

# ストリームアプリケーションを対象としたスケーラブルFPGA システムの提案

久原 拓也

## 1 はじめに

近年、科学計算に要するデータ量や演算量が膨大となり、大規模処理にはスーパーコンピュータや PC クラスタなどの高性能計算システムが用いられるようになった。これらのシステムにより、大規模処理を現実的な時間で実行可能になったが、その一方で、システムの導入に要する設備コスト、電力や空調、メンテナンスなどの運用コストの増大が問題となってきた。

そのような状況の中で、定型的な処理を専用ハードウェアで行うことにより、高い演算性能を維持したまま、システムのサイズや電力消費量の低減を可能にする手法が研究されている。この手法に適した処理として、動画処理やストリーム暗号処理のような、入力データを順に計算するストリームアプリケーション処理がある。ストリームアプリケーション処理に特化したハードウェアは、代表的なものでは GPU や Cell があるが、その他に FPGA(Field Programmable Gate Array) による専用ハードウェア構築も多く研究されている。FPGA は、柔軟に専用ハードウェアを再構成できるデバイスとして注目を集めており、実際に Splash2 や CUBE などの優れた性能を持つシステムの構築例もある<sup>1)</sup>。FPGA は複数接続によるシステム拡大が可能であり、より優れた専用ハードウェアの構築が期待できるが、増設する FPGA のコストや、扱いが困難である点などいくつかの問題も持ちあわせている<sup>2)</sup>。そこで、本研究では、ストリームアプリケーションを対象とした、安価で扱いやすいスケーラブルな FPGA システムの提案、構築を行う。

## 2 スケーラブル FPGA システムの提案

図 1 に提案するシステムの構成図を示す。

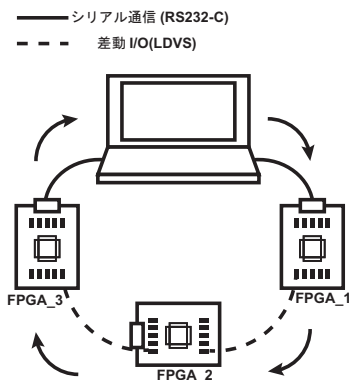


Fig.1 スケーラブル FPGA システムの構成

図 1 に示すように、PC からデータを受信する FPGA(図 1 の FPGA\_1)、FPGA 同士でデータを中継

する FPGA(図 1 の FPGA\_2)、PC にデータを送信する FPGA(図 1 の FPGA\_3) の 3 種類の FPGA がシステムに接続されており、PC からの入力信号が、複数の FPGA を介して処理され、PC に出力される構成を取る。FPGA は直列接続し、シストリックアレイでデータを処理する。シストリックアレイは制御信号でコアを同期させると共に、近接コア同士でデータを授受して、データをパイプライン処理する処理方式であり、ストリームアプリケーションに対する高速な処理が期待できる。

3 種類の FPGA に対し、ハードウェアを Verilog-HDL を用いて設計する。本システムでは、全てのハードウェアを、送信モジュール、受信モジュール、処理モジュールの 3 つに分離して設計する。図 2 にモジュール設計図を示す。

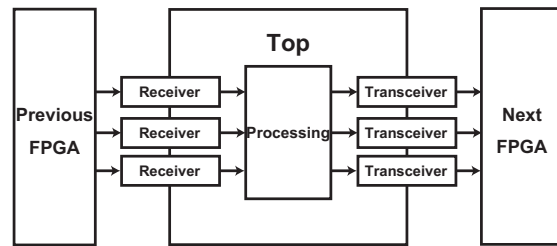


Fig.2 モジュール設計

図 2 のようにシステムをモジュラーに構成することで、処理モジュールを置き換えて、様々なアプリケーションをシステム上に実現可能になる。

## 3 スケーラブル FPGA システムの実装

提案したシステムについて、実装を行った。システムに組み込む FPGA ボードとして、50MHz 駆動の Xilinx Spartan-3AN を 3 台用い、PC と FPGA の接続にはシリアル通信 (RS232-C) を、FPGA 間の接続には差動 I/O(LDVS) を用いた。通信速度は、シリアル通信、差動 I/O 共に 9600bps となるように設計を行った。

また、動作確認の為に、本システムにアプリケーションとして 32bit 版 Mersenne Twister を実装した<sup>3)</sup>。32bit 版 Mersenne Twister とは、1 クロックサイクルにつき 1 つの 32bit 乱数を生成するアプリケーションである。今回の動作確認では、Mersenne Twister を、FPGA\_1 にのみ実装した。FPGA\_1 は常に乱数を生成し続け、PC から ASCII 文字を受信したときに、1 つの乱数を残り 2 枚の FPGA を通して PC に送信するというような設計を行った。

## 4 評価

以下の3つの構成で、乱数を1000個生成し、PCで取得するまでの時間を測定する実験を行った。図3に実験構成を示す。

- (1) FPGA1枚をPCとシリアル接続したもの。
- (2) FPGA2枚をPCとシリアル接続し、2枚を差動I/Oで接続したもの。
- (3) FPGA2枚をPCとシリアル接続し、2枚の間にFPGAを1枚差動I/Oで接続したもの。

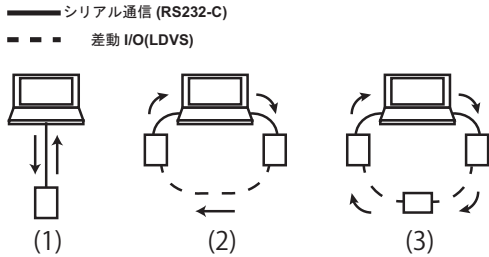


Fig.3 実験構成

各構成において10回ずつ実験を行い、時間の平均、受信した有用なデータ量 (net), 受信した総データ量 (gross) を算出した。結果を表1に示す。

Table1 通信性能 (理論性能 9600bps)

構成	時間 [s]	net[bps]	gross[bps](理論値との比率)
(1)	5.85	6830.35	8537.81(88%)
(2)	7.17	5576.07	8364.11(87%)
(3)	8.22	4861.50	8507.63(88%)

本システムのMersenne Twisterでは32bitの乱数データを生成するが、これをシリアル通信でPCに送信するために、32bitの乱数データを8bitずつに区切り、4回に分けて送信している。そのため、スタートビット、ストップビットを含めた1回の乱数送信辺りの送信データ量は40bitとなる。以上より、構成(1)(2)(3)でPCが受信した有用なデータ量を、表1のnet項のように求められる。

このデータ量は、理論的な通信量である9600bpsと比較すると、(1)では2800bps程度、(2)では4000bps程度、(3)では4800bps程度少ない値となっている。今回、乱数送信のトリガであるASCII文字の送信を、PCが一つの乱数を受信してから行うようプログラム側で設定した。これにより、FPGAがPCに乱数データを送信しない時間帯が発生し、データに有用でないデータが混じったと考えられる。図4に、PCからASCII文字が送信され、FPGA.1がFPGA.2に、FPGA.2がFPGA.3に、FPGA.3がPCにデータを送信するまでのタイムチャートを示す。

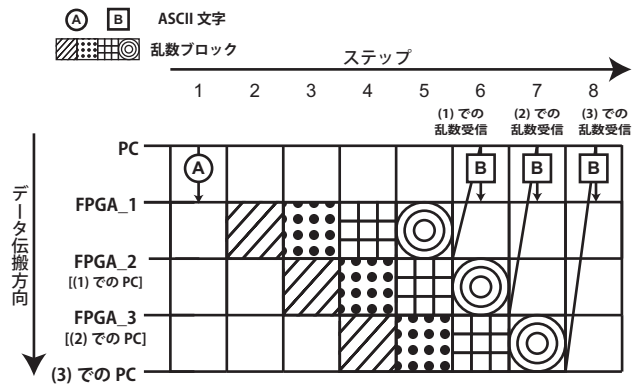


Fig.4 データ送信のタイムチャート

図4に示したように、(1)では送信サイクル5ステップ中1ステップで、(2)では送信サイクル6ステップ中2ステップで、(3)では送信サイクル7ステップ中3ステップでFPGAがPCに乱数データを送信できない時間帯が発生している。これらを含めると、構成(1)(2)(3)でPCが受信した実際のデータ総量は、表1のgross項のように求められる。

net項の値に比べ、gross項の値と理論値との差が小さいものになっていることがわかる。gross項の値と理論値との1400bps程度の差は、PCがFPGAから乱数を受け取ってから次のASCII文字を送る際の遅延や、乱数生成の時間が積み重なって生まれたものと考えられる。

以上の結果から、本システムは、通信処理がボトルネックとなつてはいるが、アプリケーション処理、信号の送受信が正常に動作していることが確認できた。今回の実験ではFPGA1枚にアプリケーション処理を実装したが、複数枚のFPGAに処理を分散することで、高速なアプリケーション処理が可能となる。また、本システムはFPGAを増設することが可能であり、より高速なシステムを構築も可能である。今回用いたプログラムでは、FPGAの増設に伴いスループットが低下していくが、プログラムを、乱数データ受信途中で次のASCII文字を送信できるように変更することで、高スループットな通信が可能になると考えられる。

## 5 ストリームアプリケーション

### 5.1 アプリケーションの選定

Mersenne Twisterを用いた実験では、スケーラブルFPGAシステム上でのアプリケーションの動作、及びPCとの相互通信を確認できたが、システム上で、アプリケーションが実装されているFPGAは1台のみであり、他2台は通信しか行っていない。1章で述べたように、各FPGA間でデータをパイプライン処理し、順に受け渡すストリームアプリケーションを本システムに実装することで、アプリケーションの高速な処理が期待できる。そこで、システムに実装するためのアプリケーションの選定を行った。今回は、結果が分かりやすく、視覚的である点を考慮し、ゲーム分析を選定した。

## 5.2 ゲーム分析

ゲーム分析は、人工知能研究の目的とする複雑な条件下での意思決定法のモデルケースとして、有力な手法である。分析対象となるゲームの中でも、二人零和確定完全情報ゲームと呼ばれるオセロやチェス、将棋、囲碁などのゲームは、ゲーム分析の結果が勝敗の形で得られる為、人工知能アルゴリズムの客観的な評価が行い易い。ここで、二人零和確定完全情報ゲームはゲームの分類であり、【二人】は人数を、【零和】はプレイヤーの利得合計が0になることを、【確定】は偶然の要素が入り込まないことを、【完全情報】は、プレイヤーが各プレイヤーの状況を全て把握できることを指す<sup>4)</sup>。

二人零和確定完全情報ゲームの分析に用いられるアルゴリズムとして、主に MiniMax 法とモンテカルロ法が挙げられる。MiniMax 法では、まず、現在局面から数回の着手を行って数手先の局面を列挙した後に、列挙した局面の中から有利な局面を選択し、着手を選択するアルゴリズムである。この時、自プレイヤーは常に自プレイヤーにとって利得が最大の手を、相手プレイヤーは常に自プレイヤーにとって利得が最小の手を打つ、という考え方に基づいて、着手を選択する。利得評価については、盤面の状況や駒の配置から評価関数を設計して求める。モンテカルロ法では、まず、現在局面からランダムな着手を行い、大量の終局状況を作り、その勝敗結果を集計し、勝数の多い手を強い手とするアルゴリズムである。モンテカルロ法は、MiniMax 法のようなゲーム特有の評価関数を設計する必要がなく、どのようなゲームにも用いることができるという利点があるが、一般的に MiniMax 法よりも多くの試行回数が必要である。囲碁のようなルールの複雑なゲームでは、モンテカルロ法を用いたアルゴリズムが有力であるとされている。

MiniMax 法、モンテカルロ法ともにある局面を元にした未来の局面を予想する処理が必要であるが、この処理における局面の進行はお互いに影響しないため、並列での処理が可能である。特にモンテカルロ法では、評価関数を用いないため、並列処理の性能が勝敗を大きく左右する。そのため、モンテカルロ法による囲碁分析の GPU 実装や、FPGA 実装が研究されている<sup>4) 5)</sup>。また、MiniMax 法においては、決まった順番でデータを処理するというアルゴリズムから、ストリームアプリケーションとして FPGA アレイの性能検討を行うのに適していることが期待できる。以上の点から、本システムに実装するアプリケーションとして、ゲーム分析アルゴリズムは適しているといえる。

## 5.3 Connect6

以上の点から、分析するゲームは二人零和確定完全情報ゲームであり、かつ、モンテカルロ法を採用した際に MiniMax 法に対して優位な効果が出るように、ある程度複雑であるゲームが望ましい。Connect6 は、五目並べを拡張した二人零和確定完全情報ゲームである。二人のプレイヤーが、交互に盤上に2個ずつ石を置き、先に自分の

石を6個並べたプレイヤーが勝利する。五目並べは、単純さと必勝形の多さから、先手必勝であることが知られているが、Connect6 は交互に置く石を2個ずつとすることで、複雑になり、先手と後手の公平さを保たれている。現在は、コンピュータ Connect6 を実装し、Connect6 を分析するモンテカルロ法アルゴリズムを作成している。

## 6 まとめと今後の展望

本研究では、ストリームアプリケーション処理を対象とした、安価で扱いやすくスケーラブルな FPGA システムを提案した。また、実際に3台の FPGA を用いてシステムを構築し、Mersenne Twister を実装して動作確認実験を行った。また、システム上で実行するストリームアプリケーションとして Connect6 のゲーム分析アプリケーションを採用した。

今回、構築したシステムでは、通信処理が大きなボトルネックとなっているが、シリアル通信でより高速なデータレートを用いたり、差動通信で用いるピン数を増やし、パラレルで送受信を行うことで、通信速度を高速化できると考えられる。今後の展望としては、システムの通信処理改善や、Connect6 アルゴリズムの完成およびストリームアプリケーション化を行い、システム上での評価実験を行いたい。また、システムをより扱いやすいものにするため、最小構成の FPGA ボードの作成などを行いたい。

## 参考文献

- 1) 吉見 真聡 and 西川 由理 and 天野 英晴 and 三木 光範 and 廣安 知之 and オスカーメンサー. ストリームアプリケーション向け大規模 FPGA アレイ CUBE の性能評価. 情報処理学会論文誌. コンピューティングシステム, Vol. 3, No. 3, pp. 209–220, 2010-09-17.
- 2) 高前田伸也, 渡邊伸平, 姜軒, 藤枝直輝, 植原昂, 三好健文, 吉瀬謙二. メニーコアアーキテクチャ研究のためのスケーラブルな HW 評価環境 ScalableCore システム. 情報処理学会研究報告. 計算機アーキテクチャ研究会報告, Vol. 2009, No. 3, pp. 1–10, 2009-10-19.
- 3) 渡部信吾, 阿部公輝. 疑似乱数生成器 Mersenne Twister の VLSI 設計. 情報処理学会研究報告. CSEC, [コンピュータセキュリティ], Vol. 2005, No. 41, pp. 13–18, 2005-05-19.
- 4) 田野文彦, 近山隆. 4t-4 グラフィックエンジンを用いたゲーム探索の高速化 (ゲーム, 学生セッション, 人工知能と認知科学, 情報処理学会創立 50 周年記念). 全国大会講演論文集, Vol. 72, No. 2, pp. 2–163–2–164, 2010-03-08.
- 5) 小泉賢一, 石井康雄, 美添一樹, 三好健文, 菅原豊, 稲葉真理, 平木敬. Fpga 基板を用いたモンテカルロ碁の高速化 (アクセラレーションと回路設計, 2009 年並列/分散/協調処理に関する『仙台』サマー・ワークショップ (swopp 仙台 2009)). 電子情報通信学会技術研究報告. CPSY, コンピュータシステム, Vol. 109, No. 168, pp. 55–60, 2009-07-28.