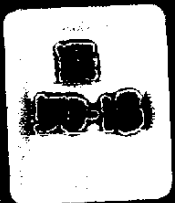


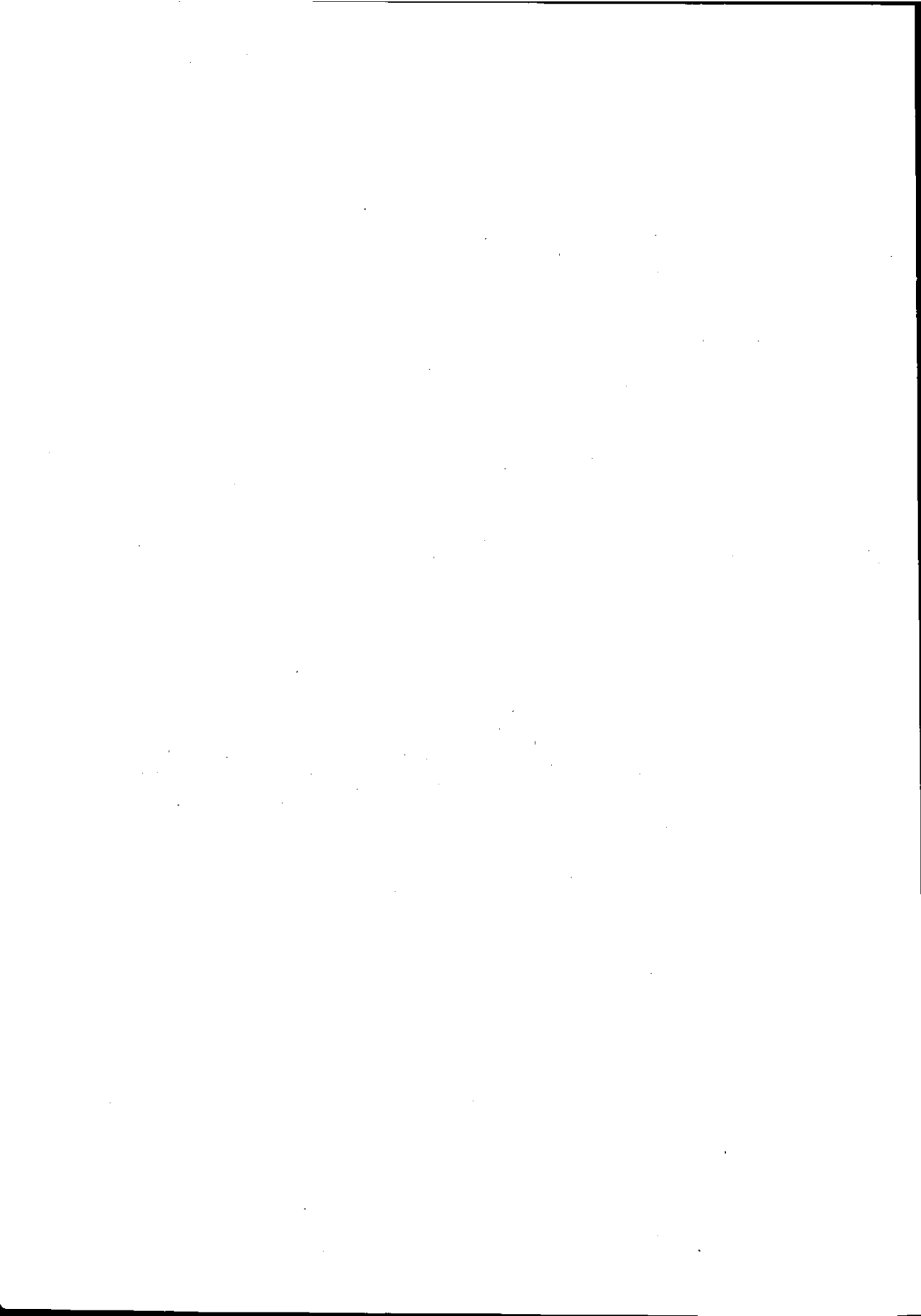
# マイクロ・プロセサー

## 御制への応用編

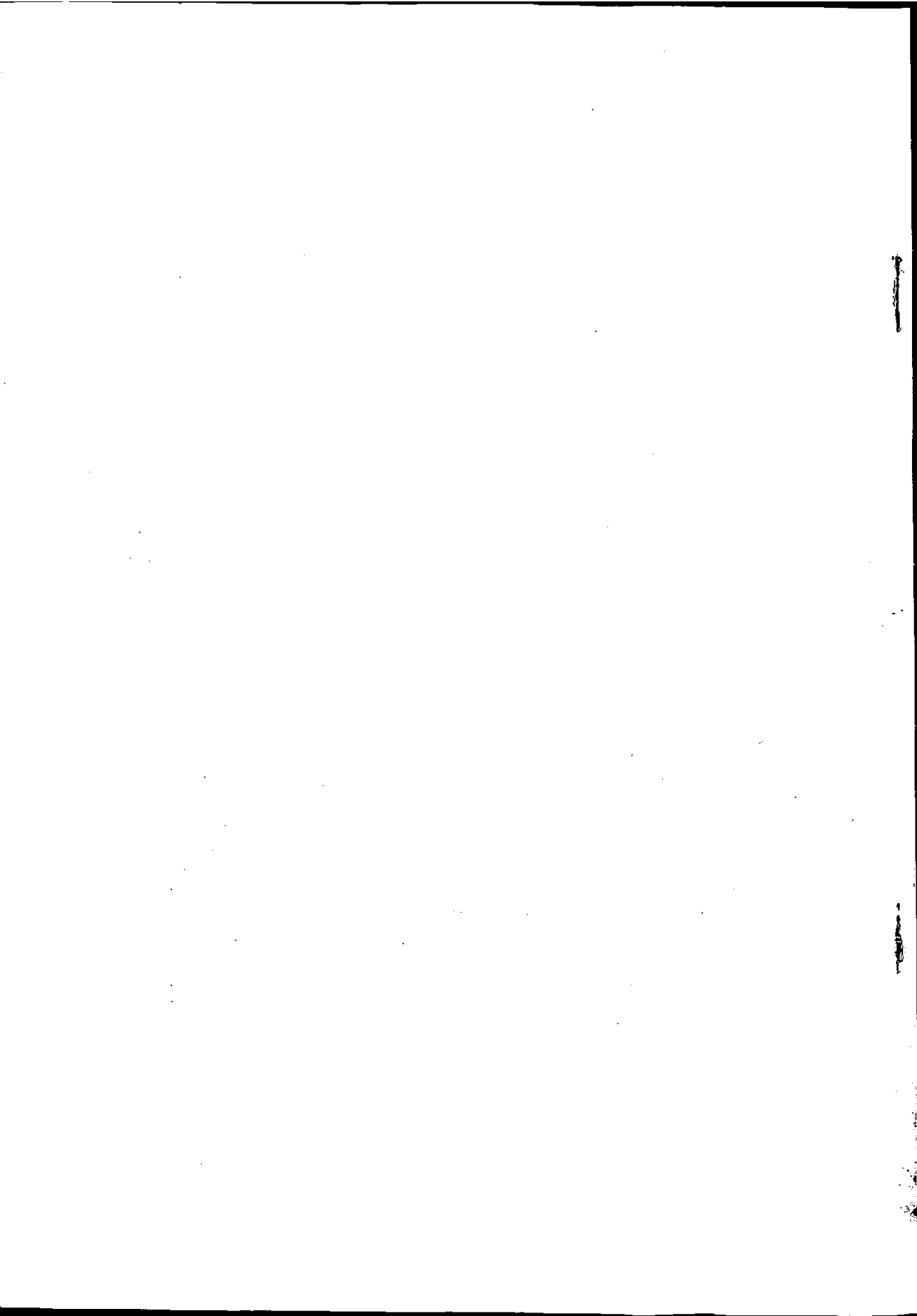
昭和50年度

財団法人 日本情報開発協会





この資料は、日本小型自動車振興会から小型  
自動車競走法に基づく小型自動車等、機械工業  
振興資金の交付を受けて作成した資料の頒布用  
増刷である。



## は じ め に

### < 刊行の目的 >

半導体 L S I を素子とするマイクロプロセッサは、超小型・軽量・低消費電力・低価格を特徴とするコンピュータ・システムとして出現以来・L S I 技術のシーズとしての発展とともに、国民生活および社会ニーズに対応する応用範囲も今後飛躍的に拡大されることが予想されています。

しかし乍ら、重要な応用分野である「制御」については、発想・機能・技術等従来と異った開発を考えて行かなければならず、全くオリジナルな見地からその開発に携わる技術者を主対象とし、管理者および経営者の立場にある方にも指針としていただくことを目的とし、昭和 50 年度日本小型自動車振興協会の補助金の交付を受けて編集したものであります。

< 編集委員 >

通商産業省工業技術院

電子技術総合研究所

制御部論理システム研究室長      工学博士      森      亮      一

京都大学教授

工学博士      西      川      稔      一

株式会社明電舎

重電事業本部

コンピュータシステム部部长      伊      東      昭      宏

三菱電機株式会社

技術管理部部長代理      千      葉      正

富士電気製造株式会社

情報処理推進本部

営業助成部次長

兼 第2開発部主査      林      部      秀      治

富士電機製造株式会社

情報処理推進本部

第1開発部主務      今      泉      吉      弘

株式会社応用システム研究所

副主任・研究員

山      根      義      彦

## ＜ 利 用 方 法 ＞

コンピュータでは、普通ソフトとハードの分野に分けますが、この講座ではハードの又外側にあるもっとハードなもの、すなわち、機械、検出器、制御器等に直接にかかわる分野でのマイクロプロセッサの効用という、最も重要であるが書きにくい分野を中心としてまとめてあります。

この講座の1つの特徴は、特定メーカーの製品にかかわらない形をとった事であります。むしろ、どのメーカーにも共通する基本的な知識や技術について、十分な理解が得られる様配慮してあります。

現在のマイクロコンピュータ技術は日進月歩で進歩しており、同じメーカーのマイクロコンピュータでも半年前と今とではインターフェイスチップのLSI化等によって著しく異なることも起こります。その場合、技術の進歩の方向を見失わないで、いつも前向きの姿勢で、対処して行くために、この講座を利用していただくことが、非常に有益であると同時に、個々のLSI素子や周辺装置等については、いつも停止することなく最新の技術を吸収していく努力を期待しております。

## ＜ 講 座 の 内 容 ＞

講座は5章から構成されており、第1章では、制御技術の基本的体系について述べさらにマイクロコンピュータとの関連をまとめてあります。第2章では、マイクロコンピュータのシステムの導入について各種の手法をあげて比較し、その過程で陥り易い問題点を具体的に解説してあります。第3章では、コンピュータの市場性、経済性、特に信頼性について、出所の明らかな多数の権威あるデータをもとに具体的かつ実用的にまとめてあります。第4章では実例と図解により実際の応用を考える方々の役に立つ多くのシステムの具体例をとりまとめてあります。第5章では大規模制御系の設計、システムの発想、階層構成など、および工業用のロボット、センサなどについて、将来におけるあるべき姿をまとめています。

本書を利用されるに当っては、常に技術の進歩の方向を見失うことなく同時に、読者が実際に接するマイクロコンピュータについては、それぞれのマ

ニユアルについて、一步一步何度も読み返し、1つの回路、1つの動作を理解するだけでも徹底した勉強をされる事を望みます。このような狭くとも深い勉強と、本書の意図する広く包括的な勉強とが補い合う事が最も重要なこととであります。

#### < お わ り に >

今回の執筆者の皆さんは、学界で、また各社の現場の第一線で活躍しておられる方々であり、多忙な時間をさいて、尽力をいただいたことに対し、深い敬意と感謝の意を表する次第であります。

なお、お気付きの点があれば、大小にかかわらず事務局までご連絡くださるようお願いいたします。



# 目 次

## 第 1 章 制御技術と制御機能の分類

|                               |    |
|-------------------------------|----|
| 1.1. はじめに .....               | 1  |
| 1.2. 制御工学とその発達 .....          | 2  |
| 1.2.1. 制御の概念 .....            | 2  |
| 1.2.2. 単ループのフィードバック制御系 .....  | 4  |
| 1.2.3. 制御理論と制御技術 .....        | 6  |
| 1.2.4. システムの概念と計算機ヒエラルキ ..... | 18 |
| 1.3. 制御系におけるセンサと情報処理 .....    | 19 |
| 1.3.1. センサとトランスデューサ .....     | 19 |
| 1.3.2. アナログからデジタルへ .....      | 23 |
| 1.3.3. デジタル計測量の処理 .....       | 25 |
| 1.3.4. デジタル・データ伝送 .....       | 30 |
| 1.4. 物体の動きの制御 .....           | 31 |
| 1.4.1. 数値制御 .....             | 31 |
| 1.4.2. ロボット .....             | 35 |
| 1.5. 流量、圧力の制御 .....           | 38 |
| 1.5.1. 流量の制御 .....            | 38 |
| 1.5.2. 圧力の制御 .....            | 43 |
| 1.5.3. 液位の制御 .....            | 43 |
| 1.6. 温度プロセスの制御 .....          | 44 |
| 1.6.1. 伝熱の法則 .....            | 44 |
| 1.6.2. 温度の計測器 .....           | 44 |

|                              |    |
|------------------------------|----|
| 1.6.3. 熱交換器の制御 .....         | 45 |
| 1.7. 電力系統における制御 .....        | 49 |
| 1.7.1. 火力発電所における制御 .....     | 49 |
| 1.7.2. 系統における制御 .....        | 50 |
| 1.8. シーケンス制御 .....           | 52 |
| 1.9. マンマシン・インターフェース .....    | 56 |
| 1.9.1. インテリジェント端子 .....      | 56 |
| 1.9.2. インテリジェント・ディスプレイ ..... | 57 |
| 1.10. おわりに .....             | 58 |

## 第2章 制御技術への新しい発想

|                                      |    |
|--------------------------------------|----|
| 2.1. はじめに .....                      | 61 |
| 2.1.1. マイクロコンピュータのインパクト .....        | 61 |
| 2.1.2. マイクロコンピュータの歴史的背景 .....        | 66 |
| 2.2. 制御機能の機能モジュール分割と再構成 .....        | 68 |
| 2.2.1. 制御機能モジュールの分割 .....            | 68 |
| 2.2.2. 制御機能モジュールの分割の整理と再構成の考え方 ..... | 72 |
| 2.2.3. システムとサブシステムの関係 .....          | 80 |
| 2.3. ソフトウェア技術 .....                  | 83 |
| 2.3.1. マイクロコンピュータ、ソフトウェアの特長 .....    | 83 |
| 2.3.2. ソフトウェア製作上の問題点 .....           | 86 |
| 2.4. ソフト・ハードの一体化 .....               | 90 |

|                             |     |
|-----------------------------|-----|
| 2.4.1. マイクロコンピュータシステム製作上の特長 | 90  |
| 2.4.2. ソフト・ハード一本化のメリット      | 95  |
| 2.5. 制御技術者の発想の転換            | 95  |
| 2.5.1. 従来の制御装置の限界           | 95  |
| 2.5.2. マイクロコンピュータ制御システムの狙い  | 99  |
| 2.5.3. 将来の制御装置のイメージ         | 101 |
| 2.6. おわりに                   | 102 |

### 第3章 制御における市場性、経済性、信頼性などの諸問題

|                                     |     |
|-------------------------------------|-----|
| 3.1. はじめに                           | 104 |
| 3.2. マーケット調査の概況                     | 104 |
| 3.2.1. 生い立ちと急速な成長                   | 104 |
| 3.2.2. 制御への応用の市場                    | 107 |
| 3.2.3. 我国の状況                        | 111 |
| 3.3. 従来の制御との比較                      | 111 |
| 3.3.1. プラントの運転、制御と情報の取り扱い           | 111 |
| 3.3.2. コンピュータ制御とマイクロプロセッサ制御         | 116 |
| 3.3.3. ワイヤドロジックとの比較                 | 120 |
| 3.4. マイクロプロセッサの特質と制御における応用上の<br>諸問題 | 124 |
| 3.4.1. マイクロプロセッサの長所と短長              | 124 |
| 3.4.2. マイクロプロセッサ応用制御の経済性            | 125 |
| 3.4.3. 信頼性と検査の問題                    | 133 |
| 3.4.4. メーカーとユーザの分担のあり方              | 147 |

|                                            |     |
|--------------------------------------------|-----|
| 3.4.5. 制御用を目指した最近の新製品 .....                | 157 |
| 3.5. マイクロプロセッサに最も適した用途 .....               | 162 |
| 3.5.1. 応用分野と使い方 .....                      | 162 |
| 3.5.2. データ処理への応用 .....                     | 168 |
| 3.5.3. 計測分析システムへの応用 .....                  | 172 |
| 3.5.4. コントロール(プロセス、マシン、システム)<br>への応用 ..... | 173 |
| 3.6. おわりに .....                            | 177 |

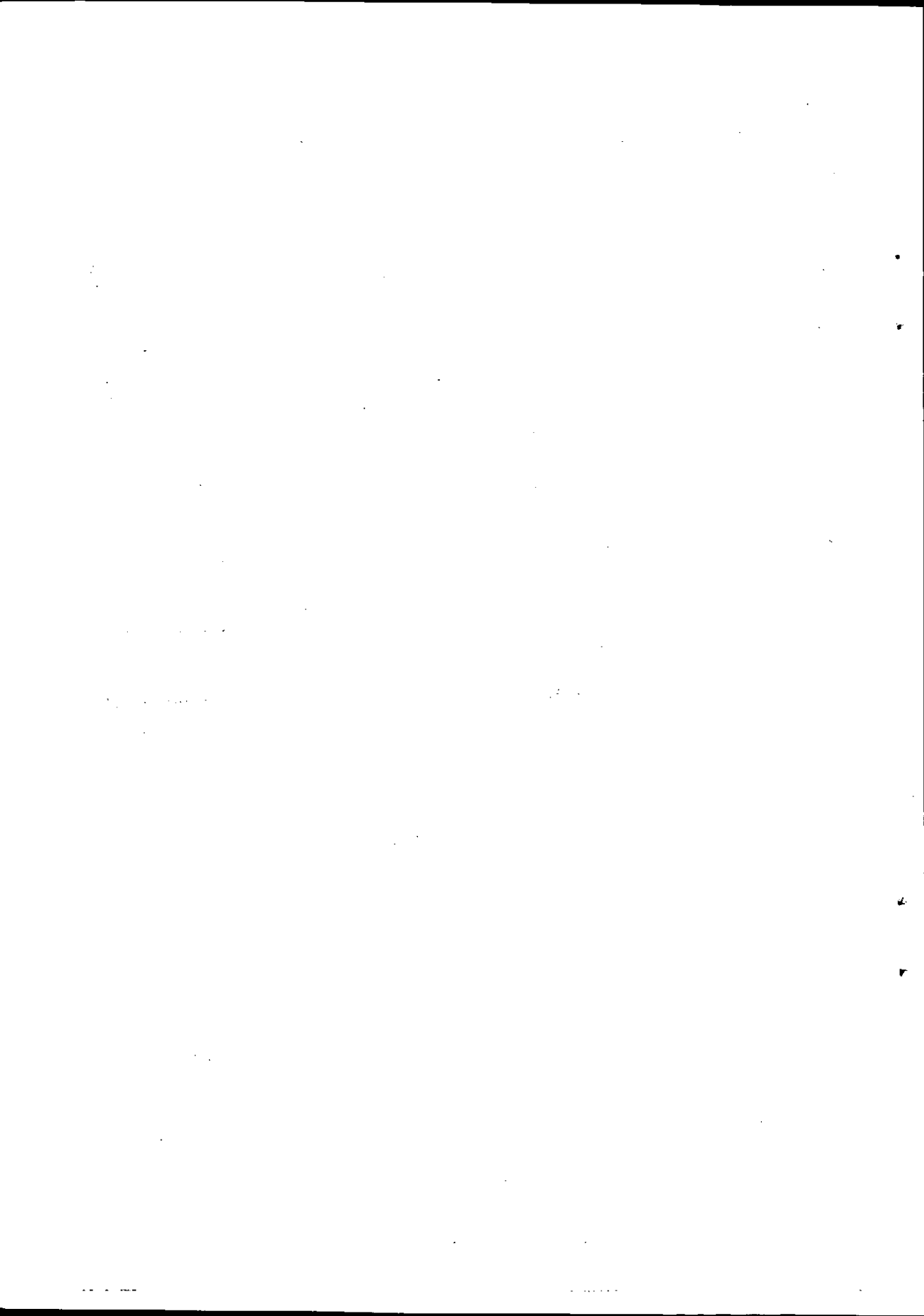
## 第4章 制御システムの具体例とその展開手法

|                                  |     |
|----------------------------------|-----|
| 4.1. 概 要 .....                   | 178 |
| 4.2. 制御システムの構成 .....             | 179 |
| 4.2.1. マイクロコンピュータ制御システムの構成 ..... | 179 |
| 4.2.2. 制御システムの具体化 .....          | 187 |
| 4.3. 制御システムにおけるデータ伝送 .....       | 201 |
| 4.3.1. 伝送の形態 .....               | 201 |
| 4.3.2. ハイアラキシステム .....           | 204 |
| 4.3.3. 制御システム間の伝送 .....          | 206 |
| 4.4. 制御機能と制御要素 .....             | 210 |
| 4.4.1. 制御システムの仕組み .....          | 210 |
| 4.4.2. 制御機能と制御要素 .....           | 210 |
| 4.5. 制御システムの具体例 .....            | 215 |
| 4.5.1. 制御システムの分類 .....           | 215 |

|                                   |     |
|-----------------------------------|-----|
| 4.5.2. 制御システム的具体例 .....           | 219 |
| 4.6. 制御システムの展開手法 .....            | 228 |
| 4.6.1. 展開手法 .....                 | 228 |
| 4.6.2. サポートシステム .....             | 233 |
| 4.6.3. POL及びFIF .....             | 234 |
| 4.6.4. マイクロプロセッサ応用の可能性と適用限界 ..... | 243 |

## 第5章 制御機能の将来動向

|                               |     |
|-------------------------------|-----|
| 5.1. はじめに .....               |     |
| 5.2. システム化の意味とマイクロプロセッサ ..... | 245 |
| 5.3. システムの分割と複合制御系 .....      | 248 |
| 5.4. ロボット-人間的機能の機械化 .....     | 250 |
| 5.5. マン・マシン・オペレーション .....     | 253 |
| 5.6. おわりに .....               | 256 |



## 第1章 制御技術と制御機能の分類

### 1.1 はじめに

1963年、アメリカのDEC社がPDP-5を発表したとき、制御技術には1つの変革が起こった。すなわちミニコンピュータ時代の幕明けである。当時、デジタル計算機の世界ではIBM社が圧倒的な力をもっていたが、その指向するところは、汎用大形、バッチ処理の計算機であり、科学技術用というよりもむしろビジネス情報処理用であった。そこへDEC社が全く正反対の指向を持って、専用小形、オンライン処理を目的とした計算機を登場させたのである。低価格な計算機を現場に配置できるということは、制御情報処理技術に大きなインパクトを与えた。それまでトランジスタ、IC、リレーなどをリード線でつなぐ、いわゆる布線論理（ハードワイヤード・ロジック）の設計を余儀なくされ、多様化し複雑化する設計要求に苦しんでいた技術者に、ハード的手段からソフト的手段へ転換する極めて重宝な道具が提供されたのである。

アメリカにおいてミニコンピュータは、たちまちにして普及し、1970年代の初めには設置台数にして50%近い割合に達し、計算機市場で確固たる地位を占めるに至るのである。各種のデータ処理、集録、データ伝送の端末、計測情報処理、プロセス制御、工作機械の数値制御、シーケンサ、交通制御等々、広い分野で成功を収めたミニコンピュータも、また同時にオーヴァースペックに悩まされることが多かった。すなわち端末における専用の情報処理装置としては、機能過剰、逆にいえば経済性が悪いと結論される場合が数多く出てきたのである。例えばコンピュータNCやプログラマブル・ロジック・コントローラ（PLC）などでは、かなり特殊なものにしかミニコンピュータは適用できない。

そこへ1971年、Intel社によるMCS-4（4004）ワンチップ・プロセッサの発表である。より限定された機能ではあるが、より小形、より小形、より低価格、これこそはそれまでの悩みを解決する朗報として、世に迎えられたのである。それ以後今日まで、チップそのものの改良から応用に至るまで、ある意味では着実な歩みを示してきてはいるが、当初

の予想ほど伸びていないという見方もある。周辺器具の価格やソフトの整備が障害になっているわけであるが、この辺で応用の重要な一分野である制御、制御における情報処理の技術を振り返ってみて、その過程でマイクロプロセッサの現在並びに将来占めるべき位置を考えてみようというのが、本章の試みである。

理論、技術、計算機応用の3つの柱を中心として発展しながら、今日の制御工学技術は極めて広い範囲に及んでいる。それらのすべてをカバーし、一定の筋道をつけて整理することは至難の業であつて、著者の力の及ぶところではないが、制御技術の時間的経緯と空間的広がりについて、何らかの概念をつかんで頂く手がかりとなれば幸いである。

## 1.2 制御工学とその発達

### 1.2.1. 制御の概念

あるシステムがあり、われわれはそれを何らかの手段で操作することができるでしょう。そのときこのシステムは、それが意識されているといないと拘らず、1個の制御系である。システムを操作しようとするとき、われわれは必ず操作の目標、目的、あるいは操作の結果に対する評価の基準を持っている。そういうもの無しに操作を行なうことは考えられないからである。一方、操作に際しては通常多くの制限が課せられている。直接操作し得る量の種類及びその操作範囲は限られている。また、操作の結果進行するシステム状態の変化は、一定の物理化学的（ないしは社会的）法則に支配されている。

従つて制御とは、限られた操作量を用いて一定の法則に従いながらシステム状態を合目的に変化させること、と言うことができよう。（変化の中には一定状態に保つということも含まれている。）例えば、炉の温度変化を制御しようとするとき、炉内の温度そのものを操作することはできないので、バーナの燃料供給量を弁の一定範囲内の開閉によつて操作することになる。燃料の量と炉の温度変化の関係はある物理的法則が支配するわけである。

制御技術の歴史はかなり古くまで遡ることができるが、独自の工学分野



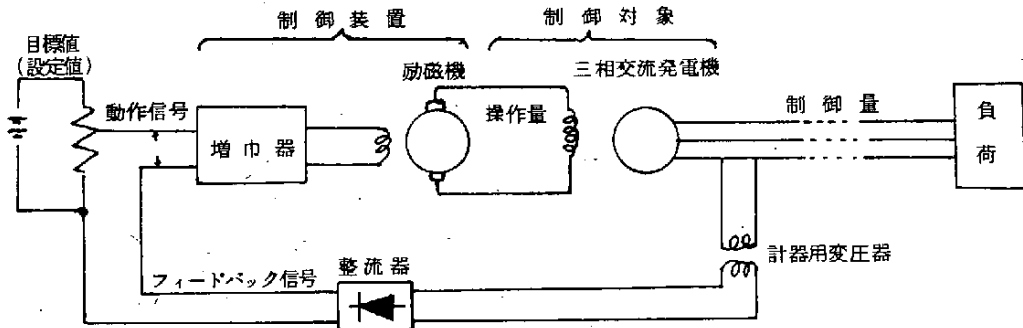
として明確に意識され、理論、技術、応用の各面で急速な発展を見せるようになったのは、第2次大戦中、1940年代に入ってからであろう。すなわち、制御の根本原理としてフィードバック制御の概念が確立され、物体の位置、方位、姿勢などを制御量として、時間的に変化する目標値に追従するように制御するサーボ系（またはサーボ機構）の理論と技術が長足の進歩をとげた。また、工業プロセスにおいて、装置内の温度、圧力等の環境条件と原料やエネルギーの供給量、製品の取出し量を調節して、製品の品質や生産量を管理しようとするプロセス制御にも、フィードバックの手法が組織的に導入された。

フィードバックの手法は、1778年に、J. Watt がエンジンの調速のために取付けた遠心力ガバナにおいて、初めて用いられたと言われている。その後、船舶や飛行機の自動操舵、液面や温度の調節、工作機械のならい削り等、工業の各種分野で利用されてはいたが、理論及び技術として体系化されたのは、先にも述べたように第2次大戦中、主としてアメリカにおいてであった。我国においても戦後いち早く研究が開始され、工業生産の拡大、高度化の要求と各種機器の製作技術、特に電子技術の発達とが相まって、急速な発展を遂げた。時代の発展と共に、大勢においては1ループの制御（スカラー制御）から多ループの制御（ベクトル制御）へ、アナログ制御からデジタル制御へと推移し、現在では、多くの制御装置の核心を成すものはさまざまな規模のデジタル計算機であると言えようが、手法は依然としてフィードバック制御が中心となっている。

フィードバックの他に見落としてはならないものに、シーケンス制御がある。これはあらかじめ定められた順序に従って多段階の制御作業を逐次進めてゆく制御であつて、フィードバック制御とは本質的に異なる。工業プロセスや工作機械の作業順序の指定に使われ、各段階の制御にはフィードバックが用いられることが多いので、その意味で上位の制御と考えることができよう。シーケンス制御装置は、もともと情報の論理処理を中心とした装置であるが、ここにも小形ないしは超小形の計算機の進出が目立つようになってきている。

### 1.2.2. 単一ループのフィードバック制御系

第1-1図の三相交流発電機の自動電圧制御系は、典型的な1ループ・フィードバック制御系の例である。この系で制御対象は三相交流発電機（界磁巻線を含む）、制御量は、その出力電圧である。



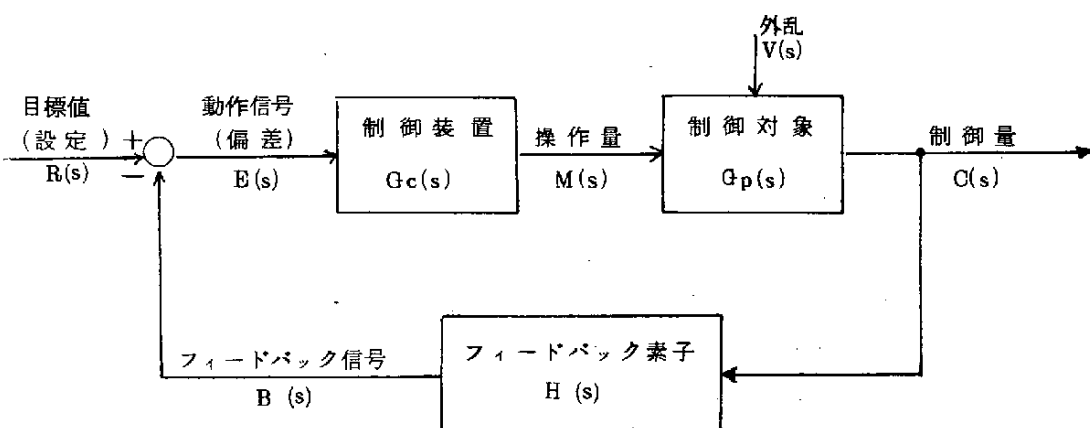
第1-1図 三相交流発電機の自動電圧制御系

制御量の値はフィードバック要素を通してフィードバック信号となり、偏差検出装置によつて目標値（一定の場合は設定値）を与える基準入力と比較される。偏差は動作信号となり、動作信号処理装置（最も普通のP動作なら増巾器）及び励磁機から成る制御装置（プロセス制御では調節計と言う）に加えられる。制御装置からの出力は操作量であつて、制御対象に入力され、閉じたループが完結するわけである。

以上の制御系の信号伝達の流れをブロック線図で表わしたものが第1-2図であり、これは1ループ・フィードバック制御系の一般的な構成を示している。ループのいろいろの点に外乱が加わる可能性があるが、第1-2図では操作量に加わる外乱で代表した。このようなフィードバック系が有効なのは、

（1）いかなる外乱に対しても、その影響を軽減することができる。

（2）制御装置や制御対象のパラメータ変動に対する出力変動の感度を下げることができるからである。外乱やパラメータ変動を個々に検出し



第 1-2 図 フィードバック制御系のブロック線図

て補償することは極めて困難であることを考えれば、それらの影響をまとめて軽減することがフィードバックが、如何に有力なものかがわかるであろう。

制御系の設計にあたっては、当然、追従性（または速応性）を良くすることが要求される。すなわち目標値に変化が生じたとき、出力がなるべく速くこれに追従することが必要である。そのためには例えば増巾のゲインを上げればよい。しかし、ほとんどの場合制御対象（或いは制御装置の内部にも）の動特性には おくれやむだ時間 が含まれているため、ゲインを上げ過ぎると系は不安定になり、ハンチング現象を生じたりする。従つて 安定性 を保つことがもう 1 つの重要な要求となる。この追従性と安定性の両方を満足させることが、フィードバック制御系の設計では最も基本的なことなのである。

制御の動的過程における制御量のふるまいの他、入力エネルギーの消費量などを含めて、制御の評価基準が数式的にコスト関数として明確に定義できる場合は、コスト関数を最小にするような制御、すなわち 最適制御 を実現することが追求される。制御量或いは内部状態量を用いて 最適フィードバック制御 が実現できれば理想的であるが、正確な最適制御を求めるには動作信号の処理が複雑になり過ぎることも多い。そのときは準最適制御

が追求されるが、いずれにしても動作信号の処理は単なる増巾器などでは不可能なので、計算機による計算処理が有力になる。

### 1.2.3. 制御理論と制御技術

制御理論と制御について限られた紙数内にまとめることは不可能であるが、ここでは後の説明に必要な最小限の事柄を簡単に述べておく。安定性の理論その他省略した重要事項については、専門書を参照して頂きたい。

#### (a) 制御対象の特性

液体を出し入れするタンクを考えてみよう。これはプロセス化学工業でよく見られる装置である。タンクの液位を  $h$  [m]、断面積を  $A$  [m<sup>2</sup>] とし、タンクへ  $P$  [m<sup>3</sup>/sec] の流入量があるとする。一方、流体低抗  $R$  [sec/m<sup>2</sup>] の弁を通して液を抜き出すとすれば、流出量は  $p = h/R$  [m<sup>3</sup>/sec] である。液位の時間変化率は  $A dh/dt = p - q$  で与えられるから、 $RA = T$  として

$$T \frac{dh}{dt} + h = R p \quad (1)$$

なる微分方程式が得られる。ラプラス変換すれば

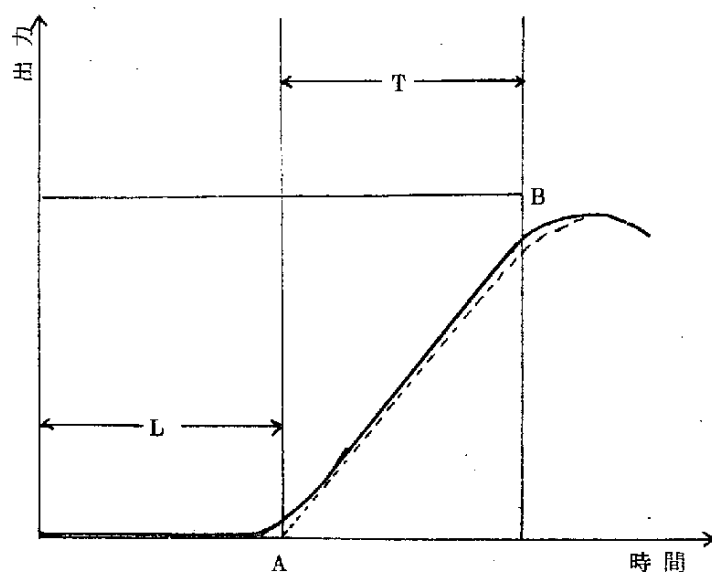
$$H(s) = \frac{R}{1 + Ts} p(s) \quad (2)$$

となる。

$p$  を入力、 $h$  を出力とみると、入出力関係が (1) 或いは (2) 式で表わされることになる。このような要素を 1次おくれ要素 と言い、いろいろな制御系に現われる基本的な要素である。 $T$  はその時定数、 $R$  はゲイン、(2) 式の  $1/(1 + Ts)$  は伝達関数である。2 個の 1 次おくれ要素を縦続に接続すれば 2 次おくれ要素となり、 $n$  個接続すれば  $n$  次おくれ要素となる。

ある要素に入力が増えられてから時間  $L$  の間は何ら出力に応答が現われないとき、その要素はむだ時間  $L$  を含む。むだ時間の伝達関数は  $e^{-Ls}$  である。時間領域で入出力特性を表わすのに、(ユニット) ステップ応答 がよく用いられる。プロセスのステップ応答は通常第 1 - 3 図のようないわゆる S 字特性を示すので、これを 1 次おくれとむだ時間、或いは 2 次おくれとむだ時間で近似することが多い。

1 次おくれとむだ時間とするとき、最急傾斜の切線 AB を引き、第 1



第 1 - 3 図 プロセスの単位ステップ応答

図 1-3 のように  $L$ 、 $T$  を定める。一般に制御対象のおくれが大きいほど制御しやすく、またむだ時間が大きいほど制御しにくい。

伝達関数を  $G(s)$  とするとき、 $G(s)$  の逆ラプラス変換が (ユニット) インパルス応答 であり、また  $G(s)/s$  の逆ラプラス変換がステップ応答である。さらに、 $G(s)$  において  $s$  を  $j\omega$  で置き換えた  $G(j\omega)$  は周波数伝達関数であり、要素に複素正弦波入力  $e^{j\omega t}$  を加えたときの出力は、 $G(j\omega)e^{j\omega t}$  で与えられる。

#### (b) 制御動作

制御動作を決定するのは 1.2.2 節で述べた制御装置 (或いは調節計) である。そして制御偏差或いは動作信号をどのように処理して操作量を決定するかにより、制御の良し悪しが決まる。一般には比例 (P) 動作、積分 (I) 動作、微分 (D) 動作、及びオン・オフ動作を単独に、または組み合わせて用いる。

偏差を  $e(t)$ 、操作量を  $m(t)$  としよう。上記の各動作における  $e(t)$  と  $m(t)$  との関係は次式で与えられる。

$$\text{P動作:} \quad m = K e \quad (K: \text{ゲイン}) \quad (3)$$

$$\text{I動作:} \quad m = \frac{1}{T_I} \int e dt \quad (T_I: \text{積分時間}) \quad (4)$$

$$\text{D動作:} \quad m = T_d \frac{de}{dt} \quad (T_d: \text{微分時間}) \quad (5)$$

$$\text{オン・オフ動作} \quad m = \begin{cases} M & e > 0 \\ N & e < 0 \end{cases} \quad (6)$$

P動作でKが大きいとゲイン余有が減つて系の安定性が悪くなり、Kが小さいと追従性が悪くなる。I動作の重要な役割は、オフセット（定常偏差）を打ち消すことである。G(s)の分母に $S^n$ が含まれるとき、それをn形の系と言うが、フィードバックをかけたとき0形ならばオフセットが生じる。I動作によつて1形（必要ならばI動作を複数段用いて2形以上）にすればオフセットはなくなるわけである。D動作は $e(t)$ の変化の傾向によつて操作量を決めようというもので、いわば先取り形の決定をする。オン・オフ動作はリレーによつて実現でき、その点簡単である。実際によく用いられるものは、P、PI、PID、及びオン・オフ動作である。

プロセス制御におけるPID調節計の最適パラメータ調整（チューニング）条件としては、（1）減衰比25%、（2）制御面積最小、（3）行過ぎ無し制御面積最小、（4）行過ぎ時間最小、などが採用される。Ziegler-Nicholsの条件が今日でもよく用いられるが、これは過渡応答の減衰比を25%ぐらいにするのが適当であるとの考えに基づいている。そしてプロセスの特性を第1-3図のように1次おくれとむだ時間で近似し、調節計をP動作だけにした場合の系の安定限界における比例ゲイン $K_o$ と、そのときの自励振動の周期 $T_o$ を求め、第1-1表に従つてパラメータを定める（限界感度法）。ただし調節計の伝達関数を

$$G_c(s) = K \left( 1 + \frac{1}{T_{Is}} + T_d s \right) \quad (7)$$

とする。またプロセスの時定数 $T$ とむだ時間 $L$ を基準にしてパラメータ値を決めると、第1-2表のようになる。

|       | K                  | T <sub>I</sub>    | T <sub>D</sub>      |
|-------|--------------------|-------------------|---------------------|
| P 動作  | 0.5 K <sub>o</sub> | —                 | —                   |
| PI 動作 | 0.45K <sub>o</sub> | 0.83 o            | —                   |
| PID動作 | 0.6 K <sub>o</sub> | 0.5T <sub>o</sub> | 0.125T <sub>o</sub> |

ジューラ・ニコルスのチューニング条件(1)

第 1 - 1 表

|       | K                 | T <sub>I</sub> | T <sub>D</sub> |
|-------|-------------------|----------------|----------------|
| P 動作  | $\frac{T}{L}$     | —              | —              |
| PI 動作 | $0.9 \frac{T}{L}$ | 3.3 L          | —              |
| PID動作 | $1.2 \frac{T}{L}$ | 2 L            | 0.5 L          |

ジューラ・ニコルスのチューニング条件(2)

第 1 - 2 表

### (c) サンプル値制御

デジタル制御装置では、連続信号を周期的にサンプルしたパルス状の信号を扱う。このような制御系をサンプル値制御系と呼んでいる。基本的な考え方は連続系と変わらないが、いくつかの相違点も存在する。

第 1 - 4 図はサンプル値制御系の基本的構成を示す。サンプラ S<sub>1</sub> のサンプリング周期を  $\tau$  とすれば、その出力  $es(t)$  は

$$es(t) = \sum_{n=-\infty}^{\infty} e(n\tau) \delta(t - n\tau) \quad (8)$$

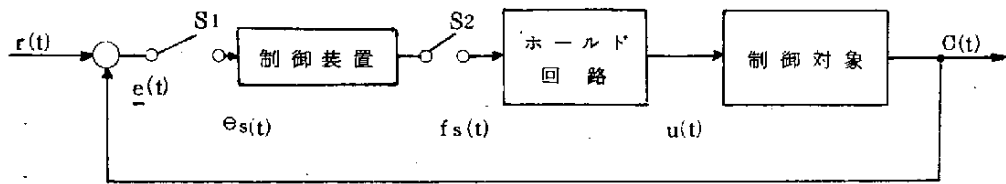
で表わされる。ホールドには通常零次ホールド

$$Gh(s) = \frac{1 - e^{-ts}}{s} \quad (9)$$

を用いる。  $ets = Z$  とおけば、  $es(t)$  のラプラス交換は

$$Es(Z) = \sum_{n=0}^{\infty} e(n\tau) Z^{-n} \quad (10)$$

となり、これを  $es(t)$  の Z 交換という。またある要素の入出力が共にサンプル値信号であるとき、入力と出力の Z 変換の比をその要素のパルス伝達関数という。



第 1 - 4 図 サンプル値制御系の構成

サンプル値制御系にはサンプラやホールド回路が必要であり、またサンプリングによって連続信号の持つ情報の一部が失われる。それにも拘わらずサンプル値制御が用いられるのは、以下の理由による。

- (1) 信号の連続検出が不可能または困難なとき、すなわち信号の検出に走査を要するとき、連続検出では検出装置が過負荷になる或いは寿命が短くなるようなとき、などに用いることができる。
- (2) デジタル計算機を用いることができる。むだ時間や非線形性を含む対象、或いは最適制御に特に有効である。
- (3) 1つの制御装置で多くの制御対象を時分割多重制御することができる。

サンプリング周期 はどのように選べばよいであろうか。τ が大きいと連続信号の情報の一部が失われるであろう。これに関して重要なのは、情報理論で確立されたサンプリング定理である。すなわち「連続信号  $f(t)$  が含む最高周波数の 2 倍の周波数でサンプリングすれば、サンプル値列  $f(n\tau)$  ( $n=-\infty \sim \infty$ ) より  $f(t)$  を完全に再現することができる。」一般に τ が大きくなると制御性能は悪くなるが、τ が小さいと計算量はふえる。また、むだ時間を含む系では、τ を適当に大きくした方が制御性能が良くなることがある。プロセス系で経験に基づいた値として第 1 - 3 表が提案されている。

デジタル計算機を用いた DDC (direct digital control) はサンプル値制御系である。PID 制御のサンプル時点  $n$  における調節計の出力  $mn$  は、

$$mn = K \left[ en + \frac{1}{T_I} \sum_{k=0}^n ek + \frac{T_D}{t} (en - en-1) \right] \quad (11)$$



|     | 標 準 [秒] | 範 囲 [秒]   |
|-----|---------|-----------|
| 流 量 | 1       | 0.1 ~ 1.5 |
| 圧 力 | 3       | 1 ~ 5     |
| 液 位 | 5       | 10 ~ 20   |
| 温 度 | 10      | 5 ~ 20    |
| 成 分 | 20      | 10 ~ 80   |

第 1 - 3 サンプルング周期の値

で与えられる。この信号を直接操作量とするものを位置計と言う。それに対し前のサンプル時点からの増分

$$\Delta m_n = K \left[ (e_n - e_{n-1}) + \frac{\tau}{T_I} e_n + \frac{T_D}{t} (e_n - 2e_{n-1} + e_{n-2}) \right] \quad (12)$$

を用いるものを速度計と言う。速度計は位置形に比して、誤動作したときの影響が少ない、自動-自動切換えがバンプレスに行なえる、等の利点がある。サンプルング周期が制御対象の時定数にくらべて十分小さいときには、チューニングは連続系の場合と同様に考えてよい。

サンプル値制御系では、一定の条件のもとに、ステップ、ランプ、或いは加速度入力などに対し、あるサンプル時刻後の各サンプル時点における偏差が0となるように、すなわち有限整定時間応答となるように、制御装置を設計することができる。これは連続系では考えられない1つの特徴である。さらにDDCにおいてはプログラムの柔軟性を利用すれば、PIDの原形にこだわる必要はないという考え方がでてくる。そこで例えば、積分にも各種の数値積分法を利用する、単純な微分の代りに平均微分を用いる、ゲインKの値を適当に切り換える偏差の大小によりPIDとオンオフとを切り換える(デュアル・モード制御)、等のアルゴリズムが実用されている。

なおDDCに対しては、1963年にアメリカのDu Pont社等、有力ユーザ3社が標準仕様(guide-line)を示した。その骨子は次のよ

うなものであり、現在でもこれが標準として用いられている。

(1) 稼働率が99.95%以上あること。

(2) 上位のSCC (Supervisory Control Computer) とデータ通信が可能なこと。

(3) 従来用いられている変換伝送系からアナログまたはデジタル信号を受けられること。

(4) オペレータ操作盤で定数その他の設定が可能であること。

(5) 演算制御装置は測定値などのデータの記憶が可能なこと。10ビットの演算精度をもつこと。

(6) 装値故障時に、故障直前の制御出力を保持すること。

(d) 制御技術の発達と計算機

第2次大戦中から本格的に始まった制御工学の研究は、その初期の段階(10～15年間)では主として、1入力1出力1ループのフィードバック系を対象とした。そこではラプラス変換及びZ変換を基礎とした周波数領域における解析、設計が議論の中心であった。これはスカラ制御論とも呼ばれる。この時代の実用技術を代表するものは、モータを用いたサーボ機構や、空気圧、油圧、または電気・電子式のアナログ調節計を用いた1制御量(温度、圧力、流量、液位等)のプロセス制御である。安定解析や設計の計算は手計算、計算尺、それにそれほど普及はしなかったがアナログ計算機、によって行なわれていた。アナログ計算機の結線は、伝達関数のブロック線図表示とよく適合していた。最も重要な問題は安定性、安定度の決定であったが、それにはラウス・フルヴィツの定理の他、フィードバック増巾器の設計理論として確立されていたナイキストの定理、ボード線図などが盛んに用いられた。その他にも、制御工学より先に発達した通信関係の理論や技術が母体となつたものが少なくなかつたのである。

その後、産業工程における自動化の進展に伴つて、多量の計測データを監視、記録するデータ・ログが登場し、温度、圧力、流量などの検出器、調節計の性能も向上し、また小形化した。1955年頃からは多数の調節計をプロセスの至るところに配置して、それをグラフィック・パネルで一目瞭然に監視することが行なわれるようになる。また、ブール代数を基礎

とした論理設計法も形を整え、シーケンス制御に応用された。

その頃一方では、トランジスタを用いたデジタル計算機が実用化されてくる。初めはコストや信頼性の点から直接制御に用いることはなかった。しかし、時あたかもアメリカ、ソ連を中心とした宇宙航法開発が進められ、そこでは経済性が問題とされることはなかったし、複雑な非線形不規則過程の最適制御なども要求されたために、特殊なケースではあったが、制御用計算機として威力を発揮する場を与えられたわけである。それがまたさらに、材料や回路の集積化を含むデジタル技術の進歩を促した。民間産業においても、複雑な多変数動的過程の高度な制御（非干渉、非線形、適応、学習、最適など）、高精度な制御に対する要求と認識が高まるにつれ、かつまたデジタル素子の小形化、低コスト化、高信頼度化が進むにつれて、デジタル計算制御が次第にアナログ制御に取って変わるようになる。

以上のように、産業における自動化・省力化・宇宙開発のようなナショナル・プロジェクトの実施、デジタル技術の発達、の3者が互いにうまく絡み合つて（ポジティブ・フィードバック作用を及ぼし合つて）、制御技術の変遷、デジタル化が急速に進められてきたのである。もちろんアナログ調節計やアナログ計算機が消え去つたわけでもないし、また近い将来に完全に姿を消すとも思われない。アナログの演算増巾器もIC化が進み、単体或いは専用の演算要素として応用分野を広げ、またデジタル要素と結合したハイブリットの中で、連続動的過程の演算に強いという長所を生かしている。しかしLSI、超LSIと進むにつれて、デジタル情報処理の比重がますます高まっていくのが大勢であろう。

計算機を中心とした技術発達の主な出来事を第1-4表の年表に掲げた。デジタル・プロセッサの小形化、超小形化の歴史において、特に2つの出来事に注目したい。1つは1958年に始まるICの実用化、殊に63年頃のFETの完成である。高い入力インピーダンスのトランジスタ回路が実現されて完全に真空管を駆逐し、同時により高密度の集積化、すなわちLSIへの道を開いたのである。もう1つは、DEC社が63年に発表したPDP-5、引き続いて65年に発表したPDP-8、すなわちミニコンピュータの登場である。その頃、既にIBM社はデジタル計算機の分

野で、技術、市場占有率共に抜群であつたが、その指向するところは超大形汎用、バッチ処理、ビジネス情報処理、であつた。DEC 社は発想を 180° 変換して、低価格、(超)小形専用、オンライン・リアルタイム処理、制御を中心とし

| 年    | 制 御 工 学                       | 計 算 機                   |
|------|-------------------------------|-------------------------|
| 1946 |                               | ENIAC (Pennsylvania 大)  |
| 1947 | 記述関数 (Nichols 他)              | 電子管式アナログ計算機             |
|      | 電子管式プロセス制御装置                  |                         |
| 1949 | サンプリング定理 (染谷-Shannon)         | EDSAC (Cambridge 大)     |
| 1951 |                               | UNIVAC-I, コア・メモリ        |
| 1952 | NCフライス盤 (MIT)                 | ETL MARK-I (電気試験所)      |
|      | サンプリング制御系                     | DDA (Northrop Aircraft) |
| 1954 | サーボ・アナライザ (Honeywell)         | IBM-650                 |
|      | データ・ログ                        | ELD (GE, WH 等)          |
|      |                               | パラメトロン (後藤)             |
| 1955 |                               | FACOM 128, 138          |
| 1957 | SCR (GE)、ステップ・モータ             |                         |
|      | パーセプトロン (Rosenblatt)          |                         |
| 1958 |                               | IC (Texas Instruments)  |
|      |                               | MUSASINO-I (電々公社)       |
| 1959 | SPUTNIK-I (ソ連)                |                         |
|      | 流体論理素子                        |                         |
| 1960 | 現代制御理論 (Pontryagin, Kalman 等) | IBM-7090                |
|      | NCボール盤 (富士通)                  | ハイブリット計算機 (日立)          |
| 1962 |                               | DDC (Ferranti)          |
| 1963 |                               | FET, PDP-5 (DEC)        |
| 1964 | 新幹線みどりの窓口 (国鉄)                | IBM-360                 |
| 1965 | 計算機制御自動給電 (九州電力)              | PDP-8 (DEC)             |
| 1966 |                               | TSS (MIT)               |
| 1969 | APOLLO-11 月着陸 (アメリカ)          | PDP-14                  |
| 1970 |                               | IBM-370                 |
|      |                               | ARPA ネットワーク             |
| 1971 | PLC (日本各社)                    | マイクロプロセッサ MCS-4 (Intel) |
|      |                               | データ通信 (電々公社)            |
| 1972 |                               | MCS-8 (Intel)           |

第 4 表 制御工学と計算機の発達

た産業情報処理、をはつきりと指向したのである。開発投資も比較的少なくすむために、新しい技術を吸収したモデルチェンジもし易く、ユーザも手軽に購入し無駄なく使用できるために、急速に普及し、既に1970年代の初期に全コンピュータの50%近い設置台数を獲得することになった。

上記の2つの流れが(もう1つの電卓の発達普及も加えるべきかもしれないが)、やがて1971年末のIntel社による最初のマイクロプロセッサMCS-4(4004)、72年のMCS-8(8008)の発表へとつながる。このように歴史の流れを振り返ってみれば、産業用機器におけるマイクロプロセッサ、マイクロコンピュータの占めるべき位置と将来性がおのずとわかってくるように思われる。

ディジタル制御の導入と発達につれて、(a)~(c)で述べたようなスカラー理論は不十分で、理論と技術の適合性が悪くなってきた。すなわち多変数系の表現、入出力だけでなく内部構造、内部状態の表現、非線形性の表現、等が必要になってきた。そこで次に述べる状態方程式による理論化が進められたのである。

(e) 状態変数による表現—近代制御理論の出発点—

まず先に述べた伝達関数による表現の欠点について、まとめてみよう。

- (1) 入出力関係を知ることはできるが、内部状態はわからない。
- (2) 初期状態の影響が無視される。初期状態には過去の入力の影響が集約されているので、それをすべて無視することになる。
- (3) 多入力多出力系の取扱いに適さない。
- (4) 時変系、非線形系に適用できない。

これらの難点を克服して高度な制御を実現するために、状態変数の微分方程式による系の表現が用いられる。状態変数とは、ひらたくいえば、系の内部状態を表わすのに必要にして十分なだけの変数である。従って、状態変数は1つの系に対して1組だけ定まるとは限らない。

状態変数が $n$ 個あったとして、それらを $x_1(t)$ 、 $x_2(t)$ 、…… $x_n(t)$ としよう。さらに $m$ 個の入力を $u_1(t)$ 、 $u_2(t)$ 、……、 $u_m(t)$ とし、 $r$ 個の出力を $y_1(t)$ 、 $y_2(t)$ 、……、 $y_r(t)$ とする。表現を簡単にするために、

それぞれをベクトル  $x(t)$ ,  $u(t)$ ,  $y(t)$  で書けば、一般に連続系の状態変化とそれに伴う出力は、次の方程式で表わせる。

$$\dot{x}(t) = f(x(t), u(t), t) \quad (13.a)$$

$$y(t) = g(x(t), u(t), t) \quad (13.b)$$

(13.a) の一階(連立)微分方程式を 状態方程式、(13.b) 式を 出力方程式 と呼ぶ。状態変数の変化を直観的に分かり易く幾何学的に表わすのに、 $n$  個の  $x_i(t)$  を座標とする  $n$  次元空間を考え、これを 状態空間 と言ひ、その中での状態変化の軌跡を 状態軌道 と言う。

(13) 式の  $f$  及び  $g$  が共に線形ならば、行列  $A(n \times n)$ ,  $B(n \times m)$ ,  $C(r \times n)$ , 及び  $D(r \times m)$ , を用いて

$$\dot{x}(t) = A(t)x(t) + B(t)u(t) \quad (14.a)$$

$$y(t) = C(t)x(t) + D(t)u(t) \quad (14.b)$$

と書ける。これを 線形系の標準方程式 と言う。時間的に不変な系ならば、 $A, B, C, D$  はすべて定数行列であり、

$$\dot{x}(t) = A x(t) + B u(t) \quad (15.a)$$

$$y(t) = C x(t) + D u(t) \quad (15.b)$$

となる。

定係数線形系の伝達関数が与えられたとき、それを標準方程式に書き直すには、次のような考え方をすればよい。入力  $u$  と状態  $x$  の間の伝達関数が基本形  $1/(s+a)$  ならば、変数  $x$  について

$$s x = -a x + u \quad (16)$$

なる状態方程式が得られる。高次の伝達関数の場合はこれをいくつかの基本形に分解し(積の形、或いは部分分数展開による和の形)、その結合を考慮して標準形を導けばよい。フィードバック系のブロック線図から状態方程式と出力方程式を導くこともできる。それにはまず各ブロックの状態及び出力方程式を求め、次に各ブロックの状態変数の集合を全系の状態変数とし、ブロックの結合を考慮してまとめればよい。

逆に、定係数線形系の標準方程式から伝達関数を求めることもできる。

(15.a) 式の解は

$$x(t) = e^{At} x(0) + \int_0^t e^{A(t-\tau)} B u(\tau) d\tau \quad (17.a)$$

で与えられる。出力  $y(t)$  は (15.b) と (17) 式より直ちに求まり

$$y(t) = C e^{At} x(0) + C \int_0^t e^{A(t-\tau)} B u(\tau) d\tau + D u(t) \quad (17.b)$$

となる。(17.b) 式をラプラス変換すれば、 $e^{At}$  のラプラス変換は  $(sI - A)^{-1}$  であるから ( $I$  は  $n \times n$  単位行列)、

$$Y(s) = C (sI - A)^{-1} x(0) + [C (sI - A)^{-1} B + D] u(s) \quad (18)$$

となり、定義により入出力間の伝達関数  $G(s)$  は

$$G(s) = C (sI - A)^{-1} B + D \quad (19)$$

である。

状態変数による系の表現によれば、入出力と内部状態との関係を考慮に入れて系の解析や設計を行える。通信系からのアナログで入出力関係だけに着目していると、情報のみでなく物質やエネルギーを含む制御系では、内部状態の考慮によって初めて明らかになる可制御性や可観測性を損った設計をする恐れがある。

可制御性とは、入力により状態を希望通りに制御できるということであり、可観測性とは、出力の観測により状態を決定できるという性質である。これは1960年にKalmanによって初めて議論された。より正確に言えば、可制御性：任意の初期状態が与えられたとき、系の状態を有限時間内に任意の目標値に移すことができるならば、系は可制御であると言う。

可観測性：初期時刻から有限時間内の出力の観測値によって初期状態を決定できるならば、系は可観測であると言う。(15)式の定係数線形系に対しては、係数行列を用いて直接に可制御性及び可観測性を判定できる。すなわち、

可制御性の定理 (15) 式の系が可制御であるための必要十分条件は

$$Q \triangleq [B, AB, \dots, A^{n-1}B]$$

なる行列  $Q$  のランクが  $n$  なることである。

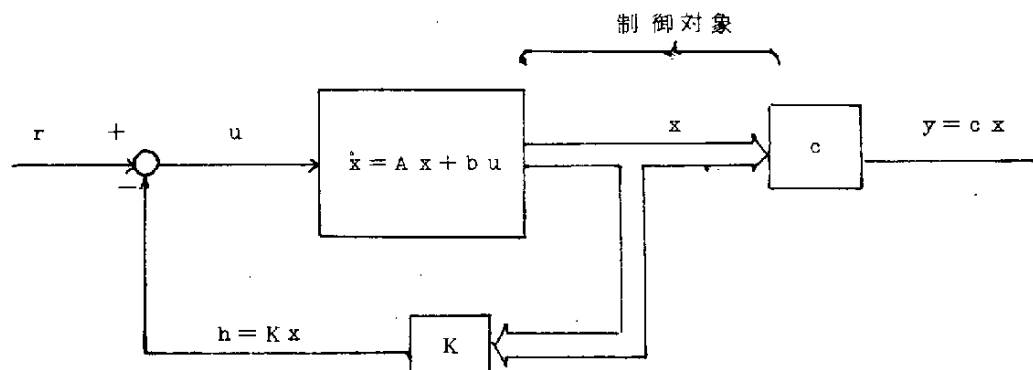
可観測性の定理 可観測であるための必要十分条件は

$$P \triangleq [C', A'C', \dots, A^{n-1}C'] \quad ( ' \text{ は転置} )$$

なる行列  $P$  のランクが  $n$  なることである。

出力  $y(t)$  から状態  $x(t)$  を求めるには、雑音の無いときにはオブザーバを用い、雑音 (正規性白色) の有るときにはカルマン・フィルタにより

最適推定を行なう。状態が求まったら、第1-5図のような状態フィードバックによる閉ループ系の設計が可能になる。制御対象が可制御なら



第1-5図 状態フィードバックによる制御系

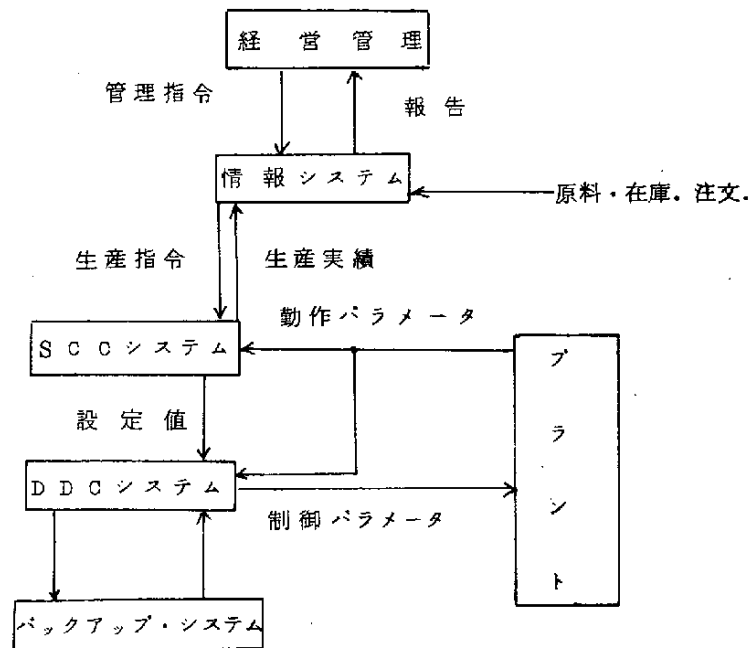
ば、フィードバック係数ベクトル  $K$  を適当に選ぶことにより、閉ループ伝達関数の極を任意に指定することができ、 $K$  の値にかかわらず制御系を可制御にすることができる。その他リアプノフの定理、ポポフの定理による非線形系の安定判別（安定化）や、ポントリャーギンの最大原理、ベルマンの動的計画法による最適制御などにも、状態変数表現の威力が発揮される。

#### 1.2.4. システムの概念と計算機ヒエラルキ

NASAの宇宙開発計画、なかんずくアポロ計画の成功に触発されて、システムという言葉や概念が極めて活発に語られるようになった。最近ではやや落ち着いてきたものの依然として言葉の乱用、概念の混乱は続いている。システム概念が多くの専門分野から別々に発生している以上、混乱はある程度致し方のないところであろう。ここでは計算機を中心とした生産システムに限って、極く簡単にふてみよう。

デジタル計算機の大形高速化に伴って情報処理能力が向上すると、単に生産の一工程をDDCで制御するだけでなく、第1-6図のような





第 1 - 6 図 生産管理制御システム

計算機のヒエラルキ・システムによつて、総合的な生産管理制御システムを構成することが行なわれるようになる。情報システムは企業レベルであり、SCCシステムはプラント・レベルであつて、プロセス変数の最適設定値をDDCに指令する。計算機は下位になるほど、ミニコンピュータ、マイクロプロセッサというように、小形専用になる。マイクロプロセッサが大きなトータル・システムのフロント・エンドとして用いられることも、今後ますます盛んになるであろう。

### 1.3. 制御系におけるセンサと情報処理

#### 1.3.1. センサとトランスデューサ

制御は制御対象の状態を把握することから始まる。その先端に位置するものがセンサ（検出器）である。近来、計測の対象となるシステムは複雑化する一方であり、計測データの量も膨大になることが多い。また

それを基にして質の高い制御を行なうことが要求される。その故に、多量の計測データが生のまま利用されることはむしろ稀で、それを集録、整理、処理、加工して必要な情報を抽出することが多く行なわれる。情報処理のためのさまざまな要求に、最も柔軟に対応できるのは、いうまでもなくデジタル計算機である。計算機処理のためには電気信号が必要であり、その場合センサは、各種の測定量を電気量に変換するトランスデューサ、電気信号を伝送する通信伝送器と一体となって計測系を構成している。

トランスデューサによって測定される量には、変位（長さ）、回転角、力、トルク、速度、振動などの機械量（力学量）と、温度、圧力、流量、流速、液位、成分などのプロセス量とがある。トランスデューサ分類するには、測定量により分ける方法、測定原理によって分ける方法などが考えられるが、実用上は、何から何に変換されるかに注目することが便利であろう。第1-5~7表では縦軸に入力量、横軸に出力量をとって、それぞれの変換に現在用いられている主な法則、手段を示したものである。出力が電気量以外である場合はそれをさらに電気量に変換することが必要であるが、例えば、変位の場合は差動変圧器、圧力の場合は圧電素子を用いれば容易に変えられる。一般にトランスデューサを組み合わせ、 $A量 \rightarrow B量 \rightarrow C量 \cdots$ という複合トランスデューサを構成することもよく行なわれる。

計測系で適当なトランスデューサを選定することは、実際にはなかなか難しい問題である。選定にあたって考慮すべきことは、測定の目的、測定量の選定と測定範囲、測定精度、設置場所と環境条件、さらに伝送系と処理装置の種類と特性、威振電源の容量、負荷、フィルタの必要性であり、また設計にあたっては変換原理の選定、入、出力インピーダンス、寿命、検査・校正の方法、操作性、などに配慮すべきである。

近年のトランスデューサ技術の1つの傾向として、固体素子の利用、特に半導体の利用があげられる。1種類の半導体で、光、熱、磁気に敏感なものも多く、多用途のトランスデューサを構成できる。

|       | 電 圧                                 | 電 流   | 周波数           | 抵 抗                         | 容 量   | インダクタンス | 光                | 圧 力                          | 変位    |
|-------|-------------------------------------|-------|---------------|-----------------------------|-------|---------|------------------|------------------------------|-------|
| 変 位   | 差動変圧器<br>インダクション<br>磁気格子<br>ホール素子   | 可動電極管 | 振 動 弦         | 液体低抗<br>ポテンシヨータ<br>ストレインゲージ | 容量マイク | 誘導マイクロ  | 油圧マイクロ<br>空気マイクロ | 光波千渉<br>もあれ縞<br>光学繊維<br>焦点ずれ |       |
| 厚 み   | 電磁<br>渦電流<br>放射線                    |       |               | 抵 抗                         | 容 量   |         |                  |                              |       |
| 角 変 位 | 差動変圧器<br>リゾルバ<br>光電オートコリメータ<br>ジャイロ | 可動電極管 | 磁気ドラム         | 対流<br>ポテンシヨータ               | バリコン  | 誘導マイク   |                  | 光波千渉<br>もあれ縞<br>符合板<br>光てこ   |       |
| 速 度   | 動コイル                                |       | もあれ縞<br>ドラム効果 |                             |       | 誘導      |                  |                              |       |
| 角 速 度 | タコジエネ                               |       | 歯車<br>リングレーザ  |                             |       |         |                  |                              | ドラタカテ |

第 1 - 5 表 トランスデューサの分類 (その 1)

|       | 電 圧           | 電 流                     | 周波数             | 低 抗                 | 容 量 | インダクタンス | 光 | 圧 力            | 変 位                                |
|-------|---------------|-------------------------|-----------------|---------------------|-----|---------|---|----------------|------------------------------------|
| 圧 力   | 電気動学的<br>圧電素子 | 感圧素子                    | 水晶振動子           | ピエゾ低抗<br>ストレイニングゲージ | 誘電率 | 磁わい     |   |                | 液柱U字管<br>リングバラン<br>ダイヤフラム<br>ブルドン管 |
| 真 空 度 | 熱電対           | 電離<br>マグネトロン<br>ベニングゲージ |                 | サーミスタ<br>ピラニゲージ     |     |         |   |                | マクラウド                              |
| 流量・流速 | 電磁<br>超音波     |                         | レザドップラ<br>カルマン渦 | 熱線                  |     |         |   | オリフィス<br>ビトーマン | 質量<br>面積式容積式<br>異 車                |
| 液 位   | 超音波           | 放射線                     |                 | 短絡                  | 電極  |         |   | 空気圧管<br>気泡     | ゲージグラス<br>フロート                     |

第 1 - 5 表の 2

|     | 電 圧              | 電 流                   | 周波数 | 低 抗                     | 容 量 | インダクタンス | 光              | 圧 力 | 変 位                           |
|-----|------------------|-----------------------|-----|-------------------------|-----|---------|----------------|-----|-------------------------------|
| 温 度 | 熱放雑<br>電射音<br>対  |                       | 水 晶 | 白金<br>ゲルマニウム<br>スタ<br>ミ | 誘電車 | 磁化率     | 色、磁気<br>放射、光   | 蒸気圧 | バイメタル<br>ゼーグル錐<br>体膨張<br>気体・液 |
| 湿 度 |                  |                       |     | 塩化リチウム<br>マグネタ<br>イム    | 誘電率 |         | 赤外線吸収          |     | 毛髪<br>乾湿球                     |
| 光   | 光電池              | 光電子<br>倍増管<br>ホトランジスタ |     | 光導電素子                   |     |         |                |     |                               |
| 熱   | 熱電対<br>サーモ<br>イル |                       |     | ボロメータ                   | 薄膜  |         | ゴレー<br>セル<br>イ |     |                               |
| 粘 度 | 振動片              |                       |     |                         |     |         | 光学式            | 細 管 | フ<br>ロ<br>ー<br>ト<br>回 転       |

第 1 - 6 表 トランスデューサの分類 ( その 2 )

|         | 電 圧 ・ 電 流                      | インピーダンス         | 光                    | 温 度                                              | 音      |
|---------|--------------------------------|-----------------|----------------------|--------------------------------------------------|--------|
| 気 体 成 分 |                                | 導電度測定セル         | 光千渉<br>屈折式<br>光電硫化水素 | 赤外線<br>磁気酸素<br>密度式<br>燃焼式<br>(ガスクロマトグラフ)<br>熱伝導式 | 音 響 式  |
| 液 体 成 分 | 電磁濃度計<br>PH 計<br>滴定式<br>ポラログラフ | 誘電濃度計<br>電極式濃度計 | 偏光<br>光電比色           | 塩酸濃度計                                            | 超音波濃度計 |
| 固 体 成 分 | X 線<br>熱起電力                    | 透磁率             | 発光分光分析               |                                                  |        |

第 1 - 7 表 トランスデューサの分類 ( その 3 )

またチタン酸バリウムよりもはるかに優れた電圧係数をもつものなどもある。これらの特徴を利用できる他、その製造技術を生かして集積化を行なったり、さらには信号処理の機能をも合わせもつような機能化をはかることもできる。

放射線や光を利用した非接触計測の方向も制御系においては重要である。それによって、連続、高速、オン・ラインの計測と制御が可能になるからである。また現在でもトランスデューサの出力はアナログ量が多いが、ディジタル処理のためには、A/D変換器を要するので、好ましくない。そこで、機械的変位や回転角のディジタル計測がNC技術の発達と共に普及し、プロセス量でもディジタル出力温度計などが開発されている。true digital transducerの研究開発は将来ますます盛んになるであろう。

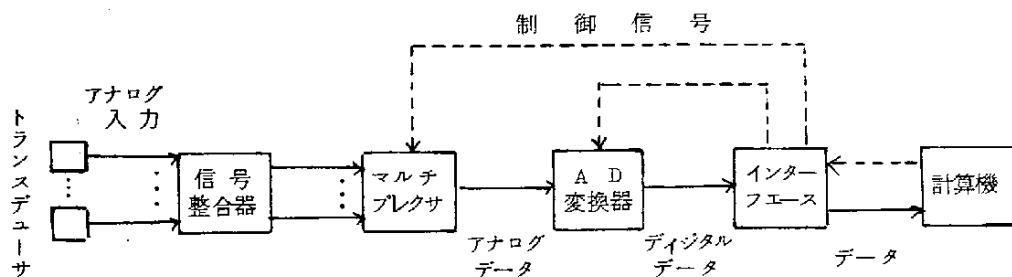
### 1.3.2. アナログからディジタルへ

ディジタル信号は、前にも述べたように演算処理を施すのに極めて都合がよい他、記録が容易なこと、伝送に際して雑音の影響を除去し易いこと、などの特徴があり、最近の計測制御系で欠くことのできない手段になっている。しかし、トランスデューサの出力は多くの場合アナログ信号であるから、それを計算機に入力したり、ディジタル記録したりするにはA/D変換器を必要とする。また逆に計算機の出力を外部のアナログ機器、操作端などに伝えるにはD/A変換が必要である。

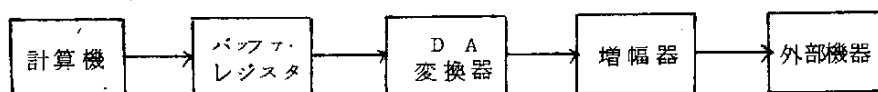
#### (a) 計算機への入力

アナログ信号をディジタル処理する過程には、オン・ライン方式とオフ・ライン方式の2方式がある。第1-7図はオン・ライン処理方式の構成を示す。トランスデューサからのアナログ入力信号はまず信号整合器においてフィルタによる雑音の除去などが行なわれ、マルチプレクサ(入力切換器)により多数の入力が順次サンプルされる。マルチプレクサには、リード・リレーが水銀接点リレーが使われることが多いが、最近ではFETなどの半導体スイッチも用いられる。その後適当な電圧( $\pm 5$  V程度)に増巾してA/D変換し、計算機に読み込まれる。

オン・ライン方式ではディジタル・データをデータ集録装置に記録しておき、あとで必要に応じて計算機で処理する。記録には印字、紙テープ、カード、磁気テープなどが用いられる。



第1-7図 計測データのオン・ライン処理系



第1-8図 計算機からアナログ機器への出力系

データ・ログは集録装置の一種で、ディジタル・データを一定時間間隔で印字記録する他、通常測定値が異常な値となったときは警報を発する監視機能をそなえている。なかには、ある程度の演算処理機能をもつコンピューティング・ログと呼ばれるものもある。ログには入力信号を線形化する回路、適当なスケーリングを行なう補正回路（以上データの前処理）、印字記録の作表を制御する回路、正常な測定値の上下限を設定する回路、測定値と設定値を比較し必要なとき警報を発する回路、などが含まれているが、小規模なログではこれらの制御及び演算を行なうのに、8ビットのマイクロプロセッサとROMで十分である。固定配線を用いた方式にくらべて、ハードの変更が極めて少なくてすむ。規模が大きいときでも、センサと組み合わせたマイクロコンピュータを現場に配置し、データの前処理、警報はそこで行ない、処理済みの必要なデータだけをモデム（変復調装置）、マルチプレクサを介して中央計算機に送り、記録すればよい。海洋観測データ用の集録装置など、特殊な用途

にも実用されている。

#### (b) 計算機からのアナログ出力

計算機から外部の操作盤などに、アナログ出力信号を取り出すこともしばしば必要である。その信号伝送系の構成を第1-8図に示した。バッファ・レジスタは、1つの計算機からいくつかの外部機器へ、並列的にアナログ出力を出すときに用いられる。

#### 1.3.3. デジタル計測量の処理

デジタル化された計測信号を計算機によって論理処理、演算処理することは容易であり、それらの処理によって計測技術の幅を拡げ、また高度の情報を取り出すことができる。ここではマイクロプロセッサ利用の立場から、そのうちのいくつかを検討してみよう。

初めに、応用からみたマイクロプロセッサの特徴をまとめてみれば次のようである。

- (1) 極めて小形であり、また低価格である。
- (2) ROM 或いは PROM と組み合わせることにより、要求された機能を満足する、可変のプログラムによる動作が可能である。
- (3) ROM 或いは RAM と組み合わせることにより、多量の数値情報を記憶し処理することができる。
- (4) 処理機能としては数値演算、論理演算、及びかなり複雑な判断ができる。
- (5) 必要に応じていくらかでも高精度の数値演算が行なえる。
- (6) 演算速度はミニコンピュータに比べて1桁程度遅く、加算時間で2 (NMOS) ~ 20 (PMOS)  $\mu$ s 程度である。将来バイポーラ素子を用いるようになれば、格段に速くなるであろう。ハードの乗除算機構をもつものは今のところほとんどなく、加算に比べて数倍遅い。
- (7) 命令は一般に少なく、割込機能も十分でないものが多い。

(6)及び(7)の欠点は日進月歩的に改善されつつある。

#### (a) 間接測定

測定には直接測定と間接測定とがある。間接測定は、測定量と一定の物理的關係にあるいくつかの独立な量の直接測定値から、計算によって

測定量を決める方法であり、導線の電気抵抗、長さ、及び断面積から比抵抗を求める場合などである。従って途中に計算が入る。この計算が自動化されるならば、結果を直読できる測定になり、オン・ライン・データとして用いることができる。

例えば管内を流れる流体の流速を超音波流速計により測定する場合を考えてみよう。超音波を流れと順方向及び逆方向に角  $\theta$  傾けて発射し、共に距離  $d$  だけ伝播させると、伝播時間は順方向及び逆方向でそれぞれ

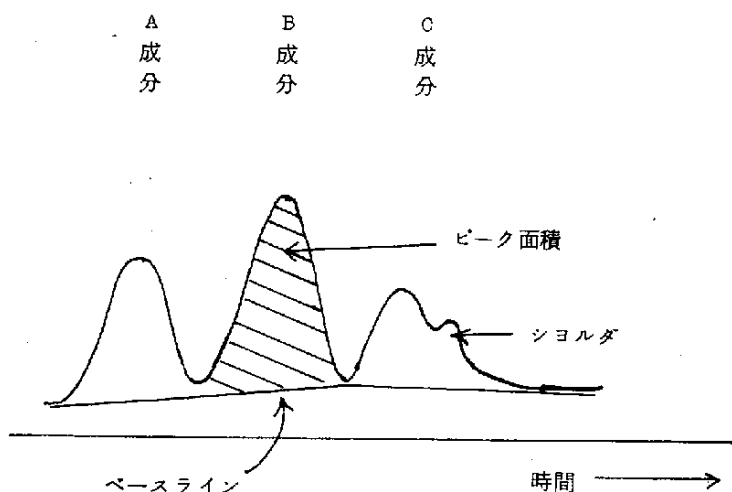
$$T_1 = d / (V_a + V \cos \theta), \quad T_2 = d / (V_a - V \cos \theta) \quad \text{である。}$$

ただし  $V$  は流体の流速、 $V_a$  は静止流体中での超音波の伝播速度である。 $T_1$  と  $T_2$  を測定し（またはそれを用いて周波数  $f_i = \frac{1}{2} T_i$  の発振波をつくり、 $f_1$  と  $f_2$  を測定し）、演算処理すれば  $V$  を直読できる。さらに音響インピーダンスから流体の密度を計算し、流速に乗ずれば質量流量が直読できる。このような簡単な計算処理ならば、マイクロプロセッサで十分である。

上の例では簡単な四則演算のみでよいが、もっと複雑な計算、例えば炉内全体の温度分布の概略を数点の測定値から偏微分方程式を解いて推定するなども、小形計算機でできるであろう。

測定中にデータ処理を要する代表的なものに各種の分析計がある。その1つとして、ガス・クロマトグラフを取り上げてみる。ガス・クロマトグラフはプロセス工業で、中間製品や最終製品の品質を分析し、工程を制御するための連続自動分析装置として広く使われている。固定相（シリカ・ゲルなど）をつめた管（カラムという）に、移動相としてキャリア・ガスと試料ガス（多成分系）を流入させると、各成分の固定相内での移動速度が異なるため、カラムの出口ではキャリア・ガスと特定成分の2成分系となる。この原理を利用して、カラムの入口で一定のサンプリング周期ごとに一定容量の試料ガスをキャリア・ガスに注入し、出口で成分を観測すると、異なった時刻に異なった成分のピークをもつ第1-9図のような波形、いわゆるクロマトグラムが得られる。各成分の濃度は、その成分に相当するクロマトグラムのピーク部分の面積、すなわち積分値で与えられる。

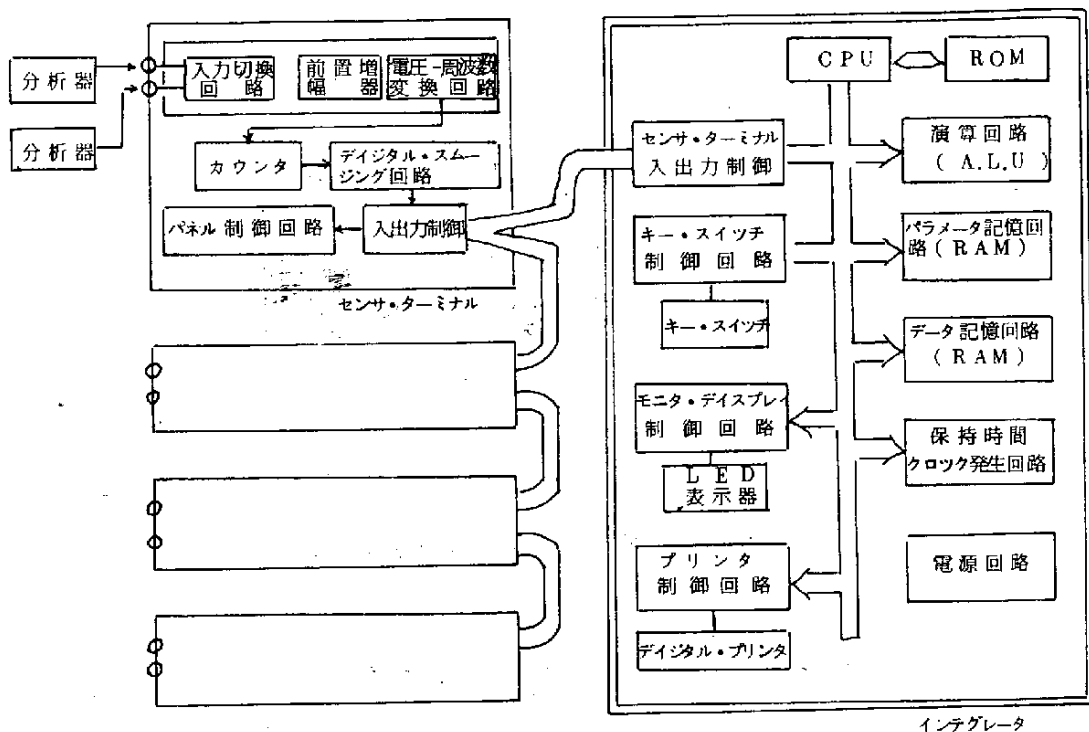




第 1 - 9 図 ガス・クロマトグラム

この積分値はほぼピーク値に比例するとして、単にピーク値だけを測定する方法も用いられている。

しかし正確な面積を算出するには、エレクトロニック・インテグレータを用いるのがよい。クロマトグラフィ用インテグレータには、電圧一周波数変換、ベースライン補正、ピーク検出、ピーク保持時間計数、面積計数、制御の各回路が必要である。主にアナログ回路で構成するものでは、ベースラインの補正、重合ピーク（フロントシヨルダ、バックシヨルダ、スライス）の分離、ピーク面積の計算などの能力に限界があり、また分析器 1 台ごとにインテグレータを必要とする不利がある。ミニコンピュータを用いてデジタル波形処理をし、分析器が多数あるときはマルチプレクサを用いれば、上記の難点は解決されるが、価格の上昇は避けられない。そこで、マイクロプロセッサの利用が考えられるわけである。演算速度はあまり要求されないので、その点でもマイクロプロセッサが適している。



第1-10図、マイクロ・デジタル・インテグレータ

第1-10図はマイクロCPUを用いて実現されたインテグレータの例である。ガス・クロマトグラフからの出力は一般に低い周波数の信号で、1ピーク経過するのに数秒～数分かかり、1周期の測定に通常数十分を要する。そこで数台の分析器からの信号を並列処理して分析の効率を上げる。クロマトグラムからピーク面積（或いはピーク値）を計算し、それからさらに一定の計算式より成分を算出し、結果を印字出力する。また、クロマトグラフの状態や試料ガスの種類に応じて面積計算のパラメータを調整する作業、ドリフトに伴い変動するベースラインをピーク波形の立上り、立下り時の値及びその間の時間より確定する作業、雑音的な微小ピーク面積を求める作業、等もすべて、4ビット或いは8ビットのマイクロプロセッサで行なっている。第1-10図のV-F変換回路とカウンタの代りに、AD変換器を用いることもある。そのときは毎秒5回程度の周期でサンプリングを行ない、10MHz程度のパルス列を周波数変調する。

なお第1-10図のセンサ・ターミナルは各分析器の近くに配置するようになっている。このインテグレータはガス・クロマトグラフだけでなく、ピーク面積やピーク値を求める他の測定装置、例えば液体クロマトグラフ、原子吸光分析器、CHN元素分析器などにも利用できる。

#### (b) 不規則信号の処理

時間的に不規則な変動をする信号やデータを得た場合に、その瞬時値に次々と注目するのは無意味であって、何ら有用な情報は得られない。多量のデータを処理して、何らかの統計的な量を算出すること、言い換えるならば、有用な形にデータを圧縮することが必要である。そこで、どのような統計量を抽出するのがよいかという問題が生じてくる。通常よく用いられるのは、相関関数及びパワスペクトルである。

単一の信号 $X(t)$ の性質を表わすのが自己相関関数であり、

$$\Phi_x(\tau) = \lim_{T \rightarrow \infty} \frac{1}{T} \int_0^T x(t)x(t+\tau) dt \quad (20)$$

で与えられる。2つの異なった信号 $X(t)$ 及び $Y(t)$ の相互関係を表わすのには、

$$\Phi_{xy}(\tau) = \lim_{T \rightarrow \infty} \frac{1}{T} \int_0^T x(t)y(t+\tau) dt \quad (21)$$

で定義される相互相関関数を用いる。 $\Phi_x()$ 及び $\Phi_{xy}(\tau)$ のフーリエ変換

$$\Phi_x(f) = \int_{-\infty}^{\infty} \Phi_x(\tau) e^{-i2\pi f\tau} d\tau \quad (22)$$

$$\text{及び} \quad \phi_{xy}(t) = \int_{-\infty}^{\infty} \phi_{xy}(\tau) e^{-i2\pi f\tau} d\tau \quad (23)$$

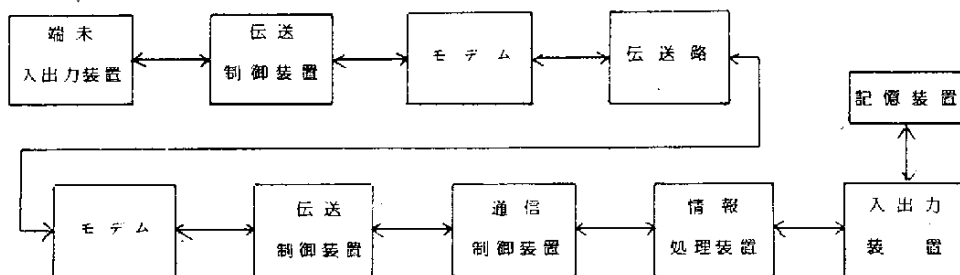
をそれぞれ  $x(t)$  のパワスペクトル及び  $x(t)$ 、 $y(t)$  のクロススペクトルと言う。パワスペクトルは、信号に含まれる周波数成分が平均としてどのようなものであるかを示すもので、バンドパス・フィルタを用いて直接測定することもできる。

実際の数値計算では  $T$  を有限値とせざるを得ないから、(20)、(21) 式の計算に一定の誤差を生ずる。また(22)、(23)式でも  $\infty$  を有限とするための誤差を生ずる。またデジタル計算では、積分計算を離散化した離散時間フーリエ変換を用いることになる。離散時間有限フーリエ変換は、データ数が2のべき乗 ( $2^P$ 、 $P$ : 整数) のとき極めて高速に行なうことができ、そのアルゴリズムを高速フーリエ変換 (FFT) のアルゴリズムと呼んでいる。以上のような統計処理を、マイクロコンピュータだけで高速に行なうのは現状では無理であるが、あまり処理速度を要しないプロセスなどで、将来実用化が考えられる。

さらに、パターン情報処理などの一部に用いることも可能になるであろう。

#### 1.3.4. デジタル・データ伝送

デジタル計測系ではデジタル信号の伝送が必要になる。より大きな観点から見ても、今日の社会生活において、情報の供給と分配を円滑に行なう必要性は急速に増しつつあり、大量の情報を誤り無く伝送する手段として、計算機の高速度処理、論理判断をオン・ラインで利用する技術が発達してきた。我国では1971年に電々公社の通信回線が民間のデータ伝送に解放されて、データ通信が本格化された。その他、比較的小規模の計測データ伝送、計算機と各種端末間の伝送、計算機互間の伝送も、さまざまな形で実現している。



第1-11図、データ伝送系の構成

第1-11図はデジタル・データ伝送系の基本的な構成を示したものである。伝送制御装置の出力は0, 1のパルスであり、モデム(変復調装置)で変調して伝送路信号に変換する。伝送にはさまざまな制御が必要である。送受信端間のデータ・リンクの設定、両端の同期調整、伝送中の誤りの修正、伝送の終止、データ・リンクの切断などである。各種の入出力端末と広範な通信網との結合を円滑にするために、最近では伝送符号、伝送形式、装置間のインターフェース、伝送制御手順、誤り制御などについて、国際的な標準化が進みつつある。

入出力端末には多種多様なものがある(銀行、座席予約、教育など)。それらの端末装置にも制御機能が設けられており、送受信装置の選定、送受信準備完了のチェックと表示、クロック、アラーム、リセットなどの制御をする。

伝送制御装置及び入出力端末において要求される制御機能は比較的簡単で、また専用的な性格のものであるから、早いうちからマイクロプロセッサの応用が試みられている。端末装置に関しては、第1・9節を参照されたい。

#### 1.4. 物体の動きの制御

物体(固形体)の位置、運動の制御は、最も基本的な制御操作の1つである。詮ずるところはサーボ機構による制御であると言えようが、実用上大きな意味をもつのは、工作機械やロボットのようにかなり複雑な動きを要求するシステムに、サーボ機構を組み込んだ場合である。そのような場合、制御対象物体が指定された径路に沿って動くようにするためには、指令の情報処理が必要となり、最近では、それに小形の計算機が使われるようになってきた。この節では代表例として、工作機械とロボットを考えてみよう。

##### 1.4.1. 数値制御

工作機械の数値制御、いわゆるNCは、物体の動きを制御するサーボ技術の代表的な応用例である。NCの初めは、1952年にアメリカMITで試作公開されたNCフライス盤である。当時は回路素子として真空管を用いたため、制御装置は工作機械本体よりも遙かに大きかったと言われているが、

その後機械工業における一方の旗頭として幾多の開発努力が積み重ねられて、今日の普及を見るに至っている。

現用のNC系では、まず工作物の寸法や加工条件及び作業順序を数値情報として紙テープにせん孔し、それを情報処理装置に入力する。その出力は指令パルス列であり、工作機械を駆動するためのサーボ機構及びシーケンス制御装置の入力となる。狭い意味でのNCは工作サーボ機構及び工具の主軸速度の制御を指し、シーケンス制御は一応別に考えることができる。

この意味でのNCは3種類に分類することができる。すなわち、

(1) 位置決めNC。これは工具の最後の位置のみを決める制御で、ボール盤による穴あけ作業がその例である。そのとき工具は所定の穴あけ位置に次々と移動するが、移動中の工具の通路制御は不要であり、速く移動させ正確に所定位置に止めればよい。最も簡単なNCであって、情報処理装置はテープの指令する位置と現在位置を記憶し、それを比較して差が零になったとき工具を停止させる。

(2) 直線切削NC。工具は直線通路に沿って移動するが、移動中にも切削等の作業を行なうもので、平削盤などに用いる。情報処理装置は工具通路の指定、移動速度の指定の他、工具の選択、工具寸法の補正、主軸速度の制御などの補助的指令も出力しなければならない。

(3) 連続切削NC。工具が任意の形の曲線通路に沿って移動しつつ切削するもので、フライス盤などに用いる。X或いはY軸方向に一定の設定移動単位(1~10 $\mu$ m程度)で動かす動作を組み合わせ(パルス分配)、指定された曲線に近い通路を実現する。すなわち、曲線の方程式が $F(x, y) = 0$ であるとする、検出された現在位置の座標 $(x, y)$ について $F(x, y)$ の値を計算し、その正負に応じてX軸或いはY軸方向のいずれかに単位距離だけ進む動作を繰り返すわけである。三次元物体の加工の場合には、もちろんZ軸方向の移動も必要であり、複雑な形状加工のためには、計算機的能力を持った情報処理装置が必要となる。また特に複雑な三次元形状の場合には、命令のプログラムも手計算では困難になるから、計算機を用いた自動プログラミングが必要になり、そのための専用言語(APT, EXAPT, SPLIT等)も開発されている。

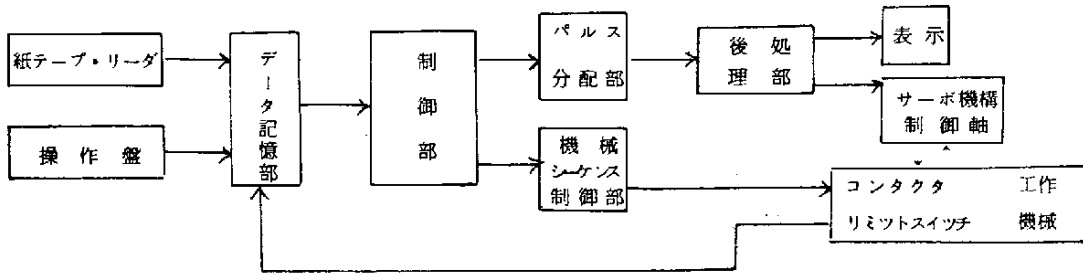
NCは数年前までは布線論理によるNCがほとんどであった。すなわち、ICやトランジスタをリード線で接続して特定の論理回路を組み、情報処理を行なうものである。しかし布線論理によるものは工作機械が変わるたびにハードウェアの設計変更をしなければならないから、各種機械への適用に際して柔軟性に欠ける。また同様の理由で量産性、保守性において問題を生じ易く、機能の高度化にも限界がある。そこで、小形の計算機を内蔵してプログラミングによって情報処理を行なわせるNC、すなわちコンピュータNC（CNC）が登場することになった。CNCでは紙テープ入力なしに繰返し作業が可能なこと、加工データの修正、変更が容易なこと、自動プログラミング機能を持たせ得ること、高度の補間処理ができること、逆プログラミングも容易なこと、など数々の長所が同時に生まれる。

今日多く用いられているCNCには汎用のミニコンピュータを用いたものと、NC専用で設計したミニコンピュータを用いたものがある。加工速度が比較的遅い機械（フライス盤、放電加工など）では汎用のミニコンピュータをそのまま適用できるが、加工速度が速い機械（旋盤など）では高速演算のできる専用コンピュータを用いるか、パルス分配を行なう補間回路を設けなければならない。そのためミニコンピュータを用いるNCは、機能、信頼性、柔軟性、汎用性の点で優れている反面、価格の点で不利となり易く、コスト・パフォーマンスを一段と改善することが望まれているわけである。

この問題を解決するためにマイクロプロセッサの利用が考えられる。適当な記憶素子と組合わせることにより、プログラム内蔵形の記憶、計算機能を持たせることができ、しかも小形、低価格という長所を生かすことができる。ただ制御プログラムの汎用性をあまり強調し過ぎると、低価格の長所が消え失せてしまうので、ある程度機能を限定した製品を目指すべきである。すなわち汎用性を強調する場合はミニコンピュータを用い、汎用性を限定する場合に応じて適当なマイクロコンピュータに置き換えていくべきである。マイクロコンピュータの場合でも、もちろん、あらかじめ準備された範囲内でプログラムの入れ換え、拡張は容易にできるので、布線論理方式のNCに比べて遙かに柔軟性は高い。

一般のCNCにおける情報処理機能をもう少し立ち入って調べてみよう。

第1-12図はNCシステムの構成を示したものである。



第1-12図、数値制御系の構成

紙テープ・リーダーはパート・プログラム、すなわち加工の形状、機械工具の動作を指令するプログラムを読み取り、操作盤からは工具の径、長さなどの補正データを入力する。データ前処理は各種のデータ（アブソリュート・データ、インクレメンタル・データ、パラメータだけ与えて円、傾斜線、格子などを指定するフィクスト・サイクル）の統一や補正データの処理を行なう。パルス分配は、工具が定められた通路に沿って移動するように、各座標軸（制御軸）方向の指令パルスを発生する機能で、NC情報処理の中心を成す。X、Y、Z軸の他、マシニング・センタなどではロータリー・テーブルの回転角も制御軸のうちに数える。サーボ機構としてはDCサーボ方式（クローズド・ループ或いはセミ・クローズド・ループ）、パルスモータ方式（オープン・ループ）などが代表的なものである。制御部はモードの切換え、割込み、インターロックなどを制御する。シーケンス制御の部分については第1.8節を参照されたい。

CPUに8ビットマイクロプロセッサを用いたNCの例では、RAMにパート・プログラムを収容し、ROMに設定移動単位、制御軸の座標補間（円、二次曲線）、指令パルス補正（バックラッシなどに対する）、制御の各プログラムの他、自己診断プログラムを内蔵したものがある。自己診断サイクルでNC内部の異常をチェックし、もし異常があれば警報を発したり、自動停止したりする。この自己診断機能を持たせることは、バックアップなどのハ



ードの増強なしにシステムの信頼性を高める手段として、最近の1つの傾向であろう。また工具の現在位置を入力することができるので、複雑な形状加工の場合、熟練者の加工動作をプレイバックする逆プログラミングも可能である。

概していえば、ボール盤、フライス盤、旋盤など、個別の工作機械についてはミニコンCNCよりもマイクロコンCNCの方が価格的に有利であり、より高度な機能を要するマシニング・センタなどでは、同程度かまたはミニコンCNCの方が優位である。1台の計算機で複数台の工作機械を管理・制御する群管理・制御方式(direct NC, DNC)では中央にミニコン・コンピューター、端末機械にマイクロプロセッサを用いて、情報処理に階層構造をもったシステムを組むことができる。

先に述べたように、複雑な形状加工の場合は自動プログラミングが必要である。そのための専用装置、すなわちNCテープ作成装置にもマイクロプロセッサを用いた製品が開発されている。加工の形状寸法、切削条件、切削速度など、加工物固有の情報をパート・プログラムとして入力すると、それらの情報から内蔵されたマスタ・プログラムによって工具通路の座標値を計算(コード化)する。さらに切削送り速度、主軸のオン、オフ、及び速度などの補助信号もコード化し、NC装置及び工作機械に固有の情報のポスト・プロセッシングも行なう。現在4ビットのマイクロプロセッサ、4KWのROMプログラム・メモリ、1240ビットのRAMデータ・メモリを備えたものなどがある。

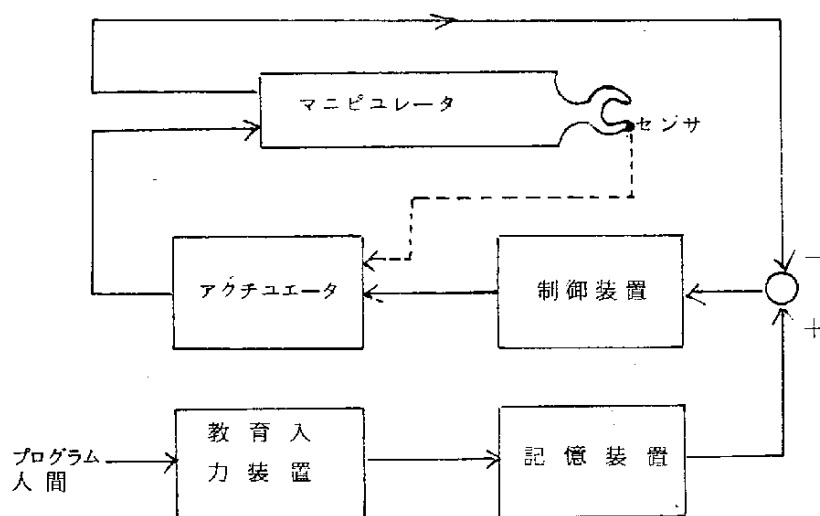
#### 1.4.2. ロボット

今日、ロボットの定義は工学的にも必ずしも明確ではないが、要するに、人間と同様な機能を備えた自動作業機械のことを言う。人間の機能は極めて複雑多様であって、それを的確に分類することは困難であるが、認識や判断などの知能に関するものと、移動性或いは運動性に関するものとに大別できよう。もちろん今日のロボットは、人間に比べて勘だ不十分な機能しか持っていない。特に知能は記憶以外の点では貧弱であり、運動能力としては手を動かして作業をし、足(車なども含む)によって移動することがどうやらで

きる程度である。しかし、単能の作業機械は普通ロボットと呼ばないので、ある程度の汎用性は持っている。

今日、研究開発ないし実用されているロボットは、工業用ロボットと知能ロボットに分類することができる。もう1つ遠隔操作ロボットを挙げることでもできよう。工業用ロボットは、普通記憶装置と1本の腕と手を備えており（足に相当する移動装置を備えたものもある）、記憶装置に読み込まれた指令に従って、比較的単純な作業を繰り返す。これは作業自動化のために広く実用されている。工場での生産工程には各種の専用自動機械が用いられているが、最近多品種少量生産の要求が多くなるにつれて、適応能力をもったロボットが登場してきたのである。作業動作は単純なものであるが、教育すなわち記憶内容の変更によって、1台のロボットで多種類の動作を行なわせることができる。

工業用ロボットの構成は、およそ第1-13図に示すようになっている。作業は手によ



第1-13 図工業用ロボットの構成

って行なわれるが、三次元空間にある物体を動かすには、X、Y、Z 3軸及び3方向の回転の6自由度が必要である。

さらに2本の指で物体をつかむ動作も加わる。

しかしあまり自由度を大きくすると、構造は複雑になり従って高価になるから、普通2～5自由度のものが使われている。腕の動きは、円筒座標形（バーサトラン形）と球面座標形（ユニメート形）があるが、前者は直観的にわかり易い代わりに、腕全体が移動するのでパワーが大きくなる欠点がある。後者はその逆である。手先には普通2本の指があり、手首もひねりや曲げなどができるようになっている。指先にセンサが付いているものは稀であるが、つかむ物体の種類によっては圧力、重量、硬さなどのセンサが必要である。アクチュエータには油圧または空気圧シリンダを用いることが多く、モーターを用いるものは少ない。これは油圧、空気圧式の方が出力対重量比において勝っているからである。油圧式は空気圧式に比べて、より重量の大きな物を扱う場合に用いる。

制御装置は記憶の内容に従って、アクチュエータを動かす制御指令を出す。自由度が2～4で、手の動きがせいぜい20点程度以下で表わせるような、簡単な作業をするロボットでは、あらかじめ作業プログラムを作成して入力するが、複雑な作業をするものではプレイバック方式を用いる。すなわち熟練した人間がロボットの腕、手をとって動かしながら、作業を教え込むのである。腕、手の位置はリゾルバまたはエンコーダにより検出され、制御信号と共に、磁気テープその他の記憶装置に記憶される。教育されたロボットが作業を行なうときには、腕、手の現位置が記憶された位置と比較され（比較または減算回路）、一致するように動作するわけである。記憶装置に磁気ディスクなどを用いると、多くの作業プログラムを同時に記憶でき、外部からの指令によって必要なプログラムを読み出すようにすれば、柔軟性、汎用性を増すことができる。なおプレイバックによるプログラミングを要しない簡単なロボットでは、記憶にピンボードを用いれば十分である。

上記のような記憶、制御を行なうには小形計算機、中でも要求される能力からみて、ROM・RAMと組み合わせたマイクロプロセッサが適している。開発例として、8ビットのCPUを用い、固定制御プログラ

ム及び固定データ用にROM 6 KW、一時的なデータ（プレイバック時の検出位置データなど）用にRAM 1 KW、それに若干のコア・メモリを併用した、自由度6のロボットなどがある。

以上に述べた工業用ロボットの他に、より人間に近い、ある程度の知能を持った知能ロボットがある。これは多自由度のマニピュレータの他、脳に相当する、やや大形の計算機をもち、視覚・聴覚・触覚といった感覚器に相当するセンサは、パターン計測やパターン認識の機能をも備えているロボットである。例えば視覚系は明るさ、色、距離の計測の他、形状パラメータ測定、形状認識、記号・文字認識などの能力を持つ。そのためにTVカメラ、光量計、光学距離計に加えてディジタル情報処理を行なうための高速AD変換器、小形コンピュータを備えなければならない。これらのパターン情報処理を行なうセンサ・システムは小形軽量でなければならないから、マイクロコンピュータの応用が極めて有望である。しかし今のところ、知能ロボットは実用の段階には至っておらず、マイクロプロセッサについても具体的な応用開発例はあまり報告されていない。

工作機械とロボット以外にも動きの制御は無数にある。例えば、各種の自動運転車輛装置（エレベータ、クレーン、コンベアなども含む）の計測情報処理システム、制御システムへのマイクロプロセッサの利用は現実のものとなりつつある。ここで強調しておきたいことは、自動化のために必要なのは動きそのものの制御よりも、環境条件を計測して動作指令を発する過程での情報処理であり、将来ますます重要な応用分野になるだろうということである。不連続部、不規則形状部の溶接を自動化するため、アークの発する光の画像を処理して溶接材の動きを制御するのに、マイクロプロセッサ（8ビット）を用いた例などは面白い。

## 1.5. 流量、圧力の制御

この節では、流れや圧力に関する現象を中心とした、プロセスの制御について考える。

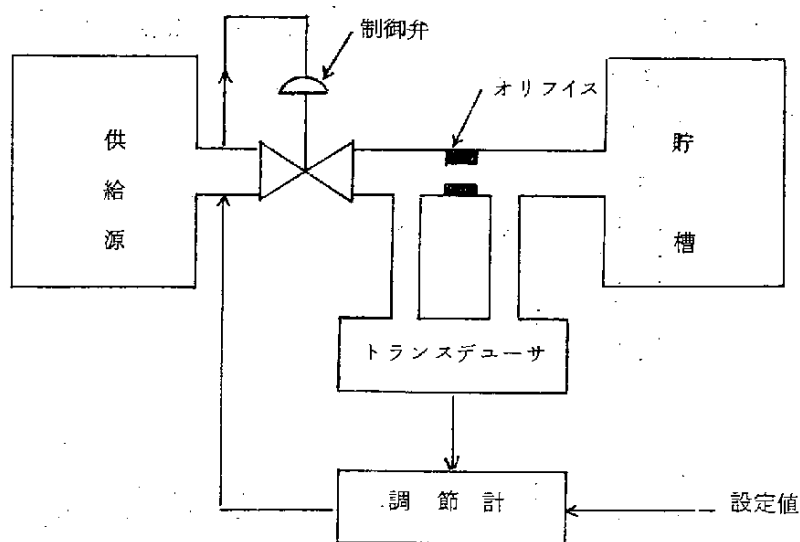
### 1.5.1. 流量の制御

流体の流れ系は供給源、パイプライン、タンク、弁、アクチュエータ

(ポンプ、コンプレッサ、ファンなど)から構成されている。流れ系の回路的なアナログを考えると、基本要素として抵抗、キャパシタンス、及びインダクタンスがある。抵抗はパイプラインの曲がり、接続、オリフィス、弁などでのエネルギー損失や、管壁での流体の粘性抵抗により生ずる。キャパシタンスは流体及び容器の圧縮、膨張に伴うエネルギーの保存に起因し、インダクタンスは流体運動の慣性に起因して現われる。

第1-14図は、流れ制御系の基本的な構成を示す。流れ系の動特性は速いから、プロセスの時定数は小さいとして無視できることが多い。弁、調節計(比例動作)、トランスデューサの時定数は、それぞれ1~3.0秒、0.05~0.1秒、0.1~0.5秒程度である。

供給源の圧力を $P_s$ 、受入タンクの圧力を $P_R$ とし、断面積 $A$ 、直径 $D$ 、等価長 $L$ のパイプでつなぐと、流量 $Q$ は



第1-14図 流れ制御系の構成

$$Q = A \sqrt{2D / f L \varrho} \sqrt{P_s - P_R} = AC \sqrt{P_s - P_R} \quad (24)$$

で与えられる。ただし、 $f$  は摩擦係数、 $\rho$  は流体の密度であり、 $C$  は流量係数と呼ばれる。パイプの途中に弁を設けたとしよう。(24)式と同様に、弁を通る流量は

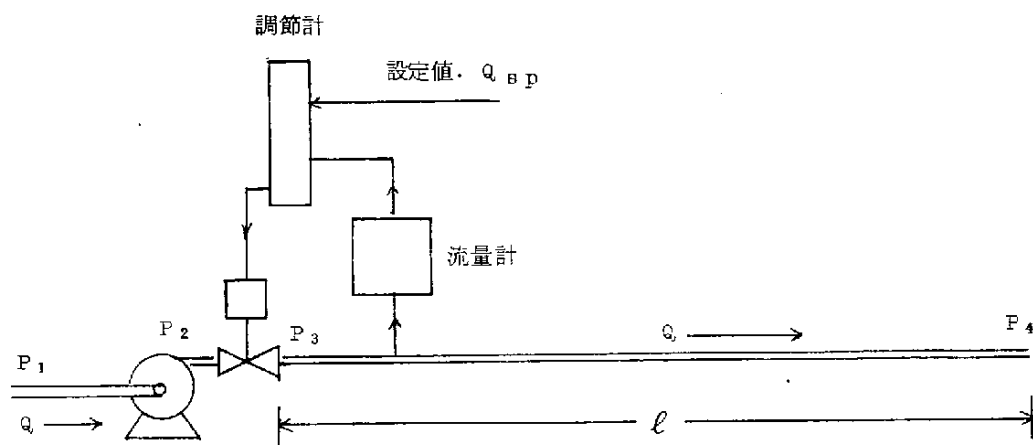
$$Q = A_v C_v \sqrt{P_o - P_R} \quad (25)$$

である。 $A_v$  は弁の開度の関数である。 $A$  が開度  $x$  に比例するとしたとき、流量を弁を全開したときの流量  $Q_m$  で正規化して、 $q = Q / Q_m$  で表わせば、(24)、(25)式より

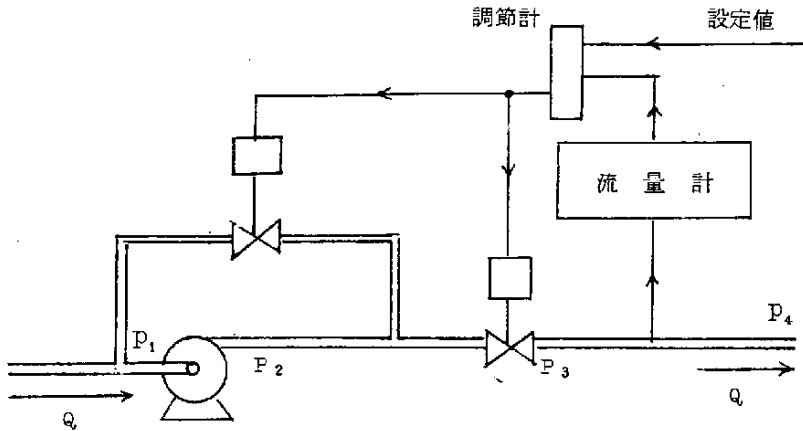
$$q = \frac{x}{\sqrt{r + (1+r)x^2}} \quad (26)$$

が得られる。ただし、 $r$  は最大流量  $Q_m$  のときの弁での圧損の全圧損に対する比である。(26)式は弁の開度を調整して流量を制御するときの基本式になる。 $r$  が制御特性を表わすパラメータになっており、 $r = 1$  (圧損はすべて弁で生ずる) ならば理想的な線形特性の弁であり、 $r$  が小さいとオン・オフ特性に近くなる。

第1-15及び16図に、直列弁を用いる制御系及びバイパス弁と直列弁を用いる制御系を示す。



第1-15図 直列弁を用いた流量制御系



第 1 - 1 6 図 バイパス弁と直列弁を用いた流量制御系

また流量制御は圧力制御と組み合わせて行なわれることも多い。第 1 - 1 7 図にその例を示した。

なお流量計測に用いられる電気式流量計は、およそ 4 種類に大別できる。すなわち

(1) 電気的二次変換器を用いる流量計。

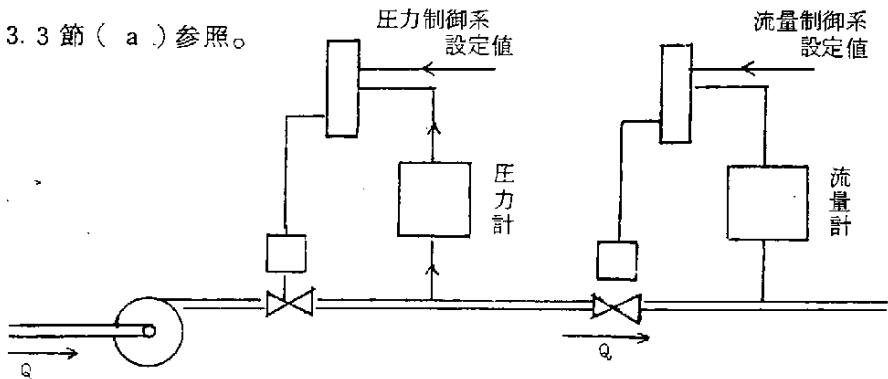
各種の機械式流量計に適当な二次変換器を組み合わせるものである。

(2) 電磁流量計。

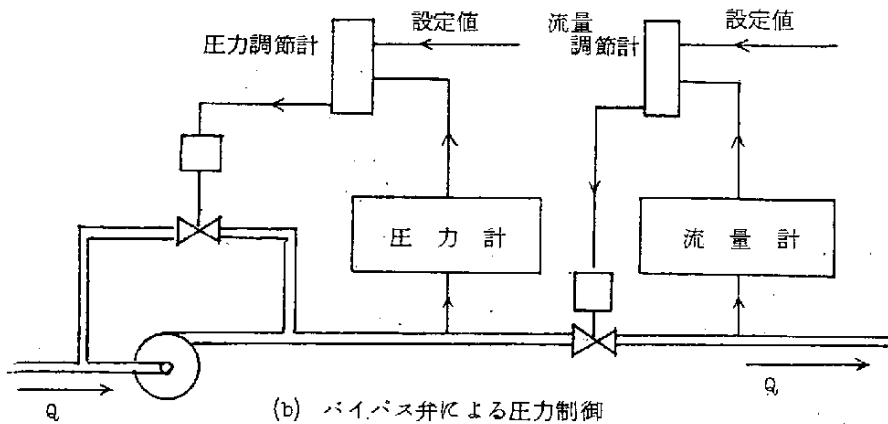
ファラデーの電磁誘導の法則を利用して、流量を直接電圧に変換する。入出力関係が線形で、応答も速い。

(3) 超音波流量計。

第 1 3.3 節 ( a ) 参照。



(a) 直列弁による圧力制御



第 1 - 1 7 図 圧力制御を併用した流量制御系

#### (4) 熱を二次量とする流量計。

流体中に置かれた加熱体からの熱放射が、流速の関数となることを利用する。

質量流量の計測制御系にマイクロプロセッサを用いた例がある。計量槽にはオン・オフ弁を通して、開弁のとき流量  $Q_1$  が流入し、一方ポンプによって流量  $Q_2$  が抜き出されている。目的は  $Q_2$  の質量流量を一定に制御しようということである。 $Q_1 > Q_2$  としてあるので、弁が開いているときは計量槽の質量  $W$  は増し、上限値  $W_H$  に達する。そのときオン・オフ弁を閉じると  $W$  は減少し始めるので、下限値  $W_L$  に達したら再び開弁する。 $W_H$  から  $W_L$  まで質量が変わる時間間隔を  $\Delta T$  とすれば、流出質量流量  $Q_2$  は

$$Q_2 = (W_H - W_L) / \Delta T \quad (27)$$

で与えられる。よって質量  $W_H$ 、 $W_L$  及び時間  $\Delta T$  を測定することにより  $Q_2$  を知ることができるわけである。 $\Delta T$  は 1 KHz のクロック・パルス数を計数して求めている。もし  $Q_2$  が設定値と異なれば、PI 制御により流出ポンプの回転を制御する。(27) 式の計算、流量表示、偏差計算、ポンプの回転数の増減量の計算のサイクルを、8 ビットのマイクロプロセッサ、512 バイトの PROM、50 バイトの RAM を用いて行



なっている。

#### 1.5.2. 圧力の制御

流体の圧力計としては、ブルドン管形、ダイヤフラム形、液柱形などがよく用いられている。ブルドン管形は構造が簡単で測定範囲は広いが精度は良くない。ダイヤフラム形は構造が簡単で感度も高いが、ダイヤフラムの変位が大きいときヒステリシスを生じ易い。液柱形は簡単で最も確実な測定ができるが、感度が悪いことがある。

プロセス用の空気供給源の圧力は6~7%であるが、それをコンプレッサを通じて空気槽につなぎ、槽内圧力を一定に保とうとするような場合、コンプレッサの電源をオン・オフ式に制御することが多い。

液体の圧力制御は第1.5.1節の最後に触れたように、流量制御の外乱をなくするための元圧制御として行なわれることが多い。蒸気圧の制御で最も一般的なものは、減圧弁を用いる方法である。すなわち高圧側の蒸気を減圧弁を通して低圧側へ流し、低圧側の圧力を検出して一定に保つ方式である。

#### 1.5.3. 液位の制御

プロセス用に使われる液位計には、フロート形、差圧形、変位形などがある。フロート形は最も簡単であるが誤差の少ないもの（滑車式）は開放タンクにしか使えず、密閉タンクにも使えるもの（回転指示式）は誤差が大きく測定範囲も狭い。差圧形は開放、密閉両タンクに使え、測定範囲も広い。さらに粘性の大きい液体や腐食性の強いものにも使えるが、ただタンクの内圧が極端に低い場合には使用できない。変圧形は応答が速く、測定範囲も広いが、精度は悪い。

制御は、弁により流入、流出量を調節して制御する。液位は流入量と流出量と流出量の再の時間的積分値に比例する。すなわち流出或いは流入が一定のときは、流入或は流出量に対し積分特性をもつ。応答性の悪い場合は、調節計（弁）は比例動作にしてゲインを高くし（P動作）、比較的応答性の良い場合は比例ゲインを小さくして、積分動作を加える（PI動作）。さらに応答性の良い場合は微分を加えることもある（PID動作）。流出側が自然流出の場合は液位は自己平衡性をもつから、制御はオン・オフ動作で十分である。

## 1.6. 温度プロセスの制御

### 1.6.1. 伝熱の法則

温度の制御の基礎となるのは伝熱の現象である。よく知られているように、伝熱は熱伝導、対流、及び熱放射によって行なわれる。熱伝導による熱流量  $q$  はフーリエの法則によって表わされる。一次元の場合は

$$q = -K_d A \frac{dT}{dx} \quad (28)$$

である。ただし、 $K_d$  は熱伝導率、 $A$  は伝導方向に直角な断面積、 $dT/dx$  は温度勾配である。流体と固体の間の伝熱は対流によってなされる。対流による伝熱量  $q$  はニュートンの法則により、

$$q = K_v A (T_1 - T_2) \quad (29)$$

で表わされる。 $K_v$  は対流熱伝達係数と呼ばれる。

熱放射に関してはシュテファン・ボルツマンの法則があり、放熱側、受熱側の絶対温度を  $T_1$ 、 $T_2$  とすれば伝熱量は

$$q = \sigma A \varepsilon (T_1^4 - T_2^4) \quad (30)$$

である。ただし  $\sigma$  はシュテファン・ホルツマン定数、 $\varepsilon$  は放射率である。

熱系では温度変化に対する伝熱量の変化率、すなわち  $dq/dT$  が熱抵抗であり、熱容量がキャパシタンスに相当する。しかしインダクタンスに相当するものは無い。(従って熱系だけでは振動現象は起こらない。)

### 1.6.2. 温度の計測器

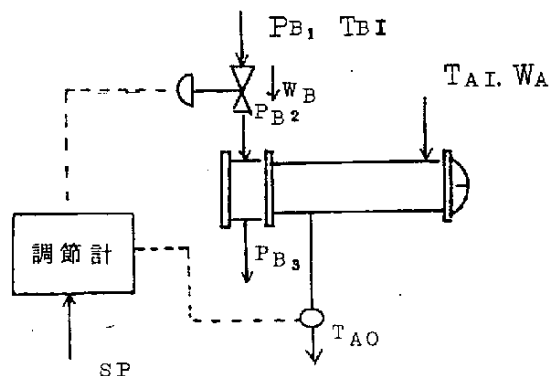
温度計には接触式の温度計と非接触式(放射式)の温度計とがある。接触式のもは測定対象の真温度を直接に測定できるが、対象の熱状態を乱す恐れがある。それに対し非接触式のもは測定対象が黒体でない限り真温度は測定できず、放射温度や輝度温度しか測定できない。しかし遠隔測定、移動体の連続測定ができるのが大きな強味である。

接触式の代表的なものは気体、液体、固体の膨張式、熱電対式(ゼーベック効果)、電気抵抗式、半導体式(サーミスタ)などがある。非接触式は一般に測定温度が高く、放射高温計、光高温計、光電管高温計、色高温計があるが、このうち制御系によく用いられるのは放射高温計である。これは対象の放射エネルギーをレンズで集め、サーモパイルによ

り測定する

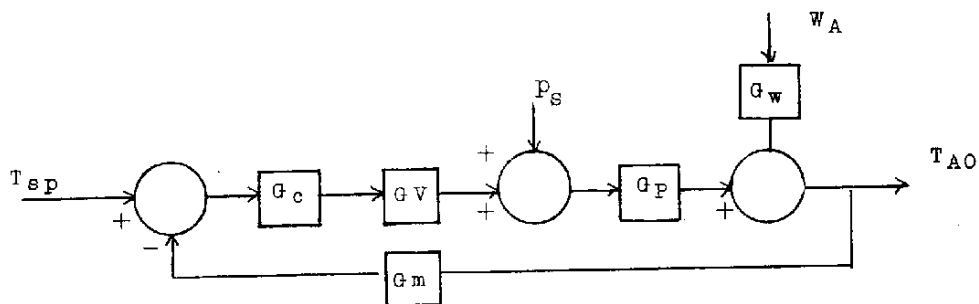
### 1.6.3. 熱交換器の制御

ここでは代表例として、熱交換器における温度制御を見てみよう。第1-18図の熱交換器の温度制御系について考えてみる。



第1-18図 熱交換器温度制御系

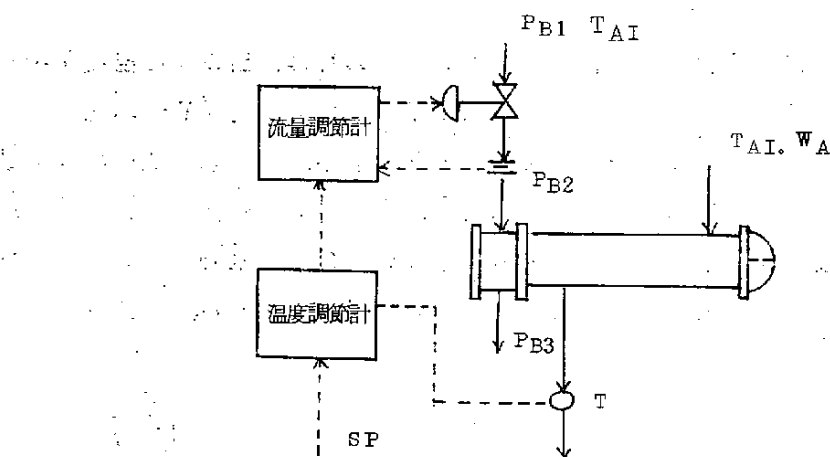
熱媒Bの流量を調節して、プロセス流体Aの交換器出口での温度 $T_{AO}$ を制御するのが目的である。入口温度 $T_{AI}$ ・ $T_{BI}$ ・流量 $W_A$ ・熱媒の圧損失 $\Delta P_B = P_{B1} - P_{B3}$ などには変動があるとする。ただし一般に温度変動はゆっくりしているから、主として $W_A$ と $\Delta P_B$ とに注目すればよい。熱・温度系では計測の遅れも無視できない。従って制御系のブロック線図は第1-19図のようになる。ただし、 $G_c$ ・ $G_v$ ・ $G_m$ 、及び $G_P$ はそ



第1-19図 熱交換器温度制御系のブロック線図

れぞれ調節計、弁、計測器、及びプロセスの伝達関数である。PI 調節計を用いた制御系の設計のためには、 $G_V$ 、 $G_m$ 、 $G_P$  がわかっておかれ  
ばよい。このうち  $G_V$  と  $G_m$  は 1 次遅れとみなしてよい。 $G_P$  も少なく  
とも 1 次遅れであり、従って閉ループの特性は少なくとも 3 次である。よ  
って調節計の性能は安定性で決められてしまう。しかも  $G_P$  の時定数は  
最も大きいので、それと次に大きい時定数の比でループ・ゲインを決め  
ると（安定余裕をみて）、結局弁と計測器によって調節計の性能が決ま  
ることになる。 $G_P$  のゲインは操作レベルで変わるので、プロセス・ゲ  
インが最大ゲインとなる操作レベルで調節計のチューニングを行なうが、  
プロセス・ゲインが低下したときには応答性は悪くなる。オン・ライン  
でプロセス特性を測定し、チューニングの条件を変えることは、マイク  
ロコンピュータなどを用いて可能になるであろう。

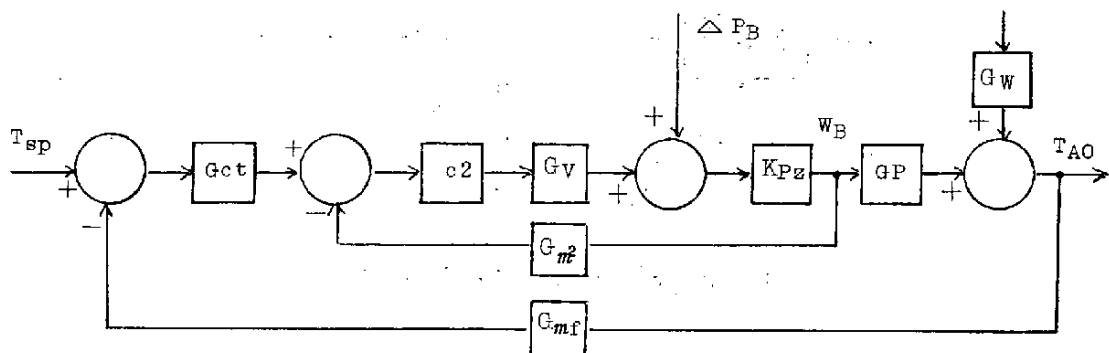
圧力差  $\Delta P_B$  の変動に対して第 1-18 図の系が不満足な場合は、第  
1-20 図のようなカスケード制御が行なわれる。すなわち温度調節計



第 1-20 図 カスケード温度制御系

の出力で流量制御ループの設定点を再調整し、熱媒体の圧力が変動して  
も温度制御ループに必要な流量を維持するわけである。そのブロック線

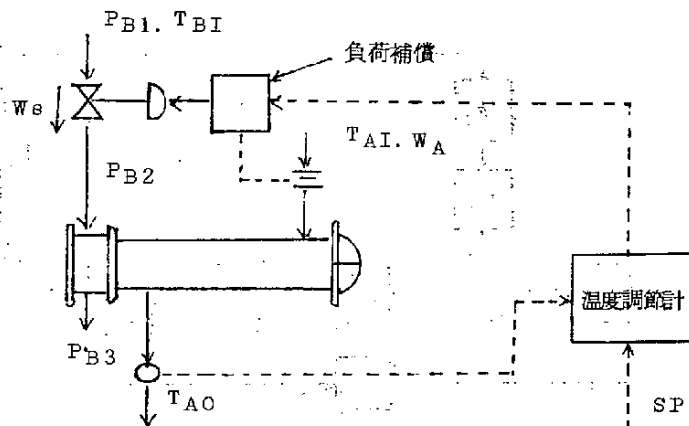
図は第1-21図となる。ただし、 $G_{c1}$ は温度調節計、 $G_{c2}$ は流量調節計、 $G_{m1}$ は温度計測器、 $G_{m2}$ は流量計測器、 $K_{P2}$ は流量プロセスのゲイン（時定数は温度プロセスに比べて小さく、無視できる）である。



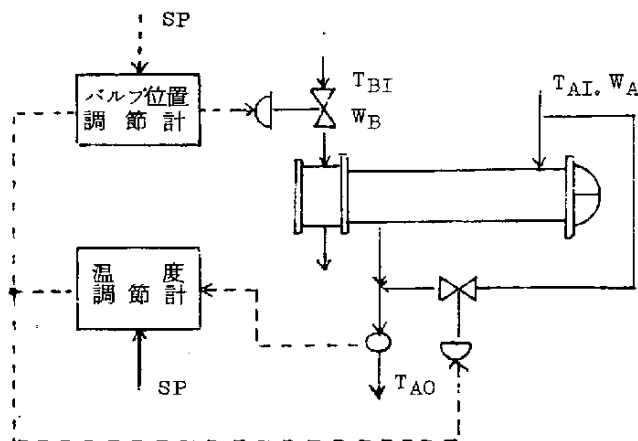
第1-21図 カスケード制御系のブロック線図

流量ループの応答は速いから、第1-18図の系に比べて、温度ループは圧力変動に対してより安定であるばかりでなく、より速く、大きなループ・ゲインを持つことになる。

流量 $W_A$ の変動が激しい場合には第1-22図のフィードフォワード制御系または第1-23図のバイパス制御系を用いる。フィード

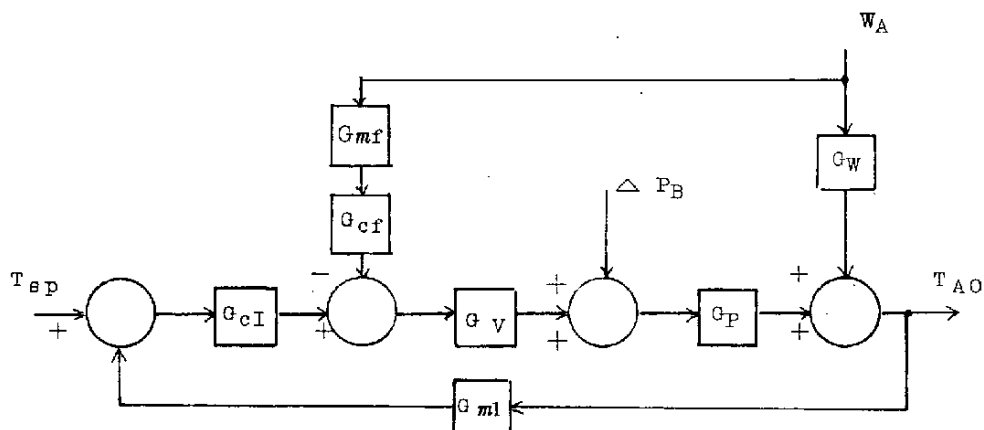


第1-22図 フィードフォワード温度制御系



第 1 - 2 3 図 バイパス温度制御系

バック制御系では、出口温度  $T_{AO}$  の変化が現われるまで調節計は作動しないが、フィードフォワードでは  $W_A$  の乱れを検出したら直ちに働き、乱れの  $T_{AO}$  に対する影響を打ち消すように  $W_A$  を調整するわけである。ブロック線図は第 1 - 2 4 図のようになる。ただし、 $G_{cf}$  はフィード補償器、 $G_{mf}$  は流量計測器である。



第 1 - 2 4 図 フィードフォワード制御系のブロック線図

## 1.7 電力系統における制御

電力事業はエネルギー産業において最も重要な位置にあり、従つて最も重要な基幹産業の一つである。電力事業の目指すところは、良質で安定した電力を安価に供給することであるから、発送配電機器の経済的な運転、電力系統の効率的で安定した運用から需要家へのサービスに至るまで、広い範囲の作業の自動化、省力化が必要であり、その中心を成すものは計算機による情報処理及び制御である。

計算機の応用分野は発電、送変給電、配電、及び事務関係に分けることができるが、各分野における主な応用は第1-8表に示すようになるであろう。原子力発電では火力発電における機能以外にも要求されるものがあるので、両者を分けて示した。

### 1.7.1. 火力発電所における制御

火力発電所は水を加熱し蒸気を作るボイラ、ボイラでつくられた蒸気を機械エネルギーに変換する蒸気タービン、タービンの機械エネルギーを電力に変換する発電機から成り、さらに電力を送電系統に送り出すための昇圧変圧器が付随している。

計算機が発電所に導入された当時の目的は、データ・ロギングのためであつた。ボイラの水流量、流体圧力、温度、タービンの圧力、温度、復水器の温度、水量、発電機の電圧、電流、周波数、力率、出力、発電電力量、無効電力量などを、測定、計算し記録して日誌を作成する。計算の中には出力の1日の積算値、平均値、最大値、最小値を求めるものなどもある。計算機はボイラ、タービン、発電機の運転状態を計測・監視し、異常を検出した時には中央制御室に表示し、警報を発する、いわゆるアラーム・モニタとしても使われる。さらにボイラ効率、タービン効率、発電効率などの性能計算、記録も引き受けている。

以上はデータ・ロギング関係であるが、制御には中央制御室から発電プラントに対して行なう自動プラント制御（APC）、自動ボイラ制御（ABC）、自動バーナ制御（ABNC）、自動タービン速度制御（ATSC）自動電圧調整（AVR）、自動同期調整（ASS）などがあり、その他特に起動停止時には各種弁類のシーケンス制御が重要である。発電プラント

を450～600MWのユニットに分けた場合、それを制御する計算機は、30 KW程度のコア・メモリ、150 KWの補助メモリを持つたミニコンピュータで十分である。信頼性向上のためにAPC、ATSC のような重要な制御にはデュアル構成をとっている例がある。

|       |                                                                                    |
|-------|------------------------------------------------------------------------------------|
| 火力発電  | データ記録・処理（日報、月報、トレンド記録、経過値記録、流量、温度、圧力等の計算、ブランド効率の計算、発電計画など）<br>プラント起動停止制御、タービン制御など。 |
| 原子力発電 | 炉特性のオン・ライン計測・計算（中性子束、放射能、出力分布、熱出力密度など）、炉の起動、出力制御、シミュレーションなど                        |
| 送変給電  | 有効電力制御、電圧無効電力制御、信頼度制御、運用データ処理、<br>送変給電所間のデータ伝送、無人運転など                              |
| 配電    | 無人検針、需要家情報サービスなど                                                                   |
| 事務    | 統計処理、料金計算処理、給与計算など                                                                 |

オ 1 - 8 表 電力事業における計算機の応用分野

### 1.7.2. 系統における制御

系統の中には発電所の他、中央給電指令所、変電所があり、それぞれにデータ処理・集録、計画、制御を行なっている。中央給電指令所は給電業務の中核であり、そこでは系統運用データ処理、発電計画、有効電力制御（AFC）、電圧無効電力制御、信頼度制御などを行なわなければならない。このうち、有効電力制御は系統の周波数を一定値に保ちながら経済的な運転をすることが目的であり、そのために発電機の出力制御



を行なう。有効電力制御では自動周波数制御 (AFC) と経済負荷配分 (ELD) を併用して、数分に 1 回の割合で ELD 計算を行ない、その結果経済的な出力値を求めて AFC を行なう。デジタル AFC では系統の状態によつて制御パラメータを変更する適応制御、負荷急変時の先行指令制御等、柔軟な制御が行なえる。コア・メモリを必要とするが、命令 0.5 kW、データ 1.5 kW 程度でよいとされている。バックアップにはアナログ AFC によるものとデュアル・デジタル構成にするものがある。

電圧無効電力制御は、電圧を適正値に保つて電力品質の向上をはかると共に、送電損失の軽減、他会社との連続線等の無効電力潮流を制御するために行なわれる。系統の接続状態、監視点の電圧、無効潮流などのデータを収集し、インピーダンス行列に基づいて電圧と無効電力に関する系統定数を計算する。電圧、無効電力が許容範囲にないときは、送電損失が最小になるように、各調整機器の電圧及び位相調整量を計算する、という手順が繰り返される。

信頼度制御は正常、警戒の各状態において、事故の波及拡大を防ぐための発電機出力の調整、負荷しゃ断などを行ない、回復状態では系統分離をなるべく速く正常にもどすなどの作業を行なう。

中央給電指令所は各発電所や、地方給電所があるときはそれとの間のデータ伝送を絶えず行なう必要があり、伝送路は時分割方式で利用される。

変電所の計算機は事故時の変圧器や線路のしゃ断、回復時の並入、正常時の母線停止及び復旧、系統切換えなどの自動操作をつかさどり、事故の記録をも行なう。

以上のような情報処理・記録、制御操作は全体としては大きなシステムであるが、その端末的な制御所の小容量自動監視記録装置や需要家の受変電設備のデータ・ロが、テレメータリングの伝送制御等にマイクロプロセッサの応用が始まっている。将来集中制御テレメータ方式が増すにつれ、末端での各種制御とデータ伝送にマイクロプロセッサの進出が目立つようになるであろう。

## 1.8 シーケンス制御と計算機

一連の制御操作の各段階を、あらかじめ定められた順序に従って進めていく制御が、シーケンス制御である。これには前段の動作が終つたことを確認して一定時間後に次段に移るとか、前段の制御結果に応じて次の操作を選定するとかの判断の機能、すなわちシーケンス論理の機能が中心となる。具体的には順序制御、時限制御、及び条件制御の3要素が主構成要素である。

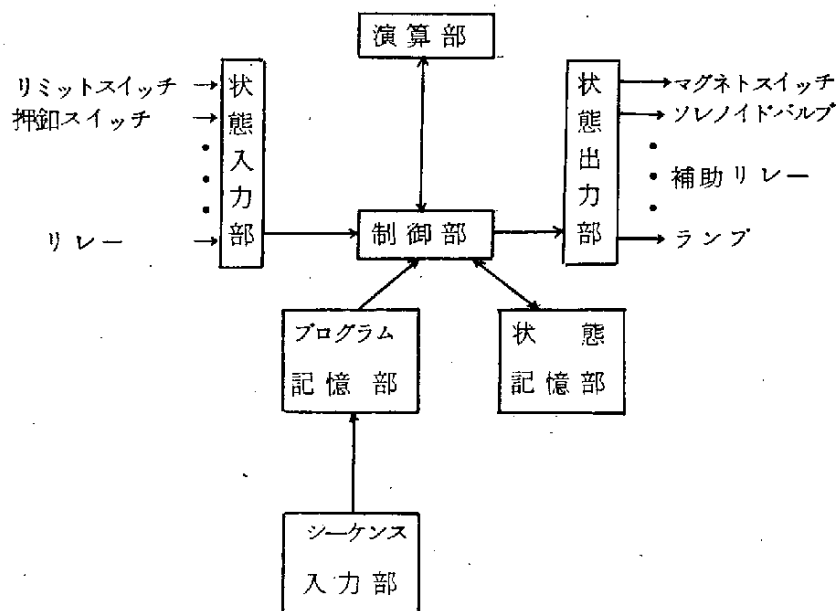
シーケンス制御装置が盛んに用いられるようになったのは、1950年代になつてからであるが、回転軸（ドラム）に取り付けたカムによつて弁を開閉する純機械方式、油圧、空気圧方式などの非電氣的な方式から始まつた。一方、電磁リレーは簡単で優れた論理素子として、長年シーケンス制御において代表的な位置を占めてきた。しかし入出力点数の増加、シーケンスの段数の増加と論理回路の複雑化につれて、主として信頼性の点で限界が生じ、半導体素子の進歩と共に無接点リレーによつて置き換えられるようになる。無接点リレーはダイオード、トランジスタ、ICなどを組み合わせてプリント板に取り付け、論理素子としてユニット化したものである。増幅・反転・比較、時限動作素子の他、フリップ・フロップ、シフトレジスタなどの素子も作られた。

制御対象の大形化、多様化につれリレーを用いた布線論理の設計、製作には膨大な労力を要するようになり、また制御内容の変更が直ちにハードウェアの変更につながるので、生産性が低くなる。1968年頃、アメリカのGM社がトランスファ・マシンの制御用に、プログラマブルであり、保守・修理が容易であり、電磁リレーより信頼性が高く小形であり、基本ユニットが拡張可能であるような装置を求めた。それに対応してメーカーが提供することになつたのが、今日のプログラム内部記憶方式のシーケンス・コントローラ（SC）である。今のところ、シーケンサ、プログラマブル・コントローラ（PC）、プログラマブル・ロジック・コントローラ（PLC）、プログラマブル・ロジック・シーケンス・コントローラ（PLSC）など、呼び方はいろいろあり、統一されていない。ただSCはプログラムの設定にカム・ドラム、ピンボード（ダイオード・マトリクス）、テープを用い

るものも含むのに対し、PLC(或いはPC)は専用或いは汎用の計算機を内蔵する方式のものを指す。

カム・ドラム方式はシーケンス段数20、出力点数50程度までの順序制御に適する簡単なものである。ピンボード方式はダイオード・ゲートにより論理回路を構成し、マトリクス状に配置された孔にダイオード・ピンなどの短絡片を挿入して、プログラムを設定する。プログラムの変更や機能の拡張が容易で、比較的低価格なのが特徴である。

PLCではおよそ1-25図のような構成となる。ICやダイオードで専用計算機としての回路を構成し、ROM及びRAMを用いてプログラム記憶及び状態記憶を行なわせる。プログラミングにはアセンブラも用いられるが、論理演算、論理代数などで表現されたキー・ボードをそなえて

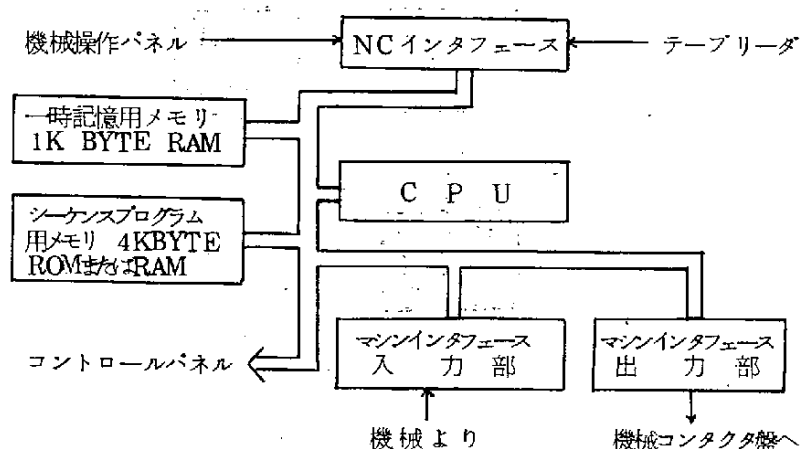


才 1-25 図 PLCの構成

いるものが多い。ハードに汎用のミニコンピュータを使用し、ソフトにはアセンブラを用いる方式は最も柔軟で高度の機能を持ち、記憶容量も大きい。また DDC、データ・ロガ、通信回線と結合することも容易であるが

一面、機能過剰になり易く、プログラミングがやつかいな他、耐環境性の点で問題を生じる恐れもある。従つて、導入にあつてはコスト・パフォーマンスを十分考慮する必要がある。一般に入出力点数及び有接点リレー換算個数が少ない方から多い方に移るにつれて、カム・ドラム方式、ピンボード方式、専用PLC、汎用ミニコンピュータPLC が順に適しており、またこの順に（必要記憶容量／入出力点数）に対する価格の上昇率が低い。

ミニコンピュータの代りにマイクロプロセッサを用いれば、汎用性の点では劣るが、小形軽量、低価格の利点を得られる。専用PLC の一例として、工作機械のシーケンス制御にマイクロプロセッサを用いたものがある。才4第に述べたように、工作機械の制御は工具の工作物に対する相対運動の数値制御（NC）と、工作順序のシーケンス制御（SC、制御モードの選定、主軸の回転速度の制御、工具の選定ATCなども含まれる）とより成る。後者のSCをつかさどるのが才1-26図のマイクロシーケンサである。NCインタフェースはテープ入力処理回路を通じてシーケンス・プログラム入力を受け、また手動操作盤よりの入力を受け取ることもできる。マシン・インタフェースはコンタクタ盤を通じて機械へシーケンス命



才1-26図 マイクロPLC

令を伝え、またリミット・スイッチを介して機械より入力を受ける。初期シーケンス・プログラムはRAMに書き込まれ、制御盤により内容のチェックや修正ができるようになっている。デバッグ済みのプログラムはPROMに固定する。マイクロプロセッサはIntel 8080（サイクル・タイム2 $\mu$ s）である。78種のバイト処理命令と17種のビット処理命令をもっており、ビット処理命令によつて3バイトのプログラムでANDやORの論理演算が可能である。

マイクロシーケンサは、ICを用いたPLCに比べて現在のところ、価格、速度の点で必ずしも優位ではないが、小形化、ハード設計のソフト化による工期短縮などの点で有利である。

## 1.9 マンマシン・インターフェース

制御系、情報処理系の端末装置は、人間とそれらの機械系の間の情報のやりとり、会話の窓口であり、マンマシン・インターフェースと呼ばれる。計算機の応用分野が広がり、多様化するにつれて、システムを構成する機器をなるべく標準化し凡用性を持たせることが必要となる。そこで一方特殊性に対応するために、端末に特殊機能を持たせる必要が生じてくる。各様の特殊性に柔軟に対応するためには、布線論理よりもソフトウェア的手法が適しているから、マイクロコンピュータなど小形計算機が活躍する場が生まれるわけである。マイクロコンピュータはかなり多量の複雑な情報処理ができるから、その小形軽量、少消費電力、信頼性、耐環境性と相まって端末装置向きなのである、処理速度が遅いことも、人間相手の仕事ではあまり大きな欠点とはならない。

ミニコンピュータやマイクロコンピュータを内蔵したプログラマブル端末は、一般にインテリジェント端末、或いはスマート端末と呼ばれている。この節ではインテリジェント端末とはどのようなものであり、そこにマイクロコンピュータなどがどのように活躍し得るかを見てみよう。

### 1.9.1. インテリジェント端末

端末装置には種々の型のもがあるが、印字装置、すなわちプリンタが現在最も広く使われている。プリンタで最も簡単なものは計算機からの出力を印字するだけのものであり、次にキーボードを備えてそこから文字入力ができるものがある。さらに制御端末と言われる、文字入出力以外に簡単な制御信号、例えば入力可能、入力要求、入力取消し、入力誤りなどを、表示ランプと押釦でやりとりできるものがあり、次に計算機と通信伝送路（電話回線など）を介して接続される通信端末がある。通信伝送路では、普通1ビットずつ信号が送られるから、文字情報や制御信号を直列に符号化して送るための、伝送制御手順を決定する回路が必要である（第1.3.4.節参照）。通信端末での情報のある程度まとめて転送し、伝送路の使用効率を上げるためバッファ・メモリを設けることもよく行なわれる。

通信端末はある程度のインテリジェンスを持つわけであるが、さらに

端末における処理機能を高めることが、マンとマシンの両面から要求される。すなわち

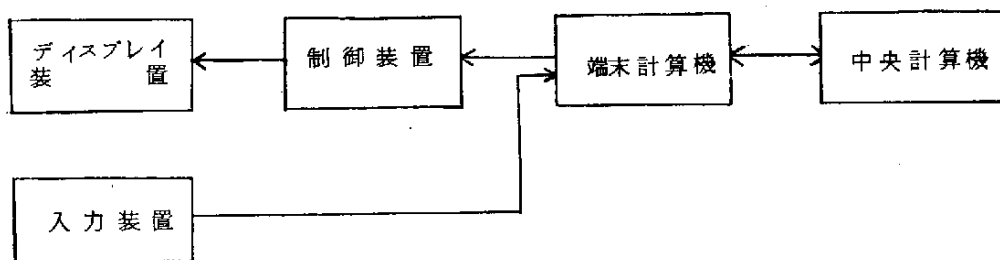
- (1) マン（使用者）はなるべく操作手順を簡単にすること、すなわち操作性を高めることを要求する。そうすることによってアクセスし易くすると共に使用効率を高め誤りを少なくする。
- (2) フロントエンドにおけるデータ処理をふやすことにより、中央計算機の負担を少なくする。これは不必要な情報を抑えて、伝送路の使用効率を高めることにもなる。

### 1.9.2. インテリジェント・ディスプレイ

CRT ディスプレイはマンが直感的にとらえ易く、また高速処理可能な端末として、近來ますます重要性を増しつつある。CRT ディスプレイは平面上における輝点のX、Y座標と輝度（Z座標）の3つの制御自由度を持っている（カラーならばもう1つ加わる）。

インテリジェント・ディスプレイでは、中央計算機とディスプレイの間に第1-27図のように小形端末計算機と表示制御装置を介在させ、各装置で次のように機能を分担する。

- (1) 中央計算機—多量、複雑な処理。またいくつかの端末間の時分割処理。



第1-27図 インテリジェント・ディスプレイ

- (2) 端末計算機—簡単な計算と図形処理。

図形の伸縮、移動、点の補間、部分消去など。マンからの入力に対して

は中央処理の必要なものに限って中央へ転送する

### (3) 制御装置—反復表示、計時など。

インテリジェンスが増すと一般の人が気軽に使える反面、誤った操作をする機会も多くなるであろう、そのために次に行なうべき操作の指示誤操作の場合、自動的に或いは使用者に指令して、正常状態に回復させる機能が必要である。

現在発表されている製品には、CPU に8ビットまたは12ビットのマイクロプロセッサ、プログラム用に8 KW程度のROM、データ用に1~8 KWのRAMを備えたものがある。RAMは入出力用バッファ、プログラム用スタック、及び伝送用バッファとして用い、DMA可能である。CPUの機能として、メモリの制御、伝送手順制御、データ制御(チェックや編集)、ジョブ・シーケンスの管理、入出力機器制御(割込管理など)がある。その他ハードウェア各部を診断する機能をもつかさとり、テスト・キーを押せば結果がCRT上に表示されることになっている。入出力機器としてはキーボードの他、ライト・ペン、小形ディスク、ハードコピー・プリンタなどが使えるようになっている。マイクロプロセッサ・モジュールの他、表示、データ伝送インターフェース、メモリなど、いくつかの標準モジュールを作っておけば、保守、診断、及び機能拡張が容易になる。

ここでは凡用のディスプレイについて述べたが、各種の専用端末装置例えば医療、生産管理、テレメータリング、地域情報、POS、銀行、証券、保険等々に、マイクロプロセッサの応用は急速に広まるであろう。

## 1.10 おわりに

計算機の処理機能というSEEDSと制御機能というNEEDSとの絡み合い、その時代的変遷・発展の経緯を念頭におきながら、膨大な範囲に拡がった現代の制御技術を、とにもかくにも一望してみた。一望はあくまでも一望であって、精細な踏査は筆者の力量と紙数不足のために、到底なし得ぬところである。マイクロプロセッサのより具体的な応用については、第2~4章を参照して頂きたい。ところで本章各節の記述を一読して頂けれ



ば、制御という機能、技術は決してそれだけで独立して存在し得るものでなく、制御対象の状態の検出・測定に始まり、測定データを誤りなく伝送し、受け取ったデータの中から適当な情報を抽出して最後にその結果と目標値を比較しながら制御操作を行なうという、一連の操作系列の中で初めて意味をもつものである。すなわち、正しい情報の取得、処理の上に立脚してのみ成功し得るのである。従って制御システムで計算機の果たす役割も、単にDDC的なものを考えるだけでは極めて不十分であることを銘記しなければならない。

マイクロコンピュータの利用は上に述べた広い意味の制御システムの技術に確かに大きな変革をもたらしつつあり、また将来さらにそのインパクトははっきりと現われてくるに違いない。それはマイクロコンピュータは一つのコンピュータであると共に、一つの機能部品だからである。コンピュータであるが故にソフトウェアによる情報処理が可能であり、情報記憶が可能であり、従って従来のハード的制御機器の持ち得なかったような多くの機能を持ち得る。例えば簡単に制御操作を変更し拡張し得る柔軟性であり、自己診断機能を持たせ得る利点である。しかしマイクロコンピュータ或いはマイクロプロセッサを汎用性・互換性のある部品として認め、標準化の徹底が実現する日がきたとき、その真価が発揮されることになるであろう。コストが低減すると共に、大きなシステムのコンピュータ・ヒエラルキの随所に自由にあてはめ、制御機能を分散して局部化し、また自由に取り換えることによって保守は極めて楽なものになる。こういった意味でコンピュータとしての認識を深め、部品としての標準化をより一層推進しなければならないと考える。しかし今のところ、マイクロコンピュータに過大の期待をかけ、ぜいたくな機能を持たせようとすると、失敗のものになるだろう。より上位の、例えばミニコンピュータとの接点を見極めて、柄に合った使い方をしなければならない、一つのマイクロコンピュータに大きな機能を持たせるよりも、それを分散して使い得るという利点の方を積極的に利用すべきである。最近我国でもマイクロDDCを含んだ総合プロセス制御装置が各社から発表されているが、これらは従来のミニコンDDCの機能を分散してマイクロDDCとし、多様

な仕様に柔軟に対応すると共に、一部の故障によるシステム全体のダウを防ぐようにしているのは、正しい使い方である。またコスト高になるデュアル構成を極力避けて、その代わり故障発見や自己診断を容易にするようにしているのも新しい行き方である。

本文では触れなかったが、将来大量の応用が期待される分野に一般家庭の中に入り込む機器類がある。例えば家庭での地域情報端末、教育用機器、ゲーム用機器などであり、さらに自動車の中に組み込まれる制御用（エンジンの点火タイミング、排気調整、室内灯調整、部分的な自動操縦など）やモニタ用の機器である。マイクロプロセッサ開発の動機が電卓用LSIの要求であったことを思えば、当然ともいえる分野である。一昔前、中学生が真空管その他の回路部品を買い集めてラジオを組み立てたように、CPUやメモリ部品のキットを求めて子供達がマイ・コンピューターを作るようにするのも、教育過程のも、教育過程の一環として有益、有用ではなかろうか。

## 第2章 制御技術への新しい発想

### 2.1. はじめに

#### 2.1.1. マイクロコンピュータのインパクト

戦後30年をふりかえって、今更のように驚嘆させられるのは、各界における技術革新のめざましさである。なかんづく、エレクトロニックスの進歩はその白眉であって、この進歩と共に制御技術はこれを栄養として発展を遂げ、新しいコンポーネントの開発、その制御への応用といったサイクルが絶えず繰り返されて現在に至っている。

その間、制御技術者は、新技術の習得、応用製品の開発に正に浮き身をやつしていたわけで、一時の油断も許されなかったのではないと思われる。

そして、ここへ来て今度は、「マイクロコンピュータ」などという厄介な代物が登場し、従来の技術や手法の延長だけでは、手に負えそうもない制御用コンポーネントが登場してきた。

マイクロコンピュータを制御用のコンポーネントと認めた時、制御技術者はそのインパクトが極めて深刻であることに思いが及ぶであろう。そして、そこにはいろいろな意味で新しい発想と、従来の制御装置の概念の転換とを余儀なくされることに気がつくであろう。

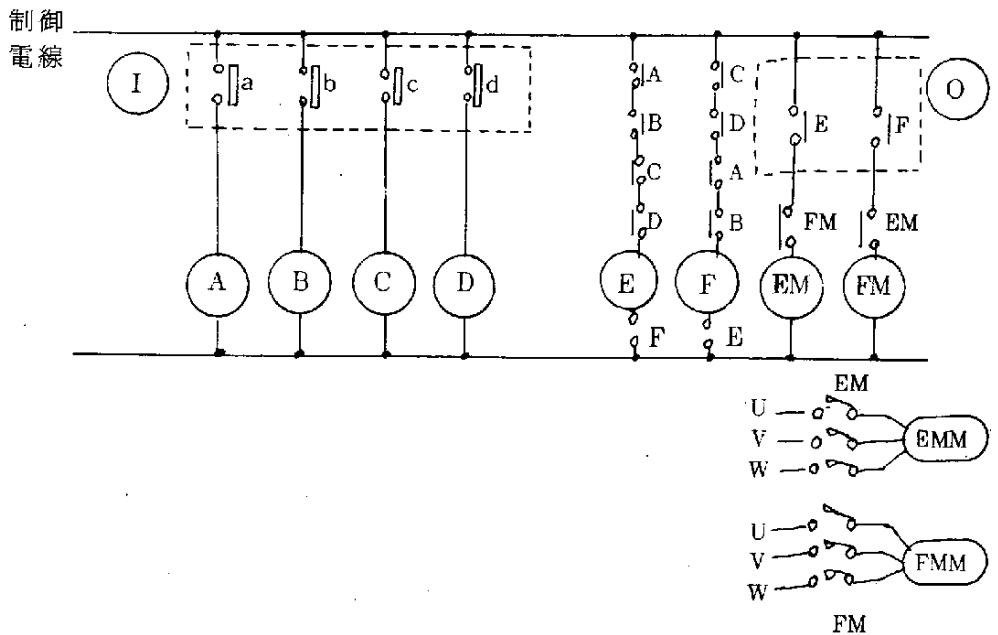
昭和20年代後半から30年代はじめにかけて「磁気増巾器」が制御装置のコンポーネントの花形であった時代があった。電子管の制御用コンポーネントとしての「たよりなさ」と、電子技術者（電子・通信関係の技術者を除く）にとって「なじみのうすさ」を補う意味で、好感がもたれたコンポーネントであるが、その後トランジスタ、サイリスタ等の半導体の素子が、制御用コンポーネントとして登場し、新旧選手交替してしまったことは周知のようである。

昭和39年後半から40年はじめにかけてICが開発実用化された。この事実は、将来のコンポーネントの方向が、「機能部品化」「複合化」と言った道を辿ることを示唆した意味で、一つの画期的な出来ごとであったといえる。開発当時のICはせいぜい数個の集積でしかなかったが、

それから3年ぐらいの間隔で集積度が一桁ぐらい高まり、MSI、LSIと飛躍的に機能の高級化が行なわれ、制御装置の性能-コスト比も大きく進歩し続けてきたといえる。

しかし、これら今までのコンポーネントを使って制御装置を設計製作する技術的な手法の中に、或る共通点がある。勿論、制御装置を設計するためには、一般的な意味で、制御の目的、制御対象の特性、要求される品質、かけ得るコスト等々のような事項は、当然どんなコンポーネントが主体となろうと共通点ではある。更に大事な共通点がある。先ず、上記のような、制御の目的、制御対象の特性、その他要求される制御機能を理解した上で、その機能を満足すべき装置の設計に入る。設計方針がきまり、次におこなうことは装置の構成要素である各種コンポーネントの選択及び、必要があれば、既存コンポーネントの改良、変更等の設計を行ない、最後にこれを接続することにより回路的な設計を終わる。以下製造のための設計を行なって製作に入ることになる。即ち、ここで言いたいことは、要求機能に見合った装置の設計製作というのは、そのために選ばれたコンポーネントを正確に電線で接続することであり、この作業を終った瞬間から装置は機能を発揮することになる。又、現場において、機能の追加や変更を行ないたい時は、それに見合ったコンポーネントを現場で取付け、然るべき点を接続するなり、あるいは電線の行先を変えて接続するなりすれば、そのことが機能の追加、変更を意味する。即ち、従来の制御装置は、ハードウェアの製作完了が、機能の達成を意味していたわけである。

この事実は、非常に大事なことなので、もう一度、多少見方を変えて、図によって説明しよう。



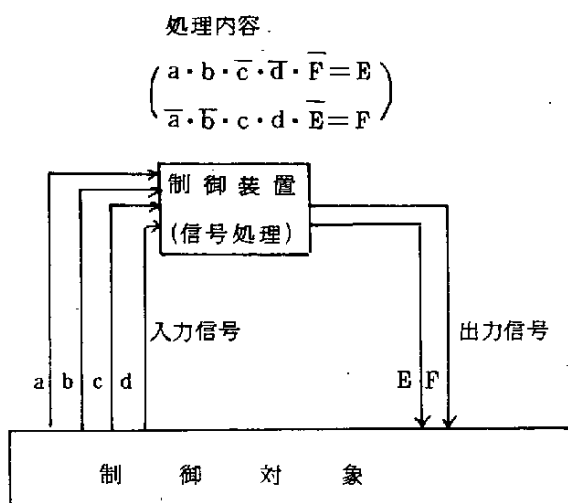
第 2 - 1 図 展開接続図の簡単な例

第 2 - 1 図は、もっともありふれた、そして簡単なシーケンス制御の例を、慣例の展開接続図という形で表わしてある。これを見た人は直ちに、a、b、c、d、というリミットスイッチの条件が制御対象の中にあつて、a、b、両リミットスイッチが動作していて、c、d、リミットスイッチは不動作である時、EMM というモータが運転され、逆に c、d、が動作、a、b、が不動作の時は、FMM というモータが運転され、EMM と FMM は同時に運転されないといった制御機能を読みとることができるだろう。そしてこの図は、接続を主体にして、制御のフロー（流れ）を表現している。即ち、制御対象機器、制御用コンポーネントと制御の論理とが、接続を通じて一体となって表現されているものと言える。

こうしたシーケンス自動制御等が発達しない前はどうかであつたろうか。多分、a、b、c、d、のリミットスイッチの動作状態に相当する状況を、人が監視し、上記制御論理を人が判断して、EMM、FMM を運転したのであろう。この場合、人間にとって、a、b、c、d、なる信号に属

する状態は、眼で確認され、判断されて、人の意志でマグネットスイッチが投入され、EMM、FMM が運転されたであろう。即ち、a、b、c、d、は人にとって入力であり、EM、FMのマグネットを入れることが出力であり、その論理的判断は、信号の処理である。この人の代りに、信号処理を行なうものが「制御装置」とするとすれば、第2-2図のような見方ができる筈である。勿論このままでは、処理の内容がまったく不明である欠点があつて、これを何らかの形で補なわないと、何のこともわからないが、例えば Boolean、Equation のような表現でも使つておけば、内容を伝達す

ることができる。こうした見方で、もう一度、第2-1図を見ると、点線で囲まれた部分①が入力信号であり、一点鎖線②の部分が出力信号に相当する。残りの部分は、信号処理に必要な部分であると言える。ここで信号処理をする制御装置は、コンポーネントとその接続により機能を達成し



第2-2図 制御装置の概念図

ていることがおわかりいただけるであろう。たとえ、この回路のIOやMSIの一部を使って表現しても、基本的にコンポーネントとその接続のみによる機能表現という根本原則は変わらないことになる。

マイクロコンピュータを制御装置のコンポーネントとして使った場合、この従来の手法のみでは機能の達成はできない。マイクロコンピュータは小なりと言えどもコンピュータであることにまちがいない。コンピュータという機械が、目的の機能を達成するのは、機械だけ用意しただけ即ち、ハードウェアを用意しただけでは何の役にも立たない。ソフ

トウェアの開発、具体的には機能を発揮するプログラムを機械にロードしてやらなければ、目的を達せられない。

一般的に言って、コンピュータを制御装置のシステムコンポーネントとして使用する場合、制御対象からの入力、制御対象への出力は、プロセス入出力装置と呼ばれる比較的定形化された論理回路群を通して、コンピュータの入出力チャンネルと称するデータの通路に接続される。入出力信号は一般にはデジタル出力、アナログ入力、アナログ出力に分類されて各々接続される。接続によって機能を表現し得るのはここまでの、信号の処理論理等の大半はソフトウェアで表現されることになる。マイクロコンピュータを制御装置のシステムコンポーネントとして使う場合にも大体同じと考えてよいので、制御技術者は、制御機能の表現手法として、新たにコンピュータのソフトウェアの技術を身につけ、しかもハードウェア技術とソフトウェア技術を、バランスよく身につけることが要求される。

次に、コンピュータというシステムコンポーネントの機能をつきつめてみれば、基本的な機能は4つぐらいしかない。即ち、

- a. 演算機能 (四則演算・論理演算)
- b. 記憶機能
- c. 比較機能
- d. データ移送機能

複雑な機能もすべてこれらの機能の組み合わせであると言える。

ふり返って、一方従来の制御装置のコンポーネントをみると、これらのどの機能を取っても豊富であるとは言い難い。むしろ、その貧弱さで間に合っている部分を間に合わせていたとも言える。それ故に、この新しい武器は、従来の制御装置の体質を大巾に変える要因をいっぱい含んだものであると言える。

こうした状況に目を向けない制御技術者がいるならば、残念ながら取り残されて、数年以内に制御技術者の看板をお倉入りにしなければならなくなるであろうと予測される。

そこで、新しい発想がこのインパクトによって余儀なくされることに

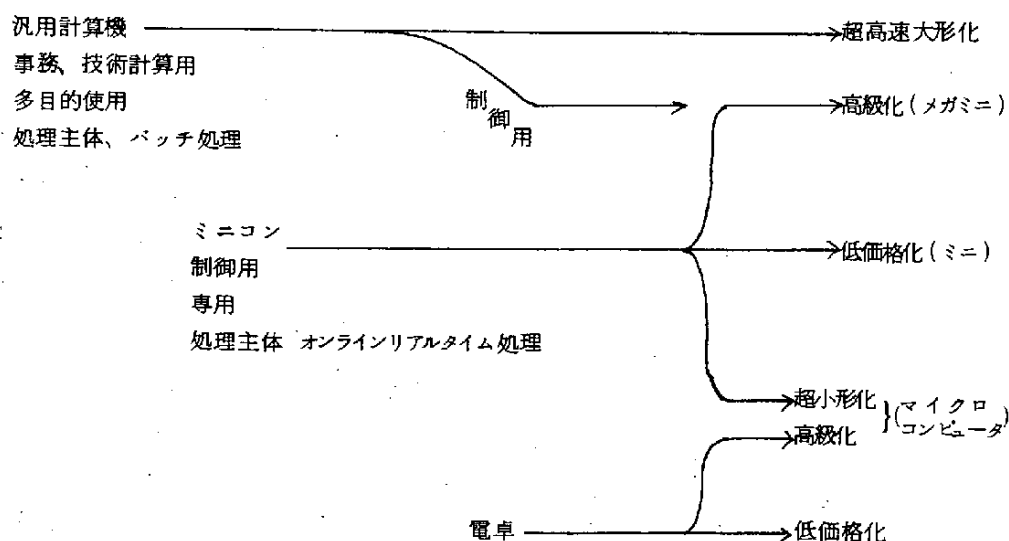
なるであろうことは、賢明な読者は既に感じられていることであろう。

筆者は、これからマイクロコンピュータを学び、制御技術の革新と共に歩もうとされる方々に、一つの考え方の例をお話しするつもりで書いているわけで、参考になれば幸いである。従って、マイクロコンピュータの個々の技術や、この利用の仕方の具体的な解説を試みようとは思っていない。むしろ新しい発想を制御技術者の方々がされることを刺激する程度に止めざるを得ない。なぜならば制御技術の発展は、現在の視点での予測と必ずしも一致しないかも知れないし、又新しい発想というのは、当然多様性に富むものであるからである。

### 2.1.2. マイクロコンピュータの歴史的背景

本題に入る前に、マイクロコンピュータの歴史的な背景を認識しておくことは、将来のマイクロコンピュータを考える場合に非常に役に立つと思われるので、ごく簡単にふれておきたい。

| 年代       | 1955  | 1960 | 1965     | 1970 | 1975      |
|----------|-------|------|----------|------|-----------|
| コンピュータ世代 | 第1世代  |      | 第2世代     |      | 第3世代      |
| 基本素子     | (電子管) |      | (トランジスタ) |      | (IC)      |
|          |       |      |          |      | (MSI,LSI) |



第2-3図 コンピュータ開発動向概念図



第2-3図は、この歴史をごく簡単に図解したものであるが、コンピュータという機械が世に出た時点で、既に、IBMは業界の巨人であった。コンピュータ業界は極端なガリバー形の業界で、1人の巨人と、多数の小人といった感じである。IBMの目指していた開発の方向は、一口に言えば、超高速大形化で、汎用で、アプリケーションの主体がビジネス用であり、処理の主体はバッチ処理であった。

これに対し、1960年のはじめ頃、米国のDigital Equipment Corporation（略してDEC）という会社が、コンピュータをいかに小さく、安くつくるか、汎用でなくても専用でもよいではないか、事務処理よりも産業用、制御用を主体に考えよう。バッチ処理よりもオンラインリアルタイム処理を主体にしたい、といった今までと大巾に異なる行方をして、いわゆる「ミニコン」を開発したわけである。大形の機種は、開発投資も気の遠くなるような予算が必要なために、何年かかけてその投資を回収しなければならないが、比較的開発投資の少なくてすむ「ミニコン」は、日進月歩のエレクトロニクス技術を吸収して進歩し、一つの大きなビジネスに発展してきた。1970年代に入り、この「ミニコン」は多様化しはじめ、高級化、低価格化、超小形化といった試みが行なわれ、これがファミリーを構成し、高級化した「ミニコン」は、IBM相当機種の中形のもののアプリケーション分野に喰い込むようになった。

一方、半導体技術の進歩はめざましく、前述したように、2～3年に1桁ずつ集積度を増すような快挙が続いてきた。そして数千個の部品を集積したLSIチップができるようになってきた。たまたま、日本の電卓メーカーが、アメリカのIntel社に高級電卓用のLSIチップを発注し、この開発が契機になって、1971年に4ビットのマイクロコンピュータ「MCS-4（型名4004）」が発表された。この時点を契機にして半導体業界にはマイクロコンピュータ・ブームが興り、Intel社は勿論、National Semiconductor社、Fairchild社、Motorola社、Rockwell Microelectronics社、等、次から次へと種々の機種を発表してきている。一方、ミニコン・メーカーも、前述し

た超小形化構想を進め、「ミニコン」のマイクロプロセッサバージョンとも言うべき機種を発表し、今やマイクロコンピュータは戦国時代に入った感がある。

しかし、ここではっきり言えることは、10年前、はじめて「ミニコン」が世に出た頃の「ミニコン」の機能と現在のマイクロコンピュータの機能を比較すると、概括的に見て、同じか或いは現在のマイクロコンピュータの方が優れているとも言える。性能-コスト比的な見方をすれば、恐らく数倍になり、同じ性能のものとすれば、コストは数分の一以下になっている。一方、人件費や諸材料の値上り等を考えると、10年前にはとても考えられなかったような仕事をマイクロコンピュータにやらせることが可能になったわけで、従来採算点に達しなかったシステムも、除々に採算に乗るようなことになりつつある。この辺のことはしっかり頭の中に入れておかないと、コンピュータ・システムは高価で、大がかりなことをやらないと採算がとれないといった従来の固定概念をマイクロコンピュータにより打破される面が出てきている事実を見逃がすことになる。

## 2.2. 制御機能の機能モジュール分割と再構成

### 2.2.1. 制御機能モジュールの分割

第1章で、制御機能の分類について、制御機能の基本的な見なおしができているものとして論を進めたい。相当重複もあろうし、意見の異なる面もあろうかと思われるが、前にも申し上げたように、これは単に考え方の一例として筆者の意見を述べているので、読者諸氏の賢察をお願いしたい。

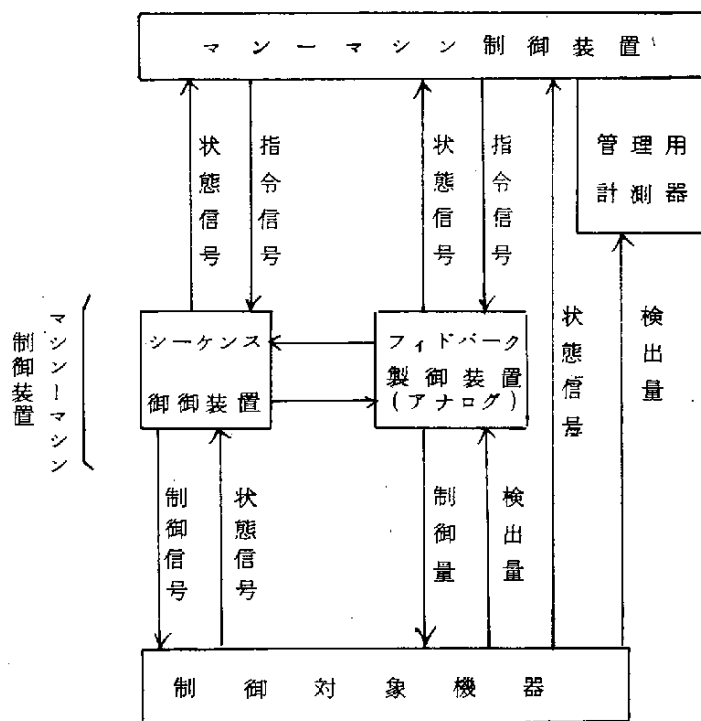
一般に従来の制御装置と呼ばれているものを、その使われている制御技術から分類すれば、第2-4図のように

- ・ マン-マシン制御装置
- ・ マシン-マシン制御装置
- { シーケンス制御装置
- { フィードバック制御装置

に分類することができる。種々の分類方法があろうかと思うが、制御関

系の技術者にとって、やはり制御技術によって制御機能モジュールに分割するのが、やり易い方向であると思われるので、この方法を取った。

今最も簡単な例として、天井走行クレーンに例を引いてみよう。しかも、このクレーンは、2点間を往復してA点で荷を積み、B点で荷おろしをする作業をしており、走行、捲上げ捲下し共に、直流モータが使われ、横行にはインダクションモータが使われているものと仮定しよう。



第 2 - 4 図 制御装置の一般的構成

一般にはこうした装置を製作すれば恐らく動力が小さければ、一体にして製作してしまうものと思われるが、これを第 2 - 4 図的な分類の仕方をすれば、マシーンマシン制御装置の各制御装置の扱い機納は、

シーケンス制御装置

運転・停止 (人間からの指令) 制御

非常停止機能 (安全機能を含む)

故障停止機能（安全機能を含む）

定位置停止機能

走行、横行、捲上げ捲下し関連制御機能

フィードバック制御装置（アナログ）

走行の加減速及び一定速度走行機能

定位置停止時のシーケンス制御との連動機能

捲上げ捲下しの速度制御機能

等を挙げることができる。

上記仮定の中で、A点とB点が固定であるという条件があれば、走行の減速制御の指令は、リミットスイッチや、場合によってはタイマなど、従来のシーケンス制御的な手法でそのままであるが、もしこれを可変とし、A、B、点共に可変で勝手に人間が設定でき、リミットスイッチ等は機械構造上設置が難しいとなると、制御機能の複雑さは一変する。恐らく、走行の回転部分のどこかに単位走行距離毎にパルスを発信するパルスジェネレータを取付け、走行位置をパルスの計数により検知する機能、或いは基準点の校正をする位置に一つだけリミットスイッチを取付け、更に、A点、B点の設定のできる機器をマン・マシン制御装置側に取付け、ここでいわゆる「数値制御機能」を付加することになる。一般には、数値制御装置といわれているのは、関連するシーケンス制御機能、フィードバック制御機能等を合せて呼ばれているが、制御機能の分割を考える時は数値演算機能およびそれに基く設定指令機能だけを取り上げて、数値制御機能としておきたい。数値制御機能とシーケンス制御機能は、同じく Discrete（離散的）な信号を扱う意味では以ているが、数値制御装置は複数のビットを数として扱うのに対し、シーケンス制御装置では、1ビットの信号を独立して扱うことに大きな差がある。こうした数値制御機能が入ってくると、従来の回路技術手法では相当高価なものになりがちで、しかも柔軟性のあるものができ難く、この辺からマイクロコンピュータが登場し易くなることは容易に想像できるであろう。そして今まで経済性の評価から実用的にあきらめていたことが、必ずしもあきらめる必要がない時代が早ばん到来するような気がする。

次にマンマシン制御装置の機能について機能モジュール的な立場で分割してみる必要がある。この機能は多様性の権化みたいなものであるが、単純な機能の寄せ集めの要素と、複雑な管理機能がある。

先ず一般的に考えられるのは、

- ・人間からのシーケンス制御指令機能
- ・人間からのフィードバック制御設定指令機能
- ・数値制御の各種数値設定機能
- ・制御対象機器のON-OFF状態及び管理計器による定性的な量の監視機能

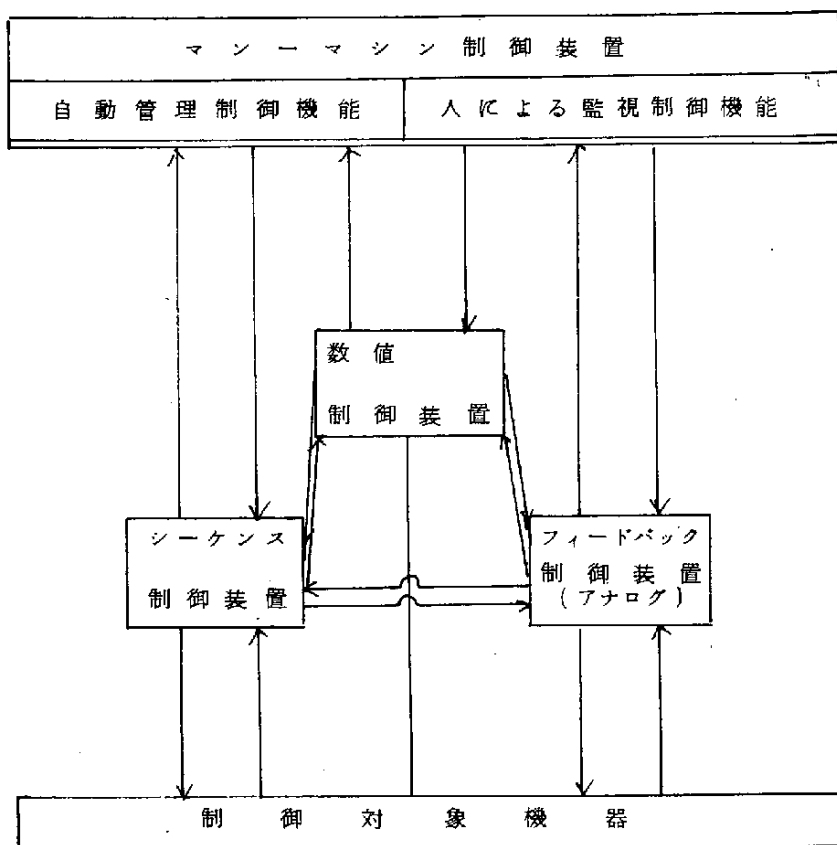
等が一般的であるが、マンマシン制御機能を高級化すると、

- ・監視量及び機器状態の自動監視機能
- ・各種の量及び機器状態の自動記録機能
- ・CRTディスプレイ等による集表示機能
- ・異常時連動対策或いはオペレーションガイド機能
- ・最適制御論理による計算結果による各種自動設定機能

等の機能を挙げることができる。後者の機能は、人にとって例外管理、例外監視制御に徹すると言った意味があり、運転管理の省力化に役立つ機能であると言える。

以上のような制御機能モジュールの大ざっぱな分割は、制御技術的な面から第一段階としてはきわめて一般的ではあるが分割し易いので、例として取り挙げてみたわけである。第2-4図をこれらの事項を入れて整理し、修正すると、第2-5図のようになる。

さて、以上のような分割を行なっても、これをどう整理するかが問題である。今更何のためにこんな分割をやるのかと疑う向きもあるかと思われるが、先ず、何等かの分割をしない限り次の整理ができないのと、恐らくこのような分割を従来の制御装置について極めて具体的にやってみられると、それだけでも「おや、！」と思われることが必ずある筈である。



第 2 - 5 図 高級な制御を含めた制御装置の一般的構成

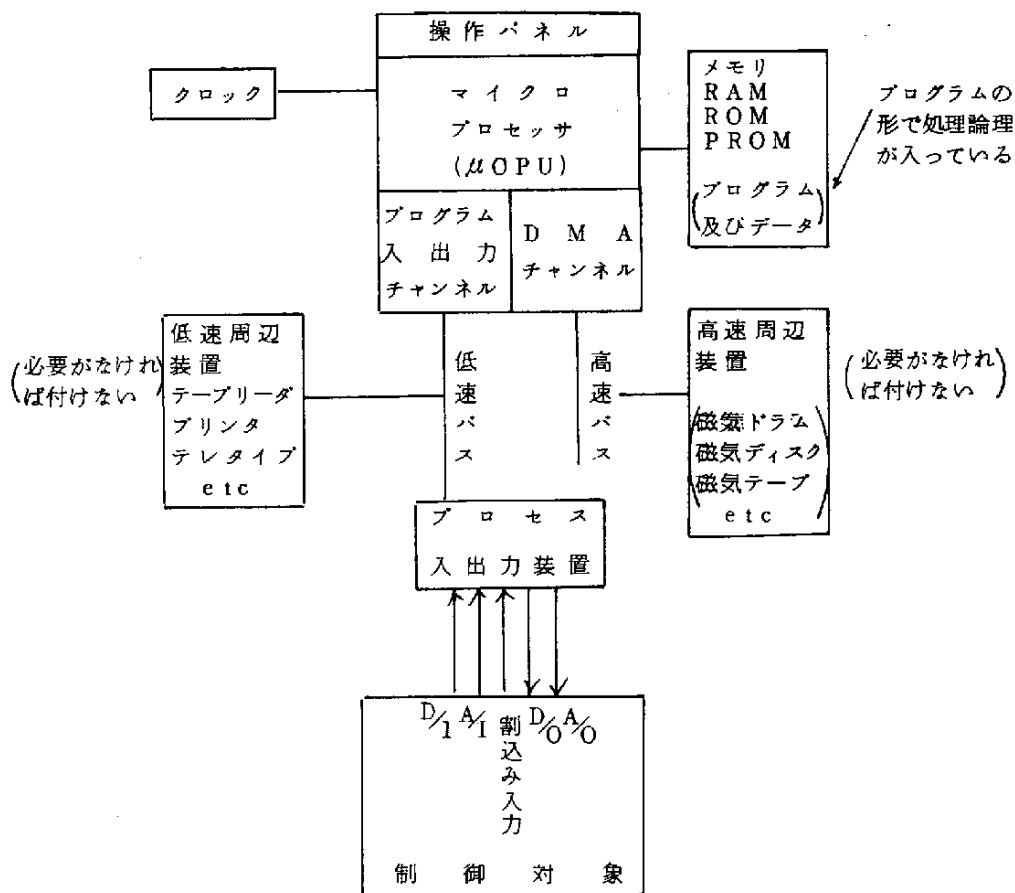
## 2.2.2. 制御機能モジュールの分割の整理と再構成の考え方

ここではっきりしておきたいのは、制御機能モジュールの分割と整理、再編成は何の目的で行なうかである。

前述した通り、マイクロコンピュータの出現は、制御装置の体質さを大きく変えるかも知れないという前提条件に立てば、制御装置のコンポーネントの主体がマイクロコンピュータで、制御装置はマイクロコンピュータ制御システムそのものである場合に、このシステムをどのようにまとめて、性能-コスト比を上げて行くかが問題である。勿論、1 台のマイクロコンピュータでまとまるものもあるだろうし、ポリプロセッサ

的なまとめがよいものもあるだろうし、マイクロコンピュータの階層構造にすることもあるだろうが、こうしたシステム構成をどうするかは次の問題であって、その前にやらなければならないことは、分割された機能モジュールを整理して、各機能モジュールの共通性や異質な部分、同質な部分を認識し、種々のフィロソフィで再編成し易くすることである。

それには、マイクロコンピュータを制御装置とした場合に、一般的にどんな形になるかを先に頭に入れておく必要がある。



第2-6図 マイクロコンピュータ制御システム概念図

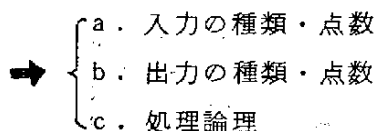
第2-6図は、マイクロコンピュータ制御システム概念図である。  
前述した第2-2図と比較対比されれば、制御装置の基本概念に殆んど  
同じ構成と言える。

制御対象機器からの信号は、ディジタル入力(D/I)アナログ入力  
(A/I)等の分類に従い、信号形態を揃えてプロセス入力装置に接続  
される。又割り込み入力とあるがこれは現段階では気にしなくてもよい。

接続された入力は、プロセス入出力装置を通じ、一般にはプログラム  
入出力チャンネルから出ているデータの通路(低速バス)に乗って、マイ  
クロプロセッサに読み込まれ、メモリに記憶される。この記憶されたデー  
タが必要な処理をされて、再び低速バスに乗ってプロセス入出力装置か  
ら必要な信号形態に変換されて、ディジタル出力(D/O)、アナログ出力  
(A/O)として制御命令となる。

ここで、賢明な読者はお気づきと思われるが、このマイクロコンピ  
ュータシステムに乗せていくために必要なことは、

#### 制御機能モジュール



このように整理することである。

前に制御技術的な制御機能モジュールの分割をしたのは、1つには、  
入出力の類似性と処理論理の類似性に着眼したに外ならない。

処理論理の表現の仕方については、できるだけ抽象化して、できれば  
数学的表現を多用することをおすすめしたい。こうした表現により、意  
外にも、一見共通性のなさそうに見えるものに共通性が発見できること  
を期待したいからである。そして、処理論理の類似性は、ソフトウェア  
の標準化につながり、又I/O点数の類似性はハードウェアの標準化に  
つながるわけである。ハードウェアとソフトウェアは、多くの場合、そ  
れぞれについて標準化することも可能である。

一般に、頭初客から示されたり要求されたりする仕様の内容を検討す  
ると、メーカから標準としている仕様など、10%ぐらいで後は特殊仕



様のように見える。

しかし、メーカーが市場で優位に立ち、性能－コスト比を上げて客に喜んでもらえるためには、一見、標準仕様10%、特殊仕様90%に見えるものを、標準仕様の組み合わせで90%、特殊仕様10%に逆転させ得るかどうか勝負どころであろう。

従って、標準化はますます多様化するニーズに対応すべく、機能モジュール的な標準化でないと意味がない。マイクロコンピュータを制御のコンポーネントすることは、この意味でもまことに好都合と言わざるを得ない。

次に再構成に当り、評価すべき要素は沢山ある。例をあげれば、

- ・ 経済性
- ・ 工事のし易さ
- ・ 信頼性
- ・ テストのし易さ
- ・ 拡張性
- ・ 商品価値
- ・ 柔軟性
- ・ 操従性
- ・ 保守性

等が考えられるが、もう一つその前に考えねばならないことは、その再構成したシステムが、実現可能なりや否やという基本的な問題がある。単純な制御機能モジュールの幾つかと、比較的複雑な制御機能モジュールを組み合わせ、複合制御機能モジュールを構成する場合には、多数の入出力に対するサービス、及び処理論理の実行を完全に行なわなければならない。起り得るあらゆる事象の組み合わせの中で、マイクロコンピュータの性能上、記憶容量、実行時間、等の制約を越えないかどうかを、具体的に使用したいと思っているマイクロコンピュータの性能上のチェックと合わせて検討する必要がある。しかし、この検討は、ミニコンやマイクロコンピュータ等のシステムを製作した経験を少なからず持っている経験者の判断が必要で、当初はあまり無理な組み合わせを行なわないことが安全である。

例として、前述した天井走行クレーンの例を挙げて、概念的な再構成の評価を試みよう。非常に乱暴で定性的な評価であるので、はじめに

おことわりしておくのと同時に実際にはもっと定量的な評価を行なったほうがよい。

第2-7図は制御機能モジュールの組み合わせの1例である。

組み合わせ(1)は、とにかく全部の制御機能をすべてマイクロコンピュータ制御システムに入れてしまう案である。

処理速度の点から言ってフィードバック制御機能の部分は、いわゆる「DDC(Direct Digital Control)」になり、相手が電気系であるため高速の処理を要求される。又、両電動機の世界速度制御を一つのシステムにまとめることは、運用上信頼度的に問題がある。即ち、マイクロコンピュータシステムの中心的なコンポーネントの故障により2台とも運転できなくなる。この意味では、個々にシステムを組まざるを得ない。即ち2システムのDDCシステムが必要となる。残りの制御機能モジュールは一つのマイクロコンピュータシステムで構成したとして、総計3つのシステムのネットワークになる。しかし、どう考えてみても現状このシステムが経済性を持つとは思われない。アナログのフィードバック制御装置のサイリスタのパワー増巾部分を除いた部分と、マイクロコンピュータによる「DDC」とが全く同じようなコストになるのには、多少時間がかかるだろう。

そこで、数値制御部分を除いてはすべて従来のやり方でいこうという案が反対に考えられる。

しかしこうすることによって、マイクロコンピュータシステムの扱う機能は位置の検出、減速点の計算とその点での出力信号ぐらいになってしまい、わざわざマイクロコンピュータなどを持ち出すことにあまり意義を感じさせないと同時に、その部分が高くなってあまり経済的でなくなる。せっかくマイクロコンピュータを使いながら、拡張性、柔軟性に欠けるきらいがある。

そこで次に数値制御機能に近いシーケンス制御機能と、マンーマシン制御機能を含めてシステム化することが考えられる。第2-7図(3)はその組み合わせである。しかしすべてのシーケンス制御機能をマイクロコンピュータ経由にすると、中心になるコンポーネントの故障で全く

| 制御機能<br>モジュール<br>組合せ | 数 値 制 御<br>機能モジュール | シーケンス制御機能モジュール       |                     | フィードバック制御機能モジュール    |                      | マン—マシン<br>制御モジュール |                     |
|----------------------|--------------------|----------------------|---------------------|---------------------|----------------------|-------------------|---------------------|
|                      |                    | 定 位 置 停 止<br>機能モジュール | 左 以 外 の<br>機能モジュール  | 走行用電動機速度<br>制御モジュール | 捲上下用電動機<br>速度制御モジュール |                   |                     |
| ( 1 )                | ○                  | ○                    |                     | ○                   | ○                    | ○                 |                     |
| ( 2 )                | ○                  | ×                    |                     | ×                   | ×                    | ×                 |                     |
| ( 3 )                | ○                  |                      |                     | ×                   | ×                    | ○                 |                     |
| ( 4 )                | ○                  |                      | ※ <sup>1</sup><br>× | ×                   | ×                    | ○                 | ※ <sup>2</sup><br>× |

○ マイクロコンピュータシステムに組み込む機能

×                      "                      組み込まない機能

※1. 手動で個々に運転停止をする最低の機能と安全装置機能

※2. ※1で制御される場合の人間による監視機能

### 第2-7図 機能モジュールの組み合わせ例

運転が不可能になると共に、安全上も問題があって、操縦性に難点が出てくる。

そこで、全く手動で動かす機能及び安全装置機能と人による最低限の監視機能を除き、更にフィードバック制御機能を除いた残りをシステム化すれば、前述の各評価要素で評価してみても、大方よい評価がもらえる可能性が濃いと言える。このような定性的な評価を表わしたのが2-8図である。

| 評価要素<br>組合せ | 経済性 | 信頼性 | 拡張性<br>柔軟性 | 操縦性 | 保守性 | 工事の<br>し易さ | テストの<br>し易さ | 商品価値 |
|-------------|-----|-----|------------|-----|-----|------------|-------------|------|
| (1)         | ×   | ○   | ○          | ○   | △   | ○          | △           | △    |
| (2)         | △   | ○   | △          | △   | △   | ○          | △           | △    |
| (3)         | ○   | △   | ○          | ○   | ○   | ○          | ○           | ○    |
| (4)         | ○   | ○   | ○          | ○   | ○   | ○          | ○           | ○    |

○ 大きな問題はなさそうである

△ 問題がある

× よくない

第2-8図 第2-7図の組み合わせの定性的評価

実際にはもっと別の評価要素も入れて評価に点数をつけ、各評価要素に「重み」をつけることを考えたら、もっと異なった見方になるであろう。

さて次の例として、幾つかの比較的独立度の高い制御対象機器があって、これらが集まって、エネルギー、資源、物質の加工、変換、処理等の

ライン設備を構成している場合の、制御機能の再構成について簡単に述べたい。具体的例としては、鉄鋼の圧延ラインや仕上げライン、上水道における各種処理機械、石油化学製品の連続製造ライン、繊維関係の各種製造ライン等を思い浮かべていただきたい。

マイクロコンピュータを意識して再構成する場合、このまとめ方が種々ある。一つのまとめ方は各制御対象器単位にまとめる方法である。今ここではこれを「縦のまとめ」としておこう。もう一つは各制御機能ごとに即ち、シーケンス制御機能はシーケンス制御機能で、フィードバック制御機能はフィードバック制御機能でまとめる方法である。これを、「横のまとめ」としておこう。フィードバック制御の「横のまとめ」としてプロセス制御関係では、数年前から「DDC」が盛んに採用されるようになった。勿論今までは制御用コンピュータとして、「ミニコン」以上のコンピュータが使われ、御御ループにして数百ループを扱い、信頼度面への配慮として2系列化しているのが一般的である。しかし、現在この手法をマイクロ化して数ループ宛まとめた「DDC コントローラ」が各社で開発されつつあり、計装関係のシステムは大きく変ぼうを遂げようとしている。こうした背景から、反応速度的に比較的遅いループのフィードバック制御装置を横にまとめる方法は既に実用段階に入りつつあるといえる。

縦、横のまとめ方については、この組み合わせを幾つか作って前述したような評価基準により評価してみると良いかと思うが、定性的な言い方をすれば、横のまとめの場合には集中化のメリットがあつて、経済性、総合的な運用の面では効果があるが、信頼性の面で問題があつて、この面を補うとすれば2重化の必要も生じ、逆に経済性をそこなってくる。

縦のまとめの場合、各機器ごとの標準化ができるような制御対象機器であれば標準化の効果による経済性を考えることができるし、又一つの故障の波及効果が全体に及ばないという意味で信頼性面では楽になる。特にA、B、C、D、が殆んど機能的に同種のものであれば、予備をおくこともできるから、全体のシステムの稼働率を上げることもできよう。

しかし、縦のまとめ、横のまとめに一番大きな影響をもっているのは、

制御対象機器同志が目的を達成するためにどのような結合がされているかである。

例えば圧延ラインのように、一つの材料が加工されてできるプロセスの中では、どの機器が故障してもラインは止ってしまう。

それに対し繊維関係等でよく行なわれているように前段の工程の後には必ずストレージがあって、半加工製品は一時このストレージに入れられるようなラインもある。電気的な表現をすれば前者は「密な結合」であり、後者は「疎な結合」である。

一概に、これだけでは片付けられないが、大ざっぱに言って、ラインを構成する機械群が「密な結合」の際は「横のまとめ」がやり易く、逆に「疎な結合」の時は「縦のまとめ」がよいことが多い。

以上、再構成について定性的な考え方だけ述べたが、再構成の際、一つ大事なことがある。マクロコンピュータという新しい武器が提供されているわけであるから、今まで漠然とこんなことをやったらメリットがあるかもしれないと思っていることが実現できないだろうか、逆に、マイクロコンピュータを生かして使うような制御機能は何であろうか、と言った発想から従来からある制御機能だけではなくて、新しい発想や検討に基く制御機能のモジュールを入れて再構成を考えていただきたい。

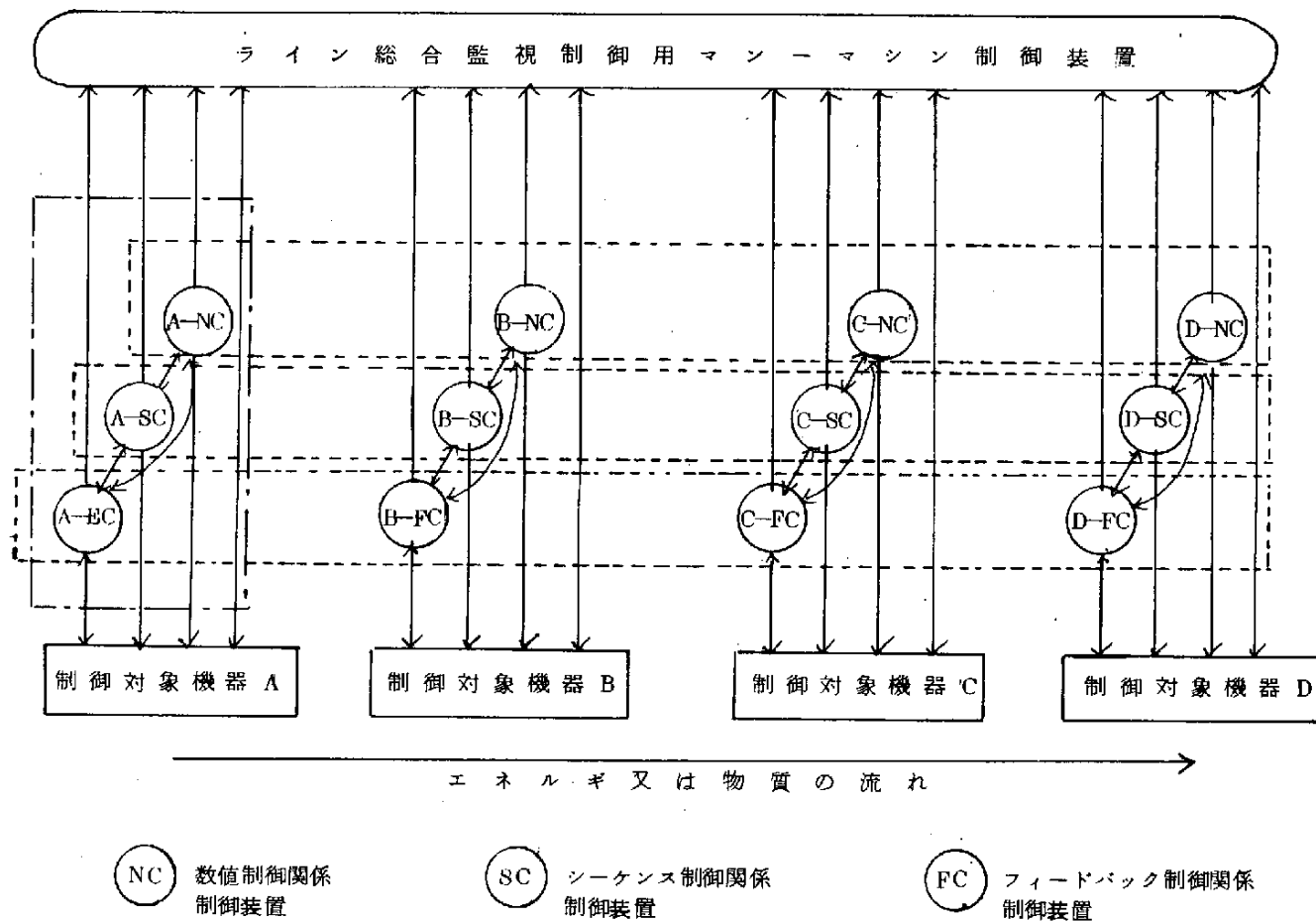
### 2.2.3. システムとサブシステムの関係

制御機能モジュールを再構成してつくられた一つのまとまった制御機能も、恐らく全体から見れば一つのサブシステムであって、これらのサブシステムが幾つか集まってシステムを構成する。

現場制御装置に、マイクロコンピュータを使う時代に、上位にミニコンや汎用コンピュータが設置されないようなケースは殆んど考えられない。

その意味で、マイクロコンピュータ制御システムをつくった時は非考えておかねばならないことがある。これ等を挙げておくので参考にされたい。

- (1) サブシステムを構成する場合、できるだけ独立度の高まる方向でまとめること。



第2-9図 ライン構成をした制御対象機器群の制御装置の概念図

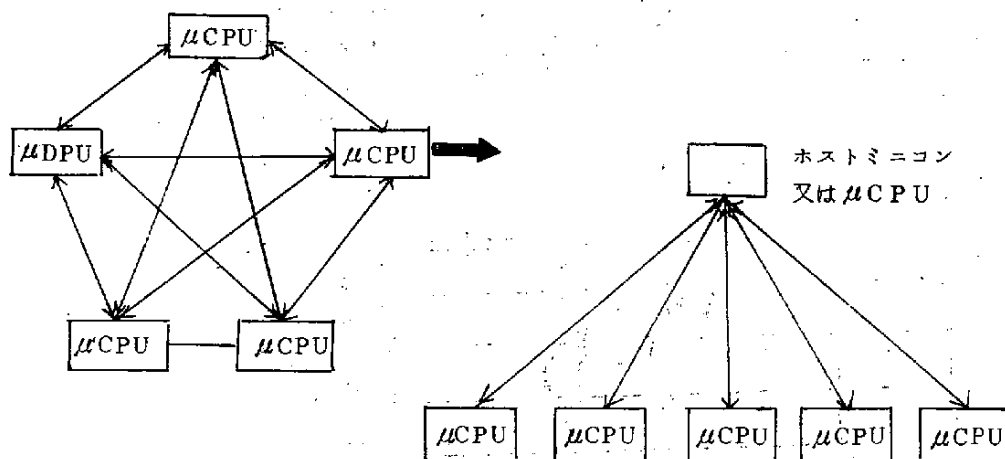
ここで独立度の高いというのは、サブシステム同志の信号のやり取りを可能な限り少なくする意味である。

- (2) サブシステムの切口、即ち信号のやり取りの方法を標準化すること。
- (3) このために必ずマイクロコンピュータシステムに何等かの通信機能をもたせ、ハード的ほもソフト的にも充分な配慮がされていること。
- (4) この通借機能による結合も、サブシステムの独立度を高める意味から「疎結合」を考慮すること。

「密結合」にすると、システム全体が一つの団子になって必ず信頼性で問題になると同時に、テストのし易さ、保守等で問題が発生する。

- (5) サブシステム同志の通信機能が増加すると、システムが複雑になり標準化という面から問題になるので、時間的に許されるならば階層構造によって上位コンピュータ経由で信号のやりとりをすること。上位コンピュータがなければ、通信制御用のマイクロコンピュータシステムを設けてもその方が簡単になる。
- (6) 制御機能面からみて、サブシステムだけで最適化するような形にすると、トータルシステム的には使い難くなるケースが多い。

制御パラメータのどれが全体のシステムの運用に関係があるかをよく見きわめて、これをマシン—マシンで変更できるような配慮が必要である。



第2-10図 サブシステム間の結合概念図



## 2.3 ソフトウェア技術

実は厄介なことに、コンピュータという機械は、「聞いてわからず、見てわからず」という機械である。どうすればわかるかと言えば「試してみる」より方法がない。使うためには、ハードウェアの勉強ばかりでなくソフトウェアの勉強をしなければならない。

一度使ってみれば、これから述べるマイクロコンピュータのソフトウェアの特長も、製作上の問題点も、恐らくわかる筈である。してみればソフトウェアについてここで述べるのは、知っている人を対象にするならば「仏に説教」、知らない人にとっては「何とかの耳に念仏」の類になるのではないかと、甚だ疑問と矛盾を感じる。

従ってここでは、興味はあるがまだ勉強途中の方に、こんなこともあるよといった程度で述べたいと思う。

### 2.3.1. マイクロコンピュータ ソフトウェアの特徴

一口にマイクロコンピュータと言ってもそのレベルはいろいろある。

従って、論を進める前にその辺を限定しておきたい。

前にも述べたように、コンピュータの作り方が「CPUチップ」「メモリチップ」と言ったLSI素子を使ってつくっており、でき上がった機能そのものは、「ミニコン」であるようなマイクロコンピュータは、ソフトウェア的に見れば、「ミニコン」のソフトウェアそのものであるもので、このような機種のソフトウェアについては論じないことにする。

ここで論じたいのは、制御装置に組み込まれるような「Component-like」なマイクロコンピュータで、多くの場合4bitか8bitの語長のものについてである。

#### (1) マイクロコンピュータのハードウェアの制約からくる基本的な特長と問題点

a. マイクロコンピュータのハードウェアの小ささに比べると命令の種類が比較的多い。

b. しかし、命令の種類が多い割に、使ってみて痒い処に手のとどくような便利な命令が至って少ない。その意味で、インストラクションセット（命令体系）は貧弱である。

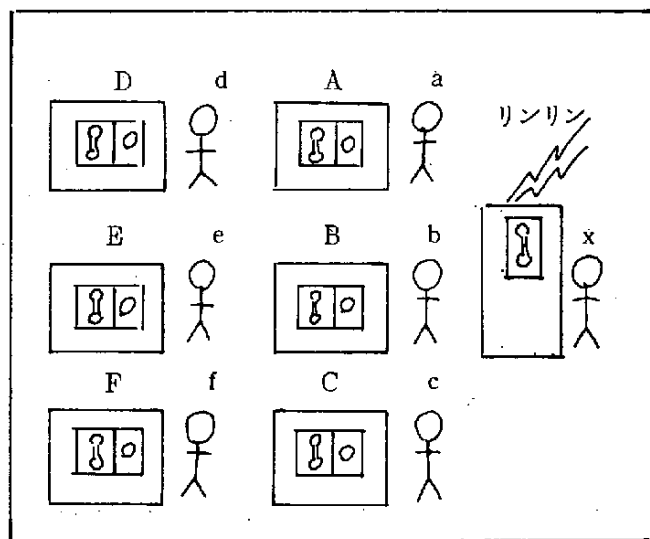
- c. 間接アドレッシングができると書いてあっても、「ミニコン」の間接アドレッシングとは異なる場合が多い。
- d. 語長が短いために何回も fetchするので命令の実行時間が長くなる。
- e. 加減算は楽であるが乗除算は厄介で、サブルーチンの扱いにも注意を要する。
- f. サブルーチンを作る場合、よほどうまく作らないとサブルーチンをコピーしてきて、ブラックボックスとして扱うことが難しい。
- g. 初期の「ミニコン」程度の割込機能はハードウェアとしては備わっているのが普通であるが、ソフトウェアで割込サービスルーチンや、多重割込をうまく扱うように組もうと思うと、うまく便利に組めるようなインストラクションセットになっていない。スタック機能があると書いてあるが、「ミニコン」の代表機種の一つである、pdp-11/10と、マイクロコンピュータシステムMCS-8のスタック機能を具体的に比較してみれば、その差の大きさに気がつくであろう。

## (2) アプリケーション上の特徴

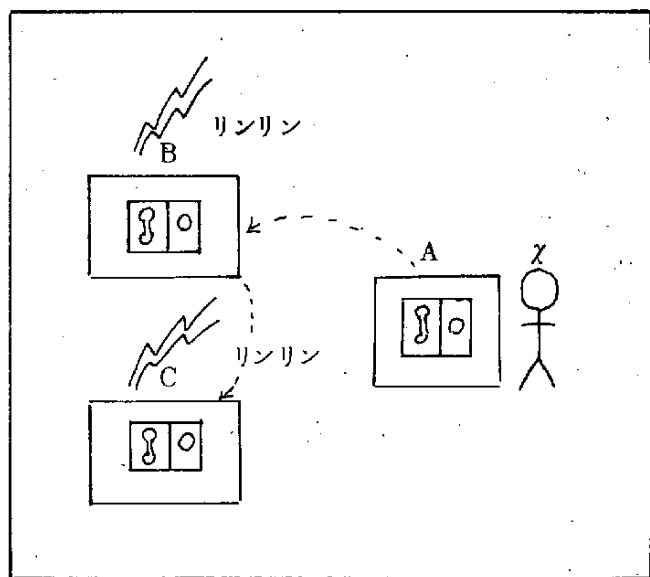
以上のような理由から、「ミニコン」並の管理プログラムを作ってもその管理の下でアプリケーションプログラムを作り走らせようとする、恐らく管理プログラムの大きさが「ミニコン」の場合の倍以上になり、オーバーヘッドタイムが非常に長くなると同時にアプリケーションプログラムに相当な制限が加わるであろう。

又こういうマイクロコンピュータは、メモリの最大容量が比較的小さい。例えば、Intel社のMCS-8では最大容量16KBである。これらの制限も考えた上でアプリケーションプログラムを作る必要がある。

この辺を技術的な説明としては正確さを欠くが、感じとしてつかめる意味で、漫画的な表現をしよう。



(a) 多重並行処理の概念図



(b) 多重並行処理をしないケースの概念図

第 2 - 1 1 図 多重並行処理の概念説明図

第2-11 図(a)は多重並行処理をする場合の漫画で、xなる課長が一人いて、aという人はAの仕事をしており、bはBの仕事をしている。以下f、Fまで同じである。外から何らかの仕事が電話で来たとすると、課長xの前の電話が鳴る。xは依頼の内容を聞いて、Aの仕事と判断するとaに電話して「君の仕事だ、やり給え」と指令する。又電話がかかってきて、それがBの仕事ならbにそれを命ずる。a、b、c……fが仕事のことで打ち合わせすることは許されているが指令はできない。指令は必ずxがする。従って外側から眺めれば、A、B、C、……F、の幾つかの仕事が同時に行なわれているように見える。

xの仕事がいわゆる管理プログラムで、aの仕事Aやbの仕事Bは、タスクと呼ばれている。

「ミニコン」等で一般に行なわれている比較的大きな制御システムは大体このような形になっている。

そうでない場合は第2-11 図(b)のようにxなる人が一人だけいて、鳴った電話器の処、例えばBに飛んで行ってその仕事をする。次にその途中でCの電話が鳴れば、Bの仕事を中断してCの仕事をする。Cの仕事が終われば再びBに戻って仕事をする。そして例えば、Intel社MCS-8(8008)では電話器が7台である。

前者は一応大企業の組織の形態をとっているが、後者は全くの小企業もいいところで、一人が社長兼事務員で、7つまで、仕事のリクエストを受けつける。但し、社長自らリクエストのない時に勝手にやる仕事は、時間の許す限りやっつかまわない。

マイクロコンピュータシステムでは、主として(b)のようなやり方をして、経費節減と能率の向上をはかっていくのが通常である。それ故に、アプリケーション・プログラムと(b)のようなパターンで作った場合に、考え得るタスクの組み合わせの中で処理時間に余裕のあるように機能を決めねばならない。その意味でマイクロコンピュータのアプリケーション・ソフトウェアでは、冗長ではない効率のよいことを特長すべきである。

### 2.3.2. ソフトウェア製作上の問題点

「ミニコン」以上のクラスのコンピュータのソフトウェア製作と、

マイクロコンピュータ・ソフトウェア製作との間には若干の差がある。

(1) 100%の正当性が製作時点で必要である。

マイクロコンピュータのプログラムは一般には、ROM、又は、PROM に入り、データはRAMに入る。ROM、PROM、RAMについては、別の篇で詳細に述べられているのでそれを参照されたい。

さて、ROM、又はPROMにプログラムを書き込むには、書き込み器を通して書き込むわけで、ROMは一旦書き込むと訂正できないので新たに訂正したプログラムを別なROMに書き込んで交換することになる。PROMでは、紫外線を照射して一旦そのPROMチップに書き込まれたプログラムを消し、あらためて書き込み器を通して書き込むことになる。いずれにしろ、現地に運んで試運転してみて、間違っていたら現場で訂正すればよいといった安直な考え方はできない。要は、現場では訂正できないと思ってやるべきである。従って、プログラムのテストを充分やって、製作時点で100%の正当性のあるプログラムにしてやらなければならない。

(2) プログラム開発費用が意外に大きい。

(1)で述べたような意味と、前項で述べた命令体系の貧弱さから、所要の機能を出すためのプログラム開発費用は「ミニコン」等と比べると相当大きい。どのくらい差があるかはシステムづくりの難易度及び技術者の能力により異なるが、上記(1)のような完全なデバックを含めて考えると、一説によれば2～4倍はあるといわれている。

よく知らない ユーザは、マイクロコンピュータのハードウェアが安く、又「ミニコン」並みの機能をもっているからという理由で、一品料理的なシステムをマイクロコンピュータでやれと言われている。こうした ユーザの中には、マイクロコンピュータを使いさえすれば安価で信頼度の高いシステムができると思っている人がいる。米国のDEC社は、LSI-11というミニコンのマイクロセッサバージョンの型録の表紙に、「LSI-11は、マイクロコンピュータ気狂に対するDEC社の解答である」という意味の文句をうたっている。現状では、このマイクロコンピュータ気狂はどうも沢山のさうであ

る。同じ程度のシステム機能を「ミニコン」とマイクロコンピュータとで製作した場合、ハードウェアのコストの差は、100万円程度の差であろうと思われる。しかしプログラムの開発費用の2～4倍という数字は、一品料理的なシステムではこのハードウェアのコスト差程度を、すぐ埋めてしまうことが多い。この意味からも、一品料理的で多少複雑になることが予想されるシステムをマイクロコンピュータで実現するのは、あまり賢いことだとは言えない。

しかし繰り返しのきくシステムでは様相が一変する。

例えば、10システム全く同じシステムを作るとすれば、開発費用は1システム当りでは $1/10$ になる。100あれば $1/100$ になるわけで、こうなればハードウェアコストの安い分がそのまま生きてくる。

現在はマイクロコンピュータシステムの揺籃期で、将来を楽しみに一品料理的なシステムでもマイクロコンピュータでやっている面もあるが、これはあくまで先行投資であって、一般的な意味で、ソフトウェア開発費の割り掛け分がシステム価格の20%以下になるようなシステムでないと、メリットはでてこないのではなからうか。逆に、それができるような制御機能の再構成が必要になると言える。

### (3) ハードウェアとソフトウェアのマッチング。

制御機能を実現する場合、種々のケースのダイナミックなタイミングの検討が必要なことは制御技術者としては周知のことである。この場合、プロセス入出力とそれに対するソフトウェアの処理時間を考える時、非常にcriticalな点にきてしまうようなこともあり得る。簡単な例で挙げると、或る種の積算量があって、単位量あたり1発パルスが発生するとし、これをカウントして積算量とするようなものであるとする。しかも、この積算量が何種類あるいは何量かあって、そのパルス間隔が非常に短い場合、マイクロコンピュータの処理時間の殆んどが、このパルスの処理に食われてしまってそれ以外のサービスができないようなことを想定しよう。この場合、例えば各積算量に対応して、10進一桁のRead out counterを用意し、このカウンタ

の数値をパルスの最小周期の10倍以下の時間内に一度読むようなことを考えれば、忙しさが1/10に感じて、他のサービスができることになる。

このように、どこまでをハードウェアにもたせ、どこまでをソフトウェアにやらせるかと言った機能に対するハード・ソフトの分割は、システムを具現する上で大切なことである。

マイクロコンピュータシステムは、それより上級のコンピュータシステムにくらべて、よりハード的といわれるのは、この辺の事情もその一因である。

#### (4) 一品料理対策

##### a. クロスコンパイラの開発

ソフトウェア開発費が大きいので、一品料理的なものにマイクロコンピュータを使うのは考えものだという事を前に述べたが、それなりの覚悟と用意をしてのことなら話は変わっている。

制御機能的にはそれほど難かしくないが、何しろ制御機能モジュールの種類が多い上にその組み合わせが一ぱいあって、実際にプログラムを作る段階では、一回一回書かざるを得ないような場合であれば、一つの方策がある。

それは、その大半の機能モジュールの表現している種々の機能がある高級言語で定義することができれば、ミニコン等を使って「クロスコンパイラ」を開発することを考えてみたらよいのではないかと思われる。この開発ができれば、この高級言語で要求される制御機能が表現され、これをそのミニコンに入力する。クロスコンパイラでコンパイルされ、機械語になったオブジェクトテープがミニコンから出力され、これを書き込み器を通してマイクロコンピュータのメモリに書きこめばよい。訂正する場合は、マイクロコンピュータのプログラムを訂正するのではなくて、高級言語で書かれたソースを訂正し、コンパイルし直し、書き込みを行なうようにすれば、ソフトウェア製作工数が格段に減る。

むしろ、積極的な意味で、こういう領域のこういう制御機能の範

囲であれば、一品料理結構ですと言ったビジネスが成立すれば、ちよとした素適なビジネスほなる可能性がある。

しかし、このコンパイラを開発することは容易ではない。よいコンパイラを開発するためには、マイクロコンピュータのプログラムをアセンブラで書いて様々な悪戦苦闘をし、問題点を充分知っている歴戦の勇士が必要になるような気がする。こうした勇士と、コンパイラを作ったことのある有能なソフトウェアスペシャリストのチームでないと、恐らくよいコンパイラの開発はできない。素適なビジネスにするには、それなりの裏づけが必要になる。世の中には、あまり調子のいい話はなさそうである。

- b. マイクロコンピュータを使って、ある種の制御機能に専用であるが、その制御機能の範囲内では汎用機であるような機種を開発する。

この一番良い例は、マイクロコンピュータに使った「シーケンサ」であろう。

「シーケンサ」はシーケンス制御にしか使えないが、シーケンス制御機能に限れば汎用機である。そして多分、入力は Boolean equation形式で書けば、そのまま入力できるとか、或る種の言語で書くか、又は特殊なハードのセットをすとかすれば所用機能が出るようになっている筈である。

## 2.4. ソフト・ハードの一体化

### 2.4.1. マイクロコンピュータシステム製作上の特長

ここで述べる製作上の特長というのは、あくまで従来の制御装置と、マイクロコンピュータ制御システムとの差を意識しての話で、上級機種のコンピュータ制御システムとの差を論ずるつもりではない。この意味で、特長といえるのかどうか分からないが、システム製作に関する特記事項を述べたい。

- (1) マイクロコンピュータ制御システムと従来の制御装置との製作上の差異



- a. ハードウェア機能は同じで、全く違う制御機能をもたせ得る。

このことにより、次のようなメリットが生まれる。

- ア. ハードが同じでも機能を変得るので、ソフトを変えることにより、製品の機能改良ができて、製品寿命を延ばすことができる。

- イ. ますます客の要求が多様化してくるが、この多様化に対応をソフトウェアに背負わせることができる。

- ウ. 新制御システム開発に対し、開発期間を短縮できる。

- b. 入出力点数をはっきりさせた時点で、ハードウェアの概略の大きさ、消費電力等がはっきりして、でき上がりの姿を予測することができる。

このことにより、計画や見積等が迅速にできることになる。

(2) マイクロコンピュータ-制御システム製作者の立場から見た特長

- a. 製作技術者の養成、教育

この新しい制御装置を製作するための制御技術者は、どういうふうにして養成するかが、製作を意図した責任者にとっては重要なことである。

2つの考え方がある。先ずソフトウェア技術者にハードウェアを教えてその任にあたらせる。もう1つは、従来の制御技術者にソフトウェアを教え、再教育してその任にあたらせる方法である。

どちらが良いかは全く当事者の能力による部分が大きいが、下記のような理由から、一般的に言えば、ハードウェア技術者でもある制御技術者にソフトウェアの教育をして、担当されることを推奨したい。

- ア. 制御技術そのものに関する過去の蓄積を生かせる。

制御装置のもつ、いやらしさとかある種の不確定性とか、ユーザーとの折衝、被制御機器の表に現われていない特性の把握や予測、運転者の心理、といったあまり教えごとではすまないような貴重な経験を、従来の制御技術者は持っている。制御システムを作るにしても、この辺の手ざわりは同じで、何とかこの経験を生かざないとよいシステムはできない。

イ. システム機能を、ハードウェア機能とソフトウェア機能にバランスよく分割するために必要なハードウェアの製作を億劫がらない。

従来の手法、即ち、ハードワイアドロジックで何か組んだ方がよい場合に勝手知った我家の台所となるので、こうした意味では、ハードウェア技術出身の方が適しているであろう。

ウ. タイミングに関する概念ができている。

アに関連するが、制御装置の設計で最も気を使うのは、タイミング及びそのマージンの問題である。制御技術者をしていた人なら、この概念が身についている。

エ. マイクロコンピュータ制御システムは制御装置の必ずしも全部ではなく、他のハードウェアとのマッチングを考える必要がある。

オ. 人材の能力開発は、企業環境変化に対応して生き続けねばならない企業にとって絶対に必要である。

このことは言うまでもないことであるが、従来の手法で作っていた制御技術者を蹴にして、新しくそれに向く人間を雇ってくるようなことは、現状の日本の企業の中では不可能である。むしろ何とかして能力開発をし、今まで企業の繁栄に貢献してきた人々に新たな力を与えるべきであろう。

## b. 企業規模、産業構造的変化

ア. 大メーカーと四畳半メーカー

コンポーネントを接続すること、即ちハードウェアの完成即機能の完成といった今までのやり方では、制御装置のコンポーネントの種類が比較的多く、又そのための製造設備に対する投資も又当然馬鹿にならない。この意味で制御装置はある大きさの企業規模を必要としていたことは事実である。

しかし、マイクロコンピュータは、Computer-like な形でも、ボード単位でも、又チップ単位でも、マイクロプロセッサやマイクロコンピュータメーカーから自由に買える。又、I/O関

係も定形化しているから大きな製造設備はあまり必要としない。勝負はどちらかといえばソフトウェアにある。制御のノウハウを知っていてプログラムをうまく作り、客の要求にマッチした機能を具現できる能力があれば、企業規模の大小はあまり関係なくなってきた。

従って、製作面からみれば大企業も四畳半メーカーも差がなくなってきた、むしろ種々な面で、大メーカーの慣性の大きさは、逆に邪魔になる。マイクロコンピュータシステムを売り物にするベンチャビジネスが多くなってきた所以である。

しかし、大企業側には有利な面が一般的に沢山ある。第1は、流通チャンネルの問題で、これが大きくビジネスを左右する。第2は品質管理に対する概念の差である。一般に小規模メーカーほど品質管理面での配慮がうすい場合が多い。第3は保守サービス面である。大企業はそれなりのサービスネットワークを持っていることが多い。日本における大企業と中小企業の構造的な格差に対する固定概念が、中小企業に対する圧迫となっていることも事実であろう。米国においては、特にエレクトロニクス関係の企業では、100人も従業員いたら太会社である。20~30人ぐらいの処が実に多く、こうした小さな会社は、すべて専門メーカーとしての技術と誇りを持っているし、社会構造が重層構造で適当な流通チャンネルを作り易い面がある。

産業構造の知識集約化がとえられている今日、やはりこうした情勢の中で、健全で、実力のある専門メーカーが育つことが期待されよう。

#### イ. 制御装置のサプライア

従来、制御装置のサプライアは、重電機メーカー、通信機関係のメーカー、配電盤メーカー等がその主力で、製造技術的にそうせざるを得なかったけれども、マイクロコンピュータの発展は、ハードウェア製作そのものを簡単にしてしまった。従って、アで述べたような理由と同じ理由で、今まで上記メーカーに頼っていた機械メーカーでも、自分等で制御装置を作って、機械と共に売

ることができるような環境になりつつある。

ずっと昔は、機械に占める電装品の比率がせいぜい5～10%であった。

しかし、現在、その比率が30～60%ぐらいのものは驚くには当たらない。こうなってくると、機械メーカーが競争力をつけるために何とかして外に金が出ていく電装品のウエイトを下げて、付加価値を高め、損益分岐点を下げようとするのは当然である。

その意味で、制御装置のサプライヤが多様化することになるものと思われる。

#### ウ. 保守サービスの問題

マイクロコンピュータ制御システムの保守サービスは、一体誰がやるのかという根本的な問題がある。

あるユーザは、コンピュータという名前がついている以上、メーカーが保守すべきであるという。それでは、メーカーの保守に見合うサービスフィーを払うかといえ、もとが安いので払えないという。

一方、メーカーは、既にブラックボックス的にあつかえるのだから、予備品か予備機をもっていれば取り換えるだけですむので、ユーザに保守をして下さいという。しかし、1つのプラントに沢山のマイクロコンピュータシステムが納入されていれば、予備をもつのに問題はないが、1つか2つであれば、予備機の負担が馬鹿にならない。又、ただ取り替えればよいというふうな形でものをいえば、ユーザの保守サービスに携わっている人等の向上心に水をさし、人間性尊重にならないという議論もある。これを解決するために、マイクロコンピュータの教育をメーカーにすべきであると言われて、やっではいるものの、3日や4日の教育ではじめての人にはわかるわけもない。

どこかに霞を食って生きる神さまのようなサービスマンがいてもよいのではないかといった議論になりがちである。

ここで今、私は腹案として、どうすればよいかを明確にもって

いるつもりであるが、この辺はこれを扱う企業の姿勢の問題もあるので、結論がましいことは言わないことにしたい。

しかしながら、ア・イ.の項で述べた制御装置のサプライア即ち、小企業のサプライア、機械メーカー出身のサプライアは、このあたりの明確な姿勢と実行なくしては、線香花火に終わるかもしれないことを配慮しておくべきであろう。

#### 2.4.2. ソフト・ハード一体化のメリット

今まで種々述べてきたことで既におわかりと思うので、蛇足ながら簡単に述べたい。

- (1) ハード・ソフトの一体化により、従来の制御装置に持ち合わせない機能を付加できる。
- (2) コスト・ミニマムの設計、性能-コスト比の高い設計等、設計活動領域が拡大し、競争力を上げることができる。
- (3) ハードウェアコンポーネントの種類が減らせるので、保守サービスがやり易くなる。
- (4) 上位コンピューターとの通信機能が従来の制御装置にくらべて簡単になり、トータルシステム化に大きく前進できる。
- (5) 多様化対策、製品寿命の延長等に対し有利である。
- (6) 制御装置の省エネルギー化である。

### 2.5 制御技術者の発想の転換

#### 2.5.1. 従来の制御装置の限界

制御技術者にとって、マイクロコンピューターの出現は、かつて石炭から石油にエネルギー源が変わった時ぐらいの感じがあるかもしれない。そこで、ここでは従来の制御装置のコストを頭にいった機能的な限界を考え、これからのマイクロコンピューターの狙いと将来の制御装置のイメージについて述べることで、発想の転換のための参考になれば幸いである。

##### (1) 従来の制御装置の限界

###### a. 表記能と裏機能

ここで表機能というのは、従来の制御装置で考えれば、制御対象機器と制御装置が順調に動作している時の機能で、一般には仕様書に載っている機能である。しかし、制御装置も制御対象機器も故障はあるから、うまく動作しないこともある。制御対象機器の明確な故障は、それなりに検知でき、停止して人間に知らせるようになっているが、それも簡単に故障検知用のハードウェアがコスト的に安くつくような場合に限定されている。

制御対象機器と制御装置のインターフェース部、及び制御装置の何かが故障した場合に、多くの制御装置は何もしなくなってしまうのが普通で、どのようなコンポーネントがどういう動作をしている時に故障が起ったかといったことは、人が後からよく調査してみないとわからない。

例えば、数台の制御対象機器があって、これが密な結合をしており、数10ケのリミットスイッチや、ON-OFF信号センサが設備されていたとする。そしてどこかに故障が発生すれば、密な結合をしているが故に、全ライン停止するようになっていたとしよう。この場合、現場のリミットスイッチかセンサのどれかが誤動作したとし、それも、その誤動作が少しの時間内に復帰してしまったような場合には、ラインが止ったけれども原因がわからず、手のほどこしようなものもない。又たとえ誤動作し放しになっていてもそれを見つけることは大へんである。しかしこれをコンピューターで制御していたとすれば、制御のステップがどこまで進んでいたか、動作の予測と異なる動作等が後で追跡容易につくることができる。即ち、記憶という機能をもち合わせているものと、そうでないものの差が出るわけで正常な動作以外に対処できる機能、これが裏機能であるが、この裏機能がすべて団子になっているのではなくて、裏機能を分類して処置できるわけである。勿論この場合、CPU（中央演算処理装置）は故障してないことを条件とすることは言うまでもない。

時間あたり、数10万円稼いでくれるラインは、世に一ばいある。これを復旧する時間を短縮することは、正に「時は金なり」である。

その意味でも、このような裏機能を必要なだけ付け加えても、大きなコストアップ要因にならない場合が多いマイクロコンピュータシステムは、これからの制御装置であろう。

## (2) 自己診断機能

従来の制御装置で弱いひとつに 自己診断機能がある。コンピュータシステムでは、デラックスなオンラインダイアグノスティックプログラムから小じんまりしたものまで、程度に応じて種々の診断プログラムをロードして、自己診断を容易にすることができる。

コンピュータという機械は本質的に時分割であるので、自己診断は処理時間のすきままで走らせておくことができる意味で、いかようにもなるが、従来の制御装置では、下手に作れば自己診断用のハードウェアの方が本体より大きくなるようなことがあり得る。

## (3) 予測、学習制御

従来の制御装置は、演算、記憶といった機能が殆んどないため、演算して予測する制御やら過去のデータを処理して、それをこれからの制御のパラメーターとして使う学習制御と言った制御のスペンを拡げることには全くお手上げである。

しかしこれからの制御機能には、是非共こうした機能を入れたくなるのではなかろうか。予測、学習制御のアルゴリズム（処理論理）がはっきりすれば、多分マイクロコンピュータでもかなりのことがやれるのではないか。

## (4) 保守

あるユーザの技術者が、「当社の生産設備に、多くのエレクトロニクスを応用した制御装置を採用した。数年かけて増設をやり、やっと完成して気がついてみたら、確かに運転の省力化は見事に達成したが、いつのまにか、制御装置の保守業務に携わる人がこんなにもいるかと思う程の大部隊になっていた。これからの省力化は、管理、運転、保守すべてを含め、バランスのよいやり方をしたい。」と言っていた。

技術の進歩が著しい現在、前に設備した装置・機器類を次の増設に

も採用するかどうかはわからない。多分必ずといっていい程、既設備の反省を含めて、改良、改善を行なうだろう。そうすると制御装置も、前のものと顔、中味共に異なったものになるだろうし、メーカーも競争力をつけるために、モデルチェンジ的な改善を行なっている。又、あるユーザーのコンペティターである別のユーザーは、コンペティターがやっているそのままを受け入れようとはしない。実績をきわめて重くみる反面、実績通りにやろうとは思っていないのがユーザーである。このような理由が重なって、極めてバラエティーに富んだ各種制御装置がずらりと並ぶ。これでは保守員が沢山必要で、しかもその保守員に高度な技術能力を要求せねばならない。

マイクロコンピュータ制御システムでは、少くともハードウェアそのものは、整理されて大体同じか又は似た顔と中味をもっている。種々のバラエティーは、おおかたソフトウェアでカバーできる。ソフトウェアが100%の正当性をもっていれば、経年変化で、いつのまにかソフトウェアは自身が変わってしまうなどということはありませんので、後はハードウェアの保守だけになるのが普通である。そのハードウェアは定形化され、或る程度の自己診断機能を持たすこともできるから、保守業務が格段楽になることは当然期待してよい。

(5) トータルシステム化(ホストコンピュータとの接続)

これからの社会の中で、少し複雑な設備やシステムは必ずコンピュータ化されるだろうことは、制御技術者であれば誰でも感じているであろう。ところが、従来の制御装置は、上位にコンピュータが据わることを意識して製作されていない。たとえそのことが仕様にならわれていたとしても、配慮をどこまでやっておくかについては、そのレベルがまちまちなことが多い。従って従来の制御装置を相手にしてコンピュータ化を進めようと思うと、条件整備に気の遠くなるような金と時間を必要とする場合が多い。

マイクロコンピュータ制御システムがもし採用されれば、この制御システム自身、ホストコンピュータに必要なデータの殆んどは、そのメモリの中にデジタルで貯わえられている。従って、ホストコ



ンピューターとの間に簡単な通信機能をもたせれば、ホストコンピューターによる管理システムは極めて容易に組み上がる。しかも、その通信内容をすべて或る種の標準コード（例えばASCIIコード）で表現できるようにしておけば、お互いの切口の標準化になって、トータルシステムが極めてつくり易くなる。

即ち、マイクロコンピューターシステムはホストコンピューターから見れば、そのコンピューターの端末装置（ターミナル）に見え、制御対象機器側から見れば、コントローラーであるように製作し得る。

## 2.5.2. マイクロコンピューター制御システムの狙い

2.5.1.に述べたように、従来の制御装置には、望むべくして望み得ない限界があった。しかし、マイクロコンピューターの制御装置への応用は、この限界をとり除いてここに大きな飛躍がでてきたわけである。その意味でマイクロコンピューター制御システムは、前にくどくど述べた種々のねらいと効用があつて、制御技術に携わる人には既にその意義を充分おわかりいただけたと思う。

従つて、ここに若干見方を変えて、補足的な意味でそのねらいを述べたい。

### (1) 制御装置の体質改善……………創造的制御装置

限界の天井が取り払われて、そこから青空が見えてきた。しかし、無限の可能性があるとも思えない。恐らくどこかで、又限界がくるかもしれない。

けれども、ここで制御装置は新しい武器を得たわけであるから、制御対象機器である各種機械の制御の目的、あるいは、制御装置を含めてその機械系のあり方といった原点にもどつて、根本から見直してよいのではなからうか。こうすることにより、従来の制御装置では、当然これくらいしかできなかったから機械もこういうつくりにするといった固定概念がいつのまにか常識化して、そういうものなのだと思ひ込んでいる面がないとはいえないのではないか。

こうしたことを考えていると、マイクロコンピューター制御システムは1つの革新であると同時に創造であつて、このような革新的な思

考が大きく制御装置のイメージを変え、又、体質が変わっていくのではないと思われる。

## (2) グローバルな標準化

制御装置ぐらい、メーカーによりバラエティーのあるものも少ないのではないかと考えている。勿論、一程の技術の流行的な風潮や、世界のリーダーであるメーカーの技術開発動向によって左右される面があって、大勢的には似たり寄ったりの面もある。

しかし、これから、マイクロコンピュータを制御装置の主要なコンポーネントとして使うようになれば、少なくともハードウェアはかなり似たものになる、これは当然世界的な広がりを持ったもので、グローバルな標準化につながっていく可能性を持つのではなかろうか。勿論、その間には種々な紆余曲折はあると思われるが、人間の知恵は、神が知っている限界の範囲の中では無限と言えるので期待したい。

## (3) 各種設備計画の容易化

前にも述べた通り、ハードウェアが定形化するために、従来設備計画面でなかなか予測の難しい制御機器の計画が非常にやり易くなる。主として大きさを決めるのは、制御機能分割の問題と入出力点数である。この概数を予測すれば、大きさや重量や床面積、これに伴う設置場所、付帯装置等が決まってきて、設備計画面が大巾に楽になる可能性がある。

## (4) 接続部の減少による、信頼度向上と工数削減によるコストリダクション

恐らくマイクロコンピュータのチップレベルでの価格は、これから急激に数量が増せば下がってくることが期待できる。しかし人件費はこれからも上昇していくと思われるので、従来の制御のように接続点数が多く、これが殆んど一品料理的であるとすれば、設計製作における工数は、だんだん高なものになることが必定である。又、接続点多ければそれだけ信頼度的に問題である。

これらの問題を、マイクロコンピュータの利用により相当程度改善できる可能性がある。

#### (5) 社会構造とのマッチング

現在の日本の社会は、若い人10人のうち4人は大学出といわれるような、高学歴社会となっている。或る精密機械加工を主体にした中小企業の社長は、中学、高校出の人の採用が思うようにいかないので、発想を変えて大学出を募集したところ、比較的優秀な学生が何人か採用できた。それと同時に、今まであった工作機械の大半をN/C 工作機械に変えた。こうすることにより、今まで機械工に指示して切削していた作業を、この大学出身者に片がわりさせたわけである。N/C 工作機になれば切削加工をする作業はすべて機械がやり、その指令はN/C テープで行なわれる。従って、「この切削加工をせよ」ということは実際の作業としては、「N/C テープを作成せよ」ということと同じで、これならば大学出の機械技術者の仕事として適当である。

この例と同じように、高学歴社会には、それにマッチした作業形態が必要である。今まで電線でコンポーネントを接続して機能を出すような設計、製作作業を、もっと知識集約化してやらなければならないわけで、この意味でも、マイクロコンピュータ制御システムの採用は、高学歴社会のニーズに充分マッチしたやり方だと思う。

#### 2.5.3. 将来の制御装置のイメージ

##### (1) エネルギー系、情報系機器の方向

前にも述べたように、和年30年後半に集積回路ICが世にでた。そしてその時点で将来のコンポーネントの方向が、機能部品化複合化の方向であることが示された。その後、半導体回路部品は急激に集積度を増し、今や10進4桁の大台にのせようとしている。

一方、例えば、東名高速道路を東京から西へ向い、富士を右手に見て走ることしばし、富士宮インターチェンジで高速を降りて約15分東京電力株式会社富士根変電所に到着する。この変電所は、66KVで受電、6・6KVで配電する配電用変電所であるが、変電所と言われなければおよそ変電所だとは誰も気がつかない。何か小さな倉庫があって、プレハブの小さな事務所があるといった感じである。その小さ

な倉庫のようなものが、実は変電所の主回路機器をすべて包含している。今までのこのような変電所は、66KVの受電用のラインスイッチ、遮断器、変圧器、避雷器、等、主回路の機器は、鉄構物で支えられて現場でそれぞれ据え付けられるのが通常であるが、この変電所はこれらの機器はすべて一体化され、油の中に浸っている。これにより、大きさは床面積にして従来の変電設備の収納面積のおよそ1/5、しかも騒音対策や環境の景観に対する配慮などがよく考えられている。当然現場での据え付け工事の工事量は大巾に減っている。又、6・6KV側の配電線の主回路機器や保護用機器は、固体絶縁を採用していて、従来の設備とくらべ、床面積で1/3、容積で1/6になっている。即ち、これらのエネルギー機器も、今や機能モジュール化、複合化の道を歩んでいることを証明している。差し当たり、この変電設備は重電機器のICといえる。

このように、エネルギー機器も情報機器も、同じような機能部品化、複合化の道を歩んでいることは、まことに興味深いことであると同時に、これからのすべてのコンポーネントの開発方向を示唆しているものといえる。

## (2) ブラックボックスコントローラ

制御装置も又、必ずやこうした集積化が進み、ソフトウェアそれ自身をも含んだ形で、ブラックボックスコントローラと言った方向に進み、モジュール化されたエネルギー機器とブラックボックスコントローラが複合化して、さらにより高い機能化が行なわれていくのでなかろうか。

しかしこのような高機能化を進めるためには、まだまだ、マイクロコンピュータ関係の技術も進歩していかなければならない。

差し当たり解決していかなければならないのは、熱の問題と、雑音の問題で、温室育ちの清浄野菜的な制御装置から、暴風雨にも負けない灌木のような強さを持った制御装置が必要になるのではなかろうか。

## 2.6. おわりに

マイクロコンピュータの出現により、制御技術者が、なぜ発想の転

換をせまられているのかについて、技術的な、又社会的な背景を中心に述べたが、前にもおことわりしたように、極めて主観的な見解も多く、独善的であったかもしれない。しかし、そんな考え方もあるのかというふうに受けとっていただいて、一人一人の制御技術者がこれからの制御技術について新たな発想のほんの刺激になれば、筆者の幸い、これに過ぎるものはないと考えている。

## 第3章 制御における市場性、経済性、信頼性などの諸問題

### 3.1 はじめに

最近、我国でも急激に普及しようとしているマイクロプロセッサおよびマイクロコンピュータは、きわめて広い応用分野が予想される。即ち既存のいろいろな機種分野にとり入れられて従来の性能を一段と向上させるとともに、融通性を増加させたい、自己判断能力を与えたいなど一段と人間に近い仕事まで機械にまかせる希望を可能にするとともに従来考えていなかった新しき製品、新しい使い方が開けて、従来技術にインパクトを与えるだろうといわれている。

今回の応用篇ではその広範にわたる応用分野のうち、現在わかっている範囲で最も大きく従来技術に革新をひきおこすと見られる制御技術を取り上げている。本第三章ではその従来技術から移行するときの問題点、マイクロプロセッサの特質、市場性と経済性を見きわめなどについて現状を概観して何等かの指針を見つけ出したいと試みたものである。

尚、今後資料としていろいろに役立てられるように、なるべく図、表を中心にまとめているので、参考として活用して載きたい。

### 3.2 マーケット調査の概況

#### 3.2.1 生い立ちと急速な成長

電子計算機がつくられ、販売されるようになってから僅か30年、その間の技術進歩と普及はまことに目覚ましいものがあるが、途中で3回の世代の革新的交代を経てきている。はじめの第1世代から第2世代は真空管からトランジスタへの交代であり、第2世代から第3世代はトランジスタからICへであった。いづれも使用されている素子の技術革新を中心としたもので、その都度進歩の歩みを急に一段階づつ飛躍する形で性能、価格に進歩をもたらせてきた。就中第2世代トランジスタから第3世代IC時代への交代にあたっては、大型機の進歩のほかに、ミニコンと称する新しい機種群が登場して、その使い易さ、手軽さが従来よりも一段と市場を広げ普及に一段と貢献したことは記憶に新しいことと思

う。そして今ふたたび I C から L S I への時代を迎へ、今度はミニコンよりも更に一段と革新性が強く、新しい使い方の可能性を内蔵したマイクロコンピュータが登場してきた。

その誕生は、当協会の会員である精工社が電卓の演算回路にプログラム制御部を含め、チップ化する工夫を米国インテル社に具体化させた事に始まるとの事で僅か4年前の事である。はじめの目標は電卓の高性能化、メモリー制御を持った使い易い卓上高級計算機であったマイクロコンピュータも、ストアプログラム方式の制御を自蔵し、実行する命令のレパートリーも通常の小型電算機と同じくらい範囲が広がることによって、その本質が単純な“計算を自動的に行なわせる機械”から、より広範な応用を指向した“プログラムの論理を制御する汎用のデジタル型制御装置”とも言うべきものに転換しつつあるこのごろである。その間の普及、進歩は僅かの期間であるにもかかわらず極めて急激で、主として米国を中心に性能の向上、種類の多彩化、量産化が進んでいる。

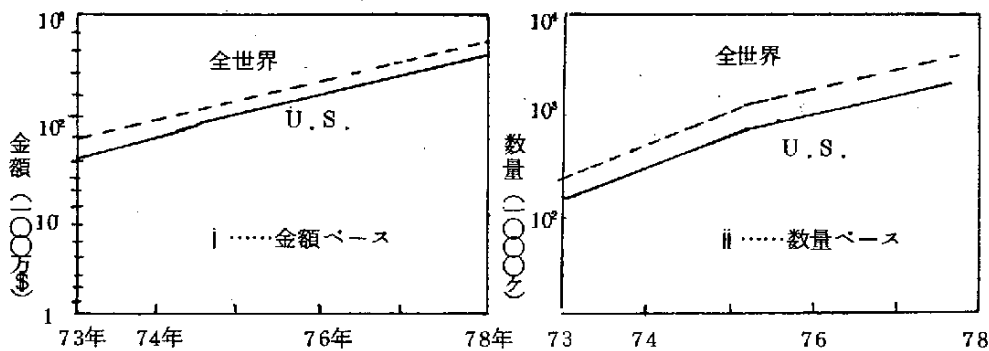
新しい製品が世に生れ、需要にマッチし、商品化に成功し普及するのは、研究段階で試作されるものの僅か3パーセントであると言われる。僅かであるかわりにその初期のマーケットの伸びは常に著しい比率を示すのが常である。これを生命にたとえ、その発達普及の段階を、幼年期、少年期、青年期、壮年期、老年期に分けると青年期迄の歩みは技術進歩、生産量の伸び、価格の低減共に著しく、最も盛んに使われる壮年期に続くと言える。そしてマイクロプロセッサおよびその関連 L S I 製品は 1950 年現在幼年期から少年期へ移りつつある段階にある。

このために未だ若い製品としてそのマーケット調査は実績の統計データがベースになり難く、主として米国の個別に行なわれている民間調査機関によるデータが殆どである。第3-1表および第3-1図にその代表例をいくつか示す。

| 例 | 調査機関             | 1974年 | 1976年 | 1978年  | 1980年  |
|---|------------------|-------|-------|--------|--------|
| 1 | C. S. I.         | 200億  | 600億  | 1200億  | 2300億  |
| 2 | G. C. I.         | 280 " | 487 " | 1458 " | 2250 " |
| 3 | I. D. C.         | 180 " | 540 " | 1100 " | 2100 " |
| 4 | Electronics<br>誌 | 100 " | 400 " | —      | 2400 " |

単位：円換算

第3-1表 米国マイクロコンピュータ市場調査例  
(含メモリ、その他補助回路)



第3-1図 マイクロプロセッサの市場動向

第3-1表の例1はREATIVE STRATEGIES INC.に当協会の依頼した調査、例2は電子工業振興協会がGNOSTIC CONCEPT INC.に依頼した調査、例3は、我国各半導体メーカー協同でINTERNATIONAL DATACORP.に依頼した調査報告、例4はよく引用される公刊誌データーであり、第3-1図は例2の一部である。

最近経済全般が低成長時代に移行すると言われて居り、各種の製品分野にも此のように高い成長が予測される製品に乏しくなっているために、数少ない稀少価値のある珍しい事業として、広い関心を集めている現状である。

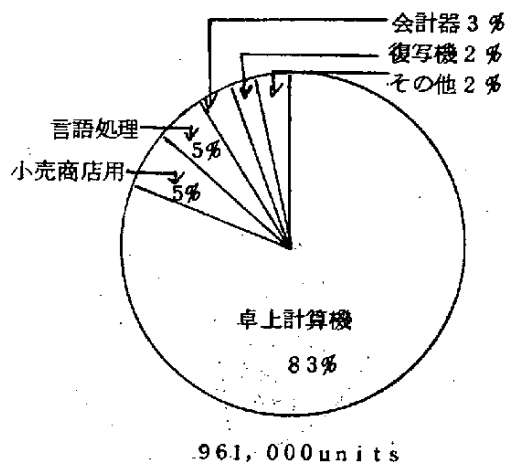
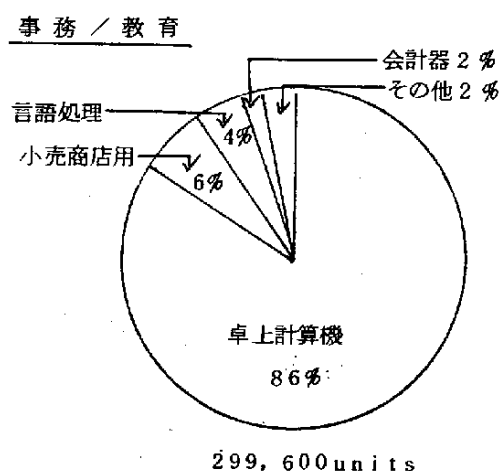
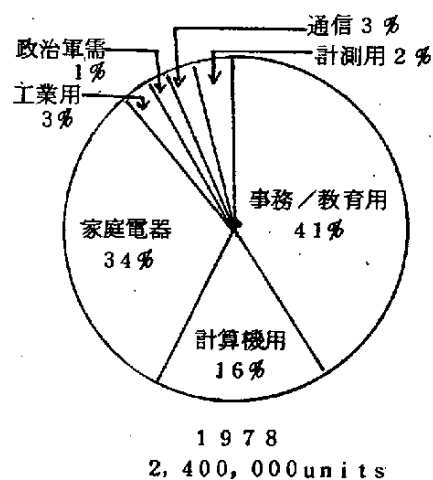
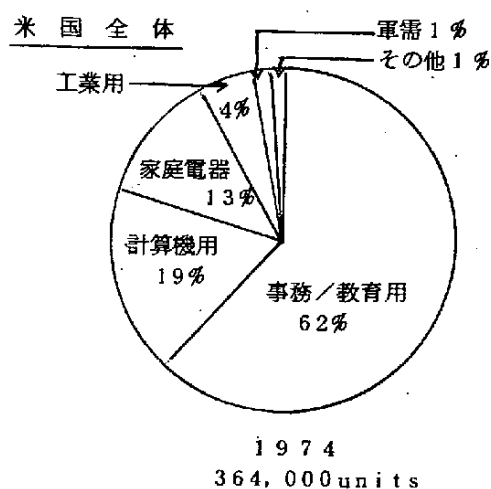
このことは、ミニコンが出現したときのブームに似たところがあるため、果して何の規模の波及効果を及ぼすか、大は産業構造の変革に迄進



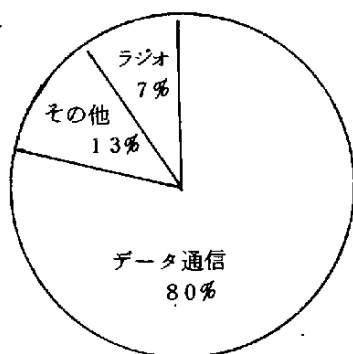
むという見方から、小は従来の電算機の普及の加速にプラスアルファ程度という見方に迄分かれている。そして米国の場合、次第に強気に近い予測が主流を占めつつあるのが現況である。第3-1表に示すように今後5年間に5倍以上を一致して予測している事は、このことを物語っているといえよう。

### 3.2.2 制御への応用の市場

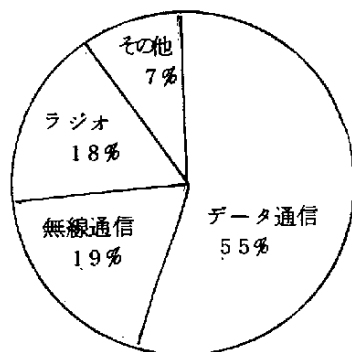
各種の応用分野別分類について1974年、および78年について第3-2図の報告がG. C. I. から出されている。



通 信

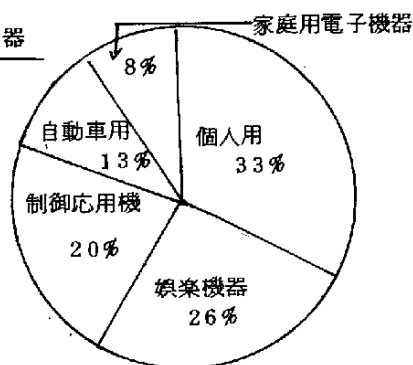


1,500 Units

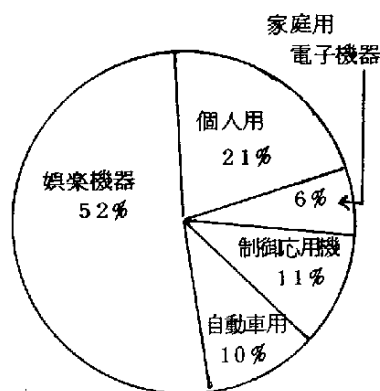


72,800 Units

家 庭 電 器

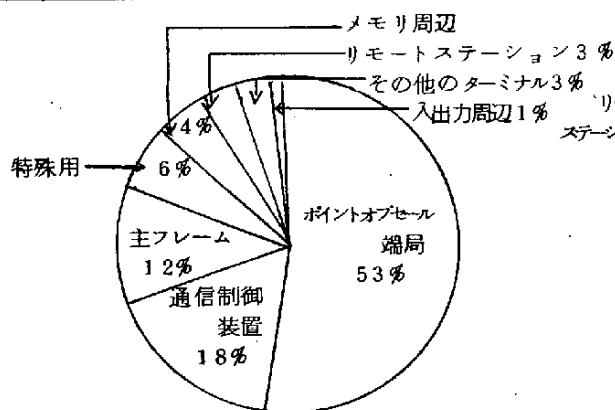


47,400 Units

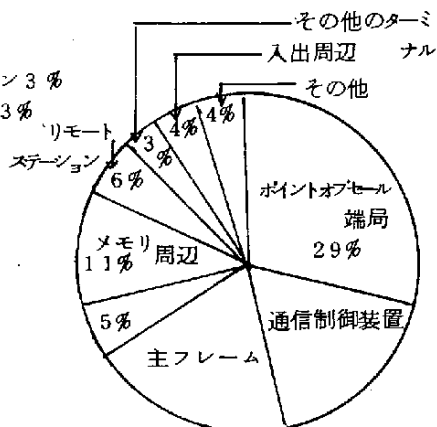


843,200 Units

計 算 機

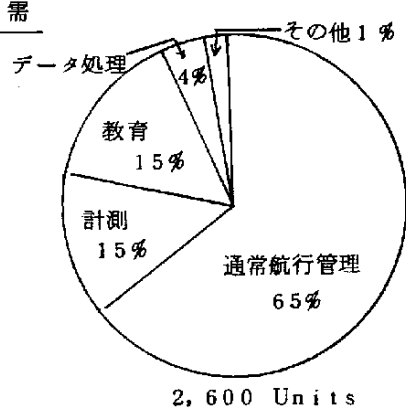


68,200 Units

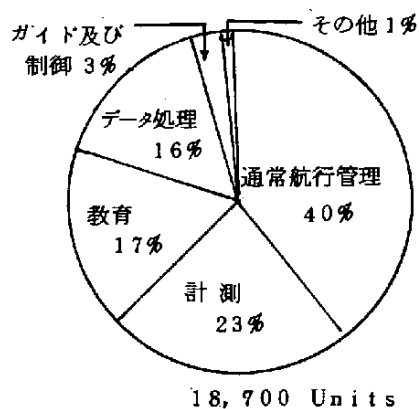


373,300 Units

軍 需

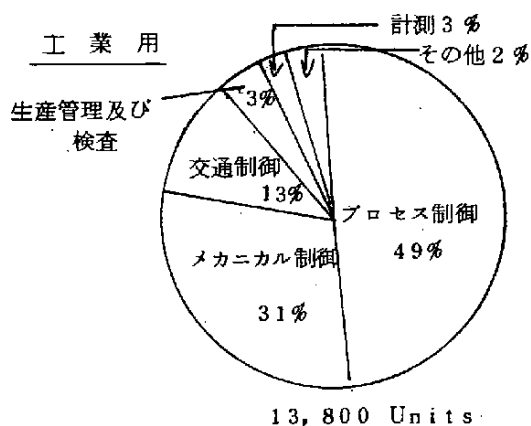


2,600 Units

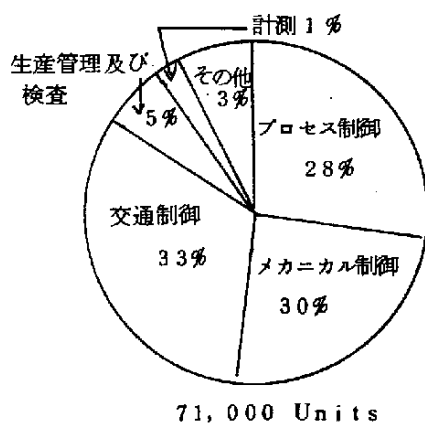


18,700 Units

工 業 用

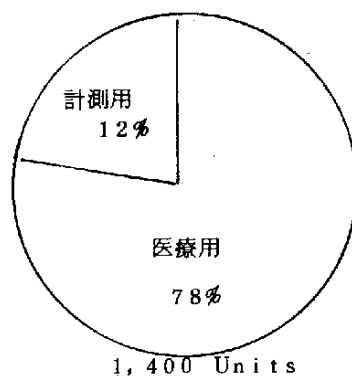


13,800 Units

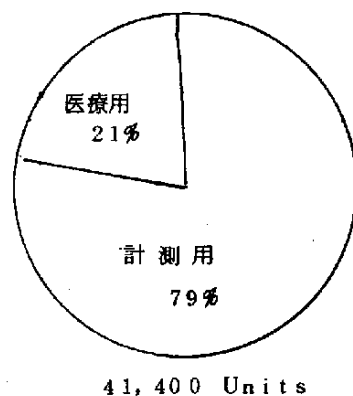


71,000 Units

計測用 / 医療用



1,400 Units



41,400 Units

第3-2図 (G. C. I. による)

最上段に全体の分類がまとめられており、第2段目以降はそれぞれの小分類である。全体分類によると、最も多数使用されるのは事務処理及び教育用で、電算機用、家庭電機用、工事用、軍需用、通信用、その他の順で続くと予測している。

我国の常識的感覚から特に目をひく分野として、事務処理のなかに、教育用を含めているということには重要な意義を認める必要がある。プログラム制御の論理の取り扱いに早くから馴れ、ソフトウェアの概念も教育の早い段階から自然に身につくようにカリキュラムの中に組み入れてゆく米国のあり方には我国の現状と著しく異った特徴がある。今後の社会のための教育内容の例として、将来インテリジェンスを持った人工機械を使う社会の教育のあり方の例として米国で教育用の用途を非常に重視していることに、教えられるものを持っているといえよう。

次の通信事業への応用ではデータ通信の大きな伸びを予想していることは常識的にうなづけると思う。ところがその次の一般消費者用家庭電器の分野で将来娯楽機器に非常に大きな用途を予想していることは我国の目からは予想外と受け取ることと思う。

さて今回の本応用篇の制御への応用として取り上げている範囲を常識的に第2図の工業用と計測用医療用をカバーするところを考えると、マイクロプロセッサ全生産量の数パーセントが使用される分野であって、必ずしも数量的に大きな使用分野ではないといわざるを得ない。

制御への応用は、個別の制御対象に適応して1ケ1ケ異ったソフトウェアを収容して制御対象とマッチングさせて初めて完成する性格が強く、量産ベースで全く同一仕様の製品を製作する分野とは異った特質があるため、数量としての統計上からは比較的小さな分野とされる。この傾向は一般の電算機の場合やミニコンの場合も同様にいえることである。次節の第3-3表に我国のミニコンの応用分野別分類を表示してあるが、これにも同様の傾向が見られる。然し乍ら制御への応用は、使用される用途分野の拡がりや極めて広範で、使用する側から見た重要性は大きく、既存技術に及ぼす影響も波及範囲が広い。この点が今回応用篇の代表テーマに制御をとり上げた意義といえよう。

尚、現時点で、現在迄に行なわれた調査についてさとるべきこととして、何うしても既存分野の分類重点になり、今までの分類ではあまり目立たなかった全く新しい分野が著しい発展を遂げる可能性を充分抱握しているとはいえないことは謙虚に認めておく必要があると思う。若干とはいえこの可能性が最近次第に認められつつある分野は、自動車、列車など移動体への搭載利用、農事用への利用などが例として示される。

### 3.2.3. 我国の現況

最近、我国の状況についても電子工業振興協会によって初めて報告が出された。第3-2表はその一部である。

我国の場合、やはり米国よりは2~3年のおくれがあるようで、特に応用面については開発の初期的傾向が強く、未だ統計資料として本来のマーケット分野の予測するのは危険であるが、広範囲に試用され始めていることは第3-2表に明らかである。

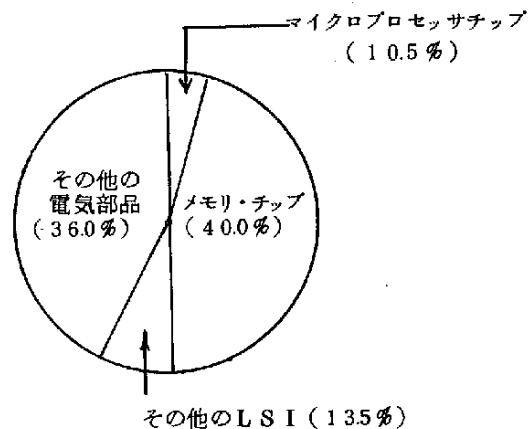
応用分野の分類とその比率について、比較し参考とするベースとして、ミニコンについての資料は略実績が固まり、明らかであるため役に立つので第3-3表に我国のミニコンについてのデーターを添付しておく。本表によるとミニコンの場合も、制御への応用は数量は多くないが広範囲にわたっていることがわかることは先に述べた通りである。

## 3.3 従来の制御との比較

### 3.3.1. プラントの運転、制御と情報の取り扱い

マイクロプロセッサの制御への応用について、その特長、可能性、位置付けを論ずるために、先づプラントの運転、制御に立ち戻って其処に使われる情報というものを見直して見る事が有効である。

| 分野別<br>回答<br>件数  | 交 通 | 専用工作機 | デバイス<br>制 御 | 通 信   | 医 用   | 自 動 車 | 事務器<br>販売器 | 光 学 | 計 測   | その他   | 平 均     |
|------------------|-----|-------|-------------|-------|-------|-------|------------|-----|-------|-------|---------|
|                  | 1 件 | 1 件   | 29 件        | 9 件   | 4 件   | 0     | 19 件       | 0   | 14 件  | 9 件   | 90 件    |
| マイクロプロセッサ<br>チップ | 6   | 7.8   | 9.22        | 9.61  | 14    |       | 6.58       |     | 11.64 | 4.67  | 10.5(%) |
| メモリ・チップ          | 24  | 33.4  | 32.01       | 34.78 | 22.25 |       | 18.26      |     | 30.4  | 27.22 | 40.0(%) |
| 上記以外のLSI         | 0   | 12    | 13.25       | 20.33 | 7.5   |       | 6.82       |     | 8.64  | 10.56 | 13.5(%) |
| その他の電気部品         | 70  | 39.2  | 39.78       | 31.83 | 51.25 |       | 45.32      |     | 44.79 | 32.22 | 36.0(%) |



第3-2表 電振協で行ったアンケート調査による応用分野別パーセンテージ (S49年)

| 用途別<br>産業別 | その他<br>(不明) | 研究   | 技術<br>設計 | 教育<br>(含<br>CAI) | 通信<br>制御<br>(ネット<br>ワーク) | 交通<br>管制 | 周辺<br>機器<br>制御 | 医療<br>用 | プロ<br>セス<br>制御 | 生産<br>ライン<br>管理 | 計測<br>分析 | その他<br>機械<br>制御 | その他<br>システム<br>制御 | その他 | 不明   | 合計    |
|------------|-------------|------|----------|------------------|--------------------------|----------|----------------|---------|----------------|-----------------|----------|-----------------|-------------------|-----|------|-------|
| 農 業        |             |      |          |                  |                          |          |                |         |                |                 |          |                 |                   |     |      |       |
| 林・野山業      |             |      |          |                  |                          |          |                |         |                |                 |          |                 |                   |     |      |       |
| 漁・水産・畜産業   |             |      |          |                  |                          |          |                |         |                |                 |          |                 |                   |     |      |       |
| 鉱 業        |             |      |          |                  |                          |          |                |         | 1              |                 |          |                 |                   |     |      | 1     |
| 建設業        | 8           | 1    | 7        |                  | 2                        |          |                |         |                | 3               | 2        |                 | 2                 |     | 1    | 26    |
| 食品業        | 1           |      |          |                  |                          |          |                |         | 5              | 1               | 5        |                 | 1                 |     |      | 13    |
| 化学工業       | 5           |      | 3        |                  |                          |          | 1              |         |                | 2               | 4        |                 | 2                 |     | 4    | 21    |
| 紙・パルプ      | 1           |      |          |                  |                          |          |                |         | 1              | 1               | 1        |                 |                   |     |      | 4     |
| 出版・印刷      |             |      | 1        |                  |                          |          | 1              |         |                | 1               |          | 21              | 1                 |     |      | 25    |
| 化学・石油      | 24          | 1    | 2        |                  | 11                       |          |                | 4       | 16             | 5               | 49       | 1               | 1                 |     | 2    | 116   |
| 鉄鋼・セメント    |             |      | 1        |                  | 1                        |          |                |         |                | 3               | 3        |                 |                   |     |      | 8     |
| 鉄 鋼        | 20          | 1    | 1        |                  | 6                        |          | 1              |         | 7              | 32              | 41       |                 | 1                 |     | 7    | 117   |
| 非鉄金属       | 11          | 2    | 1        |                  |                          | 2        |                |         |                | 3               | 8        |                 |                   |     |      | 27    |
| 機械         | 16          | 3    | 3        |                  |                          |          |                |         |                | 5               | 9        | 11              | 4                 | 1   | 6    | 58    |
| 電気機械       | 89          | 180  | 34       | 17               | 27                       | 5        | 13             | 6       | 2              | 112             | 101      | 9               | 1                 | 10  | 70   | 676   |
| 電子コン・ノーク   | 49          | 143  | (24)     | (13)             | (20)                     |          | (6)            | (6)     | (1)            | (73)            | (46)     | (6)             |                   | (5) | (50) | 442   |
| (その他)      | 40          | (37) | (10)     | (4)              | (7)                      | (5)      | (7)            |         | (1)            | (39)            | (55)     | (3)             | (1)               | (5) | (20) | 234   |
| 輸送用機械      | 18          | 7    | 7        |                  | 7                        |          | 1              |         | 3              | 16              | 38       | 5               | 8                 | 1   | 2    | 113   |
| 精密機械       | 10          | 10   | 5        |                  |                          |          | 1              |         |                | 4               | 16       |                 | 2                 |     | 4    | 52    |
| その他製造業     | 5           |      |          |                  |                          |          | 1              |         |                | 1               | 1        |                 |                   |     | 2    | 10    |
| 卸小売・商社     | 20          | 5    | 4        | 3                | 18                       |          | 17             |         | 3              |                 | 4        | 1               |                   | 1   | 9    | 85    |
| 金融・保険業     | 12          |      |          |                  | 2                        |          |                |         |                |                 | 1        | 1               |                   |     |      | 16    |
| 不動産業       | 1           |      | 2        |                  |                          |          |                |         |                |                 |          |                 |                   |     |      | 3     |
| 運輸・通信業     | 12          | 1    | 1        | 1                | 9                        |          |                |         |                | 2               | 10       | 12              | 15                | 2   | 5    | 70    |
| 電気・ガス業     | 10          | 1    |          | 1                | 1                        |          |                |         | 3              |                 | 7        | 4               | 2                 |     | 1    | 30    |
| サービス業      | 42          | 10   | 14       | 5                | 1                        |          |                |         | 2              |                 | 2        | 2               | 6                 | 10  | 2    | 96    |
| 医 療 業      | 2           | 3    |          |                  |                          |          |                | 11      |                |                 | 1        |                 |                   |     |      | 17    |
| 大 学        | 86          | 123  | 7        | 47               | 9                        |          | 5              | 26      | 1              | 9               | 74       | 4               | 1                 | 2   | 8    | 402   |
| 高 校        | 3           | 4    |          | 67               |                          |          |                |         |                |                 |          | 5               |                   | 1   |      | 80    |
| その他学校      | 2           | 1    |          | 7                |                          |          |                |         |                |                 |          |                 |                   |     |      | 10    |
| 地方公共団体     | 8           | 3    | 3        | 2                |                          | 10       |                | 3       | 3              | 1               | 20       | 1               | 1                 | 3   | 1    | 59    |
| 政 府        | 34          | 11   | 2        | 1                | 7                        | 2        | 4              | 3       | 1              | 10              | 22       | 7               | 1                 | 5   | 4    | 113   |
| 政庁関係機関     | 50          | 5    |          | 2                | 8                        |          | 3              |         | 1              | 3               | 16       | 4               | 3                 |     | 2    | 97    |
| 法人・団体・個人   | 12          | 1    |          |                  | 5                        |          |                | 3       | 2              |                 | 4        |                 | 7                 | 1   | 2    | 37    |
| 宗 教 法 人    | 1           |      |          |                  |                          |          |                |         |                |                 | 1        |                 |                   |     |      | 2     |
| 民間研究機関     | 4           | 9    | 3        |                  |                          |          |                | 2       |                |                 | 10       | 2               |                   | 3   |      | 33    |
| 軍 隊        | 156         | 60   | 21       | 2                | 7                        | 6        | 16             | 16      | 6              | 12              | 126      | 36              | 19                | 6   | 59   | 548   |
| 合 計        | 662         | 442  | 122      | 155              | 121                      | 25       | 64             | 74      | 57             | 226             | 576      | 126             | 78                | 46  | 191  | 2,963 |

(単位：台数)

第 3 - 3 表 47 年 3 月末現在産業別ミニコンピュータ使用状況と用途との関係

| 役 割                             | 主 な 仕 事 の 順 位 | 従 たる 仕 事 の 順 位 |
|---------------------------------|---------------|----------------|
| プ ロ セ ス モ ニ タ リ ン グ             | 5             | 2              |
| デ ー タ 収 集                       | 3             | 1              |
| オ ペ レ ー タ ー ガ イ ダ ン ス           | 2             | 4              |
| メ カ ニ カ ル コ ン ト ロ ー ル           | 9             | 1 1            |
| イ ン ス ツ ル メ ン ト コ ン ト ロ ー ル     | 8             | 6              |
| シ ー ケ ン ス コ ン ト ロ ー ル           | 6             | 5              |
| ダ イ レ ク ト デ ィ ジ タ ル コ ン ト ロ ー ル | 4             | 8              |
| ス ー パ ー バ イ ザ リ コ ン ト ロ ー ル     | 1             | 1 0            |
| 管 理 報 告                         | 7             | 3              |
| 工 程 管 理                         | 1 1           | 7              |
| ス ケ ジ ュ ー リ ン グ                 | 1 0           | 9              |
| そ の 他                           | 1 2           | 1 2            |

電子協報告 47-A-56

工業用ソフトウェア専門要報告より

第3-4表 プラントの運動制御におけるデータの使われ方

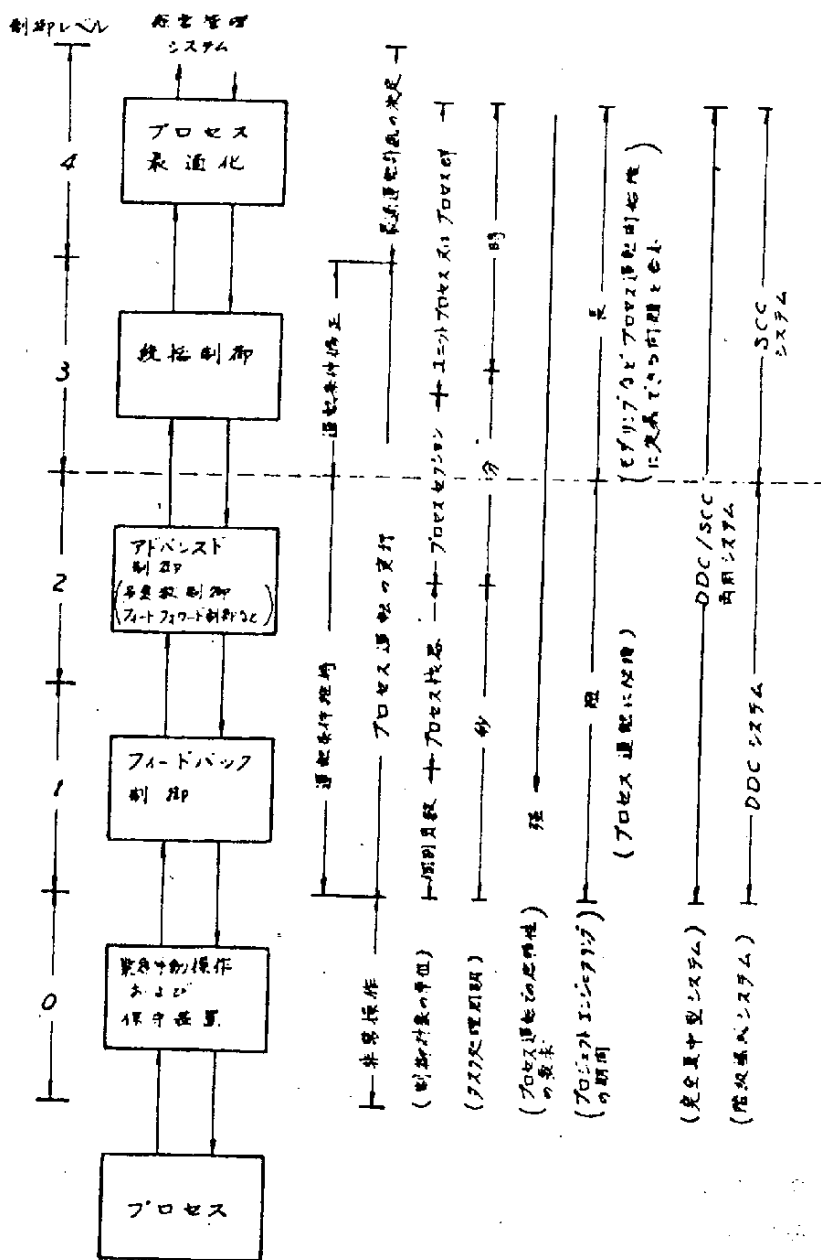
第3-4表は一般の電算機についての調査であるが、情報が何んな使  
わた方をしているかという面から見ることも可能であろう。此処では同  
じ情報が主たる用途、従たる用途として複数の用途に使われる可能性を  
もうかがわれる。

プロセス自動化、プラントオートメーションの進歩は、今世紀わが国  
産業進歩の中核技術として、最近迄の高度成長を支えて来たが、その歩  
みは一貫して、個別に分散した機械のそばの分散した運転、制御から単  
位工程毎の集中化した運転へ、更に一貫した生産の流れに乗った工程ラ  
インを全般的に通した運転、制御へと集中化を進めて来た。

このことは必然的に個別プロセス機器に最も密接した制御の段階から、  
全体的管理までその内容、レベルに応じて役割が第3-3図のように階  
層構造を形づくってきている。

即ち最もプロセスそのものに密着していなければならないのは異常時の保安処  
置、安全制御ともいうべき緊急操作である。次には平常時にプラントを、運転し  
ようとする目的に応じた一定の値に安定に継続させるフィードバック制御が位置  
を占める。これにも流量制御、圧力制御や、機械的制御のように個別の





第3-3図 プロセス制御の階層構造

状態量を直接制御するものから、時間おくれの大きな温度の制御、判断に時間のおくれを生む分析結果にもとづく制御などのように、徐々にプロセスの動きから距離を置いて全体的なとらえ方をする制御へと層状になっている。此の詳細は才1章の西川教授の分類をよく読んで載きたい。

此の全体的階層構造に応じて制御装置自身もその役割毎に機能を分割し、階層構造を持つように組み立てられる必然性がある。その中を一貫してデータが自由に適時適切な判断の材料を提供できるように流れなければならない。このような全体のシステムの中に、小まわりを効かせてマイクロプロセッサが貢献すべき可能性のある仕事は多数考えられる。その基本的な役割、使い方の着眼点は才3.5節に紹介するが、最も基本的な才3-3図の0ないし1の制御レベルの役割は、正にマイクロプロセッサの特徴を生かしたマイクロコンピュータ、制御に好適なものであるということができる。

さて才3-3図構造図、才3-4表のその中でいろいろに使われるデータの分類をまとめると結局、プラントの運転制御のために使われるデータは、3種類に分類される。即ち1-プロセスの監視用、2-記録としてそれ自身必要なもの、3-直接制御に使われるもの、の3種類で才3-4図のようにまとめられる。マイクロプロセッサは後述のようにこのそれぞれに特長を生かして使われる可能性があるがここでは3の直接制御に使われるものに限定して、従来此の範囲を分担して来たワイヤドロジックの制御と、コンピュータ、制御についても主として此の範囲を対象としている場合について比較して見ることにする。

### 3.3.2. コンピュータ制御とマイクロプロセッサ制御

一般のコンピュータもマイクロプロセッサもストアドプログラムの制御を実行するという点が共通である。従つてその適用を計画し設計する段階でコンピュータの中でも特にミニコンの使い方の中には直接共通性があり役に立つ場合が多いが、尚マイクロプロセッサの特長を生かした使い方、マイクロコンピュータ独特の長所を発揮する可能性という点に大きな期待が持たれている。そして従来のコンピュータ制御の限界、矛盾点を解決する使い方とその本領発揮の可能性が大きい。

| No. | 役 割                              | い ろ い ろ の 形                                                                                                                                                                                                                                                                                                            |
|-----|----------------------------------|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| 1   | プロセスの監視<br>Monitoring & Alarming | <p>プロセス状態を周期的に監視する。</p> <p>(レベル1) - 緊急処置対策用；異常があれば急いで安全処置をとるべき信号</p> <p>(レベル2) - 運転状態維持用；異常があれば運転状況を改善すべき信号</p> <p>情報量は最も多い</p> <p>どちらも異常がなければ忘れてよい</p> <p>それぞれ           <ul style="list-style-type: none"> <li>・異常時の制御</li> <li>・人に知らせる方法</li> <li>・記録の要るものと</li> <li>・要らない信号</li> </ul>           種類あり         </p> |
| 2   | 記 録<br>Data Logging              | <p>情報量は中間</p> <ul style="list-style-type: none"> <li>・運転改善データ用</li> <li>・運転状態維持用</li> <li>・生産記録、管理用</li> <li>・直接的運転台指針用</li> <li>・設備補修データ用</li> </ul>                                                                                                                                                                  |
| 3   | 制 御<br>Control                   | <p>情報量は1, 2より少ない。</p> <p>閉ループ制御とオープンループ<br/>PID制御とシーケンス制御<br/>線形と非線形<br/>単純ループと多変数<br/>定常態と過渡状態</p> <p>それぞれ分類できる場合と複雑に組合せの場合あり。</p>                                                                                                                                                                                    |

才 3-4 プラントの運転制御に使われる情報のいろいろ

前節に述べたように最近迄のプロセスオートメーションの道は、中央での集中制御への道という形で、情報の集中化を伴って進められ、その情報処理の担い手としてコンピュータ制御を備えることによつて高度なシステム化が大いに進んだ。

その規模は進歩すればする程大きくなり、才3-4図に分類して示した各種情報量も増大の一途をたどってきたのはむしろ当然であるが、そのための電算機は次第に大型、複雑なものとなってきた。才3-4図の情報そのものは共通の信号が多いので、データベースをプロセスに導入された電算機メモリのファイルとすることは極めて合理的といえるが、

いろいろな使い方をするということは、そのため電算機の仕事としてまかせれば益々処理が複雑化するという他に他ならず、コンピュータ制御システムとしての電算機の応答速度、信頼性、保守性などに限界にぶつかる点が出て来始めた。その間に割込処理の方法、多重処理の方法、タイムシェアリングのオペレーティングシステムの充実など、汎用のオンライン電算機システムの技術進歩に貢献した種々の技術的金字塔を建てて来たといえるが、その反面主として下記の3つの問題を提起して反省を求めることになりつつある。

#### (1) 応答性について

ソフトウェアの複雑化と共に、その全体を管理するオペレーティングシステムが重くなり、実行制御を司るプログラム自身のオーバーヘッドタイムが増大して来て、実際に必要とする機能を電算機に実行するためのプログラムの処理速度に近い能率で現実プロセスを制御することは困難な場合が次第に出て来たこと。このことは電算機本体の性能を向上させ速度を上げて単位時間の処理量を増加させるだけでは不可で、その高速性を充分発揮させるように早く仕事にとりかかれるような構造にする必要を生み出し始めている。

#### (2) 信頼性について

コンピュータ制御システムの規模の増加は次第に信頼性の限界にぶつかる場合を増加させている。プロセスの信号に数千点の入力を持つ最近の通常のケースでは稼働率は99.5%程度というのが平均した水準である。電算機に処理させる仕事のうちに此のレベルに見合った信頼性で充分な機能もあるが、とうてい不十分な機能もあることは才3-4図の分類からいい得ることである。その克服のために二重系の採用、予備部分の常時接結による待機などの冗長性を持たせる方法が屢々採用されるが、このことはコストアップのみでなく再びシステムを複雑化させて上記1項の問題点にはね返る結果になることがある。

#### (3) 保守性について

設置頭初の試運転段階での調査、ソフトウェアの修正や、運転経験を重ねることによる半永久的に続く制御アルゴリズムの改善などの修正や

変更、設備の増設その他による修正などコンピュータ、制御システムは単一機能マシンを離れてシステム化されていけばいる程、稼動開始後の修正変更が併なうのが常であるが、その変更はバッチ処理を原則としている通常の電算機程簡単容易ではない。その理由はオペレーティングシステムとして他の機能プログラムと複雑にからむために、複雑なフラグの受け渡しを常として居り、その修正変更が大きな負担となる、多数の入出力信号のファイルを持つためにこれらのデータとの情報交換が複雑であり変更が容易でない場合がある、オンライン制御を中心とした場合オペレーティングシステムは比較的小さなプログラムを多数組み合わせ処理効率、メモリ効率を巧妙に上昇させている場合が多いが、このような場合は一部の修正改造は全体の見直しになる場合がある、などである。

以上のことが従来電算機に処理させる仕事を増加させることに専念する形で進んで来た進歩の方向に反省を求め、最近再び適切な機能分担による分散化を含めた上でのオ 3-3 図のような階層構造の見直しをする傾向が現れつつある。この場合プロセスに密接した DDC、走査監視などをマイクロコンピュータに一任し、電算機は本来最も得意とする多量のプロセスデータのファイル管理、各種処理に専念することによつて総合的なシステム全体の性能を向上させることは極めて望ましい方向であるといえよう。

此処で新しい階層構成によるマイクロコンピュータ制御とミニコンを含む一般電算機によるコンピュータ制御の分担協調方式を具体的に考える場合の境界を何処に置くべきかという事について、下記の 3 つの機能、

- (1) メモリ修正内容の確認、そのドキュメンテーションをつくること。
- (2) 広範囲のソフトウェア修正とそのテストラン、その結果確認を自分自身でできること。
- (3) 自分自身の内部故障探索のサービス手段を持たせる事を必要とする使用方法、の 3 つのことが出来るような要求のある場合は、マイクロコンピュータを使用するよりも電算機にまかせるのが適当であるといえる。

マイクロコンピュータも将来プロセッサの進歩によりたちまちミニコン並みの性能を持つ可能性はあるが、その場合でもマイクロコンピュータはコントローラに近い専門機の構成とし、リソースを持ちメモリを大きく持ったデータ処理中心のマシンとは異質の特質を生かすようなシステム設計が望ましい形であることは変わらないであろう。

むしろマイクロプロセッサを手足として階層を組んだ電算機が、マイクロプロセッサのバックアップとして、自分の接続しているマイクロプロセッサについての上記3つのチェック機能を持つことが出来ることが、制御への応用を考える者の将来の夢といえるかもしれない。

尚、此処で現在マイクロ対ミニの性能には尚一桁の差があり、少なくとも数年は続くであろうことを示すデータとしてオ3-5表を参考として載せておく。

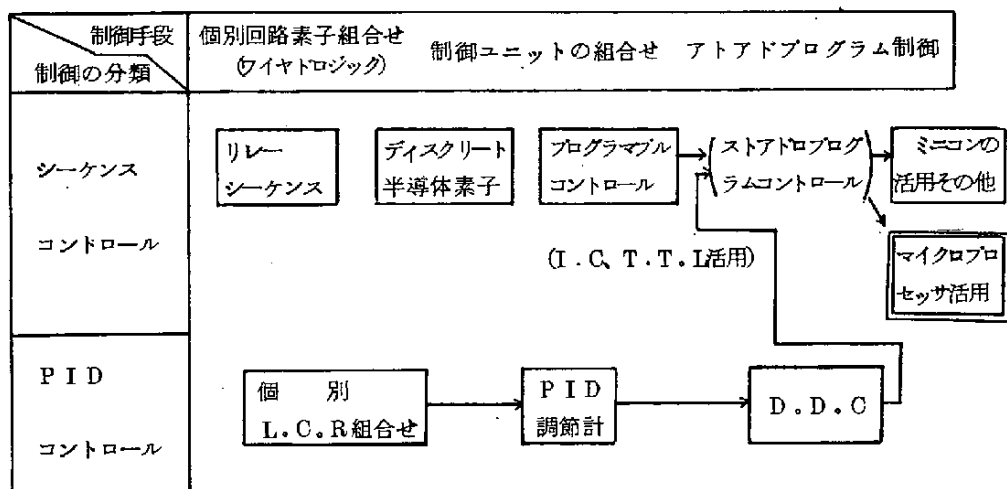
|                | 1974      |          | 1978      |           |
|----------------|-----------|----------|-----------|-----------|
|                | マイクロ      | ミニ       | マイクロ      | ミニ        |
| 実行時間( $\mu$ s) | 25-25.0   | 0.5-2.0  | 0.1-10.0  | 0.1-2.0   |
| 語長(ビット)        | 4-16      | 8-32     | 4-16      | 8-32      |
| 命令数            | < 70      | 100-200  | < 200     | 150-250   |
| 記憶容量(Kビット)     | 8-128     | 128-512  | 8-512     | 128-1024  |
| 記憶素子           | バイポーラ/MOS | コア       | バイポーラ/MOS | バイポーラ/MOS |
| 価格(1000円)      | 0.2-2.5   | 3.0-25.0 | 0.2-2.5   | 3.0-25.0  |

オ3-5表 ミニコンおよびマイクロコンピュータの比較

### 3.3 ワイヤドロジックとの比較

従来のワイヤドロジックからマイクロコンピュータ制御への変遷をふり返るとき、シーケンスコントロールのカテゴリに属するものと、アナログ量の一定値制御のたぐいの2種で代表して考えることができる。何れも大局的には個別の回路素子を組み合わせ、その間を接続して製作した時代から次第にユニットの組み合わせを経て、空間的に配置されたランダム回路で構

成組合される論理回路から、時系列的にプログラムロジックで制御される回路、即ちストアドプログラムにより制御される方式へと本質的な変化を遂げる事を意味する。その形は才3-5図のように示される。勿論初期のリレーシーケンスと雖も現在無くなるわけではなく、それぞれ個有の製品として長所、短所を持って居り広く使われている。従ってそれぞれの特長を認めて使い分け、マイクロプロセッサに適した使い方に積極的に使用する態度こそ普及の姿勢として正しいといえよう。



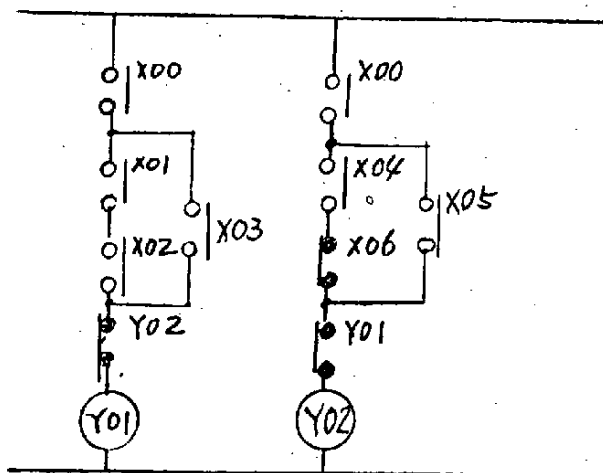
才3-5図 ワイヤロジックからマイクロプロセッサ制御への変遷

それぞれの段階をわかりやすく比較するために簡単な例で示すと才3-6図のようになる。

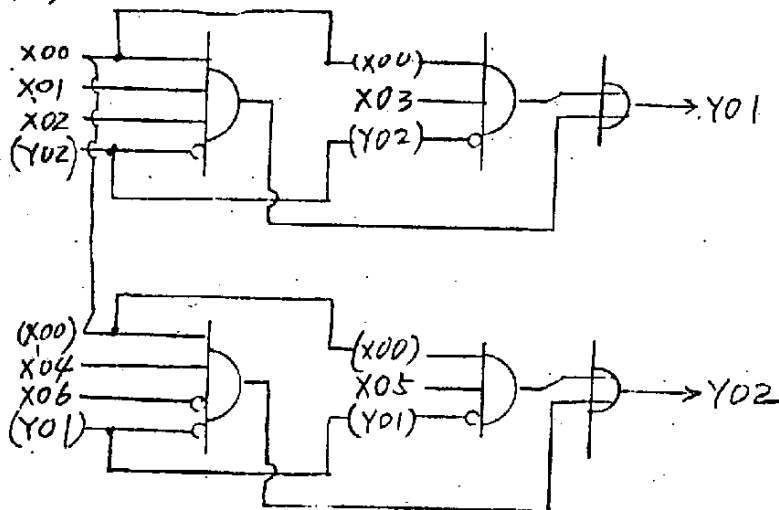
#### (1) リレーシーケンスの場合

図に示す個別X00~X06リレー接点を接続することによりY01、Y02を目的通り駆動することが出来る。製作は個別接点間を電線で個別に配線する必要があるが、リレーの総量が比較的少ない小規模回路では製作も容易であり、点検、改造変更も個々の端子を人間が見て確認出来るために容易であり、旧くから普及し使いなれて居る人口も多い利点があるといえよう。その反面大規模になった場合は此の正反対の欠点になる。

### (1) リレーシーケンスの場合



### (2) ロジック図の場合



注 ① AND  
 ② OR  
 ③ NOT

### (3) スタッドプログラムの場合

$$X00 \cdot X01 \cdot X02 \cdot \overline{Y02} + X00 \cdot X03 \cdot \overline{Y02} = Y01$$

$$X00 \cdot X04 \cdot \overline{X06} \cdot \overline{Y01} + X00 \cdot X05 \cdot \overline{Y01} = Y02$$

第3-6図 簡単な例による比較



## (2) ロジック図の場合

部品の電子化、半導体化の歩みと共に生まれ制定された表現で、個々の要素が論理機能（AND、OR、NOTなど）を端的に表すので、組み合わせて果すべき論理内容を正確に表現出来、規模が大きくなっても比較的理解し易い。回路の製作は個別におこなわなければならないことは(1)と同じである。

## (3) ストアドプログラムの場合

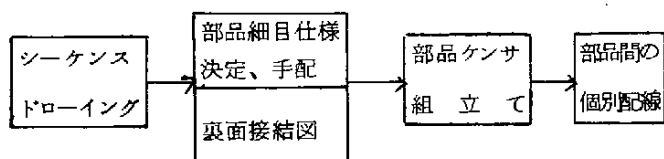
マイクロコンピュータによる制御の場合の動作は(3)に相当する。この場合の特徴は回路そのものも特定用途にその都度製作するのではなく、半導体メーカーが製作した量産ベースのマイクロプロセッサをそのまま活用する事が出来る長所があり、相当複雑なアルゴリズムでも表現し実行させることが出来る。此の場合は特定の用途に適した制御内用は、プログラムと称する従来と全く異質の仕事になって来る。チェックアウトも個々の演算素子を肉眼で確認できるハードウェア部品に対応させる事を全く棄てて、ソフトウェアのデバッグ作業に依存することになる。ソフトウェアが介在することによる著しい相違はいろいろの見方で実感としてつかんで載きたい重要事項であるが才3-7図に例として製作手順の差という面から従来のシーケンス回路との差を示した。この図のマイクロプロセッサの場合の一番下の一連の作業は従来全くなかった内容であるといえる。

マイクロプロセッサの活用のためには、従来才3-6図の(1)になれている人々にソフトウェアという仕事を身につけてもらう教育の問題と、内部の動作をほんやくし、なるべく人間にわかり易い形で使い易くするメーカー側のソフトウェアサポートが不可欠である事は一般の電算機と全く同じ問題をかかえている。この才3-7図の下の方のラインの仕事を身につける問題（即ち教育）は極めて重要であり、今後の広い分野への普及の条件となって来る。米国が此の点を認識し大きな応用分野と認めている事は既に才2節で触れたところである。

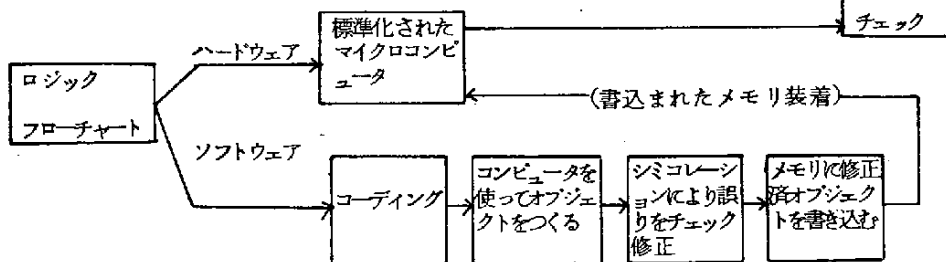
但し現在の段階では、マイクロプロセッサのアーキテクチャは単純に過ぎ、命令のレパートリは此の単純さをプログラムで補うように複雑化されて居り、通常の計算機よりも使いこなす事が難しい欠点があり普及させ難

い点がある。此の解決は今後の課題であろうが、この点は次節で再びふれる予定である。

#### ワイヤドロジックの場合



#### マイクロプロセッサの場合



オ 3-7 図 製作手順のちがい

### 3.4 マイクロプロセッサの特質と制御における応用上の諸問題

#### 3.4.1. マイクロプロセッサの長所と短所、マイクロプロセッサの特質を

まとめると、半導体集積回路としてチップ化されていること、および、コンピュータと本質的に同じストアプログラム方式のアーキテクチャを持っていることであり、これに現時点ではまだ開発されてまもない若い製品であることが付記できる。この3つの特質が、長所になったり短所になったりするわけである。その一般的評価を表にまとめてみると、オ 3-6 表のようなことが云われている。

この内容は昭和 50 年前半を基準においており、急速な進歩を続けているメーカーの状況から考えると、この評価、特に 3 番目の特質に依存している事項は事情の変化が予想されるが、現時点で早急に考慮すべき事

項として、下記の5項目が望まれている。

- (1) 価格を下げること。(実用化のためには、周辺に多数の回路外付を要し、此を含めた価格は意外に高価になること、および、少量購入価格と多量購入価格の差の大きいこと。)
- (2) 信頼性とチップ検査の方法
- (3) ソフトウェアとドキュメンテーションの充実、サポート体制の充実
- (4) 教育の問題(メーカのユーザ教育体制とユーザ内の技術革新に併うテクニカルトランスファの問題。)
- (5) 標準化、セカンドソースの問題

ここでは、これら当面の問題点を中心として若干の考慮を加えてみることにしたい。

### 3.4.2. マイクロプロセッサ応用制御の経済性

経済性の問題は、なかなか一般論でのテーマとしてとり上げる客観的データがまとまり難く論じ難いのが常であるが、実際は最も関心の深い事項なので、敢えてここに一節を設けることにした。従って、本節の価格自身は、昭和50年前半の参考値ということにとどめて載きたい。

#### (a) 価格の例

昨年米国 JOINT SEMICONDUCTOR CONFERENCE で、LSIの技術進歩について JOINT STUDY がおこなわれた。その結論によると、この分野の製品は今後10年の間、ほぼ2年間に性能が2倍に向上、価格は $\frac{1}{2}$ に低下する傾向が続くであろうとのことである。このことを念頭において、以下の参考価格の例を見て載きたい。オ3-7表は、昭和49年初頭現在の電子工業振興協会マイクロコンピュータ専門委員会で調査したマイクロプロセッサチップの価格例、オ3-8表は昭和50年上期現在公表されているキットの価格例、オ3-9表はこれをコンピュータや、デジタルコントローラに組み立てたシステム価格の例である。チップ価格のデータは、本応用篇が刊行される昭和50年下半期の例としては若干古くなることは上記の説明のとおりで、大体の目安は、制御への応用に最も多く使用される8ビットNチャネルタイプのチップも、1個100ドル以下になっている見通しである。オ3-8表のようにキットの形でチップのセットが自由に販売され、こ

れを自分でアマチュアも自由に組み立て、コンピュータをつくることができるということは、従来のコンピュータには全くなかったことで、アメリカではかなり普及しているようである。

マイクロプロセッサ  
の 3 つ の 特 質

長 所

問 題 点

半導体集積回路  
として  
チップ化されている

1. 高い信頼度
2. 小型軽量
3. 安価（但し数量による）
4. ハードウェアは標準化しやすい

1. チップの標準化と試験方法
2. 保守点検方法
3. 標準化とセカンドソースの確保
4. 小数購入時高価

ストアードプログラ  
ム方式で、小規模  
コンピュータの  
アーキテクチャ  
あり

1. 汎用性がありフレキシビリティに富む。
2. 途中の仕様変更にあたる。
3. 論理演算、算術演算、（10進数取扱）共に可能
4. タイミング単純、メモリ拡張性あり、インターフェース容易

1. サポート体制、特にソフト体制不十分
2. ソフトの負担が大きい
3. デバッグの費用負担が大
4. 教育の負担大
5. ドキュメント不十分

製品として未だ  
若く、技術進歩  
の途上にある

1. 最近の製品はミニコンに近い性能あり
  - ・命令レバトリ強力
  - ・インデックス処理可能
  - ・ビット演算可能
  - ・サブルーチンコール可能
  - ・周辺チップ充実
  - ・割込処理可能
  - ・TTC、レベルインターフェース
2. マルチプロセッサの特長を生かした分散制御の可能性（今後への期待）

1. 未だ性能に限界がある。
  - ・速度……思ったよりおそい
  - ・語長……バイト長では不足
  - ・アドレス・アドレッシング貧弱
  - ・接続……ピン数が制限される
  - ・集積度……チップの固定に複雑な回路を組まないと実用にならない
  - ・機能限界 割込機能不十分  
サブルーチンネスト不足  
命令レバトリ不足  
スタック不便  
メモリ効率悪し
2. バスの拡充、分離
3. 停電対策の要ること、二電源の要ること

表 3-6 マイクロプロセッサの一般的評価（昭和50年現在）

(100個購入時の単価)

|                  |                      |                  |                    |                   |              |
|------------------|----------------------|------------------|--------------------|-------------------|--------------|
| 4<br>ビ<br>ッ<br>ト | インテル 4004 \$30       | 8<br>ビ<br>ッ<br>ト | インテル 8008 \$60     | 12<br>ビ<br>ッ<br>ト | インタージル \$175 |
|                  | フェアチャイルド PPS-25 \$60 |                  | モトローラ 6800 \$150   |                   | IM6100       |
|                  | ロックウェル PPS-4 \$45    |                  | PPS-8 \$47         |                   |              |
|                  | インテル 4040 \$40       |                  | フェアチャイルド F8 \$75   |                   |              |
|                  |                      |                  | パローズミニット \$60      | 16<br>ビ<br>ッ<br>ト | 未 詳          |
|                  |                      |                  | モステック MK5065 \$100 |                   |              |
|                  |                      |                  | RCA COSMAC \$300   |                   |              |
|                  |                      |                  | シグネクス S-2650 \$120 |                   |              |

才 3-7 表 マイクロプロセッサチップ価格例

(昭和49初頭現在)  
電子協調査による)

|                  |                                                                   |                  |                                                 |                   |                                                  |
|------------------|-------------------------------------------------------------------|------------------|-------------------------------------------------|-------------------|--------------------------------------------------|
| 4<br>ビ<br>ッ<br>ト | インテル 4004<br>P-ROM (2K)<br>RAM (1K)<br>I/Oバッファ<br>アドレスラッチ<br>\$99 | 8<br>ビ<br>ッ<br>ト | インテル 8008<br>+ 周辺チップ<br>18種附属<br>\$250          | 16<br>ビ<br>ッ<br>ト | NSC IMP-16<br>P-ROM<br>R-ALV×2<br>その他附属<br>\$330 |
|                  | NSC IMP-4<br>P-ROM<br>R-ALV<br>FILV<br>\$105                      |                  | NSC IMP-8<br>P-ROM<br>R-ALV×2ケ<br>EILV<br>\$203 |                   |                                                  |
|                  |                                                                   |                  | インテル 8080<br>+ 周辺チップ<br>18種附属<br>\$495<br>\$695 |                   |                                                  |
|                  |                                                                   |                  | モトローラ 6800L<br>⊕RAM 2K<br>周辺4種附属<br>\$320       |                   |                                                  |

才 3-8 表 マイクロプロセッサ用LSIキット価格例

(昭和50メーカー)  
カタログ記載例)

|           | 製 品                | メモリ                           | 標準価格 (ドル) | 備 考                                          |
|-----------|--------------------|-------------------------------|-----------|----------------------------------------------|
| 4<br>ビット  | インテルIntellec 4     | 4K RAM                        | 2,545     | 4004使用 入力4、出力8、                              |
|           | ACT社PPS4 MP        | 2K (バイト) PAM<br>1K (4ビット) PAM | 2,495     | ロックウェア 入力1 (8ビット) 出力1<br>PPS4使用 テレタイプインタフェース |
|           | Pro-Log SS1A       | 128バイトP-ROM                   | 3,150     | 4004使用 8ビット入/出力×4                            |
| 8<br>ビット  | インテルIntellec 8     | 8K P-ROM                      | 3,540     | 8008使用 入力4、出力4                               |
|           | " Intellec80       | 8K P-ROM                      | 3,840     | 8080使用 "                                     |
|           | NSC社IMP 8P         | 8K RAM                        | 3,750     | NSC IMP8 8ビット入出力バス<br>フラグ4、割込1               |
|           | DEC社LDS            | 4K P-ROM<br>4K RAM            | 1,800     | 8008使用 8ビットバス (TTL)<br>パネルはオプションの裸価格         |
|           | Pro-Log SS2A       | 1K RAM<br>1.28K P-ROM         | 3,950     | 8008使用 入力32、出力32                             |
|           | モトローラ<br>Exorciser | 256 RAM<br>3K ROM             | 2,640     | MC 6800 テレタイプインタフェース<br>使用 8ビットバス            |
| 12<br>ビット | 東芝<br>TLOS12コンピュータ | 500 RAM                       | 2,320     | TLCS 12 テレタイプインタフェース                         |
| 16<br>ビット | NSC社IMP 16P        | 4K RAM<br>3K (バイト) ROM        | 3,850     | IMP 16 16ビット入出力バス                            |

附 システム例; Bell & Howell デジタルコントローラ

\$ 12,000

(インテレック 80 相当) + アナログ入力 8

デジタル入力 8

制御卓

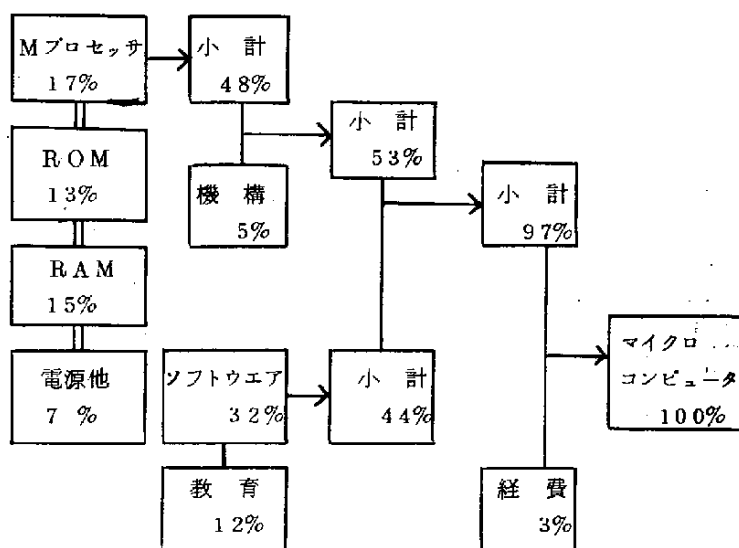
表 3-9 マイクロコンピュータとしてまとめた場合の価格例 (IEEE Spectrum  
11-12月号掲載)

マイクロプロセッサを使用しても、完成したシステムになると第3-9表のように一桁上の価格になり、ミニコンとの性能の差を考えると、むしろ特に安価だと驚くような価格ではなくなっている。勿論これは当然のことであるけれども、又一面、マイクロプロセッサを制御に使用する場合にコンピュータを使用するという概念にとらわれると意外に高価になる可能性を示唆していると云えよう。この点に应用製品を開発する立場の多くの人々の工夫が期待される。

しかしながら、チップの価格が何故2割にも満たない比率になるかという中味については、よく知っておく必要があるので、次節で価格構成の中味を調べることにしよう。

#### (b) 価格の構成とソフトウェア価格の重要性

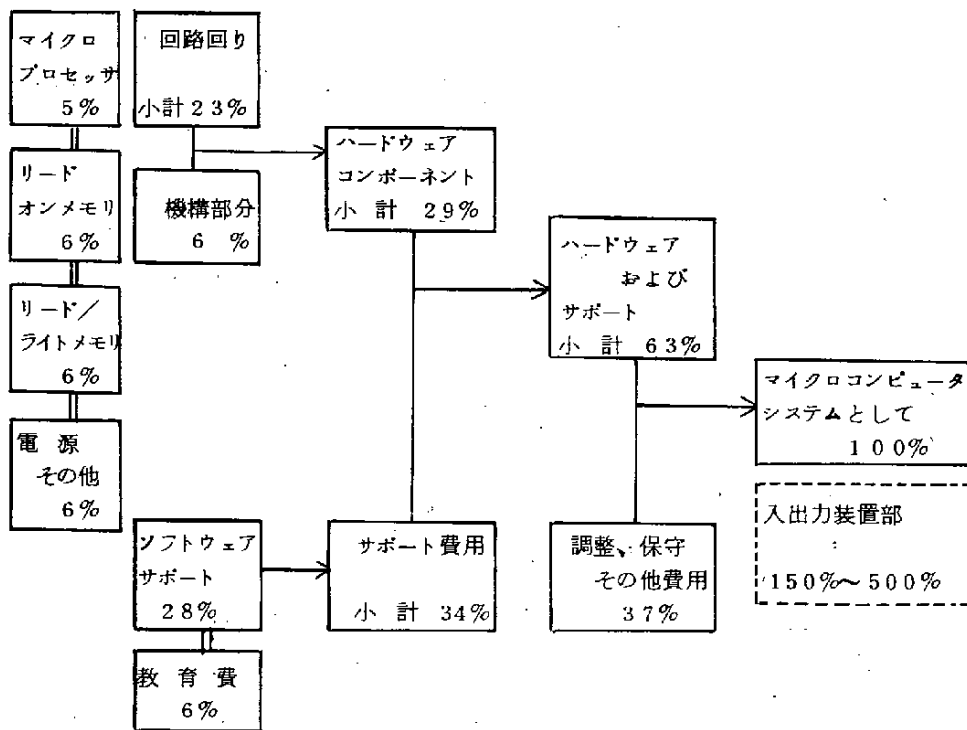
第3-8図は第3-9表に例示されているようなコンピュータを汎用計算機として素直に使用しようとする場合の構成である。この場合もコンピュータそのもののハードウェア価格にほぼ等しいソフトウェアコスト（教育を含めて）が必要である。我国でコンピュータの利用を計画する使用者は、概してソフトウェアの経費を直接コンピュータ



第3-8図 汎用マイクロコンピュータの価格構成('74at U.S.の例)

の費用として計上しない傾向が未だ強いが、マイクロコンピュータのような簡単な電算機でも第3-8図の例のように可なりの比率の費用になるのであるから今後の正しい普及（それはむしろ必然の方向であるが）のためには、本表のように明らかに費用として認め価格を明確にしてゆくことが必要である。

さて制御に使用する場合、ソフトウェアおよび設置に伴う最適調整や、設置後の保守などのハードウェア自身以外の費用は第3-8図のように電算機としてバッチJOB処理を行なう場合よりも一段と大きな比率を占める。



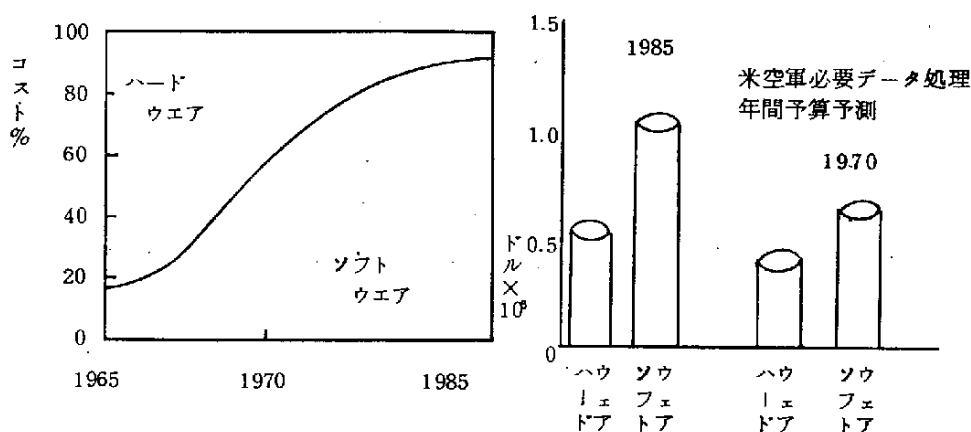
第3-9図 制御に使われる場合のマイクロプロセッサシステム価格構成

( '74 at U.S. マーケット )



第3-9図はこの内訳構成を示したもので、此の場合はもはやマイクロコンピュータのハードウェアそのものは30%足らずの比率にすぎない。従ってマイクロプロセッサ自身は5%に止ることになる。

既に市場調査の節で述べたごとく、制御への応用は、制御対象に応じて個別にソフトウェアを製作し、対象を含めて全体として調整せねばならないし、従って生産形態も多量生産より個別生産になることから、ソフトウェアの比率が高いのは当然であるが、今後益々ハードウェアの価格は低下するのに対し、ソフトウェアコストはむしろ上昇することを考えると、今後益々ソフトウェアの重要性が大きくなることは明らかである。



第3-10図 1980代における米国空軍情報処理必要予算予測

(RAND CORP.) 報告

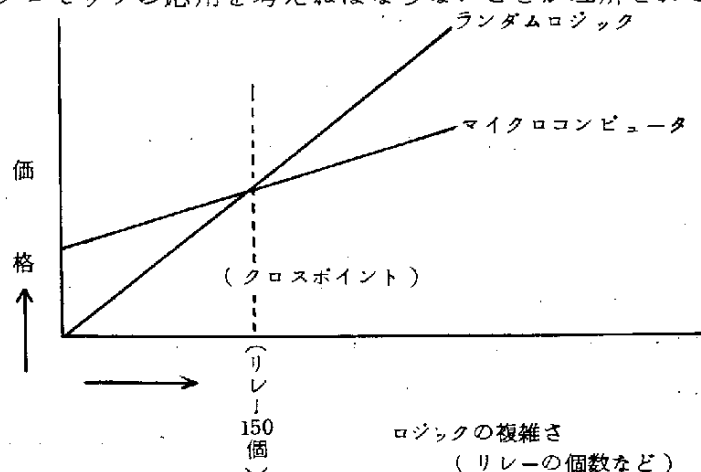
第3-10図に参考のため一般のコンピュータについてRAND CORPORATIONが空軍の委託調査で報告した例を示して、此の傾向を認識してもらいたい。

さて従来のワイヤドロジックの制御との経済性の比較は第3-9図そのものがマイクロプロセッサ制御の有利性を物語っている。即ち従来の制御はハードウェア自身もソフトウェアと同様に制御対象に応じ

て製作せねばならなかったのに比べて、マイクロプロセッサを使用する事によりハードウェアそのものは個別性から解放され、従って価格比率も小さなものになったものであり、今後の優位性を示唆しているものと理解されよう。

たとえば我国で昭和49年～50年の状況で、制御装置を個別に製作する場合リレー1個当たり少くとも15,000円ないし20,000円として価格が計算される。

これに対してマイクロプロセッサを使用すれば、最も一般的な8ビットNチャンネル型で、およそ500個ないし1000個のリレーを使用する制御を処理する能力があることになる。従って能力を十分に生かすような規模の制御システムでは従来の制御よりもはるかに経済的である。従来の制御(ランダムロジック)とマイクロプロセッサを活用した、マイクロコンピュータの使い方と価格について、一般的に第3-11図のようなクロスポイントがあり、ある規模以上の複雑なロジックの場合に、マイクロコンピュータが有利になると云われている。そしてこのクロスポイントは、昭和50年現在では、リレー150個程度の位置にあるようである。このクロスポイントは、年々、小規模側に移ってゆくことを考慮すると、今から積極的に制御の分野でも、マイクロプロセッサの応用を考えねばならないことが理解されると思う。



第3-11図

### 3.4.3 信頼性と検査の問題

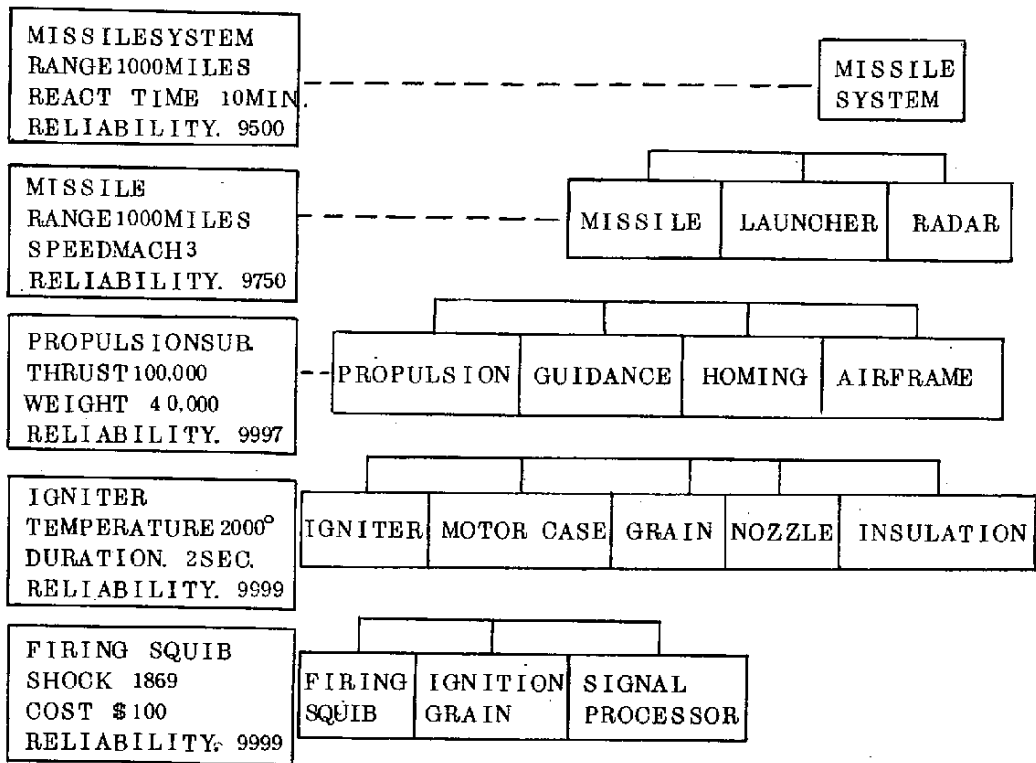
#### (a) 制御システムにおける信頼性

産業におけるプロセスの総合自動比は、生産性の向上をとおして社会生活の豊富さと便利さに大いに貢献してきたけれども、ひとたび故障を起こすと、その高度の集中化のために、波及範囲が大きく、損害も多額になる。このために制御システムを計画する者にとって、信頼性は極めて重要な問題となっている。これに対し、個々の制御部分の設置される環境条件は、温度、湿度、電源状態、振動の有無など、いろいろの意味で過酷であり、その複雑な組み合わせの故障絶無を期待することは極めて困難である。信頼性の問題を考える場合は、複雑さと個々の部品の故障を起こさぬことの2つの面から見る必要があるであり、その合理的バランスが重要である。参考のために、信頼性の問題を論ずる時しばしば引用される第3-12図を紹介しておく。この図の例のように、複雑な大規模システムを高い信頼度で稼働させるためには、それぞれの部分の信頼度を向上させるのみではなく、適切な使い方、冗長度の採り方、故障の際の安全性確保、修理の容易性などの対策を総合的に考慮する必要がある。最近のRASアプローチはこの見地に立ったものである。第3-10表にその概要をまとめたものを示しておく。マイクロプロセッサを組み込んだ制御に関する応用製品を設計する時に参考にしたい。

最近このように信頼性については、組織的、合理的、総合的に考えるようになってきたが、その歴史を振り返ると、やはり重要な個別部品があまりにも信頼度が低く故障を起こし易かったために、技術者が苦心の末に考え出したことである。

特に電子部品は小さいために、一般には機械的に頑丈なものは少なく、小型なるが故に余裕度も小さく設計され、故障を起こし易い一面があったことは否めない。しかしながらこれら過去の弱点も、歴史と共に長足の進歩がなされて見違えるように改善されてきている。

第3-13図は制御に使われる中心的論理素子が、過去20年の間に改善されてきた経過を電々公社にて、電子交換機のために調査したも



# HIERARCHY OF PERFORMANCE CRITERIA

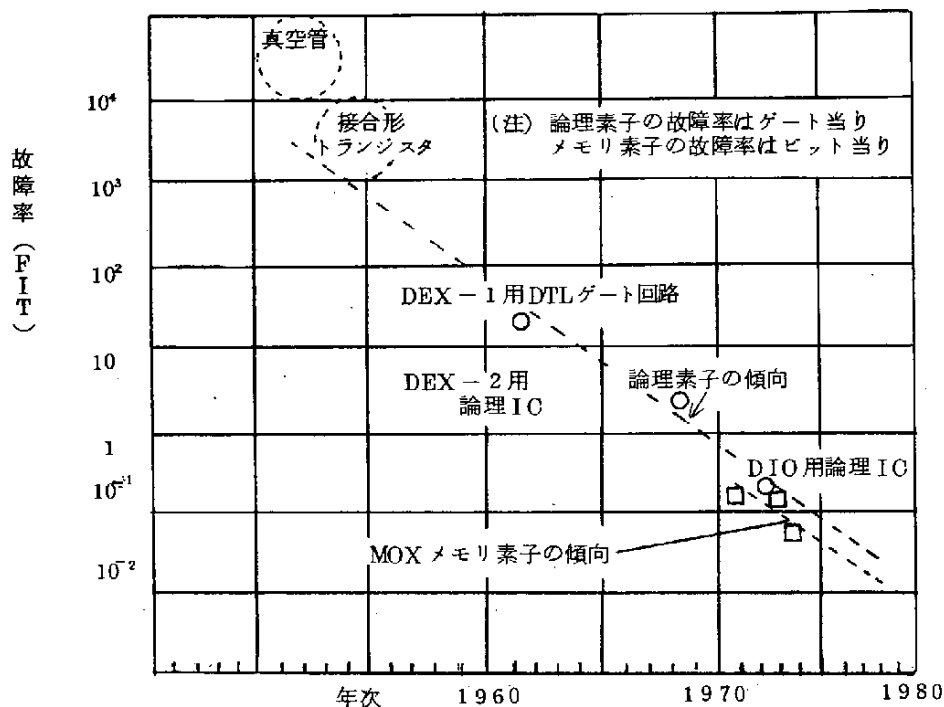
第 3 - 12 图

RASの要素と関連部門(Reliability, Availability, Servisability)

| ア プ ロ - チ        | R A S の各概念 |   |   |                  | 関 連 部 門 |    |        |    | サ<br>ー<br>ビ<br>ス |
|------------------|------------|---|---|------------------|---------|----|--------|----|------------------|
|                  | R          | A | S | シ<br>ス<br>テ<br>ム | ハードウェア  |    | ソフトウェア |    |                  |
|                  |            |   |   |                  | 設計      | 製造 | 設計     | 製造 |                  |
| 部品の信頼度向上         | ○          | ○ |   |                  | ○       |    |        |    |                  |
| 装置単位の信頼度向上       | ○          | ○ |   |                  | ○       |    |        |    |                  |
| 製造技術の向上          | ○          | ○ | ○ |                  |         | ○  |        |    |                  |
| 環境設計             | ○          | ○ |   | ○                | ○       |    |        |    |                  |
| ソフトウェアそのものの信頼度向上 | ○          | ○ |   | ○                |         |    | ○      | ○  |                  |
| 故障の自動検出・訂正       | ○          | ○ |   | ○                | ○       |    | ○      |    |                  |
| 故障の自動診断          |            | ○ | ○ |                  |         |    | ○      |    | ○                |
| 冗長設計             | ○          | ○ | ○ | ○                | ○       |    | ○      |    | ○                |
| フェイル, ソフト設計      | ○          | ○ |   | ○                |         |    | ○      |    | ○                |
| ソフトウェアによる障害対策    | ○          | ○ | ○ | ○                |         |    | ○      |    | ○                |
| マン・マシンインターフェース向上 | ○          | ○ | ○ | ○                |         |    | ○      |    | ○                |
| エージングの徹底実施       | ○          | ○ |   | ○                | ○       |    | ○      | ○  | ○                |
| 実装技術の向上          |            | ○ | ○ |                  | ○       |    |        |    | ○                |
| 保守技術・体制の確立       |            | ○ | ○ |                  |         |    |        |    | ○                |

※昭和48年11月 計測と制御 第12巻第11号より

第3-10表



第3-13図

のである。

尚、たて軸の単位FITは(1FIT=10<sup>-9</sup>、個・時間)である。

#### (b) マイクロプロセッサの信頼性

微少な半導体ウエハの中に極めて複雑な回路を構成し、細い線でボンディングされているマイクロプロセッサは、複雑さが信頼性の敵である常識から、信頼性について心配されるのは当然であり、さらに制御への応用を考える場合は益々重要問題として検討されなければならない。このためのデータとして第3-11表に主として電子部品の個別の故障率を例示、これに対比させて現時点で推定されているMOS型メモリと、マイクロプロセッサの値を例示してみた。これら部品を使用し組み立てる装置としては、部品そのものの以上に、ハンダ接合部の数、圧着接触部、カード基板その他機構部品そのものの信頼性も重

要で、装置の信頼性はそれらの集合になる。

この意味からは第3-11表のICを例にとっても、個別にトランジスタ、ダイオードを組み合わせた場合よりも明らかに故障率はるかに低い。そしてそれと同様にマイクロプロセッサも、ICを組み合わせて同じ論理のCPUを作ったと仮定した場合よりもはるかに故障率が低い。それは内部のボンディング箇所、外部接続ピン数などが圧倒的に減少するし、

IC接続のハンダづけは皆無になる事から自然の結果として得られることである。昭和50年現在の状況で見ても、各メーカーのMOS型マイクロプロセッサチップは、500～1000 FIT程度に入ってきたようで、従来のICを組み合わせて製作する場合より、一段と高い信頼度を期待できる時代に近づいている。最近、制御に使用する場合の特長の第1位にマイクロプロセッサ

の高信頼度を上げている例があるのは以上の理由で、云わば自然の結果なのである。

### (C) 検査の問題

ひとたび検査に合格したマイクロプロセッサは、前節に説明したように高い信頼度を持つが、その検査の方法、めったに故障を起こさなくても起こした場合の内部診断方法となると極めて困難であり、修理することは殆ど不可能である。従って、これから制御にマイクロプロセッサを使ってゆく場合には、検査と保守点検による不良品のとり換えが使用者の重要な任務になるのであるが、未だ現在のところこの問

| 種 類                              | 50℃における故障率<br>(単位: FIT) |
|----------------------------------|-------------------------|
| 1. 持抗                            | 3 ~ 7                   |
| 2. 可変持抗                          | 50 ~ 150                |
| 3. コンデンサ                         | 2 ~ 3                   |
| 4. ダイオード                         | 5                       |
| 5. トランジスタ                        | 10 ~ 20                 |
| 6. IC                            | 20 ~ 50                 |
| 7. リレー                           | 50 ~ 80                 |
| 8. トランス                          | 50 ~ 150                |
| 9. ヒューズ                          | 100 ~ 200               |
| 10. コア(ビット当り)                    | $10^{-2}$               |
| 11. ドラム( # )                     | $10^{-1}$               |
| 12. メータ                          | 300 ~ 500               |
| 13. MOS型メモリ(1K)                  | 200 ~ 500               |
| 14. マイクロプロセッサ<br>(MOS型)          | 500 ~ 1000              |
| (注) FIT = 個, 時間 $\times 10^{-9}$ |                         |

第3-11表 エラーレート表(例)

題をあまり掘り下げて検討されていない。本節ではこれを今後の問題として如何にあるべきかを考えてみることにする。

#### メーカーの実施する検査の内容

応用製品を制御に使ってゆこうとする立場のユーザにとって、信頼性確保、検査と保守のあり方を考える場合、まず参考になるのは、半導体メーカーが自社で生産するマイクロプロセッサを如何に検査し、品質を管理して市場に出しているかという事である。我国のメーカー側で実施している信頼性試験も、MOS タイプ製品について、電卓用チップ、メモリ用MOSを経て相当の経験を積み上げているので参考になると考えられる。

大規模な量産形態をとっているメーカーの検査は、特にLSIについては、組み立てられた後の完成品になってからでは内部の詳細な信頼性評価は困難なので、製造の工程にそって部分部分で厳重な検査をおこなっている。製品の本質的品質保持の基礎は、それら個々の部分の検査によっていると云えるが、ユーザ側で直接関係するのはそれら工程内検査よりも最終製品段階の検査なので、その内容を概観する。

| グループ | 試験項目   | 試験条件                                           |
|------|--------|------------------------------------------------|
| 1    | 高温動作寿命 | 最高動作周囲温度 1000 時間                               |
|      | 高温保存   | 最高保存周囲温度 1000 時間                               |
|      | 高温高湿保存 | 65℃, 95%RH 500 時間                              |
| 2    | はんだ耐熱性 | 260℃, 10 秒                                     |
|      | 熱衝撃    | 0～100℃, 15 サイクル, 10 分/サイクル                     |
|      | 温度サイクル | 最低～最高保存周囲湿度, 10 サイクル, 1 時間/サイクル                |
| 3    | はんだ付性  | 230℃, 5 秒, ロジン糸フラックス使用                         |
|      | 端子強度   | 引張り: 340 g, 30 秒 曲げ: 225 g, $\pm 30^\circ$ 3 回 |
|      | 振動     | 20 G, X, Y, Z 方向各 4 回, 100～2000 Hz, 4 分/サイクル   |
|      | 自然落下   | 75 cm, 3 回, 木板, Y <sub>1</sub> 方向              |
|      | 定加速度   | 20000 G, Y <sub>1</sub> 方向, 1 分                |

三菱電機技報 Vo149, No 2

第3-12表 集積回路の信頼度試験項目・条件の概要



第3-12表はメーカーで実施する信頼性試験の検査項目と試験条件(例)である。動作寿命試験は、125℃の環境で多数のチップを連続的に1,000時間ないし2,000時間動作させ、故障発生回路をプロトし、高湿保存は150℃の環境下に同様に放置するもので、熱加速テストになる。

はんだ耐熱、熱衝撃、温度サイクルの3種のシリーズテストは、同表に示す過酷な条件の下に所定サイクル動作させて異常のないことを確認するものである。第3-13表はメモリ用のMOS型LSIの試験結果の公表された1例である。

| 品種用     | パッケージ                          | 試験項目・条件 |      | 試料数個 | コンポートメント<br>アワー 個×時間 | 故障数個 | 故障内容               |
|---------|--------------------------------|---------|------|------|----------------------|------|--------------------|
| M58533S | 28ピン<br>メタルシール<br>セラミック<br>DIL | 動作寿命    | 55℃  | 38   | 114,000              | 0    |                    |
|         |                                |         | 80℃  | 20   | 20,000               | 0    |                    |
|         |                                |         | 125℃ | 15   | 15,000               | 0    |                    |
|         |                                | DCバイアス  | 80℃  | 20   | 20,000               | 0    |                    |
|         |                                |         | 125℃ | 15   | 15,000               | 0    |                    |
|         |                                | 高温保存    | 150℃ | 38   | 38,000               | 0    |                    |
| M58533P | 28ピン<br>プラスチック<br>モールド<br>DIL  | 動作寿命    | 80℃  | 20   | 20,000               | 0    | ファンクション不良<br>(8時間) |
|         |                                |         | 125℃ | 55   | 74,000               | 1    |                    |
|         |                                | DCバイアス  | 80℃  | 20   | 20,000               | 0    |                    |
|         |                                |         | 125℃ | 40   | 60,000               | 0    |                    |
|         |                                | 高温保存    | 150℃ | 30   | 30,000               | 0    |                    |
| M58531P | 16ピン<br>プラスチック<br>モールド<br>DIL  | 動作寿命    | 80℃  | 20   | 20,000               | 0    |                    |
|         |                                |         | 125℃ | 20   | 20,000               | 0    |                    |
|         |                                | DCバイアス  | 80℃  | 20   | 20,000               | 0    |                    |
|         |                                |         | 125℃ | 20   | 20,000               | 0    |                    |
|         |                                | 高温保存    | 150℃ | 50   | 50,000               | 0    |                    |
|         |                                |         |      |      | 計                    | 1    |                    |

第3-13表 メモリ用MOS LSIの寿命試験結果(一例)

| ステップ          | 概 要                                                                                                                                                                                                         |
|---------------|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| (1) 外 部 検 査   | <ul style="list-style-type: none"> <li>・リード、メッキ、はんだづけ、溶接部分の状態などの検査</li> <li>・パッケージ材料、封止、マーキングなどの検査</li> <li>・その他、保証されている外観の状態の検査</li> <li>・手法として、実体顕微鏡、金属顕微鏡、X線透過装置、ファイナリーク及びグロスリークテストなどが適宜使用される</li> </ul> |
| (2) 電 気 的 検 査 | <ul style="list-style-type: none"> <li>・電気的パラメータの測定による断線、短絡、パラメータの劣化の判定</li> <li>・シンクロスコープ、カーブトレーサなどによる特性観察及び重要な物理的特性を電気的な特性で観察</li> <li>・必要により環境試験、寿命試験などのストレス試験をおこなう</li> </ul>                           |
| (3) 内 部 検 査   | <ul style="list-style-type: none"> <li>・素子のパッケージのふたをとり除き、素子内部の構造を光学的に検査する</li> <li>・シリコンチップの表面の観察</li> <li>・適用できる場合、プローブによる電気的特性の測定</li> <li>・必要によりSEM、XMA、赤外線マイクロキャナなどが適宜使用される</li> </ul>                  |
| (4) チップ解析     | <ul style="list-style-type: none"> <li>・内部検査でおこなわれる解析を補足するために、金属学的な解析がおこなわれる</li> <li>・チップの断面切断</li> <li>・酸化膜の欠陥の解析</li> <li>・拡散の欠陥の解析</li> </ul>                                                           |

三菱電機技報Vol. 49・No2・1975

第3-14表 集積回路の故障解析手順概要

故障の解析は第3-14表に示すように、外部検査、電気的検査、内部検査、チップ解析の順に大型計算機を活用しておこなわれている。これらのメーカーで実施する各種の試験は結局、チップの電気的静特性、動作マージン、寿命、エージングを総合したものであると云えよう。以上のメーカーで実施する検査をふまえ、ユーザの側に立ってメーカーの

上記検査の結果を生かし、使用信頼度を確保するにはどんな検査をするかという事が当面のテーマである。それは結局、上記試験のうちエージングに重点を置く事になると思われる。それ以外の項目については、むしろメーカ側の実施内容に期待するのが試験に要する費用、時間などの点から妥当であるということになる。

尚、本節に紹介したメーカで実施している試験、検査の規格は、1974年11月に制定された米国国防省規格“MIL-STD-883A”マイクロエレクトロニクスの試験方法および手順に準拠している。従来あまりにも進歩が早く、試験の基準が明確でなかった集積回路、マイクロエレクトロニクスについてもMILのスタンダードが制定された。その邦訳が電子機械工業会（防衛庁半導体分科会）によって刊行されているので、必要ある方は工業会に照会して戴きたい。

#### ユーザーとしても検査について

第3-15表はNASAで69年に報告したLSIの故障の分類、第3-16表はIEEEでまとめたMOS型ICにおける特有の故障モードの分類である。このイオンマイグレーションおよび酸化膜ショートの高比率のことがMOS型LSIの特有現象であると云える。これ等は、製造した数時間後に起こることがあるために、ユーザとしてスクリーニングの意味での検査が必要と云われている点である。

次に第3-14図にLSIの故障率の温度による変化を示す。MOS型LSIは図中のBで温度による加速性が特に高い事示している（BIPOLAR型はD型）

| LSIの故障分類 |     |
|----------|-----|
| チップ      | 45% |
| パッケージ    | 28% |
| ハンドリング   | 20% |
| 他        | 12% |

NASA '69

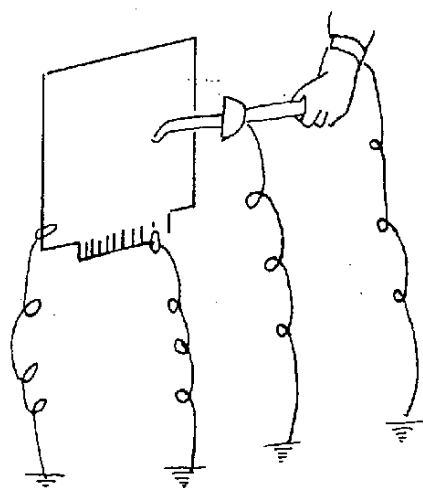
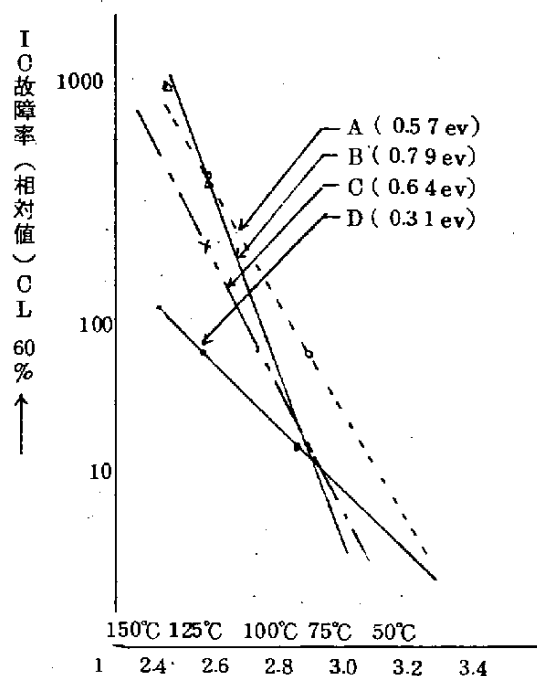
第3-15表

| MOSICの故障モード |     |
|-------------|-----|
| イオンマイグレーション | 64% |
| 酸化膜ショート     | 31% |
| パッケージ       | 3%  |
| リードボンディング   | 1%  |
| 短絡          | 1%  |

IEEE '74

第3-16表

そこでユーザとしては、高温ダイナミックテストともいべき方法  
即ち高温環境下で各種の命令を連続実行させてみる方法が、最も有効  
であるということが云い得ることになる。



第3-15図

ハンドリングの

静電気防止法

第3-14図 MOS LSIの故障率  
温度依存性

ここで第3-15表に示されている取扱い不良のパーセンテージが  
意外に高いことに鑑みて、2種類の注意事項を紹介しておきたい。

その1は放熱確除の工夫であり、2は取り扱い時にサージ電圧によ  
る絶縁破壊防止の接地線取付けの注意である。

LSIの消費電力はミリワット単位で表わされるように、決して大  
きなものではないが、素子の微少化はそれを上回っているので、チッ  
プの発熱密度は増加の一途を歩んでいる。従って、熱の放散に留意し、  
局部的に温度の上昇することを防止する配慮は忘れてはならぬことで

ある。換気ファンのフィルタ、目づまりのような意外な事によるLSIチップの誤動作の原因となることが意外に多いことを留意して、日常の取り扱いを考慮して載きたい。

もう1つは、MOSタイプLSIは酸化膜の薄い境界面が勝負であるために、帯電した静電気がチップの定格電圧よりもはるかに高い電圧を誘起するとき、破壊の危険がある。最近ではメーカーもチップ側でその対策を考慮するようになってきているが、念には念を入れた取り扱い方として、この種のチップを取り扱う時には、第3-15図に示すようにアース線を準備して帯電を防止するとか、持ち運びのときには、少し導電性を含ませるためにカーボンを混入した成分の袋を使用するなどの考慮を推奨したいと思う。

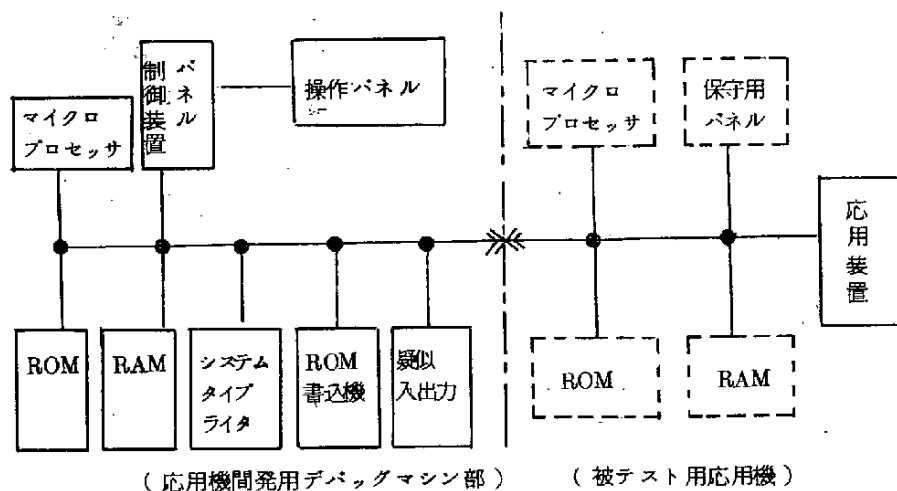
#### (d) システムチェックの進め方

検査の問題に関連するマイクロプロセッサ応用製品を開発するための検討事項に、システムチェックをどうやったら良いか、という問題がある。複雑なロジックがチップの中に収まって外からの検査が仲々難問になることは、システムチェックの進め方についてなお然りである。ハードウェアランダムロジックかソフトウェアに置き換えることに応じて、プログラムデバックの作業が必要になるが、システムとしての入出力動作はハードウェアとして残っているので、信号の入力から、処理、出力を含めた検査即ちシステムチェックには、ハードウェア、ソフトウェアを含めた総合試験が必要である。

最近までハードウェアについては従来どおりシンクロスコープなどの測定器、ソフトウェアについてはホスト計算機によるクロスアセンブラ、シュミレータの活用と、ハード、ソフトばらばらにであれば、システムチェックのサポートの方法も充実してきているが、ハード・ソフトを含めたシステムチェックの合理的な開発手段としてのサポートシステムの必要性の認識は不十分であった。

最近やっとこの点に着眼し、マイクロプロセッサ応用製品を開発するための、サポートツールとしての特殊なコンピュータ、即ちデバックマシンとも称すべきものが製品化され始めている。その出現は、ユ

ーザにとってはむしろ遅きに失する感があるが、今後益々マイクロプロセッサの応用が多方面に拡大するためには、半ば必須の開発用ツールとして活用されるであろうと思う。第3-16図はその「マイクロコンピュータ開発システム」の構成図である。

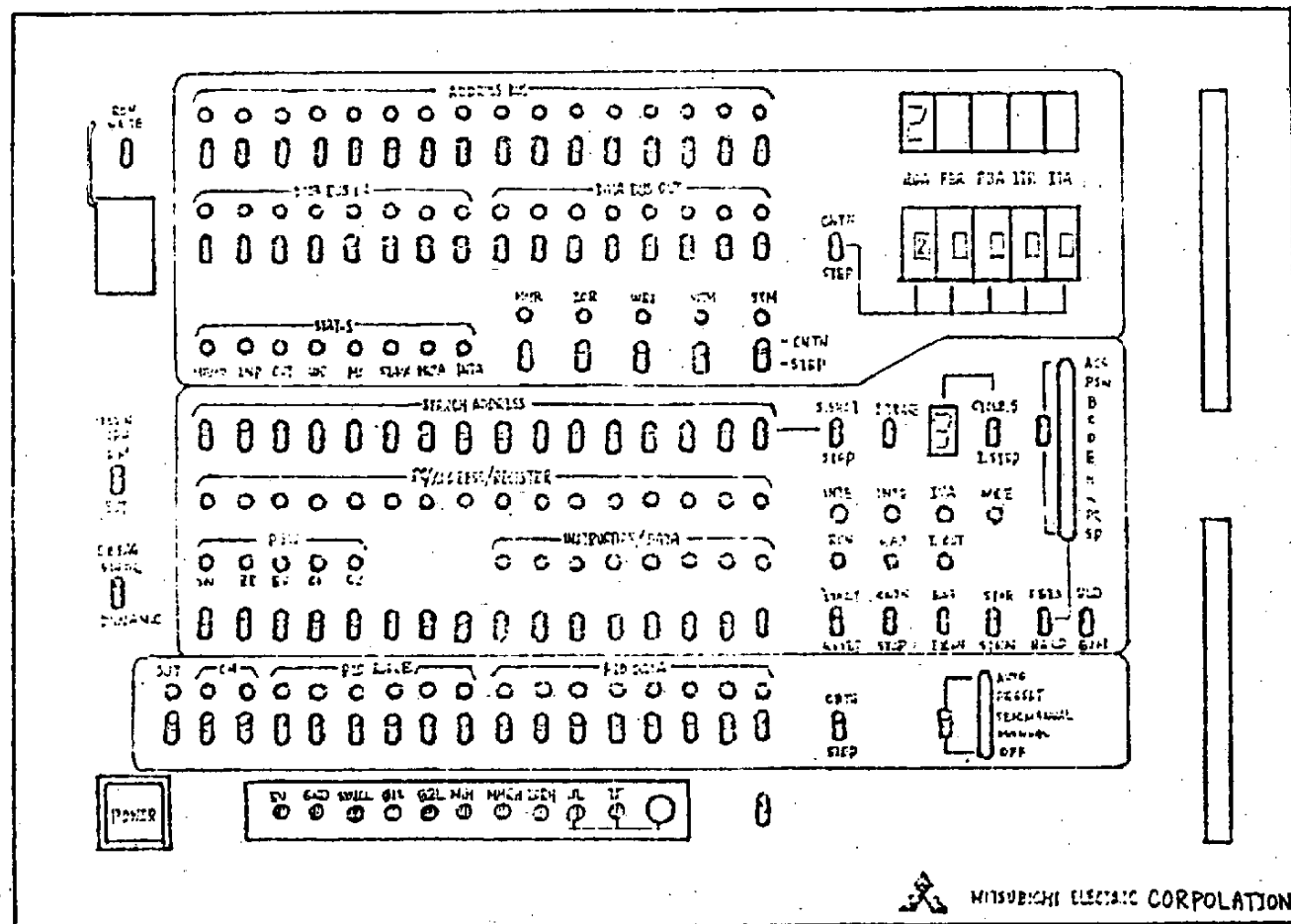


第3-16図 マイクロコンピュータ開発システム

第3-16図の1点又線の左側半分はマイクロプロセッサ応用製品を開発しチェックするデバックマシン部、右半分は開発しようとする目的の被テスト応用機(これをターゲットマシンと称する)である。

#### 開発システムの使い方

右端の応用装置を動かすためのロジックを、プログラム単位としてチェックした後デバックマシン部のRAMに収め、デバックマシン部に使用しているマイクロプロセッサと組み合わせて動作試験をおこないつつ調整作業を進める。完了し、RAMに収容されたプログラムが完全なものになったら、ターゲットマシン用のROMに書込み固定する。次にターゲットマシン用のマイクロプロセッサを装着してROMと組み合わせてシステムチェックを行ない、応用装置を完成させる手順になる。



第3-17図 マイクロコンピュータ開発システム操作パネル(例)

### 開発用システムとマイクロコンピュータの相違

第3-16図を見ると、左半分のデバッグマシン部分は一見したところ通常のマイクロコンピュータと同じように見えるが、目的が相違するとおり、構成は下記の考慮をおこなっている点が異なっている。

- (1) マイクロコンピュータ開発システム自身も入出力装置を一式具備したマイクロコンピュータでもあるが、それ自身を動かすプログラムメモリは左端のROMに入っており、そのアドレスフィールドが、ターゲットマシンを開発するとき使用するメモリアドレスの範囲に喰い込む事のないように指定されている。従ってターゲットマシン用プログラムの開発は、使用予定メモリの量に応じて自由に指定してもよい。開発用マシンのアドレスエリアは、通常使われる範囲をRAMのフリー領域に空けて、その先を使用するように配慮されている。
- (2) 割込ラインについても、ターゲットマシンで使用するレベル指定が開発システム自身の使用するラインと重複することがないように空けて製作されている。
- (3) マイクロコンピュータ開発システムの操作パネルは、通常のマイクロコンピュータと同様にプロセッサの動作が見られるというだけでなく、バスの信号、ステータスやプロセッサの制御信号も見ることができ、バスラインにワードをセットするようなテスト機能もそなえている。第3-17図はその操作パネルの1例で、通常のマイクロコンピュータよりテスト機能がはるかに拡大されていることが、ランプとスイッチの数からも見ることができる。
- (4) 開発システム自身のROMに収容されているコーティリティブプログラムは、RAMに収めたターゲットマシンの各種トレース、メモリダンプなどのデバッグングエイドとして完備しており、ここでシステムを動作させたときのソフトのチェックもできるようになっている。

従来ハードウェア回路素子で組まれていた論理を持ったシステムに等しい、又は近い装置を、これから新しくマイクロプロセッサを



使ってつくるという場合に、その開発手段の合理化設備としては上に紹介したような装置にたよることになると考えられるので、従来と異なった検査の問題の延長にある問題点として説明した次第である。

応用システムの開発を計画されるとき、ここで紹介したような特殊コンピュータ、即ちデバックマシンとでも称すべきシステムも最近やっと市場に現れてきている。インテル社で発売されているMDSシステムも、この種の応用製品開発用のシステムの例である。

従来此種の装置はあまり必要が認識されなため市場に現れることのなかった機種で、マイクロプロセッサの応用というニーズにより始めて具体化した。これも全く新しい応用例の1つということも出来る。

#### 3.4.4 メーカとユーザの分担のあり方

マイクロプロセッサの今後の課題の第3は、ソフトウェアとドキュメンテーションのサポート体制ということである。ソフトウェアについては第2章に明電舎の伊東昭宏氏によって詳しく論じられているので、ここでは少し視点を変えて、マイクロプロセッサを制御に应用する場合のメーカとユーザの分担はどのようなものであればよいか？果して電子計算機一般と同じであろうか？という点を焦点として、課題の第4である標準化とセカンドソースの問題、第5である教育の問題を含めて検討してみよう。

この問題の最も素朴で、単純な分担は、ハードウェアはメーカ、ソフトウェアはユーザという型である。一般の電算機については自明のことであるこの分担のあり方が、マイクロプロセッサについてそのまま適用されるであろうか。筆者も現在若干疑問に思っているこの点を、ここで考察を加えることにしよう。

##### 一般の電子計算機の場合

ソフトウェアの本質が、電子計算機というハードウェアそのものに対しては、それを使用する技術であることから文字のとおり素直な形は、使用する側の担当する仕事になる筈である。米国を先頭とする電子計算

機の発達、この分担の定着化、普遍化を軸に普及してきたといえることができる。このための大きな推進力となったのは、メーカ側のサポート体制の充実、就中コンパイラの開発、が最も大きな要素であった。このサポートウェアの充実と販売におけるレンタル形式という特殊な製品販売のあり方が表裏一体となって、ユーザの広範な応用ソフトウェアの開発を促進し、電子計算機をして今日の発展をあらしめたといえることができる。

ところがこの効果を十分に発揮するには、電算機のハードウェアそのものは大型になり、豊富なリソースレパートリを収める程良いことになった。少くともある程度の規模でアーキテクチャの複雑さを必須としているといえることができる状態である。このことを次のミニコンがはじめて登場してきた当時をふり返って参考にしてみることにしよう。

#### ミニコンの場合

ミニコンがはじめて世に出た当時、そのコスト／パフォーマンス比の著しい向上が賞めはやされて、従来の感覚より一段と手軽に電子計算機を使い得る可能性が開かれ、広範な多彩な応用が開かれ、そのばら色の未来について、今日のマイクロプロセッサの登場に似た予想が樹てられたことがある。

ところが、その後の普及はたしかに従来の電算機よりは多方面に使われているとは云うものの、決して既存産業にインパクトを与える程のものにはなっていない。やはり電子計算機の進歩の一段階として、普及形とも云うべき機種範囲の拡大を生んだという以上のものではない。何故最初予想した程広範に活用されないで止まっているのか？

この原因の大きな要素は、ソフトウェアが従来の大型計算機程完備していなかったこと、そのメーカのサポート体制も従来のレンタルマシンのようなサービスは得られなかったことが指摘される。

ミニコンメーカが初めて製品を世に出したとき、その価格の安いことが最大の関心事であった反面、ソフトウェアについては、その経費回収の困難さに対する回避策として、サポートに対する責任を極力軽減し、その分価格を低くする事を優先させていたことは事実であった。

自ら電子計算機を生み出し、今日まで育ててきた国として、我国よりも電算機ハードウェア、ソフトウェアの基礎的知識が普及し、教育もはるかに広くおこなわれている米国で、ソフトウェアサポートを避けてユーザに開放したミニコンの当時のあり方は、ソフトウェアの壁をある程度乗り越えられそうな感覚で受け入れられたかに見えたのであるが、今日になって顧みれば、やはりその障害は重くその普及と産業構造を変革する程のものにはなっていないと云えよう。このことは、マイクロプロセッサについて今後の普及、方向付けを考える時参考とする必要がある。

#### マイクロプロセッサの場合

マイクロプロセッサの場合は、かつてミニコンについて電算機一般と多少差のあった事柄は更に著しくなってくる。敢えて列挙すれば下記のようになる。

- (1) マイクロプロセッサは、従来IC回路などを販売してきた部品メーカーの製品であり、メモリ、入出力インターフェース回路、電源などとの組み合わせも、ユーザに開放している点で一段と部品の性格の強いものである。従ってシステム商品的なソフトウェアサポート体制を持つことは、益々困難な面がある。
- (2) 価格の面からも、ミニコンよりも更に一桁この体制の期待し難さがある。
- (3) 従来はCPUはシステムの中心として頭脳が集まった形になっていたが、マイクロプロセッサ応用製品は、その中心は組み合わせた手定の方に移り、マイクロプロセッサが附加された部品の立場になり、大きなソフトウェア開発の対象とならない用途が多い。
- (4) 現在発表されつつあるマイクロプロセッサのアーキテクチャ命令のレパートリ共に、ビット数の制限、演算速度の限界、ゲート数の限界ピン数の制限などから、小規模に小規模にまとめられており、従来の大型電子計算機と同じ感覚でソフトウェアに汎用性を持たせ、充実した十分なサービスをおこなわせることは技術的にも困難があるし、ユーザが簡単にまとめるにも仕事を進めるのに、ミニコンよりも一段と負担が重くなる可能性が強い。

### ●機械語命令一覽表

| 分類          | 命令記号          | オペランド                         |                               |                               |                               | 命令長<br>バイト数 | 実行時間<br>サイクル数 | 説明                       | フラグ           | アドレスバス         |                | データバス          |                |
|-------------|---------------|-------------------------------|-------------------------------|-------------------------------|-------------------------------|-------------|---------------|--------------------------|---------------|----------------|----------------|----------------|----------------|
|             |               | D <sub>0</sub> D <sub>1</sub> | D <sub>2</sub> D <sub>3</sub> | D <sub>4</sub> D <sub>5</sub> | D <sub>6</sub> D <sub>7</sub> |             |               |                          |               | バス100<br>bit/s | バス100<br>bit/s | バス100<br>bit/s | バス100<br>bit/s |
| 基本命令        | MOV r, r      | 0 1                           | 0 0                           | 0 0                           | 0 0                           | 5           | 1             | (r) ← (r)                | X X X X X X   |                |                |                |                |
|             | MOV MOV r, r  | 0 1                           | 0 1                           | 0 0                           | 0 0                           | 5           | 1             | (r) ← (M)                | X X X X X X   | M              | M <sub>2</sub> | (r)            | 0              |
|             | MOV MOV r, r  | 0 0                           | 0 0                           | 0 0                           | 1 1                           | 0           | 7             | ただし M ← (M(L))           | X X X X X X   | M              | M <sub>2</sub> | (M)            | 1              |
|             | MOV MOV r, r  | 0 0                           | 0 0                           | 0 0                           | 1 1                           | 0           | 7             | (r) ← M                  | X X X X X X   | M              | M <sub>2</sub> | (M)            | 1              |
|             | MVI MVI r, r  | 0 0                           | 1 1                           | 0 1                           | 0 0                           | 3           | 10            | ただし M ← (M(L))           | X X X X X X   | M              | M <sub>2</sub> | (M)            | 1              |
|             | LRI LRI r, r  | 0 0                           | 0 0                           | 0 0                           | 0 1                           | 1           | 10            | (r) ← (B)                | X X X X X X   |                |                | (B)            | 1              |
|             | LRI LRI r, r  | 0 0                           | 0 1                           | 0 0                           | 0 0                           | 1           | 11            | (B) ← (B)                | X X X X X X   |                |                | (B)            | 1              |
|             | LRI LRI r, r  | 0 0                           | 0 1                           | 0 0                           | 0 0                           | 1           | 21            | (r) ← (B)                | X X X X X X   |                |                | (B)            | 1              |
|             | LRI LRI SP, r | 0 0                           | 1 1                           | 0 0                           | 0 0                           | 1           | 31            | (SP) ← M                 | X X X X X X   |                |                | (B)            | 1              |
|             | 加減乗除命令        | SRHL SRHL                     | 1 1                           | 1 1                           | 0 0                           | 0 0         | 2             | 1                        | (SP) ← (M(L)) | X X X X X X    |                |                |                |
| STAR STAR 0 |               | 0 0                           | 0 0                           | 0 0                           | 0 1                           | 0           | 2             | (B) ← (M)                | X X X X X X   | (B)(C)         | M <sub>2</sub> | (A)            | 0              |
| STAR STAR 0 |               | 0 0                           | 0 0                           | 0 0                           | 0 1                           | 0           | 2             | (D) ← (M)                | X X X X X X   | (D)(E)         | M <sub>2</sub> | (A)            | 0              |
| LDA LDA 0   |               | 0 0                           | 0 0                           | 0 1                           | 0 0                           | 0           | 2             | (A) ← (B)(C)             | X X X X X X   | (B)(C)         | M <sub>2</sub> | (B)(C)         | 1              |
| LDA LDA 0   |               | 0 0                           | 0 0                           | 0 1                           | 0 0                           | 0           | 2             | (A) ← (D)(E)             | X X X X X X   | (D)(E)         | M <sub>2</sub> | (D)(E)         | 1              |
| STA STA =   |               | 0 0                           | 1 0                           | 0 0                           | 0 0                           | 3           | 2             | (M) ← (A)                | X X X X X X   | M              | M <sub>2</sub> | (A)            | 0              |
| LDA LDA =   |               | 0 0                           | 1 1                           | 0 0                           | 0 0                           | 3           | 2             | (A) ← (M)                | X X X X X X   | M              | M <sub>2</sub> | (M)            | 1              |
| SHLD SHLD = |               | 0 0                           | 1 0                           | 0 0                           | 0 1                           | 2           | 18            | (M) ← (L)<br>(M+1) ← (H) | X X X X X X   | M+1            | M <sub>2</sub> | (L)            | 0              |
| LHLD LHLD = |               | 0 0                           | 1 0                           | 0 1                           | 0 0                           | 2           | 18            | (L) ← (M)<br>(H) ← (M+1) | X X X X X X   | M+1            | M <sub>2</sub> | (M)            | 1              |
| XRHD XRHD   |               | 1 1                           | 1 0                           | 0 0                           | 0 1                           | 1           | 8             | (H) ← (D)(E)             | X X X X X X   |                |                |                |                |
| 加減乗除命令      | XRHL XRHL     | 1 1                           | 1 0                           | 0 1                           | 0 0                           | 1           | 8             | (H) ← (D)(E)             | X X X X X X   |                |                |                |                |
|             | ADD ADD M     | 1 0                           | 0 0                           | 0 0                           | 0 0                           | 4           | 1             | (A) ← (A) + (M)          | X X X X X X   | M              | M <sub>2</sub> | (M)            | 1              |
|             | ADD ADD M     | 1 1                           | 0 0                           | 0 0                           | 0 0                           | 4           | 1             | (A) ← (A) + n            | X X X X X X   |                |                | (M)            | 1              |
|             | ADC ADC M     | 1 0                           | 0 0                           | 0 1                           | 0 0                           | 4           | 1             | (A) ← (A) + (M) + (CY)   | X X X X X X   | M              | M <sub>2</sub> | (M)            | 1              |
|             | ADC ADC M     | 1 1                           | 0 0                           | 0 1                           | 0 0                           | 4           | 1             | (A) ← (A) + n + (CY)     | X X X X X X   |                |                | (M)            | 1              |
|             | OAO OAO 0     | 0 0                           | 0 0                           | 0 0                           | 0 1                           | 0           | 10            | (M(L)) ← (M(L) + (B)(C)) | X X X X X X   |                |                |                |                |
|             | OAO OAO 0     | 0 0                           | 0 0                           | 0 0                           | 0 1                           | 0           | 10            | (M(L)) ← (M(L) + (D)(E)) | X X X X X X   |                |                |                |                |
|             | OAO OAO SP    | 0 0                           | 0 0                           | 0 1                           | 0 0                           | 1           | 29            | (M(L)) ← (M(L) + (M(L))) | X X X X X X   |                |                |                |                |
|             | SUB SUB M     | 1 0                           | 0 0                           | 0 0                           | 0 0                           | 4           | 1             | (M(L)) ← (M(L)) - (M)    | X X X X X X   | M              | M <sub>2</sub> | (M)            | 1              |
|             | SUB SUB M     | 1 1                           | 0 0                           | 0 0                           | 0 0                           | 4           | 1             | (M(L)) ← (M(L)) - n      | X X X X X X   |                |                | (M)            | 1              |
| 加減乗除命令      | SBB SBB M     | 1 0                           | 0 1                           | 0 0                           | 0 0                           | 4           | 1             | (A) ← (A) - (M) + (CY)   | X X X X X X   | M              | M <sub>2</sub> | (M)            | 1              |
|             | SBB SBB M     | 1 1                           | 0 1                           | 0 0                           | 0 0                           | 4           | 1             | (A) ← (A) - n            | X X X X X X   |                |                | (M)            | 1              |
|             | ANA ANA M     | 1 0                           | 1 0                           | 0 0                           | 0 0                           | 4           | 1             | (A) ← (A) AND (M)        | X X X X X X   | M              | M <sub>2</sub> | (M)            | 1              |
|             | ANA ANA M     | 1 1                           | 1 0                           | 0 0                           | 0 0                           | 4           | 1             | (A) ← (A) AND n          | X X X X X X   |                |                | (M)            | 1              |
|             | ORA ORA M     | 1 0                           | 1 0                           | 0 1                           | 0 0                           | 4           | 1             | (A) ← (A) OR (M)         | X X X X X X   | M              | M <sub>2</sub> | (M)            | 1              |
|             | ORA ORA M     | 1 1                           | 1 0                           | 0 1                           | 0 0                           | 4           | 1             | (A) ← (A) OR n           | X X X X X X   |                |                | (M)            | 1              |
|             | XRA XRA M     | 1 0                           | 1 1                           | 0 1                           | 0 0                           | 4           | 1             | (A) ← (A) XOR (M)        | X X X X X X   | M              | M <sub>2</sub> | (M)            | 1              |
|             | XRA XRA M     | 1 1                           | 1 1                           | 0 1                           | 0 0                           | 4           | 1             | (A) ← (A) XOR n          | X X X X X X   |                |                | (M)            | 1              |
|             | ORA ORA M     | 1 0                           | 1 1                           | 0 1                           | 0 0                           | 4           | 1             | (A) ← (A) OR (M)         | X X X X X X   | M              | M <sub>2</sub> | (M)            | 1              |
|             | ORA ORA M     | 1 1                           | 1 1                           | 0 1                           | 0 0                           | 4           | 1             | (A) ← (A) OR n           | X X X X X X   |                |                | (M)            | 1              |
| 加減乗除命令      | CMP CMP M     | 1 0                           | 1 1                           | 0 1                           | 0 0                           | 4           | 1             | (A) - (M)                | X X X X X X   | M              | M <sub>2</sub> | (M)            | 1              |
|             | CMP CMP M     | 1 1                           | 1 1                           | 0 1                           | 0 0                           | 4           | 1             | (A) - n                  | X X X X X X   |                |                | (M)            | 1              |
|             | JMP JMP M     | 1 0                           | 1 1                           | 1 1                           | 0 0                           | 3           | 5             | (r) ← (M)                | X X X X X X   | M              | M <sub>2</sub> | (M)            | 1              |
|             | JMP JMP M     | 1 1                           | 1 1                           | 1 1                           | 0 0                           | 3           | 5             | (r) ← (M)                | X X X X X X   | M              | M <sub>2</sub> | (M)            | 1              |
|             | JMP JMP M     | 0 0                           | 0 0                           | 0 0                           | 0 0                           | 3           | 5             | (r) ← (r)                | X X X X X X   |                |                | (M)            | 1              |
|             | JMP JMP M     | 0 0                           | 0 0                           | 0 0                           | 0 0                           | 3           | 5             | (r) ← (r)                | X X X X X X   |                |                | (M)            | 1              |
|             | JMP JMP M     | 0 0                           | 0 0                           | 0 0                           | 0 0                           | 3           | 5             | (r) ← (r)                | X X X X X X   |                |                | (M)            | 1              |
|             | JMP JMP M     | 0 0                           | 0 0                           | 0 0                           | 0 0                           | 3           | 5             | (r) ← (r)                | X X X X X X   |                |                | (M)            | 1              |
|             | JMP JMP M     | 0 0                           | 0 0                           | 0 0                           | 0 0                           | 3           | 5             | (r) ← (r)                | X X X X X X   |                |                | (M)            | 1              |
|             | JMP JMP M     | 0 0                           | 0 0                           | 0 0                           | 0 0                           | 3           | 5             | (r) ← (r)                | X X X X X X   |                |                | (M)            | 1              |
| 加減乗除命令      | JMP JMP M     | 0 0                           | 0 0                           | 0 0                           | 0 0                           | 3           | 5             | (r) ← (r)                | X X X X X X   |                |                | (M)            | 1              |
|             | JMP JMP M     | 0 0                           | 0 0                           | 0 0                           | 0 0                           | 3           | 5             | (r) ← (r)                | X X X X X X   |                |                | (M)            | 1              |
|             | JMP JMP M     | 0 0                           | 0 0                           | 0 0                           | 0 0                           | 3           | 5             | (r) ← (r)                | X X X X X X   |                |                | (M)            | 1              |
|             | JMP JMP M     | 0 0                           | 0 0                           | 0 0                           | 0 0                           | 3           | 5             | (r) ← (r)                | X X X X X X   |                |                | (M)            | 1              |
|             | JMP JMP M     | 0 0                           | 0 0                           | 0 0                           | 0 0                           | 3           | 5             | (r) ← (r)                | X X X X X X   |                |                | (M)            | 1              |
|             | JMP JMP M     | 0 0                           | 0 0                           | 0 0                           | 0 0                           | 3           | 5             | (r) ← (r)                | X X X X X X   |                |                | (M)            | 1              |
|             | JMP JMP M     | 0 0                           | 0 0                           | 0 0                           | 0 0                           | 3           | 5             | (r) ← (r)                | X X X X X X   |                |                | (M)            | 1              |
|             | JMP JMP M     | 0 0                           | 0 0                           | 0 0                           | 0 0                           | 3           | 5             | (r) ← (r)                | X X X X X X   |                |                | (M)            | 1              |
|             | JMP JMP M     | 0 0                           | 0 0                           | 0 0                           | 0 0                           | 3           | 5             | (r) ← (r)                | X X X X X X   |                |                | (M)            | 1              |
|             | JMP JMP M     | 0 0                           | 0 0                           | 0 0                           | 0 0                           | 3           | 5             | (r) ← (r)                | X X X X X X   |                |                | (M)            | 1              |
| 加減乗除命令      | JMP JMP M     | 0 0                           | 0 0                           | 0 0                           | 0 0                           | 3           | 5             | (r) ← (r)                | X X X X X X   |                |                | (M)            | 1              |
|             | JMP JMP M     | 0 0                           | 0 0                           | 0 0                           | 0 0                           | 3           | 5             | (r) ← (r)                | X X X X X X   |                |                | (M)            | 1              |
|             | JMP JMP M     | 0 0                           | 0 0                           | 0 0                           | 0 0                           | 3           | 5             | (r) ← (r)                | X X X X X X   |                |                | (M)            | 1              |
|             | JMP JMP M     | 0 0                           | 0 0                           | 0 0                           | 0 0                           | 3           | 5             | (r) ← (r)                | X X X X X X   |                |                | (M)            | 1              |
|             | JMP JMP M     | 0 0                           | 0 0                           | 0 0                           | 0 0                           | 3           | 5             | (r) ← (r)                | X X X X X X   |                |                | (M)            | 1              |
|             | JMP JMP M     | 0 0                           | 0 0                           | 0 0                           | 0 0                           | 3           | 5             | (r) ← (r)                | X X X X X X   |                |                | (M)            | 1              |
|             | JMP JMP M     | 0 0                           | 0 0                           | 0 0                           | 0 0                           | 3           | 5             | (r) ← (r)                | X X X X X X   |                |                | (M)            | 1              |
|             | JMP JMP M     | 0 0                           | 0 0                           | 0 0                           | 0 0                           | 3           | 5             | (r) ← (r)                | X X X X X X   |                |                | (M)            | 1              |
|             | JMP JMP M     | 0 0                           | 0 0                           | 0 0                           | 0 0                           | 3           | 5             | (r) ← (r)                | X X X X X X   |                |                | (M)            | 1              |
|             | JMP JMP M     | 0 0                           | 0 0                           | 0 0                           | 0 0                           | 3           | 5             | (r) ← (r)                | X X X X X X   |                |                | (M)            | 1              |
| 加減乗除命令      | JMP JMP M     | 0 0                           | 0 0                           | 0 0                           | 0 0                           | 3           | 5             | (r) ← (r)                | X X X X X X   |                |                | (M)            | 1              |
|             | JMP JMP M     | 0 0                           | 0 0                           | 0 0                           | 0 0                           | 3           | 5             | (r) ← (r)                | X X X X X X   |                |                | (M)            | 1              |
|             | JMP JMP M     | 0 0                           | 0 0                           | 0 0                           | 0 0                           | 3           | 5             | (r) ← (r)                | X X X X X X   |                |                | (M)            | 1              |
|             | JMP JMP M     | 0 0                           | 0 0                           | 0 0                           | 0 0                           | 3           | 5             | (r) ← (r)                | X X X X X X   |                |                | (M)            | 1              |
|             | JMP JMP M     | 0 0                           | 0 0                           | 0 0                           | 0 0                           | 3           | 5             | (r) ← (r)                | X X X X X X   |                |                | (M)            | 1              |
|             | JMP JMP M     | 0 0                           | 0 0                           | 0 0                           | 0 0                           | 3           | 5             | (r) ← (r)                | X X X X X X   |                |                | (M)            | 1              |
|             | JMP JMP M     | 0 0                           | 0 0                           | 0 0                           | 0 0                           | 3           | 5             | (r) ← (r)                | X X X X X X   |                |                | (M)            | 1              |
|             | JMP JMP M     | 0 0                           | 0 0                           | 0 0                           | 0 0                           | 3           | 5             | (r) ← (r)                | X X X X X X   |                |                | (M)            | 1              |
|             | JMP JMP M     | 0 0                           | 0 0                           | 0 0                           | 0 0                           | 3           | 5             | (r) ← (r)                | X X X X X X   |                |                | (M)            | 1              |
|             | JMP JMP M     | 0 0                           | 0 0                           | 0 0                           | 0 0                           | 3           | 5             | (r) ← (r)                | X X X X X X   |                |                | (M)            | 1              |
| 加減乗除命令      | JMP JMP M     | 0 0                           | 0 0                           | 0 0                           | 0 0                           | 3           | 5             | (r) ← (r)                | X X X X X X   |                |                | (M)            | 1              |
|             | JMP JMP M     | 0 0                           | 0 0                           | 0 0                           | 0 0                           | 3           | 5             | (r) ← (r)                | X X X X X X   |                |                | (M)            | 1              |
|             | JMP JMP M     | 0 0                           | 0 0                           | 0 0                           | 0 0                           | 3           | 5             | (r) ← (r)                | X X X X X X   |                |                | (M)            | 1              |
|             | JMP JMP M     | 0 0                           | 0 0                           | 0 0                           | 0 0                           | 3           | 5             | (r) ← (r)                | X X X X X X   |                |                | (M)            | 1              |
|             | JMP JMP M     | 0 0                           | 0 0                           | 0 0                           | 0 0                           | 3           | 5             | (r) ← (r)                | X X X X X X   |                |                | (M)            | 1              |
|             | JMP JMP M     | 0 0                           | 0 0                           | 0 0                           | 0 0                           | 3           | 5             | (r) ← (r)                | X X X X X X   |                |                | (M)            | 1              |
|             | JMP JMP M     | 0 0                           | 0 0                           | 0 0                           | 0 0                           | 3           | 5             | (r) ← (r)                | X X X X X X   |                |                | (M)            | 1              |
|             | JMP JMP M     | 0 0                           | 0 0                           | 0 0                           | 0 0                           | 3           | 5             | (r) ← (r)                | X X X X X X   |                |                | (M)            | 1              |
|             | JMP JMP M     | 0 0                           | 0 0                           | 0 0                           | 0 0                           | 3           | 5             | (r) ← (r)                | X X X X X X   |                |                | (M)            | 1              |
|             | JMP JMP M     | 0 0                           | 0 0                           | 0 0                           | 0 0                           | 3           | 5             | (r) ← (r)                | X X X X X X   |                |                | (M)            | 1              |
| 加減乗除命令      | JMP JMP M     | 0 0                           | 0 0                           | 0 0                           | 0 0                           | 3           | 5             | (r) ← (r)                | X X X X X X   |                |                | (M)            | 1              |
|             | JMP JMP M     | 0 0                           | 0 0                           | 0 0                           | 0 0                           | 3           | 5             | (r) ← (r)                | X X X X X X   |                |                | (M)            | 1              |
|             | JMP JMP M     | 0 0                           | 0 0                           | 0 0                           | 0 0                           | 3           | 5             | (r) ← (r)                | X X X X X X   |                |                | (M)            | 1              |
|             | JMP JMP M     | 0 0                           | 0 0                           | 0 0                           | 0 0                           | 3           | 5             | (r) ← (r)                | X X X X X X   |                |                | (M)            | 1              |
|             | JMP JMP M     | 0 0                           | 0 0                           | 0 0                           | 0 0                           | 3           | 5             | (r) ← (r)                | X X X X X X   |                |                | (M)            | 1              |
|             | JMP JMP M     | 0 0                           | 0 0                           | 0 0                           | 0 0                           | 3           | 5             | (r) ← (r)                | X X X X X X   |                |                | (M)            | 1              |
|             | JMP JMP M     | 0 0                           | 0 0                           | 0 0                           | 0 0                           | 3           | 5             | (r) ← (r)                | X X X X X X   |                |                | (M)            | 1              |
|             | JMP JMP M     | 0 0                           | 0 0                           | 0 0                           | 0 0                           | 3           | 5             | (r) ← (r)                | X X X X X X   |                |                | (M)            | 1              |
|             | JMP JMP M     | 0 0                           | 0 0                           | 0 0                           | 0 0                           | 3           | 5             | (r) ← (r)                | X X X X X X   |                |                | (M)            | 1              |
|             | JMP JMP M     | 0 0                           | 0 0                           | 0 0                           | 0 0                           | 3           | 5             | (r) ← (r)                | X X X X X X   |                |                | (M)            | 1              |
| 加減乗除命令      | JMP JMP M     | 0 0                           | 0 0                           | 0 0                           | 0 0                           | 3           | 5             | (r) ← (r)                | X X X X X X   |                |                | (M)            | 1              |
|             | JMP JMP M     | 0 0                           | 0 0                           | 0 0                           | 0 0                           | 3           | 5             | (r) ← (r)                | X X X X X X   |                |                | (M)            | 1              |
|             | JMP JMP M     | 0 0                           | 0 0                           | 0 0                           | 0 0                           | 3           | 5             | (r) ← (r)                | X X X X X X   |                |                | (M)            | 1              |
|             | JMP JMP M     | 0 0                           | 0 0                           | 0 0                           | 0 0                           | 3           | 5             | (r) ← (r)                | X X X X X X   |                |                | (M)            | 1              |
|             | JMP JMP M     | 0 0                           | 0 0                           | 0 0                           | 0 0                           | 3           | 5             | (r) ← (r)                | X X X X X X   |                |                | (M)            | 1              |
|             | JMP JMP M     | 0 0                           | 0 0                           | 0 0                           | 0 0                           | 3           | 5             | (r) ← (r)                | X X X X X X   |                |                | (M)            | 1              |
|             | JMP JMP M     | 0 0                           | 0 0                           | 0 0                           | 0 0                           | 3           | 5             | (r) ← (r)                | X X X X X X   |                |                | (M)            | 1              |
|             | JMP JMP M     | 0 0                           | 0 0                           | 0 0                           | 0 0                           | 3           | 5             | (r) ← (r)                | X X X X X X   |                |                | (M)            | 1              |
|             | JMP JMP M     | 0 0                           | 0 0                           | 0 0                           | 0 0                           | 3           | 5             | (r) ← (r)                | X X X X X X   |                |                | (M)            | 1              |
|             | JMP JMP M     | 0 0                           | 0 0                           | 0 0                           | 0 0                           | 3           | 5             | (r) ← (r)                | X X X X X X   |                |                | (M)            | 1              |
| 加減乗除命令      | JMP JMP M     | 0 0                           | 0 0                           | 0 0                           | 0 0                           | 3           | 5             | (r) ← (r)                | X X X X X X   |                |                | (M)            | 1              |
|             | JMP JMP M     | 0 0                           | 0 0                           | 0 0                           | 0 0                           | 3           | 5             | (r) ← (r)                | X X X X X X   |                |                | (M)            | 1              |
|             | JMP JMP M     | 0 0                           | 0 0                           | 0 0                           | 0 0                           | 3           | 5             | (r) ← (r)                | X X X X X X   |                |                | (M)            | 1              |
|             | JMP JMP M     | 0 0                           | 0 0                           | 0 0                           | 0 0                           | 3           | 5             | (r) ← (r)                | X X X X X X   |                |                | (M)            | 1              |
|             | JMP JMP M     | 0 0                           | 0 0                           | 0 0                           | 0 0                           | 3           | 5             | (r) ← (r)                | X X X X X X   |                |                | (M)            | 1              |
|             | JMP JMP M     | 0 0                           | 0 0                           | 0 0                           | 0 0                           | 3           | 5             | (r) ← (r)                | X X X X X X   |                |                | (M)            | 1              |
|             | JMP JMP M     | 0 0                           | 0 0                           | 0 0                           | 0 0                           | 3           | 5             | (r) ← (r)                | X X X X X X   |                |                | (M)            | 1              |
|             | JMP JMP M     | 0 0                           | 0 0                           | 0 0                           | 0 0                           | 3           | 5             | (r) ← (r)                | X X X X X X   |                |                | (M)            | 1              |
|             | JMP JMP M     | 0 0                           | 0 0                           | 0 0                           | 0 0                           | 3           | 5             | (r) ← (r)                | X X X X X X   |                |                | (M)            | 1              |
|             | JMP JMP M     | 0 0                           | 0 0                           | 0 0                           |                               |             |               |                          |               |                |                |                |                |

# M58710S 8 BIT PARALLEL CPU

| 命令   | 命令記号 | 命令コード      | 命令形式      | 命令説明                                                                 | フラグ    | アドレスバス           | データバス                          |
|------|------|------------|-----------|----------------------------------------------------------------------|--------|------------------|--------------------------------|
| CALL | CALL | 11 001 101 | CD 17 3 5 | $((SP) - 1)((SP) - 2) - (PC) + 3, (PC) \leftarrow ((SP) - (SP) - 2)$ | XXXXXX | (SP)-1<br>(SP)-2 | (B)<br>(B)<br>(PC)-1<br>(PC)-2 |
| RET  | RET  | 11 011 100 | CE 17 3 5 | $(CY) = 1$                                                           | XXXXXX |                  |                                |
| CNC  | CNC  | 11 010 100 | CF 17 3 5 | $(CY) = 0$                                                           | XXXXXX |                  |                                |
| CZ   | CZ   | 11 001 100 | CC 17 3 5 | $(Z) = 1$                                                            | XXXXXX |                  |                                |
| DNZ  | DNZ  | 11 000 100 | CD 17 3 5 | $(Z) = 0$                                                            | XXXXXX | (SP)-1<br>(SP)-2 | (B)<br>(B)<br>(PC)-1<br>(PC)-2 |
| CP   | CP   | 11 110 100 | CF 17 3 5 | $(P) = 0$                                                            | XXXXXX |                  |                                |
| CM   | CM   | 11 111 100 | CF 17 3 5 | $(P) = 1$                                                            | XXXXXX |                  |                                |
| CPE  | CPE  | 11 101 100 | CE 17 3 5 | $(P) = 1$                                                            | XXXXXX |                  |                                |
| CPO  | CPO  | 11 100 100 | CE 17 3 5 | $(P) = 0$                                                            | XXXXXX |                  |                                |
| HST  | HST  | 11 AAA 111 | 11 11 3   | $((SP) - 1)((SP) - 2) - (PC) + 1, (PC) \leftarrow ((SP) - (SP) - 2)$ | XXXXXX | (SP)-1<br>(SP)-2 | (PC)-1<br>(PC)-2               |
| INT  | INT  | 11 001 001 | 09 10 1 3 | $(PC) \leftarrow ((SP) + 1)((SP) - 2), (SP) \leftarrow ((SP) + 2)$   | XXXXXX | (SP)-1<br>(SP)-2 | (PC)-1<br>(PC)-2               |
| INC  | INC  | 11 011 000 | 0B 17 1 3 | $(CY) = 1$                                                           | XXXXXX |                  |                                |
| INC  | INC  | 11 010 000 | 0B 17 1 3 | $(CY) = 0$                                                           | XXXXXX |                  |                                |
| INC  | INC  | 11 001 000 | 0B 17 1 3 | $(Z) = 1$                                                            | XXXXXX |                  |                                |
| INC  | INC  | 11 000 000 | 0B 17 1 3 | $(Z) = 0$                                                            | XXXXXX |                  |                                |
| INC  | INC  | 11 110 000 | 0B 17 1 3 | $(P) = 1$                                                            | XXXXXX |                  |                                |
| INC  | INC  | 11 101 000 | 0B 17 1 3 | $(P) = 0$                                                            | XXXXXX |                  |                                |
| INC  | INC  | 11 100 000 | 0B 17 1 3 | $(P) = 1$                                                            | XXXXXX |                  |                                |
| INC  | INC  | 11 000 000 | 0B 17 1 3 | $(P) = 0$                                                            | XXXXXX |                  |                                |
| OUT  | OUT  | 11 010 011 | 03 10 2 3 | $(A) \leftarrow ((SP) + 1)((SP) - 2), (SP) \leftarrow ((SP) + 2)$    | XXXXXX | (B)<br>(B)       | (A)<br>(A)                     |
| IN   | IN   | 11 011 011 | 03 10 2 3 | $(A) \leftarrow ((SP) + 1)((SP) - 2), (SP) \leftarrow ((SP) + 2)$    | XXXXXX | (B)<br>(B)       | (A)<br>(A)                     |
| PUSH | PUSH | 11 110 101 | F5 11 1 3 | $(SP) \leftarrow (SP) - 1$                                           | XXXXXX | (SP)-1           | (B)                            |
| PUSH | PUSH | 11 000 101 | C5 11 1 3 | $(SP) \leftarrow (SP) - 2$                                           | XXXXXX | (SP)-1<br>(SP)-2 | (B)<br>(B)                     |
| PUSH | PUSH | 11 010 101 | D5 11 1 3 | $(SP) \leftarrow (SP) - 1$                                           | XXXXXX | (SP)-1           | (B)                            |
| PUSH | PUSH | 11 100 101 | E5 11 1 3 | $(SP) \leftarrow (SP) - 2$                                           | XXXXXX | (SP)-1<br>(SP)-2 | (B)<br>(B)                     |
| POP  | POP  | 11 110 001 | F1 11 1 3 | $(SP) \leftarrow (SP) + 1$                                           | XXXXXX | (SP)+1           | (B)                            |
| POP  | POP  | 11 000 001 | C1 11 1 3 | $(SP) \leftarrow (SP) + 2$                                           | XXXXXX | (SP)+1<br>(SP)+2 | (B)<br>(B)                     |
| POP  | POP  | 11 010 001 | D1 11 1 3 | $(SP) \leftarrow (SP) + 1$                                           | XXXXXX | (SP)+1           | (B)                            |
| POP  | POP  | 11 100 001 | E1 11 1 3 | $(SP) \leftarrow (SP) + 2$                                           | XXXXXX | (SP)+1<br>(SP)+2 | (B)<br>(B)                     |
| HLT  | HLT  | 01 110 110 | 7F 7 1 1  | $(PC) \leftarrow (PC) + 1$                                           | XXXXXX | (PC)+1           | (B)                            |
| NOP  | NOP  | 00 000 000 | 00 0 1 1  | $(PC) \leftarrow (PC) + 1$                                           | XXXXXX | (PC)+1           | (B)                            |

| 記号  | 内容                           |
|-----|------------------------------|
| -   | データの移動方向を示す。                 |
| ( ) | レジスタ、メモリなどの内容を示す。            |
| V   | 論理和。                         |
| ∧   | 論理積。                         |
| ≡   | 等価。                          |
| (B) | 命令の第2バイト。                    |
| (D) | 命令の第3バイト。                    |
| X   | 命令実行後フラグフリップフロップが不安。         |
| S   | 命令実行後フラグフリップフロップがセット。        |
| R   | 命令実行後フラグフリップフロップがリセット。       |
| SR  | 命令実行後フラグフリップフロップがセット、互換リセット。 |
| F   | フラグフリップフロップ(S,Z,P,CY,CV)を示す。 |

| 記号       | 内容                  |
|----------|---------------------|
| \$\$\$   | レジスタ又はメモリによって決まる値。  |
| △        | それ以外、下線のような値が付けられる。 |
| 000      | レジスタ又はメモリ           |
| B        | 000                 |
| C        | 001                 |
| D        | 010                 |
| E        | 011                 |
| H        | 100                 |
| L        | 101                 |
| M        | 110                 |
| N        | 111                 |
| M=(H)(L) |                     |
| AAA      | RST命令におけるnの2進表記。    |
| I        | 入力モードを示す。           |
| O        | 出力モードを示す。           |
| n        | 2バイトのデータ。           |
| R        | 1バイトのデータ。           |

## ●類似命令一覧表

| 命令        | 命令   | 命令               | 命令                                                        |
|-----------|------|------------------|-----------------------------------------------------------|
| アセンブラ制御命令 | NAM  | プログラムの名前を宣言する。   | プログラムの名前を宣言する。                                            |
|           | ORG  | プログラムのカウンタを設定する。 | 以下に続くプログラムの先頭位置を設定する。                                     |
|           | ROM  | ROM領域を宣言する。      | 以下に続くプログラムを、書き込み可能領域に変換することを宣言する。                         |
|           | RAM  | RAM領域を宣言する。      | 以下に続くプログラムを、書き込み可能領域に変換することを宣言する。                         |
|           | BLK  | ブロック宣言命令。        | 以下に続くプログラムのブロック番号を宣言する。                                   |
|           | END  | 終了宣言命令。          | プログラムの終了を宣言する。                                            |
| リンク記号設定命令 | ENT  | 入口名宣言命令。         | この命令で指定された数値記号が、外部のプログラムによって参照されることを宣言する。                 |
|           | EXT  | 外部数値記号宣言命令。      | この命令で指定された数値記号が、内部のプログラムを参照する記号であることを宣言する。                |
| 数値記号      | EOU  | 数値記号設定命令。        | ある数値記号にある数値または数値記号を設定する。                                  |
| メモリ内容宣言命令 | DEF  | データ設定命令。         | データ、または変数宣言を1バイト単位で設定する。                                  |
|           | DAOR | アドレス設定命令。        | 2バイトのデータを設定する。記号の命令は、その記号がメモリとして使用されるメモリに2バイトのデータとして設定する。 |
| 領域宣言命令    | BSS  | 領域宣言命令。          | 指定したバイト数だけ、メモリ領域を確保する。                                    |
| リスト制御命令   | EJE  | 改ページ宣言命令。        | アセンブラリストにおいて、改ページを指示する。                                   |

第3-17表は、現在の代表的な8ビットマイクロプロセッサの命令の一覧表の1例である。

マイクロプロセッサは、多くの機種共に1バイト分の巾をそのまま命令レパートリに充当しているので、1チップに収容された微少な形に似合わず、ミニコンに殆んど劣らぬ豊富な命令群となっている事はこの第3-17表の示すとおりである。

しかしながら、この命令群を生かして自由にどのようなプログラムでも組んで下さいと云われたユーザで、これを融通性に富んだ使い易いと思う人はどれほどいるであろうか？

そう感ずるのは、通常の電算機のアセンブラベースのプログラミング作業に習熟したプログラマであって、一般の制御技術者あるいは広範な従来の制御の仕事に関係してきた人々の多くは、このようにマシンの言語そのものに近いソフトウェアが生じる形で提示されること自身にとまどう事になるであろう。この段階では、マイクロプロセッサは一般の電算機よりも更に電子計算機に即したプログラミング作業が要求され、電子計算機よりも気軽に使い得るという事は全く期待できない。もしその仕事を簡単だと割切ることができたとしても、それは規模が小さかった場合にそう感ずる可能性があるという事に留まる。

現在一般の電算機の場合の多くは、ユーザが分担するソフトウェアは、メーカの製作しサポートするオペレーティングシステムの下でオペレーティングシステムにつつまれて動くようになっており、直接マイクロプロセッサの場合のようにユーザにまかされる事は少くなっている。第3-17表の命令レパートリを上手に使いこなすということは教育という面では、電算機のアーキテクチャに手軽にふれることができる利点があるというものの、その作業自身かなり複雑な仕事であり、ユーザにとってその種の仕事をおこなう負担はかなり重い。それはすでに、電子計算機のプログラマであることを期待していることになる。このことは、エンドユーザにとっては、マイクロプロセッサの応用ソフトを開発する間のみ必要とする仕事で、開発が完了すれば不要となる定常的でない仕事を意味し、人的投資という点からも甚だ効率の悪

い事にならざるを得ない。

特にユーザが広い範囲にまたがり、個別に多数多種のシステムが製作される制御への応用の場合、ソフトウェアサポートは重要であるが又マイクロプロセッサメーカーに期待することに困難がある。

結局、マイクロプロセッサの効果的普及のためには、メーカーとユニザの間に更に異質のメーカー（たとえばシステムメーカーと称し得るような）の介在が正常な姿にならざるを得ない。少くとも半導体プロセス技術を駆使して、高集積度セミコンダクタ製品をつくるという意味のメーカーとは全く異質のエンジニアリングを基礎とする部門が介在する必然性があることは、誰にも認められるところであると思う。更に、そのシステムメーカーのソフトウェアの武器は、それぞれの用途の範囲をカバーする問題向き言語にあるように思われる。第3-18表はこの主張の参考のために、三菱で採用している一般の電算機制御の場合の問題向き言語の例を示すものである。これと第3-17表を比較する時、第3-18表は制御技術者に使ってもらい意志があるとすれば第3-17表は電算機プログラマに使ってもらいつもりに留まっていると批評されてもやむをえないであろう。

現在までのところ電算機の応用について、事務処理の機械化に使用されるCOBOL、科学技術計算用に使用されているFORTRAN、ALGOL、両者から発展したPL-1等の高級プログラム言語が広い普及に果たしてきた役割は極めて大きい。が、制御の分野にはこれらに比較できるまで汎用化され実績の豊かな言語はない。従ってマイクロプロセッサを使うに当って、直ちにそのような便利なソフトウェアを期待することに若干の無理があるが、今後の期待と必然性を含めて考えると、第3-18表が一見して何人にも制御に使うには第3-17表より便利であること。従って、この種の問題向き言語が広い普及のために必要であることを認識して戴けると思う。

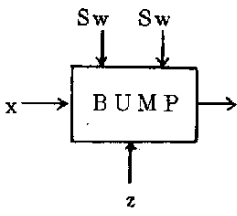
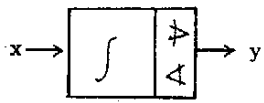
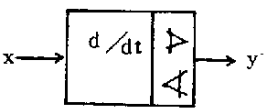
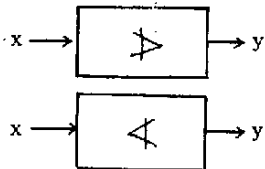
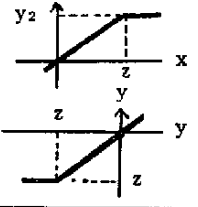
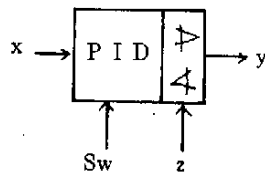
この点については、米国PURDUE大学のT.J.WILLIAMS教授を中心とする国際的ワークショップの積極的な活動が続いており、その成果が少しずつ現われつつあることに今後も期待をおくことができ

MDSS-7 アルゴリズム・パッケージ一覧表

| 名 称          | 記 号   | MPL言語での書き方                                                                                 | 備 考                                                                                                                                                                                                                                                                                                                                                                                      |       |       |       |       |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |
|--------------|-------|--------------------------------------------------------------------------------------------|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|-------|-------|-------|-------|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 論 理 積        |       | AND<br>INPUT= $x_1, x_2, \dots, x_n$<br>OUTPUT= $y$                                        | $y = x_1 \wedge x_2 \wedge \dots \wedge x_n$                                                                                                                                                                                                                                                                                                                                             |       |       |       |       |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |
| 論 理 和        |       | OR<br>INPUT= $x_1, x_2, \dots, x_n$<br>OUTPUT= $y$                                         | $y = x_1 \vee x_2 \vee \dots \vee x_n$                                                                                                                                                                                                                                                                                                                                                   |       |       |       |       |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |
| 排他的論理和       |       | EOR<br>INPUT= $x_1, x_2$<br>OUTPUT= $y$                                                    | $y = (x_1 \wedge \bar{x}_2) \vee (\bar{x}_1 \wedge x_2)$                                                                                                                                                                                                                                                                                                                                 |       |       |       |       |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |
| 否 定          |       | NOT<br>INPUT= $x$<br>OUTPUT= $y$                                                           | $y = \bar{x}$                                                                                                                                                                                                                                                                                                                                                                            |       |       |       |       |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |
| むだ時間<br>ロジカル |       | DTMLG<br>PAST= $p$<br>INPUT= $x$<br>OUTPUT= $y$<br>TIME= $t$                               |                                                                                                                                                                                                                                                                                                                                                                                          |       |       |       |       |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |
| フリップフロップ     |       | FLPFLP<br>PAST= $p$<br>SET= $x_1$<br>RESET= $x_2$<br>TRUE= $y_1$<br>FALSE = $y_2$          | <table border="1"> <thead> <tr> <th><math>x_1</math></th><th><math>x_2</math></th><th><math>y_1</math></th><th><math>y_2</math></th></tr> </thead> <tbody> <tr> <td>0</td><td>0</td><td>不</td><td>変</td></tr> <tr> <td>1</td><td>0</td><td>1</td><td>0</td></tr> <tr> <td>0</td><td>1</td><td>0</td><td>1</td></tr> <tr> <td>1</td><td>1</td><td>1</td><td>0</td></tr> </tbody> </table> | $x_1$ | $x_2$ | $y_1$ | $y_2$ | 0 | 0 | 不 | 変 | 1 | 0 | 1 | 0 | 0 | 1 | 0 | 1 | 1 | 1 | 1 | 0 |
| $x_1$        | $x_2$ | $y_1$                                                                                      | $y_2$                                                                                                                                                                                                                                                                                                                                                                                    |       |       |       |       |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |
| 0            | 0     | 不                                                                                          | 変                                                                                                                                                                                                                                                                                                                                                                                        |       |       |       |       |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |
| 1            | 0     | 1                                                                                          | 0                                                                                                                                                                                                                                                                                                                                                                                        |       |       |       |       |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |
| 0            | 1     | 0                                                                                          | 1                                                                                                                                                                                                                                                                                                                                                                                        |       |       |       |       |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |
| 1            | 1     | 1                                                                                          | 0                                                                                                                                                                                                                                                                                                                                                                                        |       |       |       |       |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |
| パルス発生器       |       | PLSGEN<br>PAST= $p$<br>OUTPUT= $y$<br>TIM <sub>1</sub> = $t_1$<br>TIM <sub>2</sub> = $t_2$ | 論理値を周期的に変化<br>                                                                                                                                                                                                                                                                                                                                                                           |       |       |       |       |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |
| サブタスクトリガ     |       | TRGR<br>SUBTSK= $i$<br>TRGSW= $swi$                                                        | <ul style="list-style-type: none"> <li>• Swi = T のとき: サブタスク i をビット</li> <li>• Sw = F のとき: 無動作</li> </ul>                                                                                                                                                                                                                                                                                 |       |       |       |       |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |
| タ イ マ        |       | TIMER<br>PAST= $p$<br>INPUT= $x$<br>OUTPUT= $y$<br>TIME= $t$                               |                                                                                                                                                                                                                                                                                                                                                                                          |       |       |       |       |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |

第 3-18 表-1



| 名 称                    | 記 号                                                                                 | MPL 言語での書き方                                                                                                                                                                           | 備 考                                                                                                                                                                                                                                                                                                              |
|------------------------|-------------------------------------------------------------------------------------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| バンプレス<br>トランスファ        |    | BUMP<br>PAST = p<br>STIME = $\Delta t$<br>PS = x<br>ADJUST = y<br>RTIME = t<br>TRACK = z<br>AMSW = swl<br>GHSW = sw                                                                   | <ul style="list-style-type: none"> <li>• swl = F (MANU) のとき :<br/><math>y = z</math></li> <li>• swl = T (AUTO) のとき :<br/>y の値を z <math>\rightarrow</math> x に徐々に近づける</li> </ul>                                                                                                                                  |
| 積 分                    |    | RESET<br>PAST = p<br>STIME = $\Delta t$<br>INPUT = x<br>OUTPUT = y<br>RTIME = t<br>HILIN = h<br>LOLIM = l                                                                             | $y = \left[ \frac{1}{ts} \right] x$                                                                                                                                                                                                                                                                              |
| 微 分                    |    | RATE<br>PAST = p<br>STIME = $\Delta t$<br>INPUT = x<br>OUTPUT = y<br>RTILIN = t<br>HILIN = h<br>LOLIN = l                                                                             | $y = \left[ \frac{ts}{1+ts} \right] x$                                                                                                                                                                                                                                                                           |
| リミッタ                   |   | HLILM<br>INPUT = x<br>OUTPUT = y<br>LIMIT = z<br>LOLZM<br>INPUT = x<br>OUTPUT = y<br>LIMIT = z                                                                                        |                                                                                                                                                                                                                               |
| PID<br>コントローラ<br>(位置形) |  | PIDAM<br>PAST = p<br>STIME = $\Delta t$<br>INPUT = x<br>OUTPUT = y<br>GAIN = R<br>RESET = t <sub>1</sub><br>RATE = t <sub>2</sub><br>HILIM = h<br>LOLIM = l<br>TRACK = z<br>ANSW = sw | <ul style="list-style-type: none"> <li>• SW = F (MANU) のとき :<br/><math>y = z</math></li> <li>• SW = T (AUTO) のとき :<br/><math>y = R \left[ \frac{1}{1+t_1s + 1+t_2s} \right] x</math></li> </ul> <p>ただし<br/> <math>\begin{cases} y &gt; h \rightarrow y = h \\ y &lt; l \rightarrow y = l \end{cases}</math> </p> |
| 一次遅れ                   |  | LAG<br>PAST = p<br>STIME = $\Delta t$<br>INPUT = x<br>OUTPUT = y<br>TIME = t                                                                                                          | $y = \left[ \frac{1}{1+ts} \right] x$                                                                                                                                                                                                                                                                            |

と思うが、今後その成果が最も期待され直接的効果を生み出すのは通常の電算機よりもむしろマイクロプロセッサを使ったマイクロコンピュータの制御用モデルとなるであろう。

尚、上記国際ワークショップには、我国からは電子工業振興協会がとりまとめて参加して、積極的に活躍しているようである。

本節の考察はソフトウェアを中心とし、その面からシステムメーカーの存立の必然性を示唆してきたが、この種、第3のメーカーの存在の意義は、ソフトウェアの問題に限られるわけではない。マイクロプロセッサが、月産数千個の規模の量産性を基礎にしてはじめて成立する製品であることから、半ば自然の結果としてかなりの汎用性を持つことになる。これは第3-17表の命令レパートリが物語っているとおりである。これを制御用として特殊化することは、チップのパターンに生かそうとすればすなわち量生産がくずれてくるし、逆にエンドユーザが個別に半導体チップに特殊仕様を盛り込んで製作しようとすることは、ユーザの本来の職務と異質の仕事を負担することにならざるを得ない。その間に相当の距離がある現在、その間を埋め、ハードウェアをROM化まで含めてサポートする任務を持つメーカーの存在は、必然性を持っていることが可能性として認められよう。その可能性がいかなる形で現実化されるかは今後の課題と云うべきである。

標準化の問題は、この種の分担が確立されれば、マイクロプロセッサの部品としての地位が明確になり、将来標準化の対象としてとらえられる可能性があるが、早急に具体化の見通しはあまり明るくないようである。むしろ、セカンドソースが第3-19表のように代表機種について確保されてきたために、システムメーカーにとって選択の基準がむしろ固まってきつつあると云えよう。

本節では、システムメーカーとも云うべき中間とりまとめメーカーの存在の必然性を論じたが、実際にどのような形で普及するかは、システムメーカーの附加価値、それに伴う経済性との兼合いで今後具体的に形づくられてゆくであろう。

| セカンドソースの現状 |                      |
|------------|----------------------|
| インテル 8080  | 日電、三菱、AMD、TI、COMSTAR |
| モトローラ 6800 | AMS、MOS-Tech         |
| インテル 4004  | NSC                  |
| インテル 3000  | Signetics            |

当協会海外視察団報告書による

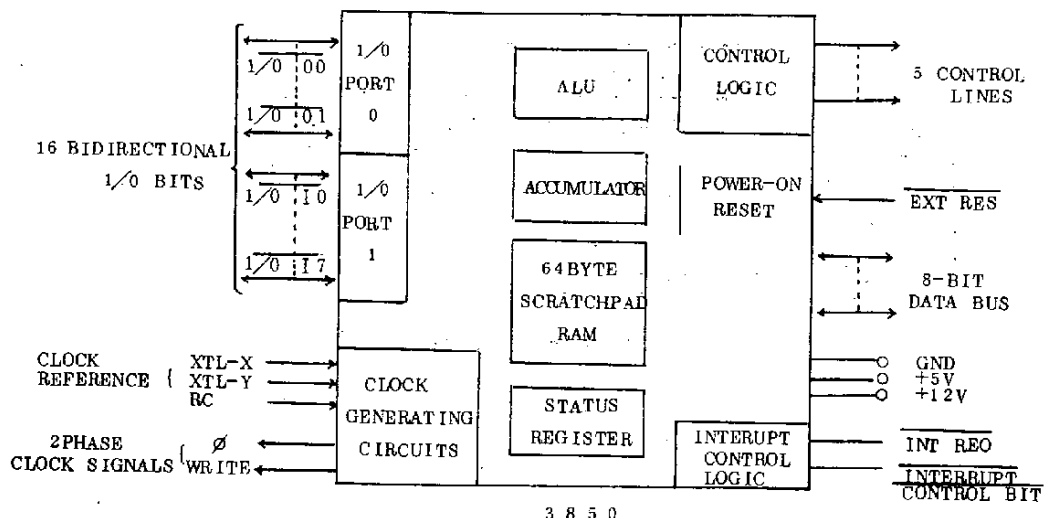
第3-19表

### 3.4.5. 制御用を目指した最近の新製品

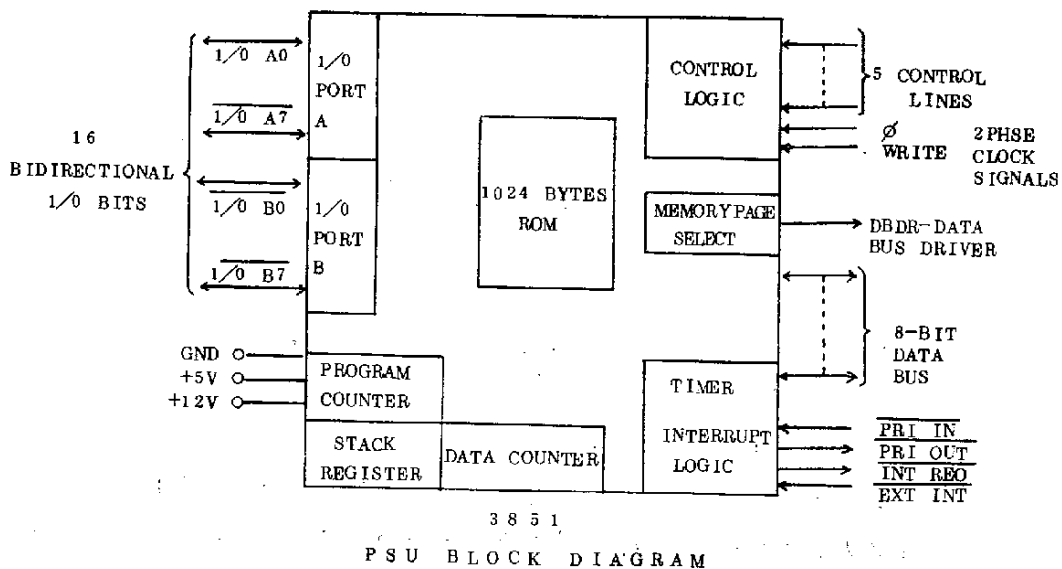
マイクロプロセッサの代表機種が第3-19表のように固ってきつつあると共に、別にバイポーラベースのビットスライスマイクロプロセッサチップが種々出現し、これは自由に組み合わせることにより既存のミニコンのエミュレーションを指向しているのが、昭和50年のマイクロプロセッサの姿である。ところが、これはどれもあまりにもコンピュータに近い、あるいはコンピュータそのものであるという批判を受ける面があることも又事実である。制御のためのプロセッサのあり方は、コンピュータのアーキテクチャを忠実にまねることが最適とは限らないという考え方も十分に成立する。この見地から、コンピュータアーキテクチャから一步を踏み出した新しい製品を見るのも意義のあることであると思う。

#### (1) FAIRCHILD F-8

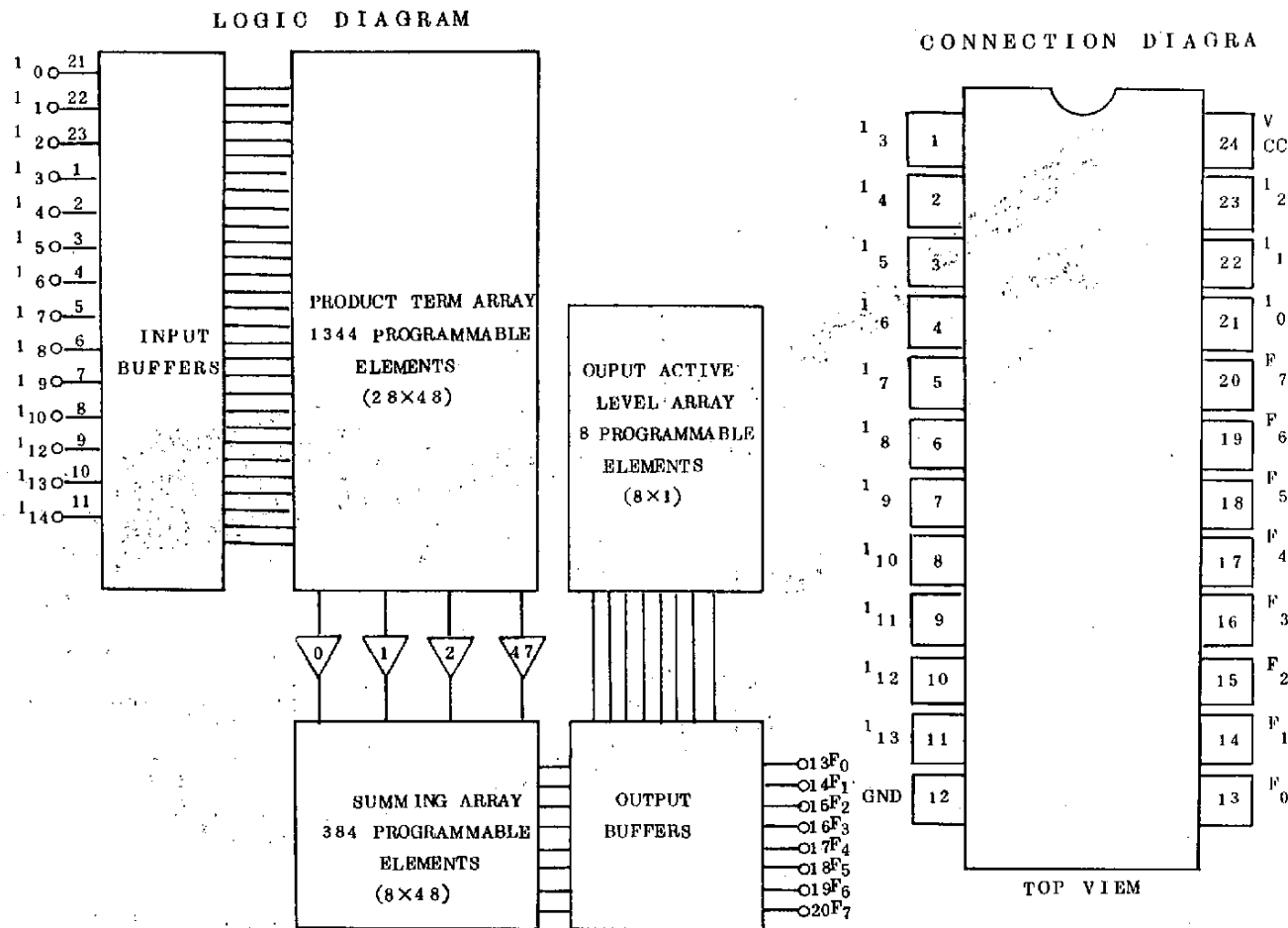
現実にマイクロプロセッサを使用して応用システムを製作しようとする、周囲に多数の回路を附加せねばならず、LSI化のメリットは減殺される。この点に着眼して、チップにできる限り必要な回路を収め、コンポーネントの合理的分配を考え直したことを特長とするマイクロプロセッサファミリである。第3-18図はCPU、第3-19図はメモリ部（プログラムストレージユニットと称している）を示す。第3-18図のCPUにはI/O PORTを2ヶ内蔵、クロック発生回路



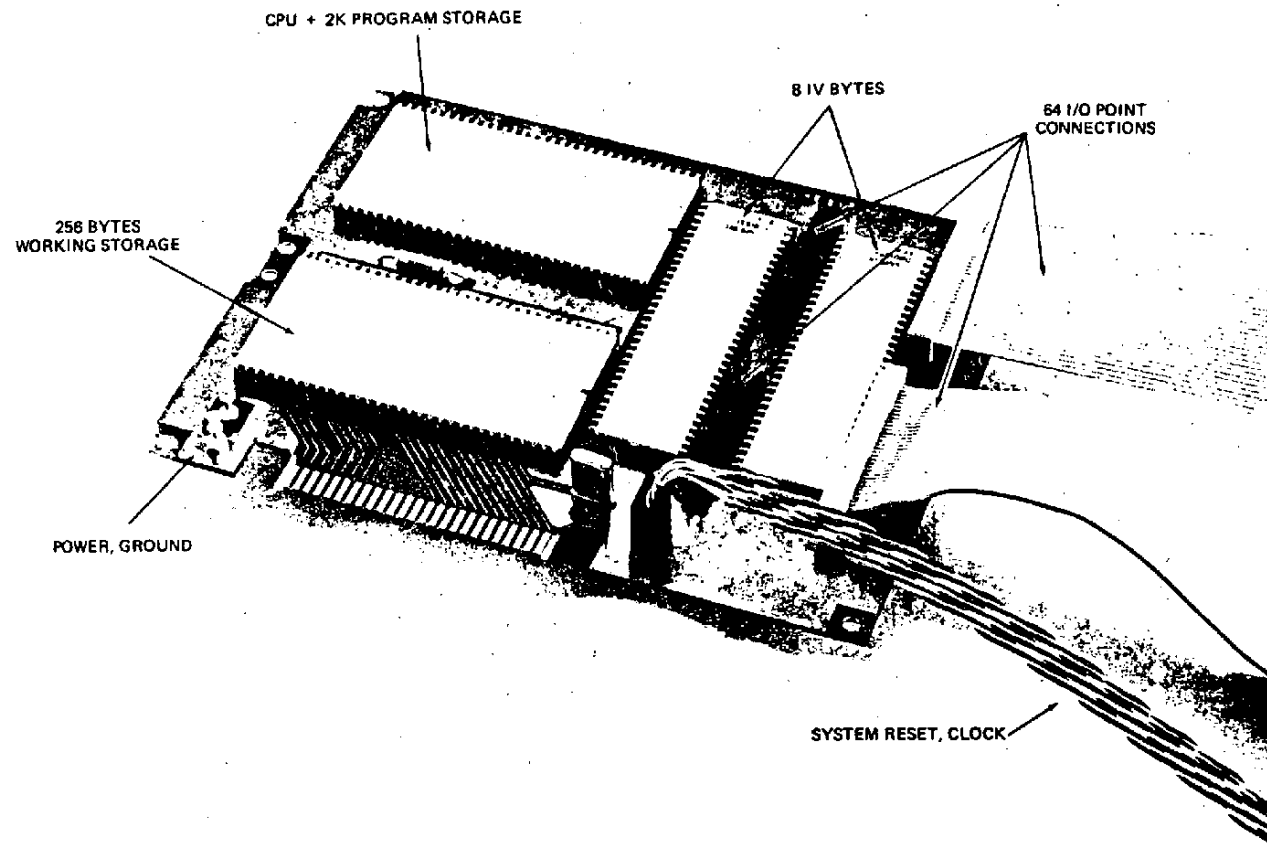
第 3 - 18 图



第 3 - 19 图



第 3 - 20 図 プログラマブルロジックアレー ( インタージール IM5200 の例 )



第 3 - 2 1 図 MICROCONTROLLER SYSTEM 20

を内蔵、更にスクラッチパッド用として64バイトのRAMを大胆に内蔵させると共に、逆にプログラムカウンタ、データカウンタ、そのアドレスロジック一式をメモリ側チップPSUに移している。またこれと共にPSU内に割込制御ロジックを一式持たせてある。

尚、命令レパートリは、情報処理用の複語命令等を省きコントロール用に集中させると共に、目立たないが内部の大きな特長としては、命令群をランダムロジックからプログラマブルロジックアレーに変え、チップ製作時のパターンを規則的なものにできるため、チップ寸法の縮小簡単化が、信頼性向上、歩止り上昇、コスト低減に効果があるということである。FAIRCHILDでは、現在このF-8シリーズLSIのチップレパートリ充実に努力を傾けている。

## (2) PLA

上記F-8に採用したロジックアレーを独立してLSIチップで提供する製品が、NSC、ロックウェル、ナショナル、インタージルのどのメーカーから出現し始めている。第3-20図はインタージルの製品の構成および接続図例である。左上の14点の入力およびその反転信号はプロダクトタームアレーで、48項の組み合わせのANDゲートで論理積となりその下のORゲートを通り、8本の出力が生成される。従ってそのゲートの組み合わせにより第3-6図の(3)に例示したような演算を施すことができるという例である。その構成がLSIで製作するのに適しているため最近再び注目を集め、今後の技術進歩と共に広範に使用される可能性が認められつつあるので、ここにごく概要のみ紹介しておくので、マイクロプロセッサと共に今後の技術進歩に注目して戴きたい。

## (3) SMS社のマイクロコントローラ

制御専用を目標とし、従来のマイクロプロセッサのあまりにも計算機寄りの性格を脱却し、思い切って命令を8種類(MOVE、ADD、AND、XOR、XMIT、JUMP、NZT、XEC)に絞りバイポーラプロセスを使用し、高速化(300ns)、周辺とメモリを含めてハイブリッドLSIチップにまとめたユニークなマイクロプロセッサである。第3

— 21 図はその構成図で、CPUと2ケのROMが1ケのチップ、256バイトのRAMが1ケのチップ、8ケの入出力ベクトルを1ケのチップとしてそれを2個、合計4ケのチップだけでマイクロコントローラとしての全機能を持ったもの（最小構成）がまとまってしまっている。汎用ではなく、コントローラ用を最初から目指した製品としてユニークであり、速度がもう1桁上昇し価格が落着けば、特殊の市場を持つようになる可能性のある製品であるということができよう。尚、このマイクロコントローラには、3.4.3節で必要性を説明したデバッグ用のコンピュータがマイクロコントローラシュミレータとして発売されている。

この種の特種な製品が、量産規模に達するマーケットを獲得するかどうかは今後注目すべきところであるが、応用分野拡大の規模からは、充分制御に適した特殊性を生かした量産機種に到達する可能性が大きいと思う。

### 3.5 マイクロプロセッサに最も適した用途

#### 3.5.1 応用分野と使い方

マイクロプロセッサを制御に使用する場合に、上手に使用すれば優れた特長を発揮する可能性があることが認められたと思うので、本節ではその上手な使い方をもう少し具体的にしらべてみる事にする。

第3—20表は国内のマイクロプロセッサの応用を計画しているところを、アンケート調査からまとめたものである。（昭和49年現在日本電子工業振興協会）この表によれば制御に関連する殆んど分野にも、いろいろな使い方採用されようとしていることがわかる。そこで本表に示された応用分野のいくつかを具体例ベースに、何のような点に特長を生かそうと計画しているかを調べて、マイクロプロセッサに適した使い方を見きわめるガイドにすることにしようと思う。



B マイクロコンピュータに適切な応用分野

| №  | 応 用 分 野                                                                                                                                                                  | 件数  | 構成比%  |
|----|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------|-----|-------|
| 1  | データ処理<br>科学技術計算、データ処理、データ集録、<br>データログ、各種窓口業務、データ変換、<br>データ検索 etc<br>ビルディングマシンシステム、在庫管理、工程管理<br>在庫計算、財務管理、伝票発行システム、<br>売上管理 etc                                           | 73  | 14.6  |
| 2  | 通信情報<br>ネットワーク制御<br>通信制御、通信回路制御、遠隔制御、<br>データ伝送コントロール etc                                                                                                                 | 13  | 2.6   |
| 3  | プロセスコントロール<br>プロセスコントロール、DDC、自動制御、<br>プラント制御、シーケンスコントロール、<br>ロボット制御、プラントシュミレータ、<br>工程ライン制御、生産設備の制御 etc                                                                   | 66  | 13.2  |
| 4  | マシンコントロール<br>I/Oコントロール、デバイス制御<br>ハンドリング装置、自動製図機、<br>簡単な機器の制御 etc                                                                                                         | 60  | 12.0  |
| 5  | システムコントロール<br>システムコントロール、テストシステムのコン<br>トロール、プログラム制御、放送番組システム<br>リアルタイム制御、自動倉庫 etc                                                                                        | 13  | 2.6   |
| 6  | 医療システム<br>医療システム、医療機器 etc                                                                                                                                                | 11  | 2.2   |
| 7  | 計測分析システム<br>数値制御、計測制御、計測システム、公害計測<br>センサの制御、音声分析、分析機器、<br>自動計測、試験分析 etc                                                                                                  | 76  | 15.6  |
| 8  | 交通管理システム<br>交通制御、ATC、タクシー配車管理<br>駐車場管理 etc                                                                                                                               | 11  | 2.2   |
| 9  | 教育システム<br>教育用コンピュータ、教育設備 etc                                                                                                                                             | 4   | 0.8   |
| 10 | 監視システム<br>回線監視装置、公害監視システム、<br>自動監視、テレメータの監視 etc                                                                                                                          | 6   | 1.2   |
| 11 | そ の 他<br>家庭用コンピュータ、家庭電器、高級電卓、<br>インタリジェントターミナル、自動車、<br>POS、各種事務機、ゲーム・マシン、<br>周辺端末機器、ランダムロジックの置きかえ、<br>各種計測器、通信機器、生産機器、<br>簡単なシステム、簡単な自動化、小型交換機、<br>高級玩具、テープの編集、コード交換 etc | 168 | 33.4  |
|    | 計                                                                                                                                                                        | 501 | 100.0 |

第 3 - 20 表



31

姫路第二発電所

第 3-22 图 (A)



年 月 日

電機

# 6号機性能計算日誌

姫路第二発電所

| 項目<br>時間 | 発電<br>電力<br>量 | 送電<br>電力<br>量 | 送電<br>効率<br>率 |        | 発電<br>効率<br>率 | 発電<br>効率<br>率 |        | ボイラ<br>効率<br>率 |        | ター<br>ビン<br>効率<br>率 | 制御不可能な効率偏差  |        |                                                | 制御可能な効率偏差        |                  |                                           |                                 |                            |                                      |                                 |                       |                                 |                                      | 所内<br>電力<br>効率<br>偏差 | 有効<br>CWP<br>運転<br>台数 | 有効<br>CWP<br>異<br>常<br>度 |                  |  |  |
|----------|---------------|---------------|---------------|--------|---------------|---------------|--------|----------------|--------|---------------------|-------------|--------|------------------------------------------------|------------------|------------------|-------------------------------------------|---------------------------------|----------------------------|--------------------------------------|---------------------------------|-----------------------|---------------------------------|--------------------------------------|----------------------|-----------------------|--------------------------|------------------|--|--|
|          |               |               | 基<br>準        | 実<br>績 |               | 未<br>正<br>誤   | 正<br>誤 | 未<br>正<br>誤    | 正<br>誤 |                     | 未<br>正<br>誤 | 正<br>誤 | 制<br>御<br>不<br>可<br>能<br>な<br>効<br>率<br>偏<br>差 | 大<br>気<br>温<br>度 | 海<br>水<br>温<br>度 | 制<br>御<br>可<br>能<br>な<br>効<br>率<br>偏<br>差 | ター<br>ビ<br>ン                    |                            |                                      |                                 |                       | ボイ<br>ラ                         |                                      |                      |                       |                          |                  |  |  |
|          |               |               |               |        |               |               |        |                |        |                     |             |        |                                                |                  |                  |                                           | 主<br>蒸<br>気<br>温<br>度<br>圧<br>力 | 再<br>熱<br>蒸<br>気<br>温<br>度 | 復<br>水<br>器<br>細<br>管<br>清<br>浄<br>度 | 復<br>水<br>器<br>冷<br>却<br>水<br>量 | ス<br>プ<br>レ<br>水<br>量 | 排<br>ガ<br>ス<br>過<br>剰<br>空<br>気 | ス<br>ト<br>ー<br>ブ<br>ロ<br>ウ<br>蒸<br>気 |                      |                       |                          | 補<br>助<br>蒸<br>気 |  |  |
| 単位       | MWH           | MWH           | %             | %      | %             | %             | %      | %              | %      | %                   | %           | %      | %                                              | %                | %                | %                                         | %                               | %                          | %                                    | %                               | %                     | %                               | %                                    | %                    | %                     |                          |                  |  |  |
| 点番号      | WGEN          | WNET          | ERR           | ER     | DER           | EG            | EGC    | EB             | EBC    | ET                  | ETC         | U0     | U1                                             | U2               | CD               | CT1                                       | CT2                             | CT7                        | CT8                                  | CT9                             | CB1                   | CB6                             | CB7                                  | CH                   |                       |                          |                  |  |  |
|          |               |               |               |        |               |               |        |                |        |                     |             |        |                                                |                  |                  |                                           |                                 |                            |                                      |                                 |                       |                                 |                                      |                      |                       |                          |                  |  |  |
|          |               |               |               |        |               |               |        |                |        |                     |             |        |                                                |                  |                  |                                           |                                 |                            |                                      |                                 |                       |                                 |                                      |                      |                       |                          |                  |  |  |
|          |               |               |               |        |               |               |        |                |        |                     |             |        |                                                |                  |                  |                                           |                                 |                            |                                      |                                 |                       |                                 |                                      |                      |                       |                          |                  |  |  |
|          |               |               |               |        |               |               |        |                |        |                     |             |        |                                                |                  |                  |                                           |                                 |                            |                                      |                                 |                       |                                 |                                      |                      |                       |                          |                  |  |  |
|          |               |               |               |        |               |               |        |                |        |                     |             |        |                                                |                  |                  |                                           |                                 |                            |                                      |                                 |                       |                                 |                                      |                      |                       |                          |                  |  |  |
|          |               |               |               |        |               |               |        |                |        |                     |             |        |                                                |                  |                  |                                           |                                 |                            |                                      |                                 |                       |                                 |                                      |                      |                       |                          |                  |  |  |

|   |   |   |   |   |   |   |   |   |
|---|---|---|---|---|---|---|---|---|
| 大 | 技 | 調 | 運 | 管 | 3 | 1 | 2 | 3 |
| 長 | 師 | 査 | 転 | 理 | 直 | 直 | 直 | 直 |
|   |   |   |   |   |   |   |   |   |



### 3.5.2. データ処理への応用

マイクロプロセッサの使用計画もデータ処理装置として使用しようとしている比率が高い。この使い方はミニコンの使い方の延長でミニコンの代りに使用しようとするのである。この場合はもともとミニコンがプログラムロジックでシステムを制御する点で、本質的に同一思想がその際生かされる事は3.3.2.で説明した通りで、容易に計画の具体化が進められるために普及が早いと考えられる。制御に関連した分野でこの種の可能性を拾ってみよう。

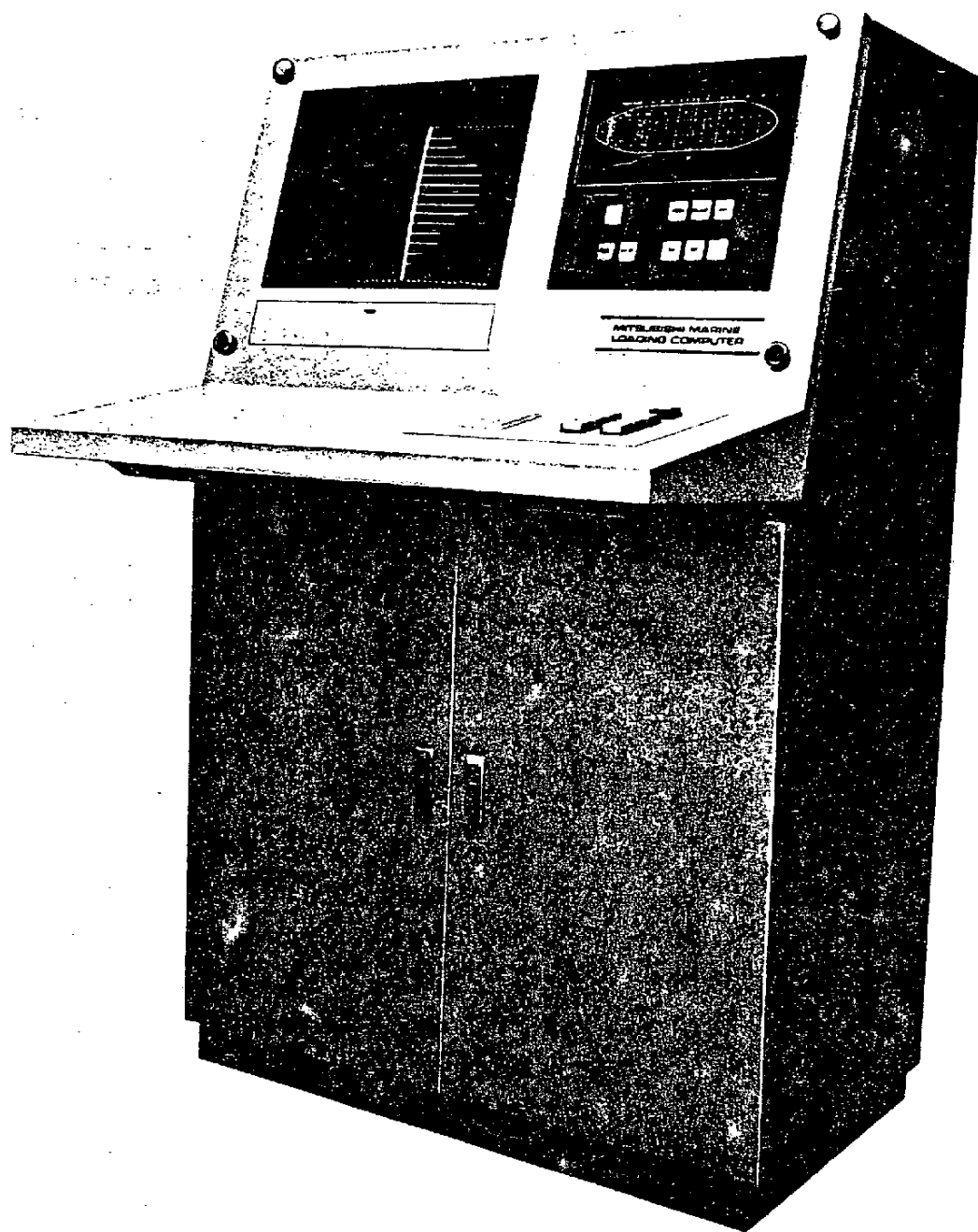
#### (例1) 火力発電所の効率監視

火力発電所の運転員の主要任務のひとつが発電所をできるだけ高い効率で運転し続けることである事は云うまでもない。このために現在自分が運転している発電プラントが、どの程度の効率で運転されているかを知る事が重要であるが、それは実は意外に容易でなかったのである。第3-22図(A)、(B)は、関西電力姫路第二発電所で使用している定時作表データの一部である。この記録項を見ると(A)、(B)に大きな相違がある。即ち(A)は温度、流量、圧力などの計測値であるが、(B)は数多くの計測値をもとに計算して初めてわかる効率そのもの、更にはその効率を生んだ運転条件の、定格運転の時との条件の偏差をデータとして記録している。現在は通常の制御用計算機を設置した事業用の大型発電プラントでのみ実施しているこの種の効率監視法が、マイクロコンピュータを使用すれば手軽に広く実用化できる。このように計測器の直接測定できる量を基に複雑な計算をおこなって初めてわかるパラメータを記録し、運転監視に役立てる事が可能になる例である。

尚、当協会で実施した海外調査団(50-6)の報告で、すでに米国のロスアンゼルス市立ヘインズスチームプラントでマイクロプロセッサを使用して、この種の用途に役立てている事が紹介されている。

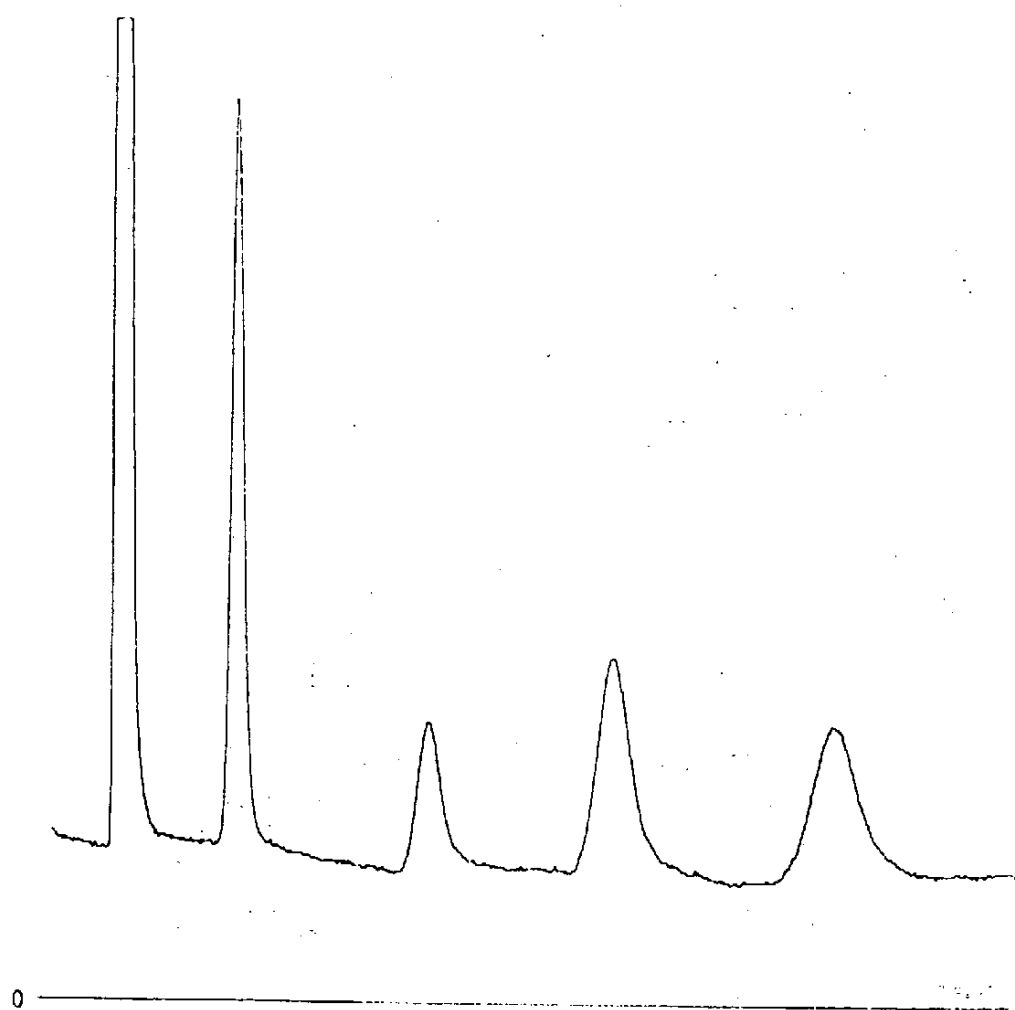
#### (例2) 船用積付計算表示装置

貨物船で荷物を積載するときに、数多く分かれた船槽の何処に載むべきかの判断をするために、塔載荷重総トン数と船槽番号をもとにそ



第 3 - 2 3 図 船用積付計算表示装置





第 3 - 2 4 図 ガスクロ出力データ



の船の船殻の構造から積荷による応力と歪の分布を計算して、歪が安全圏内か否かを表示する例で、第3-23図に示すようなものである。

### 3.5.3. 計測分析システムへの応用

第3-20表によれば、データ処理とならんで計測分析システムへの応用が多いが、この場合も従来から計算機を持続して利用してきた経験のある例が多い。マイクロプロセッサの採用は、そのコンパクト性を利用して従来の個別計器に簡単に組み込む事により、従来大掛りな組合せ構成となっていたものを一層使い易くしようとする考えである。

#### (例3) ガスクロマトグラフデータの処理

気体成分  $H_2$ 、 $O_2$ 、 $N_2$ 、 $CO$ 、 $CO_2$  等々の分析装置として、ガスクロマトグラフは盛んに使用されている。通常の場合は第3-24図のように、分析結果は記録計に順次ピークの形で出力される。このピーク値の面積を計算する事により分析成分を計算するために、従来も、しばしば電算機が利用されてきたので、マイクロプロセッサにより一層広く自動データ処理がおこなわれるようになるであろう。実際には精度を向上させるために、フィルタリング、スムージングなどいろいろなテクニックがプログラムに組まれ実用化されている。

この例は分析成分自動測定の実例であるが、この種の分析成分をもとにした制御が可能な場合、極めて広い応用分野がある。次の例4に、鉄鋼業における転炉製鋼法をサンプルにとり上げた。これも分析成分をもとにした制御の一種である。

#### (例4) 転炉のブレンディングコントロール

第3-25図にこの制御システムの構成を示している。高炉から転炉工場に送られてくる熔銑の重量、温度、成分をもとに目標とする。鋼種の規格に合格する成分の製品を精練するため混合すべきスクラップ、鉾石と石灰の必要量、吹錬すべき酸素量を計算、更に微調整用に混合すべき副原料（加炭剤、フェロマンガ、フェロシリコン、フェロクロム等）の投入必要重量を計算するもので、銑鉄の分析成分が迅速におこない得る事によって実現した。高能率製鋼法へのオンライン制御である。

この種のブレンディングコントロールは、電力ガス製造業における燃料の混合によるカロリー制御、セメント原料のブレンディング制御、各種化学工業における応用など広い応用分野があり、それ等の分析装置の附加装置として、あまり目立たない形ではあるが、意外に数多くマイクロプロセッサが使われてゆくであろうと予想される。

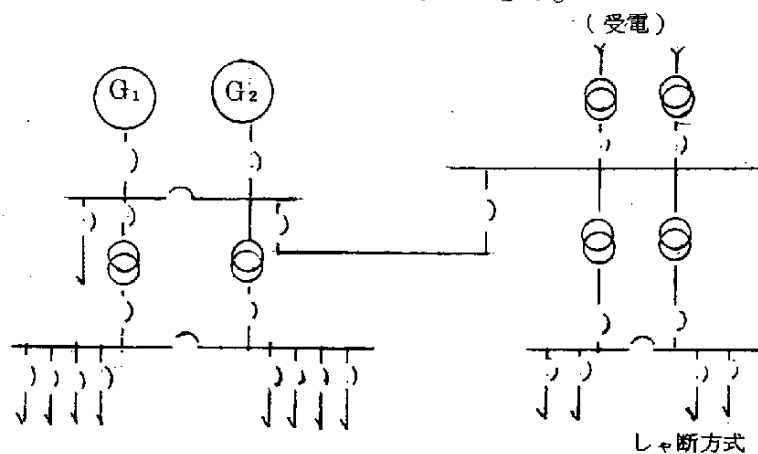
#### 3.5.4. コントロール（プロセス、マシン、システム）への応用の場合

第3-20表ではコントロールをプロセス、マシン、システムに分けて集計しているが、これらを一括集計すると、さすがに広い範囲にわたり、アンケート回答の25%を超えている。

この分野は本応用篇の中心となるもので、西川教授の分類から次章富士電機林部、今泉両氏の具体例に及ぶ詳細システム構成に詳しく説明されているので、此処ではマイクロプロセッサの応用によって普及がはじめて本格化すると考えられる特殊な例を紹介し、応用のポイントを示すにとどめる。

##### （例5） 選択負荷遮断装置

第3-26図に示す自家発電設備を持っている電力系統を例として説明する。この系統に事故が発生した時、瞬時にその事故系統を切り離すと同時にその脱落分に等しい負荷を遮断して、残存供給能力に見合うだけ給電を続け、停電すると大きな事故に波及するような重要負荷の停電を防止しようとするものである。



第3-26図

たとえば、この図の電力会社からの受電が停電になった時、自家発電の供給能力を大巾に上回る負荷が接続されていれば発電機はたちまち過負荷となり、次々にトリップして全停電に至る。そこでこのような事故発生時に、あらかじめ決定しておいた順位に従って負荷を次々と遮断して、残存負荷を自家発電のその時の発電能力に見合う量に制限する。遮断する負荷の決定は次式による。

$$W_{cs} \leq \sum \eta_i W_{fi} - W_d$$

ここに  $W_{cs}$ ; 事故により不足する電力

$W_{fi}$ ;  $i$  番目のフィーダの負荷

$n_i$ ;  $i$  番目のフィーダの状態

(  $\eta_i = 0$  …… 休転中

$= 1$  …… 運転中 )

$W_d$ ; デッドバンド

このような負荷を遮断せねばならない事故も次のようにいろいろのケースがある。

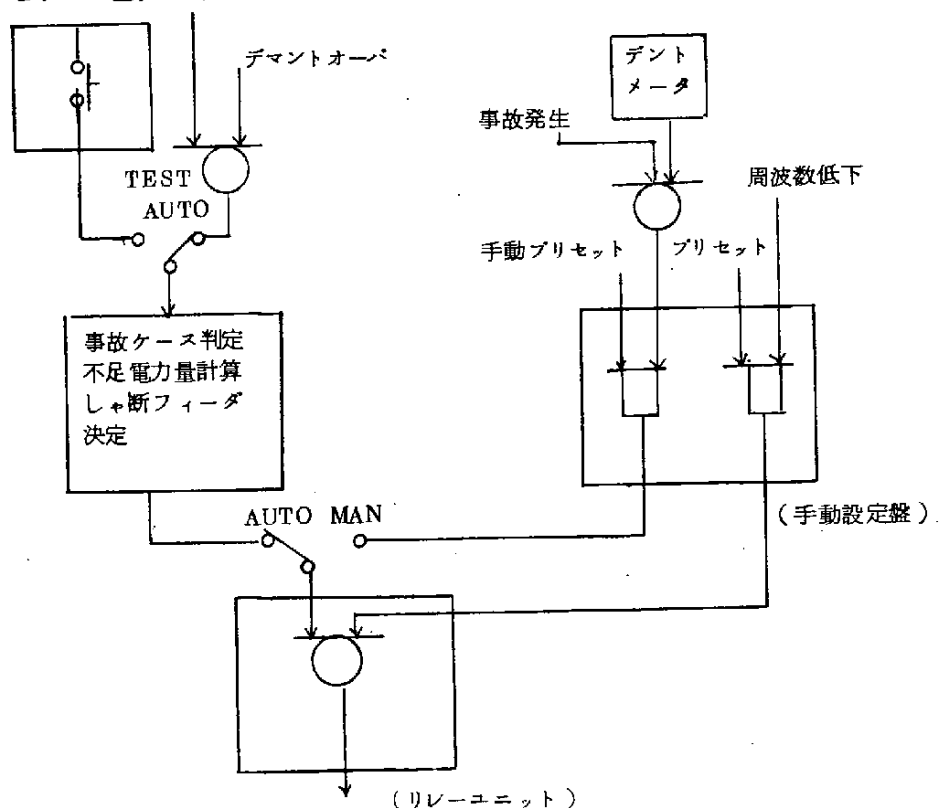
- (イ) 受電事故
- (ロ) 発電機事故
- (ハ) 母線事故、タイケーブル事故
- (ニ) 変圧器事故

また少し性質の異なる負荷遮断として、受電デマンドオーバの監視制御による場合もある。いずれもめったに起こらぬけれども、一度発生すると大きな損害を生じるおそれがあるため、極力この種の装置を設置することが望まれるが、従来なかなか適当なコンパクトな装置になり難いため設けられることの少なかったこの種の装置は、マイクロプロセッサの出現によって急速に普及する可能性がでてきたと云える。第3-27図はそのシステム構成例である。

#### (例6) 移動体に搭載する必要がある制御

上記の例5は、常時制御に使用するというよりも、常時は監視装置であり、事故発生という特殊な時のみ迅速適切な制御をおこなう使い方の例である。もうひとつのマイクロプロセッサの制御への応

(シミュレータ盤) 事故発生



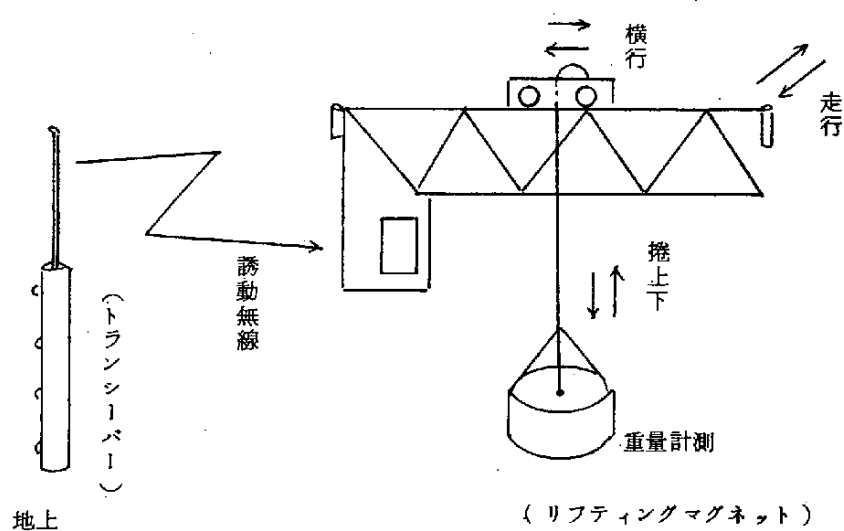
第 3 - 27 図

用として、乗物に載せて使用する分野が大きく開けると思われる。例 2 も船舶に搭載使用する例であるが、自動車のようにより小型の乗物にも使われるようになるであろう。現在 51 年規制対象車の生産が開始される段階であるが、すでに排気ガス対策に合格しながら燃費をあまり増加させない手段として、速度に応じた適切な点火時期の制御、排ガス再循環弁の制御などに従来より一段と複雑な電子回路を使用した制御装置が搭載されはじめており、次の車種では、マイクロプロセッサを使うことになる可能性が出つつある。そしてこの種の用途は、生産数量が多いために特殊な制御分野であるが、

大きな分野になると思われる。

次に乗物としては若干特殊であるが、クレーンに搭載して無人運転クレーン用に使用する計画の例もある。第3-28図はリフティングマグネットを持ったクレーンで、人間は地上からトランシーバで指令を送信、クレーンはその指令を受信、解読するマイクロプロセッサによって運転される例である。

以上一見して、マイクロプロセッサ、ないし、マイクロコンピュータの特長を生かしそうな使い方のポイントをいくつかの例から拾ってみた。これ等をまとめれば、コンパクトに下記のような制御が



第3-28図 無人クレーンの制御

実現できる。ストアドプログラム方式の特長を生かすことにつけると云えよう。

- (1) 3桁以上の精度をとり得ること。
- (2) 単純なP.I.Dだけでないループを任意に組み得ること。
- (3) 上位計算機へのデータ送受信が容易。
- (4) シーケンス制御とPID制御の混在が苦にならないこと。

(5) 著しくタイムラグの長い制御も取扱いやすいこと。

現在のところ第3-20表に表わされているように、ミニコンの小規模システムのかわりの用途と、ランダムロジックのかわりが主たる応用であるけれども、次第に全く新しい使い方が発展してくるであろう。その内容は未経験であるけれども、将来の主流はそちらになるかもしれない。本節の例は、応用の広がりを推察し、使い方がいろいろ考えられるということの糸口を、従来すでに経験している分野からサンプルしたにすぎない。

### 3.6 おわりに

マイクロプロセッサの制御への応用について、その特長とその生かし方当面の問題点をざっと概観した。

進歩の早いこの分野であるから、本稿の主張している事項の寿命はそう長くはないと思われるが、制御の分野に大きな技術革新、即ち、ランダムロジックからストアドプログラムへ、ハードウェアからソフトウェアへ、の波が訪れてきたことを認識して戴いて、その革新にふさわしいエンジニアリング、即ち

- (1) アプリケーション
- (2) ソフトウェア
- (3) ハードウェア

の3つの要素からまとめられる新しい制御のための小さな巨人、マイクロプロセッサに積極的に取り組んで戴きたいと思う。



## 第4章 制御システムの具体例とその展開手法

### 4.1 概 要

マイクロコンピュータの制御システムへの導入については、マイクロプロセッサが出現したときから有望視されていた。しかし、それがなかなか実現しなかった理由として次のことがらをあげることができる。

#### (1) システム全体

- a 実績が乏しいため、信頼性に難がある。
- b マイクロコンピュータ制御システムの背景となっている各応用分野（計算機、計測制御、電力、工業など）の技術、体制の融合が必要である。

#### (2) ハードウェア

- a マイクロコンピュータのメモリで制御システムに適合したものがない。
- b CPU以外の周辺機器、特にプロセスインターフェースのハードウェアの開発が遅れている。
- c 構造があまりコンパクトでない。

#### (3) ソフトウェア

- a ソフトウェアのコスト比率が高くなる。
- b 制御システムはパッケージ化がむずかしい。
- c 計算機の知識が必要になる。

それにもかかわらず、マイクロコンピュータはこれをささえる半導体のテクノロジーとそのコンパクトなイメージによって制御システムの理想型を具体化する有力な武器として、注目されている。第4章ではマイクロコンピュータの適応性を考え、その制御システムの具体例と展開手法を述べる。

マイクロコンピュータ制御システムの具体化およびその展開手法で特に注意しなければならないのはハードウェアとソフトウェアの有効的な使いわけである。以下制御システムの構成、伝送、機能、具体例、展開手法について順次述べる。

## 4.2 制御システムの構成

### 4.2.1. マイクロコンピュータ制御システムの構成

#### (a) マイクロコンピュータの制御システムへの適用

制御システムには二つの相反する条件がある。一つは機能分割による全体システムの危険分散であり、もう一つは情報の集中による管理の質の向上である。従来のミニコンによる計算機制御システムは主に後者の条件を満足したのに対し、マイクロコンピュータは前者の条件を充たすものとして期待されている。マイクロコンピュータはその適用の規模、価格、形状などのコンパクト性により、制御システムの機能を分割し、互いに独立させることにより、一つのサブシステムの故障が全体システムに波及しない、いわゆる危険分散を実現する有力な手段である。したがってマイクロコンピュータは下記のような適用が可能である。

#### (1) 従来システムの置換

マイクロコンピュータの機能により、従来の計算機制御装置または電子制御装置の置換ないしはその機能の向上が可能である。

#### (2) 従来システムとの共存

従来のシステムの部分的置換を行ない、従来の計測制御装置および電子制御装置、さらにプロセス計算機制御装置と共存、融合して用いる。

#### (3) 新分野への適用

制御計算機には規模が小さすぎ、電子制御装置では機能が充分はたせない新しい分野への適用。

また、これらマイクロコンピュータ応用の制御システムの設置方法としては、第4-1図のように

#### A-単独設置

各制御システムをその機能に応じて単独設置し(スタンドアロン)して使用する。

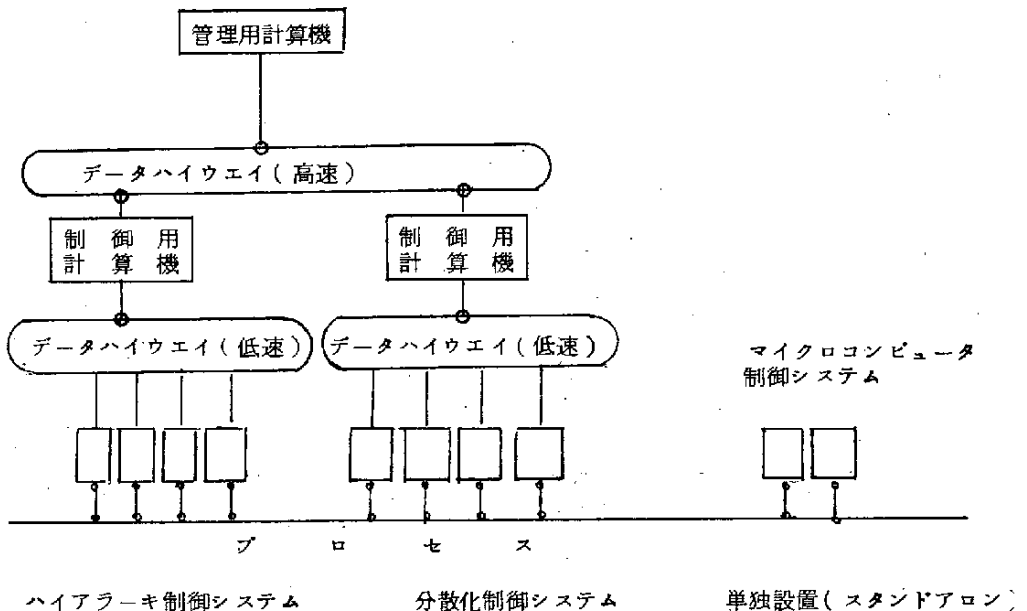
#### B-分散化制御システム

複数の制御システムを各機能に応じて、分散して設置し、有機的に構成することにより、危険分散と機能アップをはかった分散

化制御システムが構成できる。

#### ○ハイアラキシステム

上位計算機システムと結合することによりハイアラキ制御システムを構成することができる。

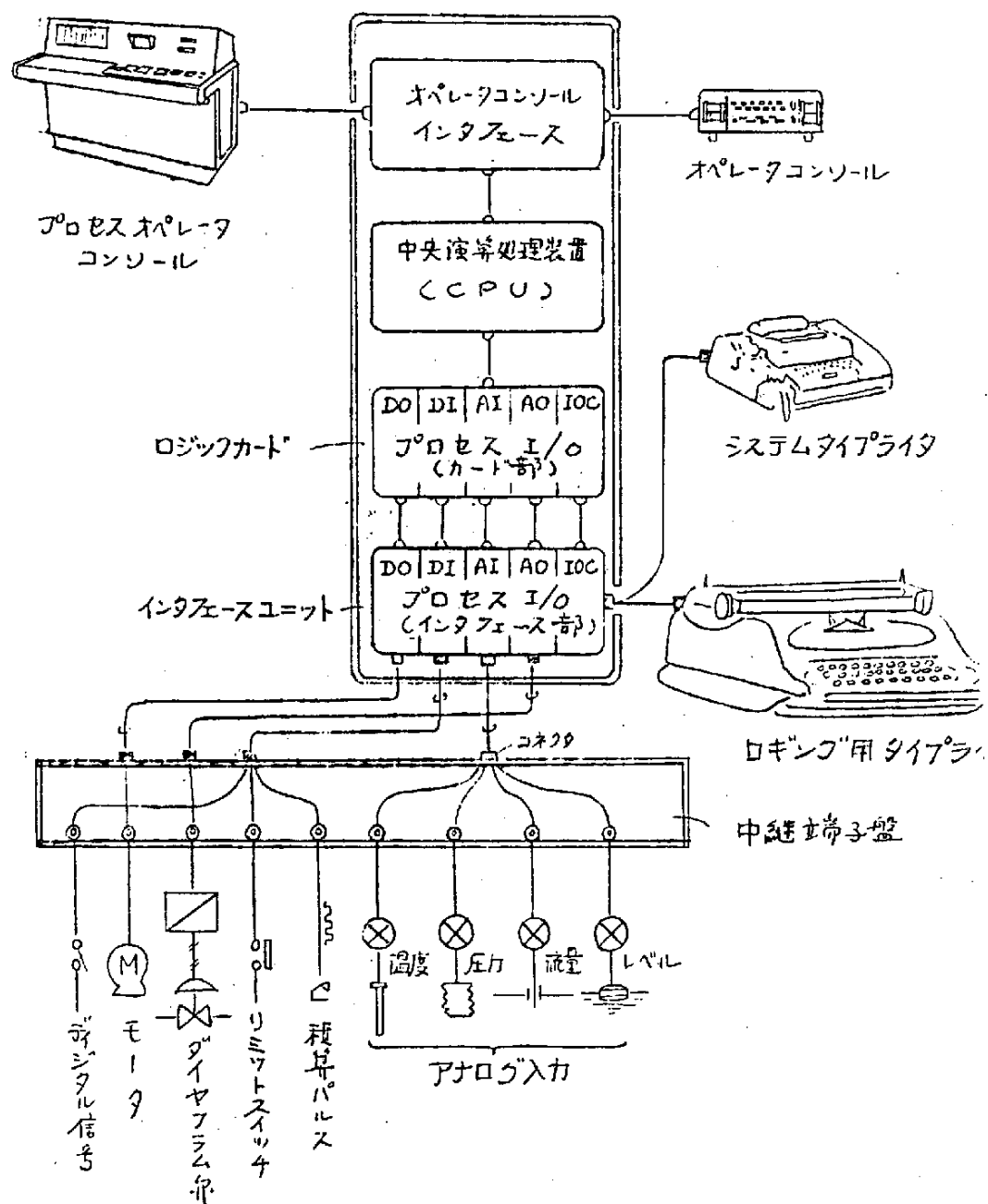


第4-1図 マイクロコンピュータ制御システムと設置形態

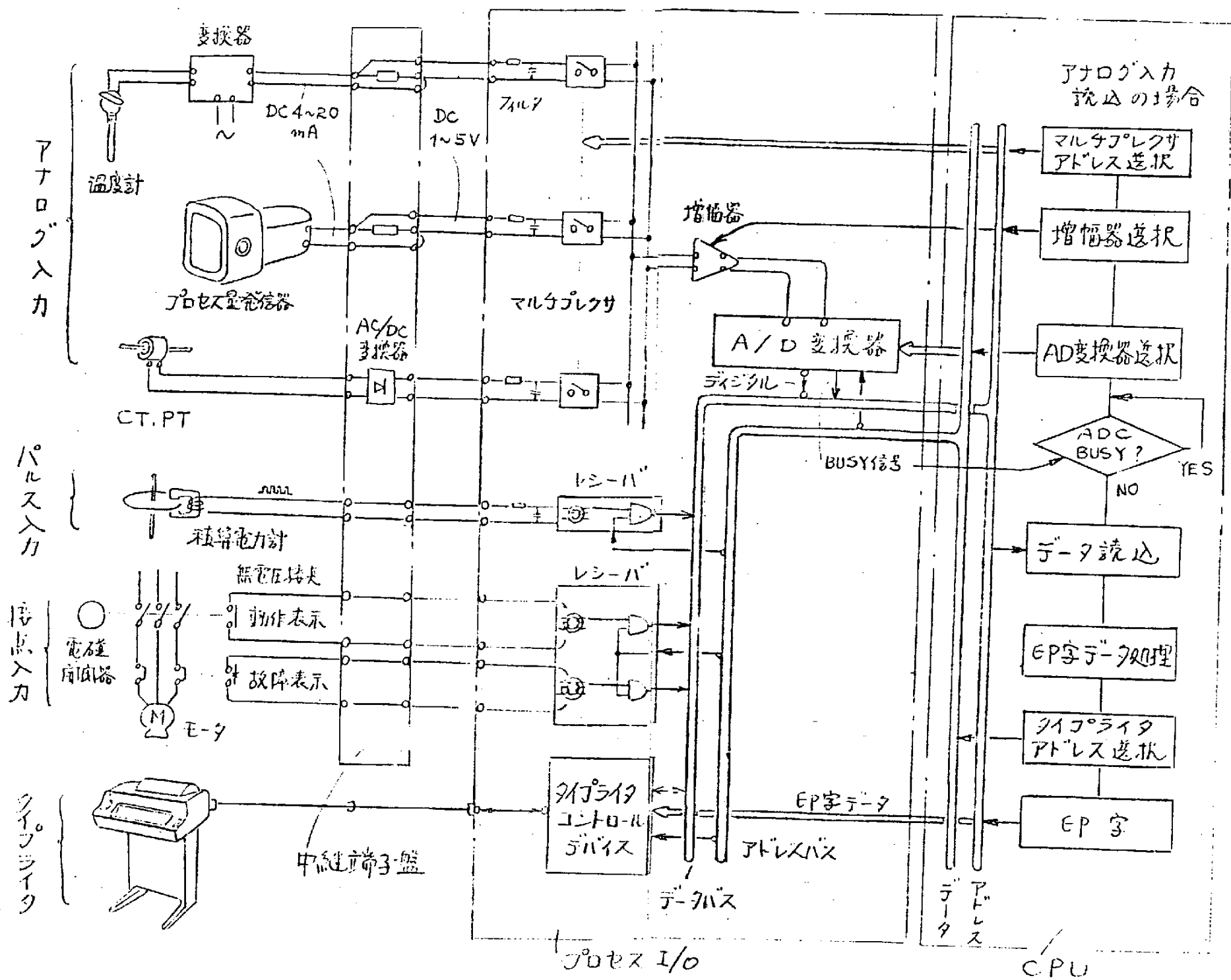
#### (b) マイクロコンピュータ制御システムの構成

計算機制御システムではあらかじめの半固定的にプログラムされた手順に従ってプロセスの状態変化を実時間でキョッチし、これに所定の制御演算処理を施として、制御出力を実時間で出力して制御対象に働きかける。このほか、プラントの操作員とのマンマシンインターフェースの機能を含めて下記の機能が要求される。

- (1) 入力処理
- (2) 制御演算
- (3) 出力処理



第 2 - 4 図 計算機制御システムの構成



第4-3図 データロガにおけるハードウェアとソフトウェアの関連



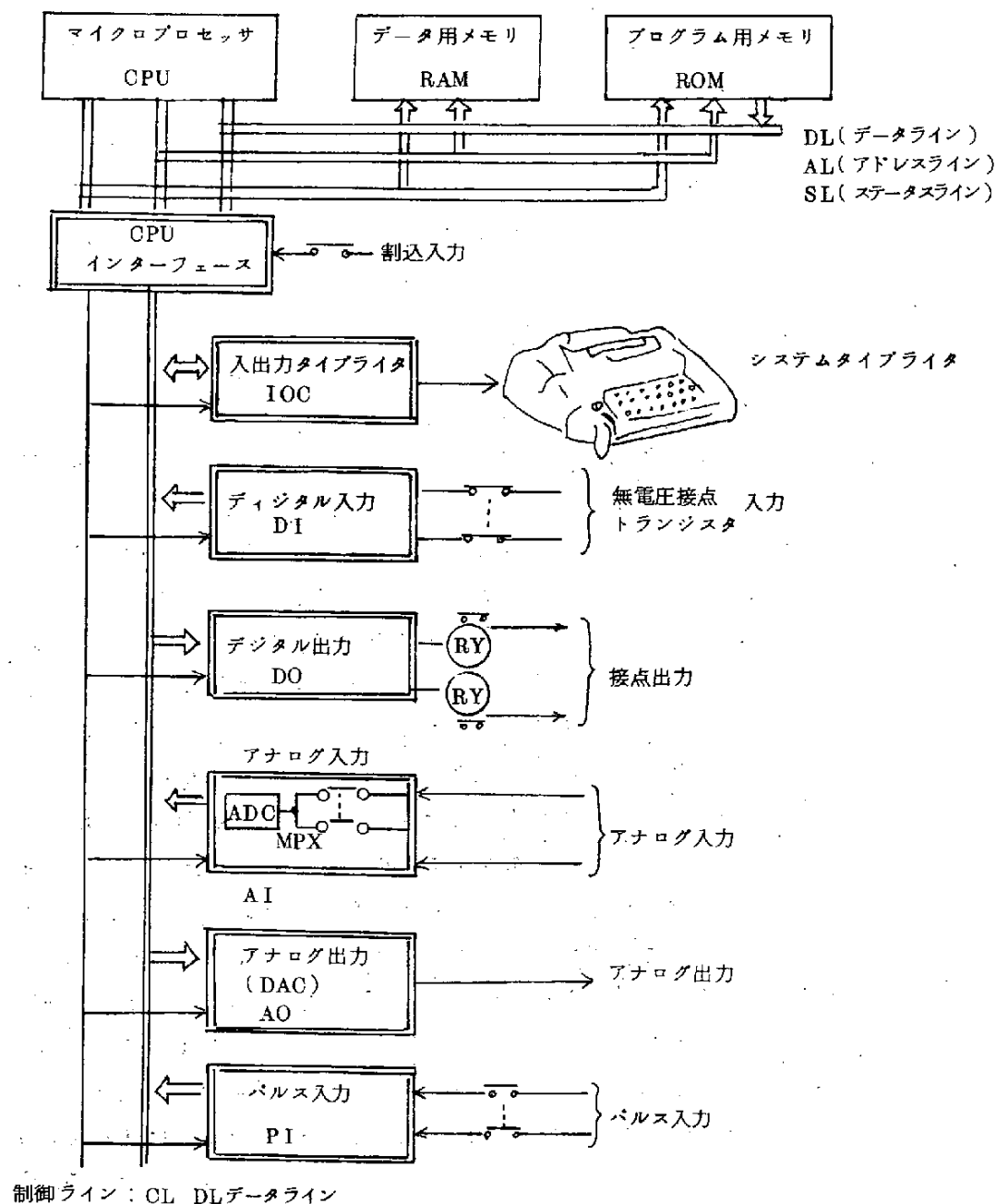
- (4) データ処理
- (5) データ伝送
- (6) マンマシンインターフェース

これらの機能を満たすため、計算機制御システムは一般に第4-2図のようなハードウェア構成を持つ。すなわち、プロセス変数を検出するための発信器やプラントの状態変化を検知するための接点入力などはプロセスインターフェース部(PIO)を経て制御計算機にとり入れられ、所定の制御演算を施した上で、その結果を出力し、PIOを経て、調節弁やモーターなどを駆動する。

このような入出力信号を処理するハードウェアとこれをもとにして制御演算処理を行なうソフトウェアとの関連をデータロガの例で示したのが第4-3図である。図でたとえば温度は温度計、変換器により統一電流信号変えられ、マルチプレクサで選択される。この選択は入力処理プログラムにより切換スイッチを通じて行なわれる。選択されたアナログ信号はA/D変換器でデジタル信号に変換され、データ処理プログラムにより、所定の印字フォーマットに従って印字される。印字は入力処理とは別に出力処理プログラムによりPIO部のタイプライタコントロール装置を介してタイプライターを駆動して行なう。

#### (C) マイクロコンピュータ制御システムのハードウェア構成

第4-4図はマイクロコンピュータ制御システムのハードウェア構成の例である。この例ではマイクロプロセッサとROM、RAMのメモリ部とからなるCPU部と、デジタル入出力、アナログ入出力などのモジュールからなるプロセスインターフェース(PIO)部との間にCPU/PIOインターフェースをおき、CPU-メモリ部のバス構造とCPU/メモリ-PIOとのバス構造を異にしている。すなわち、コモンバス方式は採用していない。これはマイクロコンピュータのような比較的規模の小さいものには好都合なアーキテクチャで、結果的にハードウェア、ソフトウェアの両面にわたり、いわばマイクロな構造とすることができる。



第 4 - 4 図 マイクロコンピュータ制御システムの構成例



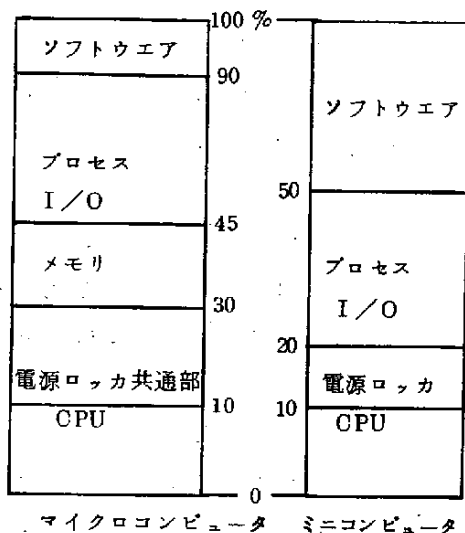
(d) マイクロコンピュータ制御システムのソフトウェア構成

マイクロコンピュータを制御システムに導入する場合は、マイクロコンピュータの特長であるコンパクト性を生かして、従来の制御計算機が行なっていた機能を分割し、単機能システムを作り上げることが要求される。これに応じて、ソフトウェア構造もマルチジョブからシングルジョブへ移行するため、ミニコンの構成とはかなり異なったものになる。一方、マイクロコンピュータ制御システムのコスト構成は、

第4-5図のごとく、従来のミニコン制御システムと比較すると、かなり異なっている。すなわち、マイクロコンピュータがコンパクトで安価であるというイメージからすれば、マイクロコンピュータ制御システムでは、そのソフトウェアの全体システムに対するコスト比率を10%~20%程度に抑えることが要求される。このためには制御システムをいくつかの汎用システムに機能分割し、システムをハードウェア、ソフトウェア一括パ

ッケージ化することが必要である。計算機システムのソフトウェアでかなりの部分を占めるOS(OPERATING SYSTEM)などの共通部を単機能化すれば、メモリ容量を小さくすることができる。またマクロ言語、問題向言語(POL)、テーブル方式プログラミング(FIF)などのアプリケーションエンジニア向きのプログラミング方式を採用することにより、プログラミングに要するソフトウェア費を軽減することができる。

マイクロコンピュータの場合はユーティリティ、アセンブラ、シミ



第4-5図 マイクロコンピュータ制御システムのコスト構成

ュレータなどのソフトウェアサポートは上位計算機で行なうのが普通であり、マイクロコンピュータの内部には常時おかない。

以上を総合すると、マイクロコンピュータ制御システムのソフトウェア構成は第4-6図の例のようになる。また全体のメモリ容量は4 K語から8 K語程度である。

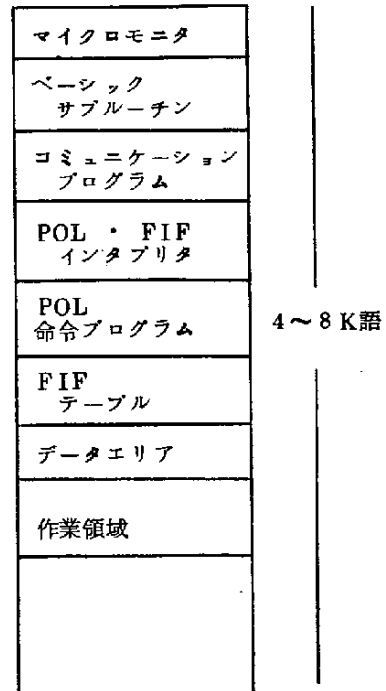
#### 4.2.2. 制御システムの具体化

##### (a) 制御システムの形態

制御システムを具体化する際その応用製品としての形態はソフトウェアによるもの、ハードウェアによるもの、種々考えられる。第4-7図はその典型的なもので、この中で一品料理的なソフトウェアの形態をマイクロコンピュータで具体化するのはソフトウェアのコストがかさむので、不適當であると思われる。これらソフト、ハードの形態はどのような組合せも可能であり、たとえば専用型ソフト+単目的型ハード、翻訳型ソフト+単目的型ソフト、FIF型ソフト+多目的型ハードなどの製品が考えられる。

##### (b) コンポーネントの選定

マイクロコンピュータはMOSやバイポーラLSIなど新しいテクノロジーをその背景としているため、プロセッサだけでなく、メモリやAD変換器、伝送制御デバイスなどのファンクションデバイスを積極的に導入することが、パフォーマンスのよい制御システムを作る上で重要なことである。第4-1表はマイクロコンピュータ制御システムのコンポーネント一覧表である。

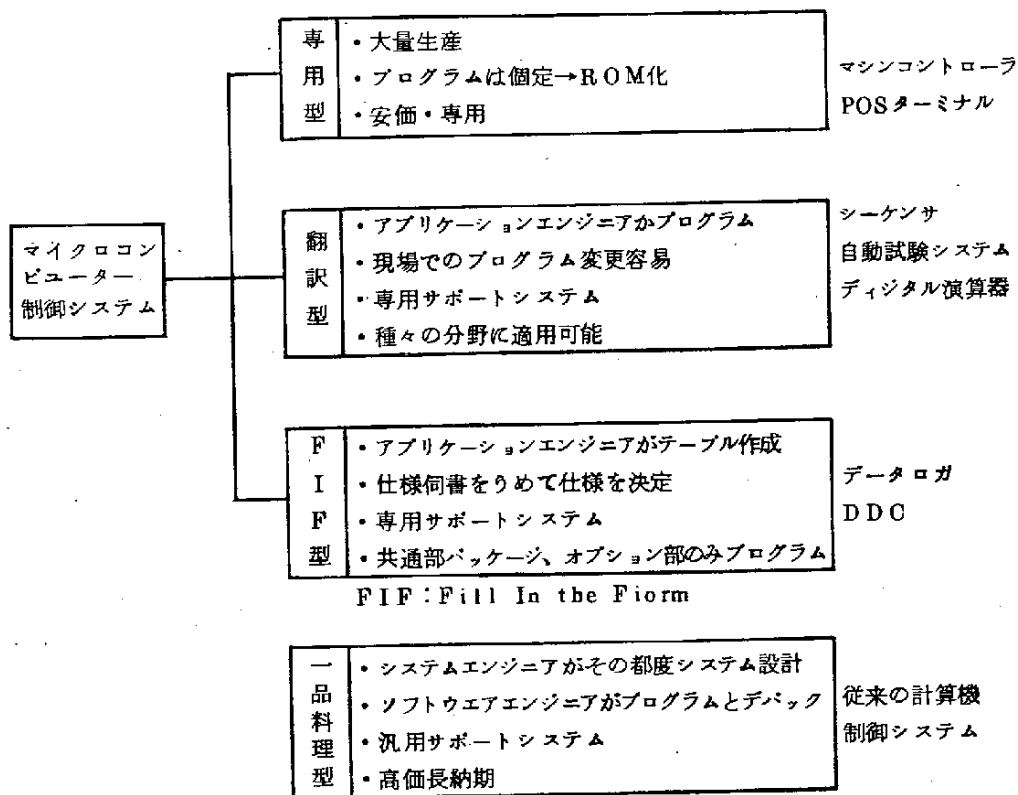


第4-6図

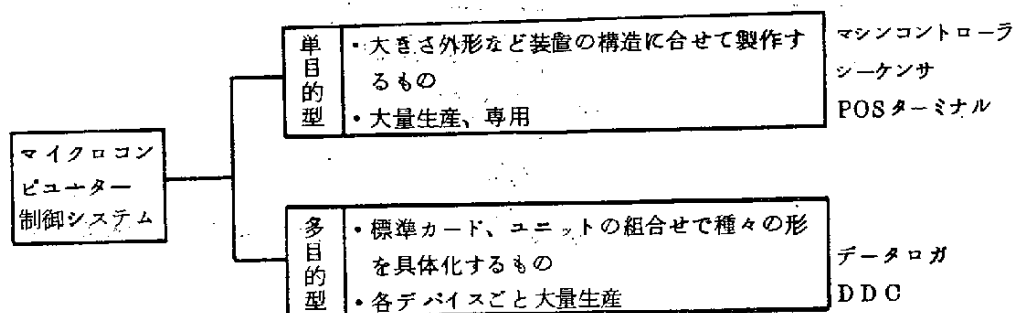
マイクロコンピュータ制御システムのソフトウェア構成

ソフトウェアからみたマイクロコンピュータ制御システム

例



ハードウェアからみたマイクロコンピュータ制御システム



第4-7図 マイクロコンピュータ制御システムの諸形態



| 分類         | MOS チップ名称                                                             | 備 考                                      |
|------------|-----------------------------------------------------------------------|------------------------------------------|
| CPU        | マイクロプロセッサ<br>ラッチ・インターフェース<br>クロック                                     | PMOS、NMOS、CMOS、バイポーラ<br>プロセス周辺インターフェース   |
| メモリ        | スタティック RAM<br>ダイナミック RAM<br>マスク ROM<br>フィールドプログラマブル ROM<br>イレーサブル ROM | 文字発生器、IPL<br>PROM、ヒューズROM<br>EROM(紫外線消去) |
| 伝 送<br>結 合 | 直列伝送インターフェース<br>DMA                                                   | MODEM                                    |
| I/O        | ディジプリンタインターフェース<br>TTYインターフェース<br>タイプライタインターフェース                      |                                          |
| P I O      | ADコンバータ<br>DAコンバータ<br>マルチプレクサ・データセレクト<br>プログラマブルタイマ                   | 逐次比較型、V/F交換型                             |

第4-1表 マイクロコンピュータ制御システムのコンポーネント(LSI)

(c) マイクロプロセッサ

マイクロプロセッサは1971年INTEL社の4ビット1チップCPUが登場してから第4-8図のようにいくつかの製品が開発発売された。その傾向はテクノロジーとしてはPMOS→NMOS→CMOS→バイポーラであり、語長としては4→8→12→16ビットである。第4-9図は16ビットの1チップCPUである

(d) メモリ

マイクロコンピュータのメモリとして、ROMやRAMについてはすでに述べたが、ここでは制御システムとして、機能別にどのような性質のメモリが必要か述べる。第4-2表はマイクロコンピュータのメモリを私見をまじえて用途別に分類したもので、制御システムでは当然のことながら不揮発性のメモリが要求され、磁気コアがまだ重要な存在であることが結論づけられる。

(e) PIO

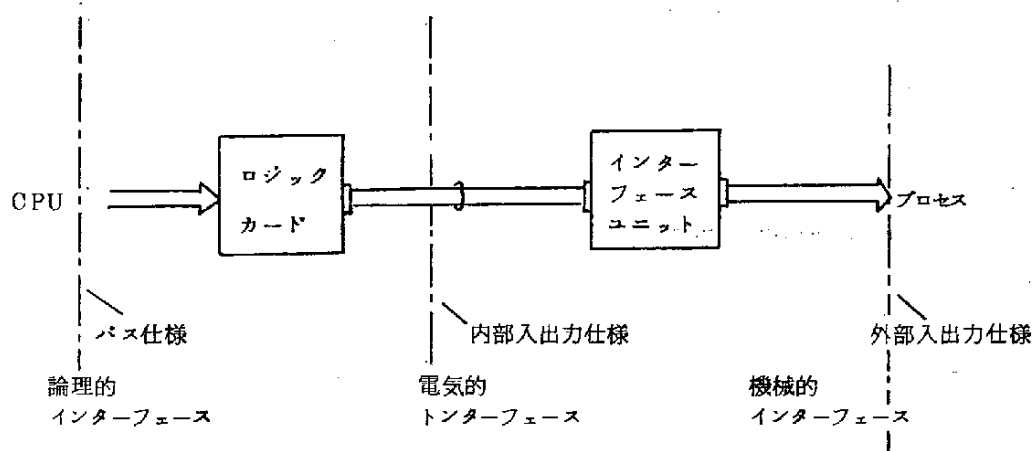
マイクロコンピュータ制御システムとして重要なプロセスインターフェース(PIO)は、プロセスの入出力信号をプロセッサへのICレベルの論理信号に変換するため、機械的インターフェース、電気的インターフェース、論理的インターフェースから構成される。ここでいう機

| 分類    | 機能                 | 具体例                            | メモリ |     |      |                    |
|-------|--------------------|--------------------------------|-----|-----|------|--------------------|
|       |                    |                                | RAM | ROM | PROM | CORE<br>又は<br>WIRE |
| プログラム | ベシックスポートプログラム      | IPL                            |     | ○   |      |                    |
|       | ベシックスソフトウェア        | モニタ サブルーチン                     |     | △   | ○    |                    |
|       | POL FIF実行プログラム     | インタプリタ テーブル                    |     | △   | ○    |                    |
|       | ファンクションプログラム       | PIDアルゴリズム                      |     |     | ○    |                    |
|       | アプリケーションプログラム      |                                |     |     | ○    |                    |
| 定数    | FIF用テーブル定数         | 工業用変換データ                       |     | △   | ○    |                    |
|       | アプリケーション定数<br>制御定数 | 入出力処理データ<br>リニアライザ定数<br>DDC設定値 |     | △   | ○    | ○                  |
| データ   | 一般データ              |                                | ○   |     |      |                    |
|       | 制御入力(瞬時)           | 積算パルス                          |     |     |      | ○                  |
|       | 制御入力(保持)<br>制御出力   | インタラプト入力                       | ○   |     |      | ○                  |

○: 適当 △: 場合に応じて適用

第4-2表 機能別メモリ適用一覧表

機械的インターフェースとは外部信号をねじ端子またはコネクタでとり入れ、絶縁を行なって標準のアナログまたはデジタルの電気信号に変換する部分をいい、電気的インターフェースはこれら電気信号をI/Oレベルの論理信号に変換する部分である。さらに論理的なインターフェースにより、これら個々の論理信号をコンピュータの手順に適合した形におき換える。これら電気信号および論理信号は第4-3表～第4-5表のごとく、内部入出力および外部入出力一覧表としてまとめることができる。(第4-10図参照)

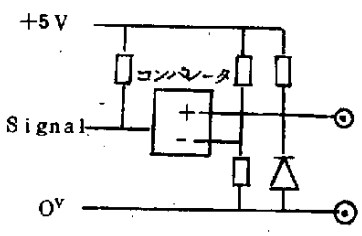
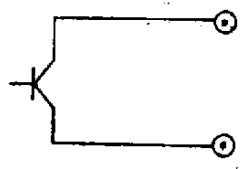


第4-10図 CPU-プロセスインターフェース

以上CPU、メモリ、PIOについて、ロジックカード単位に標準的なものをまとめると第4-6表および第4-7表のようになる。

#### (f) I/O機器

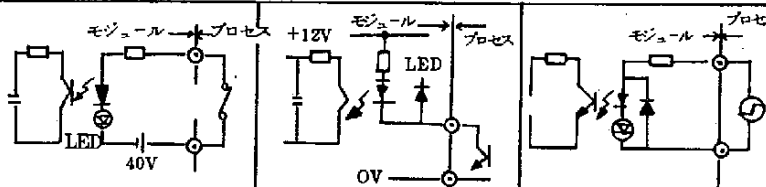
マイクロコンピュータ制御システムのI/O機器はプログラミング用と制御システム用に大別される。プログラミングといってもここでは上位計算機によるサポートシステムではなく、プログラムローダのような専用プログラム作成装置を指す。プログラム作成装置に要求される機能は次のようなもので、これを実現するためには、プログラム

|         | 入 力                                                                                         | 出 力                                                                                |
|---------|---------------------------------------------------------------------------------------------|------------------------------------------------------------------------------------|
| 信号レベル   | 論理 "1" = <span style="border: 1px solid black; padding: 0 2px;">+5</span> V<br>論理 "0" = 0 V | 論理 "1" = Tr. ON<br>論理 "0" = Tr. OFF                                                |
| 基 本 回 路 |            |  |
| 耐 圧     | D C ± 5 V                                                                                   | B C + 5 V                                                                          |

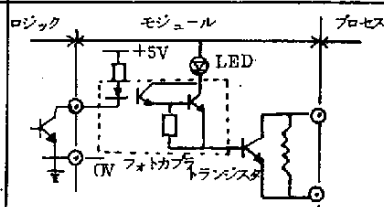
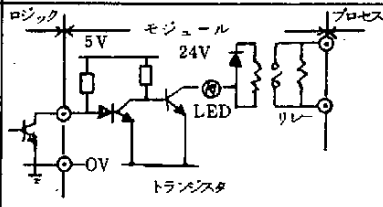
第 4 - 3 表 内部入出力仕様一覧表



デジタル入力

| 入力信号               | 電 源 内 蔵 形                                                                          |                               | 電 源 外 附 形      |
|--------------------|------------------------------------------------------------------------------------|-------------------------------|----------------|
|                    | 接 点                                                                                | TTL、T <sub>r</sub> (オープンコレクタ) | 電 圧            |
| 印 加 電 圧            | DC 48 V (モジュール内蔵)                                                                  | DC 12 V (モジュール内蔵)             | DC 12 V (外部印加) |
| 入 力 電 流            | DC 10 mA                                                                           |                               |                |
| ロジックカードとのインタフェース電圧 | DC 5 V                                                                             |                               |                |
| 回 路                |  |                               |                |

デジタル出力

|           | トランジスタ出力                                                                           | 接点出力                                                                                |
|-----------|------------------------------------------------------------------------------------|-------------------------------------------------------------------------------------|
| 絶 縁 方 式   | フォトカプラ                                                                             | リレー                                                                                 |
| 絶 縁 附 圧   | AC 500 V 1分間                                                                       | AC 2000 V 1分間                                                                       |
| 印 加 電 圧   | DC 5~24 V                                                                          | AC 250 V                                                                            |
| 許 容 電 流   | DC 150 mA 最大                                                                       | AC 7.5 A 最大                                                                         |
| 基 本 路     |  |  |
| スイッチング周波数 | 1,000 Hz                                                                           | 10 Hz                                                                               |
| 耐 圧       | 一般仕様 AC 500 V 1分間<br>電力仕様 AC 200 V 1分間                                             |                                                                                     |

第4-4表 外部入出力仕様一覧表(デジタル入出力)

|         |                        | 低 速 形                           | 中 速 形                                                                     | 高 速 形     |
|---------|------------------------|---------------------------------|---------------------------------------------------------------------------|-----------|
| 絶 縁 方 式 |                        | 有                               | 有                                                                         | 無         |
| A/O変換方式 |                        | 積 分 形                           | 逐 次 比 較 形                                                                 | 逐 次 比 較 形 |
| 変 換 時 間 |                        | 3 2 点/秒                         | 4000 点/秒                                                                  |           |
| 変 換 積 度 |                        | 10bit、 $\pm 1/2$ LSB            | 8+11bit、 $\pm 1/2$ LSB                                                    |           |
| マルチプレクサ |                        | リードリレー(両切り)                     | 半導体スイッチ                                                                   |           |
| 入<br>力  | 入力電圧                   | +125、25、50、100MV<br>+1、5、10V    | $\pm 125$ 、 $\pm 25$ 、 $\pm 50$ 、 $\pm 100$ MV<br>$\pm 1$ 、15、 $\pm 10$ V |           |
|         | 絶縁耐圧<br>(対地、各<br>入力点間) | DC 500V<br>AC 300V              | DC 500V<br>AC 300V                                                        | ナ シ       |
|         | 同相入力電圧<br>除 去 比        | 100dB 以上<br>(商用周波数に対し)          | 126dB 以上<br>80dB 以上(10KHZ)                                                |           |
| 出 力 形 式 |                        | 符号+10bit(純2進)<br>定格入力で1000(10進) | 符号+11bit(純2進)<br>負数は2の補数<br>定格入力で2000(10進)                                |           |

第4-5表 外部入出力仕様一覧表(アナログ入力)

| 大項目 | 小項目             | 仕様                                       |
|-----|-----------------|------------------------------------------|
| CPU | CPUカード(4ビット)    | I 4040(インテル)など                           |
|     | " (8ビット)        | I 8080(インテル)など                           |
|     | " (16ビット)       | MN 1610(パナファコム)など                        |
| メモリ | IC・PROMカード      | 4 KByte/カード                              |
|     | IC RAMカード       | 4 KByte/カード                              |
|     | IC PROM/RAMカード  | PROM 2 KByte<br>RAM 2 KByte/カード          |
|     | 磁気コア又はワイヤメモリカード | 4 KByte/カード                              |
| 結合  | プログラムIO結合カード    | マイクロコン 他のシステム                            |
|     | DMA結合カード        | "                                        |
| PIO | PIOインタフェースカード   | CPU $\longleftrightarrow$ PIOインタフェース     |
|     | インタラプト入力カード     | レベル1 ..... 8点<br>レベル2 ..... 8点           |
|     | アナログ入力カード       | 16点                                      |
|     | デジタル入力カード       | 64点                                      |
|     | パルス入力カード        | 8ビット×4+1ビット×32                           |
|     | アナログ出力カード       | 2点                                       |
|     | デジタル出力カード       | 64点                                      |
|     | パルス出力カード        | 32点                                      |
|     | 直列コード伝送カード      | 110、1200、1400、4800、9600 bit/sec<br>送受信一組 |
|     | 並列コード伝送カード      |                                          |

第4-6表 ロジックカード一覧表

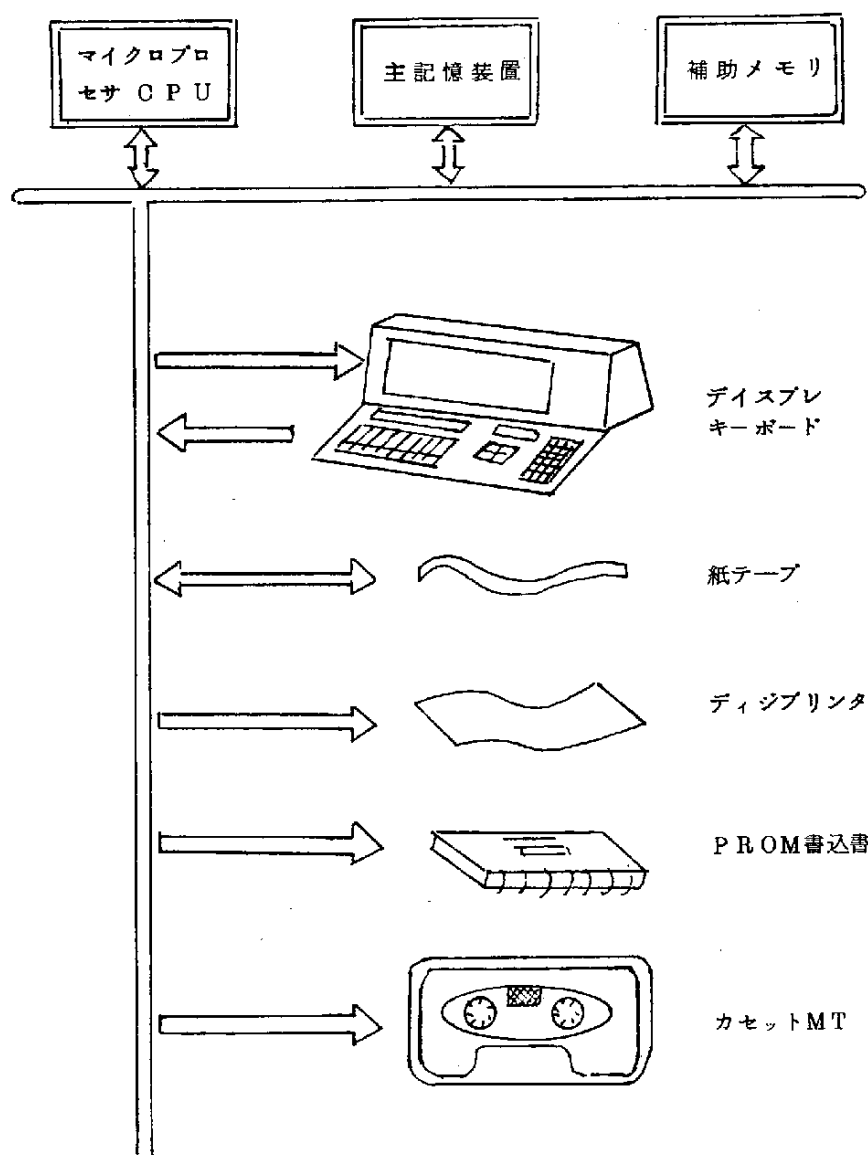
| ロジックカード                               | モジュールユニット           |                                |                      |
|---------------------------------------|---------------------|--------------------------------|----------------------|
|                                       | 種 類                 | 仕 様                            |                      |
| アナログ入力カード                             | 低 速 形               | マルチプレクサ：リードリレー 16点<br>変換方式：V/F |                      |
|                                       | 中 速 形               | マルチプレクサ：FET 32点<br>変換方式：逐次     |                      |
| デジタル入力カード<br>パルス入力カード<br>割込入力         | 接 点<br>入 力 用        | 一般                             | 32pt、フォトカブラ絶縁、LED表示付 |
|                                       |                     | 電力                             | 同 上                  |
|                                       | オープン<br>コレクタ<br>入力用 | 一般                             | 32pt、フォトカブラ絶縁、LED表示付 |
|                                       |                     | 電力                             | 同 上                  |
|                                       | 電 圧<br>入 力 用        | 一般                             | 32pt、フォトカブラ絶縁、LED表示付 |
|                                       |                     | 電力                             | 同 上                  |
| デジタル出力カード<br>パルス出力カード<br>ビットシリアル電送カード | オープン<br>コレクタ<br>出力用 | 一般                             | 32pt、フォトカブラ絶縁、LED表示付 |
|                                       | 接点<br>出力用           | 電力                             | 32pt、リレー絶縁、LED表示付    |
|                                       | 電 圧 出 力 用           |                                | MODEM インターフェース       |
|                                       | 電 流 出 力 用           |                                | ASB33 インターフェース       |

第4-7表 インターフェースユニット一覧表

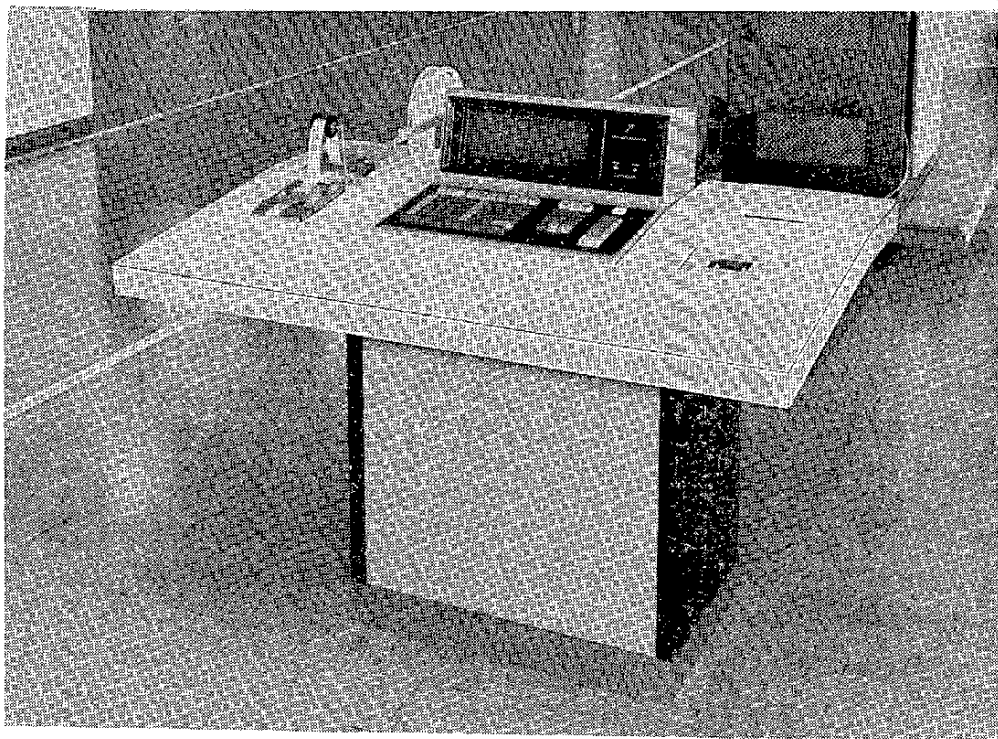
作成装置自体がマイクロプロセッサを搭載した第4-11図のようなハードウェアが必要である。

- (1) IPL (イニシャル、プログラム、ローディング)
- (2) アセンブル
- (3) PROM書込み
- (4) POL (問題向言語) の翻訳
- (5) FIF (テーブル方式プログラム) の変換
- (6) 会話型プログラミング処理
- (7) 紙テープのリード、パンチ
- (8) プログラムなどの印字
- (9) カセットMTによるプログラムの保存
- (10) ディスプレイ

第4-12図はプログラム作成装置の一例である。



第 4 - 1 1 図 プログラム作成装置のブロックダイアグラム



第 4 - 1 2 図 プログラム作成装置の例

### 4.3. 制御システムにおけるデータ伝送

#### 4.3.1. 伝送の形態

##### (a) 概要

データ伝送の手段として、計測制御の分野で用いられてきた方式には第4-8表の種類がある。

| 種類   | 変調 | 信号形態                        | 例                              | 備考 |
|------|----|-----------------------------|--------------------------------|----|
| アナログ | 直送 | 0.15~3mADC 3KΩ以下            | アナログテレメータ<br>2芯 1量             |    |
|      | 搬送 | 12~24 HZ FS変調<br>(トーンチャンネル) | アナログ式搬送<br>テレメータ<br>(通常1回線18量) |    |
|      | 直送 | 並列コード                       | IBM 735 タイプライタ                 |    |
| デジタル | "  | 直列コード                       | テレタイプ<br>リレー式テレコン              |    |
|      | 搬送 | 直列コード                       | デジタル式テレメータ<br>テレコントロール         |    |

第4-8表 従来の各種データ伝送方式

マイクロコンピュータを利用した制御システムでは階層構造の制御システムを組むため、あるいは、制御の分散化を行なうため、データ伝送が必須のもととなる。

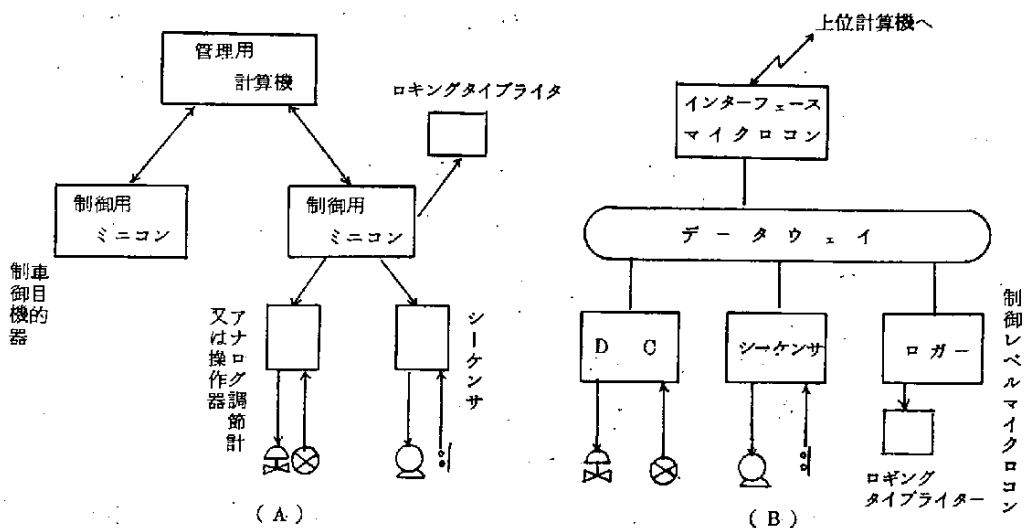
##### (b) 階層構造とシステム間伝送

制御システムでは、その機能を分析してみると単に一つのプラントの一部分の自動制御から工場全体の運転まで、その関連する分野が広い。そのため階層細造の思想がとり入れられてきた。

第4-13(A)図に従来の階層構造システムの例を又、第4-13(B)図にマイクロコンピュータが導入された場合の階層構造の例を示す。

第4-13(A)図ではアナログ調節計とシーケンス制御機器との情報の交換は直接行なわれず、上位の制御用ミニコンピュータが、それ





第 4 - 13 図 ハイアラキシステムの例

それとの情報交換の中継を行なうと同時に、それぞれの機器の管理を行なう方式である。上下方向のデータ伝送が主である。単目的制御機器と制御用ミニコンピュータ間に取り扱われるデータはほとんど加工されていないものが多いシステムである。

第 4 - 13 (B) 図では、制御機器をマイクロコンピュータで置換した場合であり、簡単な情報の処理はそれぞれの制御機器で行ない、加工されたデータが同位の制御機器へ伝送され制御が行なわれ、管理に必要なデータのみが上位へ伝送されるシステムである。同位のシステム間のデータ伝送が考慮されたシステムである。

マイクロコンピュータを制御機器へ導入する際は、伝送機能を持たせることが必要不可欠な条件となる。個別の制御機器、例えばシーケンサ、DDC システム等が伝送機能を持つことにより、制御機器は独立して使用されると同時に、制御システム全体としては有機的に連携して運営させることが出来るようになる。

すなわち、制御機器は分散化し、情報は集中化することが可能になり、危険の分散化とマンマシンインターフェースの集中化が達成できるようになる。

### (3) 伝送形式（制座システムとしての特色）

制御装置は、フィードバック制御でもシーケンス制御でも、その基本的機能は

- ・ 検出端より情報を受けとる
- ・ PID演算、論理演算を行なう。
- ・ 操作端へ出力信号を送る。

等3つの機能より構成されている。

すなわち検出端よりの情報と操作端への情報は密接な相関がある。演算処理部が1ヶ所に集中化されている場合は、検出端、操作端をそれぞれ子局とした1:Nのデータ伝送形式が有利となり、一回のデータ伝送で入力と出力を同時に処理できる伝送フォーマットがより良いものとなろう。しかし演算処理、入出力処理が一体化されている制御機器の組み合わせで制御システムが構成されているような場合には、制御機器間のデータ伝送と情報の集中化のためのデータ伝送が必要であり、N:Nのデータ伝送形式が有利となる。

### (4) 各種伝送システムの形態の分類

制御システム全体を経済的に又有効に動作させるためにはその制御の用途に応じて各種のデータ伝送方式が採用される。

- ・ 1:1 データ伝送
- ・ 1:N       "
- ・ N:N       "

#### (イ) 1:1 データ伝送

データの送受信が2ヶ所に局在化している場合に利用される形式、従来のテレメータテレコン等が採用されることが多い。

#### (ロ) 1:N データ伝送

N個の子局と1個の親局とで構成されるもので、分散化した機器の上下方向に結合が密な場合に採用される。前項の演算部が集

中化されているような場合、又は分散化したそれぞれの制御機器の仕事が独立しており、機器間のデータ伝送が余り必要のないシステムに向いている。

#### (イ) N : Nデータ伝送

それぞれのデータ送受信端が対等の位にあり、どの機器間のデータも自由に直接送受信できるデータ伝送システムである。(第4-13図参照)制御機器間が相互に密な結合を必要とする場合、又は制御機器間のデータ伝送量が平均化している場合に採用される伝送形式である。

尚、N : Nのデータ伝送が採用されるシステムを大別すると、制御機器間のデータ伝送、例えばシーケンサとDDC装置間等と、工場間の管理用計算機レベルのデータ伝送とに分けられる。前者ではインターロック信号等、情報量も余り多くはないが、後者では管理データ、スケジューリングデータ等情報量も多くなる。

### 4.3.2. ハイアラキーシステム

#### (a) 概要

従来の制御機器をマイクロコンピュータで置換するシステムでは

- ・上位計算機システムとの結合方法
- ・ジョブの分担
- ・他システムによるサポート

等を考慮して、より有効なハイアラキーシステム(階層構造システム)を構成せねばならぬ。

ハイアラキーシステムの基本思想は

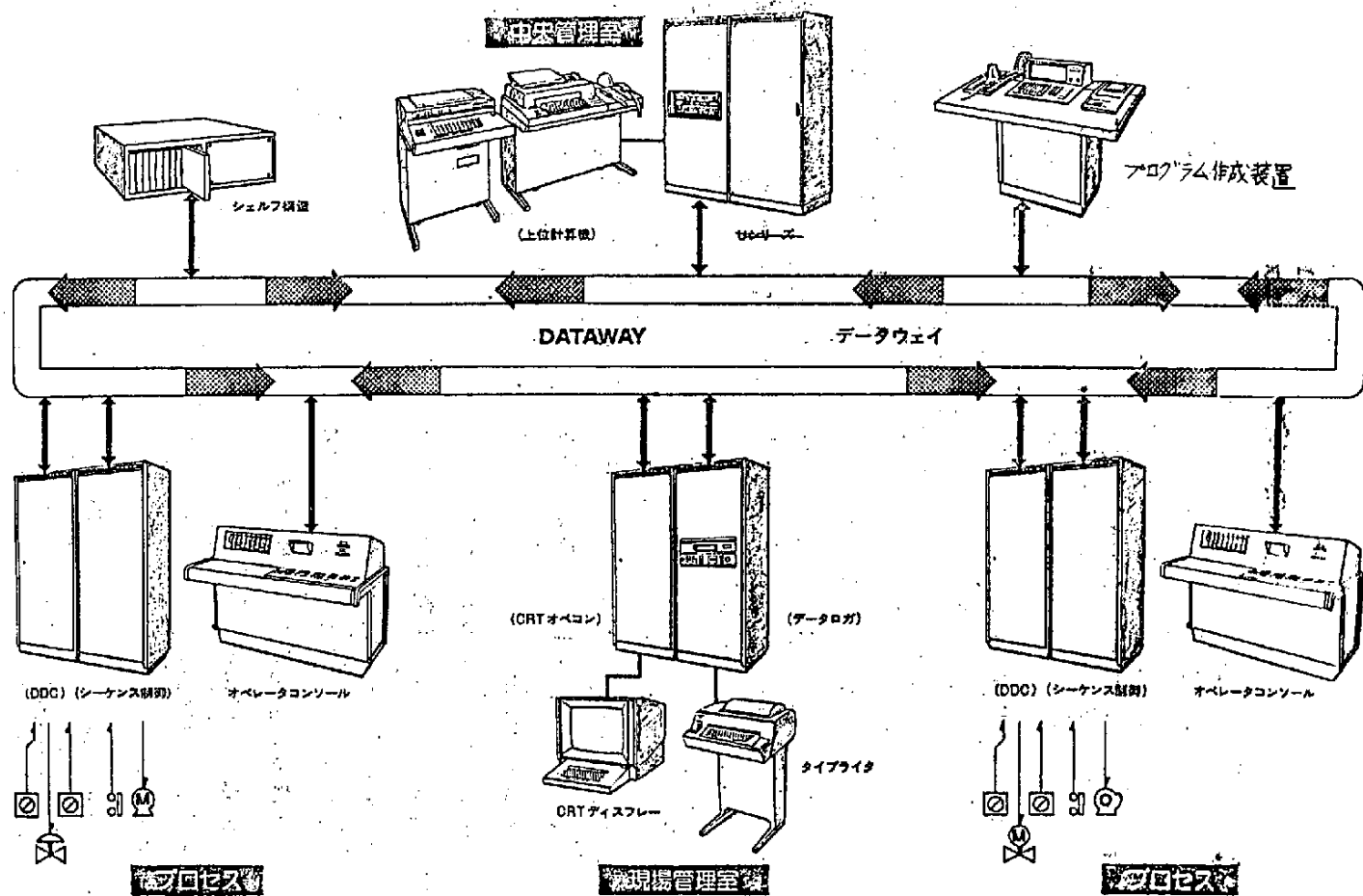
- ・制御の分散化による危険分散
- ・情報の集中化による全システムの運営の容易化
- ・ジョブの分担によるシステムの簡素化

等である。

#### (b) 上位計算機システムとの結合方法

上位計算機システムとの結合方法には

- A プロセス入出力装置による結合



第4-14図 マイコンを導入したハイアラーキシステム例

B モデム結合

C テレメータ、テレコンによる結合

D 計算機間チャンネル結合

E データハイウェイによる結合

等がある。A～Dは主として1：1の結合方式である。EはN：Nの結合である。

(c) ジョブの分担

ハイアラキシステムでは制御レベルと管理レベルの2つのレベルに大別できる。制御レベルのものにマイクロコンが適用できる。

各ジョブに応じて

- ・DDC装置（フィードバック制御用）
- ・シーケンサ（シーケンス制御用）
- ・データロガ（データ処理用）
- ・監視装置（監視操作用）
- ・インターフェース装置（上位レベルとの結合用）

等に分類出来る。

第4-14図にハイアラキシステムの一例を示す。

(d) 他システムによるサポート

従来のストアードプログラム方式のシーケンサでは、シーケンサ本体にプログラム作成装置を持っているものが多かった。しかしマイクロコンピュータを利用した制御機器の場合、従来通り制御機器本体にプログラム作成装置を持たせる方法の他に、他のマイクロコンをプログラム作成装置として使用し、各制御機器からプログラミング機能をはぶくことも可能となりシンプルな構成にすることができる。尚ハイアラキシステムでは上位レベル（管理レベル）の計算機に、これらプログラミング機能をサポートさせ、全体システムとしてより経済的合理的な制御システムを構成することが出来る。

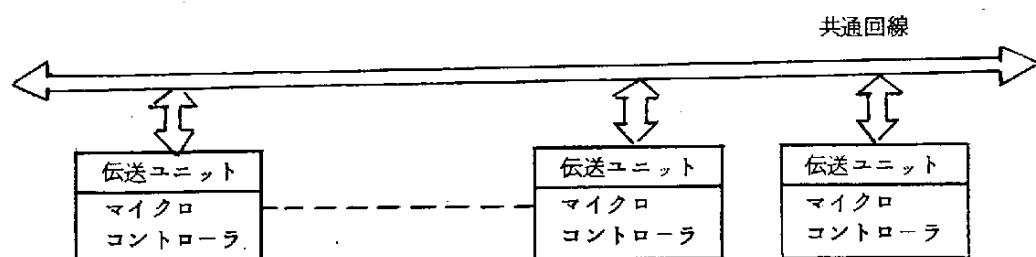
4.3.3. 制御システム間の伝送

制御レベルの制御機器間のデータ伝送（制御システム間の伝送）を考えると、その伝送形態は、N：N方式がすぐれている。N：N方式は、

各種のデータハイウェイ装置として近年脚光をあびてきている。ここではマイクロプロセッサを内蔵した各種制御機器間の任意のデータ伝送を行なう簡易データ伝送システムであるデータウェイシステムの解説を行なう。

#### (a) システムの構成

第4-15図に示すように各制御機器は伝送ユニットを介して、共



第4-15図 マイクロコントローラ間のデータ伝送例

通回線（マイクロデータウェイ）に接続されている。

#### (b) 伝送システム仕様例

共通回線に接続可能な制御機器数；16台

伝送方式；N：N

共通回線の総延長；最大 25 Km

データ伝送速度

| 共通回線の長さ | 1.5 km    | 5 km      | 15 km     | 25 km   |
|---------|-----------|-----------|-----------|---------|
| 伝 送 速 度 | 9.600 B/S | 4.800 B/S | 1.200 B/S | 300 B/S |

信号；半二重直列パルスコード無変調（NRZ）

伝送路；一般通信ケーブル（CPEV 0.9φ4芯）

伝送信号レベル；DC 24 V

同期方式；調歩同期方式

誤り制御；垂直水平パリティ

再送機能；誤りの場合自動再送

(c) 伝送制御機能概要

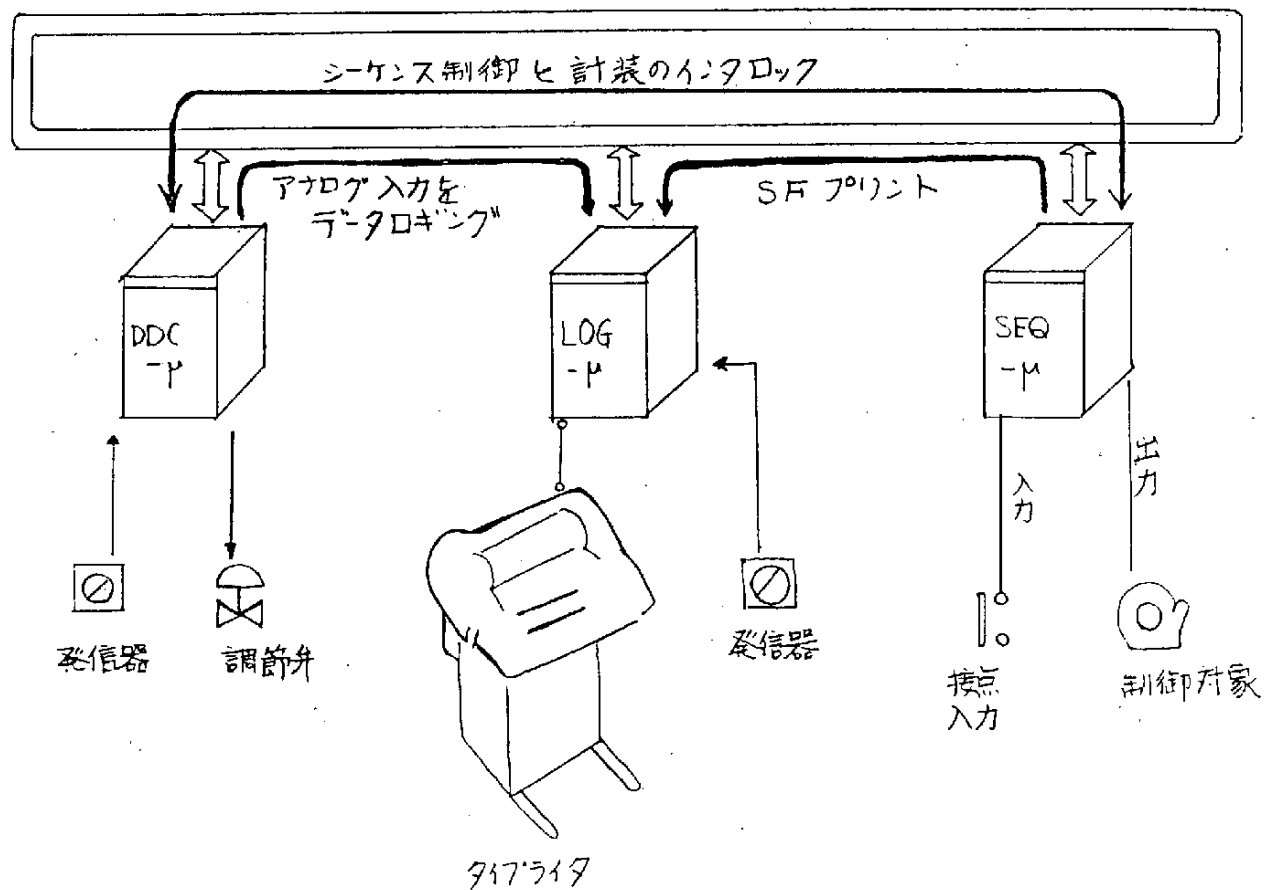
伝送制御方式は、各制御機器の伝送ユニットの機番の若い方から順次、共通回線の回線支配権を確立し、その制御機器の伝送が終了したら次の制御機器に回線支配権を渡してゆき、最後の制御機器まで終了したら又先頭の機器を渡してゆくサイクリック方式である。もし途中で制御機器が故障した場合、その制御機器は自動的に共通回線から切り離される。

(d) 制御システムと伝送機能

第4-16図に示すように共通回線を介してシーケンサ、DDC装置、ロガー等が接続しており、プラントのスタートアップ時のインターロック信号や定時刻データ等がデータとして伝送される。これらのデータ伝送によりシーケンス制御、フィードバック制御及びデータ記録が有機的に結合された制御システムが実現できる。又監視装置からDDC装置に対する設定変更等が伝送回線を通して送られる。

適切なデータ伝送システムを採用することにより、制御システムのジョブの分担、危険度の分散、複合システムの簡素化等が実現可能になる。このように制御システムに於けるデータ伝送は重要な役目をはたしており、どのような伝送システムにするかは制御システム構成上重要な課題となる。

# マイクロデータウェイ (DPCS-M)



第 4 - 1 6 図 マイクロデータウェイによる各コントローラの結合



## 4.4 制御機能と制御要素

### 4.4.1 制御システムの仕組み

マイクロコンピュータ制御システムはハードウェアとソフトウェアが融合して、所定の機能を果たす。この機能を分解していくと、第4-17図のように、制御システム→制御機能→制御要素（またはデバイス）→サブルーチン（またはコンポーネント）のようになる。制御システムはいくつかの制御機能の組合せからなる。たとえば、データロガシステムは入力処理、印字制御（印字フォーマットの制御など）、出力処理、監視、オペレーターコンソール、伝送の各機能からなり、これらは制御システム同様ハードとソフトが一体となつて一つの機能を持っている。さらに、制御機能はソフト面では制御要素、ハード面ではデバイスの組合せで構成される。たとえば入力処理機能はA I（アナログ入力）またはD I（デジタル入力）などのデバイスを用い、データ読込み、リニアライズ、工業値変換、ファイリングなどの制御要素から組立てられる。制御要素はさらに、いくつかのサブルーチンから構成され、デバイスは数個のコンポーネントの集りであると考えられる。

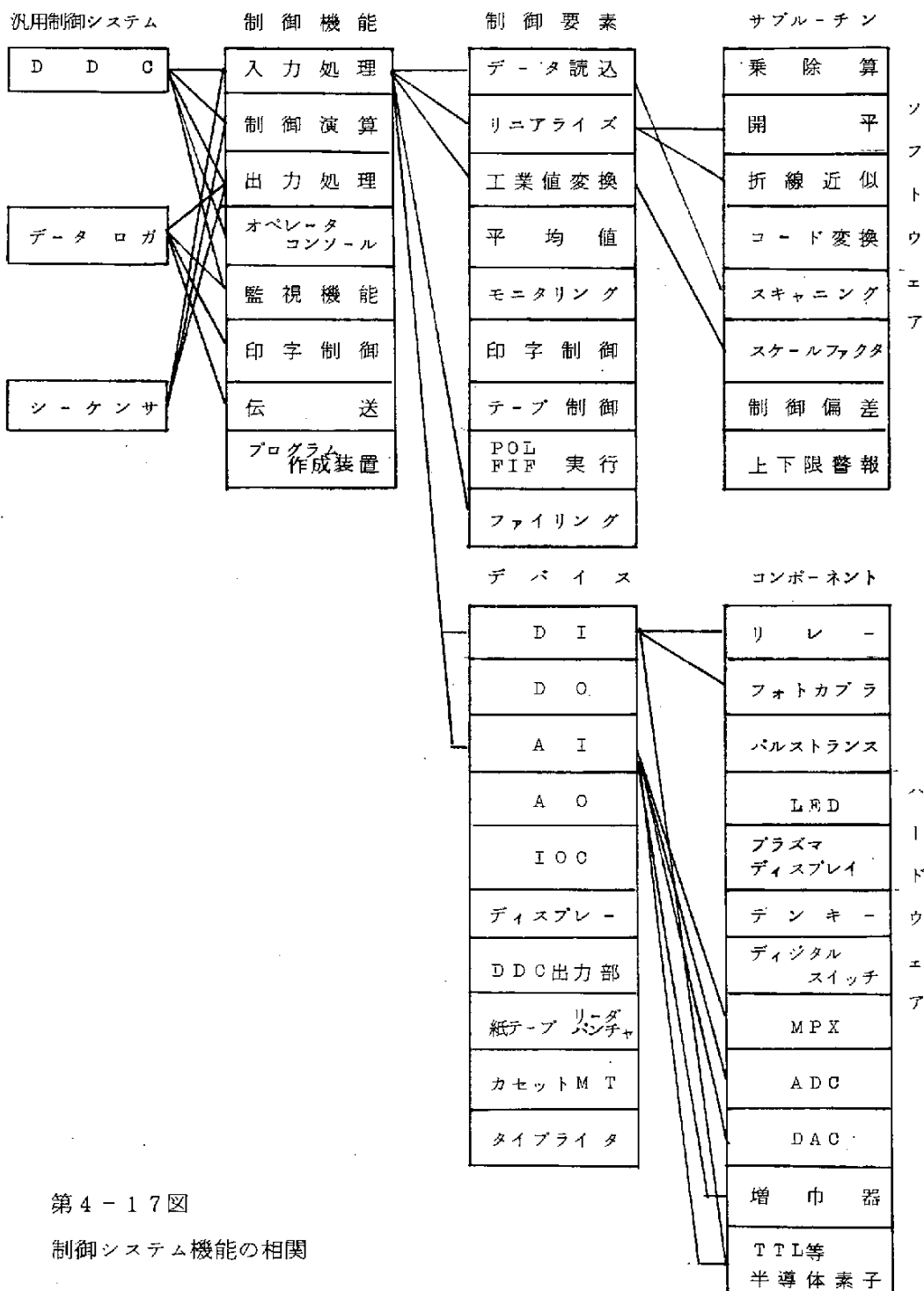
従来の計算機制御システムでは第4-17図の汎用制御システムのいくつかを1つのシステムで行なうため、第4-18図のように1つのハードウェアに対して制御機能→制御要素→サブルーチンにいたる一連のソフトウェアを組み込むことにより、その都度システム全体の要求にかなつたものを作り上げた。（実線の範囲）

これに対してマイクロコンピュータ制御システムは、これらの汎用システムそれぞれに個別にハードウェアを考え、図のように必要な制御機能のみ組込んでパッケージするのがその特色である。（点線、一点鎖線の範囲）

従つて、これら制御機能をハードウェア（デバイス）重点に実現するか、ソフトウェア（制御要素）主体に具体化するかが、パフォーマンスのよいシステムを作る一つの鍵である。

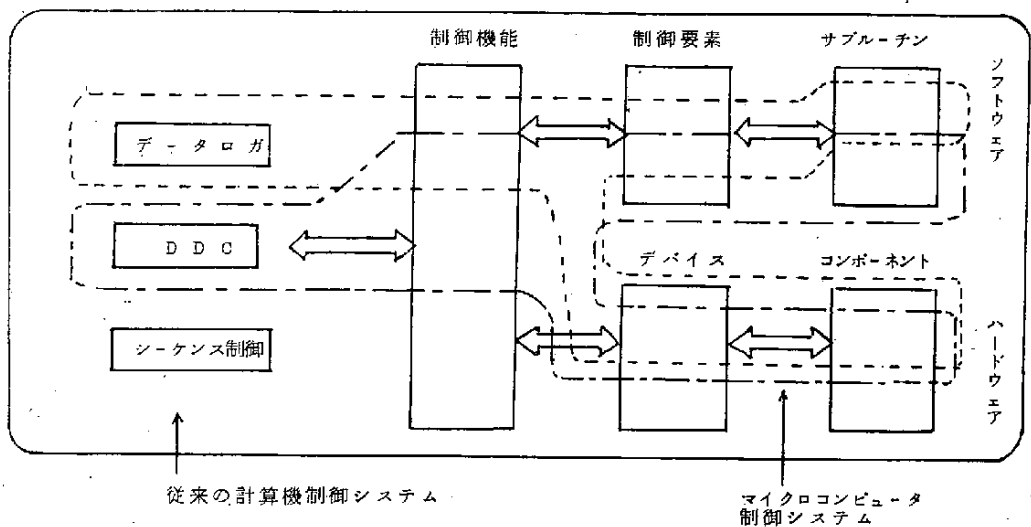
### 4.4.2 制御機能と制御要素

制御機能、制御要素などの定義や区分は明確ではないが、ここではこ



第4-17図

制御システム機能の相関

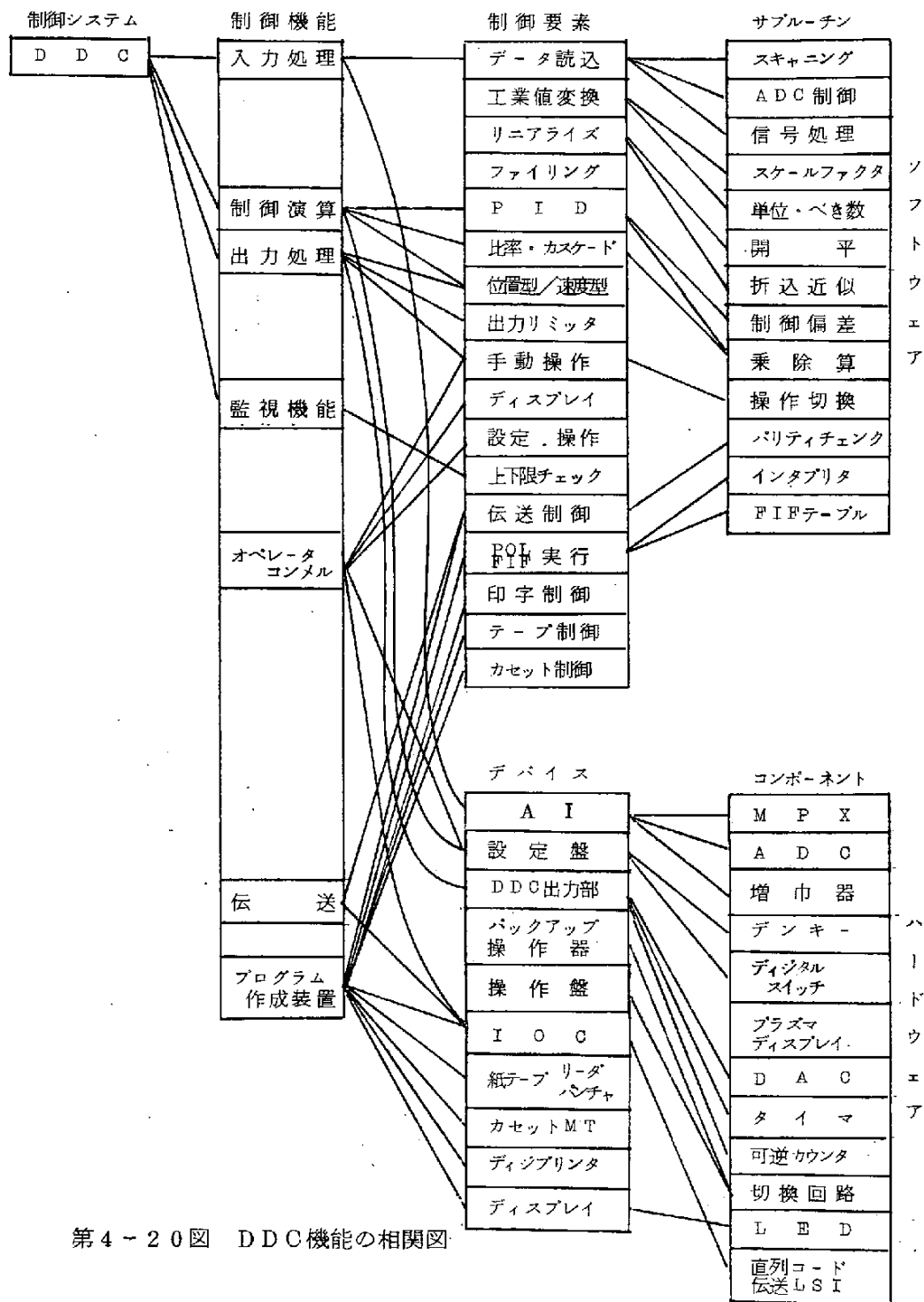


第4-18図 マイクロコンピュータ 制御システムの機能分割

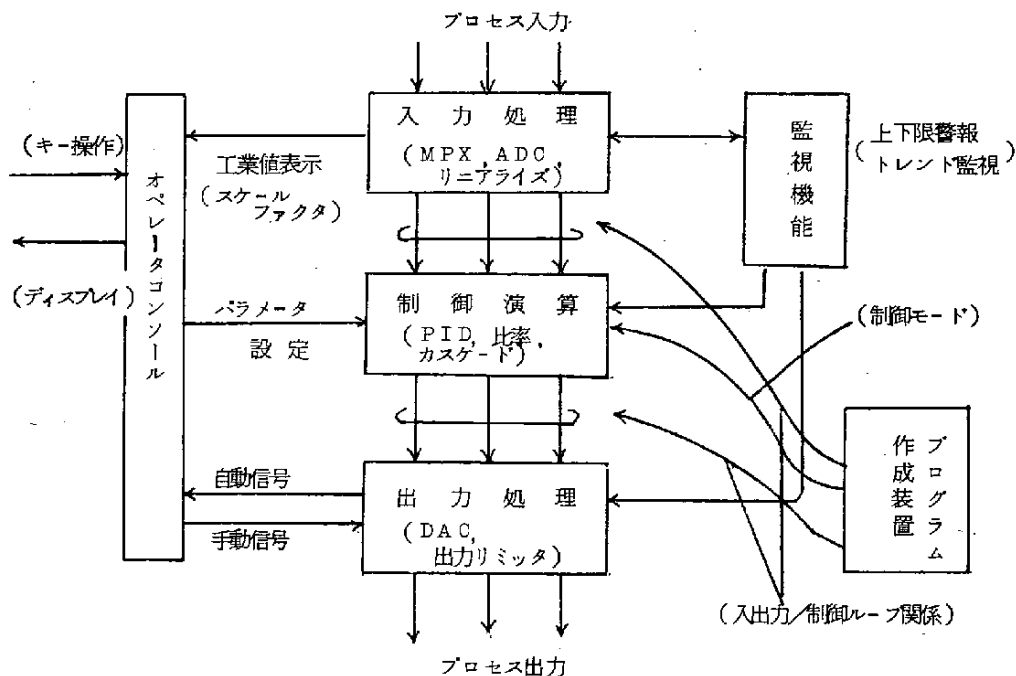
これらの関連を DDC に例をとって解説する。

第4-19図は信号の流れからみた DDC のブロックダイヤグラムである。プロセスからの入力信号は入力処理に入り、データ処理されたのち、制御演算を行なう一方、工業値変換されて、オペレータコンソールにその表示を行なう。オペレータコンソールから設定された設定値や制御パラメーターに従って、定値、比率、カスケードなどの演算を行なう。出力データは出力処理により制御信号としてプロセスに送り出される。手動制御のための信号がオペレータコンソールから出力処理に送られ、所定の制御モードに応じて調節弁などを開閉する。入力処理された信号はこのほかに監視機能を経て、常時プロセス変数、設定値、出力制御信号などのモニタリングを行なう。さらにプログラム作成装置により、入力処理、制御演算、出力処理の相互関係をプログラムし、制御ループの構成や比率、カスケード制御の関連を決定する。

これらの関係を第4-17図の相関図に従って明らかにすると、第4-20図のようになる。中継端子盤などで整線、信号処理されたプロセ



第4-20図 DDC機能の相関図

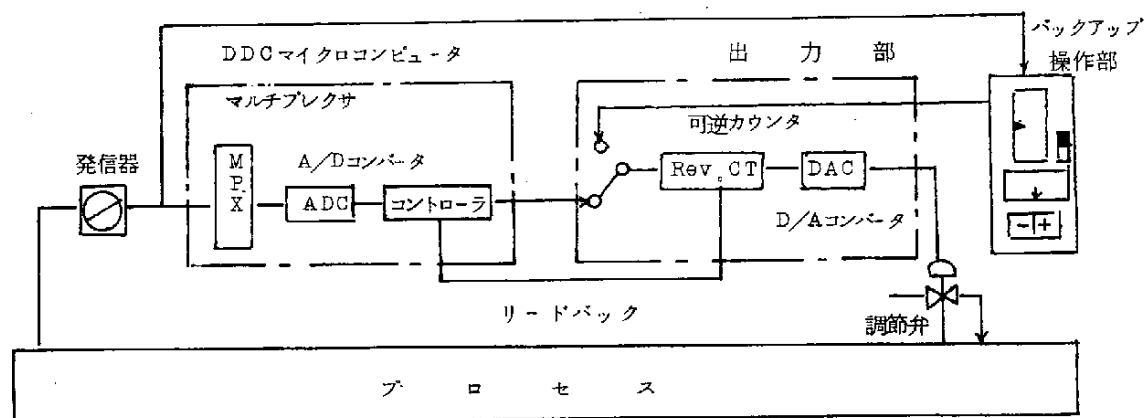


第4-19図 信号の流れからみたDDCの構成

ス入力は入力処理される。入力処理はマルチプレクサ (MPX)、AD変換器 (ADC) および増巾器などのコンポーネントからなるアナログ入力デバイス (AI) を用いて、データ読込、ソニアライズ、工業値変換の制御要素からなる3つのジョブを行なう。データ読込はさらに所定のチャンネルを選択するスキヤニング、増巾器選択、デジタル出力読込を含むADC制御、信号のベース処理やデジタルフィルタなどの信号処理からなる。また、工業値変換はスケールファクタ、単位・べき数などの処理を行ない、リニアライズは開平・折線近似などのサブルーチンを組合せて使用する。

入力処理機能を通じて得られたデータは、ファイリングされたあと制御演算を施す。制御演算はDDCの場合PID演算が基本であるが、比率、カスケードなどの演算も必要ならば行なう。PID演算は制御偏差を入力とし、比例、積分、微分演算を行なつて制御出力を算出するため、偏差、乗除算などのサブルーチンを用いる。デバイスとしては設定値や制御係数を与える設定盤が用いられる。

出力処理は制御演算よりの制御信号を受信し、これを位置型PIDアルゴリズムのときはDA変換し、速度型のときはパルス出力に変換する。さらに、出力リミッタや手動制御出力の切換処理を行なう。手動制御にはオペレータコンソールレバックアップ操作器が用いられ、出力デバイスでその信号が切換えられる。出力デバイスは可逆カウンタDA変換器、切換回路などのコンポーネントよりなる。(第4-21図参照)



第4-21図 DDC出力デバイス

オペレータコンソール制御機能はディスプレイ、設定、操作盤、コンソール制御デバイスなどの各デバイスよりなり、これらはさらにテンキー、LED、プラズマディスプレイ、直列コード伝送用LSIなどのコンポーネントよりなる。これらのデバイスを使用してコンソールはデータ表示、パラメータ設定を行なう。

監視機能は入力処理された信号をもとに工業直、設定値、制御偏差、変化率などの上下限警報を行なう。

## 4.5 制御システムの具体例

### 4.5.1 制御システムの分類

#### (a) マイクロプロセッサ適用基準

マイクロプロセッサ適用上の基準、目安として、下記のようなものがある。これらはたがいに関連を持ち、制御システム具体化の判断基準となる。

- (1) コスト ( 100~1000万円/システム )
  - (2) ビット長 ( 4, 8, 16 ビット )
  - (3) 汎用制御システム ( データロガ、DDC など ) の適用
  - (4) 適用分野 ( 交通、医用、マシンコントロール など ) の特質
  - (5) 具体化 ( コンベア制御、浄水場制御 など ) の問題点
  - (6) サブシステム ( I/O、センサ など ) とのバランス
  - (7) 製造技術 ( 生産性、パッケージ化 など ) 追従性
- (b) ビットによる適用性
- 現在マイクロプロセッサの語長は4ビット8ビット(12ビット)、16ビットがあるが、それぞれ下記のように使い分けられる。
- 4ビット………10進計算 ( 例：プログラマブル計算機、科学計算、P.O.S.、クレジット照合 )
- 低速制御 ( 例：マシン制御、医用、測定器、データ収集、コピー、電子ゲームマシン等 )
- 8ビット………BCD/2進演算 ( 例：インテリジェント端末、ワード処理、科学計算、小型会計器、信号変換等 )
- 高速制御 ( 例：電算器用辺機器制御、プロセス制御、通信制御、ラボ計測、データ収集等 )
- 16ビット………高速高精度演算 ( 例：科学計算、コンピュータグラフィック )
- 高速度精度制御 ( 例：プロセス制御等 )
- (c) マイクロコンピュータ アプリケーションの具体例 ( プロセス制御システム )
- これらは従来ミニコンや個々の制御装置により行なわれていたものの置換である。
- (1) 鉄鋼
- 高炉 ( 原料装入、高炉制御、熱風炉 )、転炉、焼結、連続鑄造、

スラバヤード、分塊、圧延、精製ライン、炉燃焼制御、秤量、エネルギー・センタなど

(2) 非鉄

アルミ精錬、自溶炉、亜鉛蒸溜炉など

(3) セメント

長距離コンベア、コンベア総括制御、キルン制御、ブレンディング、ミルなど

(4) 上下水道

ポンプ制御、薬注、浄水場制御、管網制御、排水制御、下水処理制御、畑地灌漑など

(5) 電力

ゲート制御、安定化装置（AVR、ABQRなど）集中制御用中継、自動選択、自動復旧、SFプリント、データ交換、自動起動、総合監視など

(6) 化学

一般化学プロセス、石油精整、重合反応制御、ブレンディング、陸海上出荷、電解槽、パイプラインなど

(7) 流通、工作機械

ローダ、アンローダ、エレベータ、ロボット、工作機械制御、プレス制御、シア、放電加工、塗装、溶接ライン、空気移送、自動倉庫、搬送システム、自動コンベアライン、工場内流通、コンテナヤードなど

(8) その他

醸造プラント、サイロ、バッチプラント、硝子切断、抄紙機、照明制御、シアタコントロール、ビル管理、ボイラACC、大気汚染監視、水質モニタ、乳剤調合、新交通システム、トンネル監視制御、交通管制など

(d) マイクロコンピュータ アプリケーションの具体例（その他）

この範ちゆうとしては自動試験や自動測定、自動分析などのデータ処理システムを制御システムとして掲げることができる。このほか



POS ターミナルや会計器などマイクロコンピュータ本来のアプリケーションも参考までに掲載した。

## (1) 自動試験

製品試験機：タイヤ試験機、エンジンテスト、モータ、特性試験機、WHM 試験機、電源装置試験機、布線試験機、コンピュータ試験機

保守点検試験機：車輛試験機、自動車検など

信頼性試験機：耐久テスト、ライフテスト、LCR テスタ、環境テスト

部品、ユニット試験機：IC テスタ、デジタル／アナログテスタ、ロジックテスタ、プログラマブルテスタなど

以上心臓部はマイクロコンピュータで良いがセンサやインターフェース及び専用言語の開発が必要である。

## (2) データ処理

データログ：連続プロセスデータログ（日報主体）、多点モニタ、受変電データログ、発電所データログ、エネルギーセンタ、公害モニタなど

データログ＋小規模制御：バッチプロセスデータ処理、制御（誘導炉、共同溝、ポンプ場、発酵槽）、デマンド制御電解槽、非鉄金属精練、秤量所の自動化、電力系統運用など

オンライン分析データ処理：成分分析（鉄鋼、セメント）、自動分析、海洋観測資料分析、製品品質分析、プロセスアナライザ、ガスクロ、排気ガス分析など

検査：工程検査、ロジック回路自動検査、生化学検査自動処理、電極棒形状検査、きず検査、鏡板の形状検査、システムの異常検出診断など

試験（特性実験）：エンジンテスト、測定器出力データチェック、実験データ解析処理、材料試験、信頼性テストなど

自動測定：LCR 測定、公害自動測定、データレコーダ、車リヤウイドヒータ抵抗測定、自動測定補正、インテリジェントインスツルメント、インテリジェントセンサなど

技術データ処理：水配分シミュレーション、高電圧電気過渡現

象データ処理など

科学計算：NCボール盤用コントロールテーブル作成、アナコンの代用、エミュレーション、設計計算、光学計算、数値制御のソフト作成、熱効率計算、統計分析、資源探査情報処理など

事務管理：生産、売上事務管理、在庫管理、請求書作成、経理帳簿、財務統計の作成、人事管理、給与計算、学生の成績管理、入試業務、生産データ集計・処理、公営競技のデータ収集、資材管理、自動検診、生産管理、図面管理、図書管理、粗利益計算など

### (3) 新分野

計算：パーソナルコンピュータ、プログラマブル桌上計算機、電卓など

会計：電子レジスタ、医用会計機、在庫会計機、銀行窓口会計機、フロント会計機、ビルディングマシンなど

精算：キャッシングデスベンサ、POS 端末、ルートセールス用端末、伝票発行機、クレジット照合機、磁気カード照合機、バーコード読取機、給油所向端末、駐車場料金計算機など

事務：在庫管理機、集中管理機、タクシー配車管理、工程管理、インクワイアリ端末など

計算機用周辺機器：CRT 制御、ディスク制御、テープ制御、タイプライタ制御、データ収集端末、ディスプレイ端末など

通信：データ通信、電話伝送など

医療：血液分析、医療機器、カルテ管理など

自動車：シートベルト、スキッドコントロール、燃料計、安全装置、エンジンテストなど

教育：ティーチングマシン、訓練所シミュレータなど

## 4.5.2 制御システムの具体例

### (a) 汎用制御システム

すでに述べたように、制御システムはいくつかの制御機能からなっているが、その典型的な制御機能を主体にしてまとめると次のような

汎用制御システムが掲げられる。

- (1) データロガシステム
- (2) DDC (ダイレクト・デジタル・コントロール) システム
- (3) シーケンスコントロールシステム
- (4) 位置ぎめDDC システム
- (5) 遠方監視制御システム
- (6) データ収集システム
- (7) 自動試験システム
- (8) 監視システム
- (9) マンマシンインターフェースシステム
- (10) ディスプレイシステム

この中から、制御システムに不可欠の制御機能として、(1)データロガシステム、(2) DDC システム、(3)シーケンスコントロールシステムについてその概要を述べる。

(b) データロガシステム

データロガは制御機能の基本となるもので、アナログまたはデジタル入力を順次スキャンし、入力処理を施したのち、一定のフォーマットに従って、種々の印字を行なう。さらに、監視機能により、上下限警報やSF (Sequence Fault) プリントも可能である。このほか、オペレータコンソールによりデータ表示や上下限值設定も可能である。

データロガシステムは主に印字フォーマットの仕様の違いから、一定の仕様のシステムを作ることはむずかしい。第4-22図はデータロガシステムの一例で、その仕様は下記のごとくである。

入力点数： 128点 (アナログ及びパルス)

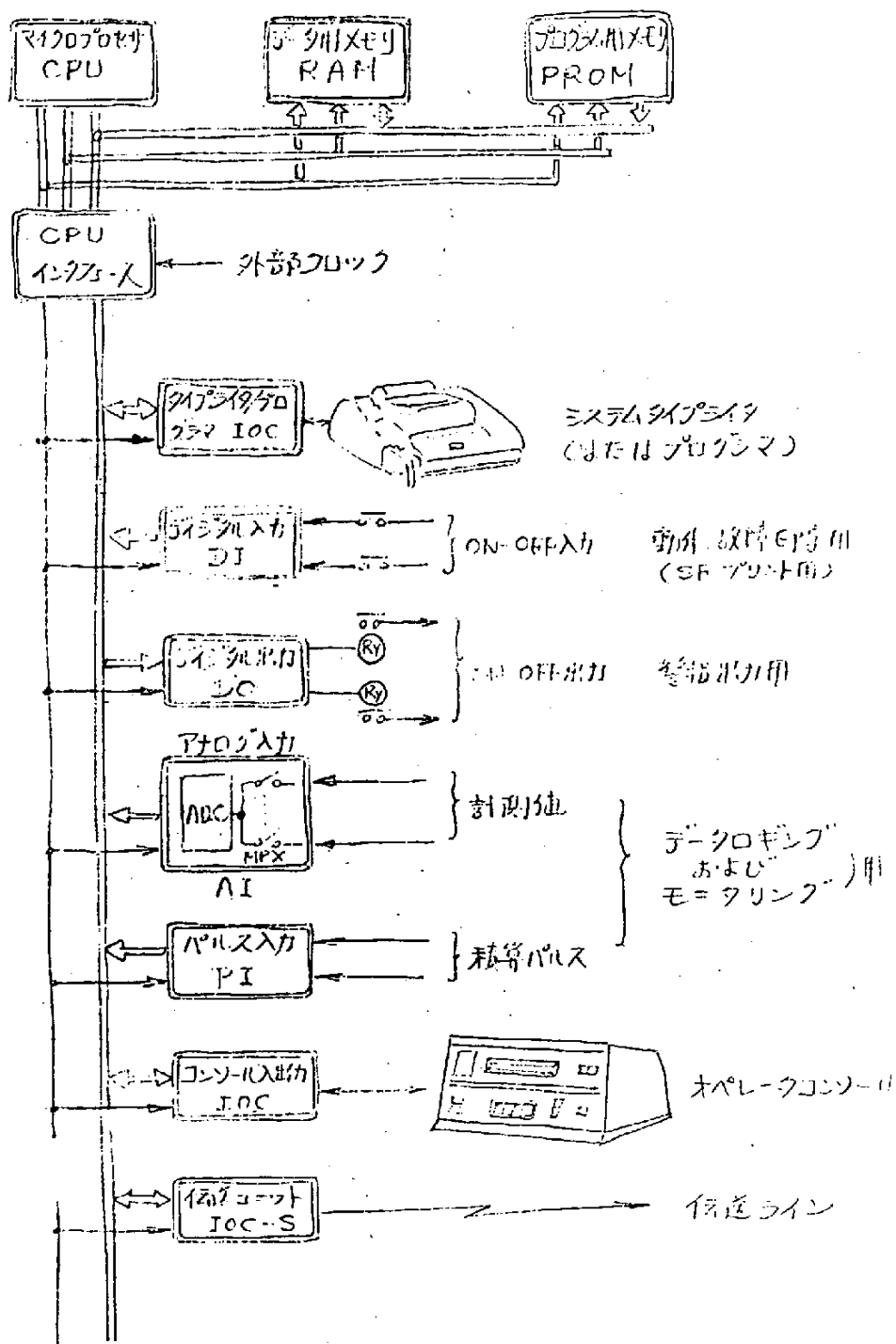
アナログスキャン速度： 32点/秒

入力パルス周波数： 16 Hz 以下

コード入力点数： 8ワード (1ワード：16ビット)  
(最大) } 合計  
128ビット

動作故障入力点数： 128点 (最大)

上下限警報： 128点 (最大)



第 4 - 2 2 図 データロギングシステムのブロックダイアグラム (例)

日報処理点数： 40点（最大）

タイプライタ： ASR、IBM、DIABLO

印字形式： 時報——3文字+1スペース/項目

日報——5文字+1スペース/項目

任意印字、警報印字： 有

コンソール： 一体形 別置形

警報出力： 256点（最大）

伝送： データウェイ方式

電源： AC100V $\pm$ 10% 50/60Hz

(c) DDC システム

DDC（ダイレクト・デジタル・コントロール）システムはすでに計算機システムを用いて実用化されているが、システムの信頼性や価格の面で一部の分野にしか適用されなかつた。マイクロコンピュータ利用のDDCはこれらの問題点を解決して、より広い分野に使用される可能性をもっている。経済的に釣り合う制御ループ数も従来のDDCが100から200ループであるのに対して、マイクロコンの場合は10から60ループ程度である。第4-23図はDDCの一例で、その仕様は次のとおりである。また第4-24図はDDC用オペレータコンソールの一例で、プラズマディスプレイとキーボードを使用している。

制御ループ数： 8/16

演算方式： デジタル16ビット並列演算

演算ユニット： 16ビットマイクロプロセッサ

演算周期： 0.5秒サイクリック（最高）

制御入力： アナログ入力16点、1~5VDC

制御出力： 8/16点

増減2方向パルス幅または4~20mAADCバックアップ

切換回路付

オペレータコンソール： オプション

伝送： MODEM結合方式

## チャンネル結合方式

バックアップ： バックアップ操作器

電源： AC100V $\pm$ 10% 50/60Hz

制御演算： ○制御モード——定値、比率、カスケード、プログラム  
制御

○制御演算——PID、フィードフォワード

警報： 測定値、偏差、変化率、設定値

### (d) シーケンスコントロールシステム

シーケンスコントロールをマイクロコンピュータで行なう方法は種々あるが、ここでは最も汎用的なシーケンサについて述べる。シーケンサはマイクロコンピュータ応用の形態では翻訳型の典型である。すなわち、ブール代数をもとにして書かれたシーケンサの言語をインタプリタで翻訳しながら実行するわけであるから、命令実行スピードが問題になる。この点を除けば、シーケンサはマイクロコンピュータを適用する上で、最も簡単なシステムの一つといえよう。

マイクロコンピュータ応用のシーケンサの一例を第4-25図に示す。このシーケンサの仕様は下記のとおりで、その命令一覧表を第

表に示す。

プログラム方式： ストアード・プログラム方式

命令種類： 35種類

語長： 16ビット+2パリティビット

演算方式： 常時繰返し演算

演算機能： 論理演算

シーケンスステップ数： 5000ステップ

1000ステップ

演算時間： 100 $\mu$ s/1命令

タイマ： 最大64点(1秒単位、1秒~9時間設定)

入力、出力： 最大合計512点

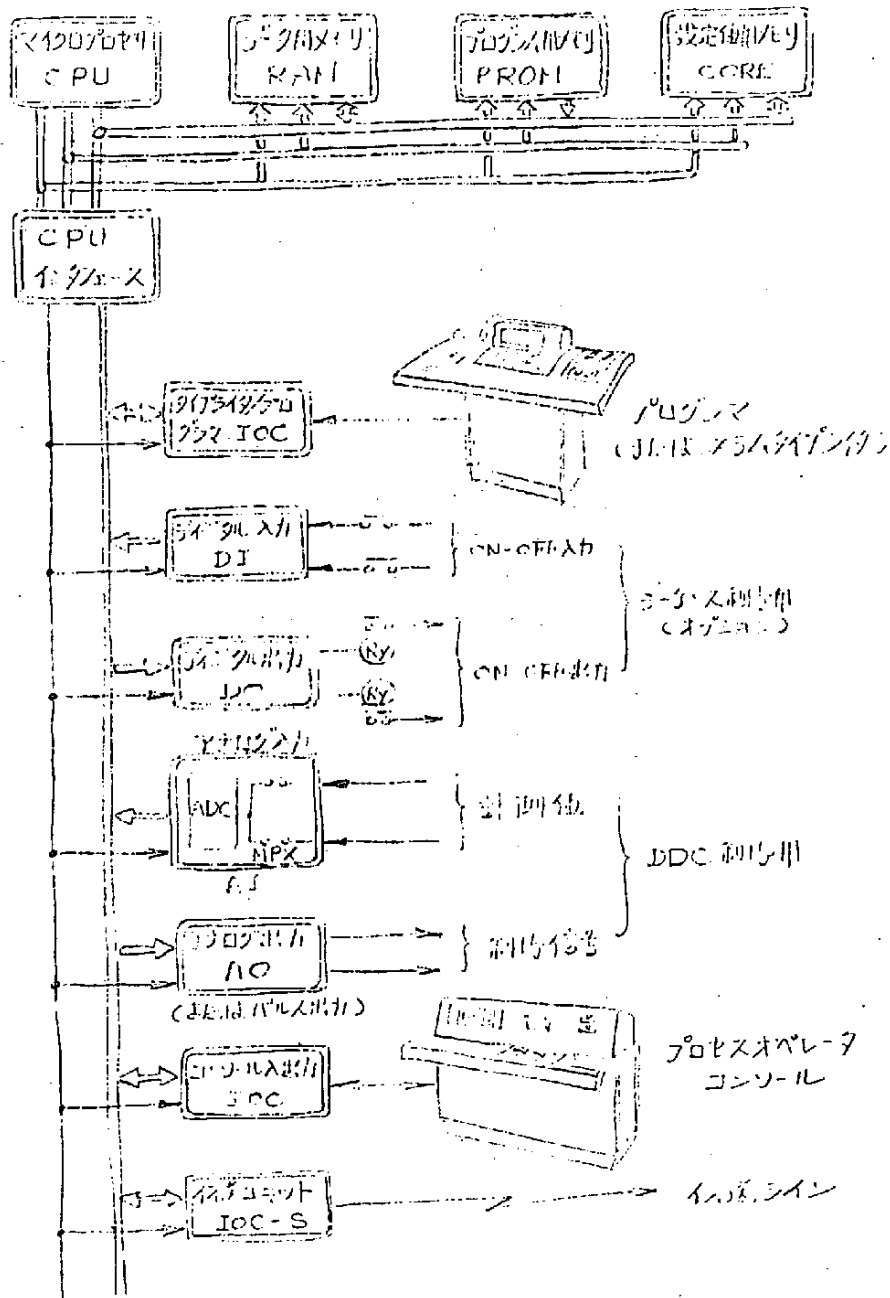
入力： 絶縁形

出力： 小形リレー

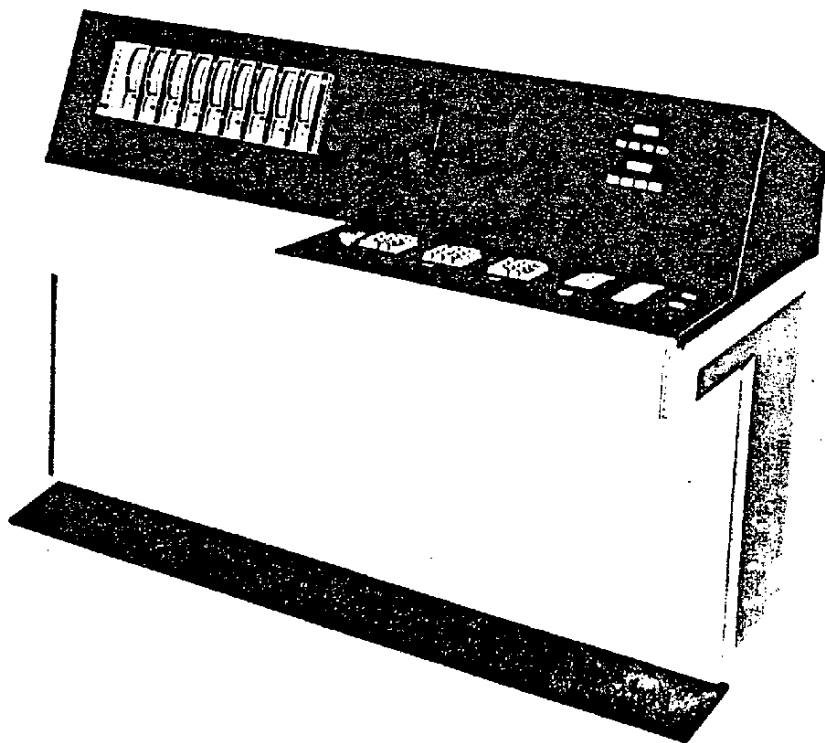
トランジスタスイッチ

電源： AC100V $\pm$ 10%

50/60Hz

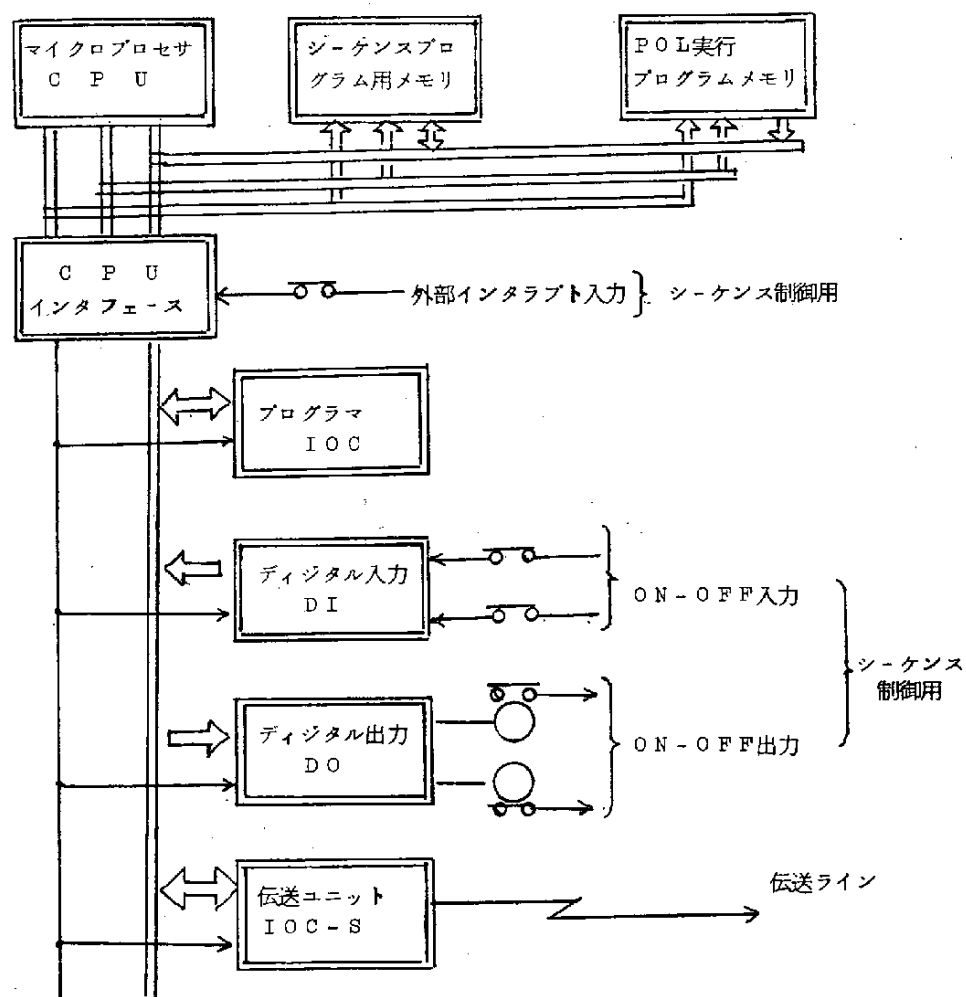


第 4 - 2 3 図 DDCシステムのブロック  
ダイアグラウ(例)



第4-24図 DDC プロセスオペレータコンソール





第 4 - 2 5 図

シーケンス制御システムのブロックダイアグラム (例)

|                            |       |        |             |            |                                                                                      |
|----------------------------|-------|--------|-------------|------------|--------------------------------------------------------------------------------------|
| B<br>I<br>T<br>命<br>令      | 命 令   | 命令語    | デバイス番号      |            | 演 算 内 容                                                                              |
|                            | READ  | (正) R  | J           | B          | $(A) \rightarrow MOO(J \cdot B) \rightarrow A(SQCT) + 1 \rightarrow SQCT$            |
|                            |       | (逆) RN | J           | B          | $(A) \rightarrow MOO(\overline{J \cdot B}) \rightarrow A(SQCT) + 1 \rightarrow SQCT$ |
|                            | AND   | (正) A  | J           | B          | $(A) \cap (J \cdot B) \rightarrow A(SQCT) + 1 \rightarrow SQCT$                      |
|                            |       | (逆) AN | J           | B          | $(A) \cap (\overline{J \cdot B}) \rightarrow A(SQCT) + 1 \rightarrow SQCT$           |
|                            | O R   | (正) O  | J           | B          | $(A) \cup (J \cdot B) \rightarrow A(SQCT) + 1 \rightarrow SQCT$                      |
|                            |       | (逆) ON | J           | B          | $(A) \cup (\overline{J \cdot B}) \rightarrow A(SQCT) + 1 \rightarrow SQCT$           |
|                            | WRITE | (正) W  | J           | B          | $(A) \rightarrow J \cdot B(SQCT) + 1 \rightarrow SQCT$                               |
|                            |       | (逆) WN | J           | B          | $(A) \rightarrow J \cdot B(SQCT) + 1 \rightarrow SQCT$                               |
|                            | 命 令   | 命令語    | ステップ番号      |            | 演 算 内 容                                                                              |
| 分<br>岐<br>命<br>令           | JUMP  | B      | S           |            | 無条件分岐 (S) $\rightarrow$ SQCT                                                         |
|                            |       | J      | S           |            | $(A) = 1(S) \rightarrow SQCT, (A) = 0(SQCT) + 1 \rightarrow SQCT$                    |
|                            |       | JN     | S           |            | $(A) = 0(S) \rightarrow SQCT, (A) = 1(SQCT) + 1 \rightarrow SQCT$                    |
| 特<br>殊<br>命<br>令           | 命 令   | 命令語    | モディファイ      |            | 演 算 内 容                                                                              |
|                            | SET   | S      | M = 0       |            | $1 \rightarrow A_0 \sim F(SQCT) + 1 \rightarrow SQCT$                                |
|                            |       | SN     | M = 1       |            | $0 \rightarrow A_0 \sim F(SQCT) + 1 \rightarrow SQCT$                                |
| W<br>O<br>R<br>D<br>命<br>令 | 命 令   | 命令語    | デバイス<br>番 号 | モディ<br>ファイ | 演 算 内 容                                                                              |
|                            | LOAD  | LD     | J           | W          | $(J \cdot W) \rightarrow A(SQT) + 1 \rightarrow SQCT$                                |
|                            | STORE | ST     | J           | W          | $(A) \rightarrow J \cdot W(SQT) + 1 \rightarrow SQCT$                                |
|                            | ADD   | AD     | J           | W          | $(A) + (J \cdot W) \rightarrow A(SQT) + 1 \rightarrow SQCT$                          |
|                            | SUB   | SB     | J           | W          | $(A) - (J \cdot W) \rightarrow A(SQT) + 1 \rightarrow SQCT$                          |
|                            | AND   | AR     | J           | W          | $(A) \cap (J \cdot W) \rightarrow A(SQT) + 1 \rightarrow SQCT$                       |
|                            | OR    | OR     | J           | W          | $(A) \cup (J \cdot W) \rightarrow A(SQT) + 1 \rightarrow SQCT$                       |
|                            | EOR   | ER     | J           | W          | $(A) \oplus (J \cdot W) \rightarrow A(SQT) + 1 \rightarrow SQCT$                     |

第 4 - 9 表 シーケンサの命令一覧表

#### 第4-9表中

A : 演算レジスタ ( 16 ビット )

MOO : 演算メモリ ( 1 時メモリ )

( ) : 内容

$\overline{(\quad)}$  : 内容の逆信号

SQCT : シーケンスプログラム ステップ カウンタ

デバイス番号 : J : デバイスの種類 ( X : 入出力、M : 一時メモリ、

T : タイマ )

B : デバイス番号

ステップ番号 : S : 分岐シーケンスステップ番号

そ の 他 : M : モディファイ

W : デバイスWORD アドレス

$\cap$  : 論理積 ( AND )

$\cup$  : 論理和 ( OR )

$\oplus$  : 排他的論理和 ( EOR )

### 4.6 制御システムの展開方法

#### 4.6.1 展開手順

##### (a) 一般的な展開方法

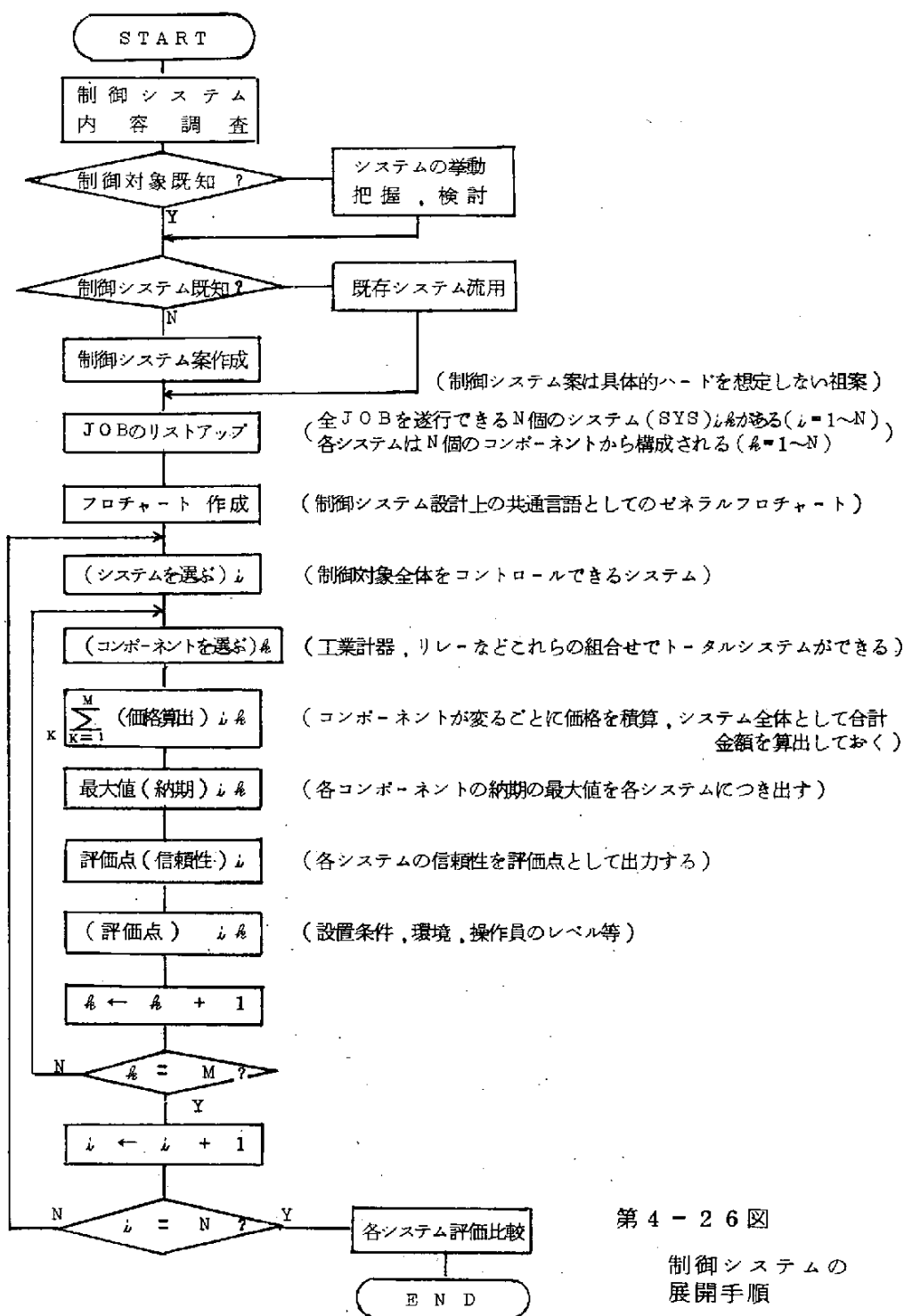
制御システムの一般的な設計の展開手順を第4-2.6図に示す。

ステップ1 設計を行なおうとする制御システムの概要を把握し、過去に同じような制御システムが既に存在しているか文献等により調査を行なう。

ステップ2 制御しようとするもの ( 制御対象の特性が既知のもの ) であるか検討し、未知の部分について制御対象及び制御装置を含むシステムの挙動を明確にする。

ステップ3 制御システムが既存のもので、十分その性能が満足すべきものかを検討する。もし機能的に満足すべきものがあればそれを流用する。

ステップ4 制御システム案を作成する。この際に具体的なハード



第 4 - 2 6 図

制御システムの  
展開手順

ウェアを想定せずシステムを合理的に運営できるような案を作る。

ステップ5 制御システムのジョブをすべて列挙する。それぞれのジョブを制御装置で行なわせるか、オペレータの判断操作にまかせるかを概略決定し、全制御JOBを遂行できる制御方法が何とおり考えられるかリストアップする。制御装置はどのようなコンポーネントから構成されるかも同時に検討する。例えば、シーケンス制御部を有接点リレーで行なうか、トランジスタにするかTTLにするか等の検討。

ステップ6 制御機能とオペレータの操作監視動作、制御対象の動作等システム全体の挙動をフロチャートにする。

ステップ7 制御対象全体をコントロールできる制御システムを選択する。

ステップ8 制御装置のコンポーネントを選択する。工業計機、リレー、マイクロコン等の組み合わせでトータルシステムができる場合、それぞれのコンポーネントを選択する。

ステップ9 コンポーネントが変わるごとに価格を積算し、システム全体の合計金額を算出しておく。ソフト、ハード費用及び試験、総合テスト等イニシャルコストと保守等のランニングコストに分けて算出する。

ステップ10 各コンポーネント(ハード、ソフト)の納期の最大値を出す。

ステップ11 各コンポーネントの信頼性を評価点に換算する。

ステップ12 設置条件、環境、操作員のレベル等を考慮して、ステップ9、10、11の評価をする。コンポーネントのすべてにつき同様の評価をくりかえして、そのシステムにつき総合評価をする。

ステップ13 考えられる他の制御システムについて、それぞれステップ7から同様の作業をくり返す。

ステップ14 すべての制御システムにつき評価をし、最高のシステムについて詳細設計に入る。マイクロコンピュータを制御装置とするシステムの展開につき次項に記述する。

(b) ハードウェアの展開

マイクロコンピュータを利用した制御装置では、ハードウェアのみでは動作しないことは通常の計算機システムと同じであるが、そのハード的取扱いは、一個の部品として処理されるべきものである。

ハードウェアの展開上、考慮すべき点を次に示す。

- A プロセス入出力装置の選択及びソフト、ハードのJOBの決定
- B メモリー容量
- C 停電時のデータ類の保護の要否
- D CPU、メモリの決定
- E 電源、構造
- F 設計、製作、試験方法
- G 保守方式

まず第一に、制御対象からのデータをどのような形で受け入れるか、又どのような形で出力するか、検出器、操作器を含め、その設置条件を考慮して、入出力装置の仕様を決定する。又、オペレーターとのインタフェースを決定する。この際どこ迄をワイアードで行ない、どこ迄をストアード（プログラム）で行なうかを決める。例えば、電動弁の開閉信号を出力する場合、パルス巾（時間）をプログラムにより時間をカウントして、デジタル出力をオンオフする方法と、時間巾に応じたデジタル出力を出し続けるパルス巾出用のハードウェアを使うか等の検討である。前者はソフトだけで処理でき、特殊な入出力装置が不要であるが、ソフトに負荷がかかり、後者はその逆である。

ソフト、ハードのジョブの分割後、メモリー容量を決定する。

第三に停電時にデータ類が消えてしまっても重大な障害にならぬかを検討する。もし不揮発性のものが必要であるなら、現時点では磁気メモリ（コア、ワイヤメモリ）を採用するか無停電電源を使う。

CPU の決定は通常の部品と異なりソフトウェアという財産と関連があるため、一度採用すると簡単に機種の変更ができにくいので選定には注意が必要である。

電源及び構造は制御装置の目的に応じて決定される。防塵構造とするとか、又停電検出処理等が通常要求される。

ハード設計に際してどのような方法で試験をするかをあらかじめ明確にしておき、それに適したハード分割にすべきである。又、保守方法、予備品の種類、トラブルシューティング方法等まで考慮して設計する必要がある。特にソフトウェアと一体となつて動作するマイクロコンシステムではトラブルシューティング方法をソフト面、ハード面の両面から行なう必要がある。

計測制御機器としてマイクロコンピュータを使う場合、設置条件は、従来の計算機システムよりは制御装置並みを要求され、システム的にはハイアラキシステムに容易に組み込めるようにデータ伝送機能が要求される等、多様な要求に適合させねばならぬので、ハードウェアの展開にあつてはその構成をフレキシブルなものにしておく必要があり、なおかつコンパクト性を失わないようにせねばならぬ。

#### (c) ソフトウェアの展開（パッケージ化）

従来のミニコンを使つた制御システムではオーダーメイドの色合いが強かつたが、マイクロコンピュータシステムでは、従来のワイヤード機器の置換、及び機能アップ等が目的であるため、又ソフト比率を大きくしないため、レディメイドのソフトウェアが要求される。しかし制御対象の機能は制御対象の特性に応じて種々の変化が要求される。これらの要求を満足させ、かつレディメイド的なソフト構成にするために、制御装置の機能をそれぞれ基本的機能に分割し、それぞれの機能ごとにパッケージ化し、これらを必要に応じて組み合わせて制御装置用のソフトウェアにする手法（パッケージ化）が必要となる。パッケージ化の展開の仕方の代表的なものが4.2.2項の第4-7図に示されている。

専用型： 仕様の変化が少ない。端末などでそのソフトは専用に作られる。特にメモリ効率を最高にして、ハードコストを下げるように設計される。同一ソフトで多量に販売できるものに向いている。

汎用型： 制御機能を基本的な要素に分解してそれに対応したソフトを製作しておき、アプリケーションエンジニアはそれらの制御機能を適宜組み合わせ、制御システムを簡単に作りあげられるようにする方法で、POL方式（翻訳型）とFIF方式がある。

#### (b) ハードのFabrication

マイクロコン利用の制御装置のハードウェアには、その用途に応じて専用型ハードと汎用型ハードがある。前者は大きさ、外形等装置の構造が用途別に設計されるもので、そのプリント板の分割方法も、その装置に専用に考慮される。そのためジャンボプリント板やフレキシブルプリント板の採用等も考えられる。多量生産が可能な機種に向いている。後者はCPU、メモリ、プロセス入出力等機能別にプリント板化し、それ等を組み合わせるハード構成法である。制御装置としてはその機能が多様なものを要求されるので、これらに適合させる方法として後者の構成法をとることが多い。これ等の例を写真第4-27図に示す。

#### 4.6.2 サポートシステム

マイクロプロセッサ応用製品の開発にはハードウェアのサポートとソフトウェアのサポートが必要であり、どのようなサポートシステムを使うかが、製品開発時間に大きな影響を及ぼす。

サポートシステムの一例を第4-28図に示す。

ソフトウェアのサポートには次のようなものがある。

A アセンブラー

B コンパイラー

C シミュレータ

これらはマイクロコンそれ自体を利用する方法とホストマシンを



利用する方法がある。

前者は手軽ではあるが、I/O 機器の性能やマシンの性能のため色々制約が多い。本図はホストマシンを利用した例を示してある。

ハードウェアサポートは、汎用型のハード構成のものに、デバッグ用コンソールをつけたシステム開発用評価マシン（システムエミュレータ）を利用する。本機にハードウェアの展開上で必要になったプロセス入出力を取りつけ、ソフトサポートによるシミュレーションの終了したプログラムと組み合わせて実機シミュレーションを行なう。ハードソフトの分割が適切であつたか等、系統的にデバッグして初期の目的を達しているならば、プログラムを R $\bar{O}$ M 書き込み器により R $\bar{O}$ M 化し、製品用ハードに取りつけて完成品とする。もし機能不足の場合又はソフトハードの分割が不適当であるならば修正する。第 4-29 図に製品と開発用評価マシンとの関係を示す。専用型のハードウェア及びソフトウェアの開発でも、まず評価用マシンでプログラム及びハードの機能を含むシステム全体の評価をして専用ハードに展開する方が、ソフト、ハードのバランスがとりやすく、システム開発の時間が短縮される。

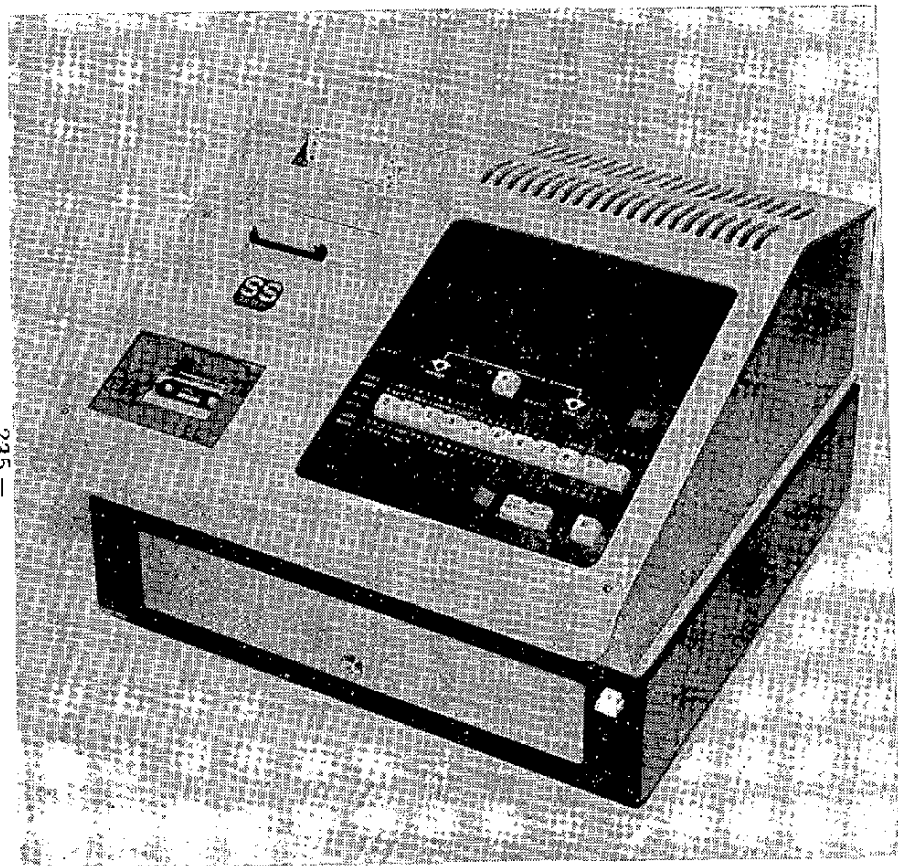
#### 4.6.3 POL 及び FIF

##### (a) 概要

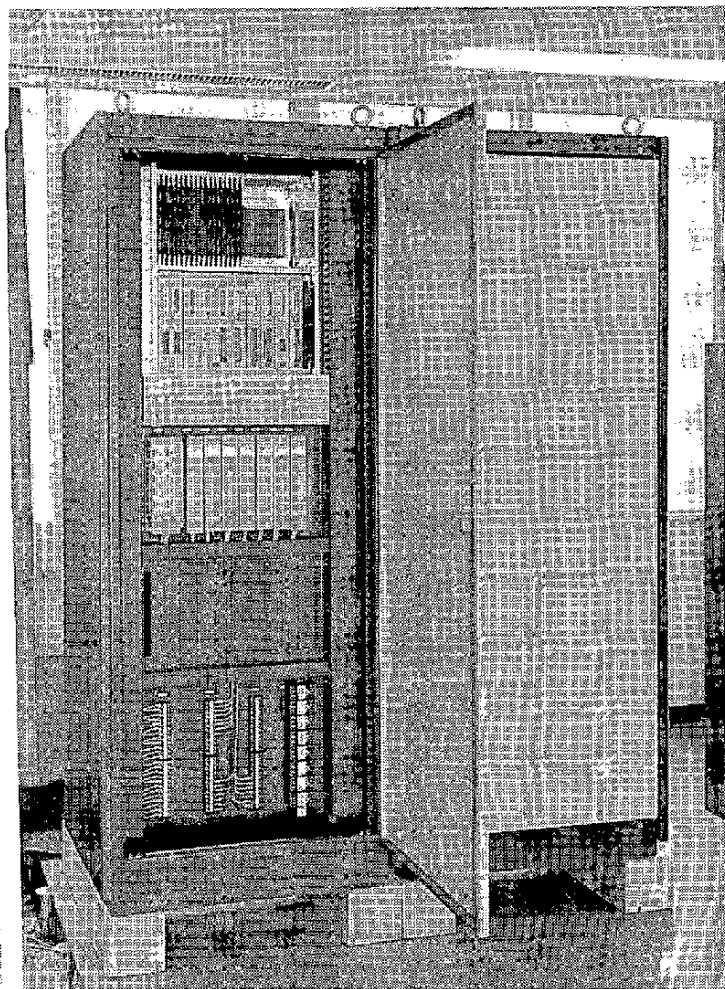
制御機能をそれぞれの基本的要素に分割して、それぞれに対応したプログラムを作っておき、アプリケーションエンジニアがマイクロプロセッサの機械語を意識せずアプリケーションシステムを作れるようにする方法の代表的なものに、翻訳型（POL 方式）とフィルインザーフォーム型（FIF 方式）とがあることを前項で記述した。

前者はある決まった制御要素で構成されるが、そのアプリケーション毎にその手順が変わるような制御システムに向けたパッケージの仕方である。

後者は、制御手順はほぼ決っているが、アプリケーション毎にその処理点数やスケールが変化するシステムに向いている。



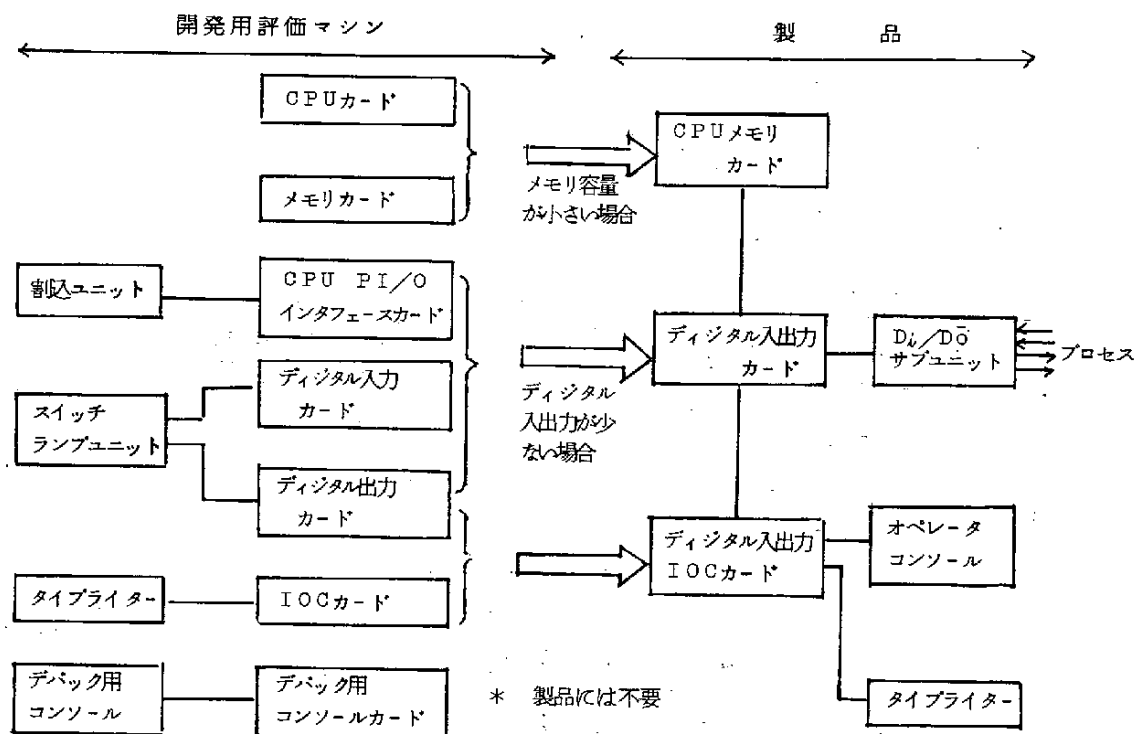
専用型ハードウェア例



組み合わせ型ハードウェア例

第4-27図 ハードの Fabrication の種類





第4-29図 開発用評価マシン製品の関係例

前者の代表例はシーケンサであり、後者の代表例がデータロガである。

#### (b) POL（翻訳型）方式の展開

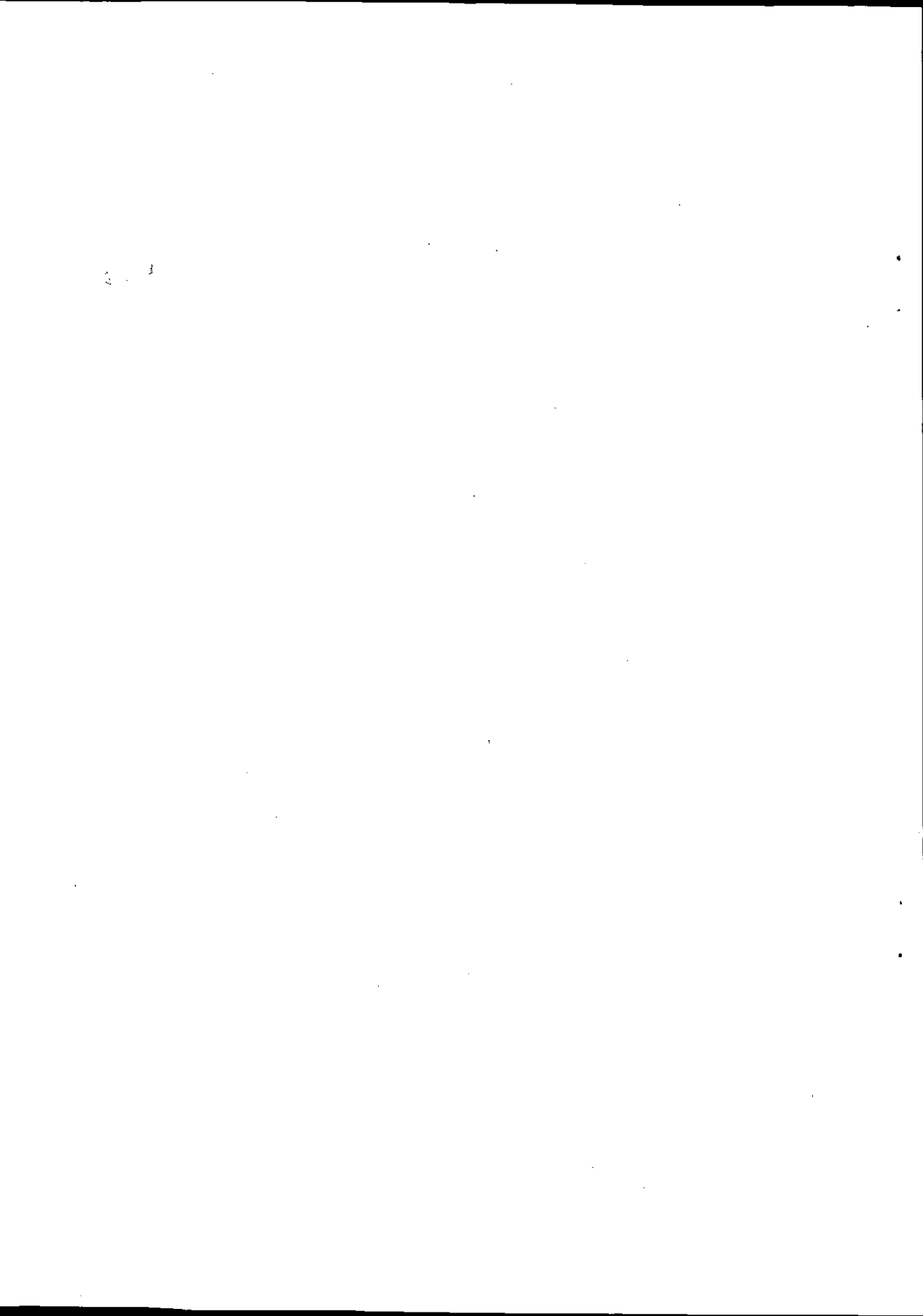
制御要素を基本的な作業に分解し、それぞれに対して一つのステートメント（命令）を与える方法で、アプリケーションエンジニアはこれらの命令を組み合わせてアプリケーションを作る方法であり、その例としてシーケンサの例を第4-9表に示した。ここではPOL方式の展開方法を簡単なシーケンサの例をとり示す。例えば、シーケンス言語として第4-10表に示す5種のステートメント（命令）を決める。基本的にはシーケンス制御は本ステートメントを組合

わせて実行できる。但しアプリケーションがしやすいよう実際には、第4-9表のような命令体系になる。今、接点 $\times 1$ 、と接点 $\times 2$ とのANDをとつて、その出力を $Y_1$ に出すシーケンスの場合、シーケンスを組む人はこの問題向言語を使つて $R, X_1 ; A, X_2 ; W, Y_1 ;$ と記述する。これを中間言語に変換し中間言語の形でマイクロコンピュータにセットされる。マイクロコンピュータのプログラムでは、この中間言語を一つ一つ実行していくのがインタプリタである。第4-30図にPOL、中間言語、インタプリタとの関係を示す。

| ステートメント    | 略号        | 例                                                                   |
|------------|-----------|---------------------------------------------------------------------|
| Read       | R         | $R, X_j$ 入力 $X_j$ の信号を読みとり演算レジスタにセットする                              |
| And        | A         | $A, X_k$ 入力 $X_k$ と演算レジスタの内容とAndをとりその結果を演算レジスタにセットする。               |
| $\bar{O}r$ | $\bar{O}$ | $\bar{O}, X_k$ 入力 $X_k$ と演算レジスタの内容と $\bar{O}r$ をとりその結果を演算レジスタにセットする |
| Not        | N         | N 演算レジスタの内容を反転して演算レジスタにセットする                                        |
| Write      | W         | $W, Y_k$ 演算レジスタの内容を出力 $Y_k$ へ出力する。                                  |

第4-10表

最も簡単なシーケンスのステートメント一覧表



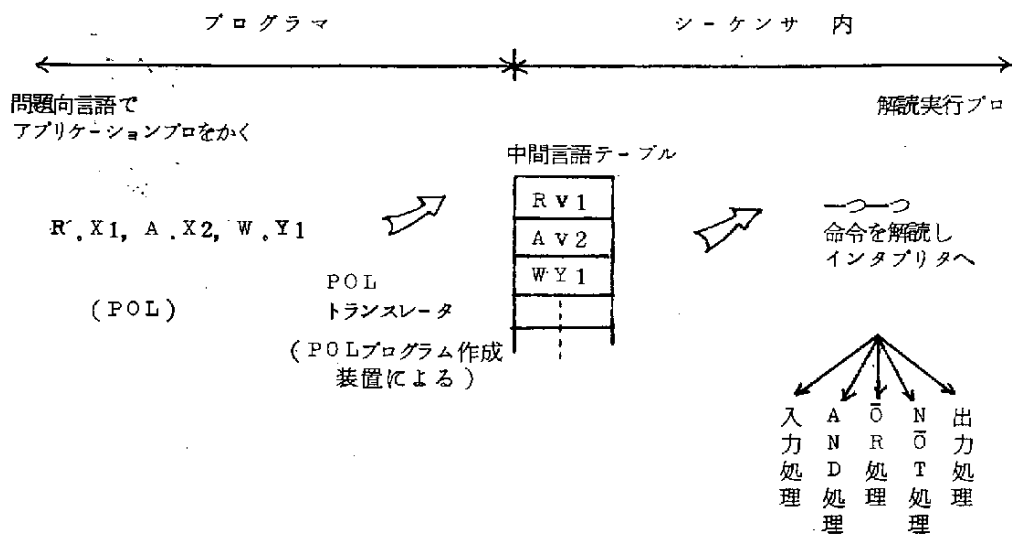
第4-31図 データログソフト構成例





[illegible]

第 4 - 3 2 図 データロガシステム仕様図書



第4-30図 POLとインタプリタとの関係

#### (c) FIF (ファイルインザフォーム) 方式の展開

制御手順及び要素がほぼ決まっており、アプリケーション毎にそれらの処理が必要か否か、又処理点数が増減するか等の変化がある制御システムのパッケージ化に適しているのがFIF方式である。データログの展開例を次に記述する。

データログはプロセス量(アナログ量、パルス、ON-OFF信号)を順次スキャンして読み、これらを常時入力処理するとともに上下限監視を行ない、上下限警報のときは、その瞬間にその時の時刻と全項目を印字する。これらのほかに、いわゆるSFプリントと称する操作機器の動作故障印字があり、この場合は操作機器に状態変化が起つた時刻とその内容を印字する。このようなデータログのプログラム例とそのデータテーブルの関係を第4-31図に示す。アナログ入力信号は、アナログ処理テーブルを参照してスケールファクター、リニアライズ、AD変換器の増巾器選択、単位、べき数選択を行なう。

(1)のテーブルはすべてコードで示されるので各テーブルに対応する値が仕様に応じて決められる。(2)は信号のベースとフルスケールとコードの関係を示す。入力信号はすべて工業値に変換されたものがファイルされる。又、設定値はデジタルスイッチによりセットされる。

第4-32図にデータロガの仕様伺書の一例を示す。

#### 4.6.4 マイクロプロセッサ応用の可能性とその適用限界

現在マイクロプロセッサが出現して数年がたち、制御用として使用できそうなものもかなり増してきた。そして制御への応用もますます増加してくるものと思われる。ここでその応用の可能性としてその適用限界について考察する。

マイクロプロセッサ応用製品が従来の制御用計算機やシーケンサ等の専用機の領分をかなり置きかえることが可能であろうが、やはりその限界があると思われる。すなわち、

##### (1) JOB の量

シングルJOBがマルチJOBか？

##### (2) システムの広がり

ソフトウェアのサポート及びプログラムライブラリーの大きさ

##### (3) 実行スピード

##### (4) コストと量

##### (5) トータルコストと信頼性

等を考慮するとおのずとその限界が明確となろう。ジョブが多く、マルチジョブを要求されるものは、パッケージ化が難かしく、一品料理的な性格を持たざるをえず、マイクロプロセッサを適用するのは不利である。但しマルチジョブでも機能分割が可能なものは、多数のパッケージ化されたマイクロプロセッサ応用製品を有機的に組み合わせ使用することが可能であろう。ハイラークシシステムに於て機能分割が可能が制御ジョブをマイクロプロセッサにやらせ、システム毎に変わる管理ジョブやスケジューリング等をミニコン以

上の計算機で処理するのが現時点では最適なものとなろう。又、データの処理量の多いもの、特に補助記憶を要するようなものも、パッケージ化がむずかしいと思われる。

処理スピードが問題となる場合も一つの限界であろう。但しスピードの早いプロセッサの出現やマイクロプログラミング技術が開放されることによつて、POLによるパッケージ化されたものがそのままマイクロプログラミングに移行して、スピードの早い制御機器となることは現時点で十分に考えられる。

従来製品の分野へマイクロプロセッサを応用する際には、上記の問題が限界となろうが新しいシステムへ適用する場合にもほぼ同様のことがいえる。

しかし、開発コスト、製品の仕様のリフレックス等考慮すると、ある程度処理スピードがおそくとも、ワイアードによる設計より、ストアードによる処理の方が有効である。

## 第5章 制御機能の将来動向

### 5.1 はじめに

これまでの章で、制御とは何か、制御技術とは何か、それらがどのような過程をたどって発達してきたか、マイクロプロセッサが登場して制御機能にどのようなインパクトを与え、また与えようとしているか、といったことについて、様々な角度から検討が加えられてきた。マイクロプロセッサの登場は比較的新しいことであり、特にその制御機器への応用になると、最近やっと軌道に乗り出したところであつて、本格化はこれからというところである。従つてこれまで述べられてきたことも、過去のことというよりはむしろこれからの方向を示唆したもの、という要素を多分に含んでいる。

ここで改めて制御機能の将来動向についての章を設けてみたが、将来の動向について正面きつたことを述べるのは、今日のように技術も社会も急激に変化しつつある時代には、至難のことである。だからあまり形式ばつたことを述べるつもりはないので、これまでの章に既に示唆されていたいくつかの事柄を締めくくりの意味でまとめ直してみた、とご理解願いたい。

### 5.2 システム化の意味とマイクロプロセッサ

制御機器を設計し、制御系を構成するのに、システム化の観点が重要であるとは、従来からもしばしば言われてきたことである。ここでマイクロプロセッサ利用の立場から、システム化の意味を考え直してみたいと思う。

システムの発想とは何かという問いには、それこそ百人百様の答がでてくるのであるが、一つの答え方として、「制御系なら制御系を設計するとき、制御対象の内容、特性に応じて個々の設計法から出発するのではなく、制御の目的をまず明確にし、それに応じて系全体の構成法、使用する要素、部分を技術的、経済的立場から、トップダウン式に選定、決定するための一般的方法論を基礎にする立場である。」とすることができよう。すなわち個々の対象に特有な計画手段なり、設計思想はあるにしても、それを第一の出発点にするのではなしに、「目的を実現する組織体」に対す

る一般の方法論をまず考えるというやり方である。それによつて計画や設計の思想立脚点をはつきりさせ、伝統や因襲にとらわれなくて手段を選択し、さらに設計上の問題点やボトル・ネックを浮かび上がらせることができる。

しかし残念ながら、今日のところどのような対象にも適用する万能のシステムの設計法は確立されていない。いくつかの数理的方法（LP、NLP、IP、DPなどの数理計画、PE、RT、CPMなどのネットワークの手法、待ち行列の理論、信頼性の理論、効用関数など決定理論、ARMA、GMDH、DEMA TELなどの予想手法、多変量解析の理論、シミュレーションの技法、その他）は、それぞれ特定の問題解決に明確な指針を与える強力なツールであるが、それらを駆使して実際問題にいつでも立ち向えるような体系化は未完である。またそれらソフトの手法を用いただけで、ハードの問題まで深く立ち入って検討できるものではない。従つて、システム的手法は、それと裏腹に、次のような手段の必要性を浮かび上がらせることになる。

- (1) 上記の意味でのシステム設計法はいわば抽象的な、ソフトの構成法であつて、実システムを対象とするには、言うまでもなくハードの裏付けが必要である。そしてシステムの発想の筋道を逆にたどつて、部品、要素、サブシステム、システムという統合が実現可能であるか否か、ボトム・アップ式にチェックしてみる必要がある。

ハードの手段、機器は、対象とする系の種類によつて異り、多様化するのはやむ得ない。将来におけるNeedの多様化は、部品、要素の多様化の傾向をますます強める力を持つのはまた必然である。しかし、ハードの多様化が拡がる一方ならば、そもそもシステムの発想は危機に頻することになる。ソフトの段階でシステム的に出発しても、ハードの段階でそれぞれの系に個有の選択、設計の問題が発生し、それに対処するために個有の専門家を必要とし、結局は一般論の利点が失われてしまうことになるであろう。

従つて、ハードの段階で部品や要素の信頼性を極力上げること、部品の数をなるべく減らすこと、部品や要素を機能モジュール化・標準化し、

代替性の高いものにすること、などが要求される。それらの要求に応えるためには空間的には部品の高密度集積化をはかり、また時間的には時分割多重使用する方法をとるのが有効である。

これらの要求項目を見れば、マイクロプロセッサはシステム化のために、まさに最適の機能部品或いは機能要素であると言うことができよう。ここで強調しておきたいことは、マイクロプロセッサは部品として見たときに、小形軽量、低価格であるばかりでなく、信頼性が高く、多様な要求にソフトに、フレキシブルに対応できる優れた特性を具えている、という点である。部品としての利点を最大限に生かすためには、将来充分な標準化がはからなければならない。すなわち、ビット数、ピン数、ピンの配列順序等、国際的な規格化がなければ宝のもちぐされになりかねない。

- (2) 大きな系を全体として一つのシステムと考えるとき、そこでの全ての情報を集中的に処理することは良い方策ではない。各種の端末から入力される多量の生データを直接中央情報処理装置に伝送するとすれば、膨大な伝送線路を設備しなければならず、必然的に伝送誤りの発生する率は高くなる。また情報処理装置の一部における故障が、系全体のダウンにつながる可能性が高くなる。

その難点に対処するために、情報処理機能或いはインテリジェンスの分散が有効である。

ローカルなインテリジェンスを、しかも階層構造を持たせて配置するならば、入力される生データを加工して圧縮された情報を逐次上の階層のインテリジェンスに伝えることができ、それによって伝送線路の容量を、相当量減らすことができる。そればかりでなく、故障をローカルに検出して系の他の部分に波及するのを防ぐこともできる。

このようなインテリジェンスの分散による系の分割と階層構成は、大形の制御システムの設計において欠くことのできない基本的な思想となりつつあり、そこでマイクロプロセッサやICメモリが情報処理装置或いはコンピューターとして持つ機能が欠くべからざるツールとなると断言できるのである。

以上のように、システムの発想の重要性と、それをハードの面から裏付けるための手段としてのマイクロプロセッサの重要性をみてきたのであるが、次の節で制御情報処理機構の複合化、階層化と、さらにそれと関連してシステムにおけるマン・マシン協同作業の意味、手段を、もう少し立ち入って考えてみよう。

### 5.3 システムの分割と複合制御系

大きなシステムを取り扱うとき、それをいくつかのサブシステムに分割することは自然で、かつ必然的な発想である。それは単に大きなシステムを1つのコンピューターや制御機器にのせることができないというような、技術的限界によるばかりではなく、われわれ人間のシステムをとらえる認識の能力や構造の問題と交わり合っていると考えるべきである。大きなシステムには、単に大くの物や量やパラメータが存在するだけではなく、異種のサブシステムや要素が組み合わさっているという特徴がある。それら異種のサブシステム、すなわち異つた概念でとらえられる複数個の対象を認識し思考するとき、概念体系の構成に「分類」が持ち込まれ、その上に「分割」が成立する。そこに現実の技術の問題として計測 通信伝送、情報処理、制御のハードウェアの量的限界が重なってくる。

大きなシステムの取り扱いで、もう一つ注意しておかなければならないのは、システムのマクロ的把握、或いはシステム要素のアグリゲーションということである。これはいくつかのシステム要素を束ねて把握することであつて、例えば個々の製造ラインに細かく注目せず何本かまとめて制御の対象とするとか、1つの工場全体の生産性を単位に考えると、国家経済を数十個の変数で表わしてその最適制御を考えると、かである。

システムの分割とマクロ化から導かれる構造の一つが、階層構造である。

階層システムに特徴的なことは次のことである。

- (1) 各サブシステムは一応閉じたシステムであり、独自の構造、機能、目標、情報機構及び決定機構を持っている。
- (2) 下位レベルの決定機構は、固有の目標、限定された情報、及び上位からの統合指令に基づいて決定を行い、上位レベルの決定機構は、合目的に下位を統合(協調)するように、決定を行なう。各サブシステム、各レベルの目標に適当な整合性、柔軟性が無ければ統合不可能となり、従つてシステムは成立しない。
- (3) 下位の目標、決定は上位のそれに強く影響される。一方、下位の目標は設定或いは把握し易いが、それらに多様性、矛盾が存在すれば、上位の目標は一義的に設定し難い。また全体の目標をアプリアリに設定できず、下位からのフィードバック情報により逐次修正しつつ決定すること



が必要な場合もある。

- (4) サブシステム間の代替はききにくく、またどのサブシステムが欠落しても、全体システムは完結しない。
- (5) サブシステム間の連関が数量的に定式化しにくく、また数量的のみに  
は扱えられない場合がある。

もう一つの構造として、分散構造がある。これは必ずしも、上位、下位のレベル分けをせず、分散的に配置された複数個の方策決定者、制御者が個有の情報構造に基づいて決定、制御を行なうもので、限定された固有の情報の他、決定者相互間の情報交換がある程度可能であるとする。その場合、情報構造の不完全さと制御方策の不完全さとはトレード・オフの関係にあり、実用性、経済性などを考慮して両者の適当な妥協をはかることになる。例えば限られた比較的狭い地域内の交通情報を基にして、交差点の信号制御を行なうような場合を考えればよい。

以上に述べたような階層構成或いは分散構造を持つ制御系は、既にいくつものプラントやネットワークの制御に用いられている。しかし、階層制御において如何にシステム分割を行なうのがよいか、何レベルにするのがよいか、統合の方策はどうあるべきか、また分散制御においていくつの制御装置に分けるのがよいか、情報構造や情報配分は如何にすべきか、情報と制御を含めた全体としての最適設計はどうなるか、などといった問題には明確な解答が得られていない。制御装置やそのフィールド・インタフェース、マン・マシン・インタフェース、上位コンピューター・インタフェースなどを考えるとき、制御装置は経済性や信頼性から容量を決める、フィールド・インタフェースは制御対象の所在によつて分割する、マン・マシン・インタフェースは操作の機能によつて分割する、などいわば直観的な方策がとられているのが現状である。

プロセス制御で階層制御が行なわれているのは第1章でも述べた通りであるが、最近では大規模なシーケンス制御の設計をサブシーケンスに分割して行ない、コントローラによつてサブシーケンス・プログラムの合成を行なう方策も考えられている。そのときの分割は、機能単位ごとにまとめる、論理的結合の密なものをまとめる、シーケンスの物流に注

目してまとめる、試験や診断が行いやすい単位にまとめる、などを基準にする。

将来、マイクロプロセッサを用いたマルチプロセッサ・システム、ポリプロセッサ・システムが一般のものとなると、アナログ機器や人間とのインタフェースをも含めて、明確な設計思想と設計基準を獲得するよう努力しなければならないであろう。

#### 5.4 ロボット—人間的機能の機械化

今日技術の世界で言われるロボットとは、「大なり小なり、人間的な機能をもった機械」のことである。大別して、工業の生産工程において省力化などのために直接役立てる目的で開発されている工業用ロボットと、より基礎的に人間の運動機能や感覚、感覚情報の知能的処理機能の解明やシミュレーションのために考究されている研究用ロボット（或いは知能ロボット）とがある。

工業用ロボットは我国においても1960年代半ばから開発・生産され、高度成長経済のもと労働力の不足の折柄、種々の作業工程に取り入れられて、かなり順調な経過をたどって発展してきた。しかし今日、経済環境が変化しつつあることにもよるが、技術的にも一つの反省期にさしかかった感がある。ロボット—人間の機能の代行—汎用機械、というイメージに対して、現在の工業用ロボットは性能的にも価格的にも、まだまだ一般ユーザにとって充分満足のいくものにはなっていないようである。

一般的な見方をすれば、ロボットというより、プログラマブルで多少の柔軟性はあるがそれほど広範な要求には対応できず、機構的にも人間や高等動物の手足の動作と比べると極めて簡単であり、マニピュレータに過ぎない。触覚や視覚に相当する感覚器管も全く欠如しているか、或いは不十分なものであつて、従つて触覚で物体の位置を検出して正しくつかむとか、やわらかさを検出して握力を変えとかの機能は無い。視覚の方も無いから、物体の位置の検出、形状や色による物体の識別判定ができない。すなわち感覚器管による信号の検出、センサ・フィードバックが無いために、微妙な手の動作ができず、また新しい環境に適応して動作を変えとか、

経験を積み重ねて学習効果を発揮するとかいうこともできない。

このように不十分な機能を補い、また改良する手段として2つの方向が考えられる。1つは単能的なロボット、或いは標準仕様の機能モジュールに徹する考え方である。つまり機能モジュールを部品として量産し、それらを系統的に組み合わせる、或いは生産工程の各所に分散して配置しようという考え方である。そのときは、制御装置も簡単なオン・オフ形のもので済ませるようにする。

もう1つの方向は単体として高度な機能を持ったロボットを目指すものである。その場合はインテリジェンス・ロボットの方向であり、例えば目、手、知能を充実させて物体認識、パターン計測を行なわせる。その他、手及び指の自由度を増すこと、運動の精度及び分解能を増すこと、視覚、触覚等の複合感覚による認識機能を持たせること、適応、学習制御の機能を持たせること、などが目標になる。

いわば中途半端な形態から、上記の2つの方向のいずれかに分極していくのが今後の動向であろう。そのいずれの極においても、今までしばしば見てきたマイクロプロセッサの特徴を、最大限に生かす場が提供されることになるであろう。

既に述べたように、機能の高度化のためにはセンサ及びセンサ・フィードバックが1つの重要な鍵になるように思われる。ロボットのセンサは、外部からの信号を検出・観測すること、内部状態量の監視、及び人間や他のロボットとの信号の授受をしなければならない。われわれ人間が何らかの作業を開始する前を考えると、作業に取りかかる前にその計画を立てる。その計画は作業の種類、自分の置かれている環境（外部状態）、それに自分の状態、能力（内部状態）によつて、いろいろ違ったものを考えなければならないはずである。それらの条件から1つの作業モデルを想定し、それに従つて実際の作業を開始することになる。作業を始めた後でも、外部環境の変化等によつて作業の進捗に狂いが生じたり、想定外のトラブルが発生すればもう一度観測をやり直して作業モデルを変更するのが人間のやり方である。そうすることによつて異なつた種類の作業や外部環境に自ら適応することができる。

そういう適応能力が無いロボットでは、要するに教育されたことしか行なえない。自ら経験を記憶に蓄えて、適応をより速く円滑に行なう能力も持ち得ない。人間や動物の観測では、単に入力信号をそのまま時系列的に受け入れるだけではない。多量の情報の中から、現在必要な情報を抽出して、それをパターンとして整理する能力がある。それによつて、例えば極めて多量の情報量を持つ画像入力を処理し、有効に利用することができている。現在パターン識別、パターン情報抽出の研究が盛んに行なわれているが、人間や動物の機能の核心にまではまだだいぶ距離があるようである。

動物のセンサには、今日の技術界で用いられているセンサと可成り違った、巧妙な仕組みがあると言われている。例えば、皮膚の表面には極めて多くの、また極めて感度の高い触覚のセンサ受容器神経細胞が配列されていて、それぞれがA/D変換器となっており、出力はアナログ入力信号、すなわち刺戟の強さによつて周波数変調されたパルス列である。刺戟が強いとパルスを送る神経繊維の数も多くなる。つまり多チャンネル伝送になるわけであるが、これも単にパルス列をそのまま中枢に送り込むのではなく、適当な前処理が行なわれる。例えば神経細胞は刺戟の時間的的平均値があるレベルより小さいと出力パルスを出さないというしきい値特性を持ち、また横方向につながった神経繊維からの入力に神経細胞の出力を抑制するという働きを持っている。これによつて刺戟の空間的变化の特徴を抽出するという重要な前処理機能を果たしている。

その他パルスが神経繊維を伝わる間に、その形状が一定に整えられる性質が具わつていたり、また刺戟が時間的に増加するときに出力を出す細胞と減少するときに出力を出す細胞を具えていて、刺戟の時間的变化の特徴抽出を行なう働きをしたりもする。

その他、限られた数のセンサ素子がある部位には密に配置し、他の部位には疎にばらまいて効率よく分解能を高めるようになっていること、感度の異なるセンサを組み合わせでダイナミック・レンジを広げていることなど、いわば何億年にもわたる実験を積み重ねて、巧妙な感覚受容器とその情報処理のシステムを完成させているのである。こういったセンサによるマイナ・フィードバックが大腦中枢を経由するメージャ・フィードバック

と何重にも重なりあつて、パターン認識、適応、学習といった高度な人間の能力を形成しているのであろう。それを技術的に実現することは、材料的にもシステムのにも、まだまだ夢物語としか言いようがないが、センサやプロセッサの集積度が高まるにつれて、そのような夢の実現の展望が僅かずつ開けてくるものと思われる。

## 5.5 マン・マシン・コオペレーション

最近騒音やら故障の多発やらでだいぶ評判を落しているが、1964年に営業運転を開始した新幹線は未だ一度も致命的な事故を起したこともなく、我国技術の一つのレベルを示すものとして評価されてきた。新幹線のような高速列車運転では運転者という人間の操作機能だけに頼ることは危険であるとして、列車運行・運転についての主な指令はすべて中央管制室から現場に与えられている。運転者は発進、加速、減速、停止などの指令に忠実に従つて操作器を動かしておればよい。それならばいつそのこと運転者をスキップして、指令によつて直接操作することまでできるかという、信頼性に関するシビアな要求を除けば可能である。しかしそこまで機械化した運動システムにせずに、運転者を介在させたことに成功の大きな原因があると言われている。

航空機の着陸操作についても同じようなことが言えよう。管制塔からの指令に従い、地上からの誘導電波の助けを借りて、操縦桿を動かすのはパイロットである。もちろん時に失敗があつて事故を起しているけれども、完全無人の自動着陸制御装置は未だ実験の段階を出ていない。全面的に無人化することは恐らく近い将来には行なわれないうであらう。

一方、石油化学の工場などで、近来よく大きな爆発事故などが起きる。プラントの自動化が進んで、モニタリングなどの要員が少数居るだけのところもふえている。また鉄鋼のプラント制御装置を収めて運転開始したら、その後しばらくして異常が検出された。現場にはユーザの人しか居なくて、メーカーの担当者は土曜日の夜中に電話で叩き起され、千キロ以上も離れた現場へとんで行ったら、制御装置のトラブルではなくて、プラント自体のちよつとした異常だつたというような話がある。

上に述べたのは、ある意味で典型的な例なのであるが、それらを分析してみると人間と機械の関わりについての意味が分ってくるのではなからうか。新幹線や航空機の運転の例は、マン・マシン・コオペレーションの成功例である。機械の持っている長所は、定められた手続きによる状態観測の正確さ、情報処理の正確さ、高速さ、故障しない限り疲れを知らず長時間にわたって動作する安定性、などである。さらに必要に応じて観測量をふやしたり、情報処理の仕方を変えたり、改良を加えることができるのも長所である。一方人間の方は以上のような能力では機械に遙かに劣るが、状況を統合的に素速く認識する能力、予期以外の異状を検出する能力、判断する能力、適応、学習する能力、などでは断然優れている。ちょうどこの2つの長所が相補的に組み合わさって、高度な制御系を構成し得ているのである。

プラントのトラブルの方は失敗例ということになるのであるが、何事故にまで立ち至るのか。原因はケース・バイ・ケースであり、また複合的なものであるだろうが、根本的にはマン・マシン・コオペレーションの失敗と言えよう。プラントの監視装置や制御装置はあらかじめ想定された動作状態、動作範囲があつて、それ以外の状態を検出したり、検出しても応答できなかつたりすることがある。その場合フェイルセーフが働いて自動停止するか、アラームを発して人間に告げるようになっていた筈であるが、設計上そうならないという落とし穴がある。またアラームを発したとしても、周囲の人間がそれに対して適切な処置を取ってくればよいが、どう対処したらよいのかが分からず、まごまごしているうちに可操作の限界を越えてしまうことがある。

失敗例における最も深刻な問題は、システムが機械も人間も含めて適応能力に欠た、或いは学習によつて適応能力を高めていくことに失敗したという点である。従来形の機械系にはもともと適応能力がない。(将来は適応形にしていくのが1つのポイントであることは前節にも述べた。)それを人間がうまく補つて、マシンを使いながらマンがちゃんと学習する仕組みになつておれば良いのであるが、マンが単なるモニタとして働いているだけではコオペレーションの利点は出て来ないのである。

将来、マン・マシン・コオペレーションを積極的に生かしていかなければならないが、そのとき両者がそれぞれどれだけの機能を分担するのかという問題と、両者のインタフェースをどうするかという問題とが大切である。後者の問題はさらに、マンからマシンへの入力と逆の出力の問題とに分かれる。

インタフェースについては第1章のインテリジェント・ターミナルの節で触れた。もう少し補足しておく、マンからマシンへの入力はなるべく手間のかからないもの、平たく言えば人間どろしが情報交換している形態と同じか、それに近いものが望ましい。機械語でコーディングし、紙テープにせん孔して入力するというのが初期の形であるが、ソフト的には機械語からアセンブリ言語へ、アセンブリ語からコンパイラ言語へと発達して、人間の負担を減らすようになってきた。紙テープ或いはカードの読み取り機構はまだあまり変わっていないようであるが、通信回路を介してTTSでリモート操作したりする便宜は増してきた。それを将来、自然言語での命令読み取り、自然言語での音声入力という形にまでもって行くことが理想である。

出力の方はプリンタで多量のデータをはき出したり、大きなグラフィック・パネルに表示するやり方から、CRTディスプレイでチェックし、必要なものだけを数表やグラフの形でハードコピーして取り出す方向へ変わってきた。CRTディスプレイのインテリジェント化は、最近マイクロプロセッサを組み込んで盛んに行なわれるようになってきている。インテリジェント化の目的は、データを整理された形(データの階層化も含めて)で表示すること、グラフその他パターンとして表示すること、フラッシングや色つけによって注意すべき点を強調したり、アラームを発したりすること、表示が人間にとって見やすいものにすること、などである。画像は人間の視覚に伝え得る情報量が、他の手段に比べて格段に多いから最も重視されるが、その他音声言語も活用する必要がある。

要するにマン・マシン・コオペレーションでは両者間の情報のやりとりが手軽にしかも効率よく行なわれることと、マンがシステムの状態を自由に知り得て、自ら学習する機械を得やすくすることが大切である。そのた

めにマシン側にマンを適度に助けるインテリジェンスを配置することが必要なのである。

## 5.6 おわりに

以上では少しばかり勝手な私見を述べさせて頂いた。数行でも参考になるところがあれば幸いである。遠い将来はともかく近い将来では、マイクロプロセッサそのものの発達動向をにらみながら、地道な使い方を考えていくのが正道であろう。DDCレベルのアルゴリズムを洗い直してみる、制御対象のオフ・ライン同定からオン・ライン同定のアルゴリズムへ発展させてみる、測定器にちよつとしたインテリジェンスを持たせる、ミニコンの機能を分散させてみる、アナログを部分的にデジタル化してみる、等々である。現場からの報告を聞かせて頂きながら、筆者ももう少しまとまった将来動向の予測を考えてみたいと思っている。

おわり





— 禁 無 断 転 載 —

マイクロ・プロセッサ

— 制御への応用編 —

発行日 昭和51年3月

編集・発行 (財)日本情報開発協会  
東京都千代田区霞ヶ関3-2-5  
霞ヶ関ビル30階

〒100 電話(03)581-6401(代)

印刷 エイワ印刷社

東京都千代田区内神田1-11-7

〒101 電話(03)291-7464(代)

|             |           |            |              |          |  |
|-------------|-----------|------------|--------------|----------|--|
| 請求<br>番号    |           | 日<br>50-18 |              | 登録<br>番号 |  |
| 著者名         |           |            |              |          |  |
| 書 名 マイノリティー |           |            |              |          |  |
| 所 属         | 帯 出 者 氏 名 | 貸出日        | 返 却<br>子 定 日 | 返却日      |  |
|             |           |            |              |          |  |
|             |           |            |              |          |  |
|             |           |            |              |          |  |

