

# 通信処理用 FPGA の CAD システム

室岡孝宏<sup>†</sup> 高原 厚<sup>†</sup>  
筒井章博<sup>†</sup> 宮崎敏明<sup>†</sup>

我々は、高速性と柔軟性が要求される通信処理分野をターゲットとした FPGA と、そのアーキテクチャに着目した専用 CAD システムを開発した。本論文では、専用 CAD システムの構成とその処理アルゴリズムについて述べる。本システムは論理合成後のネットリストを入力とし、テクノロジーマッピング、配置配線、FPGA のプログラムデータの生成という一連の設計環境を提供している。特に、信号の流れに着目した配置処理および、配線構造を意識した高速な配線処理が特徴であり、効率的な回路実現が可能である。通信処理回路を用いて評価を行った結果、我々が以前開発した CAD システムと FPGA の組合せと比較して、短い処理時間で高い配線率の回路を得ることができた。

## A CAD System for Telecommunication FPGA

TAKAHIRO MUROOKA,<sup>†</sup> ATSUSHI TAKAHARA,<sup>†</sup> AKIHIRO TSUTSUI<sup>†</sup>  
and TOSHIKI MIYAZAKI<sup>†</sup>

This paper describes the dedicated CAD system for newly developed telecommunication-oriented FPGA. This CAD system supports the total design environment which is composed of technology-mapping, placement, routing and verification-tools. In the placement and routing algorithms, the properties of our target FPGA architecture, are well considered to achieve the efficient implementation of given circuits. Experimental results indicate that the newly developed FPGA and its dedicated CAD system realize high performance circuits more effectively than our previous FPGA and CAD system do.

### 1. はじめに

現在、Field Programmable Gate Array (FPGA)<sup>1)</sup> が様々な分野で用いられている。通信処理においても多様なプロトコルへの対応を柔軟に行えるシステムへの利用が考えられている。

我々は、プログラマブル通信装置の構成要素として、通信処理用 FPGA PROTEUS<sup>2)</sup> と専用 CAD システム<sup>3)</sup> を開発した。実際の通信処理装置に適用し評価を行った結果、PROTEUS はレジスタトランスファレベル (RTL) からのトップダウン設計では十分な性能を引き出せないことが明らかになった。この主な原因は、自動配線処理時の配線資源の不足であった。

上記結果に基づき、我々は、水平/垂直チャネルの配線トラック数、およびそれらを接続するためのスイッチ数を PROTEUS と比較して約 2 倍に拡張した新たな通信処理用 FPGA として、PROTEUS-Lite<sup>4)</sup> を開

発した。また、配線資源の構成については、FPGA のアーキテクチャと CAD アルゴリズムを同時に評価するシステム<sup>5)</sup> を使用し、自動設計に適したものとした。

FPGA の CAD システム中で、そのアーキテクチャに依存し、アプリケーション回路の性能に影響するのは、配置処理と配線処理である。PROTEUS-Lite 用 CAD システムでは、PROTEUS-Lite の特徴を活かす配置、配線処理アルゴリズムを検討した。配置処理では、アプリケーション回路の信号の流れに着目し、そのトポロジーから初期配置を決定し、FPGA の方向性を考慮したシミュレーテッドアニーリング法で最適化する。また、高速動作実現のために用いられるハードマクロを含んだ回路を効率的に扱う手法も備えた。配線処理では、アプリケーション回路のクリティカルパスを考慮した配線処理の優先順序付けと、PROTEUS-Lite のアーキテクチャを考慮した線分探索手法<sup>6)</sup> を基とする経路探索アルゴリズムを導入することにより、結果の高品質化と処理の高速化を実現した。

本論文では、多量の配線資源を持つ FPGA PROTEUS-Lite の専用 CAD システム (PLCAD) に

<sup>†</sup> NTT 未来ねっと研究所  
NTT Network Innovation Laboratories

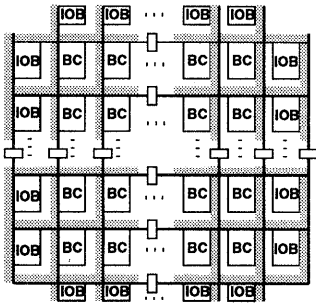


図1 PROTEUS-Liteの基本アーキテクチャ  
Fig. 1 The architecture of PROTEUS-Lite FPGA.

関して、そのシステム概要、および処理アルゴリズムについて述べる。以下では、2章で PROTEUS-Lite の構造的特徴を述べた後、3章で PLCAD の処理の流れと、処理アルゴリズムについて述べる。4章で通信処理回路を用いて PLCAD システムを評価した結果を示す。5章にまとめを示す。

2. PROTEUS-Lite の概要

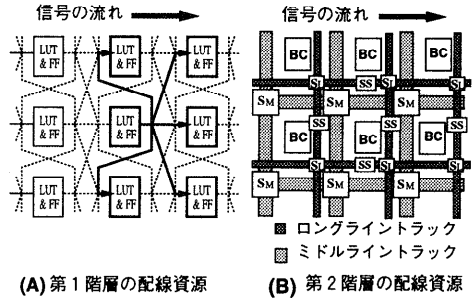
2.1 全体構成

PROTEUS-Lite の基本アーキテクチャを図1に示す。格子状の対称的アーキテクチャ<sup>7)</sup>の SRAM 型 FPGA であり、格子状に配線チャンネルを有している。配線スイッチ類はチャンネルの格子点上に配置されている。論理ブロックである基本セル (BC) は各格子に配置されており、外部との接続に使用する I/O (IOB) も縁辺部の格子に配置されている。BC はチップ水平方向、垂直方向に 28 × 10 個配置されている。IOB は上下の各辺に 28 個、左右の各辺に 10 個配置されている。また、アプリケーション回路の信号の流れは、図中左から右である。

通信処理のアプリケーションでは、通常 8 bit 単位での処理が行われる場合が多い。それに対応するため、PROTEUS-Lite の BC は 4 個の 3 入力 1 出力のルックアップテーブル (LUT)、4 つのフリップフロップ (FF) で構成される。各 LUT の出力には FF が準備されており、その使用/不使用はプログラムにより設定可能である。IOB は、トライステートバッファと FF を有した I/O モジュールから構成される。チップ左右の IOB には 4 個の I/O モジュール、上下の IOB は 1 個の I/O モジュールを含んでいる。

2.2 階層的な配線資源の構成

PROTEUS-Lite の配線資源は 2 階層で構成される。第 1 の階層は LUT、FF および、ローカルラインと呼ぶ隣接する LUT 間を接続する低遅延の配線から構成



(A) 第 1 階層の配線資源 (B) 第 2 階層の配線資源  
図 2 各階層の配線資源の構成  
Fig. 2 Routing resource architecture.

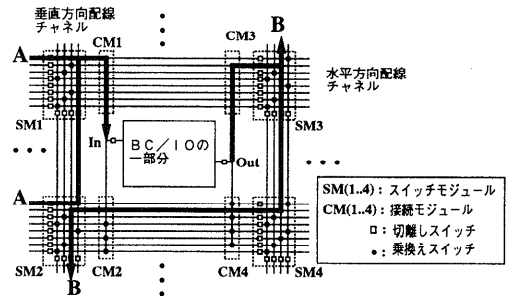


図 3 SM と CM の構造  
Fig. 3 Switch patterns in SM and CM.

される。第 2 の階層は BC の 1 辺と同じ長さのミドルラインと呼ばれる配線と、チップ 1 辺の半分の長さを有するロングラインと呼ばれる配線で構成される。それらは乗換えスイッチ、切離しスイッチを介して接続できる。ミドルラインは自由度のある配線資源を提供し、ロングラインは一定の距離を離れた BC 間の低遅延での接続手段を提供している。

図 2 に各階層の配線資源の構成を示す。図中 (A) は第 1 階層の構成を示している。ローカルラインは上、下、右上、右下、右に隣接した 5 個の LUT に配線資源中で最も小さい遅延を持つ配線資源で接続可能である。この階層の資源を用いることで、通信処理回路に多用される高速なカウンタ/シフタ等を容易に実現することができる。図中 (B) は第 2 階層の構成を示している。ミドルラインのチャンネルは、スイッチモジュール (SM) 内に複数の配線乗換えスイッチと切離しスイッチを持つ。ロングラインのチャンネルは、ミドルラインチャンネルに沿って配置され、チップ中央で切離しスイッチ (SS) により分割可能である。ロングラインのスイッチモジュール (SL) は乗換えスイッチのみを持つ。

図 3 に BC 周辺のみドルライントラック、乗換えスイッチ、切離しスイッチの配置を簡略化して示す。

表 1 BC 周辺の配線資源数

Table 1 The number of routing resources around one BC.

スイッチ類	配線資源の種類				ローカル
	ロング		ミドル		
	水平	垂直	水平	垂直	
425	6	4	32	16	5

PROTEUS-Lite のスイッチ配置は配線の接続性の確認処理を容易にし、配線の経路探索処理を高速にする。BC の入力側 CM が接続するミドルライントラックが確定すれば、接続先側の CM と候補となるミドルライントラックが一意に決定できる構造になっている。たとえば、上下の水平チャンネルのミドルライントラック A が選ばれた場合、チップ上のどの BC へも、CM1 で示される CM 内のスイッチを介して接続できる。また、垂直チャンネルのトラック B が選ばれた場合、CM3 で示される CM のスイッチを介することでチップ上のすべての BC の出力と接続できる。

乗換えスイッチは単一の N-MOS トランジスタで構成される。そのため、配線遅延を増加させるので、低遅延の経路を得るには乗換えスイッチが少ない経路を探索する必要がある。切離しスイッチは同一方向の隣接するミドルライントラックの接続/切離しを行う。また、切離しスイッチは信号を増幅するバッファを有している。

表 1 に BC 周辺の配線資源数を示す。PROTEUS-Lite は 280 個の BC を有している。すべての BC は、表に示す数の配線資源を持つため、その組合せ数は膨大となり、配線処理では効率的な経路探索が必要になる。

### 3. PLCAD

#### 3.1 PLCAD 概要

PROTEUS-Lite の専用 CAD システムである PLCAD の処理の流れを図 4 に示す。PLCAD は既存の論理合成ツールで生成されたネットリストを入力とし、テクノロジーマッピング (PLCOVER)、配置 (PLFIX)、配線 (PLROUTE) 処理を行うことにより、PROTEUS-Lite に対応したネットリストを出力する。また、手設計が可能で視覚的な回路エディタ (PLEDIT) も提供している。設計された論理回路は Verilog-HDL に変換 (UPNF2VL) し、論理シミュレーションを行える。配線処理が終了したネットリストは電気的な正当性の検証 (UPNFCK) の後、FPGA へロードするプログラムデータへ変換 (UPNF2PBF) される。配置、配線が終了したネットリストをハードマクロとして保存し、再利用できる。

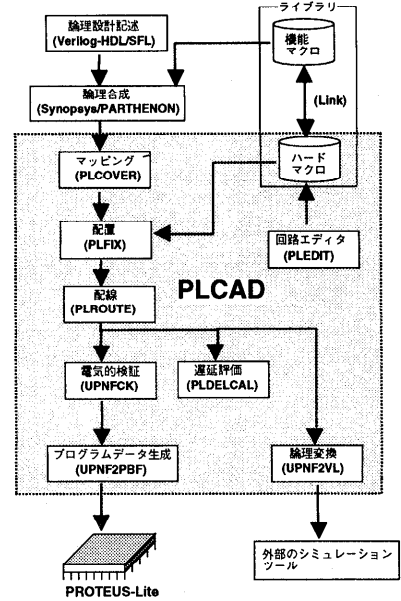


図 4 PLCAD の処理の流れ

Fig. 4 An overview of PLCAD system.

#### 3.2 テクノロジマッピング処理

PLCOVER は、2 入力の AND/OR 等と FF で記述されたアプリケーション回路のネットリストを読み込み、PROTEUS-Lite の LUT に合わせたネットリストを生成する。入力ネットリストでは、論理合成時に使用されたハードマクロは端子のみの実態のないモジュールとして扱われ、他の回路との接続の検証のみ行う。

テクノロジーマッピング処理では、入力ネットリストを論理をノード、配線をエッジとした有向グラフに展開し、リフォーミングとカバーリングの 2 段階で処理する。リフォーミングでは、後行程のカバーリングが効率的に行えるように、グラフに対してノードの複製や、論理の変更によるインバータ、バッファ、固定値を意味するノードの消去処理を行う。カバーリングでは、リフォーミングが終了したグラフに対して、LUT と同一の 3 入力 1 出力のサブグラフによる被覆処理を行う。このとき、できるだけ多くのノードをサブグラフに含むようにする。

#### 3.3 配置処理

PLFIX では、配線の局所集中を避け、配線長が最小となるようにシミュレーテッドアニーリング法 (SA) を用いて配置を決定する。初期配置では、配線資源の構成を考慮してチップ左側を入力、右側を出力として論理回路をレベルソートして配置する。SA での最適化処理では、配線がチップの左から右に向くようにコ

ストの差を調整する。配線の分岐枝のコスト ( $Cost_n$ ) を以下の式で決定し、その総和を最小とすることを目的に SA を行う。

$$Cost_n = Nl_n + \alpha \times Dc_n - \beta \times Lc_n$$

ここで、 $Nl_n$  は配線枝の長さである。 $Dc_n$  は配線の方向による係数であり、構成に準じた方向では  $Dc_n = 0$ 、逆では  $Dc_n = 1$  となる。 $Lc$  はローカルライン接続可能な位置関係であるか否かによる係数であり、可能な場合は  $Lc_n = 1$ 、不可能な場合は  $Lc_n = 0$  となる。また、 $\alpha$ 、 $\beta$  は定数であり、配線資源の構成に順じた配置結果を得るために実験結果を基に決定した。 $\alpha$  は第 2 階層の方向性に順じない配置の場合にコストを大きくするため、PROTEUS-Lite 上の最長配線となる対角線長と同等の値とした。 $\beta$  はローカルライン接続可能な場合に大きくコストを低下させるため  $Nl_n$  の定数倍の値とした。この式では、配線枝がローカルライン接続可能であるか、第 2 階層の構成に適合する場合にコストが低下する。

ハードマクロは物理的な大きさが LUT/BC と異なるために、SA での配置位置の入換え処理で LUT/BC と同一に取り扱うことが難しい。そこで、マクロの配置については、チップの領域を配置するマクロの大きさに応じた格子状の領域に分割し、初期配置でのレベルソートの結果で位置を決定している。

### 3.4 配線処理

配線処理では、PROTEUS-Lite の膨大な配線資源の組合せから最も低遅延の経路を効率良く探索することが重要である。また、アプリケーション回路の性能に影響を与える経路の配線に優先的に配線資源を与え、低遅延の経路を得ることも重要になる。PLCAD の配線ツール (PLROUTE) ではこの 2 点を考慮した配線処理を行う。

図 5 に PLROUTE での処理の流れを示す。配線処理は、処理順位付け処理 (ROP)、ローカルライン配線処理 (LRP)、線分探索を基本とする配線処理 (LSRP)、未配線処理 (GRP) の 4 つの処理からなる。

PLROUTE では効率的な配線経路の探索を行うために、2 階層の配線資源の構成に応じて、探索アルゴリズムが異なる LRP と LSRP の 2 つのアルゴリズムで経路を探索する。制約が大きい第 1 階層の配線資源については、配線枝のローカルラインでの接続の可能性のみを評価して、経路を決定する方法を用いた。自由度の大きな第 2 階層の配線資源には、スイッチモジュール内のスイッチの配置パターンにより配線トラックの選択で経路を決定できる特徴を活かした高速な経路探索手法を用いた。また、搭載する回路の性能

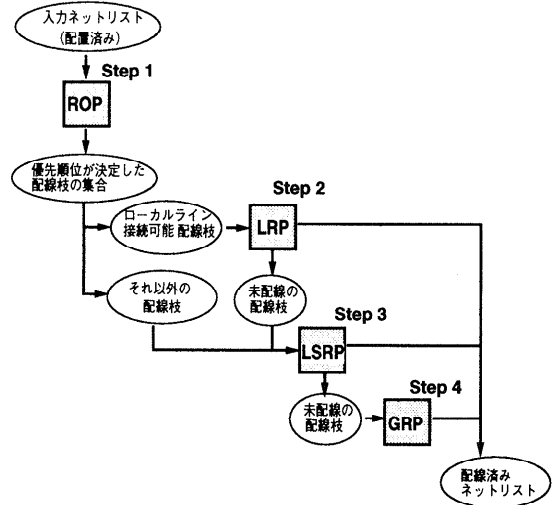


図 5 PLROUTE の配線処理の流れ  
Fig. 5 Procedures used in the PLROUTE.

向上には、配置されたアプリケーション回路の FF 間遅延を考慮した配線枝の処理優先順位付けと、遅延を考慮した分岐処理を行っている。

一般的に使用されている方向性を有しない XILINX<sup>8)</sup> 型の FPGA については、配線処理の高速化手法として、Thakur ら<sup>9)</sup> のスイッチモジュールの構成に着目した整数線形計画法による手法が、最短経路探索の手法としては Alexander ら<sup>10)</sup> のスタイナー木を用いた IDOM が提案されている。しかしながら、これらの経路探索手法は、特定方向の信号の流れを考慮した階層的な配線資源のアーキテクチャは考慮されておらず、PROTEUS-Lite に適用するのは難しい。

#### 3.4.1 処理の優先順位の決定処理

優先順位は配線枝が属する配線経路の遅延値、配線枝の属性と仮配線長を基に決定する。FF 間の遅延値は、配線枝のマンハッタン距離を仮遅延値とし、アプリケーション回路内の FF の端子 (FF の入力/出力端子、制御端子) を仮の外部端子として、外部端子間を結ぶ最長経路上の仮遅延値の総和とした。また、回路性能の向上のために、配線枝の接続先の種類により優先順位を上げる。外部回路とのタイミング制約がある I/O 端子への配線と、回路の動作自体に大きな影響を与える FF の制御端子への配線は優先順位を上げる。我々は、処理の優先順位を以下の式を用いて決定した。 $V$  の値の大きな配線枝ほど優先順位が高い。

$$V = Cd \times Nl \times IOv \times FCv$$

ここで、 $Cd$  はクリティカルパスの遅延値、 $Nl$  は配線枝の長さ、 $IOv$  は I/O との接続の場合の優先度を

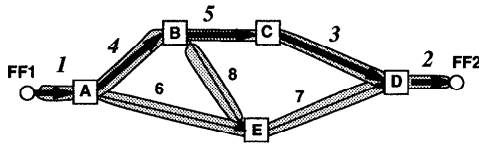


図6 配線処理の優先順位付け  
Fig. 6 Net ordering for routing.

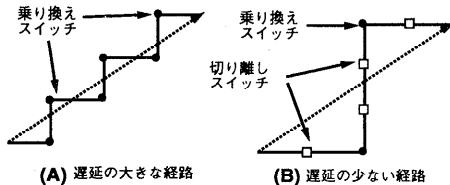


図7 低遅延な配線経路  
Fig. 7 Effective route on the PROTEUS-Lite.

上げる定数,  $FCv$  は FF の制御端子接続の優先度を上げる定数である。  $Cd$  はクリティカルパス探索から決定され,  $Nl$  は各配線に依存して決定される。  $IOv$ ,  $FCv$  は経験的に決定する。 PLROUTE では, 周辺回路とのタイミング制約を受ける I/O 端子への配線の優先度を上げるために  $IOv > FCv$  とした。 この式を用いて, アプリケーション回路内のすべての配線枝に対して  $V$  を決定し, 降順にソートすることで, 優先順序を決定する。

図6に PLROUTE の配線の優先順位の決定例を示す。 図中, FF1, FF2 は FF の入出力端子を示している。 A から E は論理要素, 数字は処理順序である。 ここで, 端子 FF1 を始点として論理要素 A, B, C, D を経て端子 FF2 につながる一連の経路が, この回路のクリティカルパスである。 矢印は配線枝を示している。 FF 間を結ぶ最も長い経路に属する配線枝で, FF の入出力に接続するものが最も優先順位を高く設定され, 他の配線枝は, 仮配線長に応じて順位が設定される。

### 3.4.2 線分探索を基本とする配線処理

LRP で経路を決定できなかった配線枝を含めて LSRP で経路探索を行う。 PROTEUS-Lite では, 同一方向の配線要素や異なる種類の配線資源の接続に乗換えスイッチを使用する。 乗換えスイッチはパッシブな素子であるため, 複数段連続して使用すると経路遅延を大きくしてしまう。 そのため, 配線の折れ曲がりを抑制する必要がある。

図7中 (A) は複数回折れ曲がった配線経路を, (B) は折れ曲がりが2カ所の配線経路を示している。 配線経路の折れ曲がりには, 乗換えスイッチが使用されて

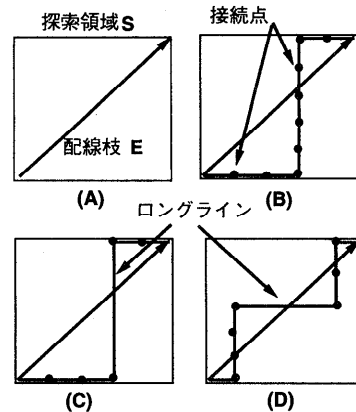


図8 LSRP の経路探索  
Fig. 8 Route search by the LSRP.

いる。遅延的には, 複数の乗換えスイッチを使用した (A) より, 2カ所のみ (B) の方が有利になる。 LSRP では, 配線経路の折れ曲がりを可能な限り抑制し, (B) の経路を得ることを試みる。

折れ曲がりを抑制した配線経路を得るために, LSRP では線分探索法を基本として, 第2階層の配線資源の構成に合わせて改良したアルゴリズムを使用した。従来の線分探索法はネットの始点/終点の双方から上下左右の4方向に最長線分を探索し経路を決定している。 LSRP では, 折れ曲がりが少ない配線経路を得るために, 線分探索の方向を, ネットの始点と終点を対角に有する長方形の辺に則した2方向のみに限定している。これにより, 迂回した配線経路を得ることはできないが, 探索領域を限定できるために, 第2階層の豊富な配線資源から高速に経路を探索することが可能になる。最長線分は, スイッチモジュールごとに, 乗換えスイッチ, 切離しスイッチの使用状態を変数テーブルに登録し, そのテーブルを検索することで得られる。我々はこの経路探索を Longest line Search Heuristics (LSH) と呼ぶ。 LSH は最悪でも, 領域内のすべてのスイッチモジュールを探索するだけで経路を発見することができ, ダイクストラ法と比較して計算量はきわめて少ない。

ロングラインは, 探索する領域の1辺が長く, ミドルラインの接続と比較して遅延的に有利な場合に使用する。図8(A)は, LSRP で経路探索される配線枝 E と, 探索領域 S を示している。ミドルラインのみの経路探索結果の例を (B) に示す。経路は複数のミドルラインで構成され, 接続点が多く生じる。接続点では, 切離しスイッチや乗換えスイッチを介してミドルラインを接続する。接続数が一定数を超えると, ロングラ

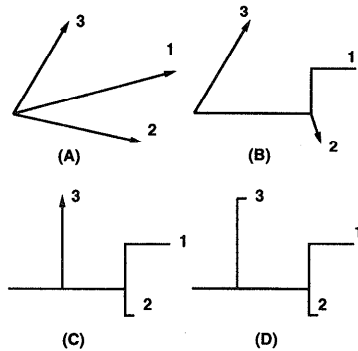


図9 配線分岐の処理

Fig. 9 Fanout handling in the PLROUTE.

インを使用する方が遅延的に有利になる。領域Sの垂直方向の長さがロングラインでの配線に有利になった場合に(C)の経路を選択する。また、水平方向が長くなった場合は(D)の経路を選択する。

### 3.5 配線の分岐点の決定

通常の配線は配線枝を有するので、配線経路上の分岐点の決定処理が必要である。分岐点の決定は、配線遅延の増加を抑制するために、最長の配線枝の配線経路の総延長を、他の分岐枝の経路が超えないようにする。PLROUTEでは、既配線の経路上で配線の始点からのマンハッタン距離が最も短くなる点を分岐点とした。候補となる分岐点が複数存在した場合は、配線の始点に最も近い分岐点を選択する。

図9に分岐処理の例を示す。配線の分岐点の決定は、配線処理の優先順序付けで決定された順序で行う。図9(A)が与えられた配線とする。最初に最も順位の高い配線枝1の経路を決定する。次に、次順位の配線枝2の始点を(A)で決定した経路上で配線の始点からのマンハッタン距離が最も短くなる点に移動し(B)、経路を決定する。同様の処理を次の順位の配線枝3に対しても行う。分岐点の集中を避けるため、配線要素上の乗換えスイッチ数を基に、1つの配線要素から分岐可能な分岐数の上限を定めた。分岐数の上限は分岐点の配線要素上の乗換えスイッチ数と、接続先の配線要素の配線容量に依存する。配線要素の接続を回路シミュレーションで評価した結果、分岐数は、ローカルラインで2、ミドルラインで2、ロングラインで4とした。ロングライン上には多数の乗換えスイッチが存在する。ロングライン自体の配線負荷は大きく、分岐した配線自体も新たな配線負荷となる。そのため、分岐数が多くなると、それに応じて遅延が大きくなる。そこで、ロングライン上の分岐可能な数は乗換えスイッチ数より少なくしている。

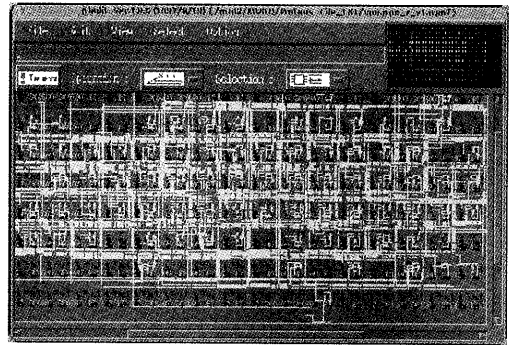


図10 PLEDITの画面

Fig. 10 A snapshot of the PLEDIT.

ローカルラインで接続可能なLUTの位置関係は、第1階層の構成から一意に決定できるため、LRPではテーブル検索での接続可能性の確認で経路を決定している。GRPでは、ここまでの処理で残った配線資源に等価的に配線として使用可能なLUTを加えて、ダイクストラ法で経路を探索している。GRPは完全配線を実現することを目的にしているため、得られた経路は最適でない可能性がある。

### 3.6 手動設計ツールと検証ツール

PROTEUS-Lite内の配線資源/BC/IOBを図的に表現し、人手で論理、配線の編集が可能な回路エディタ(PLEDIT)を準備した。PLEDITの編集集中のイメージを図10に示す。PLEDITは、論理回路の手作業による設計、配置処理、配線処理後の回路編集等をサポートする。また、ハードマクロと呼ばれる論理部品を作成できる。PLCADでは、設計した論理回路をPROTEUS-Liteに搭載する前に論理検証、遅延検証、電気的な接続検証を行うためのツールを準備した。論理検証には、市販シミュレータへのインタフェースとして、PLCAD内で使用するUPNFと呼ばれる形式のネットリストをVerilog-HDLに変換するツール(UPNF2VL)を準備した。論理回路の性能を見積もるために、搭載した回路の性能を静的に遅延評価するPLDELICALを準備した。PLDELICALでは、切離しスイッチにより分離される領域内での配線の組合せごとに実チップでの測定値を基にした遅延評価関数を用意し、それらによる計算結果を積算する方法<sup>11)</sup>で遅延値を算出している。そのため、配線経路を分布乗数回路モデルで評価するより、高精度かつ高速な評価が可能である。PLEDITでの手動設計では、配線のショートや未配線等の設計誤りが起こりうる。これらの設計誤りを検出するツール(UPNFCK)を準備した。これらの検証が終了した後にPROTEUS-Liteに論理を

表2 サンプル回路と、そのテクノロジマッピング結果  
Table 2 Sample circuits and their technology-mapping results.

Circuit	Before		After		Time (sec)	Circuit function
	# of gates	# of FFs	# of LUTs	# of FFs		
fcoun	62	13	45	13	0.1	13 bit counter
sc_mng	103	11	65	11	0.1	Small controller
hec	111	48	85	48	0.1	CRC generator
cellgen	192	54	136	54	0.1	ATM cell generator
stm1gen	320	93	228	93	2.0	STM frame generator
stm1t	608	118	378	118	6.0	STM frame terminator
sigg	899	209	611	209	13.0	Large controller

搭載するためのプログラムデータを UPNF2PBF を使用して生成する。

#### 4. 評価

実際の通信処理に使用される回路をサンプルとし、PLCAD の自動設計ツールを使用して PROTEUS-Lite への回路搭載実験を実施した。Verilog-HDL の RTL 記述から Synopsys 社のツールで論理合成を行い、PLCAD でテクノロジマッピング、配置、配線を行った。PLCAD は C++ で開発し、実験には SPARC ULTRA-II (1 CPU, 200 Mhz, 128 MB) を使用した。

表2 にサンプル回路の PLCOVER でのテクノロジマッピング結果を示す。入力ネットリストは2入力の AND, OR 等とインバータ, バッファの基本ゲートで構成されたものである。処理の結果, 平均で 1.5 個程度の基本ゲートを1つの LUT に搭載している。PLCOVER では FF は最適化しないので, テクノロジマッピング前後で変化しない。

表3 に PLFIX による配置処理の結果を示す。冷却単位を現温度の 0.1% とした。SA の冷却処理は終了まで, 約 8000 回繰り返された。配置結果では, 平均で 2.5 個の LUT を1つの BC に格納している。

表4 は PLROUTE の各段階で配線処理された配線枝数と, 経路探索に要した時間を示している。処理時間は, 経路探索に要した時間であり, ファイルの入出力やデータのメモリへの展開に要する時間は含まない。結果では, すべてのサンプルで約 30% の配線枝が LRP で処理されている。LRP はローカルライン接続可能な配線枝のみを処理する。配置処理では, 遅延の低いローカルラインで接続可能に配置することをコストの評価式で試みている。LRP での高い配線率は, この評価式が配線資源の第1階層の構成に適合していることを示している。回路 sigg を除くすべてのサンプルで, LSRP までの処理で配線が終了している。回路 sigg は制御論理の部分が大きな回路である。制御

表3 配置処理の結果  
Table 3 Placement results by the PLFIX.

Circuit	Before		After	Time (sec)
	# of LUTs	# of FFs		
fcoun	45	13	18	48
sc_mng	65	11	32	50
hec	85	48	35	50
cellgen	136	54	51	54
stm1gen	228	93	99	59
stm1t	378	118	150	68
sigg	611	209	220	79

表4 配線処理の各段階での配線数  
Table 4 The number of routed nets handled by each procedure.

Circuit	Procedures			Process Time (sec)	
	LRP	LSRP	GRP	Rtime	Time
fcoun	46	76	0	0.4	16.7
sc_mng	54	115	0	0.6	17.5
hec	73	128	0	0.8	18.3
cellgen	113	221	0	1.5	23.3
stm1gen	179	433	0	2.5	33.8
stm1t	286	730	0	6.1	54.9
sigg	469	1108	5	37.0	124.8

の論理は通信の信号と異なり方向性を有しない。そのため, 自由度の高い配線経路を探索する必要が生じ, 第2階層の配線資源を多量に使用している。その結果, GRP での探索処理が実行された。

表中 Rtime は経路探索処理に要した時間を示している。GRP が実行された回路 sigg を除いてすべて 10 秒以内に処理が終了している。処理速度については, 基本的な経路探索アルゴリズムであるダイクストラ法と比較を行った結果, PLROUTE の配線処理は 350 倍高速であった<sup>12)</sup>。

図11 は PROTEUS と PROTEUS-Lite の自動配線ツールでの配線率を比較したものである。論理の搭載は, それぞれの専用の CAD システム (PROCAD)<sup>3)</sup>, PLCAD で行った。図11 の結果から, PROTEUS では LUT の使用率が約 20% を超えると自動配線が困

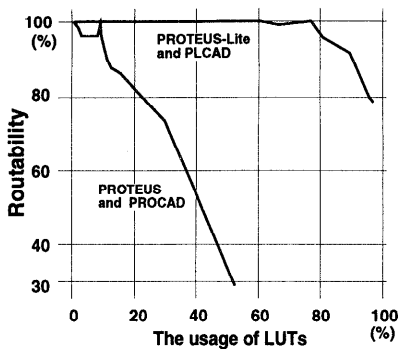


図 11 配線率の比較

Fig. 11 Routability comparison.

難になるが、PROTEUS-Liteでは、LUTの使用率で80%までは100%の配線が可能である。

PROTEUSでは、配線資源が少ないために1つのBCで使用するLUT数を制約してLUT使用密度の低い配置を行う必要があり、実質的な搭載可能な論理規模は小さいものになっている。一方、PROTEUS-Liteでは配線資源量を増加し、LUT使用密度の高い配置が可能になっている。そのため、高いLUT使用率での完全配線が実現されている。一般には、配線資源量の増加は配線処理の計算量を増加させる。そこで、配置処理で配線資源の方向性を考慮した配置を行うことにより、後行程の配線資源構造を意識した配線処理(LRP, LSRP)での高い配線率が実現されている。

PLCADシステムとPROTEUS-Liteの総合評価を行うために、CADシステムを含めた市販FPGAとの比較を行った。我々は、表2のサンプル回路をXILINX社のCADシステム(XACT)を使用して、PROTEUS-Liteと同一規模で製造プロセスも同一(0.5 $\mu$ m)であるXC4010Eに搭載し、CADの処理時間と生成された回路のクリティカルパス遅延を比較した。

表5に結果を示す。XC4010EのCLBは2つの3入力LUT(3.2gate)、1つの4入力LUT(8gate)とFF(10gate)で構成される。ゲート規模としてはFFまで含めて約25ゲート、FFを外すと約15ゲートになる。一方、PROTEUS-LiteはBC内に4個の3入力LUTと4個のFFを有している。単純に積算するとBCは約53ゲートとなるが、FFを除くと約13ゲートとなり、CLBとほぼ同等の論理実現能力があると考えられる。また、回路のCADでの搭載結果のクリティカルパス遅延は、FPGA自体の物理的な特性よりCADでの搭載処理に強く依存している。そのため、

表 5 市販のFPGAとの比較

Table 5 Comparison between PROTEUS-Lite and a commercial FPGA.

Circuit	PLCAD PROTEUS-Lite			XACT XC4010E-5		
	# of BCs	Time (sec)	delay (nsec)	# of CLBs	Time (sec)	delay (nsec)
fcount	18	80.8	34.0	32	44.0	35.8
sc_mng	32	87.6	49.2	38	47.0	40.1
hec	35	92.4	24.5	59	50.0	22.6
cellgen	51	115.4	61.5	82	76.0	55.1
stm1gen	99	167.8	70.1	150	125.0	80.3
stm1t	150	238.9	57.1	209	178.0	48.3
sigg	220	377.8	114.3	306	304.0	81.4

この評価ではFPGA単体の性能ではなくCADシステムを含めた論理実現の結果として比較している。

配置処理後のBC数とCLB数を比較すると、どの回路に対してもBC数がCLB数より少ない。PROTEUS-LiteはXC-4010Eと比べて、平均0.68倍の論理ブロックの使用で同一回路を搭載できる。これは、通信処理回路の特徴を考慮して設計したBCの構造<sup>2)</sup>が、実際の通信処理回路に適合していること示している。

搭載した論理回路の性能では、PLCADの結果はXACTの結果と比較して同等である。処理時間では、PLCADはXACTより多くの時間を要している。これは、PLCADが複数の行程を独立したツールで行っているため、ファイルの入出力やデータのメモリへの展開処理がXACTと比較して冗長に行われるためである。

生成された回路性能では、XC4010Eと同程度になっている。PLCADでは、ハードマクロを使用した設計をサポートしている。PROTEUS-Liteの構造を活かした通信処理ハードマクロを準備し、それを使用した設計を行うことで、より効率的に高速な回路をインプリメントすることが可能になると考えられる。また、配線処理と同様にアプリケーション回路のクリティカルパスと、信号の流れの方向性を考慮した配置処理を行うことを試みている。これによる、アプリケーション論理のさらなる高速化が期待できる。

## 5. まとめ

豊富な配線資源を階層的に持つ通信処理用FPGAであるPROTEUS-Liteの専用CADシステム(PLCAD)について述べた。PLCADはRTL記述からのトップダウン設計環境を提供している。PLCADの配置処理と配線処理ではPROTEUS-Liteのアーキテクチャを考慮したアルゴリズムで処理を行う。こ



れにより、アプリケーション回路を高性能に実現することが可能となる。今後は、通信用ハードマクロの充実と配置処理の改善をはかり、PLCADを含めてPROTEUS-Liteを実際の通信処理装置に適用し、その性能と適用性を確認していく。

### 参 考 文 献

- 1) Brown, S.D., Francis, R.J., Rose, J. and Vranesic, Z.G.: *Field-Programmable Gate Arrays*, Kluwer Academic publishers (1992).
- 2) Ohta, N., Nakada, H., Yamada, K., Tsutsui, A. and Miyazaki, T.: PROTEUS: Programmable Hardware for Telecommunication Systems, *Proc. IEEE International Conference on Computer Design (ICCD)*, pp.178-183 (1994).
- 3) 筒井章博, 宮崎敏明, 中田 広, 山田一久, 太田直久: 通信処理用FPGA専用CADシステム, 95回路とシステム軽井沢ワークショップ, pp.73-78 (1995).
- 4) Miyazaki, T., Takahara, A., Katayama, M., Murooka, T., Ichimori, T., Fukami, K., Tsutsui, A. and Hayashi, K.: CAD-oriented FPGA and Dedicated CAD System for Telecommunications, *Proc. Field-Programmable Logic (FPL)* (1997).
- 5) Miyazaki, T., Tsutsui, A., Ishii, K. and Ohta, N.: Fact: Co-evaluation environment for FPGA architecture and CAD system, *Proc. Field-Programmable Logic (FPL)*, pp.34-43 (1996).
- 6) Hightower, D.: A Solution to Line-Routing Problems on the Continuous Plane, *Proc. 6th Design Automation Workshop*, pp.1-24 (1969).
- 7) AT&T Microelectronics: *Optimized reconfigurable cell array (ORCA) series Field-Programmable Gate Arrays Advance Data Sheet*, AT&T Microelectronics (1993).
- 8) XILINX: XC4000 シリーズ フィールド・プログラマブルゲートアレイ, ザイリンクス (1998).
- 9) Thakur, S., Wong, D.F. and Muthukrishnan, S.: Algorithms for FPGA Switch Module Routing, *Proc. ACM/SIGDA Workshop on FPGAs* (1994).
- 10) Alexander, M.J. and Robins, G.: Optimized reconfigurable cell array (ORCA) series Field-Programmable Gate Arrays, *Proc. ACM IEEE DAC*, pp.562-567 (1995).
- 11) Katayama, M., Takahara, A., Miyazaki, T. and Fukami, K.: Delay Calculation Method for SRAM-based FPGAs, *IEICE Trans. Fundamentals*, Vol.E80-A, No.6, pp.60-65 (1997).
- 12) 室岡孝宏, 高原 厚, 宮崎敏明: FPGAの配線構

造に着目した高速配線処理, 信学技報, VLD98-49 (1998).

(平成10年9月18日受付)

(平成11年2月8日採録)



室岡 孝宏

1989年佐賀大学工学部物理学科卒業。同年日本電信電話(株)入社。同社LSI研究所にて、LSI上位設計手法の研究に従事。1994年より、通信用FPGAおよび専用CAD, さらにそれらを用いたプログラマブル通信装置の研究開発に従事。現在、同社未来ねっと研究所研究主任。電子情報通信学会, IEEE各会員。



高原 厚 (正会員)

1983年東京工業大学工学部情報工学科卒業。1985年同大学院修士課程修了。1988年同大学院博士課程修了。同年日本電信電話(株)入社。同社LSI研究所にて、LSI論理設計システムの研究開発に従事。1994年より、通信用FPGAおよび専用CAD, 通信装置の形式的仕様検証手法の研究開発に従事。現在、同社未来ねっと研究所主任研究員。工学博士。電子情報通信学会, IEEE, ACM各会員。



筒井 章博

1988年神戸大学工学部システム工学科卒業。1990年同大学院修士課程修了。同年日本電信電話(株)入社。通信処理用VLSI, FPGAアーキテクチャおよび専用CADシステム, その応用であるプログラマブル通信処理システムと通信プロトコル処理技術の研究開発に従事。現在、同社未来ねっと研究所研究主任。電子情報通信学会, IEEE, ACM各会員。



宮崎 敏明（正会員）

1981 年電気通信大学応用電子工学科卒業。1983 年同大学院修士課程修了。同年、日本電信電話公社（現 NTT）入社。厚木電気通信研究所にて、DSP 用 CAD, LSI 上位合成, CAD フレームワークの研究および論理設計用 CAD の実用化に従事。1993 年より、通信用 FPGA および専用 CAD, さらにそれらを応用したプログラマブル通信装置の研究開発に従事。現在、同社未来ねっと研究所主幹研究員。工学博士（東工大）。電子情報通信学会, IEEE 各会員。

---