

## 映像機器におけるマイクロプロセッサの構造と応用 に関する研究

メタデータ	言語: ja 出版者: 静岡大学 公開日: 2014-09-09 キーワード (Ja): キーワード (En): 作成者: 山田, 罔裕 メールアドレス: 所属:
URL	<a href="https://doi.org/10.14945/00007903">https://doi.org/10.14945/00007903</a>

理工学研究科山E

GD

K

0002530814

R

324

静岡大学附属図書館

静岡大学 博士論文

映像機器におけるマイクロプロセッサの構造  
と応用に関する研究

2002年6月

大学院理工学研究科  
設計科学専攻

山田 罔裕



# 目次

論文要旨	iii
図表一覧	vi
1. 序論	1
1. 1 研究の背景及び目的	1
1. 2 従来の研究の概観	3
1. 3 研究の特徴	6
1. 4 論文の構成	8
2. 映像機器制御用マイクロプロセッサ	13
2. 1 まえがき	13
2. 2 4ビットマイクロプロセッサ	16
2. 2. 1 性能向上	16
2. 2. 2 マイクロプロセッサ開発効率向上	23
2. 2. 3 評価	27
2. 3 32ビットマイクロプロセッサ	34
2. 4 ソフトウェア開発ツール	41
2. 4. 1 ソフトウェア評価用マイクロプロセッサ	41
2. 4. 2 モニタ	46
2. 5 むすび	56
3. マイクロプロセッサ援用信号処理	58
3. 1 まえがき	58
3. 2 オーディオ	59
3. 2. 1 時定数部品切替型回路	60
3. 2. 2 ゲイン制御型回路	65
3. 2. 3 実験結果	70
3. 3 VTRデジタル特殊効果システム	76

3. 3. 1	汎用メモリをもちいたシステム	78
3. 3. 2	TV映像信号専用メモリを用いたシステム	83
3. 3. 3	システム構成	90
3. 3. 4	特殊効果の実現	95
3. 3. 5	開発評価	102
3. 4	むすび	105
4.	マイクロプロセッサによる信号処理	107
4. 1	まえがき	107
4. 2	アルゴリズム	108
4. 2. 1	動画像のIDCT必要演算量	109
4. 2. 2	IDCT処理ソフトウェア	111
4. 3	第2ALUによる信号処理	113
4. 3. 1	IDCT高速処理アーキテクチャ	113
4. 3. 2	IDCT処理アルゴリズム	115
4. 3. 3	エバリュエーション	116
4. 4	アーキテクチャの改善による高速化の検討	118
4. 4. 1	IDCT処理の高速化	118
4. 4. 2	第2ALUの性能向上	120
4. 4. 3	バススイッチ結合メモリ・マルチプロセッサ構造	130
4. 4. 4	マイクロプロセッサによるMPEG-2の実現検討	132
4. 5	むすび	135
5.	結論	136
	謝辞	140
	参考文献	141
	筆者発表論文	154

## 論文要旨

マイクロプロセッサはコンピュータのプログラムストアード方式を先ず導入した汎用制御装置からスタートした。その後コンピュータのアーキテクチャを順次受継ぎまた独自の発展を続けるが、今や上位のものはコンピュータそのものにまで達している。1971年に4ビットのマイクロプロセッサに引き続き1973年に8ビットのものが開発され1枚のボードでコンピュータを構成するワンボードコンピュータが生まれ、主に事務用電卓やPOS端末、シーケンサなどの事務用機器や工場機器に適応された。一方1974年、主に民生機器の制御用に1チップでCPUをはじめとしてプログラムメモリ、データメモリと周辺を含む4ビットのマイクロプロセッサが生まれた。これら2つの流れは独自に、あるときは影響をしながら成長展開してきた。そして特に後者の1チップで構成されるマイクロプロセッサは映像機器に先ず使われた。

マイクロプロセッサは誕生以後映像機器において主に機構制御とヒューマンインタフェースの向上を実現し、スローやスチルなどいくつもの付加機能を生みながら現在では映像信号を直接処理するところまでにいたっている。映像機器へのマイクロプロセッサ適応に関して次の重要課題があった。

(1) 映像機器にマイクロプロセッサを正しく応用し、映像機器の成長を最善にすること。時と共に性能向上するマイクロプロセッサにより、製品として最善の性能と付加価値を実現することである。

(2) マイクロプロセッサ応用のシステム設計及びソフトウェア設計の手法を映像機器開発の設計部門に広げていくこと。

(3) 映像機器に対し、その時々々の半導体製造技術を十分生かした最も適合するマイクロプロセッサと開発ツール、それに応用技術を開発していくこと。

このような背景のもとに、映像機器におけるマイクロプロセッサの構造と応用に関して、次の3点の技術を確立することが本研究の目的である。

(1) 映像機器制御に適合したマイクロプロセッサ技術の確立

(2) 映像機器へのマイクロプロセッサ援用信号処理技術の確立

(3) 映像機器へのマイクロプロセッサによる信号処理技術の確立。

本研究はこれらについてマイクロプロセッサ及びマイクロプロセッサの応用技

術の観点よりまとめた。

映像機器制御に適合したマイクロプロセッサ技術の確立において、マイクロプロセッサの性能とその開発効率の向上をマイクロプロセッサの構造（アーキテクチャ）の改善により実現した。

1980年代において4ビットマイクロプロセッサにおける性能向上は未だ十分でなく、また、コスト面の制約で十分なハードウェアを得ることが難しい状況であった。そこで、限られたハードウェアでいかに多くの仕事をできるようにするか、具体的言えばプログラムをいかに少ない命令コード数で実現するかということが課題であった。それで複数機能命令、多分岐命令と連続記述命令の設定、命令の1ワードを8ビットから9ビットに1ビット増すことにより、1ページのプログラム空間を64語から128語に増加させること及びサブルーチン専用ページ設定の改善を行った。これらの機能を十分に使うことによって、1つのプログラムで約50%の命令コード削減を実現した。またソフトウェア品質とソフトウェア開発効率の改善のため、アセンブラ言語で書かれたプログラムの可読性向上も合せ対策した。

4ビットマイクロプロセッサの開発効率向上は内部共通バスを3本配すること、命令実現の制御信号発生の集中化と制御信号の時分割化により制御の一元管理と見通しのいいハードウェアシステムが、論理設計時の不具合減少と開発人工と期間の短縮を可能にした。またソフトウェア開発支援ツールとしてインサーキットエミュレータのエバリュエーションチップも同じマイクロプロセッサで共用化をはじめて実現した。これらの新しい構造（アーキテクチャ）の採用と改善策により実現した開発効率向上は、30品種の開発品種で58%の削減の成果を得た。

8ビットマイクロプロセッサ用モニタはミニコンピュータに比べ、メモリ空間が小さいことから、実行用モニタとソフトウェア開発用モニタなどその利用目的により専用化させた。また機器組込マイクロプロセッサ用として、モニタをマスクROMにするとともにそのモニタ機能の拡張を可能とする仕組みを開発した。また特殊なハードウェアを必要とせず一般の入出力機器システムタイプライタやCRTに500バイト程度で実現するタイムシェアリングモニタも実現した。

次に映像機器に適合する2つの新しい処理システムを開発した。それはマイクロプロセッサ援用信号処理とマイクロプロセッサによる信号処理である。マイク

ロプロセッサ援用信号処理とは、信号処理の一部をマイクロプロセッサで実行することにより、性能向上や外付部品削減を実現することを言う。マイクロプロセッサによる信号処理とは、信号処理のほとんど全てをマイクロプロセッサで直接処理することである。この場合マイクロプロセッサは、AD コンバータや DA コンバータ、それに信号処理のためのアナログ回路やデジタル回路を内蔵する場合がある。

マイクロプロセッサ援用信号処理において、現行の音声処理に比べ外付部品の 75%削減を行い、信号処理 IC とマイクロプロセッサのシステムオンチップ化の可能性を見出した。また VTR のデジタル特殊効果で、論理回路とマイクロプロセッサの 2 レベルの制御系を確立し、システムの開発効率向上と画質改善を実現した。32 ビットマイクロプロセッサに 2 つめの ALU を追加することで、従来に比べ 1/3 の時間で IDCT を実現し、デジタルテレビに使用される MPEG-2 デコーダをソフトウェア処理で実行する可能性を見出した。

映像機器におけるマイクロプロセッサの構造と応用に関して、半導体の生産技術が重要な意味を持つ。すなわち半導体の生産技術が今後とも進展し、これによりマイクロプロセッサの構造（アーキテクチャ）と応用技術は目覚しく発展ならびに新しい構想のものを多く生む可能性がある。そしてこれらに方向性を与えるのがマイクロプロセッサを要する製品、すなわちマイクロプロセッサ応用機器であり、本論文では映像機器である。そして今までもこのように、まさに映像機器が強いニーズを発し水先人になりマイクロプロセッサ及び半導体の成長を導く。マイクロプロセッサは生まれたばかり、映像機器はアナログ方式のものが成熟期の 1973 年よりほぼ 30 年、マイクロプロセッサの設計と半導体の応用技術に携り、映像機器におけるマイクロプロセッサの構造と応用技術を研究し、以上のような研究成果を得た。

## 図表一覧

図 1. 1	論文の章立て . . . . .	1 0
Fig. 1.1	Construction of this paper	
図 1. 2	マイクロプロセッサ応用機器開発体系 . . . . .	1 1
Fig. 1.2	Microprocessor application development system	
図 1. 3	映像機器におけるマイクロプロセッサ応用技術 . . . . .	1 2
Fig. 1.3	Application engineering of microprocessor for visual equipment	
図 2. 1	アーキテクチャ (構造) の改善項目 —性能向上と開発効率向上のための改善項目— . . . . .	1 5
Fig. 2.1	Improvements for architecture	
図 2. 2	複数機能命令 XAMI . . . . .	1 7
Fig. 2.2	Multi function Instruction XAMI	
図 2. 3	複数機能命令によるメモリデータの交換プログラム . . . . .	1 8
Fig.2.3	Program example of memory data exchange with multi function Instruction	
図 2. 4	サブルーチン格納専用ページの機能 . . . . .	1 9
Fig. 2.4	Function of exclusive program page for subroutine	
図 2. 5	アセンブラ言語の可読性向上 . . . . .	2 1
Fig. 2.5	Improvement of assembly language	
図 2. 6	アセンブラ言語 可読性向上のためのニーモニック . . . . .	2 2
Fig. 2.6	Signal mnemonic for the improvement of assembly language	
図 2. 7	4 ビットマイクロプロセッサのデータ転送経路図 . . . . .	2 4
Fig. 2.7	Data pass of 4 bit microprocessor	
図 2. 8	制御信号発生の集中化と制御信号の時分割使用 . . . . .	2 5
Fig. 2.8	Concentration of control signal generation and control line used time sharing	
図 2. 9	3 語命令、2 語命令の構成 . . . . .	2 7
Fig. 2.9	Organization of 3 and 2 word instruction	



図 2. 1 0	3 語、2 語命令実行中における割り込み禁止の仕組み . . . . .	2 7
	Fig.2.10 Contrive of interrupt disenable at execute 2 and 3 word instruction	
図 2. 1 1	ブロック図 . . . . .	3 0
	Fig. 2.11 Block diagram	
図 2. 1 2	性能機能概要 . . . . .	3 1
	Fig. 2.12 System performance and function	
図 2. 1 3	命令コード対応表 M58845A-XXXXP . . . . .	3 2
	Fig. 2.13 Instruction code	
図 2. 1 4	命令コード対応表 . . . . .	3 3
	Fig. 2.14 Measurement table Instruction code	
図 2. 1 5	マイクロプロセッサのビット幅による処理能力差 . . . . .	3 6
	Fig. 2.15 Difference of the capacity at each bit microprocessor	
図 2. 1 6	主要処理のビット巾と必要処理時間 (マイクロプロセッサの組込応用) . . . . .	3 7
	Fig. 2.16 Required processing time and main processing bit width	
図 2. 1 7	マイクロプロセッサの周波数トレンド . . . . .	3 8
	Fig. 2.17 Microprocessor Frequency Trend	
図 2. 1 8	マイクロプロセッサのビット幅と各 CPU のゲート数の比較 . . . . .	3 8
	Fig. 2.18 Microprocessor bit wide and each CPU gate number comparison	
図 2. 1 9	1 つ又は複数の増設 ALU を持つマイクロプロセッ . . . . .	3 9
	Fig. 2.19 Microprocessor with one or some 2ndALU	
図 2. 2 0	増設 ALU . . . . .	4 0
	Fig. 2.20 2nd ALU	
図 2. 2 1	ソフトウェア開発ツール体系 . . . . .	4 3
	Fig. 2.21 Software development tool system	
図 2. 2 2	エバリエーション機能実現のためのポートの時分割使用 . . . . .	4 4
	Fig. 2.22 Port time-division use for evolutionary function	
図 2. 2 3	ポートにおける時分割タイミング . . . . .	4 5
	Fig. 2.23 Time-division timing on port	

図 2. 2 4	時分割入出力のプロセッサ内部と外付き回路モデル . . . . .	4 5
	Fig. 2.24 Internal processor of time shared I/O and external circuit model	
図 2. 2 5	モニター一覧 . . . . .	4 8
	Fig. 2.25 Monitor List	
図 2. 2 6	開発用モニタ BOM-PTS の機能 . . . . .	4 9
	Fig. 2.26 Monitor for development BOM-PTS function	
図 2. 2 7	制御用モニタのコマンド例 . . . . .	5 0
	Fig. 2.27 Example of monitor function	
図 2. 2 8	モニタ機能のトレース、アドレスボルトの実行 (その 1) . . . . .	5 1
	Fig. 2.28 Trace with monitor function and address bolt operation No.1	
図 2. 2 9	モニタ機能のトレース、アドレスボルトの実行 (その 2) . . . . .	5 2
	Fig. 2.29 Trace with monitor function and address bolt operation No.2	
図 2. 3 0	ROM に格納されたモニタの機能追加法 . . . . .	5 3
	Fig. 2.30 Method of additional stored-ROM monitoring function	
図 2. 3 1	ROM のプログラムを任意のアドレス番地に設定を可能にするため命令追加 (リロケートブル ROM) . . . . .	5 3
	Fig. 2.31 Additional order (relocatable ROM)	
図 2. 3 2	マルチタスク制御 . . . . .	5 4
	Fig. 2.32 Multitasking control	
図 3. 1	交流バイアス音声記録の基本システム . . . . .	6 1
	Fig. 3.1 Basic AC biased recording system	
図 3. 2	再生イコライザの周波数特性 . . . . .	6 1
	Fig. 3.2 Characteristics of the playback equalizer	
図 3. 3	再生イコライザの回路例 . . . . .	6 2
	Fig. 3.3 Example circuit of the playback equalizer	
図 3. 4	記録イコライザの周波数特性 . . . . .	6 3
	Fig. 3.4 Characteristics of the recording equalizer	
図 3. 5	記録イコライザの回路例 . . . . .	6 4
	Fig.3.5 Example circuit of the recording equalizer	

図 3. 6	ALC の回路例 . . . . .	6 5
	Fig. 3.6 Example of ALC circuit	
図 3. 7	再生イコライザ回路 . . . . .	6 6
	Fig. 3.7 Playback equalizer circuit	
図 3. 8	記録イコライザ回路 . . . . .	6 7
	Fig. 3.8 Recording equalizer circuit	
図 3. 9	マイクロプロセッサ制御による A L C 回路 . . . . .	6 8
	Fig. 3.9 Microprocessor controlled ALC circuit	
図 3. 1 0	ALC の入出力波形 . . . . .	6 9
	Fig. 3.10 Waveform of ALC	
図 3. 1 1	GCA モジュールの構成 . . . . .	7 0
	Fig. 3.11 Module of the gain controlled amplifier	
図 3. 1 2	再生イコライザの実験結果 . . . . .	7 1
	Fig. 3.12 Experimental results of the playback equalizer	
図 3. 1 3	イコライザ前の録再特性 . . . . .	7 2
	Fig. 3.13 Recording and playback characteristics without equalizer	
図 3. 1 4	記録イコライザの実験結果 . . . . .	7 2
	Fig. 3.14 Experimental results of the recording equalizer	
図 3. 1 5	記録再生周波数特性 . . . . .	7 4
	Fig. 3.15 Frequency characteristics of recording and playback	
図 3. 1 6	A L C の実験結果 . . . . .	7 5
	Fig. 3.16 Parameters of algorithm	
図 3. 1 7	映像画面の構成 . . . . .	7 9
	Fig. 3.17 Picture structure	
図 3. 1 8	映像信号のフィールドメモリ配置 . . . . .	8 0
	Fig. 3.18 Field memory construction of video signal	
図 3. 1 9	システム構成 . . . . .	8 2
	Fig. 3.19 System structure	
図 3. 2 0	映像信号の 3 つの形態によるデジタル特殊効果の実現 . . . . .	8 5
	Fig. 3.20 Digital special effect realized by three modes of VCR signal	

図 3. 2 1	フィールドメモリのブロック図	8 8
	Fig. 3.21 Block diagram of field memory	
図 3. 2 2	フィールドメモリのシリアル入力タイミング図	8 9
	Fig. 3.22 Diagram of timing for serial input data in field memory	
図 3. 2 3	TV映像信号用専用メモリシステム	9 2
	Fig. 3.23 Memory system exclusively for TV picture signal	
図 3. 2 4	TV映像信号専用メモリによる映像画面の構成	9 3
	Fig. 3.24 Picture construction of memory exclusively for TV picture signal	
図 3. 2 5	映像信号のメモリへの書き込みと読み出し	9 4
	Fig. 3.25 Read and write on picture signal memory	
図 3. 2 6	マルチ画面の作成方法	9 6
	Fig. 3.26 Method of drawing up multi-screen	
図 3. 2 7	画面拡大方法	9 8
	Fig. 3.27 Method of screen enlargement	
図 3. 2 8	モザイク	9 9
	Fig. 3.28 Mosaic image	
図 3. 2 9	ピクチャインピクチャ	100
	Fig. 3.29 Picture in picture	
図 3. 3 0	デジタルスピードサーチ	101
	Fig. 3.30 Digital speed search	
図 3.3 1	デジタル特殊効果；9画面表示	103
	Fig. 3.31 Digital special effect ; 9 pictures	
図 3.3 2	モザイク	103
	Fig. 3.32 Digital special effect ; 9 pictures	
図 3. 3 3	デジタル特殊効果；ソラリゼーション	104
	Fig. 3.33 Digital special effect ; Solarization	
図 3. 3 4	デジタル特殊効果；ピクチャインピクチャ	104
	Fig. 3.34 Digital special effect ; Picture In Picture	
図 4. 1	MPEG-2 デコーダの機能構成	108
	Fig. 4.1 MPEG-2 decoder	

図 4. 2	画像構成 . . . . .	109
	Fig. 4.2 Picture construction	
図 4. 3	フレーム構成 . . . . .	109
	Fig. 4.3 Frame construction	
図 4. 4	マクロブロックの構成 . . . . .	110
	Fig. 4.4 Macro block construction	
図 4. 5	ブロック構成 . . . . .	110
	Fig. 4.5 Block construction	
図 4. 6	IDCT の Chen アルゴリズム . . . . .	111
	Fig. 4.6 Fast IDCT algorithm of Chen	
図 4. 7	複数の ALU を持つ RISC 32 ビットマイクロプロセッサ . . . . .	113
	Fig. 4.7 32-bit RISC processor with multi-ALU	
図 4. 8	第 2 ALU の構造 . . . . .	114
	Fig. 4.8 Structure of 2nd ALU	
図 4. 9	制御テーブルの構造 . . . . .	115
	Fig. 4.9 Structure of Control Table	
図 4. 10	IDCT のデータ送信 . . . . .	116
	Fig. 4.10 Data transfer of IDCT	
図 4. 11	1 秒間にかかる処理時間 . . . . .	117
	Fig. 4.11 Total process time for 1 second	
図 4. 12	第 2 ALU の高速化 . . . . .	120
	Fig. 4.12 Fast 2nd ALU	
図 4. 13	IDCT 用に改良された第 2 ALU の構成 . . . . .	120
	Fig. 4.13 Structure of 2nd ALU improved for IDCT processing	
図 4. 14	第 2 ALU の実行テーブル . . . . .	122
	Fig. 4.14 2nd ALU control table	
図 4. 15	第 2 ALU 演算子一覧 . . . . .	122
	Fig. 4.15 2nd ALU Operator	
図 4. 16	第 2 ALU の制御レジスタ . . . . .	123
	Fig. 4.16 2nd ALU control register	

図 4. 1 7	実行テーブルによる演算の実行 . . . . .	123
	Fig. 4.17 Operation performance by imperative table	
図 4. 1 8	高速化のためのメモリの多重化 . . . . .	124
	Fig. 4.18 Multi memory for high-speed	
図 4. 1 9	データメモリと演算器間のデータ転送とバス方式 . . . . .	125
	Fig. 4.19 Data memory, data transfer in the operation and bus system	
図 4. 2 0	データ転送バス方式と処理時間 . . . . .	126
	Fig. 4.20 Data transfer bus system and processing time	
図 4. 2 1	データバス方式とデータバス巾と処理時間の関係 . . . . .	126
	Fig. 4.21 Relationship of between data bus system, data bus width and processing time	
図 4. 2 2	第 2 ALU による高速化 . . . . .	128
	Fig. 4.22 Speeding up by 2nd ALU	
図 4. 2 3	第 2 ALU で処理した場合の考察 . . . . .	129
	Fig. 4.23 Consideration as in processing 2nd ALU	
図 4. 2 4	バススイッチ結合マルチメモリマルチマイクロプロセッサ構造 (アーキ テクチャ) . . . . .	131
	Fig. 4.24 Structure (Architecture) of multi-microprocessor with multi-memory connected by bus switch	
図 4. 2 5	データ転送比較 . . . . .	131
	Fig. 4.25 Comparison of data transfer	
図 4. 2 6	IDCT 処理の各演算ステップ . . . . .	134
	Fig. 4.26 Each operation step for IDCT processing	
表 3. 1	再生イコライザの時定数 . . . . .	6 2
	Table 3.1. Time consonants of the playback equalizer	
表 3. 2	再生イコライザの諸元 . . . . .	7 1
	Table 3.2. Parameters of the playback equalizer	
表 3. 3	記録イコライザの諸元 . . . . .	7 3
	Table 3.3. Parameters of the recording equalizer	

表 3. 4	記録 GCA の緒元 . . . . .	7 3
Table 3.4.	Parameters of the recording GCA	
表 3. 5	ALC 用 GCA の諸元 . . . . .	7 4
Table 3.5.	Parameters of GCA for ALC	
表 3. 6	アルゴリズムの諸元 . . . . .	7 5
Table 3.6.	Parameters of algorithm	
表 3. 7	2 階層の制御におけるマイクロプロセッサと 論理回路との役割分担 . . . . .	8 6
Table 3.7	Roles of Microprocessor and Logic circuit by dual hierarchical control	
表 3. 8	フィールドメモリの概要仕様 . . . . .	8 9
Table 3.8	Specification for outline of Filed Memory	
表 4. 1	4 乗算・加算による IDCT のアルゴリズム . . . . .	11 2
Table 4.1	IDCT algorithm using 4 multiply - add operation	
表 4. 2	4 積和による高速 IDCT アルゴリズム . . . . .	11 9
Table 4.2	Fast IDCT algorithm using four-multiply-add operation	

# 1 序論

## 1. 1 研究の背景及び目的

映像機器の発展にはマイクロプロセッサ (micro-processor) が大きく寄与しており、現在においてはシステム制御、ヒューマンインターフェースと機構制御など映像機器の制御の中核的存在である [1]。またマイクロプロセッサも映像機器の制御に使われたことにより大きく発展した [2、3]。

初期のマイクロプロセッサ (1971 年～1979 年) は機器の制御部を構成するのに CPU (Central Processor Unit) (CPU をマイクロプロセッサと称することもある)、ROM (Read On Memory)、RAM (Random Access Memory)、タイマなどの周辺機能及び入出力 I/O (Input/Output) など 10 LSI 前後必要で、映像機器や家庭用電気製品への適応はコストも高くまた容積も大きいため難しかった [4]。その後半導体製造技術の向上により 1 チップでマイクロプロセッサが実現できるようになってきて (1976 年～)、映像機器にはコスト面及び容積面で適応可能になってきたが、マイクロプロセッサを映像機器の制御部への適応の具体化の技術が弱かったというより存在すらしなかった [5、6]。また映像機器においてはマイクロプロセッサ出現の早い時期よりその応用に着手したため、後から応用を試みるいくつかの機器より先行した応用技術開発をしていくことになった [7、8、9]。

映像機器へのマイクロプロセッサ適応に際して重要課題は 3 つある。

(1) マイクロプロセッサ適応で製品に付加価値を付け、事業として成功させると共に正しい製品の成長を促すこと。時と共に性能向上するマイクロプロセッサの応用により、いくつもの価値すなわち新しい機能、性能向上、品質向上それに原価低減があるが、その製品の正しい成長を見抜き、それを着実に具体化すべきである。

(2) マイクロプロセッサ応用のシステム設計技術とソフトウェア設計技術を映像機器開発の電子設計者に広げていくこと。映像機器への半導体素子のそれまでの適用は、映像信号処理や音声信号処理のアナログ IC が主であった。その後デジタル IC によるデジタル信号処理が増加していくが、しかしマイクロプロセッサの応用及びソフトウェア設計はこれらと全く異質なものであった [7、8]。

(3) 半導体製造技術の向上を十分捉えた、映像機器により一層適合するマイ



クロプロセッサを開発していくこと。機能向上するマイクロプロセッサを十分生かす新しいシステム構成法と新しい応用技術を開発をしていくことである [5、9、10]。

このような背景のもとに、映像機器におけるマイクロプロセッサの構造と応用に関して、下記3点の技術を確立することが本研究の目的である。

- (1) 映像機器制御に適合したマイクロプロセッサ技術の確立
- (2) 映像機器へのマイクロプロセッサ援用信号処理技術の確立
- (3) 映像機器へのマイクロプロセッサによる信号処理技術の確立

本論文は、これらについてマイクロプロセッサ及びマイクロプロセッサの応用技術の観点よりまとめたものである。

## 1. 2 従来の研究の概観

### (1) 映像機器制御に適したマイクロプロセッサ関連技術の確立

映像機器とりわけ家庭用 VTR (Video Tape Recorder) にマイクロプロセッサを適応しはじめたのは 1979 年からである。当時 4 ビットマイクロプロセッサで ROM は 2K バイトのものを 2 つ使い、機構 (メカ) 制御とタイマ機能 (番組予約) を行わせていた [11]。TV (Television) には 1981 年よりマイクロプロセッサが適応され始めた。

プロセッサ、コンピュータ又は電子式計算機としては 1940 年に J.V.Atanasoff が ABC 計算機を、1946 年に Jr.J.P.Eckert と J.W.Mauchly が ENIAC (Electronic Numerical Integrator and Calculator) を開発したが、これらは現在言う Von Neumann アーキテクチャまたはストアードプログラム方式のコンピュータではなかった [12、13、14]。すなわちその差はデータと同じようにプログラムを取り扱えること、つまりメモリがランダムアクセス可能であることがキーポイントであった。それを実現したのが 1949 年 M.V.Wilks によって開発された EDSAC (Electronic Delay Storage Automatic Calculator) であり、これが現在では世界初のコンピュータの実現と言われている [15、16、17]。

最も基本的な機能であるサブルーチンや間接メモリアクセスが、1949 年にすでに M.V.Wilks によって明らかにされていた。1951 年には割込機能が、1953 年にはマイクロプログラム (micro-program) 方式が、1957 年にはパイプライン (pipeline) 制御方式が発明された。1949 年の世界初のコンピュータ EDSAC より 10 年間でこのような重要な技術が集中して発明開発されたことになる [18、19、20]。またさらに 1962 年にはバーチャルメモリ (virtual memory) が、1968 年にはキャッシュメモリ (cache-memory) が発明された [21、22]。これらの技術開発のねらいはコンピュータがより多くのメモリをいかにうまく持つかということ、そしてプログラムをいかに速く実行するかという 2 点であった。これらは特にハードウェア構造 (アーキテクチャ) の改善研究の成果であった [25、26、27]。そしてコンピュータ開発の開発期間の短縮と開発成功確度の向上のためマイクロプログラムが実用化されていった [18、28、29]。また並列コンピュータを含めたコンピュータの体系化が 1966 年 M.J.Flynn によりなされた [34]。

そして 1971 年にはマイクロコンピュータ (micro-computer) / マイクロプロセ

ッサが発明された [23]。さらに RISC (Reduced Instruction Set Computer) が 1980 年に提案された [30、31、32]。1990 年代にスーパスカラ (superscalar) が開発される [33、35]。これは 1 つのプロセッサでいかに高性能を出すかということに主眼をおいた研究開発で十分に役割を果たしているものの、コンピュータ/マイクロプロセッサの大きい発展の流れからは外れることになると考えている。それは半導体生産技術向上による高速化と大集積化に、ハードウェアすなわちアーキテクチャの適応性が問題になるからである。

映像機器へのマイクロプロセッサの応用は VTR が先駆をなしその数年後に TV、自動車のエンジン制御、家電、オーディオと続いた。これらに使われるマイクロプロセッサはワンチップマイクロコンピュータと呼ばれたり MCU (Micro Control Unit) と呼ばれたりするもので、1 チップに CPU、ROM、RAM、I/O などを集積されたものであった。また、1 チップに集積されずシステム構成の自由度の大きい、マルチチップ構成のマイクロプロセッサは一般産業用に主に使われ、これによりボードコンピュータの応用が進んだ。VTR への応用に引き続き TV においても、マイクロプロセッサがリモコン送信、リモコン受信、タッチスイッチ、電子ボリュームさらに電子チューニングに使われだした。VTR の場合と同じくヒューマンインタフェース機能の向上がマイクロプロセッサ適応の初期においては主であった。そして 1986 年から 1994 年にかけて VTR の特殊再生の発達に大きく寄与した。マイクロプロセッサ援用信号処理によるフィールドメモリシステム制御とオートトラッキング制御の 2 つの用例がある [2、3、36、37、39]。

1980 年度前半は、プログラムメモリすなわち内蔵 ROM は 2K バイトから 8K バイトまでと小さく、システムの機能向上にはマイクロプロセッサの数を増やすことで対処していた。ここではソフトウェアの言語はアセンブラが主で、プログラム容量としていかに効率のいいソフトウェアを設計するかということが応用技術の役割であった [40、41]。1990 年代では 8 ビットマイクロプロセッサの内蔵 ROM 容量は、64K バイトの通常その最大の値に近づき、それより大きい ROM 容量を必要とする応用は 16 ビットマイクロプロセッサに移っていった。

また 1990 年の後半よりエージェント機能の取込が試みられている [42]。現在 2002 年では、32 ビットマイクロプロセッサ内蔵 ROM 容量の最大のものは 512 バイトから 768K バイトと大きく、ピン数も 100 ピンを越える。カメラ一体型デ

デジタルVTRでは3～4チップのマイクロプロセッサとASICで構成されていたが、さらにASICと32ビットマイクロプロセッサが統合されSOCへの流れが進展している[101、111]。

デジタルTVにおいても32ビットマイクロプロセッサが使われ、100Kバイト以上のOSを持つシステムもある。これはテレビの映像信号を論理回路で処理するもので、マイクロプロセッサは主にGUI (Graphical User Interface) を受け持つ[100、102、110]。またマイクロプロセッサは映像信号を直接処理することが出来なかったが、マシンサイクルの高速化とその構造(アーキテクチャ)の改善により実現の試みがなされようとしている[9]。

### 1. 3 研究の特徴

本研究では映像機器の機構制御、信号処理におけるマイクロプロセッサ応用の可能性を、マイクロプロセッサ、ソフトウェア開発ツール、ソフトウェア及びシステムを各々開発をすることにより例証し検討している。

映像機器に適合したマイクロプロセッサ、ソフトウェア開発ツール、マイクロプロセッサの応用技術の確立において目指すものは、システム開発の容易なことと性能コストに優れていることである。これに対応して4ビットと32ビットのマイクロプロセッサの性能向上とソフトウェア開発支援ツール、特にエミュレーション機能のマイクロプロセッサへの取込と用途別モニタの開発を行い、これらの開発を通して研究を進めた。一方マイクロプロセッサの性能向上において注意すべき重要なことは、性能向上のために行った構造（アーキテクチャ）の改善がその後の半導体生産技術向上の微細化による性能向上の足かせにならないことである。マイクロプロセッサの性能向上は、半導体生産技術向上によるものとマイクロプロセッサの構造（アーキテクチャ）の改善の2つで行われる。これは互いに独立して行われる場合と同時にされる場合がある。アーキテクチャの改善による性能向上は次の世代の半導体生産技術向上にも順応できるものと、難しいものがある。今後の半導体技術向上を読み次の世代にも有効なアーキテクチャの改善を積み重ねていく必要がある。1980年代において半導体生産技術の向上は、現在と同じく進歩すると考えていた。それで最低限ハードウェア量の増加による性能向上は有効で、映像機器へのマイクロプロセッサの提供という意味においてマイクロプロセッサの開発の頻度も多くなると考えた。ゆえにマイクロプロセッサ及びソフトウェア開発ツールの開発の効率向上も重要な技術項目で、マイクロプロセッサ及びソフトウェア開発ツールの内部構造の改善と革新を行った。

映像機器へのマイクロプロセッサ援用信号処理技術の確立は通常のリモコン、キーボードの入力や表示等のヒューマンインタフェース、シーケンス制御や、メカコントロールに比べ、音声や映像信号に直接または間接的に制御を行うもので、いかにこれらの信号にマイクロプロセッサが関わり有効であることを示した。1つはアナログの音声処理に対して、他方はデジタル化された映像信号に対して試みSOC（System On Chip）への検討も行った。

映像機器へのマイクロプロセッサによる信号処理技術の確立は、マイクロプロセッサで映像信号を直接処理しようとする試みで、現在の論理回路によるものに比べ、開発期間と開発工数の大幅削減及び半導体のコストすなわち同じプロセスの場合、チップサイズも小さくなるこれらの可能性を追及しようという考えである。もこれらは32ビットマイクロプロセッサの動作周波数と1チップに内蔵できるメモリとマイクロプロセッサのゲート数がある値を越えることが必要である。このある値と言うのは動作周波数が250MHzであり、ゲート数はトランジスタ数に置換えれば5000万であり、早くてウエハプロセスが0.12 $\mu$ mの年時から実現可能と考えている。これらの数値は現在、半導体製造技術向上の予測されている範疇にあり、現在の最新の試作プロセスでも十分確かめられる技術となっている。今後製造技術がまだまだ進展すると考えられるが、それをいかに使い有意義な物を作るか、または今までのやり方を変えられるかという問題に取り組んだと考えている。

ここでの試みは、アナログ回路またはデジタル回路による信号処理部とマイクロプロセッサによる制御部という従来からのシステム構成を執着しているだけでは、次の発展はないとの確信によるものである。すなわちバイポーラのアナログ回路もCMOSの大規模論理回路もマイクロプロセッサも半導体の製造技術の向上によりその役割が大きく変わってくると考え、それがマイクロプロセッサ構造（アーキテクチャ）の改革につながり、映像機器などのマイクロプロセッサ適応機器に大きく貢献する。

## 1. 4 論文の構成

本論文は全5章から成る。第1章は序論で研究の背景及び目的、これまでの研究の概観と本研究の位置及び特長について述べる。

第2章では映像機器に適したマイクロプロセッサの構造、すなわちアーキテクチャの改善とソフトウェア開発ツールの改善について述べる。映像機器へのマイクロプロセッサの適応は、映像機器におけるヒューマンインタフェースと性能の向上が目的である。マイクロプロセッサ誕生の早い時期より映像機器への応用がはじまり、特に1チップ構成マイクロプロセッサの性能向上と機能拡大は、映像機器の影響が大きい。映像機器のなかでも早くからマイクロプロセッサを応用したVTRでは、当初機構部のシーケンス制御と番組予約をマイクロプロセッサで行う。次いでシステム制御、リモコン受信、リールカウンタ、チューニングシステム、さらにソフトウェアによるモータサーボ制御、オンディスプレイ表示、テレテキスト機能と拡大していく。これらVTRに必要な機能を、最小限のハードウェアをマイクロプロセッサに付加しソフトウェアにより実現していった。このような状況下においてマイクロプロセッサの性能向上とマイクロプロセッサの開発効率向上を重要課題として強く求められた。性能向上は、半導体生産技術による進展で処理速度のアップと内蔵ハードウェアの規模の増大により実現できるが、ここでは特にマイクロプロセッサの構造（アーキテクチャ）の改良を主題とした。

構造（アーキテクチャ）の改良による性能向上では、4ビットマイクロプロセッサにおいて、命令コード格納メモリがいかに有効かという観点より、1命令の機能向上とサブルーチン機構の改良による命令コード格納メモリの使用削減を検討し、実際のマイクロプロセッサの開発を行い、ソフトウェアを評価しその効果を検証した。

32ビットのマイクロプロセッサの性能向上は、第2ALUを付加することにより、くり返し演算の実行を3倍の速度で行う工夫をした。さらに性能を向上するには第2ALUの複数配置やバススイッチによるメモリとマイクロプロセッサの切替などの高速化技術を検討した。

ワンチップ4ビットマイクロプロセッサの開発効率向上策の1つとして考え

たデータの流れを単純化し明確にすることは、命令の高速実行にも有効であった。具体的には内部データベースを3組配置し、またバススイッチでそれらが互いに連結させる構造である。さらに命令実行のシーケンスの単純化と制御信号発生の集中化は見通しのきく簡明な構造にすることになり、それを具体的に開発し評価した。またソフトウェア開発支援ツールに使うエバリュエーションチップ (Evaluation Chip) はマイクロプロセッサにエバリュエーション機能を付加することで個別のエバリュエーションチップ開発を不必要にした。

8ビットマイクロプロセッサ用のモニタを検討した。ミニコンピュータ (Mini-computer) 用モニタと同じ機能をはたすためには、その目的に応じソフトウェア開発用のものとソフトウェア実行用の2つに分けて開発した。これはミニコンピュータに比べ8ビットのマイクロプロセッサのメモリ空間は小さく、またパネルコントロールのような入出力装置を持つことが少ないことによるものである。

第3章では、映像機器へのマイクロプロセッサの新しい応用を検討した。映像機器の音声信号と映像信号に間接的に関わり、これらのシステムで性能向上、開発効率向上、部品点数削減の効用を得ようとするものである。音声信号の場合、音声信号をセンスすることによりそのレベルを認識し増巾度を制御する。結果として外付回路を減ずることにより SOC の可能性を検討している。また、映像信号の場合として、デジタルの特殊効果を高速の論理回路とマイクロプロセッサの2つで制御する方式を開発し評価した。映像信号の制御は映像信号を一時メモリに保持し必要タイミングで読み出すため水平映像信号の2桁上の処理速度が必要であり、この処理を論理回路で対応させた。この論理回路の機能を変更可能にすることや他の処理ブロックやヒューマンインタフェースを実行するのにマイクロプロセッサは有効である。

第4章では映像信号をマイクロプロセッサが直接処理する方法を検討した。RISC 32ビットマイクロプロセッサに第2章で検討した第2ALUを付加し、MPEG-2デコードのIDCT処理の検討をした。

第5章は結論として本研究の成果をまとめ、そして今後に残された課題について述べる。以上述べた1章から5章までの構成を図1.1に示す。



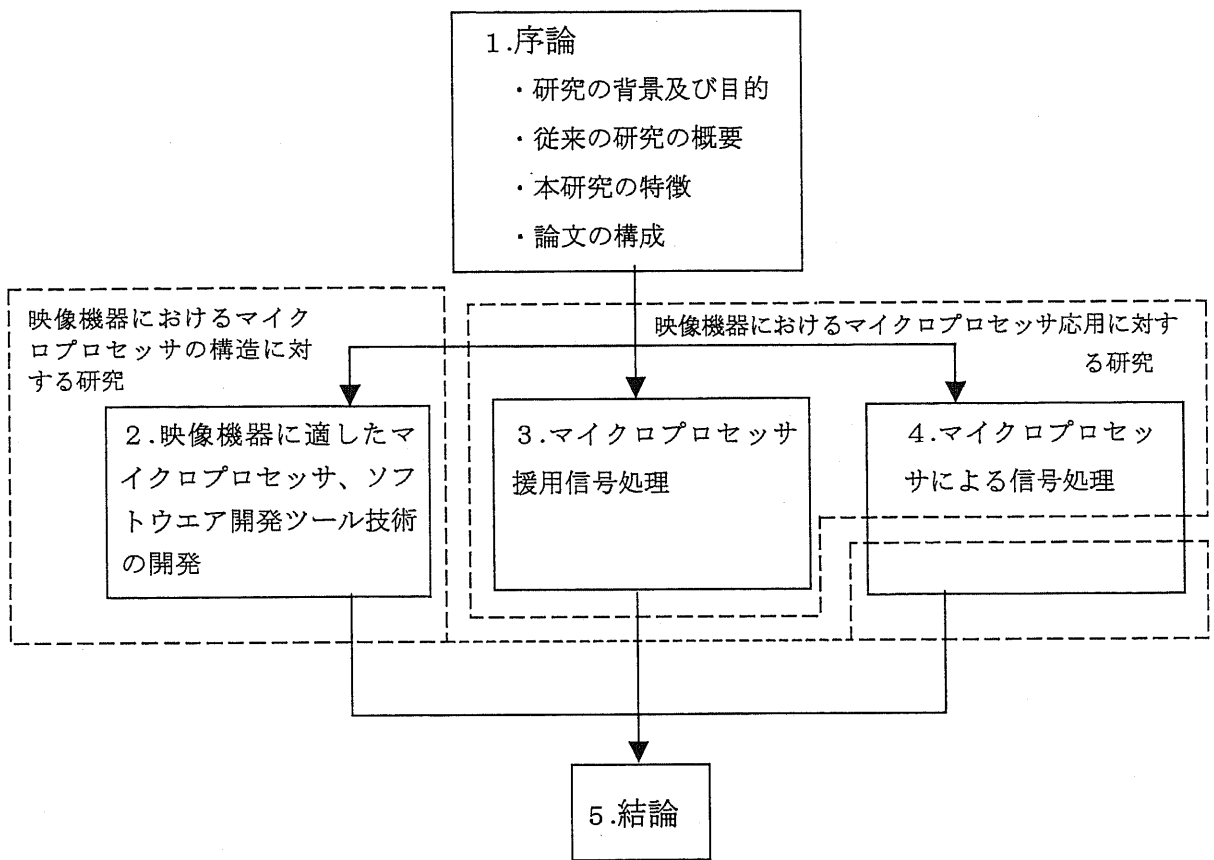


図 1. 1 論文の構成

Fig.1.1 Construction of this paper

次に映像機器とマイクロプロセッサ、それに映像機器に適応されるための応用技術を開発体系として図 1. 2 に示す。映像機器はマイクロプロセッサ及びその応用技術により、映像機器としてのいくつかの重要な機能の実現と性能向上をはたす。またマイクロプロセッサは映像機器とその実現手段である応用技術の2つよりニーズとしての仕様要請を受け、マイクロプロセッサの開発の方向性を得ることができる。応用技術は映像機器の実現したい機能とマイクロプロセッサの性能との差を何で対処するかを判断することになる。すなわちソフトウェアとしてはソフトウェアの改造、新たな処理ルーチンの追加または新たなソフトウェアシステムの設計、ハードウェアのシステムとしては一部のハードウェアの変更、新たなハードウェアの追加または新たなハードウェアシステムの設計の各々を、システムコストと開発費と開発期間を比較検討し何も対処しないことも含め選択

することになる。

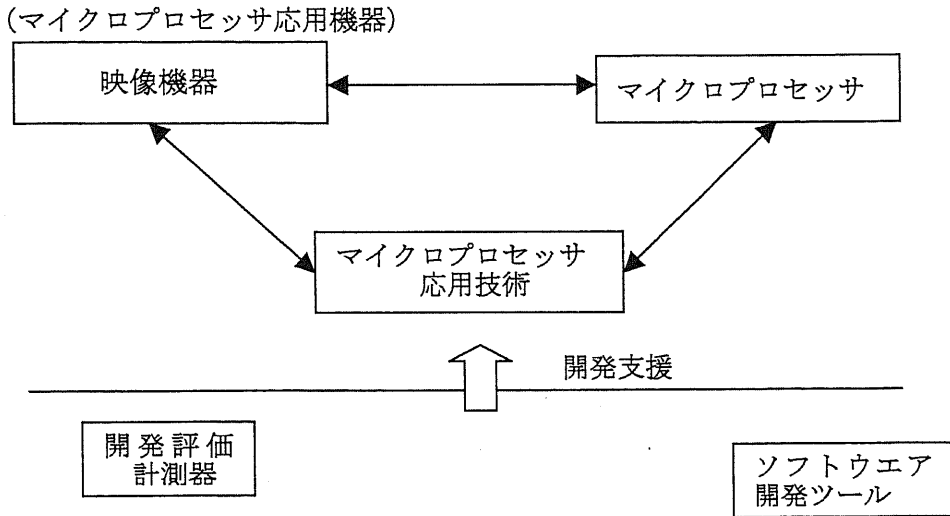


図 1. 2 マイクロプロセッサ応用機器開発体系

Fig.2.1 Microprocessor application

マイクロプロセッサとその応用技術と映像機器において、本論分の各章がどう  
いう関係にあるかを図 1. 3 に示す。映像機器での制御対象項目は映像信号、音  
声信号、モータ（サーボ）、シーケンス制御、ヒューマンインタフェースなどで  
ある。これらはマイクロプロセッサで直接信号データを処理する直接処理と、直  
接処理している IC や LSI をマイクロプロセッサが制御をする間接処理に分類で  
きる。図 1. 3 の映像機器ハードウェア部分の矢印 a と b は、信号処理における  
マイクロプロセッサ処理範囲が今後ますます拡大し、そして IC、LSI の役割が減  
少していくことを示す。半導体生産技術の向上はマイクロプロセッサ、IC と LSI  
に対して各々の性能と機能を向上させ、またコストパフォーマンスもよくする。  
今後マイクロプロセッサが周辺 IC、LSI をソフトウェア処理で取り込むか、ASIC  
としてハードウェアとして取り込むか、またはその両方かであり、全体としては

これら全てが起ると考える。そしてこれはまさに SOC への流れであると考え。

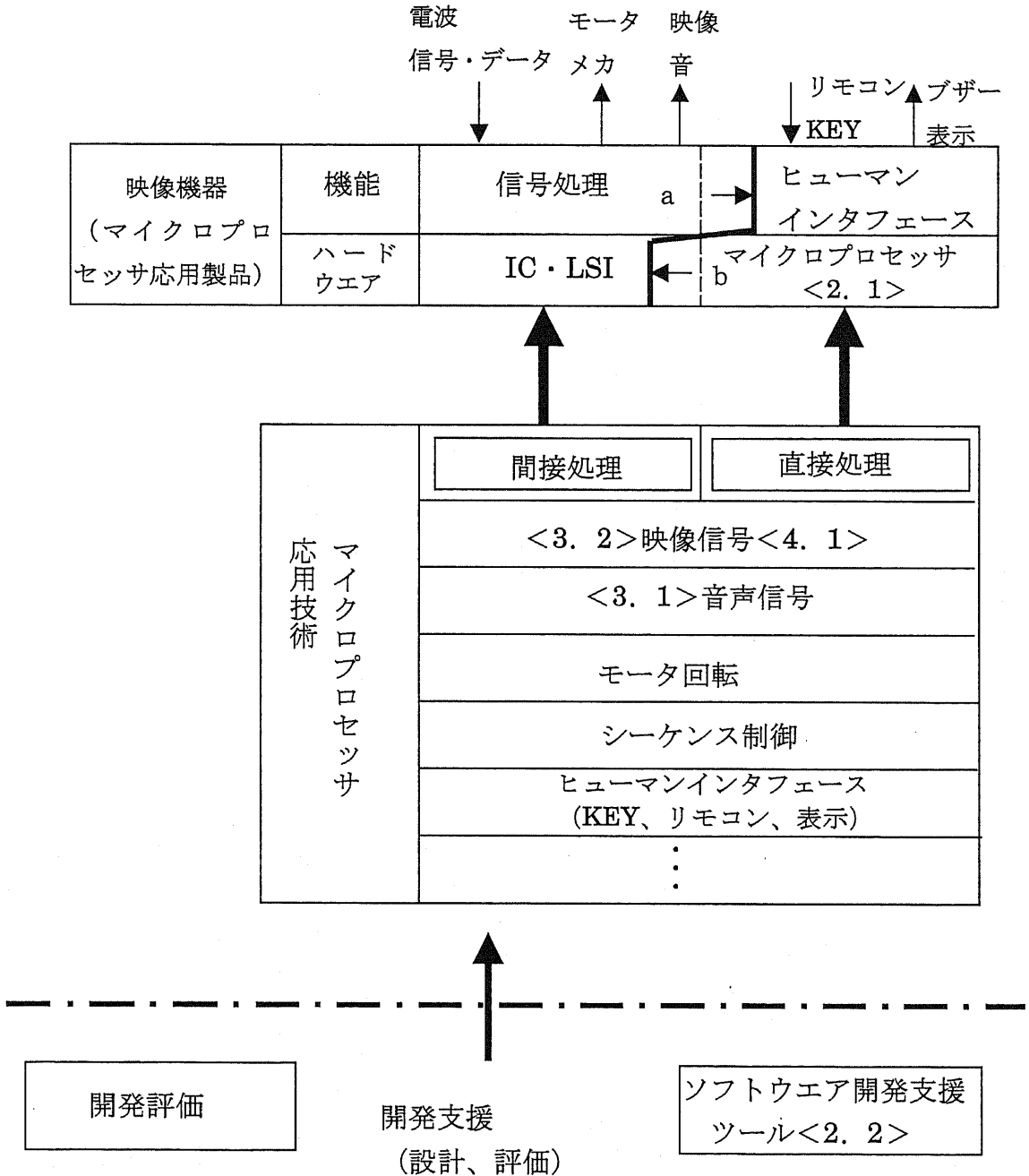


図 1. 3 映像機器におけるマイクロプロセッサ応用技術

Fig.1.3 Application engineering of microprocessor for visual equipment

## 2 映像機器制御用マイクロプロセッサ

### 2. 1 まえがき

映像機器制御すなわち VTR へのマイクロプロセッサの応用は、その応用されるマイクロプロセッサの数量が年間1億個を越えた最初の応用機器である[43]。マイクロプロセッサは1971年に4ビットの4004が開発された。これは当時の事務用電卓を開発する過程の中で誕生した[23]。事務用電卓はワイヤードロジック回路と呼ばれる論理回路 IC を複数配置したいくつかの基板で構成されていた。マイクロプロセッサは開発効率改善の目的でそれらの仕様の変更を論理回路の変更を伴わず ROM のコード、すなわちソフトウェアの変更で可能にする試みであった[97]。すなわち設計対象が論理回路からソフトウェアへの移行である[79]。

マイクロプロセッサはコンピュータのプログラムストアード方式を先ず導入した汎用制御装置といったものであり、その後独自の発展を続けるが、今や上位のものはコンピュータそのものにまで達している[44、45]。世界最初のコンピュータは、1949年 M.V.Wilkes らにより開発された EDSAC である[15]。これは1945年 Von Neumann 提唱によるプログラムストアード方式のコンピュータの実現である[13]。マイクロプロセッサの開発後、引き続き1974年に8ビットの8080、6800、1976年に Z-80 などが開発され、1枚のボードでコンピュータを構成するワンボードコンピュータが生まれた[47]。一方、主に民生機器の制御用に2チップもしくは1チップでプログラムメモリ、データメモリと周辺も含む8ビットの8048、PPS 4/2、M58840 などの4ビットマイクロプロセッサが生まれた[6、48、49]。これは1チップマイクロコンピュータとも呼ばれ、これら2つの流れが独自にあるときは影響し合いながら成長展開してきた[106]。

特に、後者の1チップで構成されるマイクロプロセッサは映像機器に先ず使われ、その応用とマイクロプロセッサの性能とソフトウェア開発支援ツール及び OS を発展させた[50、51]。しかしこの1チップのマイクロプロセッサの半導体としての信頼性確保において難問が多かった。これは1チップにマイクロプロセッサ以外に RAM、ROM、PROM、高耐圧ポートを組み込むので、各々個別の故障機構から共通のマスクとプロセスへの的確なフィードバックが当初難しかった[52、89]。VTR のマイクロプロセッサ適応の変遷は VTR の製品成長の歴史で

もある。70年代後半から第1に操作性の向上と機能の拡充で、第2に小型と軽量化であった。高性能化と品質の向上、開発期間の短縮及び部品、開発コストの削減は順次遅れて課題とされ対応されていくことになる [53、54]。

以下、2.2節では4ビットマイクロプロセッサ、2.3節では32ビットマイクロプロセッサの構造（アーキテクチャ）の改善について述べる。

2.4節では、ソフトウェア開発ツールについてOSモニタを述べる。それはミニコンピュータの開発ツールをそのままの形で受け継ぎつつ、マイクロプロセッサの特性に合せ込み独自の展開をする。2.5節は本章のまとめとして今後の課題を整理する。

4ビットマイクロプロセッサと32ビットマイクロプロセッサにおいて構造（アーキテクチャ）の改善を述べる [5、9]。この改善の目的はマイクロプロセッサの性能向上、マイクロプロセッサ開発効率向上とシステム開発の利便性向上の3つである。マイクロプロセッサ性能向上は図2.1の(i)～(iv)の4項目である。またマイクロプロセッサ開発効率向上は開発成功確度の向上と開発期間短縮が主要課題で、次の(v)と(vi)の2項目である [55、56]。システム開発の利便性の向上は性能向上された項目は全て使い易いということになるのは当然であるが、性能向上より利便性が高いものはここでは(v)項目の完備性のある機能構成である。

4ビットマイクロプロセッサの性能向上には、Nチャンネルプロセスを採用し命令の高速化を実現するものや、ROMコードの有効性を増すために命令をその応用開発により合うように一部仕様変更を可能にするものがあつた [44、117、120]。しかし映像機器用の4ビットマイクロプロセッサは1命令の実行時間がその時の応用では10 $\mu$ secで十分であつたし、また応用に合わせて命令を変更してROMコードの効率を少し上げるより、もっと大きくROMコードの効率を上げる方が勝る。それを実現する効果的な方法は以下の項で示すが、多機能命令やサブルーチン専用ページの設定が重要なものとなる。

構造（アーキテクチャ）の改善項目		向上		
		性能	利便性	開発効率
(i) CPU の高速化	<ul style="list-style-type: none"> <li>命令実行時間の高速化</li> <li>命令ビット巾の増数</li> <li>命令セットの最適化</li> <li>乗除算、浮動点演算器の付加</li> </ul>	レ レ レ レ	レ レ レ レ	
(ii) ハードウェア量の増強	<ul style="list-style-type: none"> <li>周辺機能を充実</li> <li>専用回路の取込み</li> <li>外付素子の削減</li> <li>メモリ容量増</li> <li>プログラムメモリの使用効率向上</li> <li>ポート数の増数</li> </ul>	レ レ レ レ レ レ	レ レ レ レ レ レ	
(iii) 電気的特性許容範囲の拡大	<ul style="list-style-type: none"> <li>選択できる電源電圧</li> <li>電源電圧の許容範囲</li> <li>低消費電力</li> <li>ポート耐圧</li> <li>ポート出力電流</li> </ul>	レ レ レ レ レ	レ レ レ レ レ	
(iv) ノイズ	<ul style="list-style-type: none"> <li>外乱ノイズ耐性</li> <li>ノイズ発生</li> </ul>	レ レ	レ レ	
(v) 完備性のある機能構成	<ul style="list-style-type: none"> <li>アセンブラ言語可読性向上</li> <li>内部機能に特異点のない</li> <li>直交性のある命令セット</li> </ul>		レ レ レ	レ レ
(vi) 簡明構造（アーキテクチャ）	<ul style="list-style-type: none"> <li>データパス</li> <li>制御方式</li> <li>エバリュエーション</li> </ul>			レ レ レ

図 2. 1 構造（アーキテクチャ）の改善項目

-性能向上と開発効率向上のための-

Fig. 2.1 Improvements for architecture

-Performance improvement and developmental efficiency-

## 2. 2 4ビットマイクロプロセッサ

4ビットマイクロプロセッサの生産数の多くを占めたのは、1チップ4ビットマイクロコンピュータと呼ばれるCPU、ROM、RAM、I/Oと周辺を1チップにしたマイクロプロセッサであった。1チップが実現出来たことにより、その価格と必要体積を大幅に小さくすることができ、その応用範囲を大きく広げることになった。それゆえにマイクロプロセッサは各種機器に必要な機能を取り入れ、また性能向上を計り短期開発をする必要が出てきた。

4ビットマイクロプロセッサは特に映像機器の中のVTRへの採用が早く、その操作性向上と性能向上機能拡大にこの4ビットマイクロプロセッサほど効果を出せるものはなく、他の半導体素子や機構部やテープの磁性体では及ばなかった。4ビットマイクロプロセッサの性能向上と開発効率向上の2つの面からの改善改革について、実際に4ビットマイクロプロセッサを設計開発し確認した[5、48]。

### 2.2.1 性能向上

マイクロプロセッサの性能向上を実現するためCPUの高速化とハードウェア量の増強を計った。また、これらの性能向上策はマイクロプロセッサを使い製品開発を行ううえにおいても利便性はあるが、特に利便性に有効なアセンブリ言語の可読性向上についても述べる。

#### (1) CPU高速化

・1語9ビットの命令コード：4ビット・マイクロプロセッサでは命令コードは8ビット(バイト)で1語を構成するのが一般であるが9ビットを1語にした。1語を8ビットより1ビット増やすことにより2ワード、3ワード命令を少なく1ワード命令を多くした命令セットにする事ができた。また命令コードの1ビット増加により1語命令の機能向上が計れ、それらの命令の使用頻度が増え、その結果ソフトウェアのアセンブラのステップ数削減で高速化すると共にソフトウェア設計者の負荷も削減することができた。

具体化した仕様は、ページアドレスを一般に4ビットマイクロプロセッサでは

64アドレスのところ128アドレスに増加させることが出来た。それは1語命令のジャンプ命令 B (Branch) と1語命令のサブルーチンコール命令 BM (Branch Mark) を各128コード合計256コードを配することが出来たためである。すなわち9ビットコードの半分をジャンプ命令とサブルーチンコール命令に割付けたことになる。またRAMアドレスも64語(1語4ビット)1命令LXY (Load X register Y register) でXレジスタとYレジスタを同時に設定可能にした。

・複数機能命令の採用：メモリとレジスタAとの間のデータ転送命令において、複数の機能を持たせることにより、命令コードを大幅に削減させると伴に処理時間短縮を実現させた。複数機能命令として XAMI j の場合この命令1語で本来なら4つの命令に相当する機能を実行する。4つの機能とは、図2. 2に示すようにレジスタAとメモリの値(データポイント(DP)により指定される)を交換する《a》、データポインタ(DP)を構成する下位4ビットのYレジスタの値を+1する《b》、データポインタ(DP)を構成する上位2ビットのXレジスタの値を指定の値0、1、2、3の中から1つと排他的論理和をとる《c》、レジスタYの値が+1された結果0になると命令1語スキップを発生させる。それで複数機能命令を使い、図2. 3に示す最長16語の2つのデータファイルを6命令で交換することが出来る。これはRAMの2つの領域10~1Fと30~3Fの各16語(4ビット)単位のデータはレジスタAを介して交換される。プログラムは命令のTAM1からBまでを16回繰り返し、16回目でB(ブランチ)命令をスキップしてこのルーチンを終了する。TAM 1、XAM 1、XAMI 1のオペランドのイミディエイト値は、レジスタXと排他的論理和をとり、データポインタ(DP)のRAMアドレス領域を変更する。

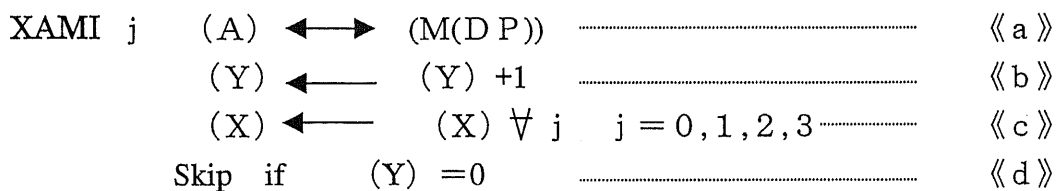


図2. 2 複数機能命令 XAMI j

Fig2.2 Multi function Instruction XAMI



LZ     0     )     アドレスデータを指示する  
 LXY    3 0   )     データポインタを30番地(16進)  
                   )     初期設定する。

ABC    TAM    1     (A) ← (M (DP )), (X) ← (X) ∇ 1  
           XAM    1     (A) ↔ (M (DP )), (X) ← (X) ∇ 1  
           XAMI   1     (A) ↔ (M (DP )), (X) ← (X) ∇ 1  
                           (Y) ← (Y) + 1、Skip if (Y) = 0  
       B   ABC     ABC 番地にジャンプする

・プログラム例

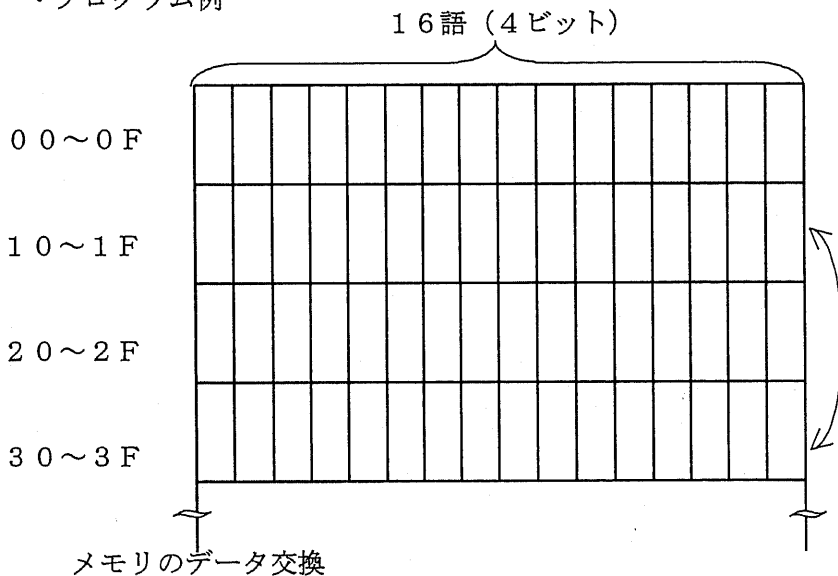


図2.3 複数機能命令によるメモリデータの交換プログラム

Fig2.3 Program example of memory data exchange

by multi function instruction

## (2)ハードウェア量の増強

命令と構成を工夫することにより実質のハードウェア量の増強を行ったと等価な効果を実現した。

・サブルーチン格納専用ページの設定：プログラムコード削減のため1語長のサブルーチン呼出命令を設定した。一般にサブルーチン呼出命令はサブルーチンのアドレスを指定する必要上、最小2語長を必要とする。そこでサブルーチン格納専用ページを設定することにより1語長のサブルーチン呼出命令を設定することができた。このサブルーチン格納専用ページは128ステップ（語）あるが、この専用ページより1語でページ外分岐ができる拡張ページを1ページ設定することで、さらにこの機能を強化した。このサブルーチン格納専用ページの機能を図2.4に示す。1語長のサブルーチン呼出命令（BM）は任意のページよりサブルーチン格納専用ページのSUB1のルーチンを実行する《a》。14ページ、15ページにおいて1語のサブルーチン呼出命令（BM）は14ページ内の分岐命令《e》、もしくは14ページへのページ外分岐命令《d》として働く。また1語の分岐命令（B）は15ページ内の分岐命令《c》、もしくは15ページへのページが分岐命令《b》として働く。

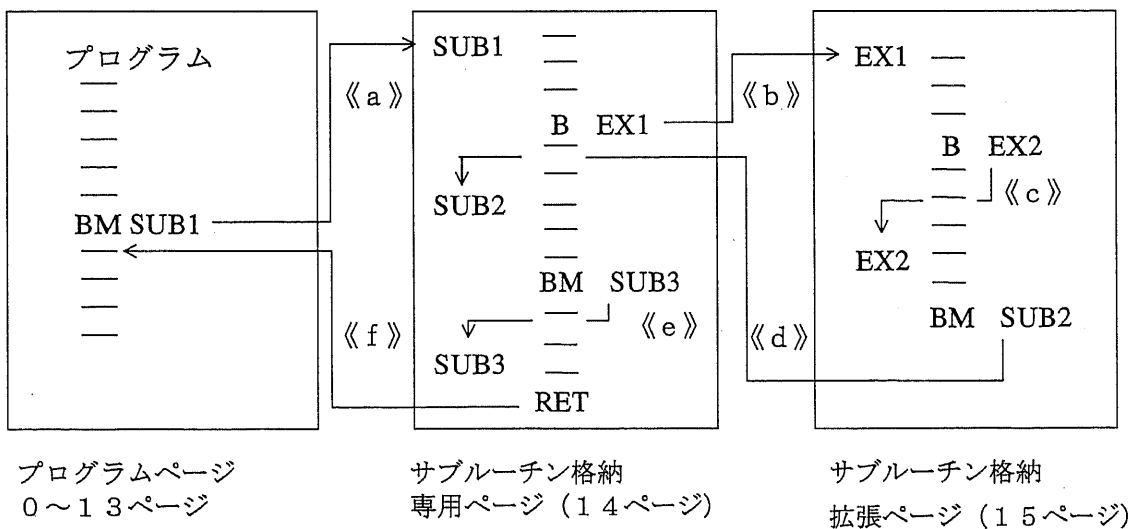


図2.4 サブルーチン格納専用ページの機能

Fig.2.4 Function of exclusive program page for subroutine

### (3) 可読性のよいアセンブリ言語

ソフトウェア開発で見通しよいプログラムをアセンブリで書くために、新たな二種類の命令として多分岐命令と連続記述命令を設定し、また可読性向上のためのアセンブラ言語仕様を設定した。また多分岐命令と連続記述命令は ROM コード (プログラム) 削減にも有効である。

・多分岐命令：分岐命令とサブルーチン呼出命令において、これらのアドレスの下位 4 ビットがレジスタ A の値に置換えられる命令を設定した (BA, BLA, BMA, BMLA)。レジスタ A に変数を入れるルーチンを置くことにより多分岐させることができ、サブルーチン実行時に変数をメインルーチンに容易に入れることが可能で、分かり易い可読性の良いアセンブリプログラムが出来を得ることができる。

・連続記述スキップ命令：RAM アドレス指示レジスタのデータポインタ (DP) とレジスタ A にイミディエイト値を設定する命令 LXY (Load X register Y register) と LA (Load A register) は連続記述されていると 2 つ目以降異なる命令が来るまでスキップする。すなわち最初に記述された命令のみが有効になる。これは 1 つの処理ルーチンに、条件に応じてパラメータを与えるときに有効で、プログラム ROM の削減とプログラムのシーケンスが単純になり、プログラムの可読性が向上する。

・可読性の向上アセンブラ言語：ニーモニックを 1 文字、2 文字、3 文字、4 文字の 4 種類にしたことによりソフトウェアのプログラムリストは処理ごとにそれぞれ異なる模様を生じ可読性の向上に寄与することになる。その具体例を図 2. 5 示す。次いでそれらアセンブリ言語 (命令) の構成において 1 文字目は命令の機能を表し、2 文字目、3 文字目、4 文字目はレジスタ、メモリまたは 1 文字目の機能を補完する英文字とした。図 2. 6 に各命令のニーモニックとその定義を示す。A は Add すなわち加算を表し、命令として A、AM、AMC、AMCS と加算命令のグループを作る。次の B は Branch、すなわち分岐を表す。BM は Branch Mark、分岐してもどり先を Mark する。すなわちサブルーチン読出命令を表す。そして L は Long、すなわちページ外を表す。A は Register A Modify すなわち分岐先番地修飾である。分岐命令、サブルーチン読出命令は、これら B、M、L、A の 4 文字を組合せ B、BM、BL、BML、BA、BMA、BLA、BMA の 8 つの命令を設定した。T は Transfer すなわち、転送、X は Exchange すなわち、データ変換で、先の

A 及び B と同じく、命令がいくつかのアルファベットにより構成されることになる。

	LA	0
	TBA	
	LXY	1,0
	LZ	0
	LCPS	1
ABC1	TAM	2
	XAMI	2
	B	ABC1
	AMCS	
	LXY	2,0
	XAM	1
	A	5
	LXY	0
	LXY	1
	LXY	2
	LXY	3
	BM	ABC2

図 2. 5 アセンブラ言語の可読性向上

Fig.2.5 Improvement of assembly language

1文字目の ニーモニック意味	ニーモニッ ク(命令)	ニーモックの定義	
A: Add 加算	A	<u>A</u> dd register <u>A</u> to immediate	
	AM	<u>A</u> dd register <u>A</u> to <u>M</u> emory	
	AMC	<u>A</u> dd register <u>A</u> to <u>M</u> emory with <u>C</u> arry	
	AMCS	<u>A</u> dd register <u>A</u> to <u>M</u> emory with <u>C</u> arry and if Carry=1 <u>S</u> kip	
B: Branch 分岐	B	<u>B</u> ranch page inside	
	BL	<u>B</u> ranch <u>L</u> ong (page outside)	
	BA	<u>B</u> ranch modifier with register <u>A</u>	
	BLA	<u>B</u> ranch <u>L</u> ong modifier with register <u>A</u>	
	BM	<u>B</u> ranch <u>M</u> ark	
	BML	<u>B</u> ranch <u>M</u> ark <u>L</u> ong	
	BMA	<u>B</u> ranch <u>M</u> ark modifier with register <u>A</u>	
	BMLA	<u>B</u> ranch <u>M</u> ark <u>L</u> ong modifier with register <u>A</u>	
	T: Transfer 転送	TAB	<u>T</u> ransfer data to register <u>A</u> from register <u>B</u>
		TBA	<u>T</u> ransfer data to register <u>B</u> from register <u>A</u>
TAY		<u>T</u> ransfer data to register <u>A</u> from register <u>Y</u>	
TLA		<u>T</u> ransfer data to register <u>L</u> from register <u>A</u>	
THA		<u>T</u> ransfer data to register <u>H</u> from register <u>A</u>	
TEAB		<u>T</u> ransfer data to register <u>E</u> from register <u>A</u> and register <u>B</u>	
TEPA		<u>T</u> ransfer data to register <u>E</u> <u>P</u> ass decoder from register <u>A</u>	
TAJ		<u>T</u> ransfer data to register <u>A</u> from register <u>J</u>	
TAM		<u>T</u> ransfer data to register <u>A</u> from <u>M</u> emory	
TYA		<u>T</u> ransfer data to register <u>Y</u> from register <u>A</u>	
X: Exchange	XAL	<u>E</u> xchange register <u>A</u> and register <u>L</u>	
	XAH	<u>E</u> xchange register <u>A</u> and register <u>H</u>	
	XAM	<u>E</u> xchange register <u>A</u> and <u>M</u> emory, その他	
	XAMD	<u>E</u> xchange register <u>A</u> and <u>M</u> emory, <u>D</u> ecrement register <u>Y</u>	
	XAMI	<u>E</u> xchange register <u>A</u> and <u>M</u> emory, <u>I</u> ncrement register <u>Y</u>	

図 2. 6 アセンブラ言語 可読性向上のためのニーモニック

Fig.2.6 Signal mnemonic for the improvement of assembly language

## 2. 2. 2 マイクロプロセッサ開発効率向上

マイクロプロセッサ開発効率向上を計るためシステム構成（アーキテクチャ）、制御方式、命令セットについて具体的な改善について述べる。

### （1）システム構成

システム構成の改善としてデータバスの複数化、レジスタの複数化、プログラムカウンタとデータポインタの同一レジスタファイル化の3点である。

CPU の内部データ転送は一般に1組のデータバスで行われるのが一般的であるが、図2、7に示すように3つのデータバスを持たせた。命令実行の高速化とデータ転送の命令機能実現の順次制御の最適解を得やすくすることができた。この事により同時に3組のデータ転送が可能になり、また3組のバスは3組のバススイッチでお互いを結合させデータ転送の自由度の拡大とデータ転送効率の向上とデータバスとそれにつながるレジスタ、メモリ、ALUの結合論理回路の削減を可能にした。このように複数のデータバスは高速化の1つの手法として有効と考える。

レジスタファイルと名づけた複数のレジスタを1まとめにしたものを2組設定した。1つはプログラムカウンタ（PC）と3つのスタック（STAK）及び2つのデータ（DP）、他方はレジスタA、BA、TEAPである。レジスタファイルにすることにより、各レジスタとデータバスの入出力回路の削減に寄与できる。また、プログラムカウンタ（PC）のレジスタファイルではセクタで4つの中から1つをプログラムカウンタ（PC）に選択する方法をとり、サブルーチン呼出時の処理が非常に明確で簡単にする事が出来た。またデータポインタ（DP）も同じくセクタで2つのレジスタの一方を選択する方法をとり同じく制御が簡明である。

またプログラムカウンタ（PC）とデータポインタ（DP）を同一レジスタファイルにすることによるもう1つの有効なことはROMとRAMのアドレスバスを共通にしてシステム構成をハードウェアの少ない（アドレスバスと各レジスタからの入出力回路とそれらの制御回路の削減）構成を実現することができた。

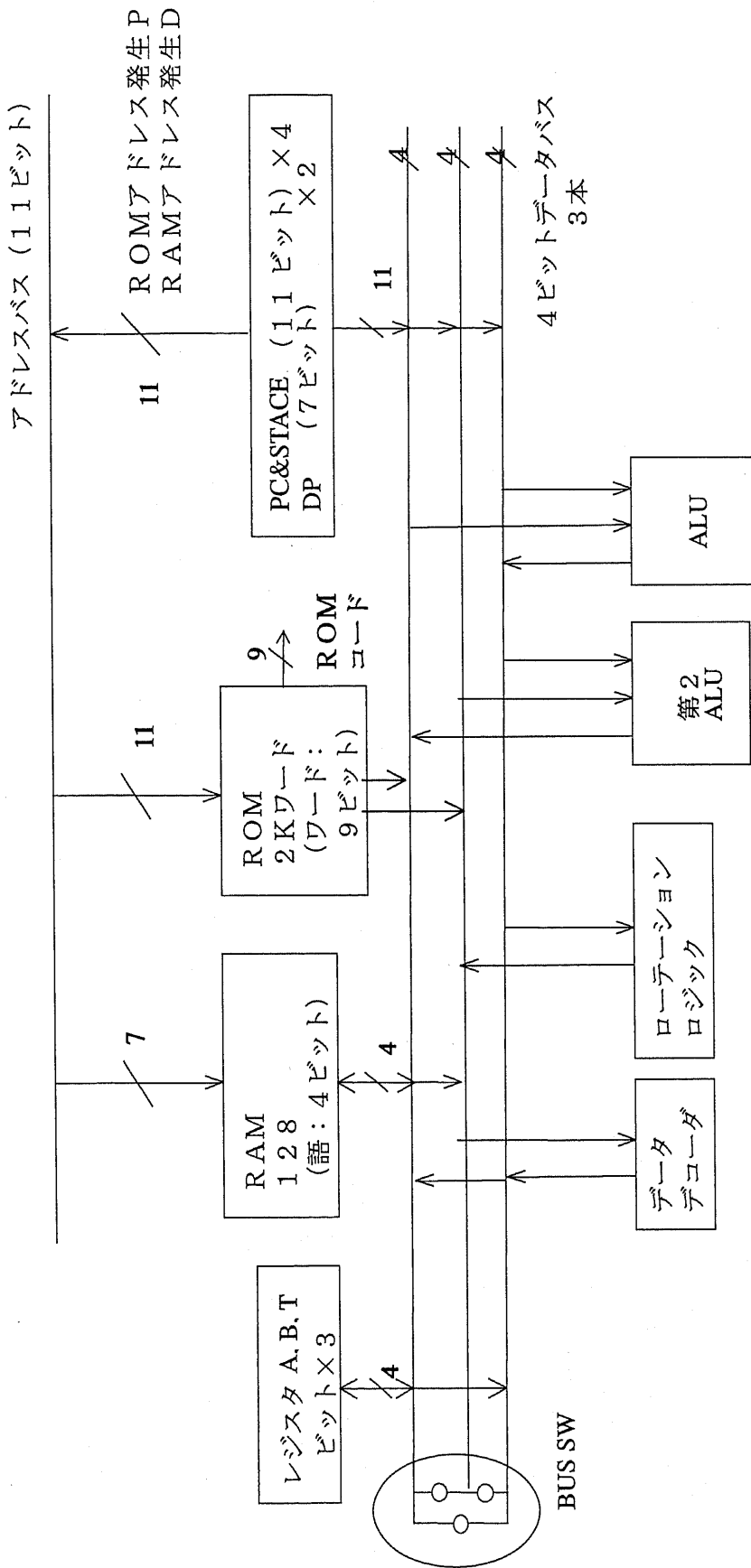


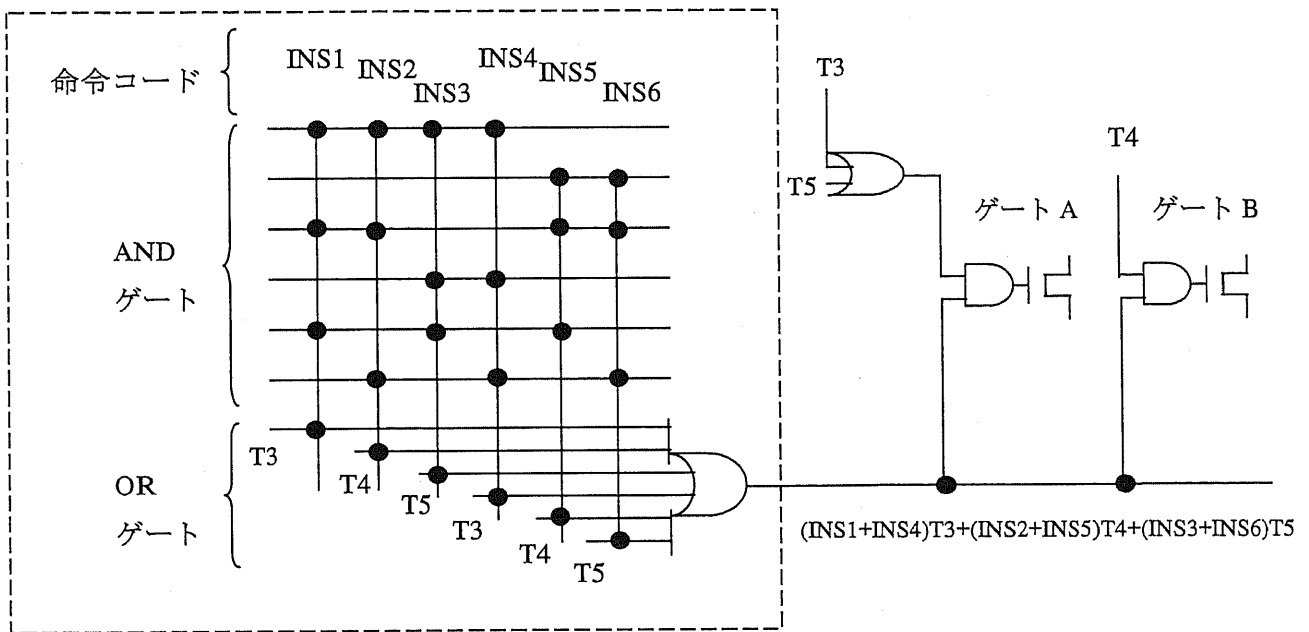
図 2. 7 4ビットマイクロプロセッサのデータ転送経路図

Fig.2.7 Data pass of 4 bit microprocessor

(2) 制御方式

命令実行において命令コードのデコードと各制御部対応の制御線に論理和をとるのに図2. 8に示すようにPLA (Program Logic Array) を採用した。そして制御タイミングを重畳し、制御線の時分割使用により制御を制御線に集中一元化した。

これにより制御線の数削減しチップサイズの縮小を実現する事ができた。また設計時の見通しがよくなり、設計時の不具合を減少させることができる [109]。



PLA にタイミング

図2. 8 制御信号発生集中化と制御信号の時分割使用

Fig. 2.8 Concentration of control signal generation and control line used time sharing



### (3) 命令セット

CPU のハードウェアの構成を簡明にしたので、命令の語長は1語、2語、3語の3種類あるが、CPUの制御の仕組みを無理なく1語長1タイプにすることができた。2語、3語の命令は各々独立している1語長命令を2語または3語組み合わせた構成にした。実行においては割込みの発生を2語および3語命令の終了まで待たすことが必要である。図2. 9に2語命令と3語命令の構成を示す。3語命令はBLAとBMLAの2つ、2語命令はBL、BML、BAとBMAの4つである。001のコードはレジスタA修飾のコード、007~07Fのコードはページ外分岐のコードで、これらに続いて命令B、命令BMが来ることにより2語命令、3語命令として働く。3語命令も2語命令も最後の命令コードで完全に3語命令として、または2語命令とし実行される。もし最後の命令コードが何らかの理由で命令デコーダにフェッチされない場合は、それまで命令は何ら影響ないノーオペレーション (No Operation) として働く。

図2. 10に3語命令、2語命令の実行中に割込が入ってもその割込を発生させない方法を示す。BLA (3語命令) の実行では第1語の001コード (分岐におけるレジスタA修飾命令) 実行において第2語が終了しないと割り込みを発生させない。また第2語の070~07Fコード (ページ外分岐のページ指定) 実行において第3語が終了しないと同様に割り込みを発生させない仕組みを考案した。

命令		コード		
		第1語	第2語	第3語
3語	BLA	(A) 001	(L) 070~07F	(B) 180~18F
	BMLA	(A) 001	(L) 070~07F	(BM) 100~17F
2語	BL	(L) 070~07F	(B) 180~18F	-
	BML	(L) 070~07F	(BM) 100~17F	-
	BA	(A) 001	(B) 180~18F	-
	BMA	(A) 001	(BM) 100~17F	-
1語	B	(B)180~18F	-	-
	BM	(BM)100~17F	-	-

図2.9 3語命令、2語命令の構成

Fig.2.9 Organization of 3 and 2 word instruction

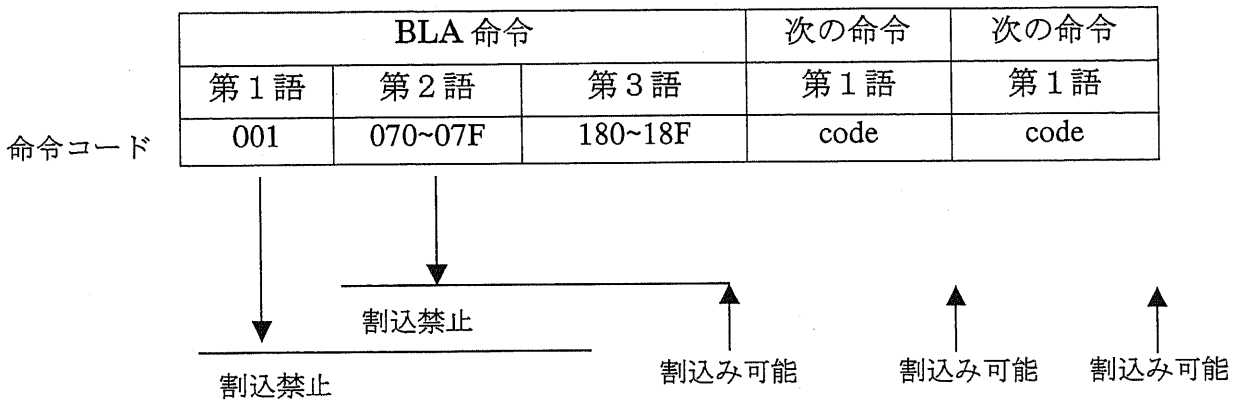


図2.10 3語、2語命令実行中における割り込み禁止の仕組み

Fig.2.10 Contrive of interrupt disenable at execute 2and 3 word instruction

### 2.2.3 評価

4ビットマイクロプロセッサの開発人工は46人月で、10ヶ月の期間を要した。これらの構成のブロック図を図2.11に、また性能機能概要を図2.12に示す。また基本命令数は68種で命令一覧は図2.13に、命令コード対応表を図2.14に示す。

(1) 性能向上

・プログラム実行の高速化

前述した図 2. 3 のメモリデータの交換プログラムで、従来の複数機能命令を持たない 4 ビットマイクロプロセッサと実行時間で比較した。1 命令の実行時間は  $10 \mu\text{sec}$  とした。

今回開発したマイクロプロセッサ (M58840 シリーズ) の場合; 2 命令のイニシヤル設定と 3 命令のデータ転送を 16 回くり返す。

$$(2 + 16 \times 3) \text{ 命令} \times 10 \mu\text{sec} / \text{命令} = 500 \mu\text{sec} \dots \dots (a)$$

従来のマイクロプロセッサの場合; 4 命令のイニシヤル設定と 8 命令のデータ転送を 16 回くり返す。

$$(4 + 16 \times 8) \text{ 命令} \times 10 \mu\text{sec} / \text{命令} = 1320 \mu\text{sec} \dots \dots (b)$$

$$b / a = 2.64 \dots \dots (c)$$

この仕様によるメモリ交換プログラムでは 2.64 倍の処理速度の高速化を実現したことになる。

・ROM コード効率

ROM コード効率は、特にサブルーチン専用ページと複数機能命令が有効であった。ROM コードの効率は従来使用した ROM コード数の 45.7% まで減少した値を得た。この 4 ビットマイクロプロセッサで開発したソフトウェアの命令数は 2048 ROM コード数であった。このソフトウェアのサブルーチン専用ページと複数機能命令のない従来の 4 ビットマイクロプロセッサを想定して必要命令数を算出すると 4485 ROM コード数が必要となった。

$$2048 \text{ ROM コード数} \div 4485 \text{ ROM コード数} \times 100 = 45.7\%$$

(2) マイクロプロセッサの開発効率向上

基本機種開発人工は 46 人月、展開機種開発は 30 品種で 300 人月であった。展開機種とは ROM、RAM のメモリ容量とタイマなどの周辺機能のラインナップである。従来の開発効率向上を考えていないマイクロプロセッサ構造のもので開発人月を推定すると、基本機種開発では 100 人月で展開機種開発は 1 件平均

24人月要し30品種で720人月となる。

本開発の場合、基本形開発1件46人月と展開開発30品種で300人月。

合計346人月・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・(a)

従来方式での開発の場合、基本形開発1件100人月と展開開発30品種で720人月。

合計820人月・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・(b)

今回の本開発と従来方式での開発人工比は  $a / b \times 100 = 42\%$

すなわち従来の構造（アーキテクチャ）時の開発に比べ58%の人月の削減で42%の人月で開発を可能にしたことになる。

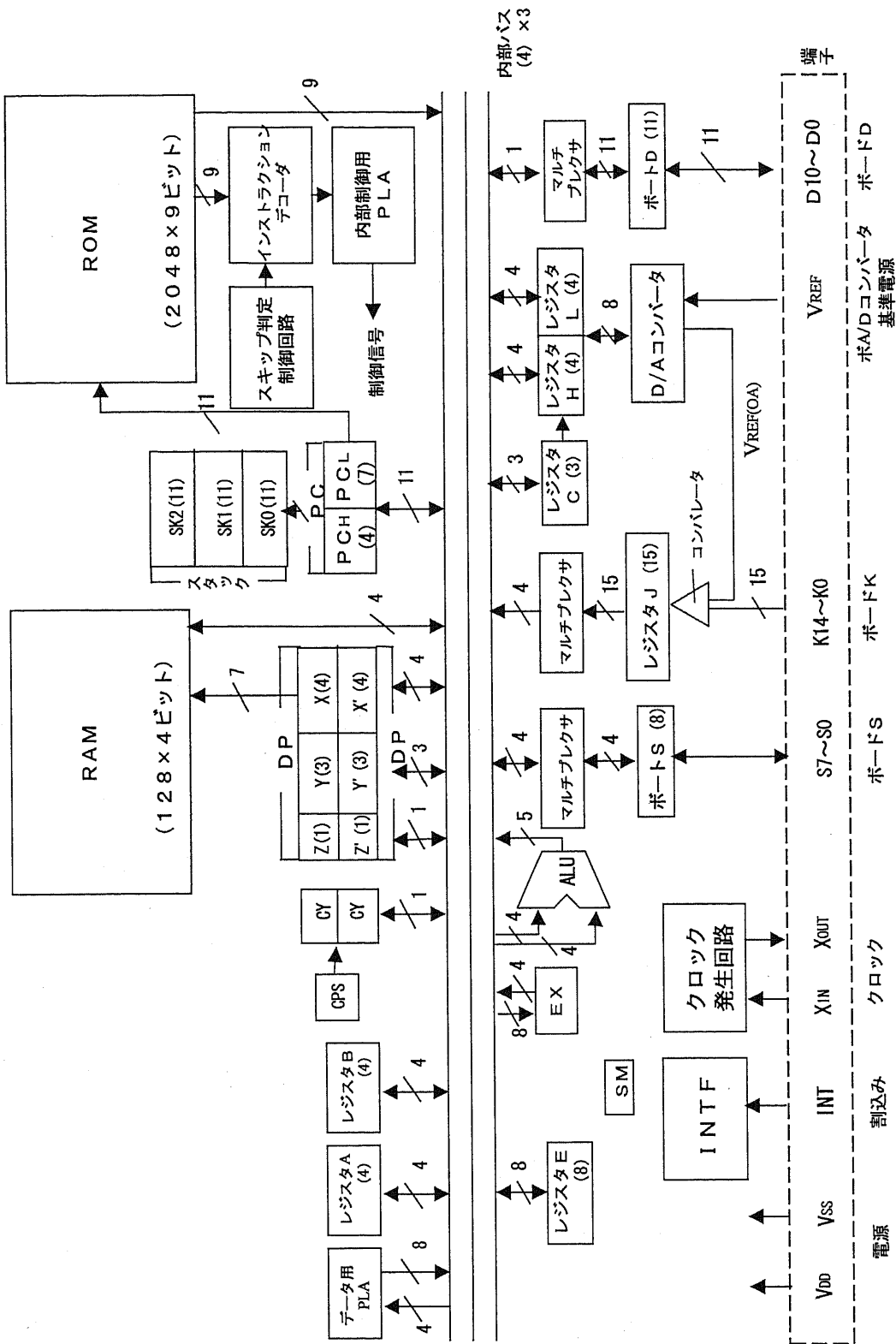


図 2.11 ブロック図

Fig. 2.11 Block diagram

項目	機種名	M58840-x x x P	
プロセッサ		PチャンネルE/D	
命令数		68種	
命令実行時間		10s	
サブルーチンネスティング数		3段(含割込み1段)	
クロックゼネレータ		内蔵(CRまたはセラミックフィルタ外付)	
A/Dコンバータ		内蔵(8ビット15チャンネル)	
メモリ容量	ROM	2,048×9ビット	
	RAM	512ビット	
ポート	入力	タッチキー	15ビット(タッチキーインターフェース内蔵)
		ポートS	4ビット×2
		ポートD	11ビット
	出力	ポートS	8ビット*
		ポートD	11ビット**
		入出力耐電圧	-33V
入出力特性	出力電流	-8mA/ポート出力、-15mA/ ディスクリート出力	
	電源	-15V	
パッケージ		42ピンプラスチック	
消費電力		250mA(標準)	
動作温度		-10~75℃	

\*, \*\* 大形蛍光表示管直接駆動可能

\* 大形LED直接駆動可能

図2.12 性能機能概要

Fig2.12 System performance and function

名前	モード	モック	長	マシンワード	機能	スキップ条件	CY
レジスタ間転送命令	TAB	1	1	(A)←(B)	-	-	-
	TBA	1	1	(B)←(A)	-	-	-
	TAY	1	1	(A)←(Y)	-	-	-
	TLA	1	1	(L)←(A)	-	-	-
	THA	1	1	(H)←(A)	-	-	-
	TEAB	1	1	(E <sub>7</sub> )←(E <sub>6</sub> )←(E <sub>5</sub> )←(E <sub>4</sub> )←(A)	-	-	-
	TEPA	1	1	(E <sub>7</sub> )←(E <sub>6</sub> )←データ用PLA←(A)	-	-	-
	TAJ	1	1	(Y <sub>1</sub> Y <sub>0</sub> )←(A)←(J <sub>3</sub> ,J <sub>2</sub> ,J <sub>1</sub> ,J <sub>0</sub> ) (Y <sub>1</sub> Y <sub>0</sub> )←(A)←(J <sub>7</sub> ,J <sub>6</sub> ,J <sub>5</sub> ,J <sub>4</sub> ) (Y <sub>1</sub> Y <sub>0</sub> )←(A)←(J <sub>11</sub> ,J <sub>10</sub> ,J <sub>9</sub> ,J <sub>8</sub> ) (Y <sub>1</sub> Y <sub>0</sub> )←(A)←(X <sub>7</sub> ,J <sub>14</sub> ,J <sub>13</sub> ,J <sub>12</sub> )	-	-	-
	XAL	1	1	(A)←(L)	-	-	-
	XAH	1	1	(A)←(H)	-	-	-
レジスタ間転送命令	TAM j	1	1	(A)←(M(DP)) (X)←(X)V <sub>j</sub> (j=0~3)	-	-	-
	XAM j	1	1	(A)←(M(DP)) (X)←(X)V <sub>j</sub> (j=0~3)	-	-	-
	XAMD j	1	1or2	(A)←(M(DP)), (Y)←(Y)-1 (X)←(X)V <sub>j</sub> (j=0~3)	(Y)=15	-	-
	CMA	1	1or2	(A)←(M(DP)), (Y)←(Y)+1 (X)←(X)V <sub>j</sub> (j=0~3)	(Y)=0	-	-
	TXA	1	1	(Y)←(A)	-	-	-
アドレス命令	LXY x,y	1	1or2	(X)←x, (Y)←y (Y)←y, (y=15~)	連続記述	-	-
	LZ z	1	1	(Z)←(Z) (z=0,1)	-	-	-
	INV	1	1or2	(Y)←(Y)+1	(Y)=0	-	-
	DEY	1	1or2	(Y)←(Y)-1	(Y)=15	-	-
	LCPSI	1	1	(CPS)←i (i=0,1)	-	-	-
演算命令	LAN n	1	1	(A)←n n=15~0	連続記述	-	-
	AM	1	1	(A)←(A)+(M(DP))	-	-	-
	AMC	1	1	(A)←(A)+(M(DP))+(CY) (CY)←キャリー-	-	S/R	-
	AMCS	1	1or2	(A)←(A)+(M(DP))+(CY) (CY)←キャリー-	(CY)=1	S/R	-
	A n	1	1or2	(A)←(A)+n	キャリー=0 (ただし=64除)	-	-
	BC	1	1	(CY)←1	-	S	-
	RC	1	1	(CY)←0	-	R	-
	SZC	1	1or2	-	(CY)=0	-	-
	CMA	1	1	(A)←(A)	-	-	-
	命令比較	SEAM	1	1or2	-	(A)=(M(DP))	-
SEY y		1	1or2	-	(Y)=y y=15~0	-	-
操作命令	SB j	1	1	(M <sub>j</sub> (DP))←1 j=3~0	-	-	-
	RB j	1	1	(M <sub>j</sub> (DP))←0 j=3~0	-	-	-
	SBZ j	1	1or2	-	(M <sub>j</sub> (DP))=0 j=3~0	-	-
割込命令	EH	1	1	INTF←1	-	-	-
	DI	1	1	INTF←0	-	-	-
	INTH	1	1	Accept INT="High"	-	-	-
	INTL	1	1	Accept INT="Low"	-	-	-
A:レジスタA、B:レジスタB、C:レジスタC、E:レジスタE、H:レジスタH、L:レジスタL、J:レジスタJ、X:レジスタX、Y:レジスタY、Z:レジスタZ、DP:データポインタ(レジスタX、レジスタY、レジスタZで構成される)、GPS:2組のDPとCYを選択するためのセレクト、PC:プログラムカウンタ、SK:スタック、CY:キャリーフラグ、D:ボードD、J:ボードJ、S:ボードS、SM:サブルーチンボード、フリップフロップ							

名前	モード	モック	長	マシンワード	機能	スキップ条件	CY
R O M アドレス命令	BL	1	1	(SM)=0: (PCL)←L6~L0 (SM)=1: (PCH)←F16 (PCL)←L6~L0	-	-	-
	BML	1	1	(SM)=0: (SK2)←(SK1)←(SK0) ←(PC), (PCH)←E14, (PCL)←L6~L0, (SM)=1 (SM)=1: (PCH)←E16, (PCL)←L6~L0	-	-	-
	BLL	2	2	(PCH)←L10~L7 (PCL)←L6~L0 SM=0	-	-	-
	BML L	2	2	(SK2)←(SK1)←(SK0)←(PC) (PCH)←L10~L7, (PCL)←L6~L0, SM=0	-	-	-
	BAL	2	2	(SM)=0: (PCL)←L6~L4, A3~A0 (SM)=1: (PCH)←F16 (PCL)←L6~L4, A3~A0	-	-	-
	BMA L	2	2	(SM)=0: (SK2)←(SK1)←(SK0) (PCH)←E16 (PCL)←L6~L4, A3~A0 (SM)=1 (SM)=1: (PCH)←E16 (PCL)←L6~L4, A3~A0	-	-	-
	BLA L	3	3	(PCH)←L10~L7 (PCL)←L6~L4, A3~A0, (SM)=0	-	-	-
	BMLA L	3	3	(SK2)←(SK1)←(SK0)←(PC) (PCH)←L10~L7 (PCL)←L6~L4, A3~A0, SM=0	-	-	-
	RI	1	1	(A)←(L)	-	-	-
	RT	1	1	(A)←(H)	-	-	-
A I D コンバータ命令	RTS	1	2	(A)←(M(DP)) (X)←(X)V <sub>j</sub> (j=0~3)	無条件スキップ	-	-
	LC7	1	1	(C)←7	-	-	-
	DEC	1	1or2	(C)←(C)-1	(C)=7	-	-
	SHL	1	1	(C <sub>2</sub> )←(L(C <sub>1</sub> ~C <sub>0</sub> ))-1 (C <sub>2</sub> )←1: (H(C <sub>1</sub> ~C <sub>0</sub> ))-1	(Y)=0	-	-
	RHL	1	1	(C <sub>2</sub> )←(L(C <sub>1</sub> ~C <sub>0</sub> ))-0 (C <sub>2</sub> )←1: (H(C <sub>1</sub> ~C <sub>0</sub> ))-0	-	-	-
	CPA	1	1	(A)←ポートKの入力とD/A コンバータの比較結果	連続記述	-	-
	EZJ	1	1or2	-	-	-	-
	CPAS	1	1	(A)←ポートKの入力とD/A コンバータの比較結果	(J)(Y)=0 但し、(Z)=1.0 ≦(Y)≦1.4	-	-
	CPAE	1	1	CPASの実行を終了	-	-	-
	入出力命令	SD	1	1	(D)(Y)←1 (Z)=1.0 ≦ (Y) ≦ 1.0	-	-
RD		1	1	(D)(Y)←0 (Z)=1.0 ≦ (Y) ≦ 1.0	-	-	-
SZD		1	1	-	(D)(Y)=0 (Z)=1.0 ≦(Y)≦1.0	-	-
OSAB		1	1	(S <sub>7</sub> ~S <sub>4</sub> )←B (S <sub>3</sub> ~S <sub>0</sub> )←A	-	-	-
OSPA		1	1	(S <sub>7</sub> ~S <sub>0</sub> )←データ用PLA←(A)	-	-	-
OSE		1	1	S←E	-	-	-
IAS		1	1	i=0: (A)←(S <sub>7</sub> ~S <sub>4</sub> ) A←j=1: (A)←(S <sub>3</sub> ~S <sub>0</sub> )	-	-	-
CLS		1	1	(S)←0	-	-	-
CLD		1	1	(D)←0	-	-	-
CLDS		1	1	(S), (D)←0	-	-	-
その他	NDP	1	1	No operation	-	-	-

←: データ転送方向、⇄: データ交換、V: 排他的論理和  
+ : 和演算、( ): カッコ内に書かれた内容を示す、サフックス: サフィックスをつけられたもののビットを示す

図 2. 1 3 命令一覧(M58840-XXXXP)  
Fig2.13 Instruction List

D3-D0 D3-D4 標準記	0															1														
	0000	0001	0010	00011	00100	0101	0110	00111	01000	01001	00110	00111	01000	01001	01100	01011	01100	01101	01110	01111	1000	10111	18-1F							
0000	00	NDP	CLS	02	SZB <sub>0</sub>	03	SEY <sub>0</sub>	04	LCPS <sub>0</sub>	05	RU	06	XAM <sub>0</sub>	07	BL BML	08	-	09	0A	0B	0C	0D	0E	0F	10-17	BM	B			
0001	BA BMA BLA BMLA	CLDS	CLD	SZB <sub>1</sub>	SZB <sub>1</sub>	SEY <sub>1</sub>	SEY <sub>1</sub>	LCPS <sub>1</sub>	LCPS <sub>1</sub>	SU	SU	XAM <sub>1</sub>	XAM <sub>1</sub>	BL BML	OFA	-	-	0A	0A	0A	0A	0A	0A	0A	0A	0A	0A	B		
0010	INY	-	-	SZB <sub>2</sub>	SZB <sub>2</sub>	SEY <sub>2</sub>	SEY <sub>2</sub>	SHL	SHL	RHL	RHL	XAM <sub>2</sub>	XAM <sub>2</sub>	BL BML	SNZ <sub>1</sub>	-	-	0A	0A	0A	0A	0A	0A	0A	0A	0A	0A	B		
0011	DEY	CLD	CLD	SZB <sub>3</sub>	SZB <sub>3</sub>	SEY <sub>3</sub>	SEY <sub>3</sub>	AMC	AMC	AMCS	AMCS	XAM <sub>3</sub>	XAM <sub>3</sub>	BL BML	SNZ <sub>2</sub>	-	-	0A	0A	0A	0A	0A	0A	0A	0A	0A	0A	B		
0100	DI	RD	RD	-	-	SEY <sub>4</sub>	SEY <sub>4</sub>	RT	RT	IAS <sub>0</sub>	IAS <sub>0</sub>	TAM <sub>0</sub>	TAM <sub>0</sub>	BL BML	TIAB	-	-	0A	0A	0A	0A	0A	0A	0A	0A	0A	0A	B		
0101	EI	SD	SD	-	-	SEY <sub>5</sub>	SEY <sub>5</sub>	RTS	RTS	IAS <sub>1</sub>	IAS <sub>1</sub>	TAM <sub>1</sub>	TAM <sub>1</sub>	BL BML	TRAB	-	-	0A	0A	0A	0A	0A	0A	0A	0A	0A	0A	B		
0110	-	TEPA	TEPA	SEAM	SEAM	SEY <sub>6</sub>	SEY <sub>6</sub>	RTI	RTI	-	-	TAM <sub>2</sub>	TAM <sub>2</sub>	BL BML	TVA	-	-	0A	0A	0A	0A	0A	0A	0A	0A	0A	0A	B		
0111	-	OSPA	OSPA	-	-	SEY <sub>7</sub>	SEY <sub>7</sub>	-	-	LCZ	LCZ	TAM <sub>3</sub>	TAM <sub>3</sub>	BL BML	TWA	-	-	0A	0A	0A	0A	0A	0A	0A	0A	0A	0A	B		
1000	OPA	XAL	XAL	-	-	SEY <sub>8</sub>	SEY <sub>8</sub>	RC	RC	XAH	XAH	XAMD <sub>0</sub>	XAMD <sub>0</sub>	BL BML	TAB1	-	-	0A	0A	0A	0A	0A	0A	0A	0A	0A	0A	B		
1001	DEO	TILA	TILA	SZJ	SZJ	SEY <sub>9</sub>	SEY <sub>9</sub>	SC	SC	THA	THA	XAMD <sub>1</sub>	XAMD <sub>1</sub>	BL BML	TABR	-	-	0A	0A	0A	0A	0A	0A	0A	0A	0A	0A	B		
1010	AM	TEAB	TEAB	-	-	SEY <sub>10</sub>	SEY <sub>10</sub>	LZ <sub>0</sub>	LZ <sub>0</sub>	-	-	XAMD <sub>2</sub>	XAMD <sub>2</sub>	BL BML	TAB2	-	-	0A	0A	0A	0A	0A	0A	0A	0A	0A	0A	B		
1011	OSE	OSAB	OSAB	SZD	SZD	SEY <sub>11</sub>	SEY <sub>11</sub>	LZ <sub>1</sub>	LZ <sub>1</sub>	-	-	XAMD <sub>3</sub>	XAMD <sub>3</sub>	BL BML	-	-	-	0A	0A	0A	0A	0A	0A	0A	0A	0A	0A	B		
1100	TYA	TBA	TBA	-	-	SEY <sub>12</sub>	SEY <sub>12</sub>	SB <sub>0</sub>	SB <sub>0</sub>	RB <sub>0</sub>	RB <sub>0</sub>	XAMI <sub>0</sub>	XAMI <sub>0</sub>	BL BML	IAF	-	-	0A	0A	0A	0A	0A	0A	0A	0A	0A	0A	B		
1101	TAJ	TAY	TAY	-	-	SEY <sub>13</sub>	SEY <sub>13</sub>	SB <sub>1</sub>	SB <sub>1</sub>	RB <sub>1</sub>	RB <sub>1</sub>	XAMI <sub>1</sub>	XAMI <sub>1</sub>	BL BML	IAK	-	-	0A	0A	0A	0A	0A	0A	0A	0A	0A	0A	B		
1110	-	TAB	TAB	-	-	SEY <sub>14</sub>	SEY <sub>14</sub>	SB <sub>2</sub>	SB <sub>2</sub>	RB <sub>2</sub>	RB <sub>2</sub>	XAMI <sub>2</sub>	XAMI <sub>2</sub>	BL BML	-	-	-	0A	0A	0A	0A	0A	0A	0A	0A	0A	0A	B		
1111	CMA	-	-	SZO	SZO	SEY <sub>15</sub>	SEY <sub>15</sub>	SB <sub>3</sub>	SB <sub>3</sub>	RB <sub>3</sub>	RB <sub>3</sub>	XAMI <sub>3</sub>	XAMI <sub>3</sub>	BL BML	-	-	-	0A	0A	0A	0A	0A	0A	0A	0A	0A	0A	B		

図 2. 1 4 命令コード対応表 M58840 シリズ M58845A-XXXP

Fig 2.14.Measurement table with Instruction Code



## 2. 3 32ビットマイクロプロセッサ

32ビットマイクロプロセッサ性能向上のためのCPUの高速化について述べる。CPUの高速化に寄与するのは、前述したように命令実行時間の短縮、命令ビット幅の増数、命令セットの最適化と乗除算や浮動小数点演算器のハードウェアの内蔵である。

命令実行時間の短縮はパイプライン技術と並列処理技術及びクロック周波数の高速化により大きく向上してきた。パイプライン技術は、さらに分岐予測技術とキャッシュメモリ技術に密接に関係し、また大量データを入出力側にCPUを介さず直接転送するDMA (Direct Memory Access) 機構を生み、アーキテクチャの発展すなわちアーキテクチャによる高速化技術の確立を推進した[22、117]。

ここまでのプロセッサの流れをRISC (Reduced Instruction Set Computer) の出現によりCISC (Complex Instruction Set Computer) と呼び区別されるようになった。RISCもCPUの高速化を目指したもので、それまでプロセッサ、コンピュータ誕生以来発展してきたCISCの流れを大変更するものである。CISCの高機能化した複雑な命令は、多くのハードウェアと内部制御自身をプログラムで制御しようとするマイクロプログラムで実現するため、これらのハードウェアが高速化の足かせになっていた。CISCの高機能化した複雑な命令は2つ以上の単純な命令に分割され、ハードウェアはより少なく簡素にすることによりクロック周波数上げまたコンパイラ技術でより効果的なプログラム設計が計られ、高速化が推進された[30、31]。

CPUの高速化は今後ともまだまだ続くが、RISCのようなアーキテクチャの大改革が出てこない限り半導体製造技術向上による微細化に負うところが大きい。この微細化を効果的に生かすマイクロプロセッサの推定仕様として、1チップで1億個のトランジスタを使うアーキテクチャの検討があり、1チップマルチプロセッサの提案もある[115]。CPUの高速化は、このようにハードウェア規模を大きくすることでも実現できるが、特殊な問題解決のためだけの試みで重要なアーキテクチャの改革につながらない限り、決して引き続くプロセッサにそのアーキテクチャは引き継がれない。

それでここではある特定の応用において有効なアーキテクチャを提示する。こ

の応用とは画像処理で、映像機器や自動車の運転補助機構等の認識に使われるものであり、それらは特定の演算、例えば1秒間に100万回とか1億回以上を繰り返す用途である。また、これらの応用は広く今後ますます増加していく。

### (1) 性能向上におけるマイクロプロセッサのビット幅とクロック周波数

CPUの性能は構造(アーキテクチャ)が同じならクロック周波数とCPUビット幅により基本的に決まる[29]。しかし、具体的な制御対象ではCPU性能はクロック周波数に比例し、ビット幅に対してはそのときの処理内容により異なる。

図2. 15に4ビット、8ビット、16ビット、32ビットの各マイクロプロセッサにおいて、32ビットデータの加算と乗算演算時の実行速度比較を示した。これは、32ビットマイクロプロセッサにおいてクロック周波数1MHzで実行するとき、この同時間で実行するには16ビット、8ビット、4ビットの各マイクロプロセッサではどれだけのクロック周波数が必要かを示したものである。但し各ビット幅のマイクロプロセッサはそれぞれのマイクロプロセッサと同じビット幅の加算器と掛算器を持つとした。結果として加算と乗算では必要クロック周波数比は大きく異なる。4ビットのマイクロプロセッサの場合、加算では32ビットマイクロプロセッサの8倍であるのに対し乗算では約3200倍になる。これは、何を明示しているかと言うと演算性能が不足の場合、その要因としてビット幅の問題か演算性能なのかにより、その最善の対応策が異なる。対応策として現状技術下では、処理アルゴリズム改善でそれがソフトウェアかハードウェア又は、システムで行うかと言うことになる。ビット幅の不足の場合、これらの対応で解決しない場合マイクロプロセッサのビット幅の増大が必要になる。

また一方図2. 16に示すようにCPU性能に対して構造(アーキテクチャ)が同じならクロックの周波数とCPUビット幅にはトレードオフの関係があることが分かる。4ビット、8ビット、16ビット、32ビットプロセッサで現在の組み込み市場で必要とする処理の実行の可否を示したものである。性能を得る必要があればクロックを速くしまたビット幅を増せばいいわけである。しかし、前述した通りビット幅の増大は全ての場合に有効であるとは限らず、またプロセッサのゲート数の増大につながる。

またクロックの高速化は一般に量産レベルで図2. 17に示すごとく半導体技

術向上の結果として5年で2.5倍の率で向上している。図2.18にマイクロプロセッサのビット幅と各CPUのゲート数比較を示す。

CPUの性能を総合的に論じる時、CPUの性能は構造（アーキテクチャ）が同じならクロックの周波数とCPUのビット幅により決まり、それで以上に示した様にCPUの性能向上はかなり正しく予測できている[43]。これらを前提により性能の高いCPUを得るための工夫、この場合構造（アーキテクチャ）の改良、改善と新たな着想の発明が必要である。

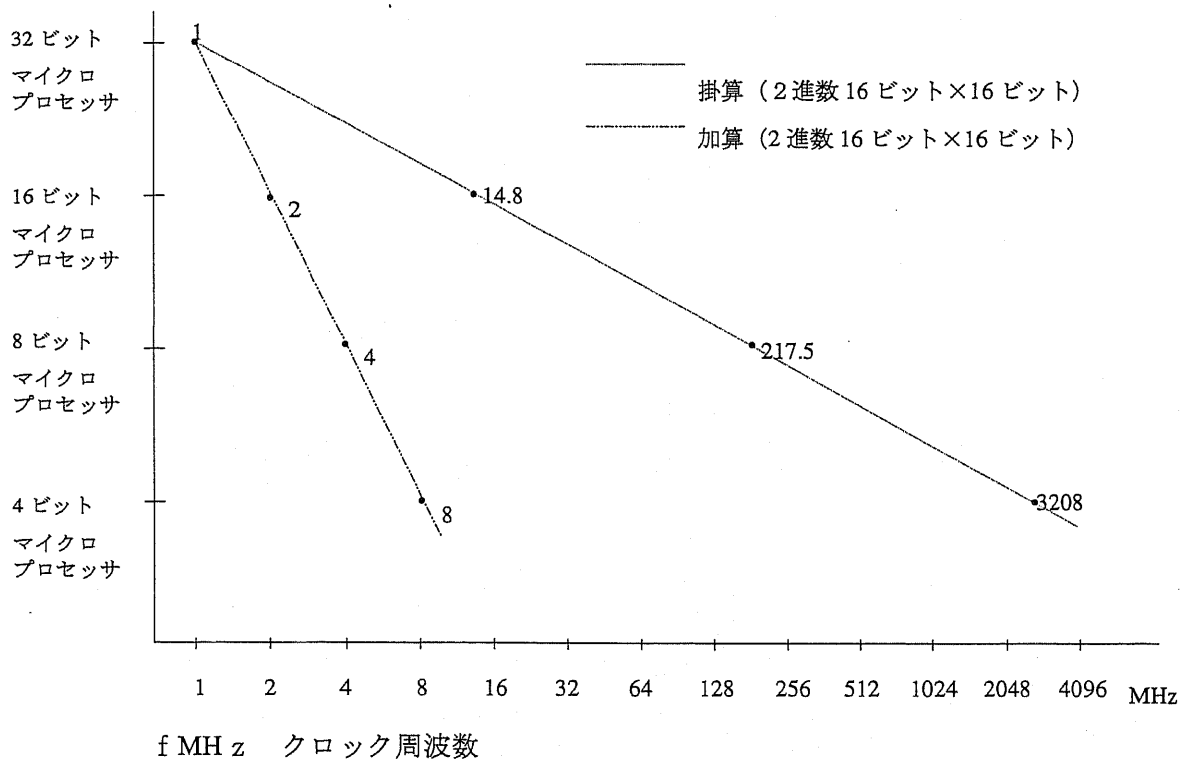
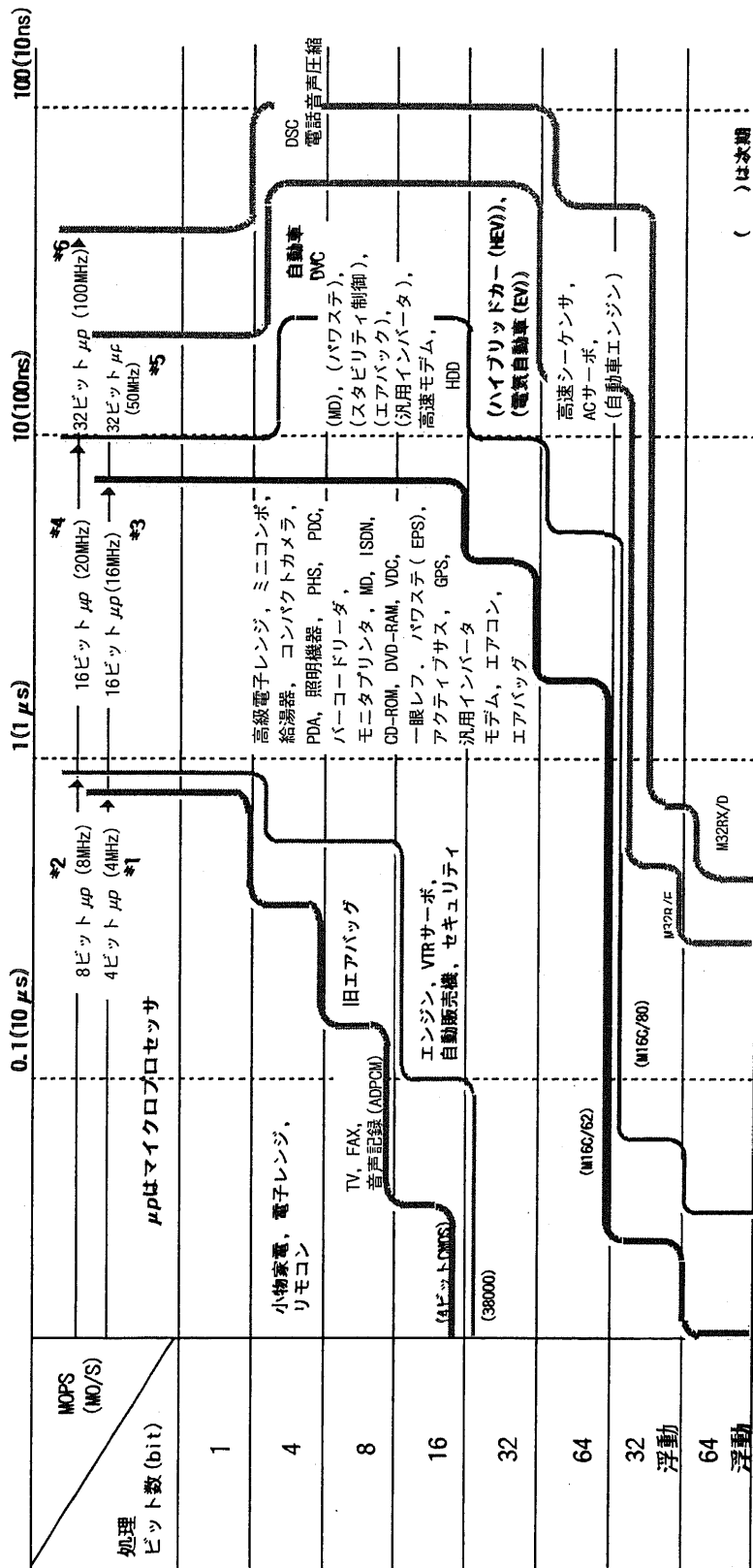


図2.15 マイクロプロセッサのビット幅による処理能力差

Fig.2.15 Difference of the capacity at each bit micro possessor



\*1, \*2, \*3, \*4, \*5 は三菱電機社製マイクロプロセッサで以下シリーズ名を示す。  
 \*1; 4500, \*2; 38000, \*3; M16C/62, \*4; M16C/80, \*5; M32/E, \*6; M32RX/D

図2. 1 6 主要処理のビット巾と必要処理時間 (マイクロプロセッサの組込応用)  
 Fig.2.16 Required processing time and main processing bit width

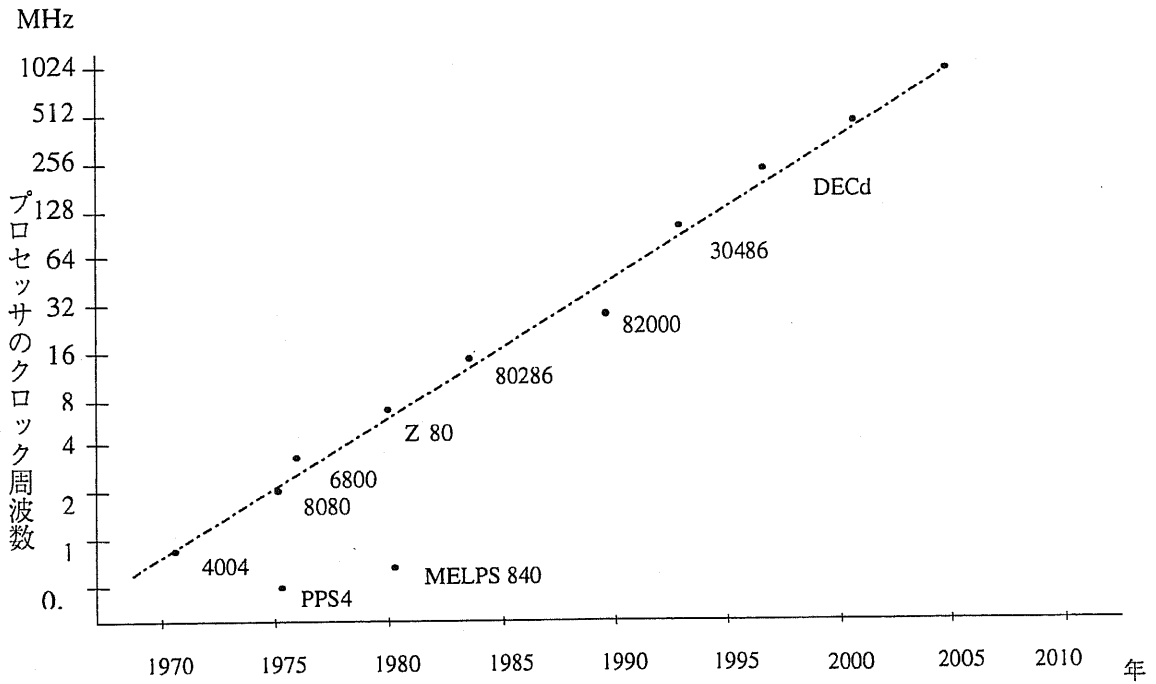


図 2. 17 マイクロプロセッサの周波数トレンド  
 Fig2.17 Microprocessor Frequency Trend

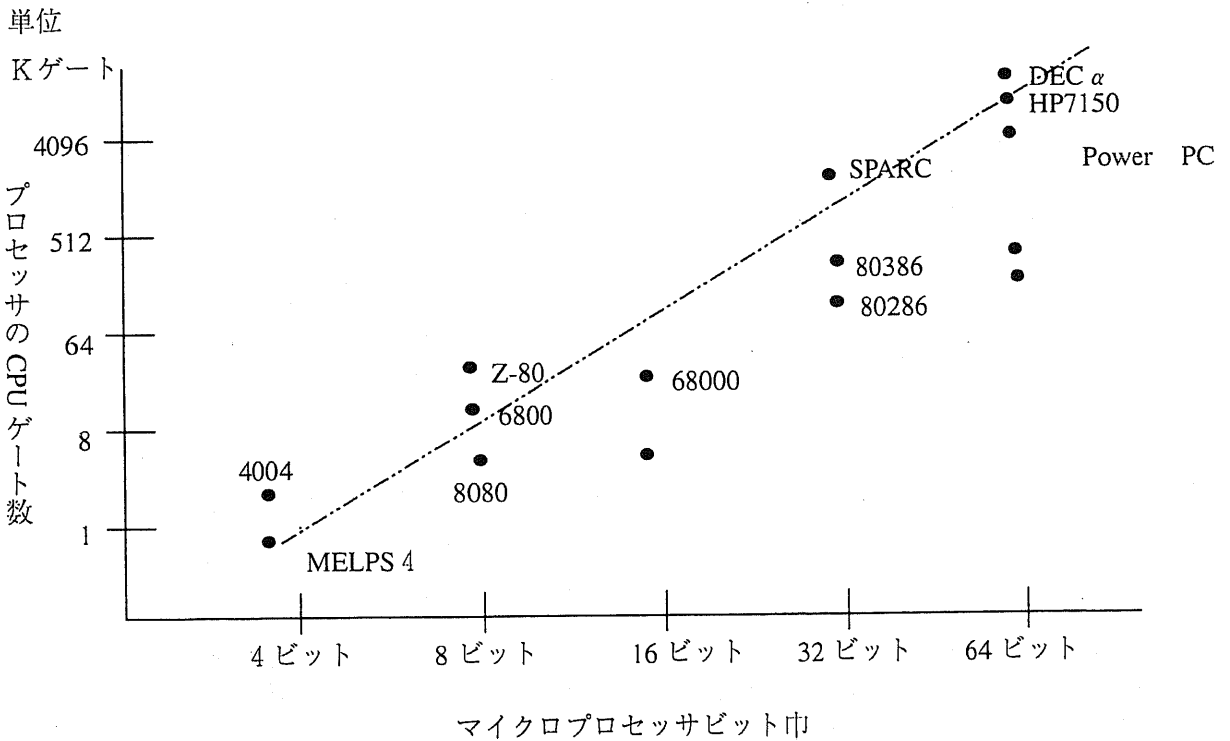


図 2. 18 マイクロプロセッサのビット幅と各CPUのゲート数の比較  
 Fig2.18 Microprocessor bit wide and each CPU gate number comparison

## (2) ALU 増設による性能改善

マイクロプロセッサの性能は、その性能を総合的に評価する場合、構造（アーキテクチャ）が同じなら速度と CPU のビット幅でその性能が決まる。現時点で今後の半導体の製造技術の向上を見込む以外に構造（アーキテクチャ）に因る性能向上の試みは重要である。またそのニーズは、本論においては映像機器の信号処理において大きく、その有効性は全体システムのハードウェアの大幅削減とシステムの簡明化及び開発期間の短縮である。

そこで、構造（アーキテクチャ）の改善、改良または着実な構想による発明として、ALU（Arithmetic and Logical Unit; 演算器）の増設によるマイクロプロセッサの高速化構造を提示する[9]。図 2. 19 に 1 つまたはそれ以上複数の ALU を増設したマイクロプロセッサを示す。増設された ALU 全の総称として第 2 ALU（The second ALU ; 2nd ALU）とする。第 2 ALU の構造を図 2. 20 に示すように演算部、演算される複数の演算子と被数及び演算結果を格納するデータメモリ、演算は連続実行できるがそのときのステップごとに演算と演算子、被数の格納アドレスを決める実行テーブル、ステップのシーケンスを決める制御レジスタとマイクロプロセッサの共通バスと接続するバススイッチより構成されている。

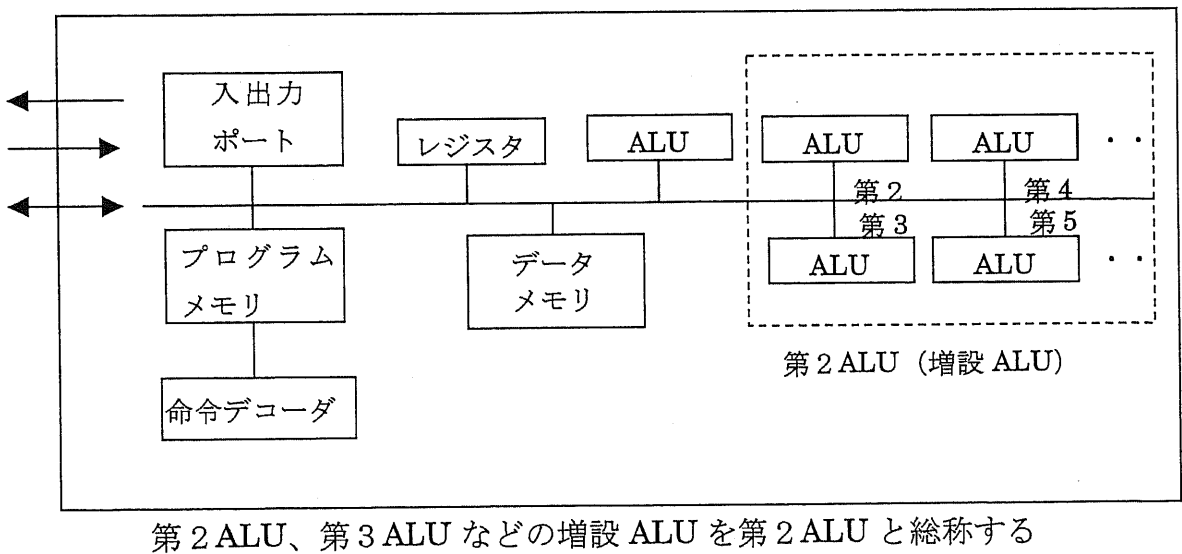


図 2. 19 1 つまたは複数の増設 ALU を持つマイクロプロセッサ  
Fig.2.19 Microprocessor with one or some 2nd ALU

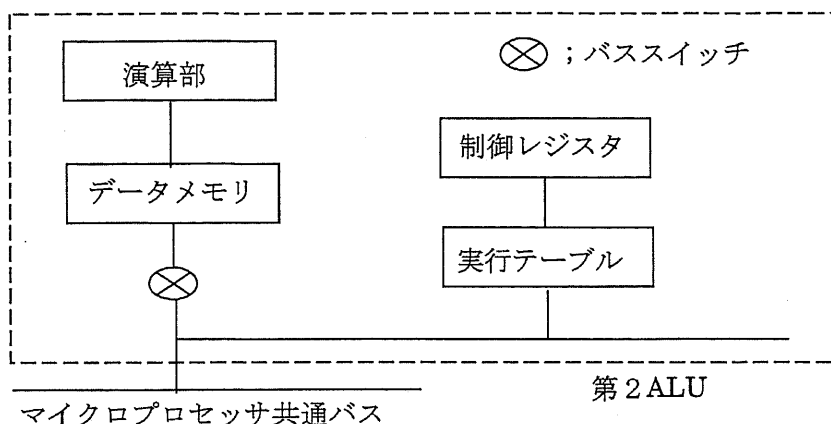


図 2. 20 第 2 ALU の構造  
 Fig. 2.20 Structure of 2nd ALU

第 2 ALU 付加によるマイクロプロセッサの性能向上の要因は次のものである。

- (1) 本体マイクロプロセッサと第 2 ALU の並列処理によるもの。
- (2) 第 2 ALU は、本体マイクロプロセッサに比べ  $1 / 10$  以下の小さな回路ブロックになる。そのため内蔵バスの付加容量に逆比例したクロック周波数で動作し、それで第 2 ALU 内部では約 10 倍の処理速度を得ることができると可能性がある。
- (3) 現在の半導体の微細化技術で、このような比較的大規模な第 2 ALU を複数個内蔵が可能でより大きい処理能力を得ることが可能である。

この第 2 ALU 付加によるマイクロプロセッサの性能向上の要因を開発するシステムに効果的に活用することが重要である。本件は前述したとおり画像処理を現在のマイクロプロセッサで実現するために比較的小さなハードウェア、すなわち独立したメモリと制御手順を持つ ALU (第 2 ALU) を付加した新たなアーキテクチャの提案を行っているものである。

この第 2 ALU を持つマイクロプロセッサは第 4 章のマイクロプロセッサによる信号処理に適応するものである。この信号処理を実現するために考案したもので、第 2 ALU 及びマイクロプロセッサの性能向上の考察と検討は第 4 章で論ずる。

## 2.4 ソフトウェア開発ツール

1970 年前半にマイクロプロセッサが開発されたとき、ソフトウェア開発ツールは当時のミニコンピュータのものを参考に開発された。しかし、ミニコンピュータとマイクロプロセッサとはコンピュータとしての機能と応用のされ方は大きく異なっていた。ミニコンピュータのソフトウェア開発はそのミニコンピュータで働かせる場合が多く、そのものでデバッグができる。しかし、マイクロプロセッサの場合は主に機器組込みでの応用がほとんどで、ソフトウェアのデバッグ機能はほとんど無に等しい。それで、機器に組み込んだ状態でデバッグのできるインサーキットエミュレータ (In Circuit Emulator) の開発が必要になる [108]。

ソフトウェア開発ツール体系を図 2.21 に示す。1970 年代と比べると高級言語 C や C++ がやっと実質的に使われることや、基本ソフトウェア、ミドルウェアの提供が重要になっている [62]。技術的にその向上のめざましいのが、JTAG (Joint Test Action Group) を含めたインサーキットエミュレータ技術とパーソナルコンピュータを中心にデバッグ機能、言語リアルタイム OS も含めた統合化開発システムの構築である。またソフトウェアの開発手法として大型コンピュータで実績のある階層化手法が提示された [63]。

### 2.4.1 ソフトウェア評価用マイクロプロセッサ

マイクロプロセッサのソフトウェア開発ツールにおいて、ソフトウェア開発の評価とデバッグを行うインサーキットエミュレータやデバッグは必須のものである [88]。またこれらは並列処理におけるプログラムや通信プログラムなどのようにより複雑になるソフトウェアの試験や、ハードウェアとソフトウェアの協調開発のため、またソフトウェアの品質確保のために重要性である。 [51、112、113]。インサーキットエミュレータやデバッグには、このマイクロプロセッサと同じハードウェアにさらに、インサーキットエミュレータやデバッグを実現する機能を持たせたインサーキットエミュレータ用マイクロプロセッサが必要である。マイクロプロセッサの新開発には本来のマイクロプロセッサとインサーキットエミュレータ用マイクロプロセッサの 2 つの開発が必要になる。これら 2 つのマイクロプロセッサを 1 品種で対応できる仕組みを開発した。



## (1) エバリュエーション機能内蔵マイクロプロセッサ

エバリュエーション機能内蔵マイクロプロセッサはエバリエーションチップと呼ばれている。マイクロプロセッサが42ピンの場合エバリュエーションチップは、アドレスバスとデータバスの端子が必要なため、64ピンの端子が必要である。マイクロプロセッサとエバリュエーションチップを共用するために、アドレスバスとデータバスを入出力ポートに時分割に重ねた。またエバリュエーションチップとして働かせるときは入出力とアドレスバスとデータバスを時分割で入出力させ外に回路を設ける。エバリュエーションチップでアドレスバスとデータバスが必要なのは、マイクロプロセッサの内臓マスク ROM や FLASH メモリを外付けの RAM などのメモリに置き換え、ここでプログラムのコードを変更することでソフトウェアの修正と評価を行うためである。

この場合これら2つの機能の選択は、エバチップとして使う場合は最小の外付回路を必要としてもマイクロプロセッサとして使う場合外付回路を全く無くすようにした。エバリュエーションチップとして使う場合、ポート端子にポート出力とポート入力に加えアドレス出力とプログラムコードの入力の、以上2つの出力と2つの入力が時分割で割り付けられる。図2.22にポート端子マイクロプロセッサ内部と外部を示す。アドレス線はROM 2kバイトのため11本、データ線は1ワード9ビットのため9本である。図2.23にポートにおける4信号の時分割タイミングを示す。また図2.24に4信号のチップ回路とチップ外回路を示す。

## (2) 評価

この方式は現在も使っており有効であったと考える。1979年より現在まで約300品種に対応しており、1品種のエバリュエーションチップに0.5人年かかるとすると、150人年のマンパワーの削減が出来たと推測できる。

言語処理	アセンブラ、マクロアセンブラ Cコンパイラ、C++ J a v a
OS	リアルタイムOS
ミドルウェア	J P E G、手書き文字認識 図形描画、DOSファイルシステム、e t c .
デバッガ	エミュレータデバッガ 統合化開発環境 開発評価ボード
基本ソフトウェア	サブルーチン集、バイナリローダー
リファレンスキット	情報端末、データ放送受信 インタネット、LAN TV、オーディオ、e t c .
スタータキット	各機種対応

図 2. 2 1 ソフトウェア開発ツール体系

Fig.2.21 Software development tool system

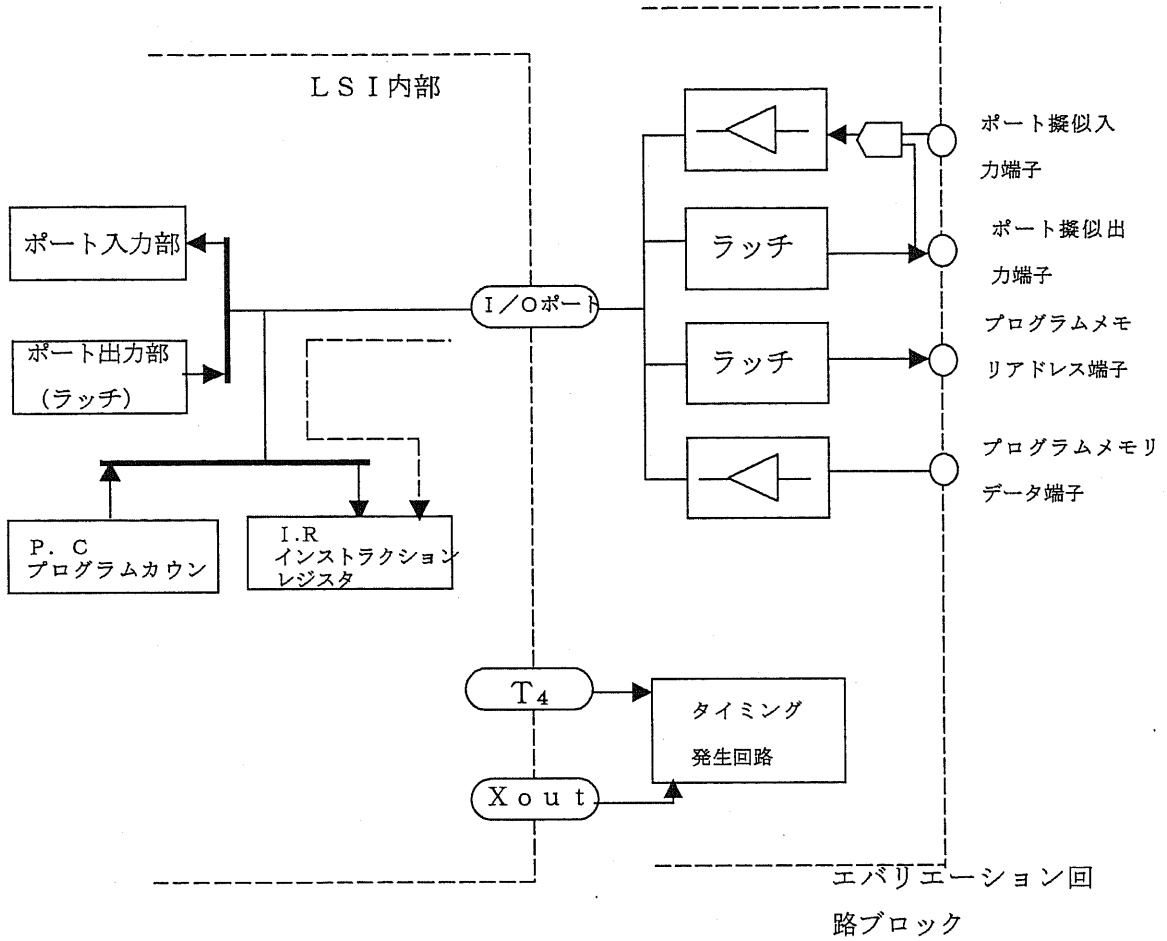


図2. 22 エバリエーション機能実現のためのポートの時分割使用

Fig.2.22 Port time-division use for evolutionary function

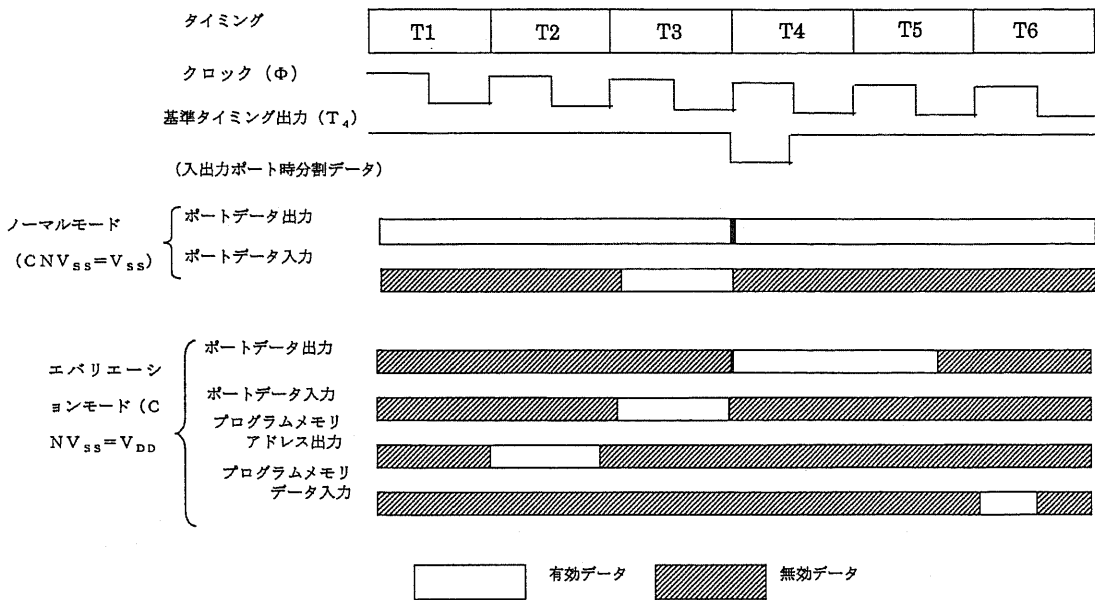


図 2. 23 ポートにおける時分割タイミング  
Fig.2.23 Time-division timing on port

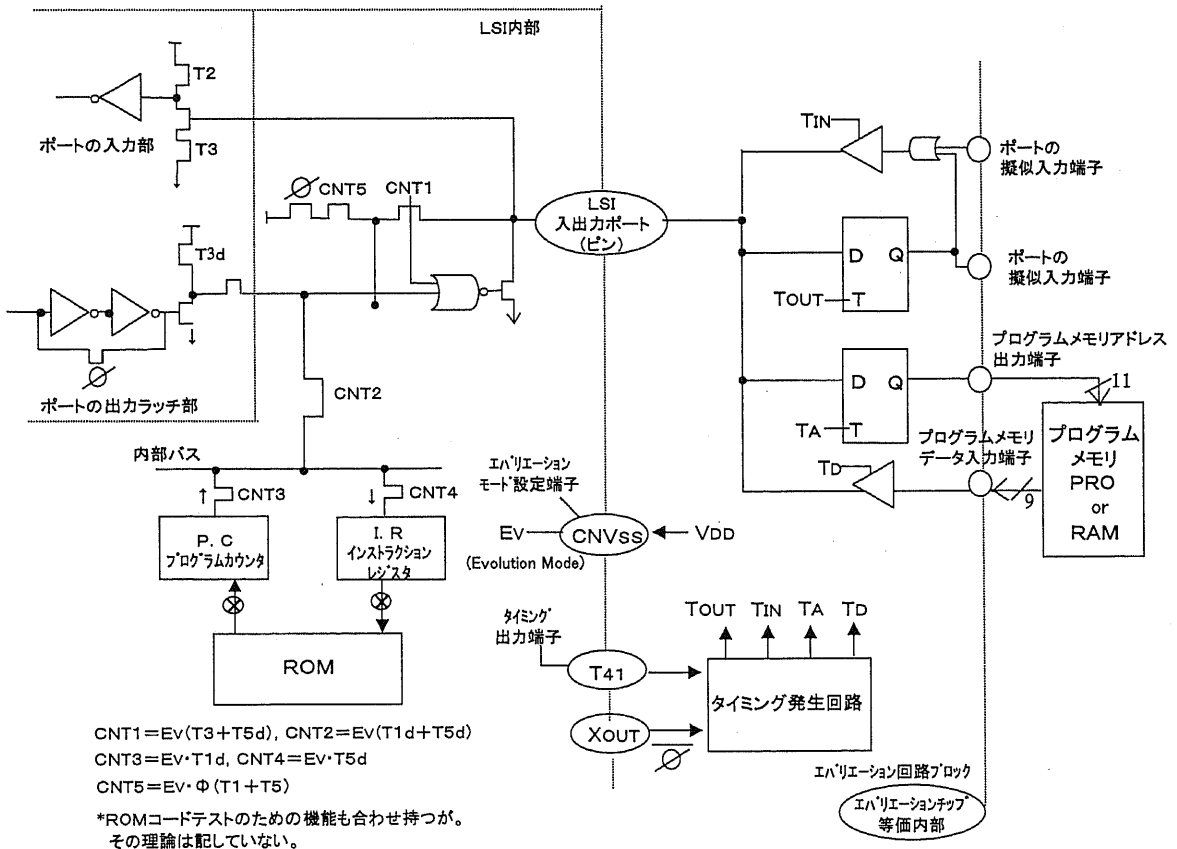


図 2. 24 時分割入出のプロセッサ内部と外付け回路モデル  
Fig.2.24 Internal processor of time shared I/O and external circuit model

## 2. 4. 2 モニタ

モニタを4品種開発してソフトウェア開発とマイクロプロセッサ応用システムに適應した[10、64、65、66、67]。マイクロプロセッサ用モニタには、ソフトウェアを評価する開発用モニタとソフトウェア実行を助ける実行用モニタに大別できる。また一方モニタはハードウェアとアプリケーションソフトウェアの間に介在しハードウェアの利用効率を高める役割もある。図2. 25に開発したモニタの一覧を示す。図2. 26に8ビットマイクロプロセッサ用モニタがマイクロプロセッサ応用製品開発のどの工程で活用されるかを示したものである。これらのモニタは、開発用モニタと実行用モニタ及びそれら両方のかねるものに分類できる。特に実行用モニタは、そのときどきのシステム上の課題解決に対応したものであるが他の問題解決にも有効になると考える[104]。

### (1) 開発用モニタ

開発用モニタは前述した図2. 26のデバックマシンに搭載され、ソフトウェアの評価とデバックの推進を助ける。同じく前述した図2. 25の開発用モニタBOM-SとBOM-Bを含みBOM-PTSについてその機能一覧を図2. 35に示す。デバックマシンに付けられる入出力機器、システムタイプライタ（キーボード、プリンタ、紙テープリーダ、紙テープパンチ）とROMライタの制御も行う。メモリのプログラムコードやデータの変更機能や指定番地をプログラムが通過すると実行を中止する機能（アドレスホルト）及びPAUSE、EXITマイクロ命令が開発用モニタの主要な機能になる。これら開発用モニタのコマンド例としてBOM-PTSのものを図2. 27に示す。

プログラムの実行の1命令ごとにマイクロプロセッサを止めて、そのときの各レジスタの値をプリントアウトする機能をトレース機能というが、トレースとアドレスホルトの実行を図2. 28と図2. 29に示す。

### (2) 実行用モニタ

実行用モニタとして図2. 25に示されているBOM-BとTSS-Mを開発し実働させた。実行用モニタは、開発にもBOM-Bの場合使われるが最終的な応用機器にそのまま組み込まれる。

BOM-B はマスク ROM として製品化したもので、2 k バイトの容量のものである。2 k バイトと比較的 ROM が小さい (BOM-PTS が 16 k バイト、BOM-S が 4 k バイト) ので機能が少ない。そこで開発応用ごとに必要な特殊な機能を追加したい。そういう場合 ROM であると、外部に RAM とか ROM 容量があいていても、それをこのモニタの機能一元的に使えない。そこで、この BOM-S は機能追加できる仕組みを入れ込んだ。図 2. 29 に示すように、ROM 化されたモニタ外の決められた RAM 番地に追加コマンドの有/無のコードを書込み、もし有のコードが書かれていたら、外部の決められた番地の RAM アドレスに、モニタのコマンド指定のルーチンが一時的にジャンプし、追加コマンドの実行をすることになる。このように追加機能をマスク ROM のモニタでも実現する仕組みを開発した。さらに、BOM-B のようにマスク ROM にされたプログラムはその時点でどのアドレスに置かれるものが絶対番地か決まってしまう。しかし図 2. 30 に示すように、アドレスを指定して、ジャンプするジャンプ命令や、サブルーチンコール命令に、ベースレジスタを設け、常にベースレジスタと加算したアドレスを選択するハードウェアと命令を追加すれば、マスク化されたプログラムも任意のアドレスに設置可能になる。このような特性を持つマスク ROM をリロケータブル ROM と名付けた。

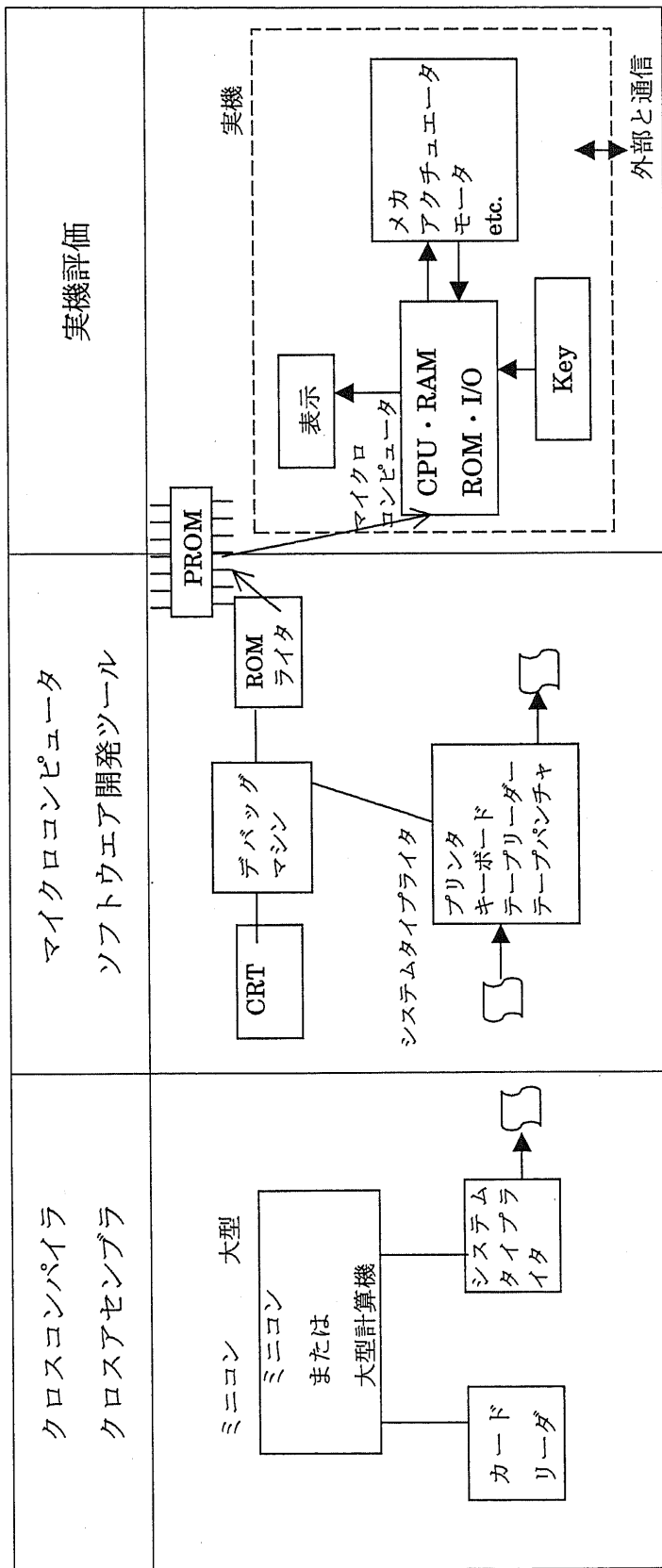
モニタ名	機能	適応
BOM-S	プログラム開発用の汎用的なモニタで、3コのマクロ命令と14コのモニタコマンドを持つ。これは、プログラム処理により実現したアドレスホルト機能を有する。(プログラム容量：4kバイト)	開発用
BOM-B	プログラム開発用の汎用的なモニタで、3コのマクロ命令と9コのモニタコマンドと三つの汎用サブツーチンを持つ。マスクROMかされたモニタであるが、モニタコマンドの拡張がユーザサイドで可能である。2kバイトのマスクROM M5831-001Sとして製品化されている。(プログラム容量：2kバイト)	開発用 実行用
BOM-PTS	デバッグマシン (MELCS 8-1) に適応するモニタで、3コのマクロ命令と、22コのモニタコマンドを持ち、特にデバッグ機能が充実し、PROM書込み機能、類似I/O機能を有する。(プログラム容量：16kバイト)	開発用
TTS-M	多重タスク処理を特別なハードウェアを用いないプログラム処理により実現する。汎用の実行用モニタである。(プログラム容量：100バイト)	実行用

開発用モニタ：プログラムデバッグを主な機能

実行用モニタ：個別のシステムで要求される実行形態を実現

図 2. 25 モニター一覧

Fig.2.25 Monitor List



(モニタ) プログラム実行制御、実行トレース (モニタ)

入出力制御、PROM ライト・リード

プログラムのロードパンチ、データ表示、

マクロ命令、マクロ命令

図 2.2.6 8 ビットマイクロプロセッサ応用製品開発におけるモニタの役割

Fig. 2.26 Role of monitor with 8 bit microprocessor application development



プログラム実行制御	G コマンド R コマンド PAUSE マクロ命令 EXIT マクロ命令	プログラム実行開始 プログラム実行再開始 実行一時中断 実行終了
入出力制御	U コマンド EXIO マクロ命令	擬似入出力処理 入出力処理
プログラムのロード及びパンチ	LM コマンド DM コマンド	MELPS8 バイナリローダ MELPS8 バイナリパンチ
データ表示	PR コマンド PM コマンド PA コマンド	レジスタデータ表示 メモリデータ表示 メモリデータ逆アセンブル表示
データ変更	MR コマンド MM コマンド MC コマンド MS コマンド MT コマンド	レジスタデータ変更 メモリデータ変更 メモリデータ補数変換 定数データメモリ設定 メモリ間データ転送
プログラムデバック	H コマンド S コマンド T コマンド PT コマンド C コマンド I コマンド	アドレスホルト スナップショット トレース デバッカテーブル表示 デバッカテーブル消去 機械語命令割込み可指定
PROM 作成	FP コマンド FT コマンド FC コマンド	PROM 書きこみ PROM データメモリ転送 PROM データメモリデータ比較

図 2. 27 制御用モニタのコマンド例

Fig.2.27 Example of monitor function

番地 (16 進表示)		プログラム	
100	START	CALL	SUB1
103		CALL	SUB2
106		NOP	
107	DVD	NOP	
110	SUB1	MVI	A, 01
112		MVI	B, 02
114		MVI	C, 03
116		MVI	D, 04
118		MVI	E, 05
11A		MVI	H, 06
11C		MVI	L, 07
11E		MVI	
120	SUB2	LXI	B, TABLE
123		MVI	A, 10
125		MVI	C, 08
127	SUB2-1	MOV	H, A
128		INR	A
129		INX	H
12A		DCR	C
12B		JNZ	SUB2-1
12E		RET	
130	TABLE	DEP	0, 0, 0, 0
134		DEP	0, 0, 0, 0

```
//TS100, 12E,,9,B-----10
//HS104, 0-----11
//SS104,MELPS8,130,137,-----12
//11-----13
//5-----5
```

```
0100
ADD  CD  MNEM          PC  A  B  C  D  E  H  L  SP  CYZSP
0100  CD  CALL    0110  0110  01  00  00  02  00  01  00  EEB9  01101
011E  C9  RET      0113  0113  01  02  03  04  05  06  07  EEBB  01101
0103  C0  CALL    0120  0120  01  02  03  04  05  06  07  EEB9  01101
0129  C2  JNZ     0127  0127  11  02  07  04  05  01  31  EEB9  01000
0129  C2  JNZ     0127  0127  12  02  04  04  05  01  32  EEB9  01001
0123  C2  JNZ     0127  0127  13  02  05  04  05  01  33  EEB9  01001
0123  C2  JNZ     0127  0127  14  02  04  04  05  01  34  EEB9  01000
0123  C2  JNZ     0127  0127  15  02  03  04  05  01  35  EEB9  01001
0129  C2  JNZ     0127  0127  16  02  02  04  05  01  36  EEB9  01000
0129  C2  JNZ     0127  0127  17  02  01  04  05  01  37  EEB9  01000
0128  C2  JNZ     0127  012E  18  02  00  04  05  01  38  EEB9  01101
012E  09  RET      0104  18  02  00  04  05  01  38  EEB9  01101
*MELPSB*
0104  00  NOP
① 10 11 12 13 14 15 16 17
***HALT***
0106  00  NOP          0107  18  02  00  04  05  01  38  EEBB  01100
```

図 2.28 モニタ機能のトレース、アドレスボルトの実行 (その1)

Fig. 2.28 Trace with monitor function and address bolt operation No. 1

```

//TS100, 11E,R,-----1
//SS12A, MELPS8, 130, 137, R -----2
//HS106, 0 -----3
//11 -----4
//8 -----5
0100
AD0      CD   MNEM          PC   A  B  C  D  E  H  L  SP   CYZSP
0100     CD   CALL    0110    0110 01 00 00 02 00 01 00  EEB9 01101
0110     3E   MV1     A,01    0112 01 00 00 02 00 01 00  EEB9 01101
0112     06   MV1     B,01    0114 01 02 00 02 00 01 00  EEB9 01101
0114     0E   MV1     C,03    0116 01 02 03 02 00 01 00  EEB9 01101
0116     16   MV1     D,04    0118 01 02 03 04 00 01 00  EEB9 01101
0118     1E   MV1     E,05    011A 01 02 03 04 05 01 00  EEB9 01101
011A     26   MV1     H,04    011C 01 02 03 04 05 06 00  EEB9 01101
011C     2E   MV1     L,07    011E 01 02 03 04 05 06 07  EEB9 01101
011E     C9   RET
0103     C0   CALL    0120    0120 01 02 03 04 05 06 07  EEB9 01101
MELPS8
012A 00      DCR      C          012B 11 02 07 04 05 01 31  EEB9 01000
0130      10 00 00 00 00 00 00 00 00
MELPS8
012A 00      DCR      C          012B 12 02 06 04 05 01 32  EEB9 01001
0130      10 11 00 00 00 00 00 00 00
MELPS8
012A 00      DCR      C          012B 13 02 05 04 05 01 33  EEB9 01001
0130      10 11 12 00 00 00 00 00 00
MELPS8
012A 00      DCR      C          012B 14 02 04 04 05 01 34  EEB9 01000
0130      10 11 12 13 00 00 00 00 00
MELPS8
012A 00      DCR      C          012B 14 02 03 04 05 01 35  EEB9 01001
0130      10 11 12 13 14 00 00 00 00
MELPS8
012A 00      DCR      C          012B 14 02 02 04 05 01 36  EEB9 01000
0130      10 11 12 13 14 15 00 00 00
MELPS8
012A 00      DCR      C          012B 17 02 01 04 05 01 37  EEB9 01000
0130      10 11 12 13 14 15 14 00 00
MELPS8
012A 00      DCR      C          012B 18 02 00 04 05 01 38  EEB9 01101
0130      10 11 12 13 14 15 16 17
0106 00      NOP          0107 18 02 00 04 05 01 38  EEBB 01101
***HALT***
0104 00      NOP          0107 18 02 00 04 05 01 38  EEBB 01101

```

図 2 . 2 9 モニタ機能のトレース、アドレスボルトの実行 (その 2)

Fig. 2.29 Trace with monitor function and address bolt operation No.2

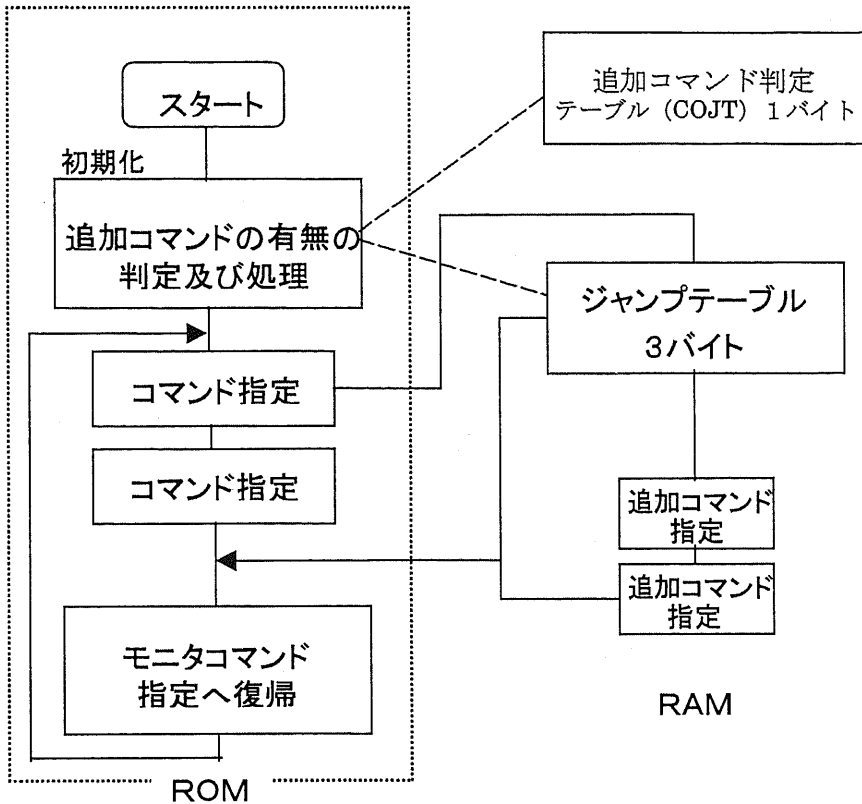


図 2. 30 ROMに格納されたモニタの機能追加法

Fig.2.30 Method of additional stored-ROM monitoring function

分岐命令	JMPB、JCB、JZB、JPEB
サブルーチンコール命令	CALLB

ベースレジスタ修飾絶対分岐命令

JMPB m 機能 ; (PC) ← m + (B) (C)

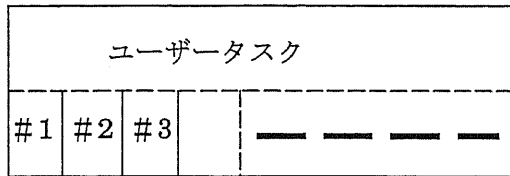
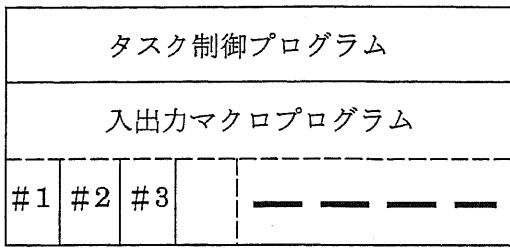
絶対分岐命令 (参考)

JMP m 機能 ; (PC) ← m

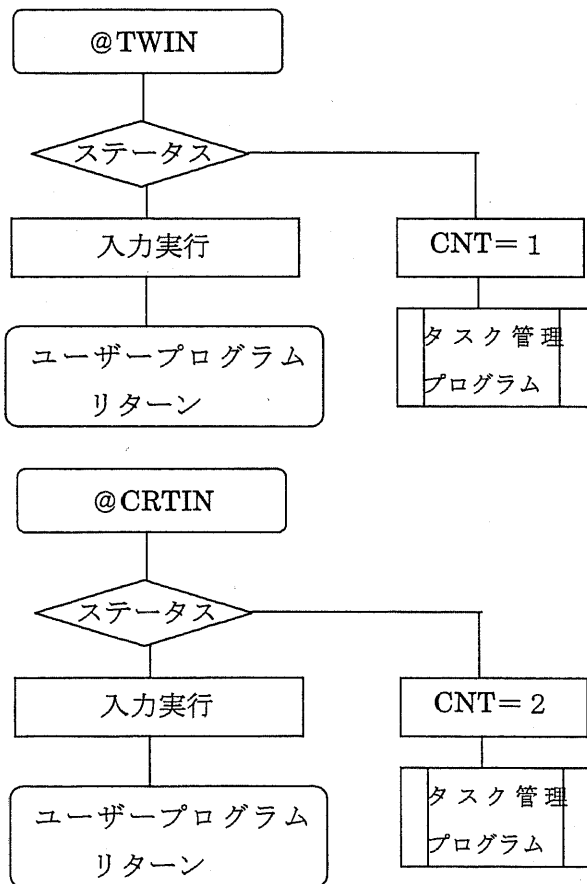
※ 但し、ベースレジスタを (B) (C) レジスタとする。

図 2. 31 ROMのプログラムを任意のアドレス番地に設定を可能にするための命令追加 (リロケータブルROM)

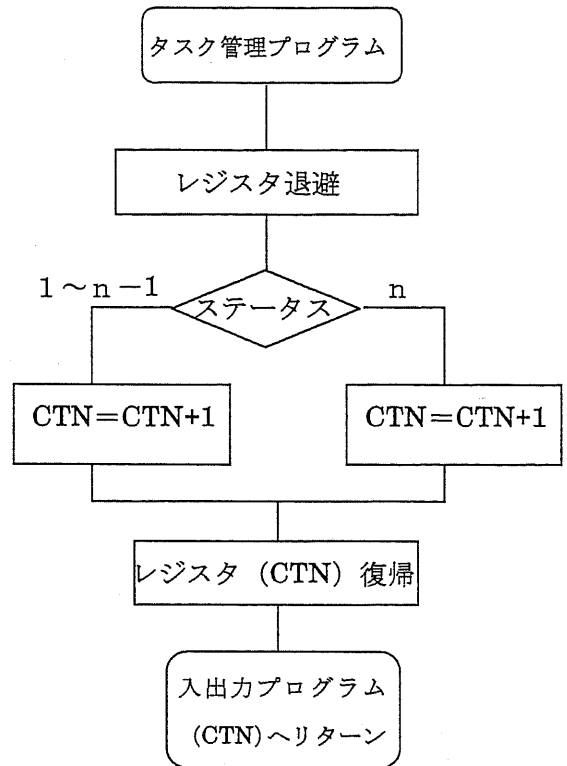
Fig.2.31 Additional order (relocate ROM)



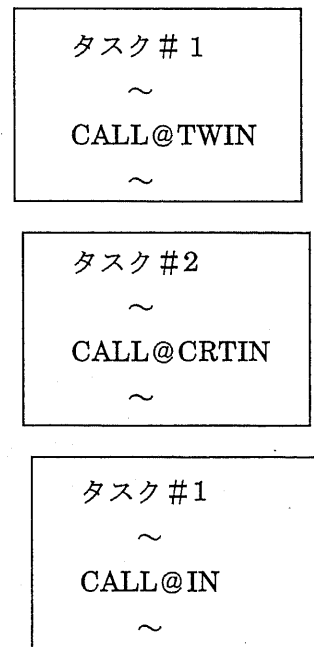
(a) マルチタスクプログラムの構成



(c) 入出力マクロプログラム



(b) タスク管理プログラム



(d) タスク

図 2. 32 マルチタスク制御

Fig.2.32 Multitasking control

次いで複数のタスク（プログラム）をタイムシェアリングで動かす簡易モニタ（TTS-M）を開発した。特別なハードウェアを用いることなくまた、割込みも使わず、各タスクの入出力の待ち時間を有効に活用する。

図 2。32 に示すように各タスクにおける入力処理ルーチンで入力状態を確認し、もしレディなら入力を実行する。そして、ビジィなら次タスクの入力状態確認を行い同様の処理を繰り返す。すなわち早くレディになった方の入力処理が実行され、そのタスクのプログラムが引き続き実行されることになる。本来ならビジィのときは閉ループでノーオペレーションをレディになるまで実行しつづけることになるが、この方法では異なるタスクを実行するマルチタスクである。

ここで述べたマルチタスクの方式は、ハードウェアの割込み機能が不要であるという利点のほか、ステータスビジィによってマルチタスク処理を行っているため、マルチタスクモニタが介在することによるオーバーヘッド、すなわち時間的遅延は 10 命令までであるので問題にならないと考える。また、マルチタスク制御モニタのメモリは 500 バイトと小さい容量で実現している。

このタスク制御プログラムでは、タスク制御獲得するチャンスは均等であるが、優先順位を持たせた制御も容易に付加できる。

## 2. 5 むすび

マイクロプロセッサ構造（アーキテクチャ）の改善の目的として、ここでの課題は2つである。1つはマイクロプロセッサの性能向上と2つめはマイクロプロセッサ設計開発の効率向上である。

マイクロプロセッサの性能向上では4ビットマイクロプロセッサの開発において、命令コードを1ワードのビット数を1バイト（8ビット）とせず1ビット多い9ビットにしたこと、複数機能命令の採用とサブルーチン機能強化により、命令コード総ビット数に対してその命令セットによるプログラムでの命令コードの大幅削減を実現した。この向上の定量的評価は実際に開発したソフトウェアのコードの実績と、これらの改善をしない場合のソフトウェアを比較することにより評価した。その結果命令機能は2.19倍になり、命令コード数が45.7%まで圧縮されたことになる。実行速度は、メモリデータ交換プログラムでは、4ビットを1語として16語の場合で、従来マイクロプロセッサでは1320 $\mu$ secのところ500 $\mu$ secに短縮で2.64倍の高速化を実現した。

32ビットマイクロプロセッサに第2ALU付加による並列処理と第2ALU内部処理速度10倍以上の可能性の提示を行った。さらに具体的には第4章にて映像信号のMPEG-2のデコード処理での検討を行う。

マイクロプロセッサの設計人工を半減させた。開発効率向上は4ビットマイクロプロセッサの場合で、基本開発1つと30の展開開発で比較すると、820人月が346人月に短縮する。すなわち42%の開発人工に削減でき58%の開発効率向上したことになる。

次いで8ビットマイクロプロセッサ用ソフトウェア開発ツールは、ミニコンピュータのものを基にした。しかし、ミニコンピュータとマイクロプロセッサとはいくつかの点で異なり、そこが以下の研究の重要課題になった。

インサーキットエミュレータに使われるエバリュエーションチップは1つのマイクロプロセッサに対し1品種ずつ開発する必要があった。これに対しマイクロプロセッサにエバリュエーションの機能を数10ゲートの追加で実現させ、一切のエバリュエーションチップの開発を不要にすることができた。1979年より現在まで300品種に対応しており、1品種に0.5人年の人工が必要と推測すると

150 人年の開発人員の削減ができたことになる。また、エミュレーションチップとしての開発効率は、エミュレーション機能の取込人工を 300 品種で 2 人年と推測すると 98.7% の開発効率向上をさせたことになる。

次にモニタを 4 品種開発した。開発用モニタはデバッグマシン上で働きソフトウェア開発のデバックを支援する。また実行用モニタはミニコンピュータに比べマイクロプロセッサの機能が低いために、その補いを実行用モニタに行わせようとするプログラム実行支援ソフトウェアである。これらの 1 つとして、マスク ROM にモニタのソフトウェア書込んだものがあるが、そこでモニタ機能を拡張ができないという問題を解決した。また多重タスク処理を入出力機器のビジィ信号をうまく使うことにより、500 バイトのプログラムで実現したモニタを開発した。

映像機器により適合した 4 ビット 1 チップ・マイクロプロセッサとその開発支援ツールを実際に開発し、2 倍以上の高速化の実現と同じく 2 倍以上の機能をはたす命令セットとその為の構造（アーキテクチャ）の開拓究明で実現することができた。また 32 ビット・マイクロプロセッサも高速化の検討として、今後の半導体生産技術の向上による高速化の足かせにならず、またますます大規模になるシステムを見通しよく開発ができ、10 倍以上の速度性能を得る可能性のある第 2 ALU の構造（アーキテクチャ）提示を行った。



### 3. マイクロプロセッサ援用信号処理

#### 3. 1 まえがき

一般産業機器、民生機器の開発時の技術面での重要課題は、新機能の実現と性能・品質の向上、部品点数の削減と開発期間短縮である。オーディオ及び映像機器においても同じ状況で、アナログとデジタルの信号技術の工夫によりゲート数を削減しSOCを目指す努力をしている [86、90]。

オーディオ信号のアナログ処理においては、アナログ IC とデジタル IC、マイクロプロセッサとのインタフェースと、外付部品点数削減が十分でなく SOC 化ができていない [112]。そこでシリアルバスとレジスタ制御によりマイクロプロセッサからアナログ信号処理回路を制御する仕組みを検討した。アナログ信号処理回路に、簡易なアナログ・デジタル変換器を付け、信号をセンスし、またその状況によりその回路の変化を、マイクロプロセッサでソフトウェア処理し、部品点数の削減と SOC 化を計る手法を試みた [103]。そして部品点数削減の十分な成果を得るとともに、アタックタイムの大幅改善も合わせて実現する事が出来た [68、69、70]。

次いで、VTR のデジタル特殊効果において、ピクチャインピクチャ、マルチ画面等の新機能及び高速サーチやスロー等の特殊再生のノイズレス効果において、この機能実現用の専用メモリと信号処理制御 LSI を新たに設計し、又それらをマイクロプロセッサのソフトウェアで処理するシステム開発することにより、新機能の実現と性能・品質の向上と部品点数の削減と開発期間短縮を実現する事が出来た [76、81、84]。

### 3. 2 オーディオ

映像機器におけるシステム LSI の発展が著しい。これは性能向上とともに部品点数削減を行うためである。これを進めるため DVD プレーヤーではマイクロプロセッサ、信号処理理論回路に加え DRAM や AD コンバータが混載され、また携帯電話機ではさらにアナログ回路も混載されている[71、72]。また、バイポーラ集積回路と遜色ない性能のトランジスタを得る BiCMOS 技術が研究され、個別のバイポーラ集積回路で構成されていたアナログ回路もシステム LSI 化が容易になっている[114]。

しかし、テープレコーダや VTR など、交流バイアス音声記録ではシステム LSI 化が進展しているとはいえない。これは単に音声信号処理のバイポーラ回路とマイクロプロセッサをシステム LSI 化するだけでは、顕著な部品点数削減がないからと考えられる。すなわち、交流バイアス音声記録のシステム LSI 化に有効な、部品削減手法が必要である。近来、マイクロプロセッサは VTR の制御に実用されてきている[1、3]。音声信号処理でもゲインやスイッチなど集積化できない外付け部品の削減が進められている[69]。しかし VTR の音声信号処理では、再生イコライザ、記録イコライザ、ALC (Automatic Level Controller) における多くの時定数部品を残している [73、118]。

そこで本論文では、これらの時定数切換えをマイクロプロセッサ援用によるゲイン制御で実現する手法を提案し、外付け部品削減の可能性を検討している。再生イコライザでは GCA (Gain Controlled Amplifier) を使った 2 重フィードバック回路でローパスフィルタを構成し、零点を時定数回路で定め、極の時定数をフィードバックゲインで切換えている。また、記録イコライザでは、フィードフォワード回路でハイパスフィルタを構成し、極を時定数回路で定め、フィードフォワードゲインで零点の時定数を切換えている。ALC では、マイクロプロセッサでプログラム制御するアルゴリズムによって、実行周期あたりのゲイン制御をアタックとリカバリーの両方で切換え異なる時定数を得ている。そして、これらの回路を VTR に実装して交流バイアス音声記録を行い、少ない時定数回路で必要な性能を確保できることから、システム LSI への有効性を確認した。

その結果、VTR 用バイポーラ集積回路の例では 10 個の時定数回路を外付け部

品としたが、提案の手法により 2 個に削減できる目処を得た。帯域幅とリカバリータイムは前記の例を同等の性能で、さらにアタックタイムは  $1/20$  に短縮できた。本提案は、バイポーラの音声信号処理回路と CMOS のマイクロプロセッサを 1 チップ化する際に有効である。すなわちマイクロプロセッサ援用ゲイン制御により、少ない外付け部品で多くの時定数動作を実現でき、BiCMOS プロセスによるシステム LSI に適していることがわかった。

### 3. 2. 1 時定数部品切換え型回路

VTR における交流バイアス音声記録の基本システムを図 3. 1 に例示した。ヘッドと入力の回路網およびバイポーラ集積回路化されたラインアンプと PB (Play Back Equalizer)、REC (Recording Equalizer)、ALC の 3 ブロックで構成され、点線のラインを介してマイクロプロセッサにより制御される。ここに例示する集積回路は、各ブロックを時定数部品切り替え型の回路で構成し、複数の時定数を得ている [73]。以下、これらの時定数の動作を調べる。

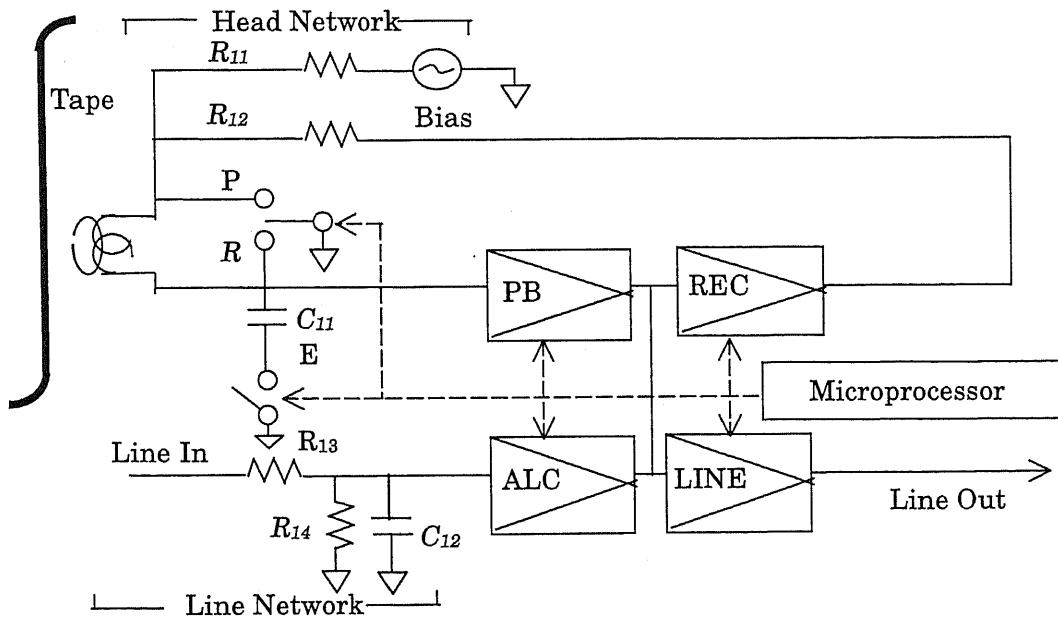


図 3. 1 交流バイアス音声記録の基本システム  
 Fig.3.1 Basic AC biased audio recording system

(1) 再生イコライザ

周波数特性は図 3. 2 の漸近線図で示す双 1 次ローパス特性が IEC (International Electrotechnical Commission) 規格に規定され、表 3. 1 のように 3 つの時定数が使われる[74]。ここに、SP (Standard Play) はテープ速度が 3. 335cm/s、EP (Extended Play) は 1. 111cm/s 用の場合であることを示す[94]。

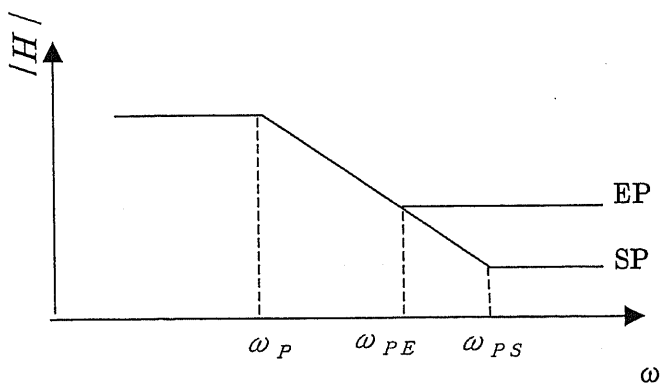


図 3. 2 再生イコライザの周波数特性  
 Fig.3.2 Characteristics of the playback equalizer

表 3. 1 再生イコライザの時定数

Table 3.1 Time constants of the playback equalizer

Mode		EP	SP
Time Constants	$1/\omega_P$	$1/\omega_{PE}$	$1/\omega_{PS}$
Values ( $\mu s$ )	3180	170	120

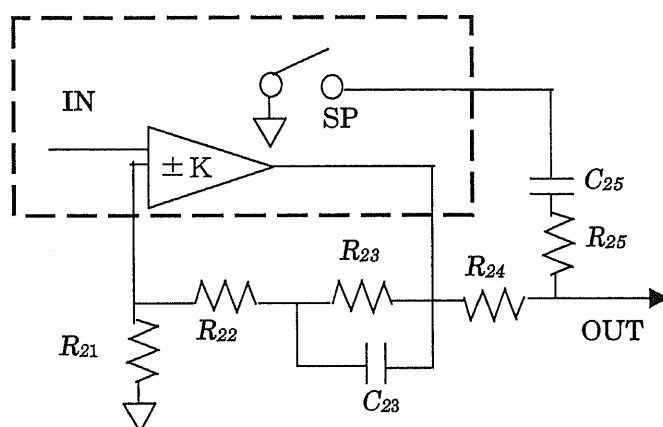


図 3. 3 再生イコライザの回路例

Fig.3.3 Example circuit of the playback equalizer

図 3. 3 回路例で、スイッチはトランジスタのコレクタエミッタ間が使われる。また、破線は集積化部分を示す。周波数特性は、帰還増幅回路で得られ、 $R_{24}R_{25}C_{25}$ の構成による縦続部分をスイッチで開閉して切り替えている。

式 3. 1 は伝達関数であり、右辺の第一項は低周波ゲインを表す。また、第 2 項は帰還増幅部の周波数特性を表し、第 3 項は縦続部分の周波数特性を表す。

スイッチが開いているときは  $C_{25}=0$  と考えると、第 3 項が 1 となり第 2 項によって周波数特性が決定される。そこで  $C_{23}R_{23}=1/\omega_P$ 、 $C_{23}R_{22}=1/\omega_{PE}$  とすると、EP の周波数特性を得る。ただし、 $K \gg R_{23}/R_{21}$ 、 $R_{23} \gg R_{22}$  である。

スイッチを閉じると、第3項が有効になる。時定数を、 $C_{25}(R_{24} + R_{25}) = 1/\omega_{PE}$ 、 $C_{25}R_{25} = 1/\omega_{PS}$ とすると、第2項の零点は等化され第3項の零点だけが有効になり、SPの周波数特性を得ている。

$$H = \frac{R_{22} + R_{23}}{R_{21}} \frac{1 + j\omega C_{23}R_{22}}{1 + j\omega C_{23}R_{23}} \frac{1 + j\omega C_{25}R_{25}}{1 + j\omega C_{25}(R_{24} + R_{25})} \quad \dots \text{式 3. 1}$$

以上のように再生イコライザでは1次の極と1次の零点を与える時定数が必要で、零点はEPとSPで切換えられる。この例は、4つの時定数を集積回路の外付け部品で構成し、部品の接続を開閉して零点を切換えて、3つの時定数を得ている。

## (2) 記録イコライザ

交流バイアス記録の周波数特性は数kHz以上で低下するので、図3.4のように $\omega_{RS}$ から帯域上限の $\omega_R$ まで高域強調して等化する。EPでは記録波長がSPの1/3になり、低い周波数 $\omega_{RE}$ から高域強調する。

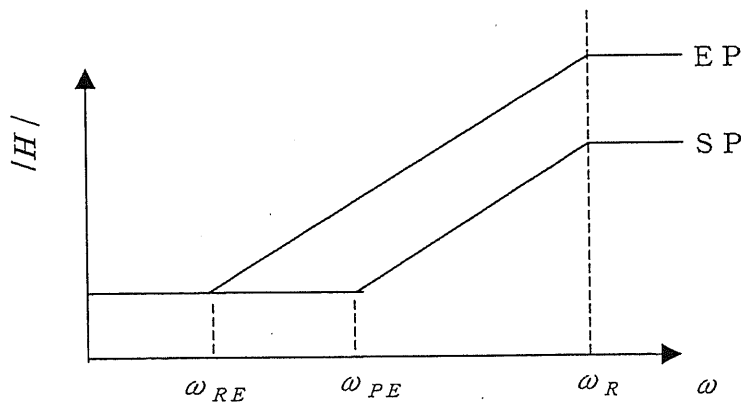


図3.4 記録イコライザの周波数特性

Fig.3.4 Characteristics of the recording equalizer

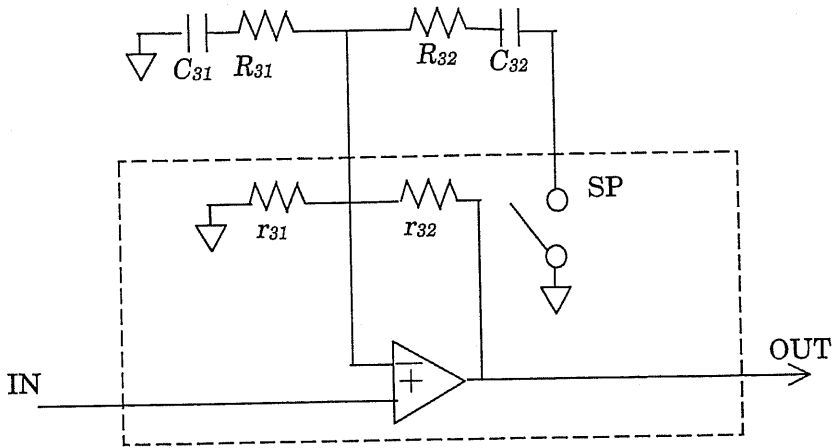


図 3. 5 記録イコライザの回路例  
 Fig. 3.5 Example circuit of the recording equalizer

図 3. 5 に回路例を示す。周波数特性は帰還増幅回路で得られ、帰還回路の  $C_{32}R_{32}$  をスイッチで開閉して切換える。

スイッチを閉じているときの伝達関数は式 3. 2 で、双 2 次となる 4 つの時定数を持っているが、 $C_{32}R_{32} = C_{31}R_{31}$  とすると  $C_{32}R_{32}$  の極が零点を等化して双 1 次特性になる。そこで  $C_{31}R_{31} = 1/\omega_R$ 、 $C_{31}R_{31} + C_{31}r_{31} + C_{32}r_{31} = 1/\omega_{RE}$  とすると図 3. 4 における EP の周波数特性を得ている。

またスイッチを開くと  $C_{32} = 0$  と考えられ、時定数を  $C_{31}(C_{31} + r_{31}) = 1/\omega_{RS}$  として SP の周波数特性を得ている。

$$\begin{aligned}
 H &= r_{32} \left[ \frac{1}{r_{31}} + \frac{1}{R_{31} + 1/j\omega C_{31}} + \frac{1}{R_{32} + 1/j\omega C_{32}} \right] \\
 &= \frac{r_{32}}{r_{31}} \frac{1 + j\omega(C_{31}R_{31} + C_{31}r_{31} + C_{32}r_{31})}{1 + j\omega C_{31}R_{31}} \quad \dots \text{式 3. 2}
 \end{aligned}$$

以上のように記録イコライザでは 1 次の極と 1 次の零点を与える時定数が必要で、極は EP と SP で切換えられる。この例は、4 つの時定数を外付け部品と 1 個の集積化抵抗で構成し、部品の接続を開閉して極を切換え、3 つの時定数を得ている。

### (3) ALC

信号が急増しても記録出力が歪まぬよう、またライン出力が閾値を越えぬよう、制御している[75]。ポップノイズを抑えるためアタックタイムは数msとされる。また信号が低下したとき、ゲイン復帰で発生する息継ぎノイズを抑えるため、リカバリータイムは1～2秒とされる。

回路例を図3.6に示す。GCAの出力のピークが整流され $C_{41}$ を充電すると、この電圧でゲインの制御が行われる。

充電ではダイオードが導通するので、アタックの時定数は、 $C_{41}r_{41}$ でありリカバリータイムを決定している。このように、ゲイン低下とゲイン上昇を与える2つの時定数を、外付け部品と1個の集積化抵抗で構成し、部品の接続をダイオードで切替えている。

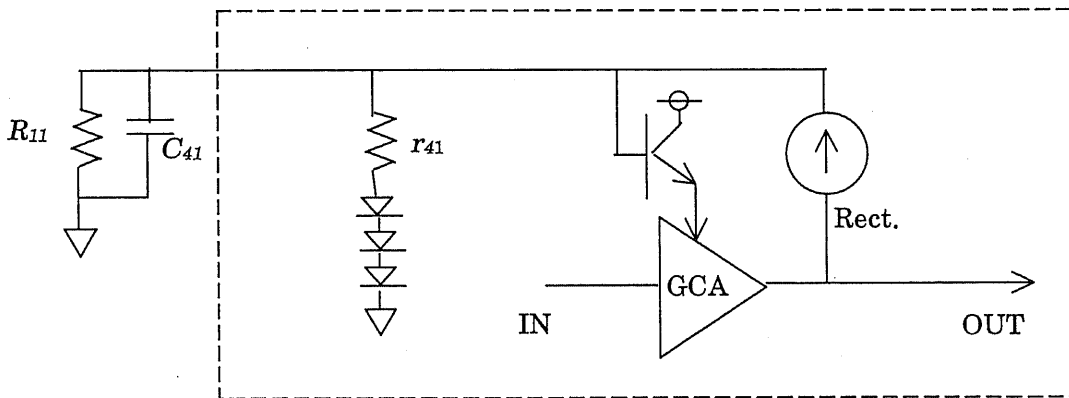


図3.6 ALCの回路例  
Fig.3.6 Example of ALC circuit

#### 3.2.2 ゲイン制御型回路

イコライザとALCのブロックで必要な時定数の切換えを、ゲイン制御で実現する方法を提案している。また、ゲイン制御はレジスタを介してマイクロプロセッサ援用によって行うことで、システムLSIへの有効性を向上する。



(1) 再生イコライザ

ゲインがAの差動増幅器の入力に伝達関数Fで負帰還すると、入出力の伝達関数は  $H = A / (1 + AF)$  となる。通常Aは大きくとり、 $H = 1/F$ とする。図3.3と図3.5の例でも同様で、極や零点の時定数を切替えるには、時定数を形成する別の素子を必要とした。

ところがAが小さいと極はAで可変となる。しかもFが双1次でなくても、Fの極がHの分子になって零点が形成される。すなわち1個の時定数回路から3つ以上の時定数を得る。以下、この手法で再生イコライザを検討する。

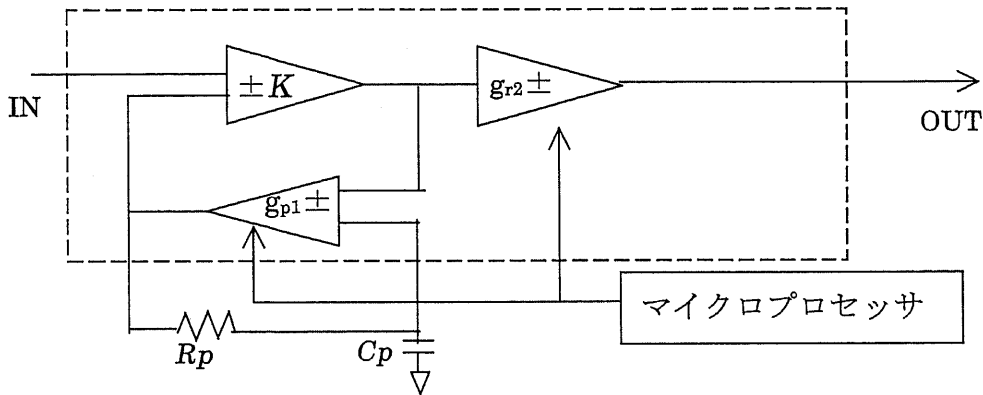


図3.7 再生イコライザ回路  
Fig.3.7 Playback equalizer circuit

図3.7に提案の回路を示す。GCAを使ったフィードバック回路でローパス特性のイコライザを構成している。その伝達関数は次式のように表される。ただし、 $K \gg gp1$ とする。

$$H = \frac{1+gp1}{gp1} \frac{1+j\omega/(1+gp1)}{1+j\omega/\omega_p} \quad \dots \text{式3.3}$$

極の時定数を  $C_p R_p = 1/\omega_p$ と定めると、零点の時定数も  $C_p$ と  $R_p$ で定まり、フ

フィードバックゲイン  $g_{P1}$  により SP と EP でフィルタの時定数が可変となる。

このようにゲイン制御型回路によれば、1 個の時定数回路で所定の特性が実現可能と考えられる。

(2) 記録イコライザ

伝達関数  $F$  を通った信号をゲインが  $A$  の増幅器でフィードフォワードすると、伝達関数は  $H=1+A F$  であり、零点は  $A$  で可変となる。しかも  $F$  が双 1 次でなくとも、 $F$  の極が  $H$  の分子になって零点が形成される。すなわち 1 個の時定数回路から 3 つ以上の時定数を得る。以下、この手法で記録イコライザを検討する。

本提案では、図 3. 8 のようにゲインが  $g_{R1}$  の GCA を使ってフィードフォワード回路でイコライザを構成している。その伝達関数を式 3. 4 に表す。極の時定数を  $C_R R_R=1/\omega_R$  と定めると、零点の時定数も  $C_R$  と  $R_R$  で定まる。そして、フィードバックゲイン  $g_{R1}$  により SP と EP でフィルタの時定数を可変にできる。

$$H = g_{R2}^2 \frac{1 + j \omega / (1 + g_{R1}) / \omega_R}{1 + j \omega / \omega_R} \quad \dots \text{式 3. 4}$$

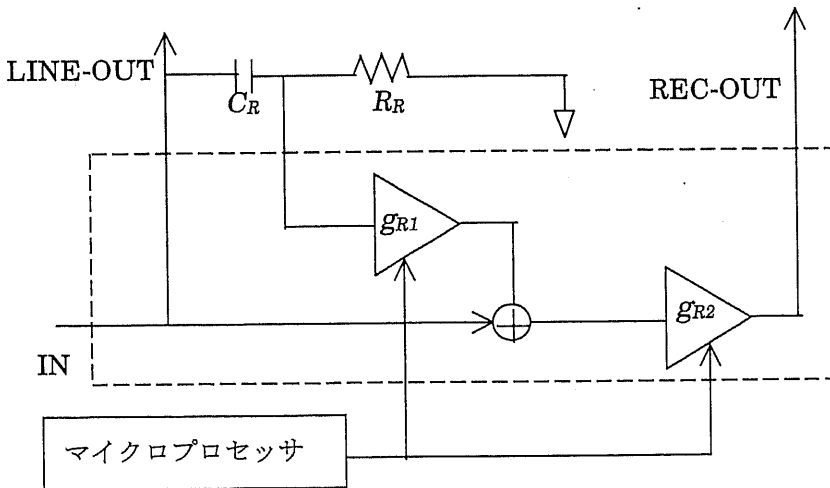


図 3. 8 記録イコライザ回路  
Fig.3.8 Recording equalizer circuit

このようにゲイン制御型回路によれば、1個の時定数回路で必要な特性が実現可能と考えられる。

### (3) ALC

ALCではマイクロプロセッサで実行時間あたりのゲイン上昇とゲイン低下を切換え、時定数回路部品を使わず2つの時定数を得る。図3.9にALCの構成を示す。ゲイン制御とALCの閾値 $V_D$ の設定をGCAで行う。 $V_{ref}$ は基準電圧で電源電圧 $V_{CC}$ の1/2が使われる。

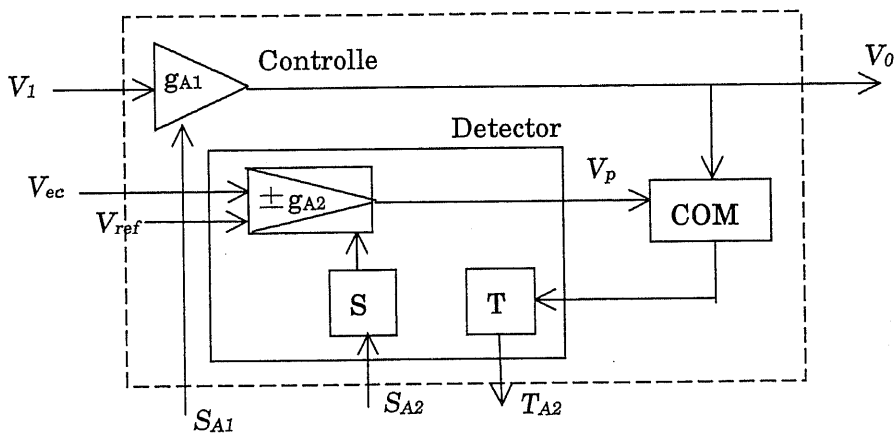


図3.9 マイクロプロセッサ制御によるALC回路  
Fig.3.9 Microprocessor controlled ALC circuit

$V_D$ は $(V_{CC} - V_{ref}) g_{A2}$ で得られ、比較器COMは実行周期 $t$ の間ALC出力 $V_o$ が $V_D$ になれば、Tレジスタに置数する。続いてマイクロプロセッサはレジスタSに $S_{A1}$ を置数し、ゲイン制御することで、 $V_o = V_i g_{A1}$ の出力を得る。

制御のアルゴリズムを式 3. 5 に、入出力波形を図 3. 10 に示す。ゲインは  $S_A$  により  $S_{AH}$ 、 $S_{AL}$ 、 $S_{AU}$  の 3 段で修正される。

$$\begin{aligned}
 & \text{IF } V_o \geq V_{DH} \\
 & \quad \text{THEN } S_{A1} = S_{A1} + S_{AH} \\
 & \quad \text{ELSE IF } V_o \geq V_{DL} \\
 & \quad \quad \text{THEN } S_{A1} = S_{A1} + S_{AL} \\
 & \quad \quad \text{ELSE } S_{A1} = S_{A1} - S_{AU} \\
 & \quad \quad \quad \dots \dots \dots \text{式 3. 5}
 \end{aligned}$$

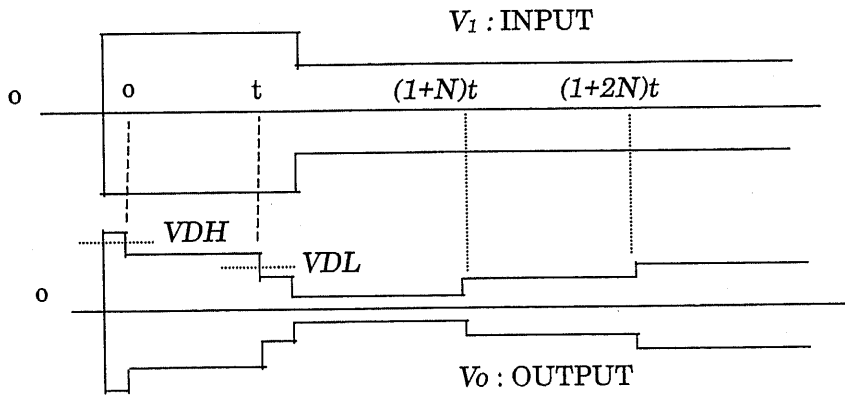


図 3. 10 ALCの入出力波形  
Fig. 3.10 Waveform of ALC

収束レベルは、式 3. 6 となる。収束の時定数は式 3. 7 で、 $N=1$   $S_A = S_{AH}$  によってアタックの時定数を得て、 $S_A = -S_{AU}$  でリカバリーの時定数を得る。そして  $V_o < V_{DL}$  を  $N$  回計測して  $S_{A1}$  のデクリメントをすると、時定数を長くできる。

このようにゲイン制御型回路によれば、時定数回路を使わず必要な特性が実現

可能と考えられる。

$$V_{DL}g_{A1}(S_{AU}) < V_O < V_{DL} \quad \dots \text{式 3. 6}$$

$$T_A = Nt \frac{g_{A1}(S_{AI}) - V_{DL}/V_1}{g_{A1}(S_{AI}) - g_{A1}(S_{AI}+S_A)} \quad \dots \text{式 3. 7}$$

### 3. 2. 3 実験結果

前項のイコライザ回路と ALC 回路を次の (1) の GCA をトランジスタと演算増幅器で組み立てた。マイクロプロセッサと試供 VTR を接続し、図 3. 1 のシステムで記録再生を行い提案の有効性を確認した。

#### (1) GCA モジュール

各ブロックを構成する GCA を図 3. 1 1 に示す。アッテネータの入出力に、増幅器を配して必要なゲインを得ている。アッテネーションはレジスタ S の置数で可変でき、マイクロプロセッサ援用制御が可能になる。

ゲインは  $g = g_1 g_2 / (1 + aS)$  である。

ACC では、比較器の出力をマイクロプロセッサに伝送するためレジスタ T を使う。そして、各ブロックは GCA モジュール 2 個と周辺回路で構成される。

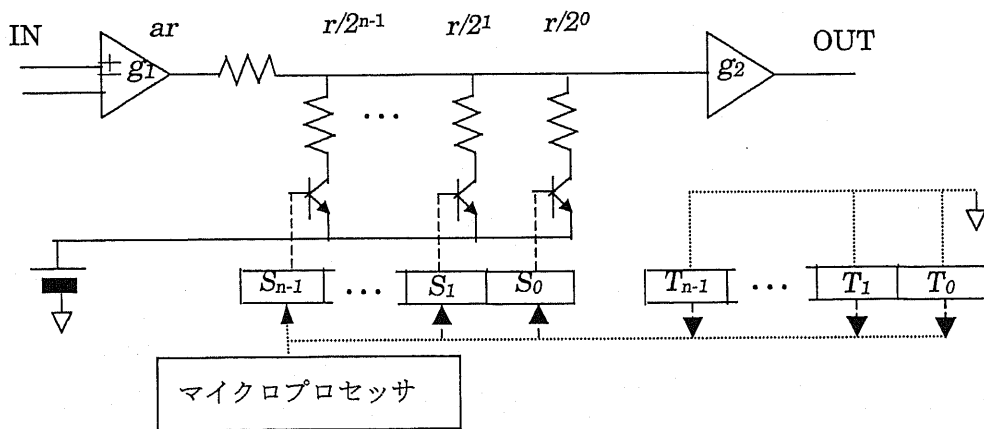


図 3. 1 1 GCA モジュールの構成  
Fig.3.11 Module of the gain controlled amplifier

(2) 再生イコライザ

図3. 7の回路と表3. 1の時定数により、再生イコライザの諸元を表3. 2のように設定した。図3. 12はこの諸元による実験結果である。100Hz~10KHzの実用帯域で図3. 2に準じた特性を得た。すなわち、1個の時定数回路で所定の特性を得た。

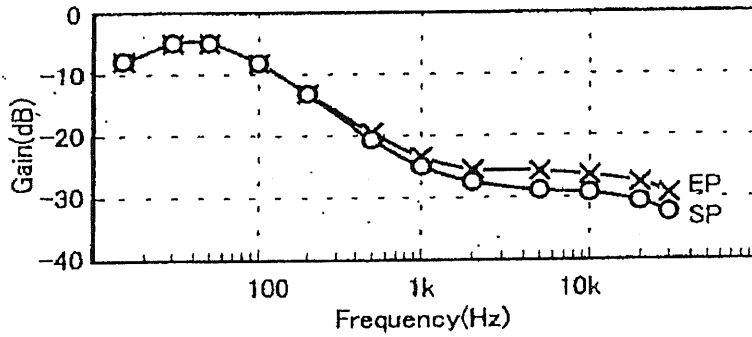


図3. 12 再生イコライザの実験結果  
Fig.3.12 Experimental results of the playback equalizer

表3. 2 再生イコライザの諸元

Table 3.2 Parameters of the playback equalizer

Mode	$g_{P1}$	S	a	n	$g_1$	$C_P$	$R_P$
SP	26	0	0.4	1	26	0.068 ( $\mu F$ )	47 (k $\Omega$ )
EP	18	1					

(3) 記録イコライザ

再生および記録等化前の周波数特性を確認した。入力は一8 dBs のとき、SP の出力を1 KHz で0 dB に正規化し図3. 13を得た。ピーク周波数以下では、図3. 12の特性により再生系で等化されるが、ピーク周波数以上は等化されない。従ってピークからの低下分、10 KHz で3 dB (SP)、5 KHz で10dB (EP) を等化する必要がある。

これを実現するため、図3. 8の記録イコライザの諸元を定めて、表3. 3に

示した。また、VTR に実装して図3. 14に示す周波数特性を得た。すなわち、1個の時定数回路で必要な特性の実現が可能なことを確認した。

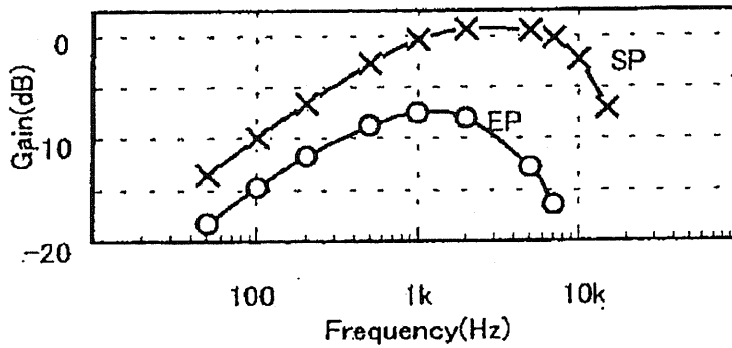


図3. 13 イコライズ前の録再特性

Fig.3.13 Recording and play back characteristics without equalizer

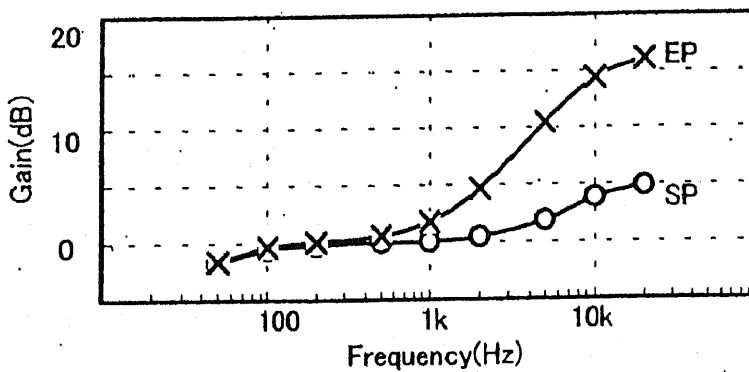


図3. 14 記録イコライザの実験結果

Fig.3.14 Experimental results of the recording equalizer

表 3. 3 記録イコライザの諸元

Table 3.3 Parameters of the recording equalizer

Mode	$\omega_R / (1 + g_{R1}) 2\pi$ (kHz)	$\omega_R / 2\pi$ (kHz)	$C_R$ (pF)	$R_R$ (k $\Omega$ )
SP	9	13	1000	12
EP	1.6			

表 3. 4 記録 GCA の諸元

Table 3.4 Parameters of the recording GCA

Mode	$G_{R1}$	S	a	n	$g_1$
SP	1	1	6.1	1	7.1
EP	7.1	0			



(4) 録再生周波数特性

図3.13の等化前の周波数特性が、記録と再生の両イコライザで等化されることを確認し、図3.15に結果を示した。入力は-8 dBs の値を得、帯域幅はSPで10.5KHzであり、またEPで5 KHzあり、例示の回路と同等の値を得た。すなわち8個の時定数回路を使っていた等化を2個で実現した。

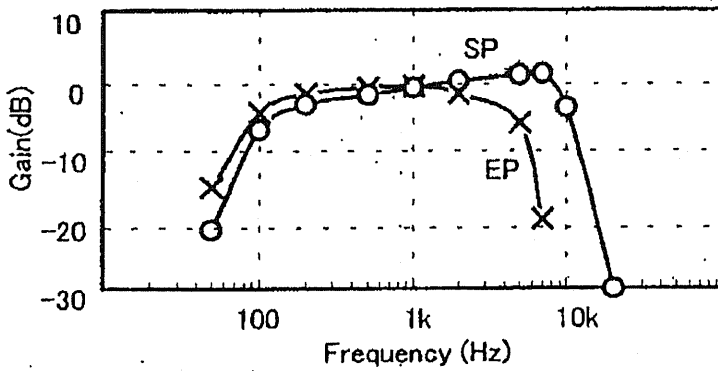


図3.15 記録再生周波数特性

Fig.3.15 Frequency characteristics of recording and playback

(5) ALC

図3.16のAは例示したALCの特性である。収束レベル1  $V_{0-p}$ に達するアタックタイムは、6 msである。

表3.5 ALC用GCAの諸元

Table 3.5 Parameters of GCA for ALC

GCA	$a$	$n$	$g_1$	$V_{DH}$	$V_{DL}$
Controller	0.1	4	1	1.7	1.0
Detector	0.7	1	0.85		

表 3. 6 アルゴリズムの諸元

Table 3.6 Parameters of algorithm

$S_{AH}$	$S_{AL}$	$S_{AU}$	$N$
8	1	1	384

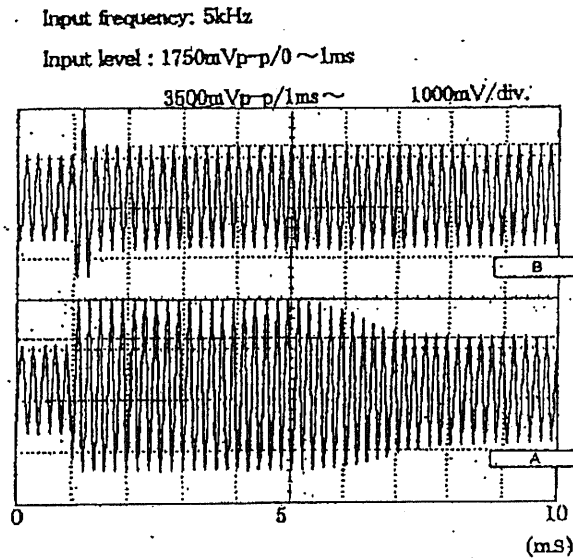


図 3. 16 ALC の実験結果

Fig.3.16 Experimental results of ALC

本提案では、表 3. 5 のように ALC ブロックの諸元設定をし、制御アルゴリズムの諸元を、表 3. 6 のように選んだ。

図 3. 16 の B はこの諸元で得た制御特性である。アタックタイムは  $1/20$  の  $0.3\text{ms}$  に短縮でき、ポップノイズを  $1/20$  に低減できた。またリカバリータイムは例示の回路と同等の  $1.8\text{s}$  を得た。そして例示の ALC では 2 個の時定数回路を使って動作させていたが、時定数回路を使わず実現できた。

### 3. 3 VTR のデジタル特殊効果

現行の映像信号処理の一部にデジタル技術をもちい、多機能高画質を得ようとする試みがある[76、77、78]。映像信号を一度デジタル化してここでいう多機能と高画質を得ることをVTRのデジタル特殊効果と呼ばれている[2]。VTRのデジタル特殊効果とは、メモリを使いフィールド内デジタル信号処理により実現されるマルチ画面、ピクチャインピクチャ、モザイク、ソラリゼーションなどの機能と、従来スピードサーチやスロー再生時に発生したノイズバーとスキュー歪を除去する機能の2つを指す。これらの各機能は映像信号をデジタル変換しメモリに一時留置き、必要なときにそれを読出しアナログに変換し出力する。メモリに書込み時と読出し時にデジタルの映像信号に対して特に時間軸の処理を施すことにより実現している[36、80]。

これらの処理を行うにあたり映像信号をコンポジット信号のままデジタル処理を行う方法と、コンポーネント信号に変換してからデジタル処理を行う方法とがある[37]。コンポジット方式はコンポーネント方式に比べ回路素子とメモリ容量が少なく済むが、色副搬送波の連続性を保つ必要があるという制約があり、カラー画像を取扱う上で難しい。またNTSC (National Television System Committee) とPAL (Phase Alternation by line color television) を同じ基本設計で開発することもまた難しい。一方コンポーネント方式では、色相は色差信号に復調してしまうため色副搬送波の連続性を保つという難題がない為、TVやVTRにおいてデジタル画像処理を行う場合コンポーネント方式を基本にして考え開発する傾向にある[81、82]。しかし、現在開発されているコンポーネント方式のものでは部品点数が多く、開発工数もかかり仕様追加と変更も難しい[37、77]。

これらの問題の解決にはメモリを映像信号専用のフィールドメモリにすることと、必要処理応答時間の長短と単位時間の繰り返し制御回数的大小により、ゲートアレイによる論理回路部とマイクロプロセッサ部の2階層に分割し分担させることである。そこで専用フィールドメモリを持ち、論理回路部とマイクロプロセッサ部に因り成る2階層制御系システムを開発して改善を確認し検討した。

部品点数削減についてはメモリを専用にし周辺の回路を取り込むことで実現する。また開発工数の削減と仕様追加、変更のスムーズな対応は論理回路部を映像

信号の垂平同期信号に対応できるタイミング制御を、マイクロプロセッサ部は映像信号の垂直同期信号に対応できるタイミングで順次制御を担当させることにより見通しのきくシステムを構築することができた[83]。

VTR や TV にこのシステムを組み込み易くするために、VTR や TV 本体とのインタフェースを3系統に集約させた。本体側に当然ある信号も開発の都合により、そのまま使えない場合が多いのでより上位の信号を得て作る。すなわちより少ない情報で必要分は作ることが一見システム全体として無駄にみえるが、より汎用性を高め開発効率の向上には有効である。この3系統の1つめはヒューマンインタフェースと動作モードである。シリアルバスにより情報を交換しマイクロプロセッサにより管理される [69、84]。2つめは映像信号より分離された水垂同期信号と水平同期信号と映像信号の奇数映像と偶数映像の判断情報であり、3つめは異常状態のセンサ入力で論理回路部とマイクロプロセッサ部と同時に受ける処理方式とした。

また検討事項としてこれらの制御部として論理回路部とマイクロプロセッサ部の最適な役割分担がある。また、これは今後の半導体プロセスの向上に因り、その役割は変化して行くと考えられる。また繰返し評価のやり易いマイクロプロセッサの分担を、より大きくすることを目指す手法を提示し設計の自由度を高める可能性を検討している [2、85]

論理回路部のかなりの部分は専用フィールドメモリに内蔵している。ハードウェアロジックの主要な部分は水平同期信号と水垂同期信号の抽出部及びYC分離、フィールドメモリへの書込みと読出しのタイミング発生である。

マルチ画面、ピクチャインピクチャなどや、特殊再生は論理回路部をマイクロプロセッサが制御することにより実現する。例えば、ピクチャインピクチャの子画面の親画面による追越し防止制御（にげ回り処理）や、スピードサーチにおける最善画面の書込みタイミング方式を示している。制御方式も含めた映像信号処理方式が、画質、機能、開発効率、コストにおいて重要であることは当然であるが、個別 LSI、IP の機能・性能の向上及び、周辺回路の取り込みも重要であり、本開発では A/D、D/A、エンコーダ、デコーダも同時期に開発した [87、88]。

今回のマイクロプロセッサと論理回路部及び、画像処理用に工夫されたフィールドメモリにより、十分な機能と性能を働かせることが分かった。

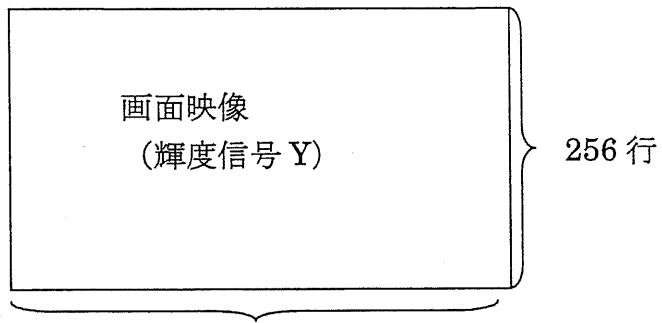
### 3. 3. 1 汎用メモリを用いたシステム

このシステムはコンポジット信号をコンポーネント信号に変換して処理を行う。フィールドメモリは、汎用メモリの 256Kビットデュアルポートダイナミックメモリ 5つを使う[37、76]。

機能としては、ストロボ効果、モザイク効果、ソラリゼーション効果とノイズバーやスキュー歪のないサーチモードの画質改善や横揺れのないスロー再生や時間遅れのない静止画の作成である。

#### (1) フィールドメモリ

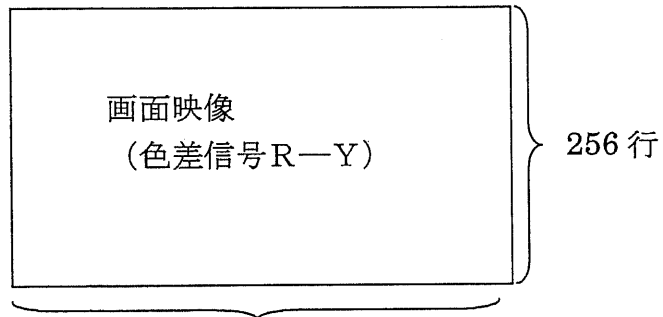
輝度信号Yは6ビット、図3. 17に示すように256行、512ポイント、サンプリングは10MHzであり、1画面当たり768Kビットのメモリが必要となる。色差信号は6ビット、256行、128ポイントサンプリングは2.5MHzのものが2組であり(赤色と輝度信号の色差信号R-Y、青色と輝度信号の色差信号B-Y)、1画面当たり192Kビット2画面で384Kバイトのメモリが必要とする。メモリ構成は、図3. 18に示すように、256K DRAM5つで、輝度信号用に3個、2つの光差信号用に1つ半使う。



512 ポイント×6 ビット/行

必要メモリビット数：512 ポイント×6 ビット/ポイント×256=768K ビット

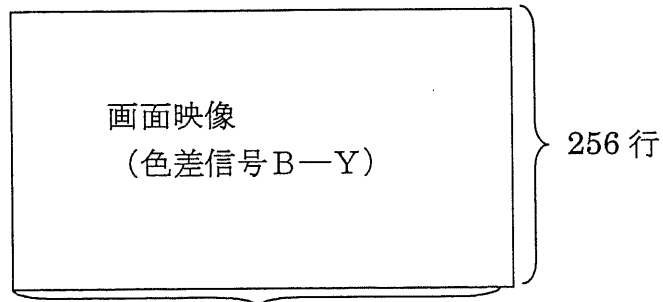
256KDRAM 3つ分



128 ポイント×6 ビット/行

必要メモリビット数：128 ポイント×6 ビット/ポイント×256=192K ビット

256KDRAM 0.75つ分



128 ポイント×6 ビット/行

必要メモリビット数：128 ポイント×6 ビット/ポイント×256=192K ビット

256KDRAM 0.75つ分

図3. 17 映像画面の構成

Fig.3.17 Picture Structure

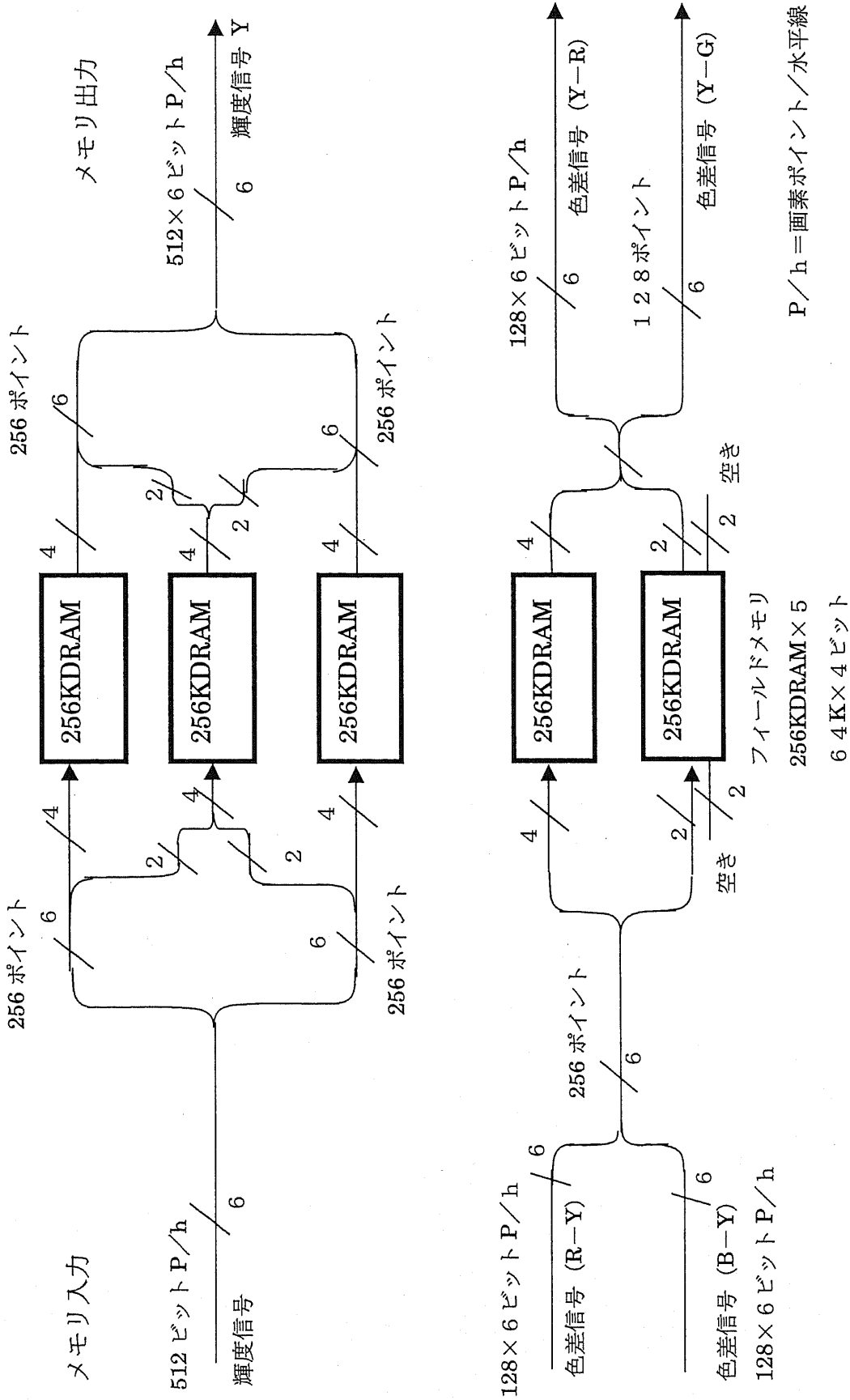


図 3. 18 映像信号のフィールドメモリ配置  
Fig.3.18 Field Memory Construction of Video Signal

## (2) メモリの書込み、読み出しの完全非同期化

映像信号のメモリの書込みと読み出しは完全に非同期にする必要がある。それは例えばピクチャインピクチャの場合、子画面をフィールドメモリに書込み、親画面の同期信号に合わせて読み出すのであるが、親画面用と子画面用の映像信号の同期は異なるためである。

本来デュアルポートの 256 K DRAM は、映像信号用に開発されたものであるが、ピクチャインピクチャのように 2 つの非同時の映像信号を扱う場合、メモリへの書込みと読み出しを非同期に行う必要があるが、読み出しのシフトレジスタにデータを転送する時 6 クロック間は、書込みができなくなる。このため、6 クロックの書込みデータを保持しあらためて、書込みをする必要がある。

図 3. 19 のシステム構成の遅延回路はこの対策用の外付回路である。

## (3) システム構成

コンポジットの映像信号をデコードし、それをさらに 6 ビットのデジタル値に輝度信号は約 10 MHz で、2 つの色差信号はそれぞれ 180 度位相をもたし、2.5 MHz でサンプリングさせる。それらのデータを 5 つの 256K ビットの DRAM に書き込まれる [37]。

それを必要なタイミングで読み出し、D/A に因りアナログに変換し、さらにエンコードして水平、垂直の同期信号を付加してコンポジット信号に戻す。

ピクチャインピクチャや多画面などの各種の付加機能やノイズレス特殊再生は、メモリへの書き込みタイミング、アドレスの制御、読み出しタイミング、アドレス制御、それに D/A のビット制御及び親映像信号とフィールドメモリ読み出しの切換え回路により実現している。

## (4) まとめ 評価

このシステムは、映像信号をデジタルにすることにより、今までアナログでは不可能と考えていたことが実現したわけで、特殊再生のノイズレス化やピクチャインピクチャや多画面で映像画面を思うように取り扱うことができた。また専用の DRAM がない場合でもかなりのことができることが分かった。

しかし、今後このようなシステムを開発していくとき、部品点数が多くなる場



合これらを専用メモリに捕込むか、制御用論理回路に捕込むか実情に合わせて検討をする必要がある。

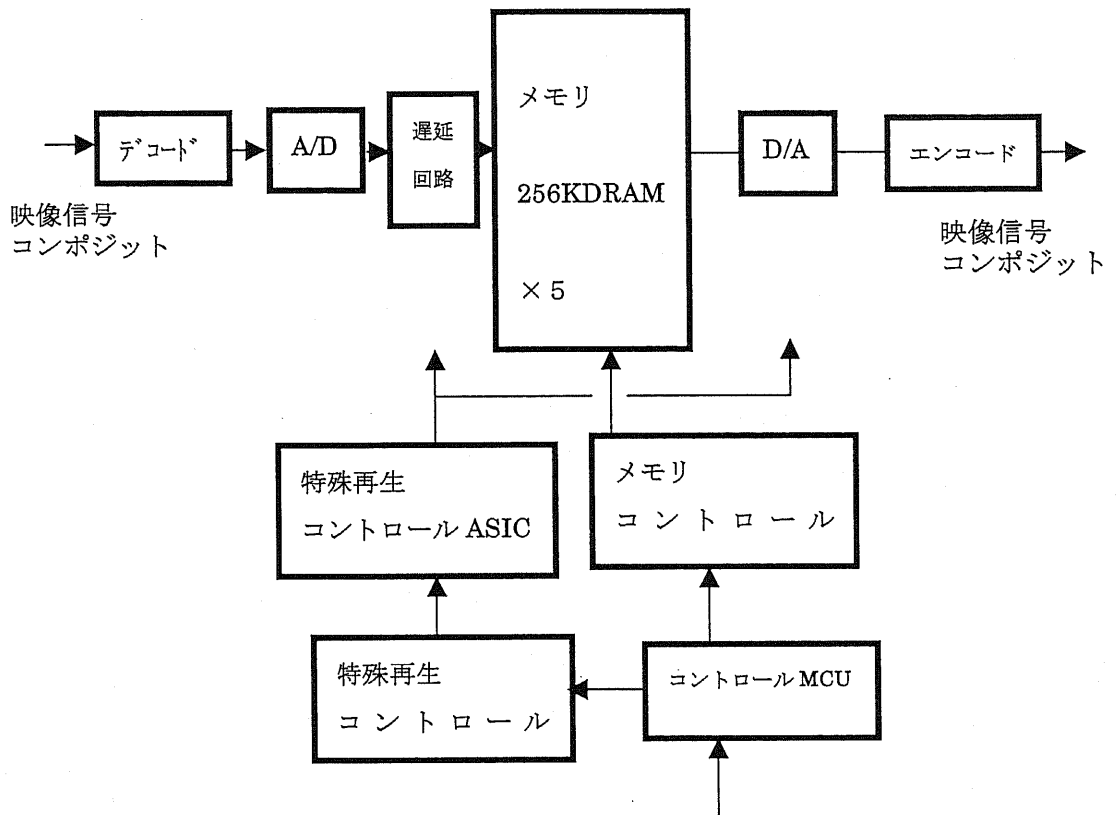


図 3. 1 9 システム構成

Fig. 3.19 System Structure

### 3. 3. 2 TV 映像信号専用メモリを用いたシステム

VTR のデジタル特殊効果を実現するのに、周辺の必要とする素子も含め、専用のメモリを開発し又制御系は高速と低速で論理回路の ASIC とマイクロプロセッサを使いわけ 2 階層としたものを提案している。開発の見通しをよくし開発期間の短縮と品質の向上と部品点数の削減ができる[2]。

#### (1) システム構成の基本概念

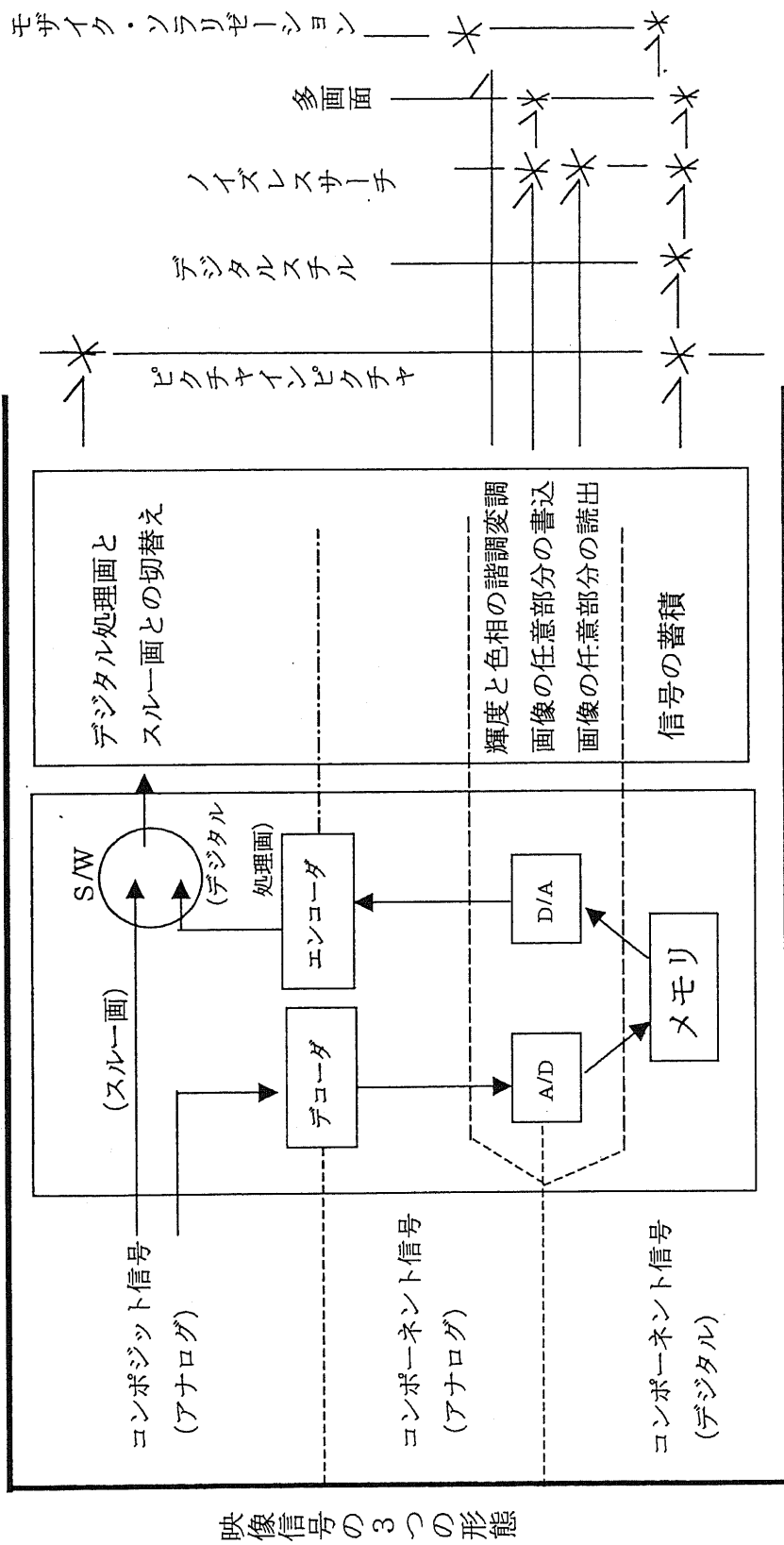
サンプリング周期を  $3 \cdot f_{sc}$  ( $f_{sc}$ : 色副搬送波周波数) の 6 ビットデジタルデータでコンポーネント信号処理を行う。デジタル特殊効果機能としては従来の特殊再生のデジタルによる性能向上によるデジタルスチル、デジタルスピードサーチ、デジタルスローやデジタル処理でないと実現が不可能に近いピクチャインピクチャ、マルチ画面、画面の拡大と縮小、モザイク、ソラリゼーションの機能である。

デジタル特殊効果処理は図 3. 20 に示すようにコンポジット信号、それをデコードしたコンポーネント信号とそれをサンプリングしてデジタルデータにした 3 つの信号の形態で実現させる。コンポジット信号は、VTR、TV の内部での基本的な映像信号で、例えばピクチャインピクチャの機能の実現には子画面をデジタル特殊効果ロジックで作り、このコンポジット信号の切換により行う。

コンポーネント信号は A/D コンバーターでデジタル化されメモリにストアされる。例えばスチル画面は一度データを取り込み同じ画面をメモリより読み出すことにより実現する。さらに高速サーチではノイズレスにするため、高速サーチの映像信号でノイズのないところだけを毎回選択してメモリに取り込み、それを 1 つの画面分読み出す方法により行う。

このようにコンポジット信号、コンポーネント信号、コンポーネント信号のデジタル化されたデータの 3 つが重要な働きをすることになる。それゆえにコンポーネント信号は、これらの処理を行う上で必要である。コンポジット信号のデジタルデータの処理が難しいのは、色副搬送波との位相差によりカラーが決められているからである。画面を切り他の画面につなぐ必要が生じ、つないだ両画面の間で位相を完全に保持することは、映像信号のデジタル化においてサンプリング周期、デジタルビット幅がコンポーネント信号の処理に比べ周期は短くビット幅は大きくなり必要論理回路が大規模になることが推定されるためである。

開発期間の短縮、開発の見通しよくすることと部品点数の削減するシステムの実現は次の3項目が重要である。それはメモリを汎用メモリから映像に適した専用フィールドメモリにする。2つめは専用フィールドメモリ、A/D、D/A、デコーダ、エンコーダ等のLSIに、目標システムが明確であるので必要周辺部品を全て取込む。3つめは映像信号処理の制御を100nsオーダの映像の書込みと読出しの制御信号を周期的に発注するロジックLSI部と、映像信号の垂直周期信号のmsオーダで、ロジックLSI部の制御モードを順次変更し、デジタル特殊効果を実行するマイクロプロセッサ部の2階層で行う制御系とする。これらの2階層制御における役割分担を表3.7に示す。



デジタル特殊効果の実現

デジタル特殊効果への個別技術

映像信号の流れ

図 3. 20 映像信号の3つの形態によるデジタル特殊効果の実現  
 Fig. 3.20 Digital special effect realized by three modes of VCR Signal

表 3. 7 2階層の制御におけるマイクロプロセッサと論理回路との役割分担  
 Table 3.7 Roles of Microprocessor and Logic circuit by dual hierarchical control

	マイクロプロセッサ	論理回路
応答時間	ms 以上	ms～ ns
役割	(1) システムコントロールマイコンとのインタフェース ・ コマンド受信 (ヒューマンインタフェース) ・ モードの送信  (2) メモリ制御用論理回路とのインタフェース ・ 初期書込み、読出しアドレスの設定 ・ ピクチャインピクチャのアドレス設定  (3) 輝度、色差用の D/A 値の変更  (4) 映像信号の偶数奇数判定	(1) TV映像信号用専用メモリ制御 ・ 水垂、水平同期信号よりメモリへの書込、及び読出しタイミングを決定実行する。  (2) ピクチャインピクチャの親画面と子画面の切り替えタイミング発生

## (2) フィールドメモリ

6ビット・3fsc・コンポーネント信号処理システムに適合するフィールドメモリとして、水平信号を一まとめにして入出力できる画面イメージのメモリ配列を持たせることが制御上得策である。そこで320行、256列で6ビットのメモリアレイと、1行分の入力と出力専用シリアルレジスタを持つフィールドメモリとした[83]。

フィールドメモリの概要仕様を表3.8に、ブロック図を図3.21に示す。メモリ制御は図3.24に示すように6ビットの映像信号はシリアル入力クロックの立ち上りエッジでトリガされて入力側シリアルメモリに順次格納される。1行分のデータ入力が完了すると入力側シリアルメモリのデータはあらかじめアドレスされているメモリアレイに転送される、次の行のデータ入力に移る。

データ転送は2種類の制御信号(IRS/CAS、RAS/IRE)により、アドレス端子から時分割入力される転送命令、転送行アドレス、シリアル入力開始列アドレスをストロブして実行される。シリアル出力についても同様の動作となる。このようにシリアル入力回路とシリアル出力回路は互いに独立させたので、入出力は完全に非同期になる。

## (3) 本フィールドメモリの特長

書込みと読出し用クロックを完全に非同期独立させることができる。このメモリの特長として、320行256列のメモリセルに対して、入力と出力に1行分すなわち256×6ビットのシリアルレジスタを各々持たせたことによる。このメモリを使うことにより、フレームシンクロナイザ、ピクチャインピクチャ、TBC(Time base collector)などの実時間で2つのソースを扱うシステムが実現される。

本システムでは図3.22の時間タイミングで本メモリを使っている。1水平同期のブランキング期間に一回だけメモリ内部転送動作制御信号を用いて、垂直方向(行)および水平方向(列)の書き込み、読み出しアドレス、行アドレスカウンタコントロールモード等を命令コードの形でメモリへ送りこむ。このようにしてライン間引や保持をする垂直方向の時間軸変換操作及び水平方向の入出力開始アドレス指定が実行できる。後は、水平方向の時間軸変換操作ができ、メモリ内蔵アドレスカウンタを有効に使用した、水平、垂直方向制御機能の独立した

使い方が可能となる。水平、垂直方向それぞれ時間軸変換率と動作開始メモリ行・列アドレスをマイコンから支えることで画面の拡大縮小やミラー効果など様々な表示画面を実現できる。

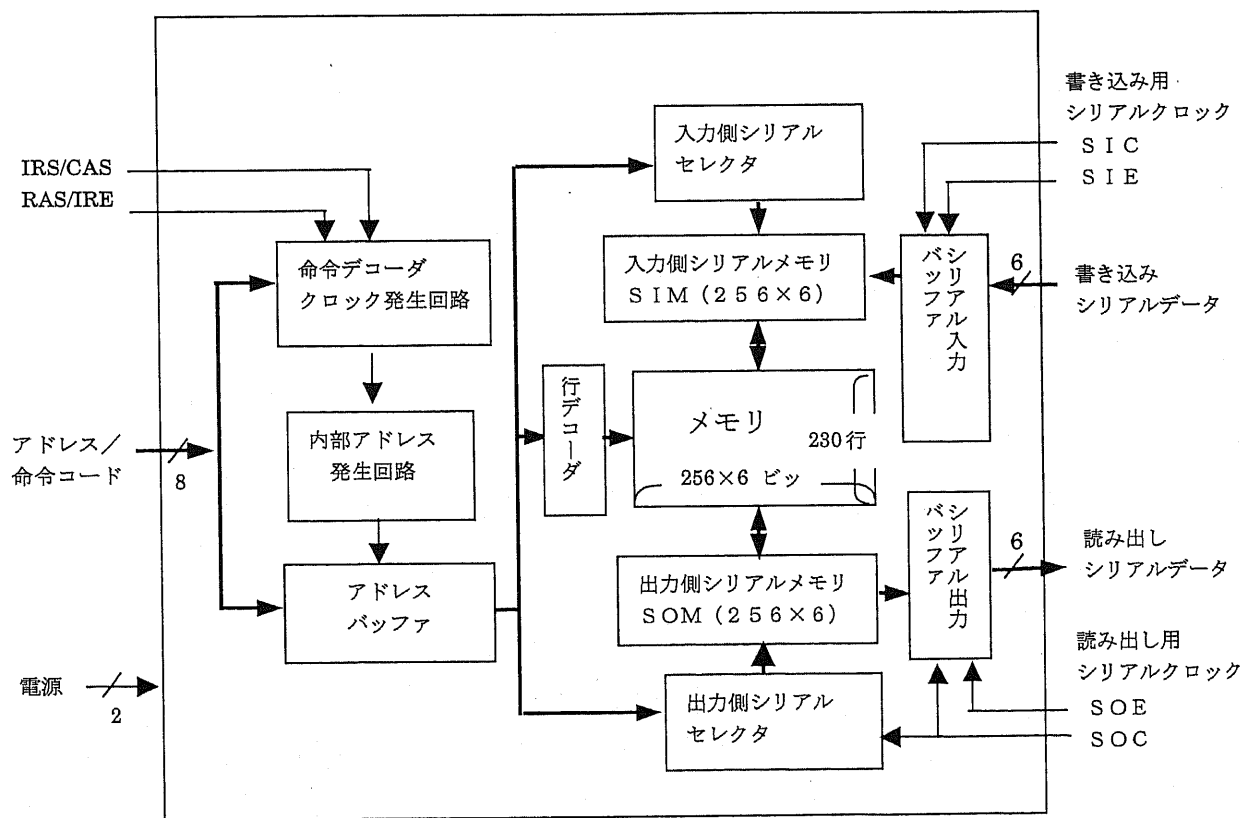


図 3. 21 フィールドメモリのブロック図  
Fig. 3.21 Blockdiagram of Field memory

表 3. 8 フィールドメモリの概要仕様  
 Table 3.8 Specification for outline of Filed Memory

総ビット数	491520 ビット
メモリアレイ編成	320 行、256 列、6 ビット
1 ワード	6 ビット
シリアル入力と出力ポート 各 1 入力と出力とも 1 行分すなわち 256 ワードのバッファメモリを持つ。 入力側シリアルメモリ (256×6 ビット) 出力側シリアルメモリ (256×6 ビット) (シリアル入出力の完全非同期動作可能)	

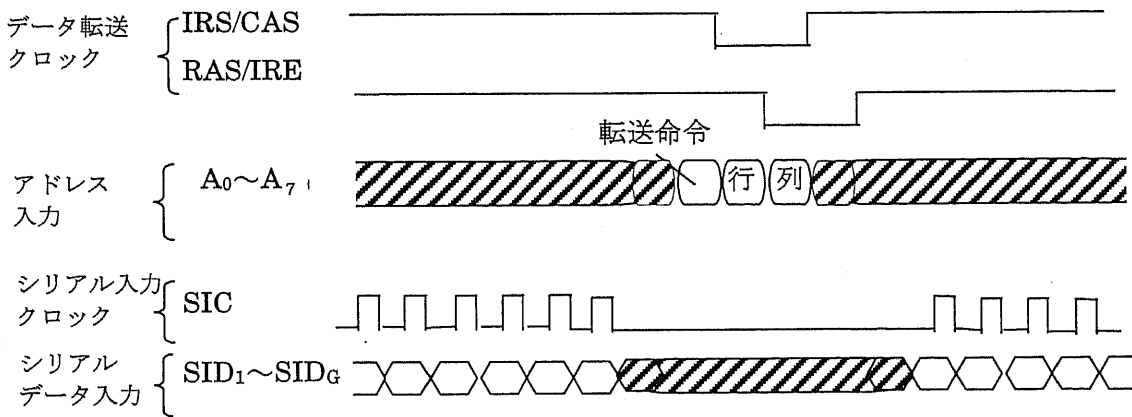


図 3. 22 フィールドメモリのシリアル入力タイミング図  
 Fig. 3.22 Diagram of timing for serial input data in field memory



### 3. 3. 3 システム構成

VTR の特殊効果用に適合した前頁で示したフィールドメモリを持つ図 3. 2 3 のシステムを開発しシステム構成、専用フィールドメモリ、2階層の制御系の有効性を評価する。9品種の LSI を開発し、システムとしては9種17LSI で構成される。

#### (1) 入力部

入力映像信号である NTSC のコンポジット信号を内部処理に必要な輝度 (Y) 信号と色 (C) 信号に、ローパスとバンドパスフィルタを使い分離する。Y信号は A/D 変換器に入力され、又 C信号は位相情報を含むので、色復調器で色差信号 (R-Y、B-Y) に変換される。

標本化周波数は、Y信号は約 10 MHz (640fH)、C信号は約 5 MHz (320fH) とした。C信号は R-Y と B-Y の2つの色差信号よりなるので、R-Y と B-Y はおのおの 2.5 MHz となる。映像信号の Y信号と2つの色差信号 R-Y と B-Y のメモリ配分を図 3. 2 4 に示す。

#### (2) 出力部

図 3. 2 5 に映像信号のメモリへの書込みと読みだしについて示している。3つのフィールドメモリから同時に、Y信号と2つの色差信号 (R - Yと B - Y) が約 5 MHz で読出される。Y信号は2つのメモリに格納されているため、読出された2つのデータは、メモリに読ませたそのときの逆のプロセスで1つに合成する。また、色信号は、R-Y と B-Y が1つに時分割圧縮しているため、これも先ず、2つの色差信号に分ける必要がある。

これらは3つの D/A コンバータでアナログ信号にもどされた後、色差信号を変調して Y信号とミックスさせコンポジット映像信号を得る。

ピクチャインピクチャのように、書込み映像信号と読出し映像信号とが非同期の場合、色信号変調時に親画面の垂直、水平同期信号に位相をあわせる。

このように書込みと読出しの水平と垂直同期信号より、各部への制御信号が論理回路により生成される。メモリ関連としては、メモリ書込みクロック、メモリ内部転送。メモリ読出しクロック、メモリ出力イネーブル、A/D 及び D/A に

関わる必要信号は A/D 変換クロックと D/A 変換クロックがある。映像信号に関わる必要信号は色差点順次用アナログスイッチ切換え、出力ブラッキング、同期信号付加がある。輝度 (Y) 信号は 1 行 512 ポイントで 320 行 6 ビットであるので、983040 ビット、約 1M ビットで、フィールドメモリを 2 ヶ必要となる。また色差信号は 1 行 128 ポイント 320 行あり、6 ビットで R-Y と B-Y の 2 組あるため 496520 ビット約 0.5M ビットでフィールドメモリ 1 ヶ必要となる。

映像信号のサンプリングからメモリへの書き込みの処理系を、図 3.25 に示す。Y 信号は 10MHz のサンプリングでフィールドメモリ 2 ヶ分容量として必要であるため、2 つのフィールドメモリに同時に書くシステムにする。2 つの色差信号 R-Y と B-Y は、2.5MHz であるため、それら 2 つを時分割に合わせると 5MHz になる。すなわち、Y 信号も C 信号も 5MHz で 3 つのフィールドメモリに同時に書き込みすることが可能となる。それで 3 つのフィールドメモリには全く同一の処理でメモリを制御できる。メモリからデータを読み出すときは、この場合と全く逆の流れになる。変調バースト付加、親子画面切換えの各制御信号である。

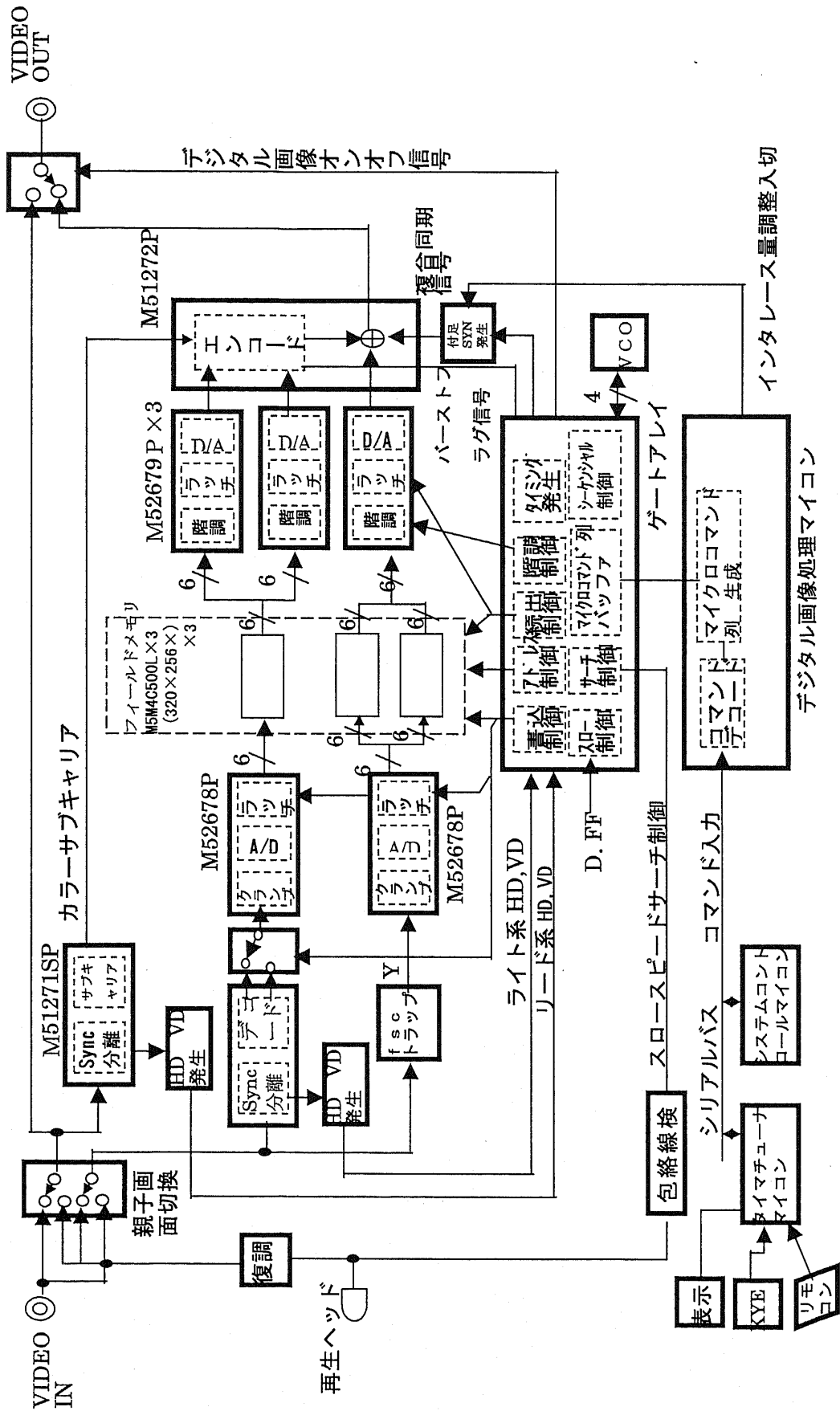
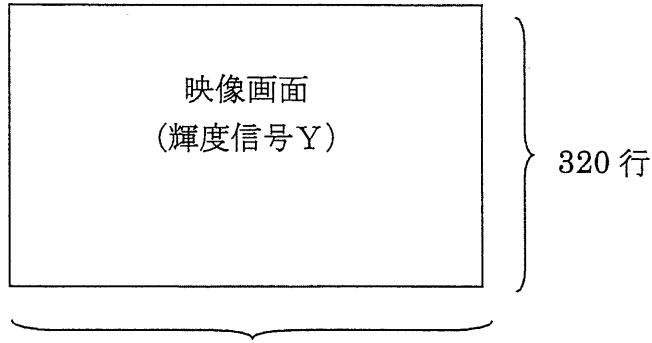
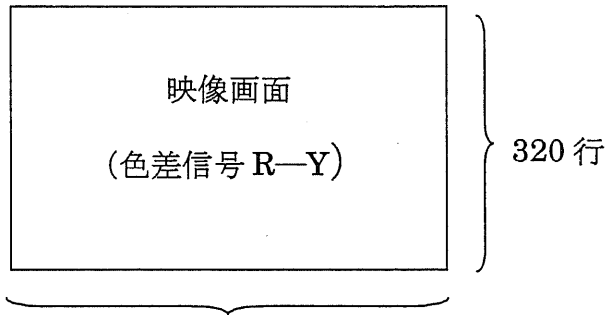


図 3. 23 TV映像信号専用メモリシステム  
 Fig. 3.23 Memory system exclusively for TV picture signal



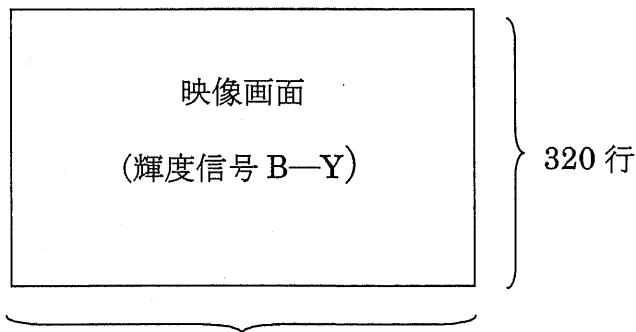
512 ポイント×6 ビット/行

512 ポイント×6 ビット/ポイント×320 行=983040 ビット  
フィールドメモリ 2ヶ分



128 ポイント×6 ビット/行

128 ポイント×6 ビット/ポイント×320 行=245760 ビット  
フィールドメモリ 1/2ヶ分



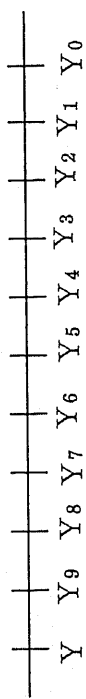
128 ポイント×6 ビット/行

128 ポイント×6 ビット/ポイント×320 行=983040 ビット  
フィールドメモリ 1/2ヶ分

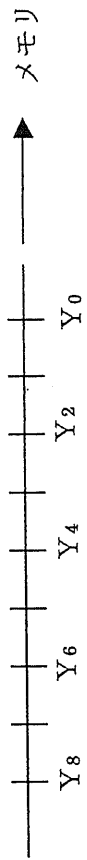
図 3. 24 TV 映像信号専用メモリによる映像画面の構成

Fig. 3.24 Picture construction of memory exclusively for TV picture signal

輝度信号 (10MHz) サンプルング周波数

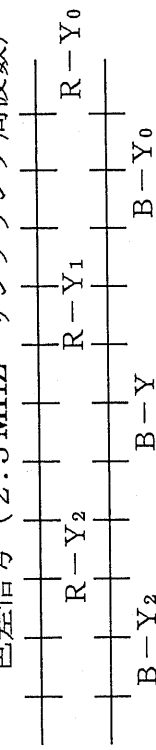


分離

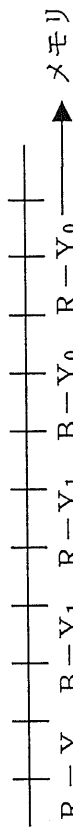


輝度信号 2 系列に分離 (5 MHz)

色差信号 (2.5 MHz) サンプルング周波数



合成



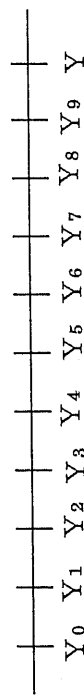
色差信号 2 系列を 1 系列に合成

(5 MHz) メモリへの書き込み周波数 5 MHz 及び

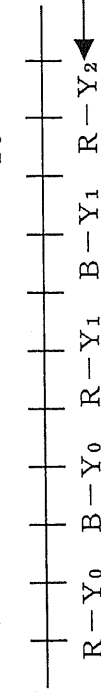
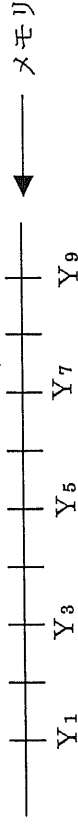
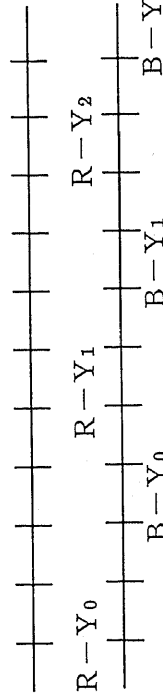
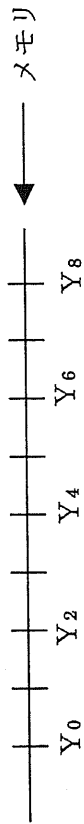
a; 映像信号のメモリへの書き込み

メモリへの書き込みアドレス行と列 3 つのメモリへの制御同じ

輝度信号 (10 MHz)



合成



b; 映像信号のメモリからの読み出し

図 3. 25 映像信号のメモリへの書き込みと読み出し

Fig. 3.25 Read and write on picture signal memory

### 3.3.4 特殊効果の実現

特殊効果の実現は、3.3.2項で示したようにコンポジット映像信号をコンポーネントに、さらにそれを、デジタル変換して各3つの映像信号において論理回路により、 $\mu\text{sec}$  オーダのタイミング制御を、マイクロプロセッサにより msec オーダのシーケンス制御により行われる。

それをさらに具体化の段階でスムーズに見通しよく開発進行でき、又、部品点数をより少なくするために専用のフィールドメモリを新たに開発した。これらの機能をフルに使い特殊効果を実現する手法について述べる [2、83、84]。

#### (1) フィールド静止画

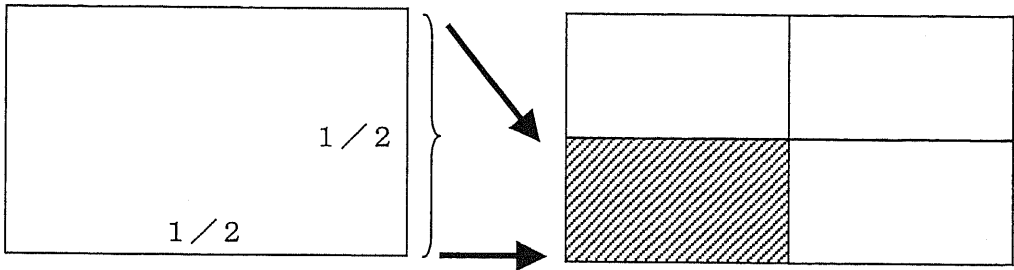
フィールドメモリには、1秒間に60枚のフィールドが書き込まれる。フィールドは奇数と偶数順次繰り返されるが、静止画の場合、止められた奇数画面かどちらか一方で静止画の再生を行うことになる。それで垂直方向の分解能力は $1/2$ となる。又、毎回奇数画面と偶数画面を繰り返すことになり、それでイレタレース量調整のため出力画面に付け足す垂直同期信号を片フィールドのみ遅延させる。

#### (2) マルチ画面

複数の画面を表示するもので2画面、4画面、9画面、16画面を選択できる。もちろん25画面とか36画面は技術的には可能である。

4画面の場合は縦枠を $1/2$ に縮小したものであり、9画面では $1/3$ 、16画面では $1/4$ である。

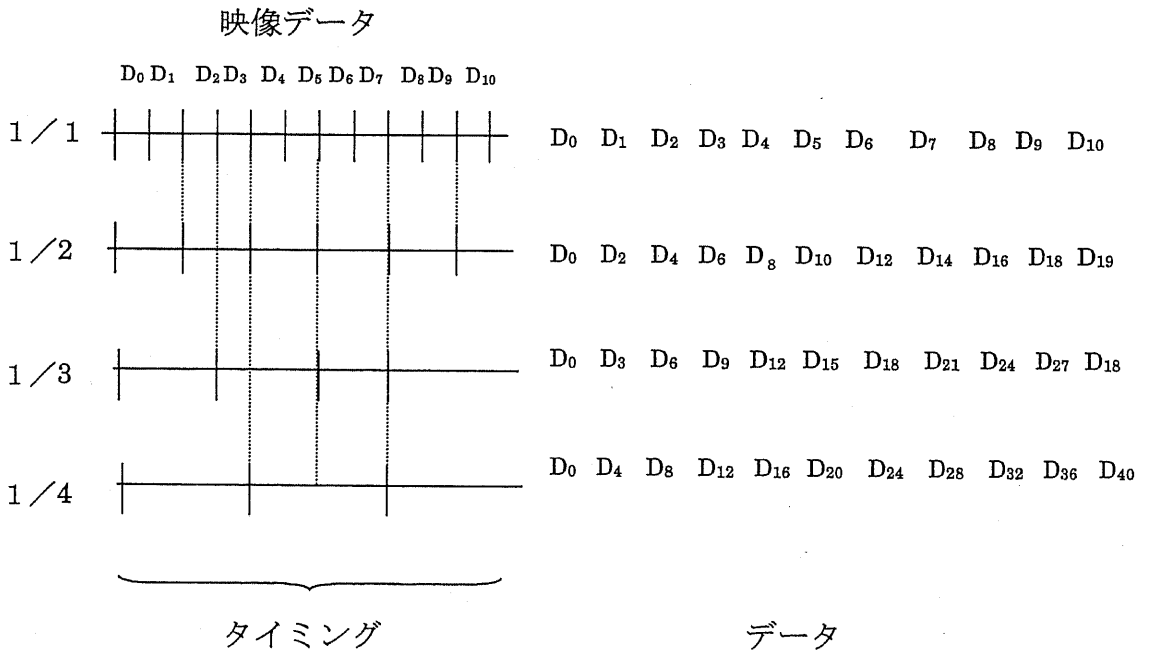
1フィールドの入力画像に対して、4画面の場合なら、2画素に1画素を、9画面の場合なら3画素に1画素を取り込む。又、水平ラインも同じく4画面2ラインに1ラインに対応し、9画面なら3ラインに1ライン対応する。メモリの書込みアドレスは4画面なら4つの内で1つを、9画面なら9つの内で1つを選ぶことになる。図3.26に多画面のための画面縮小手法を示す。



映像信号の画面

メモリの画面

(a) 画面縮小 縦、横をそれぞれ  $1/2$  にして、 $1/4$  の縮小画面を4つ作成



(b) メモリへの取込

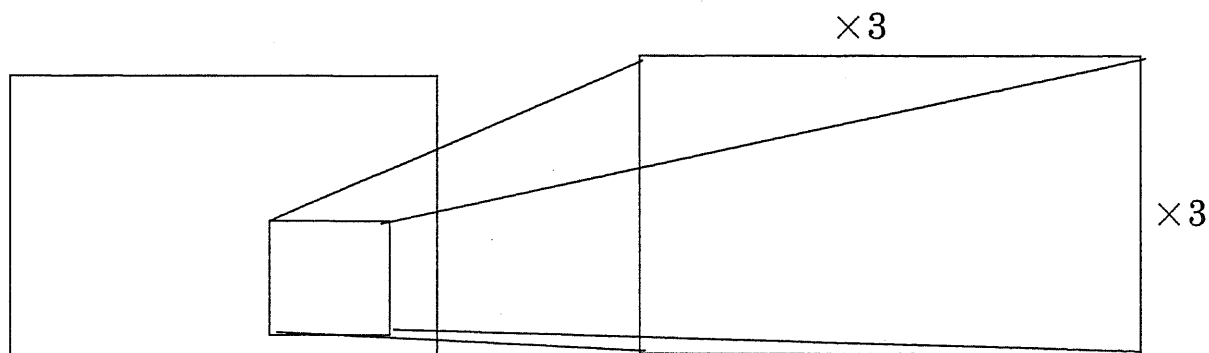
図 3. 2 6 マルチ画面の作成方法  
 Fig. 3.26 Method of drawing up multi-screen

### (3) 拡大

この機能は1点の同じ画素を拡大させ塗潰す方法で実現する。フィールドメモリの読み出し時間を、2倍、3倍、4倍と長くすることにより、1つの画素が2倍、3倍、4倍と表示され又、縦方向の拡大には同じ水平映像信号を2回、3回、4回と使うことにより、画面が4倍、9倍、16倍と拡大される。

拡大時の画質向上を行う為に2つの画素の平均をとるなどの方法があるが、今回はそのままのデータを使う簡単なやり方を採用した。





フィールドメモリ内映像データ

表示画面

(a) 画面拡大縦横3倍ずつの拡大で9倍拡大画面を得る

水平方向の拡大

アドレス  $n, n+1, n+2, n+3, n+4, n+5$

データ  $D_0, D_1, D_2, D_3, D_4, D_5$       メモリよりの出力データ

アドレス出力

1倍       $n, n+1, n+2, n+3, n+4, n+5 \rightarrow D_0, D_1, D_2, D_3, D_4, D_5$

2倍       $n, n, n+1, n+1, n+2, n+2 \rightarrow D_0, D_0, D_1, D_1, D_2, D_2$

3倍       $n, n, n, n+1, n+1, n+1 \rightarrow D_0, D_0, D_0, D_1, D_1, D_1$

たて方向の拡大

アドレス 水平データ    1倍の場合    2倍の場合      3倍の場合

$m$              $l$              $m$      $l$              $m$      $l$              $m$      $l$

$m+1$          $l+1$          $m+1$   $l+1$      $m$      $l$              $m$      $l$

$m+2$          $l+2$          $m+2$   $l+2$      $m+1$   $l+1$      $m$      $l$

$m+3$          $l+3$          $m+3$   $l+3$      $m+1$   $l+1$      $m+1$   $l+1$

$m+4$          $l+3$          $m+4$   $l+3$      $m+2$   $l+2$      $m+1$   $l+1$

$m+5$          $l+4$          $m+5$   $l+5$      $m+2$   $l+2$      $m+1$   $l+1$

(b) 水平方向とたて方向の拡大

図3. 27 画面拡大方法

Fig. 3.27 Method of screen enlargement

#### (4) モザイク

モザイクとはすりガラスを通して見える効果を期待するものであり、その実現方法には種々考えられるが、ここでは一番簡単な方式を使った。画面を正方形でうめつくし、その正方形に1つの画素 (Y, R-Y, B-Y) を全てコピーする。その正方形の大きさもデジタル化された映像信号の2つ分か3つ分か4つ分かそれ以上かは仕様として決定される。図3. 28においてモザイクの正方形の大きさを3画素3ラインとする場合水平方向の処理は3画素のはじめのものを次の2つにコピーする。読出すメモリの値を3クロック間同値にするため、同じメモリアドレスから3回読出す。たて方向は、同じラインを3回読出し、その後+3ラインして水平ラインで同じ処理を繰り返す。

ある一部のみにモザイクをかける場合、これらの処理を、あるアドレスの範囲に処理することより可能になる。

水平画素		n	n+1	n+2	n+3	n+4	n+5	n+6	n+7	n+8
水平線	1	D	D1	D2	D3	D4	D5	D6	D7	D8
	1+1	E	E1	E2	E3	E4	E5	E6	E7	E8
	1+2	F	F1	F2	F3	F4	F5	F6	F7	F8
	1+3	G	G1	G2	G3	G4	G5	G6	G7	G8
	1+4	H	H1	H2	H3	H4	H5	H6	H7	H8
	1+5	I	I1	I2	I3	I4	I5	I6	I7	I8
	1+6	J	J1	J2	J3	J4	J5	J6	J7	J8
画面イメージのメモリ内映像信号データ										
水平画素		n	n+1	n+2	n+3	n+4	n+5	n+6	n+7	n+8
水平線	1	D	D	D	D3	D3	D3	D6	D6	D6
	1+1	D	D	D	D3	D3	D3	D6	D6	D6
	1+2	D	D	D	D3	D3	D3	D6	D6	D6
	1+3	G	G	G	G3	G3	G3	G6	G6	G6
	1+4	G	G	G	G3	G3	G3	G6	G6	G6
	1+5	G	G	G	G3	G3	G3	G6	G6	G6
	1+6	J	J	J	J3	J3	J3	J6	J6	J6

図3. 28 モザイク

Fig.3.28 Mosaic image

画面イメージのモザイク処理後の映像信号データ9画素が全て同じ画素データになる。

### (5) ピクチャインピクチャ

1つの画面上に、実時間で書き込まれる1/9画面をはめ込む子画面入力情報のメモリへの書込みは、マルチ画面と同様に行う。

子画面は1/9画面であるので、メモリには4つの子画面を作成し、書込み、読出しにおけるフィールド内追越しをマイクロプロセッサで4つの子画面の選択で回避する。

この子画面を親画面にうめ込むが、このとき親画面の映像信号の水平・垂直同期信号から所定のタイミングで子画面情報を1フィールド分のみを読出すとともに、親画面の映像信号とアナログスイッチにより切替え表示画面を得る。子画面入力映像とそのメモリ上の縮小画面と親画面へのうめ込まれる画面の関係を図3.29に示す。

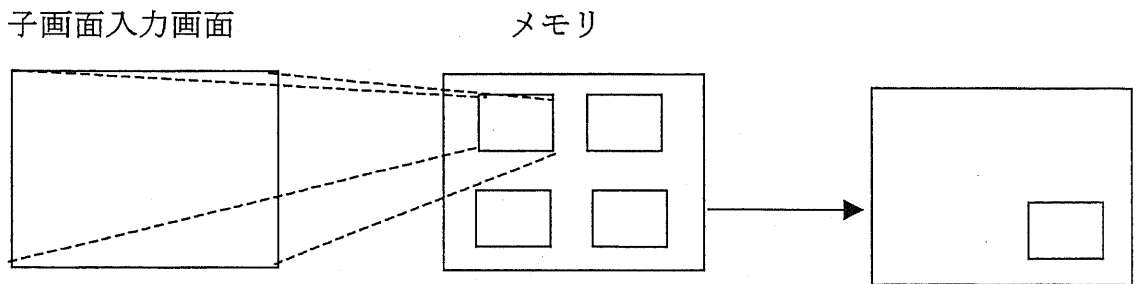


図3.29 ピクチャインピクチャ

Fig. 3.29 Picture In Picture

## (6) デジタルスピードサーチ

機構上の基本動作は従来と同じ方式である。VHS (Video Home System) の標準モードと3倍モードにそれぞれ対応する2対のビデオ・ヘッドを使う。これにメモリとマイクロプロセッサにより、標準モードで記録したテープを6倍速で再生するときノイズバーとスキュー歪を除去する。

一般にサーチモードは奇数倍速を採用してきた。これは図3.30に示すごとく画面上でノイズバーが現れる位置を固定し、ノイズバーが全画面に散乱することを止め、少しでも見やすい画面を行うためである。しかし、今回のシステムは、偶数倍速を採用する。偶数倍速にすると画面上でS/Nの高い部分とノイズバーになる部分がフィールドごとにずれて現れる。

このようにしておき、S/Nの高い信号だけをメモリに取り込み補間すれば、1フレームの期間にノイズバーのない1フィールド分の再生信号をメモリに取り込める。このようにしてノイズバーのない再生画面が作れる。またVTR出力の同期信号はヘッドからの信号とは完全に独立に付けなおすのでスキュー歪もなくなる。

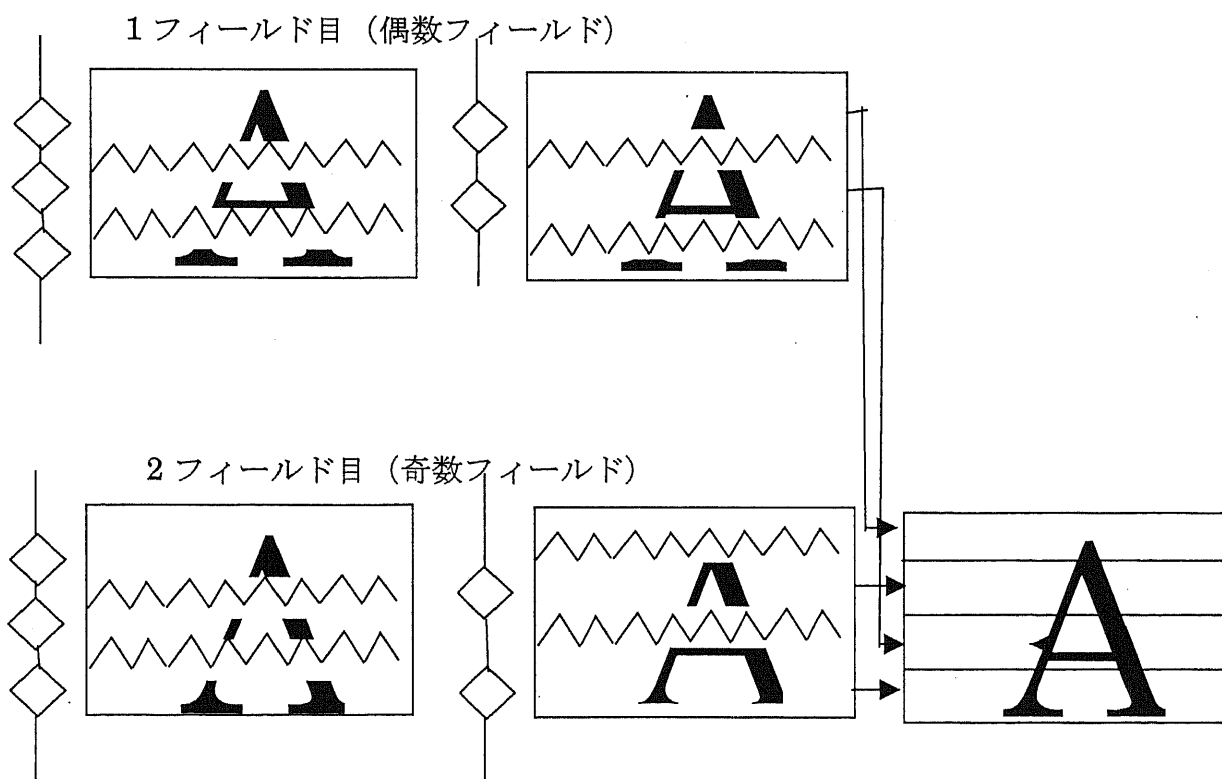


図3.30 デジタルスピードサーチ  
Fig. 3.30 Digital speed search

### 3. 3. 5 開発評価

VTR のデジタル特殊効果システムを開発した。コンポーネント方式により 3 fsc・6 ビットの量子化で新たに 11 品種の LSI を開発し、17 LSI より成るシステムになった。

デジタル特殊効果の機能としては、ピクチャインピクチャ、モザイクソラリゼーション多画面、拡大と縮小などのメモリを付加した今日のこのようなシステムでないと実現できない家庭用 VTR としては実現できない新機能と、特殊再生と呼ばれるノイズレススピードサーチ、デジタルスチル、デジタルスローなどで、ノイズバーやノイズがなく画面のユレもない特殊再生を実現した。これらの写真を図 3. 31 に多画面として 9 画面のものを、図 3. 32 にモザイク、図 3. 33 にソラリゼーション、図 3. 34 にピクチャインピクチャを示す。

システム開発は、フィールドメモリが、周辺素子を取り込み、256 ポイントのシリアル入力と出力を独立させたことなど、システム構築上有効であった。又、マイクロプロセッサと論理回路との二階層制御方式は開発の見通しをよくした。今後の開発システムにおいても、高速処理の 100ns オーダの論理回路と低速シーケンス処理の ms オーダの二階層方式は重要な方式と考える。

仕様決定後、システムは計画を大きく短縮する開発期間で完成した。当初開発期間 8 ヶ月、開発人工 80 人月のところ、開発期間 5 ヶ月、開発人工 60 人月であり、25% の開発人工を削減したことになった。

今回の開発の項目と人月を示す。また ROM、ゲート数も示す。但し、11 LSI の設計人工は含まず。

	開発期間	開発人月
全システム設計	1 ヶ月	2 人月
ゲートアレイ設計 (20Kゲート)	5 ヶ月	30 人月
マイクロソフトウェア開発 (8Kバイト)	5 ヶ月	18 人月
全システム評価	20 月	10 人月
計	5 ヶ月	60 人月

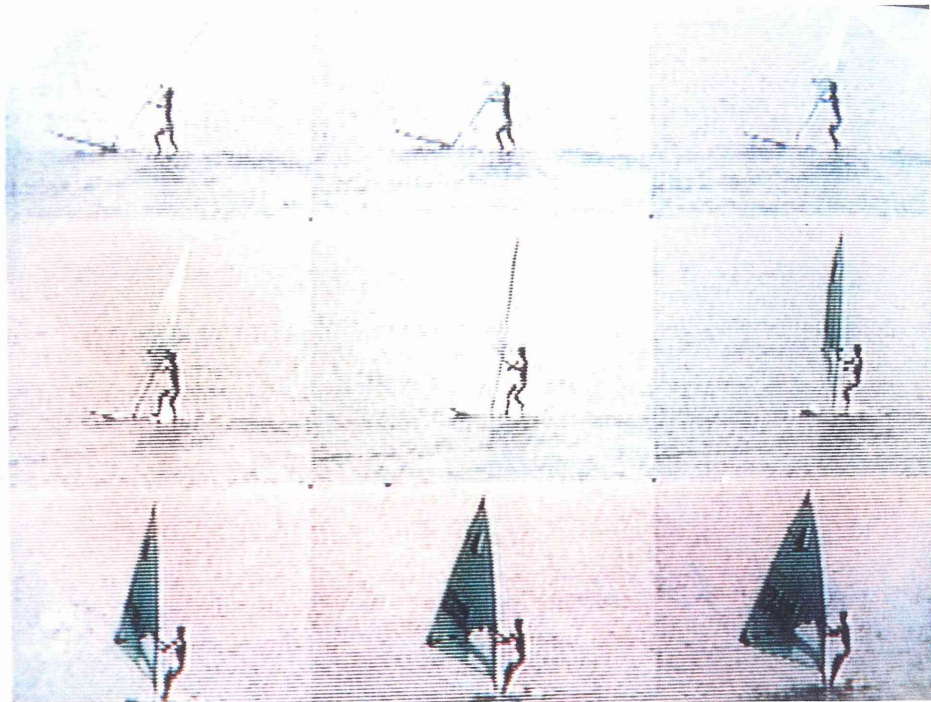


図 3. 3 1 デジタル特殊効果 ; 9 画面表示  
Fig.3.31 Digital special effect ; 9 pictures



図 3. 3 2 デジタル特殊効果 ; モザイク  
Fig.3.32 Digital special effect : Mosaic pictures



図 3.3 3 デジタル特殊効果；ソラリゼーション  
Fig.3.33 Digital Special Effect ; Solarization



図 3.3 4 デジタル特殊効果；ピクチャインピクチャ  
Fig.3.34 Digital Special Effect ; Picture in picture

### 3. 4 むすび

映像機器におけるマイクロプロセッサの応用に対する研究の1つとして、マイクロプロセッサ援用信号処理技術の確立を、オーディオ信号処理とVTRデジタル特殊効果システムにおいて試み、成果を得た。マイクロプロセッサ援用信号処理とは、マイクロプロセッサが直接信号を処理するのではなく、信号処理のLSIの中ではシリアルバスを介してレジスタ制御によりデータの出力と入力を行う処理のことを指す。

オーディオ信号処理における交流バイアス音声記録では、イコライザやALCで必要な時定数の切換えを、ゲイン制御で実現する方法について提案した。イコライザは双1次特性が使われるが、その極はフィードバックゲインで制御でき、零点はフィードフォワードゲインで制御できることを活用した。ALCでは、マイクロプロセッサの実行周期あたりのゲイン制御を、アタックとリカバリで切換えて異なる時定数を得た。VTR用バイポーラ集積回路では、10個の時定数回路を外付け部品であるが、提案の手法により2個に削減ができる目処を得た。帯域幅とリカバリタイムは性能は変わらず、さらにアタックタイムは $1/20$ に短縮できた。

VTRのデジタル特殊効果において、映像をデジタルに変換してからのメモリ、A/D, D/Aへの制御が10MHz前後の速度が必要となる。このためマイクロプロセッサのみの制御は難しく論理回路LSIを加えた2階層の制御システムで実現していくことを開発着手当初より考えていた。

しかし仕様はそのシステムの価値を決め、またLSIの規模と開発の難度に大きく左右するため、なかなか決まらなかった。すなわちデジタル処理の映像信号をコンポジット信号なのか、コンポーネント信号なのかという問題、そして各々においてデジタル化へのサンプリング速度とビット巾の選択と、それに合わせたマイクロプロセッサと論理回路LSIとの2階層の制御システムでの実現可否の検討である。この開発と製品化がうまくいったことにより、ここでの仕様決定も研究と一連の技術開発の成果とすべきと考える。これらは、デジタル処理の映像信号形はコンポーネント方式、デジタルの量子化は6ビット、輝度信号は10MHzのサンプリング速度で水平512ポイント、320行で色差信号は2.5MHzのサンプ



リング速度で水平 128 ポイント、320 行で R-Y と B-Y の 2 組である。

VTR のデジタル特殊効果でのマイクロプロセッサ援用信号処理は、高速処理制御用に論理回路 LSI と 2 階層の制御システムを構成することにより実現した。必要する制御が高速（論理回路 LSI）であるか、または低速（マイクロプロセッサ）かの役割分担も重要であるが、制御の上位をマイクロプロセッサが担当し下位は論理回路 LSI が担当する。そして仕様変更の多いヒューマンインターフェースはマイクロプロセッサで、また機器とのタイミングの合わせ込みやフィードバックループを用いる学習制御は、マイクロプロセッサと論理回路 LSI の組み合わせという構成になる。すなわち開発時における仕様変更や合わせ込みの調整などは全てマイクロプロセッサが関わり、論理回路 LSI はその高速処理の部分を活かし、その特性変更はマイクロプロセッサによりシリアルバス、レジスタ制御により行わせる。論理回路 LSI の回路変更を極力少なくすることが、マイクロプロセッサと論理回路 LSI による 2 階層制御システムの重要項目と考えた。またこれが今回の研究の成果である。これにより VTR のデジタル特殊効果システム開発の人工を当初の 80 人月から 60 人月に 25% の削減を実現した。

以上のようにマイクロプロセッサによるレジスタ制御でのオーディオ信号処理と VTR デジタル特殊効果システムにおける映像信号処理をマイクロプロセッサ援用信号処理により実施して部品点数 80% と開発人工 25% が減じられ、VTR の開発工数の低減に効果があると考えられる。またこれがマイクロプロセッサ援用信号処理技術確立推進の一方法と考える。

## 4 マイクロプロセッサによる信号処理

### 4. 1 まえがき

TV 映像信号の MPEG-2 (Moving Picture Expert Group-2) のデコードの IDCT (Inverse Discrete Cosine Transform) を RISC 32ビットマイクロプロセッサのソフトウェアにより処理することを研究し、実現の見通しを得た [8]。

ここで述べるソフトウェア処理の目指すものは、開発期間の短縮及び開発工数削減と全ハードウェア量の削減、それらは主に開発と生産に関わるものであり、もう一方はシステム構築の自由度の拡大で経年における市場動向への対応の容易性である。こういう状況の中、CRIS 32ビットマイクロプロセッサの性能を高めるためにハードウェアの強化、すなわち第2ALUを提案し、さらにそれがよりIDCTを処理する上で性能向上を計るための改善を行った。また、IDCTを含むMPEG-2のデコードをマイクロプロセッサのソフトウェア処理を主体で行う試みとして、第2ALUを含むバススイッチ結合マルチメモリマルチプロセッサ（以下バススイッチ結合マルチプロセッサと称する）を新たに提案し検討した。

マイクロプロセッサにより大きい性能と機能が要求されるが、より効果的にそれを得るために、第2ALUの提案が時宜を得たものと考え、今回IDCT処理で検討評価できた。そして新たに検討していきたいバススイッチ結合マルチプロセッサ構造（アーキテクチャ）と、第2ALUが今後マイクロプロセッサの機能・性能をさらに飛躍させると考える。

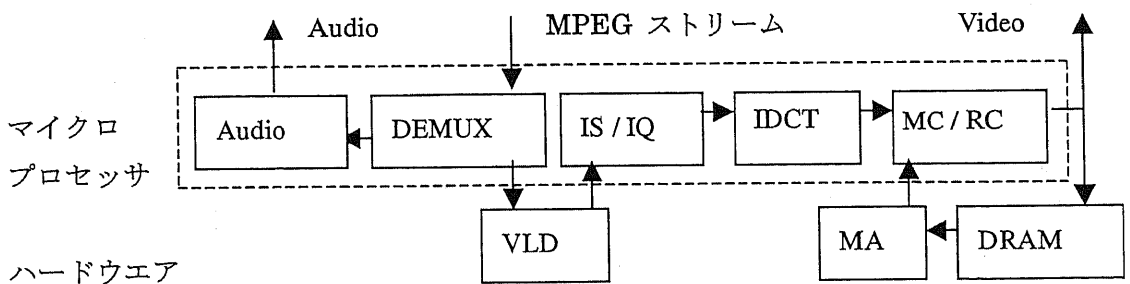
## 4. 2 アルゴリズム

8 IDCT の処理は、64 回の積と 64 回の和の演算が必要であるが、4 積和演算を使って 13 回の演算で実行するアルゴリズムを示した。

DVD やデジタル TV のシステム LSI が進展している。すなわち、プロセッサとメモリとその他の回路を SOC 化して、1 チップソリューションを得ることが期待されている [59、71、90、105]。映像機器のマイクロプロセッサの応用は制御から進んだ [1、3]。そしてデジタル信号処理に展開された [2、91]。そしてマイクロプロセッサと論理回路と混載 DRAM で構成される SOC が期待されている。

一方、MPEG-2 デコーダのシステム LSI では、メディアプロセッサと高速論理回路と外付け DRAM が使われている [59、92]。DRAM 混載はむしろ、JPEG への応用を中心に RISC プロセッサで実用されている [93]。本来 DRAM 混載は、メモリバンド幅の必要な用途で効果が享受できる [116]。すなわち MPEG-2 デコーダの方が、ノイマン・ボトルネック回避の必要度が高いので、DRAM 混載 RISC 32 ビットマイクロプロセッサによるソリューションが求められる [8]。

図 4. 1 に典型的 MPEG-2 デコーダを示す。250M 並列度 4 のメディアプロセッサを使った場合、MPEG-2 を 193MIPS で処理している。内 121.5MIPS が IDCT である。また MPEG-2 の CPU 時間専有率は 77% である [92、95]。



DEMUX: Demultiplexer      RC: Reconstruction      IS: Inverse Scan  
VLD: Variable Length Decoder      IQ: Inverse Quantization      MA: Memory Access  
MC: Moving Compensation      MA: Memory Access

図 4. 1 MPEG-2 デコーダの機能構成  
Fig. 1 MPEG-2 decoder

#### 4. 2. 1 動画像の IDCT に必要な演算量

図4. 2、図4. 3、図4. 4、図4. 5に映像のフレームおよびブロック構造を示す。

4:2:0の動画では、画像の処理単位を8ライン (lines) × 8ピクセル (pixels)のブロックとする。そして、Y (輝度) が4ブロックおよびCb (カラー・ブルー) とCr (カラー・レッド) で2ブロック、計6組をマクロブロックとして、1秒に  $243 \times 10^3$  ブロック分に相当する [92]。

すなわち、IDCTも1秒に  $243 \times 10^3$  ブロック分実行する必要がある。1ブロックのIDCTは、1024回の積と1024回の和で実行されるので、498MOPS (Million Operating per second) の演算能力が要求される [93、105]。

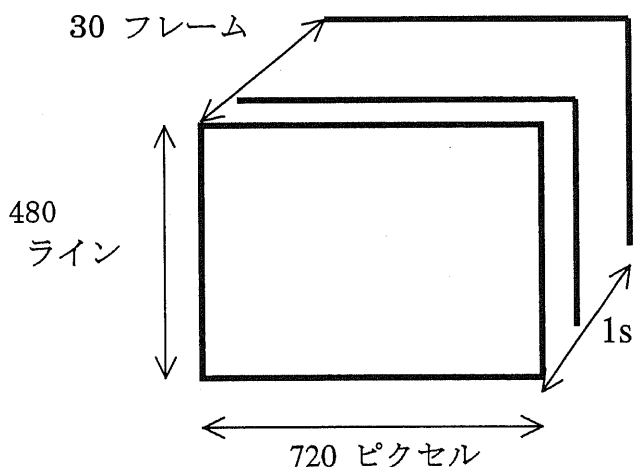


図4.2 画像構成  
Fig.4.2 Picture Construction

マクロブロック (16 ライン \* 16 ピクセル)

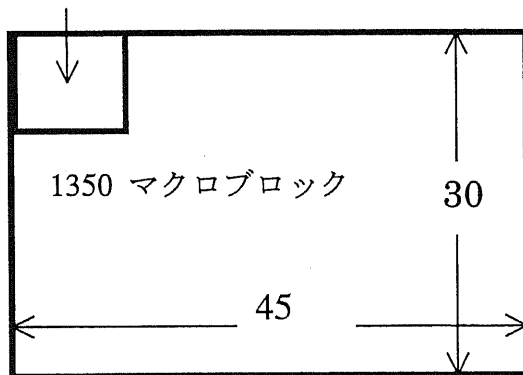


図4. 3 フレーム構成  
Fig.4.3 Frame construction

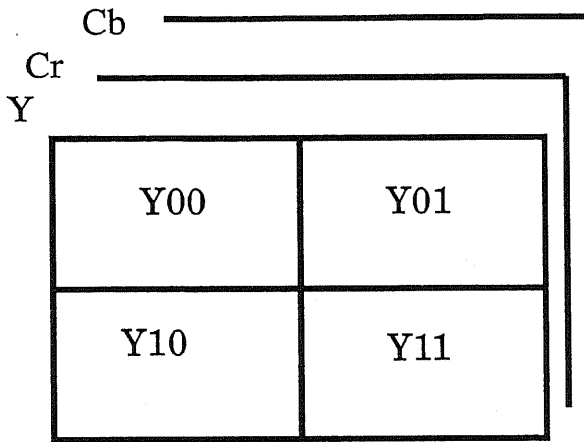


図 4. 4 マクロブロックの構成  
Fig.4.4 Macro block construction

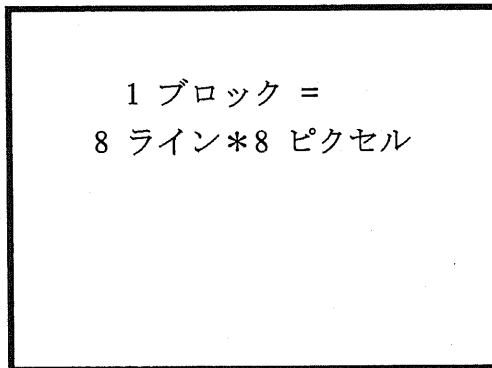


図 4. 5 ブロックの構成  
Fig.4.5 Block construction

現在、IDCT で実用化されている RISC 32 ビットマイクロプロセッサの例では並列度 2 で 100MIPS の処理能力だから 1 MOPS を 4 MIPS に相当すると仮定すると 10 倍の積和演算に相当する処理能力のシステムが求められている。従って並列命令や並列演算の応用による能力向上が必要である [96、105]。

4. 2. 2 IDCT 処理のソフトウェア

(1) IDCT 処理のアルゴリズム

1 ブロックの IDCT は、8 画素の IDCT を 64 回づつの積と和で実行し、これを 16 回繰返して実行する手法が使われている [93、99]。図 4. 6 に演算回数を減らした Chen の方法と係数を示した。

$$C^j_i = \cos(i\pi / j) \dots \dots \dots (1)$$

$$S^j_i = \sin(i\pi / j) \dots \dots \dots (2)$$

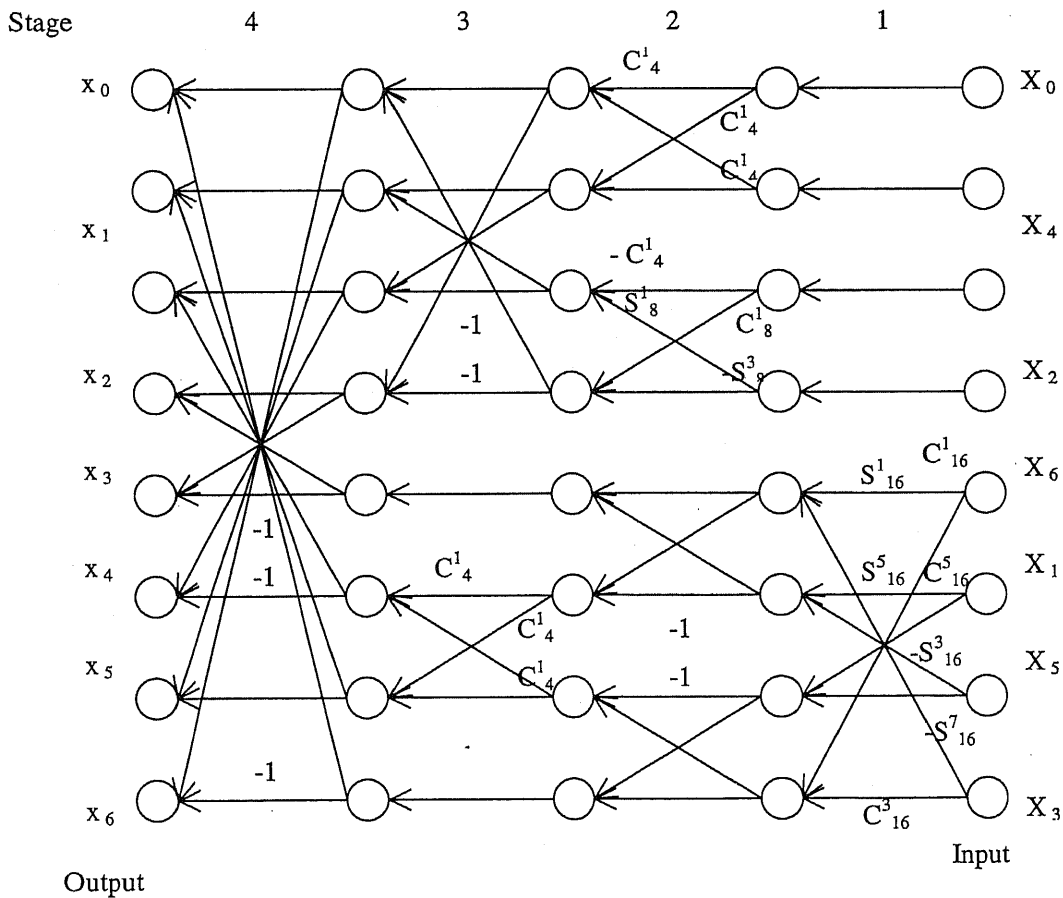


図 4. 6 IDCT の Chen アルゴリズム  
 Fig.4.6 Fast IDCT algorithm of Chen

基本方法では64回の積と64回の和、計128回の演算を必要とする。

Chenの方法は、積を52回、和を26回、計78回で可能であり、基本手法の約61%の演算回数になる。

(2) 演算回数の低減

図4.6のステージをE、ラインをF、Gとすると個々の4積和処理は $M_{E,FG}$ で表すことができる。このMによってIDCTのアルゴリズムを表す表4.1に示すようになる。すなわち13回の演算でIDCTが可能になる。

表4.1 4積和によるIDCTのアルゴリズム

Table 4.1 IDCT algorithm using 4 multiply-add operation

Step	Input	$M_{E,FG}$	Output
1	$X_0, X_4$	$M_{1,04}$	
2	$X_2, X_6$	$M_{1,26}$	
3		$M_{2,06}$	
4		$M_{2,42}$	
5	$X_1, X_7$	$M_{0,17}$	
6	$X_5, X_3$	$M_{0,53}$	
7		$M_{1,15}$	
8		$M_{1,37}$	
9		$M_{2,53}$	
10		$M_{3,07}$	$X_0, X_7$
11		$M_{3,43}$	$X_1, X_6$
12		$M_{3,25}$	$X_3, X_5$
13		$M_{3,61}$	$X_3, X_4$

### 4. 3 第2 ALU による信号処理

第2 ALU を持つ RISC 32ビットマイクロプロセッサのアーキテクチャを提案した。またこのマイクロプロセッサを使った IDCT のデータ転送を評価したところ、演算の1/3の処理量でかつ演算と並列に処理できることが判った [9]。

典型的 MPEG-2 デコーダでは、250M 並列度4のメディアプロセッサを使った場合、MPEG-2 を193MIPS で処理している。内121.5MIPS が IDCT である。また MPEG-2 の CPU 時間専有率は77%である [59、92]。

そしてこの提案では、クロックが250MHz の RISC マイクロプロセッサにおける MPEG-2 デコードの第2 ALU の時間専有率は65%となり、また内蔵バスの占有時間は19.7%となり、DRAM 混載 MPEG-2 デコード用 SOC が構成できる見込みである。

本提案は、DTV や DVD の SOC に適すると考えられる。

#### 4. 3. 1 IDCT の高速処理アーキテクチャ

複数の ALU を持つ RISC 32ビットマイクロプロセッサで IDCT をソフトウェアで処理することを提案する。

##### (1) IDCT のソフトウェアでの実行システム

IDCT をソフトウェアで実行する RISC 32ビットマイクロプロセッサを図4.7に示す。これは本来の ALU 以外に2つめの ALU をもつマイクロプロセッサである。2つめの ALU をここでは第2 ALU (2nd ALU) と称することにする。

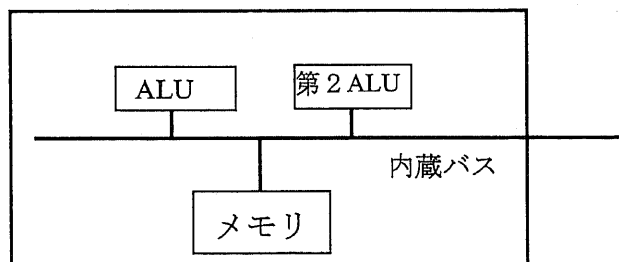


図4. 7 複数の ALU を持つ RISC 32ビットマイクロプロセッサ  
Fig.4.7 32-bit RISC processor with multi ALU



(2) 第2 ALU

第2 ALU は、高速にデータ処理を行うことに適した構造を持つ。図4.8に示すように1つ又は1つ以上のバッファメモリを持ち、それらは IDCT の処理前と処理後のデータを格納する。そして2つの積和器とバススイッチを制御する制御テーブルと制御レジスタを持つ。

バッファメモリはマイクロプロセッサの内部バス及び積和器に、バススイッチによりそれぞれ独立に接続できる。2つ以上のバッファメモリがある場合、第2 ALU はバッファメモリのデータの交換処理から独立に駆動させることが可能である。

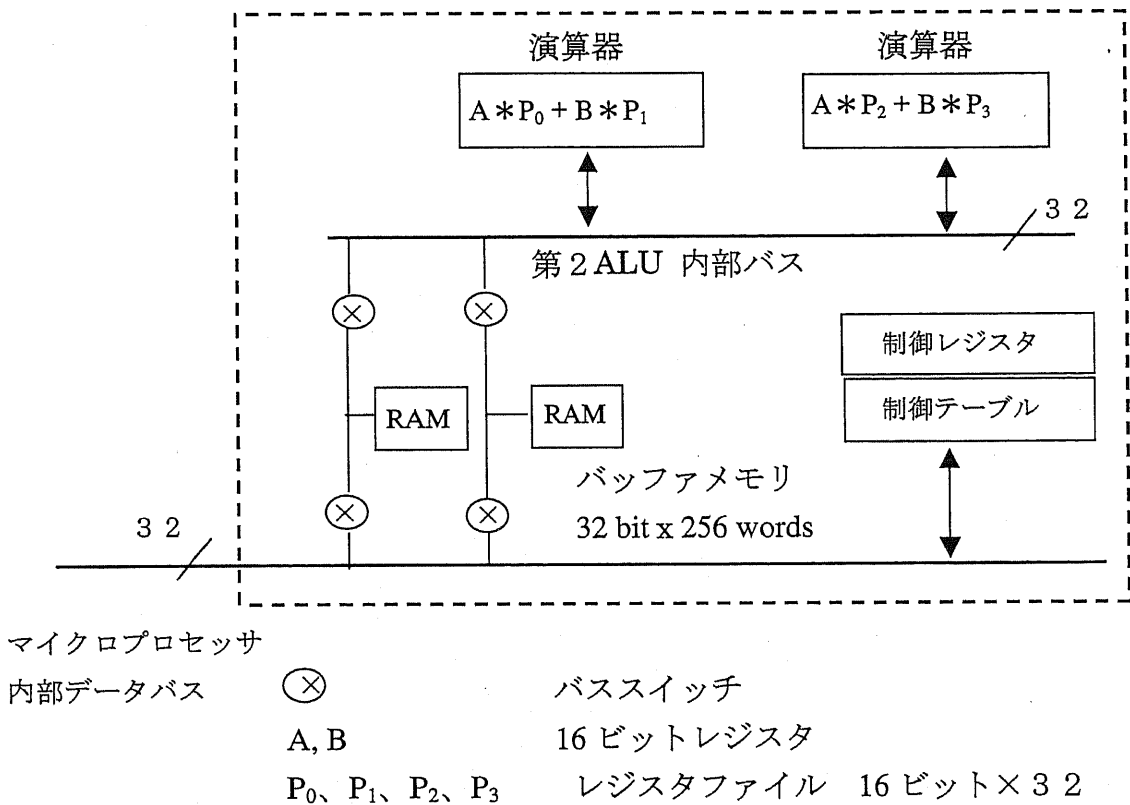


図4.8 第2 ALU の構造

Fig.4.8 Structure of 2nd ALU

#### 4. 3. 2 IDCT 処理アルゴリズム

本提案のシステムで、IDCT 処理は 101MIPS である。

##### (1) 制御テーブル

制御テーブルは図 4. 9 に示すように積和の A と B の値をバッファメモリより読み出すための A アドレスと B アドレスを 32 持つ。

A address (8-bit)	B address (8-bit)
----------------------	----------------------

図 4. 9 制御テーブルの構造  
Fig.4.9 Structure of Control Table

##### (2) 制御レジスタ

制御レジスタは、シーケンス処理の実行、バススイッチの制御を行う。  
制御レジスタの各種の機能を示す。

制御テーブルに基づく順次演算の実行指示	1 ビット
実行モニタ (実行中/終了)	1 ビット
バススイッチ制御	4 ビット
シーケンスレジスタ スタート エンド 繰り返し	3 バイト

##### (3) 第 2 ALU の実行

第 2 ALU の実行において、次の処理が必要。

2 つの積和器のレジスタファイルへ ( $P_0$ 、 $P_1$ 、 $P_2$ 、 $P_3$ )

1 3 ステップのデータ書き込み (i)

制御テーブルへのデータ書き込み (ii)

バッファメモリのデータ読み出しと書き込み (iii)

(i) (ii) (iii) が行えるようにするための第 2 ALU 内の

バススイッチの制御 (iv)

IDCT 処理の場合、(i)と(ii)は一度データを設定すると変更の必要はない。但し、(iii)(iv) は毎回読出しと書込みが必要となる。第2 ALU は、次の処理を行う。

$$A * P_0 + B * P_1, \quad A * P_2 + B * P_3$$

#### 4. 3. 3 エバリュエーション

リアルタイムの映像処理であるから、250MHz クロックバス速度の場合の IDCT 処理とデータ転送の時間分布を評価する。これらを図 4. 10 と図 4. 11 に示す。

##### (1) データ転送 (プロセッサ RAM とバッファメモリ間)

バッファメモリは6ブロック単位で IDCT 処理後のデータ読み出しとデータ設定で DMA を使い 3.1 (3.072)  $\mu$  sec 必要である。

1秒間に40500回実行され、プロセッサバスを合計124msec専有することになる。

##### (2) IDCT

1マクロブロックの処理時間は10 (9.984)  $\mu$  sec で1秒間に合計時間404msecを第2ALUが駆動される。

##### (3) IDCT の処理時間

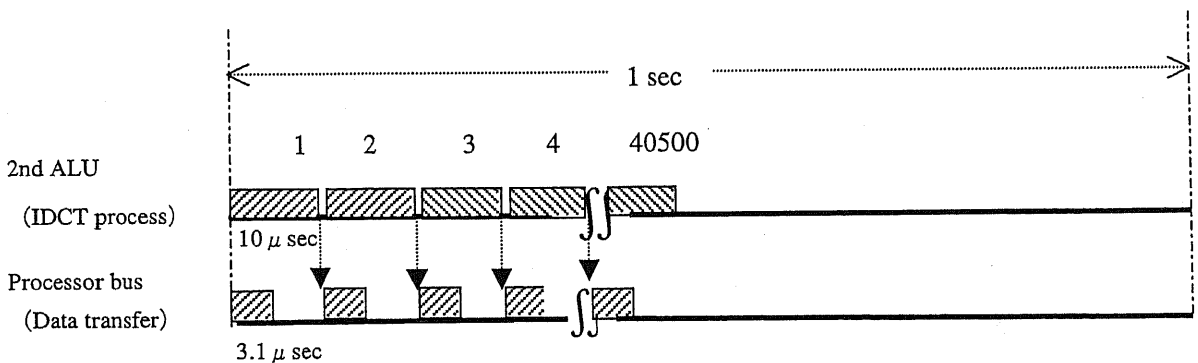


図 4. 10 IDCT のデータ送信

Fig. 4.10 Data transfer of IDCT

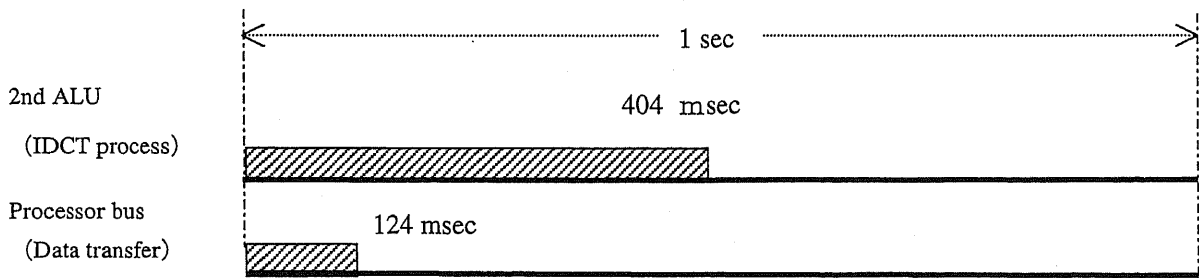


図 4. 1 1 1 秒間にかかる処理時間  
 Fig. 4.11 Total process time for 1 second

#### (4) MPEG-2 デコーダ用 SOC

図に示す提案のプロセッサを使うと IDCT が可能である。つぎに図 4. 1 に示す MPEG-2 デコーダの処理の可能性を評価する。

250M のクロックで提案のアルゴリズムによる MPEG-2 のデコードを想定すると、図 4. 1 1 に示したようにこのアーキテクチャでは、ALU が 876msec の余力と第 2 ALU が 596msec の余力を持つことが解る。一方、MPEG-2 の IDCT 以外の処理は 293MOPS 必要で、MPEG-2 実行において、IDCT のときと同じ演算効率で処理できるとすると第 2 ALU の内蔵バスの時間専有率は 64.2% となり、内蔵バスの占有時間は 19.7% となる [8、92]。この場合図 4. 1 でも明らかなように、VLD と MA、DRAM の回路は外付けとなる。これを内蔵した場合の検討は 4. 4 章に示す。

以上、DRAM 内蔵 RISC で MPEG-2 用 SOC の可能性が示された [45、93、98、107、116、119]。加えて、本提案は DTV や DVD の SOC に適すると考えられる。

## 4. 4 マイクロプロセッサ構造（アーキテクチャ）の改善による高速化検討

第2 ALU の付加で 250MIPS の RISC 3 2 ビットマイクロプロセッサで MPEG-2 デコードの IDCT をソフトウェア処理での実現を前項で確認した。それにより RISC 3 2 ビットマイクロプロセッサで MPEG-2 デコードをソフトウェアでの処理で実現の可能性が一步進んだと考えている。これを目指し RISC 3 2 ビットマイクロプロセッサの性能向上をさらに検討する。

ここでは第2 ALU の性能改善でさらに IDCT 処理の高速化の検討を行う。また MPEG-2 において IDCT 以外の 37% に当る逆量子化 (IQ)、画像作成 (RC)、動き補償 (MC)、インバーススキャン (IS) などの機能をソフトウェア処理で実現できるように、マイクロプロセッサ全体の性能向上を計る検討も行う。これらは、MPEG-2 のソフトウェア処理に限らず、他のデータのデコード処理及び高速のデータセンシングと認識処理にも展開できる研究と考えている。

### 4. 4. 1 IDCT 処理の高速化

第2 ALU を付加することで、250MIPS の RISC 3 2 ビットマイクロプロセッサで、IDCT 処理をソフトウェアで実現の可能性のあることを認識した。IDCT 処理で内蔵バスは 12.4%、また第2 ALU は 40.4% 専有されている。内蔵バスの専有に関しては IDCT 処理の前後の処理を同一メモリ上で処理する手法をバススイッチの技術と合わせて別途検討する。そこでここでは 40.4% 専有されている第2 ALU の高速化を検討し、IDCT 処理以外の処理をひきつづいて同一メモリで実行する方法を別途検討していきたい。

#### (1) 高速アルゴリズム

IDCT の処理時間短縮のため第2 ALU の 4 積和回路を増強する。前出した図 4.6 の IDCT の Chen アルゴリズムより第2 ALU の 4 積和演算器によるさらに高速を実現できるアルゴリズムを表 4.2 に得た。すなわち 4 積和演算器の数を増加させることにより IDCT の処理時間は短縮される。4 積和が 1 つの場合は 13 ステップ、2 組で 7 ステップ、3 組で 5 ステップ、4 組で 4 ステップまで短縮が可能で、Chen アルゴリズムでは 4 ステップが最短ステップになると考えている。そ

れで第2 ALUに4積和演算器を4組設定することにより、IDCTの第2 ALUの専有時間は図4. 1 2に示すように1秒間当たり404msecから約31% ( $4/13 \approx 0.307$ )の125msecに短縮できる。

また第2 ALUの構成は、図4. 1 3に示すように第2 ALUの内部バスとデータメモリは4積和の演算器1つに比べそのビット巾を4倍にする必要がある。

表4. 2 4積和による高速IDCTアルゴリズム  
Table 4.2 Fast IDCT algorithm using four-multiply-add operation

	4積和演算器の組数				Input	$M_{E,FG}$	Output
	4	3	2	1			
ステップ	1	1	1	1	$X_0, X_4$	$M_{1,04}$	
	1	2	1	2	$X_2, X_6$	$M_{1,26}$	
	2	3	2	3		$M_{2,06}$	
	2	3	2	4		$M_{2,42}$	
	1	1	3	5	$X_1, X_7$	$M_{0,17}$	
	1	1	3	6	$X_5, X_3$	$M_{0,53}$	
	2	2	4	7		$M_{1,15}$	
	2	2	4	8		$M_{1,37}$	
	3	3	5	9		$M_{2,53}$	
	4	4	6	10		$M_{3,07}$	$X_0, X_7$
	4	4	6	11		$M_{3,43}$	$X_1, X_6$
	4	4	7	12		$M_{3,25}$	$X_3, X_5$
	4	5	7	13		$M_{3,61}$	$X_3, X_4$
全ステップ数	4	5	7	13			

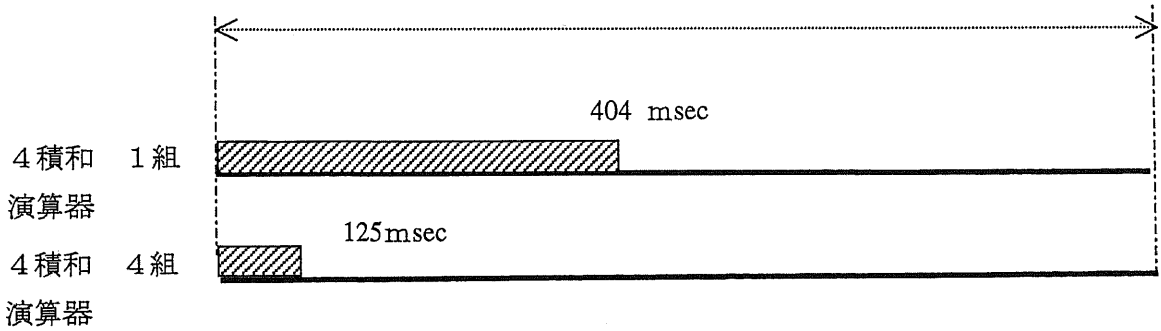


図4.12 第2 ALU の高速化  
Fig.4.12 Fast 2nd ALU

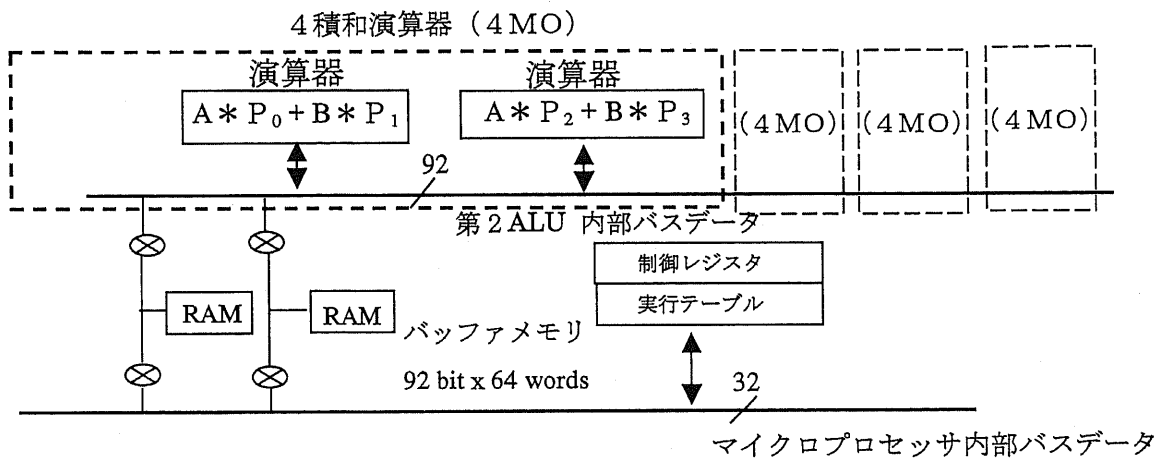


図4.13 IDCT用に改良された第2 ALU の構成  
Fig.4.13 Structure of 2nd ALU improved for IDCT processing

#### 4.4.2 第2 ALU の性能向上

4.4.1項では、IDCT処理専用の第2 ALUの高速化を検討したが、ここではよりいくつもの用途に合う第2 ALUの高速化については検討をすすめる。

##### (1) 演算シーケンスのプログラム化による性能向上

第2 ALUの演算シーケンスのプログラムによる性能向上について述べる。演算シーケンスのプログラム化は、図4.14に示す実行テーブルと図4.16制御

レジスタにより制御される。

実行テーブルは、1ステップ4バイトで256ステップの実行テーブルを構成しており制御レジスタによりこれらのステップがくり返し使われまた任意のステップを選択可能にしている。実行テーブルの一行は、Aアドレス(8ビット)、Bアドレス(8ビット)、Cアドレス(8ビット)と演算子指定(6ビット)とデータ指定(2ビット)よりなる。Aアドレス、Bアドレス、Cアドレスは、バッファメモ地の特定の番地を示し、それらをAメモリ、Bメモリ、Cメモリと称し、それらのアドレスを指定する。Aアドレス、Bアドレス、Cアドレスとも8ビットで256ワードの空間を示す。2ビットのデータ指定は、各ビットがAアドレスとBアドレスに対応しており、1の場合1つ前の演算結果のデータを使うことを意味し、高速化演算の1つの工夫である。

6ビットの演算子指定は  $a \text{ op } b = c$  の  $\text{op}$  の演算子を与えるものである。

図4. 15に演算子の一覧を示す。演算部にハードウェアの用意できる演算子を定義することが当然のことながら可能である。また拡張の演算として

$$a * b + c * d = e, \quad f * g + h * i = J$$

なども実行可能である。例えばAアドレスで  $a, b, c, d$  のデータアドレスを指定し同時にBアドレスで  $f, g, h, i$  のデータアドレスを指定することや別にアドレスレジスタを増設するなど方法がある。

制御レジスタの構造を図4. 16に示す。バススイッチの制御を持ちマイクロプロセッサからデータテーブル、制御レジスタにデータが書きこまれそのデータにより第2ALUが演算を実行する。演算テーブルはスタートアドレス、終了アドレスとくり返し回数を持ちスタートアドレスと終了アドレスの間をくり返し回数演算実行した後、次の演算テーブルを同じく実行することになり、演算テーブルを全て終了する。制御レジスタによる演算実行を図4. 17に示す。



ステップ	Aアドレス	Bアドレス	Cアドレス	データ指定 ↓	演算子指定
0	8	8	8	2	6
1					
⋮	⋮	⋮	⋮	⋮	⋮
255					

図4. 14 第2ALUの実行テーブル

Fig.4.14 2ndALU control table

- $a + b = c$
- $a \times b = c$
- $a, b$  シュト =  $c$
- $a - b > 0$  なら  $a, b$  のアドレス入れ替え
- $a - b = c$
- $a \div b = c$

(a) 演算子

- $a * b + c * d = e,$
- $(a + b + c + d) 1/4 = e$
- $f * g + h * i = j$
- $(f + g + h + i) 1/4 = j$

(b) 拡張演算子

図4. 15 第2ALU演算子一覧

Fig.4.15 2ndALU Operator

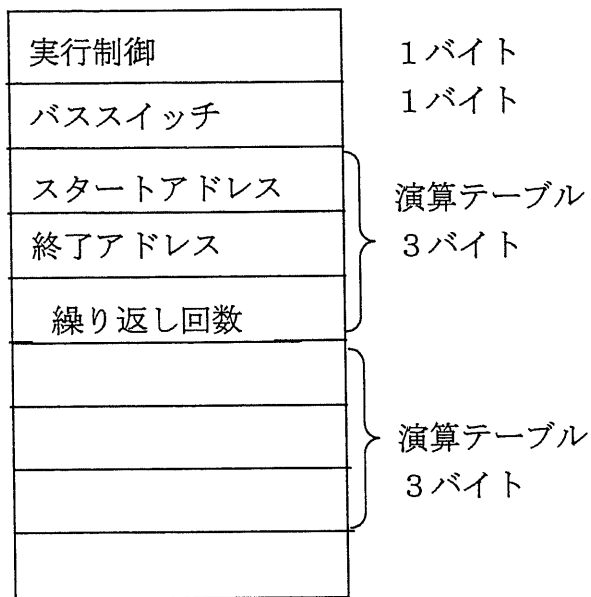


図4. 16 第2ALUの制御レジスタ  
Fig.4.16 2ndALU control register

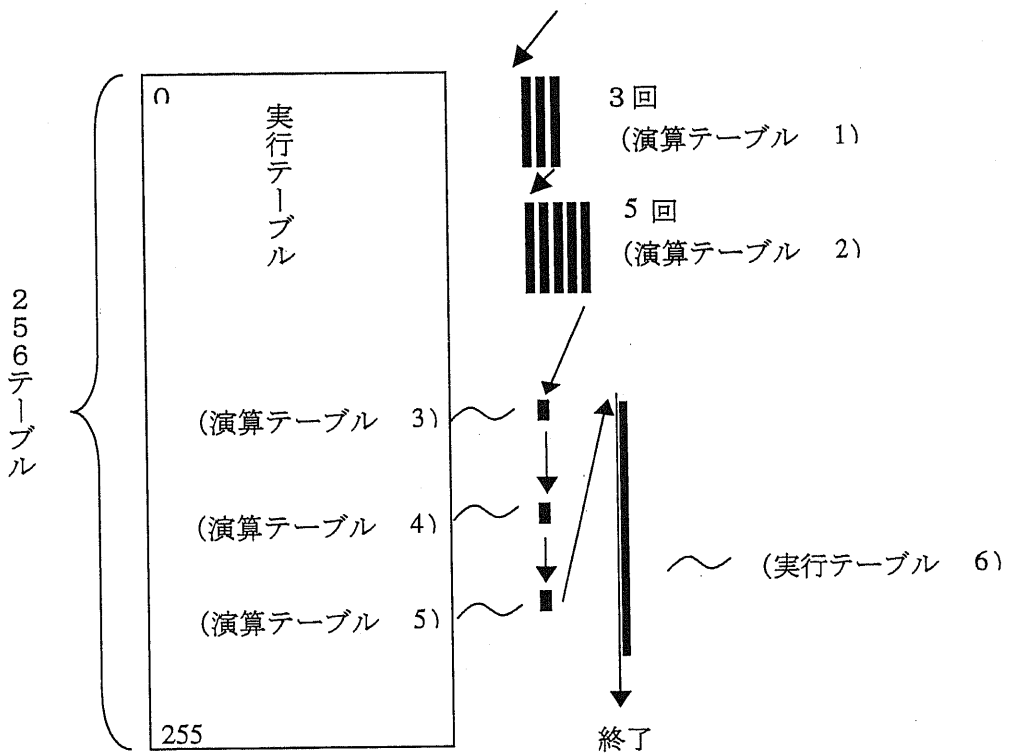


図4. 17 実行テーブルによる演算の実行  
Fig.4.17 Operation performance by imperative table

## (2) データメモリの多重化

第2 ALU のデータメモリの多重化を図4. 18に示す。演算部と1つのバススイッチにより選ばれたデータメモリが演算を実行している間に残されたデータメモリは、次に演算すべきデータの取り入れ、演算終了のデータの送出手続きを行う。本体プロセッサの直接実行と処理の並列化が可能であり、処理の高速化に寄与することになる。またこれらの高速化で処理全体に時間としての余裕が生じ、処理の手順を単純化する事が可能になり、システム構成簡略化と開発期間の短縮を可能にする。

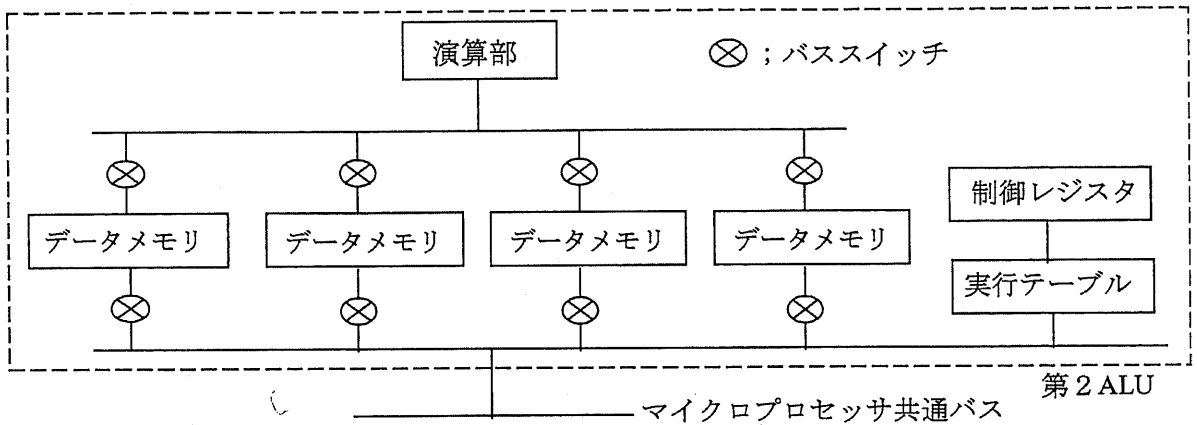


図4. 18 高速化のためのメモリの多重化

Fig. 4.18 Multi memory for high-speed

## (3) 第2 ALU のデータバスの専用化

第2 ALU 内のデータバスの専用化は第2 ALU のデータ処理速度の向上に寄与する。図4. 19に専用バス方式と共通バス方式とその中間的な半専用バス方式の構成を示す。例えば32ビットの  $a \times b = c$  の演算をさせる場合共通バス方式ではそのバス巾は32ビットになり、また半専用バス方式では64ビットに、専用バス方式では128ビットになる。このときの処理時間は図4. 20に示すように共通バスでは4T時間、半専用バスでは2T時間、完全な専用バスでは1T時間となる。Tはマイクロプロセッサの最低処理時間を示す。例えば250MHzのマイクロプロセッサの場合、4nsecになる。結局性能とバス巾がトレードオフの関係にあり図4. 21に示すように処理時間を1/4にしようとするるとバス巾は4倍必要になる。これらの選択は必要性能をどのように実現して行くかと言う選

択の中で、その時点におけるプロセスの微細化の程度に一番関係する。また1つの第2 ALU に対し前述した第2 ALU のデータバスの多重化及び第2 ALU をいくつか内蔵させるかということにも関係する。

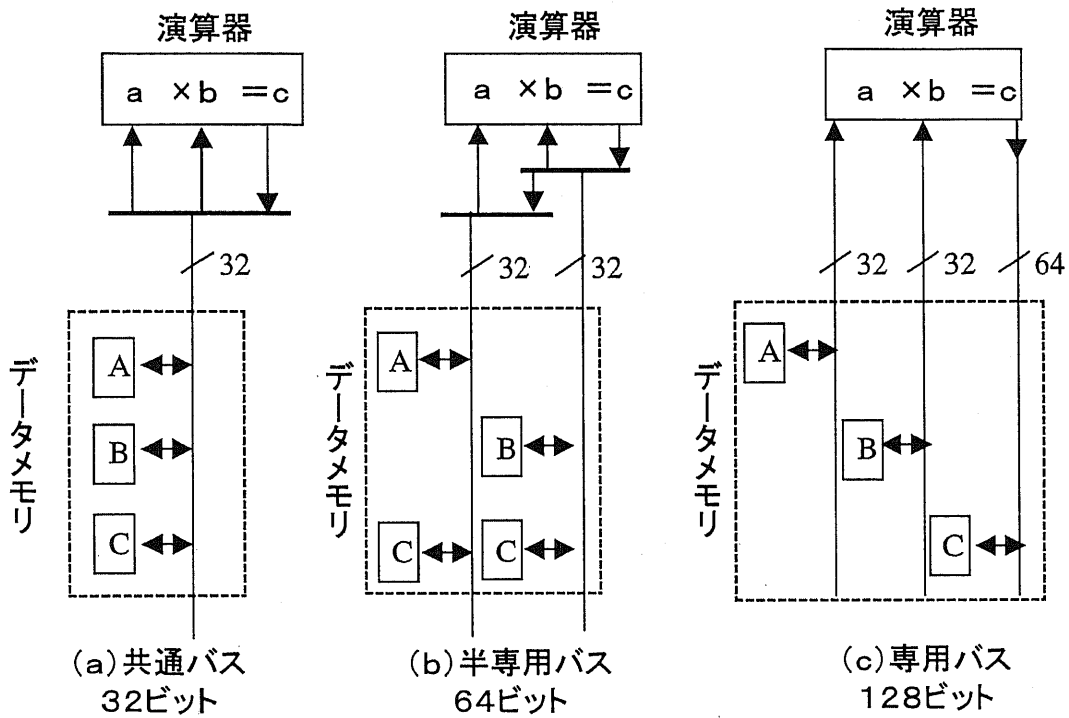


図 4. 19 データメモリと演算器間のデータ転送とバス方式

Fig.4.19 Data memory, data transfer in the operation and bus system

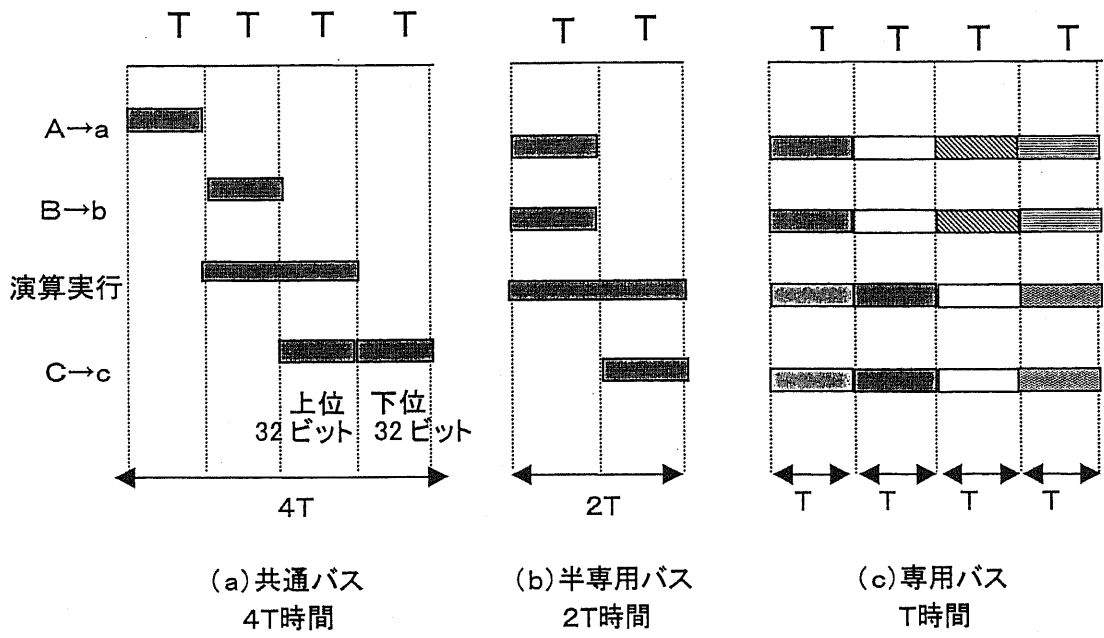


図 4.20 データ転送バス方式と処理時間

Fig.4.20 Data transfer bus system and processing time

バス方式	共通バス (a)	半専用バス (b)	専用バス (c)	c / a
バス巾 (ビット)	32	64	128	4
処理時間 (200MHZ の場合)	4T (20nsec)	2T (10nsec)	T (5nsec)	0.25 (1 / 4)

図 4. 21 データバス方式とデータバス巾と処理時間の関係

Fig.4.21 Relationship of between data bus system, data bus width and processing time

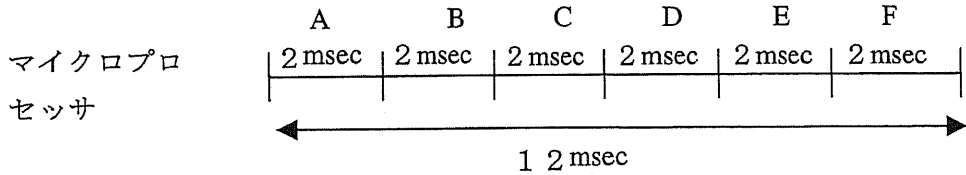
#### (4) 第2 ALUによるマイクロプロセッサの高速化

第2 ALUによる性能向上は3つあると考えている。1つは並列処理によるもの2つめは、第2 ALUは本体マイクロプロセッサに比べ1/10以下の小さな回路規模になるため、内蔵バスの静電容量は小さくなり、また配線が短くなるためスキューが小さくなる。これらによる高速化は単純に考えて10倍の高速化が望める。3つめは、現在の半導体の微細化技術でこのような比較的大規模な第2 ALUを複数個内蔵出来ることが可能となったことである。このことにより性能向上のため、また個別機能実現のため増設されるハードウェアでなくて、より多くの広い機能をソフトウェア制御のもとに実現する事ができるし、またすべきである。これが新しいマイクロプロセッサの進む方向と考えるもので、第2 ALUの提案につながるものである。

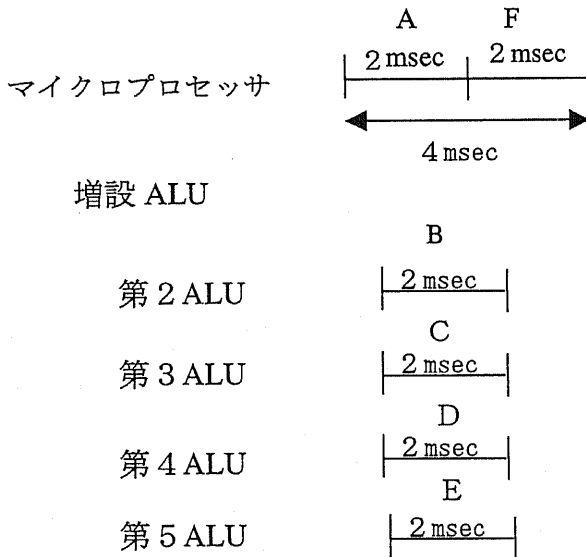
第2 ALUによる高速化の概念は、図4. 22で処理B, C, D, Eの各々が並列処理実行可能であること、またそのように設計する事が重要である。ここに示すように処理AとFに対して各処理B, C, D, Eが独立であるように、第2 ALUを有効に働かせる条件の1つは、各々分割された処理においてお互い順序を伴わない独立の処理に分割できることである。このような処理方式に適合するものは、くり返し処理が多く、例えば $10^3 \sim 10^9$ 回/秒以上の処理がいくつかに分割できるものである。このような条件に合うものとしては現在ハードウェアロジックで実現している画像処理、音声処理、認識処理やセキュリティ処理があると考えている[57、58、59、60、61]。

ハードウェア処理の部分を第2 ALUで処理させる場合、図4. 23に処理をモデル化して検討した。ハードウェア処理は一般に(A)に示すようにh1、h2、h3、h4といくつか処理過程に分れており、データはそれらの処理過程を通るごとに処理実行され自然に次の処理過程に進むようにみえる。ソフトウェアでの処理は何も次の過程に進めることは必要でなく、次のソフトウェア処理を同じプロセッサで実行することは容易である場合が多い。また処理が多く処理が間に合わず第2 ALUを複数使う必要があるとき、ハードウェアの処理の場合は(a)のように直列にし、また第2 ALUすなわちソフトウェア処理では(b)のように並列するのが有効と考える。またハードウェアを第2 ALUに置換える場合、並列処理をさせることが望ましく、これはデータの転送をより少なくすることがソフト

ウェア処理では有利であるためである。



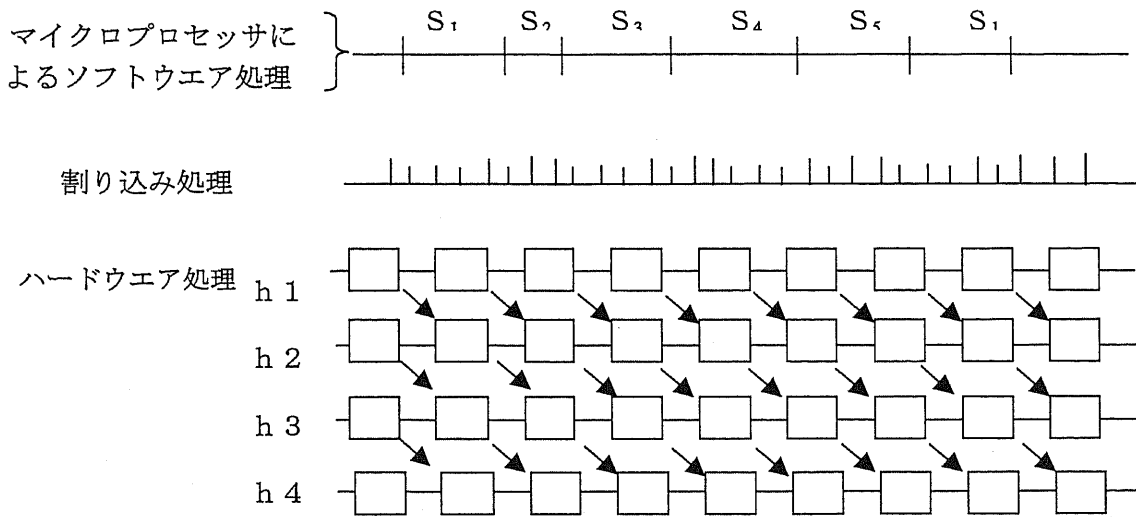
(a); A, B, C, D, E, Fを順次処理していくとこの場合12 msecの処理時間を要する。



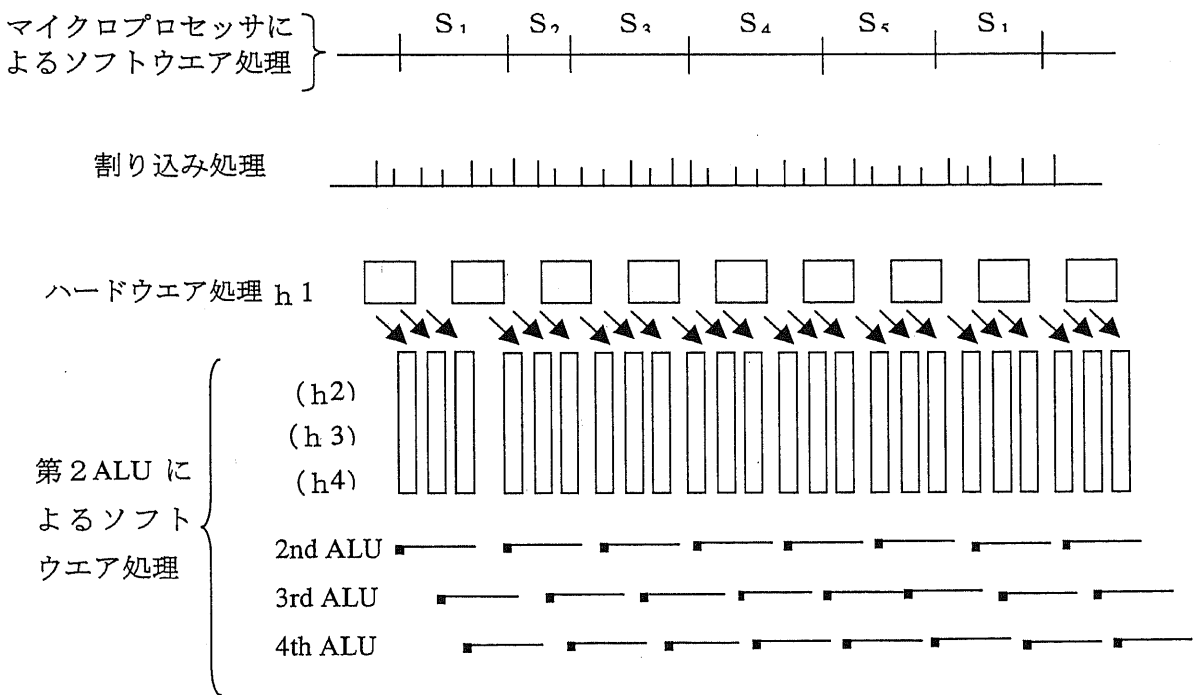
(b); 4つの増設ALUによる並列処理で高速化の実現、12 msec 必要のところ4 msec で処理可能。

図4. 22 第2ALUによる高速化

Fig.4.22 Speeding up by 2ndALU



(a) 現行のソフトウェアとハードウェアによる処理をモデル化したもの



(b) (a) の処理のハードウェア処理部分を一切 2nd ALU で処理する

図4. 23 第2ALUで処理した場合の考察

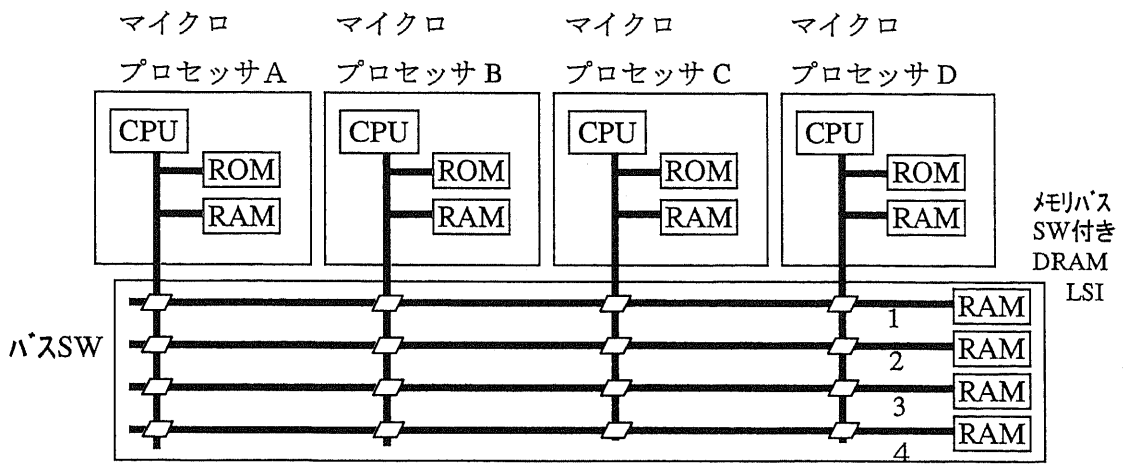
Fig.4.23 Consideration as in processing 2ndALU



#### 4. 4. 3 バススイッチ結合メモリ・マルチプロセッサ構造

複数のマイクロプロセッサを持つシステムにおいて、1つのマイクロプロセッサで処理したデータを次のマイクロプロセッサに転送し、次の処理を行わせる場合、当然のことながら、データ量に応じたデータ転送時間が必要になる。例えば4.3.3項に示したIDCT処理の場合、第2ALUでの処理が1秒間当たり404msecかかるが、それを他のメモリに転送する時間は1秒間あたり124msecかかり、転送にかかる時間はデータ処理時間の約31%にも達する。これはデータをマイクロプロセッサのソフトウェアで処理する場合、自然に起こることであるが、システムとして改善する項目であり、DMA機能もこれに対応するものであるが、十分でない場合もある。また、データ処理を第2ALUで処理するなど、本体マイクロプロセッサの実行時間を極力小さくしようとしても転送にこれだけかかるのは問題であり、ハードウェアが増加してもこの問題の解決は望まれる。

これに対して、複数の独立したマイクロプロセッサと同じく複数の独立したメモリがバススイッチによりお互い結合するバススイッチ結合によるマルチプロセッサと複数のメモリを持つシステム構造（アーキテクチャ）を新たに検討していきたい。これをバススイッチ結合マルチメモリマルチプロセッサ構造（Architecture of multi-microprocessor with multi-memory connected by bus switch）と名付ける。図4.24に示すものは、これの1つのシステム例で4つの独立したマイクロプロセッサと4つの独立したメモリにより成り、16のバススイッチにより各々のマイクロプロセッサとメモリが結合するシステムである。システムとしてはどのようにバススイッチがオンして結合してもよい。CPU AがRAM 1とCPU BがRAM 2と結合しており、RAM 1のデータをCPU Bが処理する必要のある時、バススイッチにより切替えRAM 1をCPU Bに結合することで実現できる。これは2つのCPUで同期を取る必要はあるが切替え時間は $10^{-6}$ secオーダーである。すなわち図4.25に示すように、例えば1回で1秒間当たり $124 \cdot 10^{-6}$ secの時間かかっていたデータ転送が $10^{-6}$ secで実現できることになる。



5チップ、2チップまたは1チップ構成（1チップの場合；250pin 2~4MB x 6~8ヶ = 12~32MB

図4.24 バススイッチ結合マルチメモリ  
マルチマイクロプロセッサ構造（アーキテクチャ）

Fig.4.24 Architecture of multi-microprocessor with multi-memory connected by bus switch

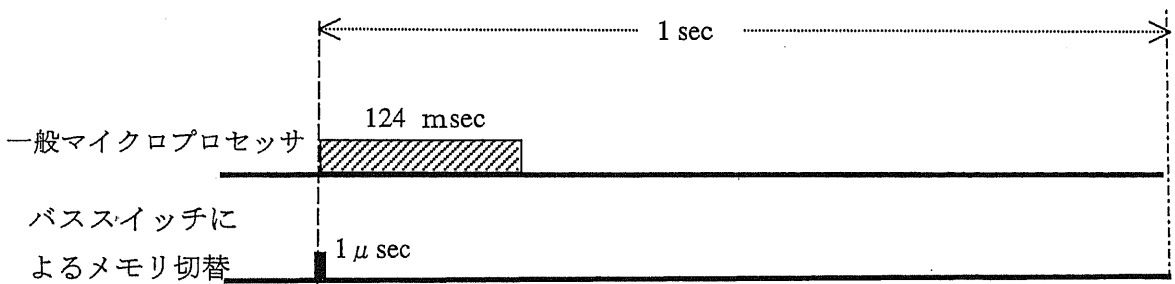


図4.25 データ転送比較

Fig. 4.25 Comparison of data transfer

#### 4. 4. 4 マイクロプロセッサによる MPEG-2 の実現検討

RISC 3 2 ビットマイクロプロセッサのソフトウェアで MPEG-2 デコード処理の検討を行う。MPEG-2 のデコードには音声と画像の処理を行うのに、VLD (可変長コード)、IQ (逆量子化)、IDCT、RC (画像形成) などいくつかのデータ処理が必要である。その中で IDCT についてソフトウェア処理での方法を検討した。この IDCT は VLD 処理をのぞき MPEG-2 デコードの中で 63% の処理時間を占める一番大きい処理である [92]。

IDCT 処理のソフトウェアでのステップ数削減を図 4. 26 に示す。1 ブロックの画像処理に加算と積の演算器を持つ 3 2 ビットマイクロプロセッサでは、1 度にできる演算を 1 ステップとした場合、128 ステップ必要であり、これを Chen のアルゴリズムを使うことにより 78 ステップまで短縮させ、さらに第 2 ALU を設置することで 13 ステップに、さらに第 2 ALU に 4 積和を 4 組持つことで 4 ステップまで短縮することができる。これにより 250M 並列度 4 のメディアプロセッサを使った場合、IDCT は 1 秒ごとに 486msec 必要とするものが、約 1/3 の 125msec まで短縮されることになる。

また IDCT 処理のデータに関して、処理後のデータと次の処理すべきデータの交換時間の短縮は、全体処理の高速化において重要である。それで第 2 ALU 及びマイクロプロセッサがデータ交換のために使う時間を極力小さくするために、前出の図 4. 18 および図 4. 24 に示すようにバススイッチで切替える方法が勝ると考える。これは 1 つの第 2 ALU またはマイクロプロセッサにおいて毎秒当り 124msec 負荷を軽減することに相当する。

MPEG-2 の高速処理における今後の課題の 1 つとして、IDCT 処理の前後の処理、すなわち IS/IQ と MC/RC を同じメモリ上で処理をさせることは、高速化において有効である。これを実現するのもバススイッチが重要な役割を果たすと考える。

現時点で、マイクロプロセッサによる MPEG-2 の実現のためのハードウェア規模を推測する。先ず第 2 ALU は  $0.18 \mu\text{m}$  のプロセスでは  $0.411\text{mm}^2$  のチップサイズであると推定した [119, 121]。これを基に MPEG-2 の IDCT 以外の処理も含め、すなわち図 4. 1 の全ての機能を含めるとする。この場合 1 チップに、250MHz の第 2 ALU を持つ RISC 3 2 ビットマイクロプロセッサ 2 組、8 M バイトの DRAM 4 組、2 K 空間のデータ変換 ROM または SRAM 4 組と 50 K ゲートの論

理回路が必要と考え、 $0.15\mu\text{m}$  プロセスを使い  $9 \times 9\text{mm}^2$  のチップサイズの LSI になると推測した。

この推測をより正確にしていくこと、及び具体的なマイクロプロセッサの構成については今後の研究の課題としたい。

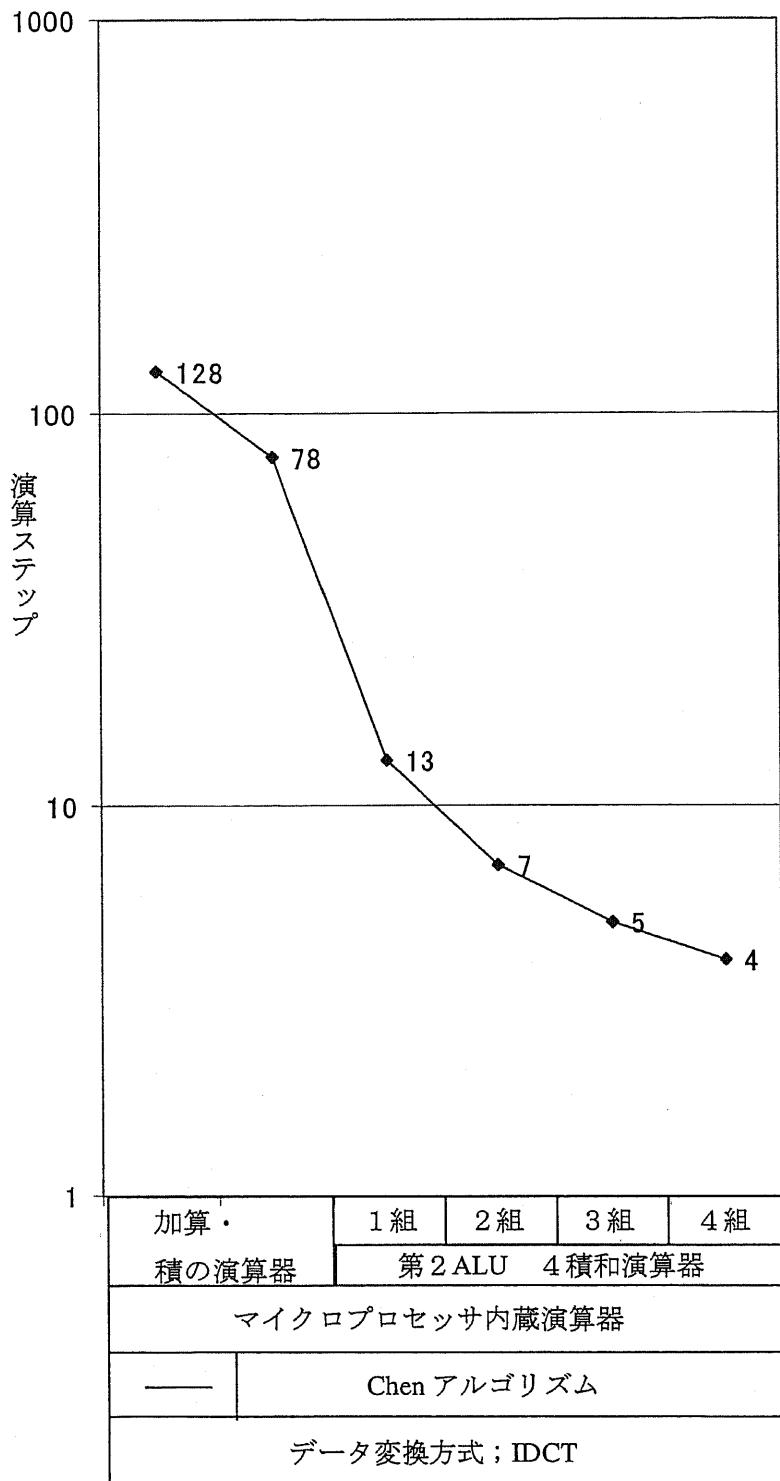


図 4. 26 IDCT 処理の各演算ステップ

Fig. 4.26 Each operation step for IDCT processing

## 4. 5 むすび

映像機器におけるマイクロプロセッサ応用に対する研究の1つとして、マイクロプロセッサによる信号処理技術の確立を、TV映像信号のMPEG-2デコードのIDCTをRISC 32ビットマイクロプロセッサのソフトウェアで処理する研究を行い実現の見通しを得た。マイクロプロセッサによる信号処理とはマイクロプロセッサが直接信号を処理することを指す。

RISC 32ビットマイクロプロセッサに第2ALUと称する演算器を追加付加することで同じ速度で実行するDSPの1/3の時間でIDCT処理を実現した。さらに第2ALUにおいて演算器の増設(図4.13)、バススイッチによるデータメモリの多重化(図4.18)、データメモリと演算器間のバスの専用化(図4.19)で第2ALUを強化すること、およびバススイッチ結合マルチメモリマルチプロセッサ構造(アーキテクチャ)(図4.24)にすることで、MPEG-2デコード処理の全てを1チップで実現する可能性を確認した。

また延長線上の新たな研究課題になるが第2ALUにおける演算とそのシーケンスをソフトウェアで設定できるようにすることにより、複雑な演算アルゴリズムも処理可能(図4.17)になり、MPEG-2のデコード処理への適応に限らず各種信号のエンコード、デコード、フィルタ、認識、これらの処理にも有効な働きをする。またバススイッチ結合マルチメモリマルチプロセッサ構造(アーキテクチャ)は、第2ALUと組み合わせ、汎用のSOC(図4.23)の目指すものであると考える。これの目的とする重要事項は半導体プロセス技術の今後の進展を背景に従来のハードウェア処理からソフトウェア処理に変革することである。これはシステム構成にも大きく考え方を変える必要が生じる。これによりシステム的大幅開発人工の削減と開発期間の短縮及びハードウェアの大幅削減の実行が今後の課題である。

## 5 結論

マイクロプロセッサが登場した 1970 年代よりマイクロプロセッサの応用機器とマイクロプロセッサはこれらがお互いに強い影響を与えながら大きく成長してきた。組み込み型マイクロプロセッサの応用機器の代表である映像機器とマイクロプロセッサ応用の進展は大きく、現在まで4つのステップで進行してきたと考えられる。

第1ステップは、マイクロプロセッサが登場した 1970 年初頭から 1979 年までの時期で、マイクロプロセッサをいかに応用するかが問題で、応用製品に付加価値をつけることが課題であった。特に4ビットマイクロプロセッサなどの1チップマイクロプロセッサが生まれ、またソフトウェア開発ツールが確立されたときでもあった。次いで第2ステップは 1980 年から 1989 年の10年間で、本格的にマイクロプロセッサの応用が拡大し、またマイクロプロセッサの需要が飛躍的に伸びた。マイクロプロセッサも1台の応用機器に複数個使われマイクロプロセッサの性能も延び、マイクロプロセッサが応用製品の本質的な改善に使われた。例えばマイクロプロセッサを使うことで、その応用製品の使う機構部分を簡略化し、またコストは安い精度の悪いモータを、ソフトウェア制御で必要な精度にまで高めて使うことで原価の低減を計ることである。

第3ステップは 1990 年より 1999 年の10年間で、前半は大規模な周辺コントローラ、例えば VTR テープ走行モータ用のサーボ制御や、HDD のディスク回転制御回路内蔵化の時期であり、また後半は CAD を酷使したマイクロプロセッサ内蔵大規模 ASIC が本格的に市場にでてきた。前半、後半通して SOC の立ち上りの時期であった。

第4ステップは 2000 年より現在まで、アナログ回路をデジタル化してデジタル LSI に取込むことや、1パッケージに複数チップをアセンブリするなど更に SOC 化が進展する。そしてさらに次の段階であるハードウェアをソフトウェアで実現する試みがはじまってきた。

こういうマイクロプロセッサとその応用製品が進展、展開するなかで本研究を進めた。第1ステップの時期においては4ビットマイクロプロセッサを開発した。これは簡明で見通しがよく、また効率のよいシステム構成（アーキテクチャ）を

研究し見出したもので、性能を高めまた開発期間と人工を改善したものである。性能向上としては ROM のコード使用量を従来の約 50% に削減した。またこれらのインサーキットエミュレータ用のエバチップはマイクロプロセッサにエバチップ機能を入れることによりエバチップ単独の開発を不要とした。8ビットマイクロプロセッサ用のモニタとしてソフトウェア開発用と実行用を開発した。ミニコンピュータに比べ、マイクロプロセッサの特徴とその実用に合わせた専用の特殊機能付実行用モニタを開発した。マスク ROM モニタの開発ではマスク ROM であっても、そのプログラムの機能拡張を可能にする仕組を付加したものや、既存の入出力機器のビジィ信号をうまく活用したタイムシェアリングモニタを開発し実用とした。またマイクロプロセッサのアドレッシング機能を向上させ、マスク ROM であるが任意のアドレス空間に設定可能にする提示も行った。

第2ステップにおいてはまだまだマイクロプロセッサの機能が低く、多くのハードウェアを持つことができない時代において、タイマ1本の割込みを周期的にかけることで、同時に処理をする必要のある表示機能とリモコン等の非同期入力パルスにおいても表示がチラツキを起さず、またリモコンのパルスを読みのがしたりしないソフトウェアシステムを開発した。

第3ステップにおいては、大規模な制御機能のマイクロプロセッサへの取り込みである。VTR ソフトウェアサーボ回路や、テレテキスト回路を取り込むのに成功した。第3ステップの後半は、VTR の交流バイアス音声記録方式のシステム LSI 化に有効な外付部品削減とアタックタイムの性能改善を、マイクロプロセッサのシリアルバスとレジスタによる制御で実現した。帰還ゲインをマイクロプロセッサで切換えて、時定数を変える手法により10組の時定数素子を2組に低減して、SOC 化を可能にする見込みを得た。

また VTR の映像信号をメモリに一時的に保持することにより、いくつもの機能、多画面やピクチャインピクチャ、及びスピードサーチやスローの画質改善をはたす VTR のデジタル特殊効果の開発においてマイクロプロセッサと論理回路の2階層制御系システム技術を習得し、開発期間短縮と開発人工の削減に寄与した。

第4ステップとしては、画像処理をはじめ今まではハードウェアで処理するのが当然に考えられていたものを、ソフトウェアで処理することを研究し、その実現のため目的に適合するマイクロプロセッサの構造（アーキテクチャ）を改善し



て大巾機能向上を検討し提案した。それは複数の増設 ALU (2nd ALU) を持ち、バススイッチと独立する複数のメモリを持つ構造 (アーキテクチャ) のマイクロプロセッサである。

本研究は製品開発を軸に展開してきたものである。4ビットマイクロプロセッサ開発時、私自身の課題は先ず開発をスムーズに成功に導くことであった。いかに誤りの少なくまた何が誤りかをよく分かるようにとの考えが、設計全般をより見やすく見通しのいいシステム構成に導いた。それが少し経済的に高くつくかもしれないという心配を、半導体プロセスが向上するなかで吸収すると考え進めた。ところが、新たに考案したシステムすなわち見通しのいいシステムは、それ由に論理数も冗長部分も極端に少なくまた歩留も高い結果をもたらした。これらによりマイクロプロセッサの開発人工の大幅削減を実現することになった。また4ビットマイクロプロセッサの性能向上は、あまり当時性能向上のことは考えられていなかったので、多くの性能向上策を見出し開発に取込むことができた。

32ビットのマイクロプロセッサの性能向上は、映像信号を32ビットマイクロプロセッサのソフトウェア処理で実現しようとしたときの高速化の考えよりスタートしている。32ビットのマイクロプロセッサの性能向上は第2ALUとバススイッチがキーポイントと考えているが、これらは4ビットマイクロプロセッサの性能向上と開発効率向上のために、主な基本構成 (アーキテクチャ) として考え採用した第2ALUと3組のデータバスに付随するバススイッチと偶然にも同じ名称でありまた機能の基本的な考えは同じである。4ビットマイクロプロセッサにおける性能向上と開発効率向上の考えは、32ビットにおいても有効であることが本論文をまとめるにおいて気づいた次第である。4ビットマイクロプロセッサの主に開発効率向上策である第2ALUとバススイッチは、32ビットマイクロプロセッサでは有効な性能向上策になっていることは注目すべきことである。これは研究、開発における方法論としての研究課題になると考えられる。

この第2ALUとバススイッチは今後マイクロプロセッサの展開する1つの方

向である。このようにマイクロプロセッサの性能向上策も映像機器などのマイクロプロセッサ適応機器のニーズを見て技術開発と研究を行うべきことが重要である事が分かった。今回の研究課題の1つである第2 ALU とバススイッチをもつ32ビットマイクロプロセッサは、次の世代にも通用する構造(アーキテクチャ)と考えられる。さらに技術開発と研究を重ね映像機器により適合させると共に、マイクロプロセッサの新たな構造(アーキテクチャ)を目指したい。そして、より多くの電子機器に有効なマイクロプロセッサ開発を実現する研究開発の端を開くことができた。

これらの基本的考えは、ハードウェアの処理からソフトウェアの処理に移すことである。ソフトウェア処理が開発しようとするシステムの開発期間短縮と必要人工を削減し、ハードウェアの大幅削減にも寄与する。これは半導体プロセス技術向上により、1チップにメモリと周辺機器とマイクロプロセッサを十分に内蔵できる状況にあり、第2 ALU と分割されたメモリとバススイッチによるこれらの並列処理構造(アーキテクチャ)により実現できると考えられるので実際のマイクロプロセッサの開発に本研究の成果を生かす事が今後の研究課題である。

## 謝辞

本研究の過程において終始客観的な立場からの適切な御助言と親身な御鞭撻を賜り、研究をまとめるにあたり懇切なる御指導と励ましを頂いた、静岡大学情報学部情報科学科教授 水野忠則博士に衷心より御礼申し上げます。

本論文をまとめる過程で種々な御指導と御鞭撻を頂いた、静岡大学情報工学部情報科学科教授 富樫敦博士に対し感謝申し上げます。また、研究の過程においてご指導と御助言を頂いた同学科助教授 佐藤文明博士に対し深謝申し上げます。また、本論文をまとめる過程で終始適切な御助言と励ましを頂いた静岡大学工学部システム工学科教授 八巻直一博士並びに静岡大学情報学部情報科学科助教授 西垣正勝博士に感謝の意を表します。

本研究をまとめるにおいて親身で力強い御指導とご鞭撻を頂いた、元三菱電機株式会社、現大阪工業大学情報学部情報システム学科教授 小島正典博士に対し深く感謝の意を表します。

本研究は、著者が三菱電機株式会社北伊丹製作所モス製造部、半導体応用システム開発部、マイクロコンピュータ部並びに三菱電機セミコンダクタ・アプリケーション・エンジニアリング株式会社において諸先輩方より薫陶を受けて行ったものであり、長年にわたり所属を変えつつもほぼ一貫して同一分野の仕事に従事できたこと、常に上長の御理解と援助を得たことが研究遂行の大きな力となっています。三菱電機株式会社専務半導体事業本部長 長澤絃一博士、通信システム事業本部長 伊藤善文常務、液晶統括事業部長 松本平八博士、三菱電機セミコンダクタ・アプリケーション・エンジニアリング株式会社代表取締役 土橋宏二社長、徳島文理大学工学部教授 小山健博士、三菱電機セミコンダクタシステム株式会社代表取締役 高山哲男社長、龍谷大学理工学部教授 長谷智弘博士に対し厚く感謝の意を表します。

研究の過程では多くの皆様方の御指導、御協力を得ました。武部秀治統轄部長、松井秀夫部長、宮本和俊部長、山口雅史部長、亀井達也部長、川原林隆部長に対し謝意を表します。特に直接協力を得た尚永幸久次長、鈴木次男部長、柴崎武部長、林和夫専任、部長 清水徹博士、部長 岡田圭介博士、部長 岩出秀平博士、専任 吉田豊彦博士、課長 坂本直史博士、藤原完美さん及び三菱電機株式会社半導体事業本部システムLSI統括部、三菱電機セミコンダクタ・アプリケーション・エンジニアリング株式会社の関係の皆様に対し謝意を表します。

## 参考文献

- [1] 山田罔裕, 川原林隆, 松井秀夫, 藤沢行雄, 浅野真弘, 玉木浩子: “ワンチップマイクロコンピュータの応用技術とソフトウェア”, 三菱電機技報, Vol.66, No.2, pp. 229-235, 1992.2.
- [2] 山田罔裕, 浜野尚徳, 森下誠, 中島義充, 増田紀隆, 尚永幸久: “ホームデジタルビデオ用システム LSI キット”, 三菱電機技報, Vol.62 No11, pp.942-947, 1988.11.
- [3] M. Kojima, K. Hatanaka and S. Kawabe: “Dynamic Tracking System using Moving Coil Actuators for a Consumer VCR”, IEEE Trans. On Cons. Elec., Vol.40, No.4, pp.969-975, 1994.11.
- [4] 三菱電機株式会社: “MELPS 8 ハードウェア説明書”, GAM-HR00-01A (54A0) , pp.2-28, 1976.
- [5] 松尾和義, 藤田紘一, 山田罔裕, 磯田勝房, 畑田明良: “4ビットワンチップマイクロコンピュータ”, 三菱電機技報, Vol.52, No.4, pp.273-277, 1978.4.
- [6] Rockwell International: “Application Notes Parallel Processing System (PPS4)”, Pub No.2518-D-26, pp.1-4, 1974.2, Rev.1, 1974.6.
- [7] Yoshide Hara, Masanori Kojima, Kunihiro Yamada and Hiroshi Shibata : “A New Digital FM Demodulator for VCRs”, IEEE Trans. on Cons. Elec., Vol.47, No.2, pp.243-248, 2001.5.
- [8] Kunihiro Yamada, Masanori Kojima, Toru Shimizu, Fumiaki Sato and Tadanori Mizuno: “A New RISC Processor Architecture for MPEG-2 Decoding”, IEEE Trans. on Cons. Elec., Vol.48, No.1, pp.143-150, 2002.2.

- [9] Kunihiro Yamada, Masanori Kojima, Tadashi Sakamoto, Toru Shimizu, Fumiaki Sato and Tadanori Mizuno: "A new IDCT Solution using RISC Processor for MPEG-2", Proceedings of the ACIS 2<sup>nd</sup> International Conference on Software Engineering Artificial Intelligence, Networking & Parallel/Distributed Computing SNPD '01, Nagoya Institute of Technology, Japan, pp.557-563, 2001.8.
- [10] 山田圀裕, 小笠原光孝, 水野忠則: "拡張デバック機能を持つマイクロコンピュータ・モニタ", 昭和 50 年情報処理学会 第 16 回大会, pp. 557-558, 1975.10.
- [11] 中島, 正田, 三橋, 小島正典, 宮後: "電子制御方式ビデオカセットレコーダー", 三菱電機技報, Vol.54, No.6, pp. 439-443, 1980.6.
- [12] J.V. Atanasoff: "Computing Machine fir the Solution of Large System of Linear Equation", Internal Report, Iowa State Univ. 1940.
- [13] J.V. Neumann: "First Draft of a Report on the EDVAC", Contract No.W-670-ODR-492, Moore School of E.E., Univ. of Pennsylvania, pp.101,1945.7.
- [14] J.G. Brainard and T.K. Sharpless: "The ENIAC", Electrical Engineering", Vol.67-2, pp.163-172, 1948.2.
- [15] M.V. Wilkes and W. Renwick: "An Ultrasonic Memory Unit for the EDSAC", Electronic Engineering, Vol.20. pp.208, 1948.
- [16] M.V. Wilkes and W. Renwick: "The EDSAC an Electronic Calculating Machine", J. Sci. Instrument, Vol.26, pp385-391, 1949.
- [17] M.V. Wilkes, D.J. Wheeler and S.Gill: "The Preparation of Program for an Electronic Digital Computers", 1<sup>st</sup> ed., Addison-Wesley .1951. & 2<sup>nd</sup> ed. 1957.

- [18] M.V. Wilkes and J.B. Stringer: "Micro-programming and the Design of the Control Circuits in an Electronic Digital Computer", Proc. of the Cambridge Philosophical Society, Vol.49, pt.2, pp.2230-2238, 1953.
- [19] Digital Computer Laboratory: "On the Design of a Very High-Speed Computer (ILLIAC-II)", Report No.80, Univ. of Illinois, 1957.10.
- [20] D.A. Patterson and J.L. Hennessy : "Computer Architecture a Quantitative Approach, Second Edition", Morgan Kaufmann publishers, inc.
- [21] T. Kilburn, D.B.G. Edwards, M.S. Lanigan and F.H. Summer: "One-level Storage System", IRE Trans, EC-11-2, pp.223-235, 1962.
- [22] C.J. Conti, D.H. Gibson, S.H. Pitkowsky and J.S. Liptay: "Structural Aspects of the System/360 Model 85", IBM Syst. J., 7-1, pp.2-21, 1968.
- [23] F. Faggin, M. Shima, M.E. Hoff, S. Feeney and S. Mazor: "The MCS-4- an LSI Microcomputer System", IEEE '72 Region Six Conf., pp.1-6, 1972.
- [24] W. Buchholz: "Planning a Computer System", PROJECT STREETCH, McGraw-Hill, 1962.
- [25] A.L. Leiner, and S.N. Alexander: "System Organization of the DYSEAC", IRE Trans. of Electronic Computer, Ec-3-1, pp.1-10, 1954.3.
- [26] 高橋, 西野, 松崎, 近藤: "トランジスタ計算機 ETL MARK III", 電気学会雑誌, 77-825, pp.675-679, 1957.6.
- [27] F.C. Williams, and T. Kilburn: "A Storage System for Use with Binary Digital Computing Machine", Proc. IEE, Vol.96, pt.3, pp.381-100, 1949.3, and Vol.97, pt.3, pp.453-454, 1950.11.

- [28] 中澤喜三郎：“計算機アーキテクチャと構成方式”，朝倉書店，1995.11.5.
- [29] 萱島興三：“電子計算機の方式設計”，産報，1972，再版，1975.
- [30] A.D. Patterson and D.R. Ditzel: “The case for the Reduced Instruction set computer”, Computer Architecture News, PP.8-6, 1980.
- [31] A.D. Patterson and S. Sequin: “A VLSI RISC. IEEE Computer”, Vol.15, No.9, pp.8-21, 1982.9.
- [32] R. Radain: “The 801 Minicomputer”, IBM J.R and D, 27-3, pp. 237-246, 1983.3.
- [33] 中島，北村，田村，滝内：“VPP500 スカラプロセサの特徴”，情報処理学会研究報告，94-ARC-104, pp. 129-136, 1994.1.
- [34] M.J. Flynn: “Very High-Speed Computer System”, Proc. IEEE, 54-12, pp.1901-1909 , 1966.12.
- [35] 田中英彦編：“超並列コンピュータシステム論文特集”，電子情報通信学会論文誌，J78-D-1-2, pp. 57-259, 1995.2.
- [36] 熊野，早川，三橋：“VCR 用高画質フィールドメモリーシステム”，昭和63年電機関係学会関西支部連合大会，G14-3, pp.G419, 1988.
- [37] 中島，水谷，玉木，大川，大橋，吉富，田口，矢澤：“フィールドメモリを内蔵したサーチモードでのノイズバーやスキュー歪みも改善する家庭用VTR”，日経エレクトロニクス，1986年10月20日号，No.406, pp. 195-214, 1986.10.20.
- [38] 伊藤，中川：“ビデオ処理 LSI”，テレビジョン学会誌，Vol.40, No.12, pp.1181-1187, 1986.12.

- [39] 小島, 塚原, 増田, 岡田, 北澤, 佐野: “VTR のオートトラッキング技術”, 三菱電機技報, Vol.64, No.8, pp, 631-634, 1990.
- [40] 高瀬広居, 玉木浩子, 尚永幸久, 山田圀裕: “VTR 用タイマ+チューナマイコンのソフトウェア開発”, MSC 技報, No.1, pp.4-8, 1983.1.
- [41] 田代哲, 山田圀裕, 山口雅史, 石丸善行: “CMOS 8ビット1チップマイコン M50754-XXXSP”, 三菱電機技報 Vol.59, No.8, pp.551-555, 1985.8.
- [42] T. Taka, K. Isogai, T. Watanabe, H. Tarugushi and T. Mizuno: “Improvement of TV Programs Selection Support Agent with History Database”, 2<sup>nd</sup> International Conference on Software Engineering Artificial Intelligence, Networking & Parallel/Distributed Computing, Nagoya Institute of Technology, Japan, pp.1041-1047, 2001.8.
- [43] 日経エレクトロニクス: “昭和 62 年度の日本電子産業展望”, No.418, pp.215-221, 1987.4.
- [44] 嶋正利: “マイクロプロセッサの25年”, 電子情報通信学会, Vol.82, No.10, pp. 997-1017, 1999.10.
- [45] 新井, 矢野: “マイクロプロセッサ”, 電子情報通信学会誌, Vol.81, No.11, pp.1107-1112, 1998.11.
- [46] 山田圀裕, 尚永幸久, 鷺尾幸夫: “半導体は21世紀のライフスタイルをどう変える”, 三菱電機セミコンメイト, 三菱電機, pp.10-11.
- [47] Electronics review: “Design enhancements boost performance of 8-bit microprocessors” Electronics, 1976.1.
- [48] 三菱電機株式会社: “三菱ワンチップ4ビットマイクロコンピュータ MELPS 4”, マニュアル.H - C4074-A 北 - 7804 (TOT), 1978.4.



- [49] Rockwell International; "PPS 4/2 Two-chip Microcomputer System" Document No.11660N40 Revision, 1975.9.
- [50] 林和夫, 尚永幸久, 武部秀治, 山田圀裕, 元吉啓登, 鈴木次男: "VTR ソフトウェアサーボ用 16 ビットマイクロコンピュータ", 三菱電機技報, Vol.66, No.2, pp.205-213, 1992.2.
- [51] 佐藤文明: "並行処理におけるプログラム試験 通信プログラム試験", 情報処理, Vol.39, No.1, pp.13-18, 1998.1.
- [52] 大槻貞二郎, 松本平八; "マイクロコンピュータ用 LSI の信頼性", 電子材料, pp108-115, 1976.11.
- [53] 水野忠則, 井手口哲夫, 大槻貞二郎, 山田圀裕, 浜野尚徳: "マイクロプロセッサシリーズの最近の実績", 三菱電機技報 Vol.51, No.3, 1977.3.
- [54] 中谷孝ほか: "TELETEXT スライサ内蔵 OSD コントローラの開発", MSC 技報, No.11, 1999, pp.10-14, 1999.11.
- [55] H. Maejima, M. Kainaga and K. Uchiyama: "Design and Architecture for Low-Power/High-Speed RISC Microprocessor Super H", IEIFC TRANCE ELECTRON, Vol.E80-C, No.12, 1997.12.
- [56] 萩原宏: "命令セットアーキテクチャ", 情報処理 Vol.29, No12, pp.1381-1386, 1988.12.
- [57] M. Yoshimoto, S. Nakagawa, K. Murakami, S. Asai, Y. Akasaka, Y. Nakajima and Y. Horiba.: "A Digital Processor For Decoding of Composite TV Signals Using Adaptive filtering 1986", IEEE International Solid-State Circuits Conference (ISSCC) Digest of Technical Papers, THAM12.5, pp.152-153, 1986.2.
- [58] 加茂明監修: "最新画像メモリとその応用", (株) トリケップス

- [59] 中村文久：“民生分野におけるシステム LSI”，電子情報通信学会誌, Vol.81 No.11, pp.1167-1173, 1998.11.
- [60] 木村浩三ほか：“ソフトウェアでの実時間処理を実現した民生用メディア処理のプロセッサ Media Core Processor”，Matsushita Technical Journal, Vol.45, No.2, pp.9-16, 1999.4.
- [61] 中嶋純子, 松井充：“共通鍵暗号 MISTY1 の最適なソフトウェア実装について Fast Implementation of MISTY1 in Software”，The 2001 Symposium on Cryptography and Information security, pp.23-26, 2001.1.
- [62] 山田圀裕, 水野忠則：“最も基本となるプログラム バイナリ・ローダ”，インターフェイス, pp.72-82, 1977.4.
- [63] 鈴木君子：“ワーニエ・プログラミング・メンドロジイ”，数理科学, No.125, pp.78-89, 1973.11.
- [64] 山田圀裕, 水野忠則, 田中禎一：“マイクロコンピュータにおけるマルチタスク制御の一方式 1265”，昭和 51 年電子通信学会総合全国大会, pp.6-111, 1976.3.
- [65] 山田圀裕, 水野忠則：“マイクロプロセッサにおける ROM 化プログラム機能拡張法”，昭和 51 年情報処理学会 第 17 回全国大会, pp.353-354, 1976.11.
- [66] 山田圀裕, 水野忠則, 土屋鍊平：“マイクロプロセッサにおけるプログラム開発用モニタ”S1-4, 昭和 52 年電子通信学会総合全国大会, pp.6-274. 1977.3.
- [67] 山田圀裕, 水野忠則, 瀬戸川妙子, 田中禎一：“マイクロコンピュータ用モニタ”1329, 昭和 50 年度電子通信学会全国大会, Vol.1975, pp.1349, 1975.3.

- [68] 樋口敬三, 山田罔裕, 杉本正樹: “システム提案型マイコン応用技術”, 三菱電機技報, Vol.63, No.3, pp.241-246, 1994.3.
- [69] 山田罔裕, 小島正典, 近藤光重, 佐藤文明, 水野忠則: “システム LSI に適した交流バイアス音声記録回路”, 電気学会論文誌 C, Vol21-C, No.5, pp.947-952, 2001.5.
- [70] Kunihiro Yamada, Masanori Kojima, Teruo Furukawa, Fumiaki Sato and Tadanori Mizuno: “Microprocessor-associated Audio signal Processing System for VHS VCRS”, IEEE Trans. on Cons. Elec., Vol47, No.2, pp.236-242, 2001.5.
- [71] 田中正文: “システム L S I の応用分野”, 電気学会誌, Vol.120, No.3., pp.140-143, 2000.3.
- [72] 桜井貴康: “システム LSI のアプリケーションとシステム LSI の課題”, 電子情報通信学会, Vol.81, No.11, pp.1082-1086, 1998.11.
- [73] 三菱電機株式会社: “VTR 専用 IC”, ‘96 三菱半導体データブック, pp.393-399, 1996.2.
- [74] International Electrotechnical Commission IEC Standard : Helical-scan video tape cassette system using 12.65 mm (0.5inch) magnetic tape on type VHS, Publication 774, pp.1-39, 1983.
- [75] 日本電子機械工業会: “ビデオテープレコーダと周辺機器の相互接続”, EIAJ CP-1211, pp.16-17, 1990.1.
- [76] 畠中, 大川, 中島: “家庭用 VTR のデジタルスピードサーチ技術”, 1986 年テレビジョン学会全国大会講演予稿集, 論文番号 7-13, pp.181-182, 1986.

- [77] 山田圀裕：“デジタルビデオ用 IC システムの開発”，Semicon Mate 三菱電機, pp.12-13, 1989.4.
- [78] 松永：“フィールドメモリ専用 1 M チップ VTR/テレビに向けて一挙に出そう”，日経エレクトロニクス, No.421, pp.147-162, 1987.5
- [79] Edited by Laurence Altman ; “Microprocessors”, Electronics Book Series, pp.12-13, 1975.
- [80] 熊野：“VCR 用高画質フィールドメモリーシステム”，MMS 技報, No.1, pp.51-56, 1990.
- [81] 渡部一喜, 玉木茂弘, 山田圀裕：“VTR 用フィールドメモリ応用システム IC” G14-5, 昭和 6 2 年電気関係学会関西支部連合大会, pp.G391, 1987.
- [82] 在本ほか：“VTR・TV 用ピクチャインピクチャシステム IC”，昭和 6 2 年電気関係学会関西支部連合大会, G14-4, 1987.
- [83] 玉木茂弘, 山田圀裕, 宮本崇行：“ビジュアル機器と画像メモリ”，日刊工業新聞社 電子技術, pp.28-34, 1988.7.
- [84] 山田圀裕, 鷺尾幸夫：“画像バッファによる付加機能設計”，高画質ビデオカメラ開発動向と設計技術（工業技術セミナー T-2095）日本工業技術センター, pp.29-51, 1988.10.
- [85] 社団法人 日本電子工業振興協会：“マイクロコンピュータに関する調査報告書 [ I ]”，89-パ-4, pp.15-30, 1993.5.
- [86] Kazushi Kitagawa, Masanori Kojima, Kunihiro Yamada and Hiroshi Shibata: “A New Video Front-end Processing System”, IEEE Trans. on Cons. Elec., Vol.47, No.2, pp. 231-253, 2001.5.

- [87] T. Miki, Y. Nakamura, M. Nakaya, S. Asai, Y. Akasaka and Y. Horiba: "An 80 MHz 8b CMOS D/A Converter", 1986 IEEE International Solid-State Circuits Conference (ISSCC) Digest of Technical Papers, THAM11.2, pp.132-133, 1986.2.
- [88] 熊本, 中屋, 本田, 浅井, 赤坂, 堀場: "8-bit 30 MHz CMOS A/D コンバータ", 昭和61年度電子通信総合全国大会講演論文集, 論文番号436, pp.2-194, 1986.
- [89] Mitsubishi Electric Corporation: "Microcomputers", Mitsubishi Denki Giho, Vol. 74, No. 1, pp. 88-90, 2000.1.
- [90] 清家忠義ほか: "DVD用システムLSI", 電子情報通信学会誌, Vol.84, No.8, pp.572-583, 2001.8.
- [91] K. Hatanaka, Y. Shirasaki, N. Fujiwara, M. Watanabe, T. Furukawa and S. Kawabe: "A Digital FM Signal Processing System for VCRs", IEEE Trans. on Cons. Elec., Vol. 41, No. 3, pp. 405-413, 1995.8.
- [92] 吉田豊彦: "VLIWメディアプロセッサ", 情報処理学会, Vol. 38, No. 6, pp.499-506,1997.6.
- [93] T. Sakamoto and T. Hase: "Software JPEG for a 32-bit with Dual Issue", IEEE Trans. on Cons. Elec., Vol. 44, No. 4, pp. 1334-1341, 1998.11.
- [94] 加代孝夫: "VTRの規格(III)", テレビ誌, Vol.45, No.4, pp.527-530, 1991.4.
- [95] Y. Takeuchi and M. Ito: "Recent Technology for Digital Image Compression Algorithm", J.IEE-C, Vol.114, No.7/8, pp.767-772, 1994.7.

- [96] T. Sakamoto and T. Hase: "JPEG Software Solution for a 32-bit MCU", IEEE Transactions on Consumer Electronics, Vol.43, No.3, pp. 410-417, 1997.8.
- [97] 国分明男: "MOS LSI 技術とマイクロコンピュータ", 電子技術総合研究所・パターン情報処理システム調査・研究報告 PIPS-R-No.3, pp.33-42, 1973.10.
- [98] Mitsubishi Electric Corporation, "Microcomputers", Mitsubishi Denki Giho, Vol. 71, No. 1, pp. 56-58, 1997.1.
- [99] W.A. Chen, C.Harrison and S.C. Fralic: "A Fast Computational Algorithm for the Discrete Cosine Transform", IEEE Trans. on Communications, Vol. COM-25, No. 9, pp. 1004-1011, 1977.9.
- [100] 宮部一裕ほか: "ハイビジョン受信機 HV20", National Technical Report, Vol.41 No.4, pp.373-391, 1995.8.
- [101] 岩村聡ほか: "パームコーダーPV-D705", National Technical Report, Vol.41 No.2, pp.212-217, 1995.4.
- [102] 古閑敏夫: "映像符号化の流れ", 映像情報メディア学会誌, Vol.54, No.5, pp.620-628, 2000.5.
- [103] 寺根, 中川, 吉本, 嶋田, 篠原, 畑中: "24ビット, 50ns 画像・映像処理用シグナルプロセッサ (DISP)", 三菱電機技報, Vol.63, No.11, pp.929-933, 1989.11.
- [104] A. タウンバウム, 水野忠則, 鈴木健二, 宮原太郎, 佐藤文明: "分散オペレーティングシステム", プレンティスホール出版, 1996.
- [105] Kazuaki Murakami: "Parallel Processing inside RISC Processors", J. IEE-C, Vol.113, No. 11, pp. 906-911, 1993.11.

- [106] マイヤーズほか; 富沢, 神城訳: “高性能マイクロプロセッサアーキテクチャインテル 80960 の設計”, 丸善, 1990.
- [107] 高田, 近藤, 清水: “4 M バイト DRAM 内蔵 32 ビット RISC マイクロコントローラ M32Rx/D”, 三菱電機技報, Vol. 73, No.3, pp.182-185, 1999.3.
- [108] 大川清人, 小笠原光孝, 保刈明彦: “マイクロプロセッサ応用機器の開発用システム”, 日経エレクトロニクス, pp.92-111, 1976.
- [109] 山田圀裕: 特許公報 JP1468304, 庁内整理番号 6749-5J, 公告, pp.85-89, 1988.2.
- [110] 八木伸行: “放送技術とコンピューター放送におけるコンピュータ技術導入の動向”, 映像情報メディア学会誌, Vol.51, No.7, pp.975-982, 1997.7.
- [111] 黒田忠広: “システム LSI の可能性と課題”, 電子情報通信学会誌, Vol.84, No.8, pp.552-558, 2001.8.
- [112] 今井正治, 松永裕介: “ハードウェア/ソフトウェア・コアデザイン”, 情報処理, Vol.36 No.7, pp.604-619, 1995.7.
- [113] 山田茂: “ソフトウェアの品質評価に関する考え方と動向”, 情報処理 Vol.32, No.11, pp.1189-1202, 1991.11.
- [114] 山内経則, 土屋主税: “アナログ/デジタル混載 LSI に適した BiCMOS 技術”, 電学論 C, 116 巻 8 号, pp.905-911, 1996.8.
- [115] L. Hammond, B.A. Nayfeh and K. Olukotun: “A Signal-Chip Multiprocessor, IEEE computer, pp.79-85, 1997.9.

- [116] 山田道裕, 布村泰浩: “メモリ混載 LSI の現状と将来”, 電子情報通信学会誌, Vol.81 No.11, pp.1093-1099, 1998.11.
- [117] D.W. Anderson, F.J. Sgaracio and M. Tomasulo: “The IBM System 1360 Model 91”, Machine philosophy and Instruction Handling, IBM J,R&D, pp.8-24, 1967.
- [118] J. Wondra: “I<sup>2</sup>C-Bus Controlled Hifi Audio-processor”, IEEE Trans. on Cons. Rlec., Vol.36, No.3, pp.344-348, .1990.8.
- [119] A.Yamada, Y. Nunomura, H. Suzuki, H. Sato, N. Itoh, T. Kurafuji, N. Yoshioka, J. Nakanishi, R. Akiyama, A. Iwabu, H. Takata, T. Shibagaki, Y. Mori, Y. Kittaka, T. Arakawa, and S. Iwade, “A 400 MHz RISC Microcontroller”, “An International Symposium on Low-Power and High-Speed Chips (COOL Chips V) 2002.
- [120] Texas Instruments Incorporated, “TMS 1000 Series Data Manual”, 1975,12
- [121] Kunihiro Yamada, Masanori Kojima, Toru Shimizu, Shuhei Iwade, Fumiaki Sato, Tadanori Mizuno: “A new IDCT Solution using a RISC Processor Architecture”, KES'2002 Sixth International Conference on Knowledge-Based Intelligent Information & Engineering Systems, the University of Milan, Crema, Italy, 2002.9. (投稿中)



## 筆者発表論文

- (1) 山田圀裕, 水野忠則, 瀬戸川妙子, 田中禎一: “マイクロコンピュータ用 モニタ”、昭和 50 年度電子通信学会全国大会, Vol.1975, pp.1349, 1975.3.
- (2) 山田圀裕, 小笠原光孝, 水野忠則: “拡張デバック機能を持つマイクロコンピュータ・モニタ”, 昭和 50 年情報処理学会 第 16 回大会, pp. 557-558, 1975.10.
- (3) 山田圀裕, 水野忠則, 田中禎一: “マイクロコンピュータにおけるマルチタスク制御の一方式 1265”, 昭和 51 年電子通信学会総合全国大会, pp. 6-111, 1976.3.
- (4) 山田圀裕, 水野忠則: “マイクロプロセッサにおける ROM 化プログラム機能拡張法”, 昭和 51 年情報処理学会 第 17 回全国大会, pp.353-354, 1976.11.
- (5) 山田圀裕, 水野忠則, 土屋鍊平: “マイクロプロセッサにおけるプログラム開発用モニタ”S1-4, 昭和52年電子通信学会総合全国大会, pp.6-274. 1977.3.
- (6) 水野忠則, 井手口哲夫, 大槻貞二郎, 山田圀裕, 浜野尚徳: “マイクロプロセッサシリーズの最近の実績”, 三菱電機技報 Vol.51, No.3, 1977.3.
- (7) 山田圀裕, 水野忠則: “最も基本となるプログラム バイナリ・ローダ”, インターフェイス, pp.72-82, 1977.4.
- (8) 松尾和義, 藤田紘一, 山田圀裕, 磯田勝房, 畑田明良: “4ビットワンチップマイクロコンピュータ”, 三菱電機技報, Vol.52, No.4, pp.273-277, 1978.4.
- (9) 高瀬広居, 玉木浩子, 尚永幸久, 山田圀裕: “VTR 用タイマ+チューナマイコンのソフトウェア開発”, MSC 技報, No.1, pp.4-8, 1983.1.
- (10) 田代哲, 山田圀裕, 山口雅史, 石丸善行: “CMOS 8ビット1チップマイコン M50754-XXXSP”, 三菱電機技報 Vol.59, No.8, pp.551-555, 1985.8.

- (1 1) 渡部一喜, 玉木茂弘, 山田圀裕: “VTR 用フィールドメモリ応用システム IC” G14-5, 昭和62年電気関係学会関西支部連合大会, pp.G391, 1987.
- (1 2) 玉木茂弘, 山田圀裕, 宮本崇行: “ビジュアル機器と画像メモリ”, 日刊工業新聞社 電子技術, pp.28-34, 1988.7.
- (1 3) 山田圀裕, 浜野尚徳, 森下誠, 中島義充, 増田紀隆, 尚永幸久: “ホームデジタルビデオ用システム LSI キット”, 三菱電機技報, Vol.62 No.11, pp.942-947, 1988.11.
- (1 4) 山田圀裕, 鷺尾幸夫: “画像バッファによる付加機能設計”, 高画質ビデオカメラ開発動向と設計技術 (工業技術セミナー T-2095) 日本工業技術センター, pp.29-51, 1988.10.
- (1 5) 山田圀裕, 川原林隆, 松井秀夫, 藤沢行雄, 浅野真弘, 玉木浩子: “ワンチップマイクロコンピュータの応用技術とソフトウェア”, 三菱電機技報, Vol.66, No.2, pp. 229-235, 1992.2.
- (1 6) 林和夫, 尚永幸久, 武部秀治, 山田圀裕, 元吉啓登, 鈴木次男: “VTR ソフトウェアサーボ用 16 ビットマイクロコンピュータ”, 三菱電機技報, Vol.66, No.2, pp.205-213, 1992.2.
- (1 7) 樋口敬三, 山田圀裕, 杉本正樹: “システム提案型マイコン応用技術”, 三菱電機技報, Vol.63, No.3, pp.241-246, 1994.3.
- (1 8) 山田圀裕, 小島正典, 近藤光重, 佐藤文明, 水野忠則: “システム LSI に適した交流バイアス音声記録回路”, 電気学会論文誌 C, Vol.21-C, No.5, pp.947-952, 2001.5.
- (1 9) Kunihiro Yamada, Masanori Kojima, Teruo Furukawa, Fumiaki Sato and Tadanori Mizuno: “Microprocessor-associated Audio signal Processing System for VHS VCRS”, IEEE Trans. on Cons. Elec., Vol.47, No.2, pp.236-242, 2001.5.
- (2 0) Yoshide Hara, Masanori Kojima, Kunihiro Yamada and Hiroshi Shibata: “A New Digital FM Demodulator for VCRs”, IEEE Trans. on Cons. Elec., Vol.47, No.2, pp.243-248, 2001.5.

- (2 1) Kazushi Kitagawa, Masanori Kojima, Kunihiro Yamada and Hiroshi Shibata: "A New Video Front-end Processing System", IEEE Trans. on Cons. Elec., Vol.47, No.2, pp. 231-253, 2001.5.
- (2 2) Kunihiro Yamada, Masanori Kojima, Tadashi Sakamoto, Toru Shimizu, Fumiaki Sato and Tadanori Mizuno: "A new IDCT Solution using RISC Processor for MPEG-2", Proceedings of the ACIS 2<sup>nd</sup> International Conference on Software Engineering Artificial Intelligence, Networking & Parallel/Distributed Computing SNPD '01, Nagoya Institute of Technology, Japan, pp.557-563, 2001.8.
- (2 3) Kunihiro Yamada, Masanori Kojima, Toru Shimizu, Fumiaki Sato and Tadanori Mizuno: "A New RISC Processor Architecture for MPEG-2 Decoding", IEEE Trans. on Cons. Elec., Vol.48, No.1, pp.143-150, 2002.2.
- (2 4) Kunihiro Yamada, Masanori Kojima, Toru Shimizu, Shuhei Iwade, Fumiaki Sato, Tadanori Mizuno: "A new IDCT Solution using a RISC Processor Architecture", KES'2002 Sixth International Conference on Knowledge-Based Intelligent Information & Engineering Systems, the University of Milan, Crema, Italy, 2002.9.  
(投稿中)