

LSI 設計エキスパート・システムの開発

2G-7

(1) 論理回路変換

宮崎俊彦 安倍義博 中野利満 菊地一成 浅野俊昭

キヤノン株式会社

【はじめに】

次世代の半導体設計ツールとして、人工知能(知識情報処理)を応用したCADシステムが注目を集めている。人工知能を応用するメリットは、設計過程において、完全にアルゴリズム化するのが難しく、自動化が困難な分野に適應できる点にある。また知識を更新、或は追加する事によって機能UPが可能のため、急速な環境の変化にも対応可能であることも大きな魅力である。LSIの設計は、論理的な側面もあるが、目標とする製造プロセスによって設計手法が全く変わってしまうという特殊性を持っており、全設計工程が自動化されるまでには至っていないのが現状である。

我々は、これまで、自動化が困難とされてきた、専門的、経験的な知識を必要とする設計工程を、知識情報処理を応用して、自動化するシステムを開発中である。本稿では、その一テーマである『論理回路変換エキスパート・システム』について報告する。

【システムの機能】

本システムの機能としては、

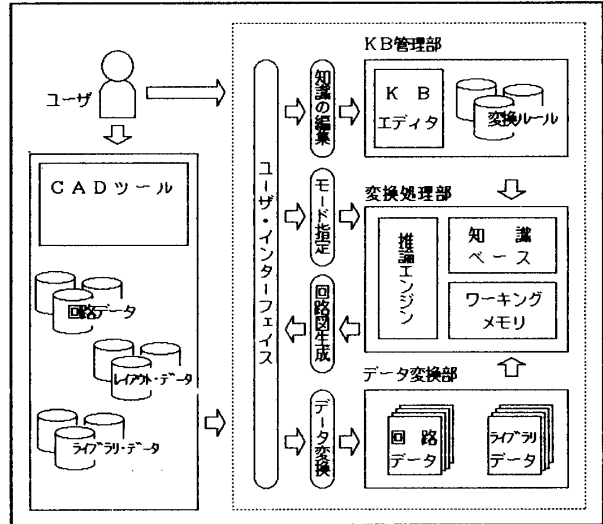
- ① 汎用ロジック等を用いて設計された一般的な論理回路を、特定プロセスに合ったLSI用の回路に自動変換する
- ② あるプロセスで設計されたLSI回路を異なるプロセスのLSI回路に自動変換する。
- ③ 変換する際、エラーチェック、論理圧縮、ファンアウト調整、等を行う。

等がある。これにより、回路の再利用性が向上し、回路設計における設計者の負担は軽減される。

【システムの構成】

『論理回路変換エキスパート・システム』の構成を第1図に示す。本システムは、市販のCADツールに組み込んで使用できるように、CADツールとのインターフェイスを備えている。この部分を修正することによって、様々なCADツールに対応できる。

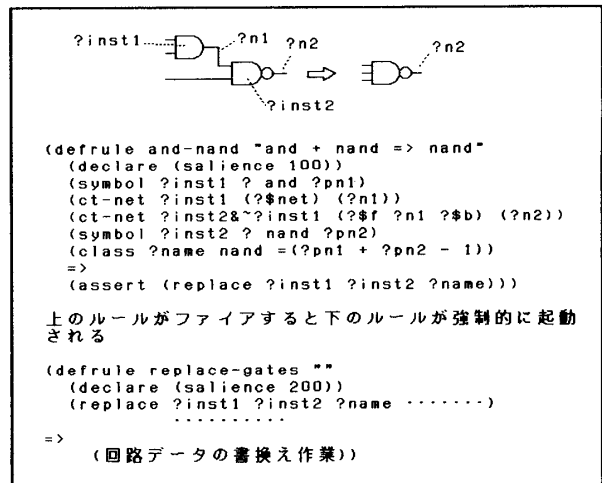
知識処理には、汎用のエキスパート・システム構築ツールのARTを用いた。また、実行マシンはSYMBOLICS-3640である。



第1図 システムの構成

【知識表現】

変換知識(ルール)の記述例を第2図に示す。本システムでは、この例の様な、局所的な部分回路の変換の繰り返しのより、全体の回路の変換処理を行なっている。



第2図 変換知識の表現方法

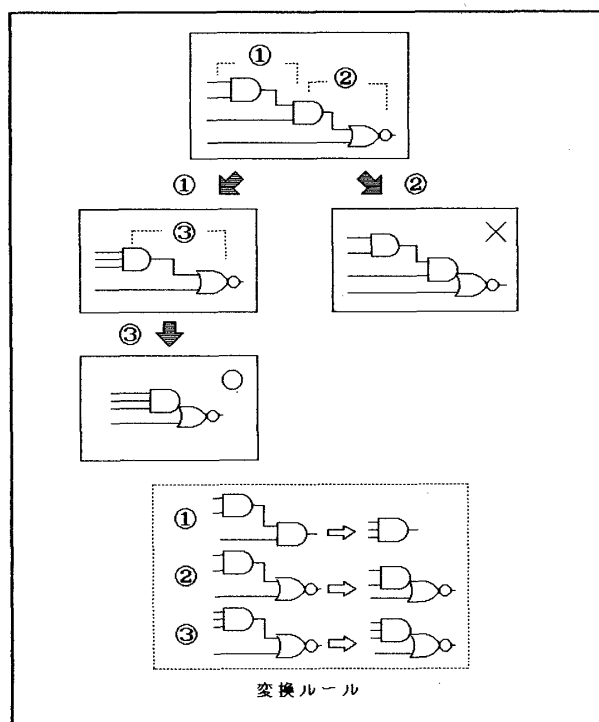
推論方式は、すべて前向き推論で、競合解消は、SALIENCE(ルールの重み付け)と、制御用ファクト(リスト構造のデータ)の埋め込みで行なっている。

第2図の例では、(REPLACE ……)が、制御用ファクトにあたり、このファクトがワーキング・メモリに現れると、次に必要なルールが強制的に起動される。しかし、これだけではルールの制御が不十分であるので、変換ルールの適用順序の最適化を、仮説推論により実現している。

### 【仮説推論による最適変換】

局所変換ルールの適用によって、回路データを処理してゆく場合に問題となるのがルール適用の順番である。第3図に簡単な例を示す。図の回路では、ルール②よりルール①を先に適用した方がより最適な変換結果が得られることがわかる。この様に、本システムでは、ある回路部分において、適用可能なルールが、同時に、複数存在した場合、それらすべてに対する仮説を立てて、可能な変換を全て行ない、その結果を評価して最適な結果を得る事が出来る。

この手法の欠点は、処理スピードが低下する事である。これを解決するためには、仮説推論を行なうための回路切り出し方法の改良、不要な仮説空間の刈り込み、ヒューリスティック関数による、無意味な探索の中断、及び最適解の早期発見、学習機能の追加等の方法を考える必要がある。



第3図 仮説推論による最適解の探索

### 【ルール数の削減】

ルール・ベース・システムの欠点は、処理スピードが遅い事である。また、ルール数や処理するデータが多くなると、処理スピードは急速に低下する。

これを回避するためには、ルールの削減や、一度に処理するデータを少なくする事が必要である。

ルール数の削減は、①ルールの条件部の共通パターンを抽出し、それを一つのルールにまとめる。②CLASS情報を付加して、同種のルールを一個のルールで記述できるようにする。等の方法で行なっている。

### 【回路データの分割について】

LSIの回路規模は非常に大きく、素子数で数千、数万に及ぶこともある。このような多数のデータをルール・ベース・システムで一度に処理するのは、現実的ではない。これを回避するためには、対象とする回路を分割して、回路データを小さくしてやる必要がある。回路分割の方法は、①回路の繰り返しパターンに注目する、②バスラインなど特殊なネットによって、回路を分割する、③変換可能な部分を最初に抽出しておき、そのデータだけにたいして変換ルールを適用する等の方法で行っている。

### 【知識ベースの管理】

本システムでは、ユーザの希望により、様々なモードの変換を行えるように設計されている。モードの変換は、機能毎に分割された知識ベースを組み合わせる(ブレンドする)ことにより行われる。これを実現するためには、知識ベースの管理を自動的に行うことが不可欠の条件である。また、知識の追加、削除、変更に対して、システムの正常な動作を保証するためにも、知識ベースの管理が必要となる。本システムでは、知識ベース・エディタを作成し上記のような知識ベースの管理を行っている。知識ベース・エディタはグラフィック入力により、ルールを作成する機能も備えている。

### 【まとめ】

ルール・ベース・システムを使った論理回路変換システムについて、その概要を述べた。

今後は、変換部の高機能化、処理速度の向上、インターフェイス部の充実等を行っていく予定である。

### 【参考文献】

- [1]David Gregory, Karen Bartlett, Aart de Geus, and Gary Hachtel "SOCRATES: A system for automatically synthesizing and optimizing combinational logic", Design automation conference June 1986
- [2]Saburo Muroga "Logic design and switching theory" 1981 Kyoritsu Shuppan Co. Ltd.