1D−7

リアルタイムマルチプロセッサシステムにおけるリカバリポイントの設定条件

峯崎 春洋, 福田 洋之, 徳永 雄一, 石田 仁志

三菱電機 (株) 情報電子研究所

1 はじめに

高度情報化社会の進展に伴い、信頼性の高いシステムへの要求が強まり、故障が発生しても正常な稼働状態を自動的に維持するフォールトトレラントシステムが要求されている。また、エンジニアリング系分野においては、リアルタイムシステムが要求されている。

本稿では、ライト - バック方式のプライベート・キャッシュメモリを備えたフォールトトレラント密結合マルチプロセッサシステムにおいて、プロセッサモジュールで故障が発生した場合にもリアルタイム性を維持できるリカバリポイントの設定条件について述べる。

2 課題

図 1のようなプラント制御の分野では、製品の高品質化、高付加価値化が要求され、これに対応するため、高度な制御技術、計測技術が必要とされている。このため、プラントの制御に直接携わるコントローラは、制御対象からの計測情報に基づき、一定時間内に制御情報を出力するリアルタイム応答性能を備えることが必要である。また、プラント制御システム全体の機能拡大、大規模化に伴い、プラントを直接制御するコントローラの故障が、システム全体の機能停止や製品の品質低下を招くため、耐故障性が要求される。

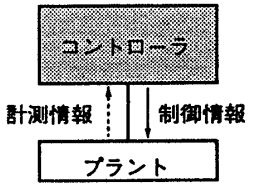

図 1: プラント制御システム

これらの要請を満たすため、プラント制御システムのコントローラの処理部で故障が発生した場合にも、リアルタイム応答性能を維持してリカバリできることが課題である。

Conditions of Recovery-Pointing for Realtime Fault-Tolerant Tightly-Coupled Multi-Processors
Shunyo MINESAKI, Hiroyuki FUKUDA, Yuichi TOKUNAGA, Hitoshi ISHIDA
Mitsubishi Electric Corp.

3 コントローラ処理部故障時のリカバリ方法

処理部の耐故障性を実現するために、図 2のような密結合マルチプロセッサシステムとする。これは、現用系と待機系のプロセッサモジュールとメインメモリモジュールがシステムバスを介して結合する。各プロセッサモジュールはライト - バック方式のキャッシュメモリを備える。現用系プロセッサモジュールは正常に動作している時にリカバリポイントを設定し、その時のプロセッサモジュールの状態をメインメモリにリカバリ情報として格納する。現用系プロセッサモジュールで故障が発生した場合、待機系プロセッサモジュールが、現用系プロセッサモジュールの正常な処理が行なわれていた時点であるリカバリポイントに遡って処理を再開する [1][2]。

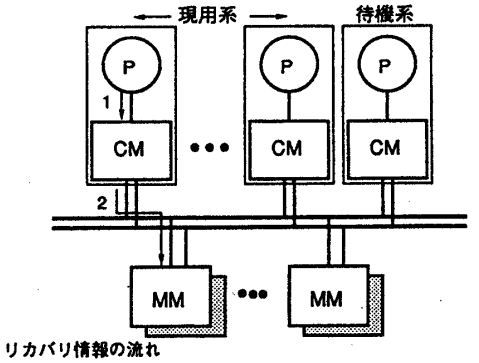

図 2: 密結合マルチプロセッサ

待機系プロセッサモジュールが現用系プロセッサモジュールの処理を引き継ぐ場合、リカバリポイントが設定された時点に遡って処理を再開するため、故障したプロセッサモジュールのリカバリ情報が、故障が検出されるまでに他のプロセッサモジュールによりアクセスされていた場合、あるいは、リカバリ情報の一部が更新されていた場合、リカバリポイントにおけるリカバリ情報の整合性が維持できない。

従って、以下の条件でリカバリポイントを設定する必要がある。

- リカバリポイントにおけるリカバリ情報の整合性を保証するための条件
  - キャッシュメモリ内の更新されたキャッシュブロックが、メインメモリのデータとリプレースされる場合。

– キャッシュメモリ内で更新されたキャッシュブロックが、他のプロセッサモジュールにより参照される場合。

また、リカバリ情報の1つであるプロセッサの内部レジスタの内容は、キャッシュメモリに格納後、キャッシュメモリ内でローカルに更新されたデータと共にメインメモリに格納される。

従って、さらに、以下の条件でリカバリポイントを設定する必要がある。

- キャッシュメモリ内にプロセッサの内部レジスタの内容を格納する空間を確保するための条件
  – キャッシュメモリ内の各キャッシュラインで更新されていないキャッシュブロックの数が、予め決められた値に達した場合。

4 コントローラ処理部故障時のリアルタイム応答性能を保証する方法

図3に示すように、現用系プロセッサモジュールは、正常な処理を実行している間リカバリポイントを設定する。現用系プロセッサモジュールにおいて故障が発生した場合、待機系プロセッサモジュールが、故障プロセッサモジュールの最も近いリカバリポイントにおけるプロセッサの内部レジスタの内容を取り込み、処理を再開する。

図3より、リカバリ時間 T_r は、

$T_r = t_2 + t_3 + t_4$ .....(A)

$t_2 \leq t_1 + t_c$ .....(B)

t_1：リカバリポイントの設定に要する時間 [3]

t_2：リカバリポイント設定終了後から故障発生までの時間

t_3：故障検出に要する時間 数 μs[4]

t_4：リカバリ処理に要する時間 数百 μs

t_c：リカバリポイントの設定間隔

である。

ここで、コントローラが制御対象からの計測情報の入力後、制御情報を出力するまでの許容時間を T、制御のための処理時間を t_0 とし、処理時間 t_0 の間にリカバリポイントを n 回設定すると仮定すると、故障発生時にもシステム許容時間 T を保証するためには、

$T \geq (n+1)(t_1 + t_c) + t_3 + t_4$ .....(C)

$t_0 \leq nt_c$ .....(D)

が成立しなければならない。

よって、故障発生時にもシステム許容時間 T を保証するためには、リカバリポイントの設定においてメインメモリに格納されるリカバリ情報の容量に限度を設け、リカバリポイントの設定に要する最大時間 $t_1(max)$ を定めると共に、リカバリポイントの最大設定間隔 $t_c(max)$ は、リカバリ情報量が最大で $t_1(max)$ 時に式 (C),(D) を満足する時間でなければならない。

従って、図2のような密結合マルチプロセッサシステムにおいて、プロセッサモジュールの故障時にもリアルタイム応答性能を維持するためには、さらに、以下の2つの条件でリカバリポイントを設定する必要がある。

- リカバリポイントの設定に要する最大時間 $t_1(max)$ を規定するための条件
  – キャッシュメモリ内で更新されたキャッシュブロックの数が予め決められた値に達した場合 [5]。
- リカバリポイントの最大設定間隔 $t_c(max)$ を規定するための条件
  – タイマ割り込みによる場合。4つの設定条件の内、いずれかの条件でリカバリポイントが設定されるとタイマをリセットする。

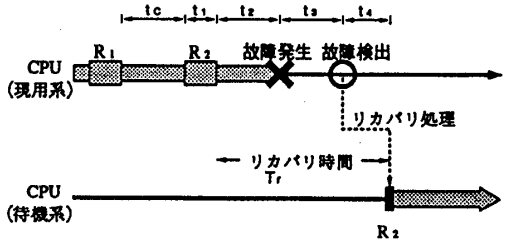

図 3: リカバリ処理

5 おわりに

図2のような密結合マルチプロセッサにおいて、以上述べた4つの条件のうちいずれかが満たされた時リカバリポイントを設定することにより、プロセッサモジュールの故障時にもリアルタイム応答性能を保証することができる。今後、これらの条件を図2のような密結合マルチプロセッサシステムに適用するために、キャッシュメモリの容量、リカバリ情報の容量及びリカバリポイントの設定間隔の適性値を解析する必要がある。

参考文献

[1] 当麻 喜弘：フォールトトレラントシステム論, 電子情報通信学会, 1990 年

[2] K.M.Chandy： A Survey of Analythic Models of Rollback and Recovery Strategies, IEEE, COMPUTER, May 1975, pages 220-232

[3] 福田 洋之, 石田 仁志, 徳永 雄一, 峯崎 春洋：フォールトトレラントシステムのメモリ制御, 情報処理学会第 44 回全国大会, 1992 年

[4] 小山田 英夫, 志賀 稔：マルチプロセッサシステムにおけるプロセッサの誤り検出法, 情報処理学会第 44 回全国大会, 1992 年

[5] 石田 仁志, 畑下 豊仁, 志賀 稔：スタックを用いたキャッシュ・フラッシュの高速化, 電子情報通信学会春季全国大会, 1991 年