

アーキテクチャ設計における 1U-4 設計支援エキスパートシステムの試作

小野昌之, 野田泰徳, 伊申泰宣, 平野達郎

【沖電気工業(株)】

1. はじめに

計算機の高性能化への要求は増々強まるばかりであり、それに伴い高度な制御機構を持つハードウェアの設計が多くなってきている。特にパイプライン方式計算機のアーキテクチャ設計の難度は高く、設計者への負担も大きい為、その設計支援が望まれている。

現在、このパイプライン方式計算機のアーキテクチャ設計支援方法について検討し、知識処理技術を応用した設計支援ツールの試作を行っているので、このシステムの構成及び処理について報告する。

2. アーキテクチャ設計とは

パイプライン方式計算機は複数の命令を並列に実行する為、通常の計算機に比べ原理的に処理スピードが速い。しかし、パイプライン動作の乱れが少ない高性能なパイプライン計算機を設計するのは難しい。パイプライン動作が乱れる原因としては、ファシリティ利用の競合や、メモリアクセスのような複数マシンサイクルかかる動作、及びJUMP命令の動作等がある。そこで設計者はこの乱れを少なくする為、設計案をつくる毎に乱れの原因をチェックし、設計案を改善している。この改善作業は人間の代表的知的活動であり、設計者の経験がその結果であるLSIチップの面積や処理スピードを大きく左右する。それ故にこれらの設計作業を支援する設計ツールが必要とされる。

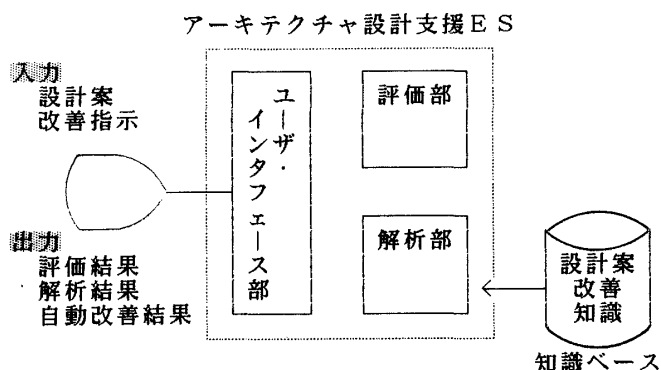


図1 システム構成

3. アーキテクチャ設計支援エキスパートシステム

本システムは、設計案の評価、評価結果の解析、及び設計案の改善といった設計者が行う一連の設計作業を支援し、処理スピードと面積の2つの制約を満足する設計解を求めるものである。本システムのシステム構成を図1に、それを用いた設計プロセスを図2に示す。

(1) ユーザ・インタフェース部

設計案にはデータベース構造、データベース上での命令の実現方法を記述した命令定義、及びパイプライン段数やマシンサイクル等の各種アーキテクチャ・パラメータがある。ユーザ・インタフェースは、この設計案を対話形式で作成する機能を持つ。

(2) 評価部¹⁾

評価部は、設計案を入力とし、設計者が評価を行う際に必要となる命令の実行状況を出力する。ここでは、命令のシーケンスであるテストプログラムを用いてシミュレーションを実施しており、1マシンサイクル毎

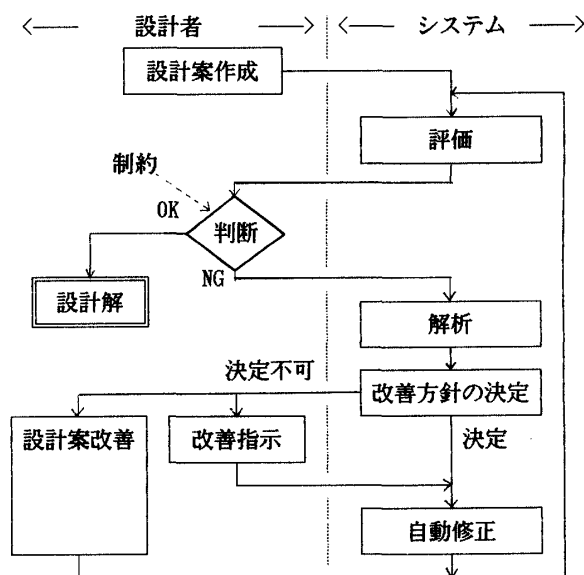


図2 本システムを用いた設計プロセス

に命令の実行待ちを検出する。現在本システムでは、ファシリティ利用の競合による実行待ちと、記憶要素におけるREAD/WRITEのタイミングの調整による実行待ちの2つの検出を行っている。

シミュレーション結果としては、1マシンサイクル毎の各命令の実行状況データを表現したタイムチャート、並びに平均並列実行度、平均MIPS値、各ファシリティの利用回数及び競合度合といった統計データがある。これらのデータは解析部で用いられる。

(3) 解析部

解析部はシミュレーション結果から制約を満足しない原因を解析し、設計案改善機能を用いて制約を満足するように設計案を改善するものである。この原因の解析では、面積が大きい時は利用度の少ないファシリティ、及び機能が単一であり且つ他のファシリティと統合できそうなファシリティ等に原因があると判断し、また処理スピードが遅い時は、ファシリティ競合度合の比較的高いファシリティに原因があると判断している。

4. 設計案改善機能

本システムでは、設計者に代って設計案の改善方針を決める為に知識処理技術を導入しており、図2に示すようにその改善方針にそって設計案が修正される。しかし、知識ベース化できないあいまいな判断の部分には設計者が介入し、設計者がシミュレーション結果から判断してシステムに対する設計案の改善方針の指示を行う。ただし、自動修正できない場合は設計者が直接設計案の改善を行う。

本システムは、処理スピードや面積に対する制約を満足させる為に、以下の改善知識を用いている。

- (1) 処理スピード向上の為にデータベース構造改善知識
 - ・ 同一ファシリティの追加
 - ・ ファシリティの機能分割等
- (2) 処理スピード向上の為に命令定義改善知識
 - ・ データ転送ルートの変更等
- (3) 面積削減の為にデータベース構造改善知識
 - ・ 機能の同じファシリティの併合
 - ・ 機能の統合した新規ファシリティへの変更等

図3に本システムで用いている改善知識の例を示す。これは面積は増えるが、処理スピードの向上が期待できる知識である。

設計案のシステムによる自動修正においては、設計案であるデータベース構造と命令定義が密接に関係している為、単に一方の修正だけではすまない。そこで本

システムはESP²⁾のオブジェクトプログラミングを用い、データベース構造と命令定義の修正に対し図4のような各々専用の修正オブジェクトを作り、メッセージ通信で互いの修正内容を密に提供しながら自動修正する方式をとっている。図3の例の場合、図4に示されるように、データベース構造に対し同一ファシリティを並列に加える修正が実施され、その修正内容が命令定義修正オブジェクトにメッセージ送信されることで、命令定義におけるデータ転送ルートの修正が行われる。

```

IF (ファシリティAでの競合度合が高い)
  & (面積に余裕がある)
THEN (Aと同一種ファシリティを並列に追加せよ)
    
```

図3 改善知識の例

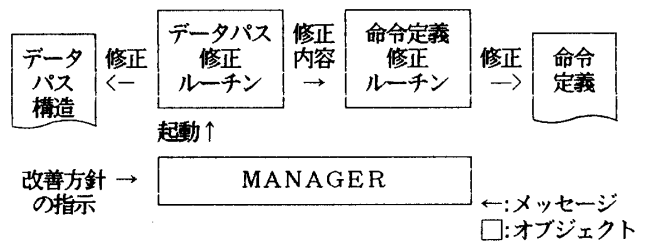


図4 自動修正におけるオブジェクト構成とメッセージ・パッシング

5. おわりに

アーキテクチャ設計支援エキスパートシステムの試作により、本システムが設計の上流工程を支援するCADツールとして有効であること、また評価-解析-改善の繰り返しの処理プロセスが、設計問題である合成型問題の解決に有効であることが明確になってきた。

現在、本システムは改善知識の数が少なく設計者の介入が多い。今後順次獲得できる設計者固有の改善知識を取り込み、より自動化を進めたシステムを狙っていきたい。

尚、本研究は第5世代コンピュータプロジェクトの一環として行われた。

【参考文献】

- 1) 蛭田 他：並列論理型言語による計算機性能評価支援システム, 情報処理学会第34回全国大会(1987)
- 2) K. Takei, T. Chikayama, S. Takagi: ESP-An Object Oriented Logic Programming Language, ICOT研究速報 TM-0075(1984)