

第5世代のコンピュータ

研究開発課題説明書

昭和56年6月



財団法人 日本情報処理開発協会



本書は、日本自転車振興会から競輪収益の一部である
機械工業振興資金の補助を受けて作成したものです。

目 次

1. 研究開発課題概要	1
2. 研究開発課題	19
2.1 基本応用システム	19
2.1.1 機械翻訳システム	19
2.1.2 質問応答システム	25
2.1.3 音声応用システム	29
2.1.4 図形・画像応用システム	32
2.1.5 応用問題解決システム	36
2.2 基礎ソフトウェア・システム	41
2.2.1 知識ベース管理システム	41
2.2.2 問題解決・推論システム	48
2.2.3 知的インタフェース・システム	61
2.3 新アーキテクチャ	75
2.3.1 述語論理マシン	75
2.3.2 関数型マシン	81
2.3.3 関係代数マシン	90
2.3.4 抽象データ型マシン	95
2.3.5 データフロー・マシン	103
2.3.6 新ノイマン・マシン	111
2.4 機能分散アーキテクチャ	119
2.4.1 機能分散アーキテクチャ	119
2.4.2 ネットワーク・アーキテクチャ	129
2.4.3 データベース・マシン	137
2.4.4 高速演算マシン	146
2.4.5 高度マンマシン・コミュニケーション・システム	152
2.5 VLSI技術	159
2.5.1 VLSIアーキテクチャ	159
2.5.2 知的VLSI-CAD	166

2.6	システム化技術	175
2.6.1	知的プログラミング・システム	175
2.6.2	知識ベース設計システム	181
2.6.3	アーキテクチャ関連システム化技術	187
2.6.4	データベースおよび分散データベース・システム	193
2.7	開発支援技術	202
2.7.1	開発支援システム	202

1. 研究開発課題概要

1. 研究開発課題概要

本書は、第5世代コンピュータ・システムの研究開発プロジェクトにおいてとりあげるべき研究開発課題の概略仕様書である。研究開発課題は、表1-1に示すごとく7グループ26課題であり、夫々の課題について表1-2により概要を示し、第2章で各課題ごとにその必要性、内外技術の現状、研究開発項目、スケジュール、研究開発目標・仕様が詳細に述べられている。

表 1-1 第 5 世代コンピュータ・システム 研究開発課題

基本システム	1-1) 機械翻訳システム 1-2) 質問応答システム 1-3) 音声応用システム 1-4) 図形・画像応用システム 1-5) 応用問題解決システム
基ソ 礎フ トシ ウス ステ アム	2-1) 知識ベース管理システム 2-2) 問題解決・推論システム 2-3) 知的インタフェース・システム
新 アー キ テク チャ	3-1) 述語論理マシン 3-2) 関数型マシン 3-3) 関係代数マシン 3-4) 抽象データ型マシン 3-5) データフロー・マシン 3-6) 新ノイマン・マシン
機能分散 アー キ テク チャ	4-1) 機能分散アーキテクチャ 4-2) ネットワーク・アーキテクチャ 4-3) データベース・マシン 4-4) 高速演算マシン 4-5) 高度マンマシン・コミュニケーション・システム
V L S I 技術	5-1) VLSI アーキテクチャ 5-2) 知的 VLSI-CAD
シ ス テ ム 技 術 化 術	6-1) 知的プログラミング・システム 6-2) 知識ベース設計システム 6-3) アーキテクチャ関連システム化技術 6-4) データベースおよび分散データベース・システム
開 発 技 術 支 援 術	7-1) 開発支援システム

表 1 - 2 研究開発課題概要

課題グループ	研究開発課題	研究開発内容	目標・仕様	備考
基本応用システム 聞く, 話す, 見る, 描く, 考える, 解くなどの機能を代表する基本的な応用システムを研究開発する。 研究開発費: 100億円	機械翻訳システム ドキュメンテーション技術の研究, 知識利用に関する人工知能研究等の成果を総合し, 多国語間翻訳を行う総合システムの研究開発。	- 機械翻訳システム・コア部分設計 - 各種言語の文法開発 - 中間言語の設計 - 文生成用文法の開発 - 人間の介在を含む総合的機械翻訳システムの開発 - 専門用語データベース(知識ベース)の開発 - 専門用語データベースのためのマシンの開発 - 高機能ワード・プロセッシング技術の開発	- 使用語彙数: 10万語 90%の精度で翻訳し, 人間の介在によって, 残り10%を処理する。 - テキストの編集, 翻訳結果の印刷までの各工程で計算機が関与する総合システムとする。 - 全コストは, 人間が翻訳した場合の30%以下とする。	
	質問応答システム 知的CAE/CADシステム, DSS, 知能ロボット等の各種専門分野に共通の質問応答システムの研究開発。	- 会話解析技術 - 図形入出力装置 - 専門家知識作成技術 - 誤入力処理技術 - 自然言語サブセットの設計と文法および辞書の開発 - 総合化技術	5年後の中期目標として, 一特定専門分野に限定した質問応答実験システムを試作する。 - 使用語彙数: 2000語(日本語) - 曖昧さ解消のための補助的情報を使用者が与える。 - 推論規則数: 1000個 - 中期目標で試作した質問応答実験システムを評価し, 各種専門分野に対する質問応答システム・プロトタイプを作成する。 - 使用語彙数: 5000語以上 - 推論規則数: 10000個以上	知的ユーティリティ・システムの開発にも使用される。

課題グループ	研究開発課題	研究開発内容	目標・仕様	備考
基本応用システム (つづき)	音声応用システム 機械翻訳の入出力として用いられる汎用音声応答システム、音声タイプライタ、電話を用いた問合せ等で重要な話者認識システムの研究開発。	- 音声タイプライタの開発 - 音声応答システムの開発 - 話者認識システムの開発	- 音声タイプライタ ① 中期目標：数百～数千単語に簡単な構文情報を利用したシステム。 ② 後期目標：単語数を一万語程度とし同時に意味解析を加え、音声認識上の誤りを自ら修正し、全体としてわかり易い文章を出力する。 - 音声応答システム ① 中期目標：対象語彙を数千語程度とし、分析合成方式を中心とする。 ② 後期目標：一万語程度を対象とし応答内容の意味を把握し、自然な会話となるように高度化する。 - 話者認識システム ① 中期目標：数10名程度の話者 ② 後期目標：数100名以上の話者に対して実用的な認識時間で動作するシステム。	
	図形・画像応用システム 図形・画像情報を構成的に蓄積し、知識情報処理に利用するために	- 図形・画像情報蓄積・検索方式の研究 - 図形・画像情報検索用言語系の開発	検索の対象となる図形・画像データベースの図形・画像枚数は10万枚程度。	

課題グループ	研究開発課題	研究開発内容	目標・仕様	備考
基本応用システム (つづき)	効果的に検索するシステムの研究開発。	- 図形・画像データベース・マシンの開発 - 図形・画像情報蓄積・検索システムの開発	- 抽象レベルの記述を含めた図形・画像格納所要時間は、数秒以内とする。 - 図形・画像検索所要時間は、平均 100 m sec 以下とする。 - 中期目標値として、取扱う図形・画像の枚数 1 万程度、処理速度は最終目標値の 5 割程度とする。	
	応用問題解決システム 一般的で高度な問題解決を目指し、“問題”の記述を入力すると、その“解答”を出力する数式理解システム、基プレイング・システムの研究開発。	- 数式理解基本方式の研究 - 数式理解システムの開発 - 基プレイング基本方式の研究 - 基プレイング・システムの開発	- 数式理解システム ① 中間目標：処理機能としては、現在の MACSYMA 程度のものに不等式と簡単な方程式の処理機能を合わせたものを、知識ベース化したシステム。 ② 最終目標：高度の数式処理アルゴリズムを組み込んだ数式関連の知識表現・問題解決システム。 - 基プレイング・システム ① 中期目標：アマチュア 10 級程度のシステム。 ② 最終目標：アマチュア初段程度のシステム。	

課題グループ	研究開発課題	研究内容	目標・仕様	備考
- 基礎ソフトウェアシステム - 第5世代コンピュータ・システムの中核となる部分で、情報処理基本機能（管理、処理、対話）に対応するモジュール群を研究開発する。 〔研究開発費：150億円〕	知識ベース管理システム 人間の持つ知識を一定の形式で表現してコンピュータ内に蓄積し、利用することによって問題解決プロセスにおいて人を援助する知的システム管理技術の研究開発。	- 知識表現・利用技術の研究 - 知識獲得・学習の研究 - 大規模知識ベース・システムの研究 - 知識ベース管理システムの開発 - 知識ベース・マシンの開発	- 知識ベース管理システム ① 中間目標：ルールとデータの同時管理、データベース・アクセスの最適化機構、矛盾解消機構、推論マシンとのインタフェース。 ② 最終目標：多世界知識ベース、分散知識ベース、帰納的推論に基づく学習、推論マシンとの融合化。 - 知識ベース・マシン ① 中間目標：2000個のルールと、100万項目のデータを収容検索可能（1項目、10^3B）。 ② 最終目標：20000個のルールと、1億項目のデータを収容検索可能（100GB）。	
	問題解決・推論システム 第5世代コンピュータの処理機能の中核をなすもので、問題解決・推論システムの処理モデルの設定によりその処理能力を理論的に明らかにし、問題解決シ	- 問題解決・推論アルゴリズムの研究 - 問題解決向記述言語の開発 - 推論マシンの開発	- 推論マシン ① 最終目標：$10^2 \sim 10^3$ Mega LIPS程度の性能。 - 問題解決向記述言語 関数型、並びに論理型プログラミングをサポートし、かつ対象指向に基づくモジュラー化プログラミングを可能とすること。生産され	推論実行速度 1LIPS（Logical Inferences per Second）とは、「三段論法」による推論操作を1秒間に1回行うことを表す。1回の推論操作を現行のコンピュータで行うとすれば、100～1000ステップを要すると思われるので、1LIPSは100～1000 IPS（Instructions Per Second）に相当する。現世代コンピュータは、$10^4 \sim 10^5$ LIPS程度である。

課題グループ	研究開発課題	研 究 内 容		目 標 ・ 仕 様	備 考
・基礎ソフトウェアシステム (つづき)	システムの実現に向けての基本技術の研究開発。			たプログラム・モジュールはソフトウェア部品として知識ベース化され、知的プログラミングに於て有効に利用されること。	
	・知的インタフェース・システム 柔軟な会話機能を実現し、人間とコンピュータとの間の使用言語（自然言語や音声、図形、画像も含まれる）の相違に基づくギャップの解消を目指した技術の研究開発。	自然言語・音声系	・構文解析技術 ・意味解析技術 ・談話解析技術 ・文章合成技術 ・言語データベースの構築 ・自然言語処理装置 ・多言語基本文法の作成 ・音素識別方式の研究 ・文章理解方式の研究 ・音声合成方式の研究 ・話者個人差の学習方式等の研究 ・音声理解システムの開発	・自然言語・音声という言語情報、もしくは図形・画像という視覚情報を媒体として、人間がコンピュータと知的に対話するための技術を確立し、知的インタフェース・システムを提供することを目標とする。 ・自然言語・音声系の目標： ① 対象語彙は、コンピュータ関連用語や、科学技術文献等の内の1分野とし、専門用語および常用単語を含める。 ② 話者適応機能を持ち、不特定話者を対象とする。 ③ 日本語および英語の音声出力が可能とする。	

課題グループ	研究開発課題	研究内容		目標・仕様	備考
・基礎ソフトウェアシステム (つづき)	・知的インタフェースシステム (つづき)			④ 識別処理は実時間に近い能力を目標とする。	
		図形・画像系	・図形・画像の構造化技術の研究 ・図形・画像生成アルゴリズムの研究 ・システム化技術の開発 ・図形・画像処理マシンの開発 ・自然言語と図形・画像との関連づけの研究	・図形・画像系の目標： ① 図形・画像を介して、人間がコンピュータと円滑に対話するシステムの構築に必要なソフトウェアならびにハードウェアの開発を行う。 ② 対象とする図形は、中規模以下の機械設計図面程度の複雑さ、画像は医用写真程度の複雑さが取扱えるものとする。 ③ 処理速度は、人間との対話手段として円滑さを損わない程度に十分速いものとする。 ④ 中間目標として、扱う図形・画像の複雑さを最終目標の約7割を狙う。処理速度については、中間目標は特に定めずに、処理方式の確立に主力を注ぐこととする。	
・新アーキテクチャ 知識情報処理システムからの要求を満たす第5世代コンピュータのアー	・述語論理マシン 推論の基本となり、また自然言語に近い表現力を有する述語論理の計算モデルおよびプロ	・述語論理の開発と実装化 ◦ PROLOG系 ◦ 新言語系 ・基本技術開発 ◦ 並列方式の研究 ◦ 専用的機構の開発		① 拡張 PROLOG言語 知識情報処理システム開発のツールとなりうること。 ② 新述語論理型言語 長期使用に耐えうる完備した仕様をもつもの。	

課題グループ	研究開発課題	研究内容	目標・仕様	備考
キテクチャの研究開発を行う。 [研究開発費： 300億円]	グラミング言語をサポートするアーキテクチャの研究開発。	・述語論理マシンの実現 ①ファームウェア・ベース・マシン 0.1 Mega LIPS ②パーソナル述語論理マシン 0.1～1 Mega LIPS ③並列述語論理マシン 数10Mega～1 G LIPS	③ 述語論理マシン	
	・関数型マシン 記号処理に適し、理論的基盤に基づく、関数型のモデルおよびプログラミング言語をサポートするアーキテクチャの研究開発。	・関数型言語の開発(LISPを含む) ・並列処理記述方式 ・並列計算モデル(データフローモデル) ・ファームウェア・ベースLISPマシン ・記号処理用VLSIとVLSI化 ・プロセッサ間結合方式 ・連想プロセッサ、連想メモリ	① パーソナルLISPマシン 汎用コンピュータ(4 MIPS)でのリスト処理の2～3倍。 ② 並列型リダクション・マシン 汎用コンピュータでのリスト処理の10倍。 ③ データフロー方式関数型マシン 汎用コンピュータでのリスト処理の数100～数1000倍。	
	・関係代数マシン 将来のデータベース・システムの中核と考えられる関係代数をインタフェース言語とし、集合演算等処理するマシン・アーキテクチャの研究開発。	・インタフェース言語と処理系の開発(関係論理、関係代数、マシン基本演算) ・データベース管理システムの開発 ・マシン基本演算用アルゴリズムの研究 ・マシン・アーキテクチャの開発 ・要素プロセッサおよび接続用ハードウェアの開発 ・記憶階層の構成方式と記憶デバイスの開発	・並列処理プロセッサの要素プロセッサ数 ① 中間目標：100台以下 ② 最終目標：数100台以上 ・記憶容量 ① 小容量・高速用：10～100 MB ② 中容量・中高速用：100 M～10 GB ③ 大容量・低中速用：10～1000 GB	・データベース・マシンと密接に関連し、特に関係データベース・システムのサポートに用いる。

課題グループ	研究開発課題	研究内容	目標・仕様	備考
・新アーキテクチャ (つづき)	・抽象データ型マシン 巨大で複雑なソフトウェアのモジュール化をアーキテクチャ側からサポートするための技術としての将来の計算機のメモリ構造やプロセッサの機能の研究開発。	・抽象データ型モデルの整理体系化・言語系の研究開発 ・対象空間から物理リソースへの写像 ・ガーベジ・コレクション ・構造メモリ ・抽象データ型プロセッサ・アーキテクチャ ・並列処理 ・その他 (I/O, 従来言語との関係, OS, DB, 分散処理)	① 100台程度の並列ノイマン型抽象データ型マシンの開発。 ② 1000台程度の並列非ノイマン型抽象データ型マシンの開発。	・述語論理マシンおよび関数型マシンとの密接な関係を持つ。
	・データフロー・マシン 高度な並列処理の実行を目的とし、並列処理を基本的に含むデータフロー・モデルに基づくアーキテクチャの研究開発。	・機械語命令セットの設計 ・データフロー向き高級言語の設計 ・マシンの全体構造の決定 ・結合網の構成 ・構造メモリの実現 ・アクティビティ制御方式の確立 ・OS機能の実現方式 ・障害対策・保護対策 ・従来計算機との融合構造 ・データフローマシン・プロトタイプの開発 ・パーソナル・データフローマシンの開発 ・データベース管理機能との融合	① 初期目標：基本操作レベル プロセッサ16台、メモリ 8MB ② 中間目標：実用規模プロトタイプ プロセッサ100台、メモリ 100MB, 50 MIPS ・プロセッサ向結合網：LSI向き 構造 $10^3 \sim 10^4$ 台収容 ③ 最終目標：超高速データフロー マシン プロセッサ $10^3 \sim 10^4$ 台、メモリ 1 ~ 10GB, 1 ~ 10BIPS ・パーソナル・データフローマシ ン プロセッサ32台、メモリ 10MB, 10MIPS	並列述語論理マシン、並列関数型マシン等と密接に関係する。

課題グループ	研究開発課題	研究内容	目標・仕様	備考
新アーキテクチャ（つづき）	新ノイマンマシン ノイマン型マシンの改良，ならびに高度のVLSI化により，ノイマン型マシンの利点を生かすアーキテクチャの研究開発。	100万トランジスタ／チップ並びに1000万トランジスタ／チップのVLSI新ノイマン・アーキテクチャの研究 - アーキテクチャ・データベースの研究 - マイクロ90（目的指向アーキテクチャ）の開発	① 中間目標：新ノイマンベースの100万トランジスタ／チップ・プロセッサ ② 最終目標：新ノイマンベースの1000万トランジスタ／チップ・プロセッサ	
機能分散アーキテクチャ VLSIを前提とし，漸進的アーキテクチャの側面を重視して新アーキテクチャとVLSIアーキテクチャの融合を具象化するためのアーキテクチャの研究開発を行う。 〔研究開発費：100億円〕	機能分散アーキテクチャ 一貫した理念のもとに，高効率，高信頼で使い易く，構築しやすく，将来の技術改良に対応しやすく，高機能で各種のマシン／システム・レベルに対応できる機能分散アーキテクチャの研究開発。	① 機能分散基礎システムの開発 - 論理モデルの確立 - アーキテクチャ諸方式の確立 - ダイナミック・アーキテクチャ - インプリメンテーション - 専用マシン開発技術 ② 機能分散実験システムの開発 - パーソナル・コンピュータ - 高級言語マシン群 - ローカル・ネットワーク ③ 総合システムの開発		
	ネットワーク・アーキテクチャ 地域的に離れたコンピュータ・システムを疎	- ネットワーク・アーキテクチャの標準化 - プロトコルの記述・作成・検証技術		

課題グループ	研究開発課題	研 究 内 容	目 標 ・ 仕 様	備 考
・機能分散 アーキテクチャ (つづき)	結合するためのアーキテクチャ。第5世代で想定している高速のローカルネットによる分散情報システム実現のためのグローバル・ネットワークとの融合利用技術の研究開発。	・ネットワークOSの実現技術 ・網間接続技術 ・マルチメディア処理技術 ・機密保護機構 ・VLSI技術 ・光ファイバ通信技術 ・衛星通信技術 ・ローカルネットワーク		
	・データベースマシン データベースの処理に適したアーキテクチャを持ち、大容量データベースに高速アクセスできる専用マシンの研究開発。	・データベース（非数値）処理に適した新しいアーキテクチャの研究 ・高機能データベース・マシンの研究 ・マンマシン・インタフェースの研究 ・分散データベースの研究 ・既存DBからの変換あるいはエミュレーション技術の研究 ・VLSI技術により実現可能になる新しいデバイスの効果的な利用法に関する研究 ・データベース・マシン設計のための基礎データの収集・分析 ・実験機の開発 ・実用レベル・マシンの開発	① 実験機 ・容 量：最大100GB ・処理能力：10^3トランザクション/秒 ・データモデル：リレーショナル ② 実用レベルマシン ・容 量：最大1000GB ・処理能力：10^4トランザクション/秒 ・データモデル：リレーショナル （他のモデルのデータベースからの変換やエミュレーションをサポート）	
	・高速演算マシン 実験に替わる数値シミュレーション等のための科学技術計算を高速	・高速論理素子 ・高密度実装技術 ・冷却技術 ・アーキテクチャ（論理仕様）	・新しい高速素子を用いた40～100 MFLOPS 程度のプロセッサ・エレメントの開発 ・4 MFLOPS 程度の性能を有する	

課題グループ	研究開発課題	研 究 内 容	目 標 ・ 仕 様	備 考
機能分散アーキテクチャ (つづき)	処理する専用マシンの研究開発。	- 高速演算方式 - 専用オペレーティング・システム - 高級言語コンパイラ	プロセッサ・エレメントを開発し、これを1000個程度同時に動作させ、全体として1BFLOPS程度の性能を発揮する並列処理方式を開発する。 数10GBの固定ヘッドディスク	
	高度マン・マシン・コミュニケーション・システム 人間向きの対話能力(インテリジェンス)を計算機システムに付与するための技術としての文字、音声、図形・画像入出力システムの研究開発。	- 漢字を含む文字入出力装置 - 図形画像入出力装置 - 音声入出力装置 - 文字・図形・音声の各入出力をインテグレートしたシステムの開発	- 漢字を含む文字入出力装置 ① 中間目標：3000～4000字種の入力機能、4～5種類の異なる字体を持つ表示装置。 ② 最終目標：音声入力との組合せによる漢字入力機能、意味理解を含むカナ漢字変換機能等を付加することを目指すものとする。 - 図形・画像入出力システム ① 中間目標：5000×5000～10000×10000点のタブレット座標入力装置 ② 最終目標：図形・画像応用システムの研究課題よりの仕様に基づいた、より知的機能を有するものとする。 - 音声入出力システム ① 中間目標：500～1000語の識別能力 ② 最終目標：音声応用システムの課題で示す、意味理解を含み、自然言語も一部可能な、より知的な仕様を持つものを目指す。	

課題グループ	研究開発課題	研究内容	目標・仕様	備考
			- マルチメディア入出力機能を持つ総合端末。 - 上記の機能を組合せ、VLSI化を行い、パーソナル・コンピュータ用端末としてインテグレートしたもの。	
- VLSI 技術 - 構成素子からの第 5 世代コンピュータ・アーキテクチャへのアプローチとして、VLSI を最大限に利用するアーキテクチャの研究開発を行う。 研究開発費： 200 億円	- VLSI アーキテクチャ 1990 年頃に登場すると予想される 1000 万トランジスタ／チップ程度の VLSI の特質を最大限に生かすことを目的としたアーキテクチャの研究開発。	- 新アーキテクチャ構築技法（基礎研究） - VLSI デバイス・ルール・ブック設計 QA システム - アーキテクチャ・データベース - VLSI アーキテクチャのための CAD - VLSI アーキテクチャ - 完全 1 チップアーキテクチャ（100 万トランジスタ／チップ、1000 万トランジスタ／チップ） - 機能パーツ・アーキテクチャ - VLSI システム - 機能分割・接続法	① 中間目標：100 万トランジスタ／チップの完全 1 チップ・アーキテクチャ ② 最終目標：1000 万トランジスタ／チップの完全 1 チップ・アーキテクチャ	
	- 知的 VLSI-CAD - 設計のノウハウを蓄積することができ、それを有効利用できる VLSI-CAD 総合システムの研究開発。	- アーキテクチャ・データベース - ノウハウ・データベース - VLSI-CAD 設計問合せシステム - ヒューリスティック・デザイン手法の開発	アプリケーション・デザイナー自身が 100 万トランジスタ／チップの VLSI カスタム・チップを 1 カ月以内にマスク・パターン設計までできること（目的チップが 3 カ月以内に得られること）。	

課題グループ	研究開発課題	研究開発内容	目標・仕様	備考
システム化技術素子、アーキテクチャ、基礎ソフトウェア、応用ソフトウェアの一貫した体系でのシステム化およびシステムの設計、開発、保守・管理のライフサイクルに関する技術を研究開発する。 〔研究開発費：〕 50 億円	知的プログラミング・システム ユーザの要求にしたがってアルゴリズム・バンク（知識ベース）から必要な機能を持つプログラムを取り出し、推論をしながら、要求に合うように合成するとともに、作られたプログラムが要求を最適に満足していることを容易に検証できるシステムの研究開発。	- モジュラー・プログラミングと検証理論の研究 - 仕様記述とプログラムの合成理論に関する研究 - プログラムの検証・合成システムおよびプログラム・ベースの開発 - プログラムの保守・改良・管理システムの開発 - プログラム設計コンサルタント・システムの開発	- プログラムの検証・合成システムおよびプログラム・ベース ① 中間目標：データベースへの検索手続きなど比較的小さく、一定分野のプログラムの合成・変換による改良。小規模なプログラムベースの開発。関数型、論理型およびデータ抽象化プログラムの検証システムの作成。 ② 最終目標：データベース管理システム、言語プロセッサなどの大規模プログラム合成。大規模なプログラムベースの開発。 - プログラムの保守・改良・管理システム ① 中間目標：関数型、論理型プログラムについて、プログラム理解システムを作成。等価変換実験。 ② 最終目標：プログラムの性能評価システムと等価変換による改良システム。プログラム修正システム。 - プログラム設計コンサルタント・システム ① 中間目標：基本設計	

課題グループ	研究開発課題	研究開発内容	目標・仕様	備考
・システム化技術 (つづき)			② 最終目標：自然言語による質問応答。データベース管理システム、データベース応用システムなどの設計コンサルテーションが可能なシステム。	
	・知識ベース設計システム 知識情報処理システムを設計・開発・運用するのに必要な種々の技術と知識を基本知識ベースの中に組織的に所有し、その基本知識ベースに基づいて、知識ベースシステムを構築しようとする者を支援するシステムの研究開発。	・メタ知識の表現方式とその利用技術の研究 ・知識ベース設計開発支援システムの開発 ・知識ベース増殖支援システムの開発	① 相当な専門知識を必要とする高度な問題について、専門家にとっても有意義なコンサルテーションを提供する知識ベースシステムの構築を容易に行えるものとする。 ② 設計する知識ベースシステムの規模は、その知識を規則表現に換算して、約2万規則程度のものとする。 ③ メタ知識レベルで、システムの検証がある程度行え、大型の知識ベースシステムのデバックが容易に行えるものとする。 ④ 中間目標として、設計すべき知識ベースシステムの規模を、目標の3割程度とする。	
	・アーキテクチャ関連システム化技術 第5世代コンピュータをシステムとしてまとめ上げる技術であり、	・仮想化とシステム構成技術 ・融合化と負荷配分の最適化技術 ・大規模システム設計・開発技術 ・超高信頼化技術		

課題グループ	研究開発課題	研 究 内 容	目 標 ・ 仕 様	備 考
システム化技術 (つづき)	アーキテクチャ構築法を中心とし仮想化とシステム構成技術，融合化と負荷配分の最適化技術，大規模システム設計開発技術，高信頼化技術等の研究開発。	アーキテクチャ開発ツールとしてのローカル・ネットワークの開発		
	データベースおよび分散データベース・システム 第5世代としてのデータベース・システムの研究開発と複数の種々のデータベース・システムを統合利用する分散データベース技術，さらに知識ベースシステムを統合利用することを目指したシステムの研究開発。	- データ・セマンティクスとデータ・モデルの研究 - 柔軟構造データベース・システムの - スキーマ設計支援システムの開発 - データ蓄積支援システムの開発 - 自然言語によるQAシステムの開発 - 分散データベース・システムの開発 - 超文字データベース・システムの開発 - データベース・マシンの開発		
開発支援技術 ハードウェア，ソフトウェア，全体システムの開発に適した開発支援用の各種システムの研究開発を行う。 [研究開発費： 100億円]	開発支援システム VLSI-CAD，パーソナル・コンピュータ，コンピュータ・ネットワーク，ソフトウェア/知識ベース開発支援システム等をプロジェクトの前段階で準備する。	- 支援ツールとしてのVLSI-CAD - 研究開発用パーソナル・コンピュータ - コンピュータ・ネットワーク - ソフトウェア開発支援システム - 知識ベース開発支援システム		

2. 研究開発課題

2. 研究開発課題

2.1 基本応用システム

2.1.1 機械翻訳システム

〔1〕 概 要

社会の国際化・情報化の急速な進展に伴い、国際間にわたる各種の情報を迅速に収集分析し、我国の立場を速やかに世界に周知させ、資源や貿易摩擦等にみられる国際的問題の解決を図ることは、貿易国日本にとって、今後の大きな課題になっている。そのためには文書翻訳システムの開発がまず第一に必要である。

本研究で開発するシステムは、これまでの機械翻訳システムの研究・用語集作成等に見られるドクメンテーション技術の研究・知識利用に関する人工知能研究、等の成果を総合し、実際のテキストを翻訳する本格的な機械翻訳システムを目指す。また、これに伴って、機械翻訳の結果を翻訳専門家及びその分野の専門家が、適宜修正し、完全な翻訳テキストを作るための Word Processing 技術の開発も同時に行なう。

以上のシステムを開発するために必要となる自然言語処理理論の開発、大きな用語集・シソーラスを管理するためのデータベース技術の開発、対象分野の知識を体系的に整理し、機械翻訳のために活用するための知識工学的手法の開発、テキスト用データベースの開発、各種の構成要素からなる機械翻訳システムを開発するためのソフトウェア・ツールの開発、高解像度・高速なグラフィック・ディスプレイを用いた本格的なテキスト・プロセッシング技術の開発等を行う。同時に、上記の各開発プロジェクトに必要な高機能ハードウェアの作成を進める。具体的には、自然言語処理を行なうためのマシン、用語集・シソーラス及びテキスト・データに向けたデータベースマシン、テキスト処理用の高解像度・高速なグラフィック・ディスプレイを開発する。このような計算機ソフトウェア・ハードウェアの研究開発の早い時期に、ある特定分野（複数）の用語集・シソーラス等の基礎的な言語データの収集とその計算機化を進め、本研究開発の円滑な推進を図る。

実動するシステムの目標としては、辞書項目数（複数単語からなる用語も含む）10 万を持ち、90%以上の精度で人間の介在なしに翻訳し、対象分野の移行が計算機サポートによって半自動的に行えるものとする。可能な対象分野は、科学技術分野の技術報告書・マニュアル、経済市況等、一般のテキストに比べて、比較的専門用語の占める割合が大きく、用語の概念規定が明確な分野とする。

機械翻訳システムは、基礎ソフトウェア・システム（2.2節）中の知識ベース管理システム（2.2.1節）と知的インタフェース・システム（2.2.3節）をベースにして構成される。

これを図 2.1-1 に示す。

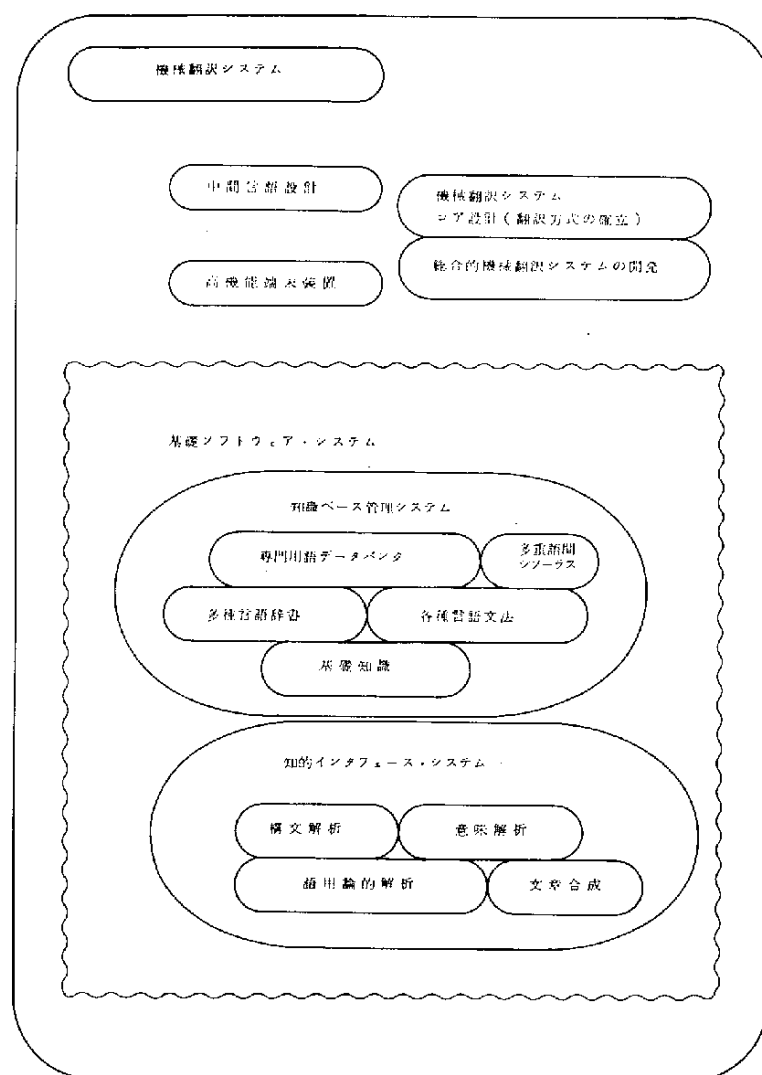


図 2.1 - 1 機械翻訳システム

〔 2 〕 必要性

社会の国際化・情報化に伴って、政治・経済・科学技術など、すべての分野において海外からの大量の情報を速やかに処理し、また、海外に対して同様に大量の情報を速やかに拡散してゆく必要性が増大してきている。しかるに一方、日本語はいわゆる孤立言語であり、各種の外国語を自由に扱える日本人が非常に少なく、また、日本語ドキュメントを直接読むことのできる外国人も非常に限られている。このことが、とくに日本において、言語障壁として、海外との情報交流を著しく阻害している。とくに、社会の国際化は、あらゆる分野で、正式ドキュメント以外の各国語で書かれた資料を、できるだけ速く入手し、それに対応することを要求しているが、日本においては、かろうじて英語が処理できる以外には、例えば、アラビア語・中国語等、これからその重要度を増すであろう諸言語に対して、全くその対応手段を持っていない。逆に、日本語は諸外国において全く未知言語であり、日本国内での諸活動が、諸外国に正当に伝わら

ず、多くのトラブルのもとになっている。

このような状況において、文書翻訳、とくに、日本語から他言語への翻訳、あるいは、英語以外の諸言語からの日本語への翻訳に対する社会的要請は極めて大きい。

以上のような社会的要請と同時に、機械翻訳システムは、人間の情報活動の中心的存在である言語情報の計算機処理という、今後の情報化社会の基幹となるべき技術の発展に対しても大きな貢献となる。Word Processing, Text Processing の技術、及び、言語を媒介とした情報の集積とその処理に関する技術、知識情報処理技術等に、大きな波及効果を持っている。

〔 3 〕 内外技術の現状

(1) 国内技術

自然言語を計算機によって処理しようという研究は、日本語固有の漢字の処理という障害のために、諸外国、とくに米国に比べて遅れていたのが、極く最近までの状況であった。しかしながら、ここ 2～3 年の間に、漢字処理のためのソフトウェア・ハードウェアが急速に進歩し、実用化されるに伴い、より高度な自然言語処理技術の開発への意欲が、各大学・研究所、メーカーの中で急速に高まってきている。

また、とくに機械翻訳についても、EC・カナダ等の多言語社会に比べて、その必要性に対する関心が薄かったが、情報化社会の進展に伴い、世界的規模で多言語社会へと移行しようとしている現在において、その関心は急速に高まってきている。しかしながら、研究それ自体は、現在プロトタイプ的なシステムが各所で開発されはじめた段階で、これを本格的な規模で実験し、実用化するまでには至っていない。

京都大学では、これまでの人工知能的手法を機械翻訳システムに適用し、対象分野を計算機マニュアルに限定した日英機械翻訳システムを開発、実験をすすめている。また、科学技術論文の英文表題を日本語へと翻訳するシステムを開発し、10万文献の表題に適用して、その結果の評価を行っている。

九州大学では、知識・概念構造を中心にした機械翻訳システムの設計を行っていると同時に、各種日本語表現の収集と、その対応する英語の構造の把握を大規模に行っている。

電総研では、自然言語処理用のプログラミング言語 LINGOL を開発、また、これを使って、自然言語の詳細な意味処理を行うシステムを開発しており、人間の知的情報処理と自然言語処理との相互関連について、すぐれた結果を得ている。

また、富士通研究所においても、機械翻訳システムのプロトタイプを完成している。

この他、機械翻訳のための基礎技術となる日本語情報処理については、東芝、富士通、日立をはじめとする各メーカーにおいて、活発に研究されている。

(2) 海外技術

ヨーロッパ EC では、EUROTRA 計画が超国家的プロジェクトとして取り上げられ、本格

的な実用化を目指して、大規模な研究が始められている。この計画が進み、EC内での多言語間翻訳が実用化された場合、EC内の政治経済の発展や、科学技術の向上に与える利益は、はかり知れないものがある。

カナダでは、モントリオール大学を中心とするTAUMプロジェクトが、政府の援助を受けて、大規模な計画を実施しており、気象通報のための英仏翻訳システムTAUM-METEOが実動している。また、より本格的な航空機マニュアル文の翻訳システムTAUM-AVIATIONがほぼ完成している。

仏のグルノーブルでは、CETAプロジェクトの後をうけたGETAプロジェクトを進めており、この計画自体はEUROTRAの中核的役割を果たしている。

また、米国CDCは、カナダのTAUMプロジェクトのソフトウェアを導入し、本格的な機械翻訳システム開発に取り組む計画をかためている。

〔４〕 研究開発内容

(1) 研究開発項目

① 機械翻訳システム・コア部分の設計

機械翻訳の中心となる入力文解析、出力文合成のための基本的アルゴリズムの設計、および両言語間の構造を結びつけるための中間言語の設計を行なう。この際、対象とする言語対が変化しても、システム概念および基本的アルゴリズムを変える必要のない多言語間翻訳のためのシステム概念を確立する。

② 各種言語の文法開発

①で設計されたシステムに従って、入力文解析の手法を確立すると同時に、日本語を含めた複数の言語に対して実際に大規模な文法を作成し、その妥当性を検証する。作成された文法は、機械翻訳のためだけでなく、一般のテキスト処理にも直接適用できる汎用的なものとする。

③ 中間言語の設計

入力文の解析結果を表示するための中間言語を設計し、これと、言語にはよらない知識記述用言語との関連を明らかにし、推論・演繹等の人工知能的な高度機能を機械翻訳に適用する手法を開発する。

④ 生成用文法の開発

単に原言語の意味を忠実に伝えるだけでなく、前後の文脈に最も適合し、かつ、人間にとって理解し易い文体で文を生成するための手法を開発し、実際にいくつかの言語に対して、その文法を作成する。ここでの文法も機械翻訳用に限定せず、データベース・アクセスを含んだ自然言語による応答システム用の文生成システムとして使える汎用性のある文法を開発する。

⑤ 人間の介入を含む総合的機械翻訳システムの開発

実際に翻訳を実行する機械翻訳コアシステムを中心部分に据え、これを取りまく形で人間との interaction を行なうための方式を設計する。このために必要な高機能 Word Processing 技術の開発、および辞書メンテナンス用ソフトウェアを含むサポート・ツールの開発を行ない、総合的な機械翻訳システムを開発する。

⑥ 専門用語データベース（知識ベース）の開発

機械翻訳システムを現実のテキストに適用し、大規模な実用化を行なうために、テキストに含まれる用語についての網羅的な辞書が必要となる。ある特定の専門分野（複数）をとって、そこでの用語を収集し、大規模な専門用語データベースを開発する。このデータベースは、各用語の各言語での訳語を持つ多言語データベースであるとともに、各用語の用例、用語間の相互関連、意味を記述した知識ベースでもある。これら複合的で、かつ、複雑な構造を持ったデータを格納し、有機的に検索・利用するためのシステムも同時に開発する。ここで開発されたデータベースは、機械翻訳システムだけでなく、人間の翻訳作業にも使用され、翻訳の質を向上するのに使われる。また、一般の知識ベースとしても機能する。

⑦ 専門用語データベースのためのマシンの開発

上記⑥の専門用語データベースは、知識ベース的機能と、大規模なテキスト的データとを管理する機能とを、同時に持ち、かつ、高速に処理できなければならない。これを実現するために、従来のデータベースマシンの技術をさらに高度化した、専門用語・知識ベース用のマシンを設計、開発する。

⑧ 高機能ワードプロセッシング技術の開発

翻訳結果の人間によるエディティング作業を容易にし、図や絵も含めたテキスト全体の編集や印刷用原版の作成を行なうために日本語およびその他の言語テキストを自由に操作し、図・絵等のテキストに含まれる非言語的情報の管理を行なう高機能なワードプロセッシング技術を開発する。

(2) 研究開発スケジュール

研究開発項目 \ 年度	1	2	3	4	5	6	7	8	9	10
(1) 機械翻訳システム・コア部分の設計	設 計		プロトタイプシステムによる実験					評価・改良		
(2) 各種言語の文法開発	各種言語の文法試作			大規模化				高 度 化		
(3) 中間言語の設計	中間言語の研究		システム設計		開 発		評価・改良			
(4) 文生成用文法開発	文生成用文法の研究		システム設計		開 発		評 価			
(5) 総合的機械翻訳システムの開発	人間の介在方式研究		小規模実験システム試作			総 合 化		評価・改良		
(6) 専門用語データベース開発	文・用語データベース試作		多言語間シソーラス試作		大規模化、 意味記述辞書の作成、 用例集の強化			評 価・改 良		
(7) 専門用語データベースマシンの開発	マシンの設計		マシンの試作				評 価・改 良			
(8) 高機能端末装置	表示機能の高度化		基本ソフト開発		高機能ワード プロセッシング用 ソフトの開発		実用機 試作		評 価・改 良	

(3) 研究開発の目標・仕様

① 国際化・情報化社会が急速に進展する状況において、言語テキストの翻訳に対する需要は、今後急速に増大する。この需要の増大に応えるために、人間援助型の本格的機械翻訳システムを開発する。

② 開発されるシステムの仕様は以下の通り。

○ 中間目標としては、小規模実験システムとして

使用語彙 5000 語で人間が介在するシステムを開発する。

精度は 80% とするが、実験のし易さを考えたシステムを設計する。

○最終目標は：

- 2-1. 使用語彙数10万語。
- 2-2. 90%の精度で、翻訳し、人間の介在によって、残り10%を処理する。
- 2-3. テキストの編集、翻訳結果の印刷までの各行程で計算機が関与する総合システムとする。
- 2-4. 全コストは、人間が翻訳した場合の30%以上とする。

2.1.2 質問応答システム

〔1〕 概 要

質問応答システムは、システム使用者がシステムに対して質問を出し、これに対してシステム側が適切な応答（これには動作も含まれる）を行うことが基本である。使用者の出した質問内容に不明確なところがあれば、適当な時期に適切な問い返しを行い、最終的に使用者の意図に沿った問題解決を行ったり、それを支援するシステムである。したがって8分科会で提案、

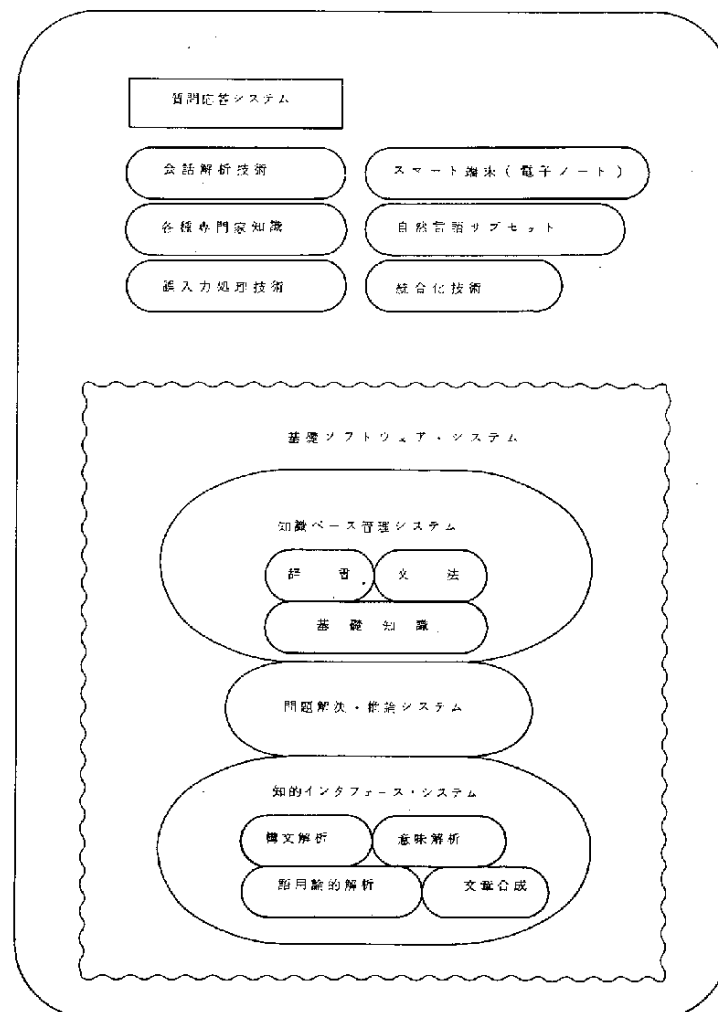


図 2.1 - 2 質問応答システム

検討されている知的 CAE/CAD システム、DSS（意思決定支援システム）、知能ロボットや、知的 CAC/CAI システムも、それぞれシステムとしての力点の相違はあるが、システム使用者とシステム相互の間に、質問応答のサイクルが含まれており、広義の意味での質問応答システムであるといつて良い。

基本応用システムとしての質問応答システムは、これらの質問応答システムに共通な問題を取り上げ、研究開発を行うものである。したがって、ここで開発されるシステムを中核にして、個々の問題領域に依存した質問応答システムが速やかに設計、製作が可能でなければならない。

第 5 世代コンピュータのなかで、質問応答システムの果たす役割は、システム使用者に柔軟な対話機能を与え、知識を駆使した高度な質問応答を行う能力をコンピュータに持たせることである。この質問応答システムは、基礎ソフトウェア・システム中の知的インタフェース・システムをベースに構成されることになる。さらに、知識ベース管理システムを用いて辞書、文法、その他の基礎および専門知識を作成し、それを問題解決・推論システムを用いて利用しながら、質問の処理が行われるので、基礎ソフトウェア・システムを統合化したシステムとして、質問応答システムが実現されることになる。（図 2.1 - 2）

〔2〕 必要性

第 5 世代コンピュータは、「使い易いコンピュータを実現する」という問題意識から発想されている。ここで「使い易いコンピュータ」とは、一般の使用者にとって「使い易い」ということであり、特定の専門家にとってのみ「使い易い」ということを意味しない。

現在のコンピュータを使用する場合に、使用者が困難さを感じる最大要因は、使用言語の相違に基づくギャップであろう。使用者が素人になればなるほど、このギャップは大きくなり、現在のコンピュータは、「使い易いコンピュータ」から遠くなる。

第 5 世代コンピュータの一部として研究開発される質問応答システムは、このような使用言語の相違に関するギャップを解消する必要がある、柔軟な会話機能、すなわち自然言語や音声、図形による会話機能をもたねばならない。

図 2.1 - 2 に示したように、質問応答システムは、基礎ソフトウェア・システムを中核に作成され、柔軟な会話機能と高度な推論能力を持つシステムであり、第 5 世代コンピュータの一部として「使い易いコンピュータ」実現のための大きな役割を果たすことになる。

〔3〕 内外技術の現状

（1）国内技術

我が国では、基本応用システムの一つとして取り上げられているような、柔軟な対話機能と高度な推論機能を有する質問応答システムを目指す試みは少なく、立ち遅れているといわざるを得ない。小規模実験用のシステムとして、電総研の VISUALIZER、電電公社武蔵野通研の MSSS 78 や音声理解システム作成の試みがある。前者は、積木の世界を対象にした文理

解の結果を作画したり、質問応答することができるものである。後者は、MITのSHRDLUと同様、文理解の結果を（コンピュータ内にシミュレートされた）ロボットの動作として表示したり、算術文を理解し簡単な計算を行うものである。九大の田町研究室では、気象文を理解して、気象図を作成する研究を進めている。いずれも実用のレベルには到っていない実験システムである。一方、日本IBMでは、実用を意識した立場から、日本語によるデータ・ベース検索システム「やちまた」を開発した。しかし、これもシステム構成法や応答速度等の問題があり、実用に供されるに至っていない。

(2) 海外技術

海外では、1970年代初頭のMITの研究、特に南米の地理についての知識を学生に教授するSCHOLAR、ロボットに指令を与え積木操作をさせるSHRDLシステムが著名である。前者は、知識表現の問題と、デフォルト推論の問題に重点がおかれ、入力英文処理は、さほど重要視されていない。後者は、入力英文処理と問題解決推論技術の開発に力点が置かれ、対象領域が制限されれば、かなりの程度の質問応答システムが作成可能なことを実証し、その後の研究を大いに刺激した。

米国のBBN研究所では、回路図を与え、それに故障要因を付加し、学生に故障を発見させるプログラムSOPHIEや、月の岩石分析用の質問応答システムLUNARを開発している。イリノイ大学のPLANESは、海軍の空軍機データベースに対する質問応答を行う。XeroxのPARC研究所では、旅行計画作成システムGUSが、ダートマス大、SRIではデータベースに対する質問応答システムROBOT、LIFERが、IBMのサンノゼ研究所では、同様なシステムRENDEVOUSが開発され実験が進められている。最近では、実験システムの作動結果を分析し、本格的な質問応答システムを作成しようとする動きもある。

西独では、政府の援助の下に、シーメンス社の質問応答システムCONDOR等が作成されている。

〔4〕 研究開発内容

(1) 研究開発項目

① 会話解析技術

柔軟な質問応答を行うためには、システムと、システム使用者との間で交わされる会話が、質問の意図に沿ったものであって、会話に首尾一貫性があるものでなければならない。そのためには、質問応答システム側が、

- (i) 会話の主題／焦点を抽出し、質問意図の理解を行う。
- (ii) 常識的な判断を行い、会話の背後に隠れている事実を顕在化する
- (iii) 会話の主導権を、システムもしくはシステム使用者のいずれも握ることが可能
- (iv) 会話理解のための適切なメモリ・モデルの設定

等を可能にする諸技術を開発する。

② 図形入出力装置

質問応答システムの端末としては、入出力の形態として、漢字を含む自然言語、図形、音声等が考えられるが、色図形の出力はもちろん、入力図に濃淡配色する機能をももつ Xerox PARC 研究所で構想されている Dynabook を高度化したスマート端末を開発する。これは、A 4 ～ B 4 程度の画面の大きさを持ち、小型、軽量で、通常のノートと同程度の簡便さで漢字や図形の入出力と、表示画面のハードコピー化が可能なものでなければならない。

③ 専門家知識作成技術

これは、基礎ソフトウェア・システム中の知識ベース管理システムで開発される知識表現、知識獲得方式に沿って、専門家知識を作成するための用具を開発するものである。これにより、専門家知識の更新、削除を容易に行うことが可能になる。究極的には、これは質問応答システムそれ自身により置き換えられる。知識ベース設計システムの成果を大いに利用する。

④ 誤入力処理技術

質問応答システムの使用者は、プロのみでなく素人も含まれる。したがって、あらゆる使用状況を設定して、それに対処しなければならない。特に大きな問題となるのは、誤入力の処理である。これには、質問文の打ち誤りのレベルから、システム側からの質問意図を、使用者側が誤解するレベルまで、さまざまなものが考えられる。システムは、これらの誤りに対して十分堅牢でなければならない。そのために、誤りの要因を分析し、それに対処する技術を開発する。

⑤ 自然言語サブセットの設計と文法および辞書の開発

各種問題（専門家）領域に対応した質問応答システムを作成する場合、システムとの会話に用いる自然言語は、一定のサブセットに落ち着くと考えられる。それらを各問題領域毎に設計する。またそれらの文法（辞書）を開発する。

⑥ 統合化技術

上記①～⑤の研究開発成果と、基礎ソフトウェア・システム中の知識ベース管理システム、問題解決・推論システム、知的インタフェース・システムを用いて開発される。辞書、文法、基礎知識と、構文解析、意味解析、語用論的解析、文章合成技術を統合化して、質問応答システムのプロトタイプを作成する技術を開発する。

このプロトタイプは、知的プログラミング・システム等の開発に利用される。

(2) 研究開発スケジュール

研究開発項目 \ 年度	1	2	3	4	5	6	7	8	9	10
(1) 会話解析技術	会話データの収集と解析			会話解析方式の研究, プロトタイプの試作				改 良		
(2) 図形入出力装置	スマート端末プロトタイプ試作			改 良						
(3) 専門家知識	専門家知識収集			知識ベース化			改 良	応用分野の多様化		
(4) 自然言語サブセット	自然言語サブセット作成			文法, 辞書の作成			改 良	応用分野の多様化		
(5) 誤入力処理技術	誤入力例の解析			誤入力処理方式の研究, システム作成			改 良			
(6) 統合化技術				第 I 期システムの試作			評 価	質問応答システムプロトタイプの試作		

(3) 研究開発の目標・仕様

① 5年後の中間目標として、一特定専門分野に限定した質問応答実験システムを試作する。

- (i) 使用語彙 2000 語 (日本語)
- (ii) 曖昧さ解消のための補助的情報を使用者が与える
- (iii) 推論規則数 1000 個

② 中間目標で試作した質問応答実験システムを評価し、各種専門分野に対する質問応答システム・プロトタイプを作成する。

- (i) 使用語彙 5000 語以上
- (ii) 推論規則数 10000 個以上

2.1.3 音声応用システム

[1] 概 要

音声によるコンピュータとの対話が、使い易い形で実用化されると、一般産業機械用やオフィス・オートメーションを始め、一般家庭まで幅広く普及することが予想される。基礎ソフトウ

エア・システムの中の知的インタフェース・システムにおける自然言語・音声系で述べられた音声理解システムが開発されると、その幅広い応用の中で、共通した基本的な応用システムがいくつか考えられる。本節では、機械翻訳の入出力として用いられる汎用音声応答マシン、音声タイプライタ、銀行や電話を用いた問い合わせ等で重要な話者認識システム等について説明する。

〔2〕 必要性

コンピュータ等への入出力媒体には、音声、キーボード、オンライン手書き文字等がある。工場等の場で、作業者が、コンピュータへ入力するとき、一般にデータ収集者はデータシートに記入し、別の作業者がカード等にキーパンチを行って入力される。一方音声を用いると、データ収集者が直接マイクロホンに発声するだけで、コンピュータへ入力として蓄えられ、オンライン処理が可能になる。また、データの発生速度も、音声入力の方が他の媒体よりも2～4倍速い。音声タイプライタは、音声入力の一つの典型例として取り挙げるもので、自然に発声した連続音声はコンピュータへ入力され（もしくは専用装置によって）、日本語の場合は漢字かなまじり文等でプリントアウトするものである。

音声出力は、記録として保存する場合は別にして、出力内容を理解する上で、出力文章を読むよりも容易で使い易いものとなる。また、電話による応答では欠かせないものである。機械翻訳でも翻訳文の音声出力が有用であり、さらに同時通訳作業では必須のものとなる。このような音声出力の基本応用として音声応答システムを開発する。

話者認識は、身分の確認手段として、預金残高の問い合わせ等の個人情報の検索や各種機密情報の検索、建物への入出時の身分の確認等で用いられることが考えられ、幅広い応用があって社会的ニーズが高いため、基本応用システムの1つとして話者認識システムを開発する必要がある。

〔3〕 内外技術の現状

知的インタフェース・システムの内外技術の現状で述べたように、実用的な音声応用システムとしては、数百単語を識別する音声認識システムや、限られた文章や単語を発声する音声出力システム等が、国内、国外に数例見られる程度である。このような単純なものはこれからも広く普及していくと考えられるが、対象語彙を十分に多くした使い易い応用システムを構成するには、基礎的研究が未だ不十分である。知的インタフェース・システムとしての「音声理解システム」の研究開発の成果を今後利用して、基本応用システムは開発される。

〔4〕 研究開発内容

(1) 研究開発項目

① 音声タイプライタの開発

音声理解システムの研究開発の成果が応用されて音声タイプライタとなる。普通に発声した文章が活字となって打ち出されるシステムであるが、前期は、意味情報等の高度な言語情

報は用いずに、簡単な構文規則と音形規則を利用して単純な文章を認識対象とする。その時、初期誤りを人間が訂正するための能率良い端末が必要とされる。中期・後期は1万単語程度を対象とし、簡単な意味や語用論的情報を用いて、音声認識の誤りを自らある程度修正し、全体としてわかりやすい文章として出力する。

② 音声応答システムの開発

システムに対する質問を理解し、その答を音声で出力するものである。構成は、音声理解部、日本語処理・解答作成部、音声出力部からなり、前2者は他の研究開発の成果を利用してシステムに組み込む。音声出力部は、知的インタフェースの項目で開発した音声出力方式に基づき、分析合成方式の開発や、規則による合成方式の基礎的研究の成果を用いる。前期は対象語彙や分野を小さく限定し、自然な合成音作成技術を開発する。中期・後期は応答内容の意味を把握し自然な会話となるように、イントネーションやアクセント等を適切に取り入れ、対象語彙を1万単語程度に増加する。

③ 話者認識システムの開発

出入管理のように、話者が特定の人物と同一人であるか否かを確認する話者照合と、犯罪捜査のように、話者がある集団の中のどの話者であるかを判定する話者識別がある。話者認識を実用化するには、その認識精度が厳しく要求されるため、特に話者認識用の特徴パラメータの抽出等の基礎研究が重要である。前期は基礎研究および数10名程度の話者を対象としたシステムを作り、中期・後期では数百名程度の対象話者が拡大し、認識時間の縮小を図る。

(2) 研究開発スケジュール

研究開発項目 \ 年度	1	2	3	4	5	6	7	8	9	10
(1) 音声タイプライタの開発	基礎研究			単純文を対象とする音声タイプライタの開発				意味情報の導入		改良
(2) 音声応答マシンの開発	基礎研究			分析合成方式、限定語彙音声応答マシンの開発				規則による合成方式の導入		改良
(3) 話者認識システムの開発	話者認識方式の研究			限定対象話者認識システムの開発				実用的システムへの高度化		改良

(3) 研究開発の目標・仕様

基本的には、知的インタフェース・システムにおける「音声理解システム」の研究の進捗状況に応じた目標を立てるのが望ましい。

① 音声タイプライタ

前期は、数百～数千単語に簡単な構文情報を利用したシステムとする。中期・後期は、単語数を増やして1万単語程度とし、同時に意味解析を加え、音声認識上の誤りを自ら修正し、全体としてわかり易い文章を出力する。

② 音声応答システム

前期は対象語彙を数千程度とし、分析合成方式を中心とする。中期・後期は、対象単語を1万程度として応答内容の意味を把握し、自然な会話となるように高度化する。

③ 話者認識システム

前期は数10名程度の話者とし、中期・後期に数100名以上の話者に対して実用的な認識時間のシステムを開発する。

2.1.4 図形・画像応用システム

〔1〕 概要

図形・画像情報を構成的に蓄積し、知識情報処理に利用するために効果的に検索する技術を確立する。

知識情報処理システムの開発の狙いは、使い易いコンピュータ・システムを提供することにあると同時に、人間の科学的思考を助長するような知的システムを提供することにもある。

そのために、人間が科学的思考に用いている種々の知識をコンピュータへ蓄え、それに基づいてコンピュータに推論や問題解決を行なわせることによって、人間を知的に支援するシステムの開発を目標にしているのである。

人間の思考過程における図形や画像の役割が非常に大であることは明らかである。人は建物や機械を設計する場合には、図面にそのアイディアを描いて他人とコミュニケーションしながら全体をまとめる作業を行う。医者はX線写真を手掛りに患者の病気の診断を行う。われわれの周囲に氾濫する文書にも図や写真が載っていて、読者の理解を助けている。

特に図形や画像を蓄積しておいて利用する場合の効果が大きな場合がある。たとえば、X線による診断に関する医学の教科書を見ると、多数の症例のX線写真に基づいて、その診断手順が記述されている。秀れた教科書を書くためには、良い症例の蓄積が必要であることが判る。

また最近では、気象衛星からの写真がテレビの天気予報で見ることができるようになり、一般の人でも雲のパターンと天気との関係に関心を持ち始めたようであるが、過去のいろいろな気象現象を写した衛星写真と、数値予報の結果とを組合せて予測を展開するといった作業におい

ても、画像の蓄積は重要である。

蓄積する図形や画像の量が多くなると、検索の際にいろいろ考慮しなければならない問題が生じる。効率の問題である。これなどは、図形や画像を抽象レベルで記述し、構造化して蓄積することによって解決できよう。また検索の速度やその出力の品質なども、高い性能が要求される。

以上のような応用分野を持った図形・画像応用システムを開発するためには、基礎ソフトウェア・システム中の知的インタフェース・システムの項で挙げた図形・画像系の研究成果をふまえ、このシステムに特有な技術の開発に重点を置いて課題の設定を行なう。

まず基本的な技術として、図形・画像情報の蓄積・検索方式の研究を行う。図形・画像に固有の構造的記述法を中心に、蓄積・検索のアルゴリズム、品質も高くかつコンパクトな蓄積方式、抽象レベルの取扱いなどの具体的な課題を取り上げる。

検索を行うために都合のよい検索用言語の開発も大切である。これは図形や画像の構造に適合した言語体系を必要とする。その検索を高速に能率よくハードウェアで行なう目的で、図形・画像データベース・マシンの開発を行なう。

最終的にまとまった図形・画像応用システムのシステム化を図る技術进行研究。そこでは、本プロジェクトの他の課題で開発された技術を総合的に用いることになるが、特に基礎ソフトウェア・システムの中の知的インタフェース・システムで開発される技術および基本応用システムの中の質問応答システムで開発される技術が主なものである。

〔2〕 必要性

社会の広汎な分野で、図形・画像を情報処理に取り入れたいという要望がある。その最も単純な形態は図形・画像の伝送である。他方、図形・画像処理技術の高度化に伴って、それを高度な情報処理システムの中で利用したいという要望も高まりつつある。

たとえば天気予報のための気象衛星からの画像や診断のための医用画像などは、過去の事例を画像の形で蓄積し、新しい事例から次の現象や新事実を推定しようとするものである。またCADでは図形を介して人間の思考を支援するシステムを目指している。これらはいずれも図形や画像を構成的に蓄積し、うまく検索するシステムを必要とする。

また膨大な図形・画像データを取扱うには、データそのものではなくて、抽象レベルの取扱いが必要になってくる。そのための図形・画像の理解の機能も必要になる。

このような図形・画像情報蓄積・検索システムは、人間の高度な知的活動を強力に支援するものであるとともに、その知的活動の結果の蓄積を可能にするものであり、知識情報処理システムの格好の応用システムである。

〔 3 〕 内外技術の現状

(1) 国内技術

コンピュータにより画像を処理しようという研究は、国内においても近年広く行なわれているが、現在のところ図形・画像理解のためのいくつかの処理方式が提案されたにすぎず、その方式が本当に複雑な画像の理解に利用できるかどうかわかっていないものが多い。

京都大学では、画像理解のための処理を構造化する研究を行った。

電総研では、対象物体の知識を画像の解釈に利用したシステムを開発した。

シーンの記述法の研究が大阪大学等で行なわれている。

(2) 海外技術

米国では、1975年から始ったARPA(Advanced Research Project Agency)の「Image Understanding」のプロジェクトで、画像理解の研究が行なわれつつある。

スタンフォード大学では、物体の認識に三次元モデルを用いた研究が行われている。

MITでは対象の記述に、詳しさによるモデルの階層構造を採用する方式を提案している。

現在SRI, CMU, USC その他多くの企業の研究グループが画像理解のプロジェクトに取り組んでいる。

〔 4 〕 研究開発内容

(1) 研究開発項目

① 図形・画像情報蓄積・検索方式の研究

図形・画像情報の蓄積とその検索を効果的に行なうためには、それらの情報が蓄積と検索に適した形態で扱われなければならない。そのための基本的技術を、図形・画像の構造的記述法に求める。

基礎ソフトウェア・システム中の知的インタフェース・システムの研究開発項目として取り上げている図形・画像の構造化技術の研究を基礎に踏まえ、応用へ発展させる技術の研究を行なうものである。たとえば図形・画像のソート技術なども、この構造的記述法に基づいて行なう方式を開発する必要がある。

次に、図形・画像を物理的なファイルへ蓄積する場合に必要な情報の圧縮、復元、あるいは雑音処理の技法の開発も行う。

この図形・画像情報蓄積・検索システムは知識情報処理システムの一部を成すものであるから、特に図形・画像に付随する各種の知識を、図形・画像のデータと有機的に使える形で知識ベース化する方法を検討しなければならない。これは図形・画像の構造的記述法とも密接に関連するものである。

② 図形・画像情報検索用言語系の開発

図形・画像それ自身、コミュニケーションの媒体であるが、図形・画像の検索の場面を考

えると、言語の助けを借りると具合のよい場合がある。

たとえば、人間は、図形や画像の全体あるいは一部分に着目し、その特徴を表わす何らかの語彙とそれによる表現系を用いて、その図形や画像をコミュニケーションの中とか、思考のプロセスの中へ取り込むことをやっている。

従って、ここでは、図形・画像情報を検索するという立場から、そのための言語系の開発を行なう。認知科学的アプローチも重要な役割を果たすものと思われる。

③ 図形・画像データベース・マシンの開発

①の図形・画像情報蓄積・検索方式の研究に基づいて、そこで開発された蓄積・検索手法を、高速で実行し、かつ十分実用に耐えられる規模でシステムを実現するのをハードウェア的にサポートする必要がある。

その場合の一つの要になるのは、この図形・画像データベース・マシンである。

図形・画像情報は、適宜に構造化されて蓄積・検索されるから、まずその構造を整理して高レベルの汎用性のある処理系が受持つべき部分と、データベース・マシンとして専用に設計されたハードウェアが受持つべき部分とを分離しなければならない。

これには、アーキテクチャの技術レベルとのバランスを検討する必要があるだろう。

図形・画像ファイルとして、電子化されたファイル、イメージのままのファイル、あるいはそれらのハイブリッド方式などの可能性について、多角的な検討のもとに設計開発を行うものとする。

④ 図形・画像情報蓄積・検索システムの開発

基礎ソフトウェア・システムの知的インタフェースで開発される技術と、本項目で開発する図形・画像情報蓄積・検索に固有な各種技術とを組合せた、システム化技術の開発を行なう。

技術的には、質問応答システムの技術並びに音声を用いた自然言語の技術を取り入れた、心地よい検索システムの開発を目指す。

(2) 研究開発スケジュール

研究開発項目 \ 年度	1	2	3	4	5	6	7	8	9	10
(1) 図形・画像情報蓄積・検索方式の研究										
	構造的記述法(I)			構造的記述法(II)				構造的記述法(III)		
	実際の基礎技術(I)			実際の基礎技術(II)				実際の基礎技術(III)		
	知識ベース(I)			知識ベース(II)				知識ベース(III)		
(2) 図形・画像情報検索用言語系の開発										
	認知科学的基礎研究(I)			認知科学的基礎研究(II)				認知科学的基礎研究(III)		
	構造と語彙(I)			構造と語彙(II)				言語系開発		
	Key 語の整備									
(3) 図形・画像データベースマシンの開発										
	第 1 期設計開発						第 2 期設計開発			
(4) 図形・画像情報蓄積・検索システムの開発										
	基本設計試作			中間目標システムの試作と評価				目標機システムの開発		

(3) 研究開発の目標・仕様

図形・画像情報を構造的に蓄積し、知識情報処理に利用するために効果的に検索する技術を確立する。

目標性能は次のとおりとする。

- ① 検索の対象となる図形・画像データベースの図形・画像枚数は 10 万枚程度。
- ② 抽象レベルの記述を含めた図形・画像格納所要時間は数秒以内とする。
- ③ 図形・画像検索所要時間は、平均 100 m sec 以下とする。
- ④ 中間目標値として、取扱う図形・画像の枚数 1 万程度、処理速度は目標値の 5 割程度とする。

2.1.5 応用問題解決システム

〔1〕 概 要

“問題”の記述を入力すると、その“解答”を出力する。しかも、その問題をできるだけ一般的で高度なものを扱えるようにする。このような問題解決機（システム）の研究は、人工知能研究の中心テーマであった。しかし、扱う問題を一般化すると解決能力は低くなり、解決

能力を高くしようとすれば、対象の特性を分析し、その結果を最大限に利用しなければならないというのが通常の結論である。

ニューエル等の一般問題解決機 (General Problem Solver) に端を発したこの研究は、まもなく、大きく2つの流れとなる。一つの流れは、プログラム言語として、その一般性を保証しようというもので、パターン照合と状態空間の探索を基本機構とする言語が提案、製作された。この流れは、定理証明器、人工知能向言語等を経て、PROLOGという言語に結論付けられつつある。これを充実、発展させるのは基礎ソフトウェア・システムの問題解決・推論システムの中心課題である。

もう一つの流れは、代表的な問題解決を対象に、人間の能力に勝る、あるいは肉薄する位のシステムを作ろうというものである。この流れに、本研究・開発課題は属する。特に対象は、すでに定式化や体系付けがなされているものが選ばれる。最近の知識ベースに基づく専門家システムが、あまり定式化されていないものを対象に、多量データと浅い推論を旨とするに比べ、ここでの対象は、非常に深い推論を必要とするものである。対象の代表例が、数式処理とパズル・ゲームプレイングである。他にプログラムを作る、プログラミングという分野があるが、これは、知的プログラミング・システムの課題である。数式処理システムは、MACSYMA、REDUCEに代表され、人工知能研究の実用化の成功例として良く上げられる。ここでは、公式や法則を知識ベース化し、数式で表わされる知識の一般的な表現システム、すなわち数式理解システムとして研究・開発する。

パズル・ゲームプレイングは、パズルやゲームを何にするかで多種多様であるが、ここでは一番難しいものとされている碁ゲームを取り上げる。碁は、その桁はずれの複雑さのため、単純な探索を高速に行なうということでは、とうてい対処しきれず、局面に対する様々な知識 (定石等) を蓄え、利用するというシステムにしなければならない。問題解決システムとして、非常に一般的で高度なものとなる。

〔2〕 必要性

数式理解システムも碁プレイング・システムも、人間かたどりついた分野としては、非常に一般的なものを対象とする。それらを深く研究し、実際に活用できるシステムを開発することは、広く各種応用システムの基本機能が得られるばかりでなく、基礎ソフトウェア・システムが備えるべき要件に対する多くの知見を与えてくれる。

① 数式理解システム

色々な量の間関係によって表わされる知識を表現するために、人類が永い年月をかけ洗練してきたのが数式である。その性質、数式の意味は、数学の体系として非常に整理、形式化されている。したがって、非常に高い解決能力を持ったシステムを作ることができる。さらに数式で表現される知識は、我々の日常生活で使うものでも、非常に多い。常に目に入っている世

界は、ニュートン力学の法則に従って構成されている情景である。その力学の法則は数式によって表現される等々である。MACSYMA等の数式処理システムは、現在の数値計算に置き換わる未来の計算システムとして、高度な数式の操作を目的として研究・開発された。したがって、一般的な知識表現システム、あるいは問題解決システムとしては不十分な点が多い。

MACSYMA等では、速度を上げるため、法則や公式はプログラムとして、システム内に書き込まれている。これらを知識ベースとして分離し、法則や公式の追加、変更を容易にし、現在のシステムが不得意とする不等式の処理や方程式の処理をさらに高度にする。すなわち、数式理解システムとして高度化すれば、非常に高い利用価値を持つようになり、各種の質問応答システムのサブ・システムとして非常に有効に利用されるようになる。

② 基プレイング・システム

パズルやゲームは、問題自体は、簡単に厳密に記述されるが、それとは裏腹に、その解法は非常に複雑になる。問題解決システムとしては、高度な技法を駆使しなければならない。特に高級なゲーム、チェス、将棋、囲碁は永い競技の歴史を経て、正確なランクづけがなされ、能力の評価が厳密に行なえる。いわば、人工知能システムのベンチ・マークと考えられる。

ゲームが高級になればなる程、様々な高度な問題解決の技法を考案しなければならない。しかも、その技法は、一般的なものをより含むようになり、他の各種の応用システムにも適用できるようになる。そこで、人類が考えついた最も高級なゲームといわれる碁をとり上げる。現在、チェスは、米国等での研究により、人間の高段者と競い合えるレベルに達している。その成功は、解法アルゴリズムは単純なものとし、超高速計算機の力によって探索を高速化して得られたものである。ゲームの複雑度を、可能なプレイの生じ方の回数で表わすと、チェスと碁では、控えめにみて、 5^{50} 、 100^{100} 通りとなる。原理的に、チェスの様に単純な探索を高速化することでは、碁のシステムを作ることとはできない。戦略とプランの考えの導入が本質的に必要となり、定石や盤面パターンの辞書化、さらに知識ベース化が必須となる。したがって、システム全体は、非常に一般的な構造を持つことになり、得られる成果は、広く利用できるものとなる。

〔3〕 内外技術の技術

(1) 国内技術

① 数式理解システム

MACSYMAやREDUCEを土台として、数式処理システムとして、高度化し、新しいアルゴリズムを追加したり、実装の方法を改良したりする研究が4～5年前から本格化した。特に理研、東大で活発に行なわれている。他の人工知能システムと結びつける試みは、あまりなされていない。

② 碁プレイング・システム

碁碁の実質的な本拠地でありながら、本格的な碁碁プログラムの研究は、欧米に比べ遅れている。最近、電総研等で研究に着手され、独自の成果が得られるようになった。とかく、我国では、碁碁等は遊びであるとし、正規の情報処理研究よりはずれるとする風潮がある。そのような考え方こそが、自由な発想を封じ、研究全体を魅力無きものとする。

(2) 海外技術

① 数式理解システム

MACSYMAやREDUCEは、10年以上にわたる地道な研究によって、米国において得られた成果である。MACSYMAはMIT、REDUCEはUtah大学で作られ、現在、着実に利用者の輪を拡げつつある。また、システムとしての一般化をねらったものとして、述語論理の定理証明器に数式処理用の推論規則を組み込んだシステムも研究・開発され、プログラムの検証システムとしても利用されている。代表的なシステムは、Texas大にある。最近、Edinburgh大でPROLOG上に実現されたシステムは、数式理解システムとしての体裁をととのえつつある。

② 碁プレイング・システム

4, 5件のシステムが研究・開発されている。最近では、Reitman等が、碁碁をパターン認識及びパターンに基づいた推論活動とみてプログラムを作成した。その技量は、27級と評価されている。

[4] 研究開発内容

(1) 研究開発項目

① 数式理解基本方式の研究

新しい数式処理アルゴリズムの研究と開発、法則・公式の知識ベース化の方式の研究等を行なう。一般の数式処理システムでは、問題の記述言語は、通常のプログラム言語の形をしている。しかし、その仕様は決して満足のものではない。基礎ソフトウェア・システムの問題解決核言語の仕様に合わせ、高度化する研究も行なう。

② 数式理解システムの開発

不等式や方程式に対する処理アルゴリズムや、その他新しいアルゴリズムが組み込まれ、知識ベース化された総合的な数式理解システムを開発する。この時、各知識ベースやシステム自体の内部構造を整理して作成し、他のシステムの有効利用を促進するようにする。

③ 碁プレイング基本方式の研究

基本的な戦略やプランの研究、局面パターンの認識方式や評価関数の研究を行なう。同時に、定石データ・ベースや棋譜データ・ベースを作成し、基本データの収集と体系化を行なう。

④ 基プレイング・システムの開発

基本方式の成果に基づき、本格的な高能力基プレイング・システムを開発する。開発に際しては、知識ベース、システム自体の内部構造自体等を整理し、他のシステムで大いに利用できるようにする。例えば、質問応答システムと結び、囲碁指導システム等が作成できるようにする。また、専用の入出力機器等のハードウェアの試作・開発も行なう。

(2) 研究開発スケジュール

研究開発項目 \ 年度	1	2	3	4	5	6	7	8	9	10
(1) 数式理解基本方式の研究	処理アルゴリズム, 公式ベース, 等, 基 本方式の研究			公式ベースの整備と, 記述言語の高度化						
(2) 数式理解システムの開発	実験システムの試作			中間目標システム・試作				目標の高機能理解シ ステムの作成		
(3) 基プレイング基本方式の研究	基本戦略の研究, 棋 譜データベースの作 成			棋譜の知識ベース化						
(4) 基プレイング・システムの開発	実験システムの試作, 基本データの収集			10 級システムの試作				初段システムの試作		

(3) 研究開発の目標・仕様

① 数式理解システム

(中間目標)

処理機能としては、現在のMACSYMA程度のものに不等式と簡単な方程式の処理機能を合せたものを、知識ベース化したシステム。

(最終目標)

高度の数式処理アルゴリズムを組み込んだ数式関連の知識表現・問題解決システム。

② 基プレイング・システム

(中間目標)

アマチュア10級程度のシステム。

(最終目標)

アマチュア初段位のシステム。

2.2 基礎ソフトウェア・システム

2.2.1 知識ベース管理システム

〔1〕 概 要

人間の持つ知識を一定の形式で表現して計算機内に蓄積し、利用することによって、問題解決のプロセスにおいて人を援助する知的計算機の技術を確認する。開発の方針としては、特定の問題分野に依存しない汎用のシステムを目指し、かつ既存の計算機技術および蓄積された情報の利用を保証しつつ、新計算機への円滑な移行が可能なものとする。本システムの主たる目的は、その高度な知的機能により、従来の計算機技術のもとでは困難とされた、設計・診断・意思決定などのプロセスやプログラム開発などにおいて人を援助すること、自然言語・音声・図形・画像等の理解を含む強力なマン・マシン・コミュニケーションの能力によって非専門家による計算機の利用を容易にすること、機械翻訳・知能ロボットなど高度の応用技術を可能にすることなどである。これを汎用性のある計算機の上で実現することにより、従来の計算機技術のもとで期待できなかった広範囲な分野に計算機利用を拡大することができる。このような新しい計算機システムを知識情報処理システムと呼ぶ。

従来の計算機とこのような新計算機の基本的な相違は、与えられた問題にたいし、前者では計算機入力以前に人がその問題の結果を得るための手順を求め、その手続きを一定の形式で表現して計算機入力とするのにたいし、後者では問題の意味そのものを一定の形式で直接に計算機に入力し、その結果を得るプロセス自身を計算機自身が構成し、結果を得る点であり、いわば前者におけるプログラム作成に相当する部分の一部を計算機が受け持つ。

結果を生成するための手順を人間が生成し、プログラム化するには問題に関する多くの知識を利用する。人間が現在行なっているこの仕事を情報処理過程と見做せば、入力である問題表現から出力であるプログラムを生成するためにこれらの知識は不可欠である。いわばプログラムのもつ情報量は問題表現に含まれる情報量にこの知識に含まれる情報量（の一部）を加えたものに等しい。この状況は知識情報処理システムでも同様であり、前述の機能を発揮するためには計算機自身が問題に関する多くの知識を保有していることが基本条件となる。

このために必要となる基本技術は、(1)知識の表現とその蓄積の技術、(2)問題解決のためにその中から必要を知識を検索し、利用する技術、(3)人間の持つ知識を移植し、また人間の知識の増進に応じて、蓄積された知識情報を更新・挿入・削除する技術であり、この上にさらにマン・マシン・システムとしての各種技術、人間の知識獲得のために計算機が援助し得るための技術など多くのものが必要となる。

従来の計算機では問題を手続きにより表現し、この手続き内の命令を形式的に実行する機械として汎用性が実現された。個々の問題ごとの固有性は基本命令の列として手続きを構成する順序に含まれ、これは手続き内の個々の命令を前もって定められた順序で逐次実行するように

構成された計算機の機構に影響しない。一方、知識情報処理システムでは問題解決に際して必要とされる知識の内容は問題ごとに異なる。従って、これを汎用計算機として実現するためには知識の記述形式を定め、それを用いて記述される知識の内容と、それを問題解決に役立てるための利用手段とは完全に独立なものとする。すなわち処理系は記述の形式に基づいて動作するようにする必要がある。この結果、問題の固有性はこの形式を用いて表現された知識の内容にすべて含まれ、これは計算機の機構に影響しない。

このように構成された知識情報処理システムにおいては、分野ごとに必要な知識の総量は異なるため必要な計算機の規模はそれに応じて異なることはあっても、機構は原則的には問題によらない。しかし、実現をめざすこのような計算機開発の第一目標として、2,000語程度の用語（共通語 1,000語、分野別専門用語 1,000語）を用いて表わされる、約 1,000 の事実、法則、諸定義を表わす知識の集まりである知識ベース、関連データの集積であるデータ・ベースおよび知識利用の際に必要な基本アルゴリズムの集まりであるアルゴリズム・ベースを備え、それらを利用して問題を解決するための推論機構を備えた知識ベース・システムを想定し、これらを統括的に管理する知識ベース管理システムの実現をめざす。

〔2〕 必要性

前節で触れた設計・研究開発のような創造的な作業、医療診断や意思決定などの作業、機械翻訳や自然言語・音声・図形・画像理解などのような情報の意味に基づく処理、知能ロボットなどの高度な応用システムなどの分野への計算機の利用はこれまで各種の試みはなされてはいたものの、実質的にはほとんど効果を挙げ得ないままに残されていた分野である。しかも社会全体の情報化、知識集約化の傾向を反映して計算機化の要求が急速に増大している分野である。すでにこのような知的な作業分野に従事する人口が急増しているが、近い将来、要求に応ずる人的資源の供給が追いつかなくなり、知的作業の効率化のために計算機による援助システムを開発することは生産性の向上のみならず、それに従事する人々を煩瑣な作業から解放し、より人間的な作業環境を整える上でも重要になる。さらに計算機産業にたいしては、この分野は非常に大きな市場を形成することが指摘されねばならない。それは従来の計算機が社会活動の特殊な部分にのみ用いられてきたのにたいし、今後は利用分野が拡大し、利用者層が急増すると予想されるからである。

従来の計算機技術が何故このような要求に応え得なかったかについては少なくとも 2つの理由がある。第 1 はハードウェアが高価であったことであり、第 2 はさらに本質的な情報処理機械の原理と構成が欠如していたことに関するものである。

前述のような要求を実現するには大量の記憶容量と高速の処理装置が必要である。また計算機にたいする負荷が大きく、計算機の占有時間も大になる。したがってハードウェアが高価であった時代には実現は困難であった。近年、ハードウェア価格は急速に低下し、性能も向上し

ている。同一性能の計算機のハードウェア価格と、人件費との比率は1960年代と比較して現在はすでに3桁の程度で変化している。このことは貴重な人間の能力を発揮するために計算機を投入することが多くの分野で経済的にも可能となってきたことを示している。

第2の理由は、計算機技術の基本に関わるものである。従来の方式のもとでは人が問題に対する解の手順を書き表わした。この方法は、記述形式が計算機の構造に基づいて定められているため、ユーザは少なくとも計算機に関して準専門家としての知識を（計算機を使うために）要求されること、問題を解くプログラムを作るために多大の時間を要すること、が利用者層の拡大の障害になってきた。そしてさらに重要なことは、プログラムを書くためにはすべての可能な処理の道筋が前もって見通されていることが前提になっていることである。すなわち、利用分野はこのような見通しが可能なものに限定される。しかるに前述の諸要求は状況に応じて動的に処理手順を変更しつつ、発見的に解を求めることを基本とするものであり、処理方式が基本的に異なる。これが従来の計算機技術の枠内でこれらの要求を満たし得なかった最大の理由であり、またこのために新しい情報処理機構を必要とする理由である。

このような問題の基本的な解決策は処理手続きを実行する前に人が行なうという方法によらず、これを問題に応じて動的に生成することである。さらに、必要なのはプログラムという中間生成物を生成することなく、問題に関する知識を有効に利用して結果を得ることである。

知識は非常に多様性に富んだものであり、かつ一般には不完全なものである。これを計算機内に蓄積し、有効に利用するために、そしてさらに新しい知識を獲得してゆくためには、知識を収容する知識ベースを管理する知識ベース管理システムの技術が重要である。この機能はユーザが与えた問題にたいし、解を得るために利用できる知識を探索し、これを用いて原問題を変換して解に近づけるというプロセスを繰返して最終的に解を見出すこと、新しい知識の挿入、削除、更新を行うこと、またそのために知識ベースの無矛盾性チェック、冗長性チェックを行なうこと、必要に応じて既存の、もしくは新しいデータベース管理システムに管理されているデータを要求し解のプロセスに用いることなどである。問題解決に際して知識の効率的な探索がなされるように知識の組織化が行なわれ、また有用な知識が複数存在する時には最適な実行順序をきめることが要求される。なお無矛盾性チェックに際しては、いくつかの相互に矛盾する仮説を挿入する場合もあるので知識の中にフレームを設定し、各フレームの中でのみ無矛盾性をチェックする方式とすることも重要である。さらに知識が不完全である場合には得られる結果も不完全となるが、知識の信頼度に基づく結果の信頼度評価や、データを通して知識の信頼度を変更してゆく学習的機能が必要となる。

このように知識情報処理システム内の知識はいわば従来の計算機の応用プログラムに相当する。したがってこれを操作する処理系、すなわち推論機構はこれより1レベル上のメタ・レベル処理系である。これは知識表現の形式情報に基づいて処理を行なうから、動作は規則的であ

り、近い将来ハードウェアによって実現することも可能である。これにより処理速度も著しく向上することが期待される。

〔 3 〕 内外技術の現状

(1) 国内技術

国内では東京大学で汎用システム化のための知識ベース管理システムの研究や医療用コンサルテーション・システムの研究、京都大学で知識表現のためのセマンティック・ネットの研究および日本語理解のための知識利用方式の研究、大阪大学でアクタを基礎にした知識利用システム、九州大学で知識利用の情報検索方式の研究、電総研で推論方式やデータベースへの動的アクセスの研究、豊橋技術科学大学で知識を用いた画像理解の研究、電電公社武蔵野通研で知識ベースに基づく日本語による図形の世界の処理の研究、東京理科大において医療診断用システムの研究などがそれぞれ進められている。最近では知識表現や推論方式の研究にたいする関心が増大しており、研究者も増えつつある。しかし現状では多くの研究は未だ開始されたばかりであり、知識ベース管理システムの必要性、その満たすべき条件等が明確にされているものは少ない。今後はこれらの条件と、それを満たすためのシステムの基本アーキテクチャを明確にし、またそれを満たす知識表現、推論方式データベースの利用等に関する研究を促進することが必要である。

(2) 海外技術

知識ベース・システムの研究は米国において活発であり、医療分野のコンサルテーション・システムとして MYCIN や EXPERT が有名であるが、この他にもこの分野では十指を越えるシステムが開発され、あるいは開発中である。最近ではさらに医療以外の分野にもこのような知識ベースを利用するシステム開発が盛りであり、これらは知識工学という新しい技術分野を形成している。

米国における知識ベース・システムの基礎的研究はこれらの具体的な分野ごとの利用システムの開発以前に知識表現、推論方式、知識表現用言語などの基礎分野の研究、自動プログラム合成の実験的試みなどという形で実績をもっているが、汎用の知識ベース管理システムについては EMYCIN のような概念が出されているものの、必ずしもその要求仕様が明確化されていない。今後、この方向の研究も増えるものと予想される。

〔 4 〕 開発内容

(1) 研究開発項目

① 知識表現・利用技術の研究

一般的な知識の表現法に関する研究を行う。第 1 に、知識そのものの研究が必要である。知識についての多くの様相を明らかにし、それを利用の観点から分類し、それら各種の知識について適当な表現法を研究開発すべきである。知識の様相として考えられるのは、一般・専

門、大局・局所、完全・不完全、仮説・恒真などである。

知識を利用の側面からみると、その利用のされ方は、多様である。そのため、各目的に合った形に容易に変換されることが必要である。たとえば、人間に対する応答では、言語や、画像の形で表現してやる必要がある。また、データベースに対する手続きなどのプログラム化も必要である。

このような知識表現の問題は、知識ベース管理システムの中核部分であり、可能な表現方式を研究開発し、その評価を行う。そして、知識表現言語を設計・開発する。

また、知識の利用技術として、知識ベースからの知識のとり出しのための検索言語の研究開発を行う。さらに、知識利用に関する知識についての基礎理論とそれを表現・利用するための基本的枠組の研究を行う。

② 知識獲得・学習の研究

知識を獲得して、システムに付け加える場合、すでに存在する他の知識と矛盾してはならない。そのためには、第1に無矛盾性のチェック機構が必要である。これは論理学で一般には限られた時間内に決定できないとされている難問である。ある条件の下で有効に働くアルゴリズムの研究開発が必要である。第2に矛盾が生じた時に、どのようにしてその矛盾を取り除くかが問題である。この2つの場合がある。それは、新たに入力された知識が誤っている場合と、元々の知識ベース中に誤りが含まれている場合である。これは、とくに Default 値との間で起こり易い。一種の Truth Maintenance System の開発が必要である。

学習については、データの集合から、規則を発見することが必要となる。この場合には、どのような規則を考えるかが最も大変な問題である。そのためには、類推や補間などの帰納的推論が重要となる。そのようにして作られた規則は、実際に正しいかどうかを確かめなくてはならない。統計的仮説検定論は、ある種のデータに対して、その検証のための有力な方法論を与える。

③ 大規模知識ベース・システムの研究

知識ベース・システムでは、変数を含まない個別データも他の手続きを表わすデータと同じ形式で扱われる。そのため、記憶効率や処理効率の面から大規模化が困難である。大規模化を達成するには、2つの方法が考えられる。第1は、上に述べた高い機能をもった知識ベースの枠組の中で、ソフトウェアおよびハードウェアの改良によって、容量・速度の両面での改善を図る方法である。たとえば、ハッシングを用いたり、あるいは連想メモリを用いたりすることが考えられる。第2は、個別データの蓄積・管理に従来のデータベースの手法が援用できることから、その部分を別に管理し、上位の知識ベースとの結合を図る方法である。この方式を確立することは、知識ベース・システムを通して既存のデータベースを利用する上で重要であり、一般には分散データベースとして知識ベースからこれらにアクセスする方

式を研究する。具体的には、項目①で開発される知識ベース検索言語を拡張し、既存のデータベース・アクセスをも含むようにする。

④ 知識ベース管理システムの開発

項目①の知識表現・利用技術の研究で開発される知識表現言語並びに検索言語をサポートする知識ベース管理システムの開発を行う。知識ベース自身は、多くの種類のものが、各所に分散されて蓄えられる分散型となることを想定し、その管理もディレクトリ管理によって行う。また、知識ベース全体が矛盾を含まない一つの世界を表わしているのではなく、仮説的な世界をも含めた、互いに相矛盾するようないくつかの世界を同時に表現できることも必要である。これらの多世界の管理も同時に行う。

さらに、項目③の研究成果を実現するために、知識ベースと一般の分散データベースを結合した大規模知識ベース管理システムのプロト・タイプを開発する。

⑤ 知識ベース・マシンの開発

項目①および③の研究によって得られる知識表現および検索言語のための高級言語マシンを作る。これは、データベース・マシンを高度化したものと考えることができる。その成分として、意味ネットワーク・プロセッサ、あるいは、よりデータベース寄りのマシンである関係代数マシンなどが考えられる。

本研究は、項目④の知識ベース管理システムの開発および使用、評価の後に、ハードウェア化によるメリットを吟味した上で行う。知識ベース・マシンの開発は、本プロジェクトの中でも、最も重要な課題の1つである。さらに、最終的には本マシンと推論マシンの融合化、統合化が図られねばならない。

(2) 研究開発スケジュール

本研究課題では、前期においては、開発支援システムの一部として、研究開発パーソナル・コンピュータ上に、小規模知識ベース用の知識ベース管理システムおよび大規模データ用のデータベース管理システムを作る。それは、プロダクション・システム、KRL、セマンティック・ネットワークなどの知識表現言語をサポートするシステムである。

さらに、新しい知識表現言語の開発を目差して、知識の表現方式、利用手法、獲得・学習手法の研究を行う。また、分散した知識の管理を含む大規模知識ベースの構築技術についての研究を行う。

研究の重点は、基礎的な理論についての体系化、中期に開発すべき知識ベース管理システムの仕様決定のための基礎データの収集におく。また知識表現言語および知識ベース検索言語を中心とする知識ベース管理システムの第一次仕様を決定する。

中期では、前期収集したデータと大規模知識ベースの構築技術に基づき、関係型等のデータベース・マシンを土台にして、知識ベース・マシンのプロトタイプを試作する。さらに、前期

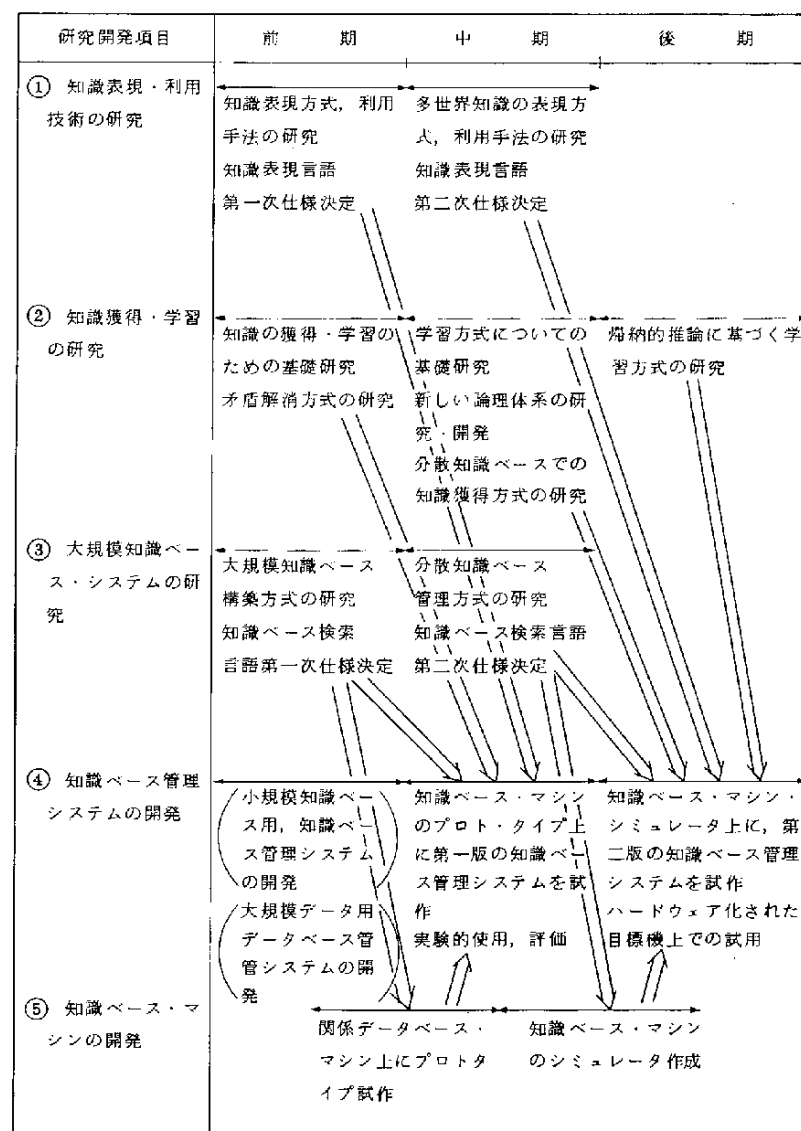
で決定した知識表現言語および知識ベース検索言語をサポートする知識ベース管理システムをこのプロトタイプ上で試作し、実験的に使用し、システムの評価を行う。

基礎研究としては、分散知識ベース系の管理方式の研究と、そこでの知識獲得方式の研究を行う。さらに学習方式についての基礎研究を前期に引き続いて行い、新しい論理体系の可能性と、その体系に基づく知識表現方式の研究開発を行う。

上に述べたシステムの評価と、基礎研究の成果をもとに、後期で開発すべき知識ベース・マシンおよび知識ベース管理システム（知識表現言語および知識ベース検索言語を含む）の第2次仕様を決定する。

後期では、中期で決められた仕様に基づき目標とする知識ベース・マシンのシミュレータを作成し、その上で知識ベース管理システムを開発する。この開発の過程においても、適宜、仕様の修正、精密化をすすめる。このシミュレータは、ハードウェアの開発に利用する。

知識ベース管理システムの研究開発スケジュール



最後に、ハードウェア化された目標機と結合し、応用システムの試用に供する。また、応用システムの試用を通して得たデータに基づき、より高度な学習機能などの付加を目差した拡充のための研究開発を行う。

(3) 研究開発の目標・仕様

前期の研究開発用知識ベース管理システムの研究開発の目標・仕様は、システム化技術の部分を参照されたい。

以下に、前期研究開発項目④および⑤について、その目標・仕様を中期と後期に分けてまとめる。なお、⑤の知識ベース・マシンの研究は、他の研究項目との関係で、中期・後期共に1～2年先行して開始される。

研究開発項目	中期目標・仕様	後期目標・仕様
知識ベース管理 システムの開発	ルールとデータの同時管理。 データベース・アクセスの最適化機構。 矛盾解消機構。 推論マシンとのインタフェース。	多世界知識ベース。 分散知識ベース。 帰納的推論に基づく学習。 推論マシンとの融合化。
知識ベース・マシンの開発	2000 個のルールと、100 万項目のデータを収容検索可能。 (1 項目 10^3 B)	20000 個のルールと1 億項目のデータを収容検索可能。 (100 GB)。

2.2.2 問題解決・推論システム

〔1〕 概 要

第5世代コンピュータ・システムの機能は現世代コンピュータのもつ機能とは質的に異なったものとなる。指示された手順のみに従った融通性のない処理から、与えられた問題を理解し、自らその解決へ向っての推論を進めながら処理を行なう。いわゆる知的処理機能を有するようになる。

このように知能化された高度な機能を現世代コンピュータ上に実現することには、ソフトウェア、ハードウェアの両面からみて無理があり、問題解決・推論システムを核に据えたコンピュータ・システムを新たな視点に立って考え、その制御機構、処理形態、マシンの構造等を明確にしていかなければならない。

問題解決・推論システムは次の2点で重要な意味をもっている。

(I) 第5世代コンピュータ・システムに於ける基本応用システム、各種応用システムの構築

者、或いは一般利用者（エンドユーザ）のためのプログラミング環境を提供する。

(2) 知識情報処理に於ける基本的処理構造を明確化し、その仕様を第5世代コンピュータ・ハードウェアへの要求条件として提供する。このように、問題解決・推論システムは第5世代コンピュータの処理系の基礎を与えるものであるが、さらに、知識ベース管理システムでの管理系の記述、知識表現法の研究、或いは知的インタフェース・システムでの自然言語の処理系記述等にも密接に関連するものである。したがって問題解決・推論システムのもつ機能が第5世代コンピュータ・システムの提供する知識情報処理の質を規定すると言っても過言ではない。

問題解決・推論システムの研究開発項目は次の様に設定される。

- (a) 問題解決・推論アルゴリズムの研究
- (b) 問題解決向記述言語の研究
- (c) 推論マシンの開発

推論アルゴリズムの研究は、演繹的推論の高速化、帰納的推論の確立等問題解決・推論システムが基礎におくアルゴリズムを体系化するものである。問題解決向記述言語の開発は、第5世代コンピュータ・システム上でのソフトウェア開発のための記述手段を開発するものである。問題解決向記述言語は推論アルゴリズムをベースに、より直観的なわかり易い問題記述機能を埋め込んだ言語・記述系として体系化される。推論マシンの開発では、問題解決向記述言語で書かれた問題解決・推論プログラムを効率的に実行するマシンのハードウェア・ベースを確立し、第5世代コンピュータのアーキテクチャを体系化する。

上記3つの研究開発項目の研究開発は理論的モデルの設定と現実システムの構築、実証の2面から進めていかなければならない。

- (a) 推論アルゴリズムの研究

推論はシステムが自ら与えられた問題の解（ゴール）を求めて処理を進める過程である。推論では述語論理等で表わされた命題の真偽を判定する証明過程が基本である。システムは知識として蓄えている公理・定理を用いて演繹的推論、或いは帰納的推論を進め、与えられた命題の真偽を判定したり与えられた命題を真ならしめるように動作する。このような推論動作過程及びそれによりもたらされる効果が問題解決・推論システムでの処理である。

推論に基づく処理に於いては、従来のデータ処理とは異なり求めるべき解へ至る手順は陽に示されない。システムは発見的方法或いは試行錯誤的方法により自ら解へ至る道を求めて処理を進める。

推論アルゴリズムの研究では、的確な発見的手法を求めること、出来るだけ無駄な試行錯誤をなくす推論法則を求めること、試行錯誤の制御を効率化する方策を求めること、推論過程での必要な公理、定理を早く見つけ出す探索手法を求めること等、とくに演繹的推論を高速化するアルゴリズムの研究が重要である。

人間の特徴である不完全な知識に基づく推論は、帰納的推論の一種であり、今後中心的な研究を進めて行く必要がある。これはデータベース或いは知識ベースの無矛盾性チェックの問題とも関連する重要なテーマである。

(b) 問題解決向記述言語の開発

問題解決向記述言語の開発は第5世代コンピュータ上で基本応用システム、並びに各種の応用システムを構築していく際のソフトウェア言語インタフェースとして重要な意味をもっている。

プログラミング言語の研究・開発はこれまで勢力的に展開され PL/I, ALGOL 68 を頂点として現世代コンピュータ上でのプログラミングの概念、言語体系が一応確立したように思える。しかし、現在の言語体系はあくまで、“どのように”処理するかを記述する手続型記述言語であり、しかもその処理記述は記憶概念に基礎を置いたものである。

知識情報処理を指向する第5世代コンピュータ上でのプログラミングは推論に基づく問題解決法を記述するものであるから、そこでは“なにを求める”かを記述する非手続的記述を主体とする言語が必要となる。したがって第5世代のプログラム言語には大きな質的変革が望まれることになるが、この変革に向けての芽は、ここ10年間の人工知能研究やプログラム基礎論での研究成果の蓄積、関数型言語および論理型言語、人工知能向言語、抽象データ型言語、等での具体的な提案や使用経験の蓄積により次第に育ってきている。

問題解決向記述言語の開発というテーマのもとで研究開発される言語には次の様なものがある。

- ① 核言語
- ② 意味記述言語
- ③ 仕様記述言語
- ④ システム記述言語
- ⑤ エンドユーザ向言語

このうち意味記述言語、仕様記述言語は知識ベース管理システム、知的インタフェース・システム、知的プログラミング・システムと密接に関連するものであり、それぞれのシステムのもとで詳しく説明されるのでここでは簡単にふれる。ここで特に注意すべきことは上記の各言語が個々ばらばらに設計されるのではなく、全てに共通の基本的な機構があり、その上にそれぞれの目的に合った言語が設計されるということである。

これらの記述言語に共通する言語構造は、関数概念に基づくプログラミングを指向した関数型言語の構造と、手続的記述を排除して述語論理に基づくプログラミングを指向した論理型言語の構造とが融合されたものである。さらにそこには、(工芸的でなく)工学的手法に基づく合理的なソフトウェア生産をサポートするための基本概念であるモジュラー化プログラ

ミングを実現させる種々の機能が盛り込まれることになる。

(c) 推論マシンの開発

知識情報処理向マシンは、問題解決・推論システムを現実化するためのハードウェア機構である。知識情報処理では推論が基本動作であるゆえ、現世代コンピュータに比してそのハードウェア機構は相当高度なものが要求される。更に試行錯誤的、非決定的実行が必然であると考えられるため、並列処理機構の導入等による高速化が必要とされる。

推論マシンは問題解決向記述言語、特に核言語に適合した高級言語マシンとして設計し、言語のもつ基本構造（セマンティクス）とマシンが備える制御機構との間に大きな差異（セマンティック・ギャップ）が生じない様にしなければならない。この様に設定された推論指向の基本機構の上に、各種の応用に沿ってそれぞれの専用マシンが、ソフトウェアによる応用システムの構築と融合しながら開発されることとなる。推論マシンの“エンジン”部とも言うべきものは、推論実行制御機構である。これを推論エンジンと呼ぶ。推論エンジンは更にその基本的構成要素として推論過程での定理の探索に欠かせないパターン照合機構、試行錯誤的実行の制御に欠かせないバック・トラック機構、非決定性制御機構等があり、それぞれ重要研究テーマである。

一方マシンを実現するための要素技術としては、VLSI技術の進展、データフロー・モデルやリダクション・モデルに基づく並列処理、分散処理概念の明確化が期待される。この様な部品、材料技術の進展とマシン・モデルの理論的進展とが融合されて、推論マシンを核に据えた第5世代コンピュータ・アーキテクチャが確立されることになる。

〔2〕 必要性

概要の項で第5世代コンピュータ・システムに於ける問題解決・推論システムの位置づけ及びその重要性についてふれたので、ここではその研究・開発の必要性を整理して述べることにする。

問題解決・推論システムの研究は第5世代コンピュータに於いて処理機能の明確化という面で中核をなす。即ち、その処理モデルの設定により理論的に処理能力を明らかにすると同時に実際にシステムの実現に向けての基本技術確立の基盤を与えるという極めて重要な位置をもつ。

第5世代コンピュータがどの程度の知能的処理能力をもち得るか、そしてこの第5世代コンピュータの上に築かれた知識情報処理システムがどの程度の“やわらかさ”を発揮して使い易いシステムとなるかは、その核となる部分のもつ処理能力に依存する。したがってその核部分のもつ能力は出来るだけ理論的モデルに基づいて明確化されなければならない、そのための理論的基盤を確立しておくことが必要である。

一方システム実現の見地からは、効率的な推論アルゴリズムの開発は不可欠なテーマである。またそのアルゴリズムをベースとした処理をわかり易く表現する記述法、言語システムの開発

は、各種の応用システムを開発する際のソフトウェア生産性を高める上で不可欠なテーマである。更に問題解決向マシンの開発は推論アルゴリズムの効率的実現のためのハードウェア基盤を確立し、問題解決向記述言語の処理系としてのハードウェア仕様を明確化して第5世代コンピュータ・アーキテクチャの要求条件を提示するという重要な位置をもつものである。

第5世代コンピュータは問題解決向記述言語および知識ベース管理システムが提供する仕様を対象とする高級言語マシンとしてそのアーキテクチャが設定される。また基本応用システム及び各種応用システムは問題解決向記述言語および知識ベース管理システムのもとで開発される知識表現言語により記述されることになる。したがって第5世代コンピュータ・システムに於いて中核となる記述系が問題解決向記述言語と知識表現言語ということになる。

情報処理システムを総合化された体系として機能させるためには、システムの構造とその制御・処理の記述系が一体化されなければならない。即ちマシンの構造とその上での記述系がある基本的機構のもとで統合化され一体となったものでなければならない。そしてこの統合・一体化のためにハードウェアとソフトウェア間の橋渡しの役目を果たすのが、問題解決・推論システムである。

〔3〕 内外技術の現状

(1) 国内技術

(a) 推論アルゴリズムの研究

人工知能研究の一環として、定理証明プログラム、知識ベース上での推論アルゴリズム、ゲーム・プログラム等の研究が電総研、武蔵野通研等の国公立研究機関、および東大、京大、阪大等の大学で行なわれているが、研究対象自体の規模は余り大きなものでなく、創造的研究成果はまだ目立ったものがない。しかし今後の研究発展のための基盤は蓄積されつつあると思われる。

(b) 問題解決向記述言語の開発

電総研、通研、東大、京大をはじめとする多くの研究所・大学で新汎用言語の研究・開発の努力が続けられいくつかの成果が得られている。しかし新しいプログラム言語（記述系）を考案し、世の中に広めていくという組織立った言語開発の試みは ALGOL N の研究開発があるという程度で欧米に比べて大きく遅れている。今までは極少数の例外を除いて、ほとんどが輸入によって賄われてきたといってもよく、日本人による言語開発を悲観視する風潮すらある。

しかし、最近の新言語研究の国内状況を見ると、電総研、通研をはじめとする研究機関、東大、京大、等の大学での人工知能研究グループの Lisp 処理系の開発およびその使用経験の蓄積がなされてきており、関数型プログラミングの技法確立のための基盤が固められつつある。特にデータフロー処理概念との融合により新たに関数型プログラミング技法、および

その記述言語設計に対する関心が、電総研、武蔵野通研、東大等を中心に高まりつつある。

また論理型プログラミングの研究は PROLOG 処理系の移植・実現により、その使用経験を踏むという形でその第一歩を踏み出した状況にあると言える。

(c) 推論マシンの開発

推論マシンの開発そのものは今後の研究テーマであり、国内外共に未だ目立った研究の動きはない。しかし推論マシンへ繋がる研究としては Lisp マシンの開発、データフローマシンの開発を上げることが出来る。Lisp マシンは電総研、武蔵野通研、京大、神戸大、慶大で研究が進められている。

データフローマシンの研究は武蔵野通研、電総研、東大等で研究が進められており、欧米で提示された処理概念の後追いの状態から抜け出して、新たな処理領域の拡大とそれに伴った処理概念の創出等、独自の研究ステップへ踏み出す素地が出来つつあると思われる。特にデータフロー処理概念を推論機構と融合させて知識情報処理向マシンのアーキテクチャ基盤として捉えている点は日本独自の新規な発想であると評価される。

(2) 海外技術

(a) 問題解決・推論アルゴリズムの研究

問題解決・推論アルゴリズムの研究は CMU, Stanford 大学等において Production System で、或いは Rule-based System 研究の一環で新ルール適用の方策を最適化する研究として進められている。推論アルゴリズムに数学的に明確な基礎が与えられたのは J. A. Robinson による導出原理の発見 (1965) があってからである。以後推論アルゴリズムを実現する単純簡潔なプログラムが計算機上に作成されるようになり、導出原理による演繹手順の効率化をめざした各種の導出手法が考案されている。導出原理による推論アルゴリズムが言語の概念にまで昇華したのは London 大学の R. Kowalski による PROLOG の提案があつてからである。

他に自然言語の意味論の研究分野では KRL 上での推論, Semantic Network 上での推論のアルゴリズム研究も進められているがまだ目立った成果はない。

最近、より柔軟な推論システムを指向するアルゴリズム研究として Non Monotonic Logic の研究が注目されてきている。

(b) 問題解決向記述言語の開発

言語の開発は欧米で極めて活発に行なわれている。

まず関数型言語については Lisp が代表的であり、欧米の多くの大学、研究所の人工知能研究者は Lisp を常用語としている。Lisp にはいくつかの方言があるが、MIT の MacLisp, BBN, Xerox PARC で開発された Inter Lisp, California 大学 Irvine 校の UCI Lisp が公用語的存在である。IBM で開発された APL 言語も関数型言語の特性をもつ言語である。

関数型言語の開発は、最近別の観点から注目されている。IBMのJ. Backusはソフトウェア工学の見地からFP (Functional Program) とそのコンピュータ・アーキテクチャの必要性を説いた。以来非ノイマン型マシンと関数型言語の融合性が強く意識されるようになりリダクション・マシン、データフローマシンを意識した関数型言語の研究開発が進められている。後者の例としてMITのVAL, California大学Irvine校のId等がある。

論理型言語についてはPROLOGがあり、仏のMarseille大学、英のEdinburgh大学、カナダのWaterloo大学等で処理系が開発されている。

抽象データ型の対象指向言語はSimula 67に端を発す。現在では、モジュラー化プログラミングの概念を実現する言語として各所で言語および処理系が開発が進められている。Xerox PARCのMesa, Smalltalk, MITのCLU, CMUのAlphard, それに米国防省の標準言語Ada等がある。

(c) 推論マシンの開発

推論マシンの開発として直接の動きはまだないが、最近のマシン・アーキテクチャの技術として注目されるのはVLSI技術の進展に支えられて、パーソナル・コンピュータ開発の動きが活発なことである。Xerox PARCのALTO, DORADO, Three Rivers社のPERQ, CMUのSPICE, IntelのiAPX 432などがある。ALTO, DORADOはMesa, Lisp, Smalltalkを、またSPICE, Intel iAPX 432はAdaを搭載言語としている。

記号処理マシンの開発としてはMITやCMUでのLispマシンの開発がある。MITのLispマシンは何回かのバージョンの進歩があり、パーソナルLispマシンとして商品化されるまでに至っている。

非ノイマン型マシンの研究はリダクション・マシンとデータフローマシンの研究が活発である。リダクション・マシンとしては独GMDのリダクション・マシン実験機、米North Carolina大学の分散型構造リダクション・マシンの構想等がある。またデータフローマシンの研究はMIT, Utah大学, California大学Irvine校, 英のManchester大学, 仏のToulouse大学等で進められている。

[4] 研究開発内容

(1) 研究開発項目

(a) 問題解決・推論アルゴリズムの研究

(a.1) 理論モデルの研究

推論機能を基礎とする処理方式の諸特性を明らかにし、その性能評価を客観化することを目的とする。本研究は第5世代コンピュータ・システムの理論的ベースを与えるものであり、したがって本研究はプロジェクト進行中全期間にわたって続けなければならない。

本研究は、演繹論理、帰納論理、様相論理等の論理学や計算意味論、計算の理論、言語理

論，オートマトン理論等の観点から進めることになる。

(a. 2) 高速化推論アルゴリズムの研究

問題解決・推論をアルゴリズム面で高速化することはシステム実現の観点から特に重要である。演繹的推論の機構は基本的であると考えられ，演繹に基づく問題解決過程を高速化するアルゴリズムの研究が必要である。そのためには問題解決の並列実行，無駄のないバック・トラック法，大規模データの高速検索などの研究を進めていく必要がある。

不完全な知識に基づく推論は帰納的推論の一種であり，システムにより高度な処理機能をもたらすためには，今後積極的に研究を進めていかなければならない。この方向の研究の進め方のひとつとして Truth Maintenance の概念に基づく推論，Default Reasoning による意味の推察とそれに基づく仮説的推論など Non Monotonic Logic による推論アルゴリズムの研究をあげることが出来る。

(b) 問題解決向記述言語の開発

(b. 1) 核言語の開発

核言語はその名の通り，第 5 世代コンピュータがもつ基本的な言語インタフェースである。核言語が基礎となってさらに上位の言語仕様が定められる。また各種システム構築のための処理プログラムは核言語によって記述される。核言語は低位の言語インタフェースではあるが，現世代の高級言語に比べてはるかに高い記述機能をもつものである。

核言語の基本的構造は論理型と関数型とが融合したものとして設定される。さらに，対象指向の要素を強く反映し，モジュラー化プログラミングによるソフトウェアの高生産性を配慮した言語仕様となる。

核言語の仕様設定に当っては次の各項を設計要因として考慮しなければならない。

① 知識ベースとの融合

知識ベースへのアクセスと，それから得たデータの処理とは一体化されなければならない。したがって核言語仕様設定に際しては知識表現系の構造，知識ベースへのアクセス記述等を考慮しなければならない。

② 記号処理（高度なデータ構造と演算）

核言語は従来の数値計算指向の言語系とは異なり記号処理（非数値データ処理）指向の言語系として設定される。推論を基礎として知識情報を処理するためには高度のデータ構造を扱えることが必要である。基本データ型として，リスト，対，集合等が含まれる。さらにデータ抽象化機能により任意の構造のデータ型を定義することが出来なければならない。

③ 並列処理

核言語は関数型と論理型の融合した構造である。言語のもつ基本的意味構造としては従来の逐次処理概念に代わって，並列処理概念に立脚したものでなければならない。さらに加え

て陽に並列処理，直列処理を表現するための高機能的表現要素を持つことが要求される。

④ 非決定性処理

核言語では知識ベースへのアクセス記述や推論動作の記述に於て非手続的表現が出来なければならない。非手続的処理ではパターン照合により，複数のゴールが非決定的に選択される。したがって非手続的記述ではパターン照合による非決定性処理の記述が言語仕様化されなければならない。

非決定性処理の記述は並列処理概念を基礎とし，これにバック・トラック制御の記述を融合させた形で言語仕様化されることになる。

⑤ 対象指向（抽象化技法）

大規模ソフトウェアを構築していくには，単に論理型と関数型との融合だけでは不十分である。最近のソフトウェア工学に於いて対象指向言語によるモジュラー化プログラミングの手法が提案され，その有効性が検証されつつある。

核言語設計に於いても，このモジュラー化プログラミングが可能な仕様を設定することが必要である。

抽象化技法によりプログラムをモジュール化し，これを知識ベース化することが出来れば半自動的なプログラム生成が可能になり，CADによるソフトウェア生産への繋がりが出来ることになる。（知的プログラミング・システムの項を参照）

⑥ プログラミング・システム化

当然の事ではあるが核言語は言語として単独に存在するだけでは用をなさない。コンパイラ，インタプリタ，デバッグ支援システム，検証システム，エディタが一体化したシステムとして知識ベース化され，知識ベース管理システムの下で機能しなければならない。

（b. 2）意味記述言語

意味記述言語は核言語をはじめ，種々の言語の意味を形式的に記述する言語である。また知識ベースにおける知識表現のための記述言語，自然言語処理における意味の形式的表現，意味辞書記述等に使用される。人工言語での意味記述と自然言語，知識表現での意味記述とは多少異なる面もあるが，機械的処理の観点からは，両者を統一した形式的記述手段が要求される。

意味記述の方法としては操作的，指示的，公理論的等種々あるが，各手法を比較検討し，より広い適用性をもつ記述法を確立していかなければならない。

知識ベース，自然言語処理での意味記述言語仕様については知識ベース管理システム，知的インタフェース・システム，知的プログラミング・システム，質問応答システムの各研究との連絡を密にしながら設計を進めていく必要がある。

(b.3) 仕様記述言語

各種システム（ハードウェアも含めて）の仕様を厳密に記述するための言語である。仕様記述には機能仕様記述と構成仕様記述がある。機能仕様には高位の抽象度が高いものから下位の詳細厳密化された仕様までいくつかのレベルが考えられる。構成仕様は対象がソフトウェアかハードウェアかにより若干異なると思われるが核言語に多少の変更を加えることにより流用できると考えられる。

データフローの概念を基礎にして仕様記述を考えると、原理的にハードウェア、ソフトウェアの境界がなくなり、さらに抽象化技法を持ち込むことによって機能仕様記述と構成仕様記述との間に連続性が出てくると期待される。

知的CADによってシステムを設計する場合、知識ベースに蓄えられたシステム構成部品群の中から要求された仕様を満足する部品が検索される。したがって仕様記述法の確立及びそのための記述言語の問題は知識ベースにおける知識表現法とも関連する。またシステムの検証は仕様記述に基づいて自動的に行なわれることになる。このように仕様記述言語は知識ベース管理システム、知的プログラミング・システム、知的CAD/CAMシステムと密接に関連する。

(b.4) システム記述言語

システム記述言語は基本応用システム、各種応用システムを構築する際のシステム設計用記述言語である。

システム記述言語には核言語がほとんどそのまま当てられるが、特に、より効率的なシステムの設計を望む設計者のためにハードウェアに近い記述要素を取り入れた言語仕様が設定される。

(b.5) エンドユーザ言語

一般の利用者にはその用途、適用分野によって種々の言語が提供されることになる。第5世代コンピュータ上での自由闊達なプログラミングのためには核言語が使用される。知識ベースへのアクセス、質問応答システムでは図形言語、表言語、自然言語等が用いられる。これらの各言語の仕様は各種の応用システムのもとで定められるべきものである。

(c) 推論マシンの開発

この研究開発項目は、第5世代コンピュータ・アーキテクチャ開発のための理論的基盤を固めると同時に、問題解決・推論システムの持つべき機能を整理し、ハードウェアへの要求条件を明確化するものである。本研究で示された要求条件に沿ってのアーキテクチャの具体化が、アーキテクチャ関連課題の研究で詳細に展開されるのでここでは基礎理論研究とアーキテクチャ研究とのインタフェースという意味で簡単にふれることにする。

(c.1) 計算機構理論

推論処理を高速化するためには、並列処理、分散処理、連想処理、非決定性処理を盛り込んだ処理概念とその計算機構を明らかにしなければならない。

計算機構をモデル化して捉える場合、そのモデルが現実のアーキテクチャと遊離したものであってはならない。VLSIによる高機能化素子等いくつかのアーキテクチャ構成要因を考慮した上で計算機構のモデルが立てられる。

計算機構としては関数型、論理型のものがモデル化され、これらを融合する計算機構が設定されなければならない。さらに、抽象データ型を処理する計算機構へと拡張されなければならない。現在リダクション・マシン・モデル、データフローマシン・モデルがVLSI技術を前提として高度の並列処理、分散処理を実現する計算機構モデルとして有望であると考えられている。今後は、論理型計算機構並びに高度のデータ構造処理機構がこのモデル上でうまく機能するかという問題を解決していかなければならない。

(c.2) 推論エンジン

問題解決・推論システムの“エンジン”部である推論エンジンのハードウェア実現は第5世代コンピュータ・アーキテクチャにとって中核的問題と言える。推論エンジンの実現に当っては次の構成要素の開発を進めなければならない。

① 高速パターン照合機構：単なる文字列照合では不十分である。変数を含むパターンへの照合に際しては値の結合を行なう、またパターン中に関数が現われたときはその関数の評価を行なう、などのUnification機能をもたなければならない。Unification機能をどのレベルまでをハードウェア実現とし、どのレベルからソフトウェア実現とするかは設計上重要な問題である。

② 非決定性制御機構／バック・トラック機構：高度の並列処理機構をベースとするか、逐次処理をベースとするかで非決定性制御機構の実現方式は大きく異なる。逐次処理ベースではスタックを用いたバック・トラック制御が大きな比重を占める。一方並列処理ベースでは複数の選択枝が同時に実行され、有効な結果だけが残されるという実行制御がとられるからバック・トラックの比重は小さくなる。（すなわち、バック・トラック制御は本質的には不要となるが、現実には選択枝の広がり制限をすることが必要であり、バック・トラック機構を排除するわけにはいかないと思われる。）

③ メモリ機構：推論システムに与えられた公理、定理等の推論規則或いはPROLOGのような論理型プログラム、推論実行中のデータを格納するメモリ機構とその装置構成も重要な問題である。推論規則や論理プログラムはパターン照合の対象となるから、これを格納するメモリ装置とパターン照合装置とは一体化させて設計を進める必要がある。

推論マシンはこれらの各構成要素から組み立てられる。マシン構成に際しては、VLSI

素子を駆使して推論マシン各部品の機能を分散化し、システムの制御はデータフロー概念に基づいて実現されるということになると考えられる。

(2) 研究開発スケジュール

推論マシンの開発は次の2面から進めなければならない。

① 開発支援機の開発：現在のコンピュータ・アーキテクチャとそれ程の開きがない実現可能なアーキテクチャで記号処理専用マシンを開発し、その上でソフトウェアにより知識情報処理システムの縮小モデルを構築して各種評価を行なう。このマシンは現在のLispマシンを発展させた記号処理向逐次型マシンである。このマシンは第5世代コンピュータへ至る途中の踏台的意味をもつ。ソフトウェア・言語、各種応用システムの評価実験はこのマシンを道具としてくりかえされる。

また推論マシン各部のハードウェア化とその評価実験もこのマシンをベースとして行なわれる。

② 第5世代目標機の開発：第5世代のコンピュータ・アーキテクチャの開発を革新的概念に沿って進める。当初から並列処理、連想処理を指向して、ノイマン原理に基づかないアーキテクチャの確立をボトムアップ的に進める。まずは小規模な実験システムによる原理的な動作検証と性能評価実験を進め、次第に処理概念を拡張しつつ実験機の規模を拡大していく。規模を拡大していく段階で①の逐次型マシン上で開発されたソフトウェアを実験機上に移植して行き評価実験の規模を拡大すると同時にソフトウェア開発を逐次型マシンから並列処理マシンと変えて行く。

具体的な研究開発スケジュールとしては例えば、次の様な設定が考えられる。

まずデータフロー・マシンの実験機を試作する。この実験機は小規模なものであり、関数型言語によるプログラミングがサポートされる。この段階では論理型プログラムに対するサポートはデータフロー・マシン上でのソフトウェア・インタプリタによる。(これは逐次型マシン上での論理型プログラムのソフトウェア・サポートに対応する。)後期スケジュールでは実験機の規模を拡大すると同時に、ソフトウェアでサポートされていた推論マシンの各部を徐々にハードウェア化してゆく。ハードウェア化に当っては逐次型マシン上で開発された部品を構成要素として用いることも考慮しつつ、データフロー・マシン上での実験評価をくりかえしてゆく。そして最後にデータフロー・マシンを基礎とする問題解決向マシン・アーキテクチャのプロトタイプを確立する。当然この研究開発スケジュールの中には、抽象データ型をサポートするための基盤となるメモリ機構の開発も含まれる。

研究開発項目 \ 年度	1	2	3	4	5	6	7	8	9	10
(a) 問題解決推論アルゴリズムの研究	演繹推論高速化アルゴリズム Non monotonic logic 基礎固め 発見的手法のアルゴリズム研究					Truth maintenance Default Reasoning を融合した 高速化推論アルゴリズム体系化				
(b) 問題解決向記述言語の開発										
① 核言語	核言語第1次仕様		逐次型処理系, プログラミング評価			第2次仕様		並列型処理系試作 プログラミング評価		
② 意味記述言語	理論的研究		記述系試作			第2次仕様		記述評価, 意味ベース		
③ 仕様記述言語	理論的研究		第1次仕様と記述系試作				第2次仕様		仕様ベース作成	
④ システム記述言語	第1次仕様		試作システム記述			第2次仕様		処理系作成 プログラミング評価		
⑤ エンドユーザ言語	表現法の検討と処理系試作, 評価							各種言語の仕様作成		
(c) 推論マシンの開発										
① 計算機構理論	計算機構理論検討					第5世代マシン計算機構理論体系確立				
② 開発支援機	関数型, 論理型指向逐次型マシン開発					推論エンジン部 ハードウェア化				
③ 第5世代目標機	非ノイマン型実験機(小規模) 試作評価					非ノイマン型(中規模)			第5世代目標機仕様 試作設計 推論エンジン部ハード化	

(3) 研究開発の目標・仕様

第5世代目標機の性能は推論処理を高速に実行できること。高度の並列処理，連想処理を駆使したマシン構成により推論実行速度にして現世代マシンの $10^5 \sim 10^6$ 倍程度の性能を発揮すること。

推論マシンの性能指標として，LIPS(Logical Inferences Per Second：推論実行速度)を単位にとると，現世代マシンは $10^4 \sim 10^5$ LIPS程度である。第5世代コンピュータでは $10^2 \sim 10^3$ Mega LIPS程度の性能を目標とする。

問題解決向記述言語系は関数型，並びに論理型プログラミングをサポートし，かつ対象指向に基づくモジュラー化プログラミングを可能とすること。生産されたプログラム・モジュールは，ソフトウェア部品として知識ベース化され，知的プログラミングに於いて有効に利用されること。

2.2.3 知的インタフェース・システム

〔1〕 概 要

第5世代コンピュータの目標は，「使い易いコンピュータを実現する」ということである。ここでいう「使い易さ」の意味は，特定の(コンピュータの)専門家は言うにおよばず，一般の(コンピュータの素人である)使用者にとっても使い易いコンピュータを実現することではない。

このような観点から現在のコンピュータを眺めてみると，「使い易さ」に対する機能が非常に不十分であると言わざるを得ない。その最大の要因は，(使用者である)人間とコンピュータとの使用言語の相違に基づくギャップにある。そこで，この様なギャップを解消するためには，コンピュータ・システムが柔軟な会話機能を持つこと，すなわち自然言語や音声，図形，画像による会話機能を持つ必要がある。第5世代コンピュータのなかの知的インタフェース・システムの果す第1の役割は，このような柔軟な会話機能を実現し，人間とコンピュータとの間の使用言語(これには自然言語や音声，図形，画像も含まれる)の相違に基づくギャップを解消することにある。

第2の役割は，知的インタフェースを実現することにより，我々の思考を支援し推進することが可能になる，という点にある。自動車のエンジンが，我々の歩行能力を推進拡大したのと同様に，第5世代コンピュータを，あたかも我々の思考能力を推進拡大するための“思考のエンジン”として機能させる必要がある。知的インタフェース・システムは，それを実現するための中核技術の一つである。第5世代コンピュータでは自然言語，音声，図形や画像は，単にコンピュータへのデータの入出力の手段と考えるのに留まらず，人間の高度な推論や判断をコンピュータが支援する際に，それを効果的に行うための手段として捉え，自然言語，音声，図

形、画像の表わす内容について理解し得るようなシステムを作り上げる技術の開発を狙っているのである。

知的インタフェース・システムは、自然言語・音声系と図形・画像系とに分かれる。自然言語・音声系は、基本応用システム中の音声応用システム、質問応答システム、そして機械翻訳システムとも密接に関連する。基礎ソフトウェア・システムとしての自然言語・音声系は、これら三者（音声応用システム、質問応答システム、機械翻訳システム）に共通する研究課題と、三者個別に、各々のシステムで本質的で中核となる研究課題となるものを取り上げて研究開発するものである。一方、図形・画像系は、基本応用システム中の質問応答システムおよび図形・画像応用システムのベースとなる諸技術を研究開発するものである。

自然言語関連では、我々人間が、日常の会話で用いている自然言語を用いてコンピュータと知的な対話を行うための諸技術の確立をする。この技術には、構文解析技術、意味解析技術、談話解析技術、文章合成技術、言語データ・ベースの構築が含まれる。以上の諸技術で開発した洗練された解析アルゴリズムを統合しハードウェア化した自然言語処理マシンの開発を行う。

音声応用システムでは、人間とコンピュータの間で“知識”が取り交わされるとき、キーボードからの入力やディスプレイの表示を読みとる作業よりも、自然音声による入出力の方が、はるかに能率が良く、しかも一般非専門家にとって使い易いものとなる。コンピュータに推論能力や問題解決能力が備わった時、いわゆる“常識”が組み込まれると、人間の間で交わされる不完全な文章も内容を補って理解できるようになる。このような状況では人間がコンピュータに、キーボードから“完全”な文章を入力する必要はなく、人間に対して話しかけるように、音声により適度に省略された自然な会話ができることが望ましい。

さて音声は、我々の通信手段の中で最も容易かつ効率良く情報を伝達できる方法である。言語情報は音声生成器官によって音声波に変換され、聴覚機構によって受容される。このシステムは、我々が幼時から年月をかけて修得あるもので、それら複雑な仕組みを解明して機械的に実現することは容易ではない。

1970年代に入って音声認識の実用化の研究が開始され、社会的ニーズと相まって国家的プロジェクトのテーマとして取り挙げられるようになった。我が国では1971年からスタートした大型プロジェクト「パターン情報処理システム」の1テーマとして音声認識の研究が始められ、米国では時を同じくしてARPAによる「音声理解システム」の研究が行なわれた。我が国の大型プロジェクトで培った音声認識技術では、実時間で単語を識別する音声認識システムが実用化し、基礎研究では、新しい方法による音声生成過程の推定や、音声分析レベルでの種々の研究成果があげられており、今後の音声認識研究の土台となるものである。

本節で述べる音声応用・知的インタフェースの中心テーマは、従来の音声認識研究を発展させて、音声理解システムを研究開発することである。すなわち、音声分析で得た音素記号列

(ある程度の誤りを含む)から、構文・意味・語用論的情報を用いてその意味内容を理解して文章として内容を把握することを目標とする。米国における「音声理解システム」開発プロジェクトの評価論文で指摘されているように、十分な音声分析技術をベースにしなければ、その上にいくら理解技術を上乗せしてもシステムとしての性能には限界がある。従って本研究開発の前期には、音素識別方式の研究が重要であって、併せて音声合成技術の基礎研究や、音声研究用共通データベースの作成等を行う。そして後期には知識情報処理技術を用いて言語情報をとり入れて音声理解研究を中心に行う。

図形・画像系では図形および画像という視覚情報を媒体として、人間がコンピュータと知的に対話するための技術確立し、知的インタフェース・システムを提供する。このシステムは視覚情報に基づく推論機構、高精度高機能のカラー図形・画像入出力装置、図形・画像処理マシンおよび図形・画像ファイルより構成される。その機能は、人間が見ることによって情報を得たり他人とのコミュニケーションに使うような図形あるいは画像が表わす内容を理解してコンピュータの中の記述に変える機能と、コンピュータ内の記述を、人間が理解しやすい図形や画像で表示する機能の2つがその柱となる。

開発の方針は、図形・画像を理解するシステムを構築するための共通技術を研究開発する事に主力を置くとともに、知的インタフェース・システムを作るために図形・画像生成アルゴリズムの研究とシステム化技術の開発も併せて行う。また知的インタフェースを自然言語理解システムと組合せて構成するための技術開発も行う。

マンマシン・インターフェースを考える際、第一に考えなければならないのは、使い方の自然さと円滑さである。その実現をサポートするためのハードウェアの開発も重要な課題である。

本節で研究開発される自然言語や音声、図形・画像による知的インタフェース構成のための要素技術をもとに、使い易いコンピュータ実現のためのスマート端末を開発する。開発の方針は、研究開発期間の前半の内に、それまでに利用可能な要素技術を総合して、パイロット・モデル(前期)を作成し、試用評価を終て、後期のモデルを設計・製作することにする。パイロット・モデル開発の途中で各要素技術へのフィードバックを密に行い、本節における研究開発を有機的かつ効果的に推進する。

〔2〕 必要性

概要のところに、必要性が比較的詳しく述べられているので、ここでは、簡単に要約しておく。第一に、人間とコンピュータ間の使用言語の相違に基づくギャップを解消するためには、第5世代コンピュータに自然言語や音声、図形や画像を用いた柔軟な会話機能を付与する必要がある。

第二に、自動車のエンジンが我々の歩行能力を推進拡大した様に、第5世代コンピュータを、あたかも我々人間の思考能力を推進拡大させるための“思考のエンジン”として機能させる必

要がある。

知的インタフェース・システムは、以上を実現するための中核的基礎技術を確立するものであり、それにより、第5世代コンピュータに対する社会の広汎な分野の需要が一層喚起されることが期待できる。

自然言語関連による会話は、柔軟な会話機能と密接に関連することは明らかだろう。音声・図形・画像についてはどうだろうか。

第5世代コンピュータは専門家ばかりでなく、一般家庭も含めた非専門家ユーザが幅広く使用すると考えられる。その時、現在のようにプログラムをキーボードから入力する方式では、手続きの煩雑さがネックとなることが考えられるため、自然な会話形式でコンピュータへ音声入出力ができることが望ましい。また、第5世代コンピュータを開発する場においても、膨大なソフトウェアを生産性良く作成する必要があるため、知的プログラミングによるプログラムの自動作成及び自動検証と併せて、音声による入出力が必要とされる。

現在実用化されている音声認識システムは、単語単位の認識方式で、話者毎に認識対象単語を登録してパターン・マッチングによって識別を行う。数百程度の単語については実時間処理が行われているが、単語数を数千から数万に拡大すると、登録に要する手間や認識時間が実用的でなくなる。そこで認識単位を小さくして音素程度にすれば、登録すべき音素数は限られており、単語数に関する制約はほとんどない。認識単位を小さくすることによって調音結合の処理、セグメンテーション等の研究が重要になってくる。

音素単位で認識した記号系列から単語や文章へと組み上げていくには、構文で意味等のいわゆる言語情報を利用する必要がある。しかし現在の数値計算向きコンピュータでは能率が悪く、言語情報の十分な利用が達成されていない。

また、人間とコンピュータの会話で重要な音声の合成では、現在は文節や単語単位で録音しておいた音声を編集して出力する方式が音質の最も良い合成音を生成する。しかし語彙数や記憶容量の限界から、合成のための単位を音節や音素にして、合成できる語彙数を大幅に拡大する必要がある。分析合成方式や規則による合成方式等の研究をさらに進める必要がある。

その他、個人差の学習方式や韻律情報の抽出方法を発展させて実用化する必要がある。

社会生活における情報処理活動の主体は、言うまでもなく人間である。コンピュータによる情報処理システムを設計・製作する場合に大きな役割を果すのは、従って人間の知的活動である。

新しい情報処理系を作り上げようとする場合に、この人間の知的活動を効果的に引き出すシステムが必要であることが、最近広く言われている。また、その可能性を示す事例も生み出されている。このようなコンピュータによる支援のプロセスの過程にあって、推論あるいは思考の手掛りを、“視覚化”する事が有効であることは容易に想像できる。

新しいシステムを作るような場合でなくとも、何かデータを分析しつつ判断を進めるような作業をコンピュータがサポートするような場合、同様のことが言える。

このような局面は従来 CAD の分野でしばしば取上げられて来ている。そこでの問題点は、使い易いシステムを決定し、設計製作することが非常に困難であり、時間と人手が必要であるという点であった。

一方、図形や画像を入力する局面について考えるならば、人間が理解できる図形や画像から、それに含まれる意味・内容をコンピュータが理解し、コンピュータの中でそれを後から使える形で記号化し記述する機能が必要である。これと、推論・問題解決のメカニズムとによって、図形や画像を介して、コンピュータが人間とコミュニケーションを行うことが可能となる。

このようなコミュニケーションがコンピュータと人間との間でできるようになると、人間の知的活動を効果的に引出す真の知識情報処理システムの実現が可能になる。従って、図形・画像による知的インタフェースを開発する必要性は非常に大である。

〔3〕 内外技術の現状

(I) 国内技術

(a) 自然言語・音声系

国内では、京都大学で機械翻訳システムの研究が、九州大学では、語彙の意味分析の延長上に、機械翻訳システムの実現が模索されている。最近では、東大、東工大、名大、北大等の大学にも研究グループが存在し、この分野の研究が一層活発化している。電総研では、自然言語理解のための諸技術の開発、また電々公社武蔵野通研では、質問応答の研究が行われている。国立国語研究所では、日本語の分析が進められている。日本 IBM では、質問応答システム“やちまた”を作動させている。東芝、富士通、日立、日電等にも研究グループが存在している。

音声認識の研究については、我が国において多くの秀れた成果があげられているが、ここでは音声理解システムについて主として述べる。

国内における音声理解の研究開発は、京都大学を始めとするいくつかの大学と、電電公社武蔵野通研を中心として行なわれている。

京都大学で研究開発された LITHAN システムでは、101 単語の語彙からなる「計算機網への状態問合せと指令」をタスクにしており、男性話者 10 名の発声した 200 文章に対して、文章認識率 64 %、単語認識率 93 % を得ている。今後は、個人差学習機能の強化と韻律情報の利用等による能力向上を目差している。

電電公社武蔵野通研で開発された、列車の座席予約をタスクとしたシステムでは語彙数 112 で文節認識率 86 % を得ている。

京都工繊大、山梨大でも BASIC や FORTRAN プログラムを音声入力するシステムを開

発し、メーカでは日本電気をはじめとして、1979年には富士通、東芝、日立、三洋電機が相次いで音声認識装置を発表している。しかし、音声理解システムとしての本格的な検討は未だ行なわれていない。

また、電総研では、音素識別方式の基礎として、調音結合の処理方式や子音の識別方式、セグメンテーション等について研究し、成果をあげている。

(b) 図形・画像系

我国の図形・画像情報処理技術の水準は、世界的に見てもかなり高い。基礎から応用まで、多くの研究者、研究グループが研究を手掛けている。ハードウェアの低廉化がもたらす効果も大きい。すなわち、この分野の研究に着手する際、ハードウェアへの初期投資が以前に比べて低減し、大学の研究室レベルでの研究がやり易くなって来た事によって、研究者層の増大が促進されている。

技術水準について言えば、応用範囲が限られているとはいえ、各種実用システムが開発されて来ている。たとえば、郵便番号自動読取り装置の実用化は、歴史的にマイルストーンと言える。工場の生産ラインにおける画像処理システムの応用（LSIボンディングの自動化等）、医用画像への応用（白血球解析装置等）、リモートセンシング画像処理など多数の応用システムが生み出されている。

このように、国内におけるこの分野の研究は非常に多い。

画像理解の研究が屋外風景や航空写真、医用画像などを対象に、京都大学、東京大学、山梨大学、電総研、東芝などで行われている。工業用ロボットの目としての画像情報処理の研究が大阪大学、電総研、日立などで行われている。

(2) 海外技術

(a) 自然言語・音声系

海外では、1970年代初頭のMITの研究が著名であるが、その後、スタンフォード大で、自然言語理解システムの開発が行われ、その技術の延長がエール大で行われている。以上の大学は実用というより、自然言語理解のための基本技術の開発に重点を置いている。同様な立場から、BBNでは、構文解析のための武器としてATN方式を開発し、自然言語理解（音声理解も含む）システム実現に向けて種々の実験が進められている。

Xeroxでは、柔軟なマンマシン・インタフェースをもったパーソナル・コンピュータが、自然言語理解を含む新世代コンピュータを意識する立場から進められている。

自然言語による質問応答システムの実用化を目指す立場からは、ミシガン大、ダートマス大、カーネギーメロン大、SRI等で進められ一部実用に供されているシステムも出現している。

機械翻訳は、カナダ（モントリオール大）、ヨーロッパ（グルノーブル大等）で活発化し

ている。パリ大では、談話理解に重点をおいている。このようなヨーロッパでの動きは、米国にも波及し、最近では、テキサス大に、機械翻訳のグループが再編されている。

IBM では、サンノゼ研究所でデータベースに対する質問応答システムの研究開発が進められていた。最近になって、ワトソン研究所でも、従来のグループの他に自然言語理解のための新しい研究計画のあることが明らかにされた。

西独ではジューゲンス社の自然言語理解システムが、政府の援助の下に進められている。

英国のエジンバラ大でも PROLOG をベースにした自然言語理解システムの研究が行われている。

一方、音声研究は米国を中心に、スウェーデン、フランス、イギリス等で盛んに行われている。

米国では、国防省の ARPA のサポートによって、1971 年から 5 年計画で音声理解システム研究開発が行なわれた。当初 8 つの研究機関が参加したが、1973 年に 3 グループに統合され、1976 年 8 月に終了した。そのうち、カーネギーメロン大学が開発した HARP Y システムのみが当初の目標（適応化した多数話者の発声による、1000 語程度の語彙の連続音声を、意味的に 10 % 以下の誤り率で認識すること）を達成した。同大学では、HARP Y の前に HEARSAY-I, DRAGON, HEARSAY-II を開発している。最も結果の良かった HARP Y システムでは、5 名の話者の発声による 184 文章に対し、文章理解率 95 % が報告されている。タスクはニュース検索であった。

その他に、BBN 社の SPEECHLIS システムと HWIM システム、SRI と SDC グループによるシステムが ARPA 計画により作られた。いずれも文理解率は 50 % 以下であり、成功したとは言えないが、言語処理法や音声分析法等に成果があったと言われている。

ARPA 計画以外では、IBM, MIT, UNIVAC, BELL 研究所等で音声研究が盛んである。IBM のシステムでは、各話者に対して 1 時間以上の音声を用いて話者適応後、250 語からなるタスクに対して文理解率 95 % が得られている。

製品化された音声認識システムでは、いずれも単語認識方式を用いて Threshold Technology 社は最大 250 単語を 99 % 以上の認識率を得ており、Interstate Electronics 社はやはり 250 語のシステムを開発している。

(b) 図形・画像系

米国の国防省がサポートする「Image Understanding」のプロジェクトが 1975 年から発足し、MIT, CMU, Stanford 大学等、米国の情報処理研究の主要な大学が研究を推進している。

Stanford 大学では ACRONYM という、三次元モデルを用いて物体を認識するシステムの開発を行っている。MIT では、明るさの情報から形へ直接変換する研究を行っている。

CMU ではロボットの眼の開発を開始している。

〔 4 〕 研究開発内容

(1) 研究開発項目

(a) 自然言語・音声系

① 構文解析技術

入力された自然言語の文を構成する単語の認定（活用語尾処理を含む）を行ない（これは形態素解析ともよばれる）、単語の品詞の並びが妥当なものかどうかを文法規則を用いて検査し、単語と単語の結び付きを示す構文構造を作り出す。この構文構造は、意味解釈をするための中間結果として次段階の意味解析で利用する。

以上の様な構文解析を行なう諸技術、特に高速アルゴリズムを開発しシステム化する。意味解析技術と構文解析技術を融合させる技術を開発する。

② 意味解析技術

構文解析により、単語の品詞の並びの妥当性が検査されたら、それが意味をなす文かどうかを調べ、言語の種類に比較的依存しない意味構造を抽出する技術が意味解析技術である。したがって、意味構造をどのような形式にするか、また語用論的な知識を意味解析でどのように利用すべきかをも考慮しなければならない。また構文解析技術と談話解析技術とを利用して意味解析を行う技術を開発する。

③ 談話解析技術

複数個の文の系列からなる談話（discourse）を解析する技術である。これには、省略語の補強、代名詞の指示するものの決定、主題の把握等の文脈解析技術が中心になるので、語用論的解析（pragmatic analysis）、意味解析が関連する。以上の諸技術を研究開発する。

④ 文章合成技術

抽出された意味構造、もしくは推論・問題解決結果を自然言語に翻訳するための技術である。ここには、文法情報が主として用いられるが、自然言語として“自然”な文を出力するためには、代名詞化、適度の省略等が必要になる。そこで談話解析技術をも応用しなければならない。ここで、構文解析で用いる文法と別の文法を利用して文章を合成すれば、機械翻訳システムが実現される。以上の諸技術を研究開発する。

⑤ 言語データ・ベースの構築

自然言語理解に必要な日常言語に対する辞書データ・ベース（数万語）と科学技術用語データバンク（50万語以上）を作成する。シソーラスの研究も行う。

⑥ 自然言語処理装置

以上の形態素解析や構文解析、意味解析等に用いられる基本技術は、非数値計算を主体

とする記号処理技術である。記号処理技術に適し、しかも効率的な構文解析アルゴリズムと、辞書や文法規則等を含む知識ベースを効率的に検索するための連想アルゴリズムをハードウェア化した自然言語処理装置を研究開発する。

⑦ 多言語基本文法の作成

日本語、英語を含む数ヶ国語の基本文法（重文、複文を含む）を作成する。

⑧ 音素識別方式の研究

音素は音声の中で最も小さい単位で、母音や半母音、 $|p|$ 、 $|r|$ 、 $|m|$ 等の子音からなる。連続音声の中では、前後の音韻環境によって音素の特徴が様々に変化するため、このような調音結合の処理法の確立が必要である。また、連続音声から音素記号列を抽出するためのセグメンテーションも重要な課題であり、言語情報の利用が考えられる。子音の中でも鼻音や流音、半母音等は識別が難しく今後の研究が期待されている。

子音と母音を組み合わせた音節単位の識別方式は、その組み合わせの数だけ識別数が増加するが、単語単位の認識よりも格段に語彙の拡張性が良く、音素識別方式と関連させて研究する必要がある。

各方面での音声研究を効率化するため、標準音声データベースを作成する。

⑨ 文章理解方式の研究

音声分析レベルで抽出した音素記号列から音節、単語、文節、文章へ認識対象を上げていくときに、構文規則、意味情報、語用論等を用いる必要がある。このような言語情報を効率良く利用するには、数値計算よりも記号処理に適したコンピュータが必要である。また、言語情報を音声理解用に、音形規則、単語辞書、構文規則等の形式で格納した知識ベースが必要であり、このようなデータを辞書形式で作成することも重要な研究課題である。

イントネーション等の韻律情報は、疑問調等の文章の意味を把握するために重要であるばかりでなく、単語や文節のセグメンテーションにも大きな役割を果たす。

また、音声波形から抽出した音素記号列はある程度の誤りを含むため、その修正には、大局的な言語情報を利用する方式を開発する必要がある。

⑩ 音声合成方式の研究

音声合成には、原音声をそのまま用いる録音編集方式、1ピッチ波形や子音素片の形式で記憶しておいて目的に応じて合成する素片編集方式、音声分析で得られた特徴パラメータを記憶しておいて合成する分析合成方式、与えられた文字記号列から言語的・音響的規則を問いて合成する規則による合成方式がある。分析合成方式は、最近T Iが“Speak & Spell”で実用化し注目されている。音質は録音編集方式に比べて遜色なく、語彙数も1万以上のものでできている。しかし規則による合成方式は音節や音素等の小さい単位を用いるので合成対象の文章や語彙数の柔軟性が格段に良い。アクセントやイントネーションを

始め多くの研究課題が残っているが、人間の音声生成過程に最も近いという意味で、研究開発の意義が大きい。規則のアルゴリズムやソフトウェア開発が重要である。

⑪ 話者個人差の学習方式等

音声生成器官の形や大きさ、発声のしかたは話者によって異なり、音声認識上で障害要因となっている。個人差の的確な抽出法、その正規化法、音声を認識していく過程で個人差を学習していく方式等を研究開発する必要がある。個人差の抽出は一方では、話者認識に密接に結びついており、電話等の手段によって遠隔地からでも話者の識別が可能になることが考えられ、応用分野は広い。

⑫ 音声理解システムの開発

音素識別方式、文章理解方式、音声合成方式、個人差の正規化等を総合して、音声専用ハードウェアや適切な規模のコンピュータを用いて、音声理解システムを開発する。

前期（第Ⅰ期）には、音素識別方式を中心に、他の諸方式の基礎研究を進める。後期（第Ⅱ期）には、音声理解方式を中心に、言語情報の利用を図り、最後に音声理解システムにまとめあげる。

（b）図形・画像系

図形・画像を人間とコンピュータとの知的コミュニケーションの媒体として利用するために解決しなければならない技術的隘路は、図形・画像をコンピュータ内で抽象レベルで扱うための技術にある。従って研究開発の方針としては、この隘路を解消する技術を最重点にし、合わせて、知的インタフェース・システムを構成するために不可欠いくつかのシステム要素、ならびにシステム化技術の研究を行い、目標達成を図る。

① 図形・画像の構造化技術の研究

入力した図形・画像から、構造化された抽象レベルの記述を作るために必要な基礎技術を確認する。本研究開発項目は、④特徴抽出の研究、⑤記述法の研究、および⑥知識利用方式の研究より成る。

④ 特徴抽出の研究では、入力された図形・画像を表現し、記述するための特徴パラメータの特徴記述因子（領域、線、孔、距離、その他）を高度化することを目指し、新しい特徴記述因子の開発をしつつ、特徴記述因子と図形・画像が本来持つ情報との関係を明確にすることによって、特徴記述因子間相互の構造化性を究明する。

⑤ 記述法の研究では、図形・画像に含まれる対象を、コンピュータによる推論や問題解決のプロセスの中で扱い得るようにするための手法の研究を行う。基本となるのは、図形・画像が表わすものの構造を明らかにする研究である。これにはいくつかのレベルが存在する。最も入力データに近いレベルでは、図形・画像入力から直接得られた特徴記述因子を構造化して、あるまとまった図形あるいは画像の部分要素を組み立てること

であり、順次高いレベル構造が抽象化されていく。

㉔ 知識利用方式の研究では、処理の対象ならびに処理方式の両方に関して、処理を進める上で不可欠な知識が駆使されるが、その知識を扱う手法の研究を行ない、知的図形・画像理解の基礎を築く。1つには、図形・画像を処理する際に、そこに表わされている対象に関する知識が不可欠である。たとえば、医用画像などでは、医学的知識がそれに当たる。他方、図形・画像インタフェースを構成する場合には、図形・画像の処理技術（第㉔項で開発される）に関する知識が必要である。たとえば、線画が表わし得る矛盾性に関する知識などである。このような知識は、高度なシステムを作ろうとすれば、たちまち大型になってしまうようなものであるから、特に、画像ファイルとの関係においても、十分な技術開発を行わなければならない。

㉕ 図形・画像生成アルゴリズムの研究

コンピュータの中で処理される各種データを人間が目で見えて理解しやすいように、図形・画像で表示する必要がある。現在すでに、オフィス・オートメーション用の端末機器等には、数表を判り易いグラフ表示に容易に変換できるシステムが組み込まれているものも出回っている。

ここでは、単に数表のグラフ表示に留まらず、図形・画像として、コンピュータの推論・問題解決の過程に現われる情報をディスプレイ等を介して表示するための、図形・画像生成アルゴリズムを研究開発する。

㉖ システム化技術の開発

知的インタフェース・システムを構成するための技術を開発する。開発の核は、前項での研究成果を取入れて作る㉗アルゴリズム・ベースの開発と、そのアルゴリズムを高度にプログラムするための㉘図形・画像処理用プログラム言語の開発である。

㉗ アルゴリズム・ベースの開発

図形・画像を媒体とした知的インタフェースを構成する場合には、それまで開発した種々のアルゴリズムを問題対象に適した形に変形して使う形式を取るようになる。そのために、そのベースとなる種々のアルゴリズムを集積したアルゴリズム・ベースを開発しなければならない。このアルゴリズム・ベースには、各アルゴリズムの特徴および使い方が、知識ベースとして推論レベルで使える形で含まれているようなものとなる。

㉘ 図形・画像処理システム構築用プログラム言語の開発

図形・画像の生成も含めた処理のアルゴリズムは、アルゴリズム・ベースとして、ファイルへ蓄積されるので、本プログラム言語は、それらを用いて、システムを記述するための言語として機能することを目標とする。それには、アルゴリズム・レベルでの推論機能を、言語自身が持っていなければならないし、また、それに付随する知識を利用

する機構も有する。

④ 図形・画像処理マシンの開発

“原理的に実現可能”であることと，“実際に実現した”こととは、全くその価値が違ふことがある。特に、情報処理技術は、言わば全て“原理的には実現可能”な技術を扱っているのであるから、知的インタフェース・システムでは、人々がその真価を認識するような“実際”のシステムのデモンストレーションが不可欠である。知的インタフェースの処理速度と質を保障するために、② 図形・画像処理用高速演算装置、⑥ 高機能高精度図形・画像入出力装置及び、③ 大容量図形・画像ファイル・マシンの開発を行う。ただし、本ハードウェアは、第5世代コンピュータの開発プロジェクトの外部に於いて、適当なハードウェアの開発がなされ、本プロジェクトへ時期的、技術的に、その機能を提供しうるのであれば、それを採用する方が望ましい。

⑤ 自然言語と図形・画像との関連づけの研究

知的インタフェースは、自然言語、音声等の技術と、図形・画像等の技術とを融合することにより完成する。本項目では、その融合技術を研究開発することを目的とし、図形・画像の内容を、言語で記述したり、あるいは言語で記述された内容を、人間の理解しやすい図形・画像の形態に変換して表示するための語彙の開発、語彙と図形・画像との変換ルールの開発等を行う。

(2) 研究開発スケジュール

(a) 自然言語・音声系

研究開発項目\年度	1	2	3	4	5	6	7	8	9	10
(1) 構文解析技術	構文解析方式の研究(I)					構文解析方式の研究(II)				
	プログラム化					改良		ハードウェア化		
	文法開発(I)					文法開発(II)				
(2) 意味解析技術	意味解析方式の研究(I)					意味解析方式の研究(II)				
(3) 談話解析技術	意味表現形式の研究(I)					意味表現形式の研究(II)				
(4) 文章合成技術	談話解析方式の研究(I)					談話解析方式の研究(II)				
(5) 言語データベースの構築	文章合成方式の研究(I)					文章合成技術の研究(II)				
	言語データベース構成法の研究(I)					言語データベース構成法の研究(II)				
	シソーラスの研究(I)					シソーラスの研究(II)				
(6) 自然言語処理装置	辞書データベースの構築					改良		運用技術		
	方式の研究					設計		開発		評価
(7) 多言語基本文法の作成	文法開発									
	プログラム化					試作		評価		

研究開発項目\年度	1	2	3	4	5	6	7	8	9	10
(8) 音素識別方式の研究	音素識別方式の開発					専用装置の開発				
(9) 文章理解方式の研究	基本方式の研究			単純文の理解方式の開発			複雑文の理解方式の開発			
(10) 音声合成方式の研究	音声合成方式の開発					音声合成システムの開発				
(11) 話者個人差の研究	基本方式の研究			個人差処理方式の研究			改良			
(12) 音声理解システムの開発	音声データベースの作成			単語辞書等データベースの作成			専用知識ベースの作成			改良
	音声理解システムの開発									

(b) 図形・画像系

研究開発項目\年度	1	2	3	4	5	6	7	8	9	10
(1) 図形・画像の構造化技術の研究	特徴抽出の研究(I)			特徴抽出の研究(II)			特徴抽出の研究(III)			
	記述法の研究(I)			記述法の研究(II)			記述法の研究(III)			
	知識利用方式の研究(I)			知識利用方式の研究(II)			知識利用方式の研究(III)			
(2) 図形・画像生成アルゴリズムの研究	図形生成アルゴリズム(I)			図形生成アルゴリズム(II)			図形生成アルゴリズム(III)			
	画像生成アルゴリズム(I)			画像生成アルゴリズム(II)			画像生成アルゴリズム(III)			
(3) システム化技術の開発	アルゴリズム・ベース開発(I)			アルゴリズム・ベース開発(II)			アルゴリズム・ベース開発(III)			
	図形・画像処理用プログラム言語(I)			図形・画像処理用プログラム言語(II)			図形・画像処理用プログラム言語(III)			
(4) 図形・画像処理マシンの開発	第1次設計・試作					第2次設計・試作				
(5) 自然言語と図形・画像との関連づけの研究	表現用語彙(I)			表現用語彙(II)			表現用語彙(III)			
	統合化(I)			統合化(II)			統合化(III)			
	認知科学(I)			認知科学(II)			認知科学(III)			
(6) 基本応用システム	基本設計・試作			中間目標システム設計・試作				目標機開発		

(3) 研究開発の目標・仕様

自然言語・音声という言語情報、もしくは図形および画像という視覚情報を媒体として、人間がコンピュータと知的に対話するための技術確立し、知的インタフェース・システムを提供することを目標とする。

音声の研究開発の目標・仕様は次のとおり。

① 対象語彙は、コンピュータ関連用語や科学技術文献等の中の1分野につき、専門用語および常用単語を含める。

② 話者適応機能を持ち、不特定話者を対象とする。

③ 日本語および英語の音声出力を可能とする。

④ 識別処理は実時間に近い能力を目標とする。

図形・画像系の研究開発の目標・仕様は次のとおりとする。

① 図形・画像を介して、人間がコンピュータと円滑に対話するシステムの構築に必要なソフトウェアならびにハードウェアの開発を行う。

② 対象とする図形は、中規模以下の機械の設計図面程度の複雑さ、画像は医用写真程度の複雑が取扱えるものとする。

③ 処理速度は、人間との対話手段として円滑さを損なわない程度に十分速いものとする。

④ 中間目標として、扱う図形・画像の複雑さを最終目標の約7割を狙う。処理速度については、中間目標は特に定めずに、処理方式の確立に主力を注ぐこととする。

2.3 新アーキテクチャ

2.3.1 述語論理マシン

(1) 概要

述語論理マシンは、述語論理型言語によって記述されたプログラムを効率よく実行することを目的にした計算機である。

述語論理型言語というのは、文字通り、述語論理の枠組に従ったプログラミング言語一般を指すものである。

論理は論証を厳密に記述するための機構ないしことばであるが、特に述語論理は、数学的な証明の表現を行なうのに必要な表現力を十分に備えているばかりでなく、厳密な表現手段としては、人間による推論形式に最も近く、したがって人間にとって自然な形式であるといえることができる。

プログラミング言語の目的は、解くべき問題とその一般的な解法、すなわち、アルゴリズムを記述して、人間から機械に伝達するための表現手段を提供することである。アルゴリズムの中味を化学方程式に書くと、

アルゴリズム = ロジック + コントロール

となる。ここで、ロジックというのは、問題を解くのに使われる情報ないし知識の体系のことであり、コントロールというのは、それらの情報ないし知識をどのように使っていくかの制御のことである。

述語論理型言語によるプログラムでは、アルゴリズムのロジック部の記述に述語論理を用い、コントロール部はそれ専用の簡単な言語を用いたり、実行機構に委ねたりすることになっている。したがって、ロジック部は公理、定理、それに推論規則などの集合であり、コントロール部は演繹の進行手順から成り、両部は完全に分離して存在することになる。

これに対して、従来のプログラミング言語では、命令あるいはステートメントなどのような、プログラム記述の最小単位のレベルで、すでにロジック部とコントロール部の化合が行なわれている。この際立った対照は述語論理型言語の革新性を端的に示すものである。この化合は、もともとノイマン型計算機の基本的な性格に由来するもので、その遺伝は高級言語のレベルにも濃厚に認められる。

これを裏返していえば、述語論理型言語をノイマン型計算機に適合させるのには、相当な困難があるかも知れないということである。しかも、この問題以前に、述語論理をプログラミング言語化するために克服すべき致命的な課題があった。述語論理型言語は計算を定理の証明という形で実現するものであるが、これは正に定理の証明を機械で自動化することを意味している。

定理証明の機械化は、原理的には、公理あるいは定理の組合せに推論規則を適用して新しい定理を導くという操作を繰返して、目的とする定理が生み出されるまで続けるという形をとる

ことになる。しかし、これを組合せをしらみつぶしに作って調べるといようなやり方をしたのでは、組合せの数が多くて実用にはならなくなる。そこで、実際的なアプローチとしては、述語論理の形式に制限をつけて（したがって、それだけ自然さが失われる）、組合せのケースを圧縮するか、あるいは、人間の直感力に相当する能力をコントロール部に組込むというようになる。後者のアプローチは、むしろ、今後の研究課題であり、現状で確実に使いものになることが実証されているのは、前者のアプローチによるものである。それでは、述語論理をHorn節という形式に制限することになっている。

そして、このHorn節形式で表現された公理あるいは定理を、従来プログラミング言語での手続き形式として読み換えることによって、ノイマン型計算機に適応させるのに成功した述語論理型言語に、PROLOGがある。

PROLOGは英国において開発され、以来、同国ならびにフランス、カナダの大学において、実装化、改良、いくつかの大規模応用における有効性実証など広汎な努力が続けられている。これらの実績を通じて、人工知能分野を中心に、データベース関係、ある種の技術計算類などにおけるPROLOGの定評は、ほぼ確立されたものとみることができる。この間にあって、米国はPROLOGに対してほとんど沈黙の状態にあるのが注目される。これは米国特有のNIH(Not Invented Here)的気風のしからしめるところであるらしい。

PROLOGによって述語論理型言語は完全に実用の段階に入ったといえることができる。今後の課題は、より大規模な応用のために、より高いパフォーマンスの水準に達することである。このためには、単にノイマン型計算機の枠内での改良を考えるだけでは不足で、述語論理本来の性格に適合した計算機アーキテクチャの追求が必要になっている。

述語論理の基本演算は、演繹(deduction)である。演繹の基本操作は、変数を含んだ2つの記号列に対して、変数に許された範囲での代入置換を行ないながら、列全体のパターン的一致を調べることである。長さの異なる多数の記号列を記憶し、記号パターンによって適当な候補の列を探し出す機能、上述のようなパターン照合(ユニフィケーション)を行なう機能、一致のとれた記号列に対してある演繹操作を行なって、新しい記号列を作り出す(リゾリューション)機能、こうした機能を繰返しても求める結果に到達できなかったとき、まえに見捨てられていた候補の列に戻って再出発する(バックトラック)機能、これらの機能を効率よく実行できる計算機というのが、述語論理マシンのイメージである。

開発すべき述語論理マシンに期待される性能は、現在のノイマン型マシンで数秒から数10秒を要する問題を、ミリ秒で処理し終える程度のものである。これだけの性能向上を実現するためには、極めて高度な並列処理機能の導入が必要である。

(2) 必要性

述語論理マシンは述語論理型言語によるプログラム(以下では単に述語論理プログラムと略

す)を効率よく実行するために存在する計算機であるから、その必要性の根源的な部分は、結局、述語論理型言語そのものの有効性・必要性に依存することになる。そして、必要性の残りの部分は、述語論理プログラムを述語論理マシンで実行することの合理性にかかっている。

述語論理型言語の必要性・有効性について考える場合に大切なことは、この言語が従来の情報処理の枠を超えた新しい応用分野、すなわち、知識情報処理を活動の場としていることである。したがって、その評価はこの新しい場に即して行なわれなければならないのである。「脱工業化社会の到来」の著者であるD.ベルによれば、知識とは“事実あるいは思想を述べた論述の組織的なまとまりで合理的な判断や実験結果を提示するもの”で、“通信媒体により、体系的な形で他人に伝達される”ものである。このような知識を対象にした情報処理の内容は、必然的に人間の知的活動と呼ばれているもののうちの機械化可能な部分ということになる。何がどのようにして機械化できるかの見通しについては、当然議論のあるところであるが、少なくとも現在までに人工知能研究の成果として得られているものは、その確定分であるといえる。

知識の機械的な表現は、記号化された形式的表現、すなわち、記号列である。このような形式的表現の中心は、いうまでもなく、述語論理による命題形式と数学における関数形式である。序に言えば、従来の情報処理に用いられているデータ構造形式も、この仲間に加えてよいものの1つである。

知識に対する知的な操作のうち、機械化が原理的なレベルにせよ保証されているものとして有力なのは、論理的な推論や関数式に対する変形操作などであり、さらに現在LISPで行なわれているような、データ構造に対する記号的変形操作もこれに含めることができる。

以上が、知識情報処理において述語論理、したがって述語論理型言語に期待される役割の説明である。つぎは、述語論理マシンの存在理由を明らかにする番であるが、それはただよりよい効率の追求ということにすべてがかかっている。前述したように、述語論理型言語とノイマン型計算機の間には大きな不適合があるので、最初から述語論理プログラムの目的にして、それに適したアーキテクチャを開発すれば、相当な性能の改善を期待することができる。まだ述語論理マシンの具体化がなされていないので、これを直接示すことはできないが、LISPマシンでの実績などからの類推によって若干の見通しをうることができる。

[3] 内外技術の現状

(1) 国内技術

述語論理における定理証明の機械化については、わが国においてもすでに1950年代からの研究史があるが、実用的な形での述語論理プログラミングの開始には、やはりPROLOG言語の導入をまたなければならなかった。

PROLOGの処理系は、電総研をはじめ、電電公社武蔵野通研、東京大学などに実装され、データベース処理、自然言語処理、問題解決などにおける実験的使用が試みられている。

その結果はPROLOGの有効性を裏付けるものであった。現在は、さらに、PROLOGに対する効率化のための改良、並列処理化、図形言語化などに研究者の関心が集まっている。

また、ここ1、2年来、述語論理マシンの実現、特にデータフロー計算機としての実現方式についての研究が各所で開始されている。

(3) 海外技術

ヨーロッパ各地におけるPROLOGへの関心は非常に高く、これを巡って多面にわたる研究が精力的に進められているようである。実用化の面では、性能の良いコンパイラが実現されたため、本格的な応用レベルでの研究が広汎な分野において展開されつつある。また、PROLOGに対する改良提案も活発に行なわれており、ソフトウェア工学指向の新機能をどのように取入れるか、コントロール部のあり方は、といった所にも興味も呼んでいる。

PROLOGの並列実行方式、データフロー形式のPROLOGマシンを対象とする研究が、その口火を米国で切っているのが注目される。

米国においては、PROLOG以外の述語論理型言語の研究が多くなる傾向にあるようである。

(4) 研究開発内容

(1) 研究開発項目

研究開発内容を大分類するとつぎのようになる。

- a) 述語論理型言語の開発と実装化
- b) 基本技術開発
- c) 述語論理マシンの実現

以下、これらの項目をさらにブレイク・ダウンして説明する。

a) 述語論理型言語の開発と実装化

基本的には、PROLOG言語に即したものと、自然演えきの考え方に従うものの2系統を、並行的に追求する。

実装化については、プログラム開発の局面での効率化を目的にした高級言語計算機としての実現と、大規模なプロダクション・ジョブの実行効率化を目的にした、コンパイル方式での実現とを、並行して進めるが、機種上での実装に当っては、その用途に合わせていずれかに特化することもある。

以上をまとめると、つぎの通りである。

- PROLOG系
 - i. 拡張PROLOG言語開発
 - ii. 実装化：コンパイル方式および高級言語計算機方式
- 新言語系
 - i. 新述語論理型言語開発
 - ii. 実装化：（同上）

b) 基本技術開発

述語論理プログラムの実行速度を飛躍的に向上させるための基本技術の開発を行なうものである。性能向上のための基本的なアプローチは、プログラム並列実行方式の導入と、述語論理型言語の特性に適合する専用のプログラム実行機構の導入の2点である。これらの基本技術は、その性格上言語のやゝ細部にわたる構造に依存する部分が多く、上述のように2系列の述語論理型言語を追求することに対応して、基本技術開発も一部で2系列化することが必要である。

ここでの開発内容の概要はつぎの通りである。

○並列化方式の追求

- | | | |
|-----------|---|------------------------|
| i. 基礎検討 | { | 並列制御方式
(PROLOG系並列化) |
| | | 並列計算モデル
(高度並列化) |
| ii. 実装化検討 | { | 資源管理手法 |
| | | 並列パターン照合機構 |

○専用の機構の開発

- | | | |
|---------------|---|---------|
| i. 連想処理機構 | { | 連想プロセッサ |
| | | 連想メモリ |
| ii. 節格納用構造メモリ | | |

c) 述語論理マシンの実現

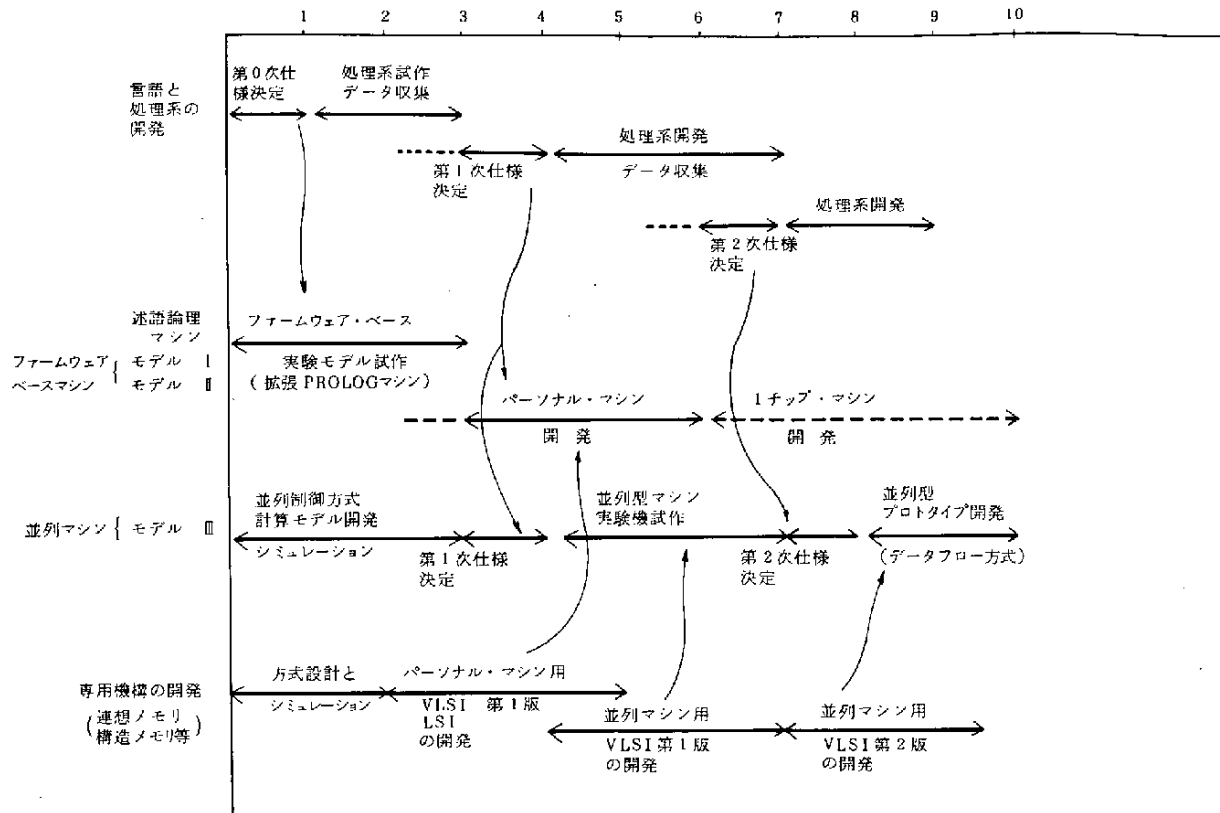
2系列の論理型言語のそれぞれについて、プロダクション・ジョブの処理を目的にする大規模マシンと、プログラム開発用を目的にする小規模(パーソナル)マシンのバリエーションが考えられる。また、開発のステップとして、開発実験的モデルと本番開発としてのパイロット・モデルの2段階が必要である。しかし、これらのすべての組合せを実現することは、明らかに現実的ではない。必要な実現時期と、前提となる要素技術の完成見込時期とを考慮に入れて、最小限必要と考えられるものを選別すると、つぎの3モデルに絞られる。

- 1) モデルⅠ：開発実験モデル
- 2) モデルⅡ：専用ファームウェア・ベース・マシン
- 3) モデルⅢ：並列述語論理マシン

(2) 研究開発スケジュール

研究開発スケジュールは次頁の線表の通り、

述語論理マシンの研究開発スケジュール



(3) 研究開発の目標・仕様

① 拡張 PROLOG 言語

主目標は知識情報処理システムの研究開発でのツールとなることであり、現在までの研究成果を集大成して、仕様に反映するという程度のものを予定。初期に第ゼロ次仕様を決定し、実験モデルを試作する。中期に第1次仕様を決定し以後実用に供せられることが必要。

② 新述語論理型言語

長期にわたる安定した使用に耐えられるだけの完備した仕様をもつものでなければならない。中期末までに最終仕様を決定し、プロトタイプ・マシン上での実用化を目指す。

③ 述語論理マシン

モデルⅠ：開発実験モデル

前期に完成を目標とする PROLOG ベースの小規模マシンで、ファームウェアによって実現するもの。

性能目標は現行の通常マシン上にソフトウェアで実現されているものの約10倍の水準である。コンパイル方式と高級言語計算機方式の両方が利用可能である。

このモデルの開発目的は、PROLOGマシンひいては述語論理マシンの性能的な可

能性を見極めるための実験台を提供することである。

モデルⅡ：専用ファームウェア・ベース・マシン

中期に完成を目標とする述語論理マシンで、理論研究や言語研究をサポートするパーソナル計算機としての規模をもつものである。専用ファームウェア・ベース・マシン（記号処理向UHM^{*}）としての形式に従うことを想定している。

性能目標は現行のものの数10倍から100倍弱の水準であるが、高級言語計算機方式としての性能に重点をおくものとする。

このモデルの開発目的は、商用パーソナル述語論理マシンのパイロット・モデルを提供することである。

モデルⅢ：並列述語論理マシン

1990年完成を目標とする高度並列化述語論理マシンで、大規模な述語論理プログラムを高速処理するための共同利用向き大型機であり、データフロー方式の利用を想定している。

性能目標は、現在の水準に対して数100倍から数1,000倍の線で、主としてコンパイル方式での性能に重点をおくものとする。このモデルの開発のために、中期に実験機を試作する。最終のプロトタイプ・マシンの目的は大規模述語論理マシンの実現可能性を実証するとともに、このクラスの商用機のためのパイロット・モデルとなることである。

2.3.2 関数型マシン

〔I〕 概要

第5世代のコンピュータにおいて重要になる知識ベースの実現とソフトウェアの生産性の向上には、関数型言語などの理論的基盤をもつプログラミング言語の利用が望ましい。一方、LISPとかINTERLISPで代表される関数型言語は以下の理由から、従来のコンピュータでは処理速度が上がらない。

① 関数型言語処理で標準的に利用されるリスト構造データを順次に追う処理は従来のコンピュータは得意でない。

また、要求される性能が現在のコンピュータの1桁～2桁上であるので、関数型言語処理に向いたマシンを新たに設計し、利用可能にする必要がある。

計算機のアーキテクチャの面から、知識情報処理をサポートするには、次の性能が望まれている。

① ネットワークの共通リソースとしての関数型マシン

現在の汎用コンピュータ（例えば、4MIPS）のリスト処理スピードの10～1,000倍。

*UHM: universal host machine

② パーソナル・マシンとしての関数型マシン

卓上型のマシンとして、現在の汎用コンピュータ（例えば、4 M I P S）のリスト処理スピードの1～10倍。

関数型マシンは関数型言語の利用者に役立つとともに、述語論理型言語等の言語の実行メカニズムとしての役を果たす。これにより、知識ベースを利用可能なものと出来るし、ソフトウェアの生産性の向上に貢献する。

関数型プログラミングとソフトウェアの生産性：

I B M社のBackus氏は現在のプログラミング言語には次のような欠点があると指摘している。

現在の言語にはフォン・ノイマン・マシンの概念を引きついだ、フォン・ノイマン・プログラミング言語（言語のモデルがフォン・ノイマン・マシン）であり、

① プログラム作成者の思考が分断される。（Intellectual Bottleneck）

一語のデータを取り出しては処理し、一語のデータを結果として格納することの繰り返し。

② ストラクチャード・プログラミングも、フォン・ノイマン・プログラミング言語に、ある種の秩序を入れたぐらいで、本質的な解答にはなっていない。

Backus氏はこの問題をF P（Functional Programming:関数型プログラミング）で解決出来ると論じている。

① F Pでは、変数の値にhistory sensitivityがないのでプログラムの論理を考え易い。

—関数合成が自然の形でやれる—

（call by name/value, 変数のスコープ等に悩まされることがない。）

② 集合演算を考えることで、繰り返し文（for I=1 to N）を排除出来る。

従来のプログラミング言語の欠点を除くべく、Backus氏の指摘のように、F Pの技術の適用を検討すべきであろう。

但し、現在P L / IとかC O B O Lでプログラミングしている者が関数型言語（例えば、L I S P言語）に切り替えて、すぐに効果を上げられるとは考え難い。

「教育」とか「慣れ」の問題の他に次の2点を研究する必要があるだろう。

(i) 関数型言語自身の表現形式

M形式 / S形式, 関数グラフ形式等

(ii) 関数型言語の持つべき機能

② 必要性

以下に現在のマシンとの相違点及び、第5世代コンピュータの他のマシンとの関係を示すこ

とで、関数型マシンの必要性を示す。

(1) 現在のマシンとの相違点

- ① 関数型言語を高速に処理する。
- ② リスト構造のデータに関する次の処理が速い。

— トリー・ウォーク (Tree walk)

— 要素選択 (Component selection)

— パターン・マッチング

低レベルの「単なる文字列の一致」から、高レベルでは「プロダクションや Horn クローズの処理」に至る、パターン・マッチングの処理の高速化が期待されている。

- ③ 関数の呼び出し、特にリカーシブな関数呼び出しを高速に処理する。

— スタッキング機構

- ④ 動的な記憶域管理を高速に実行する。

— 並行ガーベジ・コレクション機構

(2) 第5世代コンピュータの他のマシンとの関係

関数型マシンは関数型言語の利用者をサポートするだけでなく、PROLOG等を処理する述語論理型マシン、抽象データ・タイプ等を取り入れた言語のための抽象データ型マシンの実行メカニズムとしての役割をはたす。

また、集合演算機能を得意とする関係代数マシンも関数型マシンで実現することが十分考えられる。

関数型マシンの実現方式としてデータフロー・マシンがあり、興味ある研究がおこなわれている。

(3) 内外技術の現状

以下に関数型マシンの処理をインタプリタの入力となる言語レベルの観点から分類し、内外技術の現状を示す。

- (i) 言語を直接インタプリタ
- (ii) 言語→S式；S式をインタプリタ
- (iii) 言語→(専用ML)；(専用ML)をインタプリタ

この方式のLISPマシンの研究が多い。

(注) MLは機械語 (Machine Language の略である)。

- (iv) (ii)と(iii)の両方式をサポート

(電電公社 E L I S にその例が見られる)

これらのインタプリタの高速化を目指すマシンの研究が国内外で進んでいる。

次に現在研究段階にある関数型マシンの実現方式を分類する。

I. 専用MLの形式

(1) 逐次実行型専用ML

- Deutch の L I S P マシンのコード
- M I T の C O N S プロセッサのコード
- ユタ大の M B A L M コード (図 2.3 - 1 参照) 等

(2) 関数型グラフ言語

- ユタ大の F G L (関数型グラフ言語)

II. インタプリタの制御方式

(1) マイクロプログラミング方式

(a) 商用のマイクロプログラム方式コンピュータを利用する方式。

- ユタ大学の M B A L M マシン (B 1 7 2 6)

(b) 専用のマイクロプログラム方式コンピュータを設計する方式。

- C M U の L I S P マシン
- M I T , C O N S プロセッサ
- M I T , Sussman の 1 チップ・プロセッサ (図 2.3 - 2 参照)
- 電総研, L I S P マシン II
- 京大, N K 3
- 慶大, L I S P マシン
- 神戸大, L I S P マシン
- 電電公社, E L I S

(2) ワイヤード・ロジック方式

III. インタプリタの制御の多重度

(1) 単一制御

現在までに研究・試作されている L I S P マシンは逐時実行型である。

(2) 多重制御

関数評価の並列実行

(a) データフロー方式

- 東大の L I S P マシン

(b) リダクション・マシン方式

- Mago のリダクション・マシン
- Berkling のリダクション・マシン

(c) デマント・ドリブン方式

—ユタ大学の Keller マシン (図 2.3 - 3 参照)

現在の OS のプロセス管理方式の「プロセス」を「関数呼び出し」に置き換えたようなもの。

(プロセッサの個数は、例えば 1000 個を考えている)

(注) Mago のリダクション・マシンは言語の文字列を見渡し、リダクション (式の評価) のできる部分の集合に対して、並行してリダクションを行う。この動作の繰り返しで「言語の実行」をする。一次元配列のセルと、トリー型配列のセルとで構成される。VLSI 向きの構成である。(図 2.3 - 4, 2.3 - 5, 2.3 - 6 参照)

IV. シンボルテーブル・サーチ方式

シンボルテーブルのサーチの高速化のためには、順次にテーブルを検索する処理を高速化する方式やハッシング方式がある。

(1) テーブル・サーチ方式

連想メモリの利用等、ハードウェアの論理で高速化することが考えられる。

(2) ハッシング方式

ファームウェアやハードウェアでハッシング検索機構を実現し高速にする。

理化学研の FLATS マシンはハードウェアでハッシングを行っている。

V. データ形式と構造メモリ

関数型マシンの実現には、以下のようなデータ形式が利用される。

- ・タグ付データ
- ・ディスクリプタ

また、インタプリタの中では、処理上次のようなデータ形式が利用される。

- ・スタック
- ・ランタイム・シンボル・テーブル
- ・ヒープ

これらのデータ形を高速に扱う機構が必要となる。特に VLSI を使用した構造メモリ (連想メモリ等) が期待される。

IV. ガーベジ・コレクション方式

記憶域の動的な管理を行うガーベジ・コレクションの方式は、各種研究が進んでいる。

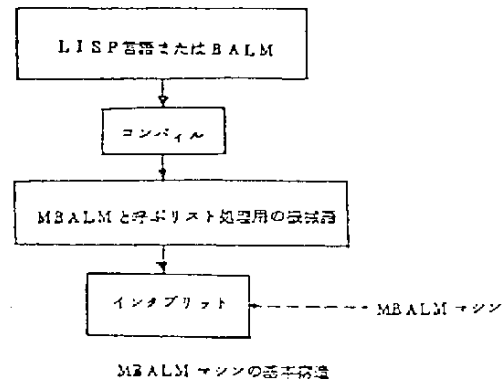
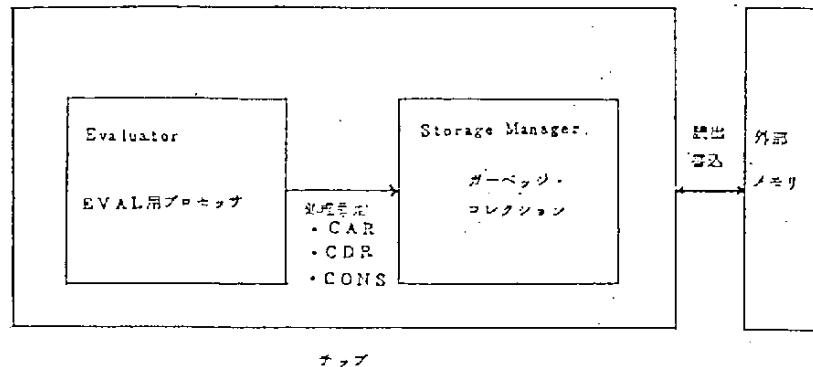


図 2.3 - 1 ユタ大学のマイクロプログラム化LISPマシン



チップ
とMITのLISPプロセッサも本質的には、ユタ大学的方式と同様
LISPプログラムを一意オブジェクト・コードに変換した後に実行する方式
(コンパイル方式)

図 2.3 - 2 MITのVLSI化LISPプロセッサ

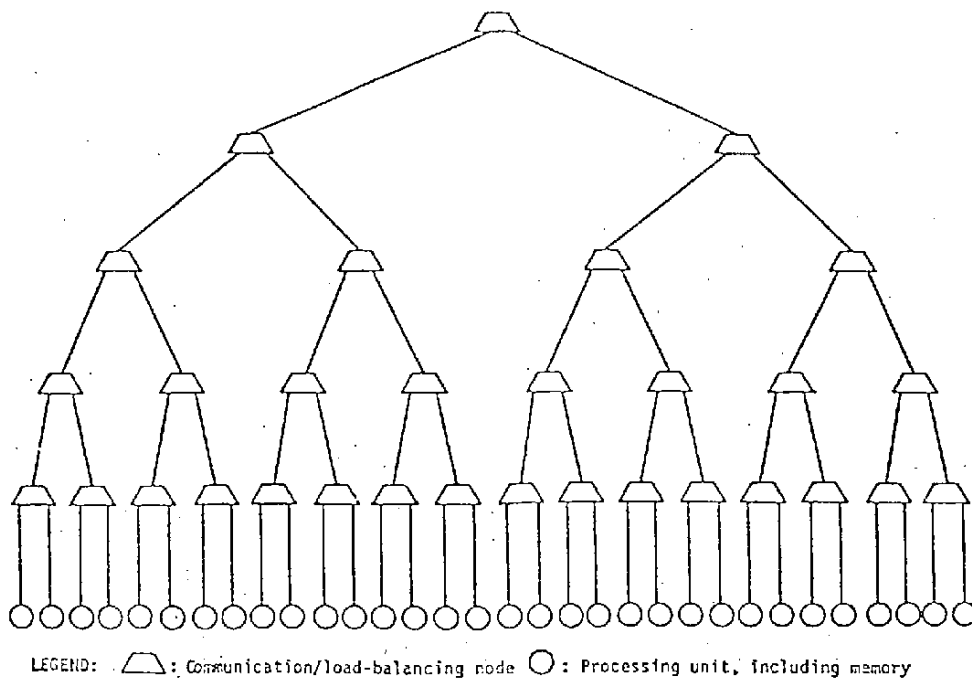


図 2.3 - 3 ユタ大学のKellerマシン

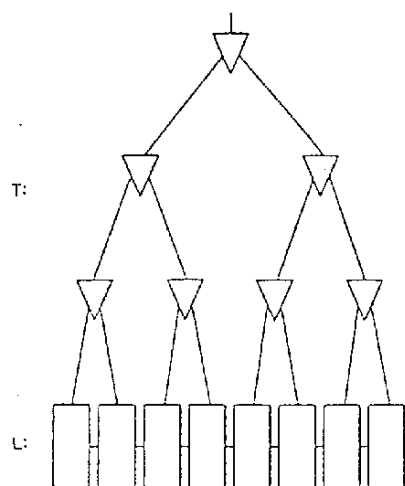


図 2.3 - 4 Mago のリダクション・マシン

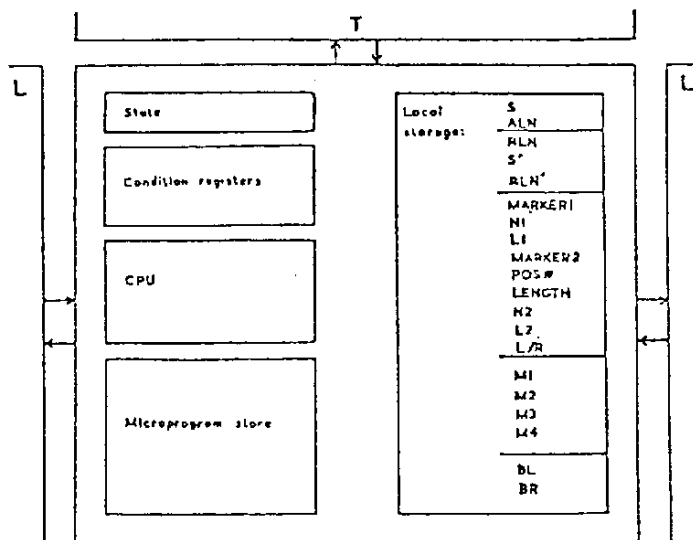


図 2.3 - 5 Mago のリダクション・マシンのセル L の構造

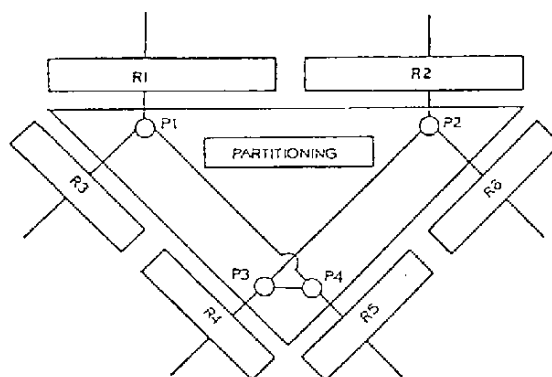


図 2.3 - 6 Mago のリダクション・マシンのセル T の構造

(4) 研究開発内容

(1) 研究開発項目

関数型マシンの開発のための、主要な研究・開発項目を次に述べる。

① 関数型言語の仕様の確立

LISP, INTERLISP等の関数型言語を参考にして、第5世代コンピュータ用の関数型言語の持つべき機能を検討し、言語仕様を定める。この際 言語の理解容易な表現(グラフ表現等)にも重点を置く。

② 並列処理記述用の関数型言語の仕様の確立

①で開発した、関数型言語を拡張して、論理的に並列処理可能な処理を、最大限並列処理することが可能なように、並列処理記述可能な関数型言語の仕様を定める。

③ ファームウェア・ベースのパーソナル関数型マシンの開発

マイクロ・プログラム方式の関数型言語のインタプリタを持つパーソナル・マシンを開発する。これは、開発の前期に主点的に開発し第5世代コンピュータの研究促進用に利用する。

④ VLSI 記号処理用UHMの開発

関数型マシン、述語論理マシン、抽象データ型マシン等のマシンに共用して利用出来る記号処理用のUHM(汎用ホストマシン)を設計し、これをVLSI化する。このVLSIプロセッサは、中期における並列プロセッサ方式による高速関数型マシンの要素プロセッサとなる。

⑤ 並列処理方式の確立

関数型言語の処理を高速化のための、並列プロセッサによる並列処理方式を確立しデータフローの処理技術と結びつける。

⑥ VLSI 並列型構造メモリの開発

一般的なパターン・マッチングの高速化の技術(多数のHornクローズを高速に処理する技術等)を確立し、VLSI 並列型構造メモリを開発する。

⑦ プロセッサ間結合方式の確立

関数型マシンと他の第5世代のコンピュータのマシンとの情報の受け渡し(処理要求と処理結果の受け渡し)を高速に行うための、プロセッサ間通信方式を確立する。

(2) 研究開発スケジュール

研究開発期間は約10年として、その期間を大きく3期に分ける。前期では、研究促進用のパーソナルな関数型マシンの研究・開発を主体とする。中期では、並列プロセッサ方式による高速関数型マシンを研究・開発する。後期では、データ・フロー方式とVLSI 並列型構造メモリを用いた超高速関数型マシンを研究・開発する。

関数型マシンの研究開発スケジュール

	前 期	中 期	後 期	速 度 ^④
パーソナル 関数型マシン	関数型言語			2~10
	関数型言語→ML			
	変換プログラム			
	ファームウェア方式 関数型マシン			
高速 関数型マシン		並列処理関数型言語		10~100
		並列処理関数型言語 →ML変換プログラム		
		並列プロセッサ方式		
		VLSI 記号処理用UHM		
		高速関数型マシン		
超高速 関数型マシン			データフロー方式	100~ 1,000
			VLSI型構造メモリ	
			超高速関数型マシン	

④ 4MIPSの汎用コンピュータでのリスト処理速度に比較しての倍数。

(3) 研究開発の目標・仕様

前期 ファームウェア方式のパーソナル関数型マシンを開発する。このマシンは、中期以降、複数台生産して、第5世代の研究促進用として、他のマシンの研究・開発や、ソフトウェアの研究・開発に利用する。

性能は、4MIPSの汎用コンピュータでのリスト処理速度の2~10倍を目標とする。

中期 並列プロセッサ方式で高速関数型マシンを開発する。

性能は、4MIPSの汎用コンピュータでのリスト処理速度の10~100倍を目標とする。

後期 データフロー技術とVLSI並列型構造メモリを持った超高速関数型マシンであり、性能は4MIPSの汎用コンピュータでのリスト処理速度の100~1000倍を目標とする。

2.3.3 関係代数マシン

(1) 概要

知識情報処理においては、いろいろな応用分野における専門的知識を、知識ベースに蓄え、これを適宜参照することで、計算機に知的な、すなわち、気の効いた処理を行わせる。

知識の内容は、各種考えられるが、単語や文法の辞書、設計・製造上の規則、作業手順のようなもので、計算機内で扱われる場合は最終的にはデータやプログラムの集合となる。

その構造は一般的に、いくつかの階層から成るが、ハードウェアに近いレベルでは、表(テーブル)のような簡単な構造のものが望しい。もちろん、表のみでは実現できず、内容呼出し記憶や、表の間を結び連想検索などを必要とするが、その基本構造は、簡単で、VLSI化などハードウェア化に適したものであることが望しい。それとともに、その構成の規則が、数学的理論に裏打ちされたものであり、検索、更新操作が関数型言語などにより記述され得るものである必要がある。

ここで述べる関係代数マシンは、関係型データモデルに基づくデータベース処理を、大規模のデータについて能率よく実行することを目的とするマシンである。関係型データモデルは、表形式の構造を基本とし、演算としては、検索、整列、結合を基礎とする関係代数演算を中心としている。さらに、この上に、データベースに対する質問応答や管理システムを実現するための関係論理に基づく言語と演算が備えられる。ユーザからこのマシンを見る場合は、関係論理に基づくデータベース向き高級言語マシンとなる。

このように整った形式的仕様を持つことから、将来のデータベースの大規模化、複雑化に対処できる新しいデータベース・マシンとして注目されているとともに、知識ベースに重要な推論のメカニズム等とも、言語面や理論面での整合性が良い。実現に際しては、関係代数マシンが、大規模システムの構築をめざし、ソフトウェアや理論面の要請が主となっていることから、ソフトウェア面での負担が、ハードウェアへと移されている。このための汎用計算機などのいわゆるノイマン型アーキテクチャでは、高性能のマシンを作り得ない。このため、新しい並列処理手法に基づく、強力なハードウェア・プロセッサの開発を必要とする。このためには、数百台以上の要素プロセッサから成り、数十ギガバイト以上の記憶容量を持つマシンをめざす開発研究が必要で、当然、VLSI技術が必須である。これと同時に、ソフトウェア面での関係論理、関係代数レベルの言語と処理系の研究開発、データベース管理用ソフトウェアの研究開発が必要である。

関係代数マシンの用途としては、その開発の中間段階で関係データベース・マシンとして実用化の可能性がある。最終的には、知識ベースをサポートする高機能マシン(モジュール)として、第5世代マシンの主要部分をなすものと考えられる。

以上のような理由から、第5世代マシン開発の重要な研究開発課題である。

〔2〕 必要性

関係代数マシンは、そのインタフェースとなる基本操作が、明確な数学的基盤に基づいて決められていることから、知識情報処理システムのような複雑なシステムの構築に適するとともに、推論などのより高レベル処理を行う述語論理マシンのデータベース・サポート機能などを受持つことができる。関係代数マシンのソフトウェア、ハードウェアを目的に応じ適合化させることで、自然言語処理、機械翻訳、コンサルテーション・システム、CADシステムなどの知識ベースの高速化、高能率化、大容量化などがはかれ、応用システムの性能が、量的にも質的にも向上する。

第5世代マシンにおいては、データベースのサポートは、きわめて重要であり、これまでの座席予約や在庫管理、文献検索などの定型的なシステムとは異なるものが要求される。

特に、知識ベースの高度な利用分野である機械翻訳や、医学や法律などの専門的知識を扱うコンサルテーション・システム、経営情報、事務手続き規則なども扱うオフィス・オートメーション・システムなどでは、単にデータを取り出したり貯えたりするのみならず、不完全な質問に対する応答、いくつもの事実を組合せ推論して新たな解答を与える能力、なぜそのような解答を与えたかが説明できる能力などが必要となる。

このような能力を実現する手法は知識ベース管理システムの研究テーマであるが、これをサポートするマシン・アーキテクチャやハードウェアとしては、きわめて高い性能が要求される。現時点で考えられるマシンの方式としては、論理的推論や演算処理の基本機能が最もよく整理されている関係型データベースとそれをサポートする関係代数マシンが有力である。また、蓄える知識の量、すなわち、マシンの記憶容量も大きくなることが予想され、汎用マシンによる実現では、要求性能の実現は困難と思われ、関係代数マシンのような機能分散の考え方をとり入れた専用ハードウェアの開発が是非とも必要である。

当然これに伴い、マシンを構成する要素プロセッサや接続メカニズムのVLSI化、大容量、高速の記憶システムを実現するための連想メモリ・チップ、高速ディスク、光ディスクのような大容量デバイスの研究開発も必要である。

〔3〕 内外技術の現状

関係代数マシンの研究は、関係データベース・マシンとして研究され、ソフトウェアによって、既存汎用機上に試作されたり、ハードウェアによる実験機が試作されたりしている。また、マシンの構成についてもいくつか提案がなされている。現状においては、関係代数マシンを集合演算を得意とする関数型マシンととらえ、そのアーキテクチャを研究する方向と、データベース・マシンとして、そのプログラミング言語、データの構成手法、データの意味論、概念変換、データのインテグリティの保障を対象とするソフトウェア面の研究の方向の2つに大別できる。

アーキテクチャ面の研究における問題は、容量や処理速度に関し、十分な能力を有するハードウェアが妥当なパフォーマンスで、実現できるかという点にあり、このためには、アルゴリズムの改良や、VLSI化などの研究と深く関係する。このため、ソフトウェアから部品面にわたる総合的な研究開発を必要とする。現時点では、ソフトウェアによるシミュレーションと、小規模な実験機の製作を併用した、アーキテクチャの模索が続けられている。

ソフトウェア面の研究は、質問応答のための言語、マシンの階層化に対応する中間レベルの言語の研究、データベース管理システムの構成の研究、マシン化する場合のアルゴリズムの研究が行われている。言語としては、ALPHA、QBE、SEQUEL (SQLともいう)などが提案され処理系も試作されている。しかし、データの意味論や、知識表現に適したデータモデル論の研究は、まだはじめられたばかりで、知識情報処理の重要な課題として、今後の発展が望まれる。

以下、主に、ハードウェアやアーキテクチャ面の研究を中心に内外の状況を述べる。

(1) 国内技術

国内では、関係代数に基づく基本演算を行うアーキテクチャの研究が近年盛んである。アーキテクチャのレベルでは、関係モデル以外のもののサポートを同時に狙ったものもある。

電総研では、バブルメモリを用いたマルチプロセッサ方式のデータベース・マシン EDC を試作した。また、ソフトウェア面では、知的データベース・アクセスの研究も行っている。通研では、最近、関係代数の基本演算を行うマシン・アーキテクチャの研究を開始している。北大では、ストリーム処理を導入した新しいマシンの構成法について研究している。このほか、関連する研究が、横浜国大、東大、慶大などで行われている。

また、東芝をはじめとするいくつかのメーカーの研究所でも、基礎的な研究が行われている。

(2) 海外技術

海外では、まず、関係モデルを提案したIBMのCoddの先駆的研究がある。IBMでは、その後、既存マシン上へ関係データベース・システム SYSTEM-Rを実装し、商用化をめざした研究が続けられている。

ハードウェアの研究としては、トロント大学のRAPシステムが有名である。これは、固定ヘッドのディスクを用い、このヘッドのおおのにマイクロ・プロセッサを付加し、連想処理を実現しようとするもので、回転記憶を用いたアーキテクチャの代表的なものである。このほか、ウィスコンシン大学では、クロスバー・スイッチによりプロセッサやディスクを結合した多重マイクロ・プロセッサ・システム DIRECTを開発している。このほか、いくつかのマシンの提案や実験機の試作の例があるが、いずれも基礎的なものである。

しかし、SYSTEM-Rにみられるように、商用化システムがソフトウェア的に整って

くるとともに、ハードウェア面も徐々に実用化をめざしたものが研究されるものと思われる。

[4] 研究開発内容

(1) 研究開発項目

① インタフェース言語と処理系の開発

マシン言語として、関係論理、関係代数、マシン基本演算の3段階のものを開発する。関係論理のレベルは、最も高いレベルとし、ソフトウェアで処理系を作る。これによりデータベースをアクセスする。関係代数は中間言語的役割を持たせ、この実行は、探索、整列、結合、制御用スカラー演算、条件判定演算などを含むマシン基本演算により行う。これら、3者間の言語変換・最適化のための処理系を開発する。

② データベース管理システムの開発

データベース・システムとして、データを構造化し、参照するのに必要な管理システム(O S)を開発する。これは、データベースのインテグリティの管理や機密保護の機能も持つ

③ マシン基本演算用アルゴリズムの研究

マシン基本演算の主なもの(探索、整列、結合、その他)について、これを効果的に実行し得るアルゴリズムについて研究する。これは、直接、マシン・アーキテクチャの決定に影響するため重要である。特に、並列処理を多用することから、並列処理向きアルゴリズムを開発する。また、V L S Iによる実装を考慮し、V L S I向きのアルゴリズムも研究する。

④ マシン・アーキテクチャの開発

並列処理を基本とし、高速化、大容量化に対応できる柔軟なアーキテクチャを開発する。また、実装に際し、V L S Iを開発する。アーキテクチャの基本方式として、マルチ・プロセッサ方式、パイプライン方式などを用い適宜組合せることとする。また、データフロー方式等の新しいアーキテクチャを取り入れる

⑤ 要素プロセッサおよび接続用ハードウェアの開発

マシン構成要素となるプロセッサ、メモリ、接続用ネットワークの構成部品などについて、個別に研究し、部品としての汎用性を高めるほか、ファームウェア技術などを活用し適応性を備えるよう工夫する。これらは、V L S I化することを目標とする。

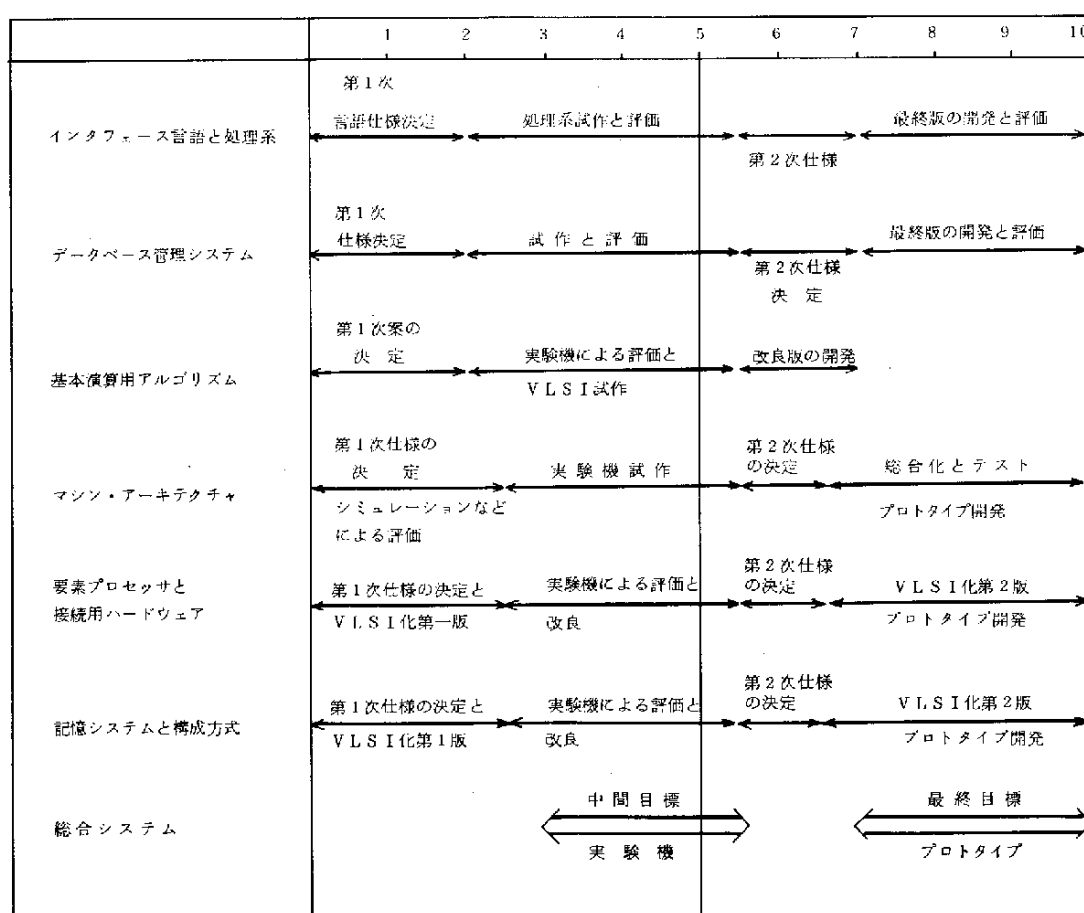
⑥ 記憶階層の構成方式と記憶デバイスの開発

マシンの大容量化を容易にし、尚かつ、高速化するために、内容呼出し記憶のV L S Iによる一部実装、階層間のデータ転送の高速・高機能化、仮想記憶化の技術開発、実装を行う。また、光デバイスなどの新デバイスの使用も考慮する。また、目的に応じ、小容量・高速型から、大容量・低速型まで、柔軟に適応できる構成方式を開発する。

(2) 研究開発スケジュール

中間段階で、実験機を試作する。この実験機上で、ハードウェアの評価と最終仕様決定のための作業を行う。また、言語や管理システムなどのソフトウェアの開発を一部行っておく。最終のプロトタイプの試作は、実験機の製作に使用したVLSI化した部品などの改良した第2段を用いて行う。プロトタイプ試作のためには、ソフトウェアも含めた評価結果もとりにこめるような体制をとる。

関係代数マシンの研究開発スケジュール



(3) 研究開発の目標・仕様

- ① 関係論理，関係代数に基くインタフェース言語と処理系を開発する。
- ② 関係型データベース・システムを管理するソフトウェアを開発するとともに，知識ベース管理システムとソフトウェア的に結合する。
- ③ 記憶容量については，小容量・高速型の構成で，10～100Mバイト，中容量・高～中速型の構成で，100～10Gバイト，大容量・中～低速型の構成で，10～1000Gバイトを目標とする。大容量型は，画像データベースも扱えるものとする。アクセス時

間は、述語論理型マシンの大規模なもののデータ格納用として十分な速度を目標とする。

④ 並列処理プロセッサにより実現する場合は、要素プロセッサ数、数百台以上を目指す。(中間目標、100台以下)

⑤ 使用する要素プロセッサ、接続用スイッチ、内容呼出し記憶などは、できる限りVLSI化することとする。

2.3.4 抽象データ型マシン

(1) 概要

情報処理が対象とする情報に注目すると、それぞれ独特のデータ構造と属性をもっていることがわかる。例えば、数値処理では、整数、実数、複素数等が基本のデータ型で、データ構造としては、ベクトル(1次元配列)や行列(2次元配列)等をとる。リスト処理では、アトムとよばれる基本要素が木構造に配置されている。このように情報に結びついた固有性は、その情報を処理する場合に非常に有効で、処理をわかり易くしている。

情報処理の発展と共に情報の特性に対する理解が深まって来た。具体的な対象に対する情報処理知識が増大する一方で、それら具象を包含して統一的観点から把えてみようとする動きもでてきた。抽象データ型の概念はその1つとして発生してきたもので、最近ソフトウェア工学における階層型設計手法やプログラム正当性の検証等に使われて、概念の有効性を示している。

狭義の抽象データ型は、データに対する抽象化で、具体的にはスタック、キュー、リスト、行列等の構造のわくを定義しておき、実体はその定義された抽象データにパラメータを渡して生成し、それを対象(Object)と呼び、同一の抽象データに対し、多数の具象データを対応させる。さらに抽象データ型は手続きの定義を許し、データへの作用も記述できる。上例ではPUSH, ENQ, CAR, MULT等の手続きが作用する。

広義の抽象データ型は、上述で定義する対象をデータに限定しないで、関数(あるいはプロセス)や制御構造にまで拡張して広汎なシステム記述能力を実現しようとするものである。この定義によるシステムの記述方法は、知識の集積に似ており、表現手段が豊富になることにより、知識の拡大につながるものである。

抽象データ型マシンは、抽象データ型言語を効率よく実行するものである。抽象データ型の有効性として、モジュール化プログラム、対象を識別する能力(Capability)からくる高信頼化プログラム等があるが、これらは、仮想的なモデルとしてみた場合であり、実際に従来の計算機で実行すると、オーバーヘッドが多くなり、実用的なシステムにはなり難い点がある。

計算機アーキテクチャの進歩およびVLSI技術の進歩は、仮想マシンとしての抽象データ型マシンを実マシン上にマッピングすることを現実的なものとしつつあり、今後の情報処理

技術の飛躍的發展の1つの方向を示すものである。

〔2〕 必要性

情報処理の対象は、初期の数値処理、事務処理から、図形・画像処理、大規模データベース・システム、信号処理、記号処理、自然言語処理等と広がっている。現在のシステムはこれら多様な分野を複雑に組合せたものとなり、この傾向はシステムの大型化と共に増大している。

情報処理システムでは、上記分野に対応してそれぞれプログラム言語を設定し、設計の容易化を図っている。分野が異なればおのずとデータ型が異なるはずで、情報の範囲が広がるにつれ、それをあつかうデータ型もふまえている。図形・画像といった情報は以前はコンピュータの処理外であったが今日では身近なものとなりつつある。さらに今後知識情報がコンピュータ処理に加わろうとしているが、その情報量は厩火である。

情報処理分野の拡大、言語の多様化、データの質的、量的拡大は、すでにソフトウェア危機を招いており、ソフトウェア技術者は、その1つの解決策として抽象データ型言語の概念を考案したがそれを現在のコンピュータでインプリメントすることは、抽象データ言語のもつ柔軟性、拡張性、ダイナミック性等のよさを制限することであり、新しいアーキテクチャによる抽象データ型言語の高速処理が必要である。

〔3〕 内外技術の現状

（1） 国内技術

抽象データ型をプログラム理論の代数的アプローチや、ソフトウェア工学における階層設計方式等に適用する研究が、大学、研究所で行われている。（表2.3-1参照）また一部メーカーやソフトウェアハウスでも、実験的にシステム設計に適用した報告がある。

これらの試みは限定した範囲にとどまっており、未だプログラム言語としては普及していない。最近米国で発表されたADA言語にこの概念が一部とり入れられており、この言語の紹介の中で抽象データ型に対する議論が活発に展開されるようになり、情報処理技術者の間にこの考え方が普及するものと思われる。

抽象データ型言語を実行する計算機アーキテクチャに関しては、ほとんど研究されておらず、言語の確立が先決の状態である。ただこの研究の重要性は認識されつつあり、今後多くの研究発表でとりあげられるであろうし、メーカーも抽象データ型アーキテクチャを有する計算機への対応策をせまられて、研究を開始するものと思われる。

（2） 海外技術

抽象データ型言語はすでに60年代に出現している。その後、表2.3-2に示すように米国においていくつかの言語が設計され、実際のシステムに適用されているものがある。長近話題になっているADAは、抽象データ型の概念をとり入れており、商業的にも使われるのでその動向が注目される。

表 2.3 - 1 国内における抽象データ型言語の試み

名 称	開発元	用 途
H I S P	電総研	階層的プログラミング
Meta-88	電総研	問題解決向き抽象アルゴリズム
CLU処理系	東工大	MITのCLUインプリメント / FACOM230-45S
イオタ言語	京 大	プログラム検証システム
ROMULUS	慶 大	抽象データ型をライブラリ化 (REMUS)
LEX	九 大	問題向き言語の作成

抽象データ型をコンピュータ・アーキテクチャに導入しようとする動きが、米国にみられる。(表 2.3 - 3 参照) すでに S / 3 8 は、対象を指向したアドレッシングをもつ商用機であり、インテルも、ADAをサポートするマイクロコンピュータを発表している。また MIT やテキサス大学 (図 2.3 - 7 参照) では、対象指向を実現するアーキテクチャを提案している。

表 2.3 - 2 海外における抽象データ型言語

名 称	開 発 元	用 途
S I M U L A 6 7	ノルウェー計算センタ	シミュレーション用
S M A L L T A L K	ゼ ロ ッ ク ス	教育用
M E S A	ゼ ロ ッ ク ス	システム記述用
A L P H A R D	カーネギーメロン大	プログラム検証用
C L U	M I T	抽象データ型一般言語
A D A	米 国 防 省	エンベッディッド・コンピュータ用

表 2.3 - 3 対象指向型計算機

名 称	開 発 元	特 徴
S / 3 8	I B M	1 レベルストア, ケイパビリティ・アドレッシング
I A P X 4 3 2	インテル	A D A をサポートする予定
C L U マシン	M I T	C L U を実行するアーキテクチャ
Object Oriented Computer	テキサス大	オブジェクト・プロセッサ, D B を用いるアーキテクチャ

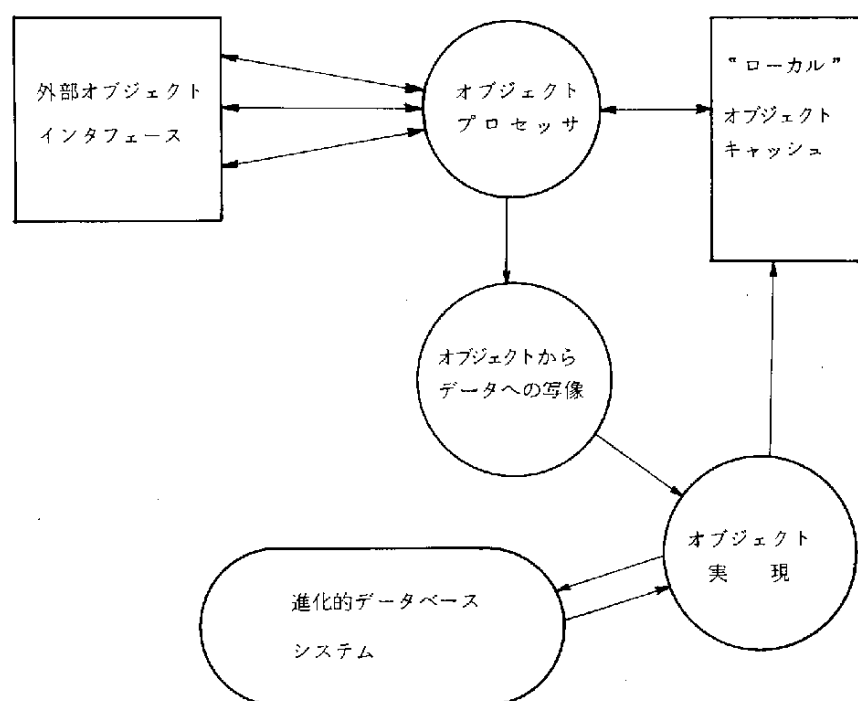


図 2.3 - 7 テキサス大学の対象指向アーキテクチャの概念図

(4) 研究開発内容

(1) 研究開発項目

① 抽象データ型モデルの整理体系化と言語の開発

抽象データ型を幅広い応用分野に適用するには、それぞれの分野の対象 (Object) の型や属性が正確に表現できること、個々の対象がユニークに識別できる capability アドレッシングをもつことが必要で、これらをストアする記憶空間 (対象空間) は、仮想空間や 1 レベルストアの拡張であり、情報の多様性からみてより我々人間の空間概念に近づくものであろう。このモデルは、形状概念だけでなく、時間概念や意味概念を含んでおり知

識情報処理のデータベース表現法とも関係する。

抽象データ型の対象をデータだけでなく、手続きや関数に拡張することは、システムの柔軟性に必要である。それにはマイクロ・プログラムによる新命令の記述能力、ハードウェア記述言語による系の再構成能力、関数型言語で使われる関数を対象とする関数形式の能力が抽象データ型言語に要請され、それを対象空間に導入する必要がある。

上記データや関数を包含した対象空間としてACTORモデルがある。ACTORとは“手続きや関数などのように計算や操作を行うもの、記憶場所、データ構造などのように情報を格納するためのもの、数や文字列などのようにデータそれ自身に対応するものの4つの異なる範疇に属するものをメッセージを受けることによって能動化される手続的な対象として統一したものである。”これは対象だけでなく対象間の通信を重視するものであり情報ネットワーク、分散処理の研究につながる。

対象空間の研究は、抽象データ型の計算機処理方式の基礎となるものだが、この研究は計算機自身もモデル化することになり、計算機アーキテクチャや人工知能の研究とも深くかかわって来る。

抽象データ型言語は上記対象空間を記述する言語であり、対象の範囲が広がっても使えるためには、問題向き言語のレベルよりも高位のものでメタ言語的性格を持つ。マシン・レベルの記述能力としては、高水準なマイクロプログラムやハードウェア記述言語まで可能なものが望ましい。

② 物理リソースへの写像方式の研究

抽象データ型で定義された対象は、通常の言語のようにコンパイルしてすぐ物理イメージができるものではなく、単に対象の原型が構成されるにすぎない。実際に起動がかかったときに、適当な物理リソースと結合するのが特徴である。従って、物理リソースと対象空間を関係づける写像テーブル（仮想空間の場合はページ・テーブル）が必要である。この構造は、Capabilityチェックを行う部分と、物理リソースをアサインする部分からなるが、この部分のオーバーヘッドはシステム性能を大きく左右するので、高速な連想処理技術の研究が必要となる。

③ ガーベッジ・コレクタの開発

抽象データ型の1つの特徴は、対象が生成/消滅を繰返すことであり、これを放置しておくと、すぐにメモリ・リソースが不足し、システムが動作しなくなる。ガーベッジ・コレクションは不用なメモリセルを回収する機能を有するもので、システム性能を上げるに

は、マシンの実行と並列して行う並列ガーベッジ・コレクションを、専用ハードウェア化する研究が必要である。

(4) 構造メモリの開発

抽象データ型のデータ構造は、単純変数や配列だけでなく、複雑な構造をもっており、1次元アドレッシングでは、処理能力が低下する。構造体メモリは、そのようなデータを自然に写像できる記憶装置で、必要な機能として、

- a. 高速大容量化のために多重アクセス可能で階層構成のメモリ装置
- b. 並列ガーベッジ・コレクションのサポート機構
- c. 対象を識別し、保護するためのタグ付きデータ
- d. メモリを効率よく利用するための、スタックやヒープ機構

等がある。

(5) 抽象データ型プロセッサ・アーキテクチャの研究

抽象データ型で使用するプロセッサは、データの多様性のため1種類ですますことは不可能である。例えば数値処理、信号処理、言語処理、画像処理を複合したシステムの場合それぞれ特有のプロセッサを配置する必要がある。しかし、それらを同一のエLEMENT・プロセッサで構成する方法は可能であり、抽象データ型がもつ柔軟性、拡張性にも合致している。またリピータブルな点で、VLSI向きともいえる。仮にこのELEMENTをUHM(Universal Host Machine)とすると次のような研究があげられる。

- a. ダイナミック・マイクロプログラミングが可能なもの
- b. プログラマブルにレジスタやバス幅を拡大/縮小できるもの
- c. 系の多重化、再構成が容易なもの
- d. 抽象データ型言語を指向したもの
- e. タスク切換え、プロテクション、例外処理等のOS機能を有するもの
- f. 分散処理に適したもの

それ以外に信頼性、高速性の追求が必要である。

(6) 並列処理方式の研究

抽象データ型言語を並列処理環境下で動作させることは、ACTORモデルの考え方の実現につながる大きな課題であり、性能の飛躍的向上が期待できる。研究としては、

- a. メッセージ授受のプロトコール
- b. メッセージ・スイッチの方式

- c. 対象空間から並列処理装置への写像
- d. データの完全性，機密性

等がある。

⑦ その他の項目

a. 入出力系の方式

抽象データ型における入出力は，ワンレベル・ストアと同様に，特定対象空間（入出力ポート）への Read/Write とみなせば，特別視する必要はない。物理的には，メモリ・マップド入出力技術によって行う方法が考えられる。

b. 従来言語との関係

抽象データ型言語は，対象空間での相互干渉を禁止する作用を有するので，従来言語で記述されたソフトウェア資産を閉じた領域に隔離する（Encapsulation）方法で，それらを有効利用することを検討する必要がある。

c. OSの研究

抽象データ型言語はシステム記述言語の能力も有しており，OSのモジュール化に適している。抽象データ型のシステムで重要なOS機能として，物理リソースの管理および障害回復技術等がある。またモジュール化による，多重OSを研究する必要があるが，これらはすべて対象空間のモデルに整合していなければならない。カーネルOSは軽くしてUHMにとり入れることを考え，大半のOS機能は，応用システム毎に最適なモジュールを組合せて構成するのがのぞましい。（図2.3-8参照）

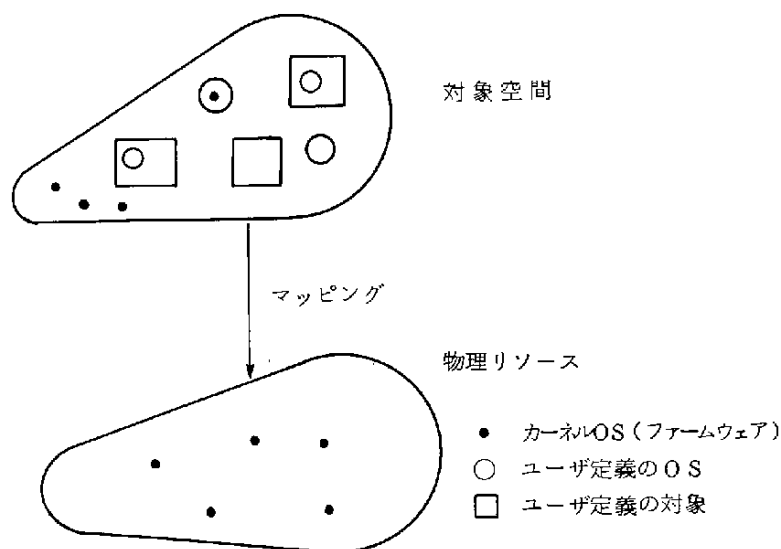


図2.3-8. 抽象データ型におけるOS

d. データベースの研究

抽象データ型でデータベースをとり扱えれば、データベースの管理を非常に容易にしてくれるが、それが対象空間にうまく埋込むことができるか、ローカリティを利用して物理リソースに効率よくマッピングできるか等を解決することが課題である。

大規模システムに抽象データ型マシンを適用するには、抽象データ型の膨大な定義部分をデータベース化し、高速にアクセスする必要がある、データベース・マシン技術の開発が必要となる。

e. 分散処理の研究

対象空間を使って分散処理をモデル化し、抽象データ型マシンのコンピュータ・ネットワーク技術を開発する。またデータベース技術とも組合せた分散データベース技術も開発する。

(2) 研究開発スケジュール

抽象データ型マシンの実現形態としては、ファームウェア・ベースの単一計算機で実行するものから、専用化したマシンによる並列処理に至るまで順々に開発して行くのがモデルの難易度からも妥当である。次表に開発スケジュールを示す。各段階ともアーキテクチャに先行して対象空間のモデルを検討する。抽象データ型言語とコンパイラ的设计もこれに歩調を合せる必要がある。

前期はファームウェアによる抽象データ型マシンで、基本的なモデルを設定する。実験用 U H M の設計が中心で、構造メモリ、ガーベッジ・コレクション、capability アドレッシングのハードウェア / ファームウェア・サポートを行う。

中期は、プロセッサを複数台にし、構造メモリの多重化を図る。モデルとしては、データベースを組込んだものを検討する。

また中期の後半では、本番用の抽象データ型マシンを複数台使ったシステムの開発で、U H M の再検討を行う。このモデルは複数のデータ、プロセスが、非同期に通信するため、A C T O R モデルを開発する。物理上は、ネットワークの方式を開発しなければならない。

後期は、実用的な抽象データ型マシンとして、シミュレーション・マシンとかデータベース・マシンを実験するのと並行して、分散処理をモデル化した分散処理用抽象データ型マシンを開発する。

抽象データ型マシンの研究開発スケジュール

開発項目	1	2	3	4	5	6	7	8	9	10
抽象データ型モデルと言語の開発	基本モデル	DHモデル	ACTORモデル	分散モデル						
	抽象データ型言語 V_1	V_2	V_3							
システム構成	ファームウェアベースの単一プロセッサ	実験機によるフルチッププロセッサ	本番用マルチプロセッサ	高性能抽象データ型マシン						
プロセッサ技術		(前期) 実験用 UHM	多重化	(中期) 大規模用 UHM	UHMのVLSI化					
要素技術	タグ、ダイナミック・マイクロプログラム	1レベルストア	DB化							
	タイパビリティ	マッピング・ハードウェア	連想メモリ							
		構造メモリ、GC機構	並列構造メモリ、専用多重GC装置							
基本ソフトウェア		カーネルOS	多重OS	非ノイマンOS	分散OS					
実証実験システム				DBマシン	シミュレーション・マシン	知識ベースマシン				

(3) 研究開発の目標・仕様

- ① 抽象データ型実験機の製作
- ② 100台程度の実験機による抽象データ型マシンの製作
- ③ 1000台程度の本番用抽象データ型マシンの製作
- ④ 分散処理用抽象データ型マシンの実験
- ⑤ 従来言語によるソフトウェアの継承
- ⑥ 多くの応用分野に適用して、その有効性を確認する（主要な分野として、データベース、シミュレーション、知識ベースへの適用を試みる）。
- ⑦ ACTOR理論や関数型言語との融合を図る
- ⑧ VLSI化した高性能なエレメント・プロセッサを多数結合して、性能価格比の向上を図る

2.3.5 データフロー・マシン

(1) 概要

従来の計算機はその基本原理を蓄積プログラムの逐次処理に置いているが、計算機利用技術の発達、応用分野の拡大にともない、幾つかの面に於いて問題点が顕著になって来た。データフロー・マシンはその基本原理を問い直そうとするものである。データフロー・マシンではプログラムの実行制御は各演算に必要なデータの準備状況によってそれぞれ個々独立に行なわれる

為、多くの演算が非同期に次々と併行して処理される。従って並列処理が自然と実行される点に特徴がある。

データフロー・マシンの研究は未だ歴史が浅く実現するに当って解決してゆかねばならないことは数多いが、アルゴリズムに内在する並列性を取出し易い、データフロー・マシン用言語はプログラムの正しさを検証するのに容易である等、非常に優れた性質を備えており、将来の計算機構造に影響を与える有力な方式である。

〔2〕 必要性

従来の計算機は次々と命令を直列に実行してゆく方式を基本としている為、元々並列に実行し得る処理でも直列に実行される。従って処理能力は1つの命令を実行する時間によって定められ、素子の速度が抑えられるとそれで能力に限界が来てしまう。従来迄にパイプライン方式等によって、処理単位を出来る限り小さくすることに注がれある程度の効果をあげて来た。しかし、世の中に於いては計算機の利用が多くなるにつれて増々高い計算機能力が求められており、その要求には際限が無い。現在の計算機構造は既にその素子限界に近ずきつつある。この限界を突破するには新しい原理に基づく素子の開発と並列処理が有力な方策でありそれぞれに努力が傾けられて来た。

プロセッサを何台も並列に並べる形の処理の研究は古く、多くの提案が為されて来たが十分実用性のあるものは出て来ていない。従来の計算機の基本構造は元々並列処理に向いていない為、解くべき問題に内在する並列性の抽出手続きが言語やプログラマに押し付けられて来た。データフロー・マシンはその基本構造が並列処理マシンであり、プログラムに内在する並列性を容易に取出すことが出来る。しかもそれがプログラマに対し殆んど負担を与えない。従ってデータフロー・マシンは並列処理の基礎を与えるものとして大変重要な技術である。

また、今後の計算機システムを構成する素子としてはVLSIが不可欠となろうが、データフロー・マシンのようにその構造に繰返しパターンが多いものは製作が容易であり適していると言えよう。

一方、現在の情報処理システムに於ける大きな問題はソフトウェアの生産性・蓄積性の向上である。これは今後情報処理が更に普及し発展してゆくに従い増々大きな問題となろう。これを解決してゆく為には、作成したプログラムの正しさを自動的に検証するシステムやプログラムの自動合成システム等の援助が必要である。所が、従来よく用いられ来た言語はこの検証に向いておらず非常な手間のかかることが知られている。一方、関数型言語と呼ばれるものは一般に副作用が少なく、検証が容易であるが、データフロー・マシンにはこの言語が非常によく適合する。従ってデータフロー言語を用いることはプログラムの生産性向上に役立つであろう。

プログラムの蓄積性に関しては論理式を次々と独立に並べたような形の論理型プログラミングが有望であるが、その処理は式の数が増すと一般に大変時間のかかるものとなる。その解決

には並列処理が有望であり、関数型言語と並列処理に向けたデータフロー・マシンは有力な武器となるであろう。

③ 内外技術の現状

(I) 国内技術

国内に於けるデータフロー・マシンの研究はまだ日が浅い。1977年に東大より発表されたSAMD計算機(単一アルゴリズム・マルチ・データ・ストリーム)はデータフローと同種の考え方に基づいていたが、データフロー・マシンとして研究が活発になり始めたのはここ1~2年である。東大のTOPSTAR, D³C, 沖電気のD³P, 武蔵野通研のデータ駆動型汎用マシン及び科学技術計算用マシン, 電総研の数値計算用マシン, 東北大の関数型データフロー・システム等であるが, この内実際に製作され動いているのは東大の2マシンだけである。他のマシンはいずれもこの1年位の間に発表され, 現在設計や試作がおこなわれている。

TOPSTARは1号機が1978年に, 2号機が1980年に完成しているが, いずれも汎用マイクロ・プロセッサを用いたもので, 手続き相互の制御をデータ駆動又は要求駆動により行なっている。2号機は24台のプロセッサを用い, C言語とデータフロー記述言語を組合せて使用しており, Lispものせられ使われている。D³Cは, 同じく汎用のマイクロプロセッサを使用したものであるが, データフロー・マシンのシステム冗長性に着目し, 適当な機能を付加することにより自律的故障診断機能を持たせた所が特徴であり, 1979年に完成している。

D³Pは階層構成のマルチ・プロセッサという形を取り, タスクと機械語の2レベルの並列処理を行うのが特徴で, 1981年には試作が完了することになっている。武蔵野通研の汎用システムはプロセッシング・モジュール多数を結合網で接続した構成であり, 各モジュールに1つの関数の実行を割付けている。科学技術計算用の方は偏微分方程式等を解く為のもので, 多くのプロセッサを2次元に配置した形を取る。現在16台のマイクロプロセッサを用いて試作中である。

電総研のマシンは, プログラム・メモリを内蔵した処理モジュールを複数バス接続した構成で, データフロー用言語としてはIdをベースにしたMIDを考えている。同じく電総研では記号処理用データフロー・マシンの為の言語としてLispの拡張も検討されている。

東北大の関数型システムは, 言語の関数としての性質を徹底させることを考えており, 試作がおこなわれている。

その後, Combinatorのリダクション・システムをデータフロー的に実現する機構の提案(東工大)や, トークンのプロセッサ割当てをハードウェアにより高速に実行するシステムの試作(阪大)等が行なわれている。

(2) 海外技術

データ駆動の理論的基礎は古く1966年頃に出された並列計算のモデルにその芽を見ることが、実際のマシン・アーキテクチャへの拡張を試みたのはMITのJ.B.Dennisである(1974)。それ以後、米、英、仏に於いて幾つかの研究が見られる。データフロー・マシンの研究は言語の研究が先行しており、ハードウェアの研究自体は遅れている。機械語に相当するものとしては次のようなものがある。

① DDF: MIT, 図式表現, 最初のデータフロー語(1974)

② DDN: Utah大学, 7種類のセルによる図式的表現(1975)

また、コンパイルすることによって上記データフロー図式に相当する機械語に変換される幾つかの高級言語も提案されている。

① TDFL: MIT, Algol60にその基礎を置く

② LAU: ONERA-CERT(仏), 単一代入言語

③ Id: UCI, 経緯依存性の取込み, 非決定性表現の実現

④ VAL: MIT, 厳密なデータ型チェック

⑤ CAJOLE: London大学, 名前の局所化機械

⑥ FGL: Utah大, フローグラフLisp

マシンが実際に製作されたものとしては、仏政府研究機関ONERA-CERTのLAU, Utah大学のDDM1がある。LAUマルチ・プロセッサは、基本命令レベルの漸新的な構造を持つシステムでプロトタイプ版0が1980年に完成し、言語LAUが動いている。DDM1はバロース社に於いて1976年初期のものが作られ、現在Utah大学に於いて改良された形のものが動いているが、DDNを直接実行するマシンで再帰構造を持つトリー型のシステムである。

この他に、英国Manchester大学に於いても単一リング構造のデータフロー・マシンが試作されており、1981年中に完成の予定である。MITに於いても、基本命令レベルのマシンを試作中であり、命令を貯えるセル・ブロックとプロセッサ群を結合網で接続してループを形成したシステムで、小規模なものを現在作っている。またカリフォルニア大学Irvine分校でも双方向リングバスを用いたシステムを試作中である。Utah大学では、DDM1の他に、FGL言語を用いるAMPSというシステムが提案されている。トリー構造の粗結合マルチ・プロセッサであるが、参照の局所性を利用して通信のオーバーヘッドを減らしVLSIに向けた構成としているのが特徴である。

(4) 研究開発内容

(1) 研究開発項目

データフロー・マシンの研究開発項目としては、次のような項目があげられる。

① 機械語セットの設計

データ駆動によって制御するアクティビティの単位をどのようなレベルの処理とするべきか、又その時用意すべき基本操作には何が必要かの検討で、そのレベルと達成し得る並列度との関係、マシン・オーバーヘッドの程度等を考慮して定める必要がある。

② データフロー向き高級言語の設計

現在迄に幾つかの言語が提案されているが実際のデータフロー・マシン上にのせられたものはL A U以外にない。その他のものはシミュレータの上で動かしたもの、又は単なる理論上の提案である。関数型言語はデータフロー・マシンに適しているとは言え、現実のあらゆる問題の記述に適している訳ではない。過去の履歴に依存する問題の記述は、データベースの扱いに必要であるし、資源管理プログラムを記述する為には、どこからその要求が来るか前以って定められない状況を記述（非決定性）出来る必要がある。また、大規模なプログラムを記述し分割コンパイルを利用する為には用いた変数のネームのスコープを明確に定める必要がある。これらの問題に対して幾つかの解が考えられているが、未だその評価は全く為されていない現状である。プログラム検証手法との関連を考慮しつつ十分な検討が必要であろう。

③ マシンの全体構造の決定

データフロー・マシンの全体構造としては、アクティビティ・メモリと処理素子及び結合網でもってループ構成を取るもの、プロセッサとメモリを合わせたものを単位としてそれらをトリー状や網状、又はバスによって結合したもの等多くの形が提案されているが、未だ試作機の数が少ないこともあって、その望ましい形が明確になっていない。発火可能アクティビティの検出ハードウェア・サポート、入出力装置の位置付け、中でも特に2次記憶装置の結合法等は殆んど研究が為されていない状況である。

④ 結合網の構成

データフロー・マシンは多くの処理素子、メモリ素子が並列に動く所に特徴があるが、それをハードウェア上で可能にする主要な要素は結合網の構成である。多くのトークン（データを含む）を同時に適切なアクティビティに渡す部分がそれで、システムのスループットを定める重要部分である。多対多の間で同時に多くの通信が可能な結合網が望まれる。

⑤ 構造メモリの実現

複雑なデータ構造を能率良くプログラムに提供する為には、データ構造の実現をハードウェアによってサポートし、プログラムとデータ構造の物理構造とを分離することが望ましい。データフロー・マシンではメモリ域を意識しないので変数名が多くなる傾向がある為、構造メモリの実現に当ってはメモリ領域の有効利用を図ることが特に大切であり、各

データに対しては参照回数というタグを付けたり、不要データを消して空き領域を確保する為のごみ集めをハードウェアで支援する等の工夫が必要である。

⑥ アクティビティ制御方式の確立

データフロー・マシンでは一般に、必要なデータがそろえば直ぐに実行を始めることが出来るはずであるが、実際上は他に条件がある。即ち、処理結果を渡す相手先のバッファが空いていない限り処理を完結出来ない。このバッファは一般に複数ヶ置けるが、その空き状況管理をアクティビティの制御と対応させないとデッドロックを生じてしまう。処理の並列度を抑えずに出来る限り有効な制御方式が必要である。更に、処理の高速化を図る為に、発見的な手法では多くの代案を並列に評価するという手法が取られることがあるが、このような場合、不要となった代案を如何に能率良く止めるかということが問題となる。従って一般に、例外が生じた時にその旨を他の関連するアクティビティへ如何に迅速に伝えるか、不要となったトークンを如何に吸収するかが問題であり、能率良い手法の開発が望まれる。

⑦ OS機能の実現方式

従来の計算機システムでは多くの利用者に能率良く使わせる為にジョブ管理、資源管理等の管理機能がOSとして実現されている。データフロー・マシンに於いてこれらの機能を如何に実現するかを検討が必要である。ジョブの優先度、スケジュール等を如何に実現するか、集中した資源管理を如何に行なうか、これらのことは分散処理システムでは一般にやっかいな問題であり、データフロー・マシンに於いてはハードウェア構造にも関係する可能性がある。

⑧ 障害対策・保護対策

計算機のハードウェア又はソフトウェアによる障害対策はシステムを使用する上で重要な問題である。一般にデータフロー・マシンに置いては各アクティビティ毎の独立性が保証されている為、対策は容易になると思われるが、関連するアクティビティをいっせいに止める等難しい問題も存在する。安全性対策も同様で、これらの事柄については全く手が付けられていない現状である。

⑨ 従来計算機との融合構造

データフロー計算機はその構造が従来のマシンと大幅に異なっている為、その実現に当ってはソフトウェアの継続性から従来マシンとの融合構造も検討するべきであろう。従来の計算機のプロセッサ内の1つの特殊演算装置として付加する方法、個々の処理自体は従来の構造をしたマシン上で行ない、それら相互の制御のみをデータ駆動で行なう等幾つかの案が考えられる。データフロー・マシンの今後の研究状況によっては、むしろ、データフローの概念をそのような形で取り込む方がより容易であり実用的であるかも知れない。

い。

⑩ データフロー・マシン・プロトタイプの開発

データフロー・マシンの能力を正しく評価する為には、実用的規模のマシンを実際に設計試作することが不可欠である。そうしてこそ真の能力、発展性、ボトルネックが明確になる。勿論これにはプログラミング言語やOSの開発も含めて行なう必要がある。実際上はプロトタイプの開発を3段階に分けて行ない、始めには基本操作レベルの実験機を試作して機械語相当のセットを定め、基本的な制御方式を確立する。次に実用的規模のマシンを開発してデータフロー・マシンの特性評価をあらゆる面から実施する。最後に、大規模マシンを試作しその能力限界と可能性を調べるというステップが考えられる。

⑪ パーソナル・データ・フロー・マシンの開発

上記第2, 3ステップと並行して、パーソナル・コンピュータをデータフロー・マシンによって実現することの研究も行なわれるべきであろう。将来のマン・マシン・インタフェースは知識情報を利用した高度なものとなることが考えられるが、その鍵を握るのは論理型プログラミング等を能率良く高速に実行するパーソナル・コンピュータである。単一ユーザ用のOSとして単純な制御プログラムの制御の下、その特定個人用の知識を活用して、複雑な上位のシステムを単純で理解し易い形に仮想化する機能が必要で、論理型プログラミングに適合したデータフロー・マシンの活躍分野となる可能性がある。

⑫ データベース管理機能との融合

今後の計算機はデータベース管理機能が不可欠であるが、データフロー・マシンに於いてそれとどのような形でインタフェースを取るか、又逆にその機能自体の実現にデータフローの考え方がどのように役立つのかという問題は重要なテーマである。一部提案も為されているが、本格的な検討は今後の課題である。

(2) 研究開発スケジュール

研究開発を3段階に分けてほぼ次のような形で実施する。

前期：

データフロー用高級言語の諸検討からデータフロー・マシンの機械語レベルの基本操作セットを決定する。アクティビティ制御方式を確立し、プロセッサ内部構造を定め、プロセッサ間相互結合網や構造メモリの検討を進めてシステムの全体構造を明らかにして、基本操作レベルのデータフロー実験機を試作する。

中期：

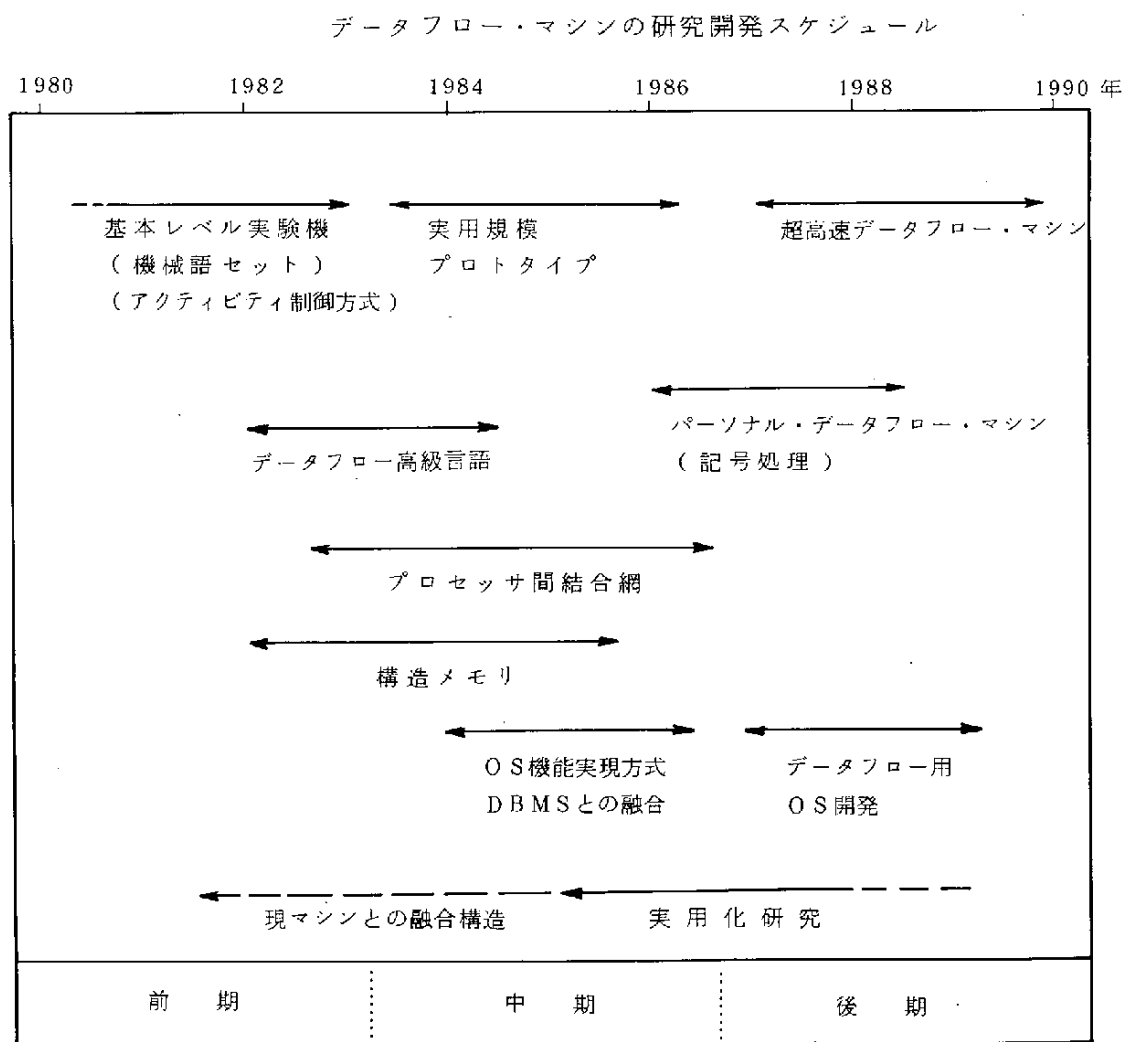
プロセッサ間結合網と構造メモリの開発をおこない、実用規模のデータフロー・マシン・プロトタイプを試作する。これにはVLSIを用いる。幾つかのデータフロー用高級言語を開発してプロトタイプ・マシン上に実装し、様々な応用に適用して汎用化の評価を行

なう。更に、OS諸機能の方式を確立し、データベース管理機能との融合を実現する。

後期：

データフロー・マシン用OSを開発してプロトタイプ・マシン上に実装する。大規模な処理（知識情報処理）を主目的とした超高速データフロー・マシンを開発する。これには超高速素子を用い、VLSIによって実装する。更に、記号処理等の非数値処理を主目的とするパーソナル・データフロー・マシンを開発し知識情報処理のフロント・エンドにする。同時に、多くの実用化の為の研究を進めるが、従来のマシンとの融合化等の試みは前期の頃から連続して進める。

以上のようなスケジュールを年代との関係で示せば、ほぼ次表のようになるう。



(3) 研究開発の目標・仕様

基本操作レベル実験機：

プロセッサ 16台

メモリ容量 16 MB

実用規模プロトタイプ：

プロセッサ 100 台

メモリ容量 200 MB

性能 100 MOPS (Millions Operations Per Second)

VLSI 構造 (初期)

超高速データフロー・マシン：

プロセッサ $10^3 \sim 10^4$ 台

メモリ容量 1GB ~ 10GB

性能 1BOPS ~ 10BOPS (Billions OPS)

VLSI 構造, 超高速素子

パーソナル・データフロー・マシン

プロセッサ 32 台

メモリ容量 10 MB

性能 10 MOPS

VLSI 構造

プロセッサ間結合網：

LSI 向き構造 (TTL)

$10^3 \sim 10^4$ 台収容

構造メモリ：

容 量 1MB ~ 10GB

2.3.6 新ノイマン・マシン

(1) 概要

(1) 第5世代プロジェクトでの位置付け

第5世代コンピュータ・プロジェクトにおける新(New)ノイマン・マシン開発計画は、大別して二つに分けられる。一つは100万トランジスタ/チップ(1チップで100万トランジスタを含むデバイス)のVLSIベース・コンピュータの開発であり、もう一つは1000万トランジスタ/チップのコンピュータの開発である。第5世代コンピュータ・プロジェクトにおける位置付けを示すならば、100万トランジスタ/チップ版(以下、MARK Iという。)は第5世代コンピュータ開発サポート・システムの中心的コンピュータ並びに第5世代コンピュータへのつなぎの役割を果すブリッジ・コンピュータとしてとらえることができ、また1000万トランジスタ/チップ版(以下、MARK IIという。)は第5世代コン

コンピュータの上に作られた知的CADシステムを使い作られる応用目的指向適応型コンピュータと考えられる。第5世代コンピュータ完成の主要な応用の一つに知的CADがあるが、これを使い1000万トランジスタ版をブートストラップし、第5世代コンピュータそのものも1000万トランジスタ/チップVLSIを使ったものにしようとするわけである。(図2.3-9参照)

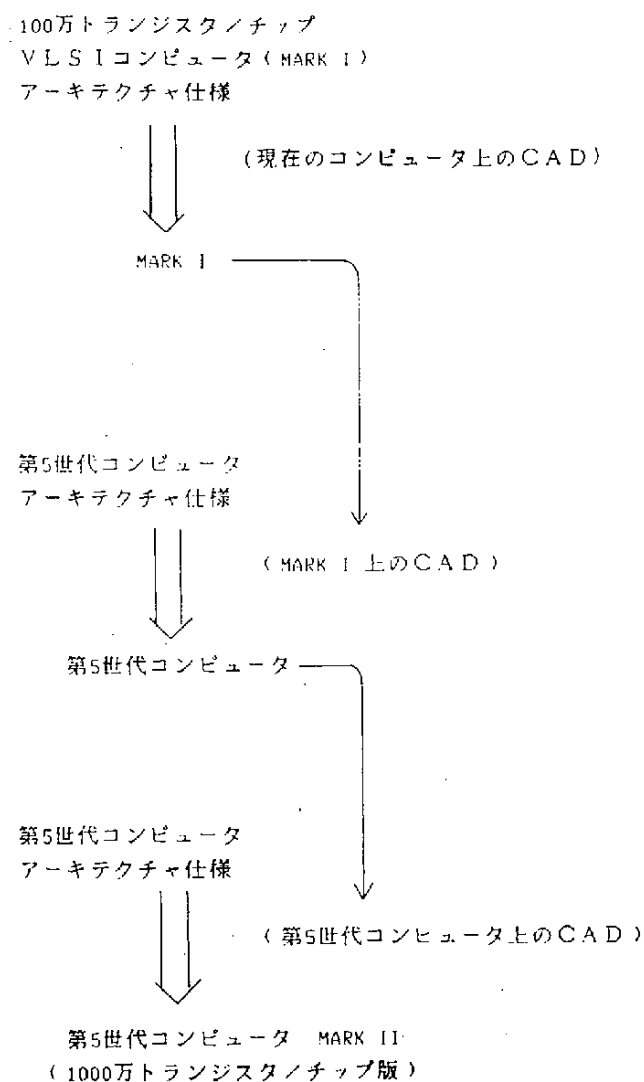


図 2.3 - 9 第 5 世代プロジェクトにおける New ノイマンマシンの開発過程

(2) 現在のマシン (ノイマン型アーキテクチャに基づくもの) の欠点

第5世代開発サポートシステムは研究者・開発者にそれぞれ専用のパーソナル・コンピュータを持たせ、それをローカル・ネットワーク技術で接続したような、いわゆる分散型コンピュータ・システムのイメージである。システムの中核となるパーソナル・コンピュータに要求される機能は、

① 高度マシン・インタフェース・サポート

(1000×1000ドット程度の高密度画面をサポートする高級エディタ, アブストラクト・データタイプ・サポート言語の装備等)

② 十分な速度

(現在の大型コンピュータを一人で使用する程度の速さ)

③ ローカル・コンピュータ・ネットワーク・サポート

などであり、これが低価格で提供されなければならない。アーキテクチャ的にも現在の汎用機はパーソナル・ユースということを意識して設計されているわけではないので問題は多いが、それよりも低価格で提供ということが現在と同じ製法・素材で作っていたのでは到底この期待に沿うことはできない。よって、MARK I, MARK IIともコンピュータ構成素材に革新性がないかぎりこの目標を達成することはできないわけである。第5世代プロジェクトではこの革新的な素材にVLSIを使うことによってのりきる。現在のコンピュータは言うまでもなくVLSIを使って作られているわけではないため、すべてのアーキテクチャをVLSIベースで今一度考えなおさなければならないと思われる。「NewノイマンのNewはノイマンをベースとしつつも従来の研究より知れたノイマン型の欠点をベースをくずさないで改良しているということを表わしている」が、このような半導体技術が未熟なため現在のノイマン型アーキ型アーキテクチャはかなりデフォルメされている。(表2.3-4参照)

そして、この影響は実行効率の向上をもたらしたがプログラムの開発効率を著しく落している。VLSI1チップ化によるNewノイマンマシンMARK I, MARK II共にこれらの欠点をすべてなくすということが開発における最大の目的となる。

(3) 望まれる機能

望まれる機能とは以下の通りである。

① ソフトウェア開発コストの低減のための工夫

前述したように現在の多くのノイマン型マシンはインプリメントする場合の素子の貧弱さから、かなり悪い方向に(デフォルメ)された方式をとっている場合が多い。具体的には従来、マシン構築にあたりまず考えたことは実行効率を上げることであり、プログラム開発効率を上げることはほとんど考慮が払われていない。従ってNewノイマンではこれらを改善することをまず考える。またソフトウェアを開発しやすくするための機能としては、プログラムを書くフェーズで書きやすくするための機能と、書いてしまったものを動くようにする(デバック)ための機能、また、でき上がったソフトウェアを管理しやすくするための機能の三つが考えられる。

表 2.3 - 4 G.J. マイヤーズによるノイマン型コンピュータのゆがみ

① 逐次型メモリ

プログラムとデータは一つの順序づけられたメモリに貯えられる。順序づけられた記憶の概念は、高級言語において名前のついた変数として離散的な集合から成る記憶の概念とはまったく類似していない。

② 線形メモリ

メモリは一次元であるが、高級言語で扱うデータには線形とは限らない多次元のデータ（配列、構造体、リストなど）がある。

③ 命令とデータの区別がない。

命令をデータのように扱ったり、データのビット列を命令として実行できるのに対して、高級言語ではデータを実行したり、命令をデータのように参照する概念はない。

④ 意味はデータに備わったものではない。

データの意味はプログラムの論理、命令によって与えられる。従って、例えば文字列やアドレスデータに対して浮動小数点の演算を行うことができる。高級言語においては、データの意味はデータそのものに結び付けられている。

⑤ 2進演算

人間とのやりとりのため10進数と2進数の変換にシステムはかなりの時間を費やしている。また、ほとんどの10進有理数は計算機内部で2進数の近似値として表現されるため、プログラミングのむずかしさ、エラー、言語の定義を混乱させる原因となっている。

⑥ 固定語長固定語長であるために表わしうる数の最大値と最小値が決まっている。高級言語の持つ可変語長の概念を固定長の語に写像するのが難しい。

⑦ レジスタ

番地づけ可能なレジスタが存在することは、高級言語の概念と合わない。すべての算術演算に必ずレジスタを使うことや少ないレジスタの最適割り当てのためにコンパイラが生成するコードはソース・プログラムの論理を表現するうえでは何の貢献もしていない。

② 高機能化

問題に適応させて命令セットを変更していくというような、主としてセマンティック・ギャップを減らす方向である。本来のノイマン・マシンにはこのアイデアは無かった。

③ 高信頼化

高信頼化は、論理的なレベルと物理的なレベルの二つに分けて考えられる。論理的なレベルでは、いわゆる不注意によるプログラム・ミスを防止するケーパビリティ・アドレッシング・メカニズムの実現などである。他方、物理的なレベルでは自己故障診断機能、または、耐故障（設計）機能の実現ということになる。

④ コンパティビリティ

Newノイマンはもともと evolutionaryな性質の方が大きいので、従来のマシンとのコンパティビリティがとりやすい。具体的にはエミュレーション機能が重要であるが、この場合単なる機械語だけでなく、オペレーティング・システムのエミュレーションをも含む広い意味でのエミュレーション能力が要求される。

又、本機能の実現のための具体的な処理方式は以下の通りである。

- ① ダイナミック（アーキテクチャ）方式
- ② スタック方式
- ③ タグ方式
- ④ ディスクリプタ方式、モジュール・オブジェクト方式
- ⑤ アドレス管理機構
- ⑥ マルチ Newノイマン機構
- ⑦ ノイマン型高速演算機構
- ⑧ プロテクション・メカニズム
- ⑨ Fault Tolerant Computing 機構

② 必要性

ここでは、Newノイマン・アーキテクチャの必要性を MARK I、MARK II の二つに分けて考察する。

(1) 共通項目

MARK I、MARK IIともに共通して必要とされる項目は、

- ① ソフトウェアの生産性向上
- ② 構成素子の進歩による高機能化
- ③ 構成素子の進歩による高信頼化
- ④ コンパティビリティの確保

である。

ここで、特に④に関してコメントをつけるならば、第5世代コンピュータ本体はそれ自身は現在のコンピュータと比べるとかなり革新的であり、コンパティビリティはとりにくいのに比べ、Newノイマン型は同じく第5世代コンピュータではあるが、ノイマン型をベースとしているため、現在のマシンとコンパティビリティをとることが可能である。そして、Newノイマンは第5世代コンピュータでは分散ノードの一つとして位置付けられる。

(2) MARK I

MARK Iは、第5世代機開発サポート・システムのノード・コンピュータであるスーパー・パーソナル・コンピュータの核として必要である。

(3) MARK II

MARK IIはMARK Iの改良、延長型という、より高機能な第5世代コンピュータ実現の1990年に手に入る素子を最大に生かしたコンピュータということの他、第5世代コンピュータ・システムの上に作られる知的CADの実用性を確かめるデモンストレーションとして必要である。

(3) 内外技術の現状

内外技術の現状を考える場合、Newノイマン型アーキテクチャの基本方式の研究は従来も行われてきた。しかし、システムとしてまとまったものはまだ存在しない。その理由は構成素子のVLSI 100万トランジスタ/チップ作成能力が現在ないからである。しかし、いくつかの芽にあたる研究開発が公表されている。1981年2月現在、実現されているものは、45万トランジスタ/チップのものであり、これは米国、Bell研究所、HP社によるものである。

(詳細はVLSIアーキテクチャの項)

また、VLSIベースではないがNewノイマン機構を積極的に取り入れているものとして、IBM社のシステム38、インテル社のIAPX432プロセッサADAMシンなどがある。我国では公表されているものはない。

さらに、MITのAIラボにおける知的CADの研究はMARK II開発研究の芽と考えることができる。なお、我国ではこの種の研究も行われていない。

(4) 開発内容

(1) 概要

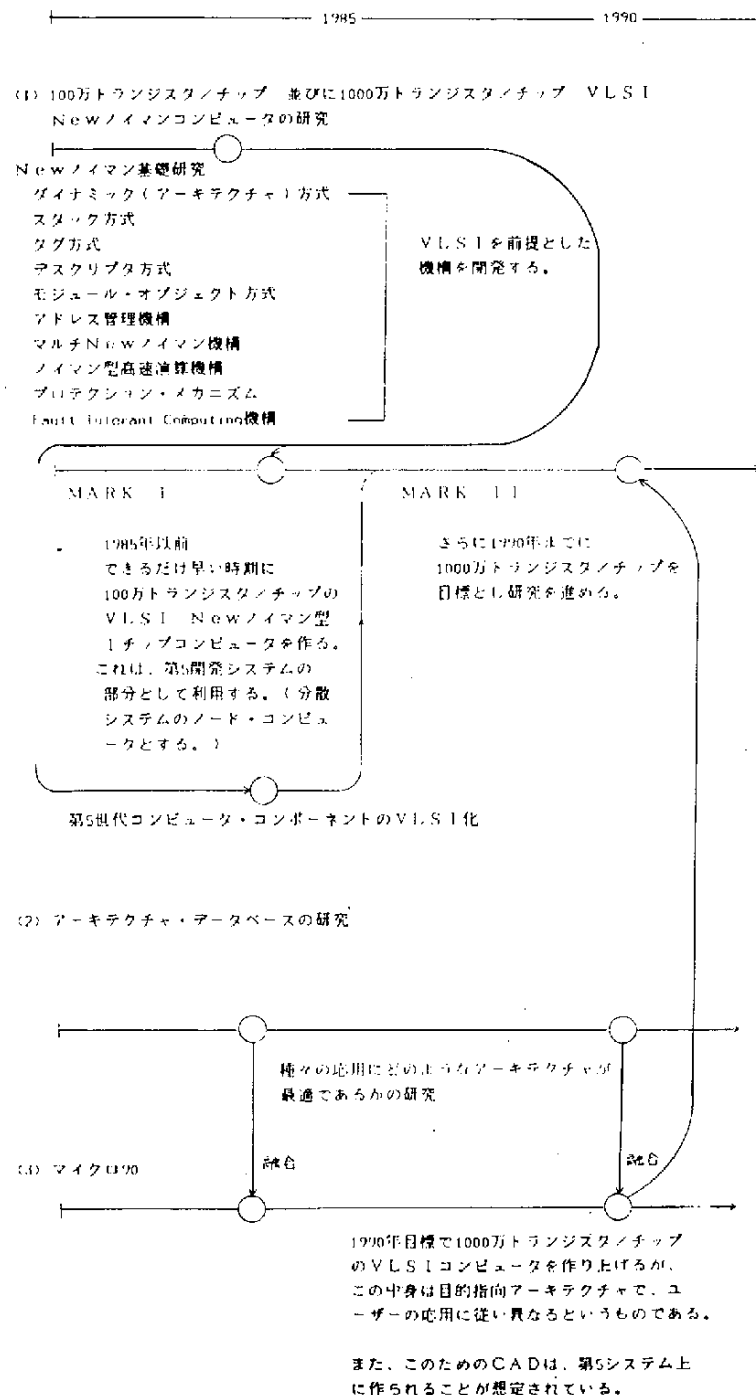
MARK Iは第5世代開発サポート・システムのノード・コンピュータ・システムであるスーパー・パーソナル・コンピュータの中身として考える。Newノイマンをシステムとし、VLSI 1チップでどう作るかだけに焦点を絞る。従って、シングル・ユーザ用を意識すればよく、従来マルチ・ユーザを前提として作られていた部分はこの際取り入れる必要はない。UHPを主体とし、(1)で述べた要求される機能が1チップにおさまるようにし、分散システムの中でのプロテクションを考え、巨大論理アドレス・アクセスが可というようなものを作る。MARK IとMARK IIは分けて考えた方が良く、MARK Iを1985年以前までに、MARK IIを1990年の目標として、プロジェクトをスタートさせる。

研究開発の進め方としては、まず(1)(3)で述べた必要な機能のVLSI化を前提とした研究が必要である。その後、MARK Iへと進める。又、Newノイマン機構が第5世代コンピュータのコンポーネントなることを研究するプロジェクトも必要である。

また、もう一つアーキテクチャ・データベース研究グループにより、このようなNewノイマンVLSIチップをどのように応用問題に適応させるか(主としてNewノイマンのファームウェアとソフトウェア)の研究を行う。さらに、それぞれの計算速度はそれほどではない

が、個々の仕事に合った計算機をそれぞれがどう持つかという、建築で言えばハウス55に相当するようなきたるべき世の国民一人一人が皆コンピュータを持つような時代に備える計画も考えられる（これをマイクロ90という）。研究の中心は、VLSI-CADであるが、これにより目的にあったVLSIチップをユーザーが作ることができるようになる。マイクロ90のためには第5世代コンピュータは必要不可欠であり、第5世代の威力を示すための計画ともいえよう。

(2) 研究開発項目とスケジュール



(3) 研究開発の目標・仕様

① MARK I

- Newノイマンベース100万トランジスタ / チップ プロセッサ
- マイクロプログラム・サポート・ツール
- シングル・ユーザ用オペレーティング・システム
- アブストラクト・データタイプ・サポート・システム記述言語
- ローカル・ネットワーク・サポート・オペレーティング・システム

② アーキテクチャ・データベース

- Newノイマシソ・アーキテクチャ要素の分類並びにデータベースの作成
- 応用分類並びにデータベースの作成
- 応用別評価機構

③ マイクロ90

主要目標はVLSI CADの項参照。

- 知的CADとアーキテクチャ・データベースの結合
- 応用別目的指向アーキテクチャ選択システム

④ MARK II

- Newノイマンベース1000万トランジスタ / チップ プロセッサ
- 適応型オペレーティング・システム
- 応用指向言語

2.4 機能分散アーキテクチャ

2.4.1 機能分散アーキテクチャ

(1) 概要

機能分散アーキテクチャは計算機システム全体の構造を定めるアーキテクチャであって非常に広い概念である。機能分散アーキテクチャにもとづくシステムを構築するためには、各コンポーネント・マシンや結合アーキテクチャを開発しなければならないが、ここでは、主にシステム構築のためのアーキテクチャという側面に重点を置いて一般論的に議論することにする。

(1) 機能分散アーキテクチャの特徴

機能分散アーキテクチャの意味は、一言でいえば、“計算機システムのあらゆる部分、あらゆるレベルにおいて、機能の分散を可能な限り行なってシステムを構築するアーキテクチャ”ということである。このような機能分散形式を採用する理由は、現在、または今後10年程度に予想されるハードウェア技術により、同じく予想される応用側の要請を満たす計算機システムを最も効果的に実現できると考えられるアーキテクチャだからである。

より具体的にいえば、このアーキテクチャの持つ次のような特徴が、望ましいものだからである。

① スループットまたは実行速度の向上

複数の処理要素を同時に動作させる並列処理によってシステムのスループットを拡大したり、同一問題を処理する実行時間を減少させるという効果である。

② システムの成長性、即ち構成の自由度の確保

情報処理システムが高度化するにしたがい、システムの連続的な成長性が強く求められるようになる。また、応用環境の多様化が進んでいるので、応用側の要請に合せた形でシステムを構築するための自由度の確保も重要である。このような目的のためには、モジュール構成の機能分散アーキテクチャが集中型単一システムに比し明らかに優れている。

③ 高信頼性高稼働率の達成

機能分散アーキテクチャでは、一部の構成装置の故障においても、全システムを停止させることなく、フェイルソフト系を容易に実現することが可能である。

④ 機能の専用化による処理効率の向上

機能単位に分割された各单位装置をその機能専用に設計することによって処理効率の向上が期待できる。

⑤ システム構築の容易化

機能分散アーキテクチャにおける各構成単位的设计は、その機能とシステム内の他のモジュールに対するインタフェースが明確に定められ(論理的にも、物理的にも)ていれば、他とほぼ独立した設計を行なうことができ、複雑なシステムをひとまとめに設計するより

はるかに容易である。

この特長や②の特長は直接ユーザから見た計算機の性能に反映されるものではないが、計算機構築の立場からは非常に価値あるものである。

機能分散アーキテクチャにはこの他にも優れた特長をあげることができるが、それらはある特定の部分の機能分散に適用できるものであったり、上記の特長から派生的に生じるもので、基本的には上記の4つの特性に集約することができる。もちろん、こうした特性を実現するには、バックボーンとなるよい理論モデルの考案、各種の実験による設計データの収集等多くの努力が必要である。安易な設計では、機能分散の最大の弱点とされる(時間的、また、物理的等)オーバヘッドによって、使いものにならないシステムができ上がってしまう可能性も少なくないことを銘記すべきである。

(2) 機能分散のレベル

機能分散の考え方は計算機システムのいろいろな部分に適用することができるが、基本的には次の3つのレベルに分けることができ、それぞれ主たるねらいや考え方が若干異なっている。

① ユーザ・サイドと中央システム間、あるいは複数の中央システム間の機能分散

ユーザの手元にあるパーソナル・コンピュータあるいは工業プロセスを管理するプロセス・コンピュータや各部門のデータ処理を行なうコンピュータ等と、中央にある強力なコンピュータ・システムとの間の機能分散である。また、複数の情報システム間の結合もこの中に含めて考えることができる。この部分の機能分散についてはネットワーク・アーキテクチャとして別の独立課題として取り上げられているが、特徴は次の通りである。

- 第5世代アーキテクチャではユーザ・サイドのパーソナル・コンピュータ等のローカルな処理機能が高度化し、通常の処理は全てまかなわれる。情報についてもローカル性の強いものや使用頻度の高いものは極力ローカル・サイトに配置される。
- ローカル・サイトの代表的装置はパーソナル・コンピュータで、これは単に処理機能とユーザ個有の情報管理を行なうだけでなく、ユーザとシステムのインタフェース装置として、システム全体を仮想化してユーザに使い易く提供する役割も持つ。このパーソナル・コンピュータは用途がパーソナルであっても現在同じ名称で呼ばれているものとは比較できないほど高度の機能を持つ機能分散アーキテクチャの重要な構成要素であり、新しい思想のもとに開発を行なわねばならないものである。
- 一方、中央サイトの共同利用される高機能コンピュータ・システムは、ローカルサイトでは機能不足な超高速処理や特殊処理機能の提供と共通情報の管理を主な役目とする。

② センタ高機能コンピュータにおける機能分散

センタの高機能コンピュータは、個々の機能を実行する専用マシンの集合体として構成

される。即ち、機能はマシンを単位に分散され、全体として汎用コンピュータとして動作する。マシンは ユーザ・サービス・マシン (U S M) , マシン・サービス・マシン (M S M) , コントロール・マシン (C T M) の 3 種に分けられる。この内 U S M はパーソナル・コンピュータでは不十分な超高速処理や特殊処理・共通データの管理等の作業を行なうマシン群であって、データベース・マシン, 超高速科学技術計等マシン, 各種高級言語マシン等が含まれる。M S M は遠隔地にあるシステムやパーソナル・コンピュータの制御を行なうマシン群であり, ネットワーク・アーキテクチャの課題中に詳しく論じられている。また C T M は特殊装置や U S M , M S M 群を制御し, 協調して動作させるためのものである。

このレベルの機能分散アーキテクチャにおいて重要な点をまとめると, 個々の超高性能マシンの開発と, それらを統合して動作させるための制御方式の開発, そのバックボーンとなるすぐれた論理モデルの設定等である。

③ コンポーネント・マシン内の機能分散

U S M , M S M 等の各マシンはそれ自身も, 更に機能分散されたマルチ・プロセッサ構造をとって, 先に述べた機能分散の特性が生かされるようにすべきである。この場合, 前の 2 つと異なる点は個々のマシンの特殊性に応じた機能分散構成をとって並列処理効果を最大限に高めねばならないことである。したがって, 高度並列処理技術, マルチ・プロセッサ・マシンの構成技術等の研究開発が重要である。

(3) システムの構成

機能分散アーキテクチャを採用することにしたとしても, その具体的なシステム構成は, 基本となる設計理念によって大きく異なる。この設計理念の確立はそれ自身一つの大きな研究課題である。そこでここでは非常にマクロに表わしたシステム構成の例を図 2.4 - (1) ~ (5) (127 ~ 128 ページ) にあげておくことにとどめる。

これらを比較するとき多階層の機能分散が行なわれ, それぞれのレベルで, しっかりした論理モデルにもとづいた接続, 通信方式が定められているという基本的考え方の一致と具体化における多様性を読みとれる。

[2] 必要性

第 5 世代のコンピュータ・システムは, 単に, 特定の応用を処理できるだけでなく, トータル・システムとして効率的で使いやすいものでなければならない。そのためには, 個々の応用の問題解決技法を改善するだけでなく, システムが, 絶えず進歩し続けるハードウェア(素子)技術を最大限に利用して, これまた進歩し続ける要求される問題の解決を最小コストで実現できるように構築されていなければならない。

このような要請を達成するために最も有効と考えられているシステム・アーキテクチャが機

機能分散アーキテクチャである。その理由は前節に述べた特徴から明らかであろう。新しいシステムはそのような方向に着実に流れている。

このようなシステムを構築するためには、全体的に深く検討された統一的設計理念にもとづき、現実の要請とハードウェア、ソフトウェア及び応用技術の状況にあわせてシステムを設計しなければならない。機能分散アーキテクチャの実現には多大の知恵と時間と費用が不可欠である。

機能分散の基本アイデアは決して新しいものではなく、ある種の新しい応用のような華やかさはないが、この際ハードウェア、ソフトウェアのこれまで蓄積された全技術を集約して、一貫した設計理念のもとに高効率、高信頼で、使い易く、構築しやすく将来の技術改良に対応しやすく、高機能を盛込むことのできる機能分散アーキテクチャの開発に努力がなされることが必要である。

[3] 内外技術の現状

機能分散アーキテクチャは従来からあった考え方を発展させ、実用に耐え、効率的なものとしようとするところに開発のねらいがある。

機能分散の考え方は、古くからあり、部分的な研究は多数行なわれている、統一された論理モデルにもとづき、確立した設計理念によって構築されたシステムの開発は、未だ全くといってよいほど行なわれていないといえることができる。

(1) 国内技術

海外、特に米国に比べるとかなり遅れているのが現状である。

ネットワーク・レベル、センタ・コンピュータ内、コンポーネント・マシンのそれぞれの分野における機能分散アーキテクチャについて、個別に、電子技術総合研究所、電気通信研究所、及び、東京大学、慶応大学等いくつかの大学において小規模の実験レベルの研究が行なわれ、ある程度の成果と基礎技術が得られている。しかし、トータル・システムとしての機能分散アーキテクチャの研究は殆んど行なわれておらず、基本的設計理念の確立を初めとして、研究開発しなければならない項目は多い。

(2) 海外技術

ネットワーク・レベルではARPA網を初めとし、多く実例がある。センタ・コンピュータにおける機能分散アーキテクチャについても古くからいろいろな研究があるが、本格的に機能分散アーキテクチャを追求したものとしてはBCC-500(1970~73)が有名である。また、最近でもカーネギーメロン大学等を中心に各種の研究が進められている。論理モデルについてもいくつかの提案がなされており、(OSI標準化モデル, Actorモデル)また、システム制御方式として分散型オペレーティングシステムについても同様に大学等で研究が進められているし、分散データベース、各コンポーネントマシンのアーキテ

クチャについての研究も少ない。

しかし、システム全体を機能分散アーキテクチャの基本思想のもとに統一的に整理して開発を試みているものはまだない。

[4] 研究開発内容

(1) 研究開発項目

① 機能分散の基本設計理念の確立

機能分散アーキテクチャにおける機能分散の方法、機能モジュール間の通信形態を統一的に表現する論理モデルの研究開発、およびそれにもとづく、効率的機能分割法、マシン間論理インタフェース、物理インタフェース、情報交換方式、情報の分散配置方式等の開発、機能分散システムの記述方法とその処理系、及びシミュレーション等の評価システムの開発。

② システム制御方式（オペレーティング・システム）

機能分散アーキテクチャによるシステムの制御方式は未知の分野である。統一的な思想にもとづくよい方法を開発する必要がある。特に、OSは従来のものが集中処理向きに設計され、一つの大きな計算機能や記憶機能を多数のユーザで効率よく共用することが主眼となっていたのに対し、機能分散アーキテクチャではその基本コンセプトが異なるためにその基本概念からして大きな変革を求められる。したがって多くの研究開発努力を注入する必要があるが、具体的には次のような事項についての研究が必要と思われる。

OSの分散方式、OSモジュール間通信方式（インタフェース）、OSモジュール間の情報交換における障害対策、モジュール間同期方式、単純で実用的なデッドロック回避方式、機密保護方式（暗号化、ケーパビリティアドレッシング）、ジョブ管理、タスク管理等の従来のOS機能の機能分散OS上への展開、ファームウェア化、VLSI利用のハードウェア化、デバッグサポート、ユーザ教育、ネットワーク・アーキテクチャと統合して設計されたローカル・OS（特にパーソナル・コンピュータ上のもの）、OS用コントロール・マシン（OSニュークリアス・マシン）の構成、モニタとその結果による自動チューニング方式、OSの開発サポート技術等。

③ 分散型データベース

計算機システムの主要な機能は、情報の処理と情報の蓄積利用である。一般的にいうと処理機能は必要なものが直ちに得られれば、どのような形で存在してもユーザにとっての違いはあまりない。しかし、情報の利用のほうは、蓄積された情報量が多く、共同利用でできる方がより有効であるから、本質的に集中化の傾向がある。したがって、データベースは機能分散の考え方にそぐわないとも見られるが、事実上、本来ローカルに存在するデータベースは結合してシステム内の全ユーザに蓄積された情報を有効に利用させるのである、と

逆に考えられるべきで、分散データベース構築方式は極めて重要な研究課題の一つである。

具体的には、異なるスキーマの変換とその処理方式、非数値処理機能、高速データ転送、信頼性の高い放送通信機構、データベース間の制御コマンドの迅速な授受方式、データ・セキュリティ（暗号化方式）、高速大容量補助記憶（ 10^3 B 幅～1 GB）、知的アクセス等の技術の開発が必要である。

④ コンポーネント・マシンのアーキテクチャ

各機能の処理を担当するコンポーネント・マシンも現存のものに比しはるかに高性能のものを開発する必要がある。6個の新アーキテクチャの課題はその構築技術としてぜひ研究しなければならないものである。個々のマシンでは計算という作業のために最高性能を出すことを目標とする高速演算マシンと情報記憶処理の中心となるデータベース・マシンは別の独立した課題としてあげられているし、マシンサービスマシンはネットワークアーキテクチャの課題に含まれている。

この他ではユーザと直接インタフェースする高性能パーソナル・コンピュータの開発とセンタ・コンピュータ内の高性能高級言語マシンのアーキテクチャが非常に重要である。具体的にいうと前者に関しては基本となるVLSI化ユニバーサル・ホスト・プロセッサとその上でのファームウェア高級言語マシンのアーキテクチャを初めとして、OS、学習機構、図形、音声、文字等の処理技術とそのための入出力装置、大容量小型パーソナル・ファイル、単純で使い易く、図形も処理できる新しい高級言語等の研究開発を行なう必要がある。後者については、中間言語の設定方式、複数言語処理アーキテクチャ、並列処理アーキテクチャ、高度デバッグ機能、適応機能、言語仕様の差の吸収方式等の研究開発が重要である。

⑤ 自動適応化方式（ダイナミック・アーキテクチャ）

機能分散アーキテクチャでは、システムの柔軟性を生かし、各レベルで与えられた要求や環境にシステム側が自動的に適応できる機能を組込むことが望ましい。そのためには、自動化方式の理論的背景の確立、命令セット、並列構造、情報配置、資源割当、マシン結合構成、信頼性と処理能力等の自動適応化方式、学習機構、動作データの収集、ダイナミック・マイクロ・プログラミング技術等の新規開発、あるいは現存技術の改良の研究が必要である。

⑥ インプリメンテーション

実際に機能分散アーキテクチャによるシステムを構築するには、実装、高速通信の可能な接続方式、VLSI化等のインプリメンテーション技術も機能分散アーキテクチャの特徴を生かせるように改良する必要がある。

⑦ その他

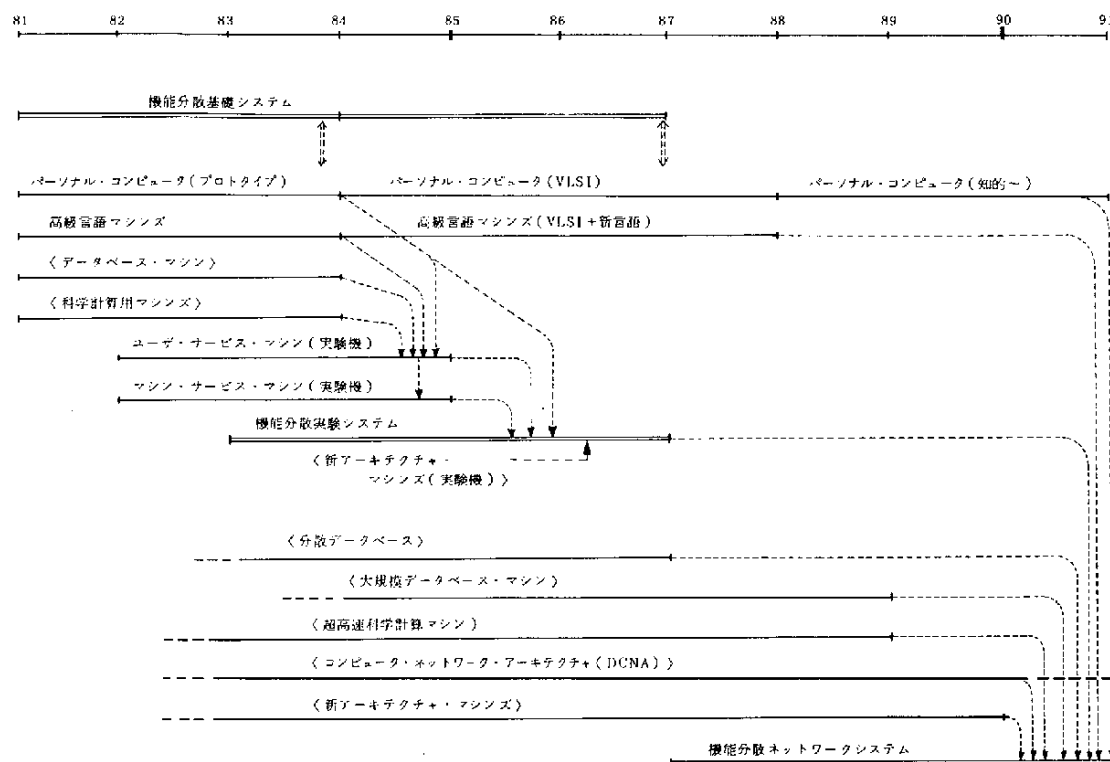
以上の他、一般的な基礎技術として、次のような技術の高度化のための努力が必要である。

機密保護方式、記憶制御方式、信頼性（可用性）向上方式等。

(2) 研究開発スケジュール

本課題の研究開発スケジュールを次に示す。

機能分散アーキテクチャの研究開発スケジュール



(3) 研究開発の目標・仕様

前項で示したスケジュールにあるように、機能分散アーキテクチャ・システムの開発は次の3つのフェーズに分けて行なうのが適当である。

① 機能分散基礎システム

機能分散アーキテクチャは、システム・アーキテクチャであるから、各種の基礎技術をうまく統合することが必要である。よって、このフェーズでは、いくつかの部分的な小規模実験システムの構築、理論的検討、シミュレーション等によって次のフェーズ以後に利用するための基礎技術を確立する。具体的には次のようなサブテーマについて研究する。

- 機能分散のバックボーンとなる論理モデルの確立
- いくつかのアプローチについて小規模実験システムを構築し、コンポーネント間通信方式、コンポーネントマシンの方式、並列処理方式等の基礎アーキテクチャを確立する。

- c. 分散OSを中心としたシステム制御方式の基礎技術、機能分散システム記述言語、その処理系の開発。
- d. 機能分散の各レベルにおける並列処理構造、命令セット、信頼度等の環境に対する動的適応（ダイナミック・アーキテクチャ）方式の開発。
- e. 物理インタフェース、実装方式等機能分散システムのインプリメンテーション技術の開発。
- f. ユーザ・マシンとして接続される各種専用マシン・アーキテクチャと、その構築技術。

② 機能分散実験システム

①及び他の課題の中間成果を取り入れながら、機能分散アーキテクチャによる実験システムを構築する。具体的にはパーソナル・コンピュータ20台程度のローカル・ネットワークとし、中心のコンピュータは、10台程度の各種高級言語マシンと他課題で完成した実験マシン（科学技術計算マシン、データベース・マシン等）を組込む。これに関し本課題中で開発を行なう分は次の通りである。

a. パーソナル・コンピュータ

基本となるVLSIユニバーサル・ホスト・プロセッサ及び、パーソナル・コンピュータの基本アーキテクチャを開発し、現在の標準大型機程度の計算能力を持ち、マシン・インタフェースを備えたシステムを開発する。

b. 高級言語マシン群

10種程度の専用又は汎用高級言語マシンを開発し、システムに接続する。

c. ローカル・ネットワーク

a,b及び他課題による成果を総合したローカル・ネットワーク中心の機能分散システムの構築。

③ 機能分散ネットワーク・システム

①、②の成果及び、他課題の最終成果を統合し、複数のセンタ・コンピュータをグローバル・ネットワークで結合したシステムの構築。

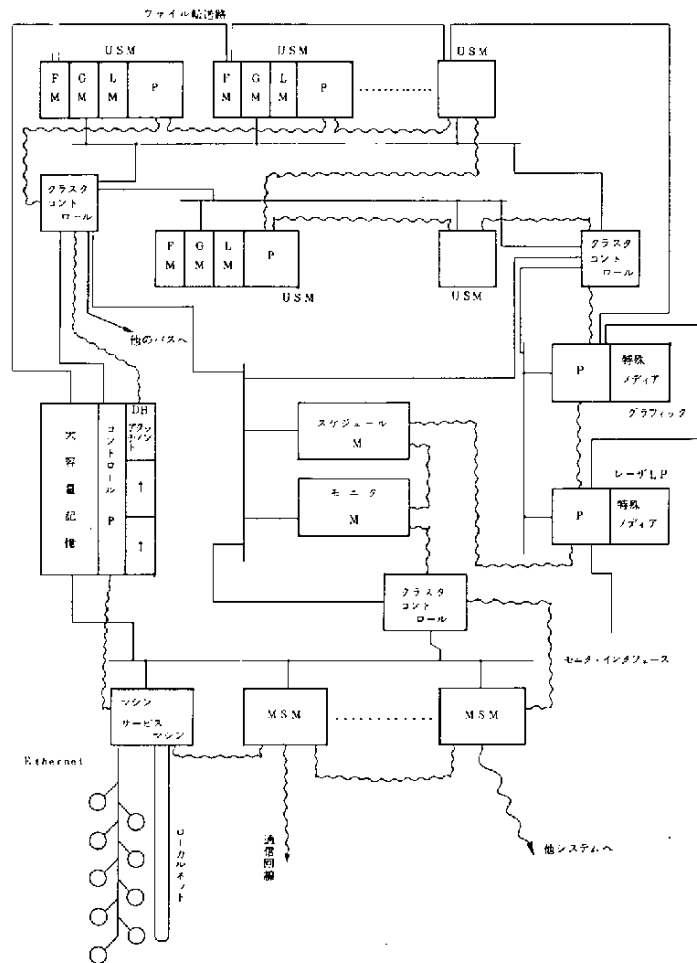
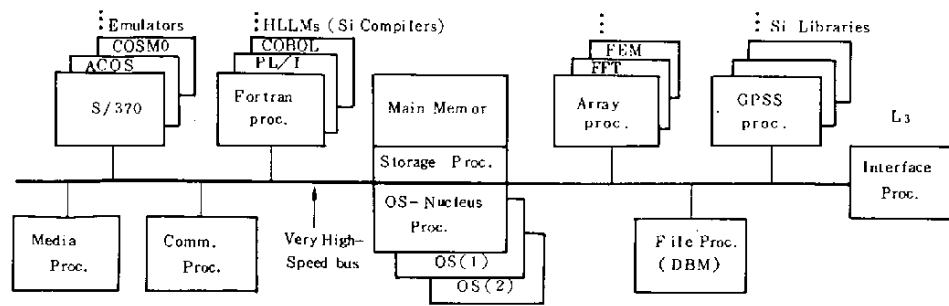
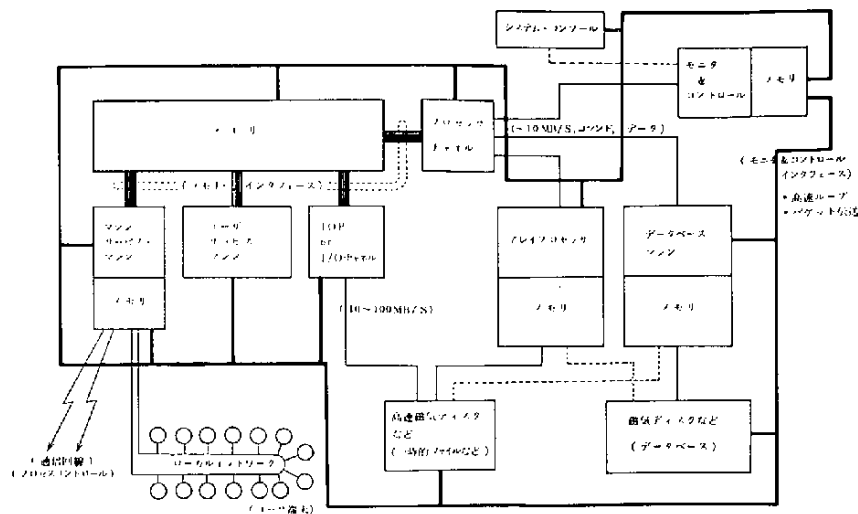


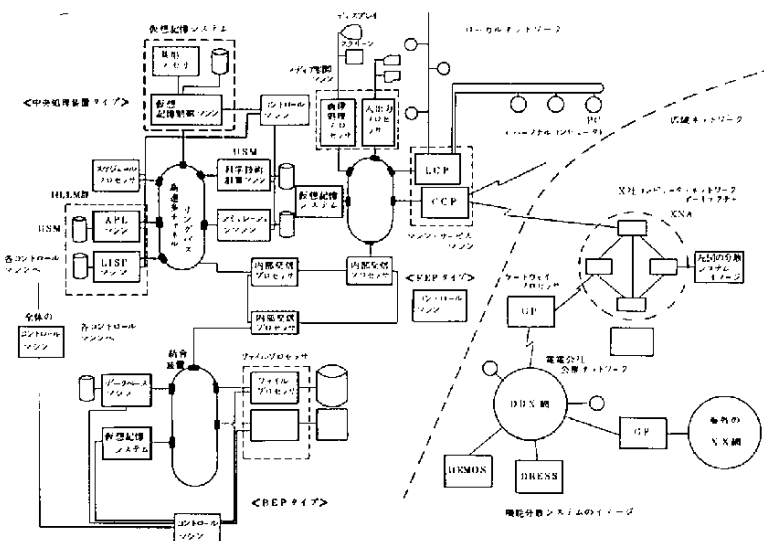
図 2.4 - 1 システム構成例 (1)



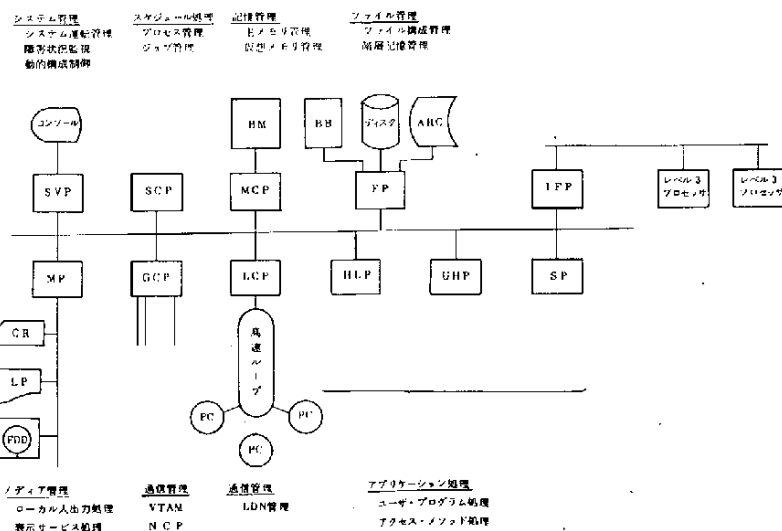
システム構成例 (2)



システム構成例 (3)



システム構成例 (4)



システム構成例 (5)

2.4.2 ネットワーク・アーキテクチャ

〔1〕 概要

コンピュータ技術と電気通信技術の進歩、個人の社会活動や企業活動の多様化・高度化・広域化に伴ない、情報化社会の構築が進展している。情報化社会では、情報の生産・流通が価値を生む社会であり、地域的に分散配置されている複数のコンピュータや入出力端末を通信回線で有機的に結合したコンピュータ・ネットワークが実現される。情報化社会の中核としての役割りを果たすコンピュータ・ネットワークは、以下の条件を満たす必要がある。

① コンピュータ・システムを構成する資源（各種マシンのハードウェア、ソフトウェア）を、そのシステムに属している利用者だけでなく、他の異なるコンピュータ・システムに属しているマシンや利用者からも、経済的に共用利用できるようにする。——— 資源共用による経済化

② データベース等を複数のマシンに分散し、災害時・システム障害時の危険分散を可能とする。——— 資源分散による信頼性の向上

③ 利用者に近い場所で一部の処理を行なうことにより、応答時間や通信コストの削減をはかる。——— 分散処理による価格性能比の向上

④ コンピュータや端末の種別、ネットワークの構成に依存しないソフトウェア作成を可能とする。また、新しいコンピュータ、端末、新しいサービスの導入を容易にする。

——— 仮想化、および開発・導入・運用・移行の効率化

ボトム・アップにコンピュータ・ネットワークを実現するアプローチや、他のコンピュータ・システムを無視して、個別に独自のネットワークを構成するアプローチでは、上述の条件を達成することが非常に困難となる。このため、コンピュータ・ネットワークを構成する各装置の役割りと、それらの間の通信に開する規約（プロトコル）を確立する必要がある。

このような考え方から生まれたものが、ネットワーク・アーキテクチャという概念である。すなわち、ネットワーク・アーキテクチャとは、物理的な構成要素から成るネットワークを、論理的なノード（例：ユーザ・サービス・マシン、マシン・サービス・マシン、パーソナル・コンピュータ、端末等をモデル化したもの）とリンク（例：通信回線、チャネル等情報を選ぶ媒体をモデル化したもの）で表現し、すべてのノードに対して共通の論理的な階層を定め、各階層ごとにプロトコルを規定したものをいう。

機能分散アーキテクチャと同じく、コンピュータ・ネットワークを実現するためには、このネットワーク・アーキテクチャの研究が不可欠である。

一方、ネットワーク・アーキテクチャに基いて実現される第5世代のコンピュータ・ネットワークは、ネットワークの地域的な広がり観点から、ローカル・ネットワークと広域ネットワークに分類され、その主な構成・機能は、以下のようになると考えられる。

① パーソナル・コンピュータが発達し、これらを結合したローカル・ネットワークが、

研究機関，大学，企業内，地域社会内等で実現される。

② 高速通信回線による資源共用ネットワークが広域ネットワークとして実現される。
また，ローカル・ネットワークは，広域ネットワークと結合される。

③ 利用者は，通常，ローカル・ネットワーク内の資源にアクセスするだけでよい。しかし，他の資源を必要とする場合は，特別の配慮を必要としないで，ローカル・ネットワークのゲートウェイを通して広域ネットワークの資源にアクセスできる。

④ 広域ネットワークは，企業等が独自に実現する専用ネットワークと，N T Tが提供する公衆ネットワークから成り，両者は必要に応じて結合される。また，これらは，海外の広域ネットワークとも結合される。

⑤ ネットワークには，電話をはじめ，データ・テレフォン，ファクシミリ，家庭用T V，パーソナル・コンピュータ，センサ，O C R，複写機，等を接続でき，音声，データ，図形，画像等の多様な通信メディアを利用した通信が可能となる。

⑥ 高度の機密保護と高い信頼性を実現している。

上記のようなコンピュータ・ネットワークを実現するためには，ネットワーク・アーキテクチャの研究の他，④高性能通信回線技術（光通信等），⑤広域通信方式（衛星通信方式等），⑥ネットワークの中核となるマシン・サービス・マシン，④ネットワーク利用を容易にするネットワーク用ソフトウェア（ネットワークO S）の実現，⑥マルチメディア処理技術，等の技術の研究が必要である。

本研究開発課題は，①ネットワーク・アーキテクチャの確立と，②コンピュータ・ネットワーク実現技術，に関するものであり，特に，機能分散アーキテクチャの研究開発課題と密接な関連がある。

「2」 必要性

コンピュータ・ネットワークは，情報の伝達・蓄積・交換・高度の情報処理を行なうシステムであり，種々のコンピュータ，入出力端末等を通信回線で疎結合したものである。このため，機能分散アーキテクチャとしてとらえることも可能である。

コンピュータ・ネットワークを効率的に実現するためには，ネットワーク・アーキテクチャという基本概念の確立が必要なことは，先に述べたことからすでに明らかである。

ネットワーク・アーキテクチャの体系化・標準化による効果を，コンピュータ・ネットワークを実現する立場から，ここで改めて要約すれば以下のようになる。

① コンピュータ・ネットワークを構成する各種マシンの機能分担を明確にし，製品の体系化をすることができる。

② 情報処理と通信処理の機能分担を明確にし，機能の重複に伴う二重投資等を事前に回避できる。

③ 独立に、目的別に実現された他のコンピュータ・ネットワークとの接続が容易になる。機能分散アーキテクチャと同じく、ネットワーク・アーキテクチャの確立は、放っておいても自然に達成できるものではない。第5世代の世の中のニーズと、それを実現するために必要とされる機能を十分に把握した上で、トップ・ダウン的なアプローチを重視して研究を進める必要がある。

これまでの経験から、コンピュータ・ネットワーク実現のために必要な技術の基礎は固まってきた。しかし、より高度なネットワーク実現のためには、多くの問題点が残されている。

例えば、ネットワーク・アーキテクチャの標準化は、ISO、CCITT等で進められているが、上位層に向けての国際標準の早期勧告が望まれている。また、既存の異なるコンピュータ・ネットワークの統合は、その計画といくつかの試みがスタートした段階にある。ネットワークOSも限定された利用範囲内で実用化が進められているのが実情である。通信処理と情報処理は、次第にその区別がつかなくなり、現在は混沌とした状態にある。

理想のゴールから見れば、現在はスタート地点に立つために必要とされる道具だてと、スタート前の基礎実験が終った段階にあるといっても過言ではない。

[3] 内外技術の現状

(1) 国内技術

ネットワーク・アーキテクチャの研究は、1974年にIBMがSNAを発表して以来、各コンピュータ・メーカーを中心に進められた。現在、各メーカーは、自社のネットワーク・アーキテクチャを開発している。これらには、ANSA（東芝）、DINA（日電）、DONA（沖）、FNA（富士通）、HNA（日立）、MNA（三菱電機）等がある。NTTは、メーカー4社と協力してDCNAを開発している。しかし、これらの間に非両立性は、ネットワークの高度利用が進むにつれて特に問題となる。また、郵政省も国内標準プロトコルの設定を目標としたCCNPをまとめており、その一部は1980年11月に告示された。

ネットワークOSの研究は、JIPNET、N-1（大学間コンピュータ・ネットワーク）等で進められている。高位プロトコル・サポート、ネットワークJCL、等が実現されている。

ローカル・ネットワークの研究も進められている。例えば、Ethernetに関しては、慶大でAcknowledging Ethernetを、東大でPriority Ethernetを、早大でReservation Ethernetを試作している。また、光ループネットワークも注目されてきている。

広域公衆ネットワークとしてDDXが1980年に主要都市に建設された。また、KDDは、国際ネットワークの実現を進めており、その第1期として、1980年にICAS（International Computer Access Service）を開始した。専用コンピュータ・ネットワーク・システムは多数実現されている。今後は、異なるシステム間の結合が進展しよ

う。

公衆通信への適用を狙った光通信方式は、一部現場試験を終了している。中容量デジタル光伝送方式（6～100Mビット/秒）は、比較的早期に商用化できよう。

衛星通信方式は、CS^{*}、BS^{**}を使用して基礎実験を進めている段階である。

(2) 海外技術

ネットワーク・アーキテクチャの代表例は、IBMのSNAである。またネットワーク・アーキテクチャ標準化組織として、ISOとCCITTがある。主として、前者はシステム利用者の立場から、後者は通信網提供者の立場から、それぞれ標準化を進めている。

ネットワークOSの研究は、アメリカでは主にARPANETを使用して行なわれている。XNOS、RSEEXEC、分散データベース用のSDD-1/2等のプロジェクトがある。例えば、RSEEXECでは、ネットワーク・ファイル・アクセスの統一、重要ファイルの多重複写、コマンド/プログラムレベルのファイル操作、既存のコンピュータ・システムで作られたプログラムがそのままネットワーク環境下で使えるカプセル化技法、等を実現している。

ローカル・ネットワークの代表例としては、カリフォルニア大アーバインのDCS、ベル研のSpider、Xeroxで開発されたEthernet、CATV利用のMITRIX、等がある。DEC、Xerox、インテルの3社は、Xeroxが特許を持つEthernetの拡張と製品化についての共同開発を進めている。

広域ネットワークの建設も進められており、Telenet、TYMNET（以上アメリカ）、Datapac（カナダ）、TRANSPAC（フランス）、EURONET（EC9か国）等がある。

光通信方式の研究も活発である。イギリスでは、8,140Mビット/秒の現場試験を既に終え、イタリアも140Mビット/秒の現場試験を終了した。^{***}

衛星通信方式は、海外では一部商用化されている。現在注目されている衛星通信を利用したコンピュータ・ネットワーク・システムには、アメリカのSBSとXTENがある。これらは、画像会議サービス、電子郵便サービス、等を提供する予定である。

マシン・サービス・マシンは、国内外で多数実用化されている。現在、ハードウェアは漸進的進歩が続いている。ネットワークOS等の機能分担とその実現が大きな問題になっている。また、LSI技術の進歩とネットワーク・アーキテクチャの標準化により、通信用LSI

*)CS：実験用中容量静止通信衛星，愛称さくら。

**)BS：実験用中型放送衛星，愛称ゆり。

***)公衆通信用

や暗号用LSIが実現されてきている。前者には、ECL-1533(DCNA共同開発), WD-2501/2511(W.D.社), Intel-8273, 等があり, 後者には, Intel-8294, TI-9940, WD-2001/2002, Fairchild-9414, 等がある。マシン・サービス・マシンの回線スループットは, 最大48Kビット/秒クラスのものが多い。また, 最大収容回線数は, 256回線クラスが中心であるが, 800~1,000回線まで収容可能なものも出現している。なお, 最近は, 構内交換機にデータの処理機能を付加したEPABXも, マシン・サービス・マシンとして利用されてきている。

[4] 研究開発内容

コンピュータ・ネットワークの実現に関する研究開発内容は, 広い分野にまたがっている。このため, 第5世代の研究では, コンピュータ・アーキテクチャに直接関連する部分を中心にとり上げ, 光伝送通信方式や衛星通信方式等, 他の研究に委ねることとし, ここでは特にとり上げない。

(1) 研究開発項目

① ネットワーク・アーキテクチャの確立

ネットワークを効率よく実現し, 各種のコンピュータ, 入出力端末等を容易に接続するために, ネットワーク・アーキテクチャの体系化・標準化を行う(基本理念の確立)。

将来にわたって発展性, 拡張性のあるコンピュータ・ネットワーク実現のためには, 次の目標の達成が重要である。

- ② 複数の異なるマシンに共通な論理構造とプロトコルを提供する。
- ③ 新データ網(パケット交換網)と, コンピュータ, 端末等との間で, 通信機能の適正な分担を行なうことができる。
- ④ 公衆ネットワークと専用ネットワークの双方に適用できる。
- ⑤ 統一された仮想端末プロトコルを提供する。
- ⑥ 効率的に実現できるプロトコルである。
- ⑦ 音声・データ・図形・画像等のマルチメディア用通信プロトコルを提供する。
- ⑧ 本格的な分散処理プロトコルを提供する。
- ⑨ 国際標準と整合する。

また, ネットワークの高度利用を可能とするため, 次に述べるネットワークOSの研究とあわせて, 高位レベルの概念を確立する研究を重視する。

② ネットワークOSの確立とその実現

ネットワークOSは, ネットワーク利用の容易化, 負荷の分散, 資源の共用等を達成するために必要なソフトウェアとプロトコルの集まりである。このネットワークOSは, 1つのマシンに集中的に配置されるのではなく, ネットワークを構成する各マシンに分散配

置され実現される。

第5世代で必要とされる主な機能には、次のようなものがある。

- ④ ネットワーク・ジョブ管理機能
- ⑤ ネットワーク・データ管理機能
 - ・ネットワーク・ファイル管理機能
 - ・分散データベース管理機能
- ⑥ データ移送機能
- ⑦ プロセス間通信機能
- ⑧ ネットワーク制御機能

なお、分散データベースの実現技術に、特に重点を置く必要がある。

③ プロトコルの記述、作成、検証技術

コンピュータ・ネットワークの高度化に伴ない、規定されるプロトコルも増加し、複雑なものとなる。このため、プロトコルを論理矛盾なく簡潔に記述する方法や、インプリメントのために切り出されたプロトコルを実験装置等を使用して検証する技術が、これまでに以上に重要となる。

第5世代では、この技術をさらに進めて、インプリメント条件（物理的な構成条件とサービス仕様）を与えることにより、検証されたプロトコルと各マシン用のソフトウェアを自動生成する技術や、プロトコルの修正・追加等に対して、すでに規定されたプロトコルとの矛盾を自動的にチェックしながら、新しいドキュメントを編集・出力する技術の研究等を行なう。

④ マルチメディア処理技術

第5世代では、高度のマン・マシン・コミュニケーションシステムを構築するため、コミュニケーション・メディアを多様化させる必要がある。したがって、音声、データ、図形、画像信号等、異なるメディアを統合して扱う技術や、各種メディア間の変換技術を研究する。

例えば、各種のメディア信号を画一化でき、さらに伝送速度の異なる信号の多重、挿入、分割が容易なデジタル・ネットワークや、パケット形式で音声や画像を伝送する方式を、マシン・サービス・マシン・アーキテクチャ開発の一環として行なう。

⑤ マシン・サービス・マシンの実現

第5世代におけるマシン・サービス・マシンの代表的な利用形態は、以下のようになると考えられる。

- ① 前置処理装置（FEP：Front End Processor）
- ② 遠隔集線多重化装置（RC：Remote Concentrator）

㊦ メッセージ交換装置 (MSW: Message Switch)

㊧ ゲートウェイ装置 (GP: Gateway Processor)

この他、テレメータやテレコントロール等を目的とする形態、端末群を一括管理する形態等がある。

このような多様な利用形態に耐えうると共に、先に述べたネットワークOS等の高度な機能を実施するマシン・サービス・マシンを開発する。すなわち、㊦より一層の高速回線のサポートや、マルチメディア機能、ネットワークOS機能をリアルタイムに処理できる高い処理能力を実現するだけでなく、㊦24時間連続して運転ができる。㊧特殊な設置環境を必要としない、㊦高い信頼性を持ち、無保守である、㊧小形・軽量・低消費電力・安価である、㊦拡張性、増設性が高く、ネットワークの変更柔軟に対処できる、等の条件を満たすマシン・サービス・マシンを開発する。また、超高速伝送インターフェース用VLSI、デジタル信号処理や伝送制御処理等を行なう通信用VLSI；次に述べる暗号処理用VLSI、等についても開発を進める。

㊦ 機密保護機構の実現

個人のプライバシーに関する情報や、企業内、国家の機密情報が伝送路上から盗まれる危険性や、特殊装置を回線に接続することにより、各種ファイルの破壊や修正が行なわれる危険性等を減少させるため、暗号処理機能や個人識別機能等をネットワークの中で実現する。

(2) 研究開発スケジュール

研究開発スケジュールを次ページの表に示す。

研究開発は、以下のように行なう。

㊦ 前期 (ステップⅠ)

ネットワーク・アーキテクチャの基本理念の確立を行なう。

㊦ 中期 (ステップⅡ)

パーソナル・コンピュータおよび各種入出力端末を結合した、実験用マルチメディア・ローカル・ネットワークと、そのために必要とされるマシン・サービス・マシンを開発する。この上で、ネットワークOSの実現等の基礎実験を行ない、コンピュータ・ネットワーク実現技術を確立する。

㊦ 後期 (ステップⅢ)

実験用マルチメディア・ローカル・ネットワークと、高性能ユーザ・サービス・マシン群を結合し、本格的な広域機能分散システムを実現する。この上で分散データベースシステム、オフィス・オートメーション・システム等を構築し、第5世代の知識情報処理用コンピュータ・ネットワークとする。

なお、電電公社は、すでに広域公衆ネットワークとしてISDN (Integrated Services Digital Network) の開発を発表している。したがって、必要に応じて強調して研究を進める。

ネットワーク・アーキテクチャの研究開発スケジュール

	1	2	3	4	5	6	7	8	9	10年
① プロトコルの記述・作成・検証	基 本 検 討					試 作				
② ネットワーク・アーキテクチャの確立	第 1 版		第 2 版					第 3 版		
③ ネットワークOSの確立と実現	第 1 版 (ジョブ管理等)		第 2 版 (分散データベース管理等)					第 3 版		
④ マルチメディア処理技術	基 礎 研 究		ローカルネットワークによる 基 礎 実 験					試 行 実 験		
⑤ マシン・サービス・マシンの実現	通信用VLSIの 開発		実 験 マ シ ン の 試 作					実用マシンの試作		
⑥ 機 密 保 護 機 構	基 礎 研 究		試 作							
(関連技術：光通信方式 衛星通信方式 などは、他の成果 を利用する。)			実験用マルチメディア・ローカル ネットワークの実現と評価					第5世代広域機能分散 コンピュータネットワーク の実現と評価		
(総 合 シ ス テ ム)										

(3) 研究開発の目標・仕様

- ① ネットワークOS機能を実現するとともに、高位レベル・プロトコルを確立する。
- ② マルチメディアを利用した各種サービスを総合して提供できるコンピュータ・ネットワークとする。
- ③ ローカル・ネットワーク内のデータ伝送速度は、100～500Mビット/秒以上とする。また、ローカルネットワークを広域公衆ネットワーク(DDX等)に接続できるようにする。
- ④ ホストとなるコンピュータ・システムの数は、3システム以上とする。
- ⑤ マシン・サービス・マシンの信頼性を現在よりも1桁以上向上させる。また、VLSI化することを目標とし、マイクロプロセッサや通信用VLSIチップを積極的に導入する。

2.4.3 データベース・マシン

[1] 概要

コンピュータは元来、数値計算のために発明されたものであるが事務処理（いわゆるデータ処理）への応用によって飛躍的にその利用が拡大し、コンピュータの著しい発展をもたらした。今日においてコンピュータによる高速、大量のデータ処理機能は不可欠のものになり、しかも応用の高度化に伴って、多数のオンライン・ユーザが共有化したデータへ並列的にアクセスするシステムが必要となり、データベース・システムが導入されてきた。

データベース・システムは今後ますます大容量化、高速化が要求されると考えられている。しかしながら現在のコンピュータのアーキテクチャは数値計算指向で作られたフォンノイマン型が主流であり、データベース・システムで要求される大量のデータの管理、ファイル装置とのデータの入出力などには適していないと云われており、既存のデータベース・システムですら十分に性能要求を満たしていない場合が多いのが現状でこのままでは将来において要求と実現性能の差は一層拡大すると思われる。そのためデータベース・システムに適したアーキテクチャを持った専用コンピュータ即ち、データベース・マシンの研究が盛んになり、既に商用データベース・マシンも出現している。

データベース・マシンの一般的な方式は；

- ① 階層的に作られたホスト上のOSのデータベース管理機能の内下層の諸機能をデータベース・マシンにオフロードする。
- ② データ操作/管理のための効果的な実行機構をもつ。
- ③ ホスト・コンピュータとデータ記憶装置間のデータ転送ネックを解消するための工夫をする。などである。

[2] 必要性

データベースの利用は拡大しており今後も拡大をつづけると予想されていて、近い将来にはピーク時 10^6 要求/秒以上の処理能力が要求されるようになるかもしれないとの予測すらある。しかしながら現在のデータベース・システムは性能の高いものでも $10 \sim 10^2$ 要求/秒程度の処理能力しかない。またデータの量も現在のそれ（現在でも大きいシステムでは 10^{12} バイトを越す）よりはるかに大きくなると考えられ一説では 10^{18} バイト程度も実現性があるといわれている。このような大規模、高速のデータベース・システムを現在のコンピュータで実現することは性能の面で困難であるのみならず必要になるであろうハードウェアやソフトウェアの規模、複雑さから考えて信頼性の面でも大きな問題を生ずると思われる。現在のコンピュータのアーキテクチャは数値計算向きでありデータベース・システムで要求されるデータ操作/管理やファイル記憶装置との高速のデータ入出力などには最適とはいえず、データベース・システムに専用化したアーキテクチャを持ったコンピュータに依らなければ高度化する要求に適合

したデータベース・システムは構築できないと考えられる。こうした背景からデータベース・システムの基本機能を専用ハードウェアで支援し、性能およびコストを改善し、ソフトウェアも単純化するデータベース・マシンが注目されている。

データベース・マシンの必要性について現在のデータベース・システムの問題点との関連でまとめると以下のようである。

① 現在のデータベース・システムのソフトウェアの複雑さの最大の原因は問合せで与えられるシンボリック名をメモリ番地へマップするために複雑なデータ構造が必要であり、挿入、削除等の基本操作に対し、この複雑なデータ構造を正しく保持しなければならない点にある。これに対し現在のフォン・ノイマン型コンピュータは数値計算や単純なデータ処理を主目的として最適化されており、この複雑なデータ構造の処理、ファイル記憶装置（磁気ディスク等）とのデータの入出力制御等の目的には必ずしも適したものと言えない。

② 現在のデータベース・システムは多くの機能的に異なる部分（例えば問合せのパーシング、ディレクトリのアクセス、ディレクトリの処理、データ検索、追加・削除、セキュリティ管理）から成っている。これらを現在のコンピュータで同一のハードウェア上に実現するとバランスのよいシステムを構成するのが困難で多くの部分にボトルネックを生じやすい。これに対し、ホスト・コンピュータとの処理機能をうまく分担したデータベース・マシンによって解決をはかることが可能と考えられるし、更にデータベース・マシンに各種の専用のハードウェアを付加することにより多様な機能に対する処理能力のバランスをはかることもできるであろう。

③ データベース・システムは今後ますます大規模、複雑化することは明らかであり、また多様な応用の発生あるいは使用パターンの変化に伴う再構成の問題を避けて通ることはできない。このため、データベース・システム用のソフトウェアは大規模化しそれを実現するには大規模ソフトウェアの設計開発技術の確立と共に、データベース・システムの基本機能のハードウェア化によるソフトウェアの単純化が必要である。

これらの問題を解決するにはフォン・ノイマン型コンピュータと本質的に異なったアーキテクチャが必要と考えられ、種々のデータベース・マシンが研究されている。その基本構成は図 2.4-2 に示すようなものであり、

基本概念は次のようである；

① バック・エンド型

フォン・ノイマン型のホスト・コンピュータをフロント・エンドに置きデータベース・マシンをバックエンドに置く、データベース・マシンにデータベース・システムのどの範囲の機能を持たせるかは各システムにより異なる。

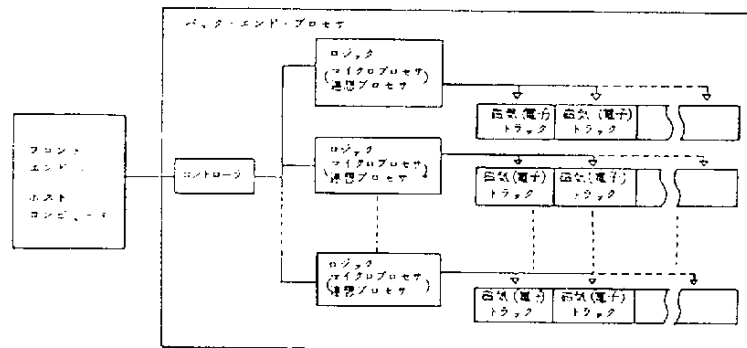


図 2.4 - 2 データベース・マシンの基本構成

② コンテント・アドレッシングと並列処理

シンボリック名から直接二次記憶装置へアクセス可能なコンテント・アドレッシングの機構がデータベースマシンのアーキテクチャの基本となる。この結果当然前述の複雑なデータ構造が必要でなくなるあるいは簡素化される。コンテント・アドレッシングでは探索単位に従って二次記憶装置を適当に分割することにより並列的探索が可能になるので性能が大幅に改善できる。

③ データベース・システムの高次機能をデータベース・マシンのプリミティブとすることにより性能向上を得られる。例えば関係データベースにおけるセレクション、ジョイン等をプリミティブとしてハードウェア機能に持たす試みは多い。

[3] 内外技術の現状

(1) 国内技術

データベース・マシンあるいはデータベース・マシンの基礎となる技術の研究は国内においても最近盛んになり、多くの大学、研究所、メーカ等で進められているが商品化されたものはまだない。大学では北海道大学、東大、慶応大、横浜国立大などで研究が行なわれている。

北海道大学では専用のハードウェアによるサーチ、ソート処理の単位とするデータフロー型のデータベース・マシンを試作中である。東京大学ではCellular Logic Typeのデータベース・マシンを発展させた可変構造多重処理データベース・マシンによってリレーショナル・データベースをサポートする研究を行っている。このシステムでは高速の光ループに多重のプロセッサ・モジュールを接続し、プロセッサ・モジュールとメモリ・モジュールの結合を可変にすることで高性能化を狙っている。東大ではこの他データ管理機能を独立したサブシステム化したデータ・サブシステムの研究や同一の処理モジュールを多数ならべた

大規模連想プロセッサの研究も行っている。慶応大学では、データの格納および演算を属性単位で行い、演算の中間結果を動的マーク・ビットに保持する方式のリレーショナル・データベース・マシンを機能分散型マルチ・マイクロ・プロセッサで実現する研究を進めている。横浜国立大は独自のデータ・モデルをベースとした汎用コンピュータとミニコンピュータの密結合型データベース・マシンを試作している。

電総研は1976年から1980年にかけて磁気バブルを用いたデータベース・マシンを開発した。このシステムはマイクロ・プロセッサ、主記憶、磁気バブルメモリを組合せたデータ・モジュールを基本単位とし、これを複数台共通バスに接続した並列処理方式をとっている。

日電中研は汎用コンピュータをホストとしこれとメモリを共有したミニコンピュータを用いたデータベース・マシンを研究している。このシステムは汎用的に設計されていて任意のデータモデルを実現できるが、現在研究中のものはCODASYL仕様のデータベース環境下で動作している。

以上の外にも電電公社、沖電気、東芝などで研究が進められている。

このように国内においても研究は盛んになっているが実用レベルの規模のマシンは少ないなど海外に比較して進んでいるとはいえない。

(2) 海外技術

ここでは米国の状況を中心に述べるがヨーロッパにおいても研究は盛んである。データベース・マシンの研究は1970年代はじめの、

- ・高速連想プロセッサSTARANのデータ管理への応用の研究
- ・ARPANET用のデータ・ユティリティとしてのCCA社によるDatacomputerの研究
- ・ベル電話研究所におけるバックエンド型のデータベース・マシンXDMSの研究

などに端を発し、米国の大学を中心に盛んになった。1972年にフロリダ大学のCASSMプロジェクト、1974年にカナダのトロント大学のRAPプロジェクトなどがはじまっている。そして1980年には初の商用データベース・マシンがBritton-Lee社から発表され(IDM500)実用化時代が始まった。これはまだ小規模なデータベース・システムを指向したものであるが、今後の発展を考えると重要な意義を持っていると思われる。

フロリダ大学のCASSMはセルラ・ロジックをデータベース・マシンに適用した最初の試みとして知られており、ディスクとその各トラックに付随したロジックから成るセルから構成し、ネットワーク・モデル、階層モデル、関係モデルのいずれもサポート可能である。

トロント大学では1975年にRAPプロジェクトを開始し1976年にはRAP. 1として完成、その後大幅な設計変更をしてRAP. 2を1977年に作っている。現在はアンカラ大

学に移った Ozkarahan を中心に R A P. 3 が開発されている。R A P も C A S S M と同様にセルラ・ロジック方式をとっており、同様の方式の研究はユタ大学、テクトロニクス社、I B M ワトソン研究所などでも行なわれている。またオハイオ州立大学では R A P と同様に、小容量高速の Content Addressable Memory と可動ヘッド・ディスクを改良した疑似的な Content Addressable Memory を組合せてこれにいくつかのデータベース機能プロセッサを付加したデータベース・マシンを研究中でその基本設計を基に、あるコンピュータ・メーカーが試作中ともいわれている。同様のデータベース・マシンには M I T の I N F O P L E X, フロリダ大学の M I C R O N E T, ウィスコンシン大学の D I R E C T, パーデュー大学の D I A L O G などがある。

Britton Lee 社はリレーショナル・データベース機能をもつ初の商用データベース・マシン、(I D M 5 0 0) を 1 9 8 0 年 2 月 に発表した。これは D B プロセッサ、2 5 . 6 K バイトの主メモリ、ディスク制御装置、ホスト接続用のチャネルなどから構成され、システム当りの最大磁気ディスク容量は 3 2 G B である。西独系のソフトウェア A G 社も 1 9 8 0 年 1 0 月にデータベース・マシンを発表している。これは、I B M と互換性のあるプロセッサ上に A D A B A S パッケージをのせ I B M 系ホスト・コンピュータとチャネル結合するものである。この他、M R I / I N T E L 社のデータベース補助プロセッサがあるし、I B M システム / 3 8 をリレーショナル・データベースを内蔵した一種のデータベース・マシンとする見方もある。

このように海外においては以前から行なわれていたデータベース・マシンの研究の成果をもとに実用レベルのマシンが出てきている点は特に注目されるべきである。将来におけるデータベース・マシンの重要性を考えると日本においてもその研究や実用化を促進する強力な施策が望まれる。

[4] 研究開発内容

(1) 研究開発項目

大容量、高性能のデータベース・マシンを実用化するための研究開発項目には次のようなものがある。

(a) データベース・マシン・アーキテクチャの研究

- データの探索、照合、などを効率よく扱える新しいマシン・アーキテクチャの研究が必要である。たとえば関係代数マシンはリレーショナル・データベースの処理に向いていると云われているがその具体的研究が必要である。
- 連想処理もデータベース・マシンの高性能化に有効と考えられ、連想処理をベースとした新しいアーキテクチャのプロセッサ(連想プロセッサ)の研究も必要である。

- 大容量のデータベースを格納するメモリは低速、即ちアクセス・タイムが長くならざるを得ず、したがって問合せに対して短い応答時間を達成するには小容量高速のメモリを組合せてうまく使用することが必須となる。この記憶階層の構成法、や制御方法の研究が必要である。
 - データベース・マシンの処理速度を高めるには並列処理の概念を導入せざるを得ないであろう（たとえば並列にサーチするなど）。したがってデータベース・マシンの中に多くの並列処理的要素を取り入れるための研究およびデータベース・マシンを並列に用いる研究を進めねばならない。
 - データベース・マシンと他の機能分散プロセッサとの間でどのように機能分担するのが最も効率がよいかの研究が必要である。通信のオーバーヘッドが少なく、論理的、物理的にシンプルな切り口（インタフェース）を設定しなければならない。
 - この他研究項目としては、複数ホストからの共有方式、複数タスクに対する排他制御やデッドロック制御の方式、機密保護方式（パスワード、アクセス制御リスト、ケイパビリティ制御、暗号化など）、などがある。
 - 大容量のデータベース・マシンに対しては高い信頼性が要求されるので、高信頼化の研究は非常に重要となる。ハードウェアの冗長構成、障害回復方式（バックアップ方式を含む）の研究が必要である。
- (b) 各種データベースのサポートについての研究
- リレーショナル・データベースがデータベース・マシンにも適していると考えられているが現存する他の各種データベースに適用できるデータベース・マシンが当面は重要である。
- それにはそれらの各種データベースに対して前述のような研究を進めなければならない。
- またハードウェアとしてはリレーショナル・データベースを基本としたマシン構造とし、その上に各種データベース・モデルを構築する方法（シミュレーションなどを含む）の研究も必要であろう。
 - さらに、既存データベースからリレーショナル・データベースに変換する方法についての研究が必要である。大容量のデータベースの変換はコンピュータで自動的に処理できないかぎり実現性がないと思われる。
- (c) 高機能データベース・マシンの研究
- 厳密な手順に従った問合せだけでなく、あいまいな問合せに対しても推論機能などを活かして応答できるデータベース・マシンが望まれ、そのために人工知能などの研究成果を取り入れた高機能化の研究が必要である。

(d) マンマシン・インタフェースの研究

- 使い易いマンマシン・インタフェースを提供するために、自然言語（日本語）によって問合せが出来るとか、音声や図形で入力や出力が行なえることも望まれ、このための研究が必要である。
- また、それ以前の問題として現在の問合せ言語をもっと使い易いものに改良することが必要で、QBEを改良したようなものとか、まったく別のアプローチによる問合せ言語を研究しなければならない。

(e) 分散データベースの研究

- 分散データベースを実現するには多くの解決すべき問題がある。データベース・マシンを分散データベース・システムに用いるときにもそれらの問題は伴うので、その解決のための効果的サポート（ハードウェア機能等）を研究する必要がある。ユーザが分散配置されたデータベースを分散を意識せずに使用できるようにする方式、重複データの管理、最適分散方式、ネットワークを介してアクセスするときのプロトコルの標準化、異なるデータベースへ自由にアクセスできるようにする方式、障害管理などのテーマがあり、データベース・マシンのハードウェアとソフトウェアへも影響があろう。それに対応した研究が必要である。

(f) VLSI技術の利用に関する研究

- VLSI技術の研究により現在は入手できない、あるいはコストが高くて利用できないような素子が利用できる可能性が出てくる。それらをうまく利用することによってデータベース・マシンの性能やコストを著しく改善できると予想されそのための基礎研究が必要である。また、逆にデータベース・マシンで利用価値の高い素子（その機能、構成、性能など）を研究して、提案しVLSI技術で実現してもらうよう方向づける必要もある。

以下に現在考えられる例について触れる。

- VLSIによって機能アレイや論理メモリなど繰り返し構造をもった素子をつくり、これを各種のアプリケーションに利用する研究が行なわれている（例：Kungのシストリック・アレイなど）。このアプローチはVLSIの特質をうまく活かしたものでデータベース・マシンへの応用も重要な研究課題である。
- 大容量の連想メモリがVLSIで実現できるようになるとデータベース・マシンでも利用価値が高い。連想プロセッサのデータベース・マシンへの適用の研究の環境としてとりあげるべき課題である。
- 磁気バブルメモリなどを利用した電子ディスクはVLSI技術によって大容量、低コスト化できると予想されるのでその利用方法、データベース・マシンに適した

磁気バブルメモリの構成などの研究が必要である。

- J JやGaAsなどが実用化されるとデータベース・マシンのアーキテクチャも現在考えられているものと根本的に異なるものになるかもしれない。たとえばJ Jによって、超高速、大容量の連想メモリが実現できたとすればそれを中心としてシンプルな構造の高性能なデータベース・マシンが実現できるであろう。

このように新しい素子の実用化は既存のアーキテクチャへのインパクトが大きく、十分な研究が必要である。

(g) データベース・マシン設計の基礎データの収集や分析

- よいデータベース・マシンを設計するには基本となる各種のデータを収集し分析しておかねばならない。たとえばデータベースへのアクセスパターンが応用分野によってどう異なるのかなどのデータを収集分析してデータベース・マシンのアーキテクチャ、構成、記憶階層などを設計に反映しなければならない。

(h) システムとしての総合評価

- データベース・マシンのように複雑で新しいシステムは完成後実際に使用して各種の性能評価データを収集し、ボトルネックとなるところを改良する研究がなければ実用的なマシンにはならないと予想される。このため実験レベルのマシンでハードウェア、ソフトウェアを含めて十分な評価研究を行い、それを実用レベルのマシンの設計に反映することが重要である。

(2) 研究開発スケジュール

1985年頃を目標に現在のデータベース・システムの10～100倍の性能、容量をもったデータベース・マシンを開発し、これを実験機として十分に評価、研究を行う。その結果を反映して、さらに10倍以上の性能をもった実用レベルのマシンを1990年頃に開発する。スケジュールの詳細についてを表6-13に示す。

(3) 研究開発の目標・仕様

① 実験機(1985年頃)

- ・容量 : 最大100GB, 但し、オンラインで高速にアクセスされる部分で、バックアップのための大容量記憶装置(MSSなど)は含まない。
- ・処理能力 : 10³件/秒
- ・データモデル : リレーショナルを基本にしてハードウェアを構成するが、システムとしてはリレーショナル以外のデータベースもサポートできるようにする。このため必要なハードウェア機能があれば付加する。

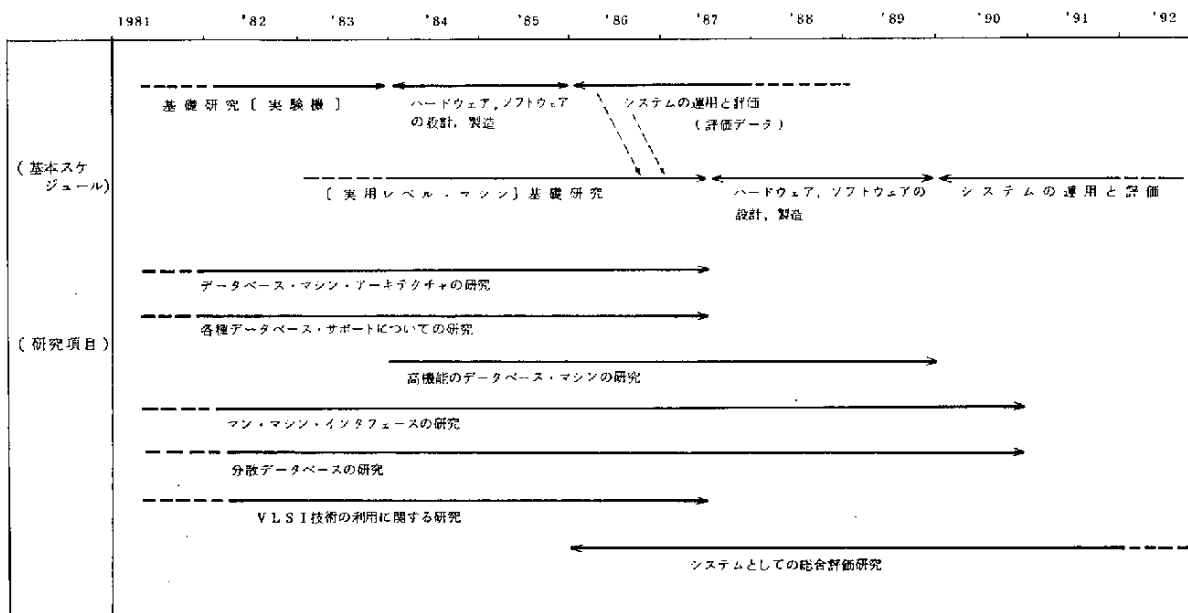
② 本番機(1990年頃)

- ・容量 : 最大1000GB, 但し前述の如くバックアップ・メモリの容量はこ

れに含まない。

- 処理能力 : 10^4 件 / 秒
- データモデル : リレーショナルを基本とするが他のモデルもサポートする。他のデータベースからの変換ツールもサポートする。
- その他 : 自然言語による問合せなど使い易いマンマシン・インタフェースを提供する。

データベース・マシンの研究開発スケジュール



2.4.4 高速演算マシン

〔1〕 概要

科学技術の進歩とともに大規模化、複雑化する科学技術計算を高速処理するための高速演算マシンが強く望まれている。

物理学上の理論研究，工学上の構造物の設計，その安全性，さらには高価な費用を必要とする実験に代る数値モデルによるシミュレーションなどのために高速演算マシンの果たす役割と重要性は，1990年代にはますます増大すると予想される。

現在のマシンで処理できない問題を，現実的に意味ある時間内に処理できるようにするために，実効的に現在の最高速マシンの100倍程度の性能をもつものが望まれる。

具体的には， $10^3 \sim 10^4$ MFLOPSの演算処理能力と $10^2 \sim 10^3$ M語を主記憶をもち，強力な並列処理向け言語とそのコンパイラをもつマシンの開発が目標である。

この目標に到達するためには，高速論理素子の開発または高密度実装の開発が必要であるとともに，並列処理方式の新しいアーキテクチャの開発が必要である。

高速論理素子としては，ガリウム砒素電界効果素子，ジョセフソン接合素子などの新しい素子の研究も必要である。また，大規模集積化技術，高密度実装技術，冷却技術の開発が必要である。アーキテクチャ面では，対象となる問題の数値解法の研究とそれに適合できる並列処理方式の開発が必要である。

〔2〕 必要性

高速演算マシンを必要とする科学技術計算分野としては，気象計算，原子力計算，核融合研究，分子科学研究，航空工学，土木建築構造計算，半導体製造，資源探査，画像処理などが挙げられる。

気象計算では，現在の予報よりも時間間隔の長い中期ないし長期の予報，台風の進路予報などの大規模計算が容易になれば，その社会的，経済的效果が大きい。

原子力関係では，原子炉設計計算をはじめ安全性の解析，シミュレーションなど原子力発電所の安全性の問題に寄与することができる。

核融合関係では，電磁流体の安定性，そのシミュレーションなどの計算により，電磁流体発電などの未来のエネルギー開発に寄与することができる。

分子科学関係では，分子の電子軌道の計算などにより，化合物生成の経済的な方法などを探ることができる。

航空工学では，高速飛翔体の設計計算に寄与するところが大きい。特に，費用のかかる風洞実験の代りに数値計算シミュレーションがある程度まで可能になるので，その経済的效果が大きい。

構造計算関係では，大型化する建造物の安全性，耐震性などの計算に寄与する。

半導体製造では、大規模集積化する電子回路の計算が、半導体の設計に不可欠であり、その計算も集積度の向上に伴いますます大規模化しつつある。

資源探査では、石油の探査などの探査技術が進むにつれて探査信号の処理量の要求は、ますます増大しつつある。

画像処理では、例えば、国土資源調査のための画像処理など新しい要求が生じつつあり、画像処理が必要とする計算量は、今後ますます増大する。

〔3〕 内外技術の現状

（1） 国内技術の現状

大型の技術計算の分野では、現在、主として汎用大型機が使用されている。これらの汎用大型機の性能は、近年の半導体技術の進歩によりLSI化が進んだことおよびキャッシュ・メモリ、先行制御、パイプライン制御などの方式設計技術の進歩などにより向上し、十数MIPS達している。この性能は海外の汎用大型機のそれと肩を並べている。

また、技術計算の高速処理のニーズに対処して、技術計算専用マシンの開発も行なわれつつある。

技術計算専用マシンとして開発されたものに、FACOM 230 / 75 APU (Array Processor Unit) および HITAC M-200H IAP (Integrated Array Processor) とがある。前者は汎用マシン FACOM 230 / 75 と非対称マルチ・プロセッサを構成し、後者は汎用マシン M-200H の内部に付加機構として内蔵される。

最近、後者のタイプのものとして、ACOS 1000 統合アレイプロセッサ、COSMO 700 IAP などが発表されている。

FACOM 230 / 75 APU は、アレイ・プロセッサ専用の FORTRAN を用意し、ユーザはアレイ・プロセッサを意識してプログラムを作成することができる。

一方、内蔵型のタイプは、汎用マシンの付加機構として用意され、標準 FORTRAN で使用できるなど、性能価格比の面や使い易さの面で特長をもち、実効的な処理能力を高めている。

国内で開発されたものは、いずれもパイプライン方式に分類されるもので、並列処理型やプロセッサ・アレイ型のものは、まだ開発されていない。

しかしこれらの技術計算専用マシンは、米国を中心に開発されているスーパー・コンピュータと呼ばれる技術計算専用マシンと比較すると、未だ十分な性能を持っているとは言えない。

一方、論理素子としては、ECL または CML 素子を使用しており、集積度は1チップあたり100のレベルに達している。ガリウム砒素電界効果素子の研究は1チップあたり100ゲート以下の小、中規模ICを作るレベルに来ている。

ジョセフソン接合素子は、現在は論理回路としては、単位ゲートの実験レベルであり、記憶回路では数10～2Kビットでの実験レベルである。

(2) 海外技術の現状

汎用大型機分野では、IBM社を中心に、高性能の機種が開発されつつある。汎用大型機の性能は、国産機と同等程度であるが、今後、IBM社による大容量の主記憶をもつさらに高性能な機種の発表が予想されている。

高速演算を必要とする分野に対しても、主として汎用大型機が使用されてきたが、既に1960年代の終りから技術計算専用マシンの開発が行なわれてきた。近年、CRAY-1の開発により、技術計算専用マシンの効果が確められ、高速演算へのニーズが高まりつつある。技術計算マシンの開発とその状況は次のようである。

1970年代始めに、ILLIAC-IV, STAR-100, ASC (Advanced Scientific Computer) などが開発された。これらはスーパーコンピュータとも呼ばれ、ILLIAC-IVは並列処理、STAR-100とASCはパイプライン処理方式のマシンである。

1976年にCRAY-1が開発され、その後、BSP (Burroughs Scientific Processor), CYBER-203, CYBER-205, DAP (Distributed Array Processor) などが開発された。これらには最新のハードウェア技術が使用され、1970年代始めのものに比し、よりコンパクトでより高性能なものになっている。

処理方式の面では、CRAY-1とCYBERがパイプライン方式、BSPが並列処理方式 (BSPは80年12月に製品化を中止された)、DAPがプロセッサ・アレイ型と多様である。

CRAY-1は、パイプラインのクロックが12.5 nsと小さく、また最新の高速半導体メモリを多量に使用して高速処理を実現しており、現在世界中に最も数多く設置されている。また、CRAY-1は標準FORTRAN仕様のままで使用でき、ある程度の汎用性をもつ技術計算マシンとして開発されている。

一方、高速処理の要求はさらに高まりつつあり、NASAではNASFプロジェクトとして1BFLOPS以上の処理能力をもつNavier Stokes Equation Solverを計画している。

技術計算プログラム用の言語としては、一般には、ベクトル計算用の拡張仕様を設けたFORTRANが使用されているが、そうした拡張仕様を統一化し、アレイ・プロセッサ向けの標準FORTRAN仕様を作成しようとする動きもある。

一方、論理素子としては、国内と同じくECLまたはCML素子が主流であり、国内とほぼ同等の集積度のものを使用している。

ガリウム砒素電界効果素子は、今後の高速演算素子として世界中で注目されている。中でも米国では、国防省の援助による研究が進められている。

ジョセフソン接合素子は、IBM社、ベル研究所を中心に10年以上の研究が続けられており、最近では30 ps / ゲートのOR回路、2 KビットのRAMが実現されている。

[4] 研究開発内容

(1) 研究開発項目

開発の最終目標は高速演算マシンであるが、総合システムとして稼働させるためには、ハードウェア技術からソフトウェア、入出力装置に至るまでの技術開発が必要である。ここでは、システムの運用に必要なものなどは除いて考えることにし、高速演算マシンの開発そのものに必要な開発項目を挙げる。

開発内容は、高速演算マシンとして必要な高速演算機構と記憶装置およびシステム制御機構である。

開発すべき技術を大分類化すると次のようになる。

- (a) ハードウェア技術の開発
- (b) アーキテクチャと論理方式の開発
- (c) ソフトウェアの開発
- (d) 大容量高速二次記憶の開発
- (e) マンマシン・インタフェースの開発

これらについてさらに分解して以下に説明する。

① 高速論理素子の開発と実用化

高速演算を行なうため、高速の論理素子の開発が必要である。同時に配線遅れ時間を小さくするため、高密度実装を可能にする必要がある。

これらを可能にする新しい素子として、ガリウム砒素電界効果素子やジョセフソン接合素子などのもつ可能性を研究する必要がある。

② 高密度実装および冷却技術

高密度実装を可能にするためのVLSI技術、マルチ・チップのハイブリッド実装技術などの開発が必要である。同時に高密度ピンのコネクタ技術の開発が必要である。

また、素子の高速化、高密度化に伴ない、単位体積あたりの発熱量が増大するため、効率のよい冷却技術、例えばフロン冷却などの技術の開発が必要である。

③ アーキテクチャ

高速演算マシンの論理仕様として、命令語の形式、種類、詳細仕様を開発する必要がある。また、アドレス方式、数表現など技術計算と密接に関連する仕様についての研究が必要である。

④ 並列処理方式

$10^3 \sim 10^4$ MFLOPSの演算を行なうためには、並列処理方式を採用して行く必要が

あると予想される。しかし、例えば、連立偏微分方程式を連立一次方程式として、これを数値解法により解く場合を考えてみると、現在は必ずしも並列処理に適したアルゴリズムとは言えない面もある。

並列処理化に向けたアルゴリズムの開発とアルゴリズムに適合する並列処理方式との両面の研究開発が必要である。

⑤ 技術計算用的高级言語仕様

現在は、物理学、工学などの数式で表現されたもとの問題を離散化し、数値解法の対象となる数式で表現し、これに数値解法のアルゴリズムを適用し、その手順をFORTRANなどの高級言語で記述している。

並列処理方式の高速演算マシンに適合したFORTRAN拡張言語仕様の開発などが必要である。さらに、離散化した数値に近い表現のままで直接に記述できるようなイメージの超高級言語の研究も必要である。

⑥ 自動並列処理化、最適化コンパイラ

コンパイラによる自動ベクトル化技術、並列処理化技術、オブジェクト・コードの最適化技術の開発が必要である。

⑦ 高速大容量二次記憶

現在は、二次記憶装置として10msオーダーのアクセス・タイムの回転体の磁気記憶装置が使用されている。計算したい問題のデータが主記憶装置の容量を越えて二次記憶装置に格納されると、計算時間が二次記憶装置と主記憶間の転送時間で決まるようになる。演算速度を高速にしても、二次記憶装置の高速化を図らなければ、全体の計算時間を短縮することができない。

主記憶は100nsオーダーのアクセス・タイムなので、 μ sオーダーのアクセス・タイムの大容量二次記憶装置の開発が必要である。

⑧ 高機能マンマシン・インタフェースの開発

計算の中間結果を容易に表示できるシステムの開発および2次元、3次元表示の容易なグラフィック・ディスプレイ装置の開発が、計算結果の確認および理解を早めるために重要である。

以上の技術開発を行なう目的で、初めに実験機を開発し、アーキテクチャ、論理方式、基本となるソフトウェア、基礎的なハードウェア技術などの確認を行なう。

次に試作機を開発する。これは高速高密度実装を用いて開発し、ソフトウェア、大容量二次記憶、マンマシン・インタフェースなどについても最終仕様のものを開発する。

(2) 研究開発スケジュール

高速演算マシンの研究開発スケジュール

開発項目 \ 年次	1	2	3	4	5	6	7	8	9	10
実験機の開発	実験機開発					評価				
試作機の開発						試作機開発				評価
ソフトウェアの開発	並列記述言語とそのコンパイラの開発									
アーキテクチャの開発	処理方式、命令語仕様の開発									
ハードウェア技術の開発	高速論理素子の開発					論理ユニットの開発				
	二次記憶素子の開発					二次記憶装置の開発				

(3) 研究開発の目標・仕様

開発は、実験機と試作機とに分けて行なう。

① 試作機

(a) 性能 $10^3 \sim 10^4$ MFLOPS

(b) アーキテクチャ

並列演算方式

(c) ソフトウェア

並列記述言語とそのコンパイラ

制御用プログラム

(d) 高速主記憶 容量 256 MB

(e) 大容量高速二次記憶 容量 1 ~ 10 GB

(f) 大容量高速ファイル

容量 10 ~ 100 GB の高性能ファイル

(g) 高機能グラフィック・ディスプレイ

② 実験機

(a) 性能 $10^2 \sim 10^3$ MFLOPS

(b) アーキテクチャ

並列演算方式

(c) ソフトウェア

FORTRAN 拡張言語とそのコンパイラ制御用プログラム

(d) 高速主記憶 容量 64 MB

(e) 大容量二次記憶 容量 256 MB

2.4.5 高度マンマシン・コミュニケーション・システム

〔1〕概 要

人間と計算機との対話の円滑さは、今後の計算機システムを特徴づける大きな要素である。知識情報処理の一つの目的は、計算機に、意味を理解できる能力や簡単な“常識”を持たせることで、この対話の円滑さの向上を実現することである。対話のメディア（媒体）としては、従来は、英数字とカナ文字が中心であった。また、言語は、プログラミング言語やコマンド言語（JCL）などであった。これを、人間の通常の対話のメディアである音声、図形、画像を含むものとし、言語は、自然言語を用いられるように、計算機的能力を質的に変換しようと試みる。

このような対話の実現には、音声や図形等の理解のための理論やソフトウェア面の研究が、第一に重要であるが、これと共に、この実現のためのハードウェアの研究も同時に進めねばならない。

このようなマンマシン・インタフェース用の入出力システム、特に、ハードウェア・システムの研究開発が、この課題の目的である。

第5世代コンピュータ・システムの研究開発においては、この入出力システムの研究開発の目的は、2種類に分けられる。その一つは、音声合成、出力システムや図形表示、印刷システムのように、理論やソフトウェア面の研究が、かなり進んでおり、ハードウェア面の研究開発が、早期に開始できるものである。この種のもののシステム開発は、主として、VLSI技術等を利用し、より高性能のものを作ることと、それを、研究開発支援ツールとして、できるだけ早く提供することが重要である。

第2のものは、音声入力や画像入力のように、その後にある認識や理解のための理論や方法に多くの研究を必要とする未知の分野を残しているものである。この種のものの研究開発においては、ハードウェアの開発は、音声理解や図形・画像理解の理論や方式面の研究と密接に関連づけて進めねばならない。

ここで開発するものとしては、音声や画像を信号レベルで処理する専用プロセッサなどが中心となり、理論やソフトウェアの研究のサポートの性格が強く、本格的なハードウェアの開発は、プロジェクトの後期に集中することになる。ここでは、カスタム化VLSIを容易に設計・製作できるVLSI-CADシステムが不可欠となる。

また、表示装置や図形入力装置等は、プラズマ・ディスプレイや、高精度のブラウン管やTVカメラ等、部品・材料面での改良研究や新しい方式の開発等が重要である。しかし、これらは、物性面の研究等、広大な分野にわたるため、この課題の範囲を越えている。

このため、ここでは、これら部品、材料については、システム開発時点で使用可能なもののうち、最も優れたものを用いることとする。

〔2〕 必要性

入出力ハードウェア・システムの主なものをメディア別に述べる。

文字の入出力としては、漢字を含む和文の入出力システムが、研究開発ツールとして、早期に必要となる。これは、高精度の表示装置、印刷装置、また、カナ漢字変換機能を含む入力装置を含む。

音声に関しては、音声合成・出力装置が、研究開発ツールとして必要である。これは、機械翻訳や自然言語理解の研究のほか、種々のソフトウェア、ハードウェアの開発支援ツールとして、実験能率の向上に、大きく貢献することが期待できる。音声入力装置は、理論、ソフトウェアの研究開発をサポートすることを目的として、できる限り高性能のハードウェアを開発する必要がある。最終的なプロトタイプにおいては、ソフトウェア面の研究成果に基く仕様に従い、VLSI等を使用した専用ハードウェアの開発が必要である。

図形・画像の入出力に関しては、まず、白黒2値図形の入出力装置を開発し、研究開発支援ツールとして提供する必要がある。入力装置としては、高精度のタブレット型の座標入力装置を開発する。また、出力装置としては、高精度のグラフィックディスプレイ装置と、ハードコピー用の印刷装置が必要である。これらは、図面の入出力、グラフや表などの図版を含むドキュメントの入出力に用いられるほか、文字入出力装置と組合せて、図面を含む書類などの作成に用いられる。

白黒濃淡、および色彩をもつ画像の入出力は、画像応用システムのソフトウェアや理論の研究のサポートを行うことが、第1の目的であるが、画像自身の意味や構造などの高度な処理を行わない応用においても、広く用いられることから、プログラミングやその他のデータ入出力用としての必要性も高い。

このため、色彩画像表示は、早期に開発支援を目的として、開発することが必要である。

以上述べたもののほか、主に、開発支援用マシンに付加して用いることを目的とする、文字、音声、図形の各入出力機能を、組合せて、一つのマルチメディア用対話ステーションとした端末装置の研究開発が必要である。

ここでは、各入出力機能を、相互に関連づけ、コンパクトにまとめ、端末自身の中に、専用プロセッサを含むものが望まれる。

そして、この能力は、音声応用、画像応用の研究の進展に促して、改良されていくことが望ましい。

〔3〕 内外技術の現状

（1） 国内技術

マンマシン・コミュニケーション・システムを構成する技術のうち、漢字かな混り文を対象とする文字入出力技術は、我国に固有のものであり、欧米では、あまり例が無い。この点を除く、英数字を用いる文字入出力、音声入出力、図形・画像入出力技術は、パターン情報処理プロジェクトの研究開発により、かなり進歩した。しかしながら、マンマシンの対話には、ソフトウェアの優劣が大きく影響することと、従来から我国では、TSSシステムの利用等が遅れたために、対話という使用形態が未発達であること等の理由から、実際のソフトウェアを含むマンマシン対話用のシステムの開発が十分ではない。

ハードウェア面での開発は、最近、進歩が見られ、これに伴って、東芝、富士通、シャープ等のメーカーから、日本語ワードプロセッサが発売され始めた。これらの中には、カナ漢字変換機能等を含むものも多いが、変換精度の向上の余地があるほか、それ自身コンピュータとしての機能を持たない等の問題点を有する。また、図面等の入出力との組合せ等、今後の改良、拡張が望まれる。

音声や図形・画像の入出力についての、基礎的な研究開発は、基礎応用システムの課題グループの中の、音声応用システム、および、図形・画像応用システムの研究開発課題中で述べられているので省略する。

ここでは、音声、図形・画像については、開発支援用マシンのマンマシン・インタフェースとして用いる場合に限って述べることにする。

コンピュータ用の端末機能としての音声入出力は、最近のパーソナル・コンピュータの普及に伴い、注目されてきた。音声の合成、出力については、一部、開発、市販されるまでになりつつある。しかし、実際には、電子翻訳機等の家電品への応用が先行している。

音声入力についても、コンピュータの端末用としては、まだ十分開発が行われていない。しかし、流通業や工場のラインの制御用としては、数十語～百語程度の音声入力装置も市販されている。（日本電気、日立など）

以上のように、技術的な基礎は、ほぼ確立されているので、入力可能語数の増加や認識、精度の向上などの研究をすすめることで、コンピュータの端末用音声入出力システムとしての実用化も可能と思われる。但し、入力した音声の意味の理解等も含めた高度な処理能力を持つものは、本プロジェクトの中心課題である知的な処理システムの開発を待たねばならない。

図形・画像の入出力については、本格的なものの開発は、十分行われていない。また、色彩画家の三次元の表示等の実用的レベルの研究開発も、米国等に比べ、立ち遅れが目立っている。基礎的な技術のうちハードウェア面は、かなりの開発経験を蓄積しているが、使い易いユーザ・インタフェースを実現するソフトウェア面と、さらに、これをパーソナル・コンピュータなどと組合せる技術の面の研究開発が重要である。

以上の文字、音声、図形・画像の入出力機能を組合せて、コンピュータ用端末とするための関連技術開発が、本課題の重要テーマであろう。

(2) 海外技術

海外での研究開発の中心はアメリカであるが、アメリカでは、TSSの早い普及、コンピュータグラフィックスの研究などに端を発するインタラクティブな対話処理の普及と、そのCADシステム等への応用の一般化、高精度カラー画像を用いた航空機や宇宙船のシミュレータの研究開発など、早くから、マンマシン・コミュニケーションに重点を置くハードウェア、ソフトウェアの研究が行われてきた。また、音声の入出力についても、マイクロコンピュータ用の簡単なシステムから、より高度なものまで、研究がすすめられている。最近では、高性能のパーソナル・コンピュータに、各種のマンマシン・コミュニケーション用のインタフェースを備え、これをネットワークで接続し、オフィス・オートメーションをはじめとする今後の応用に用いようとする試みもある。

上記の研究開発のうちでは、まず、カラー・グラフィックスについては、Evans and Sutherland社のものが有名であり、各種図形の3次元表示等の能力を持つ。濃淡色彩図形の表示とアニメーションでは、NASAの開発した宇宙船のシミュレータ用のものが有名である。

音声の入出力については、マイクロコンピュータ用のものがいくつかあるが、より本格的なものは、MITで研究されているMYTALK(音声合成出力用)などがあり、専用VLSIの研究と関連づけて開発が行われている。

マンマシン・コミュニケーションの機能の優れたパーソナル・コンピュータの開発は、いくつかあり、XeroxのALTOマシン、その後のDRADOマシンが有名である。また、MITで開発されたCADRマシンも、同様の能力を有する。これらは、高精度のディスプレイと図形入力装置、キーボードを備え、きわめてよく工夫されたソフトウェアにより制御されている。DRADOでは、音声の出力機能も付加される。

このような将来のコンピュータの一つのプロトタイプに属するものの開発の試みは、我国には見られず、本プロジェクトでは、まずこのようなコンピュータを研究開発ツールとして開発する必要がある。

(4) 研究開発内容

(1) 研究開発項目

a. 漢字を含む文字入出力装置

日本語文による対話を可能とする漢字を含む日本語文の入力、表示、印刷装置を開発し、開発支援用コンピュータへ、まず、付加する。その後、文字サイズ、書体等も可変の装置を開発し、プロトタイプ・システムへ付加する。

b. 図形・画像入出力装置

まず、濃淡のないグラフや図面等の2値図形の入出力装置を開発する。解像度は、 1000×1000 点/画面程度とし、ラスタ走査方式のものを中心とする。これは、開発支援用コンピュータへ付加する。その後、より高解像度のものを目ざし、製図等の目的に使用できる表示装置や、タブレット等の座標入力装置を開発する。また、入出力の高速化、高機能化を目的とする専用VLSIを含む専用プロセッサを開発し、付加する。

濃淡を含む一般画像の入出力に関しては、画像理解システムの課題のもとでの研究のサポートを兼ねて、与えられた仕様に基き入出力装置を開発する。これは、専用プロセッサの開発等も含まれる。

c. 音声入出力装置

音声の合成・出力に関しては、現在の技術の延長上にあり、これら技術に応じた専用VLSIチップ等を開発し、開発支援用コンピュータ・システムへ付加する。これより高度の音声出力装置については、音声理解システムの課題のもとでの研究のサポートを主に行い、与えられた仕様に基き、ハードウェアの開発を行う。

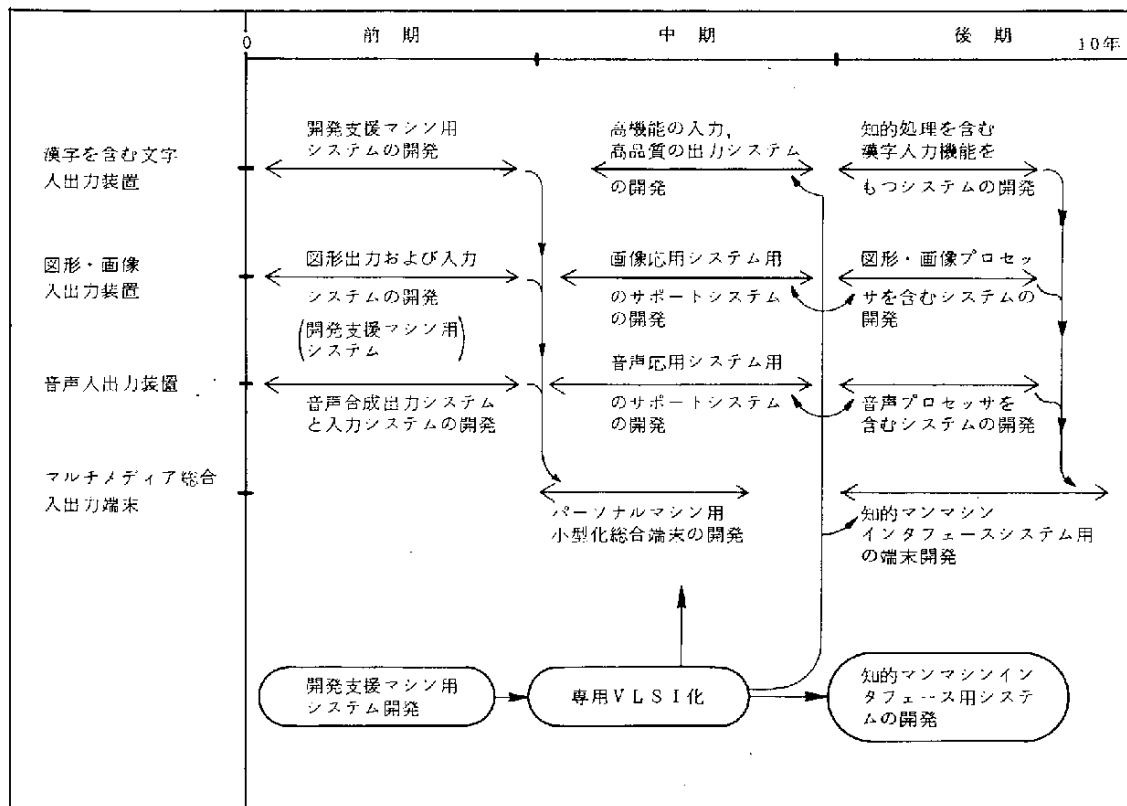
音声入力装置については、ディジタル信号処理用の専用プロセッサ等の開発を中心とするが、これは、音声理解システムの課題のもとでの研究のサポートを目的とする。

d. 文字、図形、音声の各入出力をインテグレートしたシステムの開発

これは、開発支援を目的とし、それぞれの入出力メディアごとの研究開発状況に応じた技術に基く装置を集め、コンパクトにまとめた装置である。

研究の進展に伴って、いくつかの装置を開発し、研究開発支援コンピュータを付加する。

(2) 高度マンマシン・コミュニケーション・システムの研究開発スケジュール



(3) 研究開発の目標・仕様

本課題の目標は、中期の初めまでに達成されるべき目標と、最後に第5世代コンピュータ・システムのプロトタイプに用いられるマンマシン・コミュニケーション・システムの最終目標の2つから成る。前者は、開発完了後、直ちに、開発支援用マシンの入出力システムとして実用に供されるものである。

後者は、基礎応用システムの課題グループに属する音声応用システム、図形・画像応用システムの課題に基く研究のハードウェア・サポートを行うとともに、最終的には、これら課題で開発される方式やソフトウェアと組合されて、知的インタフェース・システム用のマルチメディア入出力システムとしてまとめられるべきものである。

a. 漢字を含む入出力システム

- 漢字を含む文字3000~4000字種の入力機能、カナ漢字変換による入力とタッチメソッドによる入力の双方の機能を含む。
- グラフや線図形などとの混在を許す書類の編集能力を含むワード・プロセッシング機能。

- 音声出力による入力ミス警告機能つき。
- 4～5種類の異なる字体を含み任意に選択できる文字表示装置と、十分な速度を有する印刷装置。高品質であること。
- 外字処理の機能を持つこと。但し、オプションでよい。

上記は、中間時点までの目標である。最終目標としては、音声入力との組合せによる漢字入力機能、意味理解を含むカナ漢字変換機能等を付加することを旨とするものとする。

b. 図形・画像入出力システム

- 座標入力装置として、 $5000 \times 5000 \sim 10000 \times 10000$ 点の精度のタブレットを備える。
- 画像入力としては、 1000×1000 点程度の精度を持つ濃淡色彩図形の入力機能を目指す。
- 出力としては、 $1000 \times 1000 \sim 4000 \times 4000$ 点の解像度をもつ表示装置を備える。
- 印刷装置としては、 10000×10000 点以上のものを備える。
- 表示装置は、3次元の表示等も可能な能力を付加できるようなものとする。

上記は、中間時点までの目標であり、最終目標は、図形・画像応用システムの研究課題よりの仕様に基いた、より知的な機能を有するものとする。

c. 音声入出力システム

- 入力としては、 500×1000 語の前もって登録された単語の識別能力を有するものを目指す。
- 出力としては、 1000×3000 語の単語の合成出力が可能なものとする。

これらは、中間時点までの、開発支援を目的とした目標であり、最終目標は、音声応用システムの課題で示す。意味理解を含み、自然言語も一部可能な、より知的な仕様を持つものを目指す。

上記 a～c の機能を組合せ、VLSI 化を行い、パーソナル・コンピュータ用端末として、インテグレートしたものである。

音声、図形の入出力等のメディア別の性能は、その時点での開発技術に応じたものとし、各メディア間の相互の効果的な組合せによる対話機能の高度化を目標とする。

この中には、プログラムの音声による入力機能、図形を含む文書の編集機能、コンピュータへのコマンドの音声や図形によれ入力、また、音声、図形による応答機能など、ソフトウェア面の機能を多く含み、マンマシン・コミュニケーションの高度化を目指す。

2.5 VLSI 技術

2.5.1 VLSI アーキテクチャ

【1】 概 要

VLSI アーキテクチャとは、VLSI 素子そのものの特質を十分に考慮し、それをアーキテクチャ、方式に反映させたものである。その特質とは以下の通りである。

- (1) VLSI では、通信コスト (Communication Cost) の方が計算コスト (Computation Cost) をはるかに上回る。
- (2) 少ないタイプの単純なセルによりチップを構成する方がチップ面積使用効率が良い。
- (3) 例えば、バス構造のような単純・相対構成を取るより、ローカルな通信が多い構造の方がチップ面積使用効率も良く、速度も上がる。
- (4) データ入出力時間をも最大限利用するには、パイプラインとマルチプロセッシング混合アプローチなどが優れている。

また、VLSI アーキテクチャが従来のアーキテクチャと異なると思われる点を列挙すると以下のようになる。

- (1) VLSI アーキテクチャでは、アーキテクチャの設計手法が従来に比べより重要視される。従って、何を作るかということに合わせて、どのように作るのかが同時に決定されなければならない。しかし、VLSI アーキテクチャ構築法において有効な手段が提供された場合には、従来、製作する数に比べて設計時間がかかり過ぎるため、実現不可能とされていたようなアーキテクチャのVLSI化が可能となる。(少量多品種のチップの現実的可能性)
- (2) VLSI アルゴリズムとは、従来マルチチップにしていたために考慮しなければならなかった。アルゴリズムの本筋から離れた非本質的な部分(オーバーヘッド)をなくすことによって、革新的飛躍的に性能の向上を期待するものである。従って、同じ機能を実現することにしても全く異った手法を取る方が良いと思われる。
- (3) 設計に於ける階層、PMS構造→ISP→RT→LogicレベルにおいてLogicレベルコンポーネントが従来は与えられていた(限定したものしか無かった)ことに比べ、VLSI アーキテクチャ・アプローチではかなり自由度が高く、自由に階層のレベルが選べるという点と非常にスケールの大きいLogicレベルコンポーネントを使うことによって、設計における階層構造が扱われる。つまりアーキテクチャアルゴリズムが直接Logicレベルコンポーネント上で、実現できるようになったことが従来の素子技術では実現可能性が全く無かったようなアーキテクチャアルゴリズムを今一度検討し直す必要性を要求する。

ここでは、このような新しい可能性を持った革新的VLSI アーキテクチャ構築の方

法に対していくつかのアプローチを示す。

【2】 V L S I アーキテクチャ・アプローチの方法

ここでは V L S I 構築のアプローチを以下の三つに分ける。

(1) 応用目的指向型アーキテクチャ

前述した V L S I の特質を考慮しつつ、応用問題のアルゴリズムを構築する。例えば、V L S I の特質の一つに、少ないタイプの単純なセルによりチップを構成する方がチップの面積使用効率が良いということがあるため、まずいくつかの上記条件を満たすセルを与え、応用問題を解法できるアルゴリズムを考案する。例えば(i) C M U の Kung らはシストリック・アレイという非常に単純なセルを考え、この上で各種応用問題の構築を行っている(実例の項を見よ)が、これなどはその一例である。このアプローチでは V L S I アーキテクチャ構築成功の鍵が、必ずしもトップダウンにアーキテクチャを決定することではないことを示している。実際問題、V L S I アーキテクチャ構築において最もむずかしいのは、マスクパターンのレイアウトである。従って、マスクパターン・レイアウトをまず始めに決めてしまい、それからアーキテクチャの構築を始めると言うことが重要となろう。(ii) マスクパターン・レイアウトをよりやり易くすることを逆にデバイス側に要求することもできる。一例として P L A を挙げる。P L A はその規則的構造ゆえ V L S I 向きの構造であるが、かつ設計の方法も規則的となり、デザインオートメーション化がし易い。P L A コンパイラのようなものを作成することにより、組み合わせ回路、順序回路の設計がプログラミング言語で行うことも可能となり、論理式より構造(マスクパターン)が生成されることとなる。(iii) 別の例として、Bristle Block 設計法を挙げる。この方法では、マスクパターン・レイアウトのし易さを考慮し、まず始めにバス構造とブロックの種類を与え、そのブロック間を自由に接続し、任意のアーキテクチャを構築する。

(2) 汎用指向型アーキテクチャ

V L S I においても大量生産により著しくコストダウンが図れるという点は従来と変わらない。従って、汎用型アーキテクチャを指向し、多くのアプリケーションに適用するようなチップを作るアプローチも依然として重要と思われる。しかし、集積密度の増大に伴い機能的には従来のものと比べ高いため、汎用のレベルを再検討する必要がある、これは大きな課題である。

(i) ダイナミック・アーキテクチャ

集積密度の増大に伴う機能向上により、ダイナミック・マイクロプログラムが可能となり、従来の汎用計算機の汎用とは I S P アーキテクチャが固定で、それ以上のレベルで汎用ということであったのが、マイクロアーキテクチャ・レベルで汎用とし、それ以上で可変性を持たせることが可能となる。ここで重要なことは、V L S I 1 チップ U H P

(Universal Host Processor) 構成法および UHP を I S P レベルで問題に適応させる適応化理論の二つである。UHP については、従来理論的に可能であり、方式上の問題点が解明されているにもかかわらず、素子数や速度的に不可能であったものが V L S I により可能となる。この中には例えば、パレルシフタ、ビットアドレッシング機能、高度なレジデュアル制御などがあり、又、これらの機能の実現により、マイクロプログラムの最適化手法、マイクロプログラム開発の実現により、マイクロプログラムの最適化手法、マイクロプログラム開発高級言語などの再検討が要求されることも明らかである。適応化理論とは、従来の I S P 機械語レベルより上のレベルに中間言語を設定し、問題解決を行う場合、どのようにして問題に適応した I S P を決定していくかである。開発の現状としては、高級言語を限定し、その言語に適した I S P を決めるという高級言語計算機のアプローチが現在のところ一般的である。タグやスタックの概念などが重要であることは判っているが、どのように 1 チップ化して行くかマイクロプログラムでどのように制御するかなどのテーマが重要である。また、今後は言語レベルよりもさらに応用問題に密着したレベルでの適応 I S P の決定法、問題の性質を自動的に解析し、計算機自らが適応していくという自動チューニングの手法なども重要となろう。

(iii) 新 (New) ノイマン型完全 1 チップ・アーキテクチャ

現在作られているメインフレームを 1 チップ化するという方向は現在各所で試みられているが (1 チップ 3 7 0 など)、この手法は過渡的な方法としては評価できる。即ち、過去に蓄積のあるソフトウェア財産の多くを利用することができるからである。しかしこの方法はあくまで過渡的なものであり、第 5 世代コンピュータが実現されるといわれている 1 9 9 0 年代における重要性は高くはないと思われる。それよりも、現在広く行き渡っているノイマン型ベースではあるが、その欠点をなくしたときでの V L S I 1 チップ = New ノイマン型の研究を重視すべきであろう。(詳しくは 2.2.6 新 (New) ノイマン型アーキテクチャ 参照)

(iii) メモリ

メモリ・アーキテクチャにおいては、V L S I 化により大規模メモリの開発が可能となるため、ロジカルレベルでは、巨大論理空間のアクセス法 (例えば、3 2 ビットをアドレス指標とした場合、仮想空間として 4 G がアクセスできる。)の開発が重要となる。他方、ハードウェア的には、メモリ読み出し幅の増大マルチポートメモリなどの実現が可能となる。また、F I F O、キュー、連想処理などの機能をメモリに付加することが可能となる。いわゆる機能メモリの実現が可能となり、これらは、第 5 世代コンピュータ構築において重要な要素コンポーネントとなるであろう。

(3) V L S I マルチチップ

これは、V L S I を組み合わせて、さらに大きな一つのプロセッサを構築する場合のアプ

ローチ法である。即ちVLSIで1チップ構成が可能となってもさらにVLSIを組み合わせて超高速処理をねらう必要があるからである。問題としては、ゲート数に対して必要な接続ピン数をどのように用意するかと、その間をどう接続するかである。

【3】 内外技術の現状

VLSIアーキテクチャはまだ始まったばかりである。よって、世界レベルでその研究は盛んに行われようとしている。しかるに、特に重要と思われる研究開発の多くは米国に於て行われている。我が国では、VLSI製造工程における研究で一部行われているものもあるが、最も重要と思われるアーキテクチャ・CADにおいては十分な研究が行われているとは思われない。VLSIは今後の計算機構築に当っては必要不可欠なものであると考えると、現在のままでは1990年を待たずとも、米国からの技術導入を行わなければならない事は必須であり、研究開始は緊急を要すると考えられる。以下、いくつかの重要と思われる研究につきコメントする。

(1) CMUのKungのSystolic アルゴリズム

KungのVLSIアーキテクチャ構築法では、まず図2.5-1に示すようなセルをつなぐ方法をいくつか用意する。次に、解こうとするアルゴリズムを実現するためにはこのセルではどのような計算をする必要があるかと、データの入出力法を決めていく(これがアルゴリズムとなる)。セルの接続法は、もともとVLSIの特質を最大に利用するように決めてあるため、アルゴリズムが決まるとこれがVLSIにとっては最良のものとなる。Kungらはこの方法により、マトリクス計算、FFTを始め、データベース・マシンのコンポーネントに渡る広範囲なVLSIアルゴリズムの構築研究を行っている。

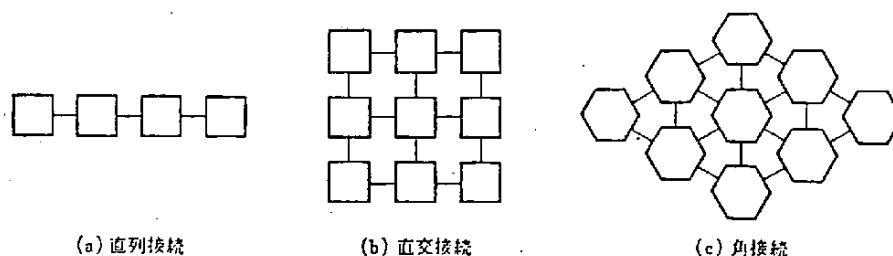


図2.5-1 “心臓”アレイ(可能な接続の例)

(2) MITの知的CAD

MITの人工知能研究所では、知的CADの研究を行っている。これは、目的指向型のVLSIチップをつくりだすための基礎研究であるが、設計ルール・ノウハウを知らなくても設計ができる事をねらいとしている。既に、この手法の一部を使いLISP専用LSIチップを数か月でつくりだすことに成功している。

(3) 32ビット・プロセッサ

1981年2月に開催されたISSCC (International Solid-State Circuits Conference) , COMPCONなどで発表されたVLSIプロセッサを表2.5-1にまとめる。集積密度としては、HP社の45万トランジスタ/チップVLSIが注目される。また、ベル研究所のCADは非常に強力で、CADデータ・ベースのデータ利用により、設計パラメータの変更に対してもある程度はのりきる事ができる。表2.5-1に示すVLSIはすべてCADで作られていることも注目すべきで、ほとんど数か月のオーダーでチップが完成している。

表2.5-1 VLSI 32ビットマイクロ・プロセッサ開発の現状 (1981年2月)

	HP	BELL MAC 32	INTEL APX 432	NS 16000
チップ構成	CPU	CPU	GDP 43201 43202 IP 43203	CPU 16032 MEMORY MANAGE 16082
構成素子数	45万	10万	11万 (43201) 4.9万 (43202) 6万 (43203)	
プロセス	2.5 μ m NMOS ↓ 1.5 μ m NMOS	3.5 μ m CMOS ↓ 1.5 μ m CMOS	HMOS	
マイクロサイクル	55 ns	31 ns	125 ns	100 ns
マイクロROM	9 K $\times$ 38bit		4 K $\times$ 16 bit (GDP) 2 K $\times$ 16 bit (IP)	1.3 K $\times$ 18 bit
アーキテクチャ	スタックマシン	ノイマン型 高級言語指向	ADAマシン (スタック)	ノイマン型 高級言語指向
高信頼性化	セルフテスト		チェッカーモード	
その他	DEBUGGING AID	CAD DATABASE	CAD	

※ 以下の項目は全社のものが満たしている。

- (1) パイプライン制御 (2) 高級言語サポート (3) OSサポート (4) バーチャル・メモリ

【4】VLSIアーキテクチャ研究開発内容

(I) 概 要

① 新アーキテクチャ構築技法（基礎研究）

前述したようにVLSIアーキテクチャにおいて、アーキテクチャの設計の方法を変えるということが重要であり、その方法論を構築する研究を早急に始める必要がある。具体的には、

(i) VLSIデザインにおいても最も困難とされるマスクパターンを容易にするため、パターンにある程度枠組を加え、論理設計を進めるというような、いわゆるトップダウンでもボトムアップでもない設計手法の開発

(ii) VLSIアルゴリズム

VLSIの特質をアルゴリズム解法のレベルより考慮するような手法の開発

(iii) 計算機支援を前提とした開発手法

② VLSIアーキテクチャ

①にもとづく具体的なアーキテクチャ。これは、大別して1チップVLSIアーキテクチャと計算機構成パーツとしてのVLSIアーキテクチャがある。

③ VLSIチップシステム構成法

個々に作られたスペシャルチップ（構成パーツも含む）のチップ間を接続して、システムを構成する技法。これは機能分散アーキテクチャの基礎的研究にも相当するが、どのように機能分散させるのか（チップ化していくか）を広いレベルで決定するアルゴリズムと具体的なチップ構成法を明らかにすることが重要である。

(2) 開発項目と必要な機能

開発概要にもとづき、項目を列挙すれば、以下のとおり

① VLSI基礎研究

(i) VLSIデザイン・ルール・ブック

新アーキテクチャ開発設計法をルール・ブックとしてまとめたもの。セルフ・スタンディング法により作られている必要があり、関係学部を卒業した学卒の人間が6か月以内にマスターできる（実習を含む）必要がある。

(ii) 設計Q・Aシステム

設計に就いてのQ・Aを計算機を介して行うシステム。解は、始め専門家が入れるが、二回目からは、システムに用意されるQ・Aデータベースにより自動回答される。またこれは、将来の知的CADのためにも使われる。

(iii) アーキテクチャ・データベース

応用別のコンピュータをVLSIを使って作るためのデータベース、アーキテクチャ

の項目分類がなされており、応用指定により最適アーキテクチャの指定が行われる機能が必要である。(なお2.2.6新(New)ノイマン・アーキテクチャの項を参照)

(iv) VLSIアーキテクチャのためのCAD

アーキテクチャ記述言語, VLSIマスク・デザイン言語を備えている事が必要。さらにアーキテクチャ・データベース, 設計Q・Aシステム, 製造用CADと連結する機能が必要である。

② VLSIアーキテクチャ

百万トランジスタ/チップ・アーキテクチャ, 千万トランジスタ/チップ・アーキテクチャは, それぞれ別に考える必要がある。開発項目としては,

(v) 完全1チップ・アーキテクチャ

(vi) 機能パーツ・アーキテクチャ

の二つがある。両者とも必要な機能をどのように入れるのかの他, 自己診断機能や, 自己故障修復機能を備えることは必要である。

③ VLSIシステム

VLSIチップをつなぎ合わせて作られたシステムで, これには色々なものが考えられるが, 知的マシン・アーキテクチャが第5世代の目的からしても最も適当である。

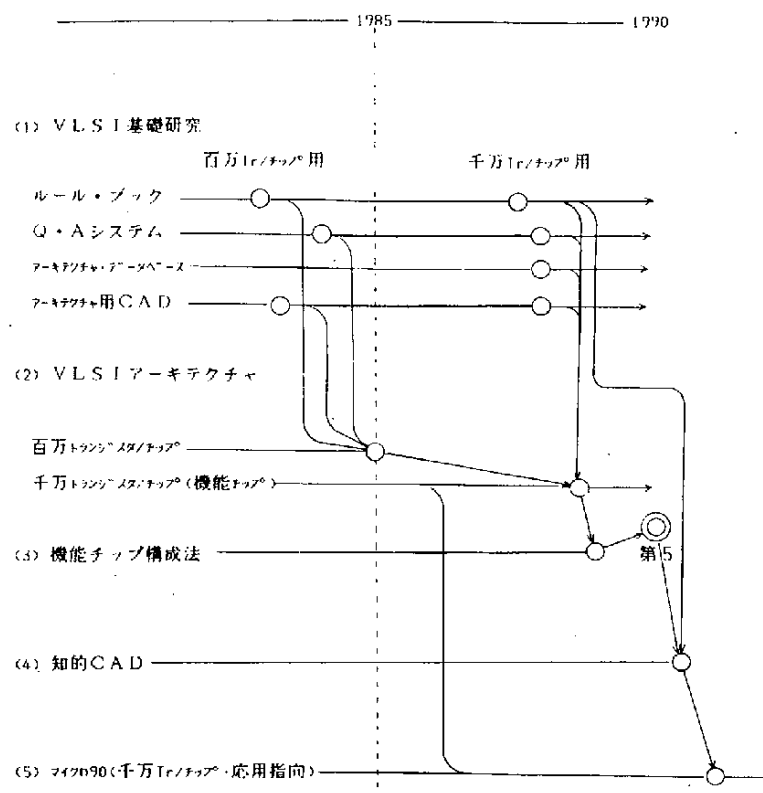
(vii) 機能分割, 接続法

そのための基礎研究で(vii)と密接な関係がある。

(3) 第5世代での位置付けとストーリー

第5世代プロジェクトでの開発ストーリーは, 以下のようになる。

まず, 基礎研究は早急に行う必要があり, 特にルール・ブックはプロジェクト開始後一年以内に完成すべきである。その他の基礎研究は, まず百万トランジスタ/チップに焦点を絞った後は, 千万へと第5世代プロジェクト全般に渡り継続させるべきである。次に, アーキテクチャではまず百万トランジスタ/チップに焦点を絞り, これの中身はNewノイマン・アーキテクチャとし, 開発システムとする。さらに千万版を構築するが, これはまず第1に第5本体(スーパー・ロジック・マシン)のためとすべきである。そして, そのためのチップ接続法が重要となる。また, 第2にアーキテクチャ・データベースを使い, 千万版の応用指向チップ開発(マイクロ90)を行う。



2.5.2 知的VLSI-CAD

【1】 概要

今後10年以内にVLSIの規模は論理系で数10万ゲートに達することが予想される。この状況はちょうど数10メガバイト付きコンピュータを与えられたプログラマに類似している。プログラマ達はそれに対して、アセンブラ言語、高水準言語、ソフトウェア工学に基づくプログラミング技術等の開発を行って進歩してきた。これと同様にVLSIの集積度が今日のように加速度的に進歩すると、新たにハードウェア危機あるいはVLSI危機というものが出現する可能性がある。ソフトウェア危機以上に困るのは、対象がハードである点で、VLSIの設計ミスはソフトウェアのバグと同様に考えることができない所に問題の重大性が潜んでいる。この困難を解決するのがVLSI-CAD技術であり、その理想はVLSIの機能、性能、構成を実際にチップを作成しないで、全く相似形でソフトウェア・シミュレーションによって実現することといえる。これが可能になれば、VLSI危機はソフトウェア危機に転化され、コンピュータ・アーキテクトや論理回路設計者は、もはや製造工数を考える必要がなく失敗を恐れず新しいVLSIの設計が可能になる。特に新しいアーキテクチャの実験評価としてVLSI-CADは必須の道具であり、ゲート数、タイミング、性能などの情報がVLSIを製造しなくてもプログラムによって可能になり、コンピュータ・アーキテクチャの進歩に非常に寄与するであろう。従来、大学や研究所では提案するだけで終わっていたシステムが現実味をおびてくる

ことも予想される。

VLSI-CADは基本的にはこれ迄の装置設計とLSI設計技術の総合化であり、従来の設計手法と全く異なるわけではない。しかし大規模論理回路が対象であるから階層化、モジュール化した設計手法が必要である。また論理記述も高レベル記述言語が使われる傾向がある。これは大規模ソフトウェアの設計と非常に類似している。データベースも重要な要素である。

VLSI-CADのデータベースは、CAD工程の一貫管理に不可欠であり、CADのオンライン化、TSS化とも密接に関係する。データベースの情報には、論理データ等のデジタル情報だけでなく、回路図やパターン情報等を扱える図形データベースも必要である。また知的CADという面からみると、アーキテクチャ・データベースと呼ばれるものに各種の設計情報をライブラリ化することが重要である。以上のような総合的一貫システムを効率よく運用するためには、設計環境の整備が必要になる。設計し易い環境としては、オンラインTSSで端末として高性能グラフィック・ターミナルがパーソナルに使えるのが望ましい。CADの普及については、CADのインタフェース仕様の標準化、特に記述言語の標準化が必要となる。さらにネットワークによるサービスも必要である。VLSI-CADでの別の問題として、わずかな外部インタフェースを介して巨大なブラック・ボックスを閉じ込めることによる故障診断技術がある。このためにテスト容易化設計法を、アーキテクチャとも関係して研究する必要がある。

VLSI-CADは単にVLSI設計支援にとどまらず、コンピュータ・アーキテクチャと一体であるという考え方もある。実際、もしコンピュータ記述言語の研究が進み、コンピュータ・アーキテクチャのデータベースが蓄積されれば、コンピュータは内包的に自己のアーキテクチャおよびVLSI-CADを保有することが可能になり、アーキテクチャのダイナミック性や自己増殖といった概念と関係して来る。VLSI-CADを能動的に把えることは興味深い問題であろう。

VLSI-CADの研究は多くの問題をかかえており、それらは簡単に普遍的な解が見つかるものではなく、デバイスの進歩、設計の規模にあわせて少しずつ変化しており、あくまでも第1の目的は、最小のマンパワーで、短期的に品質のよいVLSI設計を可能にし、生産性を向上することである。

【2】 必要性

VLSI製造における工数は、数10万ゲートで100人年以上を必要としており、今後とも増大傾向をもっている。図2.5-2は現在のLSI設計のフローを示しているが各工程は一貫しておらず、運用を複雑化している。また論理シミュレーションや配置・配線の入力データ作成は回路規模に指数的に比例して膨大な数になっており、ケアレスミスも増大して行く傾向にある。従って工程を一貫管理し、入力データの自動化、高水準化を確立する必要がある。

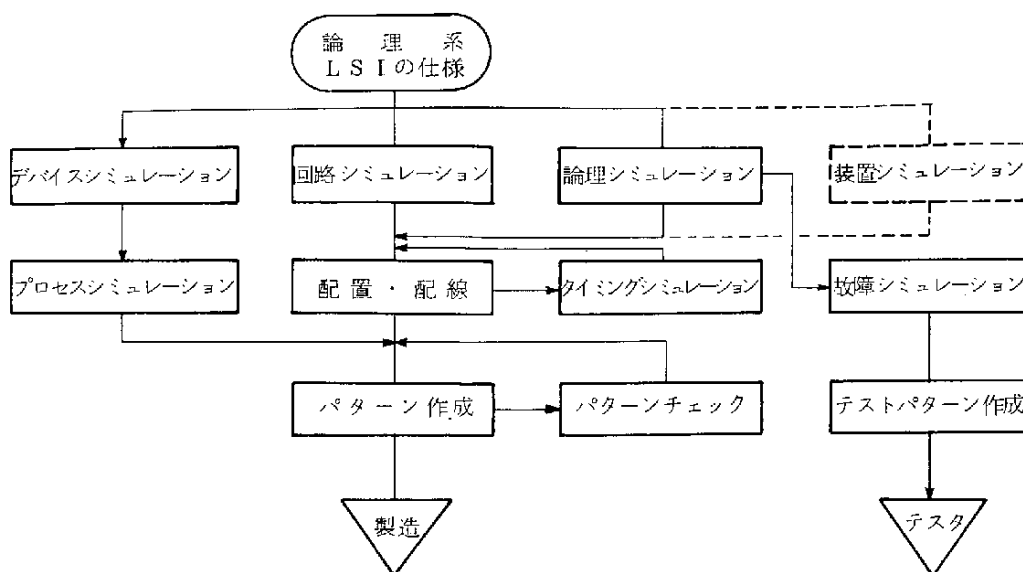


図 2.5 - 2 論理系 L S I の設計フロー

多品種少量，量産品とも V L S I 化の傾向が増大するが，多様化するニーズに対しては生産性向上策を考慮しなければ，V L S I 危機の可能性があり，設計技術者の生産性を高める V L S I - C A D が必要である。このための C A D としては，使い易さが大切な条件である。

現在 C A D は各メーカ内部で開発されており，ユーザの仕様に基づいて V L S I を製造することは，1 部のマスタスライスを除いて商業化されていない。しかし，V L S I 化の要求は，コンピュータ・アーキテクチャに関係する者にとっては強く，特に大学や研究所で顕著である。このような外部の者が共通に利用できる V L S I - C A D は今後商業的にも出現しようが，そのためには，利用言語の設定やインタフェースを統一して，C A D の標準化作業が必要である。

C A D の歴史をみるとそれはその時代の素子の技術，実装技術に依存しているのがわかる。集積度が拡大したり，素子が変われば，従来の C A D を利用することは不可能になっている。これは C A D の進歩に較べて，デバイスの進歩が急激だったからで，なるべく素子に依存しないで，V L S I 技術の進歩に耐えられる C A D を開発する必要がある。

V L S I - C A D は単にチップ製造の道具ではなく，アーキテクチャ評価のために非常に重要な技術である。第 5 世代のアーキテクチャとして多くの新アーキテクチャが提案されているが（「2 新アーキテクチャ」参照），それらのシミュレーションを V L S I - C A D で実現可能ならばアーキテクチャの進歩に貢献しよう。このためには，ハードウェアのプロトタイプ・シミュレーションと同等機能をシミュレートできる C A D を開発する必要がある。

【3】 内外技術の現状

(I) 国内技術

(i) VLSI 論理設計

従来、ゲートレベル、パッケージレベル、装置レベルと別々に作成されていたが、最近のLSI規模の拡大により、それらを統合し、混合して使用できるシステムが進んでいる。既に大型コンピュータの設計は回路規模では数10万ゲートになっており、論理記述言語の開発は各メーカーともRT(レジスタ・トランスファ)レベルで行っている。ゲートレベルは、トランスレータで行うもの、同一言語で記述できるもの、別の言語で記述するもの等さまざまである。ただし、ここで使用されるLSIは数百～数千ゲートが中心で、このレベルではハードウェア・シミュレーションを必要としない程進歩しているが、VLSIに対しては、使用する計算機の記憶容量、処理時間から問題が多い。上記混合型はあくまでもデジタル・システムの設計支援であり、VLSI-CADでの階層的アプローチはまだ研究段階である。

(ii) VLSI 実装設計

この部分のCADは配置配線、パターン・チェックがある。これらは使用する用途により各種の設計手法がある(図2.5-3)。LSIレベルでは比較的CADが自動化されているマスタスライスやビルディング・ブロックが多用されているが、VLSIになるにはチップ面積の関係からカスタム化に向わざるを得なく、この方向のCADは余り進んでいない。パターン図と論理回路図のチェックはかなり進んでおり、パターン図上での設計ルール of 自動チェックやパターン図から回路図を出力し、接続チェックを行うものも開発されている。

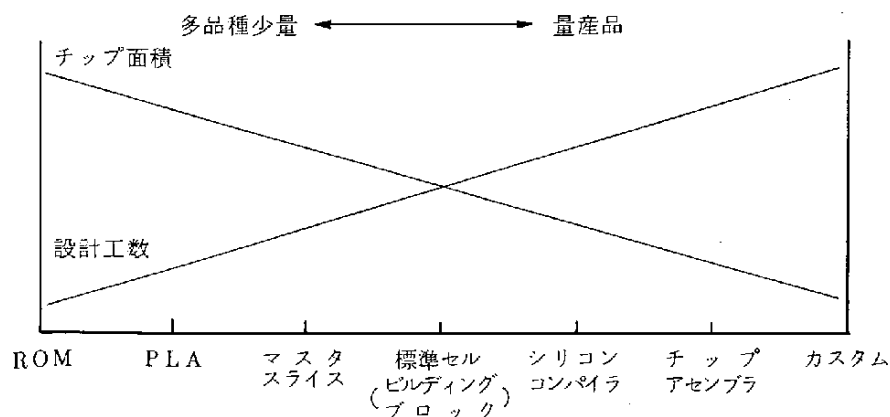


図 2.5 - 3 L S I 設計方法と特徴

・(iii) テスト容易化設計法

このアプローチには、テストパターン生成プログラムを強化することやLSSD (Level Sensitive Scan Design) が現在採用されている。LSSDは数%のオーバヘッドでスキャンパスを組込んでいる。しかしMOS系LSIでは、効率が悪い。VLSIでは両方向以外のアプローチが必要であろうという認識もある。

(iv) CAD使用環境

CAD運用の指令はTSSベースにより行われている。ホストは大型機が多い。レイアウト・パターンの修正は高精度のグラフィック・ディスプレイでインタラクティブに行っている。データベースを介したCAD工程の一貫化はLSIレベルではかなり進んでおり、回路設計、実装設計以降はほとんど統合されている。VLSIに対しては、現在開発中のものが多い。

(2) 海外技術

米国においては先駆的研究が多い。米国のCAD推進者は、コンピュータ・メーカだけではなく、大学、研究所、CADメーカでも活発に進められている。特に大学では盛んで、IC製造設備を有したり、ネットワークでメーカと直結している。図2.5-4にCAD開発状況を示す。その中で我国では余り試みられていないものとして、チップアセンブラやシリコンコンパイラによるVLSIチップ構築法、MITで行われている回路情報を知識ベース化したものがある。これらの技術は、VLSI-CADにおいて必須のものと予想され日本での遅れがめだつ。

その他にはテスト容易化設計法としてLSSDがIBMのS/38のLSIに使われて、99%の故障検出率を得ている。LSI工程の統合化はベル研究所で報告されており、35Kトランジスタ回路を1人年に完成している。知識情報処理を設計の初期の段階から製造部分まで導入しようとする知的CADシステムがMITの人工知能研究所で試みられている。

〔レベル〕 〔階層〕 〔言語〕 〔シミュレーション〕 〔VLSI設計手法〕

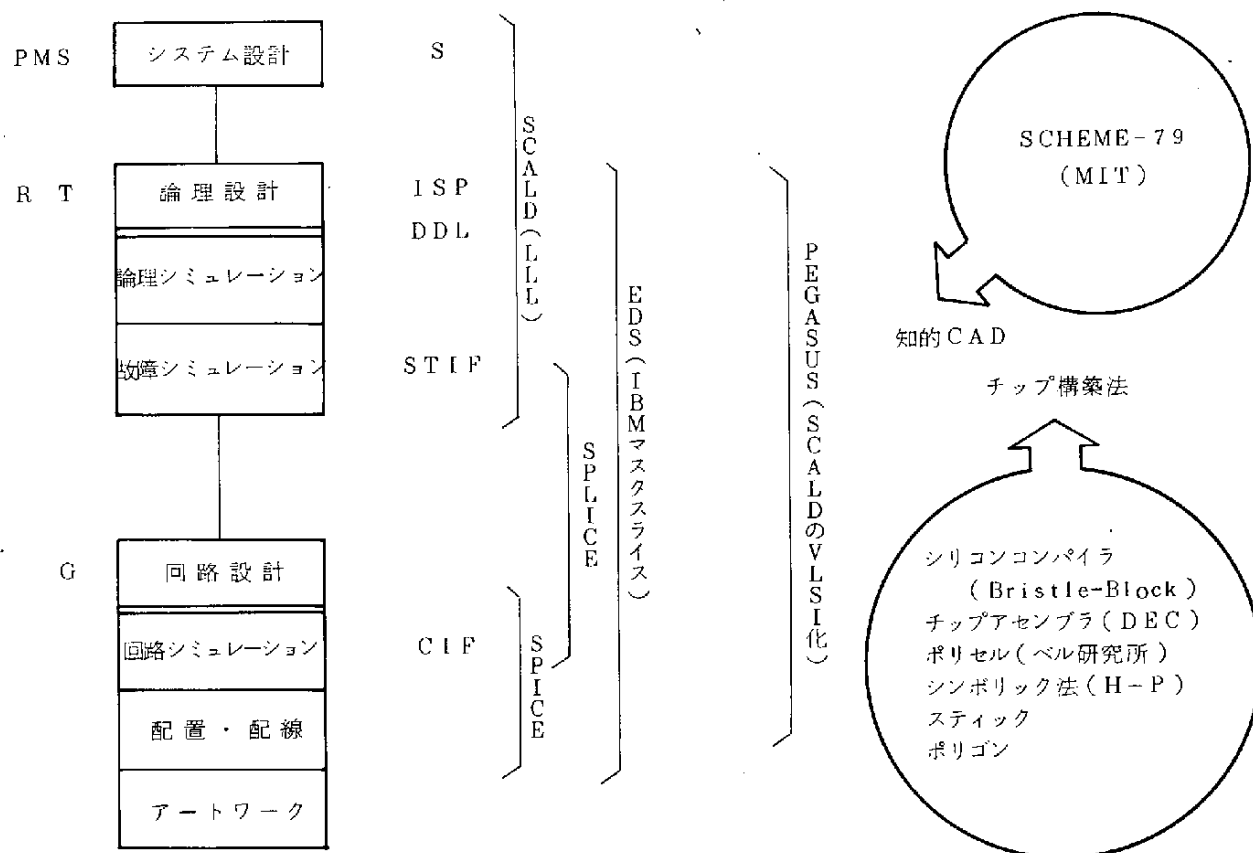


図 2.5 - 4 米国における VLSI - CAD 開発状況

【4】 研究開発内容

(1) 研究開発項目

① CAD 標準言語

CAD 資産の共同利用を可能にするため CAD 言語の標準化を行う。

CAD 言語はシステム・レベル→レジスタ・トランスファ・レベル→ゲート・レベル→パターン・レベルと階層化され、それぞれのレベル毎に標準言語を設定し、高位レベルは低位レベルの言語にコンパイルできるものとする。従って、ユーザが任意のレベルで記述したものが、標準インターフェースで混合できるが、ソフトウェアの言語と同様に、高位レベルの言語サポートを強化し、デバイス独立な CAD をめざす。各種のデバイス特性に適

応するためには、デバイス従統部は低位レベルで記述し、高位レベルとリンクすることによって汎用性をもたせる。

② CAD用統合データベース（アーキテクチャ・データベース）

VLSI開発を統括し、各工程を一貫管理するため統合データベースを開発する。

VLSI-CADに要求されるデータベースは、多数の端末から共有データベースに会話型でアクセスされること、記憶情報がCADの各レベルに対応して階層化されていること、データベース変更の頻度が多いこと、数値情報以外に図形情報をもっていること等の特徴がある。

③ テスト容易化設計法

ピン数制限下でのVLSIテストは大きな問題であり、解決手段としてVLSIに適したLSSD方式、回路を2重化、3重化したテスト方式、チップ内に故障検出診断回路を組込むBIT（Built in test）、BID（Built in diagnostics）方式等を開発する。

④ CAD専用ハードウェア

VLSI-CADを高速に実行するハードウェアを開発する。このアーキテクチャは①項でのCAD言語を高速に処理するものであり、新アーキテクチャで開発される技術に応用する。技術課題としては、高性能シミュレーション・マシン、高性能図形プロセッサがある。②項のCAD用データベースを高速化するためにデータベース・マシンを開発する。図形情報用に光ディスク等の大容量ファイルを開発する。設計情報の入出力装置としては、高性能カラー・グラフィック・ディスプレイ、高速グラフィック・プリンタ、図面読取り装置等を開発する。

CAD有効利用のためにネットワークによる会話型処理を開発する。開発要素としては光ファイバーを用いた高速データ・ハイウェイ、ローカル・ネットワーク、ローカル・データベースと高性能グラフィック・ディスプレイを有するパーソナル・ステーション等がある。

⑤ 知的CADシステム

VLSI-CADの生産性を向上させるため以下の各項目を開発する。

(i) 高度なアルゴリズム

CADの自動化を進めるため、テスト・パターン自動発生アルゴリズム、各種アルゴリズムを組合せ、最短の配線長、最小の面積解を与える発見的プログラミング技法、パターンを設計ルールに従って検査し、パターン図面から回路図に逆変換し、接続チェックを行う図形理解検査システム等を開発する。これらはライブラリ化して②のデータベースに格納する。

(ii) VLSIアーキテクチャ評価用CAD

①で開発した標準言語で記述されたVLSIアーキテクチャを評価するため、各種設計手法によるゲート数、タイミング、チップ面積、ピン数、処理能力を概算し、VLSI化の適否を判断できるようなシステムを開発する。

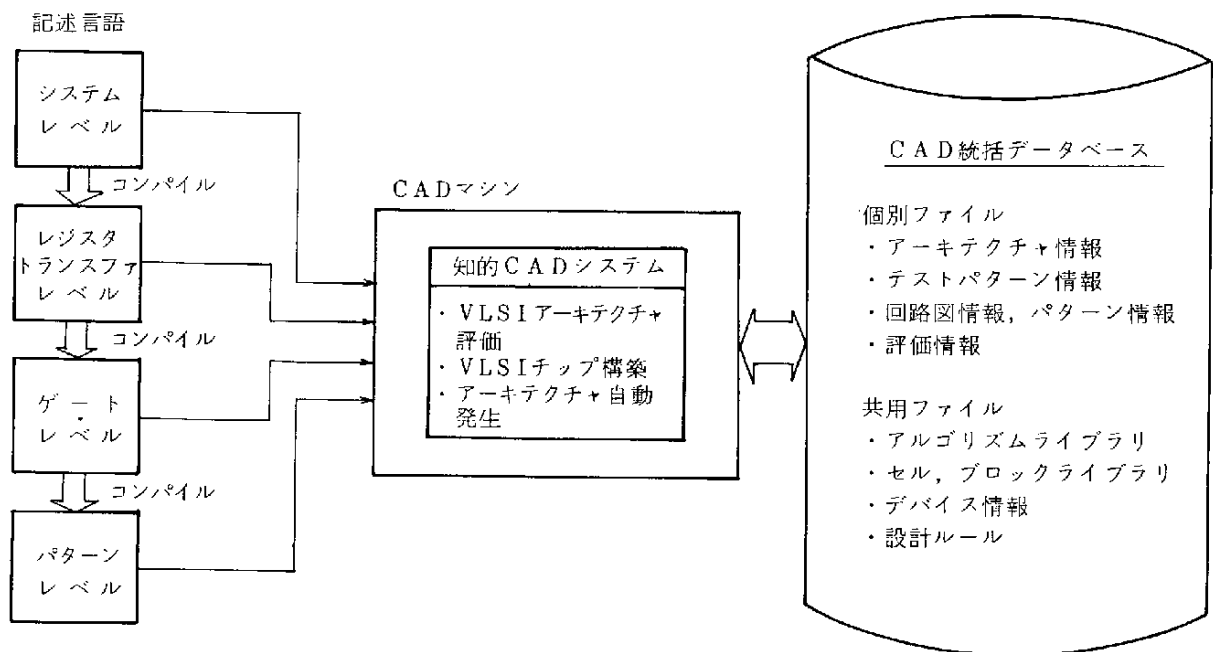


図 2.5 - 5 知的CADの概念図

(2) 研究開発スケジュール

開 発 項 目	1	2	3	4	5	6	7	8	9	10
標 準 言 語	V. 1			V. 2			V. 3			
C A D用 D B M S	基本設計		高水準 Query, 図形 DB			大規模高速化				
テスト容易化設計法	L S S D検討		設計サポート							
				B I T, B I D検討			設計サポート			
C A D専用ハードウェア	方式検討		第 1 次試作		多重化, D M B		第 2 次試作			
知的 C A D システム										
・アルゴリズム	第 1 期開発			第 2 期開発			第 3 期開発			
・V L S Iアーキテクチャ評価	V. 1 設計			V. 1 運用						
							V. 2 設計		V. 2 運用	
・V L S Iチップ構築	階層設計基本システム			V L S I化 V. 1 (シリコンコンパイル, 回路合成)			V L S I化 V. 2 (自動化推進)			
・アーキテクチャ自動発生				基 礎 検 討			V. 1		V. 2	

(3) 研究開発の目標・仕様

- ① 対象とするVLSIは論理系で数10万ゲート、記憶系でその10倍位を想定して開発する。
- ② CADマシンの性能は、処理能力：数100～数1000MIPS，記憶容量：数100MB/ユーザ，転送能力：数100MB/secを目標とする。
- ③ カラー・グラフィックの性能は、画素数：4096×4096，ディスプレイ速度：10万ベクトル/sec，転送速度：数10MB/secを目標とする。
- ④ 設計工数として、数人年/10万ゲートを目標とする。
- ⑤ 量産VLSI，カスタムVLSIのいずれにも適用できることとする。
- ⑥ VLSIだけでなく、大型コンピュータ等の大規模システムの設計にも使えることとする。
- ⑦ 使用するハードウェア（計算機，I/O）に従属しないものを開発し，広く流用できるものとする。
- ⑧ 新しい設計手法，新しいデバイス技術に迅速に対応できるオープン・エンデッドなシステムとする。
- ⑨ 設計技術者がCADをブラックボックスと考え，必要な情報を人間に分りやすい形式で入力できるようなシステムとする。
- ⑩ 言語，データ形式，入出力インタフェース，ネットワーク・プロトコル等を標準化し，アーキテクチャの情報交換を促進し，VLSIアーキテクチャの同時分散的开发を可能にする。

2.6 システム化技術

2.6.1 知的プログラミング・システム

〔1〕 概 要

プログラミング技術の過去における発展を見ると、機械語から始まって、アセンブラ言語、コンパイラ言語が発明され、プログラミングはその度に飛躍的に技術の向上を見た。

しかしながら、ソフトウェアはハードウェアに比べて技術の進展が遅いことも事実であり、ソフトウェア危機は潜在的に広まっているという見方が強い。その一番の原因は、プログラミングが非常に高度な作業であり、しかもはっきりとした方法論がないことである。すなわち、プログラミングは、現在は未だ工芸的（art的）段階に止まっているわけである。今後ハードウェアがますます進歩し、それをソフトウェアで十分に利用し切れなくなるという事態を回避するためには、プログラミングを工学的段階にまで高める必要がある。

工芸的な段階から工学的な段階に進むためには、プログラムの機能についての考察が必要となる。とくに、プログラムの自動作製システムを実現するためには、プログラムを構成する部品すなわちプログラミング言語の各言葉が表わす機能をコンピュータで処理できなければならない。これは、従来研究されてきたプログラミング言語のシンタックス的、あるいはプログラマティックス的側面の問題とは異なるセマンティックス的側面の問題である。

工学的手法の基礎は、モジュラー・プログラミング、すなわちモジュールの組合せによるソフトウェアの製造技術である。モジュラー・プログラミングを支える技術は、2つある。第1は機能モジュールとそれに基づく組合せ技術で、第2はデータ抽象化によるモジュール化と、それに基づく組合せ技術である。前者の組合せ方は、部品を横方向に結合して新しいソフトウェアを組立てていくので、水平型組合せ技術と呼ぶことにする。後者の組合せ方は、抽象データ型を操作する抽象機械の上で動くプログラムと、抽象機械を実現するプログラム群とを垂直方向に組合せてソフトウェアを組立てて行くので、垂直型組合せ技術と呼ぶことにする。

この直交する2つの組合せ技術は、プログラムの正当性の証明およびプログラムの自動合成のための基礎技術となっている。水平型組合せ技術によって、合成されたモジュールがどのような機能となるかを容易に知ることができる。また垂直型の組合せ技術は、モジュール分割技術とも考えられ、プログラムの正当性の証明は、抽象機械上のプログラムと抽象機械を実現するプログラム群とに分けてできるので、その規模が小さくなり、証明は著しく容易になる。

プログラムの自動合成は証明の逆過程とも考えられ、証明が容易になるにつれ合成も容易になる。

水平型組合せ技術は、組合せて得られたモジュールの機能が構成部品の機能と接続方法によってのみ定まることを基礎にしている。このような性質は、functionalityの原理と呼ぶ。大域変数を介した状態の変更という副作用を利用しない言語は、この性質を満足している。それ

は LISP, PROLOG などである。関数型言語でのモジュール化は, PROLOGに代表される述語論理型言語では, より一層進む。LISP での条件文は, PROLOGでは, 各条件毎の文の集まりとして表される。さらに, そのような文の集まりは, データの形でシステムに蓄えられる。文同士の組合せは不要となる。それは, パターン照合に基づく手続き呼出し機構によって実行時に行われる。

これらの言語は, Algol, FORTRAN, PL/I などのメモリの更新を直接扱う言語に比べて, 経過依存的な計算を扱えない点で言語としての能力は劣るが, 数学的な扱いが容易であり, 上に述べたような良い性質を備えている。

垂直型組合せ技術は, 仕様と言語の深いギャップを埋め合わせる大切な技術である。その間に何層かの適当な中間のインタフェースを設けて, 仕様から基本言語への橋渡しを行う。この技術によって, 要求分析—仕様記述—プログラミングに渡るプログラミングの活動を, 統一した考え方, 記述体系によってサポートすることが可能となる。

現在のところ, 水平型および垂直型の組合せ方式を兼備したプログラミング言語は未だ出現していないが, それは, これからの理論的研究のメイン・テーマの1つである。

プログラムの自動合成を実現するためには, 人工知能研究の成果を採り入れて, プログラミングの知識を利用した推論システムによって実現するのが妥当である。言語あるいはモジュールの機能の数学的記述は, 推論システムが利用できる知識として, 知識ベースに蓄えられる。そして, 目的とするプログラムの持つべき機能を手掛りに知識ベースを探索し, 推論を積み重ねることによって, プログラムの自動合成が可能となる。

知的プログラミング・システムは, 知識情報システム開発のための高度な支援システムであり, またそれ自身が, 知識情報システムとして実現されるものである。この意味では, KIPS の応用例でもあり, プロトタイプとして考えることもできる。

〔2〕 必要性

知的プログラミング・システムの必要性の第1は, ソフトウェア危機の解消であることは言うまでもない。ハードウェアのコストの減少に比べ, ソフトウェアの生産性の向上は遅々として進んでいない。その原因は, ソフトウェアの作成が工学的な技術として正しい発展形態が見られなかったからである。工学的技術の原点は, 機能のモジュール化, 部品化であり, その組合せによる製品化技術である。そのような意味で, 知的プログラミング・システムは, 本来の技術的発展過程の上にソフトウェア技術を作り直す道具立てであると言える。このことによって, ソフトウェアのかかえている問題は, その多くが解決されると期待される。

このような適正技術の進歩は, 実は処理速度や記憶容量などのハードウェア的要件を考慮せずに, 議論することはできない。モジュール化技術は, 当面は現在の計算機の使い方に比べて十分効率的とは言えない面がある。しかしながら, ハードウェアの進歩, あるいは, ソフトウ

エア指向のハードウェアの開発によって、この問題は解決されるであろう。さらに、ソフトウェアの開発費までを含めた Total Cost を考えると、新しい方法の技術的妥当性がより明らかとなるであろう。

知的プログラミング・システムは、その技術の延長上に自動プログラミング・システムの実現を見込んでいるが、これは、次世代の計算機の利用が、職業的プログラマから、一般のユーザへと拡大されることを考えると、その技術の重要性を認識することができるであろう。

知的プログラミング・システムにおける検証技術は、今後 VLSI の出現によってソフトウェアのハードウェア化が進んで行く状況を考えると、非常に大切になってくる。プログラムの単純な誤りは、ハードウェア化の前に完全に取り除いておくことが必要だからである。

〔3〕 内外技術の現状

(1) 国内技術

データ抽象化をサポートするプログラミング言語には、京都大学のイオタ言語、阪大、電総研での代数的アプローチなどがある。これらは、すべて未だ試作実験段階であり、システムの実用性についての検証はこれからの段階である。

PROLOG は、電総研、武蔵野通研、東京大学などで、実験的使用、改良などがなされている。並列 PROLOG や図形言語の提案もなされている。

プログラム・ベースの研究は、始まったばかりである。電総研で、データ抽象化技法とからめた研究がなされている。東北大で、プログラムの自動合成のためではないが、各種のプログラムの相互利用を目指したプログラム・ベースの研究がなされている。

関数型言語の研究としては、抽象データ型をもった関数型言語の研究が東工大でされている。さらに、属性文法による仕様記述から関数型プログラムを導き出す研究が同じく東工大でされている。

データベースのための関数型言語である関係代数を基にした知的データベースが電総研で研究開発されている。そのシステムは、関数型言語上でのプログラムの合成、改良、およびデータ抽象化技術によるシステムの実働化を行っている。

(2) 海外技術

データ抽象化をサポートする言語は、Algol 系の言語である SIMULA 67 が最初のものである。SIMULA 67 は、並列に動作する対象群を記述し、シミュレートさせるために開発された言語である。そのとき、個々の対象 (Object) を中心とした記述を行うために、クラス概念が発明され、データ抽象化技術に育っていった。また、Object 間の情報のやりとりに着目した計算モデルとして ACTOR が MIT で開発された。ACTOR では、データ、手続きなどをすべてメッセージとして Object 間でやりとりし、計算を進めていく。その機構はデータフロー機構を包含するもので、理論的枠組としての重要性が認識されている。その後、対

話型の言語として SMALLTALK が作られた。SMALLTALK は、Object の考えでコンピュータ・グラフィックスの機能を整理し、ユーザにとって使い易い言語となっている。抽象データ型の形式的定義と、それに基づく証明系を持った言語としては、Alphard (CMU) と AFFIRM (USC-ISI) がある。代数的アプローチに基づく言語として OBJ (UCLA) データ抽象化、手続き抽象化、制御抽象化などを取り込んだ実用的言語を目指したものとして CLU (MIT) の研究開発がある。

論理型プログラミング言語 PROLOG は、主としてヨーロッパを中心に研究開発が進められている。多くの応用プログラムの記述がなされると共に、コルーチンなどの新しい制御機構、あるいは抽象データ型などの概念を PROLOG の中に取り込む研究が続けられている。他方、PROLOG の並列実行方式とマシン化の研究も始められている (米国カリフォルニア大学)。

〔4〕 研究開発内容

(1) 研究開発項目

(a) モジュラー・プログラミングと検証理論の研究

関数型言語に基づく水平組合せ技術と、データ抽象化に基づく垂直組合せ技術によるモジュラー・プログラミングの手法を確立し、その手法で作られたプログラムの検証理論を研究開発する。

検証を行うためには、プログラミング言語を構成する各命令の機能の正確な記述と、作るべきモジュールの仕様の正確な記述が必要となる。垂直組合せ技術では、作られたモジュールは、より上位のモジュールを作る際の部品あるいは基本命令となるので、これら 2 つの言語は、一つの枠組の中で統一的に使用できるものでなくてはならない。

(b) 仕様記述とプログラムの合成理論に関する研究

モジュラー・プログラミングを基礎として、与えられた仕様を満足するプログラムを合成する方式を研究開発する。

作るべきプログラムの仕様は、単純な入力・出力の関係による機能記述にとどまるものであってはならない。すなわち、計算速度・使用記憶量の両面からの性能記述などが含まれていなくてはならない。

プログラムの合成を行うには、多くの部品 (モジュール) の中からそのプログラムを組立てるのに役立つ部品を選び出し、それらを調整して組合せることが必要となる。とくに、各部品はデータ表現の差異を吸収できるようになっていなければならない。この調整技術は、用意すべき部品の数を適当な規模に抑えるために不可欠な技術である。

プログラムの自動合成にとって、もう一つの重要な技術は、どのようにして部品を選び出して、どのようにして組立てて行くかである。これは、人間がプログラムを作るときと同様の知的な能力を必要とする。本研究では、プログラムの専門的知識を知識ベース上に再現し

て、組立て作業の自動化を図ることを目指す。そのために、アルゴリズムのパターンによる分類等の研究を行う。

(c) プログラムの検証・合成システムおよびプログラム・ベースの開発

プログラムの検証・合成は、その挙動を記述された多数のモジュールが蓄積されてはじめて有効なものとなり、実用となる。この蓄積が、プログラム・ベースとなる。その上に、(a)(b)で開発された方式に基づくプログラムの検証および合成を行うシステムを開発する。

モジュールの蓄積は、各種応用分野毎に広範囲にわたって推進することが必要である。この作業は、モジュールの標準化の作業とも考えられ、困難ではあるが大切な作業である。また、標準化を行うためには、概念の整理をつきつめて行うことが要求される。いわば、プログラミング技術の集大成となるべきものである。また将来のVLSI化の種を提供するものでもある。

(d) プログラムの保守・改良・管理システムの開発

プログラム開発の大きな分野に、プログラムの保守・管理あるいは改良の問題がある。プログラムが大きくなるにしたがって、このような活動は困難性を増してくる。知的プログラミング・システムの目的の1つは、このような作業の機械化である。とくに、仕様の変更や、パフォーマンスの改善などのために、プログラムを容易に修正できる機能は必須である。そのために、以下の点について、今後の研究が必要である。

- ・柔軟なプログラムの追求
- ・等価変換技術
- ・プログラムのドキュメント管理、セマンティクスとプログラムの中身との対応づけの技術

(e) プログラム設計コンサルタント・システムの開発

プログラムの作成にとって、問題の仕様をきちんと与えることは、大変に困難な問題である。プログラミング言語が高度になり、思考に適合した言語になるに従って、その作業は徐々に楽になってきてはいるものの、とくにプログラムの非専門家にとっては、最も困難な仕事として残されている。本研究では、プログラム設計コンサルタント・システムの開発を目指す。具体的には、自然言語による質問応答システムとするため、プログラムの設計、仕様記述に必要な自然言語のサブセットを理解する言語理解システム、設計を支援するための know how を蓄積したコンサルタント・システムの開発を行う。その結果得られた仕様記述は、プログラム自動合成システムの入力となるものとする。

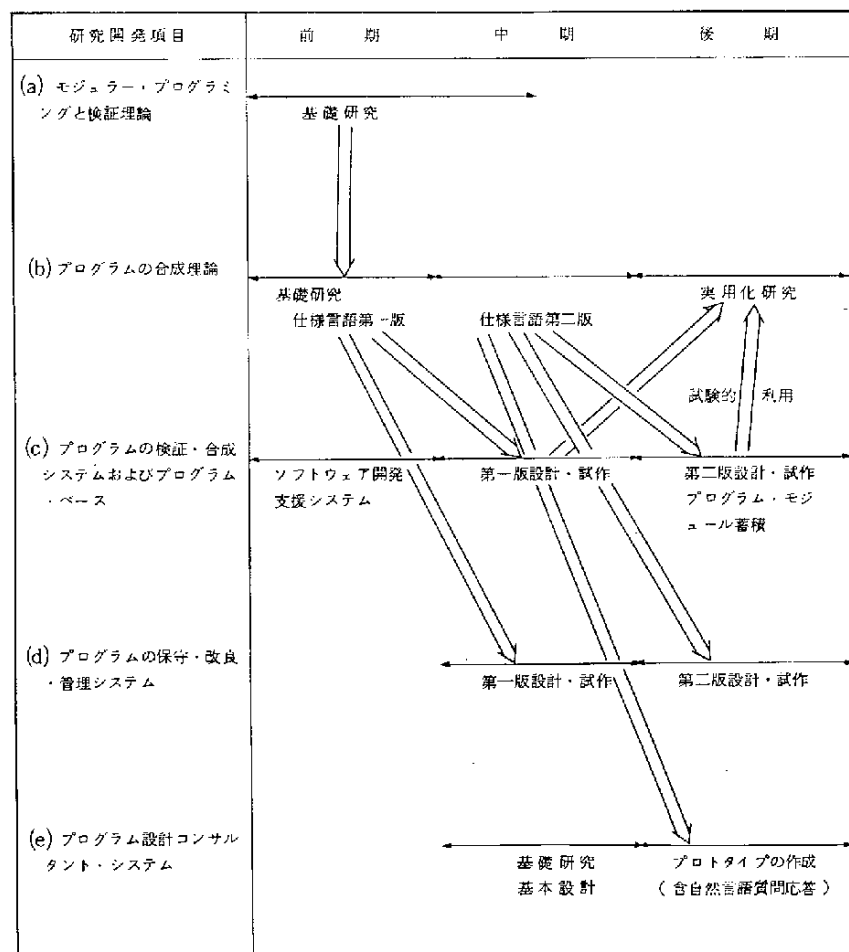
(2) 研究開発スケジュール

本研究課題では、前期においては、本プロジェクトの研究開発にとって必要なソフトウェア開発支援システムの開発を行う。その詳細については、システム化技術研究分科会報告書の第

Ⅱ部 3.1 システム化技術を参照されたい。また、ここであげた4つの研究テーマについても、基礎研究を中心に同時に着手する。

中期においては、基礎研究の成果をもとに、プログラムの検証・合成システムのプロトタイプを試作する。プログラムの保守・改良・管理システムについても同様である。プログラム・ベースの基本設計、並びにプロトタイプの作成を行う。プログラム設計コンサルタント・システムについては、そのための基本設計を行う。

後期においては、上記システムの最終版を設計・開発する。さらに、プログラム・ベースにプログラム・モジュールの蓄積を行う。プログラム設計コンサルタント・システムは、自然言語による質問応答を含めたシステムのプロトタイプを作成する。



(3) 研究開発の目標・仕様

以下に、前記研究開発項目(c)～(e)について、その目標・仕様を、中期と後期に分けてまとめる。

研究開発項目	中期目標・仕様	後期目標・仕様
(c) プログラムの検証・合成システムおよびプログラム・ベース	データベースへの検索手続きなど比較的小さく、一定分野のプログラムの合成、変換による改良。 小規模なプログラム・ベースの開発。 関数型、論理型およびデータ抽象化プログラムの検証システムの作成。	データベース管理システム、言語プロセッサなどの大規模プログラムの合成。 大規模なプログラム・ベースの開発。
(d) プログラムの保守・改良・管理システム	関数型、論理型プログラムについて、プログラム理解システムを作成。 等価変換実験。	プログラムの性能評価システムと等価変換による改良システム。 プログラム修正システム。
(e) プログラム設計コンサルタント・システム	(基本設計)	自然言語による質問応答。 データベース管理システム、データベース応用システム、などの設計コンサルテーションが可能なシステム。

2.6.2 知識ベース設計システム

(1) 概要

知識ベースシステムを設計し開発を行ない、運用段階へ至らせるまでの過程で、知識ベースシステムを作成する者を支援するための技術確立する。

知識ベース設計システムは、知識情報処理の基本応用システムの一つと見做すことができる。その基本的な機能は、知識情報処理システムを設計・開発・運用するのに必要な種々の技術と知識を知識ベースの中に組織的に所有し、その知識ベースに基づいて、知識ベースシステムを構築しようとする者を支援することである。

従って、知識ベース設計システムの構成も、基本的には、一般の知識ベースシステムと同様

に、知識ベースと推論機構から成っている。

研究開発の項目としては、一般の知識ベースシステムに関連した部分は基礎ソフトウェアシステムの知識ベース管理システムの項で研究開発を行うので、本項では取り上げないで、知識ベースを設計・開発・運用するという立場から特に重要である課題に重点を置くことにする。

知識ベースの設計・開発・運用で最も基本的な共通技術は、知識に関する知識（これをメタ知識と呼ぶ）をコンピュータ内でどのように表現し、それに基づいて抽象レベルの推論を行なうかという課題である。この課題を、知識ベース設計システムの基礎研究課題として位置づける。

次に知識ベース設計システムの核である知識ベースの開発という課題がある。この知識ベースには、知識ベース設計・開発・運用のための知識が収納される知識ベース、知識獲得のための方式や表現、あるいは人間やデータからどのようにすれば知識を抽出して得ることが出来るかというような知識が収納される知識ベースなどが主な開発対象である。

更に、知識ベースが広く一般のユーザに使われだすと、途端にシステムの規模が大型化するという傾向があるので、そのために大型の知識ベースを構築する技術の研究開発も、重要な課題である。この大型化の傾向は、研究室から現場へ出て、世界的に実用的レベルで使われている DENDRAL という知識ベースシステムで発生しているもので、実用しているユーザは、コンピュータの制約条件などはとかく無視して、性能本位でシステムを使うというのが本来の姿であるということを反映するものである。ハードウェアの大型化と、アドレス空間の拡大がこの傾向を助長するので、大型の知識ベースの構築技術の開発が不可避である。その中心的技術は、知識ベースの共同開発と拡張のための技術である。

知識ベース設計システムの目標性能は、知識の量がルールに換算して 20,000 個程度で、知識の獲得を半自動的に円滑に行う機能と、推論の過程を日本語で説明する機能とを有する知識ベースシステムを、2人/月以下で構築可能とするものとする。

〔2〕 必要性

社会の広汎な分野で知識ベースシステムに対する期待感が高まりつつあるが、知識ベース構築の技術が確立すれば、現在は潜在している知識ベースシステムへの需要が、一気に表面化すると考えられる。それは、知識ベースシステムが、問題に適した柔軟な応用システムとして最適であるからである。

知識ベースシステムは、問題の記述がやり易く、拡張性や保守性に勝れているために、大型のシステムや、カスタマイズされた細目の細かいシステムを、比較的安価に作ることを可能にする。またその推論メカニズムは、非決定論的な推論を必要とするような問題の解決システムを可能にするので、従来の手続き型プログラム技術で実現困難であった種類の問題についても、応用システムとしてまとめる手法を提供する。更にこの知識ベースシステムの基本的構造が問題解決の手順の部分と知識の部分とに分離してできていることから、知識ベースシステムの核

となる知識ベースそのものが、一度作られると一連の作業において共通に使うことができる。たとえば、応用システムとして、何か高度な装置の故障診断のための知識の他に、その装置の構造や機能に関する知識が必要であるが、これはこの装置を設計し製作する際に開発される知識ベースを再び使える。すなわち、知識ベースシステムは、それを構築する際のオーバーヘッドは、システムがいろいろ作られて拡張されるに従って、知識ベースを共通に使うことにより、急速に軽減するという利点がある。これは情報の蓄積を有効に活用できることを意味する。

このように知識ベースシステムには多くの利点があるが、これを利用するために要求されるのが、そのような知識ベースシステムを効果的に効率よく構築する技術である。従来のプログラミングシステムとは異なり、推論機構と知識ベースとが明確に分離した構造を持ち、問題に応じて最適な推論メカニズムを構成し、その問題の関連知識を知識ベースとして組織する機能を持つ知識ベース設計システムは、それ自身、知識ベースシステムの応用システムである。

従って、たとえ高級プログラミング言語があっても、知識ベースシステムの枠組である推論機構から知識ベースまでをその言語レベルから作製すると、それ程大規模ではないシステムでも少なく見積っても3～4人/年のマンパワーが必要となる。

本来知識ベースシステムは、応用分野の個性や利用者の好みに合わせて、いわば“カスタムメイド的”に使われるという魅力がキャッチフレーズでもある。この特色を経済性を保ちながら実現する技術の研究開発の必要性は明らかである。

〔3〕 内外技術の現状

(1) 国内技術

国内においても、知識ベース設計システムに対する関心が高まりつつあり、いくつかの提案がなされ始めた段階である。たとえば、東京大学では、医用コンサルテーションシステムの構築の要件の整理作業が進んでいる。

知識ベース設計のための技術の基礎は、知識ベース管理システムのための技術である。それに関しては、東京大学、京都大学、大阪大学、九州大学、豊橋技術科学大学、東京理科大学、電総研、武蔵野通研等で行なわれている。（知識ベース管理システム参照）。

(2) 海外技術

米国では、知識ベースシステムのツール（道具）の研究として、すでに7年以上の歴史がある。

代表的な例としては、Stanford 大学で開発が行なわれている EMYCIN, AGE, Rutgers 大学で開発されている EXPERT などのシステムがある。

EMYCINはプロダクションルールに基いた知識ベースシステムを構築する場合に、ルールのインプリメントを支援するシステムである。文法上の誤りやミススペルのチェックも行いう機能を持っている。EMYCINを用いて、MYCIN, PUFF, SACON等のシステムが非常に

短期間に開発することができた。

AGE は、知識ベースシステムを作ろうとしている人を支援するシステムで、黒板モデルに基いたシステムである。知識工学上の種々の技術をパッケージ化してあり、ユーザに対するアドバイスの機能も強力である。

EXPERT はやはり知識ベースシステム構築の支援システムであるが、FORTRAN で作られており、移植性が高いという特徴がある。

知識ベースの誤りを見出し、修正するシステムとしては、Stanford 大学の TEIRESIAS がある。

知識表現に関するシステムとしては、同じく Stanford 大学で UNIT, RLL などというシステムの開発が行われている。

〔4〕 研究開発内容

(1) 研究開発項目

(a) メタ知識の表現方式とその利用技術の研究

知識ベースシステムは、知識ベースと推論マシンから構成される。従って設計技術は、この 2つの要素の設計技術と、全体のシステム化を行うための設計技術が中心となる。

知識ベースシステムは、応用分野の問題の物質や、そのシステムが対象とする利用者の好み等によって、見掛けはかなり違う形を取ることになるであろう。従ってここでは、共通性が強い基礎的な技術の研究開発を重点におくことにする。

この観点から、2つのサブ項目を設定する。その第1は、知識に関する知識の表現方式の研究である。知識に関する知識を、メタ知識と呼ぶことにする。メタ知識には、知識の構造に関するメタ知識と、推論の制御手順に関するメタ知識の2種類がある。

知識ベースシステムを設計する場合に、そこで使われる知識を構造化して、知識ベースの枠組を作っていかなければならない。現在のところ知識を階層的な構造に組立てるのが良いとされている。この場合、知識のどのような側面、あるいはレベルで階層化を図るかは、問題の記述と推論の効率に重大な影響を及ぼすことは明らかである。

そのための技術は、個々の問題の特質に依存することは避けられないが、知識を「抽象レベル」で分類し、そのレベルで構造化を図り、具体的な知識を構造化する際のガイドラインを生み出すという手順によれば、共通技術として多大な効果を設計の段階に及ぼすことになる。

第2は、メタ知識を対象とした推論方式と検証理論の研究を挙げる。知識の構造と推論の制御手順についてのメタ知識を用いて、抽象レベルの推論を行なうことは、この知識ベース設計システムで設計した知識ベースシステムのシミュレーション機能を持つことになり、従って出来上がった知識ベースシステムの間接的検証技術を提供するとともに、そのシステムを

修正，拡張する際に，有効な助言を与えることを可能にする。

またメタ知識自身が完全であることを検証するための理論研究及び手法の開発も行う。

(b) 知識ベース設計開発支援システム

知識ベースシステムを構成する知識ベースと推論マシンの内，知識ベースの設計開発法の開発は特に重要である。その理由は，個別の知識ベースシステムでは，その推論マシンはシステムによって，それ程のバリエーションが存在するとは考えられないので，基本的な推論マシンが用意されていれば，それを実際の知識ベースシステムの推論マシンとして使うことは，それ程手数は掛らないと予想される。ところが知識ベースは，出来上った知識ベースシステム毎に，かなり個性的なものが要求されていて，かなりのバリエーションが存在すると考えられるので，それを統一的に取扱える設計道具の開発は，効果が大きい。その機能を果たすのが知識ベース設計開発支援システムである。

これは，2つのサブ項目よりなる。第1は知識ベース設計法の開発である。知識ベースの枠組として，現在のところプロダクションシステム，フレーム理論，セマンティックネットあるいは黑板モデル等いくつか具体的な提案がなされている。また基礎ソフトウェアシステムの知識ベース管理システムというテーマの下に新しい枠組の開発が行なわれる。

このような知識ベースの枠組は，問題によって適不適があり，知識ベースの設計において，どの枠組をどのように使うとよいかも，設計者へ助言できるシステムを開発することを目標に，基礎研究を行う。

第2のサブ項目は，知識ベース設計開発技術の知識ベース化である。前記サブ項目の研究成果をもとに，知識ベース設計開発に関する知識，すなわち「知識ベース設計開発の専門家の知識」を知識ベースとしてコンピュータ・システムに固定する作業を中心に，知識ベース設計開発支援システムを開発する。

(c) 知識ベース増殖支援システム

知識ベースシステムが広く一般に使われるようになると，システム自体が大型化する傾向にある。このような大型のシステムは，一度に完全な全システムを開発することが困難になり，基本部分から逐次大きく成長させていく手順を踏むのが普通である。そのために必要な技術が知識ベース増殖支援システムである。

特に知識を如何にシステムの中に取り込むかという技術の成否は，大型の知識ベースシステムを開発する場合のボトルネックとなる。また，知識ベースの保守技術，システムの効率を上げるための知識ベースのコンパイル技術なども重要となる。

ここでは，次の3つのサブ項目を特に取り上げ，知識ベース増殖支援システムの開発を目標に研究を推進する。

第1は知識獲得技術の知識ベース化である。知識獲得技術の基礎的な研究開発は，基礎ソ

フトウェアシステムの中の知識ベース管理システムで行なうが、ここでは、知識獲得のために必要な、いろいろなノウハウを中心とした知識ベースの開発を重点に行う。

この知識はいろいろな局面に関する知識より構成される。たとえば、知識提供者から如何にすれば、上手に知識ベースの枠組に合った形で知識を引出せるかという知識なども含まれる。

第2は無矛盾性チェック機構の研究である。ここでは、知識ベースの内容に、互に矛盾するものが含まれていないか、あるいは欠如しているものがないかを、自動的に検証するメカニズムの研究開発を行う。その結果は、知識の獲得の自動化へも利用できる。

この機構により、大型化する知識ベースの改良、保守が容易になる。

第3は知識ベース・コンパイラの開発である。知識ベースシステムとして具体的な応用システムを作ろうとする問題は、往々にして問題自身十分に解明されつくしていない場合がある。たとえば、医療診断のコンサルテーションを行なう知識ベースシステムを作る場合でも、知識ベースを作っている段階では必ずしも診断の理論が確立しているとは限らないのである。従って、作られる知識ベースの内容に矛盾がないとは言えない。これについては前述の無矛盾性チェック機構の研究により、調べることが出来る。

更に、その知識ベースの内容が、推論を実行するのに具合良く構成できているかを自動的に調べ、できるだけ効率よく知識の起動がなされるような機能を持つ知識ベースコンパイラの研究開発を行う。

(2) 研究開発スケジュール

研究開発項目 \ 年度	1	2	3	4	5	6	7	8	9	10
(1) メタ知識の表現 方式と、その利用 技術の研究		(I)			(II)				(III)	
		メタ知識の表現方式								
		(I)			(II)				(III)	
		メタ知識を対象とした推論方式と検証理論の研究								
(2) 知識ベース設計 開発、支援システ ムの開発		(I)			(II)				(III)	
		知識ベース設計法								
		(I)			(II)				(III)	
		知識ベース設計開発技術の知識ベース化								
(3) 知識ベース増殖 支援システムの開 発		(I)			(II)				(III)	
		知識獲得技術の知識ベース化								
		(I)			(II)				(III)	
		無矛盾性チェック機構								
		(I)			(II)				(III)	
		知識ベース・コンパイラ								

(3) 研究開発の目標・仕様

知識ベースシステムを設計し開発を行ない、運用段階へ至らせるまでの過程で、知識ベースシステムを作成する者を支援するための技術確立することを目標にする。その開発目標・仕様は次のとおりとする。

① 相当な専門知識を必要とする高度な問題について、専門家にとっても有意義なコンサルテーションを提供する知識ベースシステムの構築を容易に行なえるものとする。

② 設計する知識ベースシステムの規模は、その知識を規則表現に換算して、約2万規則程度のものとする。

③ メタ知識レベルで、システムの検証がある程度行なえ、大型の知識ベースシステムのデバッグが容易に行えるものとする。

④ 中間目標として、設計すべき知識ベースシステムの規模を、目標の3割程度とする。

⑤ この知識ベース設計システムによって設計すべき応用システムの例として、化学プラントあるいは原子力プラントなどの運用のためのコンサルテーション・システム、VLSI設計のためのコンサルテーション・システム、新材料開発のためのコンサルテーション・システム、医療診断コンサルテーション・システム、知的教育システム、知的エージェント・システム、あるいは法律に関するコンサルテーション・システム等、社会的要請が大きくかつ実現により大きな効果が期待されるものを想定する。

2.6.3 アーキテクチャ関連システム化技術

〔1〕 概 要

第5世代コンピュータ・システムは、知識情報処理の実現を目指す、きわめて高度なシステムの構築を目標としている。一般に、このような未知の分野を多く含む目標へ到達するためには、いろいろな要素技術をいかに選択し、整理し、まとまった体系に組上げていくかというシステム化の技術がきわめて重要である。

これは、実際に物を作る時のノウハウに類するものから、開発の哲学、方針といったもので、いろいろなレベルのものがある。ここでは、この中においても、より実地的なアーキテクチャやハードウェアを開発することを目的としたシステム化技術に範囲を絞って述べる。また、同時に、システム化のための道具についても触れることとする。

コンピュータ・システム構築のためのシステム化技術は、要素となるいろいろなサブシステム群、すなわち、ソフトウェア・モジュール群、マシンを構成するハードウェア・モジュール群、これを作るためのVLSIのような部品群、さらには、コネクタやケーブルのようなより基礎的な部品・材料群などの、設計・開発、および、それらを組立てるに必要とする技術と考える。

これは、部品の選択のための評価手法、各モジュールや部品の仕様の標準化、組立てる場合の規則の統一化、目的のものができたか否かの検証、評価の手順の明確化といった形で、具体化される。さらに、これらの規則や手順に従ったものが、スムーズに作られ、実現されるように、各種の工具や開発支援ツールが開発される。

コンピュータの開発においては、その開発支援ツールは、やはりコンピュータであり、開発目標に至る過程では、その中間的性能を有するコンピュータが、何段階かにわたって作られる。これにより、少しずつ性能の高いものを作っていくことになる。これは、まさに、自己増殖のプロセスであり、この過程を、効率よく管理していくのもシステム化技術である。

以下、このような考え方に基き、第5世代コンピュータ・システムの開発に必要なシステム化技術について述べる。

〔2〕 必要性

システム化技術は、広い概念であり、いくつか、要素に分けて考える必要がある。ここでは、次の4つに分け、その必要性を述べる。

まず、第1は、仮想化の技術である。知識情報処理を目ざす第5世代コンピュータ・システムでは、そのソフトウェアが、きわめて重要な位置を占め、その規模も大きなものとなる。また、それを支えるハードウェアも複雑で、高機能のものとなる。このため、ソフトウェアで構成する論理マシン（仮想マシン）とハードウェアで構成する物理マシン（実マシン）の間を結ぶ仮想化の技術が必要となる。

第2は、いくつかのアーキテクチャ関連技術の融合化技術である。この中には、ソフトウェアも含まれる。アーキテクチャは、機能分散を軸としたアーキテクチャ、VLSIを軸としたアーキテクチャ等、いくつかの異なる軸をもつアーキテクチャが存在し、これとソフトウェアを組合せた上での、融合化技術が必要である。この中には、応用目的とする仕事をどのように分割し、どのアーキテクチャでサポートするかといった、いろいろなレベルの負荷分散、評価の手法も含まれる。

第3は、大規模、高信頼化システムの設計開発技術である。大規模なハードウェア、ソフトウェアの構築には、人間の管理能力を越える数の部品やモジュールを用いねばならない。また、その信頼性を高めるためには、その組立てに際し、いろいろな技術を必要とする。このような条件のもとでの開発では、まず、システマチックな構成手法の開発と、その機械化（コンピュータ化）が不可欠となる。また、新しい手法の開発には、試行錯誤的プロセスが必然的に含まれる。このような技術は、未知の領域を多く含む第5世代コンピュータ・システムの開発では、きわめて重要である。

第4は、システム化技術を効果的に適用するための開発ツールとしてのコンピュータ・ネットワークである。ネットワークは、研究開発グループ間の情報交換、研究開発成果の交換に有

効だけでなく、互に、ソフトウェアやハードウェアのモジュール・インタフェースの統一に有効である。ネットワークの各インタフェース仕様に従わねば、用いることができないから、いろいろなレベルでの仕様の標準化がすすめられる。また、ネットワークに付加される開発支援用コンピュータについても、言語の統一化等をはかることが必要である。

以上述べたような技術を重点項目として、システム化技術の開発を行うことが必要である。

(3) 内外技術の現状

(1) 国内技術

国内においては、研究開発に重点を置くシステム化技術の蓄積は、十分とはいいがたい。コンピュータの本体を中心とするシステム化技術は、近年、かなりの進展を見せ、アメリカを中心とする欧米の技術水準に伍するまでとなってきた。しかしながら、ローカルおよびグローバル・ネットワーク技術については、日本では、電電公社主体で行われているもの以外は、制御用コンピュータシステム等で用いられているものが主であり、立ち遅れが目立っている。特に、ネットワーク用のソフトウェアは、近年、開発がすすめられたばかりであり、そのノウハウ等の蓄積も十分とはいいがたい。

また、大規模ハードウェアの構築用のCADシステムなどは、現在は、多くがアメリカよりの輸入に頼っており、進歩的なCADになる程、この傾向が強い。

ソフトウェアの構築に関するシステム化の技術は、多くが今後の研究テーマであるが、この面でも、立ち遅れが目立ち、今後の精力的な研究が必要である。

(2) 海外技術

海外技術の中心は、やはり、アメリカである。仮想化や融合化の技術、大規模システムの構築の技術とも、公的な研究機関（ARPAやDOD、大学）、メーカー（IBM、DEC、Xerox等）の双方において、よく研究されている。

ソフトウェアの仮想化は、ネットワーク技術と関連して研究されており、ARPAネットワーク等で実証されている。また、小規模なローカル・ネットワークにおけるシステム化技術も、XeroxのEtherネットワーク等で実証されている。

大規模なソフトウェアのシステム化技術は、IBMを中心とするメーカーで実際的な研究が行われ、大学等では、ソフトウェア工学等の基礎的な研究が行われており、多くの成果が報告されている。VLSIを目的とするCADシステムも、CALMA社を始めとするメーカーで、実用的で優れたシステムが開発されており、より高度なアーキテクチャの設計から、マスク・パターンまでの一環した設計手法とそのシステムについては、MIT、CMU等で、よく研究されている。

我国でも、ネットワークから、CADシステムまでの一連のシステム化技術の開発を強力にすすめる必要がある。

〔4〕 研究開発内容

(I) 研究開発項目

① 仮想化とシステム構成技術

プログラミング言語，オペレーティング・システム，ネットワーク・インタフェース，アーキテクチャ・レベル，機能モジュール・レベル，VLSIを含むコンポーネントのレベルのそれぞれの階層において，基準となるべき階層間のインタフェース仕様を決定する手法を開発する。この際によりどころとなる要素間接続の論理モデル（たとえば，OSI標準モデルのようなものや，ACTORモデルのようなもの）の研究を行う。これら技術によって，ソフトウェアの研究開発成果を応用目的に応じて，柔軟に選択構成するための手法の開発及び，その標準化を行う。

② 融合化と負荷配分の最適化技術

知識情報処理を目的とした応用システムの実現に際し，機能分散システム上への負荷配分と，これを行うために，ソフトウェア，新アーキテクチャに基づくマシン，その他の要素マシン，マシン中の機能モジュールのそれぞれの間での，役割と機能の分担を決定する手法について研究を行う。また，その負荷の配分の最適化や，システム構成についての評価のための手法を開発する。

③ 大規模システム設計・開発技術

新アーキテクチャに基づくマシンの設計開発のための方針や，アーキテクチャ設計のための自動化ツール（CADシステム），評価ツール，回路実装のために必要な部品，材料についての研究，開発を行う。また，大規模なソフトウェアの開発に必要な要求仕様技術，開発・評価ツール，高品質化のための技術の研究開発を行う。ソフトウェア，ハードウェア，VLSIまでの一環した設計開発支援システムをネットワーク上に開発する。

④ 超高信頼化技術

ソフトウェア，ハードウェアの高信頼化のための技術を開発する。このために，ロジック・レベル，モジュール・レベル，サブシステム・レベル，システム・レベルのそれぞれにおける耐障害化のための設計・実装技術を研究開発する。

⑤ アーキテクチャ開発ツールとしてのローカル・ネットワークの開発

ハードウェアのシミュレーション，実験機の試作，プログラミング言語や処理系の開発，VLSI等のCADシステムの開発などの支援のためにローカル・ネットワークを開発する。各種の実験・試作・評価は，このネットワークを用いて行う。特に，ハードウェアとソフトウェアを組合せた総合評価は，ネットワークが有効なツールである。

ローカル・ネットワーク間は，適宜，グローバル・ネットワークで接続する。

(2) 研究開発スケジュール

開発初期の段階で、シミュレーションやハードウェア試作のワーキング・ベンチの目的で使用するコンピュータを準備するとともに、コンピュータやCADシステムなどの間を結ぶローカル・ネットワークを付設する。

仮想化や融合化の手法については、初期にその基本方針を示し、これに従ったソフトウェア、ハードウェアの開発を促す、開発の進展に伴い、仕様の標準化などを行い、研究成果の幅広い利用と、構成の柔軟性を保障する。大規模システムの設計、開発については、自動化をすすめ、短いターン・アラウンド時間で設計や変更の行えるインタラクティブ・システムを開発していく。VLSIのCADシステムを含め、これらのツールは、ネットワークを介し、多くの研究者に利用できるようなものとする。高信頼化技術は、VLSIを含むハードウェアの能力を十分に生かす技術を開発し、設計開発に役立てる。

第5世代アーキテクチャのシステム化技術の研究開発スケジュール

	1	2	3	4	5	6	7	8	9	10
仮想化技術	階層化のための共通仕様決定方針の作成		標準化インタフェース仕様の作成							
融合化技術	システムの負荷配分や構成法の研究		システム評価用ツールの開発			評価データを用いたシステムの解析				
大規模システム設計、開発技術	CADシステムなどの整備とネットワークへの接続		機能分散システム上へのツールの実装と改良			第5世代マシンプロトタイプ試作用ツールの拡充				
超高信頼化技術	基本的手法や実装技術の研究		機能分散システムへの適用			第5世代マシンのプロトタイプへの適用				
開発用ネットワーク	開発支援用コンピュータとローカルネットワーク		機能分散ネットワークへと発展							

(3) 研究開発の目標・仕様

① 仮想化技術

開発支援マシンおよびローカル・ネットワーク上へ作られる実験用システム、中間目標システムのモジュール間インタフェースの標準化、統一化のための作業を行い、インタフェース仕様の標準化を行う。

また、この標準化仕様を普及させるためのマニュアル、手引書などを作成する。

このほか、システムの階層化の規準を作る。

② 融合化技術

シミュレーション等により、負荷配分、システム構成方法などの解析を行う。開発支援マシンおよびローカル・ネットワーク上に、マシンの負荷の測定ツールや、ネットワーク上のトラヒックの測定ツールを付加し、システム評価用のシステムを開発する。

③ 大規模システム設計開発技術

ソフトウェアの開発を容易にする各種ユーティリティ・プログラムを開発する。また、開発されたソフトウェアの蓄積、交流をはかるため、ソフトウェア・モジュールとそのドキュメント類も含めて貯えられる集中データベース・システムを開発する。これには、ハードウェア開発に用いるシミュレータ等のソフトウェアやハードウェアの説明書等も蓄えるものとする。

ハードウェアの開発に関しては、種々のシミュレータや評価ツールのうち、共通に用いられるようなものを開発する。回路等の記述言語や処理系、シミュレータなどが、この対象となる。また、CADシステム等も一部開発する。

④ 超高信頼化技術

高信頼化のためのソフトウェア、ハードウェアの実装手法を開発し、各種システムにその基本要素となるモジュール、部品等を供給する。

⑤ 開発用ネットワークと開発支援マシン

研究開発の初期にまず既存技術によるネットワークを設備する。また、開発支援マシンを開発もしくは、購入して、研究開発ツールとして提供する。ネットワークとしては、ローカル・ネットワークおよびグローバル・ネットワークを設備する。

初期の開発用ネットワークのローカル・ネットワークとしては、データ転送速度1～5 Mbits/sのものを、グローバル・ネットワークとしては、公衆回線を利用し、データ転送速度5～10 Kbits/sのものを設ける。①～④で開発したシステム化のためのツールは、すべて、このネットワークへ接続して使用できるものとする。

2.6.4 データベースおよび分散データベース・システム

〔 1 〕 概 要

データベース・システム (DBS) は、知識情報処理を含む、あらゆる情報システムの中核となるものである。今後、社会の情報化に、とまない種々の情報が、情報システム内に組みこまれていく。これらの情報としてはいは、従来の定型的な文字、数値情報に加えて、アルゴリズム、図形、画像等の新たな情報がある。データベース・システムは、これらの情報の格納場所として、ユーザの必要とする情報を、必要な時に、必要な姿で提供するシステムである。今後は、現在、データベース化し得ない情報 (例・書類、設計図、アルゴリズム) のデータベース化に取り組んでいかねばならない。

従来の CODASYL, IMS, リレーショナル, ADABAS, S2000 等の DBS は、固定されたスキーマのもとで検索中心に利用されてきた。これらの特徴は、“固い” DBS であることである。今後のユーザ・アプリケーションの多様化と、高度化に対して、現状の DBS は柔軟性を欠いているといえる。こうしたアプリケーションの変化に適応していくためには、データベースに格納されているデータの意味を表し得るデータモデル論、ユーザの必要とするデータを明らかにするスキーマ設計方法論、容易なユーザ・インタフェース (例・自然言語による QA システム)、新データ (図形、画像、アルゴリズム等) を扱い得る超文字データベース・システムの研究開発が必要になる。

更に、こうした機能を提供するための DBS 自体の管理機構として、ユーザの動的な要求に対して、最適なシステム構成 (論理及び物理構造) をとり得る柔構造 DBS が必要である。

又、通信ネットワークと、DBS 技術の進歩によって、複数の論理的に異なった DBS を統合的に利用させる分散データベース・システム (DDBS) は、主要な技術となってくる。DDBS は、上述してきた技術を総結集したシステムでもある。

ユーザの要求を、有効に物理レベルで実行させる機構として、データベース・マシンが重要である。データベース・マシンが、通信とプロセッサとの結合体であることから、通信しながら処理するという DDBS と同様な概念が必要になってくる。この意味から、DDBS と密接に関連したものとなってくる。

〔 2 〕 必要性

データベース・システム (DBS) は、今後の情報システムの核となることは疑う余地がない。第 5 世代の知識情報処理の知識ベース・システム (KBS) の基本技術であるとともに、機械翻訳、自然言語処理の大規模辞書、CAE/CAD における図形・画像を中心にしたエンジニアリング・データベース、オフィス情報システム等におけるパーソナル用データベース、更に、第 5 世代コンピュータのシステム化技術 (アーキテクチャ、DBS、素子 DBS、プログラム・ベース等) の第 5 世代コンピュータ・システムの全ての応用においてデータベース・システム

は必要になる。特に、知識ベース管理システム（KBMS）においては、複数サイトで形成されてきた特性のある情報を統合し、新たな高度な抽象情報（知識）を形成し、これをアクセスすることが必要になる。このKBMS機能を支える主要な技術は、分散データベース・システム（DDBS）である。

第5世代アプリケーションとしてのオフィス情報システム（OIS又はOA）、CAD/CAD/CAEを見ても、データベース・システム（DBS）と分散データベース・システム（DDBS）は主要な要素となっている。OISにおける高度な戦略的意思決定に対しては、関係する種々のDBSの統合が必要となる。OISの日常的な書類（フォーム）の処理も、データベース化することによって、自動化が可能となる。CAD/CAEの設計図作製においても、データベース化した図面のバージョン管理等が重要である。

又、DDBSとDBSとは、システム化技術の実証例として、研究を進めるべきものである。あるいは、システム化技術そのものの利用を図る際に、多様な面で必要となるものである。たとえば、アーキテクチャのシステム化技術では、素子データベースやアーキテクチャ・データベースが実際には非常に役に立つ。また、プログラムの自動合成では、プログラム・ベースが主要な働きをする。

このように、データベース自身の開発は、システム化技術にとっても、さらに、自然言語理解システムでの辞書や、画像データベースなど、第5世代コンピュータの基本的な応用システムにとっても必要不可欠である。

〔3〕 内外の現状

(1) 国内技術

データベースのセマンティクス及びモデル論の研究としては、富士通国際研、北大、横浜国大、日立等で研究がなされている。柔構造データベース・システムとしては、IDBS（ETL）、APAD（東大）等が研究開発されている。自然言語によるQAシステムとしては、「やちまた」（IBM）がある。分散データベースについては、JDDBS（Jipdec）、制御問題（信頼性制御、同時実行制御）（沖、日電）等の試みがある。データベース・マシンとしては、データフロー型データベース・マシン（北大）の研究が目されている。画像データベースとしては、阪大の研究がある。知識ベース・システムとの関連においては、東大、ETLにおいて、研究されている。

(2) 海外技術

海外では、多くのDBMSがつくられている。IBMのIMS、IDMS（HIS）、S2000（MRI）、ADABAS（Software AG）、Model 204（CCA）等である。この中でCODASYL型DBMSについては、国際的な標準化が進められている（CODASYL 73, 78）。大型機から、ミニコンまで、最近では、CODASYL型

のDBMSを備えてきている。ADABAS, S2000も、数百に及ぶユーザを持ち、使い易さが特徴となっている。

商用リレーショナルDBSとしては、SQL-DS (IBM), QBE (IBM), MAGNUM (Time share), INGRES (UC・Berkeley) が有名である。SQL-DSは、DOS/VSEユーザ向けで、DL/IをインタフェースとしてIMSとも通信できる。言語はSEQUENCELをユーザに提供している。QBEは、2次元表示に基づいて、例 (example) によって、ユーザにアクセスさせるシステムである。DD/Dもまた、同様に管理される。これに、トリガーとメール機能を付加した、オフィス・システムとしてのSBA (IBM) がある。他にIBMのSystem Rが開発されているが、いまだに市場に出ていない。既存DBMS (IMS) 市場の存在によって、商用システムとなるかは疑問がある。

CCA社によるSDMSは、リレーショナルDBS (INGRES) 上に、アクセス対象も2次元であるアクセスが可能である。種々の抽象化レベルを上下にナビゲートすることによって、必要な情報を得るものである。Query-by-Pictorialも図によるアクセスを試みている。

分散データベース・システム (DDBS) は、SDD-1 (CCA), POLYPHEME (Univ. of Grenoble) の開発が終わり、第Ⅱ期DDBSの研究開発にはいっている。第Ⅰ期DDBSは、広域ネットワークを用いたリレーショナルDBSからなる。同種DDBSであり、分散を意識しないアクセスを可能とした。SDD-1の商用版の分散型モデル204が今秋発表される予定である。第Ⅱ期DDBSは、ローカル・ネットワーク、OISへの適用、更新 (同時実行制御)、異種性問題の解決を目指して、研究開発されている。例として、SIRIUS-DELTA (INRIA), CCAのシステム等がある。これらは、データベース・マシンと密接に関連してきている。例として、INGRES-MUFFIN (UC・Berkeley), DIALOG (Maryland Univ.) 等がある。

自然言語のQAシステムとしては、LADDER (SRI), PLANE (イリノイ大), ランデブー (IBM), ROBOT (ダートマス大) がある。これらは、いずれも非専門家ユーザに対して、限定された英語による問合わせを許すものである。

論理とデータベースとの関連では、Truth Maintenance Systemがある。PROLOGとデータベースとの関連も、研究が盛んになってきている。

データベースとプログラミング言語におけるデータ抽象化技法との関連は、Rowe (UC・Berkeley), Wasserman 等が試みている。

〔 4 〕 開発内容

データベース・システムと分散データベース・システムの研究開発項目としては以下のものがある。

- a) データ意味論とデータ・モデルの研究
- b) 柔軟構造データベース・システムの研究
- c) スキーマ設計支援システムの開発
- d) データ蓄積支援システムの開発
- e) 自然言語による Q A システムの開発
- f) 分散データベース・システムの開発
- g) 超文字 (多形態) データベース・システムの開発
- h) データベース・マシンの開発

(1) 研究開発項目

- a) データ意味論とデータ・モデルの研究

データベースの研究の基礎となるものは、データ自身が表わしている意味および、それを基にした世界の概念化、モデル化の追求である。データの意味として考えられるものに、2種類ある。その1つは、データ化された記号あるいは数値それ自身の意味である。ある数が人の年齢であるのか、あるいは身長であるのか、あるいはある集団における平均身長であるのか、といった事柄である。他の1つは、他の概念との相互関連である。ある人が決まれば、その人の年齢は一意的に定まる。このような関連は、依存性あるいは従属性として知られており、データベースの設計に利用されている。

データ・モデルを考えると、問題とされるのは、更新操作の難易と、モデル自身の可変性である。第1の更新操作については、それを含めたデータ意味論が必要である。更新操作は、一般には対応する世界の変化に対応している。これは、世界を構成する対象の様相と考えられる。

第2のモデル自身の変更は、世界の構造的変化に対応している。あるいは、モデルが現実世界の変化に応じ切れなくなつたと考えられる。論理の世界で言えば、モデルを構成する公理系の変更である。もともと、モデルは世界のとらえ方の1つの仮説であるから、この時点で仮説の変更が必要となる。このようなモデルの変更の機構として、人工知能の分野で知られているものに Truth Maintenance System がある。そこでの研究成果の活用が本問題の解決の鍵となろう。

目標としては、上に述べた、データの更新およびモデル自身の更新を含むデータ意味論の確立と、それに伴うデータ・モデル管理機構の開発である。

b) 柔構造データベース・システムの研究

データベース・システムのライフサイクルにおいて、データベースを利用するアプリケーションの変化、新たなアプリケーションの追加、又は、コンピュータ・システムの変化（例・物理デバイス、ホスト・コンピュータ）といったシステム環境の変遷に対して、データベース・システムが柔軟に適應できる必要がある。データベースに対する変化の第1は、データの値の変化である。アプリケーションによるデータの更新（追加、消去、削除）に対して、データベース内のインテグリティと一貫性を保つことが必要である。又、この変化に對應して、最適な物理構成（例・ディレクトリ構成）を動的に取ることが求められる。複数ユーザによる共有データに対する同時更新を有効に行なう機構も必要である。更新頻度と一貫性要求が厳しくないデータについては、物理的冗長性を高めていき、各アプリケーションに適した物理構成をとらすことができる。

こうしたデータベースの物理構造の動的な適應性に加えて、その論理構造自体も、アプリケーションの要求に合う様に適應できねばならない。従来のデータベース・システムでは、固定されたスキーマを通してのみデータ利用が可能であったが、今後、アプリケーションの多様化に対してスキーマ自身も動的に変わり得ることが必要である。このために、スキーマとデータの区別なくアクセス出来る言語が、ユーザに提供されねばならない。統合概念によって、論理的に集中された管理から、論理的冗長性を許すデータベースが必要になる。各アプリケーションは、自分に適した論理構造と、これに密着した物理構造を各々が持ち、アプリケーション間で厳しい一貫性が求められるデータは、集中管理されることになる。

研究開発目標としては、次の点がある。

- ① 物理リソース（例・disk , VLSI , video , disk・MSS）の物性の記述体系の確立。
- ② アプリケーションの要求記述体系の確立。
- ③ スキーマとデータを統一的にアクセス、及び記述できる言語の確立。特にプログラム言語とのunificationが必要である。
- ④ ②と①との動的なマッピング機構。
- ⑤ 共有データに対する有効な管理機構（例・同時実行制御、インテグリティ、一貫性、セキュリティ管理）。
- ⑥ 柔軟なヒューマン・インタフェース。図形、画像等の、アプリケーションが利用するデータ・オブジェクトでアクセス出来るインタフェース（例・SDMS , QBE）。
- ⑦ データ、スキーマの時間に基づいた管理（例・バージョン管理、ヒストリー管理）
- ⑧ 仮説的なDBS設定機能（例・シミュレーション）。

c) スキーマ設計支援システムの開発

関係モデルにおいて、概念間の依存関係を基にしたスキーマ設計法がこれまでも、数多く提案されている。本研究では、その上に、さらに a) での研究成果を盛り込んで、モデルの更新のためのメカニズムを組み込み、その変更を容易にするシステムを開発する。すなわち、スキーマ設計は、データベースの初期設定段階のみに止まらず、稼働段階においてもその変更をサポートする。

スキーマとしては、概念スキーマのみならず、外部スキーマの利用者による設定を支援するものとする。この中には、用語の定義機能を含み、自然言語による質問応答システムに接続し得るものとする。

d) データ蓄積支援システムの開発

データの蓄積は、生データの加工および、既存のデータベースへの格納に対する支援が必要である。

生データの加工の段階では、既存のデータベースに適合した形にデータを加工してやる必要がある。単位をそろえたり、形式を整えたりすることが必要となる。これらは、データベースに関する知識を基にした、知識利用の対話システムによって実現できる。

データベースへの格納の段階での最も大きな問題は、データベースの一貫性・無矛盾性の維持である。特に他のデータとの所定の依存関係が満足されているかどうかの、能率の良いチェックのアルゴリズムの開発が必要である。

分散データベース・システムを用いて、情報発生源で蓄積されたデータを、統合していくことも必要である。

e) 自然言語による Q A システムの開発

データベースのアプリケーションの初心者にとって、日常用いているコミュニケーション手段、即ち、自然言語によって、データベースシステムと対話出来ることが望ましい。初心者は自分が知ろうとしていることが整理されていなかったり、明らかでない。又、データベース・システムが、どのような情報を保有しているかも、十分に理解していない。自然言語による Q A システムは、この様なユーザをサポートすることを目的としている。

自然言語処理は、基礎ソフトウェア・システムの中の知的インタフェース・システムの研究成果を利用する。さらに、質問応答の基本的な枠組は、基本応用システムの中の質問応答システムの研究成果を利用する。本項目では、具体的事例について、専門用語を定め、柔軟な会話機能の実現を図る。すなわち、あいまいな問い合せを推測で補い、誤った問い合せに対しては助言を与え、さらに得られた答は質問者の意図に合うように要約、図示などの加工を加えて応答するなどの機能を追求する。

f) 分散データベース・システムの開発

アプリケーションの多様化と高度化とは、1つのデータベース・システムの枠を越えた、データベース利用を求めてきている。例えば意思決定、設計業務等において、種々の特性の異なったデータベースを統合的に利用できることが必要である。この統合は、各データベースの意味を、アプリケーションの目的に合う様に抽象化し、必要な仮想データ・オブジェクトを得ることによってなされる。各データベースのデータの意味の理解は、重要である。この抽象オブジェクトを通してのアクセスは、各データベースを統合する通信ネットワークを用いて、データベース・システム間での通信処理によって実行される。分散データベース・システムでは、こうした新たな、抽象レベルの設定に加えて、データをその発生源と処理が必要とする場所に物理的に配置することによる高データ獲得性をもたらすとともに、アプリケーションの構造にマッチした物理構成をとることもできる。

分散データベース・システムの開発内容としては、次の点がある。

- ① データベース・システム内の格納データの意味表現と理解機構・DD/D機能のKBS化。
- ② データベースの抽象化技法。
- ③ 有効な通信処理方式（例・サイト間でのリレーションのJoin方式）。
- ④ 有効な、同時実行制御方式。
- ⑤ データベース・システムの異種性（例・モデル、言語、値表現、DBMSの機能）の解決。
- ⑥ セキュリティ、プライバシーの制御方式。

g) 超文字データベース・システムの開発

従来のデータベース・システムが、文字及び、数値データの処理を行ってきたのに対して、今後、画像、図形、文章、音声、イメージ、アルゴリズムといった超文字データの処理要求が大きくなっていく。特にCAE/CADにおけるエンジニアリング・データベース・システムでは、ドキュメント作成のために種々の形態（例・設計図、文章、写真）の情報が必要となっている。

超文字データベース・システムの開発内容としては、次の点がある。

- ① 超文字データの記述系の確立。超文字データの属性記述及び、データ間の関係性。
- ② 超文字データの物理的格納方法とアクセス方法の確立。
- ③ 特徴パターン、抽出方法と照合方法の確立。

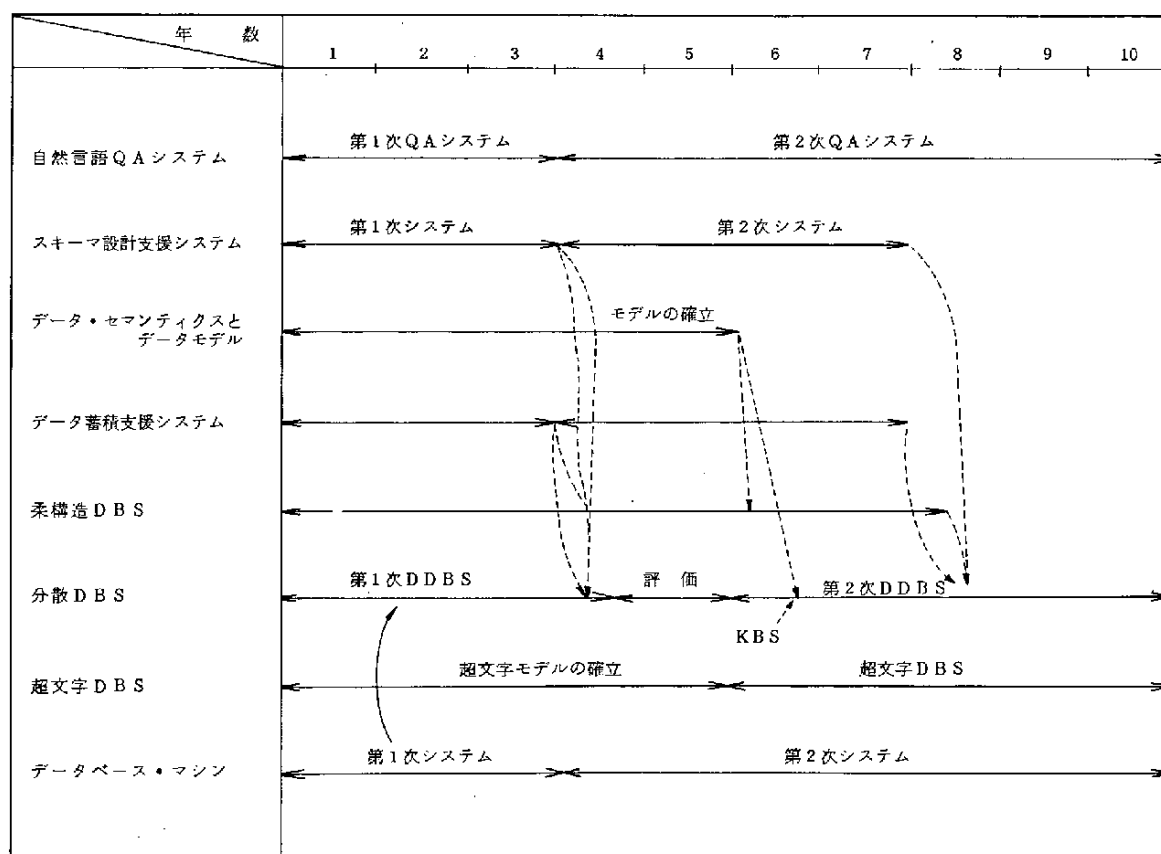
h) データベース・マシンの開発

データベース処理の特徴は、互いに関連し合った大量のデータに対して、必要なオペレーション（例・結合、更新）を行なうことである。このため、こうしたオペレーションを、

パフォーマンスよく実行する処理機構を備えたハードウェアが必要になる。VLSIと通信技術の進歩によって、通信ネットワークによる複数プロセッサが有機的に結合したシステムでの並列処理が基本となることから、分散データベース・システムにおける通信と処理の融合（例．ソーティングマシン，systolic array）が必要になる。

特に、データベース・マシンを実現するうえで、通信ネットワークが、パフォーマンス上主要な役割を演じる。データベース・マシンの機能としては、従来の検索機能に加えて、更新機能が重要となってくる。ディレクトリ管理、インテグリティ、セキュリティ、一致性の管理、同時実行制御等の有効な処理機構の実現が、今後の課題となる。

(2) 研究開発スケジュール



(3) 開発目標・仕様

a) データ意味論とデータ・モデルの研究

5年間で、データ意味論に基づいたデータ・モデルを確立する。このモデルは、データ更新及び、モデル自身の更新を、異常なく行なえるものである。このモデルは、スキーマ設計、データ蓄積、DDBS、柔構造DBS、データベース・マシン研究開発の基礎となる。又、このモデルは、図形・画像、アルゴリズムも部分的に扱えるものとする。

b) 柔構造データベース・システム

(1)のb)で述べた柔構造DBSを開発する。第1次DBSとして、既存モデルを基本としたシステムを開発する。(3年)。

第2次DBSとして、a)の新モデルに基づいたDBSを研究開発する。

c) スキーマ設計支援システムの開発

a)の新モデルに基づいたスキーマ設計技術を確立する。このスキーマは、モデル自身の更新機能によって、データベースのライフサイクルを通して、容易に変更できるものである。

d) データ蓄積支援システムの開発

(1)のd)を参照。

e) 自然言語に基づくQAシステムの開発

(1)のe)を参照。

f) 分散データベース・システムの開発

第1次DBS(5年)

ローカル・ネットワーク上に、リレーショナル・モデルに基づいて、DDBSを構築する。DBSは、個人用小型DBSを中心とし、大型DBSとしては、CODASYL型(CODAS78)を設ける。OISに対して適用し、フォーム処理を行なう。有効な同時実行制御方式の確立を行なう。図形、画像情報のコード化した処理を行なう。

第2次DBS

新しいデータモデルに基づいたデータベース統合技術を確立する。スキーマ設計支援システムによって、必要な情報を動的に生成し得るシステムである。

g) データベース・マシン

(1)のg)を参照。

2.7 開発支援技術

2.7.1 開発支援システム

〔1〕 概要

第5世代のコンピュータの研究開発は、研究面でも新しい方法論が必要となってくる。ハードウェア、ソフトウェア、全体システムのいずれの開発を行なうにしても、それに適した開発支援システムを前段階で準備することが必要である。ハードウェアでは、VLSI設計支援システム(VLSI-CAD)の開発が必要である。そのシステムは、プロジェクトの比較的早い段階で用意されなくてはならない。プロジェクトの第1ステップでは、システムはLSI技術で作られるが、第2ステップ、第3ステップではVLSIによる実現を目指している。そのためにも、第1ステップの終了時である3～4年後を目指して、VLSI-CADを開発する。

ソフトウェアの開発では、ソフトウェア開発支援システムが必要である。開発対象となるソフトウェアは、システムの階層の各レベルに互っている。よりレベルの低いソフトウェアは、ハードウェア化の対象となるものである。また、上位レベルのソフトウェアでは、高級言語プロセッサ、あるいは、知識ベース管理システムなどである。さらに、アーキテクチャの研究に関連して、上で述べたCADや各種のハードウェア・シミュレータなどの開発も対象となる。

知識ベースの開発は、非常にレベルの高いソフトウェアの開発と考えられるが、支援システムとしては、別に用意する。それによって、基本応用システムあるいは他の応用システムの開発がなされる。

以上2つの開発支援ソフトウェアを動かせる研究開発用パーソナル・コンピュータとして、関数型および論理型言語用の高級言語マシンを開発する。そのコンピュータは、研究者各個人が専有して使用するものである。

全体システムの開発を支援するシステムとして、コンピュータ・ネットワークが必要である。コンピュータ・ネットワークには、ローカル・ネットワークとグローバル・ネットワークがある。ローカル・ネットワークは、各種専用計算機を高速回線によって結合するもので、機能分散方式によってシステムを開発する際の核となる。また、グローバル・ネットワークは、分散データベースと共に、全国各地で研究開発されたユーティリティを相互に利用するための、資源共有に重点を置いたネットワークである。

上で述べた研究開発用パーソナル・コンピュータは、ローカルあるいはグローバル・ネットワークに結合されて使用される。全国各地で全く同じ研究開発用パーソナル・コンピュータを利用することによって、初めて研究成果の蓄積および相互利用が可能となるであろう。

これまでに述べてきた開発支援システムは、いずれも関連する研究開発課題の初期段階における開発目標とも一致している。すなわち、それらの各システムは、緊急に開発すべきシステムであると同時に、将来のより高度なシステムを作るためのプロトタイプとしての役目をも果

している。

〔2〕 必 要 性

前項の概要にも述べた通り、ここで開発しようとしている開発支援のための各種の道具は、第1に、今まで世の中に出回っている道具とはかなり異質のものである。第2に、それ自身が研究開発すべきシステムに組み込まれていくという性質を持っている。この、道具と開発目標が一体化する現象は、情報処理技術のもつ大きな特徴の1つである。すなわち、情報システムは、あらゆるシステムをその上で模擬でき、とくに開発目標となるシステム自身をその上でモデル化し、プロトタイプの作成が可能となる。また、そのような方法が、情報システムの研究開発に最も適していると言える。

そのような意味で、最も重要な道具は研究開発用パーソナル・コンピュータであり、その上で実現される種々の開発支援システムである。これらの道具は、安定して使い、かつ研究開発を十分に促進するものでなければならない。すなわち、知識情報処理技術の研究開発に適したコンピュータが必要である。

従来の汎用コンピュータを用いて、その上に、LISPあるいはPROLOGなどの、データ操作およびコントロールの柔軟な言語の処理系をソフトウェアで開発して道具として使うことも考えられるが、性能価格比を考えれば、専用ハードウェアを作った方がずっとよい。さらに本プロジェクトは最も先進的なものであるのでこれを推進するためには、日本全体に散らばっている第一線の研究者を官学民を問わず総動員することが是非必要である。そこで成果を集積するためには、研究に使用するコンピュータを同一化することが不可欠である。そうすることによって初めて、ソフトウェアの円滑な流通が可能になり、ソフトウェアを始めとする研究成果の蓄積を図ることが可能となる。

ソフトウェア関連の研究開発にとって上に述べた研究開発用パーソナル・コンピュータが不可欠であるのと同様、アーキテクチャ関連の研究開発にとって、ローカル・ネットワークの開発が不可欠である。ローカル・ネットワークは、新たに開発したハードウェアの検査、試験的使用、性能評価などを行なうための道具である。そこには、あらかじめ他のプロセッサや入出力機器が接続されており、新たに開発したハードウェアをそのネットワークに接続することによって、閉じた系が構成され、コンピュータとしてのテストが可能となる。

ローカル・ネットワークは、研究開発用パーソナル・コンピュータとは多少別の意味で、道具が研究開発目標と一体化する。すなわち、それは、ハードウェアとして、実際に目標システムの一部分に組み込まれることになる。

ローカル・ネットワークは、研究開発の各拠点に作られ、最初は、研究開発用パーソナル・コンピュータや、他の汎用コンピュータが接続されている。そして、それらのローカル・ネットワークを電電公社の公衆回線によって接続することによって、グローバル・ネットワークが

構成される。このグローバル・ネットワークは、遠隔地間での共同研究には必須である。同一のコンピュータを使って開発されたプログラムやデータベースは、このグローバル・ネットワークを通じて直ちに相互利用が可能となる。さらに、新たに開発された専用ハードウェアについても、遠隔地からその利用が可能となる。

ハードウェアやソフトウェアの開発に直接必要となるものとして、LSI-CADシステムや、ソフトウェア開発システム、知識ベース開発システムがあるが、これらの必要性は言うまでもない。また、それらのシステムも、関連する研究開発課題の初期の目標と位置づけられ、研究の一環として是非とも開発をしなければならないものである。

〔3〕 開 発 内 容

A. 開 発 項 目

(1) 支援ツールとしてのVLSI-CAD

① 第1ステップにおける作業と目標

VLSI-CADシステムは、プロジェクトの第1ステップの終了時点までに使用可能なものとなる必要があるが、これは、それ以前のLSI用のCADシステムとの連続性が必要である。

この点からも、第1ステップの初めにおいて備えるLSI-CADシステムは、その後続くVLSI-CADシステムの開発の基盤となるよう、その設備の構成、仕様を考慮する。

第1ステップにおけるCADシステムの第1の目的は、新アーキテクチャ実験機の試作や、開発支援ツールとなるパーソナル・コンピュータおよびこれらを接続するローカル・ネットワークの製作・拡充のために使用するカスタム化チップの製造をサポートすることである。

この時点では、まず、論理回路の記述から、そのマスクパターンを製作するためのツールを準備する。パターンを記述するための言語については、米国におけるCIF (Caltec Intermediate Form) の例に見られるような統一化、標準化をはかる。マスクパターンから実際のデバイスを製造するラインは、新たに建設することはせず、各メーカーの既存のラインを利用する。但し、VLSIのラインについては、各メーカーと協力して、建設することも考慮する。

第1ステップのCADシステムの第2の目的は、第2ステップのVLSI-CADシステム開発の基盤として、第2ステップに結びつくような論理回路についての種々の設計上の仕様記述言語 (論理機能の記述、電気的特性の記述、回路の幾何学的配置や構造の記述など) や、これらに基づくシミュレーションを行なうシステム、フリップフロップやレジスタ、さらには、より高機能モジュールを記述した一種のサブ・ルーチン・ライブラリ、テストデータを作り出すサブ・ルーチン・ライブラリなどを、順次開発して蓄積していくことである。

特に、論理回路の記述から、マスクパターンの記述への、変換用プログラム (一種のコンパイ

ラ)や、その際にボタン配置の最適化を行なうプログラム、ボタン配置が正しいか否かを検査するデザイン・ルール・チェッカーなどのソフトウェア群を開発し、まずLSIをベースとして行なっていくことが重要である。この作業は、開発支援用パーソナル・コンピュータを用いて行なうこととする。

② 第2ステップ以降の作業と目標

第2ステップでは、VLSI-CADシステムを整備する。これは、開発要素を多く含むが、1つの方針としては、第1ステップで蓄積したソフトウェア群や、使用経験をもととしてこれをVLSIの設計に利用できるように、整理体系化を行なうこと、さらに、VLSI特有の手法をこの体系に組み込み、機能の拡充をはかることである。特に、ユーザ・インタフェースについては、新アーキテクチャの研究者や、音声、図形処理の研究者が、実験用のカスタム化VLSIチップを作れるような高度なものを備えることが重要である。

このためには、設計対象とする回路の仕様記述言語のレベルの高度化、回路設計において要素となる細かな回路の記述のサブ・ルーチン・ライブラリ化、シミュレーションや評価システムの高度化を行なわねばならない。

これらを用いて、データフロー・マシンや、データベース・マシンの大規模実験機の試作、音声、図形処理などを行なう専用チップの開発などを、効果的にサポートする。

第3ステップでは、VLSIの集積度の向上や、種々の設計用ソフトウェア・システムの拡充に従い、VLSI-CADシステムとしての一応の完成を目指す。

目標としては、応用システムから、VLSIチップの設計に至る一連のコンピュータのソフトウェア、ハードウェアの設計プロセスを結ぶ、一環したシステムを目指す。

第2ステップ以降は、研究開発の重要テーマであるが、VLSI-CADシステムについては、常に開発支援ツールとしての側面があり、実際に使用できる道具としての機能も、重視して開発していく。

マスクパターン以降のファブリケーションのラインについては、その建設、運転、保守の経費や人的資源の問題があり、各メーカと協力しつつ実現し、運転や保守は、メーカ側に委託する等の開発体制も考慮に入れた整備計画を策定する必要がある。

(2) 研究開発用パーソナル・コンピュータ

本プロジェクトの早い段階で、研究開発用パーソナル・コンピュータを開発することが必要である。そのコンピュータは、TSS方式ではなく、個人専有型が望ましい。個人専有型にすることによって、オペレーティング・システムの構造が著しく簡単になり、ソフトウェアの開発が容易になる。個人専有型にすべき第2の理由は、入出力機能の充実を図りたいからである。そのためには、高解像度グラフィック・ディスプレイおよび図形入力装置を備えることが必須となるが、そのような機器を効率良く動かせるには、十分な画像メモリの用意と、画像データ

の高速転送機能が必要となる。TSSでは、十分な転送速度を得ることができない。

本パーソナル・コンピュータの用途は、それを本プロジェクトに関連する諸々のソフトウェアの開発と蓄積である。第1に、知識情報処理システムの開発に適していなければならない。すなわち、非決定的処理を含む問題解決型言語が高速に実行されることが必要である。第2に、アーキテクチャやソフトウェア・ツールの開発のために、ハードウェア・シミュレータや言語処理システムなどの開発も、その上でできることが必要である。第3に、十分に使い易いプログラミング・システムが実現できることが必要である。

このようなパーソナル・コンピュータの候補として、LISPマシン、PROLOGマシン、およびIOTAマシンの3つを考える。LISPマシンは、米国においてすでに商品化されており、システム記述言語としても、その汎用性は十分に確かめられている。日本においても、十分に使いこなされている。問題点は、非決定的処理およびデータ抽象化のハードウェアによるサポートがない点である。

PROLOGマシンは、未だ開発された例がないが、内部構造の簡潔さを考えても、LISPとあまり変わりがないので、現在の技術によってハードウェア化を行うことは、問題はない。また、PROLOG言語自体はヨーロッパを中心に多くの応用システムの記述に使われておりプログラミング・システムとしての有用性が実証されつつある。問題点としては、日本での使用経験の不足、シミュレータの記述可能性など汎用性に対する疑問、データベースやI/Oなどに対するインタフェースが不明確である点などである。また、基本的な事柄としては、非決定的処理が得意である反面、データ抽象化の実現が、これからの課題として残されている。

IOTA言語は、抽象データ型をサポートしている数少ない言語のひとつであり、日本で生まれた言語という意味でも、積極的に育て上げていきたい言語である。現在、京都大学で、言語プロセッサのみでなく、他のユーティリティを含めたプログラミング・システムとして開発されている。IOTA言語自身も、ハードウェア化の例はないが、上の2つの言語同様、言語の構造が簡潔であるので、技術的な問題はない。IOTA言語の問題点は、使用経験に乏しく、その汎用性および実用性が実証されていない点である。また、基本的問題として、PROLOGと逆に、非決定的処理をサポートしていない点である。

本コンピュータは、初めにも述べたようにプロジェクトの初期の段階での開発が必要である。そのために、設計は急がなければならない。遅くとも、第1年度内には、設計が終了しており次年度から開発ができることが必要である。

上に述べた3つのマシンは、実用性、非決定的処理、データ抽象化のサポートにおいて一長一短がある。これらの3つのマシンを並行して開発することができないのであれば、それら3つのマシンが容易に実現できるような中間言語を設定し、その中間言語を直接実行するマシンを開発するのが最も現実的な方法である。あるいは、それまでに、上記3つの言語を融合化し

て新しい言語が考えられれば最もよいが、そのような試みは、多くの時間を要するので、研究開発用パーソナル・コンピュータには間に合わないだろう。このような融合化は、第5世代コンピュータの開発それ自身にとって大きな研究テーマとして位置づけられている程である。

コンピュータ本体の他に、周辺機器として、高精度グラフィック端末、同入力装置、100MB位の差し替え可能なディスク装置は、是非必要である。これらのうち、グラフィック・ディスプレイ端末は、我が国での開発が遅れているので、開発すべきものの1つとしてあげておく。

本パーソナル・コンピュータは、本プロジェクトに参加する各研究者集団によって使用されるべきものであり、プロジェクトの規模に見合った台数を製作し、各集団に適宜配布することが望まれる。

(3) コンピュータ・ネットワーク

ローカル・ネットワークは、機能分散方式によるシステムの開発の要となる。機能分散方式とは、例えば、推論マシンと知識ベース・マシンは、それぞれ独立に開発して、最後に1つにまとめるといった方式である。この方式自身は、開発すべきシステムの能力・規模などによって、その妥当性が異なってくる。すべての機能をまとめて1つのマシンの中で実現できる場合には、効率の点から言えば機能分散方式とする意味はない。また、その接続方式が制限されることによって、かえってシステムの作製が困難となる場合もあり得る。しかしながら、たとえば図書館全体のデータより成るデータベースを考えると、それを各個別システム内に作る訳にはいかない。このように、大規模な専用マシンを必要とする場合には、機能分散方式が不可欠となる。

そのような極端な場合を想定しなくとも、機能分散方式は、システムの開発を容易にするという意味で大切な技術である。

ローカル・ネットワークの開発のためには、ハードウェアとしては、Ether-netあるいはChaus-Net程度の性能のものを開発する。そして、前項で述べた研究開発用パーソナル・コンピュータをそのネットワークにつなげて、分散システムのOS、交信用ソフトウェアを開発する。

グローバル・ネットワークについては、初期の段階では、電電公社の公衆回線を利用すればよい。将来は、より高速な回線網の安価な提供が期待される。

それらのネットワークに、本プロジェクトによって新たに開発が予定されているデータベース・マシンを接続し、分散データベースを作つて、研究成果の蓄積、相互利用を図ることが考えられている。

(4) ソフトウェア開発支援システム

本プロジェクトの研究開発に必要な種々のソフトウェアの作成を容易にするための支援システムを、研究開発用パーソナル・コンピュータ上に作り上げる。本開発支援システムは、L I

SP、PROLOGあるいはIOTA等の言語処理系と、それを取りまく種々のユーティリティとから成る。ユーティリティには、構造エディタ、画面エディタ、プリティ・プリンタなどの入出力を支援するもの、デバッガ、検証システムなどのプログラムの開発を支援するもの、および、出来上がったプログラムを管理するプログラム管理システムなどがある。

プログラムのソース・ファイルやオブジェクト・ファイルを管理するために、データベースは欠くことができない。それは研究開発用パーソナル・コンピュータのオペレーティング・システムの一部として作られる。

(5) 知識ベース開発支援システム

知識ベース開発支援システムは、知識利用システムのひな型を作り、その有効性を広く世に示すためにプロジェクトの調査段階ですでに必要とされる支援システムである。具体的には、EXPERT、EMYCIN、AGEなどのExpert Systemの開発が中心である。さらにこのシステムは、推論・問題解決システムおよび知識ベースのひな型を作り、その上で実現することになる。そして、プロジェクトの前期においては、応用システムの開発は、すべてこの段階で用意された知識ベース・システムを用いて行われる。そのため、ここで開発する知識ベース・システム自身、汎用のものでなくてはならない。なお、データベース・システムも同時に必要となるが、それは、(4)のソフトウェア開発支援システムでの開発要素に含まれているので、ここでは述べない。

B. 開発スケジュール

ここに集めた開発項目は、すべて前期で集中的に開発を行い、中期および後期は、それらの改良および維持・管理が主となる。

以下に、各項目についての開発スケジュールを表にして次ページに示す。

開発支援システムの開発スケジュール

開発項目	0	2	4	6	8	10
(1) 支援ツールとしての VLSI-CAD	LSI-CAD 開発・整備	⇒	VLSI-CAD (第1版)の 開発・実用化	⇒	VLSI-CAD (第2版)の 開発・実用化	
	VLSI設計用 ソフトウェア開発		実験機用 カスタムチップ製造		第5世代機用 カスタムチップ製造	設計用アーキテクチャ ベースの開発
(2) 研究開発用 パーソナル・コン ピュータ	実験機試作 ↓ 実用機設計	⇒	実用機開発	⇒	実用機 VLSI 化	
	高精度グラフィック 端末設計		開発			
(3) コンピュータ・ ネットワーク	ローカル・ネットワーク 設計	⇒	試作	⇒	開発	⇒ 高機能化
	グローバル・ネットワーク用 OS設計		⇒	同OS開発		
(4) ソフトウェア開発 支援システム	言語処理系開発 エディタ開発		⇒	新言語設計	⇒	新言語処理系開発
	ファイル・システム 開発		⇒	データベース開発		
(5) 知識ベース開発 支援システム	知識ベース・ システム開発		⇒	Expert System開発		

— 禁 無 断 転 載 —

昭和 56 年 6 月 発行

発行所 財団法人 日本情報処理開発協会
東京都港区芝公園 3-5-8
機械振興会館内
TEL (434) 8211 (大代表)

印刷所 株式会社 正文社
東京都文京区本郷 3-38-14
TEL (815) 7271

原本 (持出嚴禁)

受 付 No.

受付年月日

55. 6. 24

作 成 課