

KOCOSの知的インターフェースの 機能と構成

上林憲行 竹山明 徳田英幸 瀧塚博志 (慶大工学部)
西垣秀樹 (沖電気工業(株))

1. はじめに

ミニコンピュータコンプレックスにおける重要な課題として、各構成要素のミニコンピュータが持っている潜在的な能力を適切に引出し、いかにシステム全体へ反映させるといふ事があげられる。^{[1][2]}

KOCOS (Keio-Okii's Complex System) では、この課題に対し、方式設計時において、特に効率が良く可変性のあるプロセス間通信機能の構築、およびその実現方法に検討が加えられた。^{[3][4]} その結果、各構成要素間のインターフェースユニットとしてプロセス間通信機能をサポートする高度な知的管理機能を持ち、制御中枢部をマイクロプロセッサで構成したBIU (Bus Interface Unit) を設計した。中核をなすマイクロプロセッサの決定に関し、十分な処理スピードを持ち、システムコンポーネントとして効率良く適用できるユーザマイクロプログラマブルのものを選択対象とした。そこで、当時ファミリーシリーズとして種々のチップがすでに発表されていたIntelシリーズ3000を採用した。

現在、BIUはプロトタイプKOCOSで稼動中であり、プロセス間通信機能における重要な役割を果たしている。

本論文では、BIUにおける各構成モジュールの機能、制御中枢部を構成しているマイクロプロセッサアーキテクチャ、および論理的機能を実現している制御プログラムに関して報告する。

2. システム概要とBIUの位置

KOCOSは慶応義塾大学工学部矢上台キャンパスに設置されているミニコンピュータを接続した異種ミニコンピュータコンプレックスである。プロトタイプKOCOSのシステム構成はFig. 1に示されるように、OKITAC-4300C (2台)、NOVA-1200 (2台) とHITAC-10マルチシステムから構成され、各エレメントプロセッサ(以下EP)間でのハードウェア、ソフトウェアリソースの共用、および並列処理の研究を目的とした実験研究用システム^{[5][6][7]}がある。

システム内における基本的通信は、“プロセス”^{[8][9]}間で行われる。送信プロセスのアドレス空間から受信プロセスのアドレス空間へのデータメッセージ(以下DM)の通信や、プロセス間通信制御メッセージ(以下IPCM)によるプロセス間での動的なコオペレーションが実現されている。^[10] これらのプロセス間通信機能に必要な複雑な処理モジュールをEP上のローカルオペレーティングシステム(以下LOS)から分離し、BIUが通信制御することによって、結合による負荷を軽減している。BIUは、各EPと単一共通バス(C-Bus)間の単なる物理的インターフェースとしてだけでなく、コンプレックスシステムの性能を大きく左右するプロセス間通信における論理的インターフェースとしての知的管理機能を提供している。

従って、BIUにおいてプロセス間通信をサポートしている制御プログラムは効率が良く、信頼性、拡張性に対処し得るものが必要であり、実行するマイクロ

プロセッサの能力としては、プロセス間通信に付随する仕事に対して必要な基本的実行スピードと、その仕事に適した命令セットやアドレッシング機能を備えていることが必須条件であった。

以上の条件から、次の方法でBIUにおける知的制御機構を実現している。

- [1] 制御プログラムのアルゴリズムの決定
- [2] Intelシリーズ3000におけるマイクロ命令セットの決定
- [3] 制御プログラムに適したマクロ命令セットの決定(機械語に対応する)
- [4] 制御プログラムのrefinement, 問題向きマクロ命令のチューニング
- [5] サポートソフトウェアを利用したのインプリメント

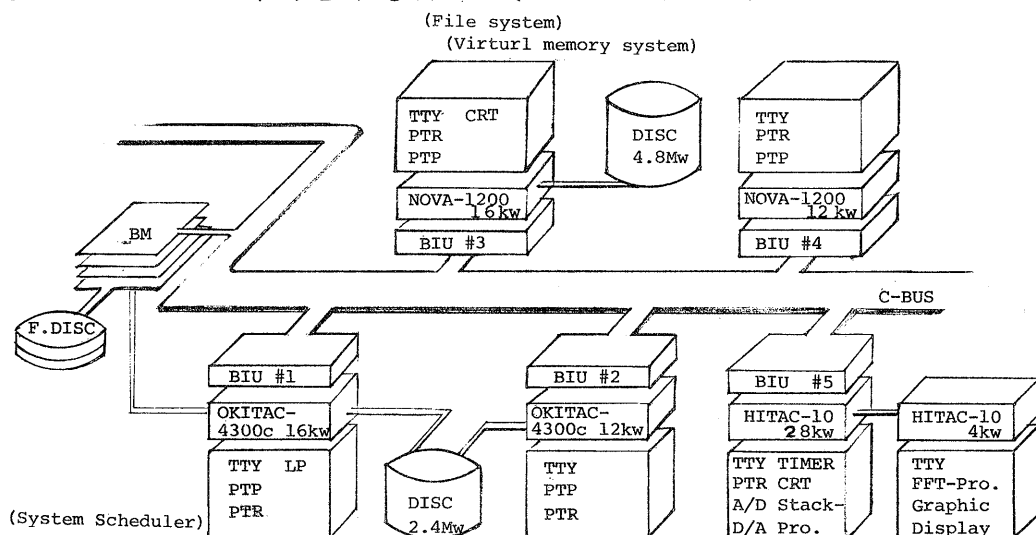


Fig. 1 System configuration of KOCOS

3. BIUのハードウェア構成^{[11][12]}

BIUの主な機能は、①プロセス間通信における論理的リンクの確立、②プロセス間通信の制御、③転送時のエラーの処理である。Fig. 2はBIU装置でありFig. 3はBIUの構成を示し、以下に各構成モジュールについて述べる。さらにマイクロプロセッサの詳細を5章で述べる。

3.1 BIU構成モジュール

(1) CMB (Communication Buffer)

EPからBIUに起動をかけるStart 功(SIO)やEPからBIUへの処理要求を知らせるListem 功(LIO)時に、BCW (BIU Control Word)アドレスやLIOパラメータがセットされるバッファである。

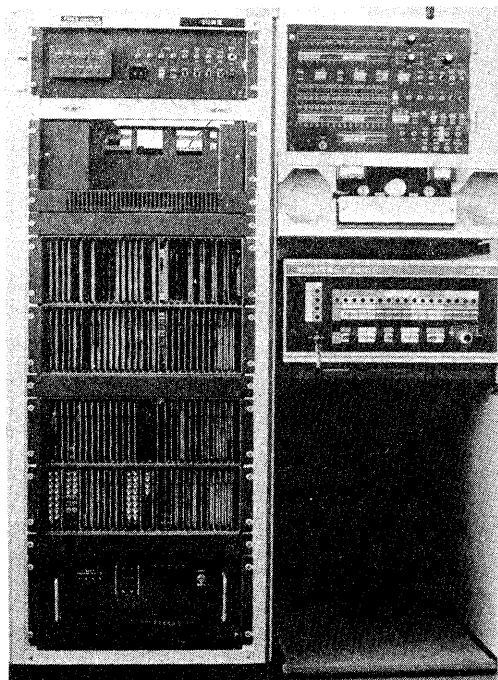


Fig. 2 BIU装置

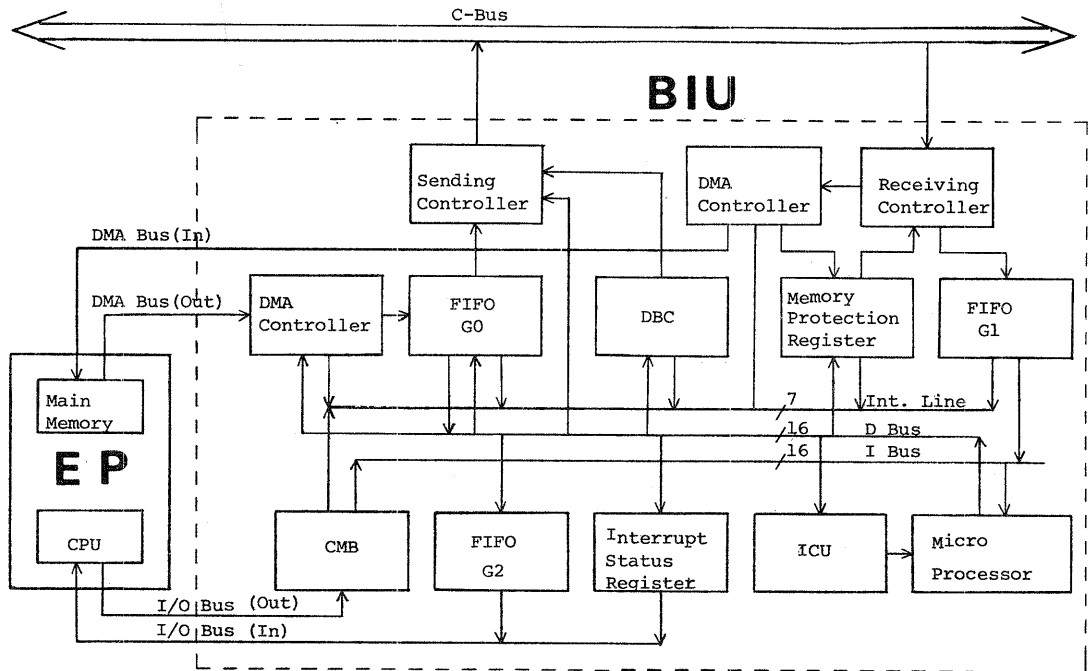


Fig. 3 Configuration of BIU

(2) FIFO G0

FIFO G0は、①BCW Read 時の入カバッファ、②IPC M転送時の出カバッファ、③EPのメモリからC-BusへのDM転送時の出カバッファとして使用される。FIFO G0は、2ビットのタグを持っており、これがC-Bus転送時の制御信号となる。

(3) Sending Controller

C-Bus上のデータ転送の制御、および受信側BIUの異常検出を行う。

(4) DBC (Distributed Bus Controller)

完全分散制御によるバスコントローラであり、ラウンドロビンを可能にしている。

(5) Receiving Controller

受信データをIPC MとDMに分類し、IPC Mの場合はそれをFIFO G1にセットし、DMの場合はメモリアドレスレジスタにアドレスをセットしてEPにDMAをかける。

(6) Memory Protection Register (Upper and Lower)

C-BusからのDM転送時に書き込み禁止領域を保護するためのレジスタである。

(7) FIFO G1

C-Busからの受信IPC Mの入カバッファである。

(8) Interrupt Status Register, FIFO G2

Interrupt Status Registerは、BIUからEPへの割込みステータスを示す。下位8ビットはマイクロプロセッサがセットし、EPへの動作要求を示す。その時の詳細情報はFIFO G2にセットする。上位8ビットはDMA時のエラー等のハード的要因によりセットされる。

3.2 プロトタイプBIU

現在、OKITAC-4300C用のBIU2台が完成して、稼動中である。プロトタイプBIUは、マイクロプロセッサが41×28[cm]の基板1枚で、他の回路は24×14[cm]の基板90枚で実装されている。使用IC数をTable1に示す。基板数、IC数とも非常に多いが、これはプロトタイプであるためマイクロプログラムとBIU制御プログラム(BCP)のデバッグ用の保守パネルインターフェイスを付加したため、現在製作中のNOVA用のBIUでは大型基板5枚で実装する予定である。^[13]

用 途	LSI	MSI	SSI
マイクロプロセッサおよび周辺回路	15	21	160
メモリおよび周辺回路	38	2	38
通信制御回路および周辺回路	6	89	312
EP-BIUインターフェイス	0	32	176
コントロールパネルインターフェイス	0	2	331
計	59	146	1017

Table 1 BIUの使用IC数

4 BIU, EP内の論理フロー

EP内のユーザプロセスAが発した転送要求は、LOSを介し、BCPによって転送の制御が行われる。この時の論理フローをForce Write (FWRITE)を例にとりFig.4に示し、以下図中の番号にそって詳細を述べる。

1. ユーザプロセスAが、システムコールFWRITEを実行する。
2. LOSはBCPにLIO, SIOを発しBCWアドレスを送る。BCPはCMBからそれを読み込む。
3. BCPはEPからDMAでBCWを読み込み、未処理IPCMキュー(送信用)にキューイングする。LOSはプロセスAをwait状態にする。
4. BCPは未処理IPCMキューからIPCMをIPCMセンディングキューに移し、またランデブーテーブルに登録する。
5. FWRITEを相手プロセスに送り、IPCMセンディングキューからDMセンディングキューにIPCMを移す。
6. 相手プロセスからの返事であるWrite Answer (WANSWER)が未処理IPCMキュー(受信用)に入る。
7. WANSWERとランデブーテーブルのマッチングを行う。
8. マッチしたならDMを相手プロセスに送る。
9. DMが正常に送信できたなら相手プロセスにFinal Information Message (FIM)を送る。
10. DM送込キューからIPCMを削除する。
11. LOSに割込みをかけ転送終了を知らせる。
12. LOSはプロセスAをready状態に戻す。

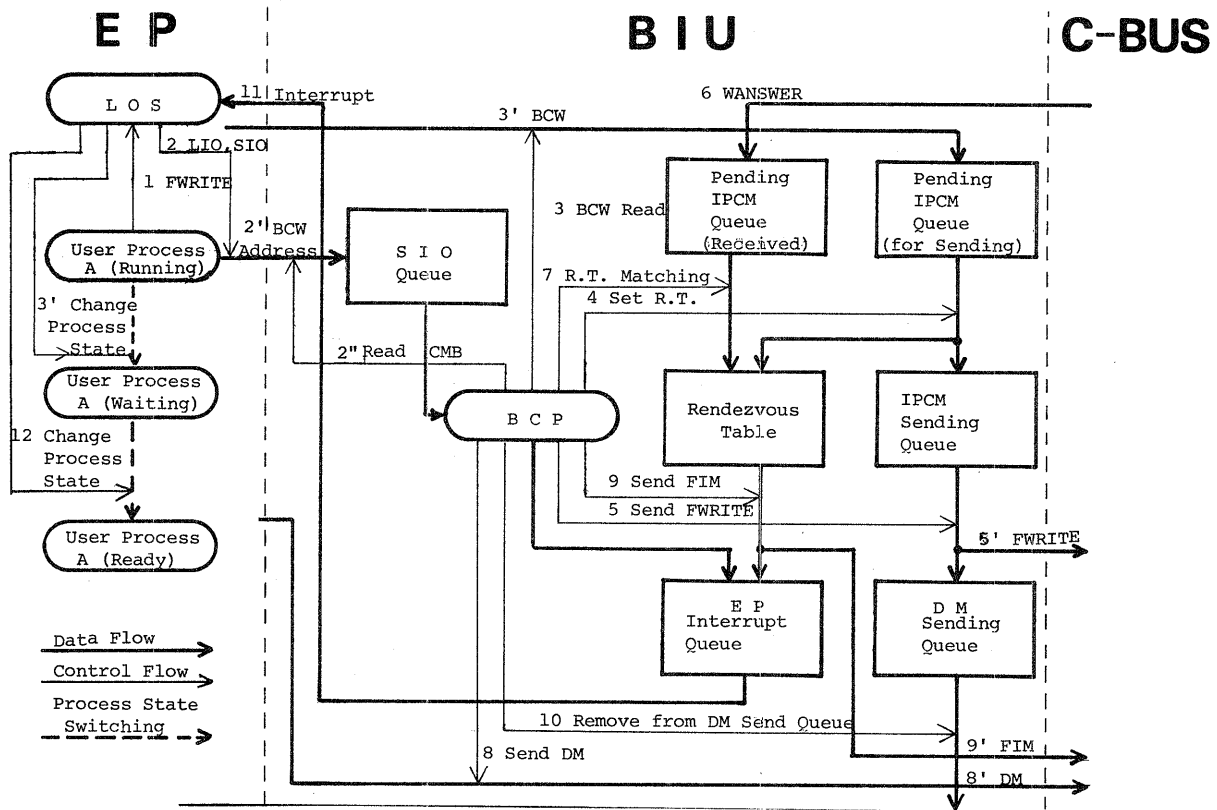


Fig. 4 An example of relation between LOS and BCP, logical flow in BCP and process state switching. (FWRITE - WANSWER)

5. マイクロプロセッサ-アーキテクチャと制御プログラム

BMP (BIU Microprocessor) は BIU 全体を制御するために、ユーザマイクロプログラマブルな Intel 3000 ファミリー^{[14][15]} を中心に構成された専用マシンである。

(1) マイクロプロセッサの構成

BMP の構成を Fig. 6 に示す。またマイクロプロセッサの実装基板の写真は Fig. 5 である。

MCU はマイクロプログラムの次アドレスの決定とフラグ、キャリーの管理を、2 ビットスライスの CPE は算術論理演算機能を遂行するユニットである。LACG は高速にキャリーを伝播させる。ICU は外部からの 8 レベルの割込みを制御する。

マイクロプログラムメモリは、アクセス時間 70ns のバイポーラ PROM 3604 を 4 個使用している。マクロメモリはアクセス時間 60ns のバイポーラ RAM 3106A 512 w と、約 1KW の PROM で最終的には実装する予定である。

現在は 1.25KW の RAM で動作している。以上のコンポーネントは LSI チ

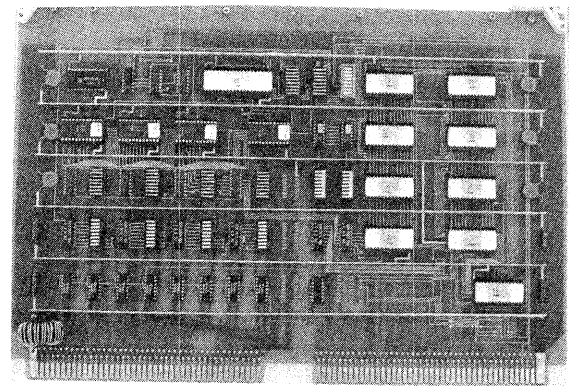


Fig. 5 BIU Microprocessor Board

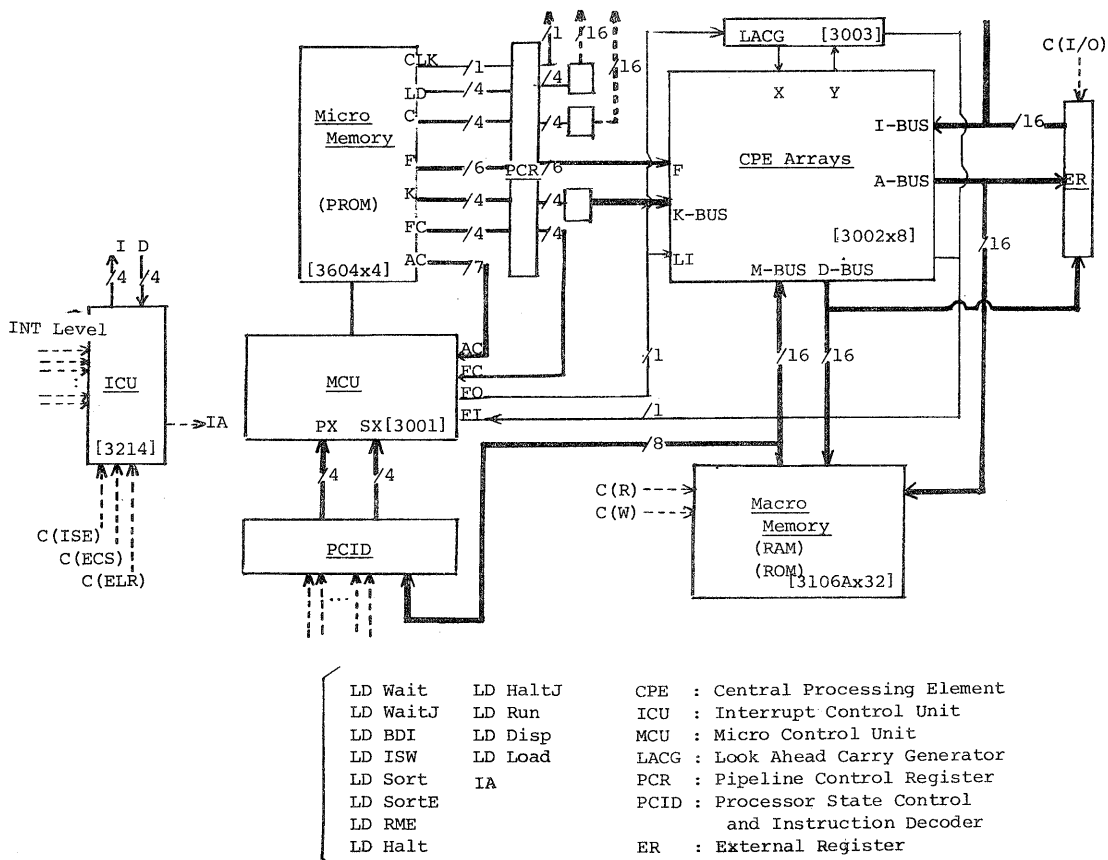


Fig. 6 BIU Microprocessor

ツプであるが、周辺インタフェース回路のPCIDはSSIとMSIで設計されている。PCIDはMCU、ICUを支援しマプロセッサの状態の管理、保守パネルからの制御、フェッチステージでのマクロ命令の解析を行う。またPCIDは制御信号を解釈しマPX、SXバスを用いママイクロ命令分岐情報をMCUに与える。またPCRを置くことによりマプロセッサはマイクロ命令レベル(マイクロフェッチ、マイクロ実行)での2ステージのパイプライン制御を実現し、高速化を計った。マイクロ命令サイクルは6MHz(166ns)で設計されている。

(2) マイクロ命令

BuIにおけるBuI(BIU Micro Instruction)の仕様をFig.7に示す。

BuIはIntel 3000の標準マイクロ命令フィールド[AC, FC, F]とユーザ定義フィールドから構成されており命令長は32ビットである。ACフィールドはマイクロメモリのメモリ空間が2次元アドレス方式であるために特異な次元アドレス制御を行っている。^[16] 多様な分岐機能、マクロ命令のコードの解釈、多岐分岐などは優れた点であるがマイクロサブルーチン機能が無に等しいこと、条件分岐の分岐先アドレスが限定されていることなどが不便な点であった。FフィールドはCPEへのマイクロファンクションであり、基本的には8つの論理動作を規定できるがKバスのパターンにより多様なマイクロファンクションが生成される。従って、Kバスのパターンの選択で効率の良い問題向きマイクロファンクションを設定できる。Kバスフィールドはマイクロファンクションの論理操作に際して、定数生成や、ビットマスク等のビット操作、

Instruction Length : 16 bit
Register : A,B,C,X,SP,CS,PC,ISW0,ISW1
Addressing Mode : Direct,Indirect,Index,Index and Indirect,Relative addressing
Special mode [ROM 0 page (512-767) direct,etc]

Instruction Format

Operation Field (Bit 15 - 8)
15 bit : 0 = Non memory reference in fetch stage, 1 = Memory reference in fetch stage
13 bit : 0 = Direct addressing mode, 1 = Indirect addressing mode
12 bit : 1 = Index
Operand Field (Bit 7 - 0)

Instruction Set

Standard Instruction

LOAD (6), LOAD IMMEDIATE (5), MEMORY INCREMENT/DECREMENT (2), STORE (4), MOVE (2),
MOVE REGISTER (14), SHIFT (2), BRANCH (4), SUBROUTINE CALL/RETURN (5), CONTROL (7),
TEST (12), COMPARE (1), ADD (5), ADD IMMEDIATE (4), SUBTRACT (5),SUBTRACT IMMEDIATE (4),
OR (2), AND (4), EXCLUSIVE OR (2), NOT (1), I/O Instruction (3)

Problem Oriented Instruction

<u>Data Structure</u>		<u>Control Structure and Bit Sort</u>	
CHANGE HEAD ADDRESS	[10]	CHECK LEVEL	[10]
CHANGE TAIL ADDRESS	[12]	BIT SORT ANALYZE	[1+9*]
TEST MEMORY LEVEL	[13]	ERROR STATUS READ/ANALYZE	[6+9*]
<u>Rendezvous Table</u>		SEND COMPLETE ANALYZE	[1+9*]
RENDEZVOUS TABLE SEARCH	[26]	<u>FIFO and I/O</u>	
<u>Internal Status Word</u>		FIFO GO WRITE	[1+20**]
TEST ISW	[10]	FIFO G1 WRITE	[1+20**]
RESET ISW	[7]	FIFO READ	[14]
SET ISW	[5]	SET DMA CONTROLLER	[6]
<u>Etc</u>		HEADER GENERATE I/D	[12]
INTERRUPT RECOVER	[27]		
WORD LENGTH GENELATE	[4]		
LOAD CONTINUE	[6]		

() = a number of macro instruction
[] = a number of micro instruction in execution stage
*,** = a number of micro instruction for common use

Table 2 Macro Instruction

Power On and Initialize module	(INT7)	ラム化して高速処理を実現している。Basic ループと呼ばれる処理モジュールのスケジュー リングによって、BCW Read 起動、IPCM転 送、DM転送、EP割込み、ヒモ処理IPCM 解析ルーチンは各々の処理キューのエントリ を確認し処理する構造となっている。各処 理キューについてはFig.4に示してある。 BCPではBIUの内部状態を表現して いるステータスビットを24ビット持ってお り、共有リソースに対する競合を防いでい る。内部処理の効率良いスケジューリ ングや信頼性の高い動作を実現しており、 このステータスビットに関連した有効な命 令を設定している。
LISTEN I/O handler	(INT6)	
START I/O handler	(INT5)	
Error handler	(INT4)	
Transfer end handler	(INT3)	
IPCM Receive handler	(INT2)	
Clock Interrupt handler	(INT1)	
Basic Loop	(LEVEL0)	
BCW Read Start module		
IPCM Transfer module		
DM Transfer module		
EP Interrupt module		
Queueing IPCM handler		
WRITE handler		RT Cut module
READ handler		
WRITE ANSWER		
READ ANSWER		
FORCE WRITE		
FORCE READ		
Control type		

Table 3 BIU Control Program Module Chart

BCPの基本的な処理方式は、外部から一
方的にくる処理要求だけを迅速に済ませ、BCPが主体的に最適時点を捕捉して
処理を行う。従って、BCPは単なる受動的なオンラインリアルタイムプログラムでなく
主体的に処理を遂行していく性格を有している。またBCPのモジュールチャートに
みられるように、BIUに課せられた多様な処理要求がBCPにおいて仮想化され
るので、内部フローが単純化されているのが特徴である。

Table.4. Comparative table of program execution times and data storage requirements

PROGRAM SIZE (DATA) (INSTRUCTION) (TOTAL)	BMI			NOVA				PASCAL			
	p1			p2		p2/p1		p3		p3/p1	
	1			33		33		4		4	
	63			168		2.7		384		6.1	
	64			201		3.1		388		6.1	
EXECUTION	exec-step			exec-time (usec)				exec-step		exec-st-line	
	e1	e2	e2' *1	e3	e3/e1	e4	e4/e2	e5	e5/e1	e6	e6/e1
INT. ROUTINE (28)*2	4.7	7.0		43	--	121.1	17.2	55	--	1	--
INT. ROUTINE (TOTAL)	1(34)	5.7	8.5	17	17	42.1	5.0	58	58	1	1
	1(62)	10.4	15.5	60	60	163.2	15.8	113	113	2	2
ROUTINE [A]	19	53.4	80.1	88	4.6	284.1	5.3	176	9.3	17	0.9
ROUTINE [B]	4	14.7	22.1	38	9.5	76.3	5.2	23	5.8	8	2
ROUTINE [C]	3	6.3	9.5	27	9	39.1	6.2	16	5.3	6	2
ROUTINE [D]	12	36.7	55.1	84	7	127.3	3.5	42	3.5	10	0.8
ROUTINE [E]	9	19.4	29.1	59	6.6	99.4	5.1	36	4.0	10	1.1
ROUTINE [F]	6	18.5	27.8	27	4.5	60.9	3.3	20	3.3	7	1.2
(AVERAGE)	8.8	24.8	37.2	53.8	6.1	114.5	4.6	52.2	5.9	9.7	1.1
INT. LEVEL 3	9.8	35.2	52.8	113.8	11.6	277.7	7.9	165.2	16.9	11.7	1.2
MEMORY*SPEED RELATIVE RATIO	1			24.8				---			

NOTE: 1. E2 shows the relative exec-time to the instruction LDA of NOVA 1200.
2. (n) shows the number of micro instruction steps.

<制御プログラムの評価>

現在のBCPはプログラムサイズがデータエリアを含め約750Wで構成されている。各モジュールの処理速度は約20~60μsである。BCPの評価のためにBCPのINT3のモジュールに関してBMIと同様な1語命令のNOVA 1200⁽¹⁷⁾の命令セットを用いて、記述した場合と、高級言語Pascal⁽²⁰⁾を用いて記述した場合との比較をTable.4に示す。NOVA 1200に比べて、プログラムサイズで3.1倍、平均実行速度で約7.9倍、メモリスピード積が24.8倍すぐれている。BCPのステップ数とPascalにおける実行ステートメント数を比較しても約1.2倍であり、各問題向き命令にセマンティックが良く吸収されており、BMIが記述性の高い命令セットを提供していることがわかる。また、処理速度においても非常に高速な処理が実現できたと思う。

制御プログラムを高速かつコンパクトに製作できた要因としては、①BMIの高い問題適応性、②割込み時のプロセッサステート等のセーブ/リセーブのマイクロ化、③メモリスizesとスピードのバランスのとれたデータ構造の適用などがあげられる。このようなアプローチにより設計当初の目標であった高速処理とプログラムのコンパクト化が達成されたと思われる。

(5) BMPサポートソフトウェアシステム

マイクロプロセッサの低価格に比べてソフトウェア開発に多大のユーザ負担があり問題とされた。BMPの開発にあたってはBCPの作成に要する労力を軽減できるように配慮した総合的なサポートソフトウェアシステムを作成した。サポートソフトウェアシステムの構成をFig. 8に示す。

• BuL (BIU Microprogram Language processor)

BuLはIntel 3000用の汎用マイクロアセンブラであるCROMISに準じた仕様となっている。BuLはBuIで記述されたプログラムをアセンブルして所定のオブジェクトを生成する機能を有している。BuLは、BuL-A (BuL-micro Assembler)とBuL-D (BuL Define Instruction)の2つのモジュールから構成され、後

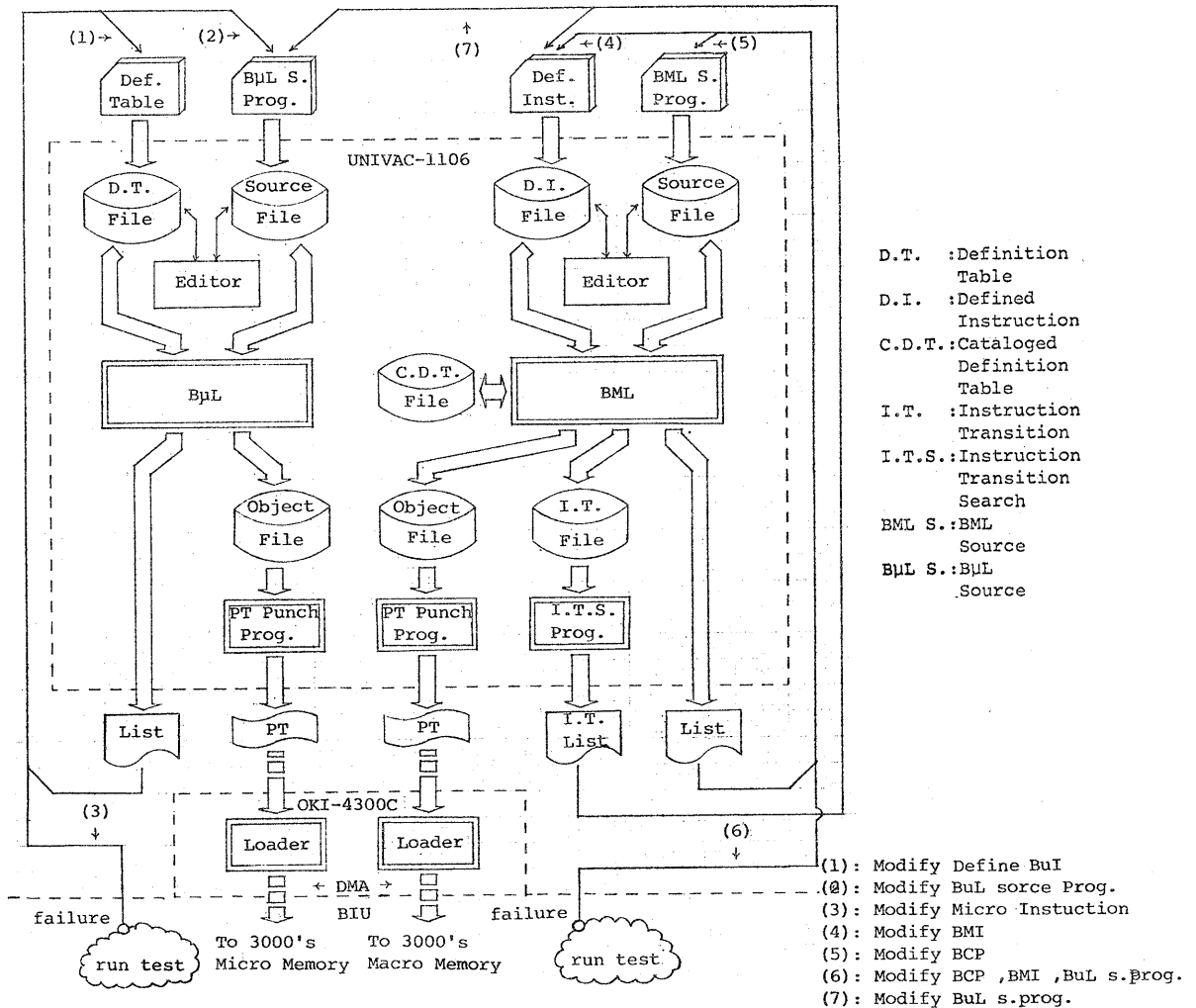


Fig. 8 BIU microprocessor Support Software System

者はマイクロ命令を定義して内部テーブルを作成する。前者は内部テーブルを参照してオブジェクトを生成する。BuLは特異なアドレスモードをチェックする機能、2次元アドレスマップイメージ等のマイクロプログラム開発支援機能を合わせて付加している。

・ BML (BIU Macro Language processor)

BMLはBuIで設定したマクロ命令を定義し、BMIで記述されたBCPのオブジェクトを生成するアセンブラである。問題向きの高水準命令を設定する過程のフィードバックに際して、本質的に派生する命令の再定義、追加、修正といったことに対応するために命令セットの定義を自由に訪ず汎用アセンブラに準じた形式をとっている。BMLではオプションとして、マクロ命令の静的出現頻度、3命令先までの静的な命令遷移情報などのBMIの設定のための評価システムがあり、有効な資料となった。

BuL, BMLはともにUNIVAC-1106システム上にFORTRANで作成されている。

謝辞

本システムの設計・製作にあたり多大なる御支援、御協力頂いた沖電気工業株式会社の方々に深謝致します。またサポートソフトウェアの開発にあたり御援助頂いた本学情報科学研究所の方々に深謝致します。日頃から熱心に御指導頂いている本学 相磯秀夫教授、および相磯研究室の皆様には感謝致します。なお制御プログラム、サポートソフトウェアの開発にあたっては松尾繁樹君(現四国電力)、福島知孝君(現富士通)の協力が尤であった。

参考文献

- [1] 相磯 : マイクロコンピュータのアーキテクチャとシステム構成, 情報処理, VOL-17, NO4, 1975.
- [2] 村上 他 : コンピュータコンプレックス(小特集), 情報処理. VOL-16, NO7, 1974.
- [3] 清水 他 : KOCOS (Keio-Okio Complex System)の方式研究, 昭和50年電子通信学会全国大会講演論文集, 1975.
- [4] 上林, 徳田, 竹山 他 : KOCOSのアーキテクチャ(1), (2), 情報処理アーキテクチャ研究会資料, 1975年7月.
- [5] H. Aiso, et al., : A Minicomputer Complex - KOCOS, Proc. 4th Data Communication Symposium, Oct. 1975.
- [6] H. Aiso, et al., : A System Software for KOCOS, Proc. IFIP TC-2 Working Conference, Sept. 1975.
- [7] 徳田, 相磯 : KOCOSにおけるプロセスの同期操作について, 昭和51年信通学会全国大会予稿, 1976.
- [8] E. W. Dijkstra, : Cooperating Sequential Processes, Programming Languages, Academic Press, 1968.
- [9] D. C. Walden, : A System for Interprocess Communication in a Resource Sharing Computer Network, CACM, Vol.15, No.4, Apr. 1972.
- [10] IPCF (Interprocess Communication Facilities) Reference Manual, KOCOS 内部資料, 1975.
- [11] BIU Reference Manual, KOCOS 内部資料, 1975.
- [12] BIU Hardware Handbook, KOCOS 内部資料, 1975.
- [13] NOVA BIU Design Notes, KOCOS 内部資料, 1975.
- [14] J. Rattner, J. Cornet, M. E. Hoff, : Bipolar LSI Computing Elements Usher in New Era of Digital Design, Electronics, Sept. 5, 1974.
- [15] Intel Serise 3000 Reference Manuals, Intel Co., 1974.
- [16] L. H. Lones, : Instruction Sequencing in Microprogrammed Computers, Proc. NCC, 1975.
- [17] 坂村, 相磯, 飯塚 : ダイナミックマイクロプログラミングによる最適命令セットの合成手法に関する考察, 信通学会 電子計算機研究会資料, EC75-28, Sept. 1975.
- [18] A. M. Abd-Alla, D. C. Karlgaard : Heuristic Synthesis of Microprogrammed Computer Architecture, IEEE Trans. Comp., Vol.C-23, 8, Aug. 1974.
- [19] Guide to NOVA 1200 Series, Data General Co.
- [20] K. Jensen, N. Wirth, : PASCAL-User Manual and Report, Springer-Verlag, 1974.