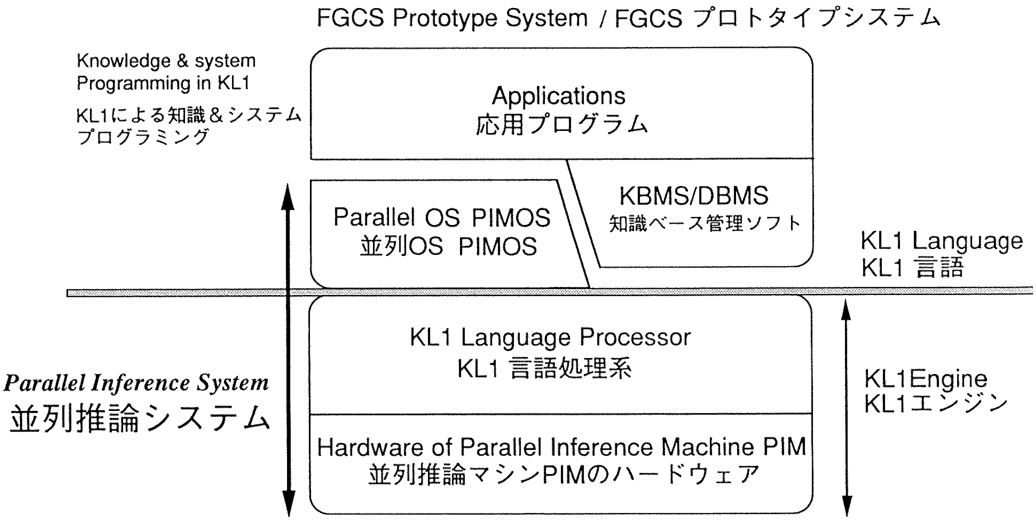
Parallel Inference System

Abstract

A parallel inference system provides the basic portion of the FGCS prototype system. The parallel inference system can efficiently solve, in parallel, symbolic and knowledge processing problems written in the kernel language, KL1.

System Components

- The kernel language KL1 is a general-purpose concurrent logic programming language which has the descriptive power and functions needed to easily describe and solve symbolic and knowledge processing problems in parallel.
- PIMOS provides a KL1 programming environment in addition to conventional OS functions.
- The design of PIM hardware architecture is well-balanced to efficiently execute KL1 from a parallel machine point of view.
- The KL1 language processor is implemented with a low parallelization and decentralization overhead.
- At the application layer, adequate speedup is obtained by dynamic and static load distribution and speculative work.


FGCS Prototype System / FGCS プロトタイプシステム
Knowledge & system Programming in KL1
KL1による知識&システムプログラミング
Applications
応用プログラム
Parallel OS PIMOS
並列OS PIMOS
KBMS/DBMS
知識ベース管理ソフト
KL1 Language
KL1 言語
KL1 Language Processor
KL1 言語処理系
Parallel Inference System
並列推論システム
Hardware of Parallel Inference Machine PIM
並列推論マシンPIMのハードウェア
KL1Engine
KL1エンジン

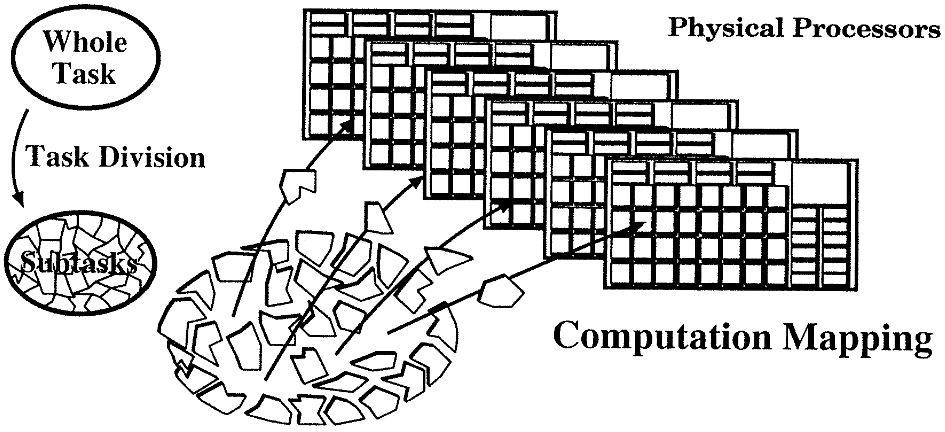
Parallel Logic Programming Language "KL1"

ABSTRACT

KL1 is the "Kernel Language", based on Guarded Horn Clauses, which gave the design principles of the whole parallel inference system, from parallel inference machine hardware to application software.

KEY FEATURES

- Allows easy description of fine-grain parallel processing, dividing the whole task into small subtasks.
- Used throughout the system, from the operating system to application software.
- Clear separation of program meaning from program processing, making load distribution much easier.
- The same language is used in all models of PIM, providing high software portability.



Parallel Inference Machine Operating System, PIMOS

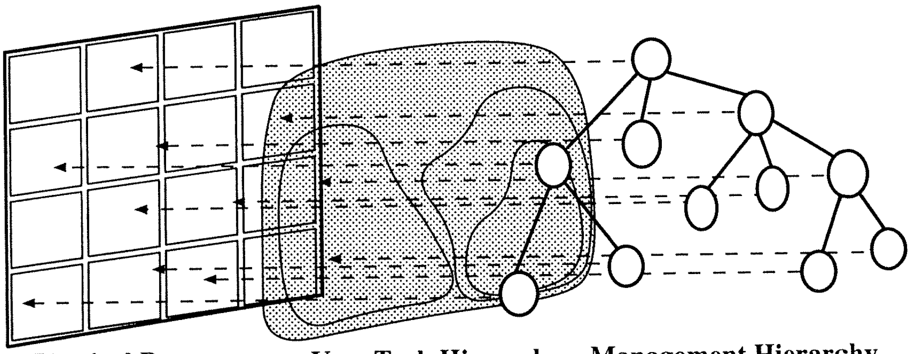
ABSTRACT

PIMOS is the common operating system for the parallel inference systems, PIM and Multi-PSI, and provides an efficient and comfortable software development environment for the systems.

KEY FEATURES

- Written completely in the KL1, a concurrent logic language.
  Realizing high portability.
- Removes management job bottlenecks with its distributed hierarchical management scheme.
- Provides powerful software development tools.
  Debugging tools, load distribution visualization, and etc.
- Tested through in parallel software R&D for more than four years.

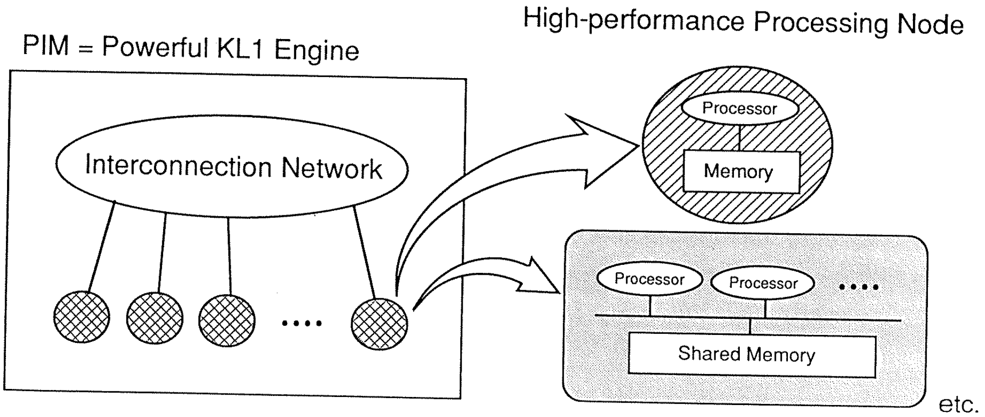
All the systems demonstrated on PIM or Multi-PSI have been developed and are running on PIMOS.



Parallel Inference Machine, PIM

What is PIM?

- **General-purpose Machine**
  PIM is an MIMD machine which can efficiently execute the KL1 general-purpose high-level language.
- **Parallel Machine with about 1000 Processors**
  Scalable architecture consisting of high-performance processors
  Adequate speedup is obtained with about 1000 processors.
- **Inference Machine**
  Dedicated instructions and hardware for efficient implementation of the concurrent logic language KL1 (e.g., dereference instruction and tag architecture)
- **Performance**
  A processing element of PIM/p and PIM/m yields 300 ~ 600 KRPS (append). Thus, a full size system of PIM/p (512 PEs) achieves 250 MRPS (append) approximately.
  250 MRPS corresponds to about 3.6 GIPS.
- **Five Modules**
  In order to examine various PIM architectures, Five modules are being developed: PIM/p, PIM/m, PIM/c, PIM/i and PIM/k.



PIM Architecture

KL1 Language Processor

Abstract

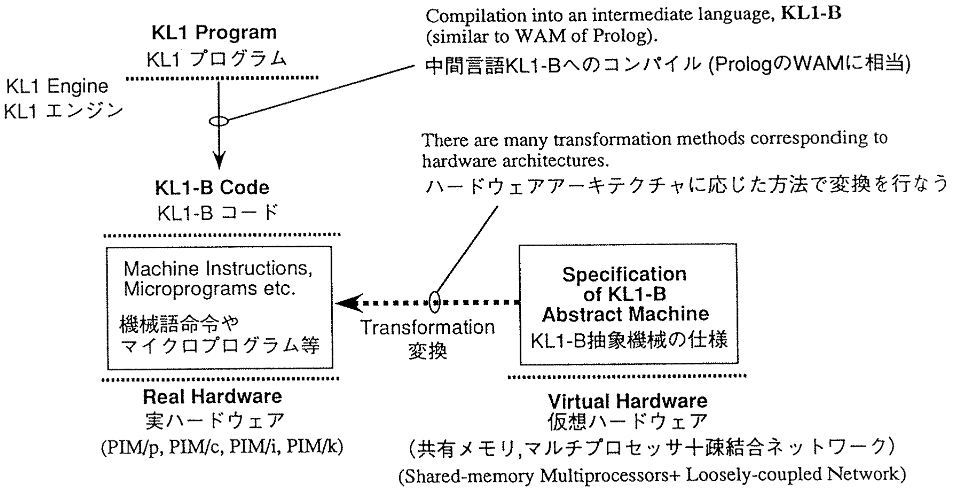
Thc KL1 language processor is software to efficiently implement a common identical KL1 interface on PIM modules with different architectures.

Functions and Features

- Our method compiles into an intermediate language, similarly to WAM of Prolog. This method is easy to develop and has high portability.
- The specification of the abstract machine for an intermediate language is transformed into machine instructions and microprograms according to the hardware architecture.
- The transformation of the abstract machine specification into a C program allows easy simulation and debugging on conventional machines.

Framework for Executing a KL1 Program

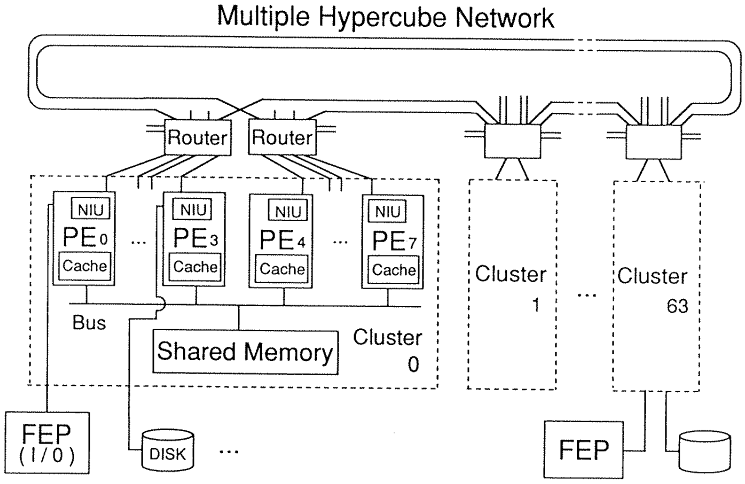
- A KL1 program is compiled into an intermediate language KL1-B.
- KL1-B codes are executed on an abstract machine.
- The abstract machine is described as a runtime system on virtual hardware.
- The virtual hardware supposes that shared-memory multiprocessors are connected by loosely-coupled networks.

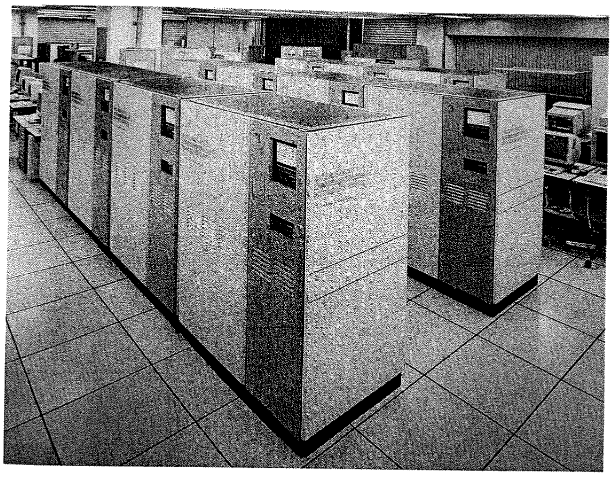

KL1 Program
KL1 プログラム
KL1 Engine
KL1 エンジン
Compilation into an intermediate language, **KL1-B**
(similar to WAM of Prolog).
中間言語KL1-Bへのコンパイル (PrologのWAMに相当)
KL1-B Code
KL1-B コード
There are many transformation methods corresponding to hardware architectures.
ハードウェアアーキテクチャに応じた方法で変換を行なう
Machine Instructions, Microprograms etc.
機械語命令や
マイクロプログラム等
Transformation
変換
Specification of KL1-B Abstract Machine
KL1-B抽象機械の仕様
Real Hardware
実ハードウェア
(PIM/p, PIM/c, PIM/i, PIM/k)
Virtual Hardware
仮想ハードウェア
(共有メモリ,マルチプロセッサ十疎結合ネットワーク)
(Shared-memory Multiprocessors+ Loosely-coupled Network)

PIM/p

- Two-level hierarchical structure – a six-dimensional hypercube network connects clusters, each of which contains eight processors sharing a memory unit.
- KL1-oriented snoop caches which realize low latency communication and synchronization
- Enhanced instruction set by macro calls

- 2階層の構成 – 6次元ハイパーキューブネットワークがクラスタを接続し，各クラスタには1台のメモリを共有する8台のプロセッサがある．
- KL1向きスヌープキャッシュによる高速な通信と同期
- マクロ呼び出しによる命令の高機能化



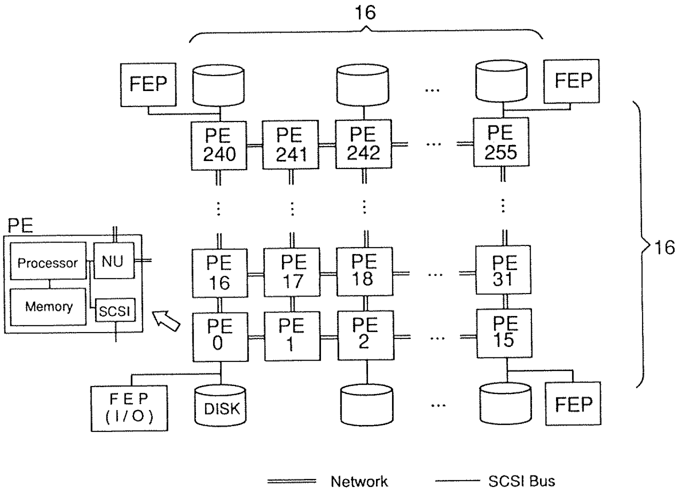


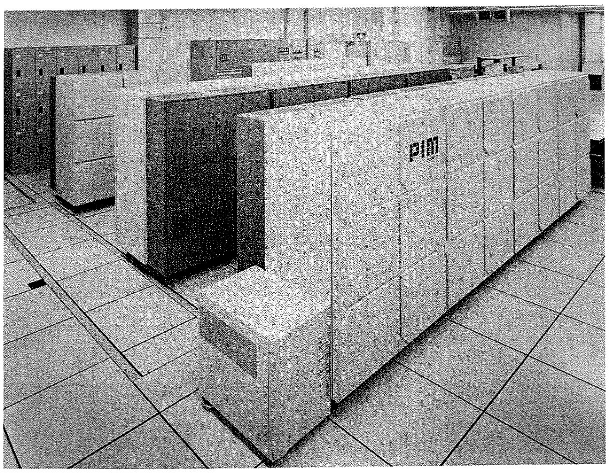
PEs/cabinet	32
cabinets/system	16
Total PEs	512
Cabinet size (m)	1.4×0.8×1.6

PIM/m

- Inherits Multi-PSI's architecture and firmware
- A single layer network and a node consisting of a CPU make for simple architecture and high scalability.
- Capable of examining various parallel processing techniques, such as task division and mapping

- **Multi-PSI** アーキテクチャ及びファームウェアの継承
- **1** 階層ネットワーク, **1** ノード **1 CPU** のシンプルなアーキテクチャで高い拡張性
- 問題分割, マッピングなどの並列処理技術の研究開発に向いている.

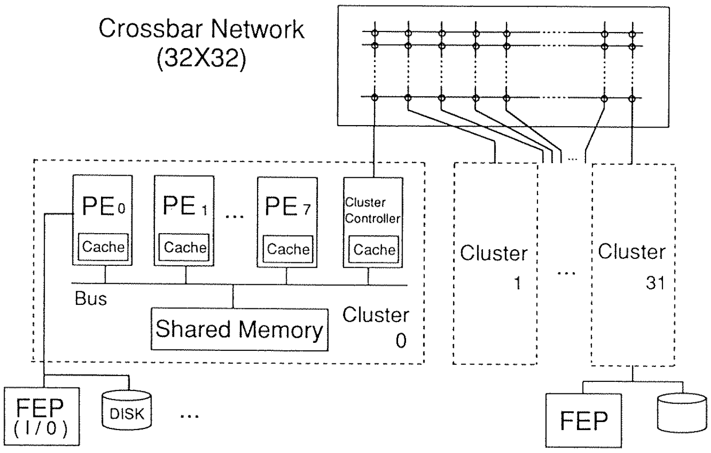


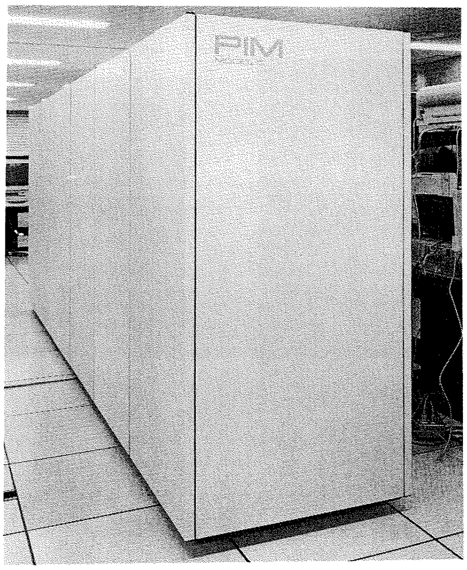


PEs/cabinet	32
cabinets/system	8
Total PEs	256
Cabinet size (m)	1.1×0.9×1.5

PIM/c

- Contains eight processing elements which employ horizontal microprogramming control and are tightly coupled
- Dedicated hardware transmits global scope variable (e.g., load information) between clusters with low latency.
- A cluster has high-speed KL1-oriented snoop caches and registers with broadcast facility
- 水平型マイクロプログラム方式の要素プロセッサ 8 台を密結合
- 負荷情報などの大域変数をクラスタ間に渡って低遅延で転送する専用ハードウェア
- クラスタ内には KL1 向きの高速なスヌープキャッシュと放送機能付きレジスタ



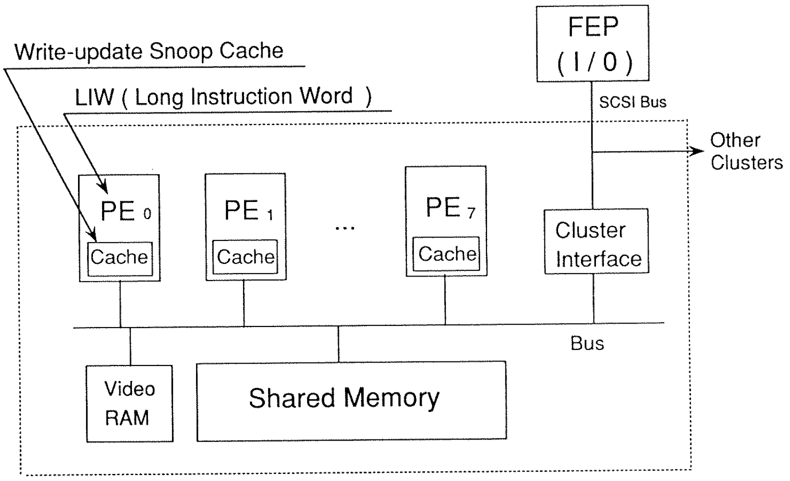


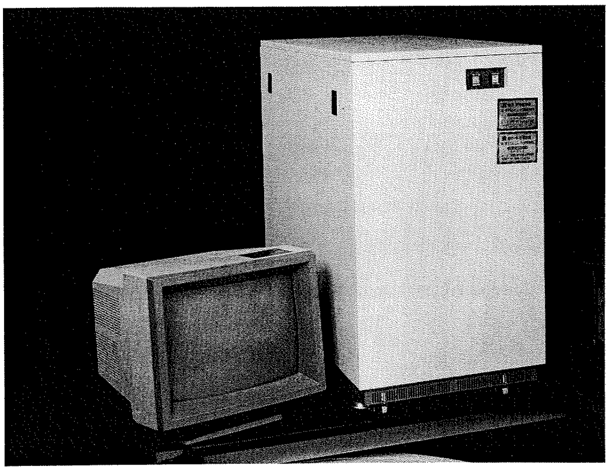
PEs/cabinet	64
cabinets/system	4
Total PEs	256
Cabinet size (m)	1.4×0.8×1.7

PIM/i

- Write-update snoop caches and LIW (long instruction word) are introduced for efficient execution of KL1 programs.
- CIF (cluster interface) works as an I/O processor.
- Effective system status monitoring by video RAM

- KL1 の効率的な実行のために書き込み更新型スヌープキャッシュと LIW (長形式命令語) を導入
- 入出力専用プロセッサとして CIF (クラスタインタフェース) を持つ
- ビデオ RAM による効果的なシステム状態の監視



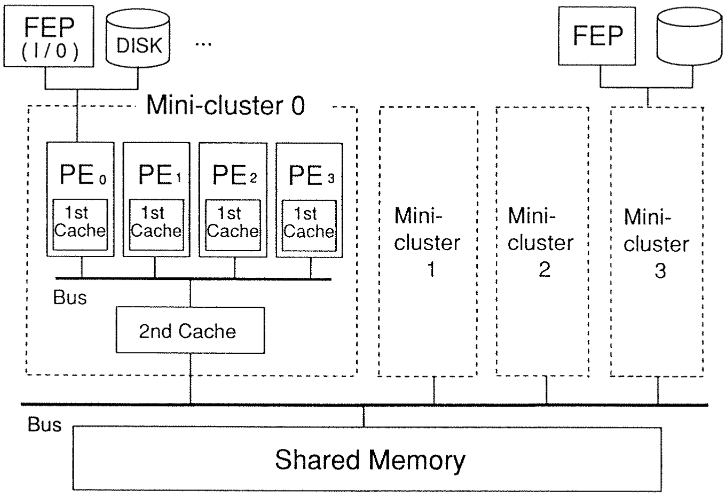


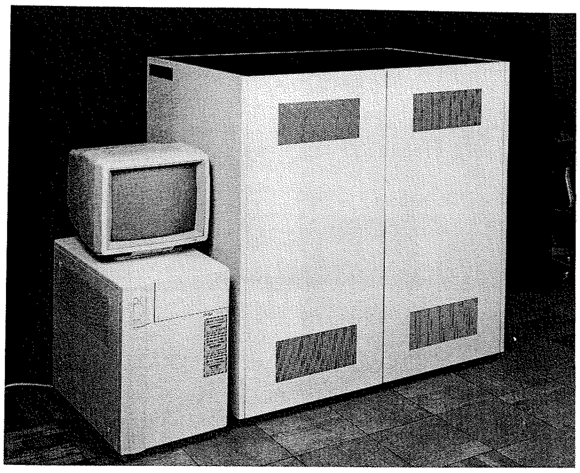
PEs/cabinet	8
cabinets/system	2
Total PEs	16
Cabinet size (m)	0.5×0.7×0.7

PIM/k

- Examines the scalability attained by multi-layer cache
- Experiments on multi-layer caches and load balancing management appropriate to KL1 execution
- Easy implementation of a KL1 language processor on a UMA (uniform memory access) architecture

- 多階層キャッシュによる拡張性の追求
- KL1 実行に適した多階層キャッシュ制御や負荷分散の実験
- UMA (一様メモリアクセス) アーキテクチャに対する KL1 処理系実装の容易さ



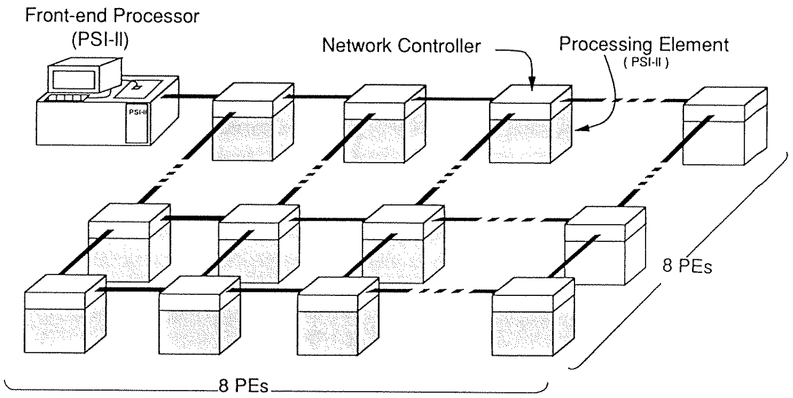


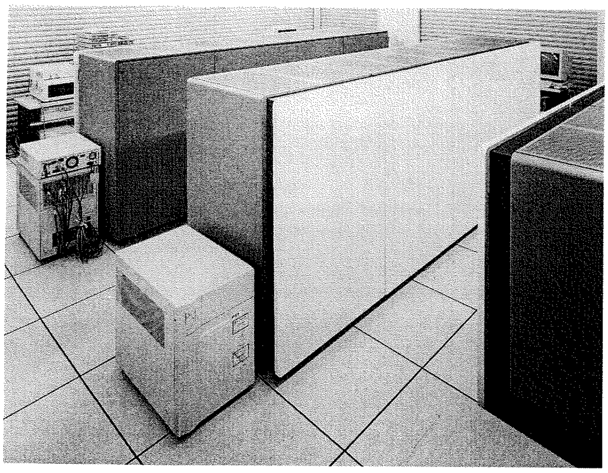
PEs/cabinet	16
cabinets/system	1
Total PEs	16
Cabinet size (m)	1.3×0.8×1.3

Multi-PSI

- A prototype of PIM developed in the intermediate stage of the FGCS project.
- Research on the KL1 parallel execution method, research on the parallel operating system, and R&D on application programs have been done on Multi-PSI.
- The CPU of the PSI sequential inference machine have been employed as the processing element and only a dedicated network controller had to be developed from scratch.

- プロジェクト中期に開発された PIM のプロトタイプ.
- **KL1** の並列実行方式の研究, 並列 OS の研究, 多くの応用ソフトウェア開発や実験が行なわれた.
- **PSI** の **CPU** を要素プロセッサとして用い専用のネットワーク制御装置のみ新規開発した.





PEs/cabinet	8
cabinets/system	8
Total PEs	64
Cabinet size (m)	0.9×0.8×1.4