

高速1チップ3D-CGレンダリングプロセッサ 「AGP」の開発

吉沢 英樹, 大塚 竜志, 佐々木 繁
(株)富士通研究所 マルチメディアシステム研究所

業界標準であるOpenGLをはじめとして、多様化するCGアプリケーションへの対応とポリゴン性能445K
グーロシェードポリゴン/秒、テクスチャ性能15メガテクセル/秒の高速性能を1チップで実現する3次元CG
レンダリングプロセッサ"AGP"を開発した。AGPは2命令並列実行可能なプロセッサコアとOpenGLに準拠した3DCG
専用パイプライン回路で構成される。使用テクノロジーは、0.5 μ CMOSスタンダードセル。ゲート数254,000ゲート、
動作周波数60MHzで消費電力5Wである。304ピンセラミックQFPに封止した。

High performance architecture of 3D-CG rendering processor "AGP"

Hideki Yoshizawa, Tatsushi Ohtsuka, and Sigeru Sasaki
Fujitsu Laboratories LTD.
1015 Kamikodanaka, nakahara-ku, Kawasaki 211, Japan

Abstracts:

The AGP (Advanced Graphics Processor) has been developed as a one-chip 3D-graphics rendering processor which can support not only the OpenGL functions but also many kinds of applications. Only one AGP can achieve 445 Kilo-polygons/sec when an isolated 25pix Gouraud shaded polygon is processed, and achieved 15 Million Texcels/sec when an isolated 200pix Gouraud shaded polygon is processed. The AGP consists of an original micro processor core and a 3D-CG pipeline circuit block based on the OpenGL functions. Its total number of gate is about 254,000, with 0.5 micron-CMOS standard cell technology. It spend power of 5 Watts at 60MHz operation with a 304pin-SQFP package.

1. はじめに

筆者らは、ワークステーション領域の性能を持ちながら、パーソナルコンピュータに適用できる、高性能3次元グラフィックスプロセッサ「AGP」を開発した。

これまでワークステーションを中心に発展してきた3次元CG技術は、米マイクロソフト社によるPC市場への採用宣言や、セガ、ソニーなどの3次元CGゲーム機の普及によって、加速度的に広まりつつある。またコンピュータグラフィックスを始めとするマルチメディア用途を意識した製品開発も活発化している。例えば、米HP社¹⁾やサマシクシステム社²⁾のRISCプロセッサでは配列演算を高速化する命令を追加しており、三菱の3DRAM³⁾のように特定の論理演算機能をメモリバイス上に集積したものなどが挙げられる。これらは全て、特定のアルゴリズムに対する性能向上を狙ったものである。しかし、いずれもコスト面で即座にPC製品に適用できるものとは言えない。

筆者らの目指す次世代3次元CGシステムは、PCやゲーム機に代表されるように、低価格且つ優れた機能拡張性と高性能を兼ね備えたものである。次世代CGシステムは、マルチメディア市場を含むコンシューマ製品市場を中心にパーソナルユースを基本とした製品であって、単に3次元CGを高速に実行できるだけでなく、多様なメディアとの融合処理も可能であることが求められる。

その例として激化するゲーム機市場を挙げることができる。最近のゲーム機では、今までワークステーションで見られなかった3次元グラフィックス技術をわずか数万円の家庭用ゲーム機に搭載している。ゲーム機のCPUは、廉価版マイクロプロセッサであるという認識は、もはや時代錯誤である。今や32ビットCPUは常識であり、最先端のRISCプロセッサを使用している。基本システム構成もパーソナルコンピュータと区別が難しくなっている。つまり、市場を区分している要因は、もはや価格だけであり、技術面ではすでにホーナスになっているのである。したがって、ワークステーションで培われた業界標準のグラフィックスライブラリは、パーソナルコンピュータやゲーム機という枠組を越えて広まろうとしている。このような背景から、筆者らは業界標準のOpenGLに基づいて3次元グラフィックス技術の中核とし、様々なアプリケーションに対応できる、高速1チップ・プロセッサ⁴⁾3次元CGレンダリングプロセッサ「AGP」を開発した。以下、本論文において、LSIアーキテクチャの詳細と性能評価について述べる。

2. AGPアーキテクチャ

2.1 次世代3次元CGシステム対応機能のLSI化

低価格の次世代3次元CGシステムの実現において主な技術的課題は、

- (1) 多様なアプリケーションに対応できる 機能拡張性、柔軟性を持つこと。

- (2) 基本性能は、専用デバイスを凌駕する高性能を持つこと。

- (3) システムコスト低減を実現するため、1チップ化あるいは、低コスト化を実現すること。

であると考ええる。

第1の課題は、 μ プロセッサを用いてプログラムで対応することで容易に解決できる。しかし、第2の課題は μ プロセッサの設計指針と相入れない内容である。何故ならば、本来 μ プロセッサは特定の演算機能に対する専用設計を行なう代わりに、プログラムによって、論理的に等価な機能を実現し、機能拡張性と柔軟性を得ているため、専用デバイスに対して性能面で劣ることは避けられない。最近では、データキャッシュによる高速アクセスや動的命令スケジュールング技術、新たな高次の命令追加などによって汎用 μ プロセッサの潜在的な性能は向上の一途をたどっている。そのため、汎用プロセッサで全てを処理するほうが合理的であると考えられる風潮がある。しかし、多種多様なアプリケーションに対応すること引き換えに、プロセッサが持っている潜在能力を無駄にしていることを否定できない。

例えば、動画伸長処理^{4,5)}におけるIDCT(離散型逆コサイン変換)やCG処理におけるZバッファアルゴリズムなどの処理では、データアクセスは決して最適に行なわれているとは言えない。データアクセスに空きができれば、どんなに演算が高速であっても、プロセッサの実効性能は低下してしまう。特に次世代CGアプリケーションのように、テクスチャ処理、ステレオ画像表示など、大量のデータアクセスを必要とする場合には、メモリアクセスがボトルネックとなって速度低下は著しいものとなる。確かに科学技術計算のように1度使用したデータを再利用する可能性の高い処理では、データキャッシュ等のアクセス高速化技術は有効である。ところが、CGアルゴリズムや動画伸長処理のようにデータの再利用が余り期待できない処理では、その効果は殆んど期待できない。つまり、次世代CGシステムでは、プロセッサの多命令実行制御技術などによる基本性能向上も必要であるが、それは間接的な改善効果に留まる可能性が高く、抜本的改善は、データアクセス性能をいかに向上させるかに掛かっている。したがって、汎用 μ プロセッサを用いて第2の課題を解決することには無理がある。仮に第1の課題を汎用 μ プロセッサで解決し、第2の課題を専用ASIC等で解決する場合を考えると、システムを構成する部品数が増加し、高いシステム性能を得ることが難しくなるだけでなく、システムコストの上昇を招く。したがって、第3の課題を解決できなくなる。このように、今までのシステム構成法では、上記の3つの課題のジレンマに陥る。

これらを解決する方法として、 μ プロセッサコアとCG専用ハイブリッドを併合させたデバイスアーキテクチャを考案した。

このアーキテクチャは、今までのエンベデッドプロセッサASIC等のデバイスと異なり、並列協調動作を実現できるよう μ コ−ドレベルまで踏み込んだ工夫が施されている。これにより、 μ プログラムによる機能拡張性と専用回路による高速性を両立し、しかも0.5 μ CMOS技術によってチップに集積することにより低価格システムの構築でき、先の3つの課題を解決することが可能になった。

2.2 3次元CGアルゴリズムと並列アーキテクチャ

次世代CGシステムでは、多彩な機能と高性能、及び低価格システムを実現する解として μ プロセッサとCG専用パイプライン回路を備えたLSI「AGP (Advanced Graphics Processor)」を開発した(図1)。

AGPは、アプリケーション依存型プロセッサアーキテクチャを設計コンセプトとしており、3次元CGアルゴリズムにおけるデータアクセスを効率よく実行できるよう、アーキテクチャ及び命令コード上の工夫が施されている⁹⁾。

筆者らは、3次元CG処理フローをラスタライズ処理を境に2つに大別した。AGPは、ラスタライズ以降の後半部分を対象にして設計されている。前半部は、主に幾何変換処理で構成され、通常は汎用 μ プロセッサや浮動小数点DSPによって処理される。AGPが守備範囲とする後半部は、DDA処理をはじめとした固定小数演算と、フレームメモリやテクスチャメモリなどメモリに密着した処理を行なう。この処理は、グラフィックスイッチリ毎に微妙に異なるラスタライズルールと処理アルゴリズムに依存した高速のメモリアクセスを必要とする。Zバッファ処理やDDA処理のように単純かつ頻繁にメモリアクセスし、データの再利用が殆んど望めない処理は、大型データキャッシュを備えた汎用プロセッサでも多大なオーバーヘッドを発生し、高速化が難しい。そこで、筆者らは更にラスタライズ以降の後半部をDDA処理を境にして2分割し、後半部分をCG専用パイプライン回路として構成し、前半部分を μ プロセッサでプログラムによって実現する並列アーキテクチャを開発した。DDA処

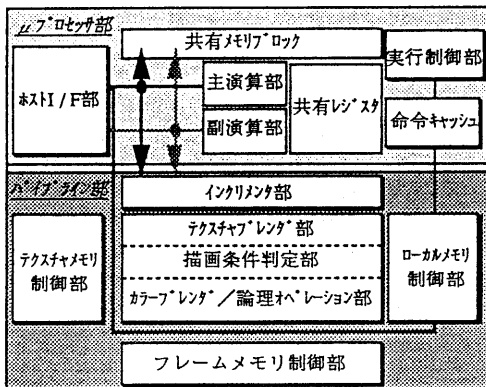


図1 アーキテクチャ

理以下は、1画素に付随する多数のパラメータ(色、Z値、テクスチャ座標など)を扱う処理であり、高速性が要求される。AGPでは、各メモリアクセスを最適制御できるハードウェアを装備することで並列処理による高速化を行なっている。

一方、前半部は、幾何変換処理との境界にありデータの通信・変換処理や、ラスタライズルールの微妙な違いを保証し、上位系の処理内容に対し柔軟に対処できるよう機能拡張性が求められる。AGPでは、ローカルメモリに蓄えられたユーザープログラムで対応する。以下に三角形ポリゴンを描画するアルゴリズムを例にAGPによる並列処理方式を説明する。

まず上位系(幾何変換部)とは図2のようなデータブロック(パケット)を介して通信により結ぶ。AGPは、通信制御部によってパケットデータを解釈変換して通信制御部内のパケットバッファにデータを蓄える。1つのパケットの入力処理が完了するか、またはパケットバッファ一杯になった時、 μ プロセッサ部のステータスレジスタに表示される。ユーザーは、この情報を調べて処理を行なうように予め μ プログラムを作成しておき、パケットの形式に応じて処理を選択し様々な処理機能を実現することができる。図3は、三角形ポリゴンを処理する場合、上位系から渡されるデータの一例を示したものである。上位系から渡されるパケットは3つの色・位置勾配と頂点情報を含んでいる。

図4にAGPでポリゴンを描画する場合、 μ プロセッサ(プログラム部分)とCGパイプライン部が、どのように協調動作するかを示した。この場合、AGPでは、各ラスタ(横線)の端点データを μ プログラムで算出し、DDA処理の初期値と増分、ラスタ方向の繰り返し回数などをCGパイプラインに設定して、非同期実行を指令する。 μ プロセッサ部は、予めプログラムされた端点演算処理をCGパイプラインとは並列に実行し、DDA処理部にあるダブルバッファされたレジスタに次のデータを書き込む。先にも触れたが、この端点計算方式やラスタ処理の終了位置の決定ルールは、グラフィックスイッチリによって微妙に異なり、AGPはその違いを μ プログラムによって吸収可能にしている。

第1ワード	0@11111111	データ数
第2ワード	0	データ (IEEE単精度浮動小数点形式)
第3ワード	1	データ (32ビットバイナリ形式)
...		
第nワード	0	データ (IEEE単精度浮動小数点形式)
	0@11111111	データ数

図2 データパケット

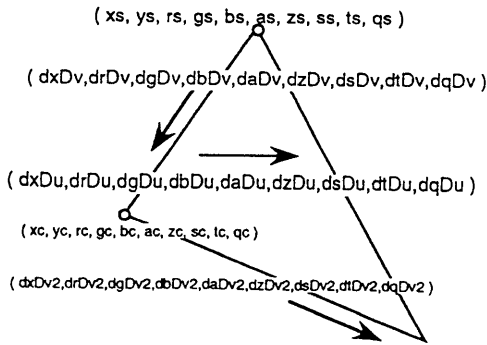


図3 ポリゴンデータ

図中色づけされた時刻で、CGパイプラインによる1ラスタ分の処理が完了する。ラスタ処理の完了は、パケット受け取り通知と同様にステータスレジスタ内に完了フラグとして伝えられる。したがって、待ち合わせ条件にこのフラグを用いた待ち合わせ命令(SLEEP命令)を用いることによって同期をとることができる。ラスタ処理の完了が確認された後、CGパイプラインは、設定された初期値と増分値に基づいて次のラスタ処理を開始し、 μ プロセッサ部は、新たな端点計算を開始する。

このようにAGPは、ラスタ処理を境界として、性質の異なる2つの回路を協調実行できるよう設計されており、これによって互いに他のデメリットを補っている。即ち、 μ プロセッサが苦手とするメモリに密結合した処理を高速の専用パイプライン回路に任せる代わりに、パイプライン回路の欠点である機能拡張性を μ プロセッサ部で確保しているのである。しかも、各々がCGレンダリング処理という1つのパイプライン処理の中にあって、並列かつ協調して実行している。次にLSIの詳細構成について触れる。

3. LSIの構成

3.1 CG専用パイプライン

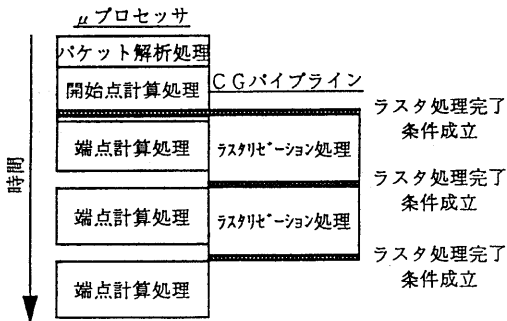


図4 協調動作

AGPのアーキテクチャをCGパイプライン処理に対応させた場合の各部の流れを図5に示す^{9,10}。CG専用パイプライン回路は、DDA処理部、テクスチャレンダ部、描画条件判定部、 α レンダ部、及び3つのメモリコントロール部からなる。

DDA処理部は、11個のインクリメントを装備し、1クロック毎に描画位置、色データ、テクスチャ座標、デプスキュー係数などを算出して、これに繋がるテクスチャレンダ部にピクセル単位で転送する。テクスチャレンダ部は、テクスチャ座標値からテクスチャ格納アドレスを計算し、テクスチャ格納用メモリを制御する専用メモリコントローラに与えて高速にアクセスし、テクスチャデータと、DDA処理部より与えられる色データに対するブレンド処理を実行する。

また、描画条件処理部は、Z値、ステンシル値、ビットマスク値など各種制御データをローカルメモリから読みだし、描画制御を行う。描画条件判定処理後のピクセルは、更に α レンダ部、論理演算部で処理されフレームメモリコントローラを介して描画される。このパイプライン回路は、LSI内部バスを経由して制御レジスタ設定を行なう方法とフラグを付けた各種設定データを副演算部内の特殊レジスタ(CONFレジスタ)に書いて設定する方法の2つの方法によって動作設定を行なう。動作設定を行なった後は、パイプライン動作を1画素単位で繰り返していく。ただし後に述べる μ プロセッサ部で実行されるプログラムとは完全な非同期実行が可能であるため、様々な描画条件変更や制御モードの変更は、通常CONFレジスタ経由の方法で行なわれる。この方法では、パイプラインの中を直前に送り込まれた処理要求を追い越すことなく実行されることが保証されるためユーザは、フラグメント単位またはラスタ単位でパイプライン動作を乱すことなく設定を変更することができる。さらに、パイプライン回路がアクセスするメモリ群は用途別に、テクスチャを格納するテクスチャメモリブロック、色データ、描画制御用データ、 μ プログラムなどを格納するローカルメモリブロック、表示データを格納するフレームメモリブロックの3つに分割し、互いに並列アクセスを可能にした。ローカルメモリに格納されたAGPを制御する μ プログラムは、パイプラインが必要とする各種データとのアクセス競合を避けるた

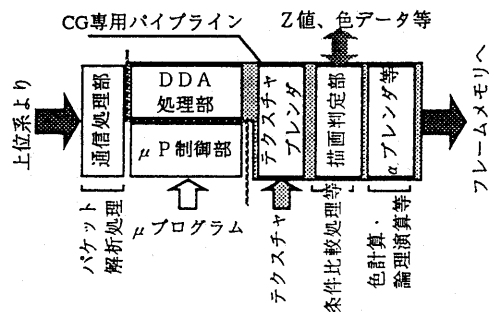


図5 データフロー

め、256ワードの2ウェイ命令キャッシュに格納・制御される。このように、CG専用パイプライン回路は、パイプライン動作に合わせて合理的にデータアクセスできるように設計されており、テクスチャデータや色データ、制御データなどのアクセス競合を回避して高速実行を実現している。

3.2 μプロセッサコア

AGPは、ライブラリごとに微妙に異なる描画制御仕様を吸収し、複雑なデータ構造を持った情報や音声などの他メディアデータとの連携処理を扱えるよう主・副2つの演算部を備え、64ビットLIW形式の命令で動作するデジタルシグナルプロセッサによって制御される⁹⁾。

このプロセッサ部は40ビット固定小数点データを標準として主演算部にALUとバレルシフト、副演算部にALUとバレルシフト付き乗算器を備えている。主・副各々に8本のローカルレジスタと、主・副どちらからでも同時に読み書き可能な16本の共有レジスタを持つ。その他、機能レジスタとしてデータセグメントレジスタ、スタックポインタ、ステータスレジスタ、2本のインテックスレジスタ、割り込みマスクレジスタを主演算器系に、パイプライン回路を制御するレポートカウンタレジスタ、コンフィギュレーションレジスタ、DDA処理部に組み込まれたパラメータ初期値、増分値を保持するレジスタ、2本のインテックスレジスタを副演算器系にそれぞれ割り付けられている。よって、μプロセッサ部で計算されたデータは、直接DDA処理部に書き込みができる。しかも、DDA処理部のレジスタは、ダブルバッファ構成であるため、パイプライン動作中でも書き込み可能である。したがって、オーバーヘッド無しでCGパイプラインに転送することができる。

また、主・副各々から並列アクセスできる256ワードの共有メモリが装備され、主・副が独立した内部バスに接続されており、並列にメモリアクセスが可能である。主演算器側バスには、ローカルメモリ、フレームメモリ、テクスチャメモリ、バケットバッファ、共有メモリ、各種制御レジスタが割り付けられ、副演算器側バスには、バケットバッファ、共有メモリのみがマップされている。

このようにしてAGPは、命令実行やデータ管理などを主演算部で実行し、パイプライン制御のための処理を副演算部で実行する方式に適したアーキテクチャで構成されている。命令は3つの部分から成り、主演算制御系、副演算制御系、CGパイプライン制御系を制御する。(図6) この内、主演算制御系命令によってプログラムの実行制御され、他の2つの命令実行順序は、これ

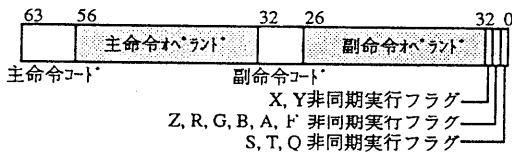


図6 命令形式

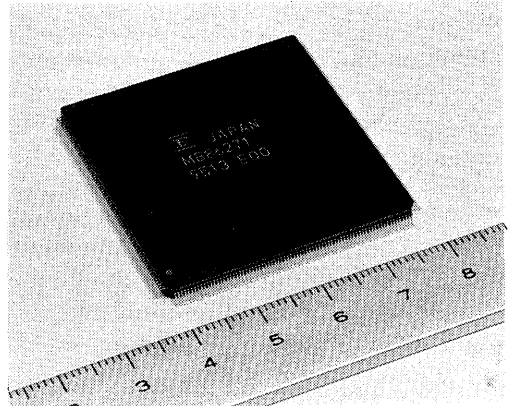


図7 LSI外観

に従う。したがって、主または副いずれかの命令でパイプライン拡張(パイプライン例外)が発生した場合、他方はそれに追従して命令実行を遅らせるよう自動的に制御される。これによって、ユーザが記述した主副の命令実行順序を保証することができる。主演算系命令80種、副演算系命令40種、CGパイプライン系命令3種を自由に組み合わせることができる。

命令は4ステージパイプラインで処理される。LIWであるため主・副の命令実行は、同期して行なわれる。CG専用パイプライン回路の実行制御は、命令の下位3ビットに設けられたフラグによって行なわれる。先の主及び副演算器とは異なり、このビットがセットされた時点でμプログラムの実行とは独立に処理を開始する。処理の停止条件は、副演算部にマップされたレポートカウンタレジスタの値が負になるか、またはパイプライン強制停止命令(PABORT命令)によって停止する。メモリ空間は16Mワード。外部・内部割り込み機能を持つ。使用テクノロジーは、0.5μCMOSスタンダードセル。ゲート数は、約25,400ゲート、消費電力5W(60MHz動作時)で、パッケージは304ピンセラミックQFPである(図7)。

4. 評価シミュレーション

AGPの性能を検証するため、評価シミュレーションを行った。シミュレーションは、LSI内部状態の詳細を調べるため、ポストレイアウト遅延情報(実遅延時間)を用いてVerilogシミュレータにより実施した。図8は、独立ホリゾン描画性能、図9は、テクスチャ付き独立ホリゾンの描画性能である。

その結果、445Kゲートホリゾン/秒、15Mピクセル/秒(200ピクセルテクスチャ付きゲートホリゾン)を達成できることが確認された。さらに、メモリアクセスを伴うCGパイプラインとの協調動作時とμプログラムのみ(非協調動作時)の性能を比較した場合、25ピクセルのゲートホリゾンを除きμプロセッサ部の処理能力にはゆとりがあることが分かった。DDA処理をハードウェア化しているため、ゲー

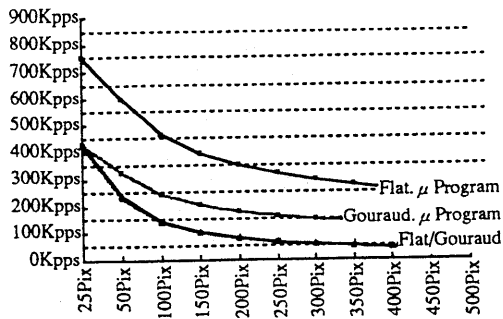


図9 独立ポリゴン描画速度

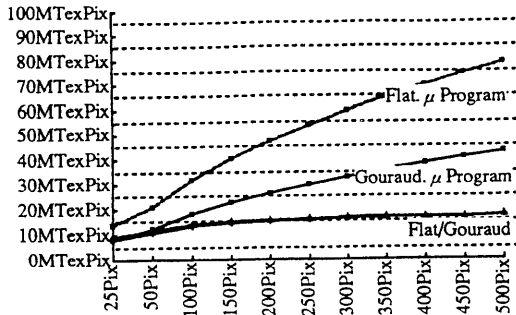


図10 テクスチャ描画速度

ポリゴンをコンスタントポリゴンと同等の速度で処理できることも確認された。

一方、ゲージ及びコンスタントポリゴン処理時の処理時間内訳を調べると、コンスタントポリゴンの場合で50ピクセル、ゲージポリゴンで100ピクセルで μ プロセッサ部とCGパイプラインの処理時間がほぼ同じになることが分かった。つまり、50から100ピクセル以下のポリゴンでは、 μ プロセッサ部の速度が支配的であり、それ以上ではメモリアクセス速度が支配的であることが分かる。

そこで、各メモリアクセスに要する時間を調べた(図10)。この図から分かるようにAGPが速度制限を受ける100ピクセル以上ではフレームメモリの消費する時間が増加していく。この時間は、フレームメモリの限界速度には

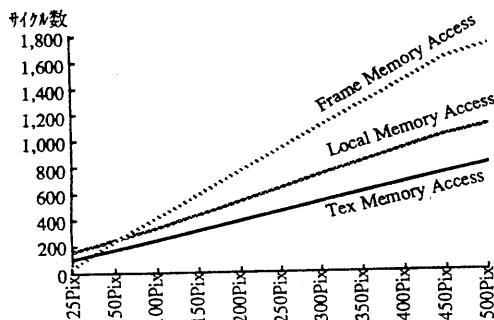


図10 機能別メモリアクセス速度比較

は等しい。したがって、AGPによってCG処理性能がメモリアクセス速度の限界まで引き出せることが明らかになった。

5. まとめ

以上のように、CGアルゴリズムを考慮し、 μ プロセッサと専用パイプラインを用いたアーキテクチャにより、445 Kゲージポリゴン/秒、15 Mテクセル/秒という、極めて高速且つ機能拡張性をもつアーキテクチャを開発できた。

またAGPのように、 μ プログラムによる制御を行ってもCGパイプライン回路との非同期動作によって実用領域(100ピクセル以上)においてプログラムオーバーヘッドは、問題にならないことも明らかになった。

以上より、AGPは1チップLSIとしてメモリ性能限界に近く、極めて優れた実用性能を持つプロセッサであると結論できる。

参考文献

- 1) R. Lee, Multimedia Enhancements for PA-RISC Processor, Symposium Record of IEEE HOT Chips VI, pp.183-192, 1994.
- 2) A. Chamas, et al., A 64b Microprocessor with Multimedia Support, Proc. of ISSCC '95, pp.178-179, 1995.
- 3) K. Inoue, et al., A 10Mb 3D Frame Buffer Memory with Z-Compare and Alpha-Blend Units, Proc. of ISSCC'95, pp.302-303, 1995
- 4) 北垣, 「マルチメディア画像LSI」, 電子情報通信学会誌, Vol.78, No.1, pp.76-84, 1995
- 5) 枝, 原田, 「低価格WSの3次元グラフィックス性能向上でポリゴンの追撃を振り切る」, 日経エレクトロニクス 1994年10月24日号, no.620, pp.75-92
- 6) J. L. Hennessy, et al., 「コンピュータ・アーキテクチャ」, 日経BP社, 1994
- 7) 「最新ASIC設計術'94」, 日経BP社, 1993
- 8) K.R.Rao, et al., 「画像符号化技術」, オーム社, 1993
- 9) J. Neider, et al., OpenGL Programming Guide, Addison-Wesley Publishing Company, 1993
- 10) OpenGL ARB, OpenGL Reference Manual, Addison-Wesley Publishing Company, 1992