

12 SIMD型並列計算機 SM-1

湯浅 太一

yuasa@kuis.kyoto-u.ac.jp

京都大学大学院情報学研究科 教授

当時：豊橋技術科学大学工学部 助教授



開発の経緯

1990年から1992年にかけて、筆者が豊橋技術科学大学に在籍していたときに開発したSIMD (Single-Instruction, Multiple-Data) 型並列計算機である^{1), 2)}。計算機の性能を飛躍的に向上させるための技術として、SIMD型並列計算機が注目されていた時代であり、いくつかの商用機も存在した。筆者らのグループでも、これらの商用機を使って、SIMD計算のさまざまな実験をしたいと考えていたが、高価であるうえに、詳細を調べていくうちに、プログラミング上の制約が厳しいことが分かってきた。そこで、価格的にも仕様のにも手軽にSIMD計算の実験・研究ができるマシンを開発しようということになった。

開発にあたっては、住友金属工業から全面的な協力をいただいた。マシン名の「SM」は、SIMDマシンのSとMをとったものであると同時に、住友金属工業の英語名Sumitomo Metalの頭文字でもある。マシン名を決めたのは、開発作業がかなり進んだ後だった。寝食を忘れて作業をしていたので、サドマドの意味もなくてない。開発チームのメイリングリストの名前は、sm-clubであった。

ハードウェアの基本設計は、1990年4月に開始した。竹岡尚三氏(現在(株)アックス社長)と松田元彦氏(現在、住友金属工業(株)総合研究所)と筆者の3人が技科大の1室に閉じこもって、ホワイトボードに書き込んだ設計図を毎日のように書き換えていった。同年8月に設計がほぼ完了し、ホワイトボードのコピーをとって、VLSIチップの設計、マイクロ・コード作成、シミュレータ開発などの分担作業を開始した。ハードウェアの詳細設計と製作はプロに任せて、大学ではソフトウェ

アの開発を並行して行った。ハードウェアは1992年11月に完成し、シミュレータを使って並行開発していたソフトウェアを実機に移行し、システム全体が完成したのが1993年春であった。

ハードウェア構成

SM-1のハードウェア構成を図-1に示す。並列計算を行う実体がPE (Processing Element) である。各PEは、8ビット演算を行うALU、128バイトの大容量汎用レジスタ、柔軟なSIMD計算機能を提供するさまざまな専用レジスタなどから構成されている。実装上は、4台のPEが1チップに収められ、約1メートル四方のボード1枚に16枚のチップが乗り、全部で16枚のボードが、約1メートル四方の立方体の筐体に収められている(図-2)。計1,024台のPEが並列に動作する。各PEチップには4MBのDRAMが付随しており、PEあたり1MBの局所メモリとして使われる。SM-1全体では、1GBの大容量である。

並列計算部を制御するフロントエンドには、SUN 4を使用した。当時のSPARCチップには、コプロセッサインタフェースを2つ備えているものがあった。片方は外付けのFPUのために使用されていたが、もう片方は未使用であった。SPARCの命令セットには、両方のコプロセッサのためのスペースが用意されていたが、通常は未使用のコプロセッサ命令を発行することはない。そこで、専用のコプロセッサを空いている方のインタフェースに接続し、これに対して発行された命令をSIMD計算命令と解釈して、並列計算部に送るよう改造した。

SIMD計算命令は一種のマクロ命令であり、これを受け取ったシーケンサが、対応するマイクロ命令列に置

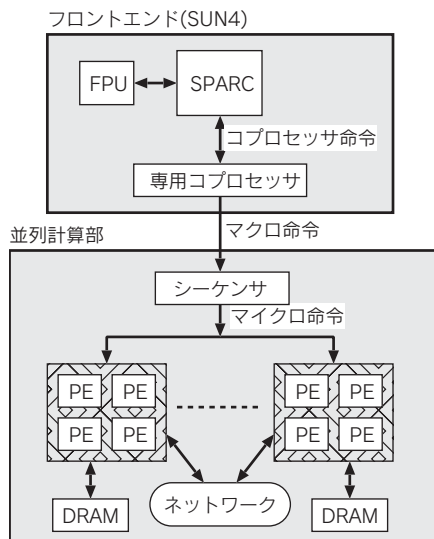


図-1 ELIS-8100

き換えて、マイクロ命令を1つずつ全PEに送り、これを全PEが並列に実行する。マイクロコードを入れ換えることによって、多様な並列計算機として動作するが、標準的なマイクロコードセットでは、フロントエンドのSPARCと類似の32ビットのマクロ命令セットを提供した。たとえば、並列計算のための加算命令は

```
add %p1, %p2, %p3
```

と記述する。ここで%p1等は、各PEの持っている汎用レジスタの指定である。

ソフトウェア

並列計算部は、フロントエンドから見れば単なるコプロセッサなので、OSを開発する必要がなかった。既存のSUN 4のUnix OSをまったく変更することなく、並列計算機システムが構築できたのである。Unixが提供するリンカ、実行時環境、ライブラリなどもそのまま利用できた。基本的なシステムソフトウェアを開発を要したのは、専用アセンブラおよび高級言語の処理系だけである。

専用アセンブラは、通常のSPARC命令とSIMD命令とを分別し、SIMD命令であればコプロセッサ用のマクロ命令に変換する。SPARCのRISC特性のために、SPARC命令は、単純な変換テーブルを用意すれば簡単に変換できる。専用アセンブラは簡単にできてしまった。

次に必要だったのが、比較的低水準の高級言語である。この言語は、プログラム記述言語であると同時に、他のプログラミング言語処理系のターゲットとしても使われる。となると、ほぼ必然的にC言語がベースになる。設計した言語を「並C」(「なみC」と読む)と呼んでいる。これは標準のC言語に、SIMD型並列実行を明示



図-2 SM-1の概観

的に指定するものである。「並C」のコンパイラは、Gnu Cコンパイラを改造した本格的なものであり、開発したソフトウェアの中で、最も時間と労力を要した。

前述のように、SM-1のソフトウェアは、ハードウェアの開発と並行して行った。これを可能にしたのが、基本設計の完成直後に開発したシミュレータである。これは、並列実行部の動作を機械語レベルで模擬実行し、実機のフロントエンドに使ったのと同じモデルのSUN 4の上で動作した。SIMD命令が発行されるとトラップをかけ、命令に従って、全PEの状態を変化させてリターンする、という簡単な仕組みであったが、実機とほとんど同様に動作した。このために、シミュレータから実機への移行は、意外なほどあっけなく終わってしまった。

その後

試作機は2台製作した。1台は技科大で画像処理や素因数分解、Kyoto Common Lispを拡張した並列Lispを使った数式処理などの実験に用いられた。もう1台は慶應義塾大学の安村通晃氏のグループが並列Fortranを開発するために利用された。

SM-1は、整数演算や記号処理の分野では当初の目標に近い性能を発揮した。たとえば32ビットの加算なら、PEごとに毎秒860K回の演算が可能であり、システム全体では毎秒860M回となる。当時の計算機としてはかなりの性能である。しかし、浮動小数点演算は専用の演算装置を持たず、マイクロコードで実装されていたために、性能が低かった。開発当初は、SM-1を商品化することも考えていたが、主要なベンチマークテストでは、浮動小数点演算の性能のために、汎用CPUに追いつけず、商品化は断念した。

参考文献

- 1) 松田元彦, 湯浅太一: SIMD型超並列計算機SM-1 (仮称) の概要, 情報処理学会計算機アーキテクチャ研究会95-17, pp.127-137 (1992).
- 2) Yuasa, T., Matsuda, M. and Kijima, T.: SM-1 and Its Language Systems, Parallel Languages and Compiler Research in Japan, Kluwer Academic Publishers, pp.349-376 (1995).

(平成13年10月16日受付)

