

A pipeline unit with a fixed & dedicated func<sup>n</sup>, such as floating point adder is called unifunctional pipeline.

A multifunctional pipeline may perform different functions, either at different times or at same time, by interconnecting different subsets of stages in pipeline.

2] static vs dynamic pipelines-

A static pipeline may assume only one functional configuration at a time. static pipelines can be either unifunctional or multifunctional. A dynamic pipeline processor permits several functional configurations

to exist simultaneously. A dynamic pipeline must be multifunctional.

### 3) scalar vs vector pipeline-

A scalar pipeline processes a sequence of scalar operands under control of DO loop. Instructions in a small DO loop are prefetched into inst<sup>n</sup> buffer. Vector pipelines are specially designed to handle vector inst<sup>n</sup> vector operands. Computers having vector inst<sup>n</sup> are called vector processors.

Q. 2. State & explain various pipeline architecture performance measures:

→ 1] clock period - The logic circuitry in each stage  $S_i$  has a time delay denoted by  $T_i$ . The clock period is denoted by  $T$ .

$$T = T_i + T_m$$

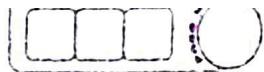
where  $T_m = \max \{T_i\}^k$

2] frequency - The reciprocal of clock period is called frequency of pipeline.

$$f = \frac{1}{T}$$

### 3) speedup

4] The product of time interval & a stage space in space-time graph is called time-space span. A given time-space span can either be in busy state or idle state but not in both.



### 5) efficiency -

The efficiency of sine pipeline is measured by percentage of busy time-space span over total time-space span which is equal to sum of all busy & idle time-space span.

$$\therefore \eta = \frac{n}{k + (n-1)}$$

$$\text{also } \eta = \frac{sk}{k}$$

### 6) throughput -

The number of results (tasks) that can be completed by pipeline in unit time is called throughput.

$$\omega = \frac{n}{t}$$

Q. 3. What are vector processors. Draw & explain Cray-1 architecture. State characteristics of Cray-1.

→ Vector processors

- Vector processor is basically a central processing unit that has the ability to execute complete vector input in a single inst.

- It a complete unit of hardware resources that executes a sequential set of similar data items in the memory using a single inst.

- It holds single control unit but has multiple execution units that perform the same operation on different data elements.

of vector.

- A vector processor operates on multiple pairs of data.

-

- A vector operand contains an ordered set of  $n$  elements, where  $n$  is called length of vector. Each element in a vector is a scalar quantity which may be floating-point number, an int, a logical value or char. Vector instructions can be classified into four primitive types:

$$f_1: V \rightarrow V$$

$$f_2: V \rightarrow S$$

$$f_3: V \times V \rightarrow V$$

$$f_4: V \times S \rightarrow V$$

where  $V$  &  $S$  denoted vector operand & scalar operand resp.

### Cray-1 architecture

- The architecture of Cray-1 consists of number of working registers, large instruction buffers & data buffers & 12 functional pipeline units.

- The clock rate in the Cray-1 is 12.5 ns.

- A front-end host computer is required to serve as the system manager.

- The CPU contains a computation section, a memory section & I/O section.

- 24 I/O channels are connected to front-end computer, I/O stations, peripheral equipment,

the mass-storage subsystem & Maintenance Control Unit (MCU).

- The front-end system will collect data, present it to the Cray-1 for processing & receive output from Cray-1 for distribution to slave devices.

computation section

- registers
- functional units
- inst<sup>n</sup> buffers

Memory section

0.25 M or 0.5 M or 1 M

64-bit bipolar words

I/O section

• 12 input channels

• 12 output channels

MCU

Mass storage subsystem

front-end computer, I/O stations & peripheral equipment

fig:- front-end system interface & Cray-1 connection.

characteristics

I/O section -

- consists of 12 input & 12 output channels

- channel priority resolved within channel groups

- Each channel has max transfer rate of 80 Mbytes/s.

- lost data detection.

- 4 input & 4 output channels operate simultaneously to achieve

mass transfer of instruction to computation section.

- MCU -
  - Handles system initiation & monitor memory system performance
  - contains  $64 \times 4$  inst<sup>n</sup> buffers
  - high speed registers for scalar & vector processing.

- four clock period bank cycle time
- one word per clock period transfer rate to B, T & V registers
- one word per two clock periods transfer rate to A & S registers
- single error correction & double error correction. detection.

- computation section -
  - 64 bit word length
  - 2's complement arithmetic
  - scalar & vector processing model
  - Eight 24-bit address (A) registers
  - four inst<sup>n</sup> buffers      • 128 inst<sup>n</sup> codes
  - int & floating point arithmetic

Q8.4.

What are different hazards in pipeline architecture?

- - A few instructions are at same stage of execution in a pft pipelined design. There is a chance that these sets of instr<sup>n</sup> will become dependent on one another, reducing the pipeline's pace.
- A hazard prevents an instr<sup>n</sup> present in pipe from being performed during specified clock cycle.
- There are three hazards in pipeline architecture

#### 1) structural hazard

- Hardware resource conflicts among the instr<sup>n</sup> in pipeline cause structural hazards.
- When more than one instr<sup>n</sup> in the pipe requires access to the very same resource in the same clock cycle, a resource conflict is said to arise.
- This is a circumstance where hardware cannot handle all potential combinations.

#### 2) data hazards

- Data hazards in pipelining emerge when the execution of one instr<sup>n</sup> is dependent on the result of another instr<sup>n</sup> that is still being processed.
  - Branches & other instr<sup>n</sup> that change the PC make the fetch of the next instr<sup>n</sup> to be delayed.
  - Data hazard can be dealt with either
- #### 2) hardware technique or software technique
- Hardware technique - interlock → Hardware detects the data dependency & delays the scheduling of the dependent instr<sup>n</sup> by stalling

enough clock cycles.

- software technique  $\rightarrow$  inst<sup>n</sup> scheduling for delayed load.

### 3) control hazards

- Branch hazards are caused by branch inst<sup>n</sup> & are known as control hazards in computer architecture.
- The flow of program is controlled by branch inst<sup>n</sup>.
- Conditional statements that are used in higher-level languages for iterative loops are converted into one of the branch inst<sup>n</sup> variations.

## Q.5. Applications of parallel architecture

- $\rightarrow$  ① one of the primary applications is databases & data mining
- ② The real time simulation of systems
- ③ technologies such as networked video & multimedia.
- ④ science & engineering
- ⑤ collaborative work environments
- ⑥ augmented reality, advanced graphics & virtual reality.

## Q.6. What are development tracks in HPCA?

- $\rightarrow$  i) Multiple processor track -

- In the multiple processor track, the source of parallelism is assumed to be concurrent execution of different threads on different processors, with communication occurring

through shared memory or via message passing.

- It can be shared memory multiprocessor or a distributed memory multicomputer.

a) Shared memory track -

shows a track of multiprocessor development employing a single address space in the entire system.

b) Message passing track -

The cosmic cube pioneered the development of message passing multicomputers.

c) Multivector & SIMD tracks -

are useful for concurrent scalar/vector processing.

a) Multivector track -

These are traditional vector supercomputers.

The CDC 7600 was first vector dual processor system. Two subtracks were derived from CDC 7600.

b) SIMD track -

The illiac IV pioneered the construction of SIMD computers.

c) Multithreaded & dataflow track

a) Multithreaded track -

The term multithreading implies that there are multiple threads of control in each processor. Multithreading offers an effective mechanism

for hiding long latency in building large-scale multiprocessors.

### b) dataflow track-

The key idea is to use a dataflow mechanism instead of control-flow mechanism to direct the program flow. Fine-grain instruction level parallelism is exploited in dataflow computer.

Q. 7. A 40-MHz processor was used to execute a benchmark program with the following instruction mix cycle counts:

instruction type	execute count	clock cycle count
arithmetic	45000	1
data transfer	32000	2
floating pt.	15000	2
control transfer	8000	2

$$\rightarrow CPI = \frac{C}{I_c}$$

= total cycles to execute a whole program  
total instr<sup>n</sup>

$$= 45000 \times 1 + 32000 \times 2 + 15000 \times 2 + 8000 \times 2$$

$$= 45000 + 32000 + 15000 + 8000$$

$$= \frac{155000}{100000}$$

$$CPI = 1.55$$

$$\text{Execution time} = C/f$$

$$T = \frac{155000}{40 \times 10^6} = 3.875 \text{ ms}$$