

## 5C-9

## 浄書システム第3版の実現

笹川重和, 門奈敦, 早川栄一, 並木美太郎, 高橋延匡

(東京農工大学)

## 1. はじめに

我々の研究室では、ソフトウェア文書をはじめとする日本語文書出力の研究として、浄書 (JOSHO: Japanese Output Server with Hospitality) [1]の研究を行っている。浄書では、

## (1) 知的組版の実現

(2) コンピュータサイエンスの文書出力、ソフトウェア文書出力を目指している。

これに対し、我々は浄書用のプリントサーバとして、第2版浄書システム[2][3]を6年前に作成したが、

- ・ MC68000 を中心としたホスト計算機が、より高度な処理を実現するには性能不足である
  - ・ LBP エンジンの耐用年数を越えている
- といった問題が生じてきたことから、浄書システム第3版の構築を行うこととした。

## 2. 浄書システム第3版の構築目標

我々は、浄書システム第3版を構築する上で、次の目標を設定した。

## (1) DTP システムのソフトウェア/ハードウェアアーキテクチャの研究を行う

我々の研究室では、次のような文書を出力したいという要求がある。

- ・ データ構造の図が付加されたプログラムリスト
- ・ 実行結果の図表、画面が付加されたプログラムリスト
- ・ タブレット入力による手書き図形の入った文書

このようなマルチメディア文書が容易に扱え、しかもこのような高度な処理を高速に行えるソフトウェア/ハードウェアの体系を研究する。

## (2) より高解像度の LBP を利用する

高品質な文書出力を得るためには、より高解像度の LBP を利用することが必要である。また、このような高解像度の LBP を使用して高速な文書出力を得るための方法を研究する。

## (3) 研究室内の分散処理システム中のプリントサーバとする

我々は、イーサネットを使用した研究室内の分散環境を構築することを考えている。浄書システム第3版は、このネットワーク中のサーバマシンとして構築する。このため、サーバマシンとしての実使用に耐え得るシステム構成を目指す。さらには、実使用から得られる経験をもとに浄書の研究にフィードバックさせる。

## 3. 浄書システム第3版の構成

## 3.1 ハードウェアの構成

我々は上で述べた目標に従い、高解像度 LBP エンジン、およびそのホスト計算機を導入した。ハードウェアアーキテクチャの研究を行うといった目標から、ホスト計算機には、ハードウェアの自作、および拡張が比較的容易に行える VME システムを選択した。この浄書システム第3版のハードウェア構成は、次のようなものである。

- ・ LBP エンジン
  - SL-2000 (解像度 400 DPI)
- ・ ホスト計算機
  - VME システム (CPU MC68030, クロック周波数 25MHz)
  - フレームメモリ : 8M バイト
  - ハードディスク : 300M バイト
  - イーサネットコントローラ
- ・ LBP インタフェースボード
  - 自作

## 3.2 ソフトウェアの構成

浄書システム第2版では、システムソフトウェアの構成を、ハイバ OS 「江戸」[4]と OS/omicon V2 [5]の構成としていた。この構成においては、時間制約の厳しいデバイスは、ハイバ OS に組み入れる方針であった。しかし、新しいデバイスを組み込むためには、デバイスドライバをアセンブラで記述する必要がある、ハイバ OS へのリンク作業が必要であるなどの問題点があり、システムの変更作業が困難であった。

浄書システム第3版では、フォントのフレームメモリへの描画、アウトラインフォントの管理などを行うシステムソフトウェアの組み込み実験を行いたい。このため、システムの変更が容易に行える必要がある。また、このような実験を効率よく行うためには、ソフトウェアの保守性が重要であると考えた。これらの問題を解決するために、我々は浄書システム第3版において、ハイバ OS に代わるソフトウェアとして、制御用 OS 「OS/礎」[6]

を作成した。OS/礎は、デバイスドライバや OS/omicron を OS/礎上のプロセスとして実行させ、プロセスに対する割込みの抽象化とプロセス間の通信を提供している。この構成を図1に示す。

我々は、このようなシステムソフトウェアの構成で、次のデバイスドライバを作成した。

- ・フレームメモリ転送のための LBP インタフェースドライバ
- ・端末コンソール用のシリアルドライバ
- ・SCSI ハードディスクドライバ
- ・フロッピーディスクドライバ

また、これらのデバイスドライバを作成して、次のような利点を確認した。

#### (1) 割込み処理を容易に記述することができる

割込み処理を記述するための OS/礎の SVC が用意されており、スタックの状態や実行環境などを考慮する必要がない。このため、言語Cによって割込み処理を記述でき、デバイスドライバの作成を容易に行うことができた。

また、言語Cを用いてデバイスドライバを作成できるため、ソフトウェアの保守性が向上した。

#### (2) デバイスドライバの組み込みが容易に行える

浄書システム第2版では、新しいデバイスドライバの組み込みはハイパ OS へのリンク作業が必要であったが、この構成により、デバイスドライバを OS/礎上で動作する実行用ロードモジュールとして記述でき、システムの起動時に、組み込みたいデバイスドライバを選択することができる。このため、デバイスドライバの変更、更新を容易に行うことができた。

浄書システム第3版のシステムソフトウェアを、上で述べたような構成で実現した。また、フレームメモリへの文字、図形の描画を管理する OS/omicron の OS 拡張部のソフトウェアである FMH (Frame Memory Handler) は、文字描画をドットフォントからアウトラインフォントに拡張し、浄書システム第2版から移植を行った。

これにより、現在、和欧混合組版、プログラムリスティングツール、ワードプロセッサの文書出力ツールが動作している。

#### 4. 今後の課題

以上のような構成で、浄書システム第3版の第1段階としての実現を行い、新たな浄書の研究のための基盤を構築した。現在のシステムの問題点としては、次のものがあげられる。

##### (1) より高速な文書出力が必要である

LBP エンジンが高解像度化したことにより、フレームメモリへの描画処理が重く、満足できる出力速度が得ら

れていない。描画処理を高速化するための描画プロセッサの採用、複数のフレームメモリを利用した描画処理の並列化などが必要である。

##### (2) フォントの管理が必要である

現在、フォントはアウトラインフォントから逐次展開しているが、文字の描画処理を高速化するためには、アウトラインフォントだけでなく、ドットフォントなどを混在して扱うことや、フォントのキャッシング処理が必要である。これらの高速化のための処理と複数のフォントを管理するためのフォント管理ソフトウェアが必要である。

#### 参考文献

- [1] N.Takahashi, O.Atoda, T.Manabe, Y.Ikeda, and M.Nakagawa: 浄書: Japanese Output Server with HOspitality, Proc. of ICTP'83, Tokyo, pp.29-34(1983-10).
- [2] 里山元章, 他: "文書の論理構造を備えた日本語清書システム「浄書」の設計と実現", 情報処理学会論文誌, Vol.30, No.9, pp.1126-1134(1989).
- [3] 関口治, 中川正樹, 高橋延匡: インテリジェント機器のシステム開発に関する一考察 -レーザビーム・プリンタのインテリジェント化を例として-, 情報処理学会第30回全国大会, 6S-4, pp.671-372(昭和60年前期).
- [4] 岡野裕之, 堀素史, 中川正樹, 高橋延匡: 多重OS江戸の設計と実現, 情報処理学会論文誌, Vol.30, No.8, pp.1012-1023(1989).
- [5] 鈴木茂夫, 他: OS/omicron 第2版の実現と評価, オペレーティングシステム研究会, 42-1, 1989.2.
- [6] 並木美太郎, 他: ビルディング・ブロック・システムのための共有ソフトウェアパス~礎~, 情報処理学会研究報告, Vol.91, No.63, 51-2(1991)

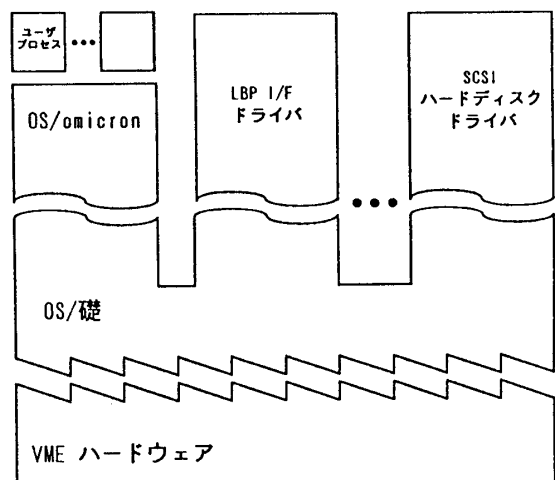


図1. システムソフトウェアの構成