Multiprocessor, Parallel Processing

By:

Dr. Subra Ganesan

Professor, CSE Department, Oakland University

Rochester, MI 48309. USA.

The Microprocessor overview

1949 Transistors
1958 Integrated Circuits
1961 ICs IN Quality
1964 Small Scale IC(SSI) Gates
1968 Medium Scale IC(MSI) Registers
1971 Large Scale IC(LSI), Memory, CPU
1972 8 BIT MICROPROCESSORS
1973 16 BIT MICROPROCESSORS
1982 32 BIT MICROPROCESSORS
1984 DSP MICROPROCESSORS – I GENERATION
1986 DSP MICROPROCESSORS – II GENERATION
1988 DSP MICROPROCESSORS – III GENERATION
1989 RISC MICROPROCESSORS – II QUALITY
1990 MISC MINIMUM INSTRUSTION SET MICROPROCESSOR

MICROPROCESSOR OVERVIEW

Microprocessor	Number of transistors	Performance	Number of Instructions
4 Bit Intel 4004 1971	2300		45
68000	70000	0.5 MIPS	80 Different ▪ 14 address mode ▪ Size B,W,L
TMS 320C80 32 bit RISC	More than a Million	2 Billion operations per second [BOPs]	

Computer Evolution

Generation I	Vacuum Tube/ Accoustic Mem	1946- 1954
Generation II	Transistor/Ferrite Core	1955-64
Generation III	Integrated Circuits	1965-74
Generation IV	LSI, Memory Chips/Multiprocessors	1975-89
Generation V	Non VonNeuman Architecture	1985- present
	Parallel Processing	

Parallel Processing is an efficient form of information processing which emphasizes the exploitation of concurrent events

Concurrency implies parallelism, simultaneity and pipelining.

Multiple jobs or programs – multiprogramming

Time sharing

multiprocessing

Parallel Processing requires knowledge of:

- Algorithms
- Languages
- Software
- Hardware
- Performance Evaluation
- Computing Alternatives

From Operating Systems View, computers have improved in 4 phases:

- Batch Processing
- Multiprogramming
- Time Sharing
- Multiprocessing

Parallel Processing in UniProcessor:

- Multiple Functional Units
- Pipelining, Parallel Adders, Carry Look Ahead Adders
- Overlapped CPU and I/O operation
- Use of Hierarchical Memory
- Balancing Subsystem Bandwidth
- Multiprogramming and Time Sharing

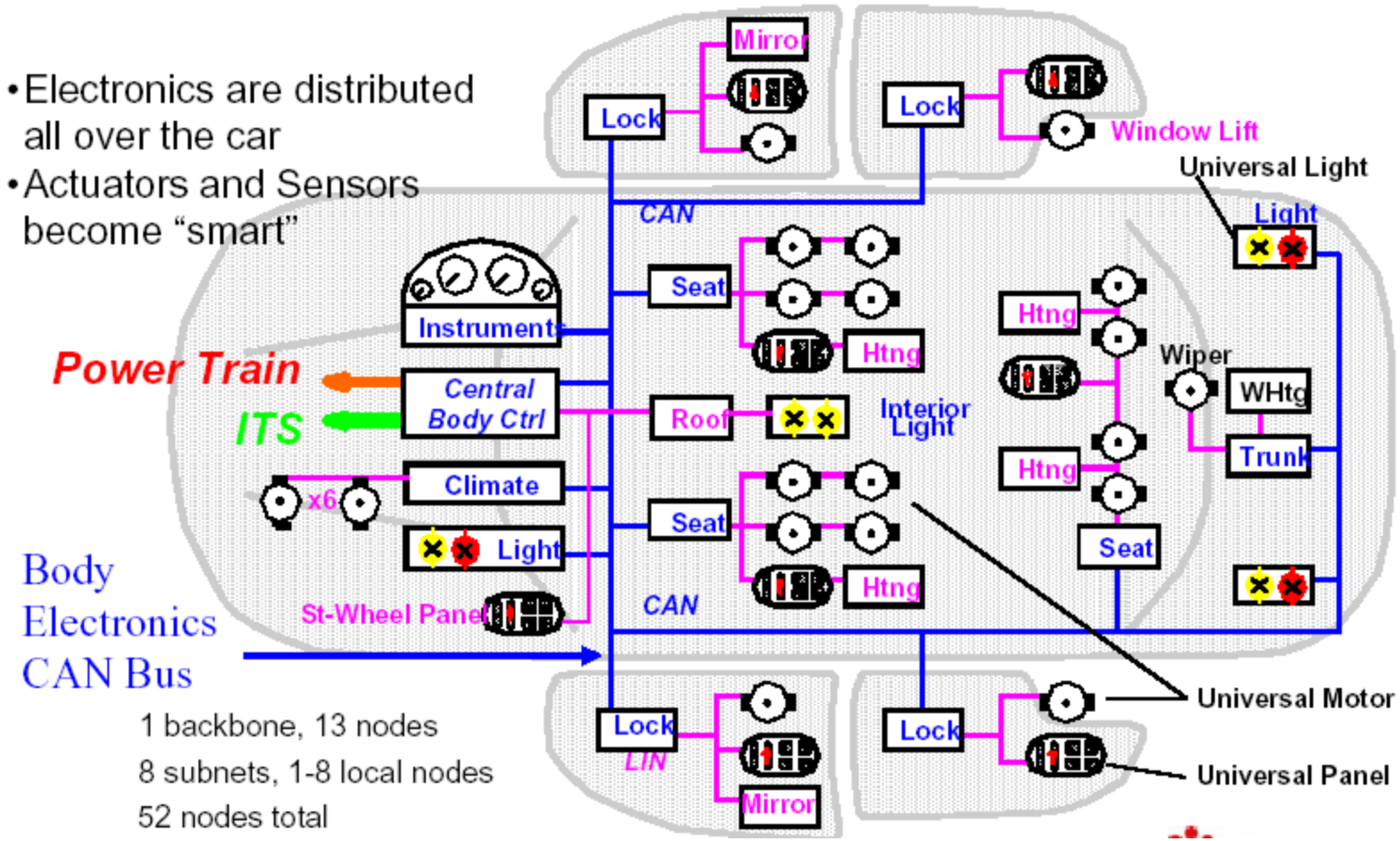
Example Parallel Processing

Embedded Systems

Scientific Applications

Automotive Body Network

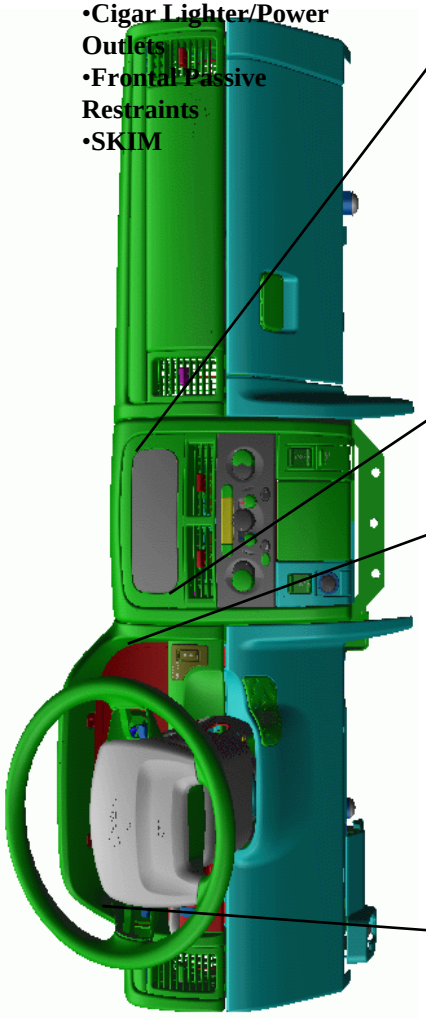
- Electronics are distributed all over the car
- Actuators and Sensors become "smart"



1 backbone, 13 nodes
8 subnets, 1-8 local nodes
52 nodes total

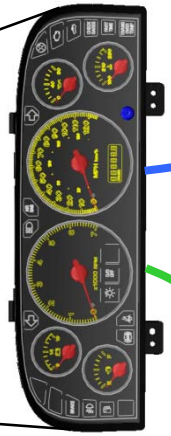
Future Car network

INSTRUMENT PANEL
- **Instrument Cluster**
- **Junction Block Module**
- **HVAC Unit/Controller**
- **Radio/ICU**
- **Speakers**
- **Switches**
- **Lamps**
- **Cigar Lighter/Power Outlets**
- **Frontal Passive Restraints**
- **SKIM**

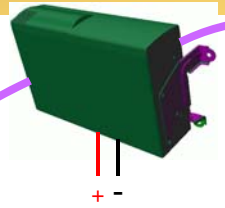


RADIO/ICU
- **CAN B / MOST Gateway**
- **NAV**
- **HVAC**

INSTRUMENT CLUSTER
- CAN B / CAN C Gateway

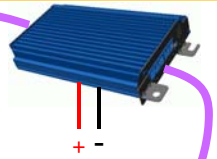


CD CHANGER



DVD PLAYER

DISTRIBUTED AMPLIFIERS

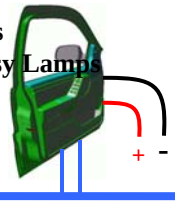


NAV

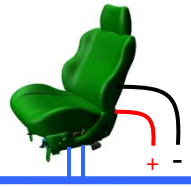
PHONE

MOST
Media Oriented Systems Transport

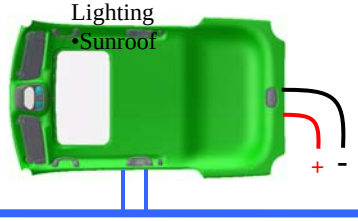
DOORS
- **Window Motors**
- **Lock Motors**
- **Switches (window, lock, mirror, memory, disarm, ajar)**
- **Mirrors**
- **Courtesy Lamps**



SEATS
- power
- heat
- memory
- switches

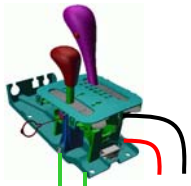


OVERHEAD
- EVIC
- Courtesy/Dome Lighting
- Sunroof

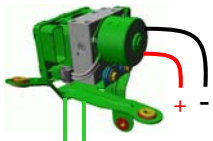


CAN B
Low Speed Data Bus

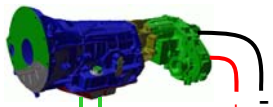
SHIFTER
- Shift by Wire
- AutoStick
- Shift Interlock
- Illumination



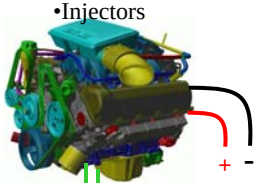
ABS



TRANSMISSION
- **Transmission Controller**
- **Transfer Case Controller**

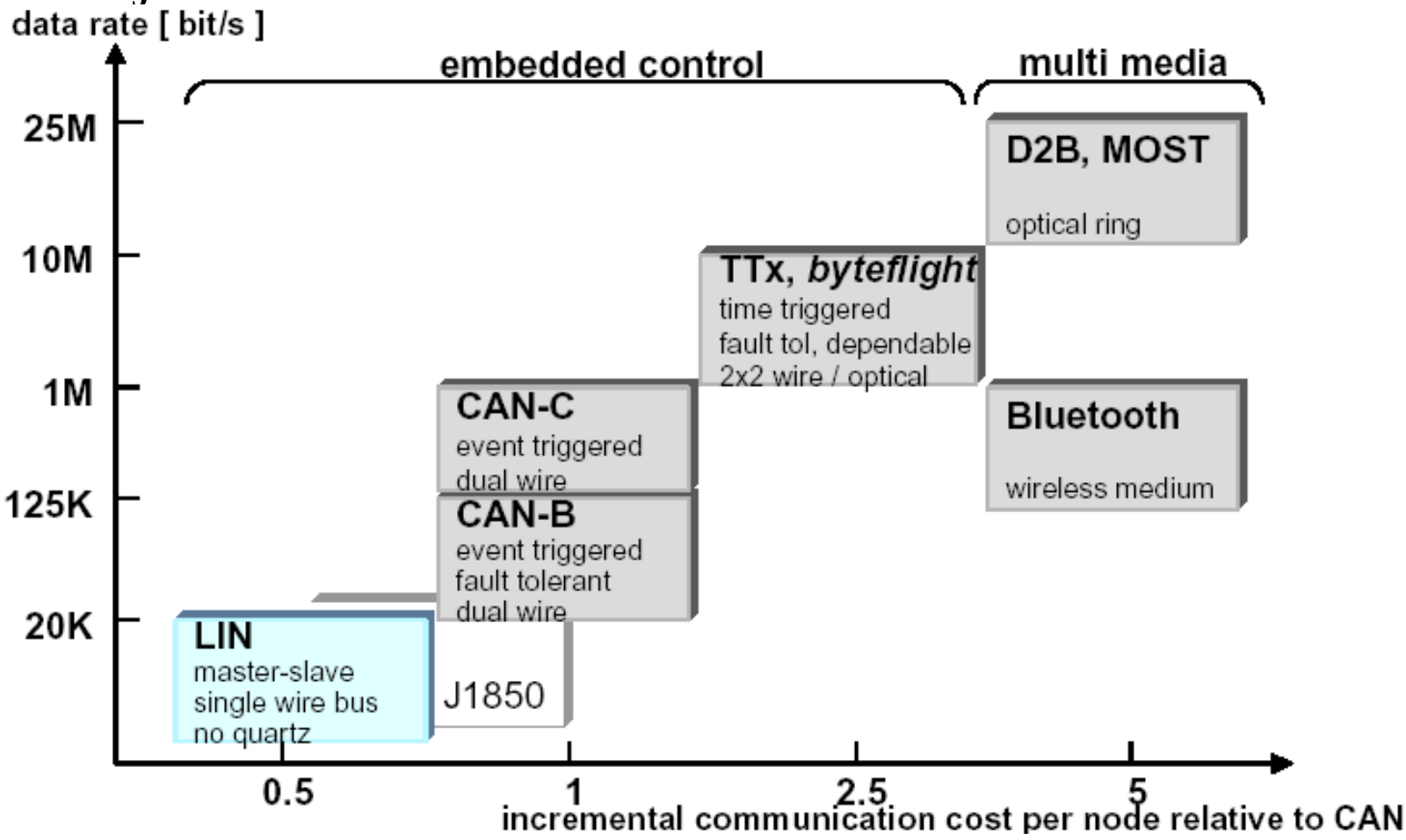


ENGINE
- **Engine Controller**
- Sensors
- Injectors



CAN C
High Speed Data Bus

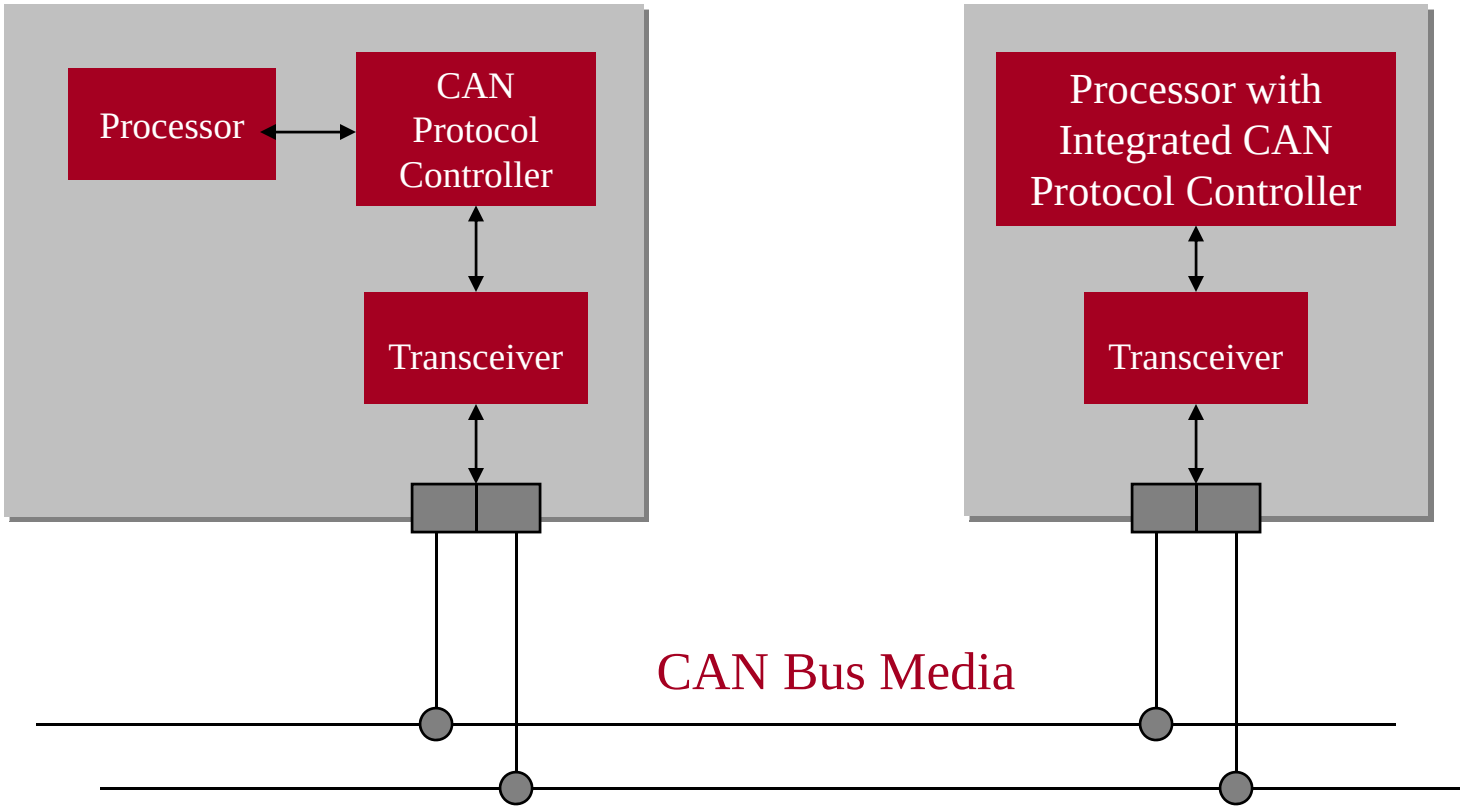
Major Protocols for Automotive Embedded Control



Hardware Implementations

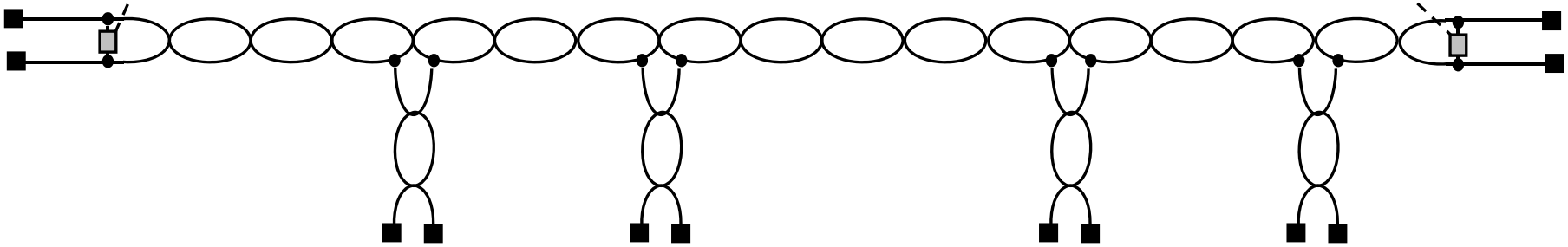
Using an external CAN controller

Using an internal CAN controller



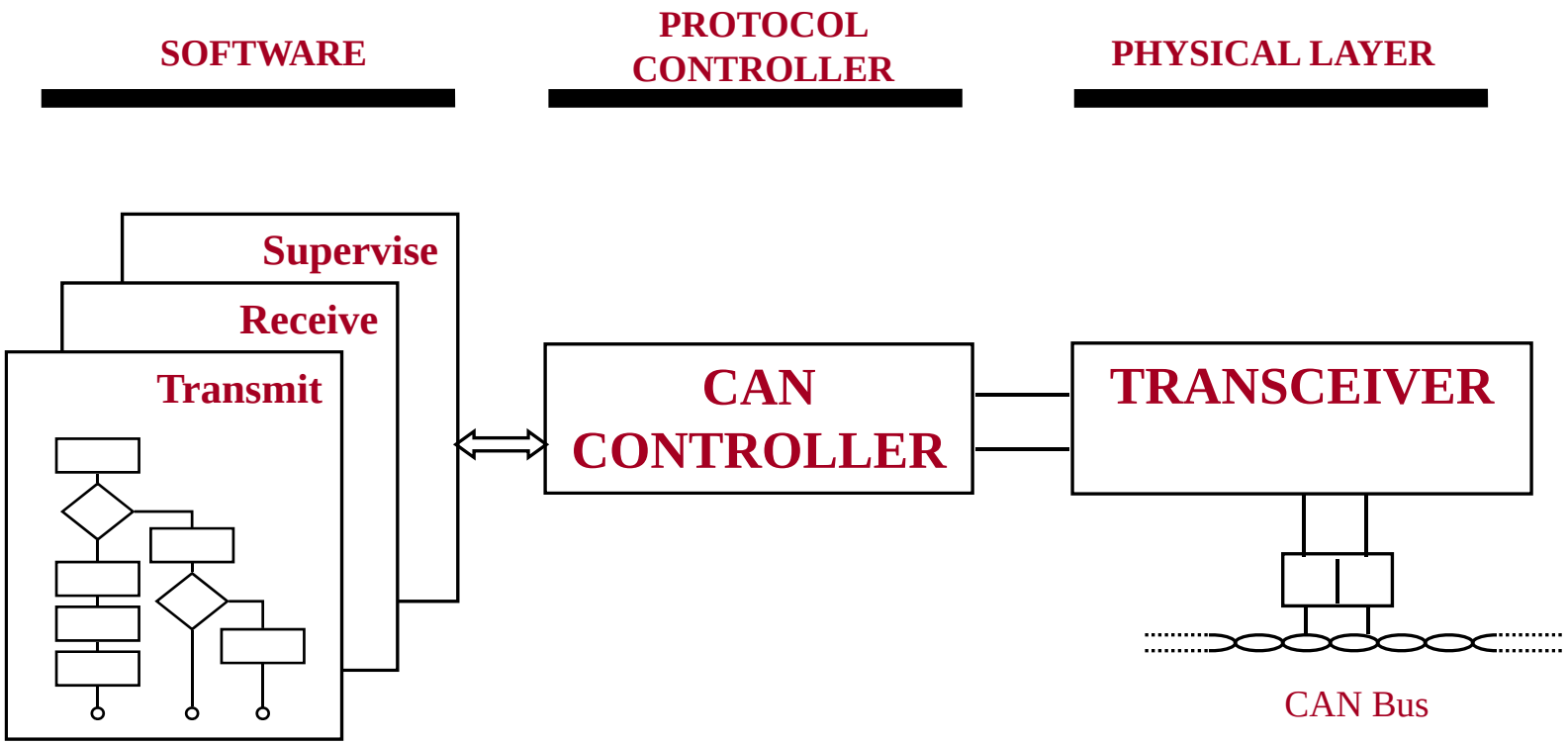
BUS Termination

Termination resistors at the ends of the main backbone
Each 120 ohms 5% 1/4 W

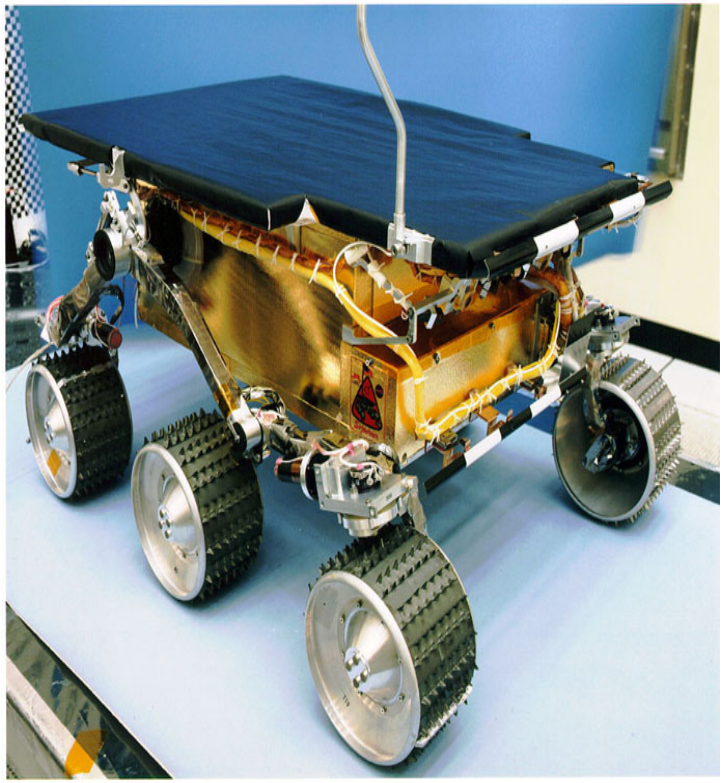


- Absolutely required when bus length and speed increases
- Not used by many medium and low speed CAN implementations

CAN Physical Layer



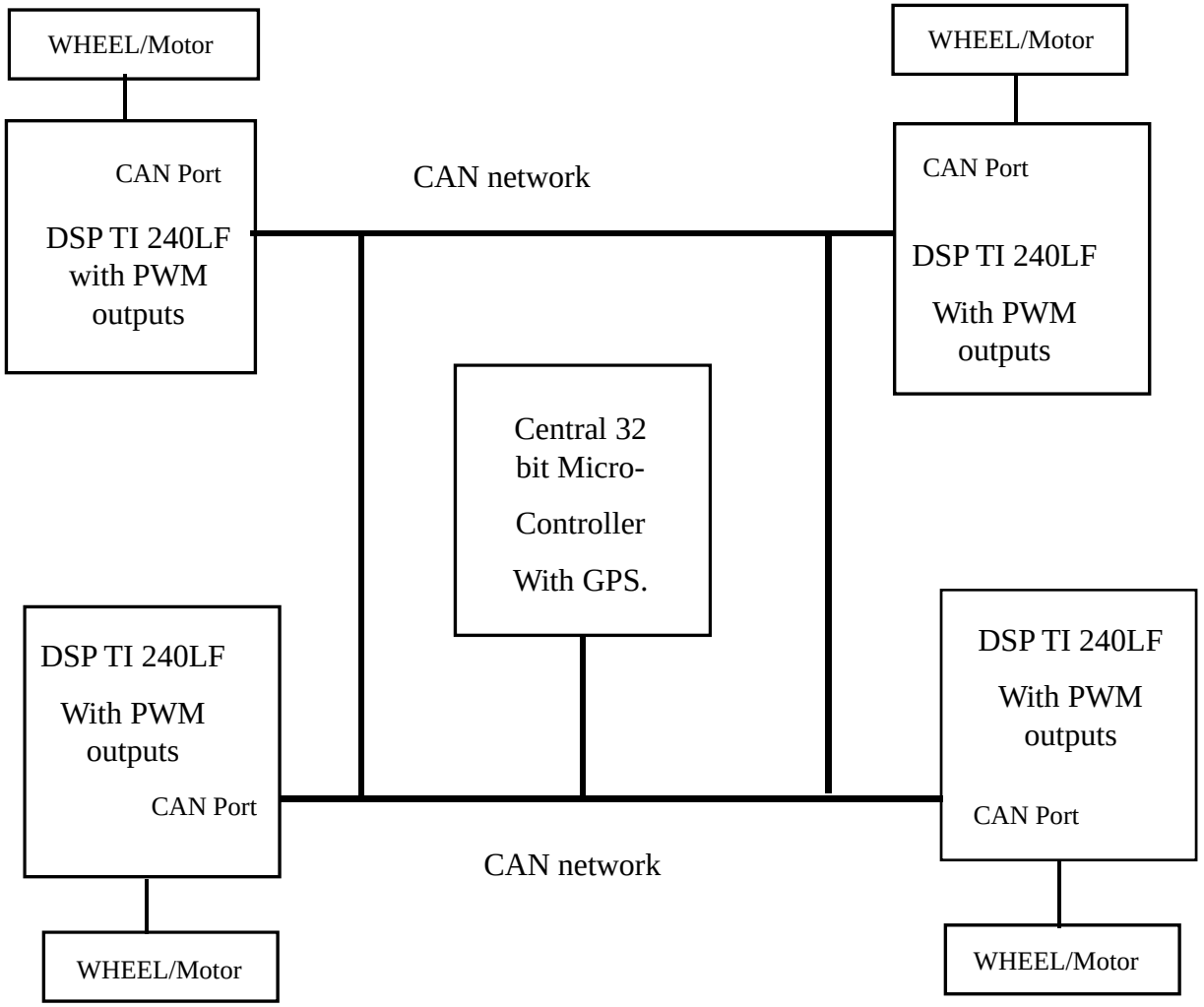
Unmanned Vehicle



Product: NASA's Mars Sojourner Rover.

Microprocessor: 8-bit Intel 80C85.

Self Guided Vehicle with 4 independent wheel controller and Central Controller with GPS.

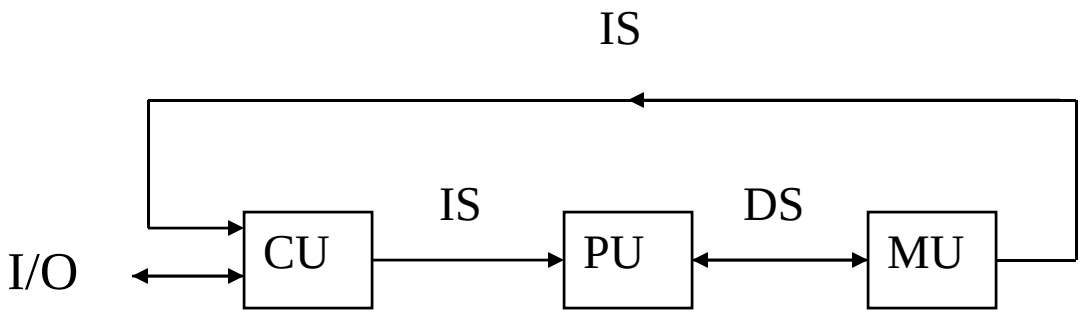


Flynn’s Classification

- SISD– Single Instruction, Single Data Stream
- MISD- Multiple Instruction Multiple Data Stream
- SIMD-Single Instruction Multiple Data Stream
- MIMD- Multiple Instruction Multiple Data Stream.

Flynn's Classification of Computer Architectures

(Derived from Michael Flynn, 1972)



(a) SISD Uniprocessor Architecture

Captions:

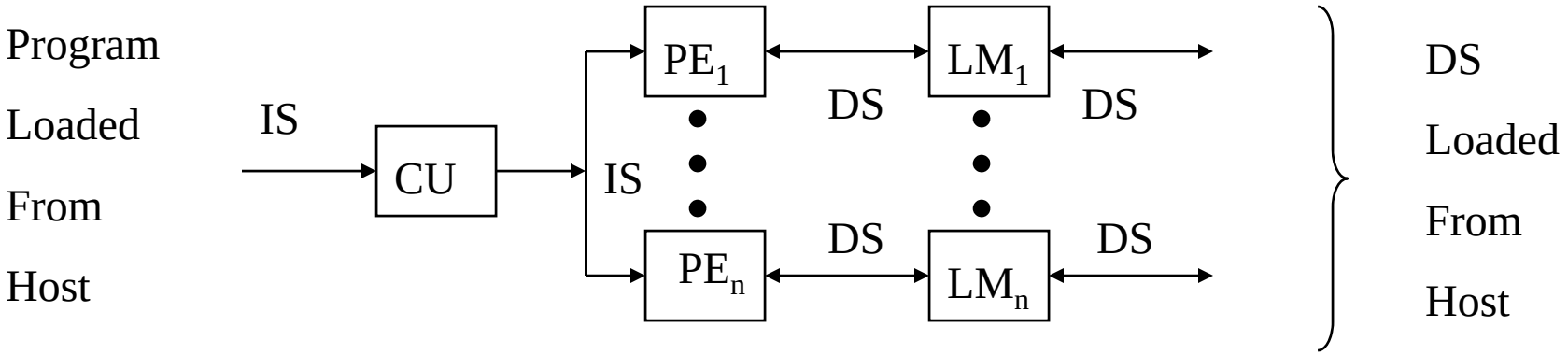
CU - Control Unit ; PU – Processing Unit

MU – Memory Unit ; IS – Instruction Stream

DS – Date Stream

Flynn's Classification of Computer Architectures

(Derived from Michael Flynn, 1972) **(contd…)**



(b) SIMD Architecture (with Distributed Memory)

Captions:

CU - Control Unit ; PU - Processing Unit

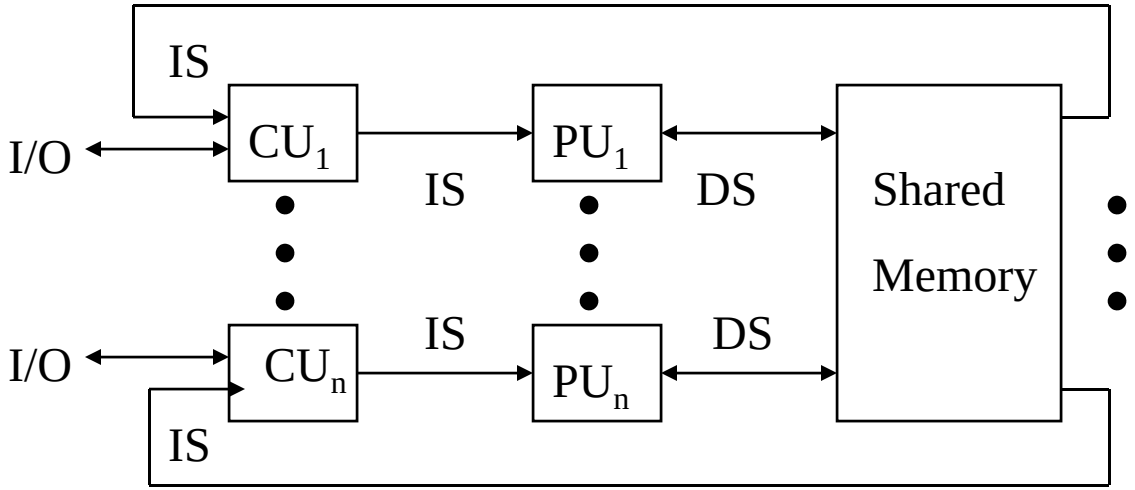
MU - Memory Unit ; IS - Instruction Stream

DS - Date Stream ; PE – Processing Element

LM – Local Memory

Flynn's Classification of Computer Architectures

(Derived from Michael Flynn, 1972) **(contd…)**



(c) MIMD Architecture (with Shared Memory)

Captions:

CU - Control Unit ; PU - Processing Unit

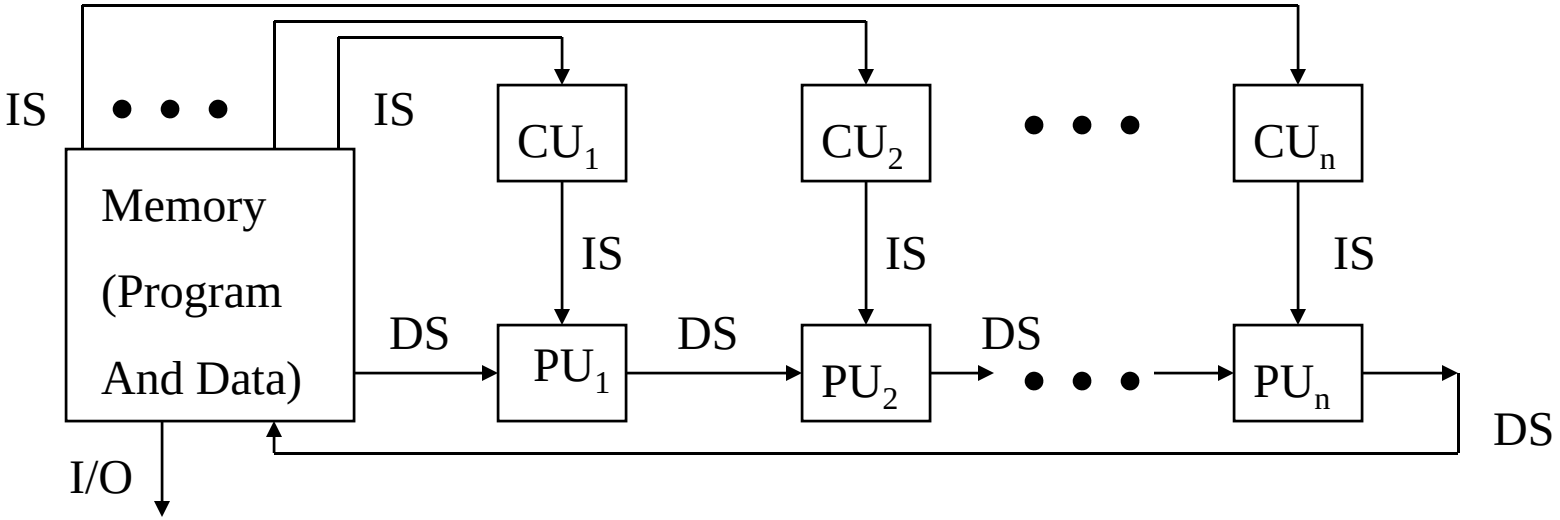
MU - Memory Unit ; IS - Instruction Stream

DS - Date Stream ; PE – Processing Element

LM – Local Memory

Flynn's Classification of Computer Architectures

(Derived from Michael Flynn, 1972) **(contd…)**



(d) MISD Architecture (the Systolic Array)

Captions:

CU - Control Unit ; PU - Processing Unit

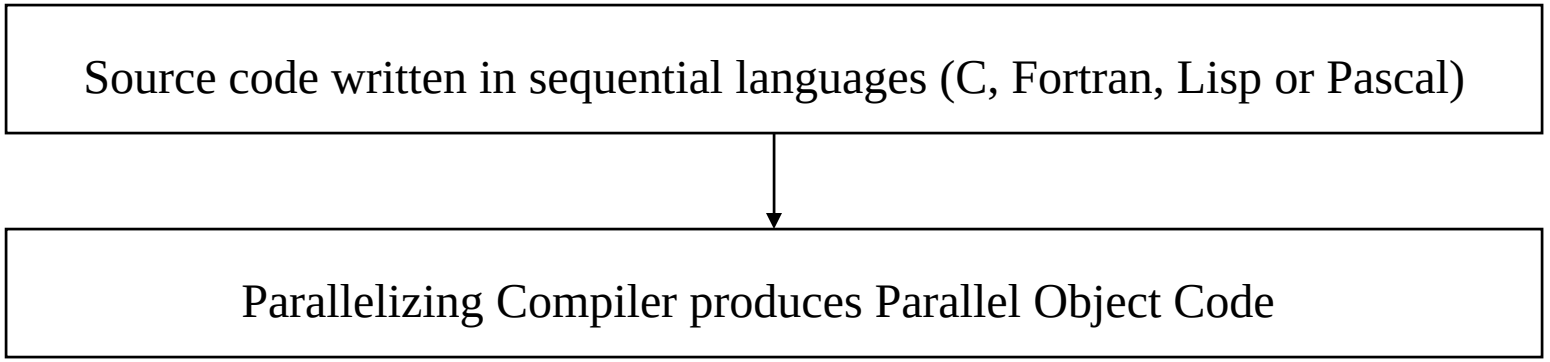
MU - Memory Unit ; IS - Instruction Stream

DS - Date Stream ; PE – Processing Element

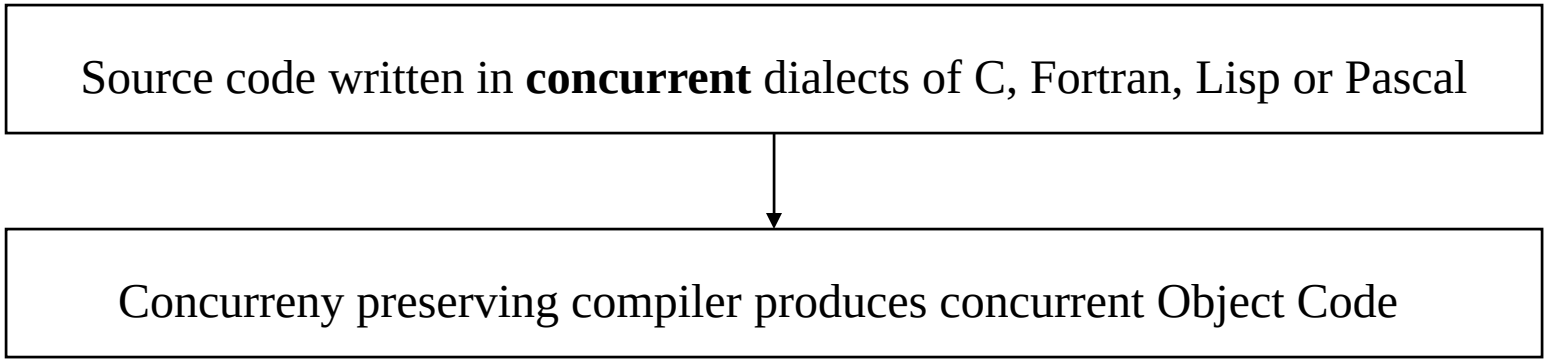
LM – Local Memory

Two Approaches to Parallel Programming

a) Implicit Parallelism



b) Explicit Parallelism

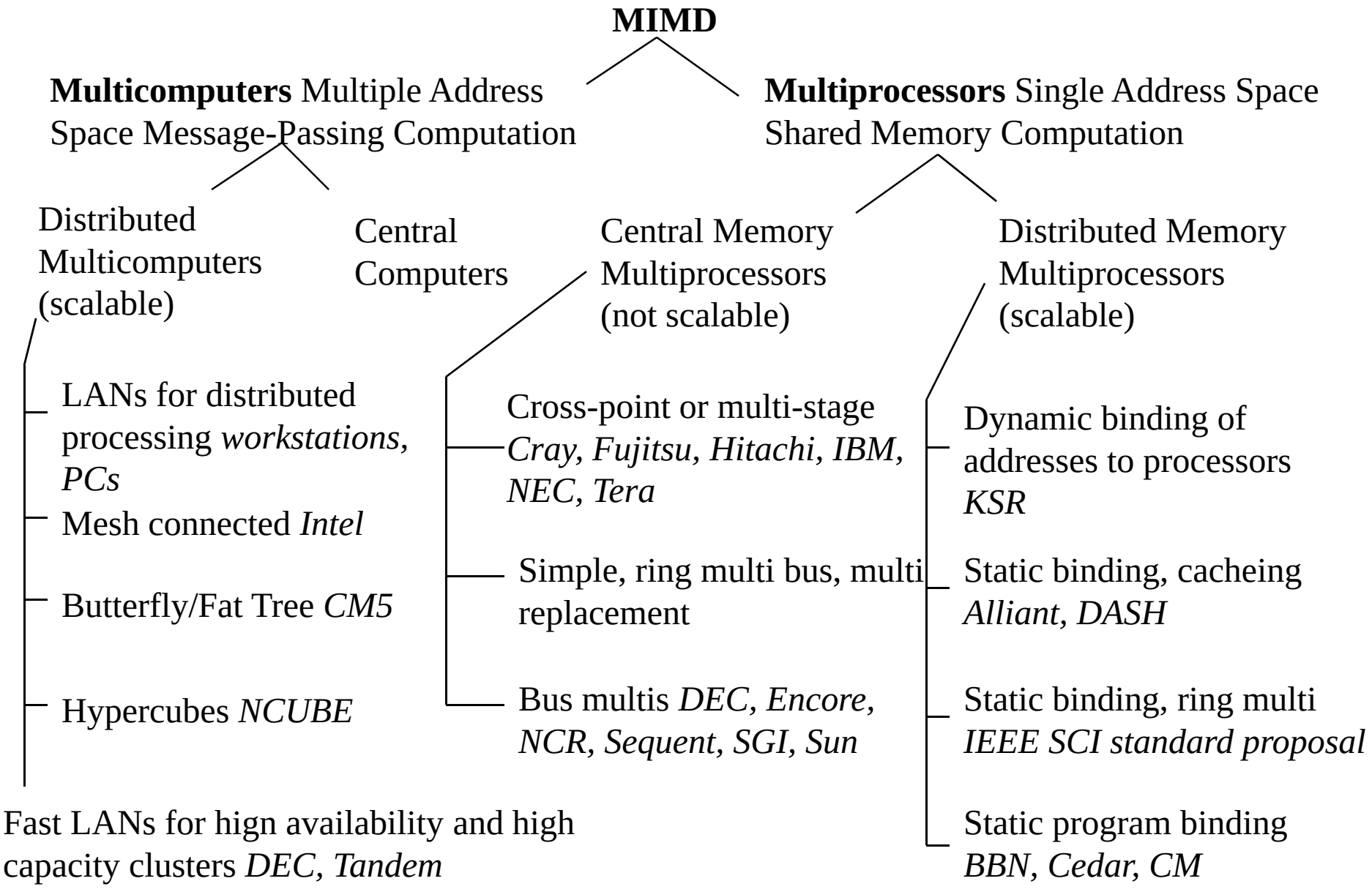


SIMD and MIMD

- SIMDSs appeal more to special purpose applications.
- SIMDs are not size-scalable.
- Thinking Machine CM2.

- MIMDs with distributed memory having globally shared virtual address space is the future trend.
- CM5 has MIMD architecture

Bell's Taxonomy of MIMD computers



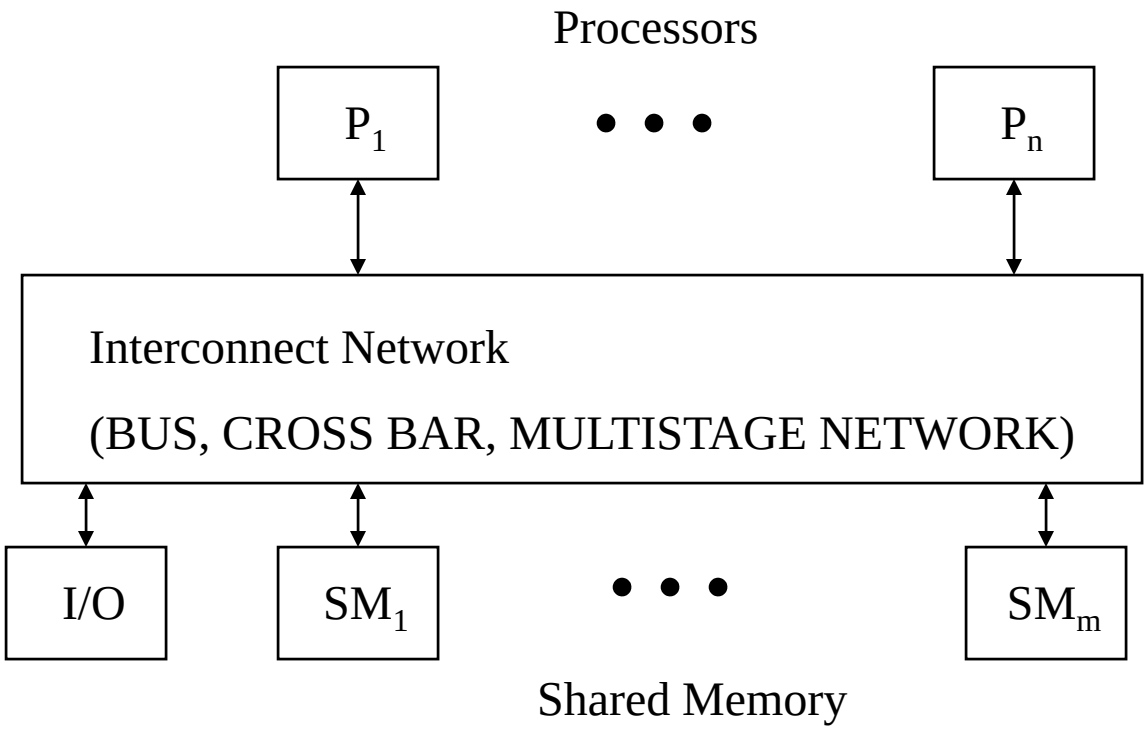
Two Categories of Parallel Computers

1. Shared Memory Multiprocessors (tightly coupled systems
2. Message Passing Multicomputers

SHARED MEMORY MULTIPROCESSOR MODELS:

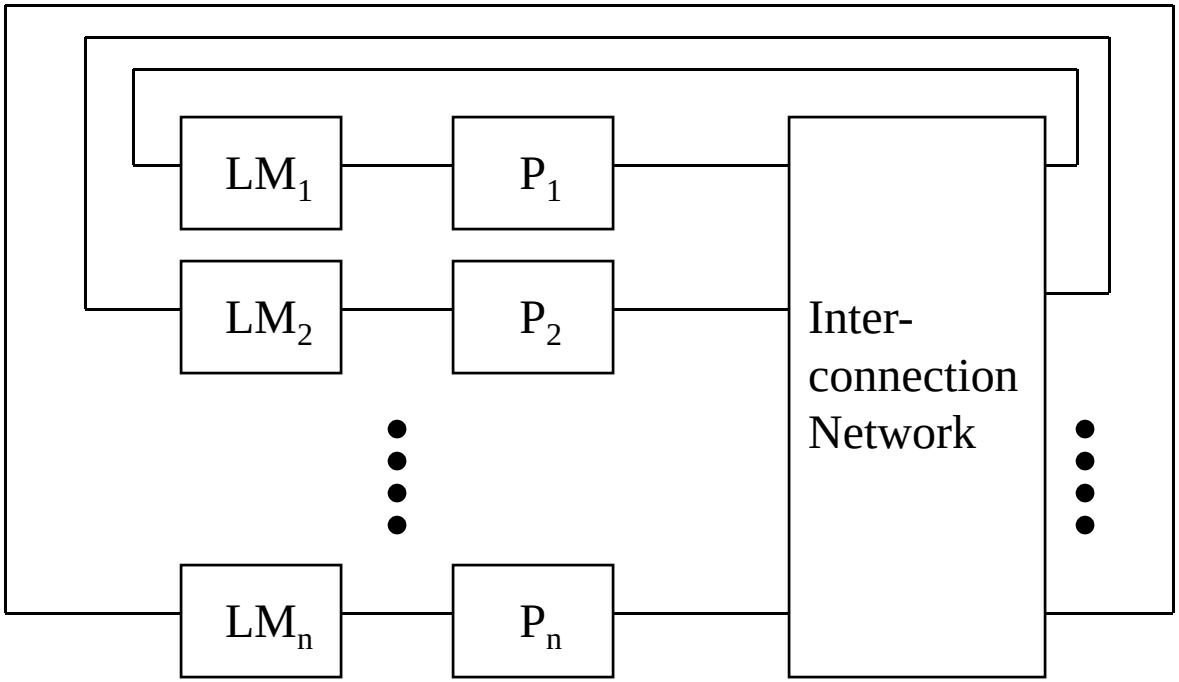
a. Uniform Memory Access (UMA)
b. Non-Uniform Memory Access (NUMA)
c. Cache-Only Memory Architecture (COMA)

SHARED MEMORY MULTIPROCESSOR MODELS



The **UMA multiprocessor model** (e.g., the Sequent Symmetry S-81)

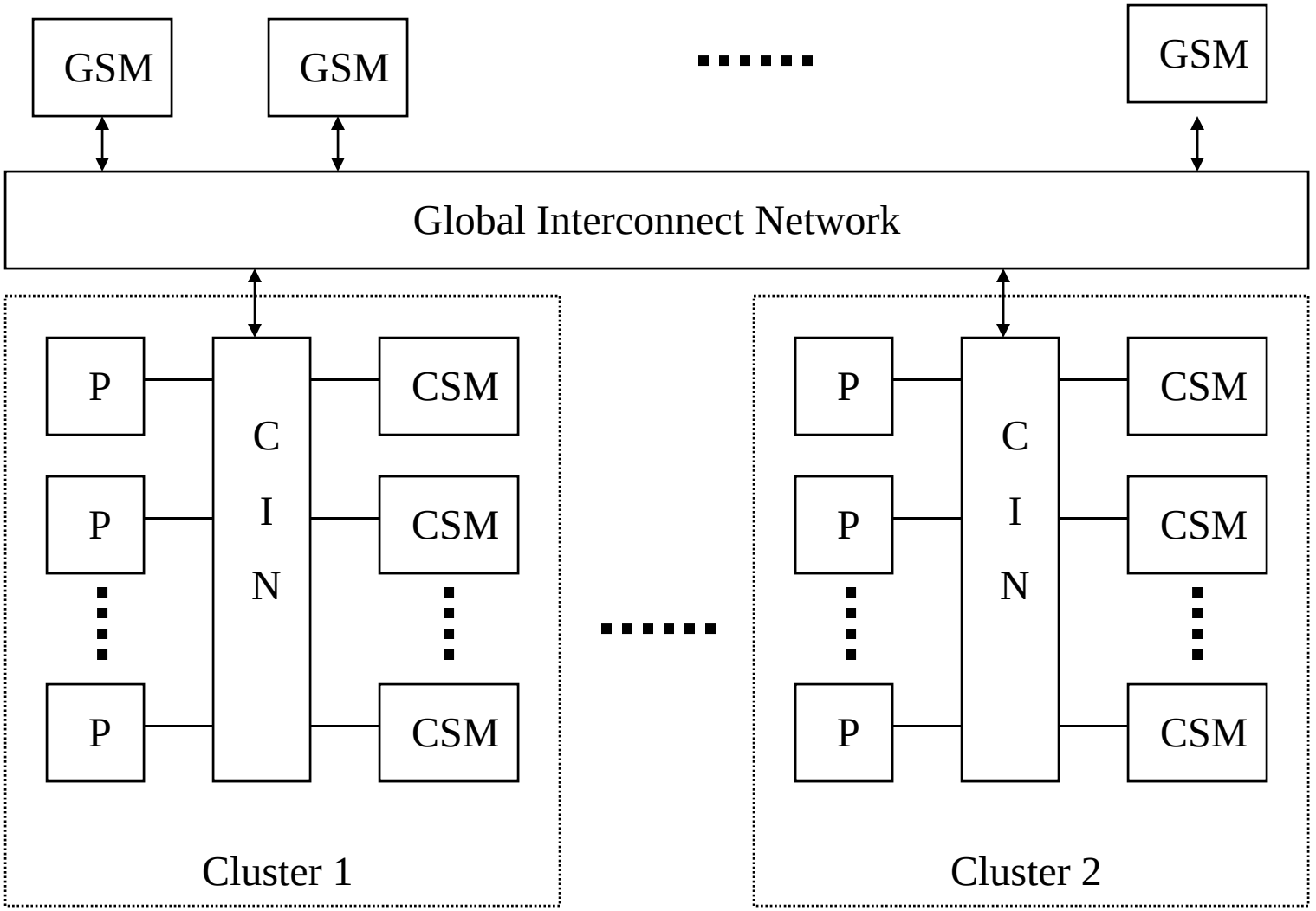
SHARED MEMORY MULTIPROCESSOR MODELS (contd…)



(a) Shared local Memories (e.g., the BBN Butterfly)

NUMA Models for Multiprocessor Systems

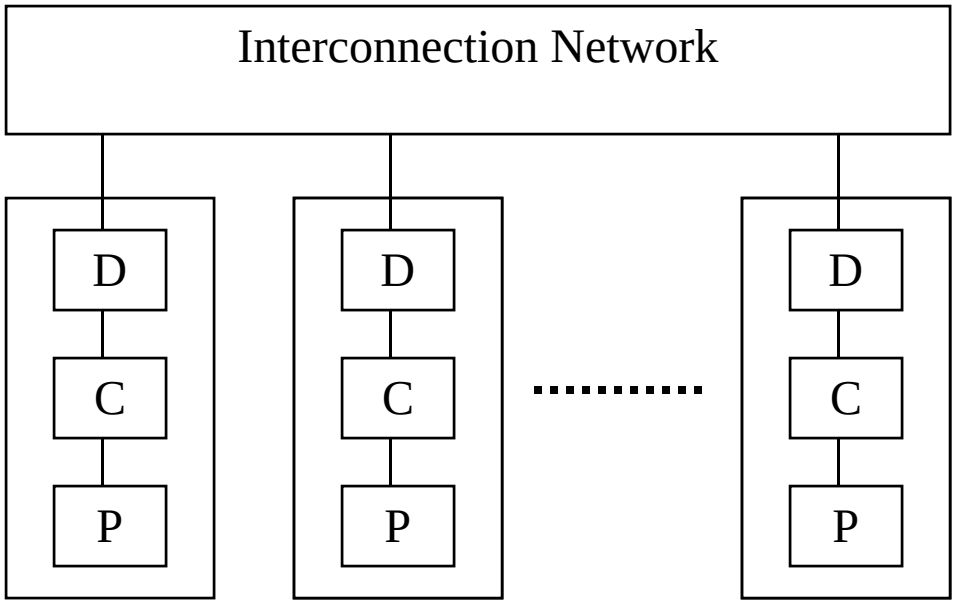
SHARED MEMORY MULTIPROCESSOR MODELS (contd…)



(b) A hierarchical cluster model (e.g., the Cedar system at the University of Illinois)

NUMA Models for Multiprocessor Systems

SHARED MEMORY MULTIPROCESSOR MODELS (contd…)



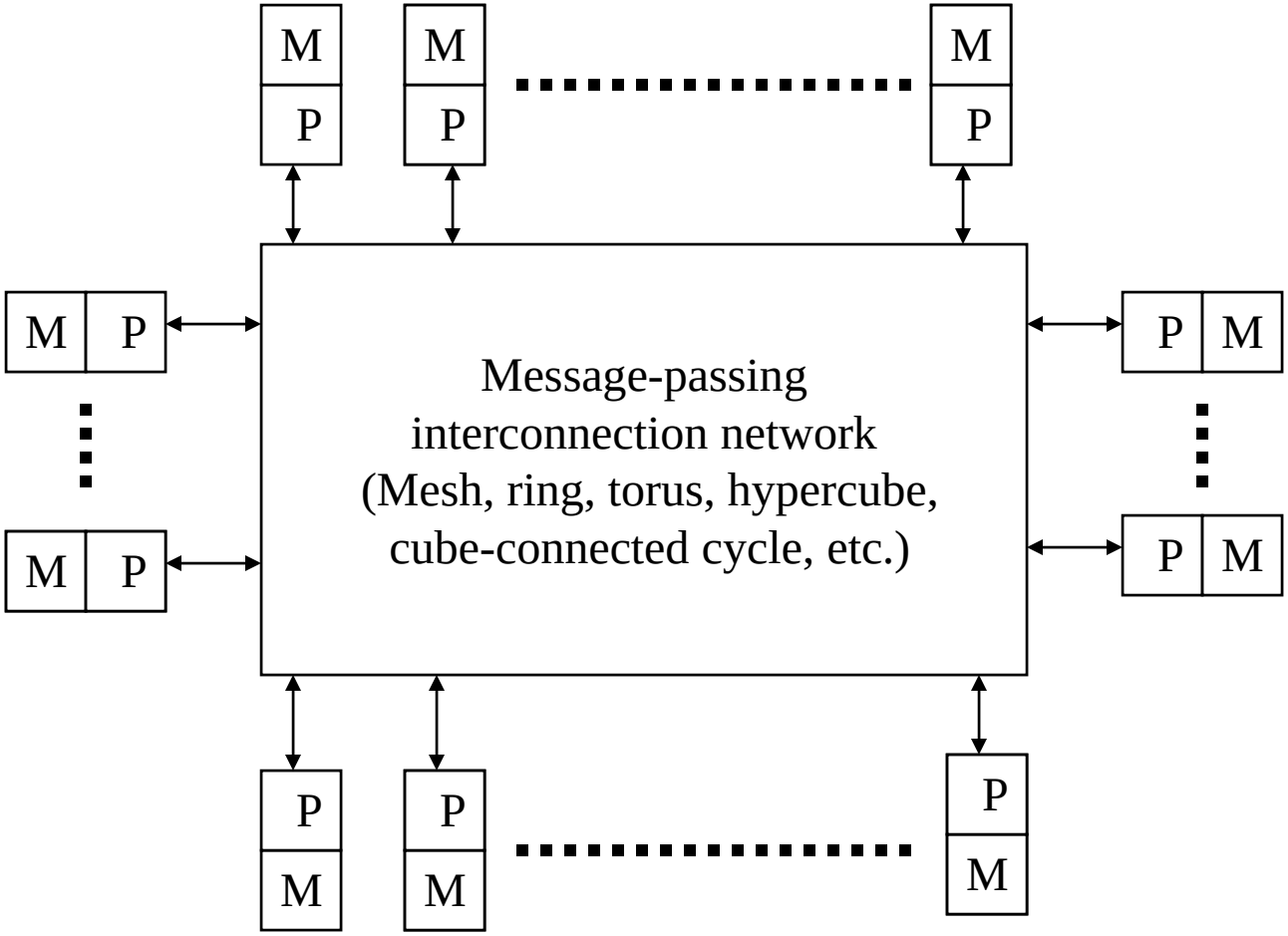
P : Processor

C : Cache

D : Directory

The COMA Model of a multiprocessor (e.g., the KSR-1)

Generic Model of a message-passing multicomputer



e.g., Intel Paragon, nCUBE/2

Important issues: Message Routing Scheme, Network flow control strategies, dead lock avoidance, virtual channels, message-passing primitives, program decomposition techniques.

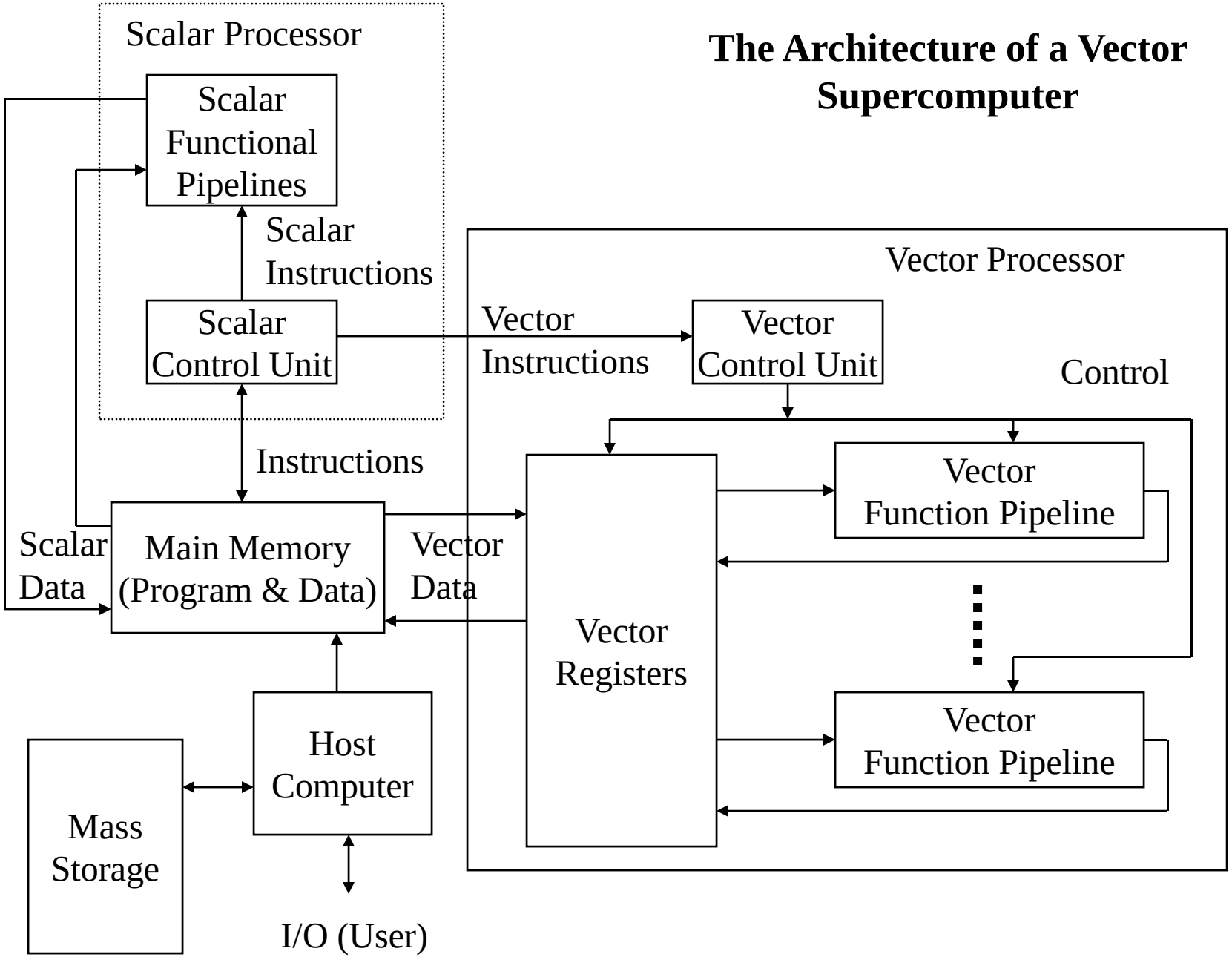
Theoretical Models for Parallel Computers

RAM Random Access Machines
e.g., conventional uniprocessor computer

PRAM Parallel Random Access Machines
model developed by Fortune & Wyllie(1978)
ideal computer with zero synchronization
and zero memory access overhead
For shared memory machine

PRAM-Variants
depending on how memory read & write are handled

The Architecture of a Vector Supercomputer



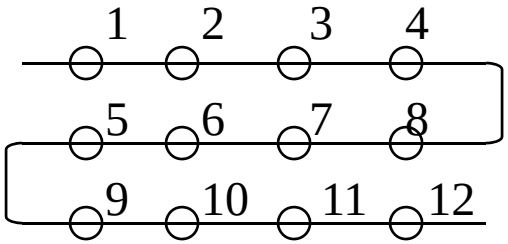
The Architecture of a Vector Supercomputer (contd)

e.g.,	Convex C3800	8 processors
		2G FLOPS peak
	VAX 9000	125-500 MFLOP
	CRAY YMP&C90	built with ECL
		10K ICS
		16 G FLOP

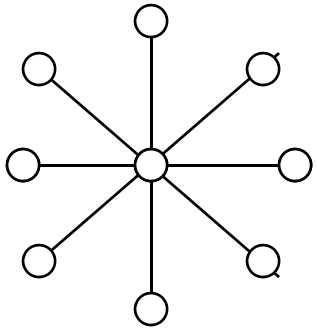
Example for SIMD machines

- MasPar MP-1; 1024 to 16 K RISC processors
- CM-2 from Thinking Machines, bitslice, 65K PE
- DAP 600 from Active memory Tech., bitslice

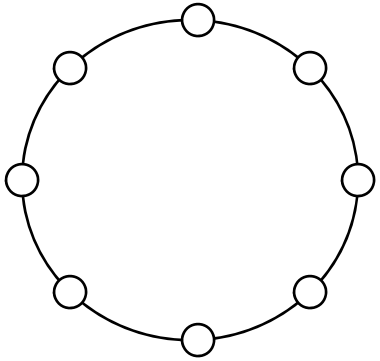
STATIC Connection Networks



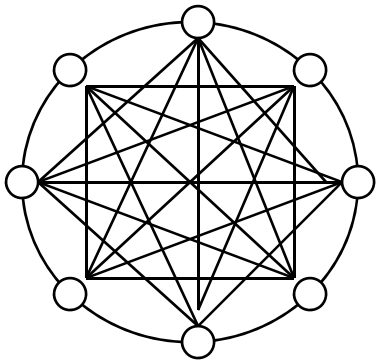
Linear Array



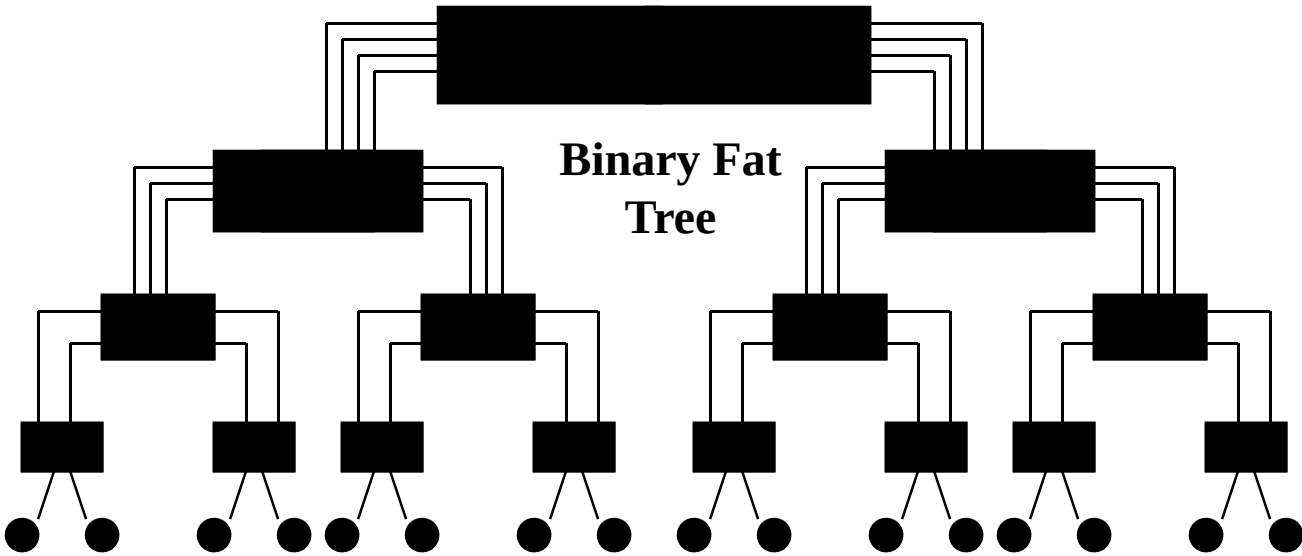
Star



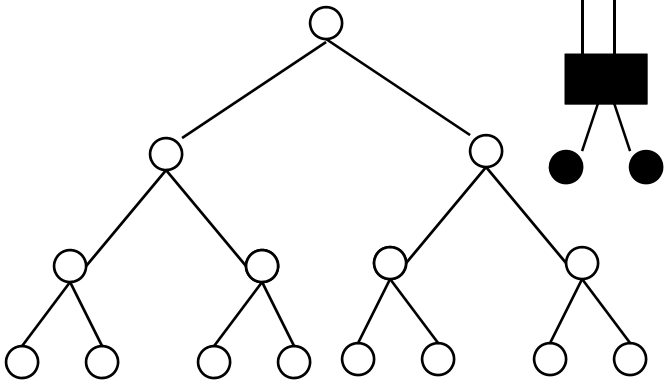
Ring



Fully connected Ring

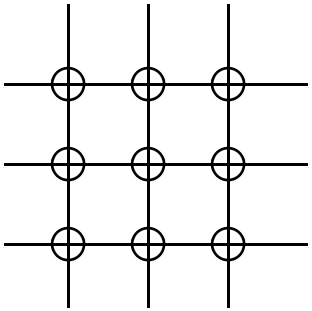


Binary Fat Tree

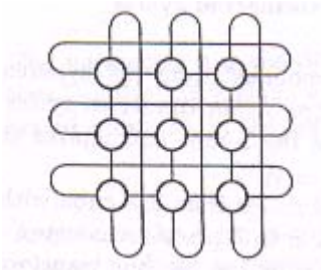


Binary Tree

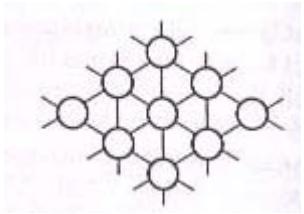
The Channel width of Fat Tree increases as we ascend from leaves to root. This concept is used in CM5 connection Machine.



Mesh

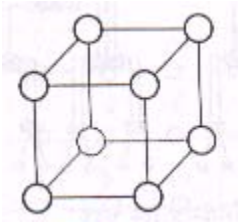


Torus

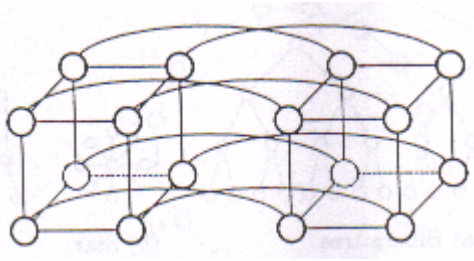


Systolic Array

Degree = t



3-cube



A 4 dimentional cube formed with 3D cubes

Binary Hypercube has been a popular architecture.

Binary tree, mesh etc can be embedded in the hypercube.

But: Poor scalability and implementing difficulty for higher dimensional hypercubes.

CM2 – implements hypercube

CM5 – Fat tree

Intel IPSC/1, IPSC/2 are hypercubes

Intel Paragon – 2D mesh

The bottom line for an architecture to survive in future systems is packaging efficiency and scalability to allow modular growth.

New Technologies for Parallel Processing

At present advanced CISC processors are used.

In the next 5 years RISC chips with multiprocessing capabilities will be used for Parallel Computer Design.

Two promising technologies for the next decade :

Neural network

Optical computing

Neural networks consist of many simple neurons or processors that have densely parallel interconnections.

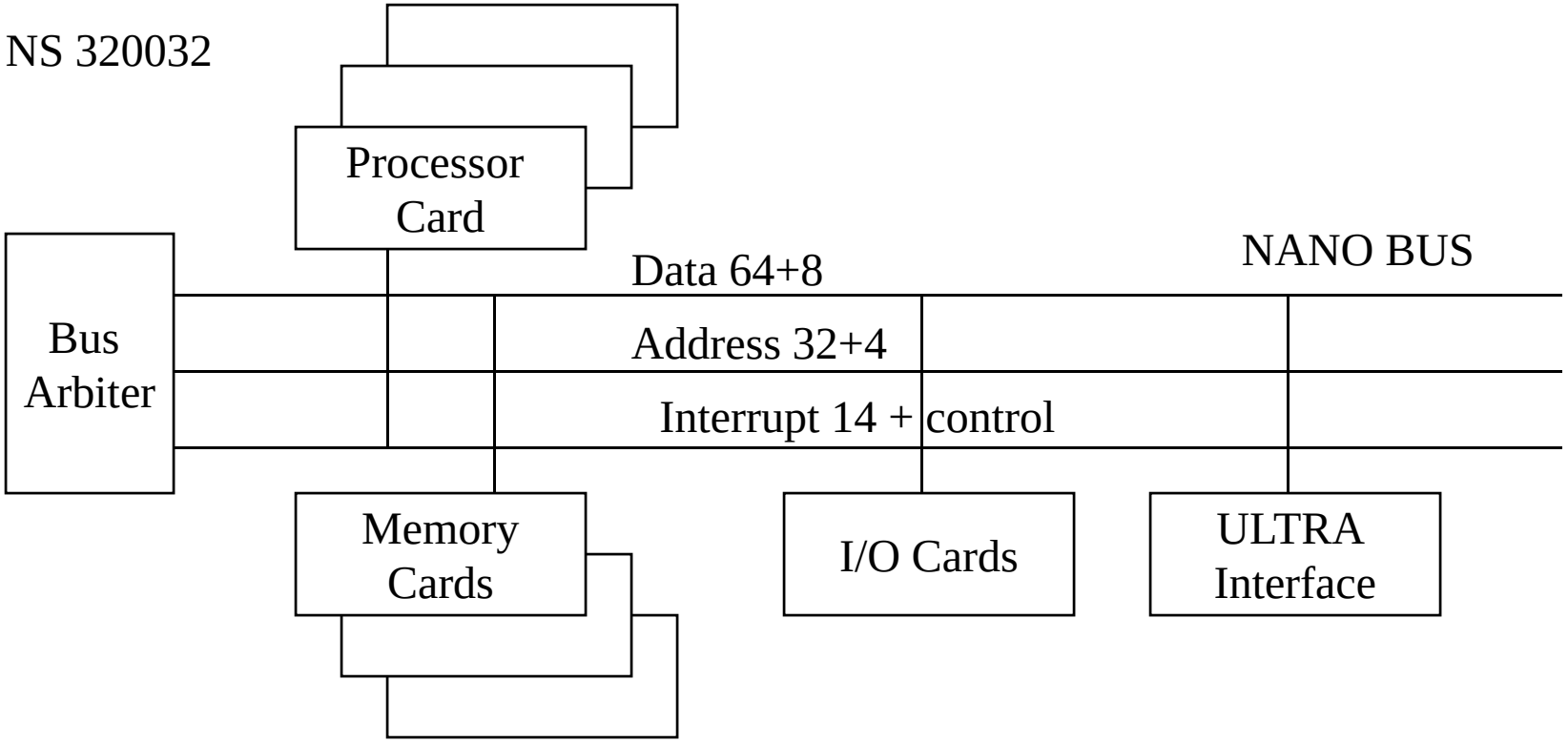
Journals/Publications of interests in Computer Architecture

- Journal of Parallel & Distributed Computing (Acad. Press, 83-)
- Journal of Parallel Computing (North Holland, 84-)
- IEEE Trans of Parallel & Distributed Systems (90-)
- International Conference Parallel Processing (Penn State Univ, 72-)
- Int. Symp Computer Architecture (IEEE 72-)
- Symp. On Frontiers of Massively Parallel Computation (86-)
- Int Conf Supercomputing (ACM, 87-)
- Symp on Architectural Support for Programming Language and Operating Systems (ACM, 75-)
- Symp. On Parallel Algorithms & Architectures (ACM, 89-)
- Int Parallel Processing Sympo (IEEE Comp. Society 86-)
- IEEE Symp on Parallel & Distributed processing (89-)
- Parallel Processing Technology (?) IEEE Magazine

Digital 21064 Microprocessor - ALPHA

- Full 64 bit Alpha architecture, Advanced RISC optmized for high performance, multiprocessor support, IEEE/VAX floating point
- PAL code – Privilieged Architecture Library
  - Optimization for multiple operating system VMS/OSF1
  - Flexible memory management
  - Multi-instruction atomic sequences
  - Dual pipelined architecture
  - 150/180 MHz cycle time
  - 300 MIPS
- 64 or 128 bit data width
  - 75 MHz to 18.75 MHz bus speed
- Pipelined floating point unit
- 8k data cache; 8k instruction cache
- + external cache
- 2 instructions per CPU cycle
- CMOS 4 VLSI, .75 micron, 1.68 million transistors
- 32 floating point registers; 32 integer registers, 32 bit fixed length instruction set
- 300 MIPS & 150 MFLOPS

MIMD BUS

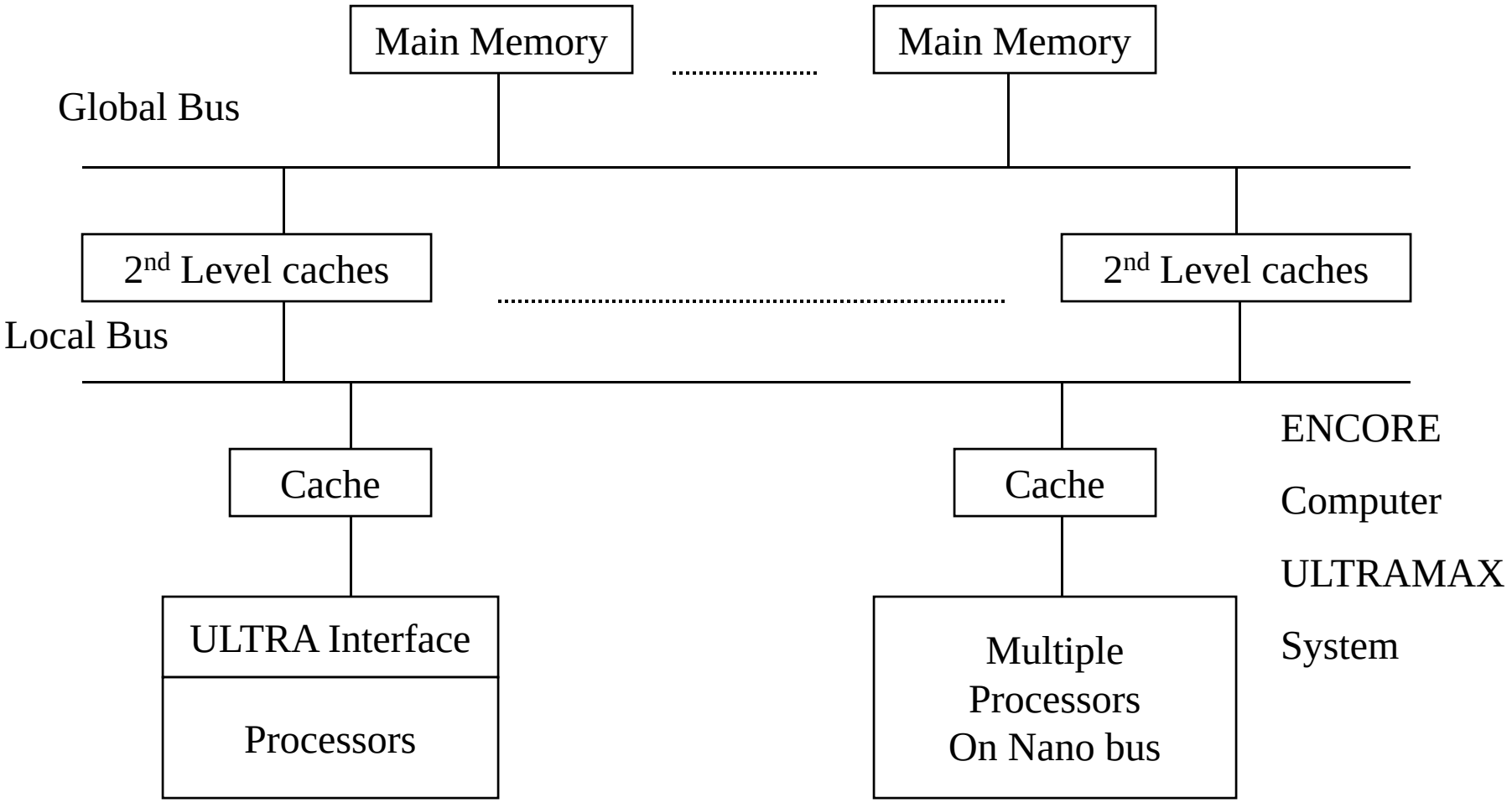

NS 320032
Processor Card
NANO BUS
Data 64+8
Bus Arbiter
Address 32+4
Interrupt 14 +
control
Memory Cards
I/O Cards
ULTRA Interface

MIMD BUS

- Standards :
  - Intel MULTBUS II
  - Motorola VME
  - Texas Instrument NU BUS
  - IEEE)896 FUTURE BUS

- BUS LATENCY
  - The time for bus and memory to complete a memory access
  - Tiem to acquire BUS + memory read or write time including Parity check, error correction etc.

Hierarchical Caching



Multiprocessor Systems

- 3 types of interconnection between processors:
  - Time shared common bus – fig a
  - CROSS-BAR switch network – fig b
  - Multiport memory – fig c

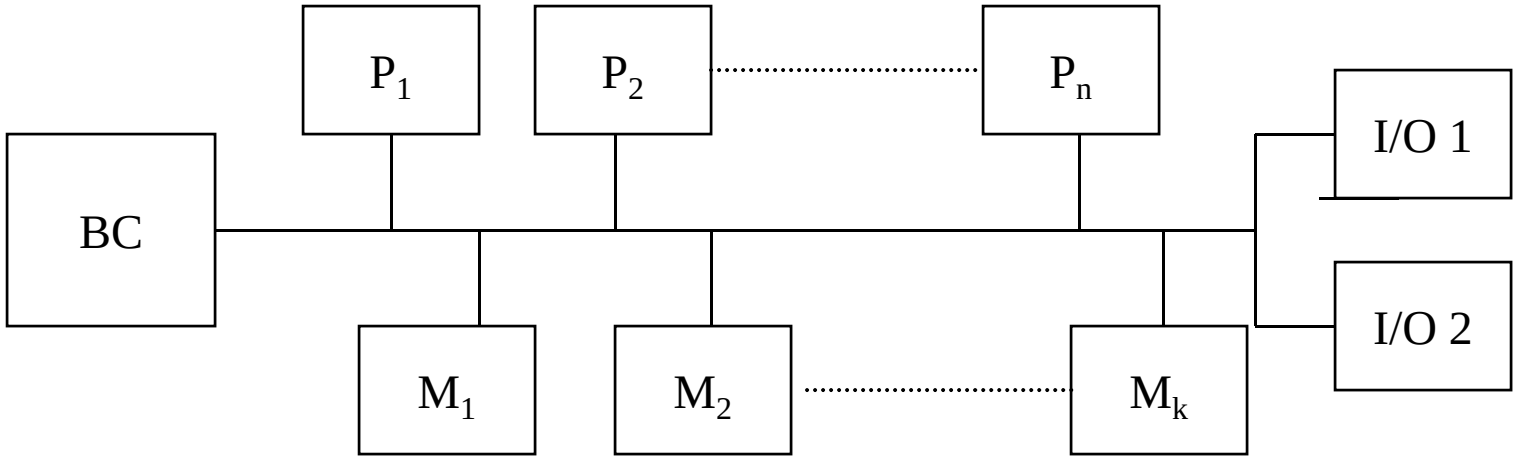


Fig a – Time shared common bus

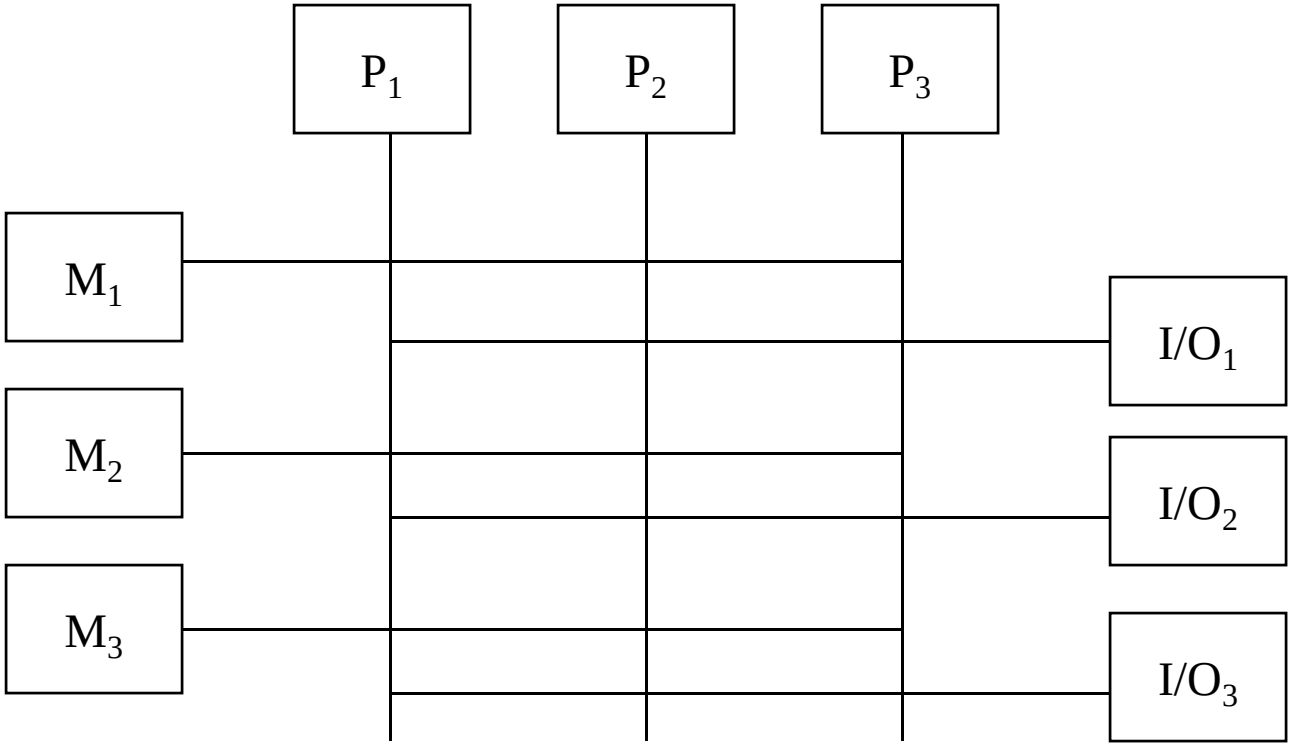


Fig b – CROSS BAR switch network

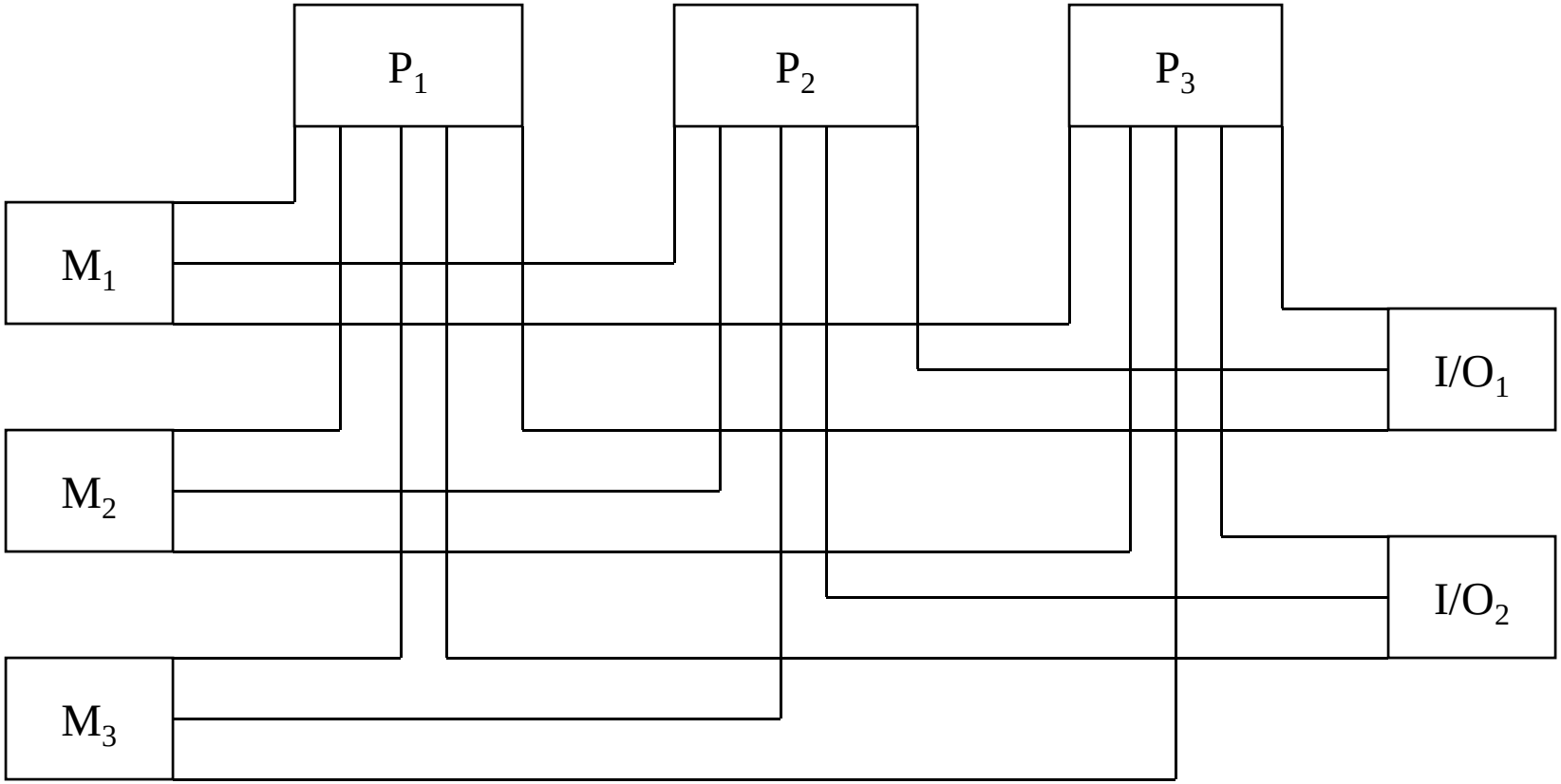


Fig c – Multiport Memory