

論理アーキテクチャ設計支援システム「RODIN」

概要

本システムは、Pascal 言語で記述された動作仕様からレジスタ・トランスファ (RT) レベルの回路を合成する。システムの開発を通して、各種設計問題に応用可能なシミュレーテッド・アニーリング法への知識の導入とその並列化の有効性を実証する。

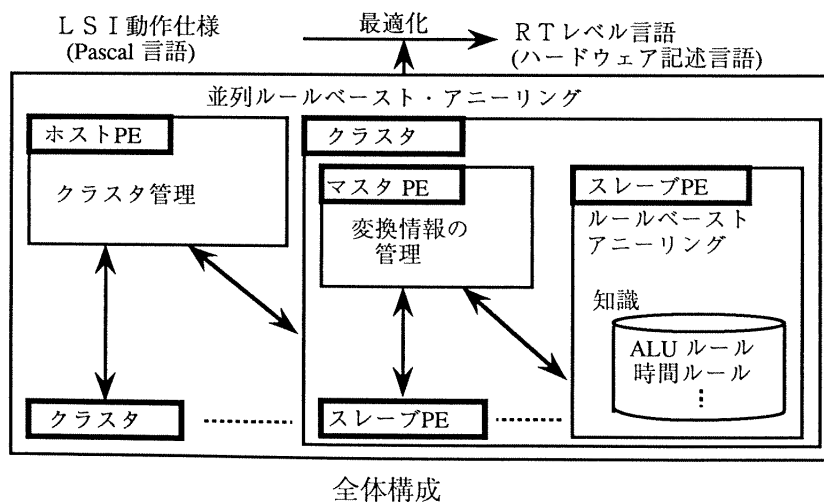
特徴

ルールベースト・アニーリング (RA)

ランダムな変換だけでなく、ヒューリスティックな変換 (ルール) を加え、ルール毎のコスト減少率に応じてルール選択確率を動的に変化させ、与えられた最適化時間内で良質な解を得る。

RA の並列処理方式

受理率に応じて複数の PE でクラスタを構成し、クラスタ内の PE が協調して 1 つの状態を最適化することにより、中低温時の受理率を増加させる。

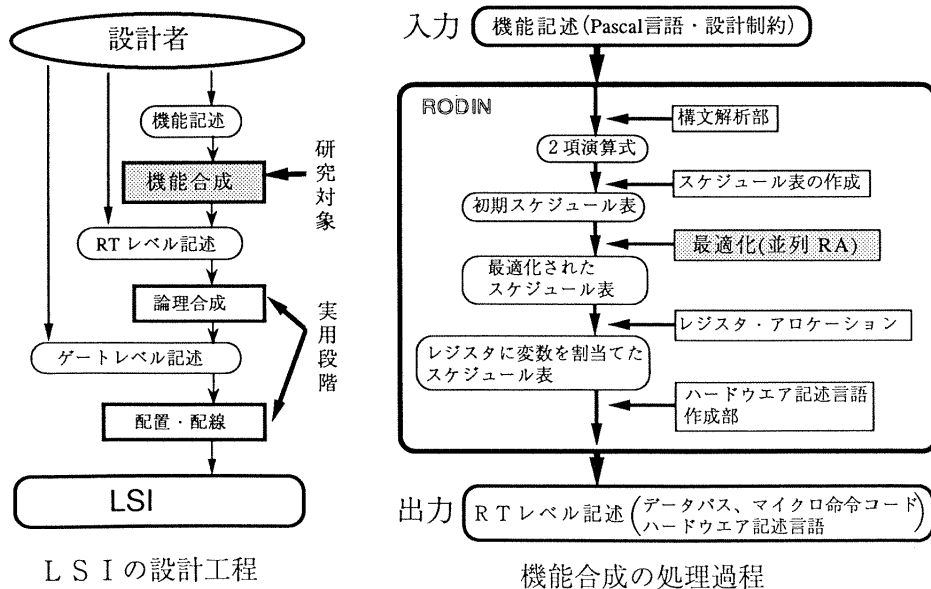


概要

現在、LSI の設計方法は、機能記述（動作仕様）や RT レベル記述（レジスタや演算器を組み合わせた記述）、ゲートレベル記述（AND 回路や OR 回路などのゲート回路を組み合わせた記述）をその抽象度に合わせて記述し、設計する方法が主流になりつつある。既に、論理合成や配置・配線は自動化されているが、機能記述から RT レベルの回路を作成する段階（機能合成）は自動化されてなく、複雑な最適化を設計者が行なう必要があった。我々の研究対象は、この機能合成の自動化である。

機能合成の処理過程

- 構文解析部、スケジュール表作成部ではシステムへ入力された Pascal 言語を二項演算式に展開し、その式を実行する演算器とタイムステップを適当に定め、初期スケジュール表を作成する。
- 最適化部では、なるべく小さいチップ面積、早い処理速度の LSI が生成できるように初期スケジュール表を最適化する。
- レジスタ・アロケーション部では、最適化したスケジュール表の変数を、どのレジスタに割り付けるか決定する。
- ハードウェア記述言語作成部では、最適化されたスケジュール表とレジスタ・アロケーションの結果からハードウェア記述言語を生成する。



ルールベースト・アニーリング (RA)

従来のシミュレーテッド・アニーリング (SA) に対して、知識による変換 (ルール) を付加し、各ルールの選択確率にもとづき確率的に適用する。また、各ルールの選択確率は動的に変化し、設計時間が短いときは貪欲なルールが優先して用いられ、設計時間が長いときはランダムな変換が優先して用いられる。本方法により、与えられた設計時間内でよい回路を設計することができる。

デモ概要 (1)

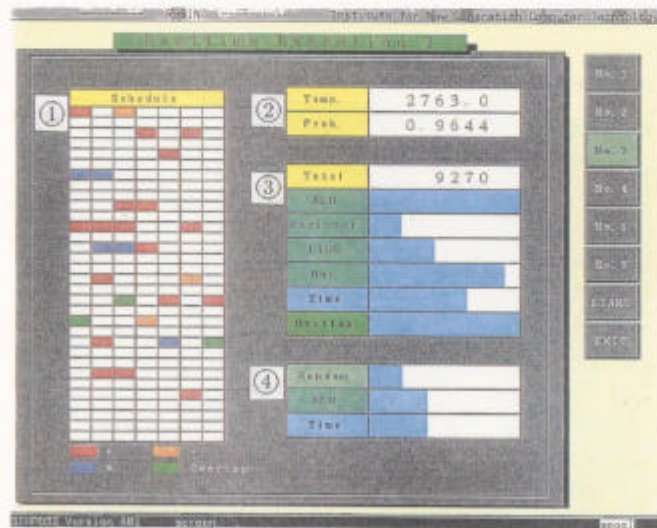
① スケジュール表

スケジュール表は二次元の表であり、横軸が演算器 /ALU を、たて軸がタイムステップを表す。各二項演算式が、どの ALU においてどのタイムステップで実行されるかを示す。

② アニーリング温度 T 、およびコストが 100 上がったとき受理される確率 ($e^{-100/T}$)

③ スケジュール表に対応する全体コストおよび各部分のコスト

④ 各ルールの選択確率をランダムな変換群、ALU に関するルール群、実行時間に関するルール群でまとめたもの



デモ画面 (1)

並列処理

- 中低温での受理率(スケジュール表が更新される割合)を増加させる。
- 一つのマスタ PE と複数のスレーブ PE で一つのクラスタを構成し、すべてのクラスタをホスト PE が管理する。

クラスタ 複数のスレーブ PE が並列に有効な変換を見つけ、その変換をマスタ PE が適用することで受理率を増加させる。

ホスト PE スケジュール表の最適化が進むにつれて、一つのクラスタに含まれるスレーブ数を増加させることにより、受理率が低下するのを防ぐ。

デモ概要 (2)

① クラスタの動的な変化

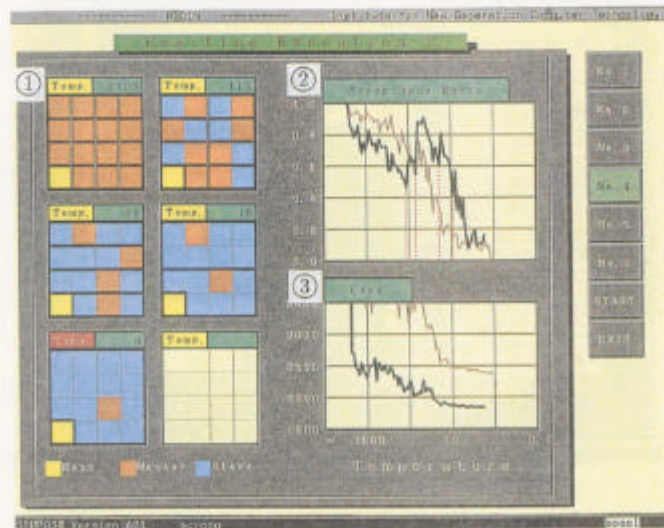
最初は、ホスト PE を除くすべての PE がマスタ PE として動作するが、徐々にスレーブ PE が増加していく様子を表示する。

② 受理率の変化

中低温でも受理率が低下しない様子を表示する(紫線は従来の SA)。

③ コストの変化

温度の低下とともに、コストが減少していく様子を表示する(紫線は従来の SA)。



デモ画面 (2)