

メッセージ通信機構を備えた

4T-2

マルチプロセッサシステムの試作・評価

永松 礼夫 森下 巖 (東京大学 工学部)

1. はじめに

半導体技術の進歩によりマルチプロセッサを構築することが容易になってきた。しかし、その構成方法については多く研究の余地がある。本稿では、小規模のマルチプロセッサのためのメッセージ通信支援ハードウェア機構について述べるとともに、試作システムで明らかとなったいくつかの問題点について検討する。

2. 試作システムの概要

2.1 メッセージ・バス・コントローラ

図1のようなシングルボードコンピュータをメッセージ・バス(Mバス)とよばれる専用の回線で結合した。Mバスは16本のデータ線といくつかのコントロール線よりなる。8MHzのクロックで1ワードを転送する。現在、4台の68000プロセッサを実装している。デバッグのための共有バスとしてVMEバスも備えており、共有メモリ・システムとしても運用できる。Mバス・コントローラは図2のような構成になっており、送信用のセンダ、受信用のレシーバとメッセージバスの送信権の調停をするアービタよりなる[1]。センダとレシーバには各々1メッセージをいれるバッファがある。Mバスの上では、ひとつのメッセージを構成するデータ群はつづけておられる。すなわち、メッセージのインターリーブは行なわない。

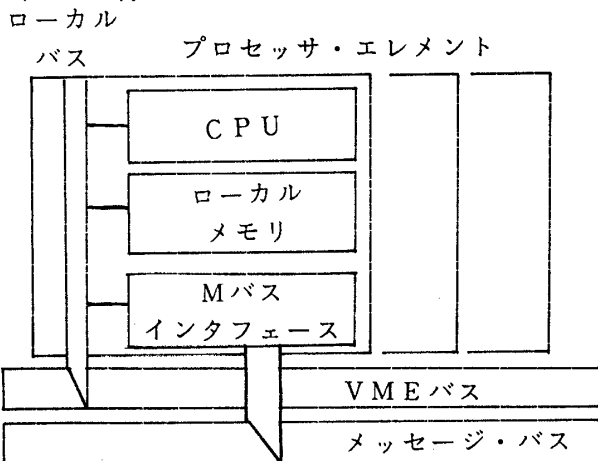


図1 システム構成

Evaluation of a message support hardware on a multiprocessor system
 Leo NAGAMATU Iwao MORISHITA (University of Tokyo)

2.2 プログラムからの起動方式

ひとつのCPUから他のCPUへMバスを使用してデータを送る際には、送信したい内容をセンダのバッファ・メモリに書いたあとでセンダを起動するコマンドをI/Oレジスタに出す。受取の場合はレシーバのバッファより各プロセスが読み出しをする。データがレシーバにあって受取の為の読み出しがまだされていない場合、同じレシーバに宛てられた他のメッセージは相手がビジーという理由で受取が拒否される。

2.3 同期機構(シンクロナイザ)

共有バス上に、図3のようなプロセッサ間の同期操作を支援するハードウェア[2]を実装した。内部はマイクロプログラム制御になっており、セマフォへのP、V命令を10サイクル程度(1サイクルは125ns)で実行する。さらに、プロセスを待たせる場合のキューの管理もおこなう。CPUからはVMEバス上のすこしウェイト・サイクルの余計にかかるI/Oとして見える。アドレスの一部を資源の識別子

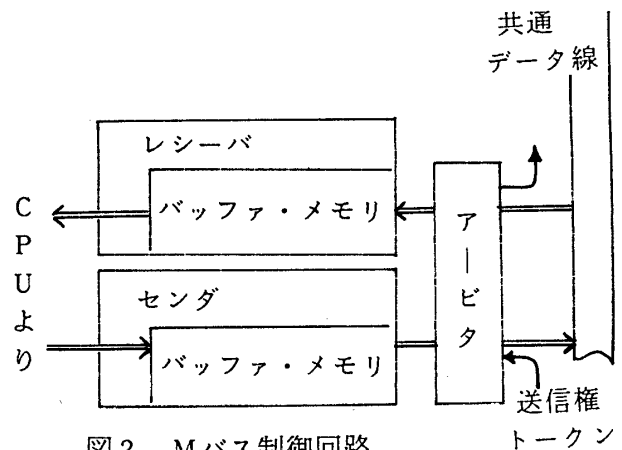


図2 Mバス制御回路

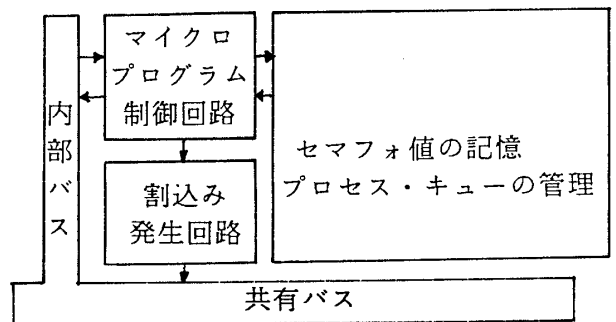


図3 シンクロナイザの構成

とし、たとえば、P命令を行なうには、そのアドレスにアクセスするだけでよい。進入が許可されれば何もおこらず、不許可であれば例外処理(バス・エラーと割込み[4])がはじまる。例外処理ルーチンでは現行のプロセスをとめればよい。マイクロプログラムを追加することでモニタの入口の排他制御ルーチンなども構成できる。

2.4 同期機構の応用

典型的なりモート・プロシジャ・コールに応用した例を図4に示す。ブロッキング・センドとその返信の所用時間は約 $700\mu\text{S}$ で、同期機構なしの場合とほぼ同じである[3]。評価に用いたOS核が異なり、主たるオーバヘッドであるプロセス履歴採取の精度が違うため単純には結論づけられない。複数プロセスがひとつのプロセッサ中にある場合、引数の中継処理と呼び/呼ばれの間隔を分離して暇なプロセッサに転送をさせたり、転送媒体を使い分けられる利点が生ずる。

3. 検討

3.1 集中の問題

Mバスは共有資源なので多くのユニットが同時に使おうとすれば、競合をおこす。しかし、Mバスの速い転送サイクルと遅いCPUのサイクルの差を吸収するためのバッファの読み書きが必須である。この読み書きはCPUのスピードでなされるので、間断なくメッセージを送ろうとしても4台程度のプロセッサではMバスを飽和させられない。

3.2 マルチ・メッセージ・バッファ

シミュレータをもちいて多段バッファの有用性について評価した。仕事量の単位を(平均命令長/転

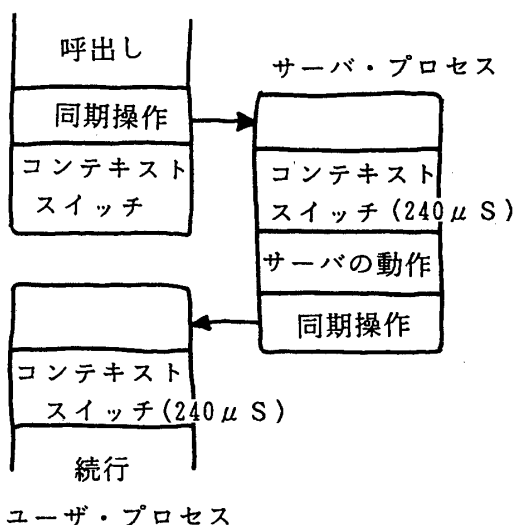


図4 リモート・プロシジャ・コール

送サイクル)として(この値を8とした)、CPU台数4台でメッセージを作り出す仕事量が50~150の一樣乱数の場合、受信バッファの最大滞留量が4、送信は最大1であった。40台でも同じであった。もちろん、すべてのCPUがひとつの転送先に集中砲火をあびせる場合はすぐに受信バッファはあふれてしまう。なお、メッセージ長は16ワードに固定とし、受信側ではひとつメッセージがくる度に、割込み(約28命令)とデータ転送(40サイクルのmov ep命令8個)が起動されバッファを速やかに空にする と仮定した。これはシンクロナイザが装備されていて、受信プロセスがすぐにデータを処理できるときにしか、実際の送信はされないという仮定である。

3.3 プロセッサに固定しないサービス

現在の実現方式では送り先はメッセージの第一語のビット・フィールドに示されていて、各インタフェース・ボードがそれを解読して取込んでいる。したがって、同様なサービスがプロセッサに依存せず利用可能な場合や、引越したプロセスに同じ送り先名を使用したい場合などに不都合があった。ポートと呼ばれる番号をサービスの識別子にして、その値で表を引く機能を付加すればプロセッサ番号と独立の送り先を用いることができる。

複数のプロセッサが関与する通信にはブロードキャスト通信もある。現状では実現していないが、これは受取バッファがビジューナ際に拒否をするメカニズムが、受信者が複数であると実現しにくいからである。

4. まとめ

試作システムでの問題を整理するとともに、今後の課題を示した。サービス・ポートを実現し、複数のメッセージをバッファに置く改良版を設計中である。

参考文献

- [1] 永松、森下：マルチプロセッサ間でのメッセージ通信オーバヘッドの短縮について、第33回情報処理学会全国大会論文集, 4C-10(1986).
- [2] 永松、神徳、ワタリ、森下：マルチプロセッサ上のイベントとプロセスの管理を行う機構、第35回情報処理学会全国大会論文集, 4C-4, pp. 153-154 (1987).
- [3] 神徳：マルチプロセッサ同期管理ユニットの設計と試作、東京大学計数工学科修士論文(1988.2).
- [4] 屈、永松、森下：セマフォ操作をサポートするハードウェア、第28回自動制御連合講演会(1985.11).