

並列画像生成システム LINKS-2 のアーキテクチャ†

河合利幸†† 山下伸一††† 大野廣司†††
 吉村浩††† 西村仁志†††† 下條真司†††††
 宮原秀夫††††† 大村皓一††

コンピュータグラフィックスの普及に伴い、高品位な画像をより高速に生成することが要請されている。我々は、視線探索法を用い、並列処理により画像を生成するシステム LINKS-1 を試作し、数多くの実験を重ねてきた。この結果を基に、飛躍的な処理速度の向上を目指し、新たなシステムを設計した。本システムのユニットコンピュータ (UC) は、LINKS-1 の UC と比較して、演算能力と通信機能が大幅に強化されている。本 UC は、32 ビット浮動小数点演算を並列処理するデータプロセッサ、アドレス計算を行うインデックスユニット、通信を制御するチャンネルプロセッサ等主要ユニットが非同期に並列パイプライン動作する。メモリアクセス競合を削減し、データ転送を並列化するため、各ユニットはクロスバースイッチにより、4 ウェイにインターリーブされたデータメモリと結合されている。これら主要部分は、カスタム VLSI を用いて実現した。その結果、本 UC 1 台の画像生成速度は LINKS-1 の約 50 倍に達する。また、並列処理効率についても改善が認められた。本論文では、その高速化の手法について述べ、シミュレーションによりその効果を明らかにする。

1. まえがき

コンピュータグラフィックス (CG) は、計算機内に物体の 3 次元モデルを構築することにより、その形状、色彩、質感、運動などをディスプレイ上に表示する技術である。工業デザイン、シミュレーション、視聴覚教育等、応用分野の拡大・普及に伴い、高品位な画像をより高速に生成することが要請されている。

この要求に、最近ではアルゴリズムに内在する並列処理の可能性に着目して、マルチプロセッサによる専用マシンの研究・開発が各所で行われるようになってきた^{1), 5)-7)}。

我々は、リアリズム重視の立場から、反射・屈折などの表現が容易であり、高品質な陰影画像を得ることができる視線探索法^{2), 3)}を用いた並列画像生成システム LINKS-1 を試作した^{8), 12)}。LINKS-1 は、本格的な画像生成専用システム開発のための実験システムと

して設計されており、1982 年 10 月の試作完成以来、安定に動作し、今日までに数多くの実験データが得られている⁹⁾⁻¹²⁾。

これらの実験結果に基づき、新たな並列画像生成システム LINKS-2 の設計・製作を開始した。LINKS-2 は、実用的な対話型 CG システム構築に必要な処理能力の達成を目標として設計された。1987 年 8 月現在、カスタム VLSI、基板、および言語処理系の作成が完了し、動作試験が進められている。

本論文では、LINKS-1 における問題点を明らかにし、新たな画像生成システムの必要性を示す。また、LINKS-2 の設計に際し、これらの問題点をいかに解決したか述べ、シミュレーションによりその効果を示す。

2. 問題の所在

2.1 視線探索法の特徴

視線探索法による画像生成の特徴を以下に示す。

(1) 影・反射・透過・屈折を一貫して処理できるだけでなく、画素ごとに全く独立に処理できる。

(2) 画素間のコヒーレンスが使えず膨大な計算量を必要とする。

(3) 各画素における処理は、どの物体表面がその画素に見えるかという処理 (交差判定) と、その物体表面の輝度を求める処理 (輝度計算) からなる。

(4) 上記の計算過程ではたかだか 3 次か 4 次の行列・ベクトル演算*もしくはスカラ演算だけから成

† Architecture of Parallel Image Generation System LINKS-2 by TOSHIYUKI KAWAI (Department of Electronic Engineering, Faculty of Engineering, Osaka University), SHIN'ICHI YAMASHITA, HIROSHI OHNO, HIROSHI YOSHIMURA (Toyo Links Co., Ltd.), HITOSHI NISHIMURA (Nishimura Technical Laboratory, Ltd.), SHINJI SHIMOJO, HIDEO MIYAHARA (Department of Information and Computer Sciences, Faculty of Engineering Science, Osaka University) and KOICHI OMURA (Department of Electronic Engineering, Faculty of Engineering, Osaka University).

†† 大阪大学工学部電子工学科

††† (株) トーヨーリンクス

†††† (株) 西村技研

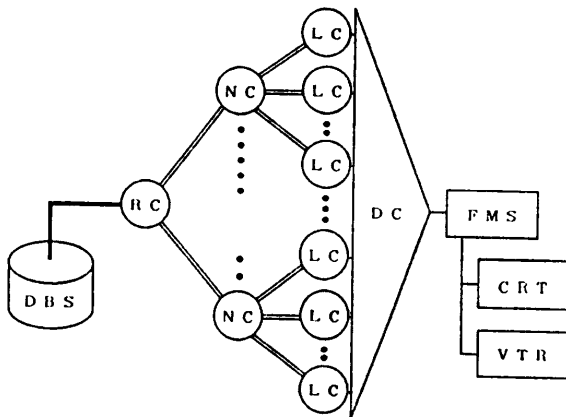
††††† 大阪大学基礎工学部情報工学科

り、処理の流れがデータに強く依存する。このため、ベクトル化率が低く CRAY-1⁴⁾に代表されるパイプラインコンピュータや、G-PSYCO⁵⁾に代表される SIMD 型マシン向き処理ではない。

以上から、視線探索法による画像生成を高速化するには、各画素の処理を並列に行う MIMD 型のシステムが有効である。しかも、各画素を処理するプロセッサ間で同期をとる必要がないため、システムを疎結合構成にすることができ、プロセッサ数に見合った安定した性能を得ることができる。

2.2 LINKS-1 の概要

前節での考察に基づき試作した LINKS-1 は、星状構造もしくは木構造の MIMD 型並列処理システムである (図 1)。各画素の画像生成を行う LC (Leaf Computer), LC への負荷 (画素) の割り当てを行う RC (Root Computer) および NC (Node Computer) より構成される¹²⁾。RC, NC および LC は同一構成の UC (Unit Computer) よりなる。UC 間は IMSU (Intercomputer Memory Swapping Unit) により結合されている。IMSU は、別々の UC の主記憶の一部となる 2 組のメモリブロックをバススイッチにより交換するもので、非同期通信を実現する。LC により計算された画素データは DC (Data Collector) により集められ、FMS (Frame Memory System) に出力される。



DBS : Data Base System RC : Root Computer
DC : Data Collector NC : Node Computer
FMS : Frame Memory System LC : Leaf Computer
== : Intercomputer Memory Swapping Unit

図 1 LINKS-1 のシステム構成
Fig. 1 System configuration of LINKS-1.

* ただし LINKS-1 では同次座標を用いていないので 4 次の行列・ベクトル演算はほとんどない。

2.3 LINKS-1 における問題点

LINKS-1 の今日までの運用経験や実験から、以下に述べるような問題点が指摘されている。

(1) 浮動小数点演算速度

C言語で書かれている画像生成プログラムをコンパイルし、その出力であるアセンブリ命令を解析した結果、単精度浮動小数点演算のアセンブリ命令数は画像生成プログラムの全アセンブリ命令のうちわずか 8.7% にすぎないが、その処理時間は全処理時間の 60% にも達する¹⁵⁾。この原因の一つは、制御用プロセッサ Z8001 と算術演算用プロセッサ 8086/87 両者間の通信のオーバーヘッドである。同様な問題が類似のシステム MC-1 でも見られる⁶⁾。

(2) UC 間通信方式

LINKS-1 では、画面を LC 台数より十分多くの小画面に分割し、その処理を各 LC に順次割り当てることによって負荷分散を行っている (画面分割法)。LC のメモリにすべてのデータが書き込める場合は、処理開始時に全物体データを各 LC に転送する (バッチ転送方式)。この場合は、小画面の要求と割り当て以外の UC 間通信が全く無く、通信のオーバーヘッドが極めて少ないため、LC の台数に見合う処理速度の向上が達成された¹¹⁾。

しかし、物体データ量が LC のメモリ容量を超える場合は、小画面の処理に必要な時点で、その物体データを LC に転送する (デマンド転送方式)。この方式では、データ量の制限はないが、通信のオーバーヘッドが増大するため、LC をある台数以上増やしても、それ以上処理速度が向上しなくなる。この台数は、LC の内部処理時間と UC 間の通信処理時間との比により決まる¹¹⁾。

2.4 LINKS-2 の必要性

LINKS-1 では、ハードウェアの変更なしにその演算能力を改善することは不可能であり、また、ソフトウェアの改良による高速化にも限界がある。

LC の台数を増やすことは、前節(2)で述べたような問題があり、現実的には、LC 64 台程度の星状構造システム、もしくは LC 256 台程度の木構造システムが限界である。さらに多くの LC を有効に稼働させるためには、演算処理速度の向上より以上の通信処理速度の向上が不可欠である。

以上の理由から、(1)単精度浮動小数点演算の高速化、(2)UC 間の通信機能の改良、高速化、により、大規模な並列処理システムの実現を目標として、新た

な UC を設計・製作する。高速化, 安定化, 小型化を図るためにゲートアレイカスタム VLSI を用いて実現した。

3. LINKS-2 の演算処理方式

3.1 画像生成処理における演算の特徴

LINKS-1 で稼働中の画像生成プログラムは, 大量のデータに対し項数の少ない演算を繰り返しており, 配列や構造体の要素など複雑なアドレス計算を要するメモリ参照が極めて多い。したがって, レジスタ割当ての最適化による演算速度の改善には限界がある上, メモリ-レジスタ間のデータ転送が演算高速化の隘路となる。それゆえ, RISC に代表されるレジスタ-レジスタ演算主体のプロセッサ向きではない。

さらに, 前述したように, スカラ, 3次元ベクトル演算が主体で, 処理の流れがデータに強く依存するためベクトル化率を上げることが難しく, ベクトルプロセッサ向きではない。

したがって, 簡素なハードウェアによる実現という観点から, スカラプロセッサを用い, 特に多用されている整数の算術演算, 論理演算, 単精度浮動小数点の四則演算, 開平, 平方根の逆数演算, 3次元ベクトル・行列演算, 型変換を高速化することが望ましい。このためには, ベクトル化が困難な多数のデータをいかに演算器に供給し, 結果を格納するかというデータ転送方式が, 演算機構の方式以上に重要となる。

116bit						
Opcode	Addressing mode			Effective address		
	Src1	Src2	Dst	Src1	Src2	Dst
8bit	4bit	4bit	4bit	32bit	32bit	32bit

Src1, Src2 : Source operands Dst : Destination operand

図 2 LINKS-2 の命令フォーマット

Fig. 2 Instruction format of LINKS-2.

3.2 UC 内部のデータ転送方式

LINKS-2 の UC では, 単一の演算器に高速に多量のデータを供給するため, アドレス生成, データ転送を極力並列化する。汎用レジスタをなくす代り, 転送のオーバーヘッドを減らすことにより, 全メモリをレジスタと同等に扱うことを可能にする。以下の手法を用いて実現した。

(1) データ型は浮動小数点数, 整数, ポインタ(すべて 32 ビット)のみとし, アドレス演算を単純化する。

(2) 2入力と 1出力アドレスを含むマイクロプログラム方式(図 2)とし, 1命令で 2項演算を実行する。プログラムはすべてこの命令形式に展開される。

(3) 3組の 32 ビットバスを設け, メモリ, 演算器にアドレス, データを並列供給する。

(4) アドレス演算専用処理ユニットを 3組設けて, 入出力アドレスを独立に計算する。

(5) データメモリとプログラムメモリを分離し, 命令・データのフェッチを独立に行う。

(6) データメモリは 4つのモジュールを用いたイ

- DP : Data Processor
- DPU : Data Processing Unit
- MPY : Multiplier
- IXU : Index Unit
- MM : Memory Module
- CHP : Channel Processor
- CHPU : Channel Processing Unit
- LCB : Links Communication Bus
- SEQU : Sequencer Unit
- μp-MEM: Micro Program Memory
- np-MEM: Nano Program Memory

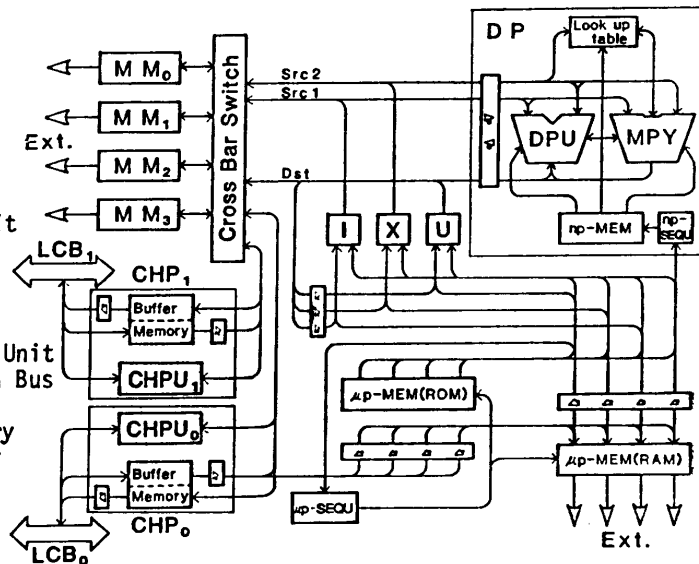


図 3 ユニットコンピュータのハードウェア構成

Fig. 3 Hardware configuration of unit computer.

インタリーブ構成とし、クロスバスイッチを介して3組のバスと接続する。アドレス空間は32ビットリニアアドレスとした。

図3にUCの構成を示す。本UCのアーキテクチャ最大の特徴は、データメモリの構成と、クロスバスイッチによる結合方式にある。以下、それぞれについて記述する。

3.2.1 データメモリの構成

図4に示すように、3次元ベクトルの各成分を順に4つのモジュールに分配すれば、連続した4ベクトルの同一成分はすべて異なるモジュールに置かれる。したがって、局所的に定義された3次元ベクトル間で演算を行う場合、被演算成分の演算器への供給および演算結果の格納が並列化できる確率が高くなる。ただし演算結果については、処理をパイプライン化することで、その格納を次の被演算成分の供給と同時に進行。

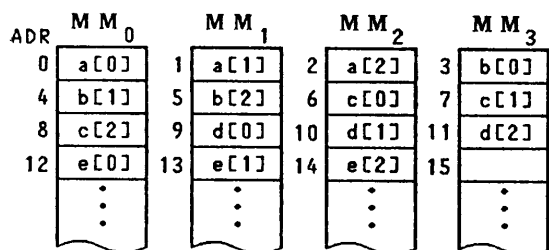
4つのモジュールに等確率でランダムに配置されたスカラデータに対し2項演算を行った場合でも、平均1.69回のアクセスで演算を終了できる。

加えて、残り1つのモジュールのアクセスが競合なく行えるので、演算処理と並行してUC間のデータ転送が効率よく行える。

アドレスは、4つのモジュールにインタリーブして割り付けた(図4)。アドレスの下位2ビットがモジュールを選択する。C言語の自動変数はスタック上にとられ、実行時にそのアドレスが決定される。このため、一般に変数の各モジュールへの実行前の配置決定は困難である。しかし本方式では、配置に関する特別な配慮を必要としない。すなわち、コンパイラはデータをアドレス順に配置するだけでよい。

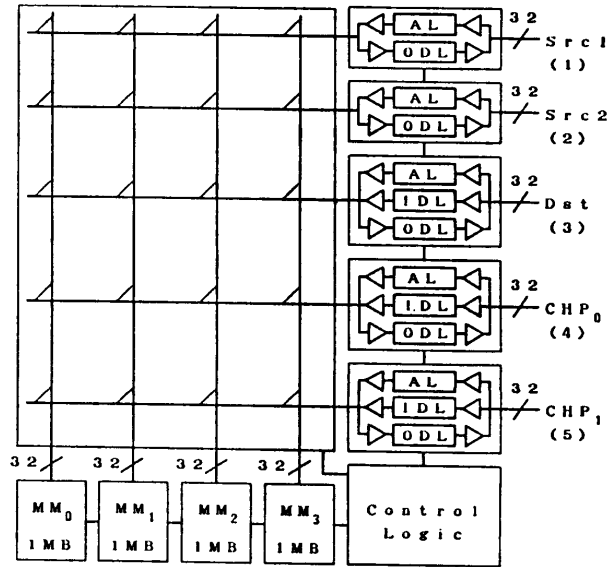
3.2.2 クロスバスイッチの制御

各メモリモジュール(MM)は、図5に示すクロスバスイッチ(CBS)により、演算器(DP)との3本のバス(Src1, Src2, Dst), 2つの通信制御プロセッサ



```
float a[3], b[3], c[3], d[3], e[3];
```

図4 3次元ベクトルのデータメモリ上への配置
Fig. 4 Placement of 3D vectors on data memory.



AL : Address Latch IDL : Input Data Latch
ODL : Output Data Latch () : Priority

図5 クロスバスイッチとデータメモリの構成
Fig. 5 Configuration of cross bar switch and data memory.

(CHP)と接続されている。CBSはアドレス、データラッチを持ち、送信側ユニットはメモリアクセス完了を待たず次の処理に移れる。

5組のうちいずれかからアクセス要求が到着次第、要求モジュールとの接続が行われるが、既に通信中であれば待たされる。アクセス要求が競合した場合は、Src1, Src2, Dst, CHP0, CHP1の順にラウンドロビン方式で決定する。

3.2.3 UCの動作

UCの各ユニットは独自のクロックで非同期に動作し、ハンドシェイクによりデータの授受を行い、必要なデータ・条件が揃えば自己の処理を他と独立に行う。これらのハンドシェイクは主としてシーケンサユニット(μ p-SEQU)の管理下で行われる。

単純なアドレッシングモードの2項演算の場合、まずマイクロプログラムメモリ(μ p-MEM)から、オペコードがDPに、オペランドアドレスがインデックスレジスタ、加算器を内蔵するインデックスユニット(IXU)に送られ、アドレス計算が行われる。次にそのアドレスがCBSを介してMMに送られ、読み出されたデータはCBSを介してDPに送られる。演算結果は再びCBSを介しMMに戻される。これらのシーケンスは、並列化されているだけでなく、パイプライン化されている。図6に、MMのアクセス競合の

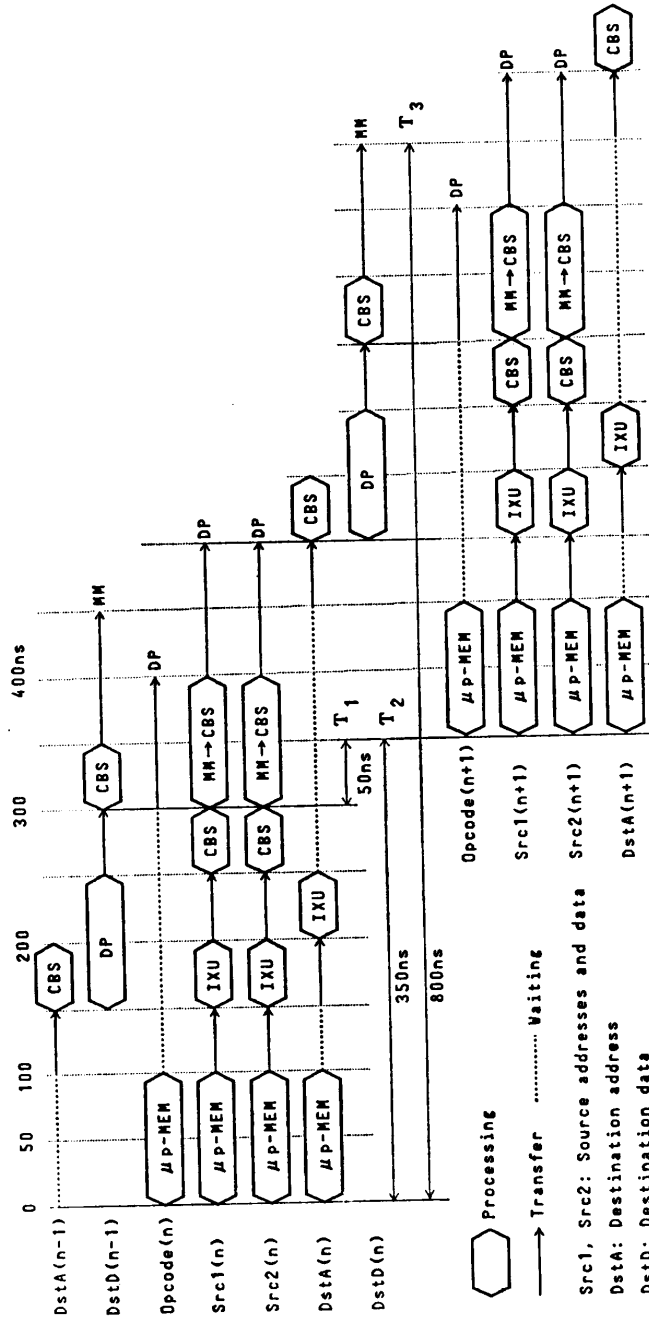


図 6 2 項演算の並列パイプライン処理 (メモリアクセス競合のない場合)
 Fig. 6 Parallel pipeline processing of binary operator without memory access conflict.

ない場合における $n-1, n, n+1$ 個目の命令実行の様子を、50 ns を単位として、各ユニットでの処理時間、待ち時間、データ転送時間を含めて時間を追って示した。ここでは、オペコードの DP への転送開始から 50 ns 後に次の命令の準備を始めるとした (図 6 T₁)。

本 UC では演算に際し、ユニット間のデータ依存関係に矛盾が生じないように以下に述べる命令実行上の制約を設けている。

(1) μp -MEM は、1つの命令のオペコードと3つのアドレスをすべて送信した後に初めて次の命令の準備を開始する。

(2) CBS (Src1, 2) はアドレス受信後、DP へのデータ転送を完了するまでバスを確保し続ける。

(3) CBS (Dst) は必要な MM を獲得後、演算結果を書き込むまで確保し続ける。

3.3 DP における演算の高速化手法

UC の演算処理は、データプロセッシングユニット

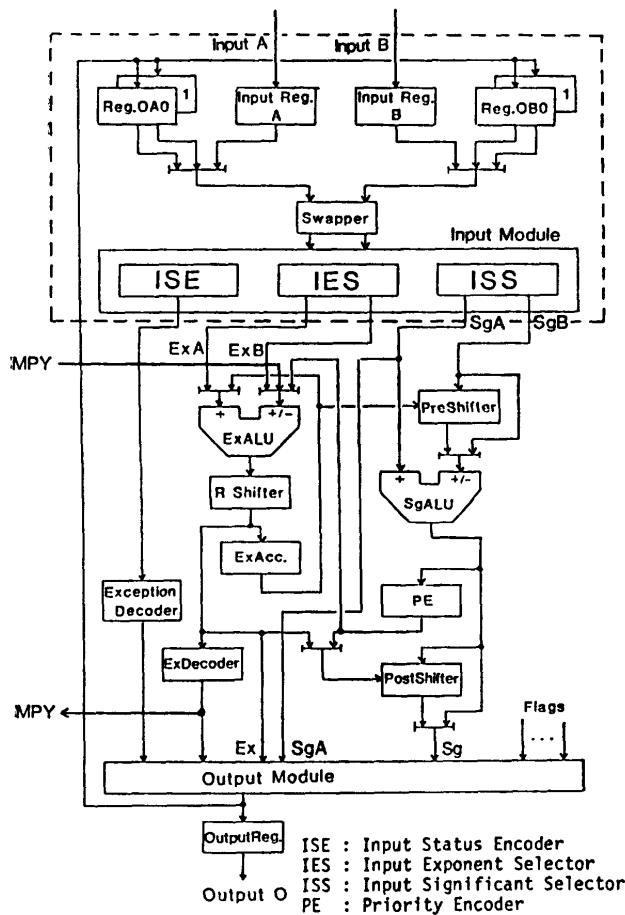


図 7 DPU の機能ブロック図
Fig. 7 Functional block diagram of DPU.

(DPU) (図 7) および乗算器 (MPY) からなるデータプロセッサ (DP) が行う。DP は、88 ビット幅のナノプログラムと呼ぶ、より低レベルの水平型マイクロプログラムで制御される。図 2 のオペコードは、ナノプログラムの開始番地 (10 ビット) の上位 8 ビットを指す。

DP には以下の高速化手法が採られている。

(1) 32 ビット長の SgALU と 10 ビット長の ExALU を用いて、浮動小数点演算は仮数部と指数部の演算を並列に行う。

(2) 加減乗算を高速化するため、逆数、開平、平方根の逆数演算は、DP のハードウェアに直結したルックアップテーブルを利用して求める。

(3) DP 内にローカルバスを持ち、ベクトル演算など DPU と MPY 間でデータ転送を必要とする演算が通信のオーバーヘッドなく行える。

(4) MPY には、 $32 \times 16 \rightarrow 48$ ビットの並列計数型

表 1 浮動小数点演算の処理時間の比較

Table 1 Comparison of floating point operation time.

命令	DP	i 8087
加 減 算	0.33	21.0
乗 算	0.3	23.6
除 算	1.05	44.0
3 次 の 内 積	1.67	112.8
平方根の逆数	0.95	80.6

(μ sec)

乗算回路¹³⁾を用いた。設計の省力化と乗算の高速化のため、機能ブロックの遅延時間に基づき最適化した回路の生成を、計算機を用い自動で行った¹⁵⁾。

表 1 に、DP の浮動小数点演算速度を i 8086/87 (クロック 5 MHz) と比較したものを示す (最小メモリアクセス時間を含む)。

3.4 演算処理方式の評価

本 UC の演算処理能力を調べるため、画像生成プログラムのうち輝度計算⁹⁾モジュールのオブジェクト (マイクロプログラム) を、UC の命令レベルのシミュレータ上で実行した。このシミュレータは、LINKS-2 用プログラムの検証を目的として作成され、MM のアクセス競合の状況と実行された命令のサイクル数からその実行時間を求めることができる。アクセス競合の状況を表 2 に、実行時間を表 3 に示す。実行時間については他の機種での実行結果と比較した。実行命令の総数は 262 命令、よって 1 命令の平均実行時間 326 ns、スループットは

表 2 輝度計算における MM のアクセス競合
Table 2 MM access conflict of intensity calculation.

競合の種類	比率 (%)
競合なし	85.6
Src1-Dst	6.0
Src1-Src2	3.4
Src2-Dst	2.2
前の Dst-Src2	1.6
前の Dst-Src1	0.7
Src1-Src2-Dst	0.5

表 3 輝度計算の処理時間
Table 3 Processing time of intensity calculation.

機 種	処理時間 (msec)	速度比
VAX 11/780 FPA 付	0.744	5.6
LINKS-1 UC 1 台	4.17	1.0
LINKS-2 UC 1 台	0.0854	48.8

3.1 MIPS となった。ただし本 UC の命令は、他のプロセッサの命令と粒度が異なり、単純に MIPS 値のみを比較することはできない。また、競合のないメモリアクセスの回数は 80% を超え、データメモリのインタリーブ構成を始めとするデータアクセスの並列化手法が画像生成処理に適したものであることがわかる。

次にパイプライン化の効果を、3.2 節で述べた 2 項演算における UC の動作モデルを例に解析した。図 6 から、MM のアクセス競合のない場合、1 命令の実行時間は 800 ns (図 6 T₃)、パイプラインの周期は定常状態では 350 ns (図 6 T₂)、したがって並列度は 2.29、スループットは 2.9 MIPS であることが導かれる。さらにこの図から、命令実行時間・パイプライン周期を決定しているクリティカルパスは、Src 1, Src 2 の μ p-MEM から DP に至るパスであることがわかる。

MM のアクセス競合のパイプラインに対する影響を調べるため、このモデルを計算機上に構築し、全く競合のない場合、表 2 に示した頻度で発生した場合についてシミュレーションを行った。その結果を表 4 に示す。競合のない場合は前述の解析結果とよく一致している。全く同じでないのは、パイプラインの立上りの過渡状態を含むためである。競合のない場合とある場合のパイプラインの平均周期の差はわずか 5 ns しかない。これは、(1)競合頻度の高い Src1, 2 と Dst は実際には同時刻にアクセスが起こらない、(2) CBS の利用率 (MM のアクセスを含む) が 84~85% と余裕があるためと考えられる。ここで利用率とは、各ユニットの内部処理時間とデータ転送時間の和の平均周期に対する比であり、純粋な待ち時間を除いた稼働率を表す。この実験は簡単な 2 項演算のモデルであ

表 4 UC のパイプライン動作のシミュレーション結果
Table 4 Simulation results of pipeline execution on UC.

	競合なし	競合あり
1 命令の平均実行時間 (ns)	800.7	810.4
パイプラインの平均周期 (ns)	351.0	356.0
パイプラインの並列度	2.28	2.28
スループット (MIPS)	2.85	2.81
μ p-MEM (Opcode) の利用率 (%)	57.0	56.2
μ p-MEM (その他) の利用率 (%)	42.7	42.1
IXU の利用率 (%)	42.7	42.1
CBS (含 MM アクセス) の利用率 (%)	85.5	84.3
DP の利用率 (%)	71.2	70.2

るが、画像生成処理では MM のアクセス競合が与える影響は少ないといえる。DP の利用率も 70% を超え、効率の良いデータ供給が実現されている。

マルチプロセッサ構成の場合、CHP からのアクセス要求による競合が発生するが、その影響はシステム構成および実行制御方式に大きく依存する。ここでは問題を単純にするため、CHP からのアクセス競合の影響を無視できると仮定した。

4. LINKS-2 の UC 間通信方式

4.1 通信処理能力の強化

デマンド転送方式における並列処理効率の低下を防ぐためには、競合のない専用バス方式が優れているが、IMSU の場合にも見られるように、接続する台数に比例して接続線数、UC 1 台当たりの通信用ハードウェア量が増大する。

一方、共有バス方式では、システム構成のソフトウェア的な制御が容易であり、耐故障性、負荷分散法の改良などの点で優れるが、バス競合は避けられない。しかし、データ転送が十分高速であれば、共有バス方式でも並列処理効率の低下を防ぐことができる。

そこで我々は、32 ビット共有バス (LCB: Links Communication Bus) を用いて高速ブロック転送を行う方式を考案した。本方式は次のような特徴を持つ。

(1) LCB におけるバス競合などが演算処理に及ぼす影響をできる限り減らすため、DP とは全く独立して動作する通信処理専用プロセッサ (CHP: CHannel Processor) を用いて制御する。

(2) データメモリへのアクセス競合を最少限にし、両者の並列稼働率を上げるため、CHP はクロスバースイッチを介してデータメモリと接続する。

(3) 多様なシステム構成を実現するため、UC は 2 つの CHP を持ち、2 系統の LCB と接続できる。

(4) バス調停のための集中的な制御機構は用いず、各 CHP が分散して行う方式とした。

(5) データ転送とは別に UC 間の動作制御 (割込、リセット等) を実現した。

(6) 外部装置との通信の高速化のため、外部装置側にも、同等の機能を有する通信制御プロセッサ (I/O プロセッサ) を持たせ、LCB に直接接続できるようにした。

(7) 大規模な並列処理システム構築のため、バス拡張用バッファを用意した。1 本の LCB に接続可能な台数は、バス利用率にのみ制限される。

4.2 CHP による UC 間通信

CHP は、図3に示すように、チャンネルプロセッシングユニット (CHPU) と 2KW のバッファメモリ (BM) から構成されている。CHPU の機能は、LCB の制御と調停、データ転送である。

LCB の転送速度は、最高 25 MW/sec (100 MB/sec) に達する。LCB に接続されている任意の CHP 間での通信が可能であるほか、バス上の全 CHP に対するブロードキャストも可能である。

同一バス上の CHPU には各々固有のモジュールナンバーが与えられており、常にどれか一つの CHPU がバスを占有している (マスタ CHPU)。

各 CHPU はマイクロプログラムや他の CHPU から転送命令を受けると、まず MM から BM ヘデータを転送した後、バス要求を出す。マスタ CHPU は、バス上のリクエストラインがアクティブになると、ポーリングにより次のマスタ CHPU (=バス要求を出した CHPU のうちの一つ) を指定した後、バスを解放する。ポーリングはハードウェア化された二分探索法を用い、最大でも $100 \cdot \lceil \log_2 n \rceil + 300$ (ns) でポーリングを終わる。

新たなマスタ CHPU は、転送先の CHPU が受信可能かどうか調べた後、パケット方式でデータを転送する。このとき、3ワードの CRC 誤り検出符号を付加する。転送クロックは送信側が決定し出力する。受信可能でなく、他にバス要求が出ていれば調停に移る。転送の際、エラーが生じた場合は自動的に転送クロックを落とし、再試行を行う。転送を終えた後はバスの調停を行う。

バスに接続されている UC の台数を n 、転送データ量を s (word) とすると、バス獲得のオーバーヘッドがなかった場合の通信時間 T は次式で表される。

$$T = 600 + 100 \cdot L + 40s \text{ (ns)}$$

ただし、 $1 \leq L \leq \lceil \log_2 n \rceil$ 。

4.3 並列処理システムの構成

図 8, 9 は、図1の LINKS-1 の星状あるいは木構造システムと等価なシステムである。LCB および I/O プロセッサを介して接続されたデータベースシステム (DBS) 上のデータおよびプログラムは、RC に転送される。LC が計算した画素データは、同じく LCB, I/O プロセッサを介して FMS に出力される。I/O プロセッサは、汎用プロセッサと CHP を用い、外部装置の種類に応じて作成した。この構成では、LINKS-1 で蓄積されたソフトウェア資源をそのまま

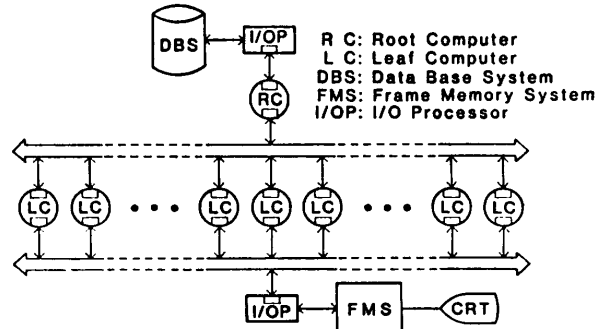


図 8 星状構造システム
Fig. 8 Star structured system.

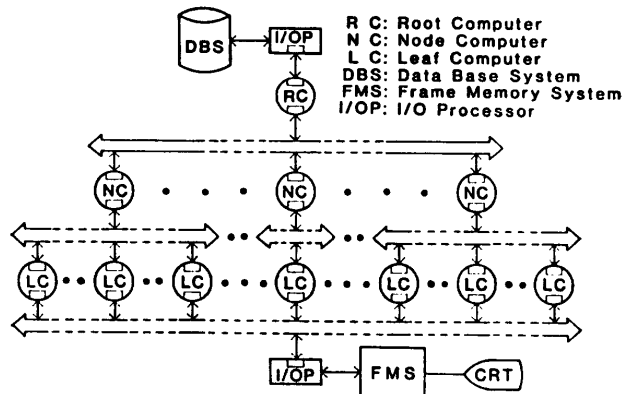


図 9 木構造システム
Fig. 9 Tree structured system.

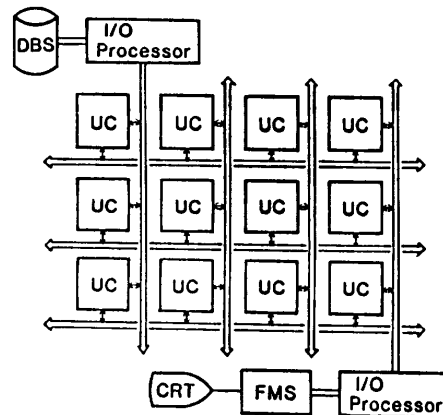


図 10 配列構造システム
Fig. 10 Array structured system.

有効に活用することができる。さらに、LINKS-1 では不可能であった NC-NC 間、LC-LC 間の通信が可能となり、画素間のコヒーレンスの有効利用ができ、メモリ容量不足の解消につながる。

図10の配列型構成は、LINKS-1 では困難なソフトウェア的な構成の変更や制御が容易であり、故障に

も対処しやすい。この場合、従来と同様な並列処理機構は、ソフトウェア的に星状あるいは木構造型の通信経路を設定することで実現できる。ただし、図 8、9 の場合に比べて、通信経路の長さが増加するという欠点がある。また、縦方向の各 UC に負荷分散すると同時に、横方向の UC に機能分散を行い、並列パイプライン処理が可能である。さらに、負荷分散手法に関しても、画面分割法¹⁰⁾だけでなく、空間分割法¹⁴⁾の実現も容易であるなど、柔軟性が高い。

4.4 通信処理方式の評価

演算処理能力に対する通信処理能力を評価するため、UC を複数台並列に動作させたときの処理速度の向上比（並列処理効率）をシミュレーションによって求めた。ここでは、総 LC 台数が N であるようなシステムによる画像生成時間を $T(N)$ と表し、並列処理効率 (Performance) を

$$P(N) = (1/T(N)) / (1/T(1)) = T(1)/T(N)$$

と定義する。

図 11 に示す球 99 個からなる画像を、デマンド転送方式により生成した場合の画像生成時間、および並列処理効率を LINKS-1 の場合と比較する。システム構成は図 8 に示した星状構造とする。実験時の条件は、反射・透過なし、全画素数が 512×512 、1 回に割り当てられる小画面中の画素数を 4×4 、 8×8 、 16×16 、 32×32 画素の 4 種類とした。

LINKS-1 では、LC 台数を変えながら、実際に画像を生成し、その所要時間を測定した。LINKS-2 の場合は、LINKS-1 と同様な実行制御を行うため、まず画像生成時の LCB を用いた RC-LC 間の通信を次のようにモデル化した。

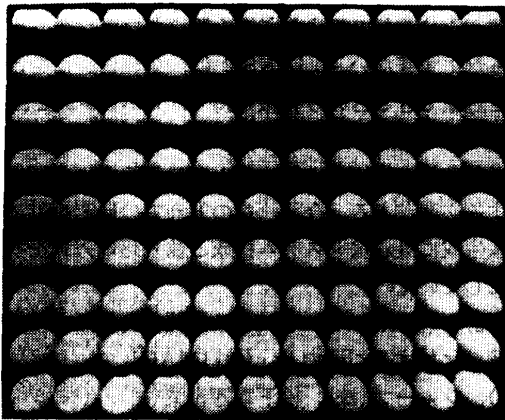


図 11 実験に用いた画像

Fig. 11 The image for experiments.

・RC の処理：

(1) LC からの割込みを待つ。

(2) LC からの小画面要求があれば、CHPU を起動、小画面データ (20 バイト) を転送。

(3) (1)に戻る。

・LC の処理：

(1) CHPU を起動し、バスを獲得する。

(2) RC に小画面を要求する。

(3) 小画面データを受領し、バスを解放する。

(4) もし物体データが必要なら、CHPU を起動、バスを獲得する。

(5) バス獲得後、RC の内部処理に影響を与えることなく、RC のデータメモリより必要な物体データを読み出す。これは、LC 側で RC における物体データのあるアドレスがわかっているためである。

(6) バスを解放し、小画面の処理を行う。

(7) (1)に戻る。

・マスタ CHPU の処理：

4.2 節に述べた手順でバスの調停を行う。

以上のモデルを、計算機上に構築した。CHPU の起動にはコマンドレジスタの設定などで $20 \mu\text{s}$ かかる。LC が RC から読み出す物体データ量と、LINKS-2 における小画面の処理時間が必要であるが、画像生成プログラムすべてをシミュレータで実行することは、所要時間の点で非現実的である。そこで、まず画像生成プログラムのうち、交差判定および輝度計算モジュールの実行時間を、3.4 節で用いた UC シミュレータにより測定した。これらは UC での処理時間に大きく寄与すると予想される。次にこれらの実行時間と画像生成プログラムの実行結果から、それぞれの分布を調べた。表 5 にその平均と分散を示す。このデータから本実験では、小画面の処理時間を正規分布、転送データ量を指数分布に従う乱数で近似した。

表 5 小画面の処理時間と転送データ量の平均と分散
Table 5 Means and variances of the processing time and the amount of transfer data at a subscreen.

小画面 (pixels)	処理時間		転送データ量	
	平均 (msec)	分散 (msec ²)	平均 (byte)	分散 (byte ²)
4×4	3.84	1.22	472.5	5.773×10^4
8×8	14.99	16.57	468.2	5.586×10^4
16×16	59.53	181.46	560.6	6.865×10^4
32×32	273.23	1645.83	675.0	6.637×10^4

表 6 LC 1 台の時の画像生成時間
Table 6 Processing time of the 1 LC system.

小画面 (pixels)	LINKS-1(sec)	LINKS-2(sec)	速度比
4×4	5716	63.97	89.4
8×8	5051	61.65	81.9
16×16	4807	61.03	78.8
32×32	4715	60.76	77.6

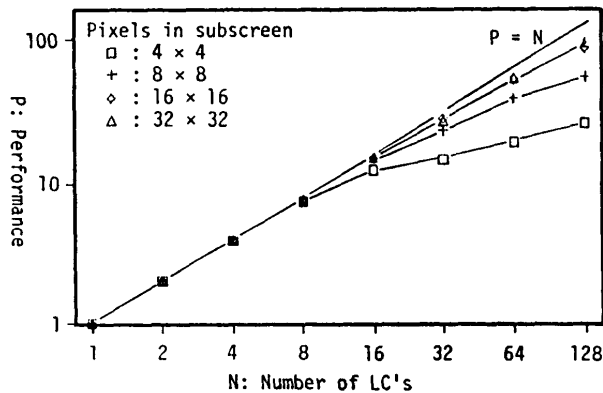


図 12 LINKS-1 の並列処理効率
Fig. 12 Performance of LINKS-1.

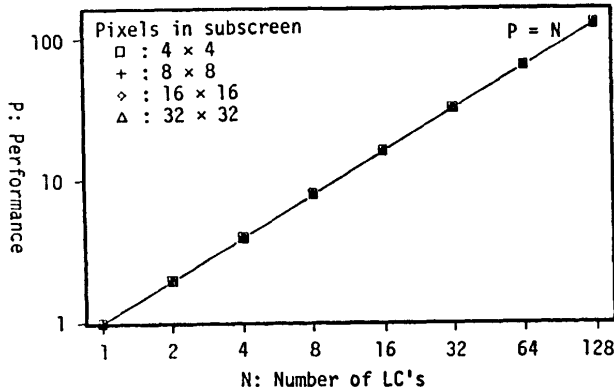


図 13 LINKS-2 の並列処理効率
Fig. 13 Performance of LINKS-2.

LINKS-1, 2 における UC 1 台の時の画像生成時間を表 6 に、並列処理効率を図 12, 13 に示す。この結果から、LINKS-1 では小画面の画素数によって並列処理効率が大幅に異なり、128 台でもたかだか 94 台分の能力しかない。これに対して LINKS-2 では、並列処理効率 $P(N)$ は画素数によらずほぼ N に等しく、最善の結果を得た。

LINKS-1 におけるこれまでの実験では、交差判定ならびに輝度計算は、全処理時間のうち約 60~70% であり⁹⁾、実際の UC 1 台の画像生成速度は LINKS-

1 のおよそ 50 倍程度であると考えられる。その場合でも、並列処理効率は処理時間 t と通信時間 τ の比で決まるため、図 13 に示す値を下回ることはない。

総 LC 台数 N が臨界値 $((\tau+t)/\tau)$ を超えると、並列処理効率 $P(N)$ は、 N に関係ない一定値に収束する¹⁰⁾。表 5 の平均処理時間、平均データ転送量から臨界値を見積もると、小画面の画素数によって異なるが、8×8 画素のときおよそ 2800 台となった。この値は大規模なシステム構築の有効性を示唆する。

以上の結果を総合して、演算速度の向上に見合うだけの通信処理速度の改善が実現された。

5. むすび

本論文では、LINKS-1 での実験に基づき設計した、並列画像生成システム LINKS-2 の浮動小数点演算の高速化手法ならびに通信機能の強化手法について述べ、シミュレーションによりその効果を明らかにした。特に演算速度の改善だけでは、並列処理時の効率が低下するため、通信速度の改善が同時に必要となる。LINKS-2 では、それぞれの改善がバランス良く行われていることがこの結果から認められる。LC が 100 台程度のシステムでもそれに見合う処理能力の向上が期待でき、さらには、設計上の目標である LC 1000 台程度の大規模なシステムの構築に関する指針が得られた。システム完成後は、高品質な画像を十分高速に生成できるものと考えられる。

このようなハードウェアの性能を十分に発揮させるには、最適化されたコードを出力するコンパイラを始めとするソフトウェアの充実が必須である。

本論文でのシミュレーションは、LINKS-1 と同様な並列画像生成処理を仮定しており、演算処理と通信処理とを分離して扱うなどごく限られた条件のもとで行ったにすぎない。LINKS-2 は、演算と通信を並行して処理できるという特徴を持ち、LINKS-1 にはないシステム構成の柔軟性に富んでいる。よってさらに多様な条件や、構成のシステムでのふるまいを明らかにする必要がある。今後、この方向での研究を進め、画像生成に最適なシステム構成と並列処理方式を明らかにしていく。

謝辞 (株)トーヨーリンクスの関係各位の本システム開発に対するご尽力と、(株)IMAGICA、日本電気(株)、シャープ(株)、松下電器産業(株)、(株)ジェック、(株)西村技研の関係各位のご助力に感謝の意を表する。

参 考 文 献

- 1) 新實治男: グラフィックス専用マシンの開発動向, 情報処理学会計算機アーキテクチャ研究会資料, 60-2 (1985. 12. 6).
- 2) Whitted, T.: An Improved Illumination Model for Shaded Display, *Comm. ACM*, Vol. 23, No. 6, pp. 343-349 (1980. 6).
- 3) 岡本充由: 3次元画像処理用マルチコンピュータシステムに関する研究, 大阪大学大学院工学研究科電子工学専攻特別研究報告 (1981. 3).
- 4) Russell, R. M.: Cray-1 Computer System, *Comm. ACM*, Vol. 21, No. 1, pp. 26-31 (1978).
- 5) 久保, 戸島, 亀井, 阿草, 大野: 3次元色彩画像表示処理専用マルチマイクロプロセッサシステム G-PSYCO, 情報処理学会論文誌, Vol. 25, No. 1, pp. 10-18 (1984).
- 6) 平井, 日高, 浅原, 鷲島: 画像生成用 SIMD 型マルチプロセッサシステム MC-2, 情報処理学会マルチメディア通信と分散処理研究会資料, 29-5 (1986. 5. 15).
- 7) 吉田, 成瀬, 高橋, 内藤: グラフィックス計算機 SIGHT の基本構成, 情報処理学会計算機アーキテクチャ研究会資料, 60-4 (1985. 12. 6).
- 8) 出口, 西村, 吉村, 河田, 白川, コンピュータグラフィックスシステム LINKS-1 における画像生成の高速化手法, 情報処理学会論文誌, Vol. 25, No. 6, pp. 944-952 (1984).
- 9) 西田, 辰己, 出口, 西村, 河田, 白川, 大村: 視線探索法に基づく三次元色彩陰影画像生成プログラムの解析, 昭 59 信学総全大, S 14-20, pp. 6-313-6-314 (1984. 3).
- 10) 西村, 吉村, 出口, 辰己, 河合, 河田, 白川, 大村, 尾崎: マルチマイクロコンピュータシステムにおける画像生成のための画面分割方式, 信学技報, CAS 82-144, pp. 31-36 (1983. 2. 9).
- 11) 西村, 出口, 辰己, 河田, 白川, 大村: コンピュータグラフィックスシステム LINKS-1 における並列処理の性能評価, 信学論 (D), Vol. J 68-D, No. 4, pp. 733-740 (1985).
- 12) 出口, 西田, 西村, 河田, 白川, 大村: 視線探索法による画像生成のための木構造並列処理システム, 信学論 (D), Vol. J 69-D, No. 2, pp. 170-179 (1986).
- 13) Stenzel, W. J., Kubitz, W. J. and Garcia, G. H.: A Compact High-Speed Parallel Multiplication Scheme, *IEEE Trans. Comput.*, Vol. C-26, No. 10, pp. 948-957 (1977).
- 14) Dippe, M. and Swensen, J.: An Adaptive Subdivision Algorithm and Parallel Architecture for Realistic Image Synthesis, *ACM SIGGRAPH '84*, Vol. 18, No. 3, pp. 149-158 (1984).
- 15) 河合, 若井, 正田, 西村, 近藤, 高山, 出口, 白

川, 大村: 画像生成用マルチコンピュータシステムとそのプロセッサについて, 情報処理学会グラフィックスと CAD 研究会資料, 19-4 (1985. 10. 25).

付 録

使用ゲートアレイ一覧 (UC 1 台)

ユニット	種 類
DPU	CMOS 10000 ゲート×1
MPY	CMOS 8000 ゲート×1
IXU	TTL (内部 ECL) 2000 ゲート×3
CBS	CMOS 5000 ゲート×4
μP-SEQU	CMOS 4000 ゲート×1
CHPU	CMOS 10000 ゲート×2

(昭和 62 年 9 月 14 日受付)

(昭和 63 年 6 月 24 日採録)



河合 利幸 (正会員)

昭和 34 年生。昭和 57 年大阪大学工学部電子卒業。昭和 63 年同大大学院博士課程退学。同年大阪電気通信大学工学部講師。コンピュータグラフィックスに関する研究に従事。

電子情報通信学会, テレビジョン学会, IEEE 各会員。



山下 伸一

昭和 35 年生。昭和 58 年大阪大学工学部電子卒業。昭和 60 年同大大学院修士課程修了。同年(株)東洋現象所(現(株)IMAGICA)入社。以来コンピュータのハードウェアおよびシステム設計, 専用 LSI の設計に従事。昭和 60~63 年(株)トーヨーリンクス出向。現在研究開発室所属。



大野 廣司

昭和 34 年生。昭和 56 年大阪大学工学部電子卒業。昭和 58 年同大大学院修士課程修了。同年(株)東洋現象所(現(株)IMAGICA)入社。同年より昭和 63 年まで(株)トーヨーリンクスに出向。現在研究開発室所属。コンピュータグラフィックス関連のシステム設計, 周辺機器のハードウェア設計に従事。電子情報通信学会会員。



吉村 浩

昭和33年生。昭和56年大阪大学工学部電子卒業。昭和58年同大学院修士課程修了。同年(株)東洋現像所(現(株)IMAGICA)入社。昭和58~63年(株)トーヨーリンクス出向。現在研究開発室所属。コンピュータグラフィックスシステムの研究・開発に従事。ACM, IEEE 各会員。



西村 仁志

昭和55年大阪大学工学部電子卒業。昭和60年同大学院博士課程修了。同年(株)トーヨーリンクス入社。現在(株)西村技研代表取締役社長。コンピュータグラフィックス, CAD システムの研究・開発に従事。工学博士。電子情報通信学会, IEEE 各会員。



下條 真司 (正会員)

昭和56年大阪大学基礎工学部情報卒業。昭和61年同大学院博士課程修了。同年大阪大学基礎工学部助手。待ち行列網シミュレータ, ローカルエリアネットワークにおけるアクセス方式の性能評価, 分散処理システムの性能評価に関する研究に従事。工学博士。IEEE, 電子情報通信学会各会員。



宮原 秀夫 (正会員)

昭和42年大阪大学工学部通信卒業。昭和47年同大学院博士課程修了。昭和48年京都大学工学部助手。昭和55年大阪大学基礎工学部助教授。昭和62年大阪大学大型計算機センター教授, 基礎工学部兼任教授。昭和58~59年 IBM トーマスワトソン研究所客員研究員。システムの性能評価, 分散処理システム, LAN の研究に従事。工学博士。電子情報通信学会会員。



大村 皓一

昭和13年生。昭和35年大阪大学工学部通信卒業。昭和37年同大学院修士課程修了。昭和40年大阪大学工学部助手。昭和48年同助教授。昭和63年大阪学院大学教授。コンピュータハードウェア, 医用電子工学, コンピュータグラフィックスに関する研究に従事。工学博士。ACM, IEEE 各会員。