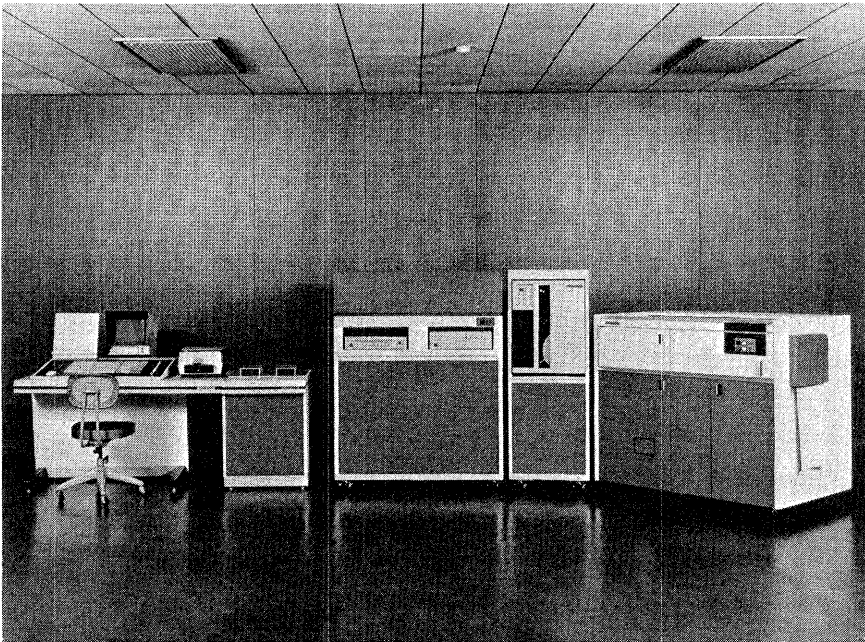


東レ株式会社
システム部 面谷和生

I TORAY 8500システムの概要

TORAY 8500システムは、プログラムメモリーを漢字メモリー部に使用した漢字情報処理システムで現在毎秒1,000字の出力が可能である。プログラムメモリーは約5cm²平方のプラスチックの板のようなものの中に、約7,000の漢字の文字パターン（各文字は32×32のドットになつている）が収められた非常に高密度なメモリーである。

プログラムメモリー上のアドレスに相当する漢字コードを用いて、レーザー光線に左右上下に操作し、文字パターンを読み出す。読み出された文字パターンは撮像管で受けとめられ、プリンターへ転送される。漢字プリンターでは、電子写真方式による印字を行なう。印字の際の用紙送りは、いわゆる間欠送りを行なつており、印字の際ソフトウェア的にいろいろな編集処理を行なうことができる。



当システムは、主として、汎用コンピュータの周辺機器として、ラインプリンターの代替、または延長を考へており、現在ラインプリンターで処理されているものや、邦文タイプ、手書きなどで処理されており、漢字の出力ができないが故に、未コンピュータ化の分野を主用途と考へて、ソフトウェアの面でもその方面の充実化を計つている。

また、オンラインCRTディスプレイを用いての情報処理も有用であり、ソフトブ

リナーとしての用途も開発中である。

II TORAY 8500 システム用ソフトウェア

TORAY 8500 システムには大別して次の 2 種類のソフトウェアがある。

1. TORAY 8500 内蔵のミニコンピュータ用ソフトウェアで磁気テープからのデータを読み、漢字プリンターを制御し、文字を出力するものや、ターミナルキーボードや、カセットテープなどの補助記憶装置の制御を行なうコントロールプログラムである。
2. TORAY 8500 のホストコンピュータ用ソフトウェアで、ユーザー、特にプログラマーが漢字プリンターをあつかい易いように考えられたソフトウェアである。

1 つはキーインされた大量の漢字データを汎用コンピュータで利用する際に、わざわざ漢字コードを用いなくても EBCDIC コード等により、かつ高水準言語であつかえるようにするデータ編集のユティリティである。もう 1 つは、既存のデータと漢字データを結びつけ、漢字プリンターで出力する際に、罫線の作成や、タイトルの挿入、ページカウントの作成など複雑なプログラム処理をパラメーターで行なえるように工夫されたプリント編集ソフトウェアである。本日の主題であるため横変換ソフトウェアもこの中に組み込まれている。

その他にもダイレクトメール関連の処理を一貫して行なうソフトウェアや、人名簿作成など、漢字情報処理の用途を考慮したアプリケーションが用意されている。

III TORAY 8500 システムにおけるたてよこ変換の必要性

TORAY 8500 システムの漢字プリンター TORAY 8570 においては、用紙の巾は最大 297mm あり、8 ポイントの英数字のハーフピッチサイズを用いれば、最大 180 字出力できる。(漢字だけの場合は 90 字)。これは、通常のラインプリンターでは 130 字程度であるので、ラインプリンターで出力されていたものを漢字処理し、TORAY 8570 でプリントアウトするならば必ずしもたてよこ変換を行なう必要はないであろう。相当複雑にプリプリントした用紙への出力も、ハーフサイズの罫線が用いられるので、多くの場合 180 字までで足りるだろう。さらに、ラインプリンターをはなれて漢字プリンター特有のものを漢字、ハーフサイズの英字、数字、特殊文字、記号を組合せて現在のラインプリンターで行なわれているようなフォーマットの設計をはなれて自由な方法で、新しい帳票の設計が可能になる。

しかし、TORAY 8500 漢字情報処理システムでは従来のラインプリンターの制約を脱して長い図表、たとえばスケジュール表等を作成することができる。また、通常の事務処理のアウトプットでも、自由な長さで作成するのが便利な場合があるので、次にのべるような、たてよこ変換ソフトウェアが用意されている。

TORAY 8500 用のたてよこ変換のソフトウェアは、汎用コンピュータ(ホストコンピュータ)を用いて行ない、一種のユティリティプログラムとしてユーザーに提供することができる。本プログラムは、JIS 7000 の FORTRAN で書かれており、プリコンパイラの形をとっている。(ユーザーの使用しているホストコン

ビュータにより、若干のかつ根本的な制約があるので)

汎用コンピュータで、たてよこ変換を考える場合、T O R A Y 8 5 0 0 システムの独自の特徴を生かすためにハードウェアはもちろん、ソフトウェアにおいても工夫すべき点がある。

T O R A Y 8 5 0 0 のたてよこ変換で、考慮しなければならない特徴は、

1. 同一行内に、自由に、大文字、中文字、小文字(中文字に対するふりがなルビ)が用いることができる。
2. 中文字には、フルサイズ(通常の漢字)とハーフサイズ(英字、数字、特殊文字)の二種類のモードがある。
3. 用紙送りを等間隔にし、特にルビがある場合、横方向のならばが不自然ではないようにする等である。さらに、一般プログラマーに対しては、コーディングの際にわざわざたてよこ変換をするための制約を作つてはならない。

IV たてよこ変換のためのハードウェア的条件

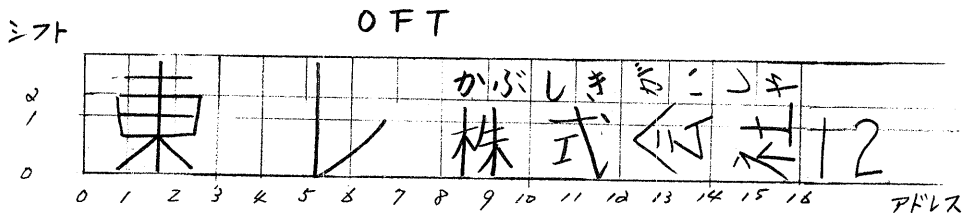
T O R A Y 8 5 0 0 の漢字プリンター T O R A Y 8 5 7 0 では、O F T (オブティカルファイバーステップ)を用いて、ホログラムメモリーから送つてこられた文字を印字している。用紙の送り方は、間欠送り方式(ソフトウェアで、改行命令を出し、1行分文字を書く間用紙は停止する)をとつている。

O F T は、全体(297mm)をたて方向3、よこ方向180の柵に分解せられ、いずれの柵に対しても、起点をきめ、横方向へのピッチを1、2、4(各々小文字、中文字、大文字に対応)で、文字を発生させることができる。

ルビにあたるものは、まず、アドレス0でシフト2で出力し、次はアドレス1、シフト2で出力する、この様に横方向のアドレスの進みは1である。

上記ルビに対応する中文字は、アドレスを0にもどし、シフト0で出力するとよい。横方向には2のピッチで進み、中文字は偶数のアドレスにバウンダリーを合わせる。

大文字は、アドレス0、シフト0で出力する。大文字の横方向アドレスの進みは都合により4にしている。



O F T の上ではルビは最上段に印字し、中文字に対するルビにしている。

次に文字の発生方法を簡単に順を送つて示す

1. 磁気テープより、-1ブロック(ブロックングは可能)コア内に読み込む。
2. データ内より、ファンクションを読みとり、出力モード、出力場所、文字サイズをセットする。(デフォルトもある)

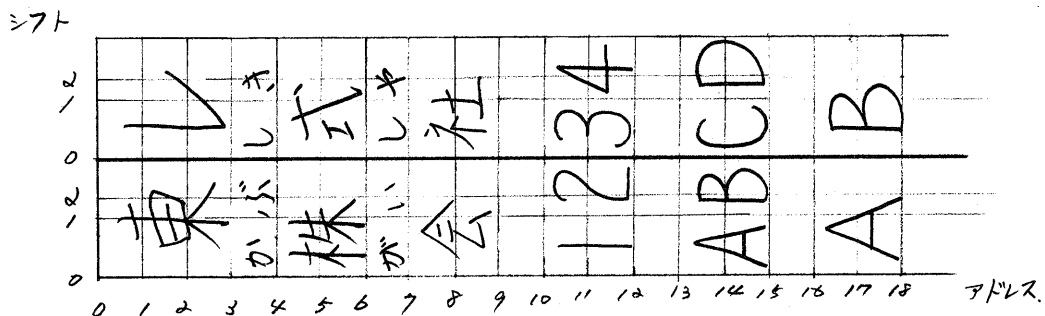
3. 漢字コード(2バイト)1文字分を、データ内より出力する。
4. もし、ファンクションに変更があれば、リセットする。文字サイズに応じて、出力場所を進める。
5. 改行命令を出す。

上記1~5をくりかえすことによつて、文字をOFT上に出力し、ZNO紙に印字することができる。

たてよこ変換されたデータを出力する場合には、ハードウェアを次の様に操作する。

1. ハードウェアの出力モードを、たてよこ変換型にする。
2. ルビにあたるものは、アドレス0、シフト0で

OFT(てよこ変換時)



まず出力し、次に、アドレスはもとのままで、シフト1に起点をうつし、小文字を出力する。

3. 中文字は、アドレスを1にうつし、シフト0で出力する。
4. 大文字は、アドレスを3、シフト0で大文字のファンクションを用いて出力する。横方向への進みは3にする。
5. ハーフサイズの文字は、アドレス4、シフト0、アドレス4、シフト1の2点を起点に出力することができる。

この場合OFT上の一行の間隔は1/6インチにとつてあり、ソフトウェアの改行命令により1/6インチ用紙は送られる。

たてよこ変換時ハーフサイズの文字や、ルビのバランスをとるために、シフト1はOFTの巾の中央にセットしてあり、シフト2は1/3のところセットしてある。通常の横書きの場合には、発生しない若干のすきまが、たてよこ変換を行なうと、ルビやハーフサイズの字の間に発生するのはやむをえない。(1/18インチ~1/36インチ) 大文字は、ソフトウェアの項でものべるが、横書きデータとの関係上、1行おきに

しか書けなくなる。

T O R A Y 8 5 7 0 には 1/3 インチ送りの機能もつているが、たてよこ変換を行なつた場合は、以上の説明の様に用いることはできない。

V たてよこ変換ソフトウェア的方法

以上の様なハードウェアの機能を用いてソフトウェア用意する。このときの条件として、一般のプログラマーの負担は増さないこと、変換スピードをあげることでより少量のメモリーで行えること。どんなホストコンピュータでも使用できることなどである。

T O R A Y 8 5 0 0 システム用たてよこ変換ソフトウェアは、前述のように J I S 7 0 0 0、F O R T R A N で書かれており、F A C O M 2 3 0 - 4 5 B (O S II) で稼動中である。

プログラムのロジックを順を送つてのべると

1. プリコンパイラ化のための準備

任意のホストコンピュータ使用するとの前程のため、高水準言語を用いる。そのためプリコンパイラとした。

特に、インプットとなるデータのレコード長、ブロック長をパラメータとして読みこむ。アウトプットは固定。

2. インプットデータの読み込みを行ない、データに従いテーブルに展開する。

インプットデータは、数種のファンクションと漢字コードからなりたつ。(それぞれ 2 バイト)

ファンクションには概略次の様なものがあり、データ内のどこに位置していてもよい。ファンクションがあらわれた時点で、ハードウェアはすべてリセットされる。

(1) 出力モードの指示

(2) 改行数、改ページの指示

(3) 文字サイズ指定 (デフォルトは中文字)

(4) 文字発生場所指示 (通常はアドレス 0 より)

当ソフトウェアでは、まず (2)、(3)、(4) の各ファンクションを確め、以下のデータを出力すべきテーブルを決定する。

テーブルは、大文字用、中文字、ハーフサイズ兼用、および、ルビ用の 3 種類がある。大文字のファンクションが出れば、次のファンクションが来るまで、大文字のテーブルにデータをストアし、そのインデックスアドレスを進めてゆく。改行命令では、各テーブルのインデックスアドレスを必要数進める。

大文字、中文字 (ハーフサイズ)、ルビのテーブルへのストアを行なり割合は、中文字 2 に対し、大文字 1、ルビ 2 つに対し、中文字ならば 1 つ、ハーフサイズならば 2 つとする。もし、データ内にハーフサイズと中文字がかさなる場合は、(重ねて印字することは通常行なわれない) あとから出てきたものが優先される。他の組合せにおいては、その様なことはなく、重ねて印字することも可能である。

よつて、大文字は、中文字2つに対して1文字出力されるので、1行おきに出力することになる。大文字は、タイトルや見出しに用いられることを考えれば、さしつかえはないであろう。

3. 改頁のファンクション ' P K ' が現われると、テーブルの内容を出力する。

' P K ' が現われると、現在処理中のレコードをそのままにし、テーブルの内容を、中文字、大文字、ルビの順に出力してゆく。その際、大文字や、ルビは多くのデータではあらわれないであろうから、漢字プリンターでブランクをわざわざ出力するのは、時間的に効率が悪いので、ストップマークを入れ、その後のデータをスキップし次の処理を行なうことができる。

4. 各テーブルのクリアーと、次の頁のためのプログラムリセット。

3つのテーブルをすべて0クリアする(0はブランクを意味する)。多くの場合その内容は、用いられないまま0であろうから、各テーブルのカラムには、そのアドレスインデックスの最大値がストアしてあり、未使用のものはクリアせずに時間の短縮化をはかっている。1頁分(' P K ' マークより次の ' P K ' マークまで)の出力が終ると、各種のカウンターをリセットし現在処理中のデータの処理を継続し、1へ帰える。

以上述べたように、T O R A Y 8 5 0 0 システムにおけるたてよこ変換は、ハードウェア、ソフトウェアの双方において、種々の工夫がなされた結果といえる。