

## V60 / V70 アーキテクチャ

高橋 利也 ・ 矢野 陽一

日本電気株式会社 半導体応用技術本部 / マイクロコンピュータ事業部

本報告は32ビットマイクロプロセッサであるV60/V70のアーキテクチャに関する。

このV60/V70アーキテクチャは、豊富なレジスタ・セットを持ち命令的にも高級言語指向を目指し、科学技術計算に対応すべく浮動小数点演算機能、仮想記憶に対応すべくその管理機構、システムの高信頼性をサポートするためのFRM (Functional Redundancy Monitoring) 機能などを統合したものになっている。

本稿では、このV60/V70アーキテクチャの特徴について示す。

### The Architecture of V60/V70 Microprocessors

Toshiya TAKAHASHI and Yoichi YANO  
Microcomputer Division and Semiconductor Application Division  
NEC Corporation  
484, Tsukagoshi 3-chome, Saiwai-ku, Kawasaki, Kanagawa, 210 Japan

This report describes the architecture of V60/V70 32-bit microprocessors. The architecture integrates various features into a single silicon die, such as a rich set of general purpose registers, high level language oriented instruction set, floating-point data handling which is suitable for scientific applications, and the FRM (Functionality Redundancy Monitoring) operation mode which supports highly-reliable systems configuration.

These features will be introduced.

## 1. はじめに

LSI技術の飛躍的進歩によるマイクロプロセッサの高性能、高機能に伴い、その応用も従来はミニコンピュータや汎用コンピュータを利用していただけの分野にまで拡大しつつある。このような応用分野においては、膨大な情報処理を効率良く、かつ安全に遂行せねばならない。V60/V70アーキテクチャは、この命題を解決すべく開発、設計されている。

本稿では、V60/V70マイクロプロセッサのアーキテクチャの特徴について述べる。

## 2. V60/V70の内部構造

図1はV70のチップレイアウトである。V60/V70は6つの独立した機能ユニットから構成され、各ユニットはパイプライン処理を行うように結合されている。機能ユニットの内訳はPFU（プリフェッチユニット）、IDU（命令デコードユニット）、EAG（実効アドレス生成器）、MMU（メモリ管理ユニット）、BCU（バス制御ユニット）、EXU（実行ユニット）である。V60/V70では、このような6段パイプライン方式を採用したことにより、最大4命令を同時実行できる。

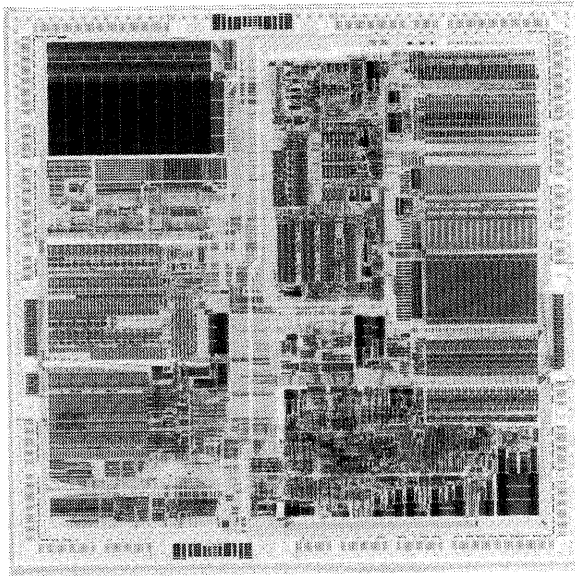


図1. V70(μPD70632)のチップ写真  
1.5μmルールのCMOSアルミ2層プロセスにより約385000トランジスタを14.35mm×14.24mmのチップ上に集積している。

## 3. レジスタセット

V60/V70は32ビットの汎用レジスタを32本持つ汎用レジスタアーキテクチャのプロセッサである。

32本という大量の汎用レジスタは、最適化コンパイラによる変数のレジスタ割り付け数を増やし、メモリアクセスの回数を減らすとともに、処理の高速化に寄与する。図2にV60/V70のレジスタセットを示す。V60/V70のレジスタセットは、一般のプログラムが活用することのできるプログラム・レジスタセットと、オペレーティングシステムのようなシステム管理者のみが参照することのできる特権レジスタセットに大別される。すべてのレジスタ長は32ビットである。

### 【プログラムレジスタセット】

#### 汎用レジスタ

|         |
|---------|
| R0      |
| R1      |
| R2      |
| R3      |
| R4      |
| R5      |
| R6      |
| R7      |
| R8      |
| R9      |
| R10     |
| R11     |
| R12     |
| R13     |
| R14     |
| R15     |
| R16     |
| R17     |
| R18     |
| R19     |
| R20     |
| R21     |
| R22     |
| R23     |
| R24     |
| R25     |
| R26     |
| R27     |
| R28     |
| R29(AP) |
| R30(FP) |
| R31(SP) |
| PC      |
| PSW     |

### 【特権レジスタセット】

#### スタックポインタ

|      |
|------|
| ISP  |
| LOSP |
| LISP |
| L2SP |
| L3SP |

#### タスク関連レジスタ

|      |
|------|
| TR   |
| TKCW |

#### システムレジスタ

|      |
|------|
| SBR  |
| SYCW |
| PIR  |

#### 仮想記憶レジスタ

|       |
|-------|
| ATBRO |
| ATLRO |
| ATBR1 |
| ATLR1 |
| ATBR2 |
| ATLR2 |
| ATBR3 |
| ATLR3 |

#### アドレストラップレジスタ

|        |
|--------|
| TRMOD  |
| ADTRO  |
| ADTR1  |
| ADTHRO |
| ADTHR1 |

#### 第2PSW

|      |
|------|
| PSW2 |
|------|

図2. レジスタセット

## (1) プログラム・レジスタセット

プログラム・レジスタセットは32本の汎用レジスタとPC（プログラムカウンタ）、PSW（プログラム・ステータス・ワード）からなる。

汎用レジスタは、32本がすべてデータ、ポインタ、インデックス用として利用することができ、浮動小数点データも整数と同様におくことができる。レジスタR29からR31の3本は、高級言語における手続き呼出し／帰還のための特別な用途として割り当てである。R29はアークギュメントポインタ（AP）として手続き呼出し時のパラメータ領域のベースアドレスを保持する。R30はフレームポインタ（FP）として手続きが呼出されたときのスタックフレームのアドレスを保持する。R31はスタックポインタ（SP）として、スタックの先頭アドレスを保持する。

PCは現在実行中の命令の先頭番地を保持する。

PSWは分岐条件に係わる条件フィールドの他に、浮動小数点の例外状態を示す浮動小数点フィールド、シングルステップや割込みの許可などのプログラムの実行状態を規定するコントロールフィールド、実行レベルなどのシステムの状態を示すステータスフィールドの4つのフィールドからなる。

## (2) 特権レジスタセット

特権レジスタは、スタックポインタ群、タスク管理、システム管理、仮想記憶管理、アドレストラップ・レジスタ群、エミュレーション用PSWに分類される。

スタックポインタ群は4つの実行レベルと割込み処理用の5本のスタックポインタからなる。タスク管理のレジスタ群はタスクの、システム管理のレジスタ群はシステムの状態をそれぞれ規定する。アドレストラップ・レジスタ群は2組存在する。

## 4. 命令セット

V60/V70の命令セットはソースオペランドとデスティネーションオペランドに対し、自由なアドレッシングモードを使用できる2アドレス方式を採用している。このため、メモリとメモリ間の演算も自由に記述できる。

V60/V70には119種273の命令が用意されている。命令セット機能概要一覧を表1に示す。

オペレーティングシステムの行う仮想記憶管理やタスク管理をサポートする命令は特権命令に含まれる。

扱うデータ・タイプは、表2に示すとおりで13種類ある。

表2. データ・タイプ

| データ・タイプ   | ビット数   |
|-----------|--------|
| ビット データ   | 1      |
| 整数 データ    |        |
| バイト       | 8      |
| ハーフワード    | 16     |
| ワード       | 32     |
| ダブルワード    | 64     |
| 浮動小数点     |        |
| ショート・リアル  | 32     |
| ロング・リアル   | 64     |
| 10進データ    |        |
| パック       | 8      |
| アンパック     | 16     |
| 文字ストリング   |        |
| バイト単位     | 8 * n  |
| ハーフワード単位  | 16 * n |
| ビット・フィールド | 1 * n  |
| ビット・ストリング | 1 * n  |

表1. 命令セット機能概要一覧

| 機能分類        | 機能概要   |
|-------------|--|
| 転送・変換       | 通常転送、拡張／縮小転送、交換、プッシュ、ポップ、特殊転送、複数レジスタセーブ／リストア |
| 整数算術演算      | 加算、減算、乗算、除算、インクリメント、デクリメント、剰余算               |
| 比較          | 比較、テスト                                       |
| 論理演算        | 論理否定、論理積、論理和、排他的論理和                          |
| シフト・ローテート   | 算術／論理シフト、ローテート                               |
| 実効アドレス計算    | 実効アドレス計算                                     |
| ビット操作       | テスト、セット、クリア、反転                               |
| ビット・フィールド操作 | ビット・フィールドの抽出／挿入／比較                           |
| ビット・ストリング操作 | 転送、論理否定転送、論理和、論理積、排他的論理和、0／1サーチ              |
| 10進演算       | 加算、減算  |
| 文字列操作       | 転送、比較、文字サーチ、文字スキップ                           |
| 浮動小数点演算     | 転送、加算、減算、乗算、除算、比較、負数、絶対値                     |
| 分岐系         | 分岐、ループ、コール／リターン、スタック・フレーム生成／削除               |
| 特権命令        | 特権レジスタロード／ストア、入出力命令、タスク・コンテストロード／ストア 等       |
| その他         | NOP、ソフトウェア・トラップ 等                            |

(1) 高級言語サポート

V60/V70では高級言語の手続き呼出し/戻りを効率的に実現するための命令が用意されている。ここでは、V60/V70における手続き呼出しの1例を示す。図3(a)は、呼び出し例と手続きのコード例、図3(b)は、スタックの動きである。このとき、汎用レジスタのうちR29からR31の3本（それぞれ、アークギュメントポインタ（AP）、フレームポインタ（FP）、スタック・ポインタ（SP））を操作する。以下が、その操作の手順である。

- ①パラメータをスタックに積む。
- ②手続き呼出し命令（CALL）により、戻り先PCと旧APがスタックに退避され、新APが設定された後、手続きへ分岐する。
- ③手続きで使用（破壊）する汎用レジスタを退避する。
- ④スタックフレーム生成命令（PREPARE）により、旧FPを退避した後に新FPを設定し、手続きに必要なローカル変数領域を確保する（すなわち、SPを進める）。
- ⑤この時点で手続きの処理本体が開始される。パラメータのアクセスはAPを、ローカル変数のアクセスはFPをベースとして行われる。
- ⑥手続きからの戻りには、スタックフレーム削除命令（DISPOSE）により、FP、SPの値を復帰し、ローカル変数領域を開放する。
- ⑦退避していた汎用レジスタを復帰させる。
- ⑧帰還命令（RET）により、呼出し側に制御が戻るとともにAPの値が復帰され、同時にスタック上のパラメータ領域を開放する（すなわち、SPを戻す）

以上の例では手続きのパラメータ領域をスタック上に確保したが、言語によっては別のデータ領域にパラメータを置くことがある。この場合は単にCALL命令のオペランドで指定する新AP値が変わると、RETでパラメータ領域の開放を行わないだけで同様な命令列を使用する。

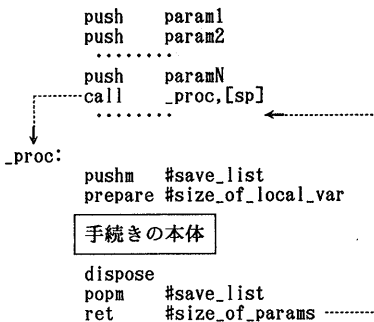


図3(a). 手続き呼出しのシーケンス

(2) オペレーティングシステム・サポート

V60/V70がオペレーティングシステムをサポートする命令としては、仮想記憶管理に関するもの、タスクのコンテキスト切替えに関するものなどがある。これらはすべて実行レベル0のみで使用できる特権命令である。

(a)仮想記憶管理

仮想記憶管理命令にはアドレス変換に用いるテーブル（エリアテーブル、ページテーブル）を参照/更新する命令、TLBのエントリをクリアする命令、仮想アドレスを実アドレスに変換する命令がある。

(b)コンテキスト切替え

V60/V70においてタスクコンテキストは次に示すレジスタ群と仮想記憶を管理するメモリ上のテーブルにより定義される。これらのレジスタ群がタスク切替え時に入替わる対象となり、各タスク毎にTCB（タスクコントロールブロック）領域としてメモリ上に格納されている。

- ①仮想アドレス空間環境  
エリアテーブル・レジスタペア
- ②プログラム実行環境  
汎用レジスタ  
各実行レベル用スタックポインタ

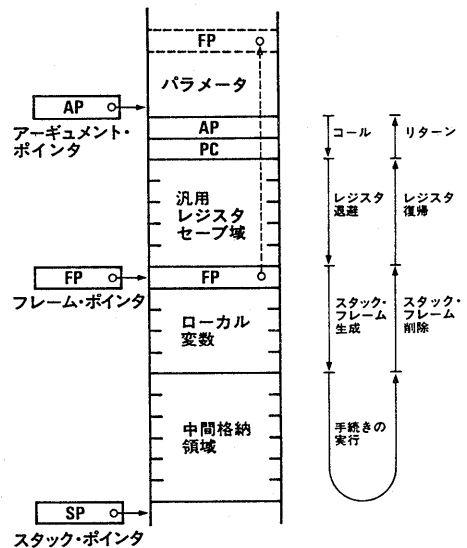


図3(b). 手続き呼出しとスタック

### ③タスク固有の状態情報

タスクレジスタ

タスク・コントロール・ワード

V60/V70ではタスクコンテキストを入替えるための命令を用意している。それがLDTASK命令、STTASK命令である。TCBの大きさは可変であり、それに含まれるレジスタはLDTASK命令、STTASK命令のオペランドおよびSYCW（システム・コントロール・ワード）によって指定する。TCB領域のベースアドレスはTR（タスクレジスタ）が保持する。

LDTASK命令を実行すると、第2オペランドで指定したTCB領域から、第1オペランドで指定したV60/V70のレジスタ群に環境を設定する。同時に、TRにTCB領域のベースアドレスをセットする。

逆にSTTASK命令では、オペランドで指定したV60/V70上のレジスタからTRが保持するTCB領域に環境を退避する。V60/V70においてコンテキスト切替えはSTTASK（旧コンテキストのストア）、LDTASK（新コンテキストのロード）、RETI S（ハンドラからの復帰）の3命令で記述できる。

## 5. 仮想記憶管理

V60/V70は内蔵するメモリ管理ユニットにより、ページング方式で仮想空間を実現できる。1つの仮想空間は最大4Gバイトであり、各タスク毎に仮想空間を割り当てることもでき、多重仮想空間構成がとれる。

### 5.1 仮想空間の構成

V60/V70の仮想空間の構成を図4に示す。最大4Gバイトの仮想空間は4つのセクションに、最大1Gバイトの各セクションは最大1024個のエリアに、最大1Mバイトの各エリアは最大256個のページに分割される。各ページのサイズは4Kバイトである。

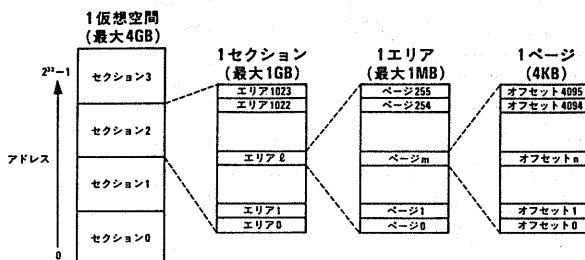


図4. 仮想空間の構成

### 5.2 多重仮想空間

通常のマルチタスキングを行っている仮想記憶システムでは、各々のタスク毎に独立した仮想空間を与えるのが普通である。すなわち、各タスク毎に別々の仮想空間が存在する。

V60/V70では、このような多重仮想空間構成をとることが可能になっている。

多重仮想空間構成を採用するシステムでは、各々のタスクに独立なアドレス域だけでなく、すべてのタスクに共通したサービスを提供するプログラム（オペレーティングシステムなど）を使用するために、各々のタスクに共通のアドレス域を提供する必要がある。

V60/V70では、仮想空間をタスク間で共有するか独立にするかを任意に指定することができる。例えば、共通アドレス域としてセクション3の1Gバイト、独立アドレス域としてセクション0からセクション2までの3Gバイトを割り振る仮想空間構成をとることができる。また、システム全体での仮想空間の共有の他にも、複数の仮想空間の間でのプライベートな共有ができる。これはエリア単位の共有である。図5に多重仮想空間の構成例を示す。

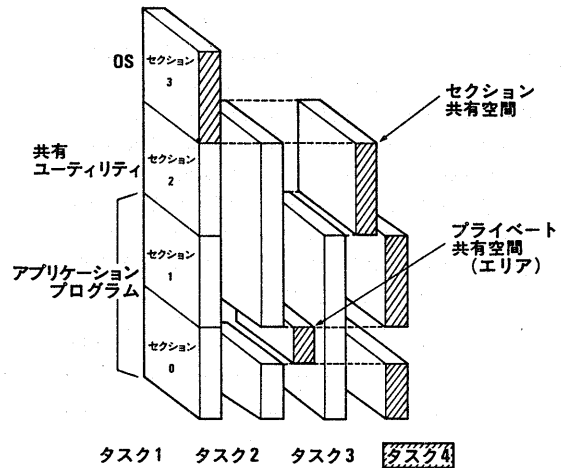


図5. 多重仮想空間の構成例

### 5.3 アドレス変換

V60/V70は仮想空間を実現するために必要となる、仮想アドレスから実アドレスへのアドレス変換機構をオンチップに内蔵している。通常の変換はファームウェアにより、オンチップのレジスタ（エリアテーブル・レジスタペア）およびメモリ上のアドレス変換テーブル（エリアテーブル、ページテーブル）を参照して行われる。

アドレス変換で参照するレジスタペアと変換テーブルを図6に示す。ひとたびアドレス変換でページ番号が得られると、その結果はオンチップ内のTLB（高速アドレス変換機構）にキャッシングされ、以後のアドレス変換はTLB上で高速に行われる。

TLBは16エントリを持つフルアソシアティブ方式であり、エントリの入換えは疑似LRUで行われる。

図7にアドレス変換機構を示す。

#### 5. 4 アクセス保護機構

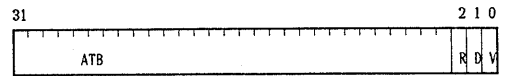
V60/V70では2段階の保護機構を実現している。プログラムの実行時には、レベル0からレベル3までの4つの実行レベルのうち1つで走行している。実行レベルは数的に小さいものほど特権性が強くなり、レベル0（特に、特権レベルとも呼ぶ）では通常の命令に加えて特権命令の実行が可能になる。

通常、レベル0ではオペレーティングシステムのカーネル部が実行され、ユーザープログラムはレベル3で実行される。

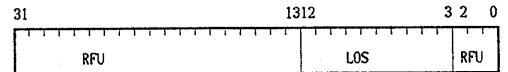
第1の保護機構はエリアに対するものである。エリアの保護はエリアテーブル・エントリによって規定される。エリアテーブル・エントリ内ではリード/ライト/実行の各アクセスタイプ毎に独立に、そのエリアをアクセスできる最低位の実行レベルが指定される。

第2の保護機構はページに対するものである。ページの保護はページテーブル・エントリによって規定される。ページテーブル・エントリ内ではリード/ライト/実行に関するアクセスができるか否かを指定する。

すべてのメモリアクセスは、エリア/ページの両方で許可された場合のみ可能になり、許可されていないアクセスに対しては例外が発生する。

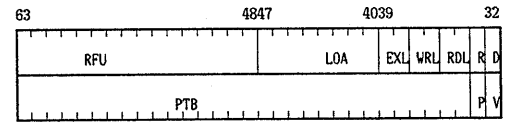


●エリアテーブルベースレジスタ (ATBR)



●エリアテーブルレンジレジスタ (ATLRS)

V Valid このエリアテーブルレジスタペアが有効か否か  
D Direction セクションの成長方向  
R Reserved for Future Use 未使用  
ATB Area Table Base エリアテーブルの実アドレス  
RFU Reserved for Future Use 未使用  
LOS Limit of Section セクションの長さ



●エリアテーブルエントリ (ATE)

V Valid このエリアテーブルエントリ (ATE)が有効か否か  
P Presence ATEの指すページテーブルが実記憶に存在するか否か  
PTB Page Table Base ページテーブルの実アドレス  
D Direction エリアの成長方向  
R Reserved for Future Use 未使用  
RDL Readable Level 保護情報：リードの許される実行レベル  
WRL Writable Level 保護情報：ライトの許される実行レベル  
EXL Executable Level 保護情報：実行の許される実行レベル  
LOA Limit of Area エリアの長さ  
RFU Reserved for Future Use 未使用



●ページテーブルエントリ (PTE)

V Valid このページテーブルエントリ (PTE)が有効か否か  
I I/O Mapped ページがI/Oマップされているか否か  
P Presence PTEの指すページが実記憶に存在するか否か  
L Locked ページがI/Oのためにロックされているか否か  
RFU Reserved for Future Use 未使用  
U User OSが自由に定義して利用できるビット  
A Accessed ページがアクセスされたか否か  
M Modified ページにライトが行われたか否か  
R Readable 保護情報：リードしてもよいか  
W Writable 保護情報：ライトしてもよいか  
E Executable 保護情報：実行してもよいか  
RPN Real Page Number ページの実アドレス

図6. アドレス変換用のレジスタとテーブル

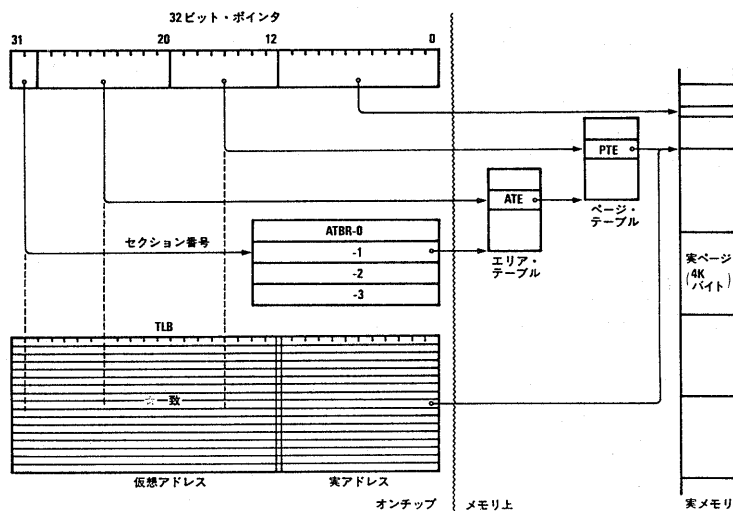


図7. アドレス変換機構

## 6. FRM機能

V60/V70には、高信頼化システムを実現する際に必要となる2重化構成や3重化構成をサポートするための、FRM(Functional Redundancy Monitoring)と呼ばれる機能を備えている。

高信頼化システムとは、ハードウェア等の故障によりシステム内のある構成要素は動作不可能になるが、その故障ユニットを冗長系により代替することで、システムの性能は低下するがシステムダウンすることなく実行を継続することのできる計算機システムである。

一般に、高信頼化システム構成を実現する時に、そのシステムでは、

- 故障の検出
- 故障部分の隔離
- 故障部分の修理・取替
- 回復処理
- 再統合

というステップを踏むことで故障からの回復を図ることになる。

現在のVLSI時代においては、マイクロプロセッサを用いて高信頼化システム構成をとるとき、プロセッサの周辺に問題を限る(メモリ系、入出力系を除く)ならば、故障および故障部分の修理・取替はマイクロプロセッサチップの故障・修理・取替ということになる。このとき、故障の検出とはマイクロプロセッサが期待どおりの動作を行うかどうかを監視し、誤動作を検出することになる。したがって、マイクロプロセッサの

チップ単位で故障検出を助ける機能がマイクロプロセッサ自身にない限り高信頼化システム構成の実現は困難である。なぜなら、複数のマイクロプロセッサからなる冗長系で、チップの全ての入力/出力の一致回路を外部に接続することで故障検出を行わなければならないからである。

V60/V70には高信頼化システムを実現するために必要となる機能をサポートしており、その機能をFRMとよんでいる。その第1の機能は、他のプロセッサの動作を監視する機能である。第2の機能はプロセッサの外部動作の凍結機能である。そして、第3はプロセッサに対する異常なしシステム構成の変更の通知機能である。

図8にV60/V70を3個用いた3重化システム構成の例を示す。V60/V70では3つのプロセッサの各端子のpin-to-pin接続だけで冗長プロセッサ構成をとり、動作の不一致を検出できるように設計してある。図8のシステム構成において、3個のプロセッサの内の1つが通常モード、他の2つが監視モードに設定され、お互に同期して並列動作している。このとき、通常モードのV60/V70は通常の命令実行を行っている。一方、監視モードのV60/V70は、外部バスを駆動せず、背中合せに接続された通常モードのV60/V70の出力信号(アドレスバス、データバス、ステータス信号)と自分自身の出力が一致しているか否かの比較を各バスサイクル毎に行う。比較が一致している限り、3つのプロセッサは正しい動作を行っているかと假定できるが、もし一致しないときはいずれかのプロセッサに何らかの故障が生じたと考えられることができる。不一

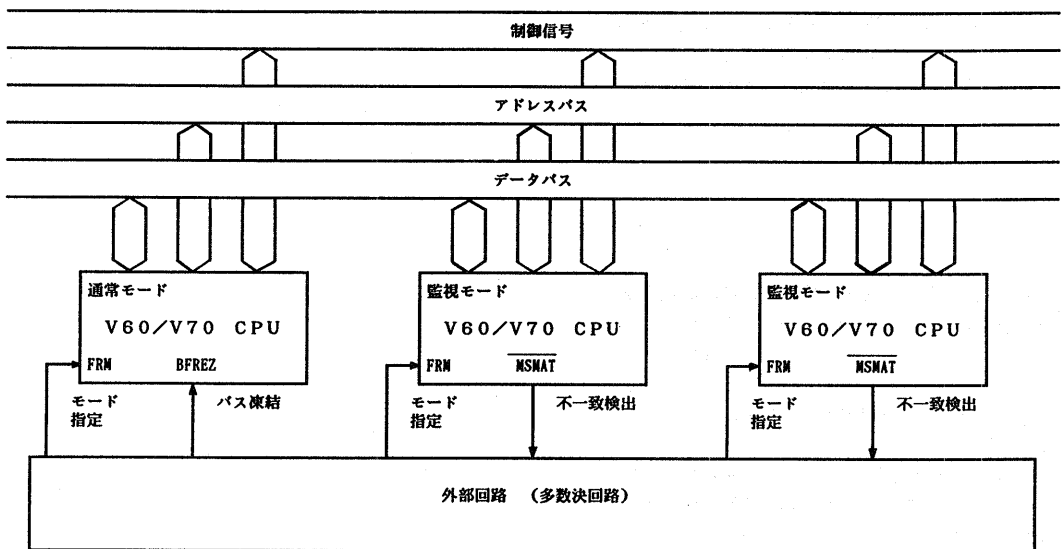


図8. 3重化システム構成の例

致が検出されると、監視モードのV60/V70は専用端子を通じ、故障処理関連の外部回路に不一致を通知する。外部回路は不一致が通知されると誤動作をしていたプロセッサのベアを外部より強制的に停止させる。これがバスの凍結（バスフリーズ）機能である。バスフリーズ状態になったプロセッサは、アドレスバスとデータバスはハイインピーダンスとなり各種の信号はインアクティブになるため、外部バスから切り離された状態になる。プロセッサのベアがバスから切り離されると、不良診断はシステムから独立に行うことができる。このとき、バスフリーズ状態のプロセッサを再スタートさせるわけであるが、V60/V70は故障が起こったときの状態からそのまま実行を継続することができる。このため、2つのプロセッサのベアの上で動作するソフトウェアにとってはあたかも異常が何もなかったように見える。

なお、V60/V70において以上の機能を実現するために要したハードウェアの量はチップ面積の約0.24%である。

#### 4. むすび

以上V60/V70のアーキテクチャの特徴について述べた。

V60/V70の応用分野としては、スーパーパソコン、EWS、大規模PBX、シングルボードコンピュータシステムなどが考えられるが、応用システムの多様化、高性能化に伴って、マイクロプロセッサに対する高速化の要求が今後一段と高まることが予想される。現在、V60/V70に引続き、同一アーキテクチャ採用でキャッシュメモリ搭載型のV80を開発中である。これは、10MIPS以上の処理能力を発揮できるであろう。

#### 参考文献

- 1) 佐藤ほか、「仮想記憶管理機構と浮動小数点機構を内蔵した32ビットマイクロプロセッサV60」、『日経エレクトロニクス』, no.391, pp.199-240, Mar, 24, 1986
- 2) 『V60/V70アーキテクチャ・マニュアル』, 日本電気, 1987
- 3) 山畑ほか、「情報処理学会マイクロコンピュータ研究会43-2, マイクロプロセッサV60のアーキテクチャ」, Feb, 6, 1987
- 4) 野原ほか、「電気通信学会技術研究報告FTS86-9~16, 32ビット・マイクロプロセッサ・V60のFRM機能

と高信頼化システムの構成」, Sep, 19, 1986

- 5) 古城ほか、「対故障機構とランデブ機能を備えるV60リアルタイムOS」, 『日経エレクトロニクス』, no.417, pp.173-201, Mar, 23, 1987