

ハードウェアソーティング アルゴリズムのFPGA実装

広島大学 松本直之

FPGAとは

- FPGA : Field Programmable Gate Array
- ・ユーザが任意に回路を書き換えることのできるLSI

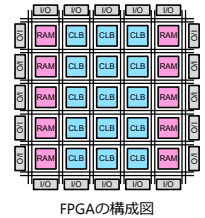
CLB (Configurable Logic Block)

ユーザが書き換え可能な論理ブロック
小規模な組み合わせ回路や順序回路を構成

ブロックRAM

CLBのメモリ能力を補う専用回路

- これらの素子を接続することで
任意の回路を実現



FPGAの構成図

研究概要

- ・2種類のソーティングアルゴリズムをFPGAに実装

ロバートニックソート

- ・ソーティングネットワークに基づいた回路
- ・任意のデータ幅のデータをソート可能

ロマージソート

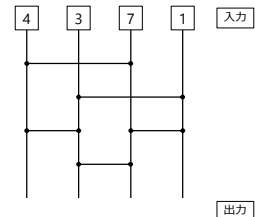
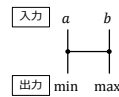
- ・FIFOを用いた回路
- ・最大で524288要素のデータをソート可能

ソーティングネットワークとは

- ・入力データをソートして出力するネットワーク
- ・複数のコンパレータから構成

コンパレータ

2つの値を入力し、小さい値を一方に、
大きい値を他方に出力（比較交換）

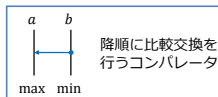
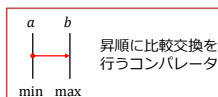
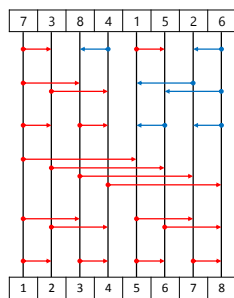


ソーティングネットワークの例

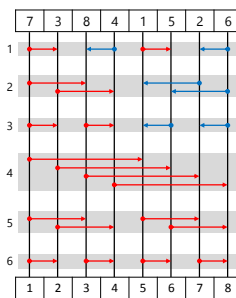
バイトニックソートのソーティングネットワーク

- ・要素数8の場合

K.E.Batcher. Sorting networks and their applications. AFIPS Spring Joint Computer Conference, pp307-314, 1968



ソーティングネットワークの段数



要素数 n のバイトニックソートの
ソーティングネットワークの段数は
 $\frac{\log n (\log n + 1)}{2} = O(\log^2 n)$

要素数8の場合、段数は6

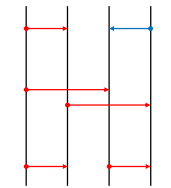
バイトニックソート回路

データ列2 01100 | 11010 | 10101 | 00011
 データ列1 111 | 011 | 001 | 100

・ビットシリアルを用いた回路を設計

ビットシリアル

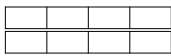
データを上位ビットから1ビットずつ入力



最上位ビットから順にソーティングネットワークに入力する

ソートされたデータが最上位ビットから順に出力される

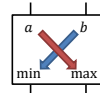
➢ 1ビットずつ入力するため、任意のデータ幅のソートを行うことができる



ビットシリアルを用いたコンパレータの動作

・最上位ビットから順に入力し、比較を行う
 ➢異なるビットが入力されると、その時点で大小が確定する

101010 100111



入力された2つのビットを比較

同じビットであれば、交換せず出力

異なるビットであれば、1をmaxに、0をminに出力
この時点で2つのデータの大小が確定

以降のビットは比較せず、決められた通りに出力

ビットシリアルを用いたコンパレータの動作

・コンパレータの動作を3つに分ける
 ➢ステートマシンを用いて制御
 ✓初期状態は $a = b$
 ✓異なるビットが入力されると状態を遷移

各状態でのコンパレータの動作

状態 $a = b$

・比較交換を行う



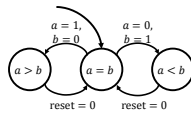
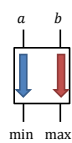
状態 $a > b$

・交換を行う



状態 $a < b$

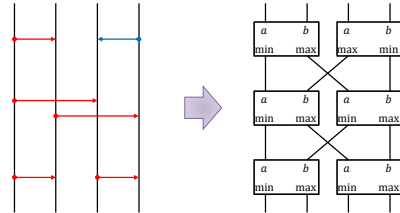
・交換を行わない



コンパレータのステートマシン

バイトニックソート回路の構成

・ビットシリアルを用いたコンパレータでソーティングネットワークを構成



実装結果

・ターゲットFPGA : Xilinx Virtex-7 XC7VX485T

・ソート可能な最大要素数 : **1024**
 ・動作周波数 : **1030.822**[MHz]
 ・処理時間(32ビット) : **84.40** [ns]

| リソース | 使用数 |
|-----------------|--------------|
| Slice Registers | 140863 (23%) |
| Slice LUTs | 141570 (46%) |

CPUとの比較

・CPU : Intel Xeon X7460 2.66GHz
 ・C言語の標準関数qsort()と比較
 ・要素数 : 1024
 ・データ幅 : 32ビット

| 実行回数 | 処理時間[μs] | | 高速化率 |
|-------|----------|---------|----------|
| | CPU | FPGA | |
| 1 | 97 | 0.084 | 1154.762 |
| 10 | 966 | 0.364 | 2653.846 |
| 100 | 9657 | 3.158 | 3057.948 |
| 1000 | 96574 | 31.097 | 3105.573 |
| 10000 | 965743 | 310.485 | 3110.434 |

パイプライン化により、実行回数が多くなると高速化率が上がる

研究概要

・2種類のソーティングアルゴリズムをFPGAに実装

バイトニックソート

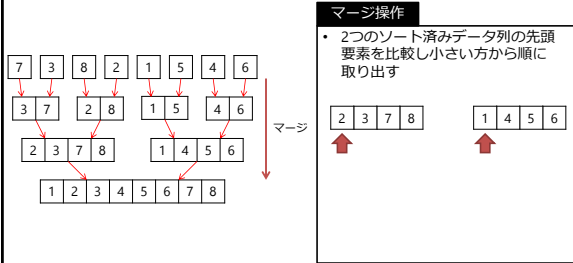
・ソーティングネットワークに基づいた回路
 ・任意のデータ幅のデータをソート可能

ロマージソート

・FIFOを用いた回路
 ・最大で524288要素のデータをソート可能

マージソート

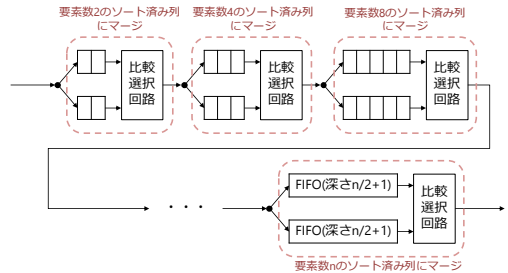
- 2つのソート済みのデータ列をマージして、1つのソート済みデータ列を作る動作を繰り返しソートを行う



マージソート回路

S.Todd. Algorithm and hardware for a merge sort using multiple processors. *IBM Journal of Research and Development*, pp509-517, 1978

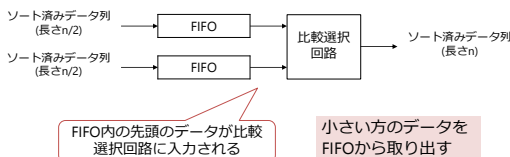
- 小さい要素数からマージを行い、大きなソート済み列を作る
- データはパイプライン的に処理される



マージの実装

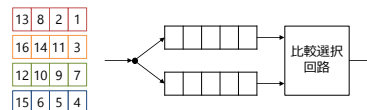
- 2つのFIFOと比較選択回路から構成

- ✓ **FIFO(First-In First-Out)**
 - 先に入れたデータが先に取り出されるデータ構造
- ✓ **比較選択回路**
 - 2つの入力のうち小さい方を出力



マージの動作

- n/2要素のソート済みデータを逐次に入力する
- n要素のソート済みデータが逐次的に出力される



実装結果

- ターゲットFPGA : Xilinx Virtex-7 XC7VX485T
- データ幅 : 32ビット

- ソート可能な最大要素数 : **524288**
- 動作周波数 : **278.339**[MHz]
- 処理時間(524288要素) : **3.767**[ms]

| 要素数 | Slice Register 使用数 % | Slice LUT 使用数 % | Block RAM 使用数 % | 動作周波数 [MHz] | |
|------|-------------------------|--------------------|--------------------|----------------|---------|
| 2 | 136 | 0 | 351 | 0 | 414.834 |
| 4 | 220 | 0 | 679 | 0 | 370.739 |
| 8 | 325 | 0 | 1040 | 0 | 372.731 |
| 16 | 423 | 0 | 1471 | 0 | 366.447 |
| 32 | 524 | 0 | 1991 | 0 | 326.813 |
| 64 | 643 | 0 | 2769 | 0 | 326.179 |
| 128 | 770 | 0 | 2249 | 0 | 324.160 |
| 256 | 908 | 0 | 2976 | 0 | 314.891 |
| 512 | 1122 | 0 | 3450 | 1 | 312.710 |
| 1K | 1345 | 0 | 4006 | 1 | 300.801 |
| 2K | 1577 | 0 | 4146 | 1 | 298.485 |
| 4K | 1818 | 0 | 4610 | 1 | 270.464 |
| 8K | 2068 | 0 | 5110 | 1 | 288.806 |
| 16K | 2327 | 0 | 5765 | 1 | 284.823 |
| 32K | 2595 | 0 | 6412 | 2 | 285.792 |
| 64K | 2872 | 0 | 7101 | 2 | 280.006 |
| 128K | 3158 | 0 | 7648 | 2 | 269.535 |
| 256K | 3455 | 0 | 8412 | 2 | 269.906 |
| 512K | 3763 | 0 | 8729 | 2 | 278.339 |

512要素以上のマージ回路でブロックRAMを使用することで、動作周波数の低下を防ぎ、リソースを節約

性能比較

- CPU : Intel Xeon X7460 2.66GHz
- C言語の標準関数qsort()と比較
- データ幅 : 32ビット

| 要素数 | CPU 処理時間 | FPGA 処理時間 | 高速化率 |
|------|---------------|--------------|--------|
| 2 | 109.32[ms] | 9.64[ns] | 11.338 |
| 4 | 197.35[ms] | 24.27[ns] | 8.129 |
| 8 | 398.07[ms] | 48.29[ns] | 8.243 |
| 16 | 868.19[ms] | 95.51[ns] | 9.090 |
| 32 | 1.96[ms] | 0.208[ms] | 9.428 |
| 64 | 4.36[ms] | 0.408[ms] | 10.689 |
| 128 | 9.69[ms] | 0.808[ms] | 11.995 |
| 256 | 21.148[ms] | 1.648[ms] | 12.833 |
| 512 | 45.958[ms] | 3.300[ms] | 13.927 |
| 1K | 99.786[ms] | 6.638[ms] | 14.593 |
| 2K | 215.658[ms] | 13.756[ms] | 15.677 |
| 4K | 464.747[ms] | 30.329[ms] | 15.324 |
| 8K | 993.881[ms] | 56.772[ms] | 17.507 |
| 16K | 2116.377[ms] | 115.093[ms] | 18.388 |
| 32K | 4502.533[ms] | 229.363[ms] | 19.631 |
| 64K | 9553.16[ms] | 468.158[ms] | 20.406 |
| 128K | 20157.653[ms] | 972.638[ms] | 20.725 |
| 256K | 43593.288[ms] | 1942.547[ms] | 22.441 |
| 512K | 94205.103[ms] | 3767.327[ms] | 25.006 |

524288要素のソートのとき、約25倍の高速化

まとめ

- ・本研究ではバイトニックソートとマージソートを実行する回路をFPGAに実装した
- ・バイトニックソート回路では、ビットシリアルを用いることで**任意のデータ幅**のデータを最大1024要素ソート可能となった
- ・マージソート回路では、ブロックRAMを用いることで32ビットのデータで**最大524288要素**をソート可能となった