

応用領域に特化した小面積再構成可能 HW のカスタマイズ方式

中谷 正吾[†] 梶原 信樹[†] 粟島 亨[†]

[†] 日本電気株式会社 〒211-8666 川崎市中原区下沼部 1753

E-mail: [†] s-nakaya@da.jp.nec.com,

あらまし 応用に応じて柔軟性を絞った領域特定再構成可能HWは、小面積なプログラマブルHWとして期待される。本研究では、スイッチの均一削減によって、ある程度の柔軟性を確保しつつ、スイッチを大幅に削減する試行を行った。削減スイッチ探索では、定量的な配線性指標を導入し削減対象を効率的に絞る方法を開発した。アプリケーションとして無線通信で使われる符号生成器を取り上げ、汎用再構成可能HWに比べて 1/3 の面積の領域特定再構成可能HWを生成した。

キーワード 再構成可能, 領域特定, スイッチ削減

Customizing of Domain-Specific and Compact Reconfigurable HW

Shogo NAKAYA[†] Nobuki KAJIHARA[†] and Toru AWASHIMA[†]

[†] NEC Corporation 1753, Shimonumabe, Nakahara-ku, Kawasaki, 211-8666 Japan

E-mail: [†] s-nakaya@da.jp.nec.com,

Abstract Domain-specific reconfigurable HWs, which have limited flexibility depending on applications, are expected as small programmable HWs. In this research, we tried a drastic switch reduction maintaining some flexibility, by using a uniform reduction process. In the search for switches to be reduced, we introduced a quantitative index of routability and developed a method for efficiently restricting the switches. Using code generators for wireless communication as applications, the domain-specific reconfigurable core generated is one third as small as general-purpose reconfigurable core.

Keyword Reconfigurable, Domain-specific, Switch reduction

1. はじめに

FPGA に代表される汎用再構成可能HW(ハードウェア)は、非常に多様な機能を実現できる高度な柔軟性を持つ反面、面積、遅延、消費電力などの点で専用回路に比べて大きく劣っているという問題がある。その対策の一つとして、応用領域に応じて柔軟性を絞った領域特定再構成可能HWが検討されている[3][4]。

Betzらは、多数のアプリケーションを配置配線しながらFPGAのパラメータを最適化した[1][2]。彼らの目的は汎用再構成可能HWの最適化であったが、アプリケーションを限定すればそのまま領域特定再構成可能HWの生成につながる。彼らは、FPGAアーキテクチャを簡単なパラメータで記述し、そのパラメータ(主に配線数/チャンネル)を最適化する方法を取った。これは、計算は簡単になるが、アーキテクチャの細かい調整ができないため高度な最適化を行えない問題がある。

Hollandらは、明確に領域特定再構成可能HWの生成を目的とし、応用領域に応じてスイッチを削減するアルゴリズムを考案した[3]。しかしながら、彼らは配置配線が不要なPALやPLAを対象としている。

Danilinらは、FPGAのようなセルアレイタイプの再構成可能HWでの領域特化を述べている[4]。しかし、応用に応じてシステムティックにルーティングリソースを最適化する方法については具体的には示されていない。彼らの評価からは、Betzら[1][2]と同じ程度の最適化と理解される。

本研究では、柔軟性が高く広く普及しているセルアレイタイプ再構成可能HWを対象として、システムティックで効率的なルーティングリソース最適化法を提案する。PALやPLAと異なり、セルアレイタイプ再構成可能HWでは配置配線が必要なため、従来、計算が簡単なパラメータフィッティングが行われてきた。しかし、それでは簡易的な最適化しかできず、大幅な面積縮小は期待できない。我々は、応用に応じて個々のスイッチをシステムティックに削減することで、大きな面積縮小を達成する。この際、実用的な計算時間でこれを遂行するため、配線性に関する定量的指標を使い、削減対象スイッチを効率的に絞り込む。本論文では、その詳細と評価例について報告する。

2. 領域特定再構成可能HWの生成

再構成可能HWで最も大きな面積を占めるのが、プログラマブルな接続網を構成するスイッチ、配線、コンフィギュレーションメモリなどである。本研究は、これらルーティングリソースをアプリケーションに応じて削減することで面積積小を図る。図1にその原理を示す。

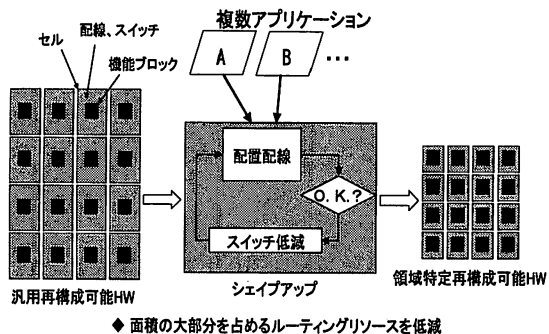


図1 領域特定再構成可能HWの生成原理

再構成可能HWは、同じセルのアレイで構成される。各セルは、種々の演算機能をプログラムできる機能ブロックと、それら之间をプログラマブルに結合するルーティングリソース(配線やスイッチ)を含んでいる。汎用再構成可能HWは、豊富なルーティングリソースを持ち、極めて多様な回路をマッピングすることができるものである。

領域特定再構成可能HWを生成するには、まず最初に汎用再構成可能HWを定義する。そして、想定するアプリケーション(図のA, B, ...)が配置配線できる範囲でルーティングリソースを削減する。アプリケーションが配置配線できる範囲で最小のルーティングリソースを持ったものが求める領域特定再構成可能HWである。

ルーティングリソースの削減の仕方は種々の方法があり得る。配線を削減するもの、スイッチを削減するもの、そして配線数やスイッチ数などのパラメータ調整を行うものなどである。このうち、本研究では個々のスイッチを削減していく方式を採用する。それは最もきめ細かいリソース調整ができるからである。配線削減の場合、その配線につながる複数のスイッチを一度に削減することになり削減単位が大きくなってしまふ。個々のスイッチを削減していく場合、使われなくなった配線は自動的に削減されることになる。Betzら[1][2]は、チャンネルあたりの配線数をパラメータにし、それを調整する方式を取った。この方式は、処理が簡単になる代わりに概略的なリソース調整しかできない

難点がある。個別スイッチ削減法は最もきめ細かいリソース調整であり、他の方法を包含しているため、小面積化には最も優れていると考えられる。

3. ツール

再構成可能HWにアプリケーションを配置配線するには、その再構成可能HWに対応した配置配線ツールが必要である。本研究では、当社のアーキテクチャ追従配置配線ツールを使用している。これは、任意にアーキテクチャを定義した再構成可能アレイに対して配置配線を行うことができる特長を持つ。

領域特定再構成可能HW生成の出発点となる汎用再構成可能HWは、目的に応じて選択する。1ビット演算主体のものや、多ビット演算主体のもの、あるいは機能ブロックの種類や配線構造など、目的に応じて汎用再構成可能HWを適宜変えるほうが効率の良い領域特定再構成可能HWが出来る。そのようなアーキテクチャ変更にもアーキテクチャ追従配置配線ツールは容易に対応することができる。また、領域特定再構成可能HW生成過程で順次スイッチを削減していくが、このスイッチ削減アーキテクチャに対する配置配線もアーキテクチャ追従配置配線ツールによって行うことができる。このようにアーキテクチャ追従配置配線ツールは本研究のキーツールである。

アーキテクチャ追従配置配線ツールでは、アーキテクチャの詳細を記述した定義ファイルを使用する。実際には、直感的で分かりやすい図面でアーキテクチャを描画すると定義ファイルを生成することができる。領域特定再構成可能HW生成の出発点となる汎用再構成可能HWはこの方法であらかじめ規定する。その後のスイッチ削減アーキテクチャは、ソフトウェアで自動的に生成される。

4. アルゴリズム

4.1 均一スイッチ削減

領域特定再構成可能HWは応用に応じて柔軟性を絞るものであるが、あまり絞りすぎると専用回路と変わらなくなる。適度な柔軟性を保持し、ある程度の範囲で想定外の回路もマッピングできるようなものが求められる。柔軟性を具体的に定義するのは困難であるものの、次に説明するスイッチの均一削減によって、残留リソースにある程度余裕を持たせることができる。

図2(1)は、汎用再構成可能HWのセルアレイを模式的に描いたものである。太線のような配線経路がマッピングされているとし、黒丸は使用スイッチ、白丸は未使用スイッチである。ここで、未使用スイッチをすべて削減すると図2(2)のようになる。配線経路がまったく通っていないところは点線のようにセルは完全に削除され、配線経路上のセルはスイッチの使用の仕方

応じてそれぞれスイッチが削除される。このように、配線経路を型抜きしたような削減を行うと、その配線経路以外の経路はマッピングできなくなり柔軟性がなくなってしまう。出来上がったセルアレイは、セルごとにスイッチ削減の仕方が異なるため、不均一なアレイとなる。

他方、本研究で使用する方法は図2(3)に示す均一スイッチ削減である。均一削減は、すべてのセルから同じスイッチを削除するもので、スイッチ削減後も同じセルが並んだ均一セルアレイのままとなる。たとえば未使用スイッチを削減する場合、すべてのセルで使われていないスイッチを削除する。この場合、各セルにはいくつかのスイッチが残っているため、最初の配線経路以外の経路もマッピング可能である。このような均一スイッチ削減によって、アプリケーションに応じたスイッチ削減はするものの、ある程度の柔軟性は残されるようになる。残留する柔軟性の定量化は今後解決すべき課題である。

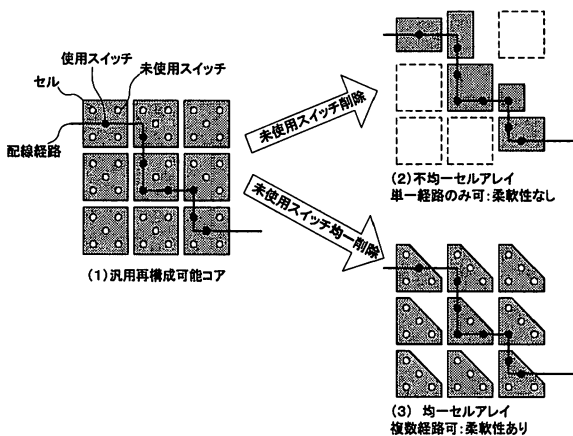


図2 スイッチの均一削減

4.2 スイッチ削減フロー

図3にスイッチ削減フローチャートを示す。まず出発点となる汎用再構成可能HWの定義ファイルを用意する(10)。汎用再構成可能HWは、すべてのアプリケーションがマッピングできる十分なリソースを持っていることが必要である。また、主体となる処理のビット幅など、主要な特徴を反映しているのが望ましい。

つぎに、それを取り込んだ配置配線ツールを生成し、アプリケーションを汎用再構成可能HWに配置配線する(20)。つぎに、配置配線した結果、未使用のスイッチをすべて均一削除する(40)。ここではすべての未使用スイッチを一度に削除するため、一度に多数のスイッチが減少する。また、アプリケーション回路が複数

ある場合、各々のアプリケーションを配置配線し、どの配置配線結果でも使われていないスイッチを均一削減する。いわば、各アプリケーションの配置配線結果の論理和をとったものにおいて、未使用スイッチの均一削減をすることになる。ここまでが初期段階である。

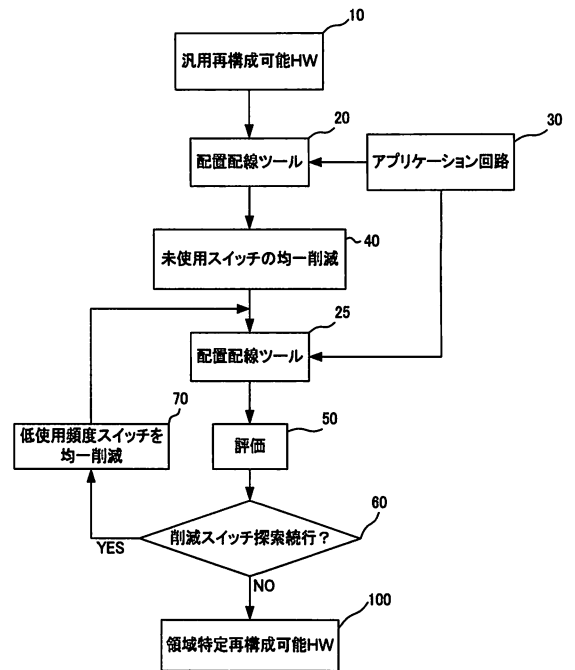


図3 スイッチ削減フローチャート

次に、未使用スイッチ削減アーキテクチャに対してアプリケーションを配置配線し(25)、評価する(50)。評価として最も簡単なのはすべてのアプリケーションが配置配線できたか否かである。そして配置配線できれば削減スイッチ探索を続行する(60, YES)。

次に、アプリケーションを配置配線した結果、S個以下のセルでのみ使われているスイッチを均一削減する。複数アプリケーションの場合、各アプリケーションの配置配線結果の合計を取る。例えば、あるスイッチが、アプリケーションAでは1個のセル、アプリケーションBで0個のセルで使用されていた場合、そのスイッチの使用個数は $1+0=1$ とする。Sの典型例は1である。一般に、S個以下のセルでのみ使われているスイッチは複数種類(M)あり、そのうち1つのスイッチを削除する仕方はM通りある。図3手続き70ではM通りの削除の仕方すべてに対するアーキテクチャを生成し、次にその各々のアーキテクチャに対してアプリケーションを配置配線する(25)。すなわち、手続き70では、M個の低使用頻度スイッチのうちいず

れか一つを削減した M 通りのアーキテクチャが生成されることになる。次の手続き 50 では、M 個のアーキテクチャ各々にアプリケーションを配置配線した結果を評価する。そして配置配線可能であったアーキテクチャのみを更なるスイッチ削減手続き 70 に回し、それ以外のアーキテクチャは棄却する。

以降同様に 70→25→50→60 のループをまわす。一回ごとにスイッチが一つずつ減少していき、あるところまで行くとそれ以上スイッチを削減できなくなる。すなわち、すべてのアーキテクチャで配置配線不能になり、処理は終了する。そして、配置配線可能なアーキテクチャの中で最少スイッチアーキテクチャが求める領域特定再構成可能 HW である。手続き 25 以降のこのループを漸次削除段階と呼ぶ。

4.3 アーキテクチャ絞込み

漸次削除段階では、最初のうち、ループを回ると急速にアーキテクチャ数が増える。例えばあるアーキテクチャにアプリケーションを配置配線した結果、低使用頻度スイッチが N 個あったとする。いずれか一つのスイッチを削除する仕方は N 通りあるので、図 3 の手続き 70 では N 個のスイッチ削減アーキテクチャを生成する。そして、その各々のアーキテクチャに対して、図 3 の手続き 25 でアプリケーションを配置配線し、評価(50)する。評価の結果が良かったアーキテクチャはさらなる低使用頻度スイッチ削減 70 にまわされるが、そこで各アーキテクチャが低使用頻度スイッチを M 個含んでいたとすると、アーキテクチャ数は M 倍に増えることになる。最初のうちは再構成可能 HW の柔軟性が過剰なため多少スイッチを削減しても評価はあまり悪くならない。すなわち生成されたスイッチ削減アーキテクチャは、ほとんど棄却されずに残る。したがって、この段階で最初の 1 つのアーキテクチャは、ほぼ $N * M$ 倍の個数に増えることになる。このようにして、図 3 の手続きを 70→25→50→60 と回ると、アーキテクチャ数は爆発的に増加していく。何らかの方法でアーキテクチャ数を抑制しないと処理時間がかかり過ぎて実行不可能になる。

本研究ではアーキテクチャ数を抑えるために、評価に工夫をしている。4.2 で説明した最も簡単な方法では、評価はすべてのアプリケーションが配置配線できたか否かであった。しかしこれでは評価値は 2 値になってしまい、たとえば配線性がどのくらい悪いかなどの有用な情報は見ていないことになる。本研究では、配線性を定量的に見るための指標を導入する。配線ツールは、何度か配線を試行して配線可能性を探索する。一定の配線試行限度を与え、それ以下の試行回数 t で配線が終了した場合、配線完了となる。配線が完了した場合、この t を配線性の定量的な指標とすることが

できる。この場合、 t が小さいほど配線性は良いことになる。また、配線が完了しなかった場合、残留コンフリクト数 T を配線性の悪さの指標とすることができる。配線が完了したか否かにかかわらずすべてのアーキテクチャに順序ついた指標を付与するため、配線が完了した場合は $-1/t$ を、完了しなかった場合は T をそのアーキテクチャの評価値とする。すなわち評価値が小さいほど配線性は良いことになる。そして、評価値が良いほうから一定の割合を合格として更なるスイッチ削減に回し、それ以外は棄却する。合格割合を調整することで(合格閾値を厳しくする)、ループを回るとのアーキテクチャ数増加を抑えることができる。また、この方法は、配線試行限度を小さくすることができ、それによる処理時間短縮効果もある。通常の配線の場合では、配線試行限度をあまり小さくするのは好ましくない。それ以上の試行回数で配線完了するアーキテクチャを配線不能と判断することになるためである。しかし、上記のように、一定の割合の良い評価値のものを残すようにすると、評価値の絶対値は問題ではなくなる。たとえ、配線が完了しなくても、評価値が一定の割合に含まれていれば、そのアーキテクチャは残すことになるからである。すなわち、通常より配線試行限度を小さくしても、結果に対する影響はあまりないと考えられる。

以上は、評価の良さを利用してアーキテクチャ数を抑えるものであるが、逆に評価の悪さを利用してさらにアーキテクチャ数を絞ることができる。一般にスイッチを削減するにしたがって配線性すなわち上記の指標は悪化する。したがって、配線性が非常に悪いスイッチ削減パターンを含むようなスイッチ削減パターンは、配置配線してみるまでもなく配線性が悪いと推定される。そこで、評価値が悪い順から一定の割合のスイッチ削減パターンを禁止リストに登録し、以降のスイッチ削減手続き(図 3 の手続き 70)において、禁止リストに含まれるスイッチ削減パターンを含むものは、配置配線に回さず棄却するようにする。これによってアーキテクチャ数をさらに抑制できる。

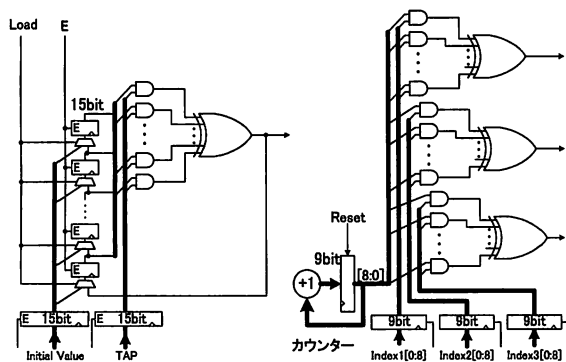
4.4 初期段階

以上のような漸次削除段階によって、一つずつスイッチを削減していき、アプリケーションが配置配線できる最小のスイッチ構成を見つけ出す。ただし、汎用再構成可能 HW からいきなり漸次削除段階のみでスイッチ削減するとうまく機能しなかったり非常に時間がかかったりする。汎用再構成可能 HW は豊富なルーティングリソースを持つため、スイッチを多少削減しても配線性はあまり変わらない。このため、どのスイッチを削減したアーキテクチャも評価値はほとんど同じになってしまい、アーキテクチャ数の絞込みがうまく

できない。このため、初期段階を設けて、最初に未使用スイッチを一気に削除する。これによって、領域特定再構成可能HW生成処理は大幅に加速される。

5. アプリケーションとアーキテクチャ

今回アプリケーションとして取り上げたのは、ワイヤレス通信などで使われる2種類の符号生成器である。図4(1)は汎用M系列生成器、(2)はOVSF符号生成器である。これらの回路規模はほぼ同じである。



(1) 汎用M系列生成器 (2) OVFSF符号生成器

図4 試行アプリケーション

次に、今回使用した再構成可能HWについて説明する。再構成可能アレイの本体を形成するPEセル(P)と、入力セル(W)と、出力セル(E)があり、WPPPEの順に配置した小規模な1次元セルアレイ構造の再構成可能HWを使用した。図4の回路はどちらも過不足なくこのセルアレイにマッピングできる。

PEセルは8bit処理系と1bit処理系を含んでおり、両者を独立に使ったり、結合して9bit処理系にしたりすることができる。

PEセルは、演算器として8bitと1bitの独自の高性能ALUを持ち、このほか8bitと1bitのレジスタを持つ。通常のALUの機能は図5(1)の一部(AND, OR, XOR)や加減算のみであるが、今回使用した高性能ALUはそれ以外の多様な機能を1つのALUで実現できる[5]。特に、図5の(3),(4)のように通常のALUで2段分の機能を1段で実現できるため、遅延、面積ともに大幅に改善することができる。

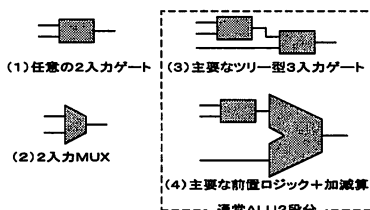


図5 高性能ALUの機能

配線をプログラマブルに接続するスイッチには、パストランジスタやトランスマッションゲート、トライステートバッファなどは使わずマルチプレクサのみを使用している。したがってすべて方向性配線である。今回は1次元セルアレイであるため水平方向にセルを繋ぐ配線のみを持つ。配線セグメントの長さとしては、4PE長、2PE長、1PE長、ダイレクトの4種類を用意した。

図6は、配置配線ツールでアプリケーション回路(M系列生成器)をマッピングしたときの画面である。左右両端は入出力セルであり、その間に4つのPEセルがある。



図6 再構成可能HWにM系列生成器をマッピングした画面

6. 評価

今回の試行では、アーキテクチャを描画した図面から定義ファイルを生成して図3における汎用再構成可能HWとし、次に、図4のアプリケーションをマッピングしながらスイッチを削減して領域特定再構成可能HWを生成した。全体の処理時間は1時間弱であった(CPU: Xeon(TM)3.60GHz)。PEセル当りのスイッチ数は、汎用再構成可能HWに比べて領域特定再構成可能HWでは1/5近くに減少した。

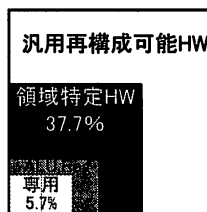
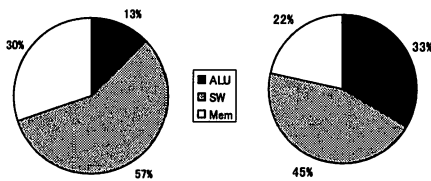


図7 面積比較

図7に、今回の試行における面積比較を示す(90nmスタンダードセルで評価)。汎用再構成可能HWに比べて、領域特定再構成可能HWは約1/3程度にまで面積が縮小された。図7において、ALUとReg(レジスタ)の大きさは、汎用、領域特定両方で変わらない。図4の両アプリケーションを専用回路で作った場合の合計面積は、領域特定再構成可能HWの1/6~1/7程度になる。両アプリケーションの面積はほぼ同じである。



(1) 汎用再構成可能HW (2) 領域特定HW

図8 各構成要素の比率

図8は、PEセル内の構成要素の面積比率である。柔軟性を実現するスイッチとコンフィギュレーションメモリの合計が、汎用再構成可能HWでは87%を占めているが、領域特化によって67%に低下している。ただ、均一スイッチ削減によってある程度の柔軟性を保持しており、領域特化後も柔軟性要素が2/3を占める。

図9にコンフィギュレーションメモリ量(bit数)を示す。領域特化によって1/4程度にまで削減されている。この領域特定再構成可能HWでは、32bit書き込みの場合、10クロック程度で全コンフィギュレーションができ、非常に高速な書き換えが可能である。

図10に各アプリケーションのクリティカルパス遅延を示す。汎用再構成可能HWに比べて領域特定再構成可能HWでは約1割程度速くなっている。専用回路に比べて遅延がかなり大きい、マッピング上の改善余地が残っており、更に数割程度は改善できる見通しである。

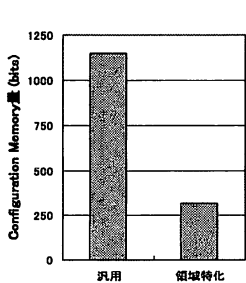


図9 構成情報メモリ量

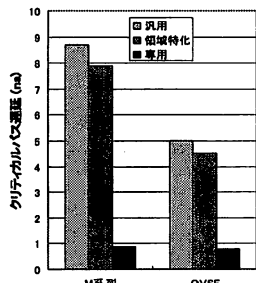


図10 遅延

7. まとめと考察

本報告書では、アプリケーションに応じて柔軟性を制限した再構成可能HWである領域特定再構成可能HWについて報告した。汎用再構成可能HWから、アプリケーションを配置配線できる範囲で極力スイッチを削減することで領域特定再構成可能HWを得る。この工程ための環境をアーキテクチャ追従配置配線ツールを使って構築した。

スイッチの削減パターンは無数にあり、現実的な時間

でスイッチ削減処理を終えるためには効率的なパターン絞込みが必須である。本研究では、アプリケーションを配線できたか否かではなく、アーキテクチャの配線性を表す定量指標をアーキテクチャ評価に導入した。良い評価の一定の割合のアーキテクチャを残し、一定以上悪い評価のスイッチ削減パターンは禁止リストに載せて以降そのパターンを含む削減パターンを棄却する。これによって領域特定再構成可能HWを短時間で生成できるようになった。

領域特定再構成可能HWでは、スイッチを削減した後もある程度の柔軟性を残すことが求められる。このために、すべてのセルで同じスイッチを削減する均一削減を採用した。この均一削減は、スイッチ削減パターンを減らして処理時間を短縮する上でも非常に有用である。

通信に使用される2種類の符号生成器をアプリケーションとし、領域特定再構成可能HW生成を試行した。その結果、汎用再構成再構成可能HWに比べて、領域特定再構成可能HWは1/3の面積になることを確認した。

領域特定再構成可能HWは、面積が小さいものの、応用に応じて柔軟性が絞り込まれているためFPGAのような汎用再構成可能HWの置き換えには使えない。同じ分野で複数品種あるLSIのわずかな差異を吸収して一品種にまとめ、低コストを実現するのに適している。また、出荷後の若干の規格変更やバグ修正などにも有用である。

今回は最初の試行であり、アプリケーションが小規模で数も少ない。今後、より大規模で、多種類のアプリケーションを使って試行を行う予定である。それを通じて、領域特定再構成可能HWの持つ柔軟性を定量化するモデルについて検討したい。

文献

- [1] V. Betz and J. Rose, "VPR: A New Packing, Placement and Routing tool for FPGA Research", Proc. of Int'l Workshop on FPL, pp.213-222, London, UK, Sep. 1997
- [2] V. Betz and J. Rose, "Automatic Generation of FPGA Routing Architectures from High-Level Descriptions", Proc. of Int'l Symposium on FPGAs, pp.175-184, Monterey, CA, Feb. 2000
- [3] M. Holland, S. Hauk, "Automatic Creation of Reconfigurable PALs/PLAs for SoC", Proc. of Int'l Conference on FPL, pp.536-545, Antwerp, Belgium, Aug./Sep. 2004
- [4] A. Danilin, M. Bennebroek, S. Sawitzki, "A Novel Toolset for the Development of FPGA-like Reconfigurable Logic", Proc. of Int'l Conference on FPL, pp.640-643, Tampere, Finland, Aug. 2005
- [5] 中谷正吾, 特許第3185727号, 2001.5.11 登録