Thinking Machine Corporation

Formed to exploit parallel processing for Artificial Intelligence.

Connection Machines

- CM-1
  - Designed by Danny Hillis (of MIT).
  - SIMD computer of 4096-65536 Single bit PEs
  - 12D Hypercube of nodes.
  - Each node is:
    - - A chip divided divided 50% packet routing, 50% processing.
    - - On chip there are 16 PEs arranged as a 4x4 grid.
    - - Off chip there are 4 kbits of static memory per PE.
  - Fully configured, the system requires 24576 bi-directional links for communication!
  - Data Parallel Languages: CM Fortran, C* & *Lisp.

Connection Machines

- CM-2
  - Faster PEs
  - More RAM
  - Optional Floating Point Accelerator:
    - - Weitek 3132 chip (14 MFlops).
    - - 1 accelerator per 32 Single bit PEs.
    - - *Unbalanced!* Memory bandwidth to Weitek is only 8 MWords/sec resulting in performance closer to 4MFlops peak per accelerator.
  - Concurrent I/O to Data Vault (array of discs) & High Speed Graphics
  - *In 1989 CM-2 won IEEE Gorgon Bell Award for 5.6 GFlops Sustained Performance on a seismic modelling problem.*

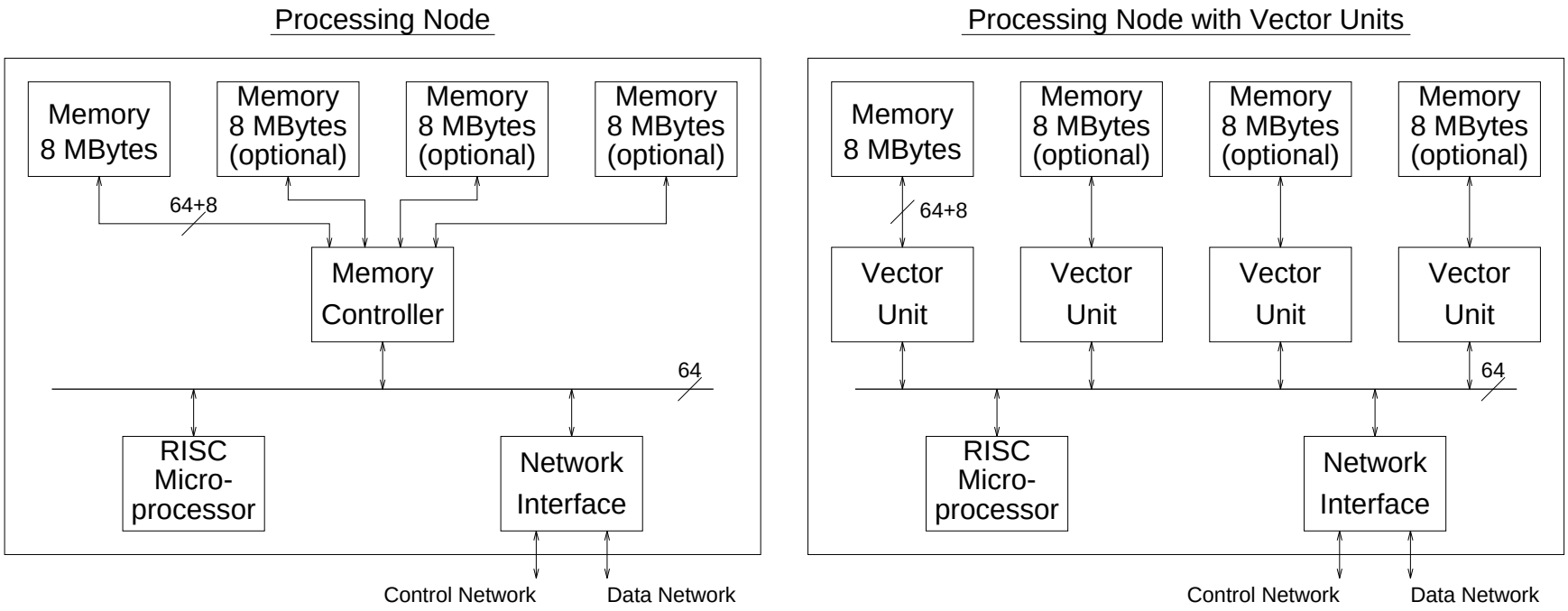
    (8-Pipe Cray Y-MP has a peak rating of 3 GFlops!).

CM-5 A Usable TerraFlop Computer

- Power
  - 16384 Nodes
  - Each node 128 MFlops.
  - Total: 2 TerraFlops
- Memory Bandwidth
  - Up to 512 MBytes/sec per node
- Communications
  - 20 MBytes/sec per node for nearest neighbour communications (over simulated 2D Grid).
  - Worst case provides 5 MBytes/sec per node.
- Programming Paradigm
  - *Data Parallel on an MIMD machine!*

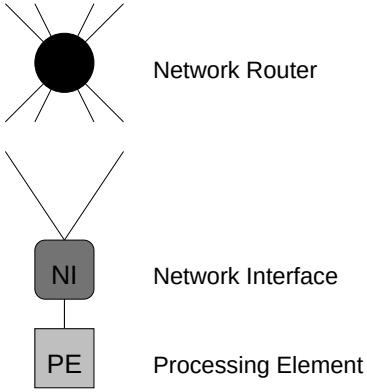
Connection Machine CM-5

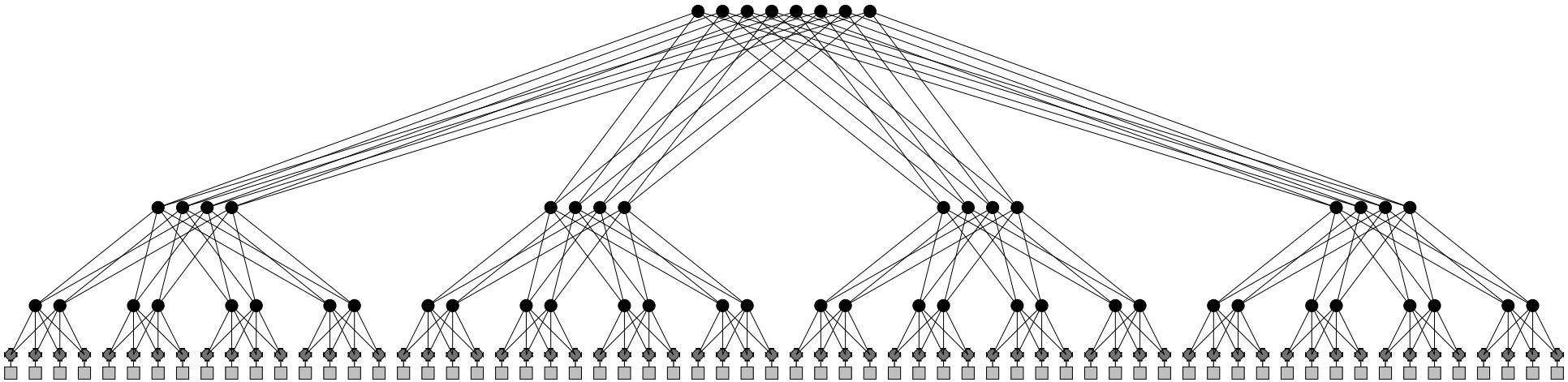
PE Architecture



- RISC microprocessor is a SPARC.
- No vector nodes have yet been built.

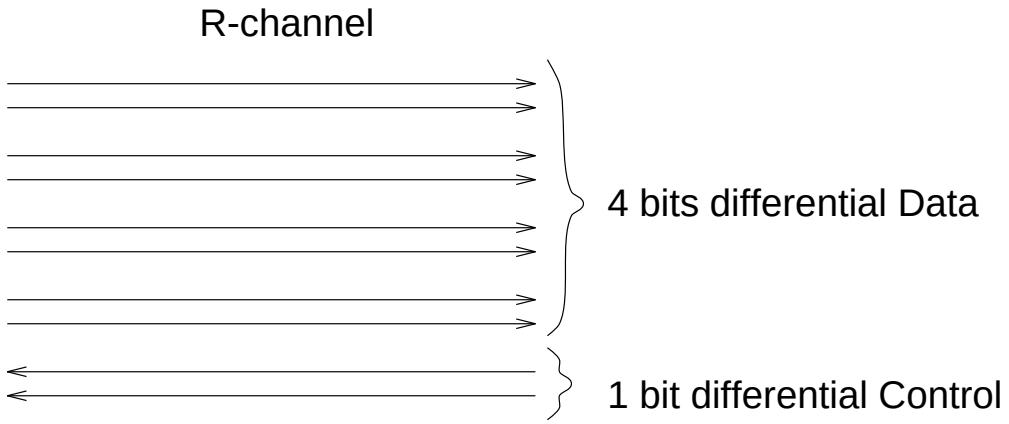
CM-5 Architecture (Data Network)





Inter-node Links

- Each bi-directional link consists of two R-channels, one in each direction.
- An R-channel consists of 4 differential pairs for data and 1 differential pair for control.



- Communication is synchronous - *all PEs and routing nodes share the same clock!*

Routing

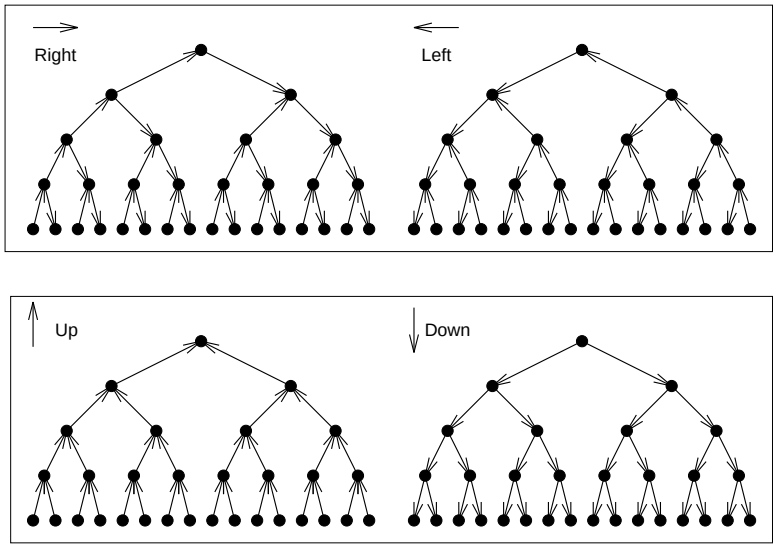
- Virtual Cut-Through Routing.

- Routing strategy:
  - Normal trees do not permit adaptive routing - there is no route choice!
  - Fat trees allow adaptive routing as the message travels up towards the root, but there is no route choice as the message travels down towards the destination leaf.
  - Where there is choice, the message will take any available channel.

Deadlock Avoidance

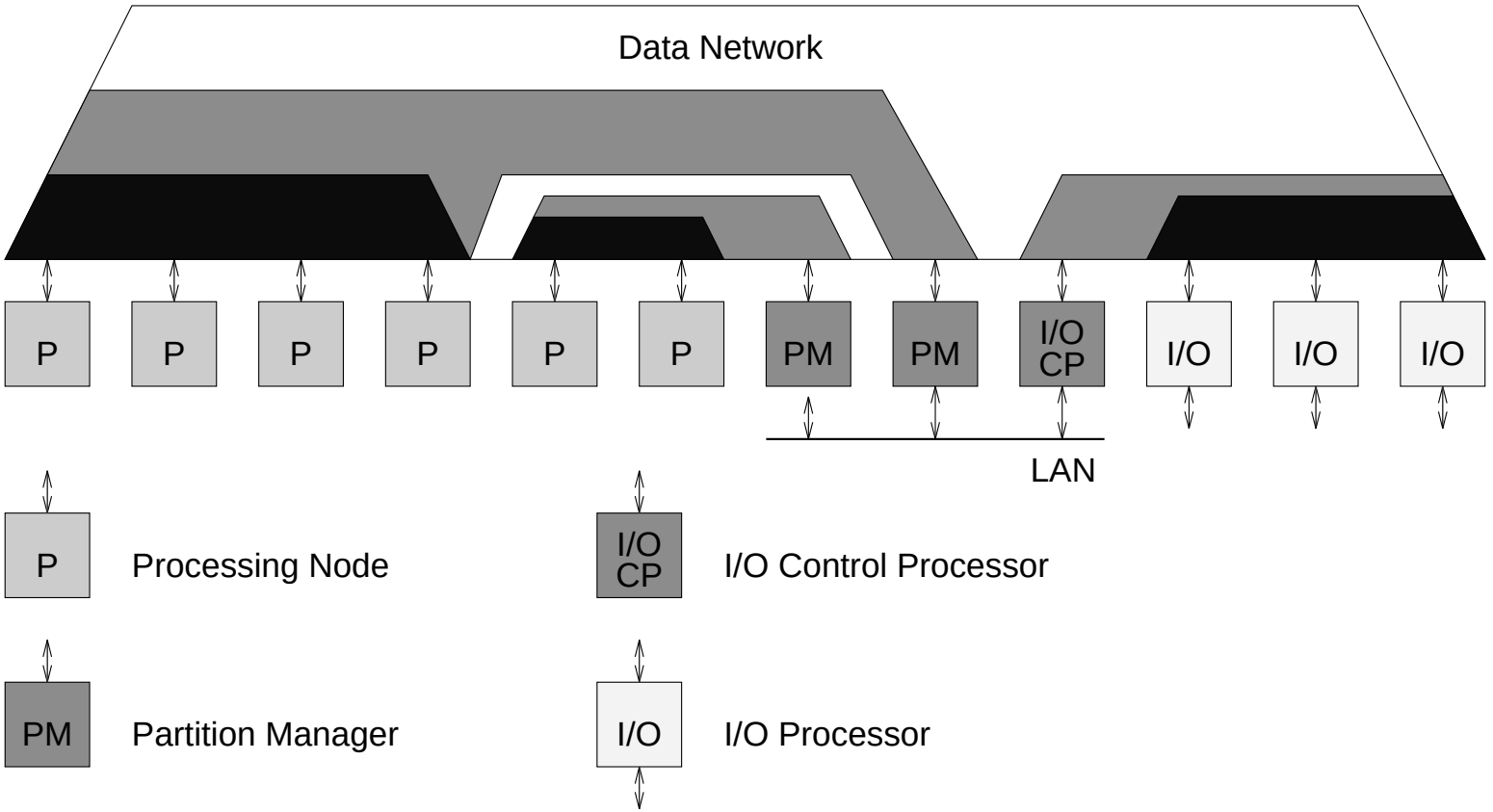
Virtual Networks

- We need only two virtual networks in order to avoid cycles of undelivered messages, for any type of tree or fat tree.
- We can use one of two sets of virtual networks:



- Having provided sufficient buffering there is no need for any additional restrictions on routing.

Partitions & I/O



Data Parallel Programming

- Although each *Processing Node* is capable of independent *Process Parallel* type operation, the software for CM-5 supports a *Data Parallel* SIMD type programming style.
- Code is written in the same Data Parallel languages; CM Fortran, C* & *Lisp.
- The processing nodes can be divided into partitions.
- Each partition is controlled by a *Partition Manager*
  - a standard SPARCstation with a *Network Interface* and software to drive it.
- Each Partition Manager may execute a number of *Data Parallel* processes[1] on its own partition, any of which may communicate with a process from another partition.

[1]The different processes are timesliced in a UNIX type fashion.