UNITED NATIONS EDUCATIONAL, SCIENTIFIC AND CULTURAL ORGANIZATION
INTERNATIONAL CENTRE FOR THEORETICAL PHYSICS
I.C.T.P., P.O. BOX 586, 34100 TRIESTE, ITALY, CABLE CENTRATOM TRIESTE

The United Nations University

UNITED NATIONS INDUSTRIAL DEVELOPMENT ORGANIZATION
INTERNATIONAL CENTRE FOR SCIENCE AND HIGH TECHNOLOGY
c/o INTERNATIONAL CENTRE FOR THEORETICAL PHYSICS 34100 TRIESTE (ITALY) VIA GRIGNANO, 9 (ADRIATICO PALACE) P.O. BOX 586 TELEPHONE 040-224572 TELEFAX 040-224573 TELEX 460449 APH I

SMR/772 - 20

INTERNATIONAL WORKSHOP ON PARALLEL PROCESSING AND ITS APPLICATIONS IN PHYSICS, CHEMISTRY AND MATERIAL SCIENCE
5 - 23 September 1994

SCIENTIFIC VISUALISATION & COMPUTING WITH PARALLEL PROCESSING SYSTEM

Prabhakar DHEKNE
Bhabha Atomic Research Centre
Computer Division
Trombay
400 085 Bombay
INDIA

These are preliminary lecture notes, intended only for distribution to participants.

BHABHA ATOMIC RESEARCH CENTRE
TROMBAY, BOMBAY 400 085

Scientific Visualisation
Computing

With

BARC PARALLEL PROCESSING SYSTEM

By

P. S. DHEKNE

Officer-In-charge

Computer Centre

COMPUTER DIVISION, BARC.

NEED FOR SUPERCOMPUTING

Scientific computing particularly simulation and modelling needs high speed number cruching capability Computational fluid dynamics, fast fourier transforms, Monto Carlo simulation, matrix multiplications etc. are some of the algorithms which require teraflops speed to obtain a relevant answer in a resonable time period.

Some scientific problems are abandoned for lack of suitable computing power. A solution of quark field theory of nuclear forces quantum chromo dyanamics orQCD would take 1 year to solve on a Supercomputer GF11 having 11 GFlops computing speed.

The $ 9 million Cray YMP for example can blast through 1·3 billion calculations per second.

BUT IS THAT ENOUGH POWER?

NO - - - - - - NO - - - - - - NO

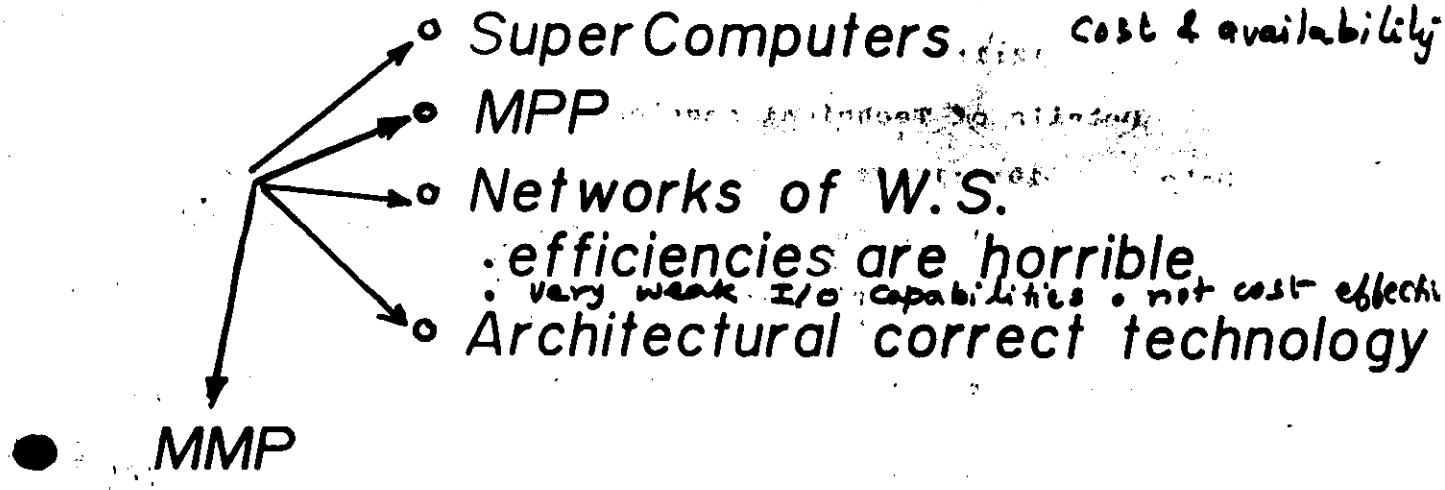
APPLICATIONS OF HIGH PERFORMANCE COMPUTING IN INDUSTRY

- COMPUTATIONAL FLUID DYNAMICS (CFD) IN AEROSPACE INDUSTRY
- SEISMIC MODELLING IN OIL INDUSTRY
- DRUG DESIGN IN PHARMACEUTICAL INDUSTRY
- WEATHER FORECASTING & CLIMATE MODELLING
- AIR FLOW & IMPACT ANALYSIS IN AUTOMOTIVE INDUSTRY
- IMAGE PROCESSING IN SATELLITE BASED SPACE APPLICATIONS
- INDUSTRIAL ROBOTS

SUPER COMPUTING ALTERNATIVES

- VECTOR PARALLEL PROCESSORS : VPP
  Traditional Cray like systems
  - Cray , NEC, FUJITZU,
- SCALABLE HIGH PERFORMANCE COMPUTING : SHPC
  Today's parallel machines
  - Intel, CM-2, CM-5, NCUBE, Transuper based
- HIGH PERFORMANCE CLUSTER COMPUTING : HPCC
  Network of workstations via Fibre Optics networks
  - HP . SUN . IBM . SGI
- MASSIVELY PARALLEL PROCESSING SYSTEMS: MPP OR TERAFLOP MACHINES
  - Intel, CM-5 , CRAY, CONUEX

  Which is architectrally correct technology ?



* Very specialised

* Still very expensive . more than even cray
  . Not able to match I/O performance

* People talk about 'MPP's are still small

* Problem size should be large

* I/O bandwidth is not adequate

* Unclear about which design will survive ?

* Predicted growth rate is not enough to sustain so many players ? & R & D expenditure .

OBSERVATIONS

- ▲ Majority of scientific/engineering appln. use more cpu time than I/O

- ▲ Large cpu bound calculations invariably involve loops that are executed several times

- ▲ Most cpu time is consumed in this loop

- ▲ Most scientific programms are written in Fortran

- Δ Most of these algorithms have built in parallelism

* The CMOS microprocessors has been achieving a 50 to 60% annual growth rate.

* Traditional Super Computer progress is 5 to 10% every year.

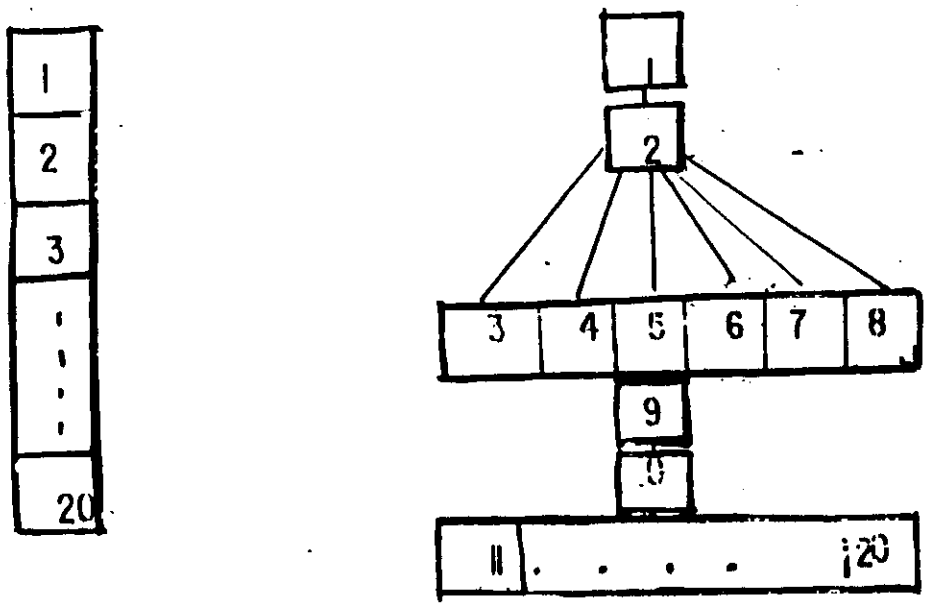
* Due to this fundamental difference in growth rate Microprocessors are overtaking SuperComputers in terms of raw computing power and cost.

UNLIKE SUPER COMPUTERS PARALLEL PROCESSORS

- ▷ COST MUCH LESS
- ▷ TAKES UP CONSIDERABLY LESS SPACE
- ▷ CONSUMES LESS POWER
- ▷ NO DEMAND FOR SPECIAL COOLING ARRANGEMENT
- ▷ PROVIDES SCALABLE PERFORMANCE
- ▷ CAN BE BUILT USING COMMERCIALLY OFF THE SHELF TECHNOLOGY (COTS)

PARALLEL PROCESSING?

* Subdividing a problem into independent tasks and performing them simultaneously allowing required information to be shared between them



* A problem subdivided in 20 tasks takes 20 units of time.

* Same problem has 3 to 8 tasks absolutely independent of each other- similarly tasks 11 to 20.

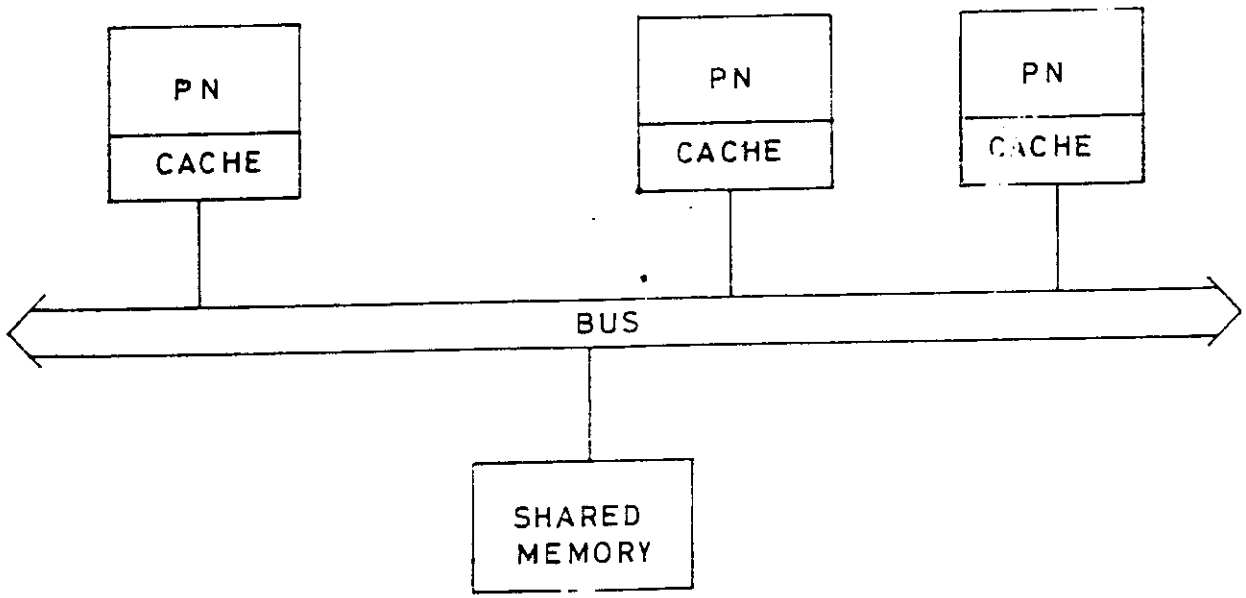
* In a single CPU, it takes 20 time units.

* In 10 identical CPUs, it takes 6 units of time.

If tasks 1,2,9 and 10 were also independent, it would have taken only 2 time units.

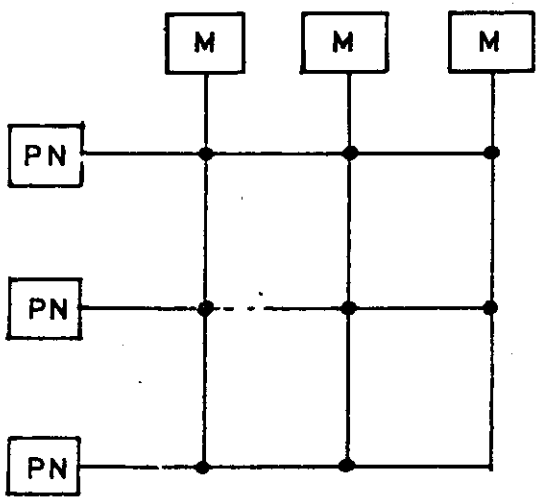
Main Components

- Architecture
  - Simple
  - Scalable
  - Processor Independent
- Inter Connecting Network
  - Scalable band width
  - Architecture independent
  - cost Effective
- Parallel Software Environment
  - User friendly
  - Portable
  - Debugging tools

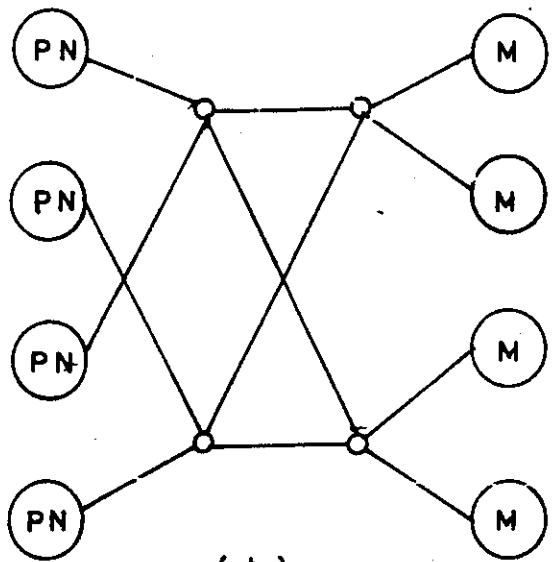


PN : PROCESSING NODE

FIG. 1 SHARED MEMORY BUS ARCHITECTURE

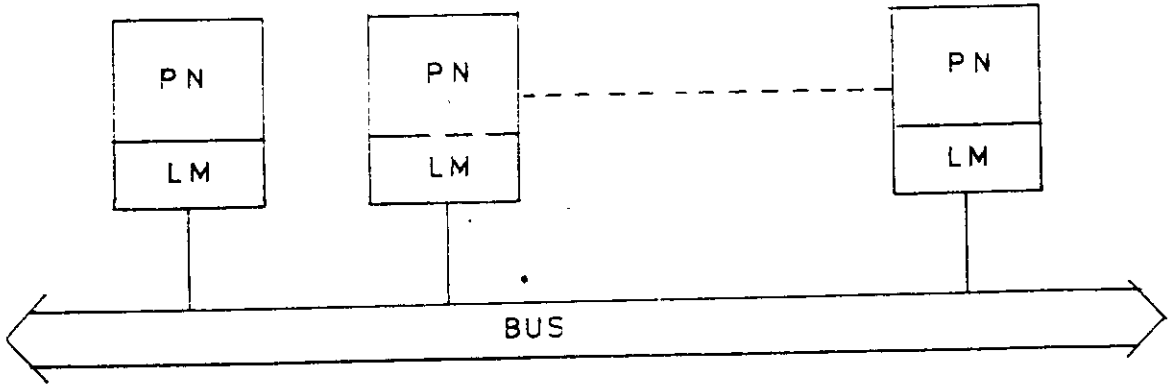


(a)
CROSS BAR SWITCH



(b)
MULTI STAGE SWITCH

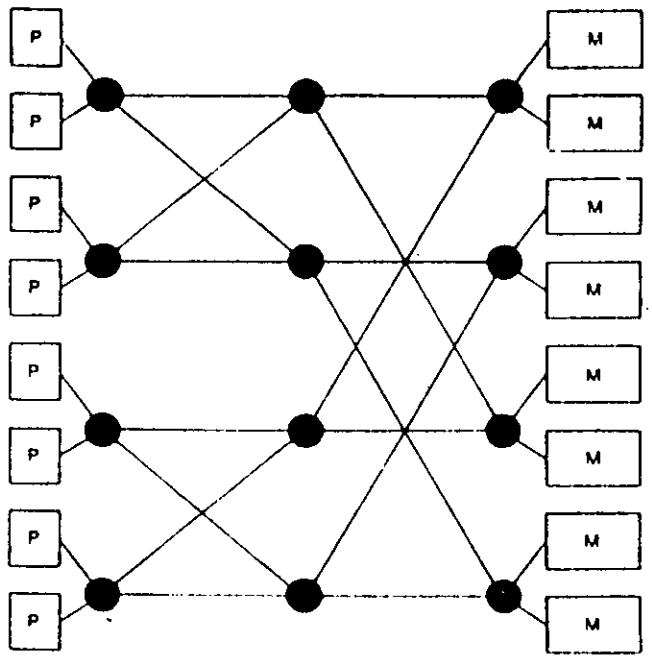
FIG.3 TIGHTLY COUPLED MEMORY



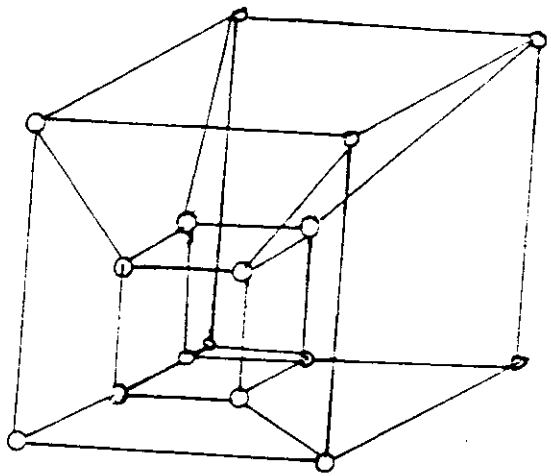
PN : PROCESSING NODE

LM : LOCAL MEMORY

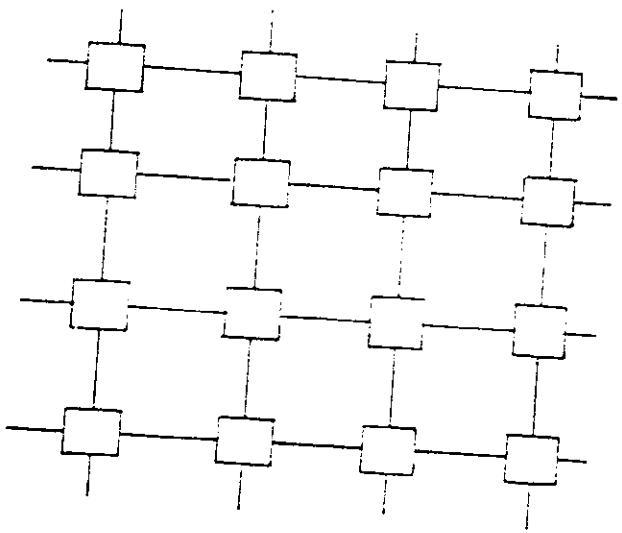
FIG.2 DISTRIBUTED MEMORY BUS ARCHITECTURE



Butterfly computers from BBN Advanced Computers are interconnect parallel machines in which multiple processors, P, are connected to multiple memory modules, M. The interconnect hardware is a collection of VLSI switching nodes that provide a path from every processor to every other processor and from the processors to all of the memory modules.



(a)



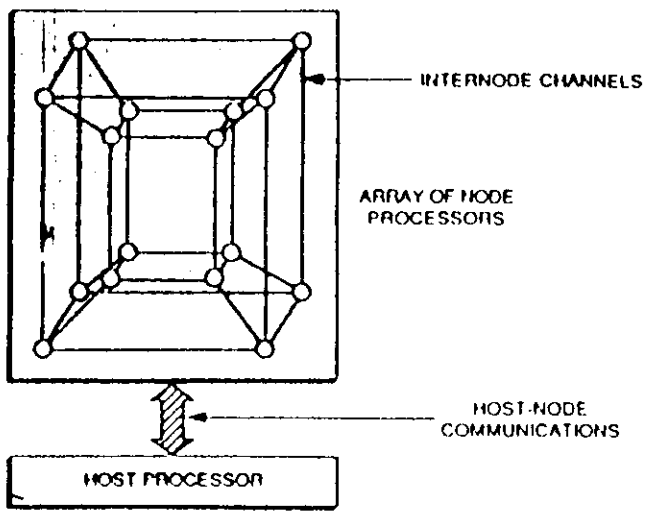
(b) MESH

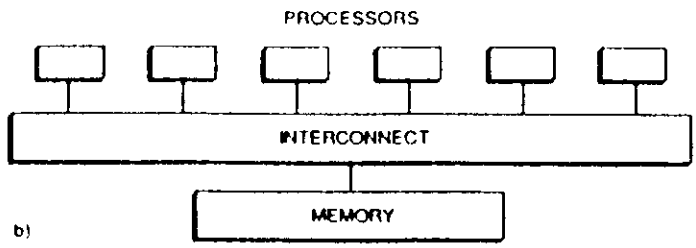
4-D HYPER CUBE

○ : PN WITH LOCAL MEMORY

□ : PN WITH LOCAL MEMORY

FIG. 4 LOOSELY COUPLED MEMORY





Host-array parallel machines have communications links between array processors, such as nodes in a hypercube, and the host processor (a). Each node processor has its own memory. Coding and allocation of computing tasks to the array processors is generally performed by the user for each application. Interconnect parallel machines, or the other hand, have a common, global memory (b). Parallel processors are connected to the global memory through common bus or through a switching array designed to relieve bus congestion.

MULTICOMPUTER NETWORKS

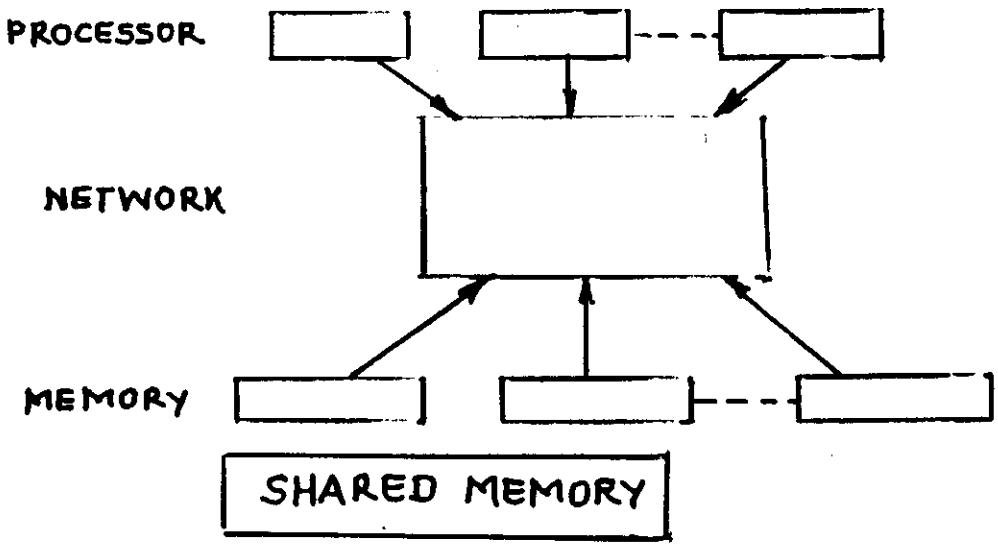
- HYPERCUBE-- LOW DIAMETER
  --EFFECTIVE USAGE OF LINKS.

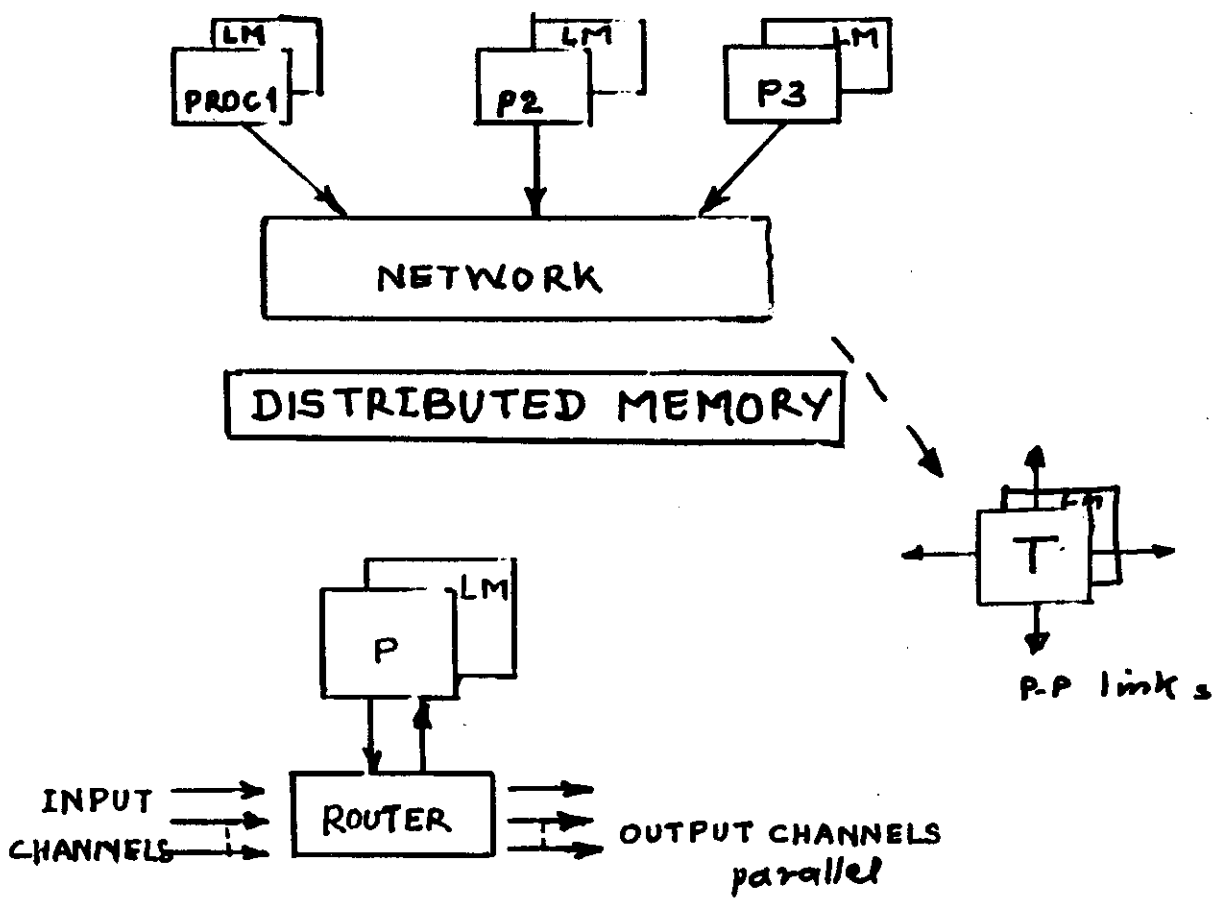
  DRAWBACKS
  - LINK PER NODE HAS TO INCREASE AS DIA. OF THE HYPERCUBE INCREASES.
  - MESSAGE HOPS INCREASE WITH LARGE NETWORK.
  - SUITABLE MAINLY FOR SERIAL LINKS.
  - SOFTWARE HAS TO BE REWRITTEN FOR CHANGE OF HYPERCUBE DIMENTION.

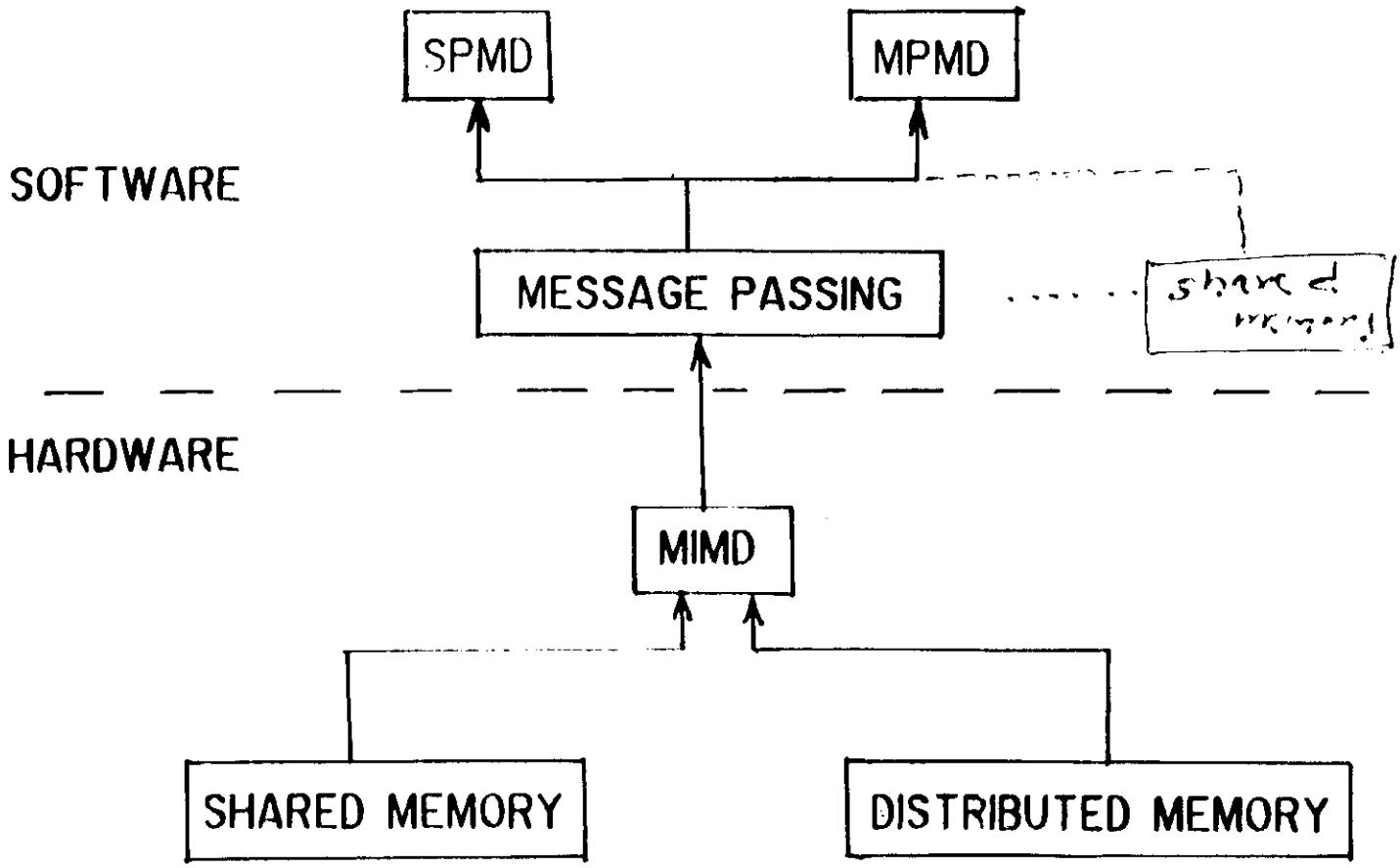
- 2-D MESH WITH WARMHOLE ROUTING MOST PREVELENT NOW & APPEARED AS ALTERNATIVE TO HYPERCUBE.--
  --WIDER CHANNELS.
  --HIGH COMM. RATES.
  --SUITABLE FOR VLSI.
  --SIMPLER HARDWARE.

  WHICH OUT WEIGH ADVANTAGE OF HYPERCUBE.
  --ROUTER PROP. DESIGN
  --NOT YET ACCEPTED STANDARD.

MULTICOMPUTER NETWORKS







- BUS
- CROSS BAR SWITCH
- LATTICE

- BUS
- HYPERCUBE
- MULTICUBE MESH

FACTORS USERS' SHOULD BE AWARE OF

- MICROPROCESSOR USED
- PROCESSOR BOARD DESIGN
- INTERCONNECTING NETWORK
- ARCHITECTURE
- SOFTWARE ENVIRONMENT
  - OPERATING SYSTEM
  - USER INTERFACE
  - PARALLEL TOOLS & DEBUGGING AIDS
  - APPLICATIONS
- OTHER SUPPORT HARDWARE & SOFTWARE
- OVERALL PERFORMANCE
- EASE OF UPGRADABILITY (SCALABILITY)/Add on cost
- COST

BARC'S APPROACH

- MESSAGE PASSING DISTRIBUTED MEMORY
  - HIGHLY SCALABLE
  - EASY TO DEVELOP
  - MOST POPULAR
  - CAN BE BUILT FROM OFF-THE-SHELF COMPONENTS
- TOPOLOGY
  - BEST INTERCONNECTION SCHEME AMONG DIFFURENT TOPOLOGIES.
  - SCALABLE IN TERMS OF PROCESSING POWER AND I/O BANDWIDTH.
  - LOW LATENCY/HIGH BANDWIDTH.
- PERFORMANCE
  - ATLEAST > 200 MFLOPS
- METHOD & TECHNOLOGY —
  - USE MAX.POSSIBLE OPEN COMPONENTS.
  - MINIMUM MODIFICATIONS FROM THE CONVENTIONAL TECHNOLOGY
  - USAGE OF CLUSTER APPROCH
    --EASY TO BUILD UP & REGGEDISE
  - USE OF LARGER RESOURCES MEMORY/CPU
  - USAGE OF INDUSTRY STANDARD BUS CONNECTION FOR P-P COMMUNICATION.

THREE COMPONENT SELECTION

I) ARCHITECHURE - 2-D MESH MOST PREVENT & EFFICIENT-COUNTERMANDED BUS PROBLEMS THREE WAYS.

- DISTRIBUTED MEMORY (LOCAL), MESSAGE PASSING, BUS.
  - ▲ REDUCES BUS TRAFFIC
  - ▲ NO CACHE COHERENCE PROBLEM
- USAGE OF LIMITED NUMBER OF NODES PER BUS.
  - ▲ CLUSTER APPROACH
- USE OF MULTIPLE BUSES TO SCALE COMMUNICATION BANDWIDTH.

II) PARALLEL SOFTWARE ENVIRONMENT

- UNIX SVR 5.4
- ANSI C & FORTRAN 77
- SIMPLE MONITOR PROG. ON NODES
- SEND/RECEIVE CALLS
- SIMULATOR

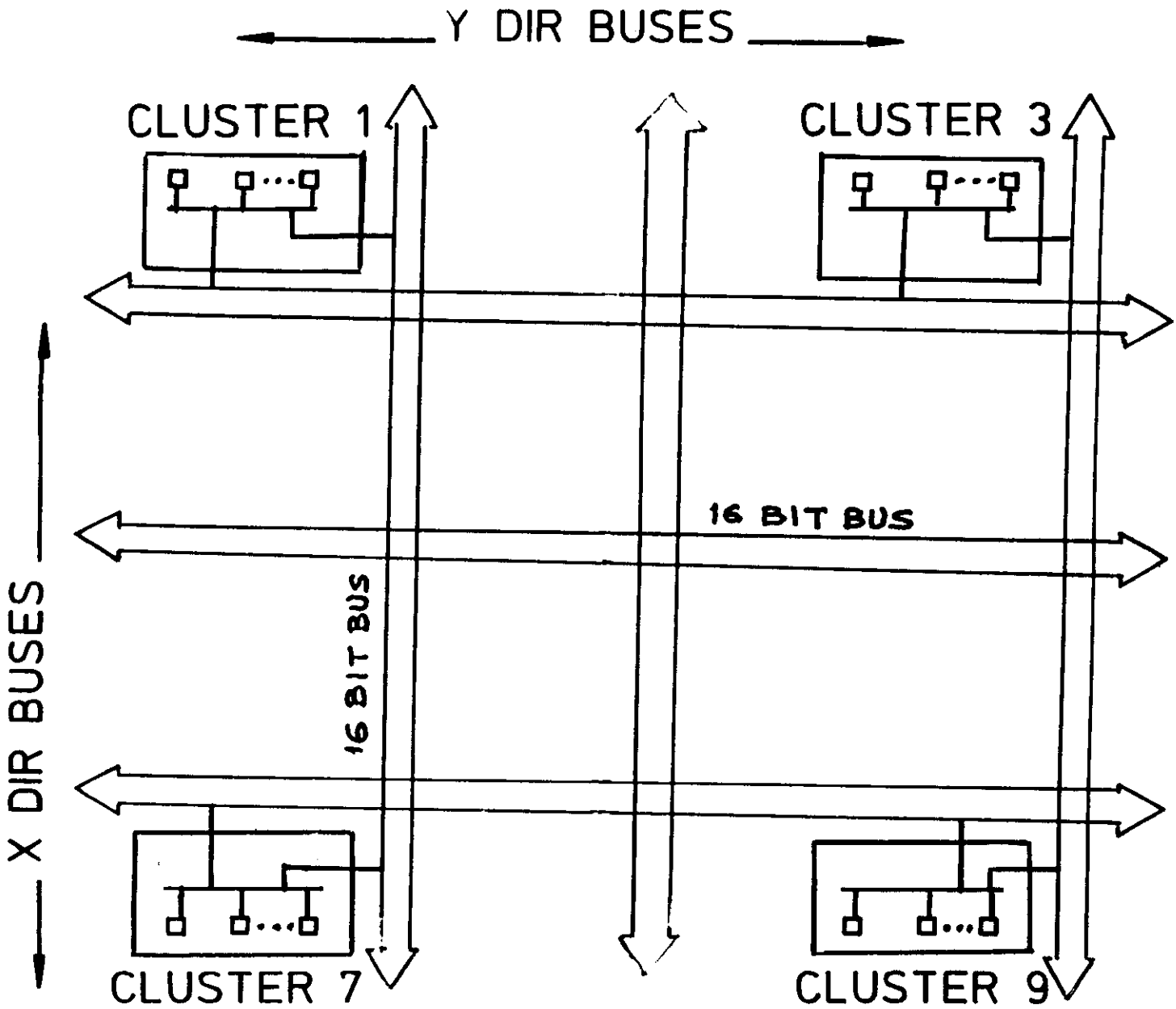
III) NODE PROCESSOR

- USE OF HIGHEST PERFORMANCE MICROPROCESSOR (FROM INTEL)
- I E E E BACK PLANE BUS FOR PROCESSOR INDEPENDENCE
- USE OF INDUSTRY STANDARD PERIPHERAL BUS FOR CLUSTER COMMINICATION

PROCESSOR BOARDS ON ANUPAM

S.NO.	TYPE	PEAK PERFORMANCE		SUSTAIN-MFLOPS	
		SINGLE NODE	32 NODE	SINGLE NODE	32 NODE
1	i860/XR @40Mhz	80Mflops	2·56G	8Mflops	200
2	i860/XP @50Mhz (CPU upgrade)	100Mflops	3·2G	11Mflops	275
3	i860/XP @50Mhz Single slot with 128KB SRAM	100Mflops	3·2G	14Mflops	350

BPPS-64 NODE SCHEME



DELAY : ONLY ONE HOP

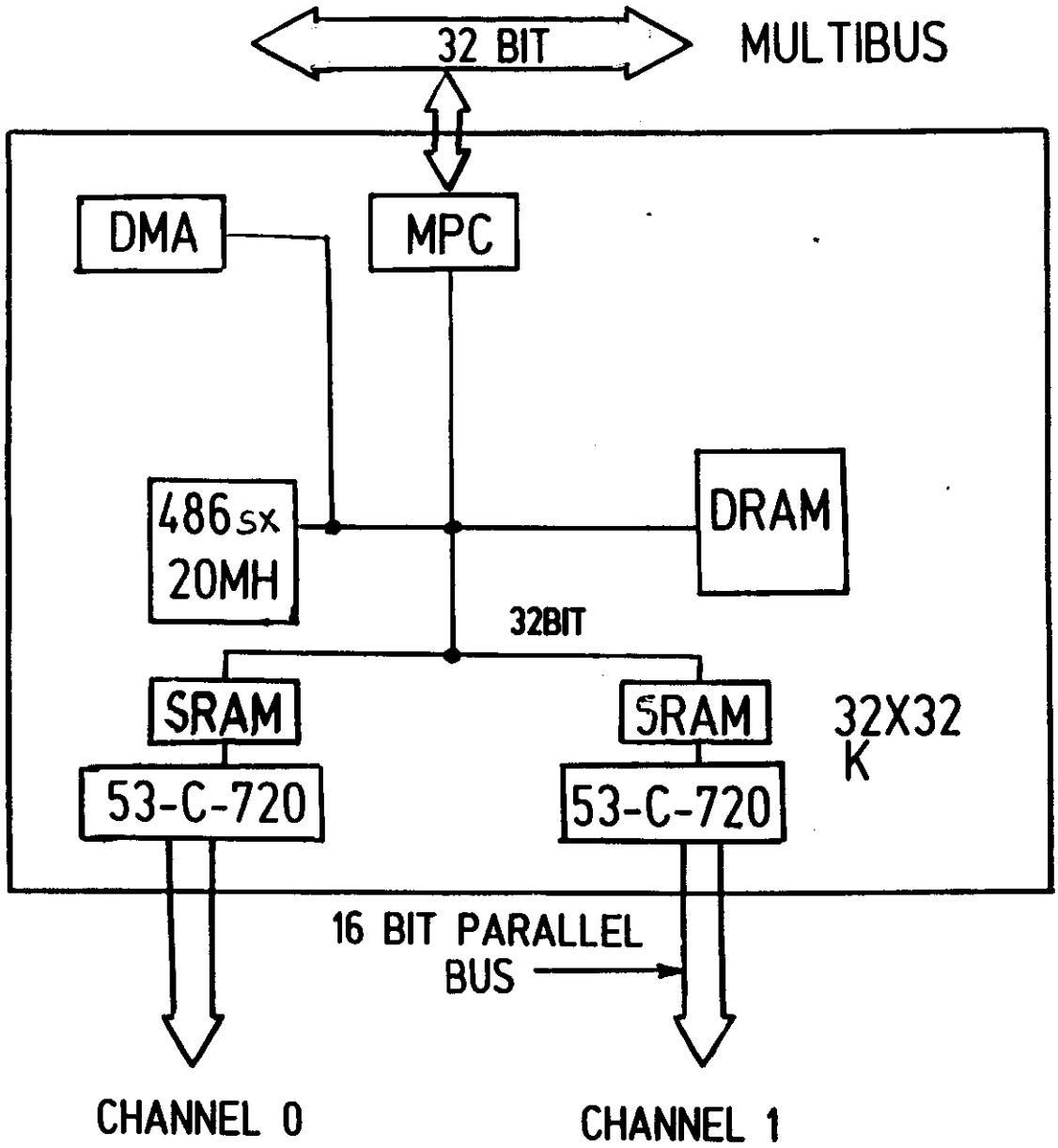
CLUSTER : 8 NODE MULTIBUS CRATE.

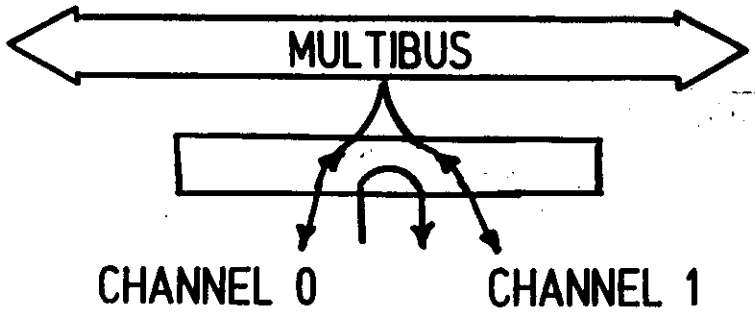
COMMUNICATION : 40MB/Sec.(Peak) on Multibus within cluster

20MB/Sec.(Peak) Cluster to cluster.

ADD ON COST PER CLUSTER - Cost of crate controller and

CLUSTER CONTROLLER

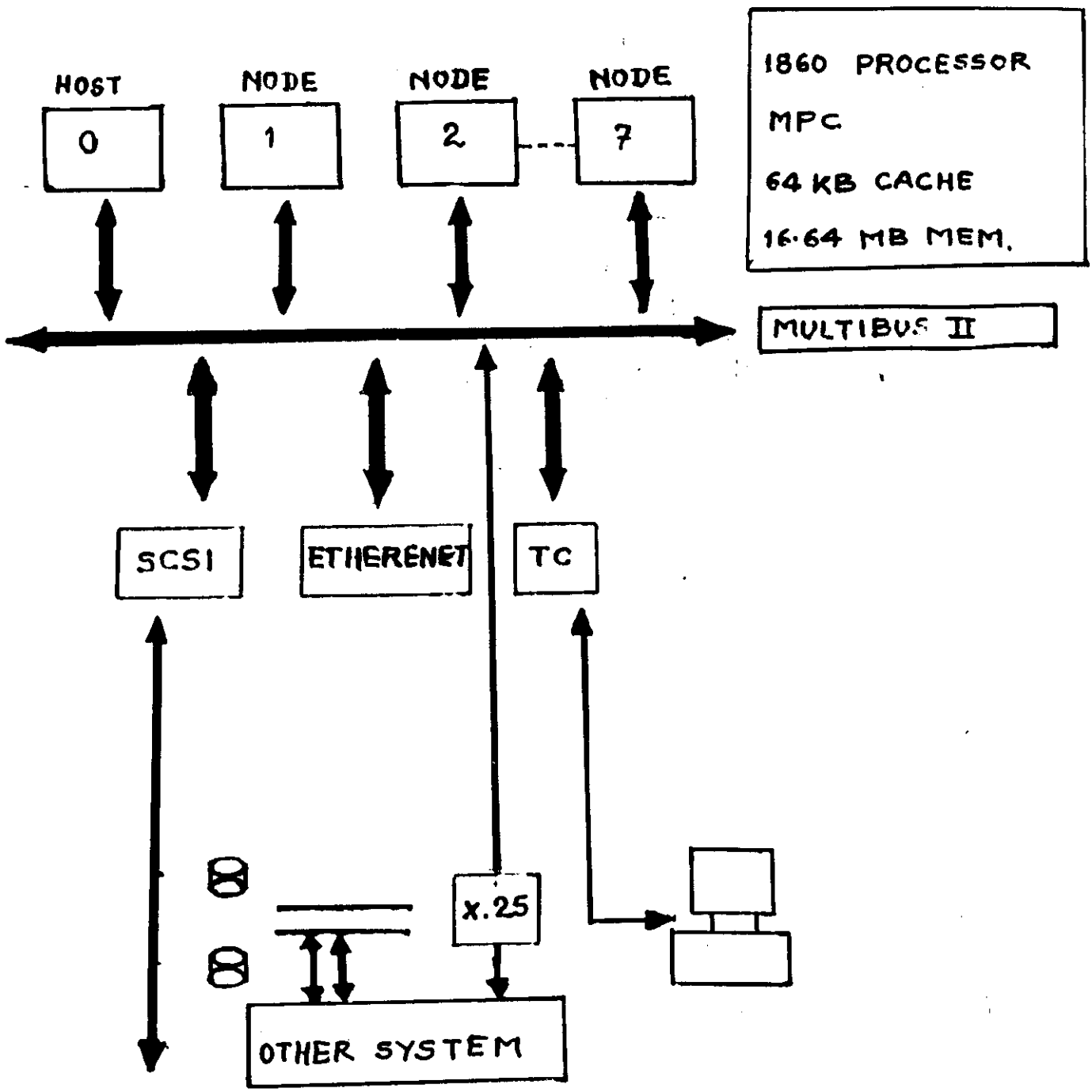




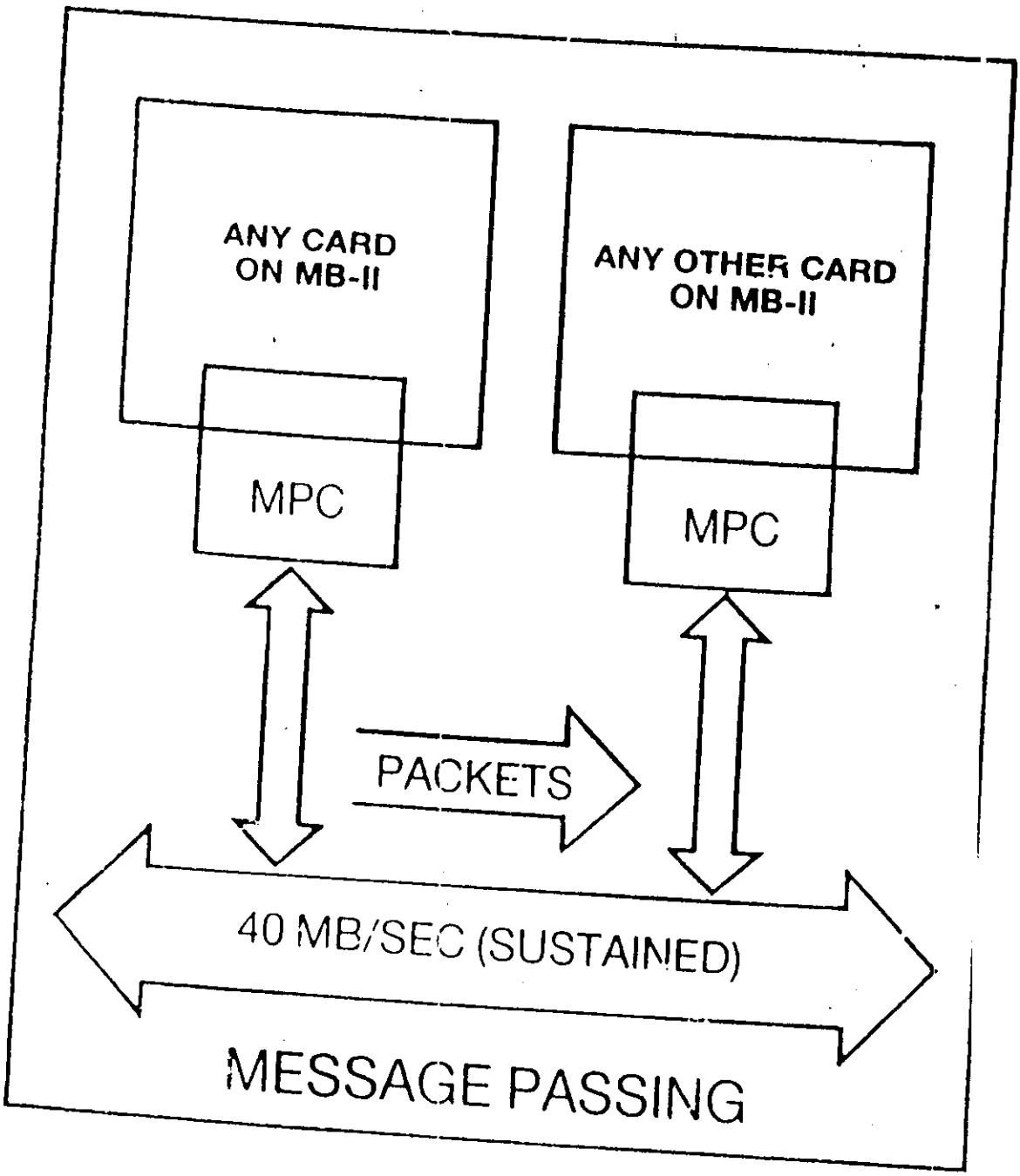
WHY WSCSI

- INDUSTRIES STANDARD PERIPHERAL BUS WITH FLAT CABLE.
- VLSI SUPPORT 53 C720. (80MHz)
- SCRIPT LANGUAGE SUPPORT.
- HIGH BAND WIDTH,20MBYTE/Sec, MAX POSSIBLE TRANSFERRATE, FLAT CABLE.
- SYNCHRONOUS,MESSAGE PASSING FEATURE.
- 16/32 BIT PARALLEL BUS.
- SPEED HIGHER THAN EVEN FDDI.
- COSTS MUCH LESS.

SINGLE CLUSTER OF ANUPAM

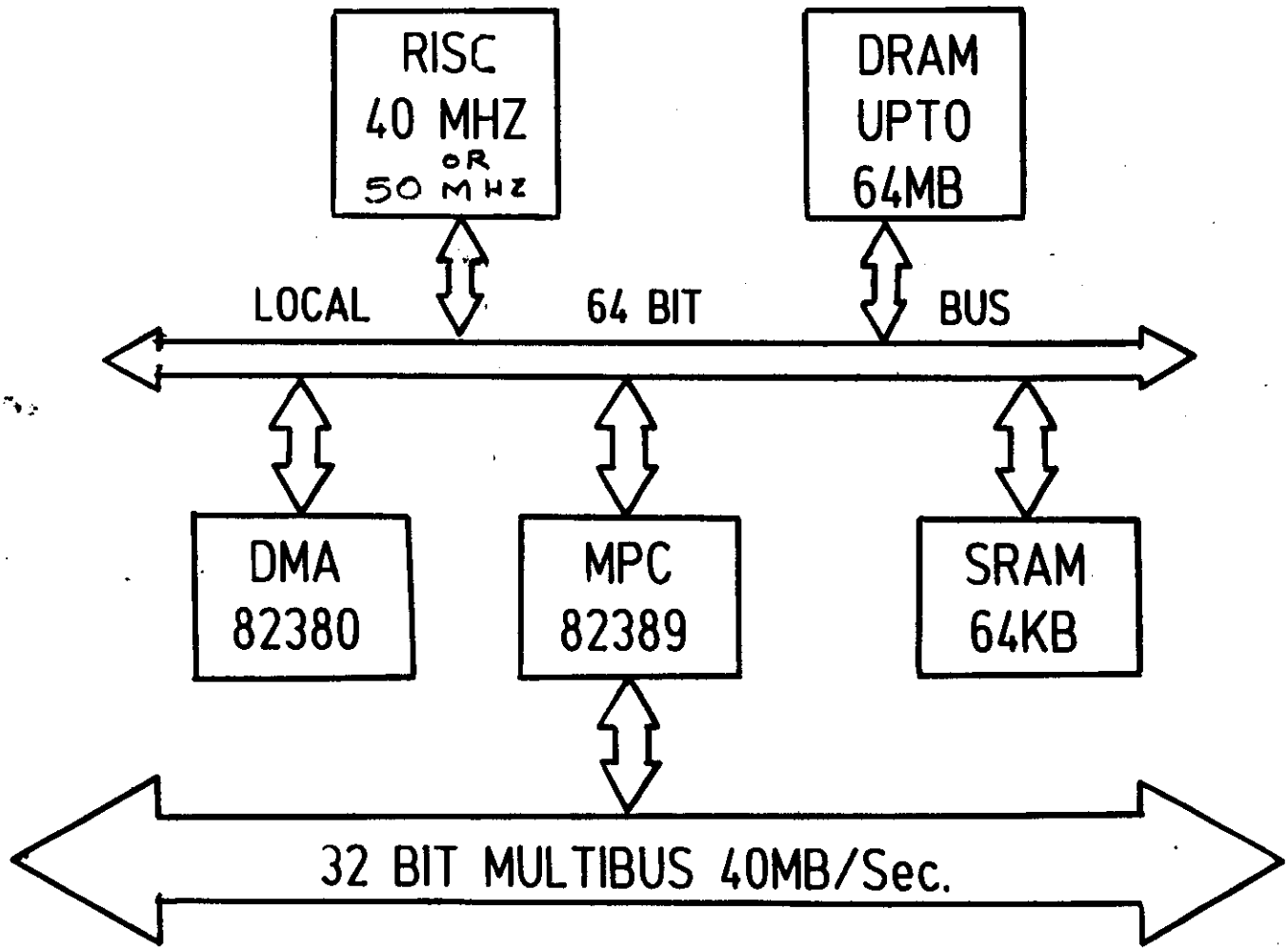


Why MB II ?



- High bandwidth
- power on reconfiguration
- message passing feature.
- possibility of multiple o.s. support
- VLSI, MPC solution
- synchronous bus
- Today's intelligent adaptor boards with DRAM

PROCESSOR BOARD



- DMA--DIRECT MEMORY ACCESS CONTROLLER.
- MPC--MESSAGE PASSING CONTROLLER.
- SRAM--HIGHSPEED STATIC MEMORY.
- DRAM--DYNAMIC MEMORY (MAIN MEMORY)
- RISC--REDUCED INSTRUCTION SET COMPUTER.

JOB UNDER DEVELOPEMENT FOR FOR 32 NODES

- SELF ORGANISED CRITICALLY in FRACTALS
  BARC
  - * CURRENTLY RUNNING ON 16 NODES

- ASTRO PHYSICS GAMMA RAY SIMULATIONS
  BARC
  - * CURRENTLY RUNNING ON 8 NODES

- WEATHER PREDICTION T-80 AND
  SSIOC Codes.
  - * PRESENTLY RUNNING ON 4 NODES

- MOLECULAR MODELLING & SIMULATION (X-PLOR
  BARC
  - * PRESENTLY RUNNING ON ANUPAM
    (SEQ.) HOST

WHY i860

- RISC, 64 bit chip with super scaler architecture
- On chip integer unit, floating point add and multiply, MMU, 8KB instruction cache, 4KB datacache, Vector processing capability & graphics processor
- Gives > 30 MFlops Linpack performance (handcoded)
- Most popular among parallel machines
- Major vendors
  - ▲ Intel — iPsc 64 / Delta / Paragon (128node) (512node) (2048node)
  - ▲ Mercury systems — i860/VME based parallel Race series system
  - ▲ Alacron — Multiple PC based i860 Cards
  - ▲ CSPI — i860 Acceletor
  - ▲ Meiko — i 860 based parallel machine
  - ▲ Alliant — i860 based 2048 nodes parallel m/c

SOFTWARE MODEL

The parallel processing in BPPS is based on communicating sequential processes.

In this model, the underlying computing system is a collection of concurrently executing sequential processes.

The communication between the processes is either through message passing or through shared memory.

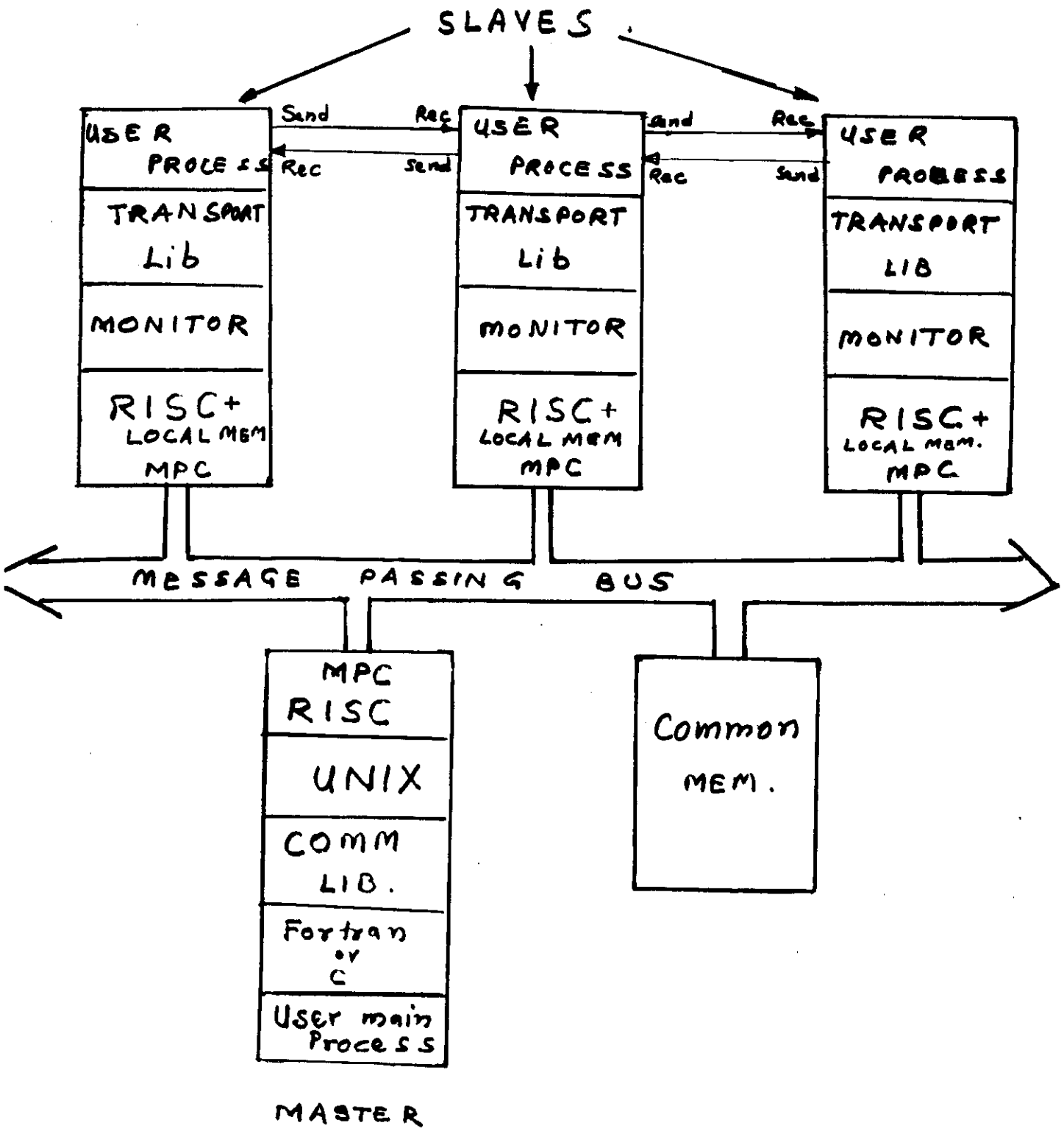
In message passing, the sender will wait until the receiver becomes ready to receive the data.

Each process, executing in a processor has its own code, data and stack.

The Master CPU initiates the execution of processes.

This type of SOFTWARE MODEL is called Single Program Multiple Data SPMD or Single Algorithm Multiple Data SAMD.

BPPS BLOCK DIAGRAM
(SOFTWARE)



- RISC - Reduced Instruction Set Computer
- MPC - Message Passing Controller

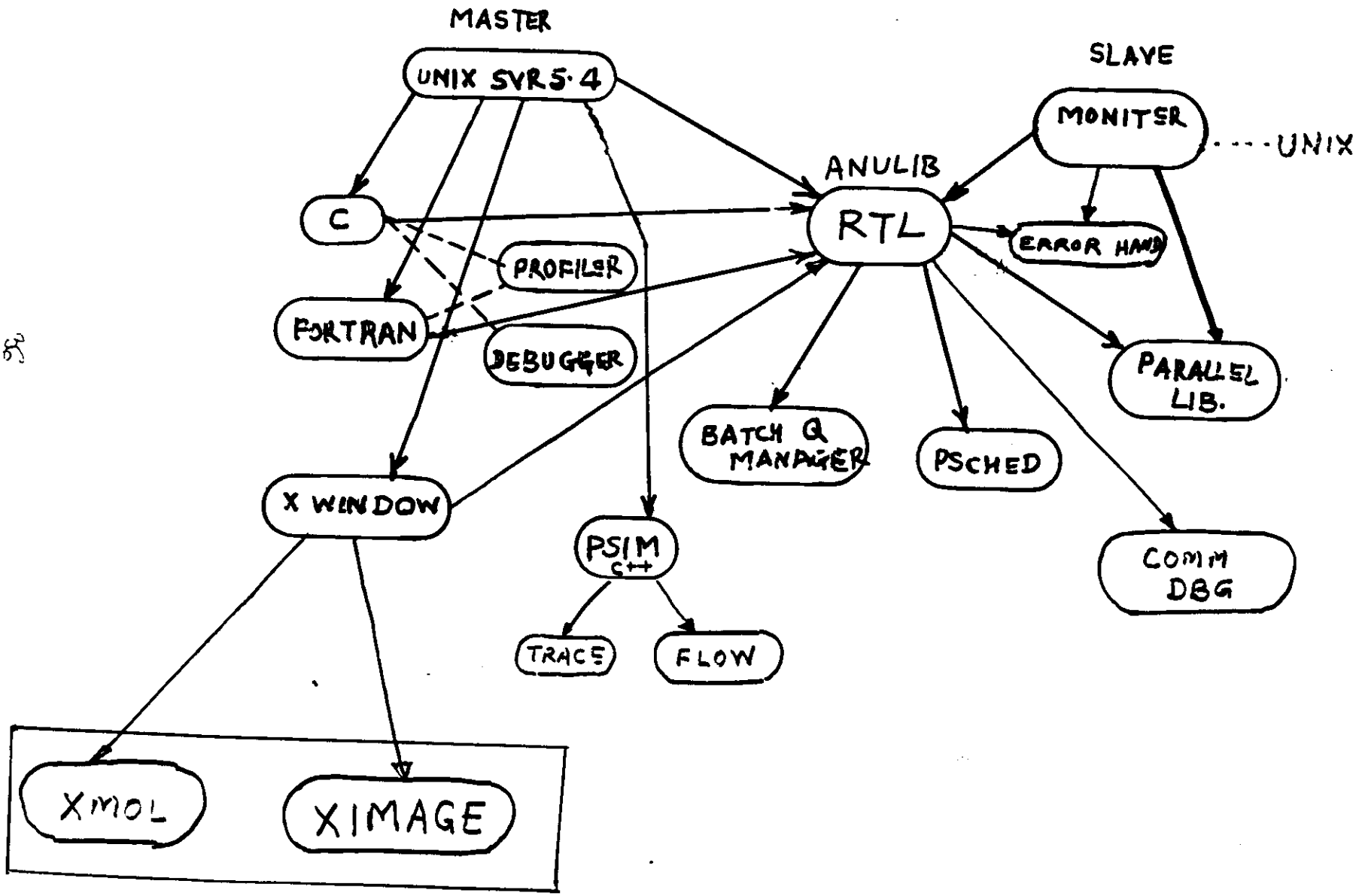
SPECIAL FEATURES OF BPPS

- HIGH PERFORMANCE 64 BIT RISC PROCESSOR
- LARGE MEMORY PER NODE
- PROCESSOR INDEPENDENT ARCHITECTURE
- HIGH SPEED INDUSTRY STANDARD MESSAGE PASSING BACK PLANE COMMUNICATION BUS, PROVIDING BOTH POINT TO POINT AS WELL AS BROADCAST DATA TRRANSMISSION
- VERY SIMPLE & FLEXIBLE COMMUNICATION LIBRARY CONSISTING OF - - -
  - INITIATION CALLS
  - TERMINATION CALLS
  - SEND/RECEIVE CALLS
  - MISCELLANEOUS CALLS
- NO NEED FOR PARALLEL COMPILERS LIKE PAR C OR FORTRAN OR SPECIAL LANGUAGE SUCH AS OCCAM. WRITING PARALLEL PROGRAM INVOLVES ONLY INCLUSION OF A SIMPLE SEND/RECEIVE CALLS IN A SEQUENCIAL PROGRAM.
- SEND/RECEIVE CALLS CAN TRANSFER DATA FOR MULTIDIMENSIONAL ARRAY, THUS AVOIDING DUMMY ARRAY,

PARALLELISATION STEPS

- DECOMPASE THE PROGRAM INTO TWO MODULES MASTER AND SLAVE.
- MASTER MODULE GETS EXECUTED ON HOST PROCESSOR AND INCLUDES INPUT/OUTPUT MODULE, FILE I/O MODULE AND COMPUTATION MODULE. IT SENDS DATA TO THE RESP SLAVES, COMPUTES ITS OWN AND RECEIVES INTERMEDIATE FINAL OUTPUT.
- SLAVE MODULE GATS COPIED ON ALL SLAVES AND GETS EXECUTED ON EACH NODAL PROCESSOR. IT RECEIVES DATA FROM MASTER OR OTHER SLAVES COMPLETES ON IT AND SENDS BACK INTERMEDIATE/ FINAL RESULTS TO MASTER OR OTHER SLAVES.
- COMPILATION AND TESTING USING SIMULATOR ON ANY UNIX MACHINE.
- COMPILATION ON THE TARGET AND EXECUTION ON THE SAME.

ANUPAM SOFTWARE ENVIRONMENT



PSIM Need

① First and foremost use of Simulator is debugging

② PSIM simulates complete parallel environment of ANUPAM on any UNIX based sequencial uniprocessor machine. (UNIX version ४. 3.2 & above)

③ Presently available on any UNIX platform such as SGI, SUN, HP & PC/AT 486 and higher with 32 mb memory

④ User interface identical to ANUPAM

⑤ Currently supports Fortran & C

⑥ C++ interface and S_trace facility available in new version

⑦ S_Trace facility provides stack tracing in case of abrupt exit

Features of PSIM

- Compatible with ANUPAM

  User only need to recompile & link source again, without changing single line of code, on ANUPAM

- Once parallel program is fully debugged it can run on the target ANUPAM machine

- Provides user friendly, menu driven integrated derelopment Environment (IDE) for editing, compiling & executing

- Makes use of extensive UNIX tools such as sdb, vi, SCCS etc, to ease user workload in debugging.

SCLIB

* PARALLEL SUBROUTINE LIBRARY FOR MATHEMATIC, NUMERIC FUNCTIONS
* HAS SIMPLE USER INTERFACE
* USER CAN CALL THESE PARALLEL ROUTINES IN HIS SEQUENTIAL AS WELL AS PARALLEL CODE
* SUPPORTS
  - MATRIX ALGEBRA
  - SOLVING DIFFERENTIAL EQUATIONS
  - FUNCTION EQUATION
  - RANDOM NUMBER GENERATOR
  - LEAST SQUARE FIT
  - LINPACK
  - VECTOR OPERATIONS
  - FINDING ZEROS OF POLYNOMIALS.
* USER SPECIFIC ROUTINE CAN BE EASILY ADDED TO THE LIBRARY

IMAGEPRO

- Image Processing subroutines implimented using parallelisation Technique.
- Has simple user interface.
- User need not do parallelisation. User can insert parallel calls in his sequential code.
- Parallel Routines such as 2-D FFT, Sobel's edge detection, histogram equalisation, high pass/low pass filters, Median filtering, thinning, are available.
- Modular software approach allows user specific function to be added in the library.

COMMON PLATFORM FOR PARALLEL & MULTIPROCESSING

- MULTIPLE SEQUENTIAL JOBS AND MULTIPLE PARALLEL JOBS EXECUTION POSSIBLE
- FEATURE NOT AVAILABLE ON ANY INDIAN PARALLEL MACHINE TODAY
- UNIX RUNS ON MASTER & SLAVES
- EASE OF DEVELOPEMENT OF PARALLEL TOOLS LIKE PARALLEL PROFILER AND PARALLEL DEBUGGER
- SHARED FILE SYSTEM
- AUTOMATIC LOAD BALANCING POSSIBLE

JOBS ON 32 NODE ANUPAM

Efficiency

- VASBI – (CFD code from ADA) — 76%
- PROLSQ – (Molecular Modelling BARC) — 68%
- LMTO – (Electronic structure calculatons calculations BARC) — 70%
- MANALI – (Monte carlo simulation for Reacter Physics BARC) — 54%
- MCNP-code – (Radiation Transport problem BARC) — 80%

FACTORS ABOUT EFFCIENCY

- COMPUTING EFFICIENCY (MFLOPs / NODE)
- NO OF NODES
- START UP TIME
- COMPUTATION TO COMMUNICATION RATIO
- NATURE OF ALGORITHM & ITS INHERENT PARALLELISM
- SIZE OF THE APPLICATION
- COMPUTE, I/O, COMMUNICATION LOAD BALANCE/ DISTRIBUTION

☆ Parallisation Methods

- ◎ Event Parallelisation
- ◎ Geometric Parallelisation
- ◎ Algorithmic Parallelisation

☆ Grain Size

- ◎ Coarse Grain
- ◎ Medium Grain
- ◎ Fine Grain

☆ Problem Size

- ◎ Large Size
- ◎ Medium size
- ◎ small Size

HH

EFFECT OF AMDALH'S LAW

$$T_{ET} = T_{seq} + T_{par}$$

$$T_{par} = T_{startup} + T_{com} + T_{wait}$$

Amdalh's law says

- Sequencial part ultimately decides the overall speed up

$T_{par} \simeq 0$ by dividing $\frac{T_{par}}{n}$ $n = n\phi$ of Pro

ET : Elapsed time

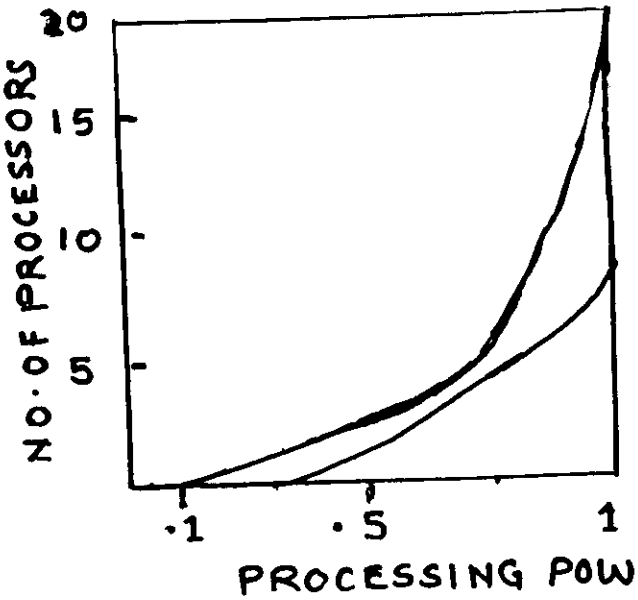
Seq : sequencial elapsed time

par : elapsed time for parallelisable code

Parallel codes

- ◎ Embrassingly Parallel codes
- ◎ Not so highly Parallel codes
- ◎ Poorly parallelisable codes

PARALLELISATION



SPEED UP RATIO

$$Sp = \frac{1}{((1-P)+P/N)}$$

– Processor with higher power
– Processor with lower power

Compute ⇔ Communication

$T_{ES} = T_{ep} + T_{es}$

T_{ES} = Execution time for sequential

Ideally $T_{EP} = \frac{T_{ep}}{N}$

T_{ep} = Execution time for parallelisable code

T_{es} = Execution time for serial code (non parallelisable

T_{EP} = Execution time for parallel code with n processor

$T_{par}(N) = T_p(N) + T_{comm}(N) + T_{wait} + T_{es}$

speed up $S(N) = \frac{T_{ES}}{T_{par}(N)}$

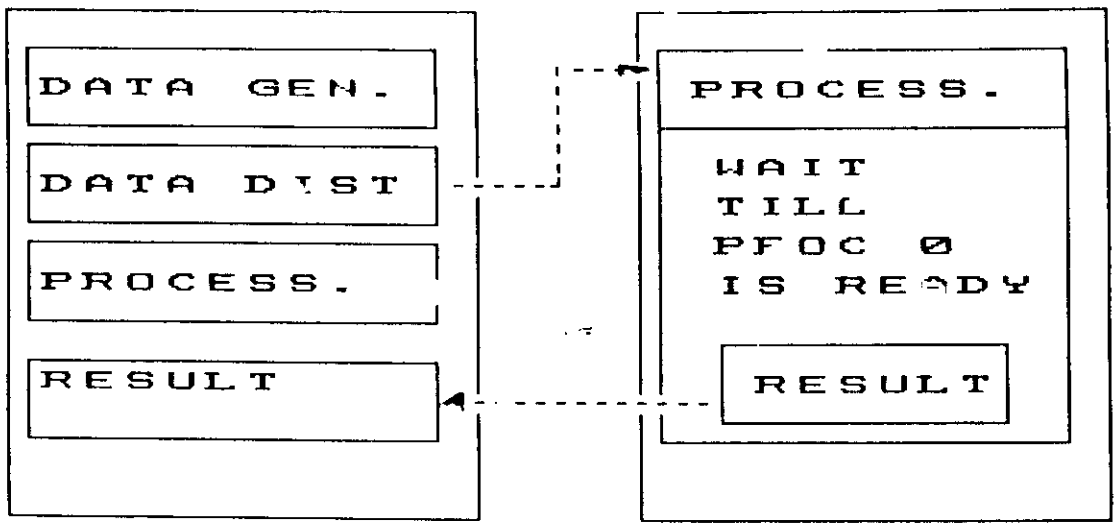
T_{par} = Time for parallel execution

Efficiency $E(n) = \frac{S(N)}{N}$

T_{ES} = Time for sequencial execution

To get $\left(\frac{T_{ep}}{N}\right)$ 'n' low all CPU's should be equally loaded.

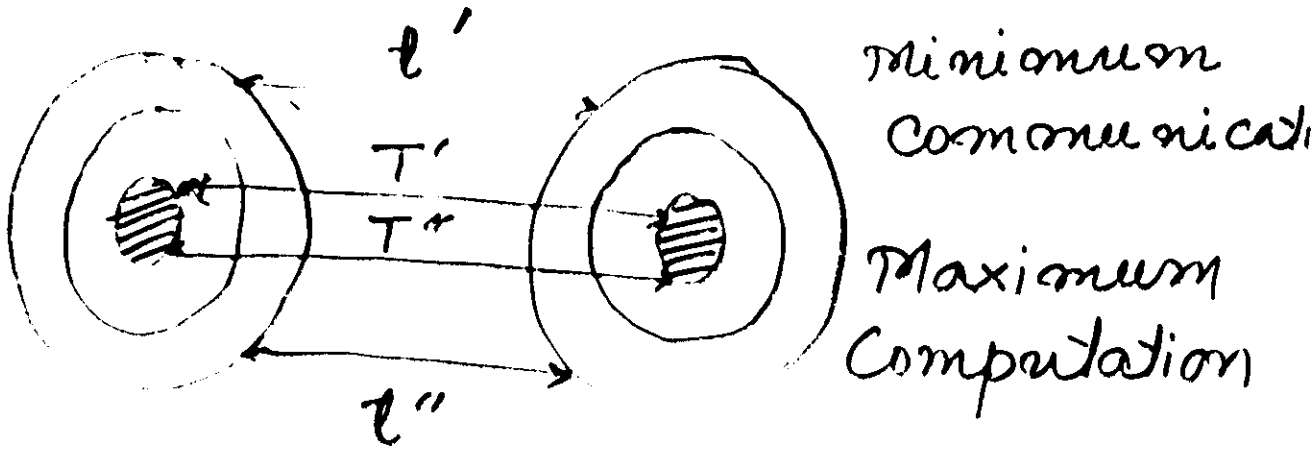
COMMUNICATING SEQUENTIAL PROCESSES



COMMUNICATION BETWEEN PROCESSES
done initially for data
distribtion.
done finally for receiving results.

PROC 1 WAITING TO SYNCHRONISE WITH PROC 0.

EXTRA ACTIVITIES FOR PARALLEL PROGRAMS
- COMMUNICATION BETWEEN PROCESSORS
- WAITING FOR SYNCHRONISATION



PROLSQ ON ANUPAM.

TABLE A
PROBLEM SIZE = X

NO OF PROC.	EXECUTION TIMINGS IN SEC. T PARALLEL	T SEQ.	T TOTAL
1	606·5	35·5	642
2	307·6	35·5	343·1
3	207·4	35·4	242·8
4	157·3	35·4	192·7
5	126·8	35·4	162·2
6	106·8	35·4	142·2
7	92·7	35·4	128·1
8	80·5	37·7	118·2

TABLE BB
PROBLEM SIZE = 4X

1	2421·6	35·5	2457·1
8	309 (305) *	35·7	344·7 (308·6) +
12	240 (223) *	36·4	276·4 (215·6) +
16	219 (195) *	36·4	255·4 (168·9) +

* WITH 32KB DATA BUFFER INSTEAD OF 16KB

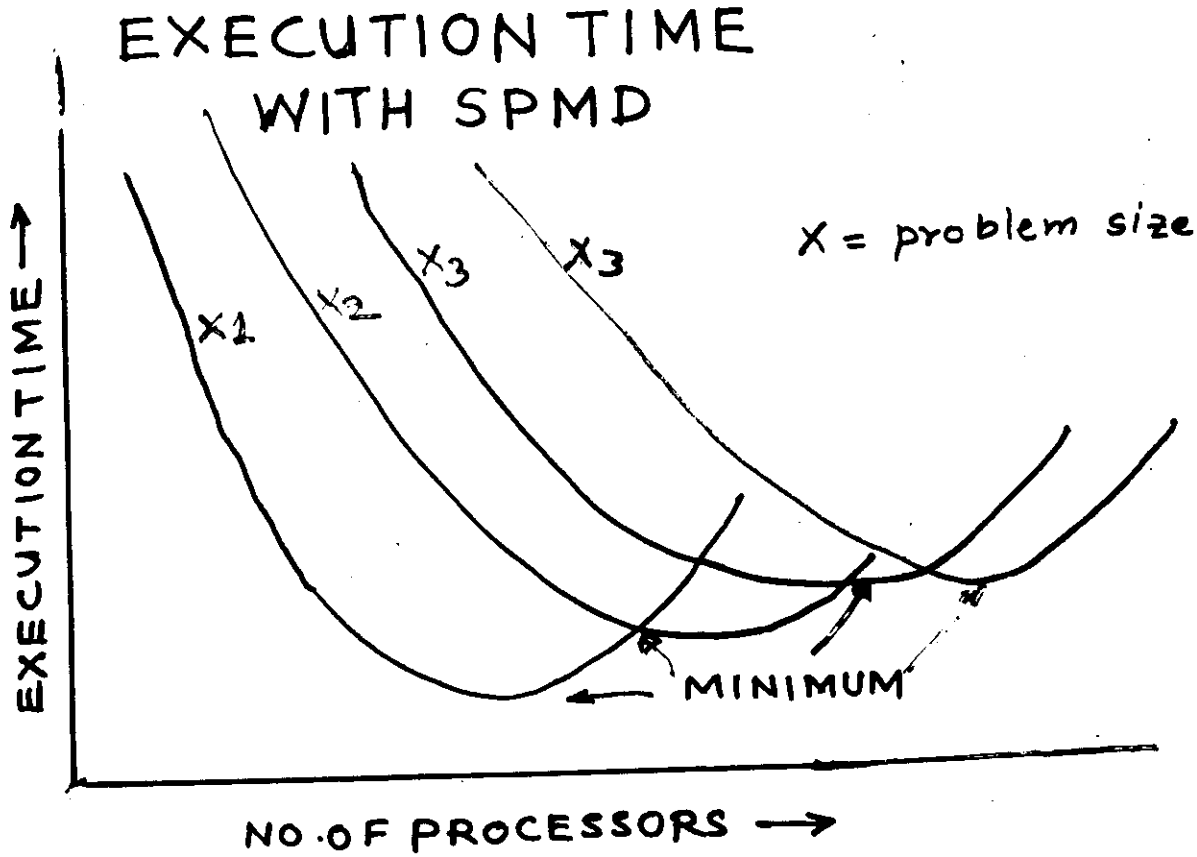
\+ WITH WSCSI BUS TIMINGS

HR

TABLE I

GRAY SCALE IMAGE: 771 X 590 PIXELS

BPPS	NO OF PROC.	MEDIAN FILTERING (3X3) sec.	SOBEL'S OP. (EDGE-DETE.) sec.	2 DFFT sec.	THINNING (20 ITRATIONS) sec.
	1	10·56	4·37	32·76	242·46
	2	5·35	2·29	21·79	121·57
	3	3·62	1·59	18·05	81·02
	4	2·76	1·28	16·25	61·68
	5	2·25	1·04	15·20	49·02
	6	1·95	0·92	14·46	41·52
	7	1·66	0·82	13·89	35·25
	8	1·49	0·79	13·47	31·52
PC/AT486 33 mhz		58·55	33·83	1050	1590·86



o FACTORS

- ARCHITECTURE
- COMMUNICATION SCHEME & SPEED OF DATA Xer
- CPU SPEED / COMMUNICATION SPEED
- LOAD BALANCING
- PROBLEM SIZE & TYPE

— Muler's Law —

OUTSIDE BARC USERS

- IOPB--COMPUTATIONAL PHYSICS
- ISRO,BANGLORE-CFD
- CENTRE FOR ARTIFICIAL INTELLIGENCE AND ROBOTICS-CAIR --IMAGE PROCESSING
- SAC,AHMEDABAD-IMAGE PROCESSING - SAR
- DEPT. OF PHYSICS,POONA UNIVERSITY -ELECTRONIC STRUCTURE CALCULATIOI
- IISC,BANGLORE --CFD -- MOL. MODEL -- Robot
- ADA, BANGLORE - Four groups
- VSSAC, TRIVENDRAM --CFD
- CW & PRS, PUNE-MATH.MODEL FOR POLLUTION MEASUREME
- DRDL and DEFENCE
- I.I.T.--BOMBAY -- PHYSICS CODE -- CFD
- ONGC, DEHARADOON -- SEISMIC
- ADRIN, HYDERABAD - - IMAGE PROCESSING
- HAL, BANGLORE
- WEATHER PREDICTION

TYPE OF APPLICATIONS

* MONTE CARLO SIMULATION FOR PHOTON TRANSPORT AND DISCRETE ORDINATE NEUTRON TRANSPORT.
* ELECTRONIC STRUCTURE CALCULATIONS
* MOLECULAR MODELLING
* MOLECULAR DYNAMICS & SIMULATION
* HIGHLY RESOLVED 3D VERSIONS OF PHYSICS CODE
* SURFACE RECOGNITION IN IMAGE PROCESSING
* RANDOM NUMBER GENERATION
* COMPUTATIONAL FLUID DYNAMICS
* SEISMIC STUDIES
* 2D & 3D FFT'S
* MATRIX MULTIPLICATION
* MULTI GROUP INERGAL TRANSPORT EQUATION

KEY FEATURES OF BPPS

- Modularity —— cluster approach
- Openness —— open backplane bus — Multibus
  Open O.S. — UNIX, SVR 4
  Std. language compilers — Fortran & C
- Compatibility —— 100% UNIX compatible — Fortran, C
- Scalability —— no. of cpu's — performance
  generation — cost
  size problem
- Cost —— unmatched best performance
  >200 MFlops for Rs. 1Crore 25 Lakhs for 32 nodes
- Appln. portability — Accross message passing, SPMD
- Performance — 4·1 times faster than IBM RS 6000/560 ON BPPS 16
  5·4 times faster on BPPS 24
  6·8 times faster on BPPS 32

MulfiComputer Approach For BPPS

- Increased performance throughput
- Increased reliability
- Graceful degradation in the event of failure
- Multibus
  - power on reconfiguration
    (not possible on VME or transputer based systems)
  - high band width
  - new enhancements
  - message passing approach
- VLSI, MPC, solution

SCIETIFIC COMPUTING

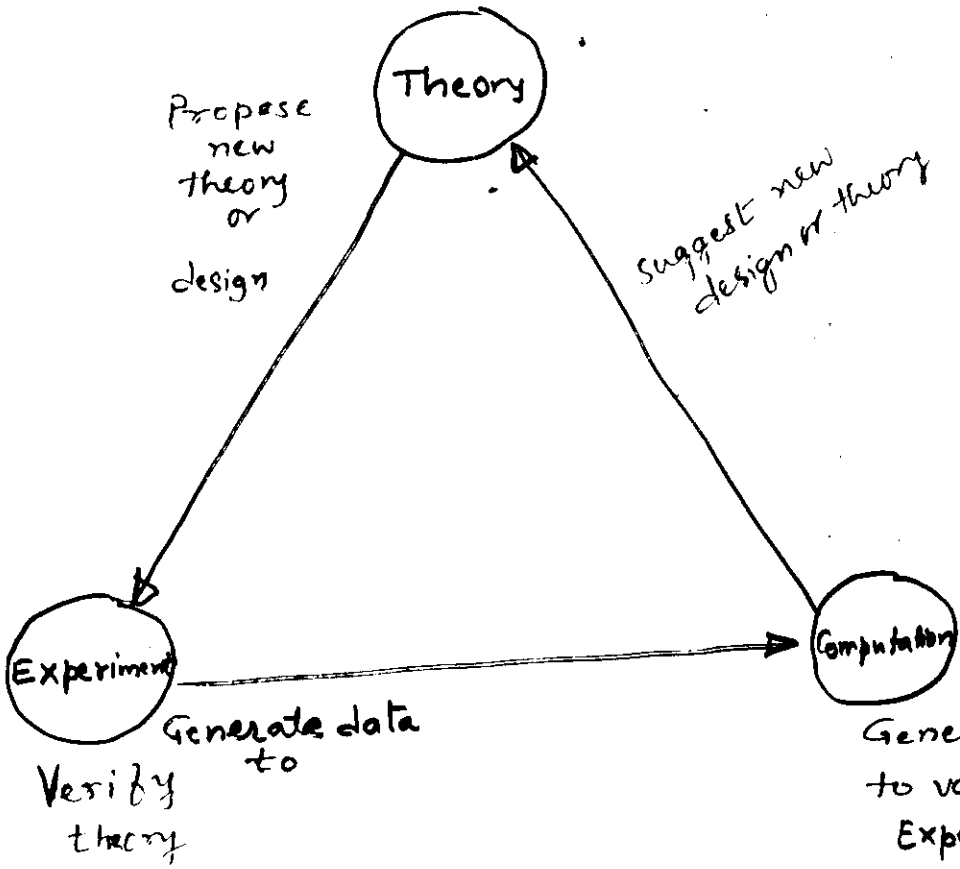
- MOLECULAR MODELLING & SIMULATION
- MOLECULAR DYNAMICS
- FLUID DYNAMICS
- MONTE CARLO SIMULATION
- SPEECH ANALYSIS
- IMAGE PROCESSING
- FINITE ELEMENT ANALYSIS
- FINITE VOLUME METHOD

Emerging Use's of PARALLEL M/C

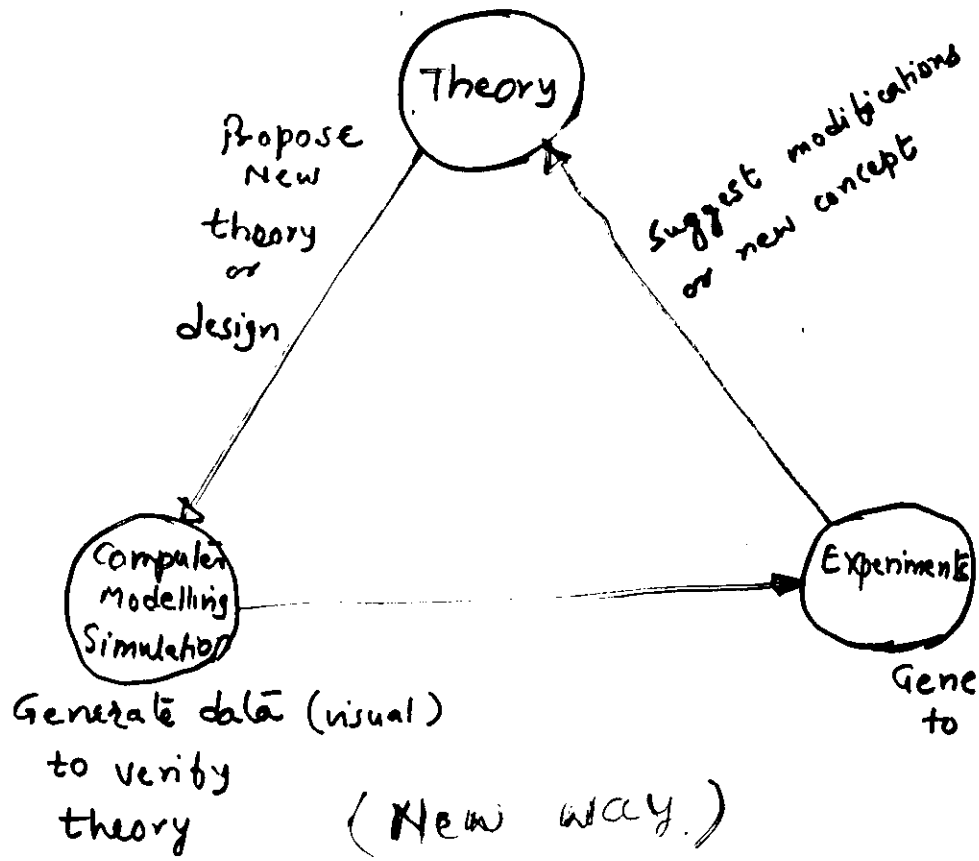
- Modelling and simulation
- Geophysics exploration
- New drug design
- Aerospace design
- Global climate modelling
- Astrophysics
- Robotics
- Image processing
- Medical Imageing
- Critical Real time jobs

Life Cycle of

• Science and Technology



(old way)



(New way)

VISUAL COMPUTING

- Computational methods are being used for simulation & modeling as an alternative to experimental methods

- These techniques require supercomputing power to obtain results in reasonable time

- Also these methods generate huge volume of data which is impossible to analyse using conventional methods

- Computer world today is moving towards object oriented graphics or what is known as "visual computing" since

- Pictorial data appeals more to human mind

- All this requires still higher computational power

- Once niche market now it is becoming common place due to the availability of low cost parallel processor

PARALLEL LINPACK READINGS ON DFPS

Size of Matrix: 100*100 Double Precision

nproc	Time	MFLOPS
1	0.15	4.58
2	0.51	1.34
3	0.40	1.71
4	0.37	1.85
5	0.35	1.96
6	0.36	1.91
7	0.37	1.85
8	0.40	1.71

Size of Matrix: 200*200 Double Precision

nproc	Time	MFLOPS
1	0.90	5.88
2	1.16	4.67
3	1.01	5.16
4	0.91	5.95
5	0.86	6.29
6	0.84	6.44
7	0.80	6.76
8	0.78	6.94

Size of Matrix: 300*300 Double Precision

nproc	Time	MFLOPS	Speed Up	Efficiency
1	2.73	6.66		
2	1.63	11.15	1.67	83.5
3	1.38	13.17	1.97	65.6
4	1.25	14.54	2.18	45.5
5	1.22	14.90	2.24	44.8
6	1.19	15.28	2.29	38.1
7	1.15	15.81	2.37	33.8
8	1.13	16.09	2.40	30.0

Size of Matrix: 400*400 Double Precision

nproc	Time	MFLOPS	Speed Up	Efficiency
1	6.07	7.08		
2	3.69	12.68	1.64	82.0
3	2.68	16.04	2.26	75.3
4	2.34	18.37	2.60	65.0
5	2.26	19.02	2.68	53.6
6	2.01	21.08	3.00	50.0
7	1.91	22.51	3.17	45.2
8	1.89	22.74	3.21	40.1

Size of Matrix: 500*500 Double Precision

nproc	Time	MFLOPS	Speed up	Efficiency
1	11.36	7.38		
2	6.15	13.63	1.84	92.0
3	4.66	17.99	2.43	81.0
4	3.84	21.83	2.95	73.0
5	3.44	24.37	3.30	66.0
6	3.26	25.71	3.48	58.0
7	3.09	27.13	3.67	52.4
8	2.99	28.04	3.80	47.5

Size of Matrix: 600*600 Double Precision

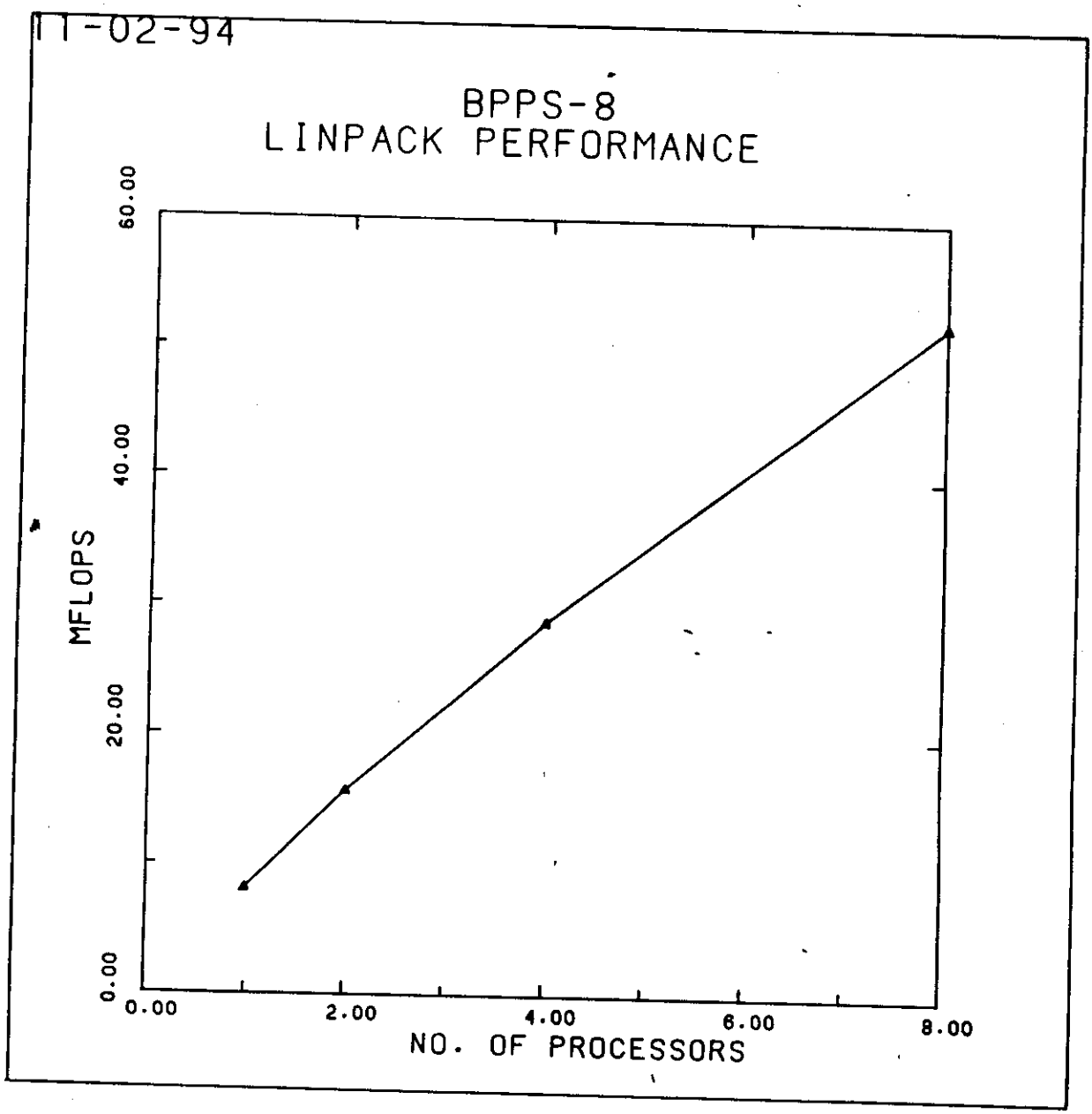
nproc	Time	MFLOPS	Speed Up	Efficiency
1	19.00	7.62		
2	9.45	15.31	2.00	100
3	6.88	21.04	2.76	92.0
4	5.50	26.31	3.45	86.2
5	4.93	29.36	3.85	77.0
6	4.40	32.89	4.31	71.8
7	4.06	35.65	4.67	66.7
8	3.92	36.92	4.84	60.5

Size of Matrix: 700*700 Double Precision

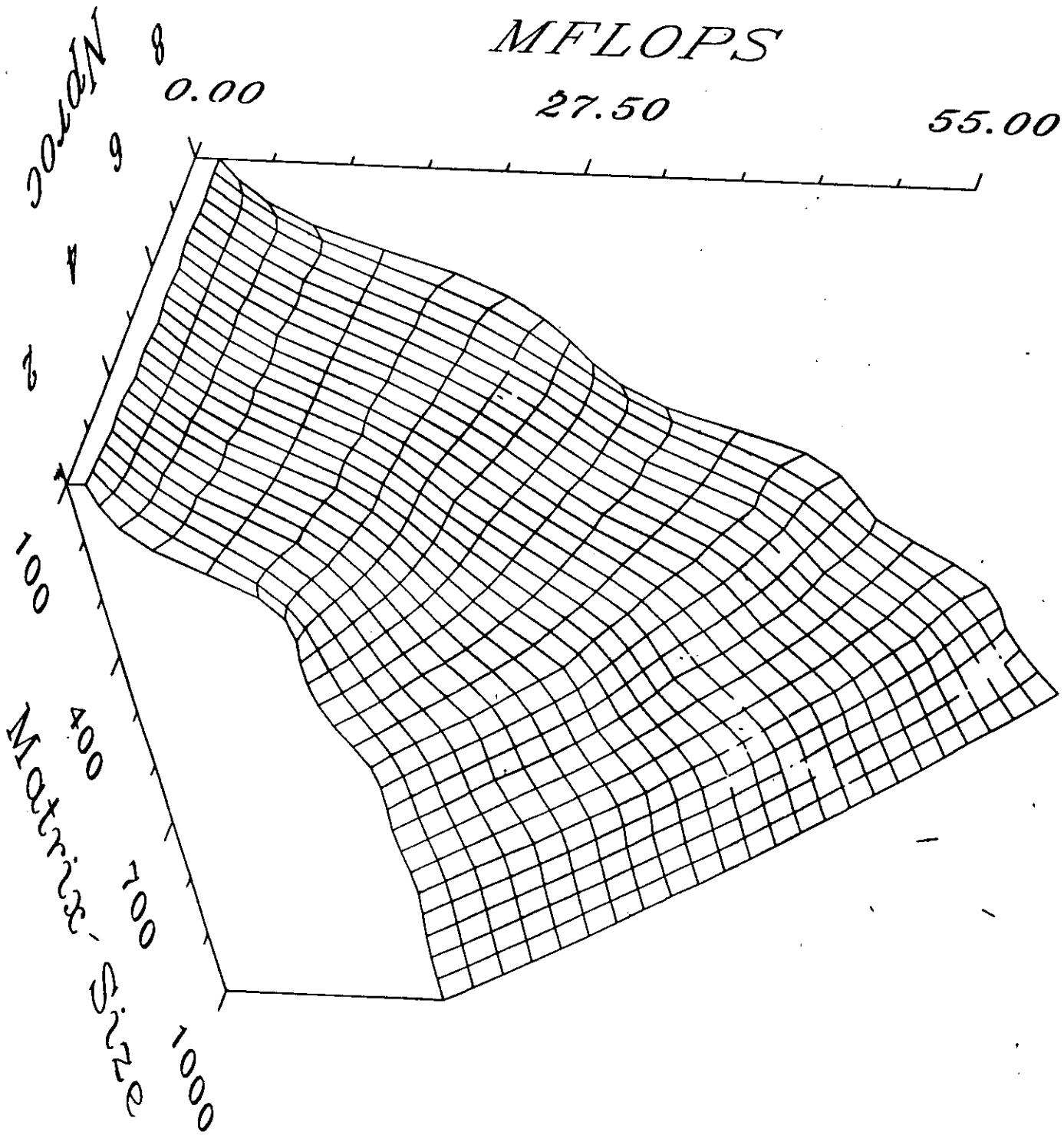
nproc	Time	MFLOPS	Speed Up	Efficiency
1	29.71	7.73		
2	14.89	15.42	1.99	99.5
3	10.85	21.17	2.73	91.0
4	8.53	26.92	3.48	87.0
5	7.47	30.74	3.97	79.4
6	6.49	35.39	4.57	76.1
7	6.00	38.27	4.95	70.7
8	5.68	40.43	5.20	65.0

Size of Matrix: 800*800 Double precision

nproc	Time	MFLOPS	Speed Up	Efficiency
1	43.27	7.89		
2	21.64	15.85	2.00	100
3	15.17	22.50	2.85	95
4	11.84	28.85	3.65	91.2
5	10.05	33.96	4.30	
6	8.77	38.92	4.90	
7	7.96	42.88	5.40	
8	7.30	46.76	5.90	


11-02-94
BPPS-8
LINPACK PERFORMANCE
MFLOPS
60.00
40.00
20.00
0.00
0.00
2.00
4.00
6.00
8.00
NO. OF PROCESSORS

2-D plot


MFLOPS
0.00
27.50
55.00
NPROC
8
9
4
2
1
100
400
700
1000
Matrix Size

3-D PLOT

GRAPHICS APPLICATION ON PPS

- LOW COST PARALLEL PROCESSING HAS GIVEN NEW PUSH TO VISUAL TECHNOLOGY
- SCIENTIFIC VISUALISATION
- COMPUTER GRAPHICS–ENGG. APPLN. FEM ETC.
- COMPUTER ANIMATION/VIRTUAL REALITY
- IMAGE PROCESSING
- MEDICAL IMAGING

SCIENTIFIC VISUALISATION

- Computer method of converting numeric data into visual format

- Researchers today restort to a computer based analysis and visualisation technique to expand their understanding of theoretical concept via geometrical constructs

- Availability of low cost parallel computers coupled with colour graphics work station (which has realtime texturig capabilities) and UNIX version of symbolic computations, has made it practable now.

VISUAL FORMATS & TOOLS

- Visual tools are needed to convert the large amount of data processed by parallel processors into optical properties such as colour, hue, intensity, opacity etc
- Today user can use 'GUI' tools available with workstation such as GNUPLOT, MATHCAD, MATHAMATICA, HPG etc
- Visualisation tools not only allow users visual representation of their data but users also can reveal certain characteristics otherwise unknown.
- With scientific visualisation users can attain better understanding and knowledge about physical process under study

What is needed ?

- Conceive the visual output.
- Model the output data formals
- Use geometrical constructs to do modelling
- Generate pictorial data for visualisation.

USER REQUIREMENTS

* OFF LINE VISUALISATION
  - LARGE CPU BOUND USER JOBS

* ON LINE VISUALISATION
  - NEAR REAL TIME GRAPHICS
    - IMAGE PROCESSING
    - SIMULATION
    - MODELING
  - REAL TIME GRAPHICS
    - VIRTUAL REALITY
    - PICTURE ANIMATION
    - MEDICAL IMAGING

TECHNOLOGY ON BPPS (ANUPAM)

* EACH APPLICATION DEMANDS DIFFERENT GRAPHICAL INTERFACE

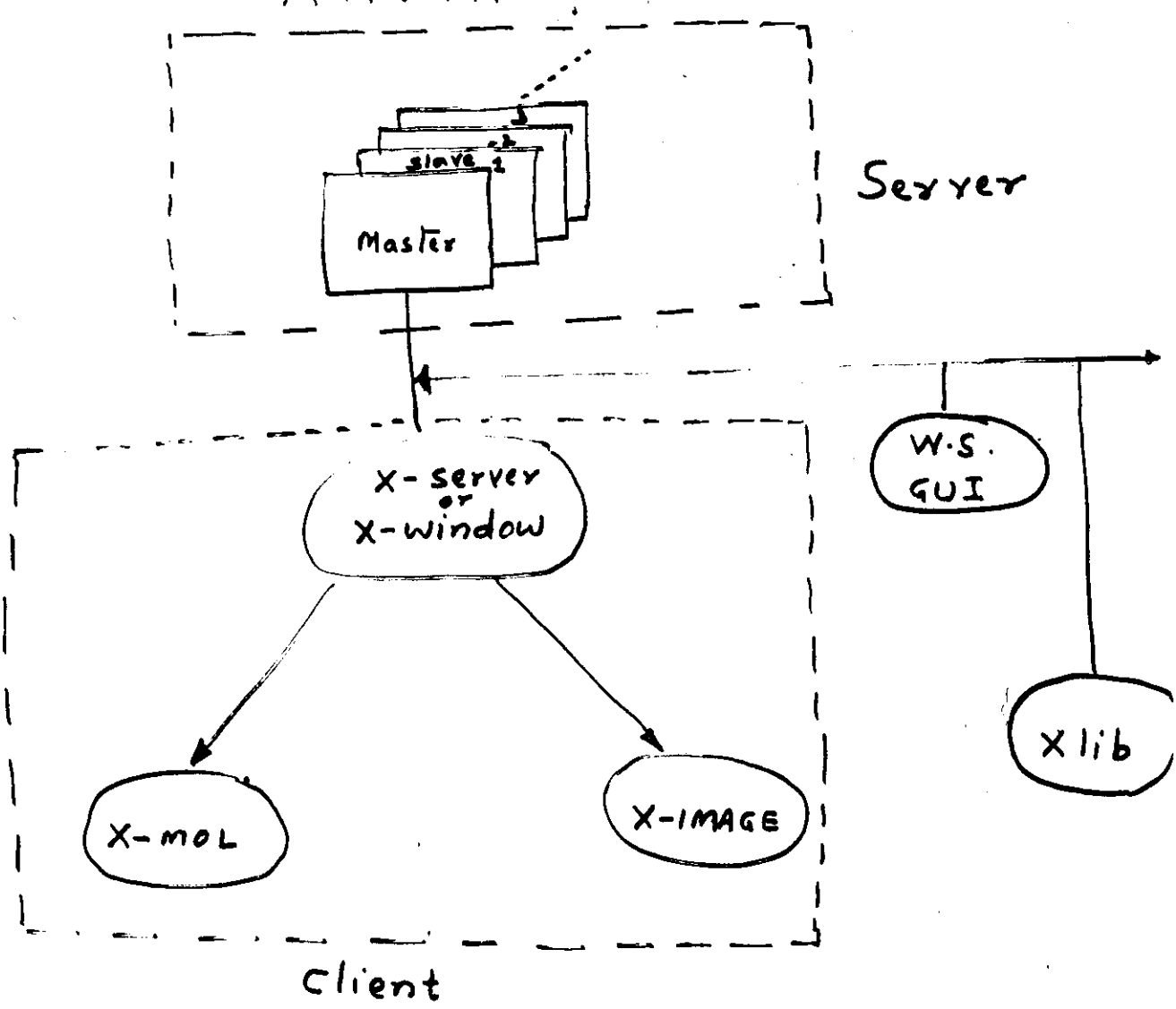
* TCP/IP-FTP → NETWORK OR LOOSE COUPLING
  - FOR OFFLINE DISPLAY/VISUALISATION GENERAL PURPOSE USER

* X-LIB -(CLIENT/SERVER)
  - FOR NEAR REAL TIME APPLICATION
    - SKILLED USER/VISUALISATION

* DEDICATED CLIENT/SERVER-SPECIFIC APPLN.
  - IMAGE PROCESSING - XIMAGE
  - MOLECULAR MODELLING - XMOL

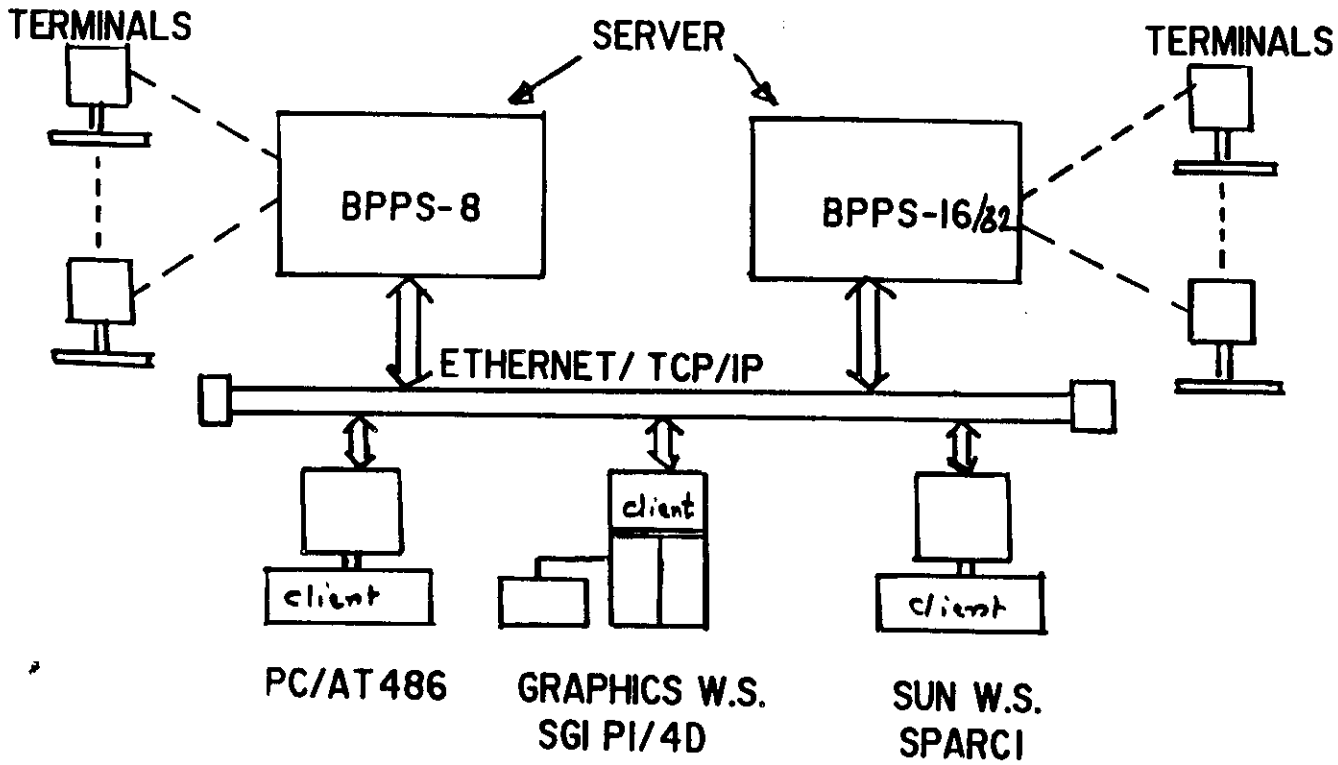
* CLIENT/SERVER — TIGHT COUPLING
  GRAPHICS - X SERVER FOR REAL TIME GRAPHICS APPLICATIONS

Scientific Visualisation ON ANUPAM (software)

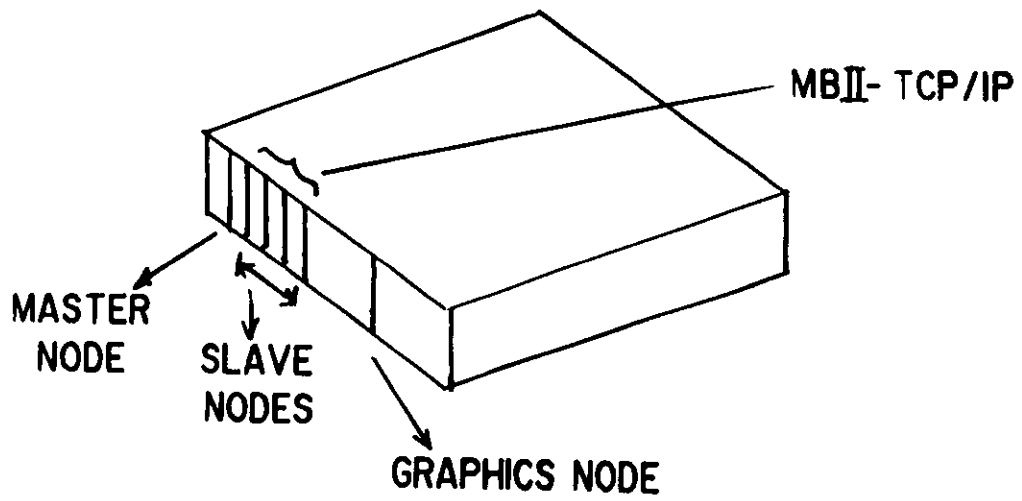


- W.S. Workstation
  - Any graphics workstation with UNIX & X support
  - SGI, SUN, HP, PC/AT 486 are supported.

·BPPS CONFIGURATION (Hardware)

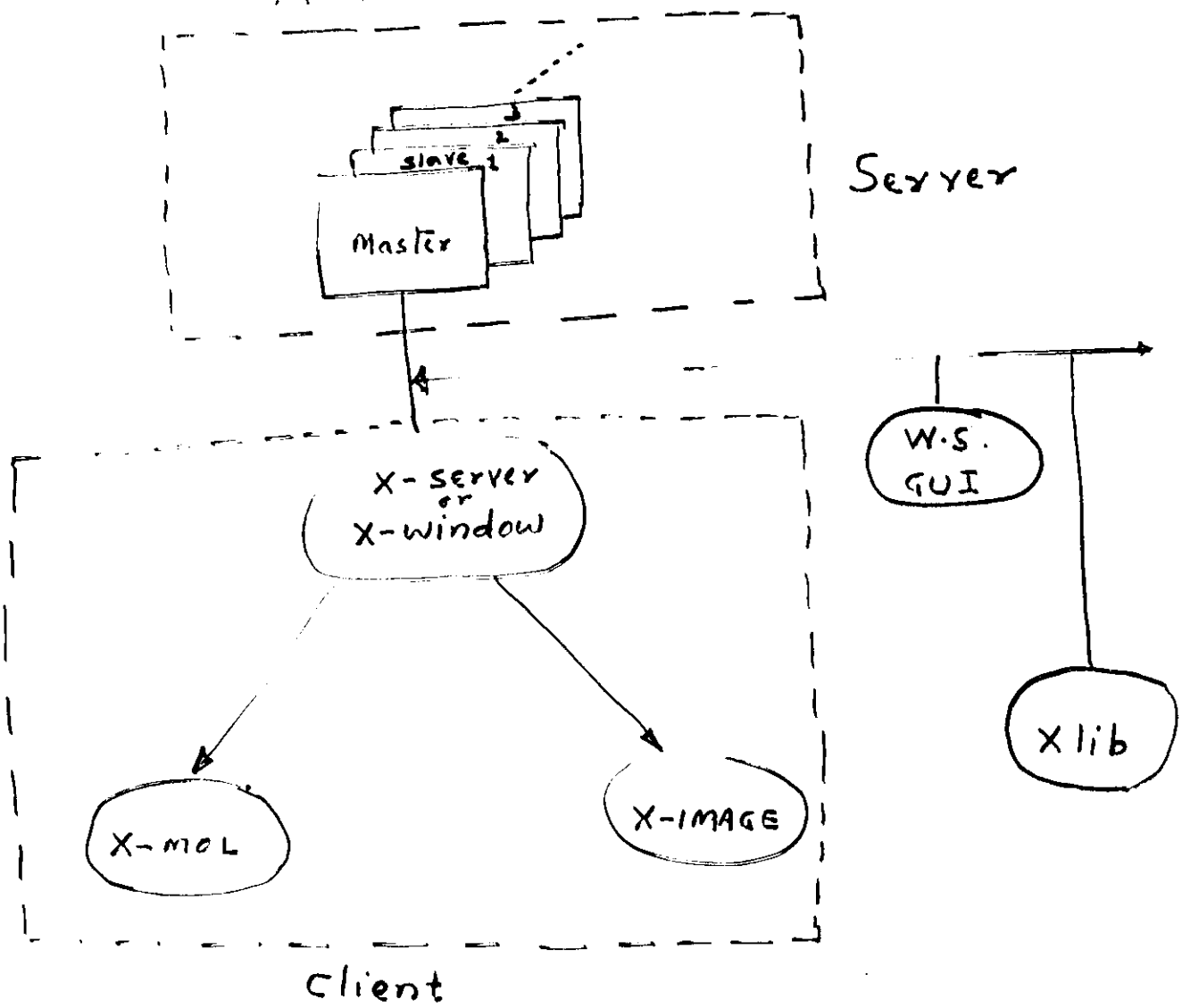


LOOSE COUPLING



TIGHT COUPLING

Scientific Visualisation ON ANUPAM (software)



- W.S. Workstation
  - Any graphics workstation with UNIX & X support
  - SGI, SUN, HP, PC/AT 486 are supported.

X IMAGE ON ANUPAM

* IMAGE PROCESSING APPLICATIONS ARE COMPUTE INTENSIVE & IDEAL FOR PARALLELISATION.

* XIMAGE SOFTWARE IS BASED ON CLIENT/SERVER MODEL.

* SERVER MODULE RUNS ON ANUPAM PARALLEL COMPUTER.

* CLIENT MODULE CAN RUN ON ANY COLOUR GRAPHICS WORKSTATION WITH X-WINDOW SUPPORT.

  * SUN * SGI * PC/AT WITH SVGA AND X SERVER.

* SERVER SUPPORTS PARALLEL IMAGE PROCESSING ROUTINES SUCH AS FFT, IFFT, FILTERS, EGE DETECTION, ETC.

* USER DEFINEDFUNCTIONS CAN BE EASILY ADDED.

* COMPLETE MENU DRIVEN USER FRIENDLY INTERFACE.

X-MOL

- BASED ON CLIENT/SERVER APPROACH WITH ANUPAM AS BACKEND SERVER PROVIDING COMPUTATIONAL POWER & X-BASED GRAPHICS WORKSTATION AS A CLIENT FOR DISPLAY PURPOSE
- PROVIDES VISUALISATION OF PROTEIN & NUCLEIC ACID MOLECULAR STRUCTURES USING BROOKEVEN PROTEIN DATA BANK DATA
- PROVIDES SIMPLE GRAPHICAL USER INTERFACE WITH POINT-AND-CLICK OPERATIONS
- PROVIDES VISUALISATION OF MOLECULES IN VARIOUS WAY LIKE STICK MODEL, STICK MODEL WITH DEPTH QUEING, STICK & BALL MODEL, SPACE FILLED MODEL, ETC.
- SCAN CONVERSION & RENDERING ALGORITHMS RUN IN PARALLEL MODE ON ANUPAM USING OBJECT PARALLELISATION APPROACH
- GUI IS COMPATIBLE WITH OPENLOOK TOOL-KIT