

並列可視化処理向け FPGA 搭載 PCI カードへの ボリュームレンダリングの予備実装

岡村 大[†] 五島 正裕[†] 森 眞一郎[†] 中島 康彦^{††} 富田 眞治[†]

[†] 京都大学大学院情報学研究科 〒 606-8501 京都市左京区吉田本町

^{††} 京都大学大学院経済学研究科 〒 606-8501 京都市左京区吉田本町

E-mail: †{okamura,goshima,moris,nakashim,tomita}@lab3.kuis.kyoto-u.ac.jp

あらまし 本稿では, Dual Link の DVI-D 入出力インターフェースと DDR-SDRAM を 2 系統搭載し, 高いメモリバンド幅を要求する並列画像処理向けに開発した FPGA 搭載 PCI カードを紹介する. また, その具体的な応用として, 並列ボリュームレンダリング向けの専用アクセラレータをこのボードに実装する構想を述べ, その簡易版として制作したボリュームレンダリング回路の紹介を行う.

キーワード 可視化, ボリュームレンダリング, 並列処理, FPGA, ハードウェアアクセラレータ

Preliminary Implementation of Volume Rendering Circuit onto an FPGA-based Visualization Accelerator

Dai OKAMURA[†], Masahiro GOSHIMA[†], Shin-ichiro MORI[†], Yasuhiko NAKASHIMA^{††}, and
Shinji TOMITA[†]

[†] Graduate School of Informatics, Kyoto University, Kyoto-shi, 606-8501, Japan.

^{††} Graduate School of Economics, Kyoto University, Kyoto-shi, 606-8501, Japan.

E-mail: †{okamura,goshima,moris,nakashim,tomita}@lab3.kuis.kyoto-u.ac.jp

Abstract This paper introduces an FPGA-based PCICard for Parallel Visualization of Large Volume Data. In order to implement memory intensive image / vision processing applications, like parallel volume rendering, this card is configured with DVI-D Dual Link Input / Output interfaces and two independent channels of DDR-SDRAM. Then we show our preliminary implementation of a simple volume rendering circuit onto the PCICard.

Key words Visualization, Volume Rendering, Parallel Processing, FPGA, Hardware Accelerator

1. はじめに

近年の計算機性能の急速な向上に伴い, 大規模かつ高精度な数値シミュレーションへの期待が高まっている. なかでも, 実時間の数値シミュレーションの可視化技術は, 大規模な 3 次元データの処理を伴う医療などの分野において, 現在研究が進められている. これまでも, PC クラスタ等の並列計算機環境を利用した, シミュレーションとその実時間可視化のためのシステムが開発されており, 次世代シミュレーション技術として, 従来の実験の代替手段となりうる「仮想実験型/仮想体験型のシミュレーション環境」の構築が望まれている. このような次世代のシミュレーション環境では, オペレータによるシミュレーション対象へのインタラクティブな操作に対応して, 実時間でシミュレーションを行うとともに, 即刻その結果を可視化など

の手段により提示することが求められる.

我々は, 個人あるいは小規模な組織単位で占有利用可能な PC クラスタを用いて, インタラクティブな数値シミュレーション及びその可視化を実時間処理する環境について研究を行っている. 大容量の数値データを一旦分散させ処理し, 再度可視化のためにデータを集約した上で表示機器に出力する, という一連の処理には高速な伝送路と実時間処理可能な可視化機構が不可欠である.

上記の可視化機構を実現可能なハードウェアアーキテクチャとして我々は VisA を提案した [1], [2]. そのプロトタイプとして処理の手順をそのままにボリュームデータの処理可能サイズを半分に制約した VisA Pro を実装し, VisA についての評価を行おうとしている. VisA は専用 ASIC で作ることを前提としたアーキテクチャであるが, 実験段階に適した FPGA 上へ

の実装を考え、高速入出力リンクと大容量高速メモリを備えた汎用 FPGA 搭載 PCI カードを東京エレクトロニクス(株)との共同企画で開発した。

本稿では、上記ボードを VisA のプロトタイプとして用いる方法を紹介します、我々が提案してきたボリュームレンダリング専用並列計算機 ReVolver/C40 アーキテクチャ [3]~[5] をこのプロトタイプ上で走らせる構想を述べる。また、ボリュームレンダリングの予備実装として制作した回路を紹介する。

章構成は以下の通りである。まず 2 章でボリュームレンダリングの概要について述べ、3 章で今回実装に用いる FPGA 搭載 PCI カードの詳細と VisA Pro の実装方法を紹介します。4 章で今回予備実装として制作した回路について説明し、5 章でまとめを行う。

2. ボリュームレンダリング

2.1 処理の概要

ボリュームレンダリングでは、シミュレーション等により得られた 3 次元空間上の数値データを、色 C と透明度 t に対応づけて可視化することで、3 次元空間内部のデータの分布状況を可視化する。具体的には、視線上のボクセルの値を視点に近い順に v_0, v_1, v_2, \dots とするとピクセル値は次の式 (畳み込み演算) で計算される。

$$C_k = \sum_{i=0}^k (1 - t(v_i)) \cdot c(v_i) \cdot \prod_{j=0}^{i-1} t(v_j) \quad (1)$$

ここで $c(v_i), t(v_i)$ はそれぞれボクセル値 v_i を、色、透明度に変換して値であることを示している。さらに、式 (1) は以下のような漸化式で表わされる。

$$C_k = C_{k-1} + (1 - t(v_k)) \cdot c(v_k) \cdot T_{k-1} \quad (2)$$

$$T_k = t(v_k) \cdot T_{k-1} \quad (3)$$

この畳み込み演算は、演算区間をいくつかの部分区間に分割し、それぞれの区間に対する計算結果に対して、再度畳み込み演算を行うことが可能であるという性質がある。そこで、1) 複数のノードに分散して割当てた部分 3 次元空間 (以下 サブボリュームと呼ぶ) に対して畳み込み演算を行い、2) 各ノードで得られた画像と画素毎の透明度を、視点からの距離の順番に従って順次パイプライン的に合成 (composition) する、という手法で並列処理が可能である。

このようなボリュームレンダリング処理の実装に際しては、可視化用のハードウェア・アクセラレータを用いるか否か、また、ハードウェア支援を行う場合に、テクスチャベースの汎用グラフィックスカードを用いるか [6], [7]、あるいは、ボリュームレンダリング専用のハードウェアで実装するかの選択が可能である。現在我々は、これら 3 つのケースに関して各々並列可視化環境の構築を行なっている。

本稿では、その中でもボリュームレンダリング専用のグラフィックスカードを用いた実装について述べる。

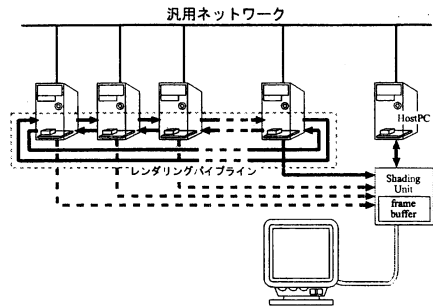


図 1 実時間可視化システムの構成

2.2 ボリューム・グラフィックス・カードを用いた可視化システム

専用ハードウェアによる可視化システムは、我々が既に開発した ReVolver/C40 で採用したアーキテクチャをベースにしたもので、 $N \times N \times L$ のサブボリューム単位で並列化し、Ray Casting アルゴリズムを用いて、1 ピクセル分のピクセル値計算をサブボリューム単位でパイプライン処理することで目標とする描画速度を得る。N=4096 とした場合、可視化システムの構成としては 128 ノード構成の PC クラスタに、ボリュームレンダリング向けアクセラレータ (VisA: Visualization Accelerator) を装備し、VisA 間を双方向高速リンクで接続する。計算結果は VisA 間リンクと同一規格のケーブルによりフレームバッファに送られ、ディスプレイへ表示される。生成された 2 次元画像に対して、さらに後処理が必要な場合は、Host PC に送り汎用のグラフィックスカードを用いて処理を行う。

ReVolver/C40 では、視線生成、ピクセル値計算、シェーディングの 3 ステージをそれぞれ専用のハードウェアを開発し構成していたが、VisA ではピクセル値計算のみを重点的に専用ハードウェア化し、その他のステージで必要であった処理は各 PC の CPU や汎用グラフィックスカードを用いて実行する。以下、VisA の主要構成要素について簡単に説明を行う。

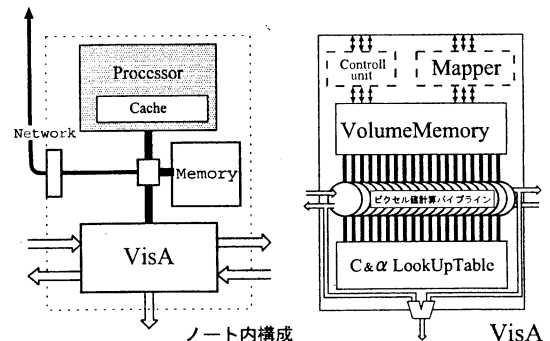


図 2 内部構成

(1) ピクセル値計算パイプライン：32個のピクセル値計算ユニット(PCU)をパイプラインに接続して構成する。表示に必要なRGBは8bitであるが、誤差伝播の影響を軽減するため16bit固定小数点として色と透明度の演算を行う。パイプライン周波数は画面サイズとフレームレートから128MHzとなる。

(2) Look Up Table (LUT C&A)：RGB各8bitの色情報と8bitの透過率を保持する256エントリのルックアップテーブルである。各ピクセル値計算ユニットが1ボクセルの演算を行う度に1回参照されるため、スループットの点ではボリュームメモリの4倍の性能が要求されるが、小容量のメモリであるためマルチポートRAMを複数個用いて実装する。

(3) ボリュームメモリ：ボリュームデータを格納するための容量2GBのメモリで、 $4000^2 \times 32$ のサブボリュームを4セットまで格納可能とする。ノード内のピクセル値計算ユニットに対して4GB/sのバンド幅を確保するため、4バンク程度のメモリバンク構成とする。

(4) プリフェッチ機構：ボリュームメモリはパイプライン構成している全てのピクセル値計算ユニットから同時にアクセスされるため、所望のパイプライン周波数を実現するために、LUTとボリュームメモリの間にプリフェッチバッファ(PFB)を装備し、ボリュームメモリへのアクセス遅延に伴うパイプラインストールを最小化する。

(5) VisA 間リンク：このリンクは、レンダリングパイプラインにおいて計算途中の色情報、透明度情報、Z値を送るためのリンクであり、約1GB/sの転送速度が必要である。各ノードに隣接ノード間の双方向リンクと、フレームバッファへの出力ポートを設ける。これにより、ReVolver/C40で対応が困難であった種々のレンダリングアルゴリズムに対応する[3], [5], [8]~[11]。具体的には、DVI規格に準拠したLVDSインタフェースを用いてネットワークを構成する。

(6) 制御ユニットおよびMapper 制御ユニットは視線情報を始めとする描画に関する情報をCPUから受け取り、VisA全体の制御を行う。MapperはCPU側からボリュームデータを受け取り、ボリュームメモリに格納する。アクティブレンダリングを行う場合において、シミュレーション結果からボリュームデータへのマッピング処理が定型的かつ簡易なものであれば、CPU側でのマッピング処理を省略しMapperに直接マッピング処理を行わせることで高速化を図ることができる。この目的のためにMapperにはFPGA的な機能を持たせる。

3. 開発対象FPGA搭載PCIカード

本PCIカードは、東京エレクトロニクス(株)との共同企画で開発を行ったもので^(注1)、高速大容量のDDR-SDRAMを2系統搭載し、高いメモリバンド幅を要求する並列画像処理向けのFPGA搭載PCIカードである。われわれは、このPCIカードを可視化のための専用アクセラレータのプロトタイプと位置づけておりVisA Proカードと呼んでいる。

(注1)：汎用評価ボード(TD-BD-PCI2DVI)として同社より販売している

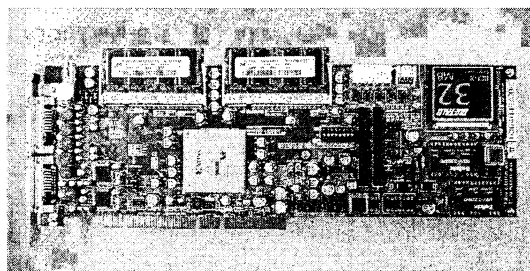


図3 VisA Pro カード (写真提供 東京エレクトロニクス(株))

3.1 カードの仕様

3.1.1 高速大容量メモリ

DDR-SDRAM SO-DIMMを2枚搭載し、別系統クロックにて最大DDR333MHz(PC2700規格メモリ使用時)にて独立に動作可能である。各メモリスロットには、ノートPC用に市販されているメモリ容量1GBまでのDDR SO-DIMMモジュールを搭載可能である。従って、128bit幅の2GBメモリ、あるいは、64bit幅の1GBメモリ2チャンネル等のメモリ構成が可能である。最大メモリバンド幅は2チャンネル合計で5.2GB/sに達する。

3.1.2 DVI-Dデュアルリンクの入出力チャンネル

高解像度液晶ディスプレイに対応したDVI-Dデュアルリンクの入出力チャンネルを持つ。これにより、本カードで生成した画像データを直接ディスプレイに出力することや、グラフィックカードのデジタル出力を一旦取り込んで加工したのちに出力すること等が可能である。

DVI-Dインタフェースで使用するLVDSインタフェースチップとしてはTI社のTFP401/TFP410を各2セット搭載し、デュアルリンク構成時には入出力それぞれ1GB/sの転送速度(165MHz動作時)を実現可能であり、この入出力チャンネルを並列処理のためのネットワークとして使用することも可能である。

3.1.3 高性能大容量FPGA

本ボードは、Xilinx社のVirtexIIシリーズFPGA XC2V6000-5(600万ゲート)を搭載する。同FPGAは、内部に18bit×18bitの高速乗算器と18Kbitのオンチップメモリブロックをそれぞれ144個内蔵し、単体ではともに100MHz以上で動作可能である。これにより画像処理で必要とされる高い整数演算性能とメモリバンド幅を確保することが出来る。また、ソフトウェアプロセッサコアを内部に置くことでより柔軟な数値処理が可能となる。

3.1.4 その他の特徴

- PCIインタフェース：ホストPCとのインターフェースとしてPCI64/66並びにPCI32/33に対応している。
- SSRAM：166MHz動作の512KB(128K×36bit)SSRAMを搭載している。ZBT(Zero Bus Turnaround)タイプなので、Read/Writeアクセス切り替え時のIdleサイクルが不要である。

3.2 VisAの実装

VisA ProはVisAの機能を約半分縮小したもので、基本的な動作はVisAと同様である。図4にReVolver/C40と同じ

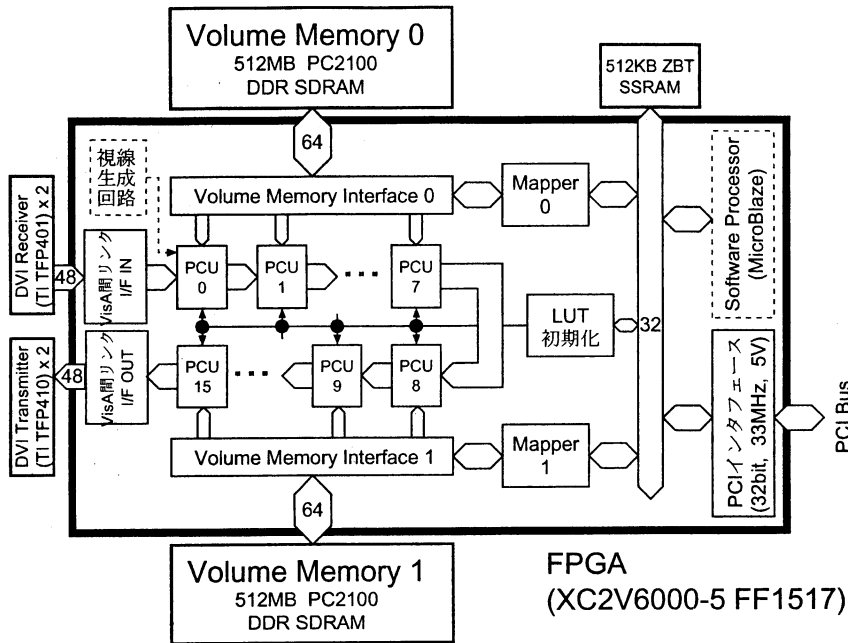


図4 VisA Pro カードブロック図

並列レンダリング・アルゴリズムを採用する場合の VisA Pro の概略構成を示す。

3.2.1 概略構成

一枚の VisA Pro カードには、ピクセル値計算パイプラインの 1 ステージを担当するピクセル値計算ユニット (PCU 図 5) を 16 段実装し、連続する 8 つの PCU がボリュームメモリ (VisA Pro カードでは 2 バンク構成) の 1 バンクを共有する。隣接する VisA Pro 間には DVI Dual Link ケーブルによりリング接続する。

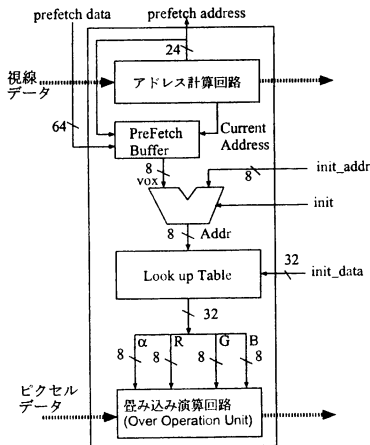


図5 ピクセル値計算ユニットの構成

3.2.2 ボリュームメモリインタフェースの実装

ボリュームメモリのバンド幅を有効活用するには、使用するメモリモジュールが対応可能なバースト長でのブロック転送を行なう必要がある。今回はバースト長を 4 とし、32 バイト単位のブロック転送とした。この際、プリフェッチバッファに無駄なボリュームデータが格納されるのを極力避けるため、一回のバースト転送で、 $4 \times 4 \times 2$ の領域に対応するデータが読み出されるような設定とした。このとき、一度のバースト転送で読みだされたデータは、隣接する 2 つの PCU のプリフェッチバッファ (PFB) に各々 4×4 づつ格納することとする。

このように、ボリュームメモリは $4 \times 4 \times 2$ の 3 次元領域が連続アドレスとなるようなアドレス付けを行なう。そのためには、格納するボリュームデータのサイズに応じたアドレス変換が必要であり、ボリュームメモリインタフェース内でこの変換を行なう。

3.2.3 LUT の実装

LUT は、サンプリングポイントのボクセル値 (vox) に対応する R,G,B,α 値を、畳み込み演算回路に対して毎サイクル供給しなければならない。VisA Pro 実装では FPGA 内に分散配置されている高速 Block RAM を用いて実装する。LUT は一旦初期化を行なうとレンダリング処理中は Read Only となるため、比較的容易にマルチポート構成が実現可能である。しかしながら、FPGA 内の長距離配線リソースの消費を減らすため今回の実装ではシングルポート構成とし、各 PCU 毎に独立に LUT を設けた。これにより高速動作を保証する。初期化やテーブル変更時には、動作速度を下げ全 LUT を一斉更新する。

3.2.4 PFBの実装

PFBには、ボリュームメモリからのプリフェッチデータのブロック書込みと、LUT側からの毎サイクル読み出しが必要のため、デュアルポート構成のバッファとする必要がある。ブロック書込み時には、 4×4 の領域に対する16バイトのデータを、LUT側の動作周波数の2倍の周波数で64bit単位で書き込む。現在の実装では、プリフェッチ効率が100%の場合は、書込みの周期は16サイクルに一回である。これに対して、LUT側は毎サイクル8bit単位の読み出しが連続して行なわれる。

ボリュームメモリへのアクセスパターンは、わずかな空間局所性が存在するのみである。したがって、PFBは比較的小容量の連想メモリとして構成する。Readアクセス時にPFB内に必要とするデータが存在しなかった場合、パイプラインをストールし、オンデマンドフェッチを行なう。

3.2.5 畳み込み演算回路

式(2),(3)の畳み込み演算を行う回路である。畳み込み演算の方法については、1つのピクセルに関するボクセルデータを同時に読み込みツリー構造で足していく方法や、1つずつボクセルデータを読み込み順次パイプライン式で足していく方法などがある。

3.2.6 Mapper回路

ここでは、2.2節(6)で述べたMapper機能の他に、PCIバス側からボリュームメモリ内のボリュームデータにボクセル座標(X,Y,Z)でアクセスするためのアドレス変換機能も備える。

3.2.7 視線生成回路

各VisA Proカード内で、独自に視線情報を生成する必要があるレンダリングモードの場合に、視点位置、スクリーン位置、ピクセル座標から、視線ベクトルを計算する回路である。

4. 簡易版ボリュームレンダリング回路の実装

3章で述べた並列ボリュームレンダリング回路の実装に先立ち、VisA Proカードの動作検証も兼ねて、単体のVisA Proカードのみで動作可能な簡易版ボリュームレンダリング回路の実装を行なった。

実装に際しては、PCIバス経由でのボリュームデータやLUT内の伝達関数(ボクセル値を色と透明度に変換する関数)の変更やレンダリング結果のディスプレイへの直接表示を可能とする等、データの更新、レンダリング、表示の一連の連携動作の確認を行なえることを第一の目的とした。

そのため、レンダリング処理自体には以下のような制約を与えている。

(1) ボリュームデータサイズは $32 \times 32 \times 32$ (32KB)とする。これは、VisA Proボード上のSSRAMに格納可能なサイズであり、DDR-SDRAMを利用することによる回路の複雑化を避けるためである。

(2) ボリュームデータの奥行き方向(z軸とする)に並行なz軸並行投影のみを扱い、原則として視点移動は行なわない。これにより視線情報の計算が単純な加算処理となり(視線生成回路がほぼ不要)、メモリアccessパターンも連続となる。

以下、回路の概要について説明する。ブロック図を図6に

示す。この回路では、まずボリュームデータと伝達関数の設定はPCIインタフェース(Xilinx社のIPであるLogiCORE PCI32/33を使用)を介して動作周波数33MHzで行なう。次にレンダリングを開始すると166MHzで動作可能なSSRAMから毎クロック4バイト(ボクセル値4つ分)のデータが連続して出力される。このデータはVolume Memory Interfaceを介して83MHz毎クロック8バイトに変換されピクセル値計算ユニットに供給される。ピクセル値計算ユニットでの計算結果は、DVIインタフェース部のフレームバッファに格納される。ピクセル値計算およびフレームバッファへの格納は動作周波数83MHzで行なう。フレームバッファに格納されたレンダリング結果は、DVIケーブル(PCの汎用グラフィクスカード)から供給されるDVIクロック(108MHz)と水平/垂直同期信号を使って、スクリーン上の適切な位置に表示される。

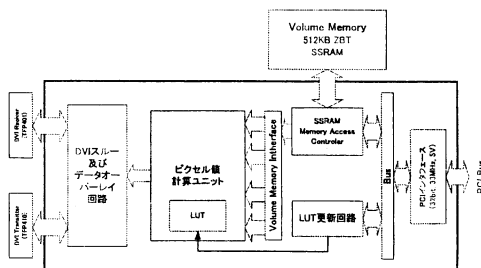


図6 簡易レンダリング回路のブロック図

以下、各部の詳細を説明する。

4.1 レンダリング処理部

ボリュームメモリから読み出される8つのボクセル値は、FPGA内のBLOCK RAMで実装した8個のLUTの並列検索のためのアドレスとして使用され、各々 (r, g, b, t) に変換される。この際、同時に読み出されるデータはボクセル空間上の x, y 座標が等しく、 z 座標が1つつ異なる連続する8座標点のボクセル値に対応するようSSRAMのアドレッシングを行なっている(図7)。次のボクセル座標はマイクロクロック毎にアドレスを8インクリメントすることで得られる。これにより、1ピクセルあたり4クロックで当該ピクセルに寄与するボクセル値を読み出すことが可能となる。

読み出されたボクセル値は連続する8個を単位として畳み込み演算回路のトリーに供給される。第一段目の畳み込み演算回路で隣接するボクセル間の寄与が計算され、その結果が次の段の入力となり、3段目の畳み込み演算回路の出力が連続8個分の寄与値となる。この段の出力は毎サイクル、隣接する連続8個分の寄与値を出力するので、最終段の畳み込み演算回路でこの値を4回累積することでz方向32ボクセル分の寄与値(=ピクセル値)が求まり、その結果はフレームバッファに格納される(図8および図9参照)。

4.2 SSRAM入出力部

レンダリング処理部とPCIインタフェース部の双方からアクセスがあるため、2系統のクロックの差を吸収し、アクセスの調

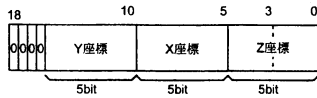


図7 ボリュームメモリ (SSRAM) のアドレス構成

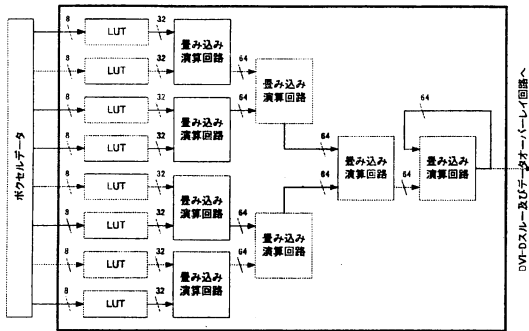


図8 ピクセル値計算ユニットの構成

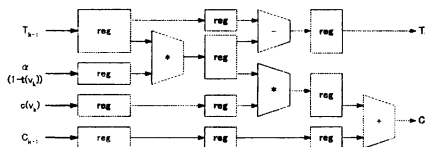


図9 ピクセル値計算パイプライン1段分の畳み込み演算回路

停を行なうための回路をSSRAMとの間に置いた。SSRAM自体は166MHzで動作するが、PCIインタフェース側は33MHzで動くバスとした。

前述の通り、SSRAM制御回路とピクセル値計算ユニットとの間には、ボリュームメモリインタフェース回路をもうけ、SSRAMからの166MHz動作4バイト幅のデータバスを83MHz動作8バイト幅のデータバスに変換している。これは、166MHz動作のSDR-SSRAMを見かけ上83MHz動作のDDR-SSRAMと見做すことで、将来のDDR-SDRAMとのインタフェースとの親和性を上げるためである。

4.3 DVI-D インターフェース部

DVI-D インターフェース部自体はSingleLink DVI ケーブルを使った場合5.2GB/sの速度が出ることが確認できている[12]。しかし、今回は1枚での実装であるためPCのグラフィックカードから出力された情報をDVI-Dインから入力しそのままDVI-Dアウトから液晶モニタに出力する回路(PCIカードに付属するサンプルコード)を転用し、入力データの一部にレンダリング処理の結果を貼り付ける構成とした。

5. まとめ

今回の予備実装により、一連のボリュームレンダリング処理をVisA Proカード単体で行なう回路が完成した。今回のレンダリング回路は制約が多く、現時点では視点移動にも対応できていないが、ボリュームメモリのアドレッシング法とレンダリングパイプラインの構成法の関係や、それをFPGA上に実装す

る場合のレイアウトとの親和性の解析に利用する予定である。

また、今後は3章で紹介したReVolter/C40ベースの並列レンダリング回路の実装、ならびに、ある一定の条件の下でボリュームデータの3重化を必要としないレンダリングアルゴリズム[8]の応用について検討を行なっていく予定である。

謝 辞

日頃より御討論いただく京都大学大学院情報学研究所富田研究室の諸氏に感謝します。本研究の一部は、日本学術振興会科学研究費補助金 基盤研究S(課題番号16100001)、21世紀COEプログラム(課題番号14213201)、ならびに、文部科学省特定領域研究S(課題番号13224050)による。

文 献

- [1] 生雲 公啓, “時変ボリュームデータの実時間可視化のための専用グラフィックスカード VisA の開発”, 京都大学大学院情報学研究所 修士論文, 2003.
- [2] 對馬 他, “ボリューム・レンダリング専用並列計算機 ReVolter のアーキテクチャ”, 情報処理学会論文誌, 第36巻, 第7号, pp.1709-1718, 1995.
- [3] 原源 他, “ReVolter/C40 を用いた時系列ボリュームデータの実時間可視化”, 情処研報 2002-ARC-142, pp.7-12, 2002.
- [4] S. Mori, et. al, “ReVolter/C40 : A Scalable Parallel Computer for Volume Rendering - Design and Implementation -”, IEICE Trans. on Inf. and Sys., Vol.E86-D, No.10, pp.2006-2015, 2003.
- [5] 山内 他, “アクティブボリュームレンダリングに基づくシミュレーションステアリング”, 信学技報 CPSY2001-35, pp.1-8, 2001年8月.
- [6] Marcelo Magallon, et al., “Parallel Volume Rendering Using PC Graphics Hardware,” *Proc. of Pacific Graphics*, 2001.
- [7] 丸山悠樹 他, “汎用グラフィックスハードウェアを用いた並列ボリュームレンダリングの実装”, 情処研報, 2003-ARC-154, pp.61-66, 2003.
- [8] 金喜都 他, “ピクセル並列処理によるボリューム・レンダリング向きの超高速専用計算機アーキテクチャ”, 情報処理学会論文誌, 第38巻, 第9号, pp.1668-1680, 1997.
- [9] 藤原雅宏 他, “階層格子ボリュームデータの実時間可視化”, 情処研報, 98-ARC-128, pp.7-12, 1998.
- [10] 山内 他, “透視投影ボリュームレンダリングにおけるサンプリング方式の評価”, 第29回画像電子学会年次大会予稿集, pp.33-34, 2001年6月.
- [11] Kevin Kreeger, et. al, “Adaptive Perspective Ray Casting,” *Proc. of 1998 Symp. on Volume Visualization*, pp.55-62, Oct. 1998.
- [12] 森 他, 960MB/s の DVI-D 入出力リンクと DDR-SDRAM を 2 系統を持つ FPGA 搭載 PCI カード- 並列可視化処理への応用 -, 11 回 FPGA/PLD Design Conference 論文集, pp.31-34, Jan. 2004.