

373

PPS-1ファームウェア技術による
PASCAL・SOLOシステムの移行

植 淵 肇 田中英彦 元岡達
(富士通株式会社) (東京大学)

1. はじめに

OS記述用言語に対する各方面の期待は日増しに高まり、いろいろなアプローチが試みられている。その1つに、1975年カリフォルニア工科大学の Per Brinch Hansen によって開発されたOS記述用言語、シーケンシャルPASCAL、コンカレントPASCALがある。PDP 1/45計算機にインプリメントされたオペレーティングシステム「SOLO」は両言語を96%記述されている。一方、ファームウェア技術の進歩にともない、ハードウェア・ソフトウェアのいかに問わす、あらゆるもののファームウェア化が進んでいる。本論文は、PPS-1にSOLOシステムを移行する手段として、PPS-1・OSの制御下に、1プロセスとしてPDP 1/45計算機のエミュレータを動作させた。さらに、SOLOを部分的にファームウェア化して行き、どのくらい処理効果があるかを調らべたのを報告する。

2. PPS-1上でのSOLOシステム

動的マイクロプログラム制御可能な多重プロセッサシステムPPS-1上でのSOLOシステム構成を図1に示す。エミュレータへの入力はVirtual Code (以下VC) のインタプリタなどからなるSOLOのKERNELである。エミュレータはメッセージ通信によりPPS-1 OSと交信できる。SOLOが用いるディスクには、PPS-1の仮想空間(8MB)の一部を用い、PPS-1 PAGERによりサポートされる。したがってエミュレータからみれば直接入出力装置を制御する必要はなく、PPS-1・OSの入出力制御モジュールが一括して処理している。これによりエミュレータの作成が非常に容易になる。エミュレータが動作する場合、VCからみれば二段階のインタプリタで処理されることになる。図1の構成で、PPS-1・OSのスケジューラによりプロセスが割当られたプロセッサがSOLOシステムの仮想マシンになる。

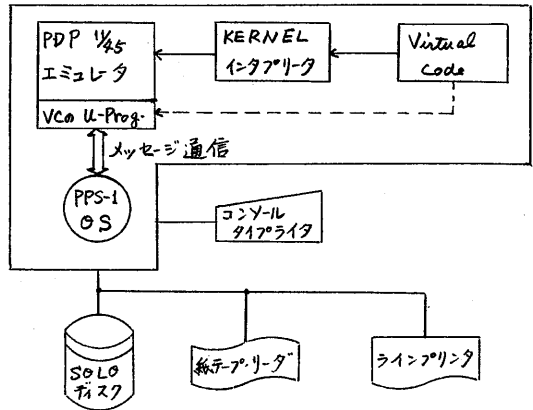


図1. PPS-1上でのSOLOシステム

命令	速度(us)	命令	速度(us)
ADD	49	MFPI	55
ASL	36	MUL	99
BEQ	5	ADDD	166

表1. エミュレータによる処理速度

VC	エミュレータ	ファームウェア化	VC	エミュレータ	ファームウェア化
02	390	97	90	1053	132
0A	344	83	98	1211	132
10	300	84	9C	1053	132
1E	379	58	9E	605	60

(ms)

表2. VCのエミュレータとファームウェア化の速度比較

3. エミュレータの実行速度

PDP 1/45エミュレータによるPDP 1/45の機械命令の処理速度の一部を表1に示す。これらの命令は、R-R命令であり、実行アドレスを求める必要のある命令ではその判定に8μs~46μsが加わることになる。

4. 部分的なファームウェア化

SOLOを構成しているVCをエミュレータを通さず、直接マイクロプログラムで処理することも考える。この場合、PPS-1からみればVCが一機械語となる。VCは全体で112個あるが使用頻度の多いもの19個を選別し、ファームウェア化した。VCをエミュレータで処理した場合と、直接ファームウェア化した場合の処理速度比較を表2に示す。この表からみると約3倍~10倍程度、処理速度が向上している。

5. 結論

PPS-1・OSの制御下の1プロセスとしてPDP 1/45計算機のエミュレータを約4Kステップで作成できた。また310ステップ程度のマイクロプログラムを追加することで処理速度の面で、約3倍という結果を得た。完全なファームウェア化を行なってもSOLOシステムの場合、改善率はこれ以上余り向上しない。

参考文献) Per Brinch Hansen, "Concurrent PASCAL Implementation Notes, Information Science California Institute of Technology, January 1976"