

Concurrent Pascal のインタープリタ

およびカーネルの製作

松方 純, 田中 英彦, 元岡 達

(東京大学 工学部)

5J-5

はじめに

マルチ・プロセッサ・システム PPS-1⁽¹⁾のマイクロ・プログラムによる Concurrent Pascal マシンについて報告する。

PPS-1

PPS-1 は、図 1 のようなマルチ・プロセッサ・システムである。各プロセッサ・ユニットは、マイクロ・プログラム方式であり、マイクロ命令は、24ビット中のいわゆる縦型マイクロ命令である。大半の命令は、一命令あたり440nsで処理される。制御記憶は書きかえ可能で、容量は、8kW (1W=24ビット) ずつである。プロセッサ・ユニットは、三台あるが、この報告の範囲では、そのうち一台のみを使用している。すなわち、PPS-1 をシングル・プロセッサとして利用している。

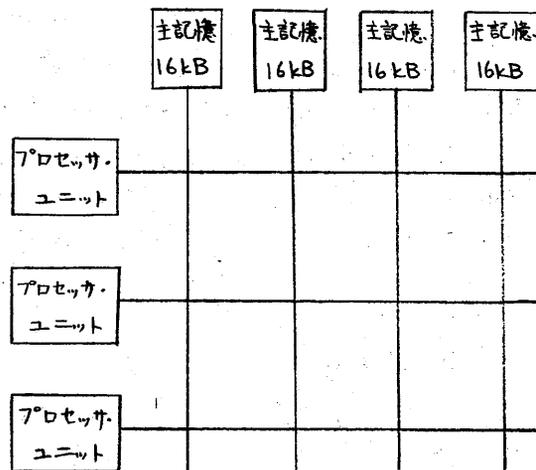


図 1 PPS-1 の構成

Concurrent Pascal

Brinch Hansen は、OS 記述用の高級言語として、Concurrent Pascal を作成した。⁽²⁾ Concurrent Pascal は、Wirth の Pascal の拡張であり、特徴としては、並行して動作する複数のプロセスからプログラムが構成されること、抽象データ型であるクラス、モニターが導入されたことなどが挙げられる。

Concurrent Pascal マシン

Brinch Hansen は、PDP-11/45 によってシミュレートされる仮想計算機 Concurrent Pascal マシン上に、Concurrent Pascal を実装した。⁽²⁾

Concurrent Pascal マシンは、インタープリタとカーネルの二つの部分からなる。インタープリタは、Concurrent Pascal コンプाइラによって生成される Concurrent Pascal マシンの命令 virtual code を実行する仮想スタック・マシンである。このインタープリタをプロセス毎に時分割で使用することにより、プロセスの並行動作を実現している。カーネルは、I/O 等のハードウェアに密着した機能の提供とインタープリタの時分割の制御などを行なう。

PPS-1 上の Concurrent Pascal マシン

Concurrent Pascal マシンのマイクロ・プログラムによる実現の方法として次のような方法が挙げられる。

(1) PDP-11/45 をマイクロ・プログラムでエミュレートする。

(2) Concurrent Pascal マシンを直接マイクロ・プログラムでエミュレートする。

我々の研究室では、既に(1)の方法により Concurrent Pascal を PPS-1 上
に実現した。⁽³⁾ さらに、インタープリタの一部を直接マイクロ・プログラムでエ
ミュレートするように改良された。その結果、処理速度が相当速くなった。⁽³⁾

現在、(2)の方法、すなわち、Concurrent Pascal マシンを直接エミュレート
するマイクロ・プログラムの作成を行なっている。PDP-11/45 に縛られない
ので、種々の効率化が容易になった。たとえば、PDP-11/45 において、主記
憶のアクセスの度に使用される仮想アドレッシング機構のハードウェアのエミュ
レーションを省くことは、処理速度の向上に相当寄与する。表 1 は、PPS-1
上の Concurrent Pascal マシンの各種の実装法における virtual code の処理
速度を比較したものである。

現在作成中の Concurrent Pascal マシンのマイクロ・プログラムの大きさは、確
定助ではないが、4kW (1W = 24ビット)程度になる予定である。これは、PDP-
11/45 のエミュレータの大きさ(約 4kW)と同じくらいである。

おわりに

今後の課題としては、PPS-1 が、マルチ・プロセッサ・システムであるこ
とを活かし、複数のプロセッサ上で動作する Concurrent Pascal マシンの実現等が
興味深いと思われる。

参考文献

- (1) 元岡, 山室: ポリプロセッサシステム PPS-1, 情報処理 16, 10, pp. 906-912, 1975.
- (2) Brinch Hansen: The Architecture of Concurrent Programs, Prentice-Hall, 1977.
- (3) 埴淵: PPS-1 ファームウェア技術による PASCAL, SOLO システムの移行, 昭和 52 年電子通信学会講演論文集.

実装方法 virtual code	PDP-11/45 エミュレータ上 (1)の方法)	一部を直接エミュレート	全体をエミュレート (2)の方法)
push constant address	390 μ s	97 μ s	15 μ s
push local address	287 μ s	58 μ s	11 μ s
push local variable	344 μ s	83 μ s	11 μ s
push indirect	224 μ s	82 μ s	9 μ s
push byte	300 μ s	84 μ s	10 μ s
range	379 μ s	58 μ s	15 μ s
copy byte	236 μ s	81 μ s	10 μ s
copy word	217 μ s	82 μ s	10 μ s
call	371 μ s	82 μ s	10 μ s
call system	390 μ s	131 μ s	15 μ s
enter	1053 μ s	132 μ s	26 μ s
exit	605 μ s	60 μ s	12 μ s

表 1 PPS-1 上の Concurrent Pascal マシンにおける
virtual code の処理時間。