

KEIO-TOSBAC タイムシェアリング システム

入出力制御とスーパーバイザ*

土 居 範 久** 佐 藤 文 孝*** 小 堀 英 雄***

Abstract

The small-sized, general purpose time-sharing system, called the KEIO-TOSBAC Time-Sharing System, is being implemented on the TOSBAC-3400-30 computer system at Keio University co-operating with Tokyo Shibaura Electric Co., Ltd. Up to date, on hardware the terminal device (used electric typewriter) and the terminal controller with maximum of 8 terminals have been developed. Using these equipments, the initial system is operating experimentally with the conversational FORTRAN system and the desk calculator system. Furthermore, now being developed the conversational formula manipulating system, the automatic plotting system used with the curve-plotter and so on, and that being formed the plan appending the various features to the system, the system is in a fair way to great growth.

Though the two terminals and the terminal controller currently used are the pilot model, the supervisor is a general purpose and open-ended system.

The terminal controller and the supervisor (Called Version 0) are described in this paper.

1. ま え が き

計算機に基づく協同体系 (computer utility) においては、計算機を他の目的のための単なる道具として利用するだけでなく、人間よりもすぐれている点 (記憶の確かさ、処理の速さなど) を存分に利用して、人間の日常生活の各種事態に対する手段として用いることを究極の目的としている。この協同体系を実際に実現しようとするとき、タイムシェアリング システムが最も適している。しかし、現在の段階では真の意味での協同体系の実現は不可能に近い。だが一定の制約 (磁気コア記憶装置および補助記憶装置などの容量、処理速度、価格および処理可能な端局の数などによ

る) を加えると、ある程度の協同体系が実現できる。この段階でも、非常に有用なシステムである。そして、この段階で得られる経験をもとに、一步一步究極の目的に近づくことができるだろう。

そこで、かなりの制約を加えても、これまでの計算機の使用方式では実現できなかったこと、あるいは実現できたとしても多大な時間と経費を要したようなことが簡単にでき、手段とまではいかないにしても、かなりの程度の道具として利用できるようなシステムの開発を目的として一最終的には、もちろん計算機に基づく協同体系の開発を目的としている一現在、慶応義塾大学では東京芝浦電気株式会社と共同で TOSBAC-3400 モデル 30 を中心に、経済的なタイムシェアリング システムを作成中である。現在までのところハードウェアの面では端局装置および端局制御装置を開発し、ソフトウェアの面では会話型 FORTRAN, LISP, SNOBOL, KFMS (Keio Formula Manipulating system) および卓上計算機 (desk calculator) システ

* KEIO-TOSBAC Time-Sharing System: Input/Output Control and Supervisor, by Norihisa Doi (Department of Administration Engineering, Faculty of Engineering, Keio University), Fumitaka Sato and Hideo Kobori (Tokyo Shibaura Electric Co., Ltd.)

** 慶応義塾大学工学部管理工学科

*** 東京芝浦電気株式会社

ムを作成して、実験的に稼動中である。これをもとに各種機能を追加開発し、除々にではあるが、システムを発展させて行く途上にある。現在の端局装置および端局制御装置はパイロットモデルであり、端局は2台ではあるが、その中心となるスーパーバイザはかなり汎用であり、諸機能を追加することも簡単に行なうことができる。

ここでは、端局制御装置およびスーパーバイザ (Version 0) について述べる。

2. KEIO-TOSBAC タイムシェアリングシステムの概要

現在のパイロットシステム (Version 0) の機器構成 (Fig. 1) およびその機能と用途の概要は次のとおり。

●磁気コア記憶装置

サイクル タイム 800 ns., 容量 16 K 語 (24 ビット/語)。

スーパーバイザ, IOCS および各種基本コマンド処理プログラムを常駐 (約 3 K 語)。

●磁気テープ記憶装置

転送速度 28.8 kc, 4 台。

MT 1 システム用各種コンパイラ, アセンブラおよび各種サービス プログラムを格納。

MT 2, MT 3 使用者ファイルを格納。

MT 4 バックグラウンド プログラムを退避。

(または、端局からの計算機室内にあるラインプリンタ、紙テープさん孔装置などの出力装置への出力要求を処理するた

めに、一時的に、出力データを格納するためにも使用することができる)

●磁気ドラム記憶装置

転送速度 150 kc, 容量 8 K 語, 2 台。

MD 1 使用者原簿, 使用者ファイル原簿およびコマンド処理プログラムを格納。

MD 2 メモリ交換 (memory swap) の際に、使用者プログラムを格納。

●特殊入出力制御装置

端局およびコンソール タイプライタとの入出力を制御。

●端局

通常の電動タイプライタを、端局用に修正したもの。

端局とのデータの授受は、1 字ごとに、特殊入出力制御装置が優先度制御装置を経て中央演算制御装置に割込みをかけることによって行なう。出力データに対するチェックは端局タイプライタの制御装置が行ない、入力データに対するチェックは特殊入出力制御装置が行なう。各端局の制御装置は特殊入出力制御装置によって統合的に管理されている (3.4 参照)。優先度制御装置は中央演算制御装置への割込みを一括して処理していて、前述の特殊入出力制御装置のほかに、周辺機器に対するチャンネル、および磁気コア記憶装置を管理している。また、これらのものはすべて並行処理が可能である。

3. TOSBAC-3400-30 の付加機能

タイムシェアリング システムを開発するために、端局および特殊入出力制御装置を東京芝浦電気株式会社と共同で開発すると共に、TOSBAC-3400-30 自身にも必要最小限の機能を追加した。

3.1 モード

マスタ (master), スレイブ (slave) の二つのモードをもつ。マスタモードにあるときには、計算機はすべての命令を実行できる。また、メモリプロテクションも行なわれない。

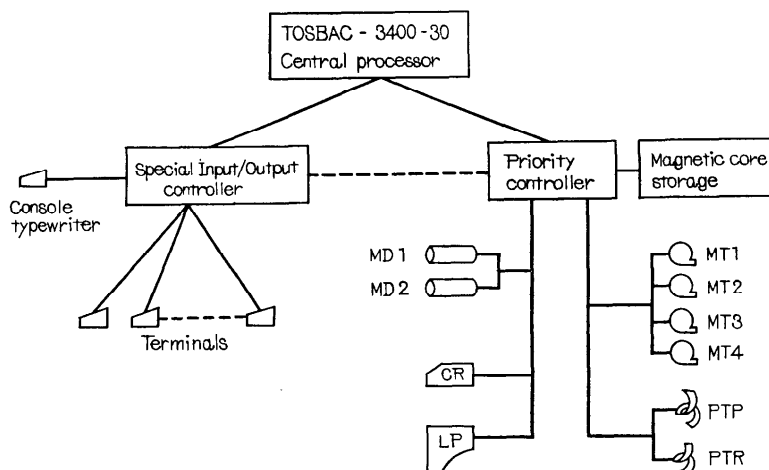


Fig 1. The configuration of the KEIO-TOSBAC Time-Sharing System

マスタ モードに入る条件

割込みがおこったとき、および本体の CONT. CLEAR キー（すべてのレジスタをクリアするためのもので、操作卓上にある。システムを始動するにあたり、スーパーバイザを、まず、補助記憶装置から磁気コア記憶装置に読み込むときには、必ず、このキーが押される）を押したとき。

スレイブ モードに入る条件

プロテクトされている領域から、プロテクトされていない領域へ飛越命令で分岐したとき。

割込処理は、特定の割込飛先番地（割込みの種類に従って、割込みがおこったときにコントロールが移る番地は定められている）に JTL (Jump and Transfer Link: この命令の番地部に指定されている位置 (m) に、この命令が格納されている番地に 1 を加えた値を格納し、($m+1$) 番地に分岐する) 命令を入れておき、割込みがおきると、割込みの種類により、JTL の番地部に指定しておいた所定の割込処理ルーティンに分岐させることによって行なう。このとき、JTL の番地部で指定されている番地がプロテクトされている領域内ならばマスタ モードのままで割込処理をすることになり、JTL の番地部で指定されている番地がプロテクトされていない領域ならばスレイブ モードで割込処理をすることになる。（現在の KEIO-TOSBAC タイムシェアリング システムでは、割込処理はすべてプロテクトされている領域内において、マスタ モードで行なっている。）

プロテクトされている領域内において、マスタ モードで割込処理後、割込まれたプログラムにコントロールを戻すには、JTL 命令の機能により割込処理ルーティンの先頭に格納されている番地へ、分岐すればよい。このとき、この番地がプロテクトされていない領域にあるときには、スレイブ モードに入る。

割込飛先番地に JTL 命令以外の命令を入れておいたときには、割込みがおきると、マスタ モードに入り、その飛先番地の命令を実行後マスタ モードのまま割込まれたプログラムにもどる。ただし、割込飛先番地に飛越命令があり、その番地部がプロテクトされていない領域を指している場合には、その命令を実行すると、コントロールは番地部に指定されている場所に移ると共に、ストレイブ モードに入る。

マスクによって割込みが禁止されている場合には、割込原因が生じても割込みはおこらず、もちろんマスタ モードに入ることもない。

3.2 特殊命令

次の命令は、マスタ モードにあるときに限り実行可能である。

SEIP (Select Input) SEOP (Select Output)
RCS (Read Console) WCS (Write Console)
SER (Select Read) SEW (Select Write)
RLCN (Reset and Load Channel)
LCN (Load Channel) SECN (Select Console)
WTM (Write Tape Mark) ERA (Erase)
BSR (Back Space Record)
BSF (Back Space File) RWD (Rewind)
HJ (Halt Jump) HP (Halt Proceed)
ETM (Enter Trapping Mode)
LTM (Leave Trapping Mode)
PM (Protect Memory)
SII (Set Indicator from Interruption)

上記命令がスレイブ モードで出されたときには、原則として、その命令は NOP (No Operation) となり、グループ 1 の割込み（優先度が 2 番目の割込みで、演算の際のレジスタ関係の割込みなどが属する）が起る。上記の命令の中には、その命令の直後で割込みを禁止しているものもあるが、スレイブ モードでこれらの命令が出されたときは、直ちに割り込む。

3.3 メモリ プロテクション

スレイブ モードにあるときには、コアを読み書きする度に境界レジスタ (boundary register) の内容と読み書きする番地とを比較する。その番地がプロテクトされるべき領域になれば、そのまま読み書きが行なわれるが、プロテクトされるべき領域にある場合には、その読み書きを行わずにその命令を終り、その後でグループ 1 の割込みをおこす。飛越関係命令に対しても同様の処理を行なう。プロテクトされるコア記憶装置の領域は二つの境界レジスタ (U レジスタおよび L レジスタ) で指定する。U レジスタはプロテクトされない領域の上限を定め、L レジスタはプロテクトされない領域の下限を定める。プロテクトは 256 語単位で行なう。

3.4 特殊入出力装置

端局としては、タイプライタとスイッチ ボードを設けた。各端局は特殊入出力制御装置 (SIOC: Special Input/Output Controller) を通して、コンソール入出力装置の一種として接続する。接続できる端局数は最大 8 局（この SIOC をもとに開発した、TOSBAC DN-230 では最大 32 局）である。

端局とのデータの授受は1文字ごとに RCS (Read Console) または WCS (Write Console) 命令を用いて行ない、各端局の同時動作を許すためにグループ2の割込み(優先度が3番目の割込みで、入出力関係の割込みが属する)をおこす。この割込みは、RCS あるいは WCS 命令がどの端局に対して働くかをもきめる。さらに、端局タイプライタに電源がはいっていないときにも、本体から電源を投入し、出力することができる。

SIOC からの割込みはグループ2の割込み1種だけで、SII (Set Indicator from Interruption) 命令を出すと、割込みをおこした端局番号と割込みの種類コードがインディケータレジスタにセットされる。種類コードとしては、次の3種類のものがある。

- (1) ERC (End of Receiving of a Character) 1文字受信完了
- (2) ETC (End of Transmission of a Character) 1文字送信完了
- (3) CS (Channel Signal) チャンネル シグナル受信完了

ERC の割込みがおこったときには、RCS (Read Console) 命令を出してデータをアキュムレータに受けとり、ETC の割込みがおこったときには、WCS (Write Console) 命令を用いてデータを送信する。RCS または WCS 命令を出さずに、ETM (Enter Trapping Mode) 命令を出すと、再び同一の割込みが生じる。また二度以上出ると、二度目以降は通常のコンソール入出力装置に対して働く。

CS の割込みに対しては、RCS 命令を出すことにより、それまでの、その端局に対する入力データに関する情報(インタラプションコード)を読み取ることができる。インタラプションコードのおもなものは次のとおり。

- SIOC の状態と出された命令とが矛盾している
- データの重なり合いが生じた
- データが完全でない
- キャリヤが切断した
- 端局からの受信が完了した

各データに対するパリティチェックは SIOC が行ない、その結果は、データのパリティビットの位置に入れる。

端局とのデータの授受を行なうために従来の命令にも改良を加えた。SEIP (Select Input) および SEOP (Select Output) 命令は、コンソール入出力装置の稼

動状態を指示するものであったが、これらの命令の第16ビットを1にしたときには SIOC に対するものであるように拡張した (Fig. 2)。

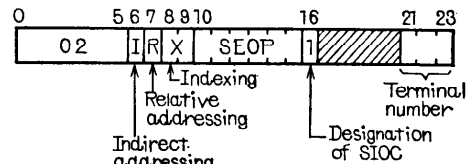


Fig 2. Extended SEOP instruction

SIOC に対するこれらの機能は次のとおり。

- SEOP この命令で指定された端局へ STX (Start of Text) コードを送り、以後データが端局へ送られる状態にする。
- SEIP 端局へのデータ送信後、この命令を出して ETX (End of Text) コードを端局へ送る。

SEIP 命令を出すと、直ちに、その端局からそれまでに送信したデータに関する情報 (STATUS コード) が送られてきて、ERC の割込みが生じる。STATUS コードのおもなものは次のとおり。

- データが完全でない
- データの重なり合いが生じた
- 端局がオフラインの状態にあった
- インタラプションスイッチが ON であった (多行にわたる印字を途中で中断したいときなどに用いる)
- データ送信後一定時間経過しても端局側では何の動作も行なわれない

4. スーパーバイザ

スーパーバイザは、タイムシェアリングシステムの中核であり、スーパーバイザに組み込まれている精神によって、そのシステムの性格がきまる。

KEIO-TOSBAC タイムシェアリングシステムでは、端局からの入力はどうのような状態にあるときにも受信可能であるように、各端局には優先度をつけず、原則的には、各端局は平等に順次処理するように、端局との応答は会話形式で行なうように、そしてシステムは拡張可能であるようにすることを前提としてスーパーバイザを作成した。

使用者の立場に立つと、端局を使用する際の状態は次の2種に大別できる。

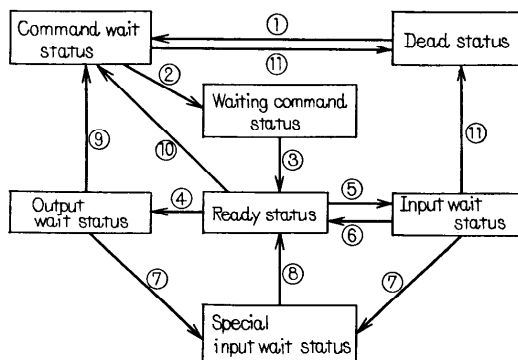
- プログラミングモード

● コマンド モード

そして、使用者がシステム内で位置する状態としては次の7種のものがある。これらの間の遷移関係を、Fig. 3 に示す。

- 端末未使用状態 (Dead status)
 - コマンド待ち状態 (Command wait status)
 - コマンド受信完了状態 (Waiting command status)
 - 実行可能状態 (Ready status)
 - 入力待ち状態 (Input wait status)
 - 出力待ち状態 (Output wait status)
 - 入力完了待ち状態 (Special input wait status)
- (SIOC の命令矛盾が生じたときに起る特殊な状態)

次にスーパーバイザを機能別に分類し、その中核となるものを記述する。



- ① 端末から HELLO と打つことによって始まる一群の会話の終了 (使用者番号, 使用サブシステム名, プログラム名の登録およびファイルの有無)
- ② 端末からのコマンドの受信完了
- ③ 処理装置を使用する順番の到来
- ④ スーパーバイザに出力要求
- ⑤ スーパーバイザに入力要求
- ⑥ 端末からの入力の受信完了
- ⑦ 端末へ出力しようとしたとき, 端末側が入力中 (端末の入力を要請するときには, 入力するように知らせる出力がまずある)
- ⑧ 端末からの入力の受信完了
- ⑨ 端末への出力の送信完了等
- ⑩ 端末からの実行中断信号の受信完了
- ⑪ 長時間端末からの応答がない

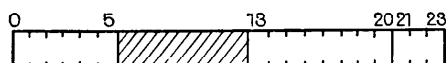
Fig 3. The transition of the user's status within the system

4.1 端末割込処理

割込みを媒介として端末との情報の交換をつかさどるものである。前述したように, SIOC の割込みには3種類のものがある。これらの割込みに付随した処置は次のとおり。

- ERC 入力データを読み取る。SEIP を出した直後の場合には, STATUS コードを読み取る。
- ETC データをコード変換し送信する。データ送信終了直後の場合には, SEIP 命令を出す。
- CS インタラプションコードを読み取る。

入力 各端末からの入力情報(インタラプションコード, STATUS コードおよびデータ)を端末番号と共に1語に構成し (Fig. 4), 受信した順にプールバッファ (大きさに関しては 4.2 参照) に格納する。プール バッファはサイクリックに使用する。



0~5 種類別コード (Sort code)

00 データ (上段) (Data (Upper case))

40 データ (下段) (Data (Lower case))

20 STATUS コード (STATUS code)

30 インタラプションコード (Interruption code)

34 STATUS コード後のインタラプションコード

(Interruption code after STATUS code)

13~20 データ, STATUS コード, またはインタラプシ

ョンコード (Data, STATUS code or Interruption code)

21~23 端末番号 (Terminal Number)

Fig 4. The format of the input information within the pool-buffer

出力 各端末へ出力すべきデータは, 出力処理ルーティンが各端末の出力用バッファへ格納後 SEOP 命令を出す。その後の ETC の割込みから端末割込処理ルーティンが1字ずつ処理してゆく。システムからの出力に対する英字は小文字で送信する。本体と端末とのコードが異なるので, 1字ずつ送信する際にコード変換を行なう。割込処理に要する時間は1データ当たり約 100~150 μ s である。

4.2 計時割込処理

時計は5番地にあり, 20 ms ごとに5番地の内容から1ずつ引き, 0から77777777₈に変わるときにグループ0の割込み (最高の優先度をもつ割込みで, 電源異常やメモリ エラーなどがこのグループに属する) がおきる。この時計を, 時分割, 使用者に対するアカウントリングおよび時刻の算出などに用いる。計時の割込みがおきる間隔は現在 200 ms である (この時間間隔にもとづいて, プール バッファの大きさが定まる)。計時割込みがおきると, まず時刻を更新し, アカウントリングを行なう (Fig. 5 (a)). 次にプール

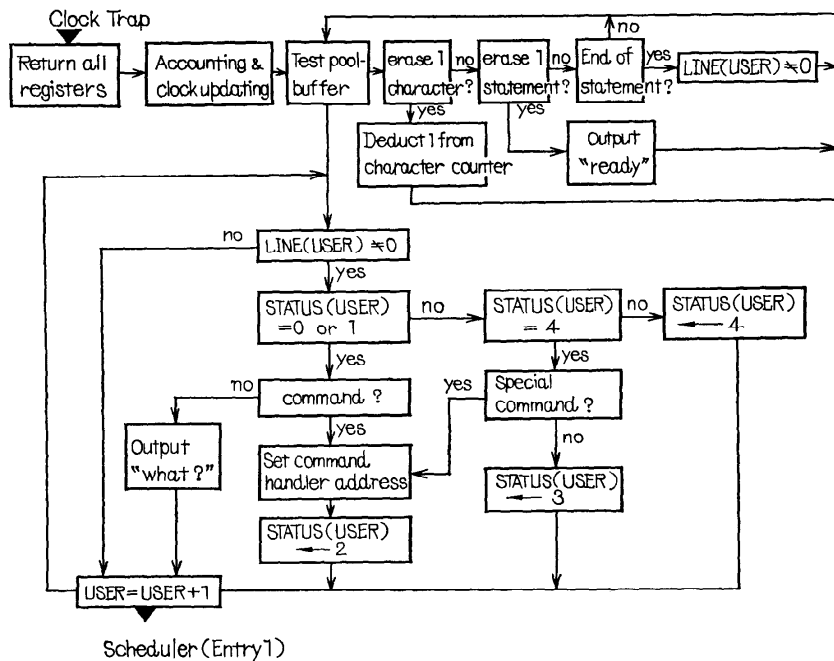


Fig 5. (a) Clock trap handler

バッファ内の情報（前回未処理の位置から現時点までにうめられた位置まで）を調べ、データの場合には端局ごとに用意した入力バッファにコードを変換し、格納する。インタラプションおよび STATUS コードの場合には、それぞれコードに基づいた処理を行なう。ただし、STATUS コードに対する処理は、そのコードに対するインタラプションコードの解析後に行なう。STATUS およびそれに対するインタラプションコードが正常な場合には、出力が完了したことになり、使用者は所定の状態に移る。入力データに対するインタラプションコードが正常な場合には、入力が完了することになる。入力が完了した使用者に対しては、ひき続き使用者の状態を調べ、システム コマンドかサブシステム コマンドか、あるいはサブシステムに対するデータかなどの検査を行なう。システム コマンドに対しては、コマンド受信完了状態となり、他の場合には実行可能状態となる。

4.3 スケジューラ

スケジューラ (Fig. 5 (b)) は、次にどの使用者を処理するのかを決め、対応する処置をとるものである。KEIO-TOSBAC システムでは、使用者間あるいは端

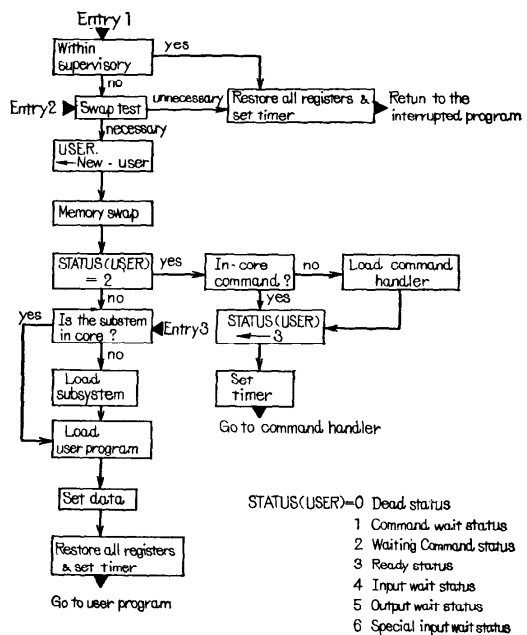


Fig. 5. (b) Scheduler

局間に優先度はつけず、すべて平等に処理を行なう。

スケジューラは、計時割込処理ルーティンからの入口（入口1）、スーパーバイザルーティンからの入口（入口2）、および、その他特別なコマンド（たとえば、PROGRAM（プログラミングモードへ切替えるためのコマンド）からの入口（入口3）の3種の入口がある。

入口1（Entry 1） スーパーバイザルーティン処理中に計時割込みが生じたのかどうかを調べ、処理中の場合には直ちにそのルーティンにもどる。その他の場合には、入口2に移る。

入口2（Entry 2） 現在の使用者が一定時間（1秒）を経過したかどうかを調べ、経過していないときには直ちに使用者のプログラムへもどる。経過した場合、あるいは、その使用者が実行可能状態以外の状態に遷移していたときには、他の使用者の状態を調べ、実行可能状態またはコマンド受信完了状態にある使用者があるときには、その使用者を次の使用者として定める。どの使用者も実行可能状態またはコマンド受信完了状態にないときには、現在のところシステムは使用者待ちの状態になる（バックグラウンドとしてパッチ処理を組み込んだときには、この使用者待ちの状態のときにこれを処理することになる）。

使用者交換の際には、使用者のプログラム部分だけを磁気ドラムに掃き出す。

次に、コマンド受信完了状態にある使用者は、実行可能状態にして、時計をセットしてから各コマンド処理ルーティンに移る。その他の場合には入口3に移る。

入口3（Entry 3） 使用しようとするサブシステムが、現在コア記憶装置内にあるものと異るときには、システムテープから必要とするものを読み込む。そして使用者のプログラム部分がすでに磁気ドラムにある場合（ファイルされているものを使用するときでも、この時点ではすでに磁気ドラム内に移されている）には、これをコアに読み込み、入力データを所定の位置に移し、レジスタ、時計などをセットしてから、サブシステムへコントロールを移す。これからプログラムを作成する使用者に対しては、時計だけをセットしてからサブシステムにコントロールを移す。

4.4 スーパーバイザ呼出処理

使用者のプログラムでデータを端局へ出力したいとき、端局からのデータを必要とするとき、実行を終了するときなどは特殊命令の内の HJ（Halt Jump）命令を用いて割込みを起すことによってスーパーバイザ

と連絡をとる。スーパーバイザの方では、その呼出命令群を用いて、要求された処理を行なう。

4.5 ファイルの管理

現在のシステムでは、使用者のファイルは磁気テープに保持している。このためファイルの管理は暫定的に簡単な方法をとっている。

ファイル管理のためには、次の3種の原簿を使用する。

- 端局別ファイル原簿〔MFD: Master File Directory〕（1語/端局）

各端局について、その端局に対する UFD（後述）の有無を記録。

- 使用者原簿〔UFD: User File Directory〕（3語/1登録）

ファイルを登録した使用者に対して作成する原簿で、1使用者につき1登録からなり、原簿の語構成は次のとおり。

0	使用者番号.....
1	
2	最初の OFD 格納位置

- 使用者ファイル原簿〔OFD: Own File Directory〕（4語/1登録）

使用者がファイルとして登録したプログラム（現在のシステムではデータだけをファイルすることはできない）の名前、使用したシステム名などを登録しておくもので、1プログラムにつき1登録からなる。原簿の語構成は次のとおり。

0	プログラム名.....
1	
2	サブシステム名
3	テープ位置と次の OFD の位置

端局別ファイル原簿はコア内に保持していくが、他の原簿はすべて磁気ドラムに保持している。これら原簿間の関係を Fig. 6 に示す。

現在のシステムでは前述のとおり、ファイルとして保存できるのはプログラム（完成、未完成には無関係）だけである。端局番号、使用者番号、プログラム名、およびサブシステム名が合致して初めてファイルが利用できる。現在ディスクに基づいたファイルシステムを開発中であるが、これが完成すると、データもファイル可能となり、さらに他の使用者が開発したプロ

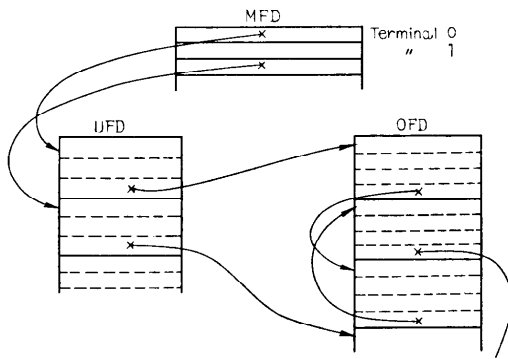


Fig. 6. The interrelation of the directories

ラムも利用可能になるなど、かなりの程度まで完備される。

5. む す び

現在すでに開発されているサブシステムおよびその概要は次のとおり。

●会話型 FORTRAN システム

モニタ、構文解説、解説実行および各種コマンド処理ルーティンからなり、すべて純手続 (pure procedure) である。ファイルとしては構文解説ルーティンで生成されたポーランド記法に基づいた木構造のプログラムが貯蔵される。実行に際しては、解説実行ルーティンがこれを処理し、プログラムリストの要求に対しては LIST コマンド処理ルーティンがもとの FORTRAN ステートメントを再成する。プログラミングに際しては、1 ステートメントごとに構文解説ルーティンが文法をチェックする。通称 KEIO システム (Keio Elementary Instructive Operating System) という。

●卓上計算機 (desk calculator) システム

FORTRAN 言語に類似のステートメントを用い、1 ステートメントごとに計算を行ない折返し答を印字する。計算結果は 100 個まで、保存しておくことができ、この結果は名前で利用することができる。初等関数を内蔵するほか、各種コマンドが用意されている。システムは純手続きである。

この他にも LISP, SNOBOL および KFMS (Keio Formula Manipulating system) が開発されており、すべて会話型で純手続からなる。また、前述のファイルシステムを初め、アセンブラ、カーブプロッタ用言語などを開発中であり、さらにファイルシステムはバックグラウンドの処理も並用できるよう計画してい

る。また、これらの作業と平行して、タイムシェアリングシステムのシミュレーションを行ない各種情況に適合したシステムの研究も行なっている。

1968 年夏には、現在のパイロットモデルをもとに開発した装置—端局装置は TOSBAC DN-510、端局制御装置は TOSBAC DN-230 (TOSBAC DN-510 を最大 32 台まで接続可能)—を採用することにより、スーパーバイザもさらに拡張充実したものになる。

タイムシェアリングシステムとしては非常に小型なものではあるが、周辺装置を拡充することによりかなりの程度のものが可能であり、この経験をもとに、一步一步真の意味での計算機に基づく協同体系の開発を進めてゆくつもりである。

最後に、この論文を書くに至るまでには、システムの開発に際し、多数の方々のご協力を戴いたが、特に、共同研究者の原田賢一 (慶応義塾大学工学部中央試験所助手)、中西正和、大野瑞夫、石渡裕之、大野義夫、松本雅雄 (慶応義塾大学工学部管理工学科) の諸兄、直接いろいろとご指導、ご援助を戴いた浦昭二教授、および多大なご尽力を戴いた山内二郎教授には絶大な感謝をする。また、東京芝浦電気株式会社電子計算機事業部第二電子計算機課の天羽課長、松下主任を初めとする課員諸氏およびサービス部整備課の大竹 繁、萩原 優の両氏、JBA コンピュータ技術部第一課の桑木課長、川野 誠、重富与八郎の各氏には、いろいろとご迷惑をおかけしたにもかかわらず、並々ならぬご援助を戴いたことを感謝する。

参考文献

- 1) TOSBAC-3400 電子計算組織命令説明書。
- 2) 土居範久、原田賢一：タイムシェアリングシステムに関する研究、慶応義塾大学大学院工学研究科修士論文 (1965) [未発表]
- 3) 中西正和、中村紳也：タイムシェアリングシステム用コンパイラ、慶応義塾大学工学部管理工学科卒業論文 (1965) [未発表]
- 4) 土居範久、原田賢一、中西正和、大野義夫、石渡裕之、松本雅雄：卓上計算機システム、KEIO/TOSHIBA TSS. TR-1 (1967)
- 5) 土居範久、原田賢一：入門タイムシェアリングシステム、KEIO/TOSHIBA TSS. TR-2 (1967)
- 6) 浦 昭二、土居範久、原田賢一：タイムシェアリングシステム、数学セミナー、10月号—12月号 (1967)
- 7) 土居範久：KEIO-TOSBAC タイムシェアリングシステム、情報処理月例会資料 33 (1968)。(昭和 42 年 10 月 16 日受付)