

鉄鋼計算機制御システムにおける 自律分散システムの適用

Autonomous Decentralized Systems for Computer Control Systems of Iron and Steel Plants

鉄鋼業では、従来の少品種多量生産から多品種高効率生産への生産形態の変更などにみられるように操業側の変化は激しく、計算機制御システムもそれに対応できるものが要求されている。また最近の経済情勢から設備投資の削減は避けられず、いかにして最小の投資で最大の効果を得るようになるかということが、計算機システムの開発担当者の使命でもある。そのため建設当初は必要最小限のシステムから出発し、稼動後の操業の変化に応じてハードウェア及びソフトウェアを容易に追加してゆけるような計算機システムが望まれた。このようなニーズに対応するため、自律分散コンセプトによる新しいシステムアーキテクチャを開発し、鉄鋼システムに適用した。その結果、拡張性の向上、保守性の改善だけでなくソフトウェアの生産性を向上させることができた。

森 淳* *Jun Mori*
鳥越英俊** *Hidetoshi Torikoshi*
中井耕三*** *Kôzô Nakai*
森 欣司**** *Kinji Mori*
増田崇雄***** *Takao Masuda*

1 緒 言

日立製作所は、鉄鋼計算機制御システムに対する拡張性、保守性へのニーズを満たすため、新しいシステム技術の構築と、それに基づく新しいソフト技術の開発を進めている。本稿は、上記ニーズに対応するために取り入れた自律分散システム技術と、これに基づく新制御用計算機V90/5シリーズの新しいシステムソフトACP(Autonomous Control Processor)及びこれを適用した鉄鋼計算機制御システムについて論述したものである。

最近、計算機技術に大きな革新があった。その第一は、ハードウェア面でLSI技術及び光通信技術である。前者はプロセッサやメモリのコストを急激に低下させ、高性能、高機能のマイクロコンピュータによる新しい計算機を適用できるようにした。後者は、これら計算機を機能的に分散させるとともに、全体として協調をとるための有効な情報連絡手段を提供した。これらのことが、従来の中央集中処理システムから拡張の容易な分散処理システムへの展開を可能とした。第二は、システムソフトウェア面での標準化である。特に、UNIXTMのような会話形OS(Operating System)はユーザー指向の計算機システムの構築を容易にした。このような技術動向を踏まえて、増設改造の容易な、より使いやすい鉄鋼計算機制御システムを実現しようとしたのが自律分散システム採用の発端である。次に、計算機システムに対するニーズをユーザー

側から見た場合及びメーカー側から見た場合で分析してみた。ユーザーニーズとしては、第一にシステムの既存部分の変更や、機能の拡張が全体システムを停止しなくてもできること(段階的なシステム構築)、第二はシステムの部分的な故障に対してもシステム全体を停止させることがないこと(ノンストップコンピュータシステム)、第三はシステム稼動中に保守ができること(オンライン保守)である。これら三つのユーザーニーズは、従来の中央集中処理システムでは達成するのが難しいものであった。次に、メーカー側のニーズとしては、特に顧客の仕様に基づき作成するアプリケーションソフトに関して、第一に計算機ハードウェアの違いに影響されないソフトを作成できること(機種異なる計算機にも利用できるソフトウェアとすることによるソフト生産性の向上)、第二に計算機システムのシステム構成が変化しても、アプリケーションソフトは変更する必要がないこと(システム構成に依存しないソフトによる拡張性の向上)、第三にアプリケーションソフトウェア間でのデータの受渡しが、共通のファイルによらずメッセージ通信の形で行われること(データ駆動形ソフトによる保守性の向上)がある。上記のような最近の技術動向とシステムへのユーザー、メーカー両者からのニーズに対応できるような新しいシステムアーキテクチャとして自律分散システムを適用した。

※) UNIXTMは、米国ATT社の登録商標である。

* 川崎製鉄株式会社千葉製鐵所設備技術部 ** 川崎製鉄株式会社水島製鐵所電気計装部 *** 日立製作所大みか工場
**** 日立製作所システム開発研究所 工学博士 ***** 日立製作所機電事業本部

2 自律分散システム

2.1 自律分散システムの処理構造

自律分散システムで基本となる概念は、データが流れる場であるDF(Data Field)の概念である。自律分散システムでは、各アプリケーションプログラムはDFに論理的に接続されており、これらアプリケーションプログラム間のリンケージは、完全なデータフロー形となっている。すなわち、各アプリケーションプログラムは、DFを流れるメッセージの中から自らに必要なメッセージを用いて処理を実行し、その出力をDFに送出するだけであり、アプリケーションプログラムは自分の果たすべき機能だけを考慮して作成することが可能である。自律分散システムの具体的な処理構造を図1に示す。ここで、新しく開発された自律分散システムのリアルタイムOSであるACPはDFとのデータ送受信、アプリケーションプログラムの起動を集中管理しており、アプリケーションプログラム内の異常検知と異常データの排除など、障害波及阻止の管理機能を具備している。このような処理構造によって、自律したアプリケーションプログラム(以下、機能モジュールと呼ぶ。)はDF上のどこに、いつ接続されても直ちに機能できる。すなわち、システム稼動中に新機能の増設や不具合機能の切り離しが現実的に可能となり、テストや保全まで含めたシステム開発環境は飛躍的に向上する。

以上述べた処理構造を、次の二つの機構によって実現している。

(1) データ駆動

各機能モジュールが必要とするメッセージがDFにそろった時点で起動する機能

(2) 機能コード通信

DF上を流れる各メッセージに、機能コードと呼ぶメッセー

ジ内容を区別するためのコードを設定し、機能モジュール間のメッセージ通信のプロトコルをこの機能コードに基づくプロトコルに統一する機能

これによって、機能モジュール間のインタフェースをネットワーク構成に依存しないものとした。このことは、同一CPU内の機能モジュール間リンケージと、他のCPUにある機能モジュールとのリンケージを全く同一としたことを意味している。

2.2 機能コード通信方式

自律分散システムの基本となるのはDFの概念であり、このDF上でのメッセージ通信方式である機能コード通信方式について、その概念図及びメッセージフォーマットを図2に示す。

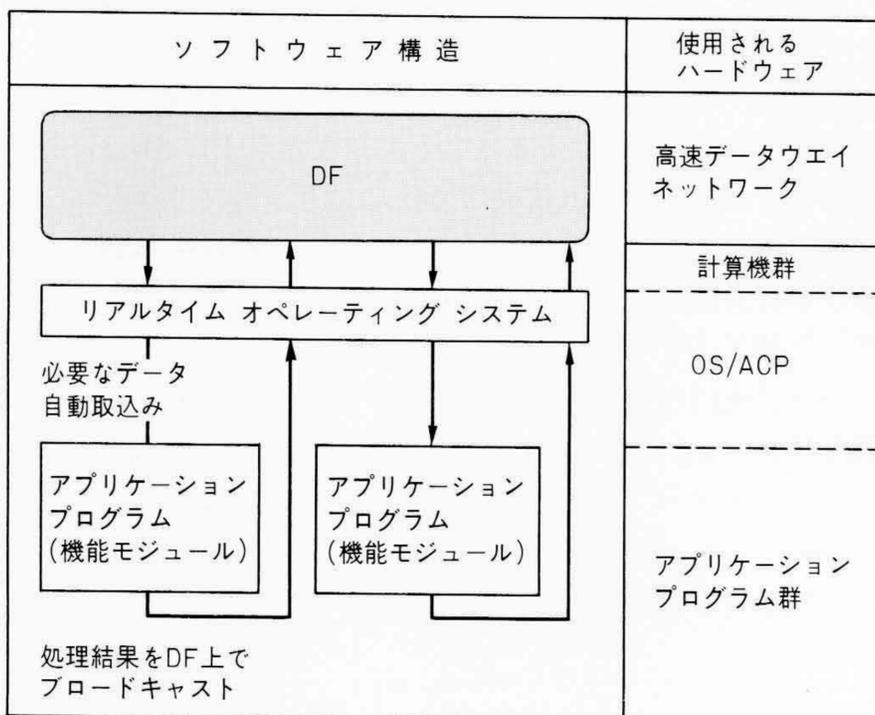
機能コード通信の管理機能は、NCP(Network Control Processor)とACPに分担させている。

(1) ブロードキャスト

自律分散システムでは、各機能モジュールはその出力データに、受取先を指定するあて先アドレスではなく、データの内容を示す機能コードを付加してACPに渡す。ACPは、機能モジュールから受け取ったデータをDFへ(具体的にはネットワーク上のメッセージフォーマットに変換し)NCP経由でネットワークに送信する。ここでメッセージはネットワーク上のすべてのNCPにブロードキャストされる。

(2) 選択受信

自律分散システムでは、各機能モジュールのACPへの登録は、各機能モジュールの入出力データの機能コードを登録することによって行う。ACPは、登録された各機能モジュールの入力データの機能コードをNCPに登録する。各NCPは、ネットワークから受信したメッセージに付加されている機能コードと、自分自身に登録されている機能コードが一致したメッセージだけを自分自身に接続されているCPU(Central Processing Unit)に送る。以上の処理によって、各CPUがDF上を流れるメッセージの中から自分自身に必要なメッセージだけを選択し受信することが可能となる。



注：略語説明 DF(Data Field)
OS(Operating System)
ACP(Autonomous Control Processor)

図1 自律分散システムの処理構造 自律分散システムでの処理構造を示す。

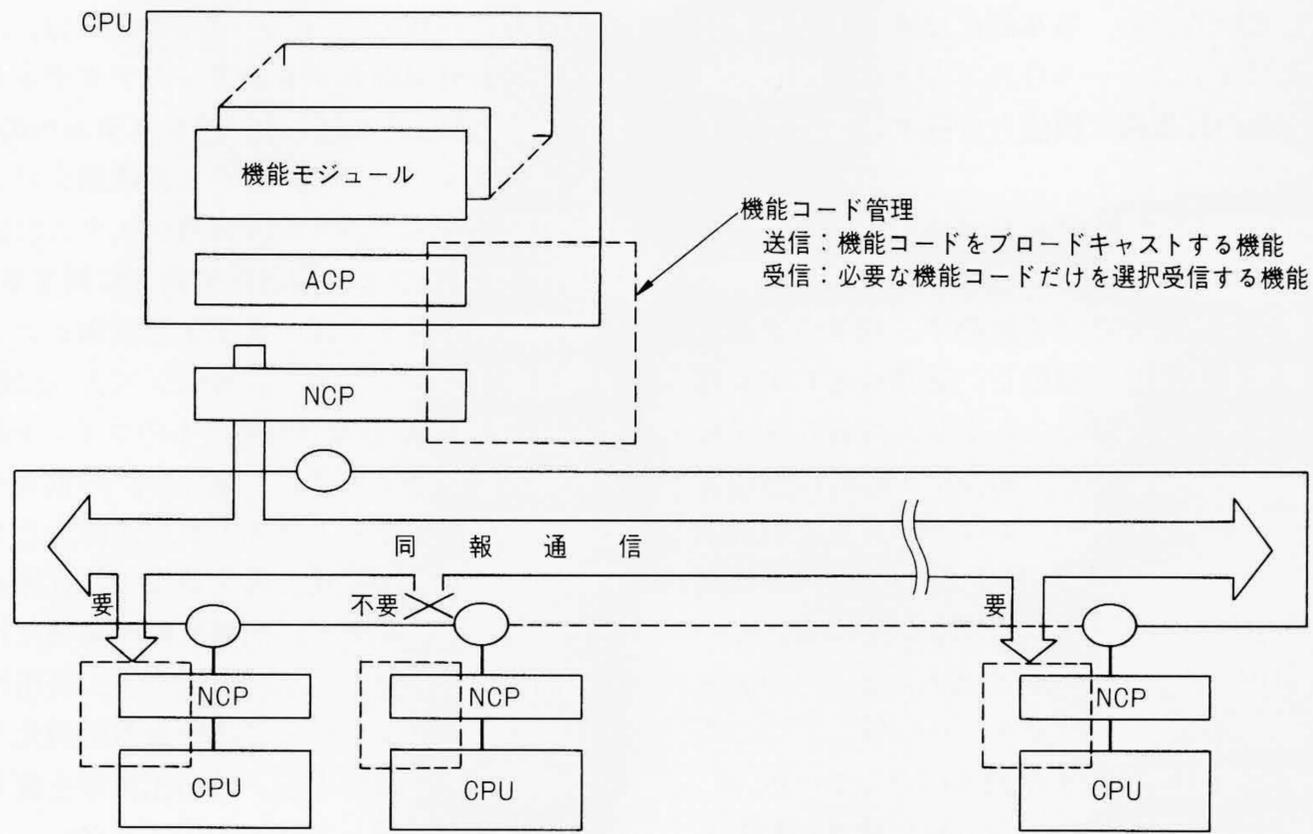
3 鉄鋼計算機制御システムへの適用

3.1 鉄鋼自律分散システムの特長

鉄鋼制御システムに適用した自律分散システムの特長を、従来システムと比較して図3に示す。

(1) システム構成

鉄鋼システムでは、従来から機能の分散、保守の集中という考え方であり、早くからネットワークシステムが適用されてきた。しかし、サテライトステーションの機能は単に端末機器(I/O：入出力装置)のドライブ処理程度であり、基本的には中央の計算機に集中したデータベースを持ち、中央からすべての端末をリモートコントロールする中央集約的なシステムアーキテクチャであった。これに対して自律分散システムでは、I/Oの処理に専念するI/Oコントローラと呼ぶCPUを持ち、各I/Oで高速な応答性を要求される処理のほとんどを吸収するようにしている。更にアプリケーションプログラムは、前章でのような機能モジュールとして単一インタフェースで



メッセージのフォーマット

機能コード	制御情報	データ
-------	------	-----

注：略語説明 NCP(Network Control Processor)

図2 機能コード通信方式 機能コード通信方式は、各CPUが必要とする機能コードをあらかじめNCPに登録し、その選択受信はNCPで行う。

項目	従来システム	自律分散システム
ネットワーク		
タスク間リンクエージ		
ファイルアクセス		

注：略語説明 I/O(入出力機器)

図3 鉄鋼自律分散システムの特長 従来システムと自律分散システムのアーキテクチャの比較を示す。

作り上げるようにしているため、基本的にはネットワークに接続されたどのCPUにインストールされてもよく、増設改造が容易なように複数台の計算機に機能を分散する構成としている。

(2) アプリケーションプログラム間リンケージ

アプリケーションプログラムの処理単位であるタスク間のリンケージで、従来システムでは必ず起動する相手のタスク名あるいはタスク番号を使用して起動し、通常は起動する前に相手に渡すべきリンケージ情報をあらかじめ共有ファイルに書き込んでから起動をかける方式であった。これに対して自律分散システムでは、アプリケーションプログラムの処理単位(機能モジュール=タスク)は処理した結果のデータに機能コードを付けて、ACPに渡すだけで処理を終了する。このとき、このデータをDFに流すのはACPであり、またこのデータ(機能コード)を必要とする機能モジュールを捜して、このデータを渡すのもすべてACPが行う仕組みになっている。したがって、機能モジュール側からは、一切DFに書き込むことも参照することもできないようになっており、このことが各機能モジュールの独立性を高めることになり、ソフトウェアの増設改造を容易にしている。

(3) ファイルアクセス

従来システムで、各アプリケーションプログラムは、共通のデータベース(ファイル)を持ち、これを読み書きして処理を行っている。したがって、ファイルの競合を防ぐため、読み書きの際ファイルのリザーブフリーの処理を必要とした。これに対し自律分散システムでは、自らが必要とするデータは、あらかじめ自分自身のファイル(これをプライベートファイルと呼ぶ。)にDFから取り込んでおく方式のため、リザーブフリーは必要としない。このことが、上記(2)と同様各機能モジュールの独立性を高め、機能モジュールの増設改造を容易にすることになった。一方、従来のように、すべての情報が一つのデータベースから参照できるという容易さと、共通情報が一元的に管理できるメリットを犠牲にするおそれを生じた。これに対処するため、従来のアプリケーションプログラムのタスク構成、及びデータベースの見直しを行い、1タスク1

ファイル(一つのファイルへのアクセスは、一つのタスクしか行わない。)を原則とする新アーキテクチャとした。

3.2 冷間圧延プロセス計算機システムへの適用

自律分散システムは、まず主要鉄鋼システムの一つである冷間圧延工場のプロセス計算機システムに適用された。

3.2.1 ソフトウェア開発効率向上に対する問題点

自律分散システムは、まず主要鉄鋼システムの一つである冷間圧延工場のプロセス計算機システムに適用された。酸洗ライン、圧延ラインなどの幾つものラインを持つ冷間圧延工場プロセス計算機システムの建設は、対象ライン数が多く個別設計ではソフトウェア開発の負担が膨大となる。また、システム開発要員の効率化と保守要員の増員抑止は、焦び(眉)の急を要する問題であり、制御技術の蓄積だけでなくソフトウェア高生産化を目的とした、システム利用技術の開発を必要としている。ソフトウェア開発量の低減化とソフトウェア開発の効率化を主眼とする、効率化対策を表1に示す。

プロセス計算機システムのアプリケーションプログラムの移植性、流用性を妨げる問題点は三つある。

(1) 装置依存性(Device Dependency)

プロセス、設備を対象とするプロコンの位置づけから、電気、計装とのインタフェース、すなわち各種のシーケンス信号やセンサ信号との接続が不可避であるが、その配置及び接続形態はプラントごとに独自に決定される。よって対象設備に依存した仕様になりがちである。

(2) システム構成依存性(System Configuration Dependency)

メーカー標準の支援プログラムはI/Oに対する動作制御を主目的に設計されており、I/O異常時の代替処理、共有I/Oの排他処理及びオンラインとオフラインのシミュレーション処理などのシステム運用に関する処理は、アプリケーションプログラムに依存する場合が多い。したがって、システム構成に依存し、同一構成でなければ、アプリケーションプログラムは流用できない結果につながる。

(3) イベント駆動形(Event Driven)

対象プロセスのシーケンス進行に沿ったデータ収集、設定

表1 ソフトウェア開発の効率化対策 ソフトウェア開発量の低減化とソフトウェア開発の効率化を主眼とする効率化対策を示す。

No.	目的	対策	ねらいのレベル
1	ソフトウェア開発量の低減化	事業所間での技術トランスファ推進	機能仕様・制御モデルの流用
		ソフトウェア流用率の向上 (1) ソフトウェアポータビリティの追求 (2) ソフトウェア共通設計化	現在の応用プログラムの問題点、装置依存性・システム構成依存性、イベント駆動形プログラムフローからの完全解放と、ソフトウェアの部品化(共通部品の確保)
2	ソフトウェア開発の効率化	ソフトウェア開発アーキテクチャの活用 (1) 全社ドキュメント標準化 (2) ソフトウェア品質の向上	アプリケーションプログラムのデータフロー化、及びインタフェースの統一化で構造設計の信頼性向上
		ソフトウェア開発環境の整備 (1) 系列別メーカー・機種統合 (2) オンラインテスト環境の整備	ソフトウェア部品の流用性を確認するため、オンライン稼動中に並行運転・テスト可能なシステム構築、同時に障害波及阻止の対策がOSレベルで確立

制御、操業ガイダンス、データ伝送などの処理を実行する(イベント駆動)ため、事象の待ち合わせ、タイマ起動などのタイミング制御を必要とする。すなわち、これは設備の運転法案に大きく依存し、かつトリガ信号はプラントごとのインタフェース条件によることを意味する。

3.2.2 冷間圧延工場プロコンシステム基本方針

冷間圧延システムの建設に際しては、主要目標として下記三つの項目が挙げられた。

- (1) 建設、拡張及び保全を統括したシステムコストの低減
- (2) ソフトウェア移植性、流用性の向上によるソフトウェアコストの低減
- (3) オンライン稼働中のソフトウェア開発環境とオンライン保守性の改善

目標達成の具体的なシステム構築の指針は次のものとした。

(a) 将来、冷間圧延工場の全プロセス計算機システムが統括管理可能なシステム構築

自律分散システムと高速データウェイ(光ネットワーク)採用によるハードウェア、ソフトウェアの共有資源とシステム拡張の柔軟性確保

(b) 冷間圧延プロコンのソフトウェア開発量の低減化

自律分散システムの採用によるソフトウェア移植性、流用性の向上につながる構造設計

(c) ソフトウェア開発環境の整備とオンライン保守性の改善

善

開発システムを活用したオンラインテスト環境の充実
シミュレータシステムを利用したオンラインシミュレータの実施

(d) システム建設、保守要員の少数化

ハードウェア管理の集中化、ソフトウェア管理の一元化
ソフトウェア開発量の低減化と開発の効率化(上記(b), (c)項の実現)

特に、アプリケーションプログラムの問題点を解消し、従来全く試みられなかったプロコンソフトウェアの移植性、流用性向上を推進するため、今回開発したI/Oコントローラで動作する入出力サーバの概念を図4に示す。

3.3 システム評価

自律分散システムを冷間圧延プロセス計算機システムに適用して得られたシステムの評価として下記があげられる。

3.3.1 拡張性の評価

図5(a)は、今回適用したコールドタンDEMミルと酸洗ラインのシステムで、段階的にシステムの増設を行った例である。この例は、タンDEMミルシステムが先に稼働している状態で、酸洗ラインのシステムを段階的に機能拡張していったものである。この拡張は、同一ネットワーク上へマイクロコンピュータを順次接続していくことで行われた。すなわち、図5(b)の従来システムの場合、各計算機がネットワーク以外に、例

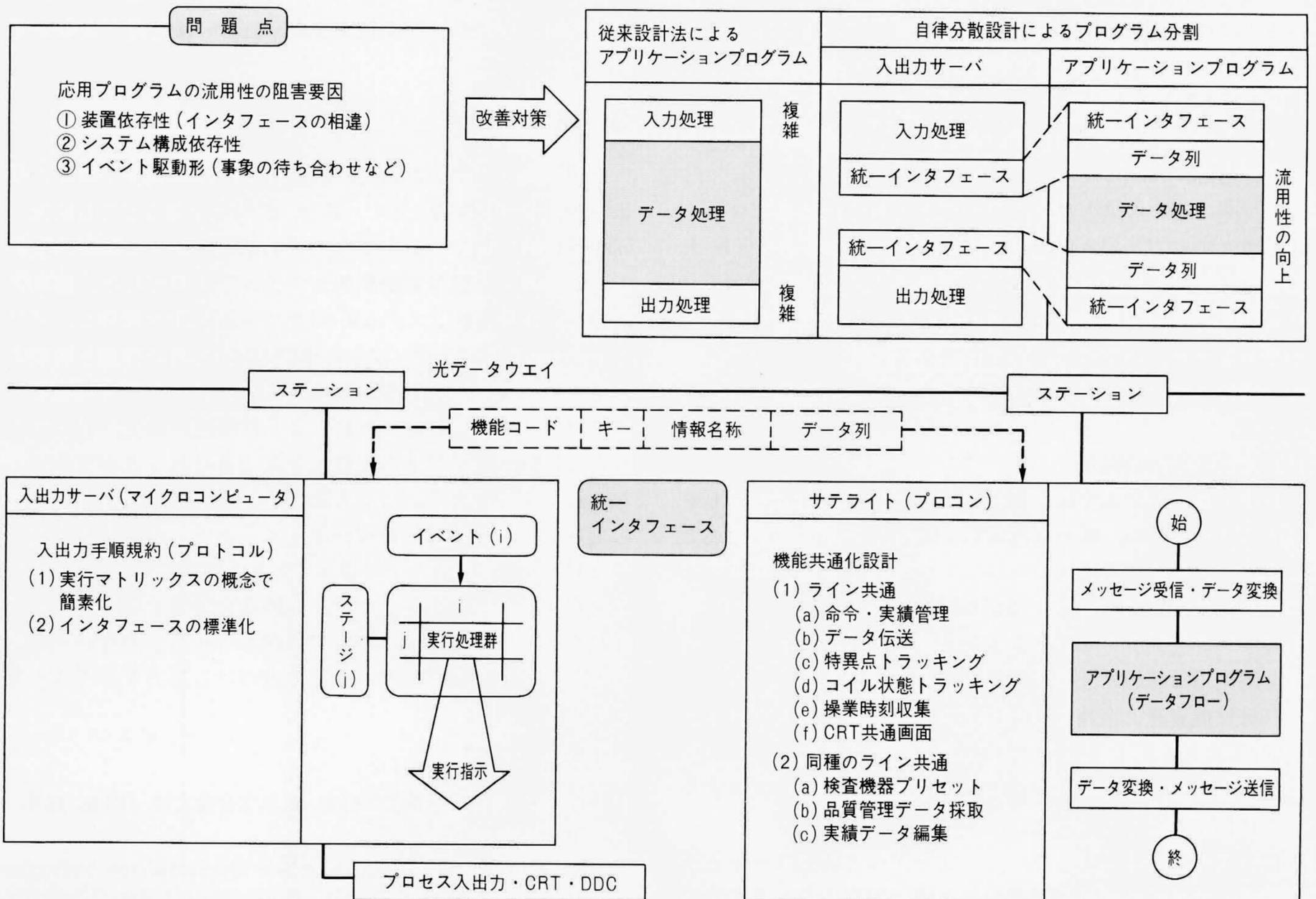
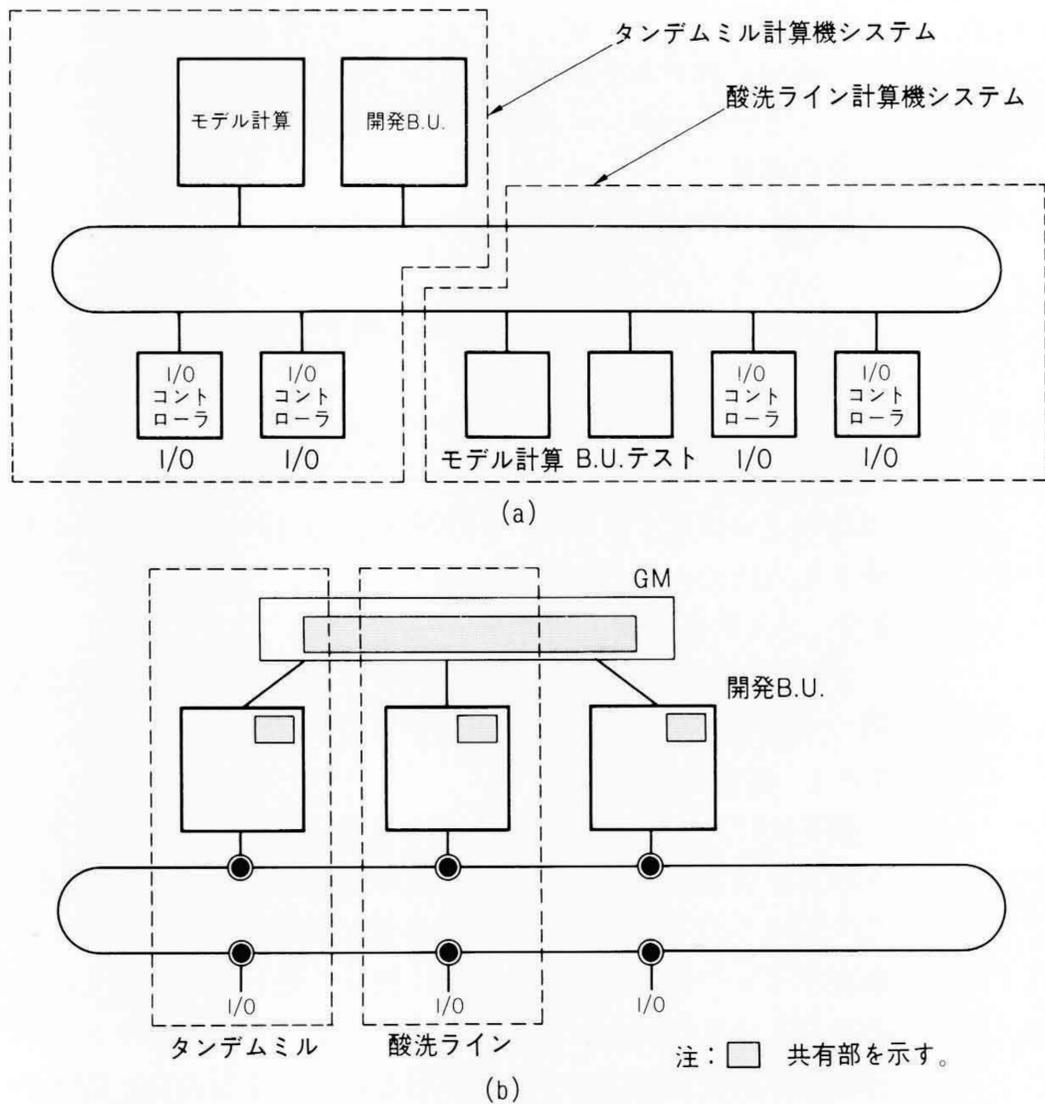


図4 入出力サーバの概念図 アプリケーションプログラムと統一インタフェースで接続される入出力サーバの概念を示す。



注：略語説明 B.U.(Backup Computer), GM(Global Memory)

図5 コールドタンデムミルシステムへの適用 自律分散システムを適用したコールドタンデムミルシステムと従来システムとの比較を示す。

えばGM(Global Memory)といった共有メモリでも接続されているのに対し、図5(a)で、各計算機を接続しているのはネットワークだけである。このことが段階的な拡張作業を容易にした。

3.3.2 開発性の評価

ソフトウェア開発での自律分散システムの特長は、次のとおりである。

- (1) 従来の集中方式と違い、プログラム単位に独立のファイルを確保することで、自己完結的に設計が進められる。
- (2) 単体テスト及び結合テストの開発途中の段階で、ハードウェア環境の影響を受けることなく、テストを実施することができるため、効率のよい作業が行える。
- (3) 情報入出力がデータフローのため、結果の確認が容易に行える。

3.3.3 保守性の評価

自律分散システムでは、情報が全部の計算機へブロードキャストされるため、開発用計算機で特別な処置を講ずることなく、オンラインシミュレーションが実施できる。具体的な効果としては、次のような点が挙げられる。

- (1) オンラインシミュレーション機能によって、被テスト機能を開発用計算機に登録することで容易にテストできる。
- (2) 計算機資源に余裕がなくなったときには、従来は該当計算機にメモリを増設するなどの処置を講じてきた。自律分散システムでは、その時点で余裕のある計算機へ容易にプログラムを移植することができる。
- (3) システム全体のインタフェースがすべて機能コード方式のため、すべての情報が開発用計算機で収集でき、トラブルの解析が容易に実施できる。

以上、実プロセスへの適用によって、自律分散システムの導入では、拡張性、開発性及び保全性といった面で効果が認められた。

4 結 言

自律分散システムの研究はまだ始まったばかりである。将来の注目すべき技術として、学会、国内外企業などでも研究、開発が始められている。本システムはこのような動向のなかで、いち早く鉄鋼計算機制御システムで実用化した。適用に際して、自律分散システム用のOSであるACPを開発し、アプリケーションソフトウェアから煩雑なI/O処理の作業を解放するためと、I/Oの処理を自律させるI/Oコントローラを開発し、実プロセスに適用した。これにより拡張性の向上、保守性の改善だけでなくソフトの生産性を向上させることができた。

これまでのシステムは、人間の高度の情報処理能力を目指してきた。これに対し自律分散システムは、システムをもう一度見直し、生物としての根元である「生きているシステム」を実現するものである。今後、人間的な要素を生物学からの知見をもとに取り込み、単なる「頭でっかち」でない「均整のとれたシステム」へといっそう近づける努力をしてゆく考えである。

参考文献

- 1) 森, 外: 自律分散概念の提案, 電気学会論文誌, 104巻, 12号, 303~310(昭59-12)
- 2) K. Mori, et al.: Autonomous Decentralized Software Structure and Its Application, Fall Joint Computer Conference Nov. 1986 pp.1056~1063.(ACM and IEEE)