

マルチプロセッサ向き画像生成ユニット

- Scanline Z-Buffer Merger -

西村 明夫、小林 忠司、若山 順彦、前信 潔、西澤 貞次、鷺島 敬之

松下電器産業株式会社 映像音響研究センター メディア研究所

〒 571 大阪府門真市大字門真 1006

E-mail: akio@isl.mei.co.jp

あらまし マルチプロセッサシステムにおいて、より高画質かつ高速な画像生成を行おうとする場合に問題となるポリゴン数の増加、スクリーン上でのポリゴンの重なりの増加に対応するため、多数のポリゴンに対するマッピング処理を高速に並列処理できるアーキテクチャとしてスキャンラインZバッファマージャーを提案する。また、試算による処理能力の見積りについても報告する。

セグメントジェネレータとスキャンラインZバッファマージャーを組み合わせたポリゴン処理ユニットあたりの処理能力は最大115.8K Polygons/sec と試算でき、このユニットを64ユニット程度並列動作させることによりテクスチャ付ポリゴン処理能力7.4MPolygons/sec が実現できる見込みである。

和文キーワード 画像、生成、スキャンライン、Zバッファ、テクスチャマッピング、マルチプロセッサ

Image Generation Unit for Multi Processor System

- Scanline Z-Buffer Merger -

A. Nishimura, T. Kobayashi, Y. Wakayama, K. Maenobu, T. Nishizawa, T. Sagishima

Media Research Laboratory, Matsushita Electric Industrial Co.,Ltd.

1006 Kadoma, Osaka 571 JAPAN

Internet: akio@isl.mei.co.jp

Abstract A high-performance image generation architecture "Scanline Z-Buffer Merger" is presented. It can deal with the increase in the number of polygons, and it's overlapping for realistic and high speed image generation.

Polygon processing unit, which consists of Segment Generator and Scanline Z-Buffer Merger, is evaluated it's performance as 115.8K Polygons/sec. 64 units of polygon processing unit can achieve the performance of 7.4MPolygons/sec.

英文 key words Image, Generation, Scanline, Z-Buffer, Texture Mapping, Multi-Processor

1 はじめに

人間と計算機を近づけようという努力が各方面で払われている。我々はその一つの方向として、映像を通じてのコミュニケーションの可能性を追求し、高速かつ高画質な画像生成に注力してきた[日高 85, 安部 89, 前信 90, 若山 90]。画像生成において高画質性と高速性を両立させるため、SIG-2 システム [安部 89] においてはラジオシティ法の計算結果およびテクスチャのマッピング処理とスキャンラインアルゴリズムによる隠面消去処理を組み合わせた。また、高速化のための有効な手段としてマルチプロセッサによる並列処理を利用した。マッピング処理により写実感が増し、動画による部屋のウォークスルーによってある種の仮想現実感を感じさせるまでになった。

さらなる高画質化のためには、ポリゴン数の増加が不可欠であり、ポリゴン生成能力の向上が画像生成ハードウェアの重要な目標となっている [Sil92, 佐々92]。

このポリゴン数の増加を別の観点で考えてみると、ポリゴン数の増加は構造の複雑度を増し、構造が複雑になるとスクリーン上でのポリゴンの重なり (以下、重なりと略す) も増加してくる。また、表現する空間自体の広がりも重なりの増加に拍車をかける。これに対して各種の隠面消去アルゴリズム (Z バッファ法など) では、輝度値を求める計算 (ライティング計算、マッピング処理など) を隠面消去処理の前に行うため、重なりの増加に伴って無駄な処理の比率が大きくなっていく。

これらのポリゴン数の増加、重なりの増加の問題に対応するため、本論文では、多数のポリゴンに対するマッピング処理を高速に並列処理できるアーキテクチャとしてスキャンライン Z バッファマージャを提案する。また、試算による処理能力の見積りについても報告する。

セグメントジェネレータとスキャンライン Z バッファマージャを組み合わせたポリゴン処理ユニットあたりの処理能力は最大 $115.8K \text{ Polygons/sec}$ と試算でき、このユニットを 64 ユニット程度並列動作させることによりテクスチャ付ポリゴン処理能力 $7.4M \text{ Polygons/sec}$ が実現できる見込みである。

2 並列画像生成処理

2.1 並列処理で画像生成を行なう場合の問題

画像生成処理を高速に実行しようとする場合、並列処理手法がよく用いられる [鷲島 91]。これは、画像生成処理全体の処理量の大きさに対応するということが第 1 の目的ではあるが、形状を表現するポリゴン間あるいは最終生成画素間の独立性が高いことを利用しようとするアプローチでもある。

隠面消去を高速に処理するアルゴリズムとしては、スキャンライン法、リストプライオリティ法、Z バッファ法などがある [FvDFH90]。

高速画像生成専用ハードウェアにおいては、Z バッファ法が単純な構造を採れるため有利である。

ここで並列処理を念頭において Z バッファ処理の流れを見直してみよう。処理の流れは図 1 のようになる。この図において左端にはポリゴン、右端には出力画像が示されており、処理は順に左から右に流れる。ワールド座標系への変換から画素生成までの処理はポリゴン独立に進めることが可能である。一方、透視変換終了時にポリゴン情報と出力画素の対応関係が明確になるため、それ以後 Z-Buffer 更新処理までは出力画素領域単位で並列処理が可能である。この 2 種類の並列処理をうまく切り替えることができれば非常に効率的な並列処理アーキテクチャとなる。ポリゴン単位の並列処理を出力画素単位の並列処理に切り替える際の情報の再編成のオーバーヘッドはかなり大きく、どの段階で切り替えるかが画像生成ハードウェアアーキテクチャにおける非常に重要な問題である。また、この並列処理にパイプライン処理を組み合わせることによりさらに高速化が可能である。

2.2 代表的なアーキテクチャの考え方

代表的なレンダリング専用ハードウェアアーキテクチャの考え方を振り返ってみると、SGI 社の IRIS シリーズ GTX, VGX, Elan, Reality Engine では、パイプライン構造をとり、パイプラインステップ内では一部並列処理を行うが、次のステップの前にデータを集中化して処理する。セグメント生成の直後にデータの再編成を行って、出力画素単位の並列処理へと切替えている [AJ88, Sil92]。

HP 社の Turbo/SRX においては、透視変換までをポリゴン単位で並列処理した後にデータを集中化し、以後をパイプライン化して処理している [鹿野 88]。

SIG-2 では、全プロセッサが形状データのコピーを持ち、処理の最初から出力画素単位で並列処理することにより、この問題 (データの再編成オーバーヘッド) を回避していた [安部 89]。

2.3 マッピング処理問題

Z バッファ処理にマッピング処理が加わるとさらに問題が複雑になる。マッピング処理は通常画素生成の際に行われ、テクスチャ情報とライティング条件とを組み合わせる RGB データに変換する処理である。このステップにおいて 1 つのマッピングデータは複数のポリゴンから参照されるため、並列処理でこのマッピング処理を行おうとする場合は、ポリゴンを並列処理する複数のプロセッサがマッピングデー

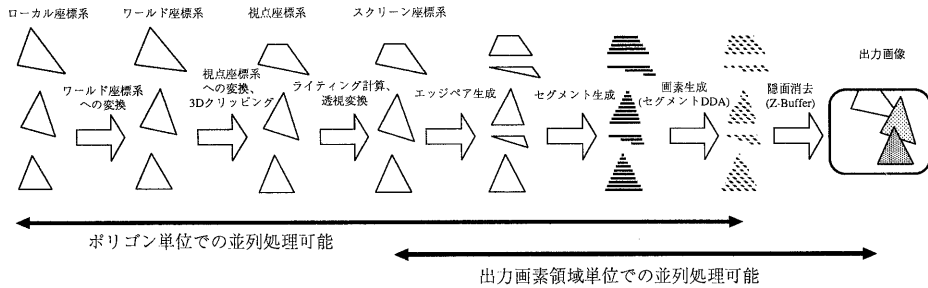


図 1: Zバッファ処理の流れ

タを格納しているメモリを共有することになる。マッピングデータの参照のためのメモリアクセスが頻繁に行われるため、ネットワークの十分な転送レートが必要になってくる。これを回避するために、個々のプロセッサがマッピングデータのコピーを持つという考え方もある。

3 ZB/M アーキテクチャ

3.1 2レベル並列処理

我々はまず、並列処理の効率的なインプリメントを優先して検討をすすめた。ポリゴン単位の並列処理に、セグメント DDA 機構を組み合わせることを基本とした。

画像生成の並列処理については、図 2 に示すように 2 レベルの並列化を採用した。第 1 レベルではエッジペア（上下の辺がスキャンラインに平行なポリゴン）生成までの処理を並列処理し、データの再編成を行った後、ポリゴン内 DDA 処理、Z バッファ処理によって画像を生成する。第 2 レベルでは、画像生成結果をスキャンライン単位で合成処理することとした。本論文では上記第 1 レベル並列処理以降のセグメントジェネレータ、スキャンライン Z バッファマージャおよびテクスチャ/照度マッピングエンジンを含む ZB/M アーキテクチャを提案する。

3.2 スキャンライン Z バッファ処理

ZB/M アーキテクチャでは、隠面消去処理にスキャンライン Z バッファ法を採用している。セグメント生成まではスキャンライン法、スキャンライン内は Z バッファ処理で隠面消去を行う。最近発表された画像生成アーキテクチャの中にもその例が見られる。[KWG92]

スキャンライン Z バッファ処理の利点は、スキャンライン法におけるセグメント処理オーバーヘッドの低減、および Z バッファ法におけるメモリ容量の節約である。前者により第 1 レベル並列処理部に対する処理能力要求を低く抑え

られ、後者によって Z バッファ/画素データメモリ部を小型化できることになる。

これに対して、スキャンライン Z バッファ処理の欠点は、ポリゴンソートの必要性であり、このために単純な並列パイプライン処理には向かず、透視変換の後に一度データを集中化する必要がある。また、高速処理のためにセグメント DDA 機構を採用する場合には、スキャンラインごとと同じポリゴンデータを読み直さなければならず、データ転送能力に対する要求が大きくなる。

3.3 テクスチャマッピング処理

マッピング処理のオーバーヘッドは無視できない。マッピング処理は基本的にメモリのランダムアクセスとなるので、メモリの速度に大きく依存することになる。昨今のプロセッサの処理速度の飛躍的な向上に対してメモリの速度は追従してきていない。したがって、マッピング処理を減らす努力は画像生成能力の向上に対して非常に有効であることがわかる。

ZB/M アーキテクチャにおいては、Z マージ後にマッピング処理を行うことにより、マッピングメモリへの参照を集中化し、無駄なアクセスを排除している。

4 セグメントジェネレータ

ZB/M アーキテクチャにおいては、第 1 レベル並列処理結果の入力部にセグメントジェネレータを持っている。このユニットは、セグメント DDA 機構を持つ場合の欠点である、スキャンラインごとのポリゴンデータの読み直しのオーバーヘッドを解消し、さらに y 方向の補間処理を同時に実行するためのものである。

4.1 セグメント生成処理

セグメント生成処理は図 3 に示した処理となる。すなわち、左上頂点の (x, y, z) 座標、エッジの横方向の長

第2レベル並列処理

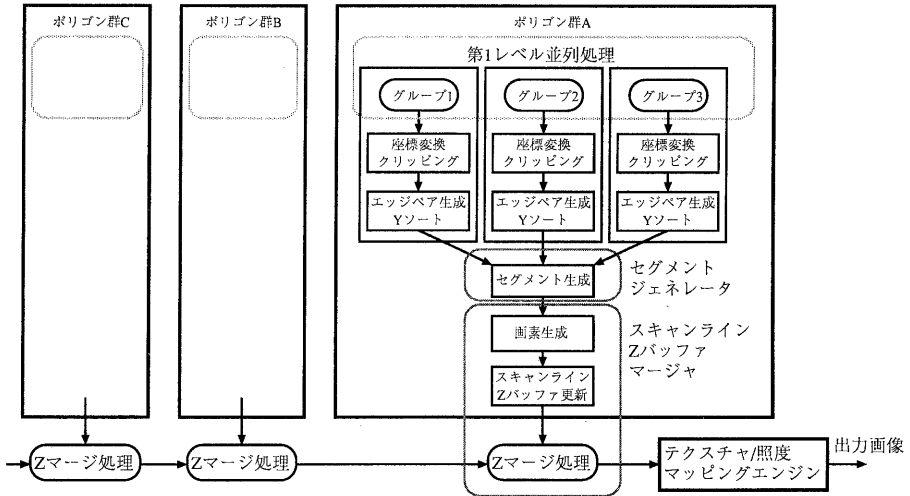


図 2: 2 レベル並列処理の概念図

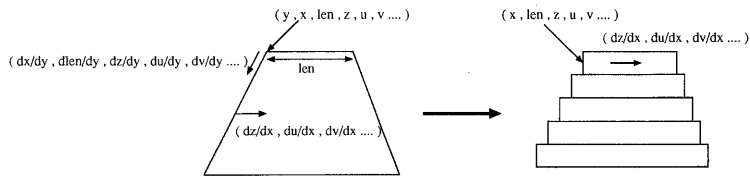


図 3: セグメント生成処理

さ (len)、マッピング座標 (u,v) およびそれらのパラメータの 1 スキャンライン下に進んだ時の増分 (dx/dy,dlen/dy,...)、さらに横方向に 1 画素進んだときの増分 (dz/dx,...) を入力とし、左端のパラメータ値と横方向の増分を含むセグメントデータを出力する。

異なるエッジペアに属するセグメントでも同一スキャンラインに存在するセグメントは続けて出力する必要があり、そのためセグメント生成中のエッジペア (アクティブエッジペア) を記憶するバッファが必要である。

4.2 ハードウェア構成

上記の処理を考慮して図 4 のようなハードウェア構成とした。

上部の入力 FIFO は、第 1 レベル並列処理部とセグメントジェネレータとの調速のためのバッファである。中央のアクティブエッジペアメモリはアクティブエッジペアを格納し、下部の加算器とレジスタはアクティブエッジペアを更新する。

アクティブエッジペアは順に読み出され、更新され、必要なものが再びアクティブエッジペアメモリに格納さ

れる。必要なくなったアクティブエッジペアはメモリに格納されずに捨てられる。

1 スキャンラインの更新が終ると、次のスキャンラインで新たにアクティブになるエッジペアが入力 FIFO より入力される。

5 スキャンライン Z バッファマージ

図 2 において、画素生成処理、スキャンライン Z バッファ処理、Z マージ処理を行うブロックをスキャンライン Z バッファマージ (以後 ZB/M と略す) と呼ぶ。

5.1 Z マージ処理

並列処理の第 2 レベルで行われた生成画像の合成処理は、生成画像の Z 値をもとに、隠面消去処理を行う形をとる (Z マージ処理)。その利点は、第 2 レベルでの並列処理の効率的な実行が可能となる点である。しかし、欠点としては半透明表現のための α 合成処理が困難であることが挙げられる。

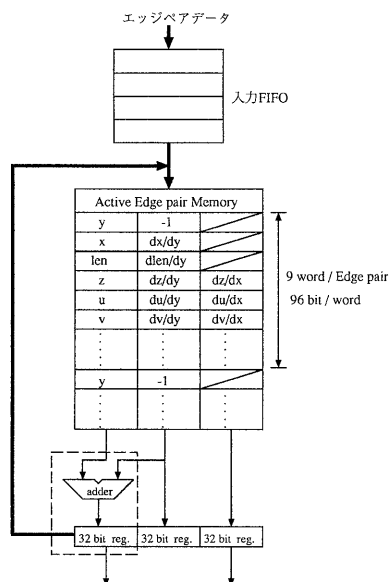


図4: セグメントジェネレータ

5.2 ダブルZバッファ処理

Z マージ処理において α 合成処理が困難であるという欠点を補うため、スキャンラインZバッファを2枚用意し、Zバッファ処理時に常に手前から2点分のZ値および画素データを保存する形を取った。Z マージ処理時にも、となりのマージャからの2点分のデータと自分の持つ2点分の合計4点分のデータを、Z値をもとに手前2点分を残して次のマージャに出力する。

最終的には、全体での手前から2点分のデータが残ることになり、この2点のデータをもとに α 合成処理を行うことになる。

5.3 インタリーブング

出力画素領域単位の並列処理可能性を生かし、かつ出力画素レートのフレキシビリティを得るため、ZB/M は水平方向のインタリーブングが可能な設計となっている。インタリーブ数は1,2,4,8,16の5種類から選択できる。

出力画素レートがインタリーブ数と比例する構成となっているので、ZB/M の応用システムの必要画素レートに合わせてインタリーブ数を決定できる。

これに対して、ZB/M の応用システムの必要画像生成能力 (Polygons/sec) に関しては、第1レベル並列処理部の設計および第2レベル並列処理の並列度で対応することになる。

このように、出力画素レートとポリゴン処理能力をほぼ独立に変更できるため、非常にフレキシブルなシステム構

成が可能となった。

5.4 内部構成

ZB/M は、 $x, len, z, r, g, b, \alpha$ に対するセグメント DDA 機構を持つが、第3.3節で述べたマッピング処理に対応するため、 $x, len, z, um, vm, ui, vi, \alpha$ に対するセグメント DDA 機構としても動作するように設計している。

ZB/M は図5に示すように、おもに8つのピクセルプロセッサとイニシャルプロセッサ、セグメントプロセッサ、マージャで構成される。8つのピクセルプロセッサとセグメントプロセッサはループ状に接続されている。これをセグメントループと呼ぶ。さらに、8つのピクセルプロセッサとマージャはマージバスに接続されている。

図中のピクセルプロセッサ、イニシャルプロセッサ、セグメントプロセッサはセグメントクロックで動作しており、マージャはそれとは独立のマージクロックで動作している。したがって、マージバスは非同期バスとなっている。

以下各部を簡単に説明する。

- イニシャルプロセッサ
イニシャルプロセッサはセグメント入力ポートに接続されており、入力されたセグメントのインタリーブ構成のための前処理を行う。その後、セグメントの内部データ形式である、セグメントパケットにフォーマット変換し、セグメントプロセッサに転送する。
- セグメントループ
セグメントループは112 bit 幅を持ち、セグメントの開始座標、セグメント長、画素情報とその増分等のデータを含むセグメントパケットが1サイクルごとにピクセルプロセッサ間を転送される。セグメントパケットは最大で8サイクル長であり、ループには同時に1つのセグメントパケットしか投入されない。
- セグメントプロセッサ
セグメントプロセッサはセグメントループを制御する。すなわち、セグメントループにセグメントパケットが入力可能かどうかを判断し、可能な場合は新たなセグメントパケットを生成し、投入する。
- ピクセルプロセッサ
ピクセルプロセッサは画素情報を保持するための内部メモリを持ち、それぞれチップ内の画素の1/8づつをインタリーブして受け持つ。ZB/M はチップ間でもインタリーブ構成をとる機能があるので、例えば、4チップでインタリーブした場合は、ピクセルプロセッサは画面上の水平32画素ごとに受け持つことになる。
内部メモリはダブルバッファ化されており、その各々のバッファがセグメントループ側とマージバス側に接続される。セグメントループ側ではセグメントパケットをDDA処理により更新し、次のピクセルプロセッサに転

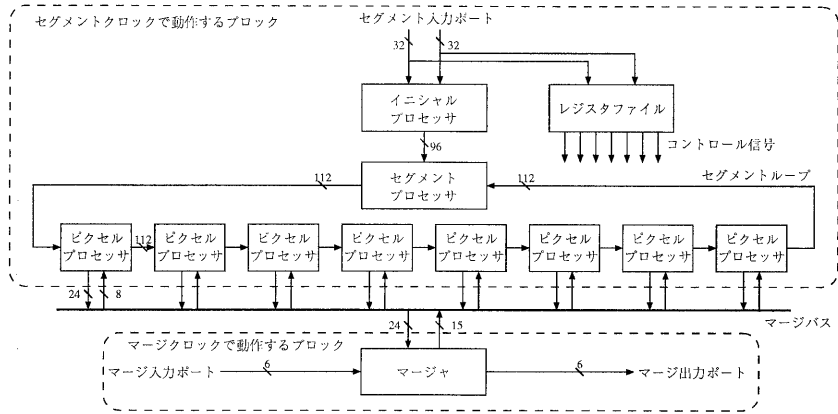


図 5: ZB/M 内部ブロック図

送する。同時に、Zバッファ法による隠面消去処理を実行し、必要な場合は内部メモリを更新する。マージバス側ではマージアの指示に従って内部メモリ中の画素情報をマージバスに出力する。

- マージバス
マージバスは各ピクセルプロセッサの内部に保持されている画素情報を読み出し、Zマージ処理を行う際に使用する。
- マージア
マージアはマージ入力ポートからの画素情報とピクセルプロセッサの内部メモリ中の画素情報からZマージ処理を行い、結果をマージ出力ポートに出力する。

6 パフォーマンス見積り

6.1 セグメントジェネレータ、スキャンラインZバッファマージアの処理能力見積り

セグメントジェネレータからZB/Mまでの処理能力を見積るため、前提条件として以下の条件をおいた。

1. 前処理で各ポリゴンは2エッジペアに展開される。
2. 平均5x5=25Pixelsのエッジペアを扱う。
3. ZB/Mの入力クロックは25MHz。
4. セグメントジェネレータへのエッジペアの入力には9サイクルかかる。
5. エッジペア入力中はZB/Mへの入力ができないため9サイクル空きができる。
6. ZB/Mへの入力は平均して1個のZB/M当りのセグメント長サイクル待たされる。入力を開始してからのサイクル数は8である。
7. 隣のZB/Mとのタイミングずれにより半分の時間は待ち状態と考える。

これらの条件下で予想処理能力を算出してみると、

- セグメントジェネレータへのエッジペアの入力には、エッジペアの大きさに関係なく9サイクルかかる。
- 8インタリーブ以上では、1個のZB/M当りの平均セグメント長が1となるため、条件6により、ZB/Mへのセグメント入力に9サイクルかかり、平均生成セグメント数が5であることを考慮すると、1エッジペアの処理サイクル数は $9 + 9 \times 5 = 54Cycles$ となる。
- したがって、処理可能エッジペア数は、

$$\frac{(25MCycles/sec)}{(54Cycle/Edge)} = 463.0K Edges/sec$$

となる。

- 条件7より、実効処理可能エッジペア数は、

$$463.0K/2 = 231.5K Edges/sec$$

となる。

- これをポリゴン数に直すと、

$$\frac{(231.5K Edges)}{(2Edge/Polygon)} = 115.8K Polygons/sec$$

となる。

- 4インタリーブでは、1個のZB/M当りの平均セグメント長は2となり、エッジペア処理サイクル数は $9 + 10 \times 5 = 59$ サイクルであり、

$$\frac{(25MCycles/sec)}{(59Cycle/Edge)} = 423.7K Edges/sec$$

となり、これをポリゴン数に直すと、 $105.9K Polygons/sec$ となる。

- 2インタリーブでは、1個のZB/M当りの平均セグメント長は3となり、エッジペア処理サイクル数は $9 + 11 \times 5 = 64$ サイクルであり、

$$\frac{(25MCycles/sec)}{(64Cycle/Edge)} = 390.6K Edges/sec$$

となり、これをポリゴン数に直すと、

97.7K Polygons/sec となる。

- 1 インタリーブでは、1 個の ZB/M 当たりの平均セグメント長は 5 となり、エッジペア処理サイクル数は $9+13 \times 5 = 74$ サイクルであり、

$$\frac{(25MCycles/sec)}{(74Cycle/Edge)} = 337.8K Edges/sec$$

となり、これをポリゴン数に直すと、

84.5K Polygons/sec となる。

以上の結果をまとめると、表 1 のような見積りとなり、セグメントジェネレータ、ZB/M の処理能力は 1 ユニット当り最大 115.8K Polygons/sec 程度と予想される。

表 1: ポリゴン処理ユニットの処理能力見積り

インタリーブ数	1	2	4	8	16
処理能力	84.5	97.7	105.9	115.8	115.8

単位: K Polygons/sec

6.2 プロセッサ部の処理能力見積り

第 1 レベルの並列処理を実行するプロセッサ部の処理能力を見積りてみることにする。仮に、この部分を i860 マイクロプロセッサ複数個で構成した場合を考えてみる。

この並列処理部分の処理を i860 で実行した場合の処理時間を実測し、その値をもとにプロセッサ部の能力見積りを行った。

処理時間を測定するにあたって、i860 プロセッサを用いた市販のボード (i860XR 33.3MHz) を利用し、1 プロセッサで実行した。

テストデータとしては 5,760 Polygon のデータを 4,8,12,16 分割して用い、結果をそれぞれ 4,8,12,16 プロセッサで処理した場合の処理時間と見做した。

処理時間の測定は C のソースプログラムをコンパイルしたものと、その一部をアセンブラで書き換えたものについて行った。

アセンブラ化は座標変換および透視変換について部分的に行った。これは、座標変換、透視変換の処理のほとんどが浮動小数点演算であり、パイプライン命令と並列実行命令 (整数演算と浮動小数点演算を同時に実行する) を用いることにより処理時間をかなり改善することができるためである。

測定した処理時間を表 2 に示す。

アセンブラ化した場合のポリゴンの生成速度は約 1.25 倍程度高速化されており、座標変換と透視変換の他にクリッピングの処理もアセンブラ化すれば一層の高速化が望める。

以上の結果より、プロセッサあたりの処理能力の見積りは 14.4K Polygons/sec 以上となる。したがって、前節の結

表 2: プロセッサ部の処理能力見積り

プロセッサ数	C	Assembler	プロセッサ 当りの処理能力
4 プロセッサ	0.107sec	0.090sec	16.0K Polys/sec
8 プロセッサ	0.054sec	0.050sec	14.4K Polys/sec
12 プロセッサ	0.037sec	0.033sec	14.5K Polys/sec
16 プロセッサ	0.029sec	0.025sec	14.4K Polys/sec

果と合わせるで考えると、セグメントジェネレータとスキャンライン Z バッファマージャを組み合わせたポリゴン処理ユニットに対して、i860 プロセッサ 8 個でバランスするという計算になる。この見積りはプロセッサ間バスの転送能力が十分にあると仮定した場合の結果であり、実際にはプロセッサ間バスの設計によって制限を受けるため、もっと少ないプロセッサ数でバスが飽和すると考えられる。

7 まとめ

多数のポリゴンに対するマッピング処理を高速に並列処理できるアーキテクチャとしてスキャンライン Z バッファマージャを提案し、さらに試算による処理能力の見積りを行った。

ZB/M アーキテクチャの特徴は、

1. マッピング処理の集中化
2. ダブル Z バッファによる α 合成処理
3. Z マージ処理による効率的な並列処理
4. ZB/M のインタリーブによるシステム構成のフレキシビリティ
5. セグメントジェネレータによる効率的なセグメント生成である。

セグメントジェネレータとスキャンライン Z バッファマージャを組み合わせたポリゴン処理ユニットあたりの処理能力は最大 115.8K Polygons/sec と試算でき、このユニットを 64 ユニット程度並列動作させることにより 7.4M Polygons/sec の処理能力が実現できる見込みである。

また、処理時間の実測値をもとにした見積りにより、このポリゴン処理ユニットに対してバランスするデータ供給を行うためには、i860 プロセッサ 8 個が必要との結果が得られた。

現在、ZB/M 機能を持つ VLSI を開発中であり、ロジック部約 27,000 ゲート、RAM 部 49Kbits で実現できる見込みである。

今後、ZB/M VLSI を実装した画像生成システムを開発し、その評価を進めてゆく予定である。

8 謝辞

本研究を進めるにあたり、御助言、御協力を頂いたメディア研究所の峰久、平井、西村健二、望月、中瀬、太田、日高、浅原の各氏に感謝いたします。

参考文献

- [AJ88] Kurt Akeley and Tom Jermoluk. High-performance polygon rendering. *Computer Graphics*, Aug. 1988.
- [FvDFH90] Foley, van Dam, Feiner, and Hughes. *Computer Graphics principles and practice second edition*. ADDISON WESLEY, Reading, Massachusetts, 1990.
- [KWG92] Michael Kelley, Stephanie Winner, and Kirk Gould. A scalable hardware render accelerator using a modified scanline algorithm. *Computer Graphics*, 1992.
- [Nea88] Teiji Nishizawa and et al. A hidden surface processor for 3-dimension graphics. *Digest of Tech. Papers, IEEE ISSCC*, pp. 166-167, 1988.
- [Sil92] Silicon Graphics Inc. *Data Sheet: Reality Engine Visual Simulation*, 1992.
- [安部 89] 安部, 他. 画像生成システム S I G-2. 情処学会 グラフィックスと CAD 研究会, 1989.
- [佐々92] 佐々木, 他. 超高速CGシステム『昴』. 第2回ハイビジョン研究会, 1992.
- [鷺島 91] 鷺島, 西澤, 浅原. 並列図形処理. コロナ社, 東京都文京区千石, 1991.
- [鹿野 88] 鹿野良夫. コンピュータグラフィックスとエンジニアリング・ワークステーション. 情報処理, Oct. 1988.
- [若山 90] 若山, 他. パソコン用3次元グラフィックスボードの開発. 信学会 マイクロコンピュータとワークステーション研究会, 1990.
- [前信 90] 前信, 他. 3次元グラフィックス動画システム C U B E. 情処学会 グラフィックスと CAD 研究会, 1990.
- [日高 85] 日高, 他. マルチコンピュータ構成画像生成システム M C-1. 情処学会 第31回全国大会, 1985.