

マイクロコンピュータ用高級言語PL/Hシステム

High Level Language PL/H System for Microcomputer

マイクロコンピュータ応用システムの開発では、プログラムの開発に多大の労力を要する点が隘路となっている。そこで日立製作所は、ソフトウェアの生産性と保守性を向上させるために、高級言語PL/Hとそのコンパイラを開発した。PL/Hでは処理手続きを数式や短い命令文として書けるので、アセンブリ言語に比べ、記述量が60%減、プログラム開発工数が40%減となる。また、応用システムを実時間で動作させデバッグする実機デバッガASEをPL/H言語レベルで使えるようにした。この論文では、マイクロコンピュータ開発支援ソフトウェアでの高級言語PL/HとASEの位置づけ、及び適用効果について述べる。

吉村一馬* *Yoshimura Kazuma*
 渡辺 坦* *Watanabe Tan*
 西野秀毅* *Nishino Hideki*
 加藤正道* *Katô Masamichi*
 中野浩行** *Nakano Hiroyuki*

1 緒 言

最近、マイクロコンピュータの応用範囲の拡大に伴い、プログラムの作成量が飛躍的に増大してきている。一方、安価で使いやすい周辺LSIやメモリの開発も盛んに行なわれており、マイクロコンピュータを使ったユーザーシステム(以下、ユーザーシステムと略す)開発費中に占めるプログラム開発費の割合が大きくなってきた。これらを背景として、プログラムの生産性や保守性の大幅向上、プログラムの早期育成への要求が高まってきている。

マイクロコンピュータのプログラム開発には、アセンブリ言語が多く用いられてきた。しかし、これでは計算機の機械語と1対1対応に記述するためプログラムの生産性や保守性が低いばかりか、マイクロコンピュータの機種ごとに教育を必要とし、ソフトウェア標準化の障害にもなっていた。そこで、数式や名前を使った短い文の形でプログラムを記述できる高級言語PL/H^{1)~3)}とそのコンパイラを開発して、前述の要求にこたえることにした。

その開発基本方針は、次に述べるとおりである。

- (1) ソフトウェア生産性をアセンブリ言語比で40%以上向上させることを目標に、PL/Hシステムを開発する
- (2) PL/Hは、マイクロコンピュータの8ビット機に対する共通言語とする。また、上位機種の16ビット機にもこの仕様を含んだPL/Hを統一思想のもとに開発する。
- (3) 実機デバッガASE(Adaptive System Evaluator)で、ハードウェアも含めた総合テストをアセンブリ言語レベルだけでなく、PL/H言語レベルでも行なえるようにする。

2 システム体系

マイクロコンピュータHMCS6800のプログラム開発には、図1に示すように、レジデント・システムとクロス・システムが利用できる。レジデント・システムH68/SD20は、マイクロコンピュータ自身を用いてプログラムを開発するシステムである。同図のEMS(Executive Monitor System)は主に割込み処理、入出力処理及び基本的なデバッグ支援処理を行なう。FDOS(Floppy Disk Operating System)は、フロッピーディスクを用いたオペレーティング・システムであ

り、フロッピーディスク・ファイル関連の入出力操作や、ファイル管理などを行なう。RMS(Real Time Monitor System)はユーザーシステムで利用できる標準的実時間モニターであり、割込み処理や基本的入出力処理、マルチ・タスク管理機能がある。マクロアセンブラ及びPL/Hコンパイラは、番地指定が相対番地型の再配置可能な目的プログラムを出力する。リンケージ・エディタはこれらの幾つかの再配置可能な目的プログラムを結合して、絶対番地型の目的プログラムを出力する。リンケージ・エディタの出力をASEに与えれば、更に実時間でのデバッグができる。原始プログラムの作成と修正は、テキスト・エディタを用いて行なう。デバッグの完了したプログラムはH68/SD20に内蔵されているEPROM(Erasable & Electrically Programmable Read Only Memory)ライターによりEPROMに書き込まれる。

クロス・システムは大形計算機かミニコンピュータを用いてマイクロコンピュータのプログラム開発を行なうものである。そのアセンブラやPL/Hで出力される目的プログラムをシミュレータを用いて模擬的に実行し、デバッグする。デバッグが終了すれば、目的プログラムを紙テープなどに出力し、マイクロコンピュータに入力してテストする。

3 高級言語PL/H

3.1 PL/H言語の機能と仕様

インテル社のマイクロコンピュータ8080用高級言語PL/Mは、簡潔でかつシステム記述に適しており、世界的に広く使われているため、PL/Hの言語仕様はPL/Mの仕様をもとにし、機能拡張を図っている。その概要を表1に示すとともに、以下、幾つかの特徴について述べる。言語の構文は大形計算機の汎用言語PL/Iに類似しており、多くの基本的な実行文の構文形式はPL/Iの縮小形となっている。プログラムを分割して作るための機能や、明確に書けるようにする構造化プログラミング向きの機能をも備えている。一方、マイクロコンピュータ特有の機能も備えており、割込み制御や機械語の入出力機能により、オペレーティング・システムの助けを必要としないプログラムを作り出せるようになっている。

更に、バイト単位のきめ細かい操作のための豊富な機能が

* 日立製作所システム開発研究所 ** 日立製作所武蔵工場

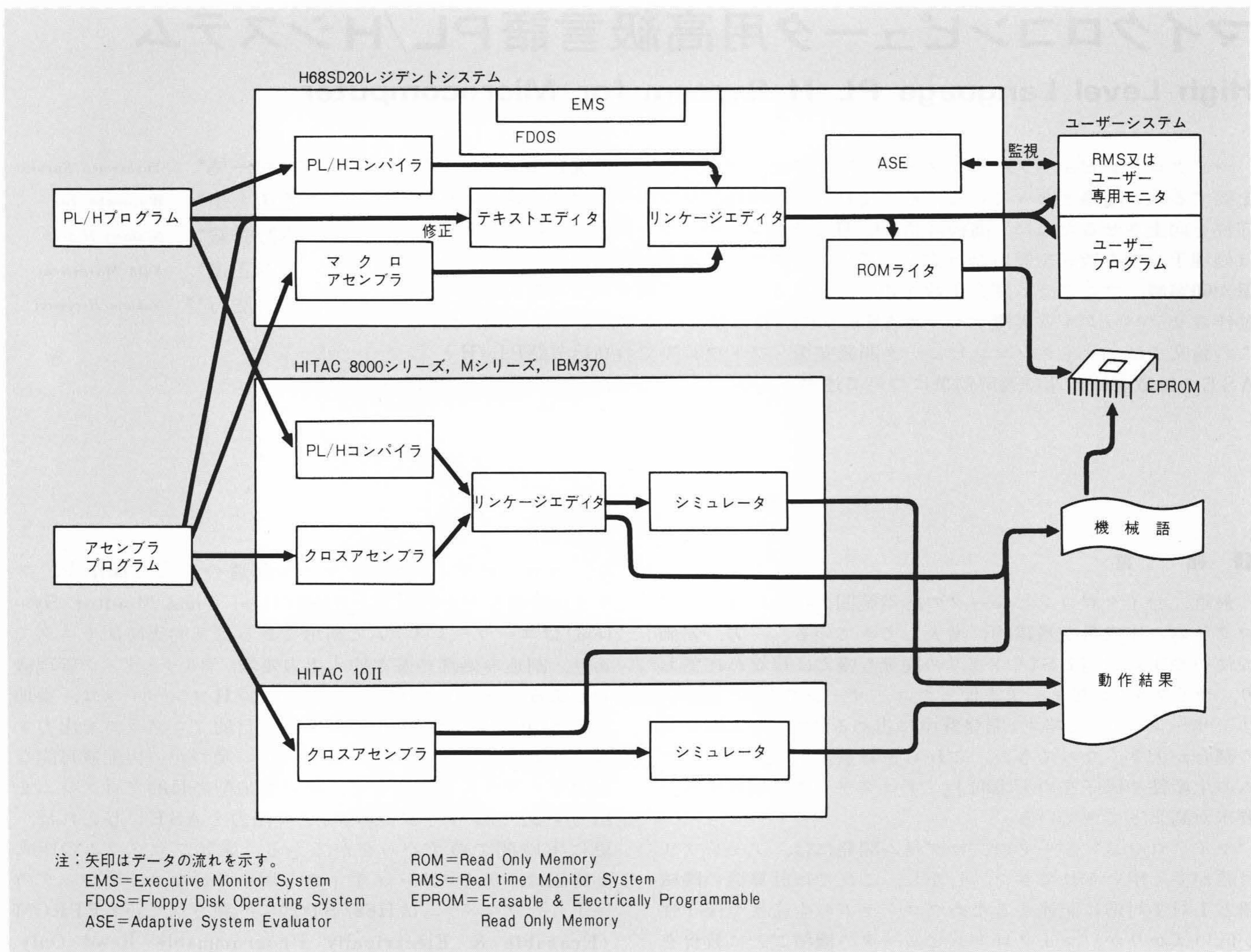


図1 ソフトウェア関連図 HMCS6800マイクロコンピュータに対するサポートソフトウェアの全容と、データの流れを中心としたプログラムの関連を示す。

表1 PL/H仕様 言語の仕様構成項目とその仕様概要を示す。

項目	仕様
定数	2進, 8進, 10進, 16進, 文字
変数	タイプは1バイト, 2バイトの符号なし整数
データ構造	単純変数, 配列[一次元], レベル1の構造体
演算機能	+, -, *, /, PLUS[キャリイ付+], MINUS[キャリイ付-], AND, OR, XOR, NOT, <, <=[≤], <>[≠], =, >=[≥], >
初期値機能	INITIAL[RAMエリア初期値], DATA[ROMエリア初期値]
変数の割付け機能	AT(絶対番地), AT(. 変数名), BASED属性
モジュール間共有属性	EXTERNAL属性[参照], PUBLIC属性[定義]
ラベル, 変数の有効範囲規定	ブロック構造, {DO ; ~END ;} {PROCEDURE... ; ~END ;}
代入文	通常の代入文の他に多重代入文[A, B, C, ..E = 式:], 及び埋込み代入文[A := 式:]
実行制御文	GOTO文, CALL文, RETURN文, 繰返LDO文, DO WHILE文, DO CASE文
IF文	IF ~ THEN ~ ELSE
割込み手続き	例 モジュール名: PROCEDURE INTERRUPT 1 ;
割込み制御文	DISABLE文[マスクビットセット] ENABLE文[マスクビットリセット]
組込み手続き	ハード依存機能 CARRY, ZERO, SIGN, OVERFLOW ROL[左回転], ROR[右回転], SCL[キャリイ含めた左回転], SHR[シフト], SHL, SAR LENGTH[配列要素数], LAST[配列要素数+1], SIZE[データ構造のメモリ占有量] LOW[2バイト数値の下位バイト切出し], HIGH[上位バイト切出し], DOUBLE[2バイト化] TIME[計測], MOVE[データ移動], INPUT[データ入力], OUTPUT[データ出力], DEC[10進化]
リテラル変数	コンパイル時にリテラル変数を定義された記号列に置き換える

注: [] はコメント

あり、メモリ効率の良い目的プログラムが作れる。

3.2 PL/Hコンパイラ

ユーザーシステムのコストには使用メモリ量の多少が敏感に反映されるため、プログラムのメモリ効率は多くの応用分野で重要視されている。したがって、PL/Hコンパイラは、極力効率の良い目的プログラムを生成する必要がある。一方、H68/SD20システムは、主記憶容量や外部記憶容量、入出力装置の性能などの面で厳しい制約を課されているので、このシステムの下で動くPL/Hコンパイラにはコンパクトさが求められる。これら二つの要求を適切に妥協させてバランスよく実現することが、このコンパイラの主要な設計目標となる。

最適化設計のための基礎データを収集するために、最適化効果の事前評価を行なった。その結果(図2)によれば、所期の目的プログラム性能を達成し、しかもコンパクトさを保つためには、最適化の処理単位を基本区間(制御の流入・流出点を内部にもたない最大の逐次実行命令系列)とするのが妥当である。文を単位とする局所的最適化では性能目標に達せず、フロー解析を伴うプログラム単位の大域的最適化ではコンパクトさを失う。最適化項目も、一部が同図に示されているように、期待効果の高い順から優先的に選択した。

こうした最適化処理の外に、コンパイラの使いやすさにも意を払った。例えば、最適化処理の実行選択、原始プログラムの自動字下げ印字指定などのコンパイラ制御オプションを豊富に設ける、エラー・メッセージはエラー番号だけでなく説明文形式で出力する、目的プログラムのリストは変数名などを用いアセンブリ言語形式で出力し、原始プログラムと併記して対応づけしやすくする、などである。

表2に、PL/Hコンパイラの開発結果を総合して示す。

4 実機デバッグASEにおける高級言語機能

ASEはハードウェアとソフトウェアを一体としたテストを可能にするものであり、必要ならばユーザーシステムにメモ

表2 レジデントPL/Hコンパイラ諸元 PL/Hプログラムは、リンケージ・エディタを介して他のPL/Hプログラムやアセンブリ言語プログラムと結合できる。

項	目	摘	要
動作環境	ハードウェア・システム	H68/SD20システム	
	所要メモリ量	RAM48kバイト	
	制御プログラム	FDOS	
入出力	入力	PL/H原始プログラム (フロッピディスク・ファイル)	
	ファイル出力	再配置可能型目的プログラム ASEデバッグ支援情報 (フロッピディスク・ファイル)	
	力印字出力	原始プログラム・リスト 目的プログラム・リスト(アセンブリ言語形式) 名前表リスト、クロス・レファレンス・リスト モジュール情報リストなど	
コンパイラ制御オプション	自動字下げ、最適化実行など19種類		
プログラム規模	PL/Hで記述して約23kステップ		
性能	コンパイル量	1回のコンパイル当たり最大1,500ステップ	
	コンパイル速度	印字時間を除き約80ステップ/分	

注：RAM = Random Access Memory

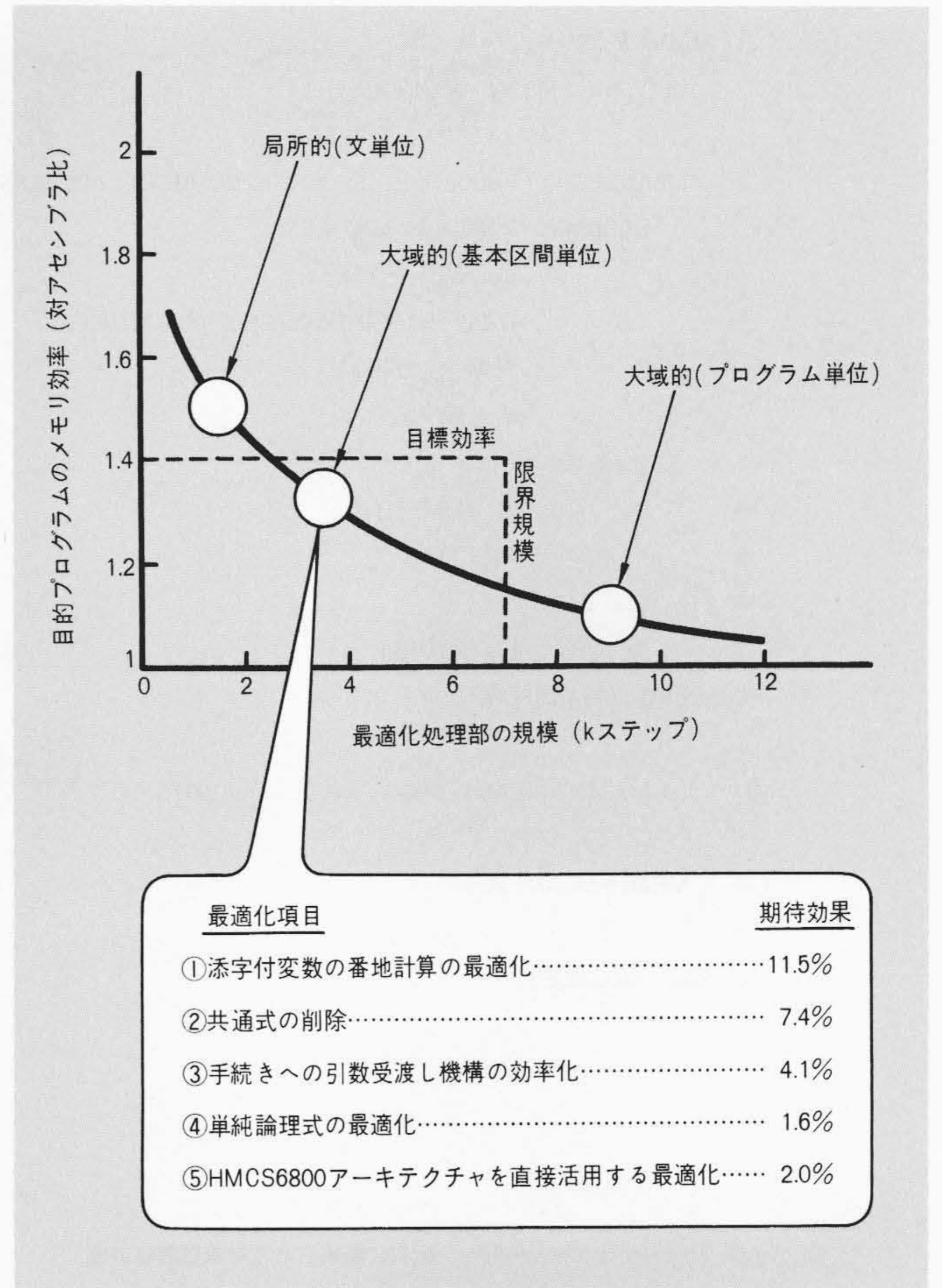


図2 最適化効果の事前評価 最適化処理単位を基本区間とするのが妥当である。最適化項目は期待効果の高いものから選択する。

リや入出力装置を貸し出して、それを実時間性を損うことなく作動させる。その際、テスト対象のPL/Hプログラムでの記述法を使って、

- (1) デバッグ操作の挿入位置(ブレークポイント)をPL/Hの文や句という論理的処理単位で指定すること。
 - (2) 実行再開位置の名札や文番号による指定
 - (3) 変数の値の表示と変更
 - (4) 文番号、名札、変数のメモリ番地の表示
 - (5) トレース情報の収集
- などのデバッグ操作ができる。

ASEはハードウェア部とソフトウェア部から構成される。ASEを使用するには、ハードウェア部を介してH68/SD20と開発中のユーザーシステムを直接に結合する。その後、H68/SD20の制御卓入力装置からFDOSのコマンドを入力してASEプログラムを起動し、ASEコマンドを用いて前述の機能などを実現する。図3にPL/Hプログラムの例と、それをデバッグするために用いられるASEコマンドの例を示す。

5 高級言語の適用効果

PL/Hを使用して多くのプログラムが開発された。表3は、その幾つかの開発結果から得られたPL/Hの適用効果をまとめたものである。開発工数、ステップ数については、12件、1,698ステップのPL/Hプログラムをアセンブリ言語で記述し直して比較算出したものであり、その後の数十キロステップのPL/Hプログラム開発でも、その効果が確かめられてい


```

1  MODULE : DO ;
    DECLARE MEAN ADDRESS,
           VALUE(100)ADDRESS ;
2  MEANSFUNC : PROCEDURE (POINTER, NUMBER) ADDRESS ;
    DECLARE POINTER ADDRESS,
           NUMBER BYTE,
           ITEM BASED POINTER (100) ADDRESS,
           SUM ADDRESS,
           I BYTE ;
3  SUM=0 ;
4  DO I=0 TO NUMBER-1 ;
5      SUM=SUM+ITEM(I) ;
6  END ;
7  RETURN SUM/NUMBER ;
8  END MEANSFUNC ;
:
:
100 L1 : MEAN=MEANSFUNC(. VALUE(0), 100) ;
:
:
110 MEAN=MEAN+20 ;
:
:
150 END MODULE ;
    
```

(a) PL/Hプログラムの例 配列に格納されている複数個の値の平均をとる関数型手続きMEANSFUNCを含んでいる。

ブレーク条件を設定するコマンドの例

- ① : BREK PC=. L1 ; A
 - ② : BREK PC=#100 ; B
 - ③ : BREK RANG #100<=PC<=#110 ; B
- 注 : ①L1を名札にもつ文を実行した直後にブレーク。
 セミコロンの次のAは「直後」を指示する。
 ②文番号100の文を実行する直前にブレーク。
 セミコロンの次のBは「直前」を指示する。
 ③文番号100から110までの範囲にあるいずれかの文を実行する直前にブレーク。

実行開始を指定するコマンドの例

- ① : GO
 - ② : GO . L1
 - ③ : GO #100
- 注 : ①現在のプログラム・カウンタのまま実行開始。
 ②L1を名札にもつ文から実行開始。
 ③文番号100の文から実行開始。

変数の値、番地を表示、変更するコマンドの例

- ① : . NUMBER/mmmm (LF) mmmm/nn
- ② : PRNT
 BEG ×××× . VALUE(0)
 END ×××× . VALUE(99)
 EXEC Y

- 注 : ①mmmmは変数NUMBERの番地, nnは値。
 ②配列VALUEの要素の値をすべて表示する。
- (b)ASEコマンドの例 (a)のプログラムをデバッグするのに頻繁に使われるコマンドを示している。

図3 PL/HプログラムとASEコマンドの例 ASEの特徴の一つはPL/HプログラムをPL/Hの記述水準でデバッグできる点にある。これによりテスト効率の大幅な向上を期待できる。

る。目的プログラムの性能は、5件1,137ステップのPL/Hプログラムをアセンブリ言語で記述した例から算出したものである。

ASEは、ユーザーシステムの実時間デバッグを機械語のレベルでなくPL/H言語のレベルで行なえるようにし、使い勝手を大幅に向上させた。また、メモリや入出力装置の完備していない試作品レベルでもデバッグや評価試験を行なえるようにした。

表3 PL/Hの適用効果とコンパイラの性能 下記数値は、いずれも対アセンブリ言語比である。

項	目	数 量
PL/H適用効果	開発工数	40%減
	ステップ数	60%減
	習得期間	70%減
目的プログラムの性能	メモリ量	1.2~1.7倍
	実行時間	1.2~1.6倍

6 結 言

マイクロコンピュータ用プログラムの生産性と保守性を向上させるために、高級言語PL/Hとそのコンパイラを開発し、また、実機デバッガASEでも高級言語でのデバッグを可能にした。

PL/H言語は、アセンブリ言語に比べて目的プログラムのメモリ量と実行時間が共に約1.4倍となる。しかし、プログラム開発工数は40%減、記述量は60%減及び習得期間は70%減となるという効果が得られた。このことからPL/Hは、メモリ量や実行時間の制約の緩い少量生産品や仕様変更の多い製品に向いているといえよう。またASEに高級言語機能を組み込むことにより、ユーザーシステムのハードウェア、ソフトウェアを込めた実時間でのテスト・デバッグ作業をPL/H言語のレベルで効率良く行なえるようにした。

参考文献

- 1) 株式会社日立製作所：日立マイクロコンピュータシステム PL/H言語マニュアル (昭54-3)
- 2) 吉村, ほか5名：マイクロコン用システム記述言語PL/Hコンパイラの開発思想, 情報処理学会第19回全国大会論文集, 133~140 (昭53-8)
- 3) 加藤, ほか6名：マイクロコン用高級言語PL/Hレジデントコンパイラの開発, 電子通信学会講演論文集, 1473 (昭54-3)