

遺伝的アルゴリズムの高速実行に適した命令セットを持つ専用 RISC プロセッサ DLX-GA

若林 真一[†] 小泉 慎哉[†] 小出 哲士^{††}
井村 紀道[†] 藤原 一成[†]

本論文では、遺伝的アルゴリズム (GA) の実行における計算時間の短縮を目的として、任意の GA を高速に実行可能な RISC プロセッサ DLX-GA を提案する。提案プロセッサ DLX-GA は DLX アーキテクチャをベースとした RISC プロセッサであり、GA の実行において多用されるビット演算命令や乱数発生命令、SIMD 型命令等をサポートし、これらを 6 段のパイプラインで処理することにより GA 実行の高速化を実現する。提案 RISC プロセッサを HDL 設計し、CMOS 0.35 μm スタンダードセルテクノロジーを用いて 4.93 mm 角の LSI チップとして実現し、評価ボード上で性能評価を行った。その結果、開発したプロセッサチップが仕様どおりに動作することを確認した。

A RISC Processor DLX-GA with Instruction Set Suitable for High-speed Execution of a Genetic Algorithm

SHIN'ICHI WAKABAYASHI,[†] SHINYA KOIZUMI,[†] TETSUSHI KOIDE,^{††}
NORIMICHI IMURA[†] and KAZUNARI FUJIWARA[†]

This paper proposes a new RISC processor for high speed execution of genetic algorithms (GAs). The proposed RISC processor was designed based on the DLX architecture, and a new instruction set, which was effective for high-speed execution of GAs, was implemented. The proposed RISC processor was designed with the hardware description language, and it was fabricated as an LSI chip with the CMOS 0.35 μm standard cell technology. From the evaluation of the fabricated LSI chip using the evaluation board, we have shown that all the functions specified by the specifications of the chip were correctly realized.

1. はじめに

遺伝的アルゴリズム (Genetic Algorithm, GA) は自然界の遺伝メカニズムに基づく探索アルゴリズムとして、1970 年代に John Holland によって開発された¹⁾。GA は優れた解探索能力を持つ一方で、汎用逐次処理コンピュータ上で実行した場合、複雑な問題に対しては多大の計算時間を要するという問題点を持つ。この問題点の改善策として GA のハードウェア化が研究されている^{2),6),8),9)}。たとえば、Scott らは再構成可能な FPGA を用いて定常状態 GA を実現するハードウェア GA を提案している⁶⁾。Yoshida らは粗粒度の並

列処理を実現した GA 向けの VLSI を提案している⁹⁾。著者らも適応的遺伝的アルゴリズムのハードウェア化について研究し、実際に Genetic Algorithm Accelerator (GAA) として LSI チップ化している⁸⁾。このアプローチの主な問題点は、ハードウェアは選択・交差・突然変異等の遺伝オペレータが前もって固定されているため、適用可能な問題が限られるということである。

そこで本研究では、任意の GA の効率的な実行を実現することを目的として、GA 実行に適した命令セットを持つ新しい RISC プロセッサ DLX-GA を提案する。汎用プロセッサに GA 向きの命令を付加することで GA 実行のパフォーマンスを向上させようとする試みは著者らの調べた限りではこれまでに報告されておらず、本研究が初めての試みである。提案プロセッサ DLX-GA は、Hennessy らが提案した DLX アーキテクチャ³⁾をベースとして設計されており、ロード/ストアや算術論理演算等の通常の命令に加えて、GA の

[†] 広島大学大学院工学研究科
Graduate School of Engineering, Hiroshima University

^{††} 広島大学ナノデバイス・システム研究センター
Research Center for Nanodevices and Systems,
Hiroshima University

実行において多用されるビット演算命令や乱数発生命令, SIMD 型命令等, GA の高速実行に有効な命令を備えている. そして, これらの命令を 6 段のパイプラインで処理することにより高速な命令実行を実現する. また, 処理の高速化のためにオンチップの命令キャッシュを備えている.

提案 RISC プロセッサを Verilog-HDL 言語を用いて RTL 設計し, CMOS 0.35 μm スタンダードセルテクノロジーを用いて 4.93 mm 角の LSI チップとして実現し, 評価ボード上で性能評価を行った. その結果, 開発したプロセッサチップが仕様どおりに動作することを確認した.

2. 提案プロセッサ DLX-GA

2.1 GA の特徴

GA の効率的な実行を実現するプロセッサアーキテクチャを得るために, GA をプログラムとして実現した場合のプログラミング上の特徴について考察した.

(1) ビット指向のオペレーション

GA は一般的に AND, ADD のような通常のワード単位のオペレーションだけでなく, ビット指向のオペレーションも頻繁に使用する. つまり, 遺伝的操作はメモリワード全体に対してだけでなく, ワード中の一部に対しても適用される.

(2) 乱数の高頻度な使用

GA は選択・交差・突然変異等のアルゴリズムの実行における様々なステージで乱数を頻繁に使用する.

(3) SIMD 的オペレーション

一般に, GA では染色体の集合が人口を構成し, 同じ遺伝的操作が人口中のすべての染色体に対して適用される. これらの操作は SIMD (Single Instruction Stream, Multiple Data Stream) 操作と見なすことができる.

2.2 命令セットアーキテクチャ

前節で述べた GA の特徴に基づいて, GA のソフトウェアによる高速実行が可能となるように提案プロセッサの命令セットを設計した. 算術論理演算命令やロード/ストア命令等通常の汎用プロセッサがサポートする命令も必要なので, ベースとなるアーキテクチャとして DLX アーキテクチャ³⁾を採用し, 浮動小数点命令を除くすべての DLX 命令をサポートしている.

DLX の命令セットに加えて, GA の効率的な実行を可能とする 27 個の新しい命令をサポートした. これらの命令は 3 つのカテゴリに分類される. ここでは紙面の都合上, 命令セットの完全な記述は省略し, 以下に各カテゴリに属する命令のうちの代表的なものについて説明する. 命令セットの完全な記述は文献 4) を参照されたい.

(1) ビット演算命令

このカテゴリに属する命令は, 1 ワード中の指定された数ビットをオペランドとして扱う. このカテゴリには算術論理演算や MOVE 等を含む 16 個の新たな命令を考案した.

```
move_bits [rs][rd][ra][rb][rc]
```

この命令はソースレジスタの指定した数ビットをデスティネーションレジスタ内の指定した位置に移動させる. すなわち, $\text{rd}[\text{rb}:\text{rb}+\text{rc}-1] \leftarrow \text{rs}[\text{ra}:\text{ra}+\text{rc}-1]$.

```
and_bits [rs][rd][ra][rb][rc]
```

この命令はソースレジスタとデスティネーションレジスタの指定した数ビットについての論理積をとる. すなわち, $\text{rd}[\text{rb}:\text{rb}+\text{rc}-1] \leftarrow \text{rs}[\text{ra}:\text{ra}+\text{rc}-1] \& \text{rd}[\text{rb}:\text{rb}+\text{rc}-1]$.

(2) 乱数発生命令

このカテゴリには乱数発生に関する 5 つの命令がある. 提案プロセッサはクロックサイクルごとに 96 ビットの疑似乱数を生成する疑似乱数ジェネレータをプロセッサ内部に持っている.

```
set_rand_num [rd][imd]
```

この命令は範囲が 0 ~ imd-1 の乱数をレジスタ rd にセットする.

(3) SIMD 型命令

このカテゴリには 4 つの SIMD 型命令が属する.

```
rotate_r [rs1][rs2][imd]
```

この命令はレジスタ rs1 と rs2 の内容を同時に imd ビット右方向に回転する.

```
exchange_rotate_r [rs1][rs2][imd]
```

この命令はレジスタ rs1 と rs2 の内容を同時に imd ビット右方向に回転し, rs1 の内容を rs2 に, rs2 の内容を rs1 に格納する.

2.3 マイクロアーキテクチャ

本論文で提案する RISC プロセッサ DLX-GA のアーキテクチャを図 1 に示す.

DLX-GA は 32 ビット RISC プロセッサである. 前節で述べたように, DLX-GA は DLX アーキテクチャ³⁾に基づいて設計しているため, プロセッサのほとんどの特徴は DLX と同じである. DLX-GA は 32 ビット

³⁾ 浮動小数点命令を省いたのは, ハードウェア資源の制約によるものである.

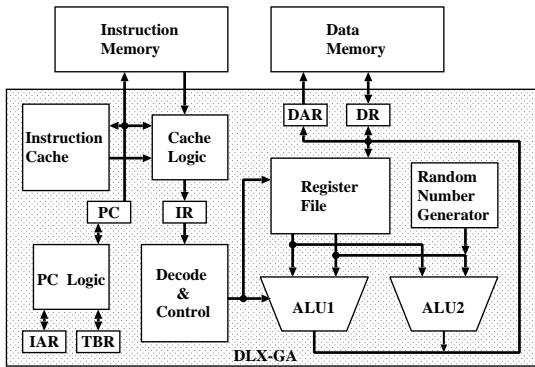


図1 DLX-GA アーキテクチャ
Fig.1 DLX-GA architecture.

表1 DLX-GA の仕様
Table 1 Specifications of DLX-GA.

命令メモリアドレスバス	15ビット
命令メモリデータバス	32ビット
データメモリアドレスバス	23ビット
データメモリデータバス	32ビット
命令キャッシュ	直接マップ方式 512ライン
レジスタファイル	32ビット×32ワード
割込み	2レベル
クロック周波数	70 MHz

の固定長命令フォーマットを持つロード/ストア・アーキテクチャであり、ハーバードアーキテクチャを採用し、命令メモリバスとデータメモリバスを別々に持つ。また、DLX-GAは処理の高速化のためにオンチップの命令キャッシュを持つ。しかし、ハードウェア資源の制約のためデータキャッシュは持っていない。表1にDLX-GAの主な仕様を示す。

DLX-GAにおけるすべての命令の実行は、命令フェッチIF、命令デコードID、実行EX1、実行EX2、メモリアクセスMEM、レジスタ書き込みWBの6つの基本ステージからなり、これらを6段のパイプラインで処理する。パイプラインストールへの対応としては、構造ハザードについては資源の追加を行っており、データハザードについてはフォワーディングを行っている。そして、制御ハザードについては命令デコード(ID)ステージで早期に分岐を行うようにし、それにとまなう1つのディレイスロットには遅延分岐を採用し、ディレイスロットができないようにしている。

プロセッサのデータバスには、32ビット×32ワードの汎用レジスタ群を実現したレジスタファイルと、SIMD型の命令を実現するために2つのALUがある。さらに、データバスは一次元セルラオートマタに基づく疑似乱数ジェネレータ⁷⁾を持つ。この乱数ジェ

ネレータは96ビットの疑似乱数を毎クロックサイクル生成する。GAで使用する乱数は適度な一様分布を示す乱数であればGAのパフォーマンスに及ぼす影響の差はないということが示されており⁵⁾、今回採用した疑似乱数は適度な一様分布を示すことが知られている⁷⁾。乱数ジェネレータにより生成された数を0から1までの固定小数点乱数と見なし、命令により与えられた定数との乗算を行い、所望の範囲の整数乱数とする。

2.4 LSI設計

提案プロセッサDLX-GAは、ローム(株)製の4.93mm角、信号ピン数111ピン、5層配線(ポリシリコン:2層、メタル配線:3層)、CMOS 0.35 μ mテクノロジスタンダードセルチップに実装した。ライブラリにはロームのパスポートライブラリを使用し、試作はVDEC(東京大学大規模集積システム設計教育研究センター)に依頼した。

回路設計はVerilog-HDLを用いてレジスタ転送レベルで記述することにより行い、各パイプラインステージをそれぞれ単独のモジュールとして構成し、トップモジュールでそれらのインターフェースを記述した。論理合成においては、後の配置配線でALUやレジスタファイル等のデータバス系と、命令デコーダ等の制御系を分けて配置するためにフラットにしないで合成している。Verilog-HDLによる回路記述は、コメントを含めて約8,000行である。Synopsys社のDesign Compilerを用いて論理合成を行った結果、セル数(キャッシュメモリを除く)が17,087、ネット数17,193となった。論理合成後、Cadence社のVerilog-XLシミュレータを用いてゲートレベルでシミュレーションを行って機能検証を行い、その後、Avant!社のApolloを用いて配置配線を行った。配置配線結果から得られたタイミング情報をバックアノートして、再度、シミュレーションによる機能検証を行い、クロック周波数70MHzでの動作を確認した。最後にレイアウト検証を行い、マスクパターンデータを作成した。

3. 提案プロセッサの評価

3.1 ソフトウェア開発環境

提案プロセッサのソフトウェアを開発するために、提案プロセッサ専用のアセンブラを開発した。提案プロセッサ専用のコンパイラについてはまだ開発を行っていない。このため、提案プロセッサのプログラム開発においてはDLXプロセッサのC言語コンパイラを流用し、C言語のソースプログラムに対してDLXコンパイラが出力するDLXアセンブリコードを入手に

表 2 1 世代あたりの実行時間の比較
Table 2 Comparison of execution time for one generation.

	評価	交差	突然変異	その他	計
DLX-GA	2660	410	244	2156	5470
割合 (%)	48.6	7.5	4.5	39.4	100.0
DLX	2701	693	4245	2222	9861
割合 (%)	27.4	7.0	43.0	22.5	100.0

より修正し, DLX-GA のアセンブリコードを生成して, DLX-GA アセンブラにより提案プロセッサの機械語命令プログラムを得ている.

3.2 シミュレーションによる評価

Verilog-XL シミュレータ上でのシミュレーションにより提案プロセッサ (DLX-GA) とオリジナルの DLX プロセッサの比較実験を行った. 対象とする最大化問題として count1 (ビット列中の 1 の数が評価値) を採用し, GA 実行において 1 世代あたりの計算時間 (単位はマイクロ秒) を比較した結果を表 2 に示す. 表中の「割合」は全体の計算時間に占める各 GA 演算の実行時間の比率を示す. 表から分かるように, DLX-GA 特有の命令を用いることにより交差および突然変異においてかなりの実行時間の短縮がなされている. また, 交差および突然変異の全体の計算時間に占める割合が, DLX プロセッサにおいては 50% であったのに対して, DLX-GA では 10% 程度になっている. 全体として DLX プロセッサより約 1.8 倍の高速実行が実現されている.

3.3 評価ボードによる実チップの検証

提案プロセッサの試作 LSI チップを評価するため, プリント基板を用いた評価ボードを開発した. 評価ボードを用いて試作チップの機能検証と性能評価を行った結果, すべての命令が仕様どおりに実現できていることが確認された. プロセッサの動作周波数については, プロセッサ周辺回路の回路遅延のために 10 MHz までの動作が確認された. これについては, メモリを含む周辺回路に高速 IC を用いることにより動作周波数を 50 MHz まで向上可能であると予測している.

4. あとがき

今後の課題としては提案プロセッサで導入した GA 専用命令の有効性の詳細な検討がある. また, 提案プロセッサ専用のコンパイラの開発も今後の課題である.

謝辞 卒業研究として DLX-GA プロセッサの評価ボード製作に携わった本学卒業生の加藤洋明君と高橋

秀宗君に感謝する. 本研究の一部は平成 12 年度科学研究費補助金基盤研究 (C) (2) (課題番号 12838008) による. 本チップ試作は東京大学大規模集積システム設計教育研究センターを通じ, ローム (株) および凸版印刷 (株) の協力で行われたものである.

参考文献

- 1) Goldberg, D.E.: *Genetic Algorithms in Search, Optimization, and Machine Learning*, Addison-Wesley Publishing Company (1989).
- 2) Graham, P. and Nelson, B.: A hardware genetic algorithm for the traveling salesman problem on Splash 2, *Field Programmable Logic and Applications*, Moore, W. and Luk, W. (Eds.), pp.352–361, Springer (1995).
- 3) Hennessy, J.L. and Patterson, D.A.: *Computer Architecture: A Quantitative Approach, 2nd Edition*, Morgan Kaufmann Publishers, Inc. (1996).
- 4) Koizumi, S.: Design and implementation of a RISC processor for high-speed execution of a genetic algorithm, Master Thesis, Graduate School of Engineering, Hiroshima University (2001).
- 5) Meysenburg, M.M. and Foster, J.A.: Randomness and GA performance, Revisited, *Proc. Genetic and Evolutionary Computation Conference*, pp.425–432 (1999).
- 6) Scott, S.D., Samal, A. and Seth, S.: HGA: A hardware-based genetic algorithm, *Proc. ACM/SIGDA 3rd International Symp. on FPGAs*, pp.53–59 (1995).
- 7) Serra, M., Slater, T., Muzio, J.C. and Miller, D.M.: The analysis of one-dimensional linear cellular automata and their aliasing properties, *IEEE Trans. Computer-Aided Design of Integrated Circuits and Systems*, Vol.9, No.7, pp.767–778 (1990).
- 8) 若林真一, 小出哲士, 八田浩一, 中山善勝, 後藤睦明, 利根直佳: 交差手法の適応的選択機能を組み込んだ遺伝的アルゴリズムの LSI チップによる実現, *情報処理学会論文誌*, Vol.41, No.6, pp.1766–1776 (2000).
- 9) Yoshida, N., Yasuoka, T. and Moriki, T.: Parallel and distributed processing in VLSI implementation of genetic algorithms, *Proc. 3rd International ICSC Symp. on Soft Computing*, pp.450–454 (1999).

(平成 14 年 10 月 15 日受付)

(平成 14 年 12 月 3 日採録)