

論理設計エキスパートシステムLODEXの概要

2L-7

角田多苗子, 丸山 文宏, 真野 良男, 林 一司, 川戸 信明

富士通株式会社

1. はじめに

我々は第五世代コンピュータプロジェクトの一環として、ハードウェアの論理設計を対象としたエキスパートシステムの研究を行っている。これまでに、ハードウェアの動作アルゴリズムからCMOS回路を生成する実験システムを試作した¹⁾。これは、従来計算機による支援が困難と考えられていた機能設計の部分を、知識ベースのアプローチにより、また回路設計の部分をアルゴリズムミックスなアプローチにより自動化するもので、論理型言語Prologにより統一的に実現できることを示した。しかし、システムの入力となる動作アルゴリズムの設計(方式設計)自体難しいものであり、この部分の支援も含めたトータルな設計支援システムが望まれる。

LODEX (Logic Design expert system) は、方式設計に関しては、高集積度の活用が容易で並列処理による効果を得るのに適していると考えられる、プロセッシングエレメント (PE) を規則的に配置したVLSIを対象として、視覚的な支援を行う。それに続く機能設計に関しては、逐次型推論マシンPSI上にESPで構築した知識ベースシステムにより自動化する。本稿ではLODEXの概要について報告する。

2. システム構成

図1にLODEXのシステム構成を示す。

本システムを用いた設計過程は、ハードウェア・アルゴリズムを設計していく方式設計と、方式設計の結果に基づいて機能設計を自動生成する2つの過程から構成される。LODEXでは、アルゴリズムを、PEの動作アルゴリズムとその接続関係で表現し、機能設計の結果得られるステートマシンを、ハードウェア記述言語DDLで表現する。

規則的な構造を持つVLSIを設計する場合、方式設計の段階からその構造を決定することが必要である。視覚的インタフェースモジュールは視覚的な支援によりこのプロセスをサポートする。この過程を通して設計したハードウェア・アルゴリズムはPEが並列動作を行うため、全体の動作を確認することが難しい。この検証を行

うために、方式シミュレータを用いる。

機能設計の過程は推論機構モジュールによって進められる。推論機構はアジェンダとルールによって、ワーキング・メモリ上の設計データを書き換えていく。ワーキング・メモリの初期値はハードウェア・アルゴリズムである。設計データは、論理設計の世界のモデルのインスタンスとして表現する。

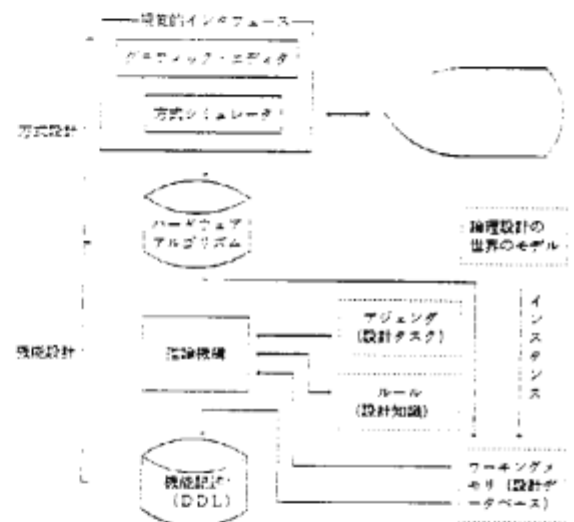


図1 システム構成

3. ハードウェア・アルゴリズム設計支援

図2は、視覚的インタフェースのグラフィック・エディタを用いて、規則的に配置したVLSIのハードウェア・アルゴリズムを設計した例である。一つのPEの動作アルゴリズムが左のウィンドウに、それらを接続した様子が右のウィンドウに現れている。規則的に配置することに対応して、PEを一次元、二次元に指定した倍數

Overview of Logic Design Expert System (LODEX)

Taeko KAKUDA, Fumihiko MARUYAMA, Tamio HANO, Kazushi HAYASHI, Shuichi KAWATO
FUJITSU Ltd.

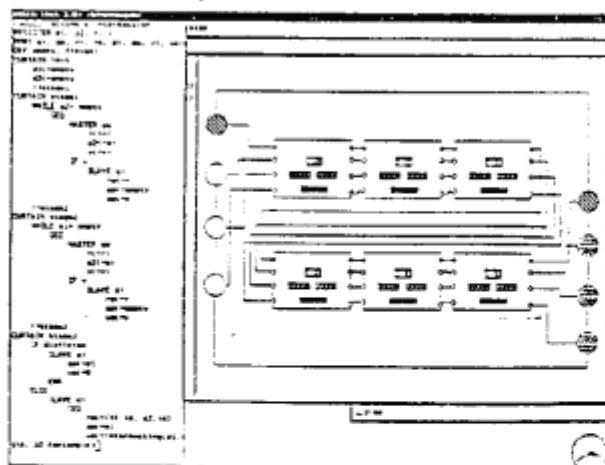


図2 ハードウェア・アルゴリズム

だけ展開する機能をもつ。動作アルゴリズムの記述言語は、従来使用していたoccamにかわり、変数ではなくハードウェア素子（レジスタ等）を対象としたオペレーションを記述でき、割り込み処理の記述が容易で、プロセス間通信を同期式に制限しない、などハードウェア記述向きの特徴をもったものを考案した。

シミュレータはオブジェクト指向の概念に基づいて構築されており、シミュレーションの制御の受渡しや実行は、オブジェクト間のメッセージ・パッシングによって表現される。実行すべきオブジェクトはスケジューラに登録される。このシミュレータでは、PEやPE内レジスタ等をオブジェクトで表すほか、動作アルゴリズム内のオペレーションのかたまり（WHILE文など）もオブジェクトとして扱っているため、スケジューラが管理する対象をすべてオブジェクトとして統一的に扱うことができる。

4. 機能設計エキスパートシステム

4.1 知識表現

LODEXでは、レジスタ、ステートといった論理設計の世界に現れる個々の概念をESPのクラスで表現し、それらでモデル空間を構成する。ワーキング・メモリに現れる個々の設計データはクラスに対するインスタンスとして表現する。個々の概念に対応するクラスには、自分自身を操作する手続が記述され、各設計データに継承される。Prologで実現した実験システムでは、個々の設計データがPrologのfactの形で表わされ、ワーキング・メモリ更新のためのデータ操作手続の詳細がルールの中に含まれていたが、LODEXでは、このような手続きをルールの中に書かずに設計デー

タにメッセージを送るだけでよくなり、知識としては本質的ではない操作手続きの詳細を知識本体から分離することができる。

設計知識を表すルールは、機能設計中の各過程に対応した知識源（KS）にモジュール化し、各KSを一つのオブジェクトとする。ルールは設計作業に対応してワーキング・メモリを更新していく前向き推論ルールと、前向き推論ルールのIF部として、条件チェックを行う後向き推論ルールから構成される。

4.2 推論機構

設計では、一つのクラスに属する多数の設計データに対して、同じ作業を繰り返さなくてはならない場合がある。LODEXではそれぞれの作業をタスクと呼び、タスクの集合を格納するアジェンダによる推論制御を行う。各設計過程に対応するKSは、メッセージの受信により起動されると、そこで行うべきタスクの集合をアジェンダに書き込む。インタプリタはアジェンダから一つずつタスクを取り出し、そのタスクに関して適用できる前向き推論ルールを選択し、そのIF部にあたる条件をチェックする際、後向き推論ルールを起動する。IF部が成り立つと、THEN部の実行によってワーキング・メモリの内容が更新されていく。アジェンダから最後のタスクが取り除かれたとき、インタプリタの実行は終了する（2）。

5. おわりに

LODEXは、規則的な構造をもつVLSIを対象を絞ることによって方式設計を支援し、オブジェクト指向の概念を活かした知識ベースシステムとして機能設計を自動化するエキスパートシステムである。今後複数のKSによる協調的な推論メカニズムについて検討し、回路設計も含めた論理設計全体の支援も行うつもりである。

謝辞

この研究を進めるにあたって御支援頂いたICOT第五研究室岩下室長（現NTT）に深く感謝致します。

参考文献

- 1) 丸山、真野、林、角田、川戸、上原、「動作アルゴリズムからCMOS回路を生成する論理LSI設計用エキスパートシステム」、日経エレクトロニクス、1985、11、4
- 2) 箕田、及川、澤田、楠本、秋元、角田、丸山、「論理設計エキスパートシステムLODEXにおける推論エンジンの開発」、情報処理学会第35回全国大会