

リコンフィギャラブル・ロジック向き論理ブロックの提案と評価

飯田 全 広[†] 末吉 敏 則^{††}

リコンフィギャラブル・コンピューティングに適したプログラマブル・ロジックの論理ブロックを提案する。本論理ブロックは、回路の構成データの単位であるコンテキストを複数格納するための構成データキャッシュと、マルチコンテキスト化およびクラスタ化の両機能を有する LUT (Look Up Table) を持つ。本稿では、ベンチマーク回路を用いた評価結果から、従来の 4 入力 LUT を用いた論理ブロックと速度性能が同じ場合には、実装密度が構成データキャッシュによって約 2.5 倍に向上することを示す。ただし、構成データキャッシュ内の回路は LUT へ書き込む必要がある。また、LUT のクラスタ化によって実装効率は約 6% 改善し、さらに LUT のマルチコンテキスト化によって実装面積および構成データ量がともに最小値を示すことを報告する。

Proposal and Evaluation of a Logic Block Architecture for Reconfigurable Logic

MASAHIRO IIDA[†] and TOSHINORI SUEYOSHI^{††}

In this paper, we propose a logic block architecture of programmable logic that is suitable for reconfigurable computing. Our logic block contains configuration data cache for holding multiple contexts and the LUT (Look Up Table) that has the functions of multi-context and clustering. The context is a unit of configuration data for circuit. we evaluate implementation efficiency, implementation area and configuration data bits through mapping some benchmark circuits. As a result of the evaluation, the implementation density is improved up to about 2.5 times than the conventional logic block using 4-LUT by means of the configuration data cache. The implementation efficiency is improved about 6% by clustering of LUT. Moreover, the implementation area and the amount of the configuration data indicate minimum values respectively by means of the multi-context.

1. はじめに

近年、FPGA (Field Programmable Gate Array) に代表されるプログラマブル・ロジックは、試作や小ロットの製品等で数多く使用され、めざましい発展をとげている。他の集積回路と同様に、FPGA は集積度と配線性の向上をめざし、論理ブロックや配線構造の改善やメモリの取り込み等、様々な研究^{1),2)} が進められてきている。また、FPGA は回路を書き換えられるという特性 (再構成性) を持つ。これを積極的に利用したリコンフィギャラブル・コンピューティング (Reconfigurable Computing, 以下 RC と略す) と呼ばれる新しい計算パラダイムも認知され、活発に研究

が行われるようになってきた³⁾。RC の特徴は、ハードウェアの高い性能とソフトウェアの柔軟性をあわせ持つところにある。

しかしながら、現在市販されている FPGA を RC に利用する場合、集積度や再構成の速度等多数の問題点が見つかっており、それらを改善しなければいっその性能向上は難しい⁴⁾。多くの問題は、プログラマブル・ロジックに対して市場が求めている能力と RC が求めている能力とに差異があるためと考えられる。したがって、RC の能力向上のためには試作に使われている FPGA の流用ではなく、RC に適したリコンフィギャラブル・ロジック・デバイス (Reconfigurable Logic Device, 以下 RLD と略す) が不可欠である。

本稿では、現行 FPGA ののかかえる問題を解消するために、構成データキャッシュと高機能化した LUT (Look Up Table) を持つ RLD 向き論理ブロックを提案する。以下、2 章では現行 FPGA を RC に使用した場合の問題点を明らかにするとともに、本研究で提案する論理ブロックの構成について説明する。3 章

[†] 熊本大学大学院自然科学研究科

Graduate School of Science and Technology, Kumamoto University

^{††} 熊本大学工学部数理情報システム工学科

Department of Computer Science, Faculty of Engineering, Kumamoto University

では評価モデルおよび計算機実験の環境を示し、4章で実験結果を述べる。5章では現行FPGAとの比較とともに提案論理ブロックの効果について考察する。そして、6章で関連研究を示し、7章でまとめと今後の課題を述べる。

2. RLD 向き論理ブロックの提案

2.1 RLD の要件

従来のFPGAを使用したRCでは、ハードウェア化による性能向上を再構成に要するオーバーヘッドが相殺されることが多い。また、デバイス容量以上の回路を実装したい場合には、処理ができないか、または長い再構成時間のために極端な性能低下が見られる。そのため、RCの用途は限定され、DNAパターンマッチング⁵⁾や暗号解析⁶⁾等の比較的処理内容の変更が少ないアプリケーションでしか十分な効果が得られていない。

元来、FPGAはASIC等と比べて実装密度が低く、動作速度も遅いという欠点がある。RCでは処理内容に応じてデータ幅や演算器構成等を最適化することで欠点を補い、いっそうの性能向上を図るものだが、その最適化の時間が性能を左右する場合もある。また、ASIC等と異なりRCではある時点で必要な処理のみを実装することで、低コスト化や低消費電力化も期待できる。しかし、これも再構成時間が長い場合や、回路規模が固定される場合は実現が難しい。したがって、RLDに求められる要件は(1)実装密度と動作速度を向上させること(2)再構成時間が短いこと(3)回路の仮想化を実現すること、の3点である。

2.2 RLD 向き論理ブロックの提案

我々が提案するRLD向き論理ブロックは、複数のLUT構成を持つマルチコンテキスト・マルチグレインLUT(Multi-Context, Multi-Grain LUT, 以下MCMG-LUTと略す)と複数のコンテキストを格納するためのメモリである構成データキャッシュ(Configuration Data Cache, 以下CDCと略す)を搭載する。

図1の構成例では、6入力3出力LUTを6つのモードで使用する。図中のn-LUTはn入力LUTを示す。前半の3モード(a)~(c)は、LUTをクラスタ化している。(a)は2入力LUTを3個として機能するモードであり、(b)は3入力LUTを2個として機能するモードである。(c)は6入力LUTである。後半の3モード(d)~(f)は、LUTをマルチコンテキスト化するモードで、コンテキスト数とLUT粒度のバランスを選択でき、さらに複数のコンテキストを切り

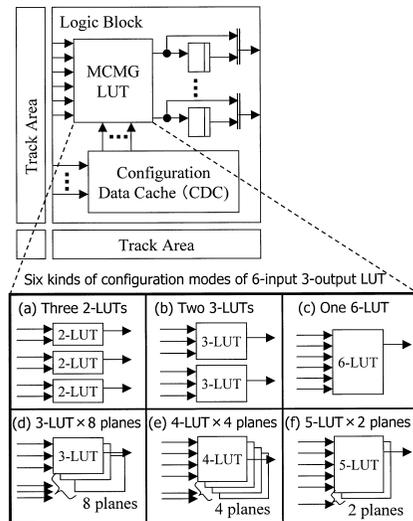


図1 RLD 向き論理ブロックの構成
Fig. 1 Logic block architecture for RLD.

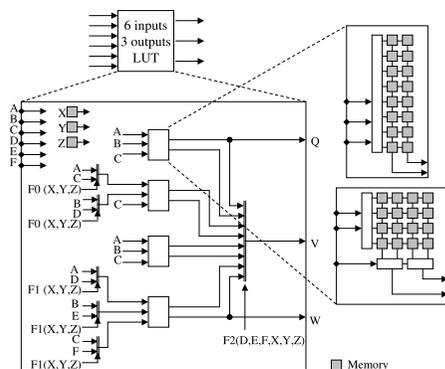


図2 MCMG-LUTの構成例
Fig. 2 A structure of MCMG-LUT.

替えながら動作が可能である。これらのモードは論理ブロックごとに設定可能である。

また、CDCには、MCMG-LUTへ書き込まれるコンテキスト(回路データ)とMCMG-LUTの構成モードを決定するモード変更用ビット等が複数セット格納され、LUT部が動作中であっても、専用のデータ線によってCDCの内容を書き換え可能な構成をとる。そのためCDCへの書き込み動作は、MCMG-LUTの動作には影響しない。

以上から、MCMG-LUTは従来のLUTより実装効率の改善と、回路ごとのLUT最適化、構成データ量の抑制等の効果が期待できる。一方、CDCは搭載可能な回路容量の増加、すなわち実装密度向上が期待でき、さらに再構成時間の隠蔽等、従来のFPGAに欠如していた機能を論理ブロックに加える。

また、図 2 に前述の MCMG-LUT の内部構成例を示す。A~F が入力、Q, V, W が出力である。また、X, Y, Z はモード変更用メモリを示し、LUT を構成するメモリは 3 入力 2 出力のメモリアレイ 4 個で構成している。F0, F1 は入力セクタの切替え信号であり、モード変更用メモリ X, Y, Z によって決められる。F2 は出力 V のセクタ切替え信号であり、入力信号 D, E, F およびモード変更用メモリ X, Y, Z によって決められる。

3. 評価モデルおよび計算機実験の環境

3.1 面積評価モデル

図 3 に示す面積評価モデルは、規則的に配置された独立した論理ブロック部と、水平・垂直方向の配線チャンネル部からなる。ただし、階層配線構造等は持たない。各トラックの幅はビット面積 BA (Bit Area) の平方根とし、論理ブロックは正方形と仮定する。また、評価モデルにおける単位面積 A_{LB} は、論理ブロックと隣接する水平・垂直方向の配線領域の面積の合計である(式 (1))。 W は配線トラック数、 FA (Fixed Area) は論理ブロック内の固定領域の総面積である。表 1 に各パラメータの定義を示す。

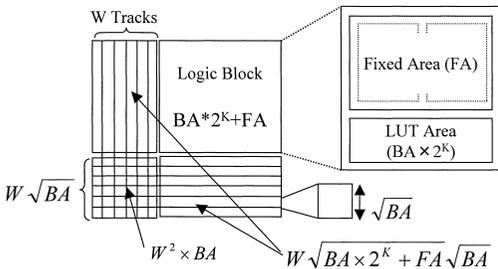


図 3 面積評価モデル
Fig. 3 Area model.

表 1 面積評価モデルのパラメータ一覧
Table 1 Parameters of the area model.

パラメータ	説明
IA	実装回路面積
N_{LB}	論理ブロック数
A_{LB}	論理ブロックの単位面積(配線領域を含む)
K	LUT への入力信号数(LUT の粒度)
BA	LUT を構成するメモリの単位面積
W	配線トラック数
a	K のトラック上の使用率係数
b	リセットやクロック等の固定配線数
FA	論理ブロック内の固定領域の総面積
FMA	FA 内の FF 等の固定領域の面積
BN	CDC を構成するメモリのビット数
R_a	BA に対するコンテキスト用メモリの面積比
CLA	MCMG-LUT のモード制御回路の面積

$$A_{LB} = (W^2 \times BA + 2W\sqrt{BA \times 2^K + FA}\sqrt{BA}) + (BA \times 2^K + FA) \quad (1)$$

このモデルは、トロント大の Rose らのモデル⁷⁾ から W と FA の 2 力所の変更を加えたモデルである。本稿では、 W を式 (2) のように LUT への入力信号数 K の関数とする。 a は LUT への入力信号数 K に対してトラックで必要とする割合を示す使用率係数である。 b はクロック、リセット等の固定配線数である。一方、 FA は、FF 等の固定領域の面積 FMA (Fixed Miscellaneous Area) と CDC のメモリ領域、MCMG-LUT のモード変更用メモリおよび制御回路領域の合計面積であり、式 (3) のように定義する。 BN は CDC を構成するメモリのビット数、 R_a は BA に対するコンテキスト用メモリの面積比、 CLA (Control Logic Area) は MCMG-LUT 内のモード変更用メモリおよび制御回路等の面積である。従来の論理ブロックは、 CLA および BN をゼロとした場合と同じである。また、CDC 用メモリのテクノロジーが LUT のメモリと異なる場合は、 R_a を用いて調整する。

$$W = a \times K + b \quad (2)$$

$$FA = FMA + BN \times R_a \times BA + CLA \quad (3)$$

以上から、評価回路を実装したときの実装回路面積 IA (Implementation Area) は、実装に要した論理ブロック数 N_{LB} と単位面積 A_{LB} の積と定義する(式 (4))。

$$IA = N_{LB} \times A_{LB} \quad (4)$$

次に、本評価で用いる各パラメータの値を示す。文献 7) から $1.25 \mu\text{m}$ CMOS プロセスのとき、 BA は $400 \mu\text{m}^2$ 、 FMA (文献 7) の FA に相当)は $5,100 \mu\text{m}^2$ となっており、4-LUT 論理ブロックの単位面積 A_{LB4} は $182,133 \mu\text{m}^2$ である。本研究もこの値を用いる。また、 CLA は $2,250 \mu\text{m}^2$ と仮定した。これはモード変更用メモリの 3 ビット ($1,200 \mu\text{m}^2 = BA \times 3$) と制御回路 ($1,050 \mu\text{m}^2$) の合計である。制御回路は文献 7) の FF を含まない FA 値 ($2,100 \mu\text{m}^2$) の半分程度の面積と見積もっている。また、 K の使用率係数 a は 1 とし、 b はリセット等の専用配線とグローバル配線を考慮して 12 本とする。コンテキスト用メモリの BA 面積比は LUT と同じと仮定し、 R_a は 1.0 とする。

3.2 遅延評価モデル

遅延評価モデルは、Rose らのモデル⁸⁾ をより詳細にしたモデルを採用する。本研究では、図 4 に示すように、平均配線遅延 DR をワイヤ遅延 DW とコネクションブロック遅延 DC に分解する。クリティ

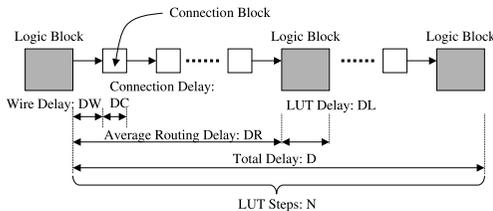


図4 リンコンフィギャラブル・ロジックの遅延評価モデル
Fig.4 Delay model.

表2 遅延評価モデルのパラメータ一覧
Table 2 Parameters of the delay model.

パラメータ	説明
D	FF 間の総遅延 (クリティカルパス遅延)
DW	ワイヤ遅延
DC	コネクションブロック遅延
N	FF 間の論理ブロック段数
R_d	DW-DC 遅延分配係数
DR	平均配線遅延
DL	論理ブロック遅延

カルパス上の総遅延 D は、論理ブロック段数 N 、平均配線遅延 DR および論理ブロック遅延 DL から式 (5) を用いて算出している。各パラメータは表 2 のとおりである。

$$D = (DR + DL) \times N \quad (5)$$

$$DR = \Sigma(DW) + \Sigma(DC) \quad (6)$$

一方、配線構造が不明の場合、 $\Sigma(DW)$ 、 $\Sigma(DC)$ は DW-DC 遅延分配係数 R_d と DR を用いて式 (7)、式 (8) と定義する。

$$\Sigma(DW) = DR \times R_d \quad (7)$$

$$\Sigma(DC) = DR \times (1 - R_d) \quad (8)$$

一般に、LUT の粒度が大きくなると (a) 論理ブロック遅延 DL が大きくなる (b) 論理ブロックの面積増によって論理ブロック間のワイヤ遅延 DW が大きくなる (c) LUT あたりの論理量が増えるために N が小さくなる、等の変化が起こる。

本研究では、論理ブロック間の配線長は、論理ブロックの辺長に比例すると仮定する。したがって、本論理ブロック間の平均配線遅延 DR_6 は、本論理ブロックの単位面積 A_{LB6} の平方根と 4-LUT 論理ブロックの単位面積 A_{LB4} の平方根との比と、4-LUT での平均配線遅延 DR_4 との積として求められる (式 (9))。

$$DR_6 = \frac{\sqrt{A_{LB6}}}{\sqrt{A_{LB4}}} \times DR_4 \times R_d + DR_4 \times (1 - R_d) \quad (9)$$

次に、遅延評価に用いる各パラメータの値を示す。文献 8) から $1.25 \mu\text{m}$ CMOS プロセスを採用した場合、4-LUT 論理ブロック遅延 DL_4 は 1.71 ns 、平均

配線遅延 DR_4 は 4.0 ns 、6-LUT 論理ブロック遅延 DL_6 は 2.38 ns である。従来の LUT に対してはこれらの値を用いる。MCMG-LUT の論理ブロック遅延 (DL_{n6}) は、従来の 6-LUT よりも 10% 程度遅延が増加すると見積もり、 2.62 ns とする。また、 R_d は 0.1 とする。

3.3 実装効率の定義

ここでは、実装効率 IE (Implementation Efficiency) を定義するために、占有ビット数 M_o と使用ビット数 M_u を導入する。 M_o は総論理ブロック数 N_{LB} 内に含まれる全 LUT のメモリビット数を示し、式 (10) によって算出する。一方、 M_u は、各論理ブロックの入力信号線数によって、LUT で使用されるメモリのビット数が異なることから、各入力 LUT 数 N_{LBn} を用いて、式 (11) によって算出する。 n は LUT への実際の入力数である。以上から、実装効率 IE は、 M_o に対する M_u の割合と定義する (式 (12))。

$$M_o = N_{LB} \times 2^K \quad (10)$$

$$M_u = \sum_{n=1}^K (N_{LBn} \times 2^n) \quad (11)$$

$$IE = M_u / M_o \times 100 \quad (12)$$

3.4 実装密度の定義

実装密度は単位面積あたりの実装論理量と定義されるが、ここでは従来の FPGA との比較のために、4-LUT を持つ論理ブロックに対する相対論理量を定義し、それを用いて実装密度を再定義する。

相対論理量 RLC (Relative Logic Capacity) は 4-LUT 論理ブロックに対して同じ回路を実装するのに要する対象論理ブロック数の比と定義する。たとえば、10 個の 4-LUT 論理ブロックを必要とした回路が、8 個の 5-LUT 論理ブロックで実装できるなら、5-LUT の論理ブロックは RLC が $10/8 = 1.25$ となり、4-LUT 論理ブロックより 1.25 倍の論理量を持つことになる。

次に、4-LUT 論理ブロックに対する対象論理ブロックの相対的な実装密度 RID (Relative Implementation Density) を式 (13) と定義する。 NC はコンテキスト数、 AR (Area Ratio) は面積比である。また、 NC と RLC の積を相対的な回路容量とする。たとえば、4-LUT を 2 個持つ論理ブロックが 4-LUT を 1 個しか持たない論理ブロックに対して 1.5 倍の面積であると仮定すると、4-LUT を 2 個持つ論理ブロックの実装密度は、 $NC = 2$ 、 $RLC = 1$ 、 $AR = 1.5$ から $RID = 2 \times 1/1.5 = 1.33$ となり、4-LUT を 1 個しか持たない論理ブロックの 1.33 倍である。

$$RID = NC \times RLC / AR \quad (13)$$

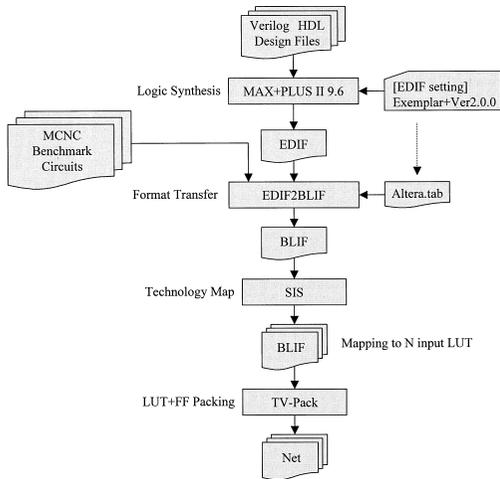


図 5 評価回路の実装フロー
Fig. 5 Flow of architecture evaluation.

3.5 構成データ量の算出方法

構成データは LUT に格納されるコンテキストと配線上的コネクションブロックのデータからなる。コンテキスト分の構成データ量は、LUT 使用数と LUT のビット数の積である。コネクションブロック分は、配線数(ネット数)、論理ブロック間の配線の平均接続点およびコネクションブロックのビット数の積として見積もる。本研究では平均接続点数を 4 カ所と仮定し、各コネクションブロックは 1 ビットのメモリで構成される。さらに、本方式の論理ブロックでは 3 ビットモード変更用メモリも構成データに加味する。

従来の LUT の構成データ量 (CD_c) の算出式を式 (14) に、本方式の論理ブロックの構成データ量 (CD_n) を式 (15) に示す。 N_{LUT} は LUT 数、 N_{NT} はネット数、 N_{LB} は論理ブロック数である。

$$CD_c = N_{LUT} \times 2^K + N_{NT} \times 4 \quad (14)$$

$$CD_n = N_{LUT} \times 2^K + N_{NT} \times 4 + N_{LB} \times 3 \quad (15)$$

3.6 計算機実験の環境

図 5 に、評価回路の実装フローを示す。Verilog-HDL で記述された評価回路は、ALTERA 社の MAX+PLUS II を用いて EDIF ネットリストを生成し、さらに、トロント大で開発された EDIF2BLIF⁹⁾ を用いて BLIF フォーマットへ変換する。そして、BLIF ネットリストを UCB の SIS¹⁰⁾ で 3-LUT から 7-LUT にマッピングし、同じくトロント大で開発された TV-Pack¹¹⁾ を用いて FF と LUT のパッキングを行うことで、最終的なネットリストを得ている。

一方、本方式のマルチコンテキスト化するモード (d)-(f) を使用する場合、LUT の各コンテキストは同時に動作しないことが前提となるが、上記のマッ

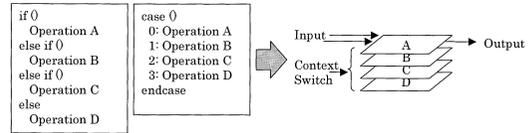


図 6 マルチコンテキストのマッピング方法
Fig. 6 Mapping method for multi-context LUT.

表 3 MCNC ベンチマーク回路一覧
Table 3 MCNC benchmark circuits.

回路名	
alu4	ex1010
apex2	ex5p
des	frisc
dsip	seq
elliptic	tseng

表 4 追加評価回路一覧
Table 4 Additional benchmark circuits.

回路名	説明
ACS4	ビタビ復号法の ACS 回路 (4 bit 精度)
DIV8	パイプライン割算器 (8 bit 精度, 4 段)
FFT6	8 ポイント FFT 回路 (6 bit 精度)
DES16	16 ループ版 DES 暗号回路
MA8I	複素積和演算回路 (8 bit 精度)
SCU	DLX プロセッサ (命令デコーダ部)

ング方法では保証できない。そこで、ここでは図 6 に示すように、Verilog-HDL の RTL 記述レベルで、明示的に同時動作しないことが分かっている case 文または if 文の各条件下の処理を、別のコンテキストにマッピングする方法を採用する。本方式へのマッピングは、オペレーションごとに図 5 に従って行い、それらを手作業で統合することで最終的なネットリストを得る。

3.7 評価回路

評価回路の選定は重要な問題である。偏った評価にならないように回路を選定する必要があるが、現実には明確な基準を設けることは難しい。CDC とクラスタ化の評価では、MCNC ベンチマーク回路¹²⁾ から表 3 に示した比較的大きな回路 10 種類と、表 4 の Verilog-HDL で記述された 6 種類の実用回路を用いる。

一方、マルチコンテキスト化の評価では、図 7 に示した 16 ビット精度、8 オペレーションの ALU 回路 (ALU16) を用いる。ここで、入力 A, B はデータ信号、入力 C はオペレーションを決めるコマンド信号、出力は Q である。図 6 に示した方法で実装する場合、一般に if 文や case 文の条件下の処理の回路規模は均一にはならない。また、条件数もデバイスのコンテキスト数と一致しない。ALU16 の各オペレーションも

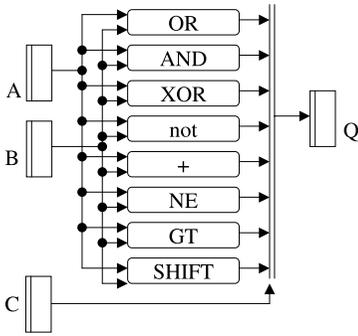


図 7 ALU16 回路の構成
Fig. 7 Structure of ALU16.

表 5 評価回路における使用論理ブロック数

Table 5 Number of logic blocks used for each benchmark circuit.

回路名	各 LUT 粒度における使用論理ブロック数				
	3-LUT	4-LUT	5-LUT	6-LUT	7-LUT
alu4	1,854	1,323	1,067	922	795
apex2	2,249	1,589	1,377	1,269	1,131
des	2,127	1,539	1,312	1,180	1,154
dsip	1,852	1,371	1,169	690	1,279
elliptic	3,892	2,916	2,500	2,043	1,988
ex1010	5,229	4,039	3,427	3,202	3,017
ex5p	348	245	137	115	104
frisc	4,644	2,973	2,535	2,252	2,191
seq	2,101	1,501	1,290	1,177	1,057
tseng	1,185	957	874	784	698
ACS4	267	179	97	100	83
DIV8	459	391	313	259	209
FFT6	3,982	2,640	1,813	1,622	1,159
DES16	2,194	1,931	950	726	981
MA8I	1,000	647	421	406	364
SCU	1,055	839	625	531	534
合計	34,434	25,080	19,907	17,278	16,744
相対論理量	0.73	1.00	1.26	1.47	1.50

異なる回路規模であり、条件数も 8 と多い。本方式の MCMG-LUT では 3-LUT \times 8 面 (モード (d)) はコンテキスト数が一致するが、4-LUT \times 4 面 (モード (e)) や 5-LUT \times 2 面 (モード (f)) には一致しない。以上から、マルチコンテキスト化の評価で採用する回路として適当であると判断した。

4. 実装結果

4.1 使用論理ブロック数

表 5 は、全評価回路を前述の方法で、3~7 入力の LUT にマッピングしたときの論理ブロック数である。相対論理量は 3.4 節の定義から求めた。4-LUT に対し、6-LUT は 1.47 倍の論理量があることが分かる。

また、マルチコンテキスト化の評価として、ALU16 を各粒度の LUT と、本方式の 3-LUT \times 8 面 (モー

表 6 ALU16 回路のマッピング結果

Table 6 Implementation results of ALU16.

LUT 粒度	LB 数 (個)	ネット数 (本)	構成データ量 (bits)	実装面積 (mm^2)
3-LUT	579	667	7,300	88.57
4-LUT	341	429	7,172	62.24
5-LUT	354	442	13,096	79.47
6-LUT	334	422	23,064	95.68
7-LUT	211	299	28,204	80.40
4-LUT*4	191	471	8,505	56.01
3-LUT*8	184	637	6,876	53.96

表 7 4-LUT に対する論理ブロック段数比

Table 7 Ratio of logic block stages in each LUT granularity to 4-LUT.

回路名	各 LUT 粒度の論理ブロック段数比 (対 4-LUT 比)				
	3-LUT	4-LUT	5-LUT	6-LUT	7-LUT
ACS4	1.38	1.00	0.73	0.69	0.50
DIV	1.28	1.00	0.88	0.76	0.76
FFT6	1.32	1.00	0.79	0.84	0.79
DES16	1.28	1.00	0.78	0.78	0.56
MA8I	1.05	1.00	0.59	0.68	0.59
SCU	1.31	1.00	0.88	0.81	0.88
平均	1.27	1.00	0.77	0.76	0.68

表 8 各 LUT 粒度における入力数の割合

Table 8 Rate of various inputs actually used in each LUT granularity.

入力信号数	各 LUT 粒度における入力信号数ごとの割合 (%)				
	3-LUT	4-LUT	5-LUT	6-LUT	7-LUT
1 入力	0.91	1.11	1.47	0.08	0.90
2 入力	17.59	13.38	7.59	9.21	6.55
3 入力	81.50	22.58	19.87	7.23	8.12
4 入力	-	62.94	24.07	16.16	18.19
5 入力	-	-	47.01	25.62	12.29
6 入力	-	-	-	41.70	17.96
7 入力	-	-	-	-	35.98

ド (d)) および 4-LUT \times 4 面 (モード (e)) とに実装した結果を表 6 に示す。表からマルチコンテキスト化によって、同じ LUT 粒度で比較した場合、ネット数は変化が少ないが、論理ブロック数は大きく減少していることが分かる。

4.2 論理ブロック段数比

表 7 は、追加評価回路における各 LUT 粒度の FF 間の最大論理ブロック段数を 4-LUT の値で正規化した論理ブロック段数比である。また、MCNC ベンチマーク回路における 4-LUT と 6-LUT の論理ブロック段数比 N_6/N_4 は、文献 8) に 0.7 程度であることが示されている。一方、追加評価回路では、表 7 に示したように 0.68~0.84 で、平均 0.76 である。

4.3 各 LUT への入力信号数

全評価回路をマッピングした結果として、実際に各粒度の LUT へ入力されている信号線数の割合を表 8 に示す。太字部分は、各 LUT 粒度の最大入力数へマッピングされた LUT の割合である。LUT 粒度が大きくなるに従って減少していくことが分かる。これは、マッピング・アルゴリズムにも依存するが、LUT 粒度を大きくするに従って、すべての入力ポートに信号を割り当てることができない LUT の割合が増えていくことを意味する。

5. 考 察

5.1 構成データキャッシュの効果

ここでは、CDC の効果を見るために、本方式の CDC を含む論理ブロック（モード (c)）と従来の FPGA で用いられる 4-LUT の論理ブロックとを比較し、4-LUT 論理ブロックに対する相対面積と実装密度を求める。

4-LUT 論理ブロックに対する相対的な回路容量は、3.4 節で述べたように、4.1 節で求めた相対論理量と、MCMG-LUT および CDC 内に搭載できるコンテキスト数の積である。たとえば、CDC 容量が 64 ビットの場合、本方式の論理ブロック内には、MCMG-LUT 分を含めて 2 つのコンテキストが保持できる。したがって、4-LUT に対する回路容量は、 $2 \times 1.47 = 2.93$ となり、1 個の 4-LUT を持つ論理ブロックに対して、本方式の論理ブロックは 2.93 倍の論理回路が搭載できる。また、実装密度は式 (13) から 4-LUT 論理ブロックに対する実装可能な回路容量に面積比を掛けた値である。

表 9 に、CDC のビット数 BN の変化に対する 4-LUT との面積比および回路容量、そして 4-LUT 面積あたりの回路容量を示す。表 9 から分かるように、4-LUT 面積あたりの回路容量は、CDC の容量が 128 ビットまでは低いが 256 ビットあたりで逆転し、それ以上では増加する。512 ビットでは 13.20 倍の回路容量を持つが、面積も 7.47 倍になるため、4-LUT 面積あたりの回路容量は約 1.77 倍になる。

次に、CDC の回路遅延への影響を考察する。表 10 は、CDC 内のビット数 BN ならびに 4-LUT と 6-LUT の論理ブロック段数比 N_6/N_4 が変化したときの、4-LUT に対する遅延比である。なお、 N_6/N_4 の変化は 4.2 節から 0.7~0.9 としている。表における太字は、4-LUT から 6-LUT への変更による論理ブロック段数の減少が CDC による配線遅延の増加を上回り、その結果として総遅延 D が小さくなるケースである。

表 9 BN に対する 4-LUT 面積あたりの回路容量

Table 9 Number of circuits per the area of 4-LUT in each BN.

CDC ビット数 BN (bit)	4-LUT に対する		4-LUT 面積 あたりの実装密度
	面積比	回路容量	
64	4.66	2.93	0.63
128	5.16	4.40	0.85
256	6.02	7.33	1.22
512	7.47	13.20	1.77
1,024	9.91	24.94	2.52
2,048	14.08	48.41	3.44
4,096	21.32	95.35	4.47
8,192	34.22	189.24	5.53

表 10 LUT 段数比ごとの 4-LUT に対する遅延比 ($R_d=0.1$)

Table 10 Delay ratio in every stage ratio of LBs to 4-LUT ($R_d=0.1$).

CDC ビット数 BN (bit)	論理ブロック段数比ごとの遅延比				
	0.70	0.75	0.80	0.85	0.90
64 bit	0.87	0.93	0.99	1.05	1.12
128 bit	0.87	0.94	1.00	1.06	1.12
256 bit	0.88	0.95	1.01	1.07	1.13
512 bit	0.90	0.96	1.02	1.09	1.15
1,024 bit	0.92	0.98	1.05	1.11	1.18
2,048 bit	0.95	1.01	1.08	1.15	1.22
4,096 bit	0.99	1.06	1.13	1.20	1.27
8,192 bit	1.05	1.12	1.20	1.27	1.35

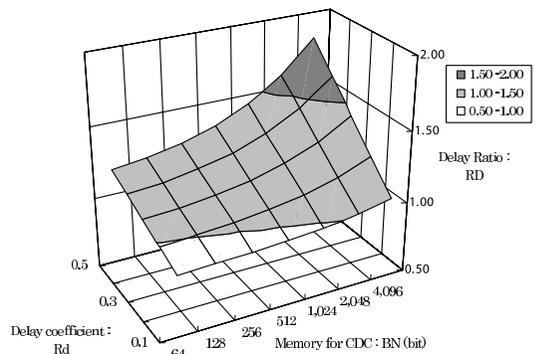


図 8 R_d による 4-LUT に対する遅延比の変化 ($N_6/N_4=0.75$)

Fig. 8 Delay ratio to 4-LUT by R_d ($N_6/N_4=0.75$).

また、テクノロジーの変化によって R_d の比率が変わった場合を図 8 に示す。なお、 N_6/N_4 は 0.75 としている。一般に、 R_d はプロセス・テクノロジーに依存し、プロセスが微細化するとスイッチング遅延に対して配線遅延が相対的に大きくなる (R_d は大きくなる)。図 8 から、 R_d が 0.1 で CDC 容量が 2,048 ビットまでと、 R_d が 0.2 で CDC 容量が 64 ビットまでは、4-LUT より遅延値が小さい ($RD < 1.0$)。しかし、それ以外の CDC 容量では 4-LUT より遅くなる。

なお、4-LUT より遅延を大きくしないという条件下では、論理ブロック段数比 0.75、DW-DC 遅延分配

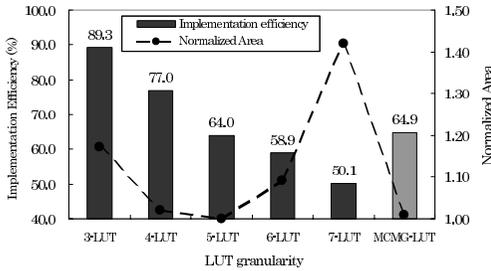


図 9 評価回路における正規化面積と実装効率

Fig. 9 Normalized area and implementation efficiency.

係数 R_d が 0.1 のとき、表 10 から CDC 容量が 1,024 ビットまで搭載でき、このとき実装密度は表 9 から約 2.5 倍に引き上げることが可能であることが分かった。ただし、構成データキャッシュ内の回路は LUT へ書き込んでから実行する必要がある。

5.2 クラスタ化の効果

次に、LUT におけるクラスタ化の効果について考察する。MCMG-LUT は、2-LUT \times 3 個 (モード (a)) または 3-LUT \times 2 個 (モード (b)) として使用することができる。一方、表 8 に示したように、最大入力数未満の LUT が、ある一定の割合で存在し、たとえば 6-LUT の場合、2 入力の LUT は約 9% あり、3 入力の LUT は約 7% ある。したがって、これらの LUT をクラスタ化することで、論理ブロック数を削減できる。

図 9 は、全評価回路を 3-LUT から 7-LUT と、MCMG-LUT にマッピングした正規化面積と実装効率を示している。MCMG-LUT への実装は、完全にクラスタ化できた場合である。また、クラスタ化の効果を見るために、本方式の論理ブロックは CDC を含まない。実装効率は式 (12) から求めた。正規化面積は最小値で正規化した値である。その結果、本方式の論理ブロックは、正規化面積は最小値に近づき、実装効率も従来の 6-LUT より約 6% 改善されている。

5.3 マルチコンテキスト化の効果

次に、マルチコンテキスト化の効果について考察する。図 10 に、4.1 節で述べた 7 種類の LUT に実装したときの 4-LUT に対する面積比を示す。比較のために、本方式の論理ブロックは CDC を搭載しない。グラフの横軸は LUT 粒度を示し、縦軸に 4-LUT に対する面積比を示した。その結果、本方式の 3-LUT \times 8 (モード (d)) に実装した場合が最小面積を示し、従来の 4-LUT と比べて約 87% で実装できることが分かる。

また、図 11 に 4-LUT の値で正規化した構成データ量を示す。モード (d) およびモード (e) とともに、

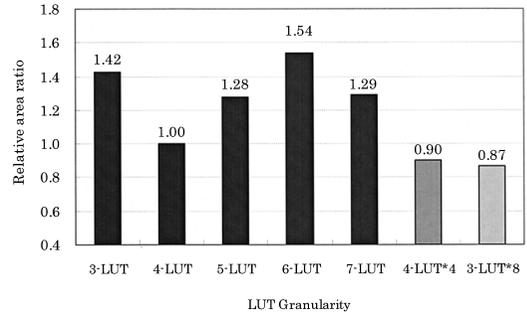


図 10 ALU16 の面積比較

Fig. 10 Area comparison of ALU16.

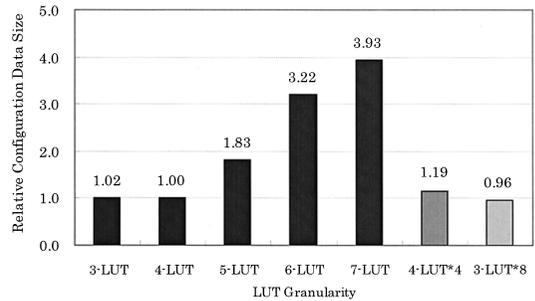


図 11 ALU16 の構成データ量比較

Fig. 11 Comparison of relative configuration data size for ALU16.

構成データ量は、従来の LUT 中で最小データビット数を示す 4-LUT と同等であることが分かる。

実装面積、構成データともに削減できたのは、従来の 3-LUT から 7-LUT では出力段のセレクトが LUT を用いて構成されるのに対して、本方式では配線の切替えによって行われたためと考えられる。

6. 関連研究

FPGA のマルチコンテキスト化の研究は広く行われている。MIT の DPGA (Dynamically Programmable Gate Arrays)³⁾ は、コンテキスト用メモリとして DRAM を持ち、LUT へロードすることで回路を切り替える。また、慶応大の HOSMIH¹⁴⁾ も DRAM をコンテキスト用メモリに使用している点で DPGA と近いが、コンテキストの切替えをデータ駆動で行うという特徴がある。これらの方式は、LUT そのものは従来の FPGA と同じである。本提案方式では LUT 粒度も可変となり、より自由度が高い。

また、NEC の DRLE (Dynamically Reconfigurable Logic Engine)¹⁵⁾ は、論理ブロックを 4×4 の UC (Unified Cell) と呼ぶ統合セルに置き換え、LUT とクロスバスイッチとして働かせる。UC は 4×4 の

MC (Memory Columns) からなり、各 MC を多ビット化することで複数のコンテキストを実装する。コンテキストの切替えは RC (Reconfiguration Controller) からの制御信号によって行われる。DRLE は LUT が複数のモードを持つ点で本提案方式に近いが、コンテキストデータを LUT のメモリエセル内に配置する点が異なる。本方式では、CDC と LUT の 2 階層でコンテキストを管理する。

7. まとめと今後の課題

7.1 まとめ

本稿では、LUT のマルチコンテキスト化とクラスタ化の両方を実現した MCMG-LUT と、論理ブロック内に複数のコンテキストを格納するための構成データキャッシュ (CDC) を搭載した RC 向きの論理ブロックを提案した。そして、提案方式の各機能の効果を実装密度、実装面積、実装効率、および構成データ量について評価した。

その結果、 $DW-DC$ 遅延分配係数 $R_d = 0.1$ 、論理ブロック段数比 $N_6/N_4 = 0.75$ 、4-LUT に対して性能を落とさないという条件下で、搭載できる CDC の最大容量は 1,024 ビットとなった。このとき、従来の FPGA (4-LUT) と比較して、実装密度を 2.52 倍に引き上げることが分かった。また、LUT のクラスタ化によって、実装効率は従来の 6-LUT に対して約 6% 改善された。そして、マルチコンテキスト化によって、実装面積は最小値を示し、構成データ量は従来の LUT 中で最小の 4-LUT と同程度であることが分かった。

7.2 今後の課題

本稿では、RC に適した RLD を実現する論理ブロックを提案した。しかし、現在の CAE ツールは本提案論理ブロックには対応していない。したがって、今後は提案方式の論理ブロックを有効に活用できる CAE ツールを検討していく必要がある。また、構成データの入れ替え方法、配線構造等の詳細についても検討する予定である。

参考文献

- 1) Brown, S.: FPGA Architectural Research: A Survey, *IEEE Design & Test of Computers*, pp.9-15 (1996).
- 2) 末吉敏則: リコンフィギャラブルロジック, 電子情報通信学会誌, Vol.81, No.11, pp.1100-1106 (1998).
- 3) 末吉敏則: Reconfigurable Computing System の現状と課題—Computer Evolution へ向けて, 信学技報, VLD96-79, CPSY96-91 (1996).

- 4) 末吉敏則, 飯田全広: リコンフィギャラブル・コンピューティング, 情報処理, Vol.40, No.8, pp.777-782 (1999).
- 5) Waugh, T.C.: Field programmable gate array key to reconfigurable array outperforming supercomputers, *Proc. IEEE 1991 Custom Integrated Circuits Conf.* (Cat. No.91CH2994-2), 6.6/1-4, 756 (1991).
- 6) 中島克人, 森 伯郎, 佐藤裕幸, 高橋勝己, 浅見廣愛, 水上雄介, 飯田全広, 新留勝広: FPGA ベース並列マシン RASH, 並列処理シンポジウム JSPP'99, p.222 (1999).
- 7) Rose, J.S., Francis, R. and Lewis, D.: Architecture of Field-Programmable Gate Arrays: The Effect of Logic Block Function on Area Efficiency, *IEEE J. Solid-State Circuits*, Vol.25, No.5, pp.1217-1225 (1992).
- 8) Rose, J.S. and Brown, S.: The Effect of Logic Block Architecture on FPGA Performance, *IEEE J. Solid-State Circuits*, Vol.27, No.3, pp.281-287 (1992).
- 9) Leventis, P.: Using edif2blif Version 1.0 (Draft), (1998).
<http://www.eecg.toronto.edu/~jayar/software/software.html>
- 10) Sentovich, E.M., et al.: SIS:A System for Sequential Circuit Synthesis, Memorandum No.UCB / ERL M92/41 (1992).
<http://www-cad.eecs.berkeley.edu/Software/software.html>
- 11) Marquardt, A., et al.: Using Cluster-Based Logic Blocks and Timing-Driven Packing to Improve FPGA Speed and Density, *ACM/SIGDA FPGA 99* (1999).
<http://www.eecg.toronto.edu/~jayar/software/software.html>
- 12) Yang, S.: Logic Synthesis and Optimization Benchmarks, Version 3.0, Tech. Report, Microelectronics Centre of North Carolina, 1991.
- 13) Tau, E., Chen, D., Eslick, I., Brown, J. and DeHon, A.: A First Generation DPGA implementation, *Proc. 3rd Canadian Workshop on Field-Programmable Devices*, pp.138-143 (1995).
- 14) 柴田裕一郎, 宮崎英倫, 凌 曉萍, 天野英晴: HOSMII: DRAM 混載型 FPGA に基づく仮想ハードウェアシステム, 信学技報, CPSY97-45 (1997).
- 15) Fujii, T., Furuta, K., Motomura, M., Nomura, M., Mizuno, M., Anjo, K., Wakabayashi, K., Hirota, Y., Nakazawa, Y., Ito, H. and Yamashina, M.: A Dynamically Reconfigurable Logic Engine with a Multi-Context/Multi-Mode Unified-Cell Architecture, *IEEE Inter-*

national Solid-State Circuits Conference, WA
21.3 (1999).

(平成 13 年 9 月 21 日受付)

(平成 14 年 3 月 14 日採録)



飯田 全広 (正会員)

1964 年生。1988 年東京電機大学電子工学科卒業。同年三菱電機エンジニアリング(株)入社。オフィスサーバ, DB エンジン等の開発に従事。1995 年同社を退職し,九州工業大学大学院情報工学研究科に入学。1997 年博士前期課程修了。同年,同社に復職。1999 年熊本大学大学院自然科学研究科に入学。現在,可変構造型計算機の研究・開発に従事。



末吉 敏則 (正会員)

1953 年生。1976 年九州大学工学部情報工学科卒業。1978 年同大学院工学研究科情報工学専攻修士課程修了。同年九州大学工学部情報工学科助手。同大学院総合理工学研究科助教授,九州工業大学情報工学部助教授を経て。1997 年より熊本大学工学部数理情報システム工学科教授。工学博士。計算機アーキテクチャ等の研究に従事。