

## インテリジェント・コンソールを用いた 計算センタ運用管理システム†

有田 五次郎††

計算機システムの運転状況、利用状況を計算機自身で監視し、利用に関する情報の管理や利用状況に応じたシステムの制御、システムの運転操作等を自動的に行うシステムは、運用管理システムと呼ばれる。このような機能を持つ運用管理システムを構築するためには、計算機上で動作している種々のプログラムの状態のみならず、計算機本体および入出力機器のハードウェアの状態、空調機や警報装置、場合によっては電話回線等の外部環境までも監視・制御する機能が必要になる。従来のオペレーティングシステム (OS) ではこのような配慮はほとんどなされておらず、これらはオペレータの仕事となっていた。オペレータはオペレータコンソール経由で計算機システムと通信しながら、システムや外部環境の監視・制御を行っている。インテリジェント・コンソール (INC) は、計算機とそのオペレータコンソールの間にそう入されたマイクロコンピュータシステムで、拡張 OS として動作し、またオペレータの機能を代行することができる。このオペレータの代行機能に着目すると、INC は運用管理に対して種々の応用を持っている。本論文では運用管理の考え方と INC の概念を述べ、運用管理システムに対するいくつかの INC の応用例を示す。

### 1. はじめに

汎用計算機における運用管理システムは大学の大型計算機センタなどで研究開発が進められてきたが<sup>1)</sup>、省力化、省資源の要求が高まるにつれ、多くのセンタ形式の計算機利用者に採用されるようになった<sup>2)</sup>。

計算機の利用形態、運用方式は使用者\* ごとに異なっており、したがって運用管理システムも使用者ごとに異なってくる。従来のオペレーティングシステム (以後 OS と書く) では、ジョブ制御マクロ、カタログド・プロシジャ、ユーザー・ルーチンへの出口の設定、システム編集等の方法で、計算機利用方式の使用者ごとに異なる部分に対応してきた。これらは OS があらかじめ定めている機能に対して変更・追加を可能にするもので、変更・追加される機能はすべて OS の管理下にある。このため、利用状況に応じてシステムパラメータを変更するなど、OS 自体を監視・制御する機能については配慮されておらず、これらの機能はオペレータ (計算機管理者または操作員) にまかされている。

自動運転を含む運用管理システムを実現するためには当然各種の外付けハードウェアを必要とするが、最も重要な問題は OS とのインタフェースであろう。OS が拡張可能に構成されていれば新しい機能を追加

することも容易であるが、そのように構成されていない従来の OS の中に、利用者が望む各種の運用管理ルーチンを組み込むことは、かなり困難を伴う作業となる。

インテリジェント・コンソール<sup>3)</sup> (以後 INC と書く) は OS の標準インタフェースのみを用いて、外付けのハードウェア・ソフトウェアによって OS の機能を拡張しようとするもので、オペレータの機能を代行することも可能である。利用資格の検査や、計算時間の計測など、OS が直接管理する必要のある仕事以外の運用管理の機能はオペレータの機能の代行であるから、INC を用いることにより比較的簡単に種々の運用管理のための機能を実現することができる。

本論文では運用管理に対する INC のいくつかの応用について述べる。本論文は次の構成をとる。まず第 2 章で運用管理システムの一般的な機能について述べ、次に第 3 章で運用管理の立場から見た INC の概念と実験システムの構成を示す。第 4 章では運用管理システムに対する INC の応用例を示し、第 5 章ではまとめとして、運用管理に INC を用いる場合の効用、問題点等について述べる。

### 2. 運用管理システム

計算機に係わる立場は、作る側 (メーカー)、使わせる側 (センタ)、使う側 (ユーザ) の三つに分類できる。運用管理システムはセンタの立場に立ったシステムである。

運用管理システムは多くの側面を持っており、これ

† An application of the intelligent console to the operation management system for a computation center by ITSUJIRO ARITA (Department of Computer Science and Communication Engineering, Faculty of Engineering, Kyushu University).

†† 九州大学工学部情報工学科

\* ここでいう使用者は計算センタまたは計算機運用責任者である。

らを次のように分類することができる。

A. 利用者の管理

利用資格の登録, 利用資格の検査, 使用記録, 利用明細 (請求書) 発行等.

B. 利用状況の把握, 制御

ジョブ種別ごとの優先制御, 改札 (利用制限) 機能, 利用状況に応じた利用方式の変更, システムパラメータの変更, 運転記録, 利用統計等.

C. 省力化および自動運転

花文字 (ジョブの区切りの情報) 印刷・LPカッタ, オープンバッチ・オープンスターション, 出力検索 (出力データ編集)・デマンド出力, 自動 IPL・自動再開機能, オペレータメッセージへの自動応答, 障害検出・警報機能等.

D. 利用者サービス

ジョブの完了表示・完了通知, ジョブの完了問合せ, システムの状況問合せ等.

これらの機能を実現する手段としては次のような方法が考えられる.

- (1) OS が本来持っている機能を利用する.
- (2) OS が組み込みを予定している使用者作成ルーチンとして構成する.
- (3) ジョブとして実行可能な応用プログラムとして構成する.
- (4) オペレータが実行する.
- (5) OS を改造して新しい機能を追加する.
- (6) 外付けハードウェア, ソフトウェアを用意する.

従来の OS では, 一般には(1)~(4)までしかゆるされてなく, したがって OS が予想していない機能の実現や自動運転には(5), (6)によって対処しなければならなかった. 実際, 大学の大型計算機センタを始め多くの運用管理システムはこのようにして実現されてきた<sup>3)-7)</sup>.

従来の運用管理システムの実現方法は, 目的に応じてハードウェアを用意し, それに対応して OS を改造する手法をとっている. このため, 目的および対象システムが異なればそれぞれに対して異なる対応を要求される. 運用管理システムは使用者 (センタ, 計算機運用責任者) ごとに異なる. したがって, 使用者が運用管理システムを自由に設計し, 製作できるように OS が構築されていることが望まれる. これは, 運用管理に利用できる汎用のユーザインタフェースを OS が用意することによって実現できよう.

INC は既存の OS に付加した, ハードウェア, ソフトウェアの汎用ユーザインタフェースである.

INC を運用管理に应用することによって, 運用管理のために OS のユーザインタフェースが持たなければならない機能が明らかになる.

### 3. INC の概念と構成

INC の一般的な機能・構成についてはすでに述べたが<sup>8)</sup>, ここでは運用管理に必要な INC の概念と運用管理を目的とした実験システムの構成とを示す.

#### 3.1 INC の概念

OS はオペレータとの通信機能を実現するコンソールサービス (CONS) を持っている. OS 内の制御モジュールは CONS の機能を用いてオペレータと通信することができる. CONS はまたユーザプログラムとオペレータとの通信機能を持っている. ユーザプログラムは WTO (Write To Operator) マクロでオペレータにメッセージを送ることができ, WTOR (Write To Operator with Reply) マクロでオペレータからの応答を受取ることができる. この様子を **図 1** に示す.

INC は **図 2** に示すように計算機とオペレータコンソールの間にそう入されたマイクロコンピュータ・システムであり, ホスト計算機の拡張 OS として動作することができる. ここで CCI, SCI, ECI, DCI は, それぞれコンソールインタフェース, システム制御インタフェース, 環境制御インタフェース, 装置制御インタフェースで, 必要に応じて設けられる.

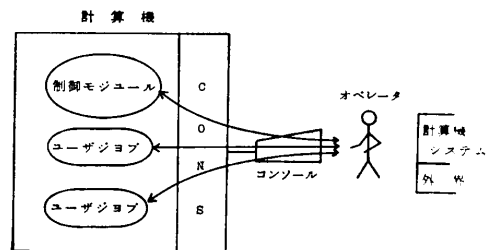


図 1 オペレータの役割  
Fig. 1 A role of an operator.

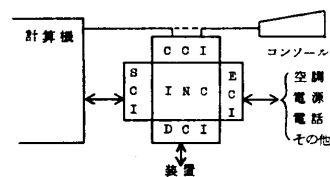


図 2 INC の概念  
Fig. 2 Conceptual structure of INC.

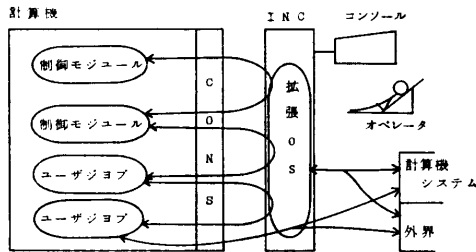


図3 INCによって可能になる通信経路  
Fig. 3 Message paths enabled by INC.

INC はオペレータの代りにシステムメッセージを監視し、操作コマンドを発行することができ、またメッセージの仲介者として、ユーザプログラム同士、ユーザプログラムと OS の制御モジュール間の通信機能を実現し、さらに SCI, ECI を通じて計算機システムおよび外界に働きかけることができる。この様子を図3に示す。

INC を利用することにより、ユーザジョブを制御プログラムとして動作させたり、ユーザジョブから OS の制御モジュールを監視・制御したり、外界および計算機システムを制御したりすることができる。これらは自動運転を含む運用管理システムにとって必須の機能である。

### 3.2 INC のハードウェア構成

ホスト計算機のコンソール装置のインタフェースを持つことを除き、INC は通常のマイクロコンピュータ・システムである。我々は情報工学科に設置された教育用計算機 FACOM 230-45 S および大型計算機センタに設置された共同利用大型計算機 FACOM M-200 (以後 F-45 S, M-200 と書く) のために 2 台の INC (以後それぞれ INC1, INC2 と書く) を製作し実験を行った。

INC1 および INC2 のボード構成を表1に示す。

F-45 S および M-200 のコンソールはいずれも IOC と呼ばれる入出力制御装置に接続される。このインタフェースは FACOM IOC インタフェースと呼ばれ、入力側および出力側にそれぞれ、8ビットのデータ線、16ビットの制御およびタグ線を持った低速の並列入出力インタフェースである<sup>9)</sup>。表1の IOC インタフェースボード、タイプライタインタフェースボードは、IOC インタフェースの装置側および計算機側のインタフェースで、これの制御はプログラムによっている。これらのインタフェースは並列入出力用 LSI, PIA を用いて簡単に実現できるが、実験システムではソフトウェアの負担を軽減するため、ケーブルドラ

表1 INC のボード構成

Table 1 A list of boards constructing INC 1 and INC 2.

ボード名称	内容	INC 1	INC 2
CPU ボード	プロセッサ M6800 ROM 16 k 2716×8	○	○
RAM ボード	RAM 48 k 4116×24	○	○
IOC インタフェースボード	IOC との全インタフェース信号	○	○
タイプライタ インタフェースボード	タイプライタとの全インタフェース信号	○	
UDC インタフェースボード	300~9600 bps 8 回線 ACIA×8	○	
汎用 I/O ボード	PIA×2 ACIA×1 SSDA×1 Timer×1	○	
SSDA ボード	ACIA×1 SSDA×1		○

イバ/レーバを含めておのおの約 40 個の TTL-IC で実装した。また SSDA ボードは大型計算機センタと情報工学科を結ぶ 48Kbps の直流モデムのための同期式回線インタフェースである。

なお、INC のハードウェア、ソフトウェアを開発するため、また運用管理のための各種の機器制御の実験を行うため、ディスプレイ、プリンタ、デジタルカセット等を持ったマイクロコンピュータ (UDC, Universal Device Controller と呼ぶ) を使用した。INC1, INC2 および UDC は同一のバス構成を持ち、各ボードには互換性がある。

### 3.3 ホスト計算機との接続

システム全体の接続図を図4に示す。F-45 S および M-200 はいずれも稼働中の計算機であり、使用中のコンソールに INC を接続して実験することは不可能である。しかしこれらの OS はいずれもマルチコンソールの機能を持っているので、IOC に接続された未使用のタイプライタをサブコンソールに割り当て、これに INC を接続して実験を行った。

M-200 の OS では、システムメッセージがいくつ

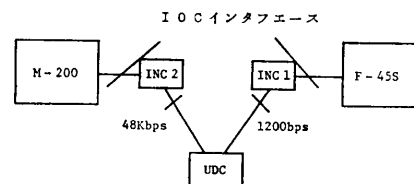


図4 システム接続図

Fig 4 Configuration of the experiment system.

かのクラスに分れており、サブコンソールから指定した特定のシステムメッセージ、たとえばジョブの開始、終了のメッセージだけをそのコンソールに出力させることができる。このシステムメッセージの傍受機能と、システムの動作状態を問合せる DISPLAY コマンドの使用により、システムの動作状態を INC で監視・制御することが可能となる。

F-45S の OS にはシステムメッセージの傍受機能が無く、サブコンソールにはそのコンソールに関係したメッセージだけしか出力されない。このため、システムの動作状態を常時監視する必要がある場合にはサブコンソールをメインコンソールに切り換える必要がある。これはメインコンソールを一時、ハードウェア的にシステムから切り離すことによって行われる。

いずれの場合も INC はコンソール装置に出力されるメッセージを途中で「見る」だけであり、ホスト計算機の OS には全く手を加えずに標準の OS の上に運用管理システムを構成することができる。

### 3.4 INC 上の基本ソフトウェア

実験システムの INC はコンソールインタフェース、回線インタフェースおよび運用管理に必要な機器インタフェースのみを持っており、一般の入出力機器、ファイルを持っていない。したがって応用を決めて運用を行うためのソフトウェアは ROM 上に固定化しておかなければならない。しかし我々の場合は各種の応用について実験を行うのが目的であるので、RAM の上で種々のプログラムを動かす、しかも UDC と INC が離れているため遠隔地からデバッグができる必要がある。このため実験システムでは INC の基本ソフトウェアを一種のネットワークソフトウェアとして構成した。

一般に複合計算機システムのソフトウェアは、その OS の構造によって次の二つの型に分類できよう。

第1は、各計算機が独立な、完結した OS を持ち、OS の機能の一つとしてプロセス間通信のプロトコルを持つもので、ここではこれを対称型ソフトウェアと呼ぶ。ネットワークアーキテクチャはこの代表的なものである。

第2は、各計算機が必ずしも独立した OS を持たず他の計算機上の OS の制御の下で動作するもので、これを分散型ソフトウェアと呼ぶ。

分散型ソフトウェアを構成するために必要な機能は以下の通りである。

(1) read/write アドレスを指定して他の計算機

のメモリ領域を読み書きする。

(2) branch 指定したアドレスから、他計算機上のプログラムを実行させる。

(3) lock/unlock メモリ領域の排他制御。

(4) system reset システムの初期設定。

INC のソフトウェアをメッセージレベルでの分散型ソフトウェアとして構成するため、これらの機能を INC の ROM 上に実現した。ただし一般の並列処理を行うのが目的ではないので、(3)の機能は実現していない。

このような基本ソフトウェアを用意することによって、ホスト計算機または UDC から INC にプログラムを送り、それを走らせ、デバッグすることができる。

なお(4)の機能はデバッグ中にプログラムが暴走したとき初期状態に戻し、システムのハングアップを防止するための機能である。

## 4. 運用管理に対する INC の応用

この章では運用管理に対するいくつかの INC の応用例を示す。

### 4.1 利用記録、運転記録

小型計算機やミニコン等ではジョブの利用記録やシステムの運転記録の収集の機能を持たないものも多い。このようなシステムをオープン使用する場合には、利用状況の管理にコンソールタイプライタの記録が用いられる。しかしコンソール出力は計算機利用者が必要とすることもあり、またその整理もめんどうである。

INC はホスト計算機の拡張 OS であり、コンソールメッセージを解析することによって利用記録やシステム運転記録の収集を自動的に行うことができる。記録用のファイルは運転記録用ジョブを起動することによってホスト計算機のファイルを利用することもできるし、INC に記録用ファイルを設置することも容易である。また、もしシステムメッセージが時刻の情報を含んでいない場合には、INC にカレンダー付きタイマを用意することにより時間記録を収集することができる。

ミニコンや小型計算機のコンソールインタフェースの多くは標準の回線インタフェース (RS-232C) であり、現在市販されているパーソナルコンピュータのほとんどはこのインタフェースを持っている。したがって、ミニコンや小型計算機のコンソールとしてこれら

のパーソナルコンピュータを接続し、この上に適当なプログラムを開発することによって INC とすることができる。

4.2 自動運転

自動運転システムにも種々のレベルが考えられる。ここではシステムの起動・停止の自動化、システムダウンの自動回復、異常警報・緊急停止等を目的とするシステムを F-45 の上で考える。

自動運転システムでは次のようなシステム制御、環境制御のインタフェースが必要になる。

SCI (System Control Interface)

CPU: スイッチ power on/off, reset, IPL

表示信号 ready, machine check

DISK: スイッチ power on/off, unload

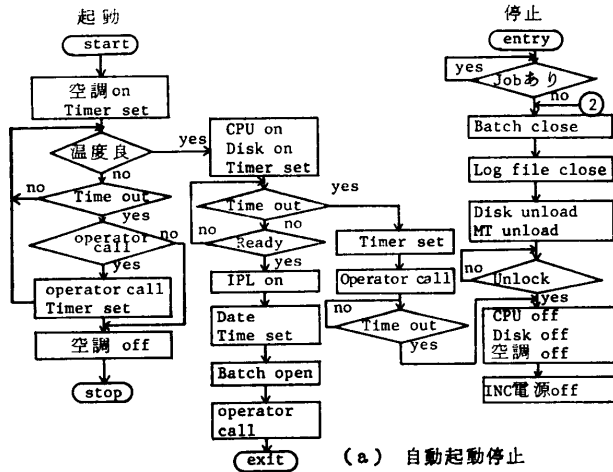
表示信号 ready, lock

MT: スイッチ load/unload

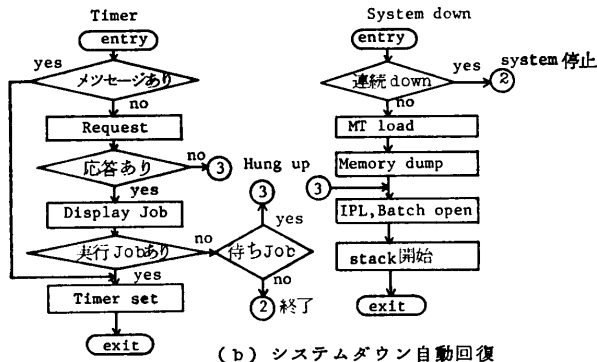
ECI (Environment Control Interface)

カレンダー付きデジタル時計とのインタフェース

日付け、分の変化は割込み要因



(a) 自動起動停止



(b) システムダウン自動回復

図5 自動運転の処理フロー (一部)

Fig. 5 Flow chart of self running operation (a part).

空調機 電源スイッチ

温度計とのインタフェース

オペレータコール用ブザー オン/オフ・スイッチ

自動運転システムのソフトウェアはすべて INC 上におかれる。各機能に対する処理手続きの流れ図の一部を図5に示す。

このシステムには種々の変形が考えられる。たとえばオープンバッチ処理や TSS のシステムでは、IPL後にオペレータコールを行わず、オペレータ代行ジョブを起動してすべてのシステムメッセージをそのジョブに送り、自分で運転管理を行うことができる。また4.3節で述べるような電話回線とのインタフェースを持てば、遠隔地から計算機を監視・制御するリモートコンソールの機能や、TSSで計算機を使用したいのにシステムに電源が入っていない時、利用者が電話をかけてシステムを起動して使用するリモートスタート/ストップ機能などを実現することができる。

4.3 ジョブ処理状況の表示・問合せシステム

バッチ処理やオープンバッチ処理の環境下、またはTSSの環境下でも長時間ジョブをバックグラウンドのバッチ処理に依頼した場合等では、依頼したジョブがいつ完了するかが依頼者の最大の関心事となる。このため、ジョブの完了表示、電話連絡、ジョブ処理状況の表示等が計算センタにあれば便利であろう。

ここではこのような目的で試作したシステムの機能と、インタフェース部のハードウェアについて述べる。

4.2.1 システムの構成と機能

図6にINCを応用したジョブ表示・問合せシステムの構成を示す。このシステムは以下の機能を持っている。

(1) ジョブ完了の表示および放送

バッチジョブ制御は各ジョブの終了時に完了メッ

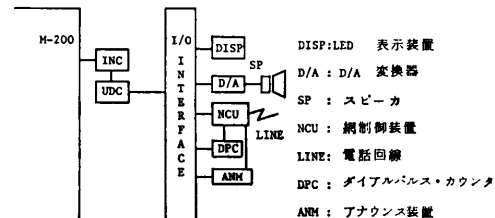


図6 ジョブ問合せ・応答システム

Fig. 6 Configuration of the inquiry and announcing system.

セージをオペレータコンソールに出力する。このジョブ番号は INC から UDC に送られ、LED 表示装置に表示されると同時に、録音編集方式で音声データに変換され、DA 変換器を通して音声として出力される。

### (2) ジョブ完了通知

ジョブ完了通知がほしいユーザは最終ジョブステップの後に完了通知ジョブステップ

// TEL tel. no.

を追加する。このジョブステップではパラメータとして与えられた電話番号を WTO (Write To Operator) で INC に送る。これは INC から UDC に渡され、UDC は NCU (網制御装置) の自動ダイヤル機能を用いてダイヤルする。ラインが通じると UDC はこれを検出し、ANM (アナウンス装置) を制御して「あなたのジョブが完了しました。」というメッセージを送る。なお一定時間呼び出してラインが通じない場合はいったん呼出しを中止し、後刻に再度通知を試みる。

### (3) 電話による問合せ

NCU の着信機能を用いて問合せに対する自動応答を行う。NCU に割り当てられた番号に電話をすると着信が NCU に検出され、UDC の制御のもとで ANM が応答する。ユーザは ANM の指示に従い問合せ種別をダイヤルで入力する。この情報は DPC (ダイヤルパルス・カウンタ) によって検出され UDC に入力される。

センタの利用案内、講習会予定等、計算機システムに関係しない一般の問合せに対しては、ANM にあらかじめ録音されているメッセージを UDC が返す。

ジョブの状況の問合せの場合は ANM の指示によりさらにジョブ番号をダイヤルする。UDC は INC 経由で DISPLAY コマンドを発行してジョブの状態を調べ、ANM を制御して適当な応答を返す。

#### 4.2.2 インタフェース部のハードウェア

DA 変換器、DPC、ANM 制御回路、NCU のレベル変換器等は I/O ボックスとして一つにまとめられ、UDC と 50 芯のケーブルで接続されている。各装置に対する信号は汎用 I/O ボードの 2 個の PIA (Peripheral Interface Adapter) の各ポートに割り当てられている。

音声出力部 (D/A) のブロック図を図 7 に示す。

音声データは、この実験では 5 kHz のサンプリングデータをそのまま使用した。「イチ」、「サンジョウ」

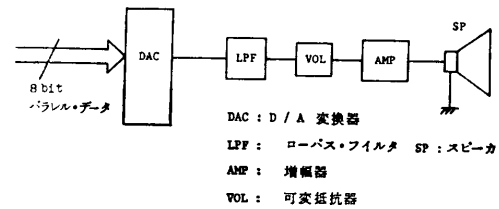


図 7 D/A 部のブロック図

Fig. 7 Block diagram of the D/A converter.

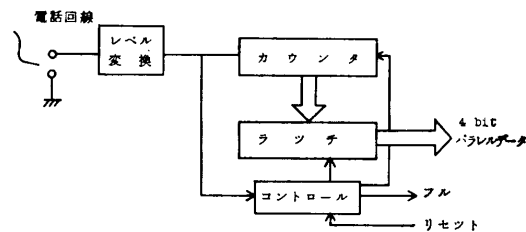


図 8 ダイヤルパルス・カウンタの構成

Fig. 8 Block diagram of the dial pulse counter.

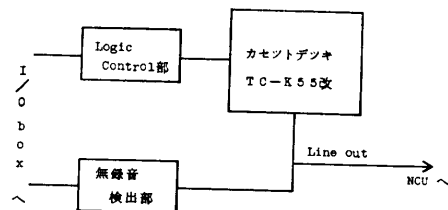


図 9 アナウンス装置の構成

Fig. 9 Configuration of the announcing machine.

のような音素ごとに A/D 変換された 8 ビット長の音声データは UDC のデジタルカセットに入れられており、これをメモリ中に編集した後 5 kHz (1 バイト/200  $\mu$ s) の速度で D/A に送りジョブ番号の発声を行った。

この部分は音声合成用 IC 等を利用すると非常に簡単になるであろう。

ダイヤルパルス・カウンタ (DPC) の構成を図 8 に示す。1 回のダイヤルが完了するとフル・フラグが立つので、そこでラッチの内容を読み込む。データが読み込まれるとフル・フラグはリセットされる。

網制御装置 (NCU) は通常、モデムと回線制御装置 (CCU) に接続されるが<sup>10)</sup>、我々の場合は UDC がモデムと CCU の両方の役割を果し、制御はすべてプログラムで行われる。自動発信、着信接続等の手順の詳細は省略する。

アナウンス装置 (ANM) には手近にあったカセットデッキ (SONY TC-K55) を一部改造して利用した。

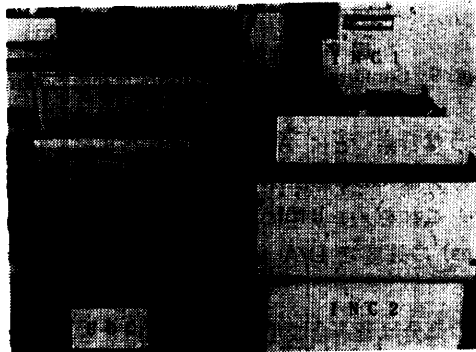
このデッキにはリモートコントロール・ボックスが付いており、これのスイッチをリレーに置き換えることによってロジックコントロールを可能にした。

ANM の構成を図 9 に示す。ANM には多くのメッセージが入っており、必要なメッセージを探し出す機能が必要になる。無録音検出部はこれを行うもので、メッセージの区切りの無録音部を検出する。メッセージは録音された位置（先頭からの順番）で識別される。

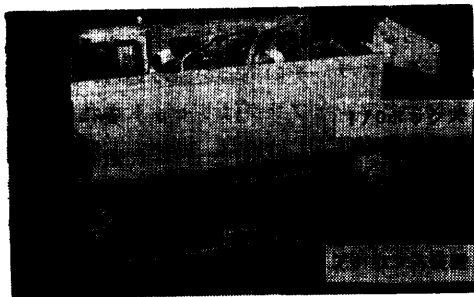
無録音部の検出はメッセージサーチのほかに、メッセージ送出の完了を検出するためにも必要である。メッセージサーチとメッセージ送出ではテープスピードが異なるため、無録音検出部は時定数の異なる再トリガ可能なワンショット・マルチ 2 個で構成されている。

なお、メッセージサーチのためには、高速走行中に再成を行う、いわゆるキュー機能が必要であるが、このデッキはこの機能がなかったので多少の改造でこれを可能にした。ANM には留守番電話等に使用される専用デッキを利用するとよいであろう。

この実験に使用した装置を図 10 に示す。



(a) INC 1, INC 2 and UDC



(b) I/O box and announcing machine

図 10 実験システムの各装置

Fig. 10 Hardware components of experiment system.

## 5. む す び

以上計算機システムの運用管理の概念と運用管理に対する INC の応用例について述べた。

自動運転システムについては対象が稼働中のシステムであることもあって実装実験は行っていない。しかし、メーカーの CE の協力程度でこのシステムの実装は可能であるし、このレベルで十分実用になると思われる。

ここで示した応用例はいずれも、いわゆる「小細工」である。自動運転システムについて見れば、防災システムと連動した本格的な無人運転システムがいくつも稼働している<sup>2)</sup>。しかしこのようなシステムを、たとえば大学の学科・研究室で持つ程度の計算機に合うようにレベルを落して適用することは、たぶん新しいシステムを作り上げるだけの労力が必要であろうし、このようなシステムに電話による応答機能を追加することも困難であろう。

INC の最大の 特徴はシステム構成の柔軟性、システム変更の容易性にある。たとえばここで示した自動運転システムにおいて、4.2 節に述べたようにオペレータの判断をホスト計算機上のジョブレベルのプログラムとして構成すれば、判断基準・行動様式の変更や高度化はきわめて容易になる。オペレータの機能が、「あらかじめ想定された事態」に対して判断、行動するものとすれば、INC を用いたシステムでオペレータの機能はすべて代行でき、適当なハードウェア、必要があれば工業用ロボット等の導入によって完全な無人システムにまで拡張することも可能である。

我々は個々の小細工ではなく、小細工を可能にした「INC の機能」に注目したい。INC はハードウェア、ソフトウェアの両方にわたる、OS の汎用ユーザインタフェースであり、この先に何を実装するかは使用者にまかされている。

オペレータコンソールは OS が動作する上で欠くことのできない重要な機能を持っている。INC はコンソールインタフェースを利用するのであるから、自動運転はもちろん、そのほかの応用においてもハードウェア、ソフトウェアの信頼性の確保、OS の保護には十分に考慮を払わなければならない。この意味で INC を使用者が設置するのは危険が多すぎる。このような機能が多くの OS の標準機能となることが望まれる。

なお、コンソールメッセージを OS 内で受け取る機能を持った OS も開発されている<sup>11)</sup>。このシステムは

自動運転のみを目的としているためこの機能をユーザインタフェースとして解放していない。この機能を拡張して使用者が利用できるようなれば、OSのソフトウェア・ユーザインタフェースとしてINCと同等になり、柔軟な運用管理システムを構成することが可能となろう。

**謝辞** この実験システムのハードウェアは主として北尾正典君（現TDA）、INCの基本ソフトウェアは主として堂込一秀君（現富士通）が担当し、実験を行うに当たっては九州大学大型計算機センター、富士通株式会社の関係各位に多大のご配慮をいただいた。また本論文をまとめるに当たり、九州大学工学部牛島和夫教授には貴重なご示唆をいただき、東京工業大学木村泉教授、九州大学工学部宇津宮孝一助教授には多くのご助言、ご鞭撻をいただいた。ここに記して謝意を表す。

### 参 考 文 献

- 1) 高田 勝他：共同利用大型計算機センターの設計とその動作解析，情報処理学会第9回プログラミングシンポジウム報告集（1968）。
- 2) 市村 洋他：汎用コンピュータ・システムの自動運転，情報処理，Vol. 22, No. 4(1981)。
- 3) 宇津宮孝一他：ジョブ工程管理，九州大学大型計算機センター広報，Vol. 5, No. 2 (1972)。
- 4) 有田五次郎他：計算機システム障害警報装置の開発，九州大学大型計算機センター広報，Vol. 5, No. 3 (1972)。
- 5) 北川 一他：京大大型計算機システム(I)～(VII)，情報処理学会第15回全国大会予稿集(1974)。
- 6) 伊澤喜三男他：計算機センター無人化の試みと問題点，情報処理，Vol. 18, No. 11 (1977)。
- 7) 猪瀬 博他：東京大学教育用計算機センター省力化システム，三菱電機技報，Vol. 51, No. 7, (1977)。
- 8) 有田五次郎：インテリジェント・コンソール，情報処理(論)，Vol. 23, No. 3 (1982)。
- 9) F1025A/B 入出力制御装置動作説明書，富士通株式会社。
- 10) FX-1A型，FX-10A型 NCU 網制御装置動作説明書，富士通株式会社。
- 11) 藤本義起他：無人化を指向した自動運転システム，情報処理学会第21回全国大会予稿集(1980)。  
(昭和56年12月21日受付)  
(昭和57年2月16日採録)