

手軽な外付け FPGA アクセラレータによるソーティングの高速化

臼井 琢真[†] 吉瀬 謙二[‡]

[†]東京工業大学 工学部情報工学科 [‡]東京工業大学 大学院情報理工学研究科

1 はじめに

ソーティングは、多くのアプリケーションにとって重要な計算カーネルであり、常に高速化が求められている。その手段として、FPGA を用いたアクセラレータの使用が挙げられる [1][2]。FPGA を用いたアクセラレータでは、アプリケーションに特化した演算パイプラインとデータ供給機構を実現する回路を FPGA 上に実装することによって、CPU や GPU と比較して高い演算性能を達成することができる。

FPGA アクセラレータを設計するときには、CPU と FPGA アクセラレータ間におけるデータの送受信のためのインターフェースを考慮しなければならない。これには様々なものが存在するが、中でも、USB3.0 は以下の 4 つの利点を持つ。

- 理論値 5Gbps の高い転送速度
- 外付けインターフェースであり、さらにホットスワップに対応しているため接続が容易
- 最大 900mA のバスパワーにより、省電力な機器であれば USB ケーブルを 1 本繋ぐだけで動作する
- モダンな計算機の大半が USB3.0 接続をサポート

これらの利点は CPU と FPGA アクセラレータ間におけるデータの送受信のためのインターフェースとして非常に好ましいので、我々は USB3.0 接続をサポートするポータブルな FPGA アクセラレータ (図 1) を提案する。

2 設計と実装

2.1 ソーティングロジック

図 2 に、FPGA 上に実現されるソーティングロジックのデータパスを示す。図 2 中の破線で囲まれた部分が Merge Sorter Tree [2] であり、このロジックでソーティングが実行される (図では Merge Sorter Tree の入力が 4 つだが、実際には 8 つの入力を持つ Merge Sorter Tree を実装した)。Merge Sorter Tree 中の四角は FIFO を表し、丸はソートセルを表している。ソートセルは 2 つの入力データを比較し、それらの入力データの比較結果にしたがってどちらかのデータを選択して出力する組み合わせ回路である。Interface Module はホスト PC から送信されるデータ列の受信、FPGA 内でソートされたデータ列の送信を担当する。ホスト PC から送信されるデータ列を受信する際、Interface Module は入力データに対して 2 要素ずつソートし、Input Buffer に格納する。Input Buffer は入力幅が 64bit (2 要素)、出力幅が 32bit (1 要素) の非対称 FIFO である。Merge Sorter Tree でソートされたデータは Output Buffer で一時的にバッファリングされた後、DRAM にライトされる。Output Buffer は入力幅が 32bit (1 要素)、出力幅が 64bit (2 要素) の非対称 FIFO である。

このアクセラレータは図 3 のような手順でソーティングを行う。まず、Process A にてデータ列の受信と Interface

High-speed Sorting using Portable FPGA Accelerator
Takuma USUI[†], and Kenji KISE[‡]

[†]Department of Computer Science, Tokyo Institute of Technology [‡]Graduate School of Information Science and Engineering, Tokyo Institute of Technology



図 1: ノート PC に接続された FPGA アクセラレータ

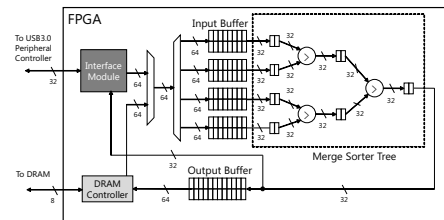


図 2: FPGA に実装するソート回路のデータパス

Module の処理を行う。そして Process B-E を適切な回数だけ繰り返し、完全にソートされたデータ列を得る。最後に、Process F にてそのデータ列を Interface Module を通してホスト PC に返却する。実装では、FPGA アクセラレータの性能向上のため、この Process B-E の処理をオーバーラップさせている。

2.2 実装

FPGA アクセラレータを実装するために、我々は TKDN_ART7_BRD-2 [3] を導入した。これは Xilinx Artix-7 XC7A100T、256MB DDR3 SDRAM を搭載し、USB3.0 Peripheral Controller に Cypress 社製の EZ-USB FX3 を採用した FPGA ボードである。

我々は、図 2 に示したソーティングロジックを Verilog HDL を用いて実装した。Interface Module の実装には、[3] が提供する IP コアを利用し、DRAM Controller の実装では Xilinx 社が提供する IP コア [4] を利用した。FPGA 内の全てのモジュールは 100MHz にて動作し、DRAM は 400MHz で動作する。なお、ハードウェア使用率は Block RAM が 20%、スライスが 18% となり、FPGA のハードウェアリソースを殆ど使用せずに実装することができた。

3 評価と考察

3.1 USB3.0 による効率的なデータ通信

図 4 にホスト PC と FPGA ボード間におけるデータの通信速度 [MB/s] を計測した。この通信速度 [MB/s] は、それぞれ異なるデータサイズについて、送信と受信を適当な回数繰り返したときの時間を計測し、それらの算術平均で求めた値である。図 4 に示すように、データサイズが小さいときは通信のオーバーヘッドのため通信速度が非常に低く、データサイズが大きいほど通信速度は向上する。今回 FPGA に送るデータのサイズは 128MB と

表 1: 評価環境

Host PC	Computer A	Computer B	Computer C	Computer D
Type	Tower	Laptop	Slim Desktop	Laptop
CPU	Intel Core i7 3770K	Intel Core i3 4010U	Intel Core i7 870	Intel Core Duo T2400
CPU Frequency	3.5 GHz	1.70 GHz	2.93 GHz	1.83 GHz
Memory Capacity	16 GB	4.0 GB	4.0 GB	1.0 GB
USB Ports	USB3.0 × 4, USB2.0 × 6	USB3.0 × 2	USB2.0 × 8	USB2.0 × 2

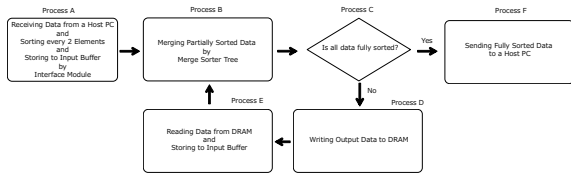


図 3: ソーティングの手順

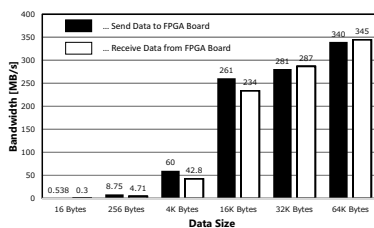


図 4: USB3.0 の通信速度

非常に大きいため、ホスト PC と FPGA アクセラレータ間におけるデータ通信は最大値に近い通信速度で実行することができる。

3.2 評価環境と性能評価

提案する FPGA アクセラレータの高いポータビリティを示すため、評価環境として FPGA アクセラレータに接続するホスト PC を表 1 のように 4 台用意した。[3] が提供する、今回用いた FPGA ボードと USB3.0 を介してデータ通信を行うためのライブラリが Windows 向けであるため、これらの全ての PC において OS に Windows を用いて評価した。

Computer A と B を USB3.0, Computer C と D を USB2.0 にて FPGA アクセラレータと接続し、FPGA アクセラレータでソーティングを実行させた場合の実行時間とホスト PC でマージソートをシングルスレッドで実行させた場合の実行時間を比較した。なお、評価に用いたプログラムは C 言語で記述し Microsoft Visual C++ Compiler 2013 にて最適化オプションを /Ox に設定してコンパイルしたものである。

前述した環境において 32M 個の int 型要素 (1 要素 : 4Bytes) のソーティングに要した時間を計測したところ、図 5 のようになった。Computer C と D においてはホスト PC のみでソーティングを実行したときより実行速度が低下したが、Computer A と B においてはホスト PC のみで実行したときより実行速度がそれぞれ 1.26 倍、2.60 倍向上した。

3.3 議論

この節では、Computer A と B において、FPGA アクセラレータにソーティングをオフロードしたときの実行速度が向上した要因について考察する。図 5 を見ると、どの PC においても FPGA アクセラレータにソーティングの実行時間は同一だが、Computer A と B においては送信時間と受信時間が大幅に削減されていることが分かる。これは、FPGA アクセラレータが USB3.0 の通信

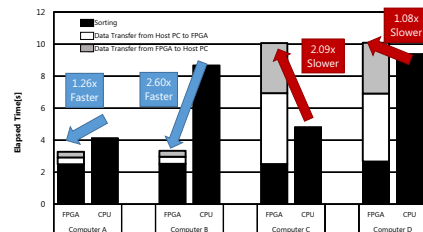


図 5: ソーティングの実行時間の比較

速度の恩恵を受けていることを示している。

FPGA アクセラレータに USB2.0 を介して接続した Computer C と D では、ソーティングを FPGA アクセラレータにオフロードしても速度向上が得られず、特に Computer C では 2.09 倍遅くなった。比較的モダンな CPU を搭載する Computer C においては、ホスト PC-FPGA 間の通信時間がホスト PC 上でのソーティング時間を上回っている。このため、FPGA アクセラレータに USB2.0 を介して接続した場合は、FPGA アクセラレータに処理をオフロードしても、逆にソーティングの実行速度が低下してしまう。

対して、USB3.0 を介して接続した Computer A と B においては、通信時間の大幅な削減により、実行速度がそれぞれ 1.26 倍、2.60 倍となった。このことから、USB3.0 接続をサポートする PC ならば、提案する FPGA アクセラレータによってソーティングの高速化を達成できる。

4 まとめ

本稿では、USB3.0 接続の手軽な外付け FPGA ソーティングアクセラレータを提案し、その設計と実装を行った。その結果、USB3.0 をサポートする PC ならばソーティングを最大 2.60 倍高速化できた。

今後の課題として、Merge Sorter Tree の入力を増やした場合の性能評価や、提案する FPGA アクセラレータのポータビリティを高めるために [3] が提供する USB3.0 通信のためのライブラリを他の OS に移植すること、他の計算カーネルの高速化が挙げられる。

参考文献

- [1] Rene Mueller, Jens Teubner, and Gustavo Alonso. Sorting networks on fpgas. *The VLDB Journal*, Vol. 21, No. 1, pp. 1–23, February 2012.
- [2] Dirk Koch and Jim Torresen. Fpgasort: A high performance sorting architecture exploiting run-time reconfiguration on fpgas for large problem sorting. In *Proceedings of the 19th ACM/SIGDA International Symposium on Field Programmable Gate Arrays*, FPGA '11, pp. 45–54, New York, NY, USA, 2011. ACM.
- [3] Tokushudenshikairo Inc. <http://www.tokudenkairo.co.jp/art7/index.html>.
- [4] Memory Interface Generator (MIG). <http://www.xilinx.com/products/intellectual-property/MIG.htm>.