

55-S003

昭和55年度
マイクロコンピュータ応用に関する
委託開発報告書

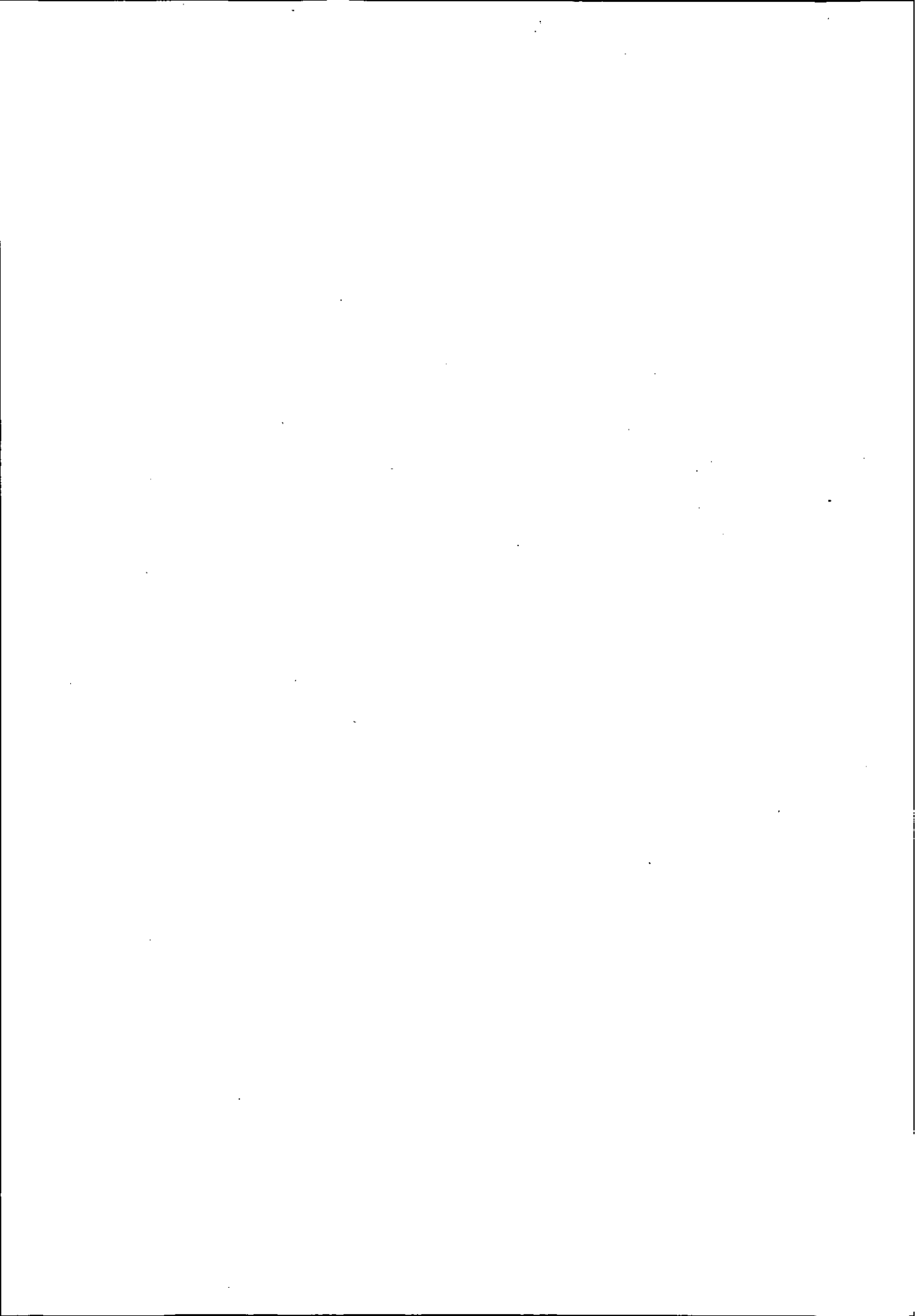
昭和56年3月

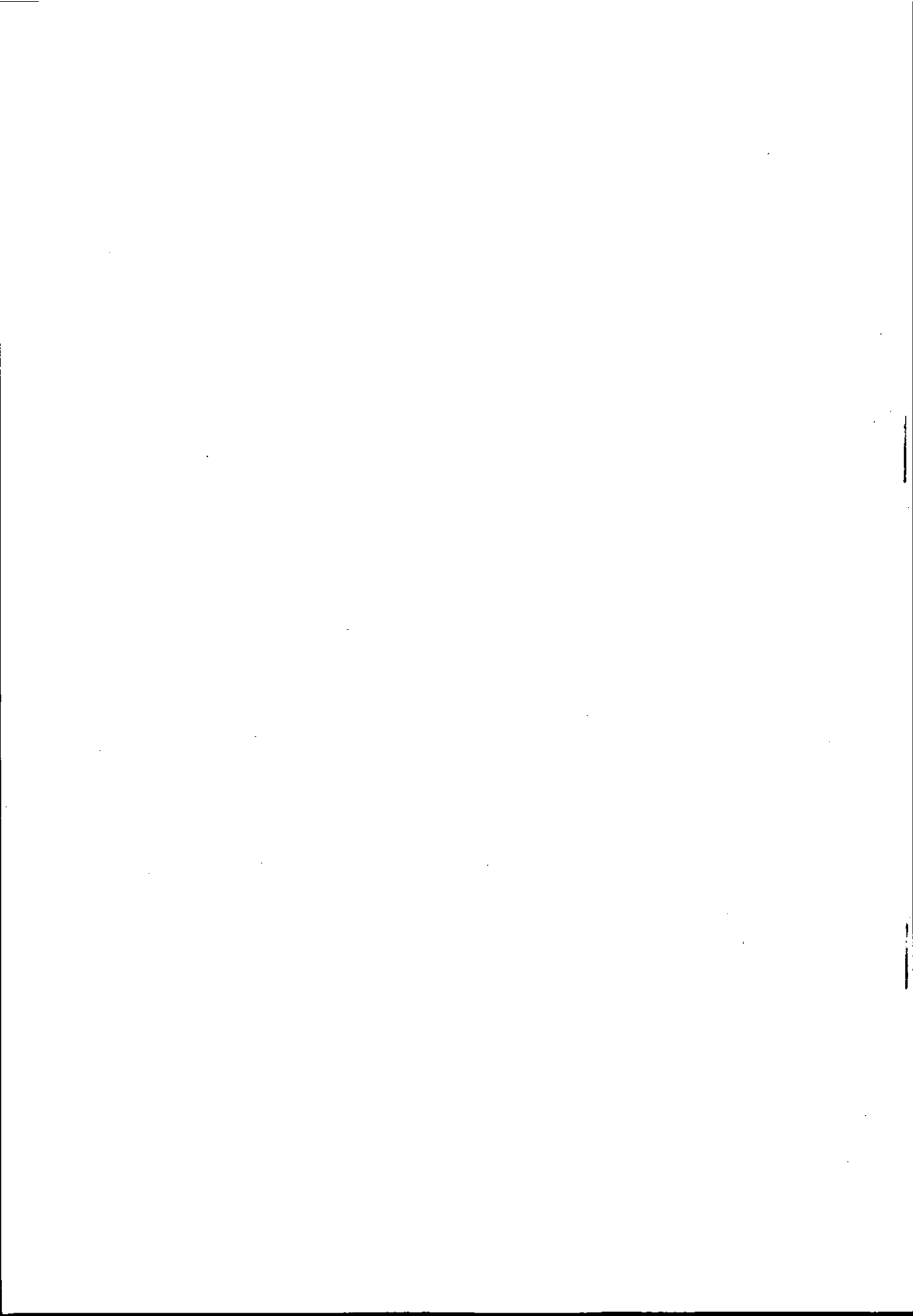
JIPDEC

財団法人 日本情報処理開発協会



この報告書は、日本自転車振興会から競輪収益の一部である機械工業振興資金の補助を受けて、昭和55年度に実施した「マイクロコンピュータの応用に関する調査研究」の一環としてとりまとめたものであります。





序

マイクロコンピュータがあらゆる分野に普及拡大を続けている現在、ハードウェアの開発についてはめざましいものがある反面、ソフトウェアについては必ずしも満足すべき状態でないのが現状であります。

これらの現状に対処するためには、システムの設計・製造を担当し、特にソフトウェアの開発にウエイトを置くシステムハウスの技術力の向上、財政基盤等を助成し、育成させていくことが我が国のマイクロコンピュータ産業における大きな課題の1つであると思われまます。

当協会では、この課題を達成するために昭和53年度より「振興助成のための事業委託制度」を設け、システムハウスを主対象にマイクロコンピュータの応用に関する共通的・基礎的技術の委託開発を実施いたしております。

本報告書は、当協会が本年度委託開発した「パフォーマンス測定器」、「4ビットマイコン用開発サポートシステム」、「8080-6800 双方向ソースプログラムコンバータ」、「マイクロコンピュータにおけるPASCALコンパイラ」の4システムの概要を説明したものであります。

これらの開発システムが広く皆様方に利用され、応用システムの開発等に寄与できれば幸甚に存じます。

終りに本開発にご尽力・ご協力下さいました各位に対し、深く謝意を表わす次第であります。

昭和56年3月

財団法人 日本情報処理開発協会

会長 上野幸七

マイクロコンピュータプロジェクト委員会

(敬称略, アイウエオ順)

委員長	田村 浩一郎	電子技術総合研究所 制御部論理システム研究室長
委員	河合 勝司	名古屋市工業研究所 電子部情報技術課技師
"	北川 愛子	日本電信電話公社 横須賀電気通信研究所研究主任
"	出口 光一郎	東京大学工学部 計数工学科助手
"	寺田 浩詔	大阪大学工学部 電子工学科教授
"	前田 英明	ソフトウェアコンサルタント
"	山上 喜吉	電子技術総合研究所 パターン情報部バイオニクス研究室
オブザーバ	梅村 勁樹	通商産業省機械情報産業局 情報処理振興課
	稲積 義登	通商産業省機械情報産業局 情報処理振興課
	勝山 治夫	通商産業省機械情報産業局 電子政策課
	長岡 久人	通商産業省機械情報産業局 電子政策課
	佐藤 昌彦	通商産業省機械情報産業局 電子政策課

マイクロコンピュータプロジェクト
委員会小委員会

委員	田村 浩一郎	電子技術総合研究所 制御部論理システム研究室長
"	山上 喜吉	電子技術総合研究所 パターン情報部バイオニクス研究室
"	前田 英明	ソフトウェアコンサルタント
"	出口 光一郎	東京大学工学部計数工学科助手
"	北川 愛子	日本電信電話公社 横須賀電気通信研究所研究主任
"	寺田 浩詔	大阪大学工学部電子工学科教授
"	島崎 真昭	京都大学工学部情報工学科助教授
"	河田 亨	大阪大学工学部電子工学科講師
"	河合 勝司	名古屋市工業研究所 電子部情報技術課技師
"	松井 信行	名古屋工業大学電気工学科助教授
"	丹羽 敏行	名古屋工業大学情報工学科助手

昭和55年度委託開発の経過

- 第1回委員会 [昭和55年6月]
 - ① 委託開発テーマの選定
 - ② 委託開発スケジュールの確定

- 第1回小委員会 [東京地区：6月][大阪地区：6月][名古屋地区：6月]
 - ① 委託開発の目的・条件についての検討
 - ② 委託開発仕様の検討

- 第2回委員会 [昭和55年7月]
 - ① 委託開発の目的・条件の決定
 - ② 委託開発仕様の決定
 - ③ 公募要領の検討・作成

- 公募要領の発送 [昭和55年7月]

- 公募締切 [昭和55年8月]

- 第2回小委員会 [東京地区：8月][大阪地区：8月][名古屋地区：8月]
 - ① 応募先と提案書についてヒアリング
 - ② 応募先の技術力についてヒアリング

- 第3回委員会 [昭和55年8月]
 - ① 応募先総合評価
 - ② 委託先の決定

- 委託契約 [昭和55年8月]
 - ① パフォーマンス測定器 : 日本システム技術㈱
 - ② 4ビットマイコン用開発サポートシステム : コアデジタル㈱
 - ③ 8080-6800双方向ソースプログラムコンバータ : 東海ソフト㈱
 - ④ マイクロコンピュータにおけるPASCALコンパイラ : 日本システム開発㈱

- 第3回小委員会 [東京地区：10月][大阪地区：10月][名古屋地区：10月]
基本システム設計について委託先からヒアリング

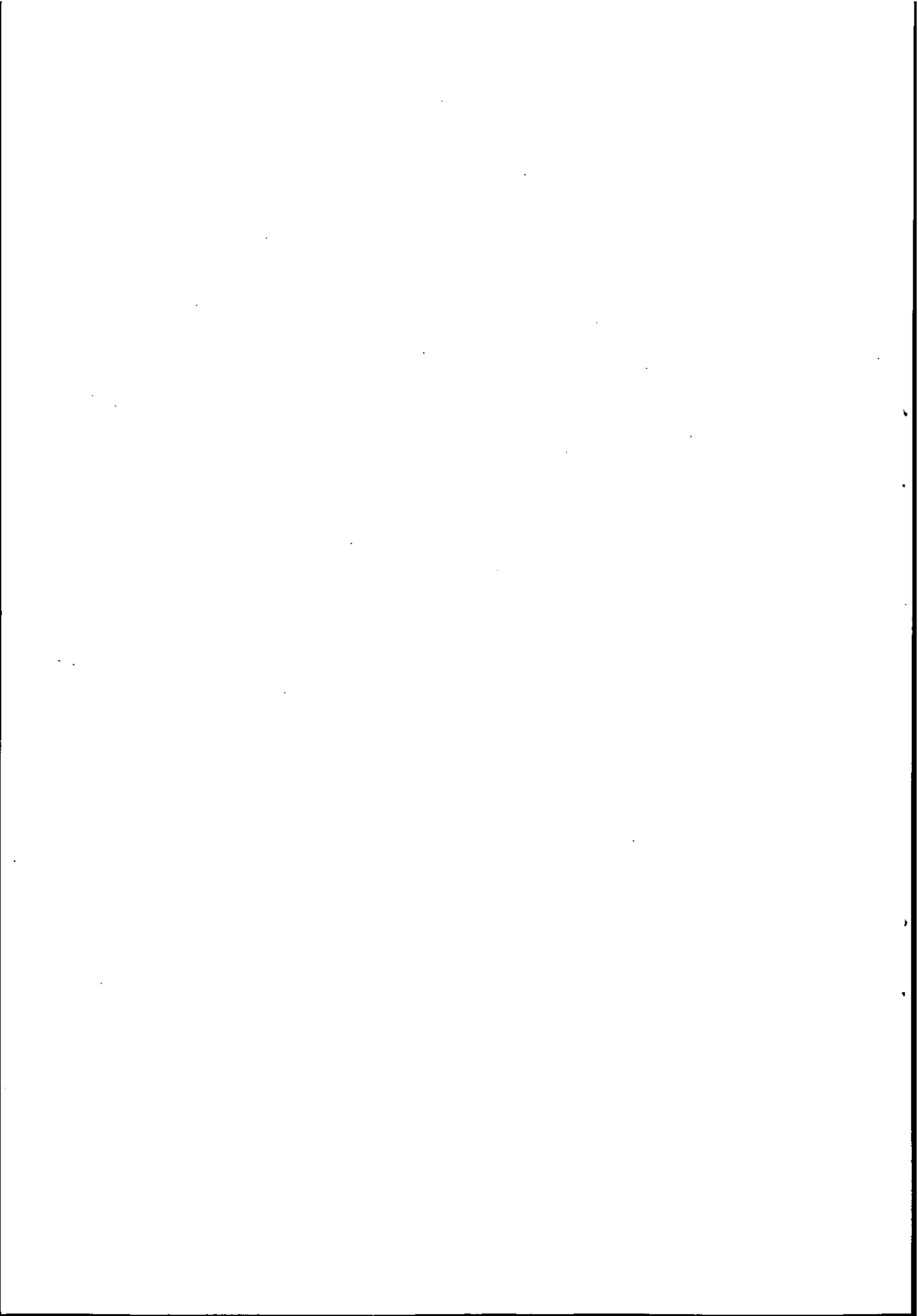
- 第4回委員会 [昭和55年10月]
基本システム設計の総合評価

- 委託開発の検査
各テーマについて、それぞれ3回検査を行った。

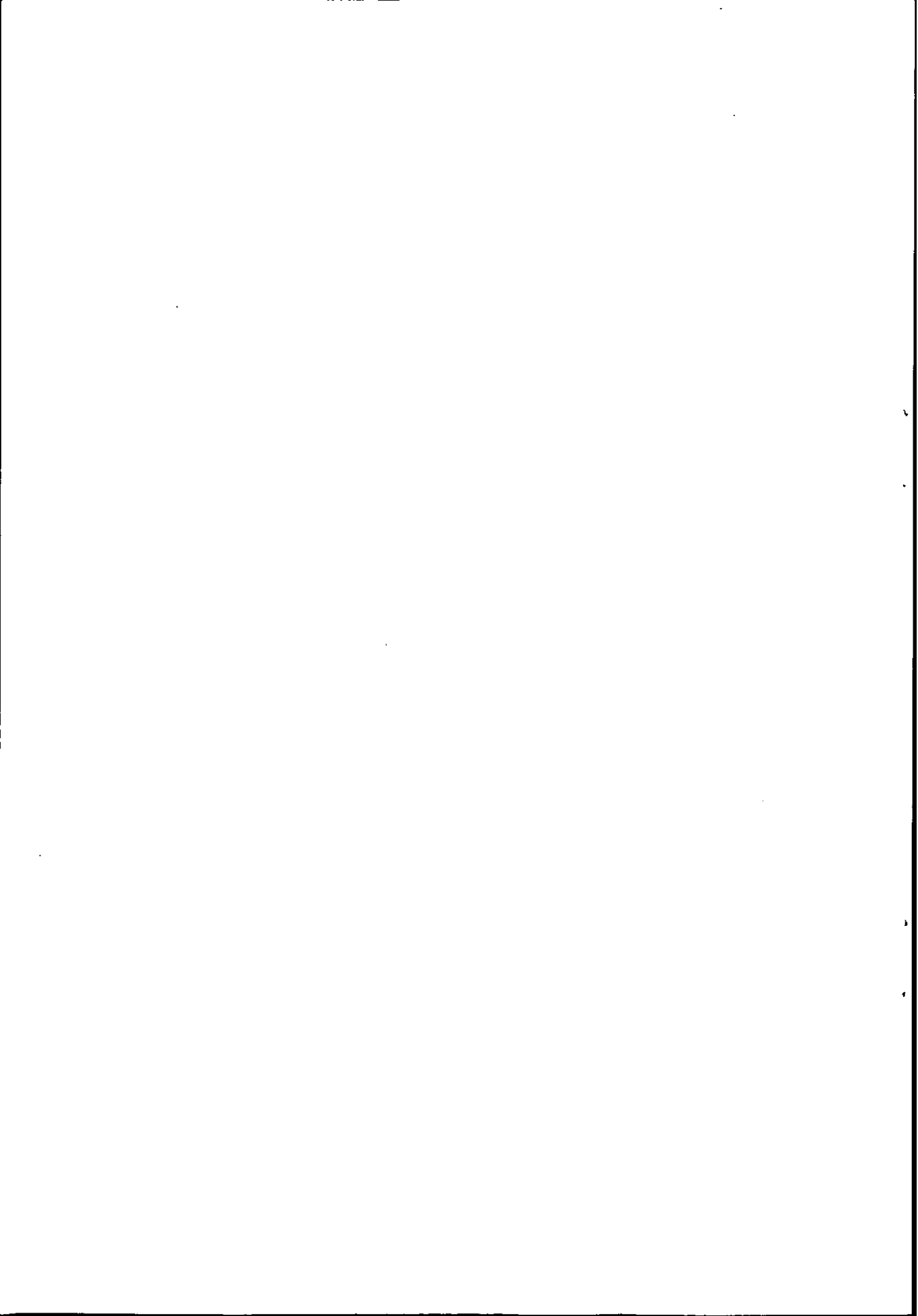
- 第5回委員会 [昭和56年3月]
 - ① 検査員による検査報告
 - ② 納入物件の評価

目 次

第1章 パフォーマンス測定器	1
1.1 装置の名称	1
1.2 装置製作の目的	1
1.3 装置の構成	1
1.3.1 ハードウェア構成	1
1.3.2 ソフトウェア構成	4
1.4 装置の機能	5
1.5 評 価	5
第2章 4ビットマイコン用開発サポートシステム	9
2.1 システムの名称	9
2.2 システム製作の目的	9
2.3 システムの構成	9
2.3.1 ハードウェア構成	9
2.3.2 ソフトウェア構成	11
2.4 システムの機能	11
2.5 評 価	13
第3章 8080-6800双方向ソースプログラムコンバータ	15
3.1 システムの名称	15
3.2 システム製作の目的	15
3.3 システムの構成	15
3.3.1 ハードウェア構成	15
3.3.2 ソフトウェア構成	18
3.4 システムの機能	18
3.5 評 価	21
第4章 マイクロコンピュータにおけるPASCALコンパイラシステム	23
4.1 システムの名称	23
4.2 システム製作の目的	23
4.3 システムの構成	23
4.3.1 ハードウェア構成	24
4.3.2 ソフトウェア構成	26
4.4 システムの機能	26
4.4.1 MCC PASCAL言語仕様	26
4.4.2 ソフトウェアの機能	29
4.5 評 価	33



第1章 パフォーマンス測定器



第1章 パフォーマンス測定器

1.1 装置の名称

パフォーマンス測定器

1.2 装置製作の目的

本装置は、マイクロプロセッサの応用装置を開発する際、ハードウェア及びソフトウェアの各部分において、マイクロプロセッサの利用度、各部プログラムの実行時間の割合、利用回数等を測定し、システムのパフォーマンスを評価することを目的としたものである。

従来、特に高度のマイクロプロセッサの応用用途、例えば、多数の割込をリアルタイムで処理したり、1個のマイクロプロセッサを多目的に同時使用する場合、又、処理時間に余裕がなく、時間関係が重要になる場合には、これを測定評価する有用なツールがなかった。

本装置は、マイクロプロセッサの利用の高度化とともに、これに見合った強力なツールを提供し、システム開発の効率化をねらったものである。

1.3 装置の構成

1.3.1 ハードウェア構成

本装置は、次の部分によって構成される。

(1) パフォーマンス測定器本体

解析用マイクロプロセッサ	1式
目的システムの状態記録メモリ	4Kステップ×48ビット
時刻記録用メモリ	4Kステップ×48ビット
タイマ及びカウンタ	1
サンブラ	7個以上

(2) キーボード ディスプレイ

1式	
表示文字数	80×24行
英数カナ文字表示可能	

(3) アダプタ モジュール

Z-80用アダプタ	1式
-----------	----

被試験器CPUが変わった時は、別途開発する被試験器用CPUアダプタモジュールに交換することによって対処することができる。

アダプタモジュールは、パフォーマンス測定器本体にコネクタによって接続される。

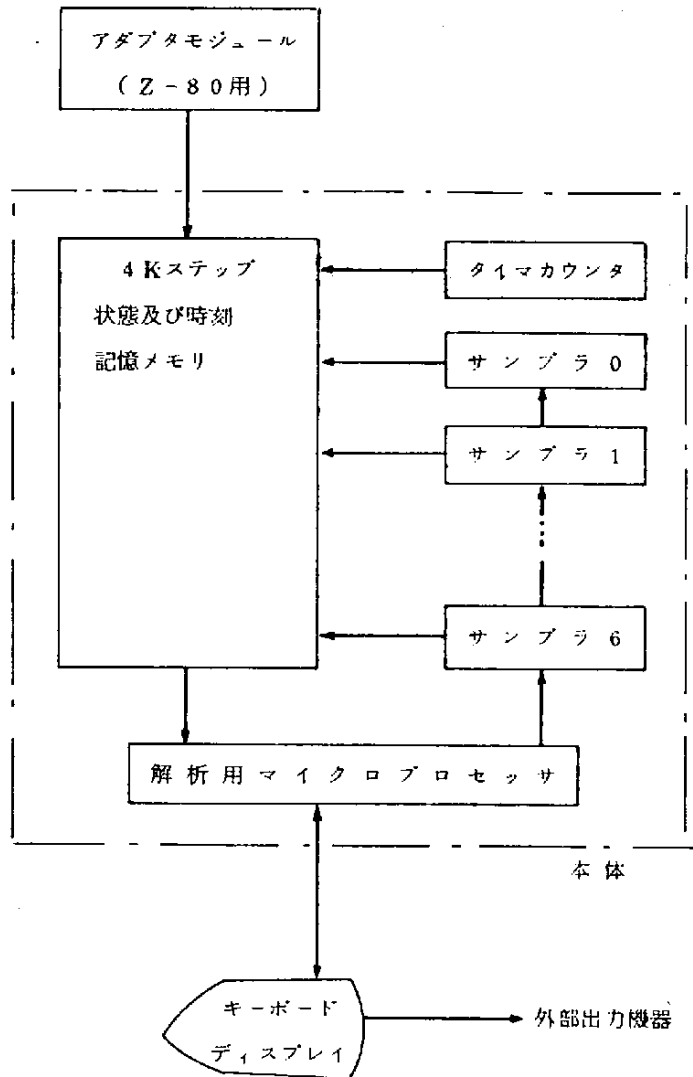


図1-1 ハードウェア構成

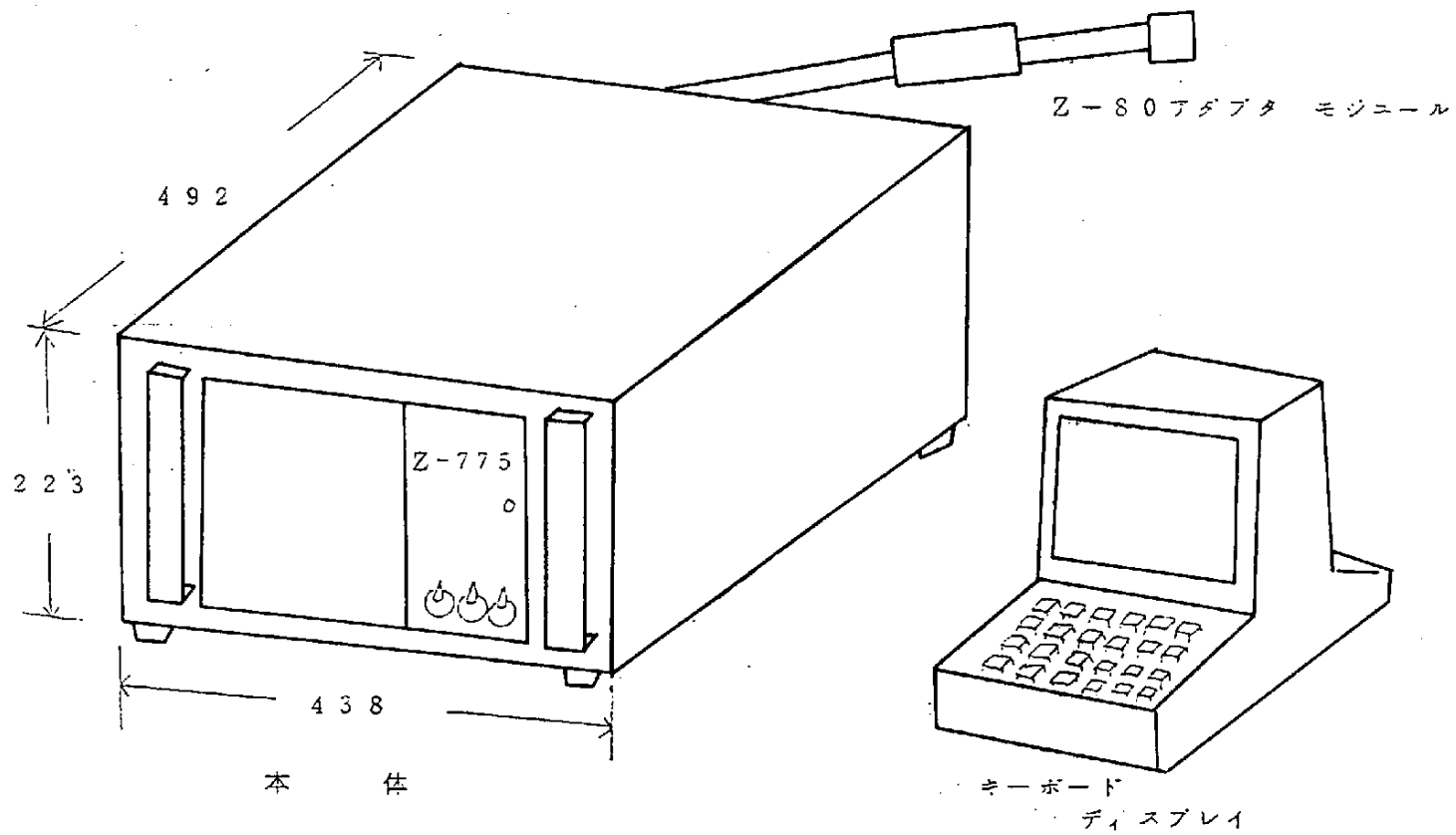


図1-2 外観図

1.3.2 ソフトウェア構成

本装置は、次のソフトウェアより構成される。

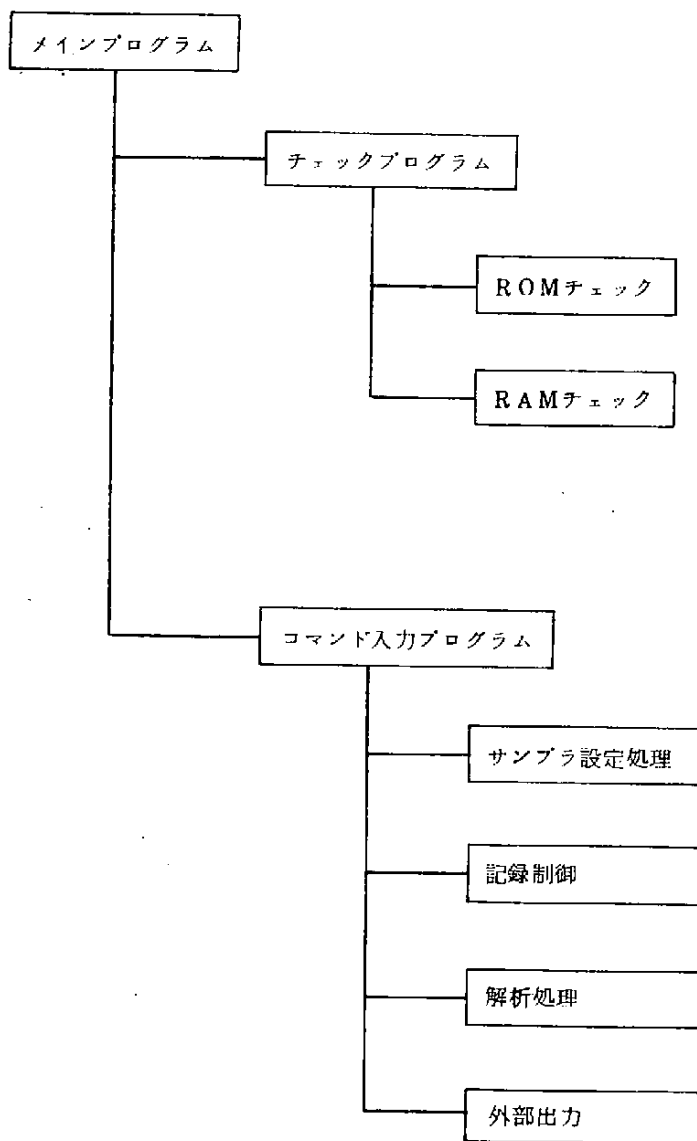


図1-3 ソフトウェア構成

1.4 装置の機能

(1) 測定機能

- ① 被測定マイクロプロセッサシステム（以下 目的システムと記す）の実行状態の必要な部分を 4 Kステップまで、状態記憶メモリに記憶する。
- ② 実行状態の必要な部分は、7個のサンプラによって検出する。
サンプラは「マイクロプロセッサのアドレス信号、データ信号、各種コントロール信号および若干の外部信号」の全体、又は一部が特定のパターンになっていることを検出する。
このパターンは、キーボードより設定する。
- ③ 目的システムの状態とともに、時刻（分解能 1 μ S）を時刻記録メモリに記録することができる。
時刻の記録は、最大 4 Kステップとする。
- ④ 特定のサンプラの出力をカウントする機能を有する。
- ⑤ サンプラの出力を利用して記録の開始、停止を制御することができる。

(2) 解析表示機能

キーボード ディスプレイによって設定された条件に基づき、解析用マイクロプロセッサは、次のことを行ないその結果を表示する。

- ① 状態メモリ及び時刻メモリの表示
- ② 同上の特定パターンのサーチと表示
- ③ プログラム特定部分の実行時間、実行回数の解析

(3) データ記録機能

状態メモリおよび時刻メモリの内容を外部に転送することができる。

(4) 装置診断機能

次の機能によって、装置を診断することができる。

- ① RAMチェック
- ② PROMチェック
- ③ サンプラ機能チェック

1.5 評価

本装置は、マイクロコンピュータ応用製品におけるマイクロプロセッサのパフォーマンスを測定するためのものである。本装置の開発にあたり、必要とされる条件は次の5点であった。

- ① マイクロプロセッサの以下のような状態である頻度、および、費やしている時間の割合を計測できること。
- ② ある範囲の番地の命令をフェッチしたこと。
- ③ ある範囲の入出力アドレスに対する命令を実行したこと。

- ㉓ ある特定の命令を実行したこと。
- ㉔ WAIT, DMAの状態にあったこと。
- ② 前項の各種状態を選択設定する機能を持つこと。
- ③ 計測した結果を表示またはプリンタに出力する機能を持つこと。
- ④ 計測中の状態を表示できること。
- ⑤ 各種の8ビットマイクロプロセッサに対して適用可能なこと。

装置の開発完了にあたり、上記条件を満たしたパフォーマンス測定器となっているかを検討した。

①, ②については、本装置は、7個のサンブラと、状態記憶メモリ、時刻記録メモリを有しておりこれらが以下のような機能を持つことで条件を満たしている。すなわち、サンブラは、目的マイクロプロセッサのアドレス信号、データ信号、各種コントロール信号、外部信号の48ビットをサンプルレ、その全体または一部が特定のパターンになっていることを検出する。このパターンは、キーボードより設定する。このサンブラの出力によって、目的システムの状態の記録の開始、停止が、同じくキーボードからの設定により制御され、各設定条件を満たした目的システムの状態が、1 μ Sの分解能の時刻と共に、4Kステップ分のメモリに記憶される。また、特定のサンブラの出力パターンを計数する機能を有する。

③については、本装置は状態記憶メモリ、時刻記録メモリの内容に対して、次のような解析表示機能を有している。すなわち、状態記憶メモリ、時刻記録メモリの特定の部分の表示。同上の特定パターンの探索、および、計数表示。ある2つの特定パターンで始めと終りが定義される一連の実行状態の探索、計数、そして、その実行時間の総計、割合、各実行時間の平均値、最大値、最小値の表示など。これらの解析表示機能を選択して組み合わせることで、目的システムの動作状態を計測することができる。

④については、計測中の状態がディスプレイに表示される。

⑤については、今回、開発期間の制限から、Z80CPUに対する装置のみが完成している。しかし、本装置では、各種CPUに対しては、アダプタモジュールのみを交換することで対応できる構成となっており、今後の各種CPUへの対応化は容易である。また、サンブラのビットに余裕があるので、16ビットCPUへの対応も可能である。

さらに、本装置では、状態記憶メモリ、時刻記録メモリの内容を外部に転送する機能も有している。

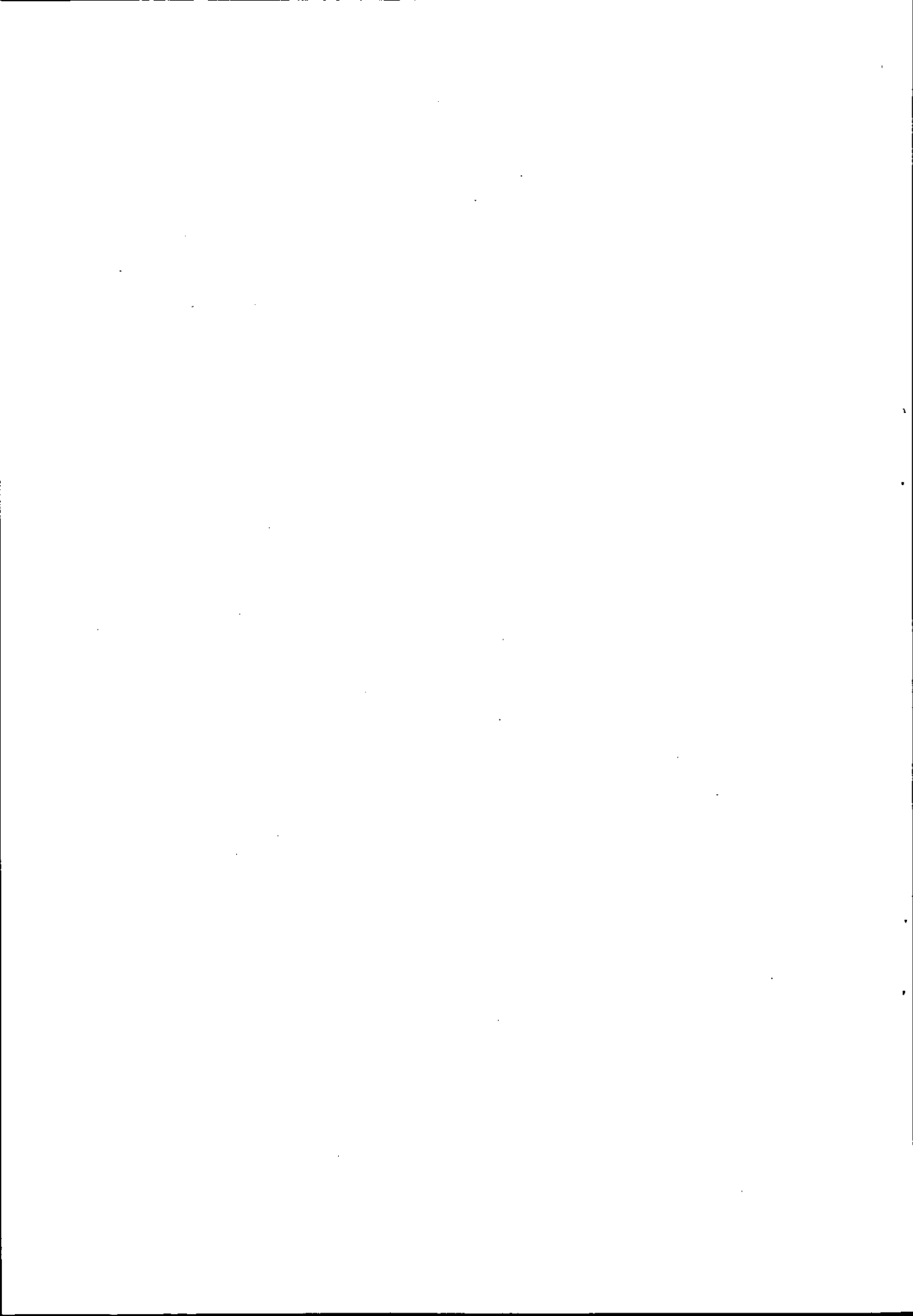
以上のように、本装置に期待していた条件はほぼ満足されており、仕様書に記載された事項に従って製作されている。本装置の出現によって、既存のテスト、デバッグでは行いえなかった、目的システムの長時間にわたる動作状態の把握が可能となった。

問題点としては、本装置によって目的システムのパフォーマンスの測定を行うには、測定条件設定での工夫から、解析プログラムの組合せ方などで、若干の習熟が必要であると思われる。また、本装置は、データ収集に主点が置かれ、現在のところ表記のパフォーマンス測定器という名称よりは、パフォーマンス測定用のデータ集収器と呼ぶ方がふさわしい機器となっている。

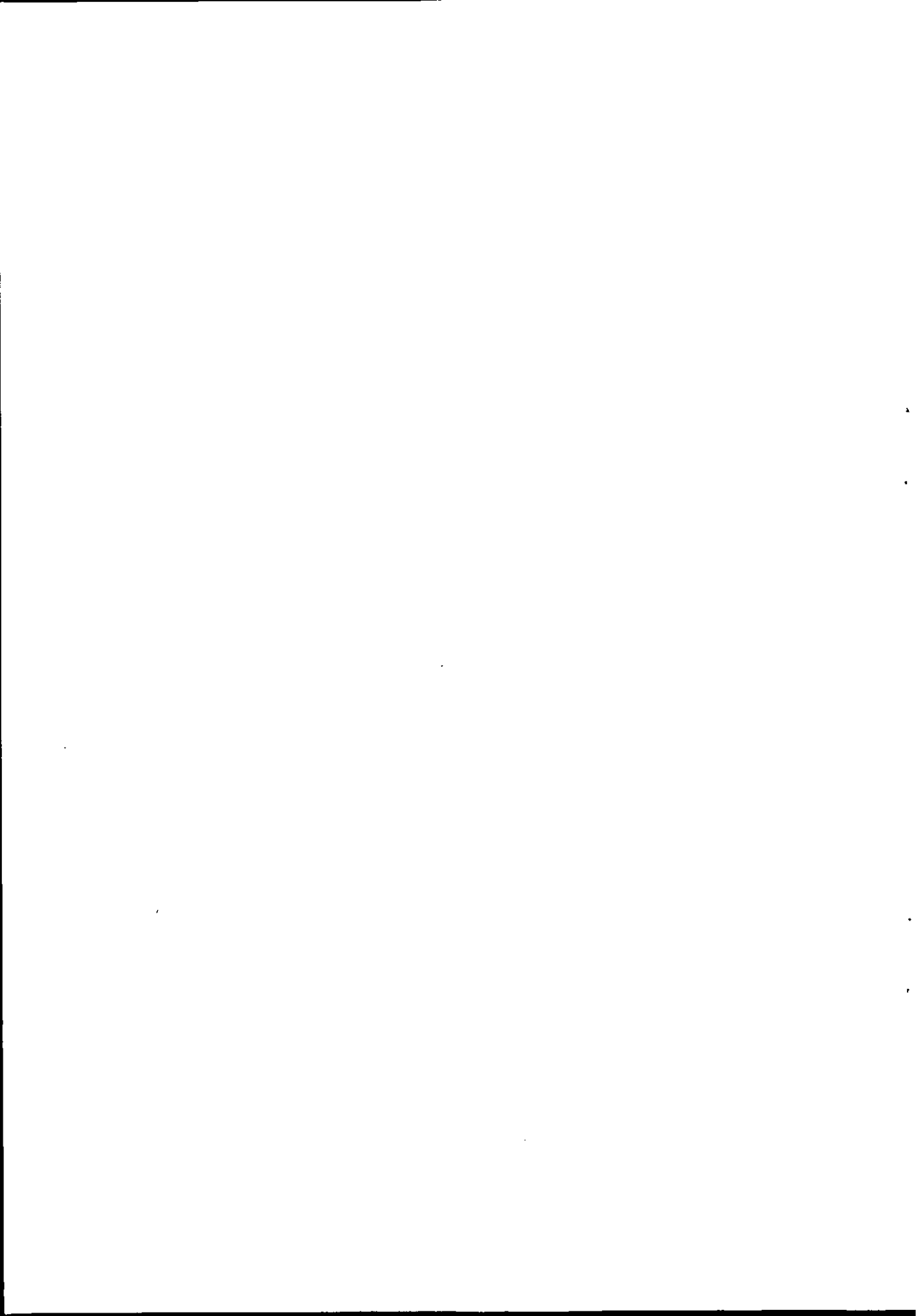
しかし、本来、パフォーマンス測定には、多くの複雑な環境要因が関与すること、また、パフォー

マンス自体の定量化・評価の方式も定まったものでないことから、汎用の機器としては、開発当初ということで、やむを得ないと思われる。これらの点については、今後、本装置が広く使用されていく中で、解析用プログラム、条件設定法などが充実されていくことで解決されていくものと思われる。

今後、本装置がマイクロコンピュータ振興センター（財）日本情報処理開発協会内に設置され、会員等に広く利用されることによりマイクロコンピュータ応用システムの開発、改良での効率向上、経済性向上に大きく寄与することと思われる。



第2章 4ビットマイコン用 開発サポートシステム



第2章 4ビットマイコン用開発サポートシステム

2.1 システムの名称

4ビットマイコン用開発サポートシステム

2.2 システム製作の目的

本システムは、4ビットマイクロコンピュータ応用製品の開発業務における各工程での作業を支援し、応用製品の開発能率の向上と信頼性の向上を図るものである。

本システムは、CP/Mと呼ばれるオペレーティング・システムをもったフロッピーディスクベースの開発サポートシステムである。開発サポートする対象は次にあげた4ビット1チップマイクロコンピュータである。

- ① μ COM43 シリーズ <日 電>
- ② MB8840 シリーズ <富士通>
- ③ TLCS43 シリーズ <東 芝>
- ④ HMCS40 シリーズ <日 立>

また、音声発生機能を有しており、キー入力の音声エコーバック、オブジェクト・コードの照合ができ、操作性が向上している。

2.3 システムの構成

2.3.1 ハードウェア構成

本システムは、本体（FDDを含む）、CRTディスプレイ、プリンタ、XYプロッタ、PTR（紙テープリーダー）、PTP（紙テープパンチャ）、音声発生機により構成される。

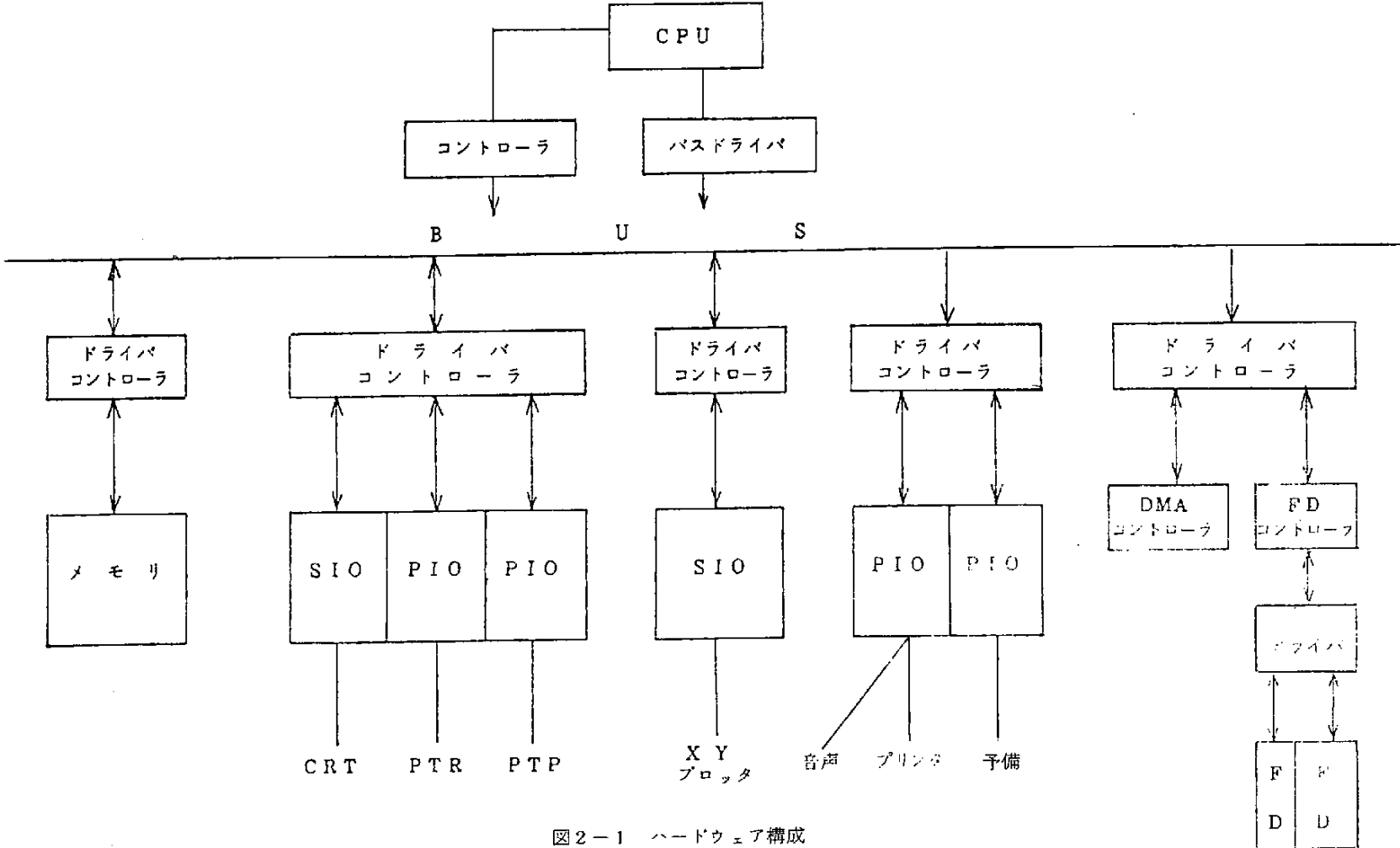
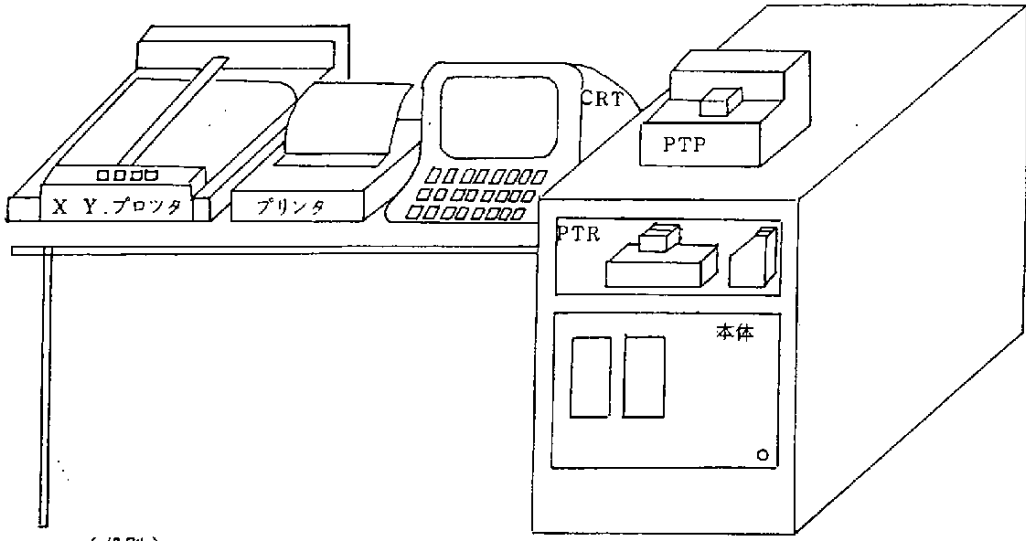


図 2-1 ハードウェア構成



(机別)

図2-2 外観図

2.3.2 ソフトウェア構成

本システムは、前述の4つのプロセッサに対して、それぞれ次のサポートを行うソフトウェアより構成される。

- ① クロスアセンブラ
- ② 逆アセンブラ
- ③ シミュレータ
- ④ フローチャータ
- ⑤ 自己診断機能

2.4 システムの機能

(1) クロスアセンブル機能(クロスアセンブラ)

4ビットマイクロコンピュータ用ソースプログラムをクロスアセンブルし、アSEMBルリスト及びオブジェクトプログラムを出力する。

4ビットマイコンのソフトを作成するユーザは、システム的设计、コーディングを行なって、ソースプログラムを作成する。ソースプログラムは、人間の目にわかりやすいニモニックで書かれており、このままではマイコンに記憶できない。そこで、このクロスアセンブラを用いて、ニモニックから機械語を作成するのである。機械語でできたプログラムをオブジェクト・プログラムといい、アSEMBルして作成されたこのオブジェクト・プログラムを、すぐに実機に載せないで、まず、シミュレータで

ロジックの誤りや、コーディングの誤りを検査する。

(2) トレース機能 (シミュレータ)

オブジェクトプログラムの命令コードにより4ビットマイクロコンピュータの各種レジスタ、RAMの状態、入出力ポート等の状態をシミュレーションし、CRTに表示しながらプログラムをトレースする。

ユーザは、このシミュレータを用いて、プログラムのトレースをし、動作を確認することができる。誤りがみつかり、命令コードを書き替えたり、レジスタの内容を変更したりして、プログラムの修正をしながら、トレースを続けることができる。こうして、ひとつおき修正箇所を抜き出して、ソースプログラムをテキストエディタで修正し、アセンブルを行なう。シミュレータでデバッグされたプログラムを評価用機器や実機に載せ、ハードウェアを含んだ総合テストを行なう。そこで発見された誤りを修正し、最終プログラムを作成する。総合テストでは、通常PROMを用いる。PROMによる命令コードの修正や、テストを繰り返して、プログラムを作成し、逆アセンブラを用いて、ソースプログラムに復元することもできる。

(3) 逆アセンブル機能 (逆アセンブラ)

4ビットマイクロコンピュータ用オブジェクトプログラムを逆アセンブルし、逆アセンブルリストを出力する。

こうして作成された最終ソースプログラムに対応したフローチャートを作成し、プログラム仕様書と共にドキュメントとして納品するのである。

(4) フローチャート作図機能 (フローチャータ)

4ビットマイクロコンピュータ用ソースプログラムからフローチャートを作図する。

(5) 音声発生機能

オブジェクトコード等の読み合わせに使用する。

(6) 自己診断機能

本システムで使用するハードウェアの動作確認を行う。

本システムは、本体、CRTディスプレイ、プリンタ、プロッタ、PTR、PTPで構成されており、本体に接続する機器が多い。自己診断プログラムは、こうした機器と本体が正常に接続されているか

どうかを調べ、その結果を画面に表示する。本体自身の検査については、CRTを接続して、内部にあるブート・ストラップROMを自己診断ROMに差し替えて行う。

2.5 評 価

本システムの開発条件は次の6点であった。

- ① 各種の4ビットマイクロプロセッサに対して適用可能なこと。
- ② プログラムの開発に際して、デバッグ時の修正およびロジック検証して役立つこと。
- ③ プログラム完了時のドキュメント作成のためのフローチャート作図機能を有すること。
- ④ クロスアセンブル機能および逆アセンブル機能を有すること。
- ⑤ 既存のシステムに対して新規性をもつこと。
- ⑥ 自己診断機能を有すること。

装置の開発完了にあたり、上記条件を満足しているか否かを検討した。

①については、対象機種として μ COM43シリーズ(日本電気)、MB8840シリーズ(富士通)、TLCS43シリーズ(東芝)、HMCS40シリーズ(日立)の4機種とした。

②については、ソフトウェアとしてシミュレータを作成し、紙テープリーダーまたはフロッピーディスクより、オブジェクトプログラムを読み込み、命令コードにより4ビットマイクロコンピュータの各種レジスタ、RAMの状態、入出力ポート等の状態をシミュレーションし、CRTに表示しながらプログラムをトレースする。

このCRT表示を見ながら、命令コードによりデバッグ及びロジック検証を行う。

③としては、紙テープリーダーまたはフロッピーディスクより、ソースプログラムを読み込み、フローチャートをX-Yプロッタに作図する。

④については、クロスアセンブル機能および逆アセンブル機能を有している。クロスアセンブル機能として、紙テープリーダーまたはフロッピーディスクより、ソースプログラムを読み込み、所定の形式のオブジェクトプログラムを紙テープパンチャまたは、フロッピーディスクに出力する。ユーザ指定はコンソールで行う。

逆アセンブル機能として、紙テープリーダーまたはフロッピーディスクより、オブジェクト・プログラムを読み込み、逆アセンブルリストをプリンタに出力する。ユーザ指定はコンソールで行う。また、このリストをフロッピーディスクに登録することもできる。

⑤については、これまでの開発サポートシステムはアセンブラやシミュレータなどであり、フローチャートまで作成するトータルサポートシステムはなかった。

また、多種の4ビットマイクロコンピュータに対応出来るサポートシステムは現在見当たらない。しかも操作性並びに正確性の向上に音声を取り入れ、オブジェクト出力の際の読み合せなどに使用出来る。

また音声出力はスピーカまたはヘッドフォン出力が可能であり、また必要のない時は出力しないこ

とも可能である。

⑥については、診断用ROMを付加し、本体とコンソールの診断を行い、次に、自己診断プログラムによりプリンタ、X-Yプロッタ、PTR、PTP、音声出力装置の診断を行う。

以上のように本装置に期待していた条件はすべて満足しており、仕様書に記載された事項に従って製作されている。

しかし、さらに機能を拡充し、使い易いものにするためには次のようなことが考えられる。

(1) ソフトウェアの充実

MN-1400シリーズ(松下)、SM-4シリーズ(シャープ)、TI社のシリーズなどの4ビットマイクロプロセッサでも使用可能とする。

(2) ハードウェアの低価格化

現在使用中のX-Yプロッタが高価なため、より低価格なものを導入し、販売時の価格を低くするよう努力する。

(3) 本装置はZ80CPUと64KバイトのRAM・および8インチ両面倍密度のフロッピーディスクを2ドライブ装備し、その他通常の周辺装置に加えて、X-Yプロッタ、音声出力装置を有し、OSとしてはCP/Mを装備している。

このため、4ビットマイコン用開発サポートシステムとしてだけでなく、8ビット用ソフトウェア開発システム等としても使用可能である。したがって、今後、本装置がマイクロコンピュータ振興センター(財)日本情報処理開発協会内の会員各位に広く使用されることにより業務の効率化と開発時間の短縮が得られるものと思われる。

一方、製品化され安価で市場に提供されることが望まれる。

第3章 8080-6800双方向

ソースプログラムコンバータ



第3章 8080-6800 双方向ソース プログラムコンバータ

3.1 システムの名称

8080-6800 双方向ソースプログラムコンバータ

3.2 システム製作の目的

本システムは、マイコンのプログラム開発として一般的に用いられている“CP/M搭載のマイコン開発システム”を使用し、代表的なCPUである8080と6800を対象としてアセンブリ言語プログラムをソースプログラムレベルにて双方向に変換するものである。

8080ソースプログラム紙テープを入力して変換後6800ソースプログラム紙テープを出力する。また、6800ソースプログラム紙テープを投入して変換後8080ソースプログラム紙テープを得るシステムである。

8080CPUと6800CPUは同じ8ビット系であるとはいえレジスタの個数、性質、演算フラグの立ち方、アドレッシング方法、ROMとRAMの配置法、入出力処理と大巾にアーキテクチャが異っており、まったく完全な変換は不可能に近い。

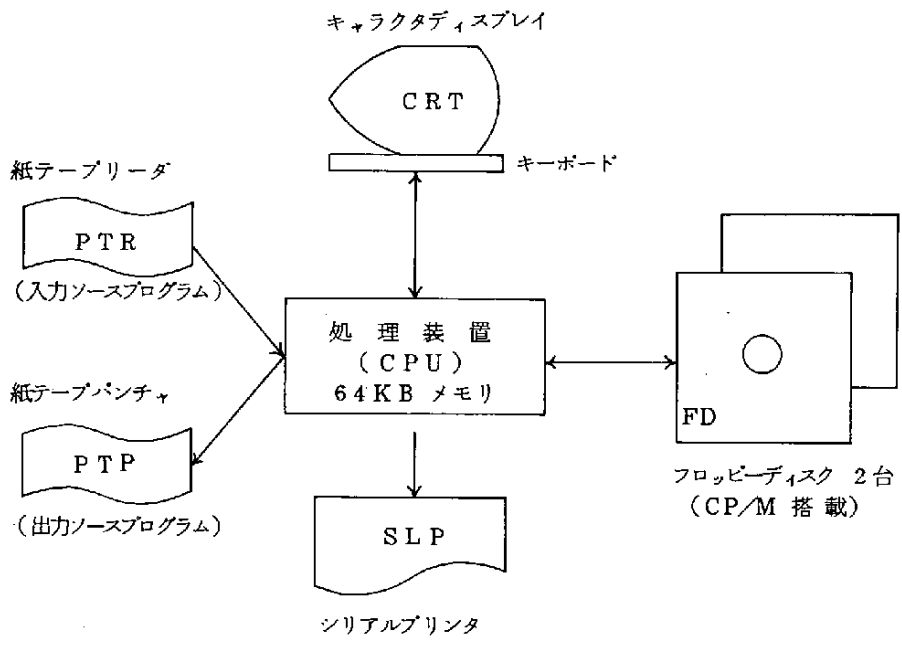
従って、本システムにおいては変換後プログラムが効率的で実用にたえられるようにメモリサイズ・実行時間が過大にならないように変換している。すなわち、命令変換の前にプログラムチェックリストを出力し、演算フラグの立ち方、変換不能命令の表示をしており、オペレータの介入によってターゲットCPUに合わせたプログラム修正を可能とし、その後、命令変換を行い、プログラム変換リストとターゲットプログラム紙テープを出力するようにしている。

このソースプログラムコンバータの開発により、マイコンソフトウェアが他系のマイコン上で動作可能となり、開発済プログラムの使用効率が高まり、流通範囲の拡大が期待できる。

3.3 システムの構成

3.3.1 ハードウェア構成

CP/Mを搭載している一般的マイコンプログラム開発システムであり次のような構成となる。



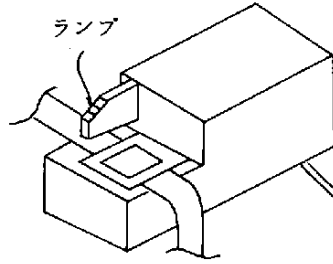
(注) フロッピーディスクは単密度と倍密度の両方が扱える。

図3-1 システム構成

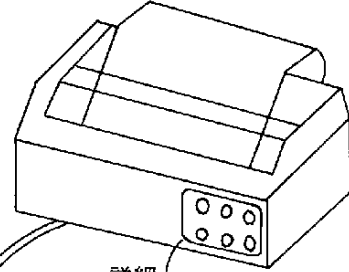
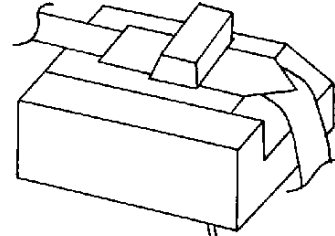
PTR (紙テープリーダー)

PTP (紙テープパンチャ)

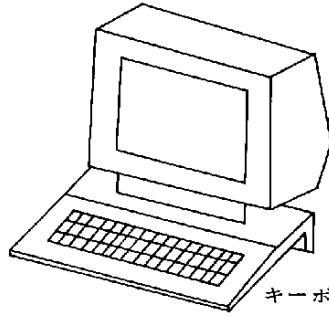
SLP (シリアルプリンタ)



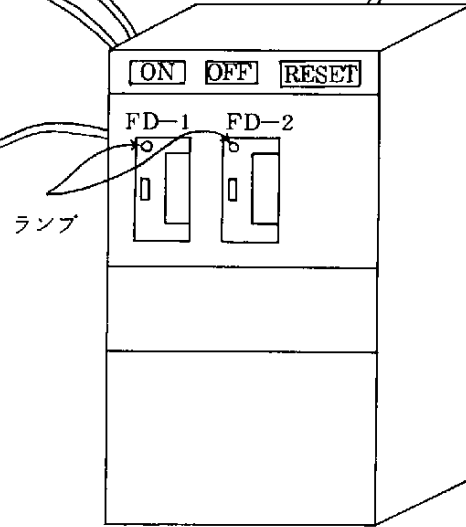
ランプ



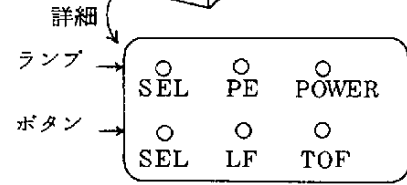
CRT (キャラクタディスプレイ)



キーボード



ランプ



FD (フロッピーディスク) 及び

CPU (処理装置)

64 KBメモリ

図3-2 外観図

3.3.2 ソフトウェア構成

CP/Mの下にて動作し、8080 ↔ 6800 変換ライブラリファイルの基本ファイルとして命令変換マクロ展開システムプログラムを中心にして次のような構成となっている。

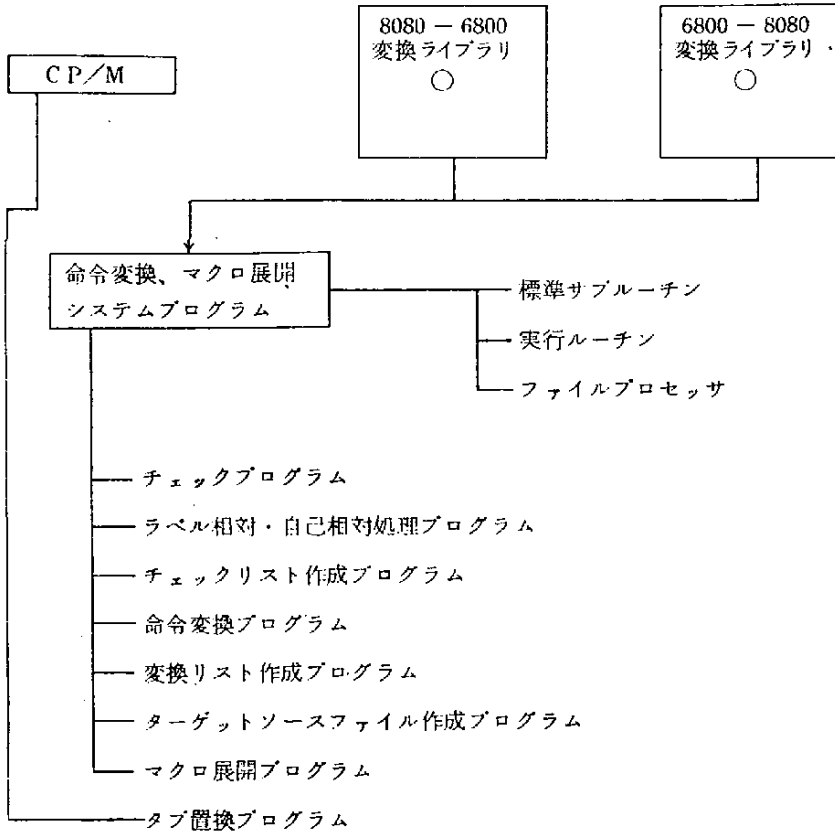


図3-3 ソフトウェア構成

命令変換、マクロ展開システムプログラムは最適化ロジックの入りこんだ変換ライブラリを読みこんで命令の変換とマクロの展開を基本とする一種のコンパイル&ゴーシステムであり、作表やファイル処理もできる言語プロセッサである。

3.4 システムの機能

(1) 変換処理の範囲

- ① 8080系は8080と8085を対象としたインテル社言語、6800系は6800を対象とするモトローラ社言語を扱う。
- ② 入出力、割込関係を除くプログラムモジュールの変換を扱う。
- ③ 入出力媒体は紙テープとし、変換後のリストを出力する機能を持つ。
- ④ ソースプログラムとしては完成品を扱う。

⑤ ソースマシン1命令よりターゲットマシン命令を作り出す方式を原則とする。

⑥ 変換後命令の最適化処理を行う。

(2) 8080→6800ソースプログラム変換機能

変換手順は次図の通りであり、その概略ステップを示す。

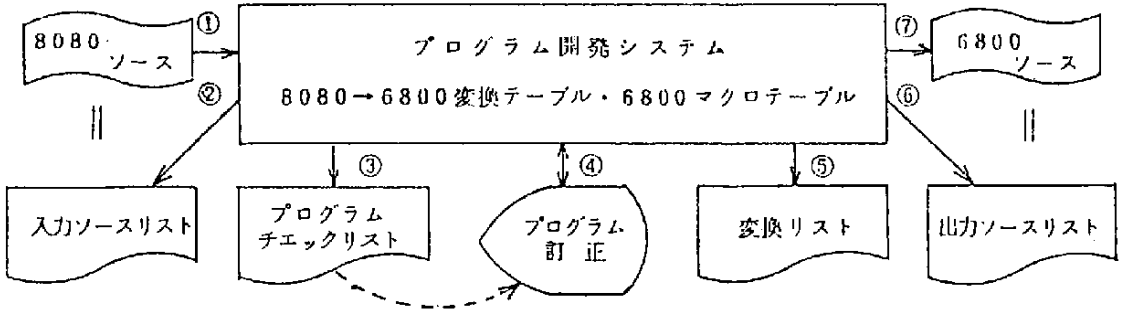


図3-4 8080→6800変換図

① 8080ソースプログラム紙テープをフロッピーディスクへ格納する。(CP/M機能使用)

② フロッピーディスク内のプログラムをリスティングする。(CP/M機能使用)

③ 8080ソースプログラムを解析し、不良、不要命令の検出を行って各命令の演算フラグを表示してプログラムチェックリストを作成する。

④ コンピュータと会話形式にてソースプログラムの整理をするところであり、i/o関係命令を取り除いたり、必要な演算フラグ命令を指示したり、代替となる標準マクロ命令を投入して、変換後6800プログラムの最適化をはかる。

⑤ 8080命令とキーインしたマクロ命令を6800命令に変換し、変換リストを作成する。

⑥ 8080命令を抜きとり、6800命令のみとして変換後プログラムをリスティングする。(CP/M機能使用)

⑦ 変換後プログラムを紙テープにパンチアウトする。(CP/M機能使用)

(3) 6800→8080ソースプログラム変換機能

変換手順は4.2と同じ手順となるため変換図のみを次図に示す。

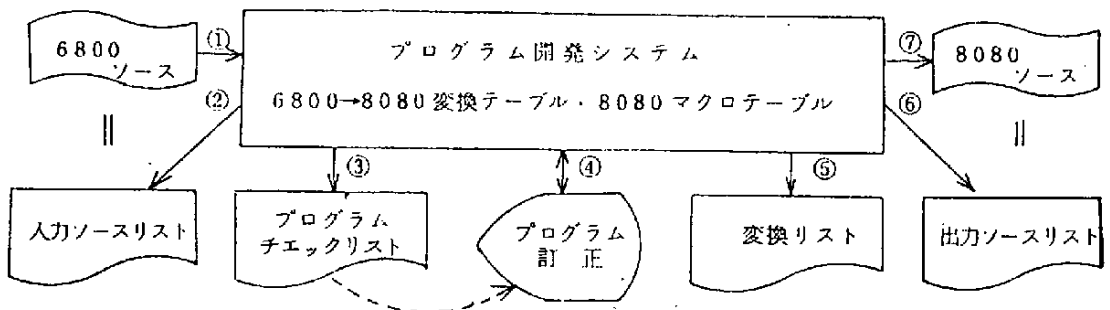


図3-5 6800→8080変換図

(4) マクロ命令展開機能

(2), (3)のステップ④と⑤の間に入る機能であり、ユーザ処理マクロをソースプログラムステップに展開する機能である。具体的には次図の様な処理になる。

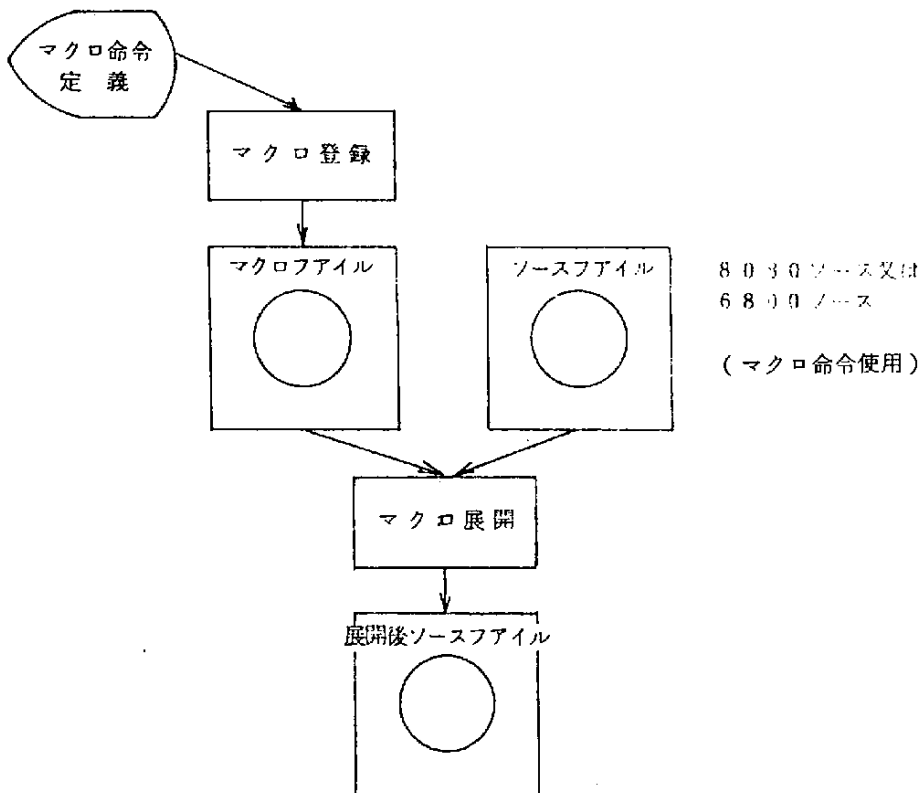


図3-6 マクロ命令展開機能図

なお、このマクロ展開機能はプログラム開発にも使えるものである。

3.5 評 価

本システムの開発にあたり、前提条件として挙げられたのは次の5点であった。

- ① 8080系は8080と8085を対象としたインテル社言語，6800系は6800を対象としたモトローラ社言語を使う。
- ② 入出力、割込関係を除くプログラムモジュールの変換を扱う。
- ③ ソースプログラムとしては完成品を扱う。
- ④ ソースマシン1命令よりターゲットマシン命令を作り出す方式を原則とする。
- ⑤ CP/M内蔵のプログラムは最大限に利用するとする。(少々オペレーションが複雑になろうとも)

本システムの開発完了にあたり、上記条件に関し、以下のような検討を行った。

①、②については前提条件通り変換が可能で、しかも変換後のソースを更に変換することも可能である。なお、ソースの入力媒体としては紙テープは勿論のこと、フロッピディスクおよびキーボードも選択できる。

また、出力媒体としては紙テープ、フロッピ・ディスクおよびプリンタが利用できる。

③、④については、ソースマシンの1命令毎の単純な変換では変換後の命令数の増加は避けられないが、本システムでは、既変換命令の扱いレジスタの値を照合することによって、冗長な命令の変換を省略(最適化)し、変換効率(メモリ、実行速度)の向上を図っている。

実際のプログラム例では命令数で約20%の減少となっていて、最適化を行った結果は次表のように実用的なものである。

		8080→6800→8080	6800→8080→6800
命 令 数	変換前	25 (1.0)	26 (1.0)
	変換後	36 (1.4)	41 (1.0)
	再変換後	66 (2.6)	65 (2.5)
実行時間	変換前	11.2秒 (1.0)	23.2秒 (1.0)
	変換後	47.7 (4.3)	37.9 (1.6)
	再変換後	05.0 (5.8)	17.12 (7.4)

⑤については、

- (1) ソースのファイルへの入出力
- (2) 入出力ソースのリスト出力
- (3) チェック・リストと変換リストの出力
- (4) 入力ソース・ファイルの編集

などは、CP/Mの機能を利用している。

その他、本システムはマクロ機能を有し、ユーザに解放されているので、マクロ登録をすることに

より、他機種への変換も可能である。また、マクロ言語プロセッサも含まれており、マクロ機能と併用することによりシステム開発用ツールとしても有用である。

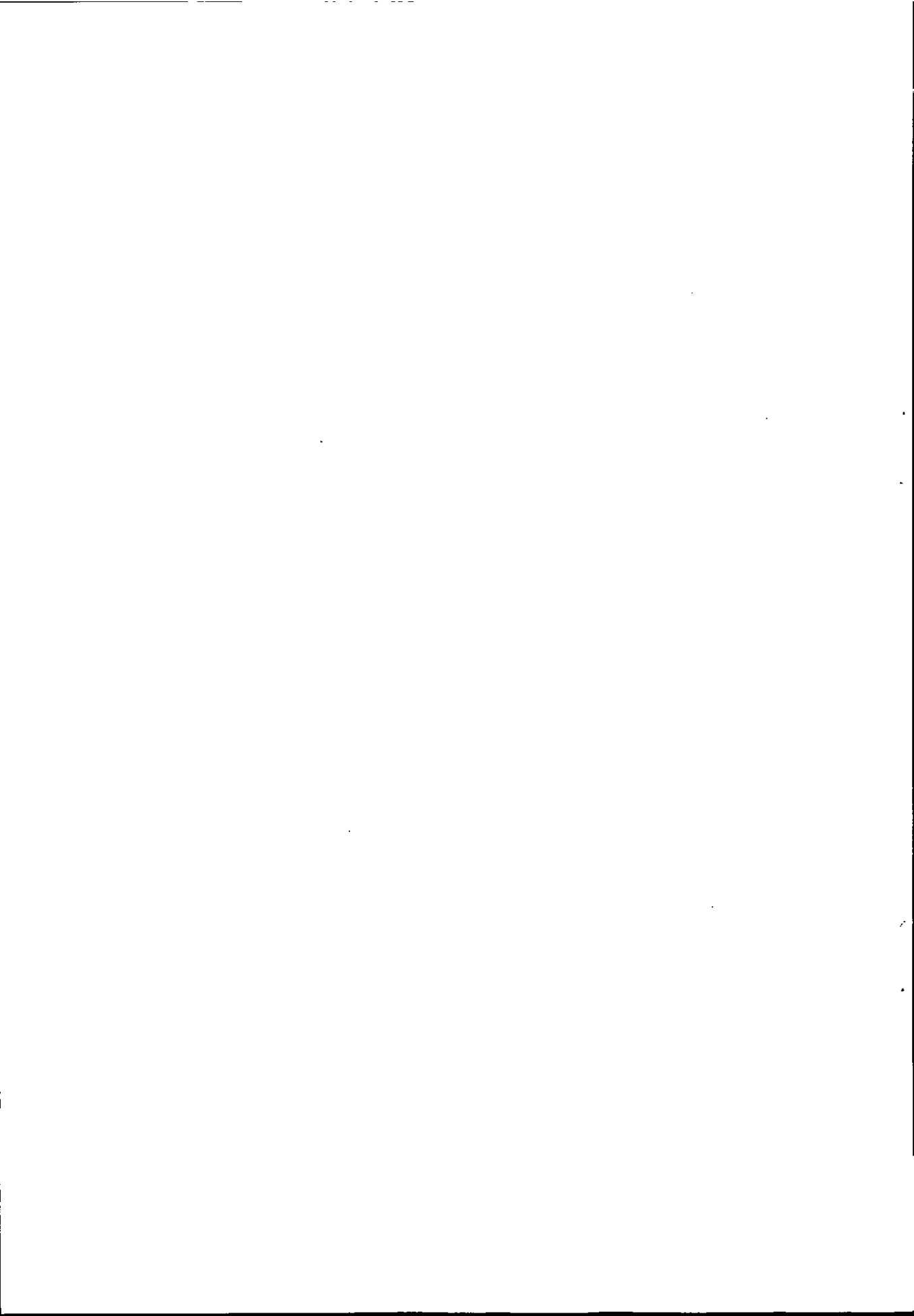
以上のように本システムは前提とした条件を満足しており、仕様書に記載された事項に従って製作されている。なお、本システムをマイクロコンピュータ振興センターに設置し、会員の共同利用に供する目的を考慮すると、今後検討すべき項目として次のものが挙げられる。

- (1) 入出力、割込関係を含むプログラムの変換
- (2) 他機種(Z80, M6809など)への変換
- (3) ユーザの必要とする任意のフラグの保存
- (4) システムからのエラーメッセージの強化

今後、本システムがマイクロコンピュータ振興センターに設置され、会員各位に広く利用されることにより、マイクロコンピュータ応用システムの開発および運用にあたっての効率向上、経済性向上に大きく寄与することが期待される。

会員各位の積極的な利用を望んで止まない。

第4章 マイクロコンピュータにおける PASCALコンパイラシステム



第4章 マイクロコンピュータにおける PASCALコンパイラシステム

4.1 システムの名称

MCC PASCALコンパイラシステム

4.2 システム製作の目的

高位言語によるマイクロコンピュータシステムのソフトウェア開発を推進し、マイクロコンピュータソフトウェア生産性向上を図ることを目的として、MCC PASCALコンパイラシステムの開発を行う。

従来のマイクロコンピュータ用汎用ソフトウェアは、米国で開発されたものがほとんどであるためユーザとしてブラックボックスの利用に限定され、積極的に改造、移植する可能性はほとんど無いのが現状であった。本MCC PASCALコンパイラシステムの開発により、コンパイラの内部構造を含めて公開された処理系を提供することにより、一層の普及拡大をもたらすとともに、ユーザの実状に合わせて、改造、機能向上、移植が可能となる。

プログラミング言語PASCALは、ALGOLの精神を受け継いだ言語として、Prof. Dr. Niklaus Wirthによって1968年に設計された。(標準PASCAL) 実際のコンパイラは、1970年にCDC 6000上で実現され、以後、他機種に対しての移植を目的とするPASCAL-P1, P2, P3, P4コンパイラが開発された。

一方、マイクロコンピュータ上で走るPASCALは、1978年にKenneth L. Bowlesによって開発された。このシステムは、UCSD PASCALシステムと呼ばれ、64Kバイトの主記憶とフロッピーディスクだけの非常に小さな環境の中で動作可能とした点が画期的である。

MCC PASCALは、P4コンパイラをベースに8080相当のマイクロプロセッサ、64Kバイトの主記憶、フロッピーディスクから構成されるシステムのCP/M上で走ることを可能とするとともに言語仕様としていくつかの拡張を行い、また、システムとして、8080へのトランスレータを備えるなどユーザにとってかなり使い易いシステムとなっている。

4.3 システムの構成

本システムは、CP/M(Control Program/Monitor)に準じたOSの下でPASCALソースプログラムからP-コード(中間コード)を生成するPASCAL-Pコードコンパイラ、P-コードから8080ネイティブコードを生成するトランスレータ、P-コードを8080上で実行させるインタープリタおよびP-コードローディングモニタの各ソフトウェアおよびこれらが稼動可能な標準ハードウェアシステムで構成される。

4.3.1 ハードウェア構成

MCC PASCALを使用するためのハードウェア構成を示す。

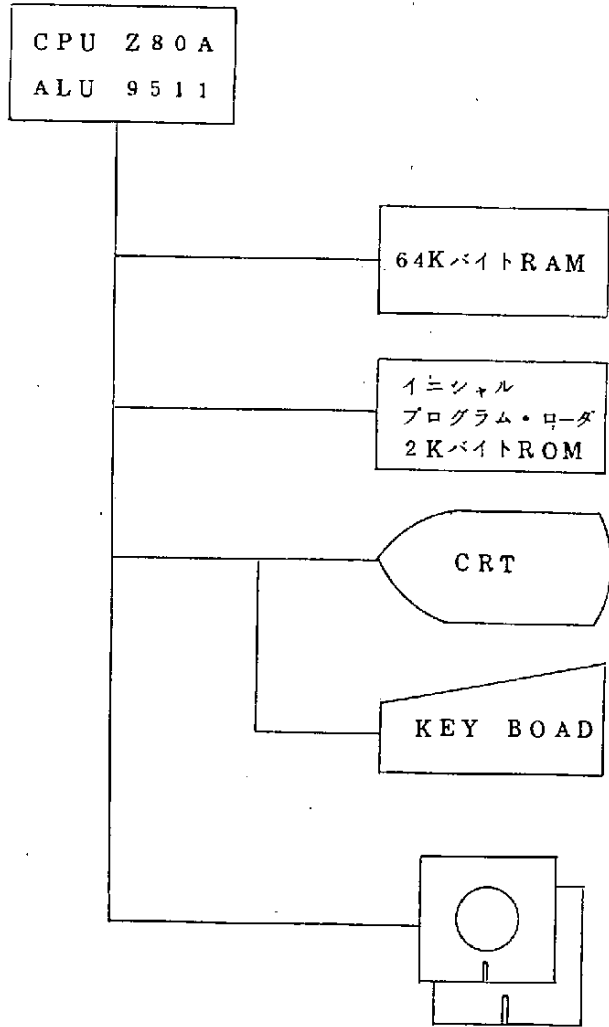


図4-1 ハードウェア構成

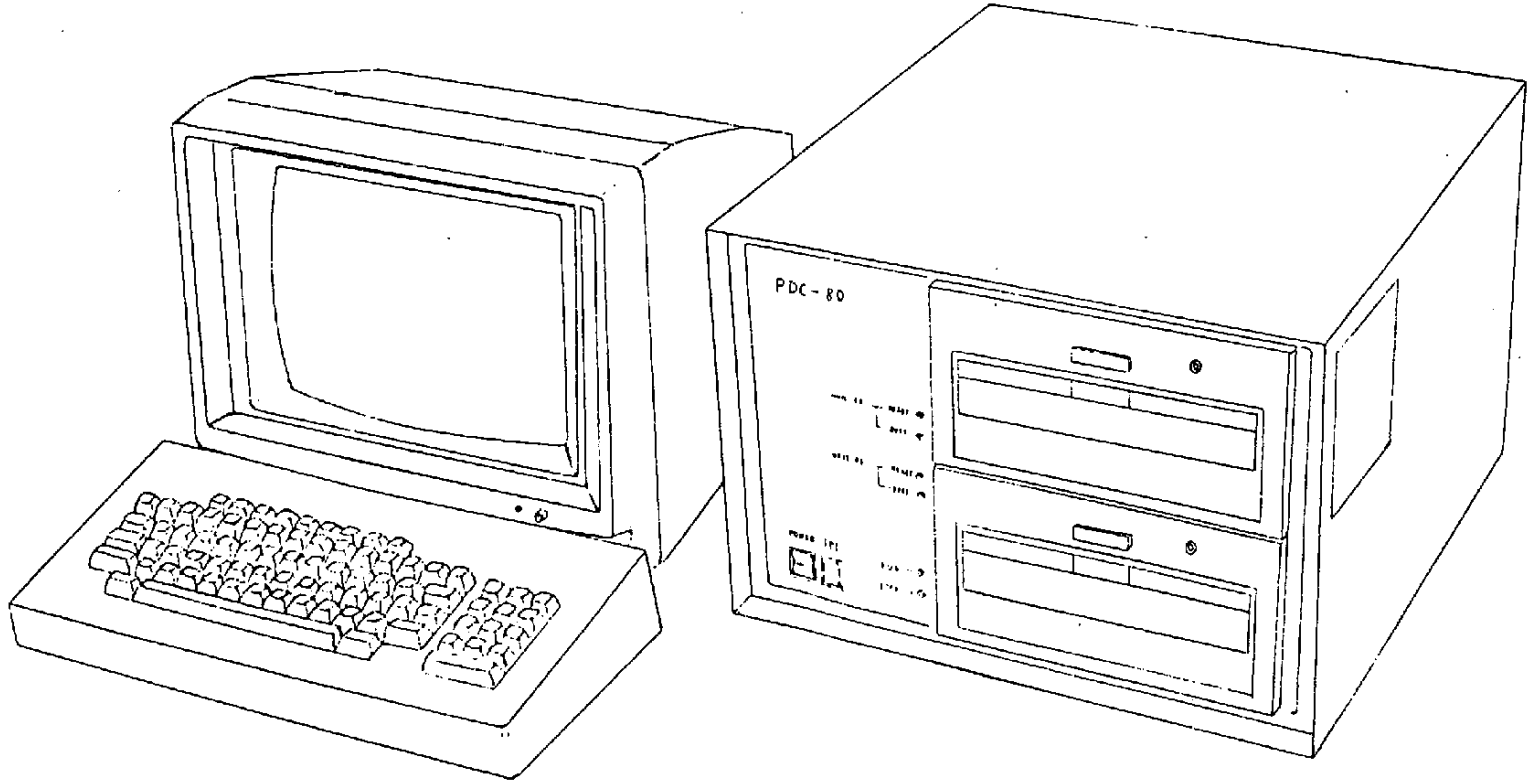


图4-2 外觀图

4.3.2 ソフトウェア構成

本システムのソフトウェア構成は次図のとおりである。

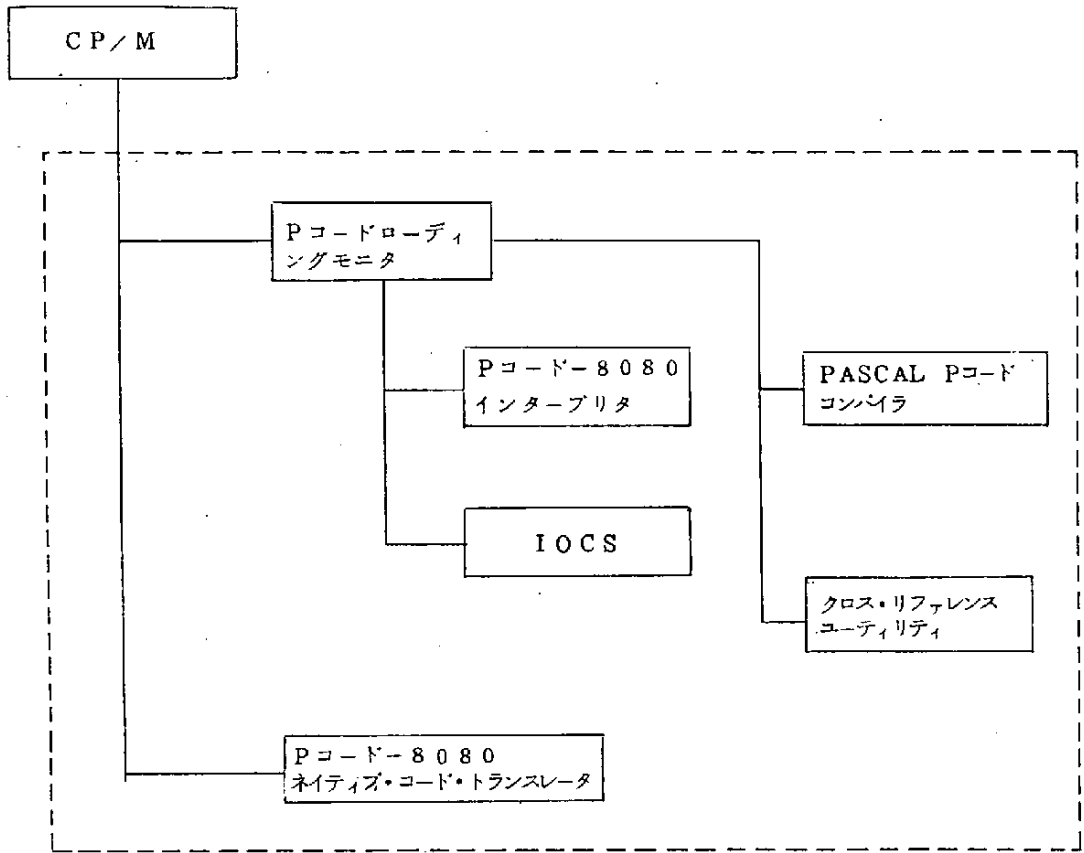


図4-3 ソフトウェア構成

4.4 システムの機能

4.4.1 MCC PASCAL 言語仕様

(1) 文 法

MCC PASCALは、Kathleen Jensen. & Niklaus Wirthの“PASCAL USER MANUAL AND REPORT.”に記載されている標準PASCALに準じる。

(2) 制 限

(a) 整 数

整数型のとりに得る値の範囲は、16ビット(-32768~32767)とする。

(b) 実 数

実数型のとりに得る値の範囲は、32ビット($\pm 3.4028 \times 10^{19}$)とする。

実数型の入出力は、有効桁6けたのE型表現とする。

(c) Set型

Set型で、扱える要素は、4096とする。

(d) 動的メモリ割り付け

動的メモリ割り付けは、MARK及びRELEASEに依る。

(e) GOTO

GOTOは、同一ブロック内のラベルに対してのみ有効とする。

(f) 手続き及び関数

手続き、または関数の引数として、手続き名、関数名を引き渡すことができない。

(3) 拡張

(a) ランダム・アクセス・ファイル

ディスク・ファイル内の特定レコードに対し、ファイル・ポインタをセットする標準関数“SEEK”を備えている。

SEEKは、引数としてファイル名とレコード番号をもつ。

SEEK(ファイル名, レコード番号)

ファイル名で指定するファイルは、Structureで定義されたものである。レコード番号は、整数型であり0から数える。

SEEK動作は、ファイル名で指定されたファイルのファイル・ポインタをレコード番号で指定したレコード位置に変更する。したがって、SEEKにつづくPUT、GETで要求したレコードに対しアクセスできる。

SEEK(FIB, 5);

GET(FIB);

(b) Case文

Case文の選択子が、いずれのラベルである定数にも該当しない時、エラーを発生させずに次のステートメントを実行する。

(c) 分割コンパイル

MCC PASCALでは、メイン・プログラムから呼ぶ手続きの一部をメイン・プログラムのファイルと異なったファイルとしてコンパイルすることができる。

分割コンパイルを行うために、メイン・プログラム側のファイルと呼ばれる手続側のファイルで最小限の情報を書く必要がある。この情報は、リンカ情報としてコンパイラから出力され、リンカによって単一セグメントの実行可能なPコードを完成させる。

メイン・プログラム側では、外部手続き名とそれらを含むファイル名を“USE”で宣言する。この宣言は、プログラム・ヘッダの直後に書かなくてはならない。

外部手続き側では、メイン・プログラムで呼ばれる手続き名と、そのファイル名を“UNIT”及び“HEADER”で宣言する。この宣言の後から“BODY”と“END”で囲まれたブロックの

中に各手続きを記述する。

<メイン・プログラム側>

```
PROGRAM プログラム名；  
  USE ファイル名；  
    手続き名；手続き名；  
    .  
    .  
  USE ファイル名；  
    手続き名；  
    .  
    .  
    手続き名；  
  END；  
  {  
    ブロック  
  }  
END .
```

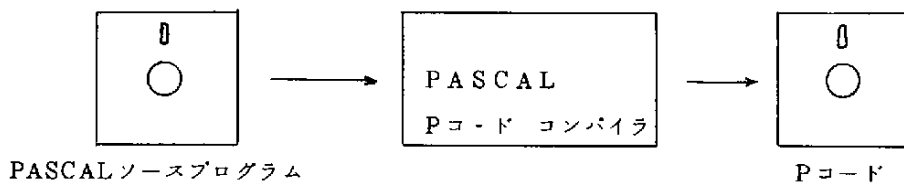
<外部手続き側>

```
PROGRAM プログラム名；  
  UNIT ファイル名；  
    HEADER  
      手続き名  
      .  
      .  
      手続き名  
    BODY  
      {  
        各手続き  
      }  
  END .
```

4.4.2 ソフトウェアの機能

(1) PASCAL Pコード コンパイラ

PASCAL Pコード コンパイラ(以下コンパイラ)は、MCC PASCAL 文法に従って記述されたソース・プログラムをPコードに変換するプログラムである。

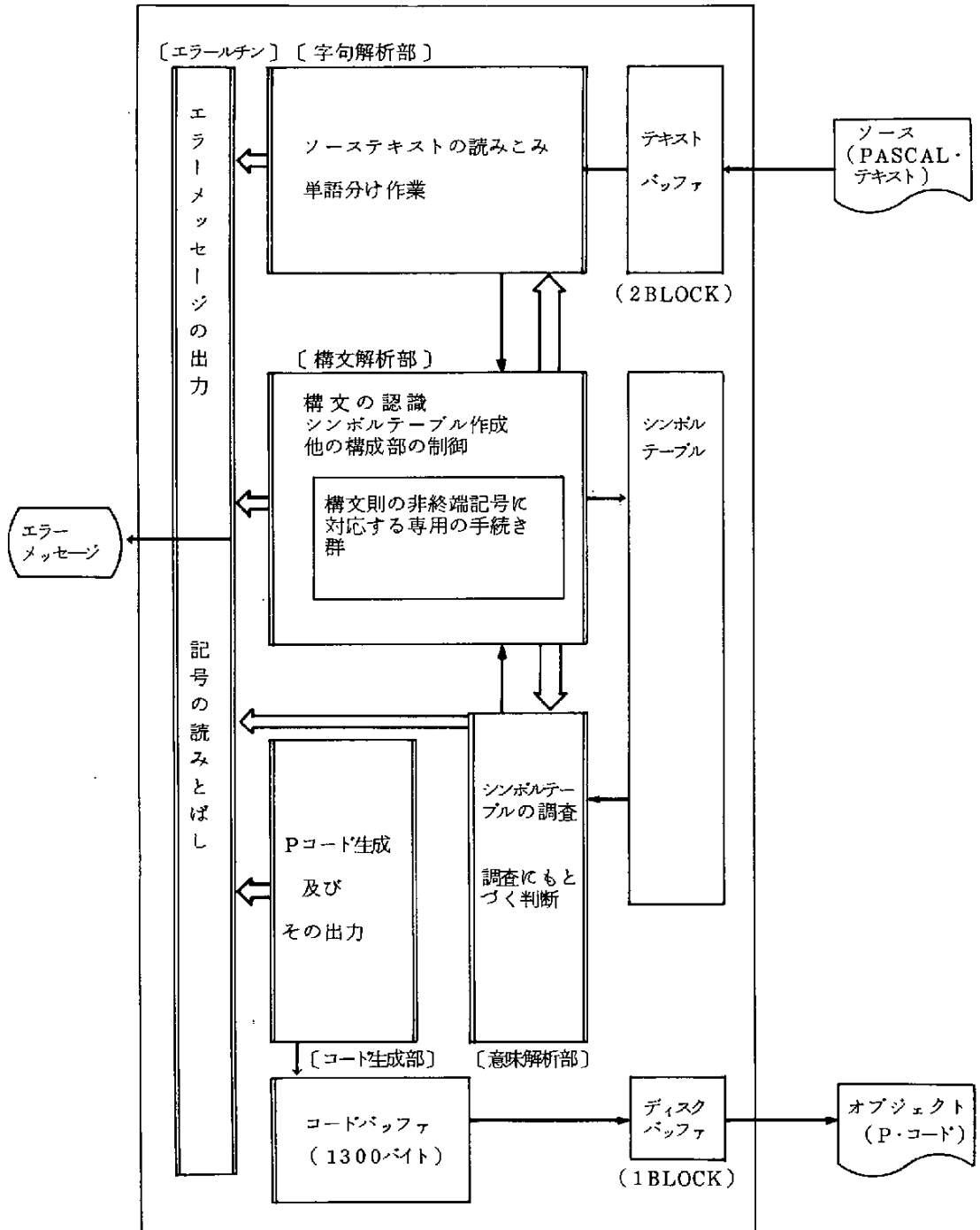


コンパイラは、大きく分けて、構文解析部、意味解析部、コード生成部から成る。

構文解析部は、ソース・プログラムからテキストを1文字ずつ読み取り、PASCAL 構文フローに基き文法のチェックを行う。

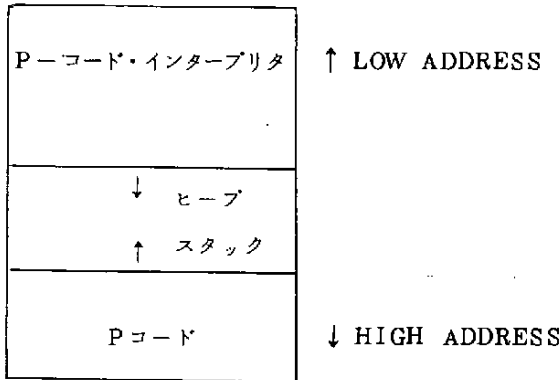
意味解析部では、構文解析部でチェックされた各文の意味づけを行う。意味づけされた文は、コード生成部で、Pコードに変換され出力される。

コンパイラの論理構成



(2) Pコード-8080インタープリタ

コンパイラから出力されるPコードは、仮想されたスタック・マシンの機械語である。従ってPコードを直接実行できる機械は数少く、ほとんどの機械では、Pコードを解釈実行する機械をプログラムで行わなければならない。このように、Pコードを解釈実行するプログラムをPコード・インタープリタと呼ぶ。



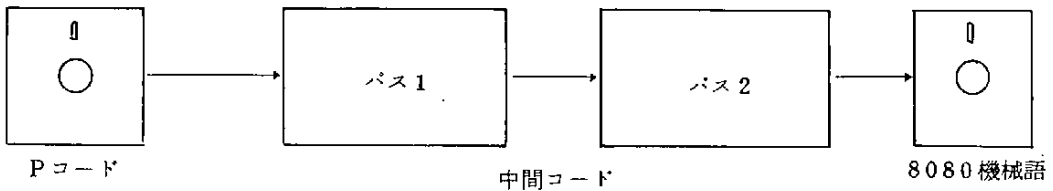
Pコード・インタープリタは、Pコード解説部と各Pコードに対応した実行部で構成される。解説部は、仮想マシンの命令カウンタ (IPC) の指すPコードを取り、そのコードに対応した実行ルーチンに制御を渡す。実行部は、そのコードに対応した機能を果し、制御を解説部に返す。

(3) Pコード-8080ネイティブコード トランスレータ

トランスレータは、Pコードを入力として、それに等価な8080機械語の並びを出力する。

トランスレータは、2パス方式をとり、パス1で飛び先番地テーブルの作成及び局所最適化を含んだ命令カウンタの計数を行い、中間コードを生成する。パス2では、パス1で生成された中間コードを入力として読み、番地修正などの加工を行い、8080機械語(ネイティブ・コード)を出力する。

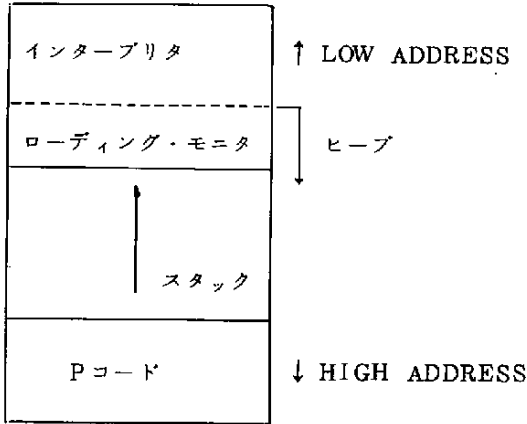
トランスレータで生成された機械語は、ランタイム・ルーチンを付加して実行可能となる。



(4) Pコード ローディング モニタ

ローディング・モニタは、CP/MのCCPとインターフェイスをとり、仮想スタック・マシンの初期化、Pコード又はネイティブ・コードのロードと、ランタイム・ルーチンの付加を行う。

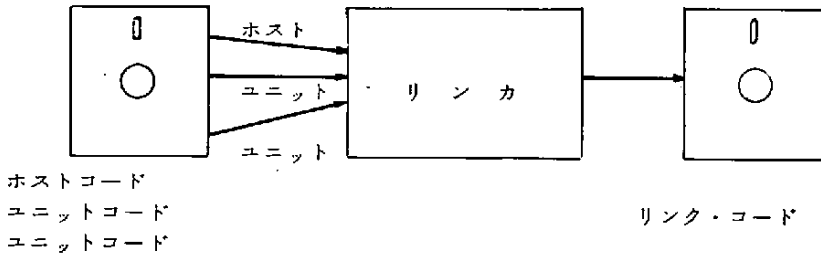
ローディング・モニタは、Pコード・インタープリタと同居しており、Pコード実行時(インタープリタに制御が移った時)に、フリー・エリアとして解放される。



(5) リンカ

リンカは、分割してコンパイルされた複数のプログラムを結合し、一本の完成されたプログラム (Pコード・ファイル) を生成する。

リンカは、分割コンパイルされた手続部分共通領域の参照オフセット、JUMPオフセット、プロセッサ・ディクショナリの書き替を行い一本のPコード・ファイルを出力する。



4.5 評 価

本コンパイラシステムの開発にあたり、必要とされる要件は次の6点であった。

- ① 言語仕様としては、標準PASCALの仕様である「PASCAL USER MANUAL AND REPORT by K. Jensen, N. Wirth」を基本とすること。
- ② PREFIX定義の機能を追加し、使用側でのルーチンを付加できること。
- ③ 標準関数、手続の一部は入口、出口を明確にし、使用側で改造可能とすること。
- ④ CP/Mに準じたOSの下で動作するシステムであること。
- ⑤ P-コードから8080ネイティブコードを生成するトランスレータを付加すること。
- ⑥ マイクロコンピュータソフトウェア開発に必要なユーティリティを可能な限り付加すること。

本システムの開発完了にあたり、上記要件を満足しているか否かを評価、検討した。

①については、テスト用に開発されたプログラムが上記の標準仕様をカバーしていること、それらのプログラムが正しく実行されていることを確認した。

②については、主プログラム側でUSE構文を使って付加されるプログラムで使用されるデータに関する宣言を行い、UNIT側のヘッダ部でこれと同じものを宣言させ、その本体部でプログラムを記述し、分離コンパイルした後、リンクする形式で実現された。この実現法は、PREFIX定義機能より強力であり実用上より好ましい形で実現されている。

③については、本コンパイラシステムを稼働させるハードウェアシステムにAm9511数値演算ユニットが組み込まれていることを前提として作成されたため、それを持たないシステムの利用者のためのドキュメントが付けられている。

④については、本コンパイラをCP/Mの1つの言語プロセッサとして稼働できるよう作成されている。

⑤については、仕様書に規定された通りP-コード列を8080の機械語に翻訳するトランスレータを作成した。トランスレータ自身はアセンブリ言語で記述されているが、プログラムの設計段階ではテーブルドリブン方式を基本として、他機種への移植性、使用者が経験的に見つけた局所的最適コード生成ルーチンの組み込みが容易になるよう十分考慮されている。

⑥については、④に関連して触れたように、CP/Mベースで稼働するため、CP/Mのユーティリティは全て利用可能であるが、PASCALコンパイラ用にスコープを考慮しないクロスリファレンスリスト作成プログラムが開発された。

以上の通り、開発されたPASCALコンパイラシステムは、仕様を十分に満足するものである。

また、実行効率性能面での評価は以下の通りである。構造文に伴う制限は、レキシカルレベル数8、プロセッサ又はファンクション数149、それらの各々のP-コードでのコード長1200バイトである。PASCALコンパイラ自身がこれらの制約内で表現されていることから、以上の制限は実用上問題とはならない。また、コンパイル時間に関しては、約4.5Kステートメント、セグメント数4のプログラム例で約16分であった。マイクロコンピュータ用のコンパイラとしては十分な性能

と評価できる。

P-コードレベルでの実行速度についても、現在市販されているUCSD-PASCALのそれよりも若干高速化されていることを確認した、

また、8080ネイティブコードへのトランスレータオブジェクトに関しては、コードサイズは3倍以内で、実行速度はP-コードインタプリタのその2～3倍高速化されている。特に、実際によく現われる添字やループカウンタの変分が比較的小さい場合の効率化が配慮されている。

今後、本システムが(財)日本情報処理開発協会マイクロコンピュータ振興センターの会員等に広く利用されることにより、マイクロコンピュータ応用システムの開発の効率の向上に大きく寄与するものと思われる。

— 禁 無 断 転 載 —

昭和 56 年 3 月 発行

発行所 財団法人 日本情報処理開発協会

東京都港区芝公園 3-5-8

機 械 振 興 会 館 内

TEL (434) 8211 (代表)

印刷所 株式会社 昌 文 社

住所 東京都港区芝 5-26-30

TEL (452) 4931

55-S003

