

(4) 逐次型推論マシン (SIM) の進捗報告

ICOT 第3研究室代理 内田俊一

ご紹介いただきました内田でございます。

私は、SIM計画のハードウェア及びソフトウェア両方の全般的な経過報告をいたしますが、主にハードウェアシステムについてお話ししたいと思います。同計画の詳細につきましては、ICOTから他に4名がこの会議で論文を発表する予定でございます。私どもはソフト及びハードウェアの開発の核言語としまして、論理的プログラミング言語を採用しましたので、この論理的プログラミングの能率的なプログラミング環境が重要となっております。そのため、逐次型推論マシンSIMと名付けました新しいコンピュータシステムの開発を行うことになりました。この開発の条件といたしましては、研究開発をごく短期間に行うこと、すなわちSIMシステムが中期に実用的な道具となるようにすること及びソフト担当者がこのハードウェアを使ってソフトのデバックを行えるように、できるだけ早くハードウェアシステムを作りあげることです。

ハードウェアシステムの構成を図1に示してあります。ハードのシステムは十分な処理スピードとメモリー容量を備えていなければなりません。標準的なソフトウェア研究開発用として、このシステムは少なくともDEC-10Prologと同スピードの処理能力と、DEC-10Prologよりもはるかに大きいメモリー容量を備えていなければならないと私どもは推算いたしました。しかしながら、より大きなソフトウェアの実験用には、それよりも5倍以上の処理スピードが必要ということになりました。加えて、ビッドマップ・ディスプレイ、図形I/O、さらに世界的なネットワークシステム

に接続可能なローカルエリア・ネットワークシステム(LAN)なども含めた革命的なマン・マシン・インタフェース装置が大変重要であると考えました。

ソフトウェアシステムには、我々が高度な対話型プログラミング環境を作ることができるような論理的プログラミング言語に基づいた強力なシステム記述言語がなければなりません。そしてこの種の環境には、特に和文入出力機能をも含めた高度なヒューマン・インタフェース及びI/Oメディア・サブシステムが必要です。こうした機能は全て柔軟で拡張性のある形でオペレーティングシステムに組み込まなければなりません。

これらの条件を満たすために、私どもはSIMのシステム全体をいくつかの部分に分けました。それぞれの部分の詳細についてお話しする前にSIMシステムの全体的な説明をしたいと思います。

まず、最初に、このシステムはハードウェアとソフトウェアに分けられます。ハードウェアとソフトウェアのインタフェースは主に機械語、KL0により行われます。ハードウェアシステムは、さらに基本ハードウェアシステムと拡張ハードウェアシステムに分かれています。基本ハードウェアシステムは、ローカルエリア・ネットワークシステムであるICOT-NetとPSI(パーソナルSIM)からなっています。拡張ハードウェアシステムは主に、PSIの後置プロセッサシステムとして接続するとして設計された高速プロセッサから成り、その他専用のI/O装置も含まれます。

SIMのソフトウェアシステムはExtended Self-contained Prolog 略してESPと名づけられたシステ

ム記述言語とSIMPOS と呼ばれるプログラミング及びオペレーティングシステムからなります。SIMPOSはオペレーティングシステムとプログラミングシステムの2つに分けられまして、プログラミングシステムはESPの言語プロセッサを含んでいます。

以上のソフトウェアシステムの他に、SIMのための開発支援システムを加えたいと思います。このシステムは、ファームウェア及びハードウェアの開発及びソフトウェアの開発を支援するために開発されました。このシステムは、マイクロアセンブラ及びマイクロシミュレータ、KL0アセンブラ及びリンカー、そしてSVP上で実行されるハードウェア・モニタリングシステムで構成されています。

さて、SIM計画がいくつかの重要な小部門から成り立っていることがわかりいただけだと思います。これから各部門の詳細、すなわち核言語(KL0)の設計と基本ハードウェアシステムである、PSI、ローカルエリアネットワーク、および拡張ハードウェアシステムなどについてお話ししたいと思います。

なお、主な部分は高速プロセッサシステムとSIMPOSです。

まず、最初の最も重要な作業はKL0の設計です。私どもが最初に直面した研究上の問題点は、KL0のレベルの選択でした。最終的にはDEC-10Prologと同レベルを選びました。そうすれば実行時間を短縮するために新しいファームウェア及びハードウェア機構を導入する余地があるからです。最初は、主にDEC-10Prologに基づいてKL0を設計しましたが、これをさらにハードウェア制御及びオペレーティングシステムを記述できるよう拡張しなければなりません。実際の拡張は、制御機構及びデータタイプに関して行われました。

こうした拡張の結果、組込述語数が増え、ハードウェア制御用の組込述語も数多く追加されました。KL0の設計は、1982年6月にICOT研究

所が設立された直後に始まり、1982年11月に最初の設計が完成しました。この設計はICOTの3名の研究員にまとめあげられたものであります。

KL0の設計と並行して基本ハードウェアの設計も開始されました。私どもは、PSIが30Kリプスを達成し、かつ、また大容量メモリをもつことを期待していました。この主な理由として、DEC-10Prolog利用者が、当時処理スピードには満足していたものの、メモリ容量に対する不満が多かったことがあげられています。私どもは、PSIを標準I/Oを備えた独立型スーパー・パーソナル・コンピュータにしたいと考えています。また、このシステムをこのプロジェクトの研究者の方々にも広く使っていただける標準的な装置にしたいと考え、コスト・パフォーマンスについても考慮しております。さらに開発期間が短いということも大きな問題となりました。かくしてPSIをバランスのとれた安定したシステムにするような研究開発戦略を選んだのであります。

PSIアーキテクチャとハードウェアの詳細につきましては、このあとでICOTの龍和男、横田実の両研究員が発表いたしますので、私は簡単な紹介だけにしたいと思います。私どもは、タグアーキテクチャを採用しておりまして、各メモセルは40ビットから成っており、そのうち8ビットはタグ、32ビットはデータ用です。また、マイクロプログラム制御を採用いたしまして、PSIは要求ベースページを行うスタック機構のハードウェアを備えています。メモリ・スペースは256の論理エリアに分かれ、各エリアがスラブ、またはスタックエリアとして使用できます。さらに多重プロセスサポート・ハードウェアも備えております。また、市販装置とも接続できるよう標準IEEEバスをつけました。また、ローカルエリアネットワーク用として10M bpsのネットワークを追加しました。

PSIのファームウェアシステムも大変重要な部分であります。なぜなら、KL0インタープリタ

すなわち、Prolog インタープリタはファームウェアによって実装されるからです。図に示されているように、KL0 を翻訳するために4つのスタックを使用しています。マイクロプログラムインタープリタは、マイクロコードユニフィケーションとバックトラッキング、拡張制御構造、実行最適化機能、ガーベジコレクション及びオペレーティングシステム機能サポート等の機能を備えています。現在、組込述語数は160以上で、マイクロプログラムのステップ数は約12kw/Kステップです。

処理速度を増加させるため、私どもはハードウェアにいくつかの専用回路を導入し、ほぼ30Kリプスを達成しました。ハードウェアの大まかな仕様は、つぎのとおりです。このマイクロプログラム記憶容量は64ビット、16kwでマシンサイクルは200 nsです。メモリは非常に大きなものを付けました。メモリ容量は40ビット、16MWです。256 Kビットのメモリチップを使用し、キャビネットに80Mバイト収容することができましたので、仮想記憶システムは付けませんでした。素子技術としては最も標準的なTTL及びNMOSを使用しました。

PSIの全体的なハードウェア機構は、他のマイクロプログラム制御のコンピュータと類似したものです。高速プロセッサがPSIのI/Oバスに接続されます。図1の右上に高速プロセッサモジュールが示されています。

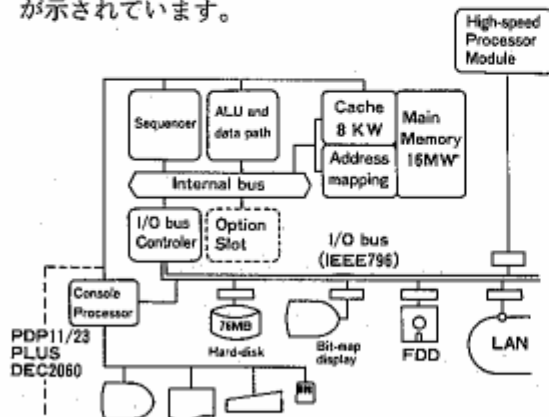


図1 PSI System Configuration

PSIの機能設計はICOTの6名の研究員により1982年8月頃開始され、その年の12月にはほぼ完了しました。詳細な設計段階の中ほどからは、メーカーも研究に加わり、1983年5月頃には詳細な設計も完了しました。そしてPSIの第1号機が1983年12月にICOTに運びこまれました。

いくつかの大きなソフトウェアの研究開発には、PSIより高速のマシンが必要であると思われました。

そこで私どもはできる限り高速のマシンを開発することに致しました。私どもはPSIより少なくとも5倍速いマシンを完成したいと考えています。

この高速プロセッサは、PSIの後置プロセッサとなるような設計されており、より最適化されたハードウェア及び構造を備えています。従って、このプロセッサの開発はPSIよりも研究指向の強いものでありますので、ゆえによりリスクの大きいものです。

この高速プロセッサの設計におきましては、PSIとは異なるアプローチをとることにしました。つまり、機械語の水準、翻訳方法、素子技術を変えたのです。機械語の水準はPSIのものよりは低く、これはデービット・ウォーレン博士及びティック博士の新しい提案に基づいたものです。これはスタック指向命令セットと呼ばれ、コンパイラの最適化はより難しいものとなります。

この機械語はKL0やESPをサポートできるように設計され、170の組込述語をもっています。

この翻訳機構は3つのスタックを使用する構造コピーに基づいています。

この高速プロセッサのハードウェア・アーキテクチャは、この機械語の実行を速めるように最適化されています。これもタグアーキテクチャを使用しましたが、各語は36ビットからなり、そのうち4ビットがタグ、32ビットがデータ用または7ビットがタグで29ビットがデータ用です。これは、マイクロプログラム制御も使用し、ハードウェア・メモリ管理システムもあります。メモリスペー

スは8つの論理エリアに分かれ、スーパーバイザ・プロセッサを経てPSIに接続されます。

このハードウェアシステムの概略を図2に示します。現在、開発は続行中ですが、同機は200Kリプスを達成できるであろうと考えております。すなわち、マイクロプログラム記憶容量は80ビット、11kw、マシンサイクル時間は100ns、メモリ容量はPSIよりずっと多い36ビット、64MWで素子技術はCMLとNMOSです。

この高速プロセッサの構造は、大まかに言って他のタグアーキテクチャ・マシンと類似していますが、細かい部分はもちろん特別に実行を速めるよう作られています。

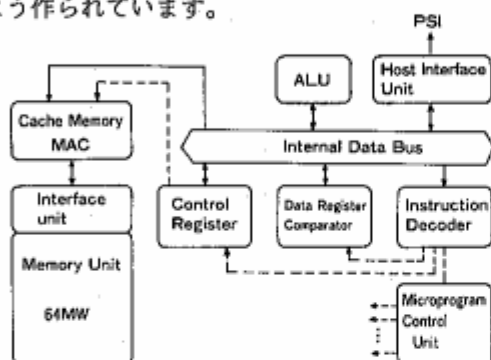


図2 High-speed Processor Module

このプロセッサの開発には未知の研究問題点が多く、機能設計には、1年以上費やさねばなりませんでしたが、開発はまだ継続中ですが、本年度内には完了する予定です。

ソフトウェアシステムのSIMPOSに関しましては、私の発表のあと、ICOTの横井研究室長が詳細を発表いたしますので、私は簡単な紹介をしたいと思えます。

私どもはスーパー・パーソナル・コンピュータ用のトータルプログラミング及びオペレーティングシステムの開発を目的として参りました。もちろん、開発では困難なこともありました。と申しますのは、SIMPOSの開発にあたりましては、全く新しいマシンを用いて開発しなければならず、全く新しい言語で記述しなければならなかったから

です。

設計の初期段階では機械語すらも確定されてはいませんでした。つまり、私どもは皆、プログラマーとしては経験不足だったのです。それに開発期間も大変限定されておりましたし、また、現在もそうであるからです。

SIMOSを作成するには効率的なシステム記述言語が必要です。このシステム記述言語ESPの詳細については、ICOTの近山研究員があとで発表しますので、ここでは簡単な紹介だけをいたします。この言語の設計はSIMOSの機能設計と平行して行われました。設計の中で最も重要な問題点は、システム言語のESPにどのようにモジュール化機能を導入するかということでした。私どもは実際にクラスシステムとマルチプル・インヘリタンス機構を使用した対象指向概念を採用することにしました。これは一部には、SIMPOSのモジュール構造も対象指向概念に基づいていることによるものです。ESPの基本言語はもちろん論理型プログラミング言語のKL0です。私どもは、強力なマクロ機能を付け加えました。

ですからESPの全ソースプログラムはKL0にコンパイルされ、実行時支援ルーチンを使って実行されます。また、DEC-10のプログラムのほとんどが、ESPのプログラムに簡単に変換できるように設計を考えました。このようにして、ESPは、現在、標準的なシステム記述言語となっています。

SIMPOSの主な機構は、つぎのとおり、要約することができます。SIMPOSは、マルチプロセスサポートを備えたパーソナル・オペレーティングシステムです。和文I/Oを含む対話型プログラムシステムに基づいたマルチウィンドウを持ち、またLAN用ネットワーク・インタフェースサブシステムも備えています。SIMPOSは対象指向モジュラー構造をもち、ソフトウェアモジュールは全てESPで書かれています。SIMPOSのサイズを正確に示すのは大変難しいのですが、ESPプログラムの行数にして、現在、約90Kラインです。

SIMPOSの機能設計は、1982年8月頃から約10名のICOTの研究者によって開始され、機能仕様は1983年3月頃集成されました。詳細な設計段階でメーカー側研究者がSIMPOSの開発チームに加わりました。現在でもSIMPOSは約30名の研究者によって開発が続行中ですが、1985年3月頃までには完了する見込みです。

SIMプロジェクトは、まだ続行中で完了していないことを私は強調したいと思います。しかし、ゴールは明らかに真近にあります。SIMのトータルシステム、すなわち、SIMプロジェクトの現在の開発状況は、FGCS'84の会場ロビーに設置致しましたデモ用システムでご覧いただけるように、PSIとICOT-Netは、ほとんど完成しています。SIMPOSに関して、このオペレーティングシステムは実際に使用可能です。プログラミングシステムはまだ開発中ですが、1984年12月には使用可能となるでしょう。拡張ハードウェアはもちろんまだ開発中です。

私どもは、また全体の評価を完了したわけではありませんが、PSIのハードウェア及びファームウェアは安定かつ使用可能で、30Kリブスを達成できると確信しております。しかし、できあがったばかりのコンピュータシステムの例にもれず、PSIも種々改良が必要で、そのいくつかは既に行われました。その一例は、オペレーティングシステムのいくつかの部分です。まず最初に、オペレーティングシステムはソフトウェアモジュール全てを安全に作りあげることができるように設計されました。しかし、現在はスピードを増すためにモジュールのいくつかを書きかえる段階にあり、既にオペレーティングシステムのモジュールのいくつかを書きかえました。

ファームウェアシステムに関して、何点かの改良を行いました。重要な改良点はファームウェアシステムの拡張とKL0の拡張です。SIMPOS開発の経験を通して、私どもは論理的プログラミング機能と同様、ソフトウェア製造に対象指向モ

ジュラー構造の有効性を確任いたしました。従いまして私どもは、KL0の仕様及びファームウェアの拡張を行い、これらの中にESP実行時サポートルーチンを含むことができるようにしました。ですからKL0の仕様は、DEC-10PrologではなくESPを基本にしているということができようかと思えます。ハードウェアに関しましてはI/Oインタフェースの改良を行うつもりです。

このSIMプロジェクトに関連した今後の計画といたしましては、ハードウェア、ファームウェア、ソフトウェアシステムの評価及び改良の完了、LSI、VLSIを使用した小型ハードウェアの開発、ICOT-Netの拡張、より効率のよい、また使いやすいプログラミング環境のためのSIMPOSの拡張などがあります。これに加えて、並列ソフトウェア開発のためにPSIを使用し、多重PSIシステムファームウェアによるKL1サポートを付け加える予定です。

私の発表を終了するにあたりまして、私はSIMシステムが中期においても使用可能なシステムであることを再度強調したいと思います。ESPもシステム記述言語として大変有効であることが証明されまして、現在、FGCSプロジェクトの標準言語ともなっています。SIMPOSとESPの機構はよく構成されていますので、アーキテクト及びアーキテクチャの研究者が実行速度を速める新しい機構を導入することを可能にするものと思います。

最後に本会場ロビーに展示してあるPSIとSIMPOSのデモマシンの見学を、どうぞお楽しみ下さい。

このデモマシンは皆様のために特に用意されたものでございます。

ご静聴ありがとうございました。