

実験データ処理のためのオンラインシステム*

石黒美佐子** 山田孝行** 次田友宣**

Abstract

This system was developed for scientific data processing of acquired data from a variety of experimental facilities. Mini-computer at remote stations send data received from those facilities to the central host computer through communication lines, while the host computer sends back the output to the originated station.

Specifications for the system are as follows:

- (1) The terminal devices consist of various types of mini-computers which are not standardized.
- (2) Terminals individually require different types of data processing.
- (3) Fundamentally, ISO code is prepared but to receive input binary data from terminals, the pseudo binary method is also acceptable.
- (4) Data processing is carried out by the unit of input messages continuing for 1 k to 8 k bytes.
- (5) Program for data processing can be written by FORTRAN.
- (6) A large amount of output information can be recorded by the line printer or X-Y plotter at the Center as a complementary output device.

1. 概要

1.1 システム導入目的

日本原子力研究所(以下原研と略す)においては、従来、実験データ処理は、大型計算機でバッチ処理で行なってきた。近年、大型機(FACOM 230-60)をセンタ側計算機としてオンラインで処理したいという要望が出てきた。ところがバッチ処理を主目的として導入した大型計算機の安定稼働をはかるためには、一足飛びに各種ミニコンと大型機とを結合することには不安があった。まずは中型計算機(FACOM 230-35)で試作実験し、大型機でのオンライン化に先立ち、センタ側システム、端末プログラム、端末インタフェース(ハードウェア)に関する問題点を解決しておこうというのが今回の目的である。

1.2 データの種類と変動

端末で発生するデータは、原研構内の原子炉や、加

速器などから発生するパルスが波高分析器・AD変換器などによりデジタル化されたものである。これらのデータが各施設専用の小型計算機により収集され、センタに送られる。端末で1回の実験により発生するデータ量は1k語から8k語(1語20ビット以内)で、5分から30分間隔である。これらの実験データは疑似バイナリ方式でセンタに送信される。

センタ側計算機でデータ処理が終了すると、端末に計算結果を送信する。端末では前回の実験結果にもとづいて次の実験を設定する場合もあるので、応答時間は10分程度が望まれる。また計算結果のうち、緊急を要しないもので量が多いものは端末に送らないでセンタのラインプリンタに出力するか、磁気テープを経由してX-Yプロッタにオフラインで出力する。

1.3 データ処理の内容

(1) 実験データのスペクトルの平滑化、ピークの探索、面積化の計算、1次微係数の計算等科学技術計算をする。なお1回の実験のための入力データは、1k語から8k語(1語4バイト)の電文群からなる。

* Online System for Experimented Data Processing, by Misako Ishiguro, Takayuki Yamada and Tomonori Tsugita (Japan Atomic Energy Research Institute);

** 日本原子力研究所

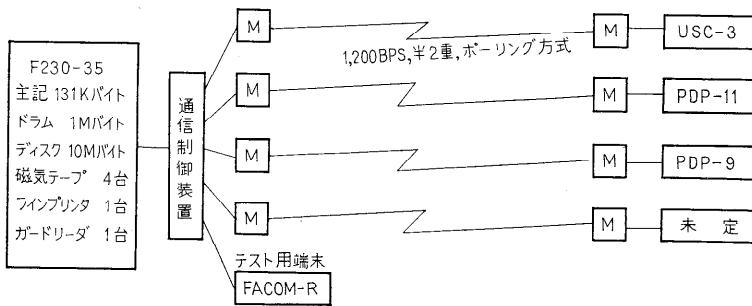


図 1 Hardware system

1.4 電子計算機組織

モデムは 1200 BPS の電電公社の規格品を用いるが、端末計算機との間でハードウェアインタフェースがとれない場合 (USC-3 など) には、インタフェースを特別に作成することが必要である。

2. 電文のデータ形式

電文コードは ISO コードを用いる。端末で発生するデータはバイナリ形式なので、ISO コードの範囲内でバイナリ (疑似バイナリとよぼう) 形式のデータを送れるようにすること、データ処理の入力データが電文群から成ること、以上 2 点を考慮して以下のような方式にする。

2.1 バイナリデータの取扱い

バイナリ伝送が可能なら、少ないバイト数で多量の情報が送信できて、その上コード変換もないので好適である。しかし原研で使用する通信制御装置 (F 1801 K) ではバイナリ伝送は許されない。そこで伝送上は ISO コードを使用するものとし、運用として整数には 5 ビットバイナリ方式、実数には 4 ビットバイナリ方式を使用する。もちろん通常の ISO コードで送ることも可能である。

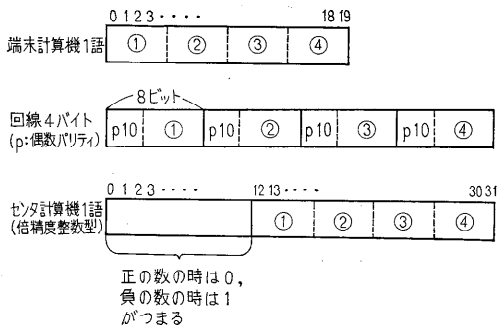


図 2

(1) 5 ビットバイナリ方式
 端末計算機の 1 語のビット数は以下のとおりである。

- PDP-11, USC-3 20 ビット
- PDP-9 18 ビット
- FACOM-R 16 ビット

端末計算機の 1 語は 20 ビットを限度とし、これをセンタ側計算機の 1 語 (倍精度整数 32 ビット) に対応させる。この方式は主に、端末からセンタに実験

データを送るのに用いる。

図 4 に見るように、回線上の 4 バイトに対して 20 ビット (7 桁程度) 送れるので、10 進化して送るのに較べれば 1.7 倍となる。

(2) 4 ビットバイナリ方式

データ処理の出力結果は、桁数が大きくなったり、浮動小数点表示になったりするので、実数型のデータも取扱えるように考慮しなければならない。もちろん ISO コードの文字表現のまま送った方が端末計算機で出力結果をタイプライタやプリンタに出すのに好都合な場合もある。

この場合、センタ計算機の 1 語に対して最低 12 バイト (FORTRAN の書式変換で E12.5 程度) 必要となる。伝送上のデータ量をできるだけ少なくするために、FACOM 230-35 の実数語が 16 進表示であることに対応づけて、4 ビットごとに切断した方法を取り入れる。

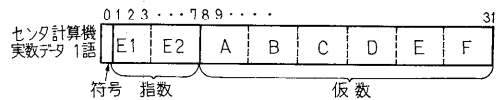


図 3

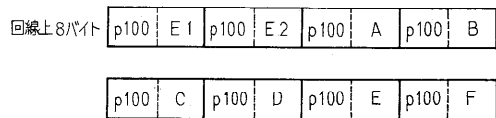


図 4

センタ計算機の 1 語が 8 バイトに対応するので、文字表現で送るのに較べると 1.5 倍となる。

2.2 実験データと電文との関係

(1) 1 電文 (256 バイト) の構成

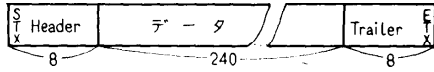


図 5

Header の内容

- 回線番号 (1 バイト)
- 電文シーケンス番号 (2 バイト)
- コード系 (1 バイト) 文字表現, 5 ビットバイナリ, 4 ビットバイナリ
- 処理プログラム番号 (1 バイト)
- 予備 (2 バイト)

Trailer

- 電文種類 (1 バイト) 最初, 途中, 最終, 通知
- 通知項目 (2 バイト) 完了通知, エラー通知コード
- 予備 (4 バイト)

(2) 電文

H電文とデータ電文がある。コード系は1電文毎に異なってもよい。電文のHeaderにコード系を指定する。

◎H電文

端末からセンタへの第一番目の電文にはHeader的役割を持たせる。文字表現でなければならない。

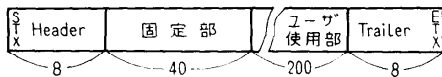


図 6

・固定部

- ①データ番号 (4桁)

端末ごとに、発生するデータに番号をつけ、この番号により、センタのデータ処理状況を端末側で知ることができる。
 - ②入電文総数 (4桁)
 - ③処理プログラム打切時間 (4桁)
 - ④出電文打切数 (4桁)
 - ⑤ラインプリンタ出力打切桁数 (4桁)

③, ④, ⑤は、データ処理プログラムをFORTRANで書くので、何かの誤りにより、ループしたりするのを防ぐため。
 - ⑥予備 (20バイト)
- ・ユーザ使用部
- 実験の日時、種類、計算条件、その他データ処理を制御する情報などを一般データに先立って任意に入れる。

◎一般電文

任意のデータを、任意のコードで入れることができる。

(3) 実験データの伝送形式

端末で1回の実験で発生するデータは次のような伝送形式でセンタに送られる。センタから端末へのデータ伝送形式も、H電文がないだけでほぼ同じである。それぞれを入電文群、出電文群と呼ぼう。

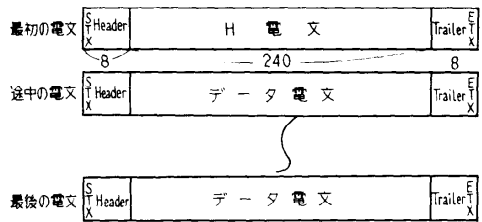


図 7

たとえば、4k語を5ビットバイナリで端末からセンタへ送る場合を考える。

$$4 \text{ k語} = 16 \text{ kバイト}$$

$$1600/240 = 66.66\dots$$

H電文を入れると68電文となる。これが入電文群である。

上記を送信するのに必要な時間は、 $68 \times 256 \times 10 / 1200$ で約2分30秒となる。

2.3 通知電文

センタから端末への通知手段として、通知電文がある。通知電文は1電文(256バイト)で構成される。端末では、通知電文により送った電文のエラーが判明し、また正常に送られたデータのセンタにおける処理状況がわかる。通知電文は、次の時に

- (1)センタで受信した電文のformatエラー
- (2)受信完了, データ処理完了, ラインプリンタ出力完了, 送信完了, 全処理完了

送られる。

3. ユーザプログラム

3.1 ユーザプログラムの構成

ユーザ処理プログラムは、ROS (Real Time Operating System) のもとで、オンラインパッケージ(COP-F)を用いて作成する。プログラムを処理タスク単位で、タスクの起動とデータの流れについて図示したのが図8である。タスク番号6から10までがFORTRANで書かれたデータ処理プログラムである。

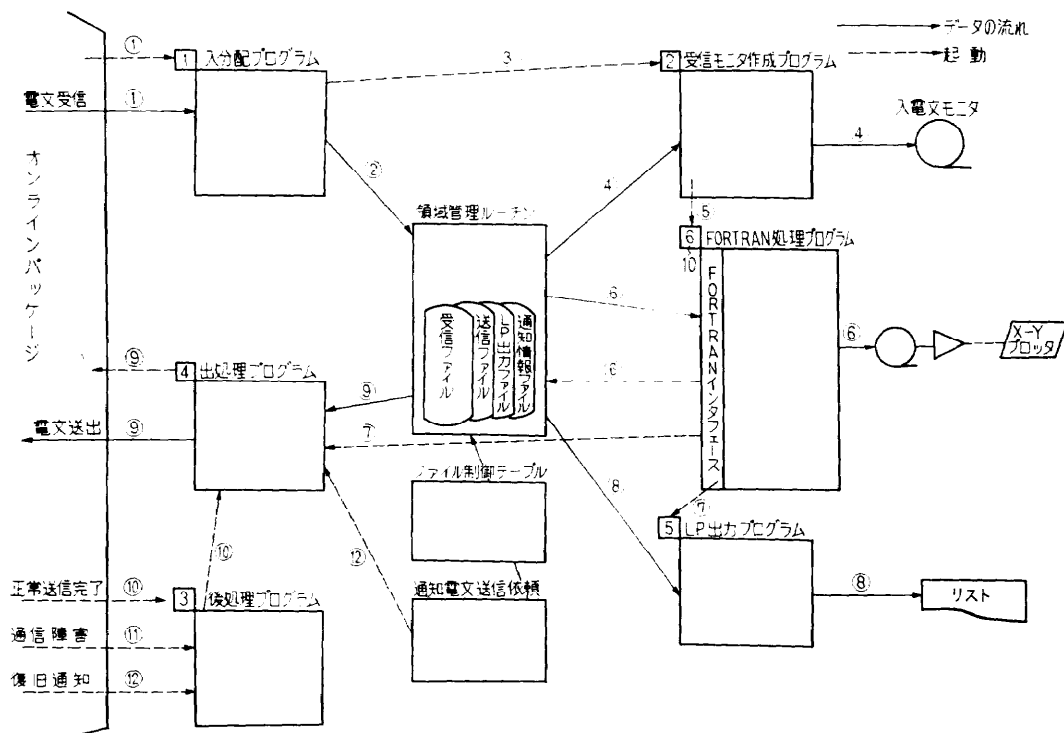


図 8 General flow of the processing kstac

(1) 入分配プログラム

回線ごとに並列で送られてくる電文を回線別に区分し、FORTRAN で使用できる形にコード変換する。所定の電文までを1データ分(入電文群)とし、受信ファイルに集収する。入電文群の収集が完了すると受信モタ作成プログラムを起動する。

(2) 受信モタ作成プログラム

受信ファイルのデータを磁気テープにコピーする。システムダウンなどの障害発生にそなえて、その時点でのファイル制御テーブル(後述)の内容も記憶する。受信データのモニタリングの完了をもって受信完了とし、端末に通知する。H電文で指定されたデータ処理プログラムを起動する。

(3) データ処理プログラム

現在開発完了しているのは、汎用性のあるガンマ線のスペクトル解析プログラムである。受信ファイルから電文を読み、計算する計算結果を端末に送信するためには、出電文形式にコード変換し、送信ファイルに格納するセンタのラインプリンタに出力するためには、書式付きのラインプリンタ出力ファイルに格納する。X-Yプロッタの出力は磁気テープを経由してオ

ラインでなされる。処理完了後、出処理プログラムとラインプリンタ出力プログラムを起動し、出力結果の処理を依頼する。

(4) 出処理プログラム

送信依頼された出電文群を送信ファイルより読み出し送信する。通知電文の送信も行なう。出電文群が互に入り乱れないように、また通知電文を優先して送信させるために、次の優先順位により1電文ずつ送信する。

- ① 通信障害などによる再送電文。
- ② 通知電文。
- ④ 現在送信中の出電文群の継続電文。
- ⑤ 新規の出電文群の最初の電文。

(5) 後処理プログラム

出処理プログラムで電文の送信が出されると、オンラインパッケージの回線制御部で実際の電文送信がなされる。そして正常に送信できたかどうか後処理プログラムに通知される。正常なら継続電文を送信するために再び出処理プログラムを起動する。出処理と後処理とが交互に働くことにより出電文群の送信が完了

する。完了後送信ファイル領域を解放し、その時すでに対応するデータのラインプリンタ出力が完了しているかどうかを調べ、完了していればそのデータに関する全処理が完了したことになるので、受信ファイル領域も解放する。通信障害やハードウェアにより送信が異常となる時は、その理由も添えて後処理プログラムに通知される。コンソル上に障害理由を出力するなど措置をし復旧を待つ。オンラインパッケージのテストコール機能により障害の回線の復旧も後処理プログラムに通知される。障害により送れなかった電文の再送が必要な時は、再送信のために出処理プログラムを起動する。

(6) ラインプリンタ出力プログラム

ラインプリンタ出力ファイルからデータを読み出してラインプリンタに出力する。1データ分の出力が終了するとラインプリンタ出力ファイル領域を解放し、対応するデータの出電文群の送信が完了しているかどうかを調べ、完了していれば全処理完了したことになるので受信ファイル領域も解放する。

(7) システム開始、復旧プログラム

開始部は、全回線の閉塞解除とユーザファイルのオープンを行ないオンラインシステムを開始させる。復旧部は、受信モニタ（磁気テープ）より、ファイル制御テーブルと、受信データを読み取り、すでに受信したデータ番号を回線ごとに表にし、コンソルに出力する。オペレータが未処理のデータの番号と回線番号をコンソルから入力すれば、対応するデータを受信モニタから受信ファイルに移し、受信モニタ作成プログラムに起動をかけ、この後開始部を行なうことによりシステムを再開させる。開始と復旧の区別はオペレータによりコンソルで指示する。

3.2 FORTRAN インタフェース・パッケージ

メーカが提供する ROS FORTRAN は本来オンラインシステムのバックグラウンドジョブに利用することを目当てに作成されていた。リアルジョブとして働らくデータ処理プログラムを FORTRAN で作成するためには、オンラインパッケージとの結合および入出力処理のために FORTRAN で記述できない部分を補う必要が生じた。そのために FORTRAN インタフェースパッケージが作成された。

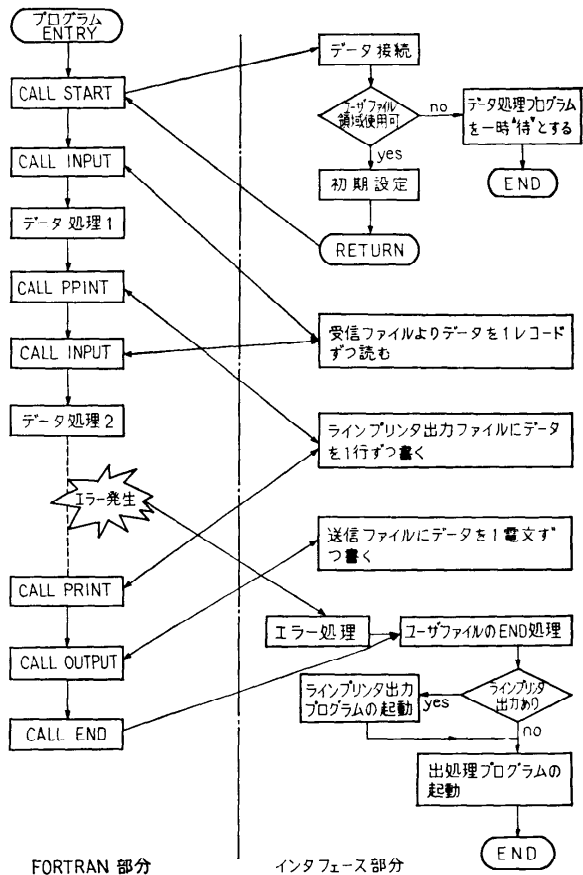


図9 Example of data processing program

一方 FORTRAN で書いた部分に、プログラム割込、入出力関係のエラー、基本関数使用上のエラーがあった場合に、タスクを異常終了させずに一括してエラー処理ルーチンに引渡すことも必要である。エラー処理ルーチンは、エラー理由を出電文とラインプリンタに出力した後、正常な終了措置によりタスクを終らせる。

4. ユーザファイル

4.1 ファイルの種類

(1) 受信ファイル

受信した電文が、FORTRAN で使用できる形式にコード変換され端末ごとに組立てられ保存される。

(2) 送信ファイル

データ処理プログラムの出力結果のうち、端末に送信すべきデータを出電文形式にコード変換され保存される。

(3) ラインプリンタ出力ファイル

同じくセンタのラインプリンタに出力すべきデータが書式つきで保存される。

(4) 通知電文用ファイル

通知電文が一時保存される。回線が空き次第処理プログラムにより端末に通知電文が送信される。

4.2 ユーザファイルの動的利用

1 データ分のファイルは、受信ファイル、送信ファイル、ラインプリンタ出力ファイルから成り、ディスクパックを使用する。ラインプリンタに出力すべきデータを一時ため込むために、かなりの領域を必要とするので(2ラインで1レコード)リスト処理により領域の動的利用をはかる。1レコードは256バイトから成り、100レコードを論理上の1ブロックとする。未使用領域は空領域として一括管理する。100ブロック(=10,000レコード)をこのために使用する。なお通知電文用ファイルは、やはりディスクパック上に取られ、固定領域とし300レコード取られる。

4.3 空領域の管理

- (1) システム開始時(復旧時も含む)は、ファイルの全領域を空領域とする。
- (2) 情報を出力する時点で1論理ブロック単位で空領域から領域を確保する。
- (3) 1データ分のデータ処理が全部完了すると、ファイル領域は解放され空領域に返却される。
- (4) 空領域が全領域の半分(50論理ブロック)以下になると全回線受信閉塞とし、受信済のデータに対するデータの処理を優先させる。
- (5) 空領域が半分以上になれば受信閉塞を解除する。
- (6) データ処理プログラムの最初で領域使用予定ブロック数と空領域ブロック数を比較し、領域が足りない場合はデータ処理プログラムを待状態にし、定時間後(1分後)に再起動し、再び領域使用可能かどうかテストする。なお領域使用予定数は、入電文群のH電文に、送信打切電分数(SMAX)、ラインプリンタ出力打切行数(LMAX)を記入してあるのでこれらより算定する。

4.4 ファイル制御テーブル

ファイル制御テーブルは、ユーザファイル領域の管理、システムダウン復旧のための情報の保存および回

線ごとに送信受信の並行処理するための制御をすることを目的としている。全部で512バイトから成り、基本情報部、リストコントロール部および回線ごとに回線制御部がある。基本情報部には、接続回線数、ユーザファイル論理ブロック数、論理ブロック内のレコード数、空領域ブロック総数、使用領域先頭ブロック数、使用領域最終ブロック数などが入っている。リストコントロール部は、次のようなテーブルである。ディスクパック上の1論理ブロックに対して、主記憶上の1語(16ビット)が対応する。ファイルに書込する際は、論理ブロックの範囲内においては、書込順に連続したレコードが取られるが、ブロックの変わり目では番地が必ずしも連続しない。今の場合100レコード未満は残となる。

0	コントロール	処理ステータス	次のブロック No.
1			
2			
...			
98			
99			

16ビット

コントロール

- 0 空領域
- 9 ファイルの先頭ブロック
- 8 ファイルの先頭以外のブロック

図 10 List Control

リスト処理の単位を論理ブロックとし、それを今の場合100レコードとしていることに議論が残る。論理ブロックをもう少し少ないレコードで構成したとすれば、補助記憶をより効率よく使用できるが、反面リストコントロール部が大きくなり主記憶が余分に要る。主記憶と補助記憶両方が節約するための折中案として当座は1論理ブロックを100レコードとした。

4.5 処理ステータス情報

受信ファイルは、将来は受信モニタの役目を果させ

ようと考えている。システム開発中は受信ファイルの内容を磁気テープに保存しこれが受信モニタとして使用される。受信ファイルは障害復旧にそなえてデータ処理が終了後も、ラインプリンタと送信が共に完了するまで保存される。またシステムダウンの復旧の際に処理の途中から再開させたいという将来への意図もあって、処理ステータス情報がとられている。処理ステータス情報は受信ファイルの第1番目のブロックに次のように取られる。

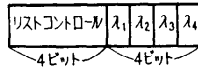


図 11

λ_1 : 受信ステータス

入電文群が受信完了したとき on になる。

λ_2 : 処理プロステータス

データ処理プログラムが完了したとき on となる。

λ_3 : 送信ステータス

出電文群が送信完了したとき on となる。

λ_4 : ラインプリンタ出力ステータス

ラインプリンタ出力が完了したとき on となる。

$\lambda_1, \lambda_2, \lambda_3, \lambda_4$ ともに on のとき全処理完了とみなされ、領域はすべて解放され空領域にもどされる。

5. 障害およびエラー処理

5.1 システムダウンの復旧処理

ファイルのバックアップが入電文に対してのみしかなされていないことを考慮し、当初は次のとおりとする。

状態	受信ファイル	送信/ラインプリンタファイル	
受信モニタ完了以前	無	無	端末より入電文群の最初から再送する。
データ処理プログラム実行中	有	途中	受信モニタより復旧する。
送信またはラインプリンタ出力中	有	有	上記に同じ。

5.2 受信電文のエラー処理

端末からセンタに電文を送る場合の考え方を述べる。

センタで入電文群を受信し、受信モニタに書込完了すると、端末に入電文群受信完了したデータ番号を通知電文により知らせる。端末では、受信完了通知を受

取るとその電文群を以後再送する必要はない。受信途中でシステムダウンになったときは、H電文より再送しなければならない。電文の format エラーがセンタから通知されたら、その電文群の継続する電文は読み飛ばされる。その上センタ側では、その入電文群を消去してしまうので、端末は適切な措置を講じなければならない。

5.3 送信電文のエラー処理

オンラインパッケージにおけるエラー処理に従がう。

6. おわりに

現状および問題点について述べる。

現在のところ、センタ側では、オンラインパッケージのゼネレータ・ログを使用して結合テストを行なっている。さらに47年4月からセンタ側にテスト用端末として FACOM-R を導入し運用面も含めてテストする。X-Y プロッタの部分を除いて一応完成している。

メーカでは、オンライン業務は金融機関に対するものが多らしく、提供された ROS、オンラインパッケージも、原研におけるように純然たる科学技術計算に使用するには、かみ合わない点が多い。特に、FORTRAN がオンラインを目的としたものでなかったということ、処理プログラムをアセンブラで書くことが主だったためかユーザ使用主記憶が ROS における領域区分（リアル単一、リアル多重、バッチ単一、バッチ多重、リアル・バッチ多重）の制約を受けて十分な容量を得られなかった点である。われわれの場合、FORTRAN で書くデータ処理の部分を 45k バイトに押えなければならなかった。このようなことは、中規模計算機でオンラインをやる時の宿命とも思える。

各タスクの所要時間は、データ処理プログラムと、ラインプリンタ出力プログラムを除いて数秒である。ラインプリンタ出力プログラムが1回の出力データが高々 1000 行なので 1分以内である。データ処理プログラムは、FORTRAN の書式変換プログラムが、提供されたものは、主記憶節約のためにドラム上にあるので 1 行分の書式変換にドラムに 10~20 回のアクセスをするのでほとんどの時間がこのために食われ、現在のところ約 6 分かかる。この欠陥を補うために、主記憶常駐の書式変換ルーチンを原研で開発中である。これが完成すれば、データ処理プログラム所要時間は

約1分である。1データ分のデータを送るのに2分～5分かかるとし、全回線5回線中の回線使用が平均2回線とすれば、一応センタ側におけるタスクの処理時間はバランスがとれる、応答時間も10分程度となる。

オンラインプログラムは通常、最終的にはプログラムエラーがないことを前提にしている。ところがFORTRANで書くと、データのエラーで思わぬ結果を招くので、プログラムエラーがあることを予想しておく必要がある。システムダウンをさせないで、エラーメッセージを出して、処理タスクの流れをくずさないようにつなげなければならない。オンラインプログラムにエラーがあっても救われるという前提に立ってこのシステムは作成されている。とはいえエラーを嚴重にチェックしてプログラムを書くことも必要である。

このシステム作成の進行状況は、46年4月からシステムデザインの検討を始め、9月から12月まで、アセンブラ部分約5000ステップ、FORTRAN部分約1000ステップのコーディングを3人で行なって、

現在結合ディバックを行なっている段階である。47年4月からテスト端末としてFACOM-Rを利用し、テストし、本格的に端末も含め稼動するのは、47年10月頃の予定である。

最後に、システム作成にあたって富士通株式会社の大越康臣、山崎敏憲、真田勝吉、伊藤博各氏に有効な情報の提供、システムデザインの検討(特に回線関係)、オンラインパッケージを使用する上での指導などにお力添えいただきましたことを感謝します。

参 考 資 料

1. 富士通株式会社
 - (1) FACOM 230-35 ROS 関係マニュアル式。
 - (2) FACOM 230-35 オンラインパッケージ外部仕様書一式。
2. 馬場澄子他, Ge (Li) 検出器による γ 線スペクトルの解析コード BOB 70, JAERI 1216.
(昭和47年3月3日受付)
(昭和47年4月6日再受付)