

解説

高速漢字プリンタシステムについて*

黒崎悦明**

1. まえがき

漢字情報処理技術は、長い黎明期を経て、ようやく実用化期に入ったといえる。漢字入出力システムの導入に際してユーザが着目すべきポイントは、細かい点を別にすれば、

- (1) ハードウェアの性能はその業務にマッチしているか。
- (2) 機器を有効に使用するためのソフトウェアのサポートは充分か。
- (3) 外字処理については能率よく対処されるか。
- (4) 漢字コードについては、入力から出力に至るまで一貫して配慮されているか。

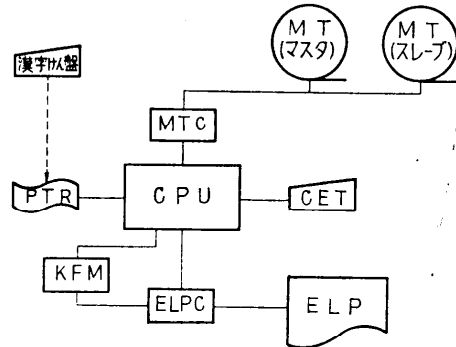
という4点である。(1)については漢字入出力装置、とくにプリンタについてみると、これは汎用型と業務専用型の2種に分けることができる。汎用型とは、字体、大きさ、縦書き横書きの制御などに flexibility を持たせた装置を指す。これに対して、業務専用型とは、一般に字種が大規模でもなく、字体、大きさも固定のものではあるが、ある種の業務の適用に際しては必要充分である、という性格をもった機器といえよう。後者の場合、前者に比較してコストが大幅に安くなるという特長がある。

本文では、上記の分類に従えば業務専用型ともいえる当社の高速漢字プリンタシステムを中心にして、漢字処理特有の上記(1)から(4)までの問題について解説を試み、その他の興味ある応用として、漢字のラベルプリンタとしての応用を紹介する。

2. システムの概要

2.1 ハードウェアの特徴

図-1は、オキエレクトロプリンタを漢字プリンタ



- ELP: エレクトロプリンタ
- ELPC: プリンタコントロール部
- KFM: 漢字フォントメモリ
- MT: 磁気テープ装置
- MTC: 磁気テープ制御装置
- CPU: OKITAC-4500コンピュータ
- CET: コンソールタイプライタ
- PTR: 高速紙テープリーダー

図-1 高速漢字プリンタシステム構成

部として採用した超高速漢字プリンタシステムの構成図である。センタコンピュータとのインターフェイスは、磁気テープインターフェイスであって、オフライン・プリンタ・システムを標準の構成としている。

表-1(次頁参照)にプリンタ装置の仕様を、図-2(次頁参照)にオキエレクトロプリンタの印刷原理を示す。

プリンタ部の印字方式には、イオン流の電界制御によるインクミスト印刷方式を採用している。これは、プラスのコロナ電極により発生されたイオンが、アパーチャの電界によって ON/OFF 制御され、ゲートを通過したイオンが、用紙後部のマイナスに印加されたプラテンに向う。イオンが霧状に流れているインクミスト中を走行する際、インク粒子に付着して、プラスのインク粒子となり、これがプラテンに引かれて用紙に付着することを、印刷の基本原理としている。

アパーチャの孔の列(ドット列)は X 方向に、1,320

* High speed KANJI Printer system by Yoshiaki KUROSAKI (OKI Electric Industry Co., Ltd.)

** 沖電気工業(株)

表-1 プリンタ装置の仕様

印字方式	イオン流の電界制御による印字
印字形式	ドットイメージ
印字速度	漢字 4,000 行/分 (3 LPI) 5,333 行/分 (4 LPI)
文字の構成	Kモード 16×18 ドットマトリクス Aモード 7×9 ドットマトリクス
1行印字数	66 字/行 (オプション 68 字/行)
漢字フォントメモリ (KFM)	基本 2,048 字 増設単位 2,048 字
用紙	普通紙
用紙の大きさ	横 6 インチ～22 インチ (標準 15 インチ) 縦 7 インチ～16 インチ (標準 11 インチ)
倍率制御機能*	* オプション 標準 (11 点) の 2, 4 倍

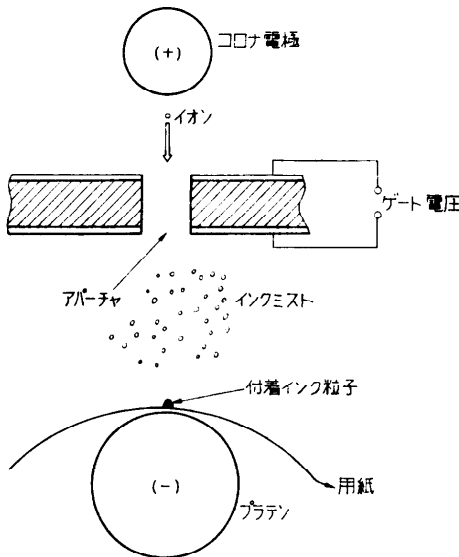


図-2 プリントの原理

ドット (100 ドット/インチ) 配列され、ドット列一斉モードで印刷される。プリンタコントロール部 ELPC には、標準 128 種のアルファニューメリックの文字発生器を有し、かつ、漢字フォントメモリに接続されている。漢字フォントメモリとしては、書き換え自由な RAM (ランダムアクセス・メモリ) を採用しており、これはプリンタシステムのコントローラとして中核をなす OKITAC 4500 CPU にも接続されている。

以上の構成により本漢字プリンタシステムのハードウェアとしては、下記の特徴を有する。

- (1) 普通紙に印字できる。
- (2) 漢字モード (16×18 ドットマトリクス/字)

と、英数カナモード (7×9 ドットマトリクス/字) を混合して行別に打ち分けられる。

- (3) インクミスト方式の、ドット列一斉モードで印字するので、毎分 4,000 行 (漢字モードの場合) あるいは毎分 8,000 行 (英数・カナモードの場合) の超高速でプリントする。
- (4) 漢字フォントメモリ KFM の字種のメインテナンスが容易である。
- (5) 漢字コードは、文字フォントが格納されている KFM のアドレスに対応しているため、フォントを格納するアドレスの変更により自由にコードを設定できる。
- (6) 通常のラインプリンタのように、キャリッジ・コントロールテープの掛けかえのわずらわしさはなく、キャリッジコントロールの情報がパンチされた紙テープを PTR から CPU メモリにロードすることだけでよい。

2.2 ソフトウェアの特徴

漢字処理システムにおけるデータ処理業務を、入力、処理、出力の段階に分けてみたとき、入力フェーズでは、通常データ処理業務と比較して、現況では殆んどオフライン形式で行なわれている。これは、漢字処理システムが未だ広く行きわたっていないことにもよるが、とくに、漢字入力の困難さのために、安価でかつ、入力手段の容易な簡易入出力端末装置の技術開発が遅れていることにも起因している。

現状の漢字処理システムによる実際の業務の流れは大略以下のようになっている。

- (イ) 原データのさん孔
- (ロ) さん孔データのチェック (モニタ・プリントリストによる)
- (ハ) 漢字データ入力ファイルの作成
- (ニ) センタ・コンピュータによる一般データ処理
- (ホ) 漢字データ・プリントファイルの作成
- (ヘ) 漢字データ・プリントファイルの印書

上記の(イ)～(ヘ)の各ステップのうち、(ニ)を除けば、漢字処理システム特有のソフトウェア技術が必要となる。これらの技術上の諸特質については、図-3 (次頁参照) の実際の業務フローに沿って、当社の高速漢字プリンタシステムのソフトウェアを例にして説明する。

本システムには、以下にあげられている各種のプログラムパッケージが用意されており、漢字処理業務を能率よく、合理的に遂行できるように考慮されている。

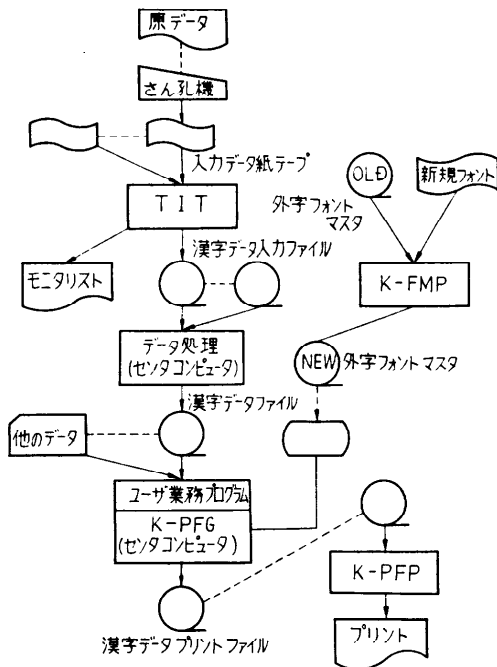


図-3 漢字データ処理業務の流れ

(1) TIT (さん孔データ処理プログラム)

けん盤さん孔機にて紙テープ等の媒体にさん孔(8単位2列/字)された漢字原データを読み込み、けん盤コードを所定の変換アルゴリズムによってプリンタコードに変換したのち、磁気テープ(漢字データ入力ファイル)へ出力する。その際、漢字コードの桁ずれのチェック、3章で後述される外字チェックとコードの割りあて、キーボード上の単語キー(“アパート”, “団地”等の単語を1打けんで入力するためのキー)のコードの、対応する漢字コード列への変換が同時に行なわれる。さらに、オペレータの要求により、モニターリストをプリントアウトする機能が備えられている。なお、各種のけん盤コードとプリンタコードとの関係については、4章で詳述する。TITは、本システムのOKITAC 4500 CPUで実行される。

(2) K-PFG (漢字データ・プリントファイル・セネレータ)

センタ・コンピュータのユーザプログラムによって使用される。漢字データ・プリントファイルを出力するパッケージである。フィード行数、ページコントロール等のフォーマット・コントロールの指定を行うことができる。

さらに、外字に対しては、あらかじめディスクにロ

ードされている外字フォントを取り出し、プリントファイルにドットイメージのまま書き出すことができる。

(3) K-PFP (漢字データプリント・ファイル印書プログラム)

K-PFGによって、センタ・コンピュータから出力された漢字データ・プリントファイルを、指定されたフォーマットに従って編集し、漢字プリンタ部へ出力する。K-PFPには、テストプリント、運用モニタリング、などの各種のオペレーション機能があり、コンソールタイプライタCETからのコマンドにより、ファイルの選択、スタート/ストップのページ指定が自由に行える。また、後述するような方法による外字の印字機能がある。

(4) K-FMP (漢字フォント・メンテナンスプログラム)

漢字フォントマスタの作成・変更、漢字フォントのKFMへの登録、外字フォントマスタのディスクへのローディング、漢字フォント・チェックリストの作成などのユーティリティからなる。

本システムでは、KFMとして、読み出し書き込み自由なRAMを採用しているため、印書業務に先立って、あらかじめ使用頻度等によって分類された漢字について、数種の漢字フォントマスタの中から必要なマスタを選択し、これをK-FMPによって漢字フォントメモリの内容を入れかえたのち、印書を実行させるなどの方法が採れる。従って、application orientedな字種の選定と分類により大容量のRAMを使用せずに業務の遂行が可能となる。

3. 外字処理について

漢字はその字種がきわめて多いことから、漢字処理システムの設計に際しては、内字と外字の問題を考慮しておかなければならない。

従来、単に外字といえは、通常、けん盤外字を指すことが多いが、入力から出力に至る一貫したシステムを論ずる場合には、図-4(次頁参照)に示されるように、

- (1) けん盤外字
- (2) プリンタ外字
- (3) システム外字

の3種に分類した方が理解し易い。

けん盤は操作者の人間工学的な立場から、最大2,500字、あるいは、特別な場合でも5,000字以内に留められるべきである。この場合、けん盤上の文字をけん盤

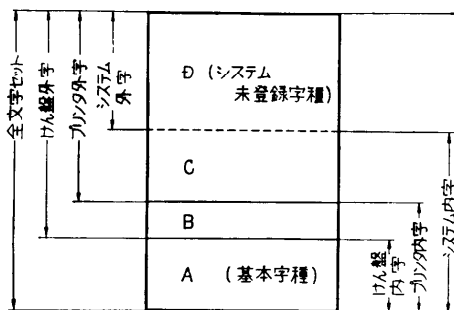


図-4 内字と外字の定義

内字と呼ぶのに対して、それ以外の字種をけん盤外字という。けん盤外字の入力方法としては、外字ファンクションキーおよび、テンキーを使用して、字引きにより直接にコードを入力するなど、いくつかの方法がとられている。

一方、プリンタの文字種（プリンタ内字）は、けん盤内字を包含するのが一般的である。しかし、プリンタ内字として出現頻度のきわめて少ない字種までむやみに多く持つことは、ハードウェアのコスト/パフォーマンスの立場から得策とはいえない。

このため、適用業務に応じて効率のよいプリンタ内字の選択が望まれる。図-4のB部は、けん盤外字であって、かつ、プリンタ内字であるから、入力処理段階において単純に、プリンタ内字コードに変換することができる。一方、プリンタ外字については特別の処理が必要とされる。ここで、プリンタ外字のうち、外字フォントマスタにドット・パターンとしてのフォントが登録されている文字グループ（C部）と、登録されていない文字グループ（D部）に分けよう。この場合D部をシステム外字と呼ぶ。C部の字種は充分多いことが必要であるが、特にシステムの初期稼働段階では完全ということとはあり得ない。従って万一、システム外字が現われた場合には特別の処理を必要とする。

本システムの場合、次のようにしてプリンタ外字の処理を行なっている。

(1) 漢字入力データファイルの作成段階で TIT プログラムによりプリンタ外字が発見され、その外字がシステム外字である否かが、コード数字列が表わす値により判定される。システム外字である場合、特殊マークでおきかえてモニタリストに出力する。

(2) システム外字のフォント（16×18 ドット）を新たに作成して外字フォントマスタテープを update する。

(3) センタコンピュータによって実行される K-PFG によりディスクの外字フォントを索引して外字フォント（16×18 ドットパターン）を漢字データプリントファイルにマージ出力する。

(4) K-PFG による印書の際、漢字データプリントファイルに外字が現われた場合、フィードの時間を利用して、その外字フォントをプリンタ部の漢字フォントメモリ（RAM）の空きエリア（外字用ワーキングエリア）にダイナミックに書き込み、そのエリアに対応するコード（アドレス）を指定することによって外字の印字を行う。

4. 漢字コードについて

漢字コードについて充分検討しておくことは、英数・カナ文字の字種がクロズドされているのに反して、漢字が数千から数万に及ぶきわめて多い字種をもつオープンの性格を有するものであることから、特に重要である。

一般に、けん盤から入力される漢字入力コードは、

- (1) 入力方式
- (2) けん盤配列
- (3) 制御コード

等の諸要因によって決定され、メーカーによってまちまちであるのが実情である。

さらに、けん盤配列あるいは内字の字種は、業務の変更にもよって、部分的に変更を受けることが多い。このような漢字処理特有の特質を無視して漢字処理システムの実現はありえない。

ある入力系に基づいてプリンタコード系を定めてしまうと、入力系の変更や他の入力系からのデータを処理する必要がある場合には、一般にばう大なコード変換表を必要とし、一定の置き換えアルゴリズムによる単純な変換法の実行が困難となる。入力コードを一つの数値としてみたとき、ある入力コード体系の与える数値列は、けん盤の一般的な性質から、図-5に示されるように部分的に詰められている（あるいは部分的に歯抜けがある）。

一方、本システムのフォントメモリ KFM は RAM

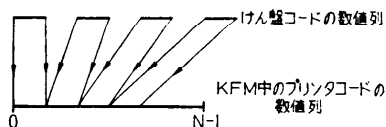


図-5 コードの変換

のメモリアドレスで文字フォントの所在をアクセスする方式がとられているため、プリンタ・コードは即ちメモリアドレスとみなすことができる。従って、プリンタ・コードを数値としてみたとき、この数値列は抜けのない非負整数列をなす。しかも RAM 内の文字フォントの配列は自由に変更できるから、次のことがいえる。即ち、

「いかなる入力系が与えられても、この入力系すべての漢字コードを一定の置き換えアルゴリズム(写像)によってプリンタ・コードに置き換えられるように設計することができる。」

従って、本システムの場合、一定のプリンタ・コードというものはありえず、また、その点が上述の諸問題を解決するための特長でもあるといえよう。

5. 漢字プリンタの他の応用

現在、ダイレクトメールなど一般の応用とは別に、梱包ラベル、荷札等に用いられる大きく拡大された漢字の字体のプリントは各方面から要求され、興味深い応用がある。ドットプリンタとしての特性を生かせば、エレクトロプリンタは、この種のアプリケーションのひとつとして最適である。

その方法に、(1) 標準モードの文字を、2倍または4倍にハード的に自動拡大して、高速印字をする。

(2) KFM より所定の漢字フォントを CPU のメモリにロードし、プログラムによって任意の倍率に拡大

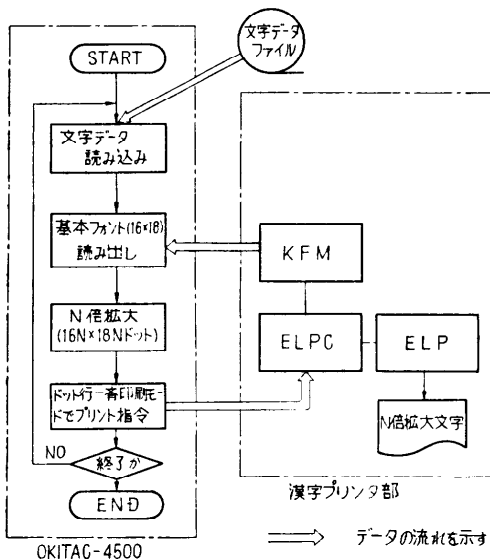


図-6 拡大プログラムによる拡大文字の印字

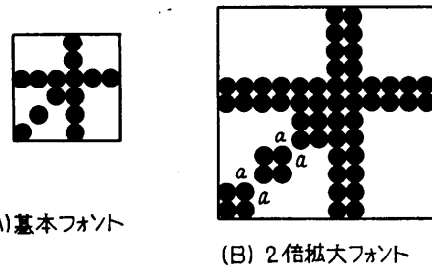


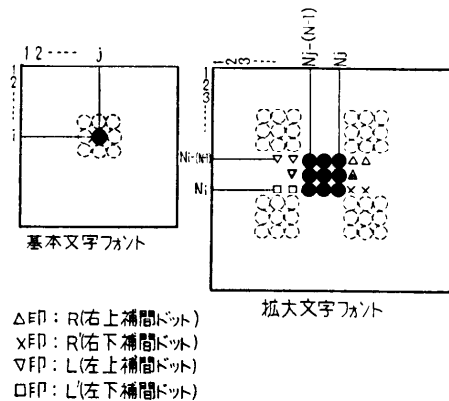
図-7 補間しない場合の拡大

して印字する。前者については、本システムではプリンタのオプション機能として提供されている。以下では、拡大プログラムによって基本文字フォントを任意整数倍に拡大して、ドット列一斉モードで印刷する方法について述べてみよう。

図-7(A) に示す基本文字フォントから N 倍の拡大文字フォントを得るのに、基本文字フォントの各黒(白)点を $N \times N$ 枚の正方状配列のドット群に対応させる原理のみを適用すると、例えば $N=2$ の場合、同図(B)に示すようにくずれ部 a が生じ品質が悪化するという問題がある。以下にこれを解決する方法について述べよう。

ドットパターンの各黒白点において、斜関係にあって隣り合う任意の2黒点についてみれば、2点 A_i, A_{i+1} の左右(または上下)に他の黒点が存在する場合と存在しない場合とがある。前者は横または縦方向直線と他の直線との交差部においてみられるが、その場合のみドット A_i, A_{i+1} 間での補間を避け、後者においてのみ補間を行なうならば、きわめて自然な拡大文字フォントを得ることができる。

したがって、補間実行条件は以下の(1)、(2)の条



△印：R(右上補間ドット)
 ×印：R(右下補間ドット)
 ▽印：L(左上補間ドット)
 □印：L(左下補間ドット)

図-8 拡大原理

件が同時に成立することである。

(1) 互いに隣り合う行 (ドットライン) 中の 2 点 A_i, A_{i+1} が黒点で斜関係にある。

(2) i 行にあって列が A_{i+1} と同一列に位置する点が黒でなく、かつ $i+1$ 行で列が A_i と同一列に位置する点も黒でない。

そのことを、図-8(前頁参照)について簡単な論理式を用いて説明しよう。

基本文字フォント上の i 行 j 列目の点 $A(i, j)$ に注目する。基本部分パターンを B , 右上補間パターン, 右下補間パターン, 左上補間パターン, 左下補間パターンをそれぞれ R, R', L, L' としよう。 $A(i, j)$ を, 基本文字フォントの i 行 j 列目が黒点のとき 1 で, 白点のとき 0 であるような論理変数とすると, B および R, R', L, L' が 1 のビットストリングとなる条件は, それぞれ (3)~(7) 式で示される。

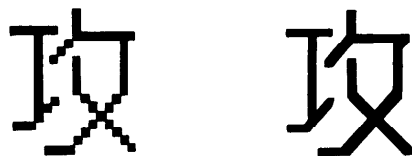
$$A(i, j) = 1 \quad (3)$$

$$A(i, j) \cdot A(i-1, j+1) \cdot \overline{A(i-1, j)} \cdot \overline{A(i, j+1)} = 1 \quad (4)$$

$$A(i, j) \cdot A(i+1, j+1) \cdot \overline{A(i+1, j)} \cdot \overline{A(i, j+1)} = 1 \quad (5)$$

$$A(i, j) \cdot A(i-1, j-1) \cdot \overline{A(i-1, j)} \cdot \overline{A(i, j-1)} = 1 \quad (6)$$

$$A(i, j) \cdot A(i+1, j-1) \cdot \overline{A(i+1, j)} \cdot \overline{A(i, j-1)} = 1 \quad (7)$$



(a)補間なしの場合 (b)補間した場合

図-9 拡大文字の印字例

したがって, $A(i, j)$ が走査されたとき同時に $A(i, j)$ を取り囲む他の 8 点 $A(i-1, j-1), A(i-1, j), A(i-1, j+1), A(i, j-1), A(i, j+1), A(i+1, j-1), A(i+1, j), A(i+1, j+1)$ を取り出して (3)~(7) の判定を行えば, $A(i, j)$ 点のまわりの補間を行なうことができる。

図-9(a) は補間しなかった場合の印字例であり, 同図 (b) は補間した場合の印字例である。

6. あとがき

本文では, 漢字プリンタシステムを検討するに当たって必ず必要とされる外字処理や文字コードの問題を中心に述べた。5. で触れた拡大漢字印刷の応用例としては, 梱包ラベルのプリント等のほかに, 合格者の発表の際の氏名の一覧の印刷など, 多くの応用も考えられよう。 (昭和 50 年 5 月 12 日受付)