プロセスデータ処理記述言語とそのリアルタイムパッケージ群―PDMS―

Process Data Description Language and Its Realtime Software Packages — PDMS—

計算機制御システムで「制御対象プロセスと計算機とのインタフェース(プロセス入出力インタフェース)」は、システムの中枢とも言うべき重要な部分である。近年のハードウェアの技術革新とシステムの制御規模・適用範囲の拡大に伴い、この「プロセス入出力インタフェース」も複雑化・多様化しており、プロセスデータ処理アプリケーションの生産性・信頼性向上の隘路となっていた。

日立製作所ではこのような状況に対処するために、アプリケーションシステムに統一的かつ普遍的な論理プロセス入出力インタフェースを与える「バーチャルシステムアーキテクチャ」、及びこの論理的なインタフェースの上で効率的なテストを可能とする「プロセスシミュレーション機構」を具備したプロセスデータ処理の標準ソフトウェアPDMSを開発した。本PDMSでは高性能のハードウェアを生かした独自の性能調整機構により、システムごとに異なる要求性能の実現を図っている。

大脇隆志* Takashi Ôwaki 大島啓二* Keiji Ôshima 林 利弘* Toshihiro Hayashi 各務博之** Hiroyuki Kakamu

1 緒 言

近年のハードウェアの技術革新により、端末I/O(入出力装置)のインテリジェント化、システムの分散化など、計算機制御システムの構成は複雑・多様化している。更に、計算機の処理能力の向上によって、制御対象プロセスの範囲・量も増大してきている。

これに伴い、計算機制御システムの中枢とも言うべき「制御対象プロセスと計算機とのインタフェース」も複雑・多様化している。

このような状況に対しては、統一した思想のもとに、普遍的なプロセス入出力インタフェースを提供するとともに、プロセスデータにかかわるデータ処理を標準的にサポートすることが不可欠である。 PDMS (Process Data Management System)は、このような背景のもとに開発したプロセスデータ処理の標準ソフトウェアである。

PDMSのねらいは、

- (1) プロセスデータ処理アプリケーションプログラムの作り やすさ、保守のしやすさの向上
- (2) プロセスデータ処理として必要なリアルタイム性能の確保の双方を同時に実現することであり、これらによって、アプリケーションソフトウェアの生産性、信頼性及び保守性の向上を図るものである。

本稿では、PDMSの開発思想、方針、構成、機能及び適用 状況について述べる。

2 PDMS開発の基本思想と方針

PDMSでのプロセス入出力インタフェースの標準化に当たっては、生産性、信頼性及び保守性の向上をねらいとして、

- (a) バーチャルシステムアーキテクチャ^{1)*)}
- ※) ハードウェアデバイスの物理的な構造や制御方式からアプリケーションプログラムを解放し、かつプログラムとは独立に性能調整を可能とすること。

- (b) 統合化・体系化システムアーキテクチャ を,また,性能向上をねらいとして,
- (c) リアルタイムシステムアーキテクチャ を、それぞれ実現することを開発の基本思想とした。これら

の基本思想を具現化するためのPDMSの開発方針は、以下に述べるとおりである($\mathbf{図1}$)。

- (1) プロセスデータ処理の環境仕様記述と手続き記述とを分離し、各々を担当者の専門分野に合った論理レベルのインタフェースで記述できるようにすること。
- (2) テストシステムとの統合化を図り、特にプロセスシミュレーションテスト機能を強化すること。

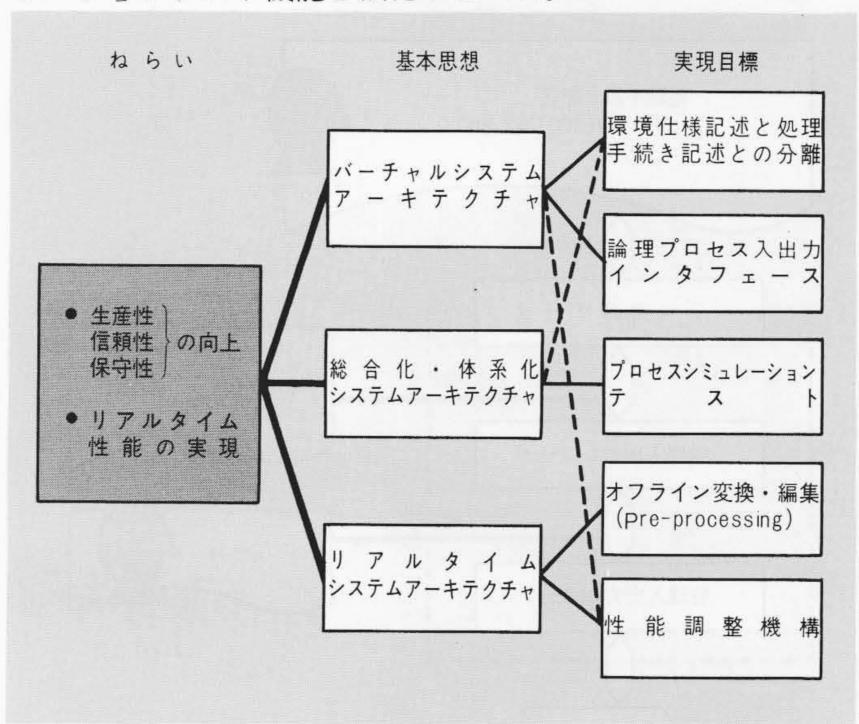


図 I PDMS開発の基本思想と方針 PDMS(Process Data Management System)は、「バーチャルシステムアーキテクチャ」、「統合化・体系化システムアーキテクチャ」及び「リアルタイムシステムアーキテクチャ」を基本思想の3本柱として開発した。

^{*} 日立製作所大みか工場 ** 株式会社日立コントロールシステムズ

(3) 事前処理の徹底によるプロセスデータ処理の高速化,及び性能調整機構による多様なリアルタイム性能要求の実現を図ること。

これらの開発方針に沿ったPDMSの具体的な実現目標を, 以下に説明する。

- (1) 環境仕様定義機能と論理プロセス入出力インタフェース計算機制御システムで、アプリケーションプログラムが扱うプロセスデータは、温度・流量、操作器のリレー信号といった、PV(Process Variable:プロセスの状態量)やLV(Logical Variable:制御情報)である。通常、これらのプロセスデータは、センサ、工業計器及びPI/O(プロセス入出力装置)といった複雑なプロセス入出力インタフェースを介して計算機に取り込まれる。したがって、これらプロセスデータを用いて処理を行なうアプリケーションプログラムを正しく設計・製作するためには、
 - (a) 制御対象プロセスについての専門知識・ノウハウ
 - (b) センサ・工業計器を含む計装システムの専門知識
 - (c) PI/Oなど計算機ハードウェアの専門知識

が必要とされていた。従来、プログラマやシステム設計者は、これら質の異なる多種の知識を同時に要求されていたため、(i)知識不足に伴う仕様不一致などによる作業の手戻りが発生する、(ii)設計は上級のプログラマ、システム設計者に限定される、(iii)プロセスとの煩雑なインタフェースを意識しなければならないため、プログラムの作成効率が低下する、といった問題が発生していた。

このような問題を解決するためにPDMSでは、

- (a) 計算機ハードウェア, 計装システムの各々について, システムエンジニア, 計装エンジニアが, それぞれの専門 分野に応じた適正作業が行なえるようにすること(環境仕様 定義)。
- (b) プロセスと計算機システムとを結ぶハードウェアの煩

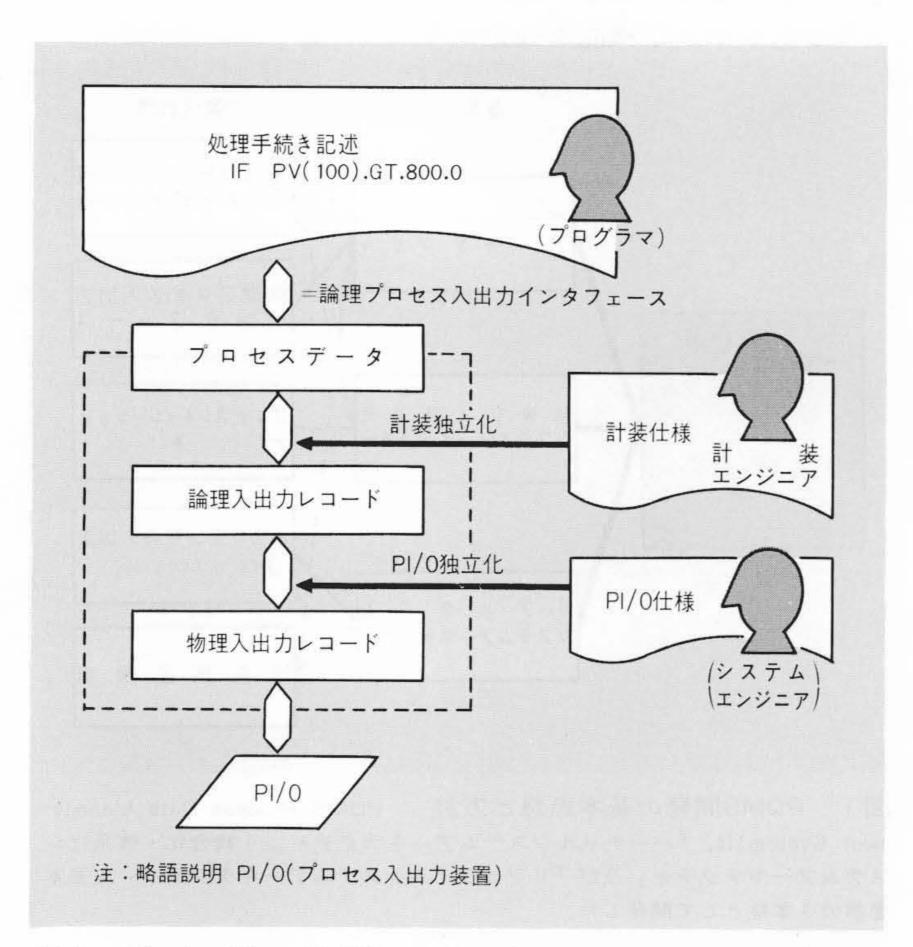


図 2 プロセスデータ処理におけるバーチャルシステムアーキテクチャ プロセスデータ処理の手続き記述と環境記述とを分離することにより、担当者の専門、レベルに応じた適正作業が行なえるようになる。

雑なインタフェースをブラックボックス化することによって、プログラマは論理レベルのプロセスデータをアクセスするだけでよいようにすること(論理プロセス入出力インタフェース)。

の実現を図ることとした(図2)。

(2) プロセスシミュレーションによる擬似オンラインテストプロセスデータ処理アプリケーションでは、その最終品質確認は、プロセスに計算機をつなぎ込んだ形で行なう必要があり、本格的なシステムテストは現地でなければできないという事情があった。このため、(i) プログラムの品質確認が遅れ、トラブル発生時の手戻りが大きい、(ii) プロセスへつなぎ込んだ形でのプログラムテストでは、再テストやプログラム増設時の確認試験が困難、という問題が発生していた。

これに対処するため、PDMSではその内部にプロセスシミュレーション機構を組み込み、これによりプロセスと計算機とを直接接続できない環境でも(また逆に、オンライン接続された環境でも)、擬似オンラインテストをテストシステムHITEST/F²)(Hitachi Integrated Test System/Function)の機能を利用して行なえるようにすることとした。

(3) リアルタイム性能確保と性能調整機構

論理的なプロセス入出力インタフェースの実現とともに制御用システムとして重要なのは、プロセスデータ処理のリアルタイム性能の確保と個別システムに合わせて性能を調整可能とすることである。これに対して、PDMSでは、

- (a) 環境仕様データをあらかじめリアルタイム処理に都合のよい形に変換・編集しておくこと(オフライン変換)などにより、オンライン時の高速なプロセスデータ処理を可能とすること。
- (b) リアルタイム処理で用いられるファイルを,主記憶(高速アクセス),補助記憶(低速アクセス)のいずれにでも配置できる機構と,このファイルの物理配置などを環境仕様として定義できる機能とにより,アプリケーションシステムの特質に応じて,システムの負荷,応答性,メモリ効率を最適に調整できるようにすること。

を具体的実現目標とした。

3 PDMSの構成

前章で述べた実現目標を満たすPDMSのシステム構成を図 3に示す。

(1) データ生成・保守サブシステム

PI/Oの構成仕様、計装仕様などのプロセスデータ処理の環境仕様データを、端末CRT(Cathode Ray Tube)やカードリーダなどから定義・保守するためのサブシステムであり、プロセスデータ処理記述言語PDDL(Process Data Description Language)とその保守ユーティリティプログラム群から成る。

(2) 入出力実行・監視サブシステム

プロセスデータ処理を実行するリアルタイムパッケージ群であり、PDDLで定義された仕様データをもとに、PV、LVの入出力実行・監視、プロセス割込解析などを行なうプロセスデータ処理パッケージと、アプリケーションプログラムとの間でデータの授受を行なうためのユーザーインタフェースマクロから成る。

(3) テストサブシステム

プロセスデータ処理アプリケーションプログラムの擬似オンラインテストを実現する部分である。制御用ソフトウェア機能一貫テストシステムHITEST/Fと有機的に結合されて

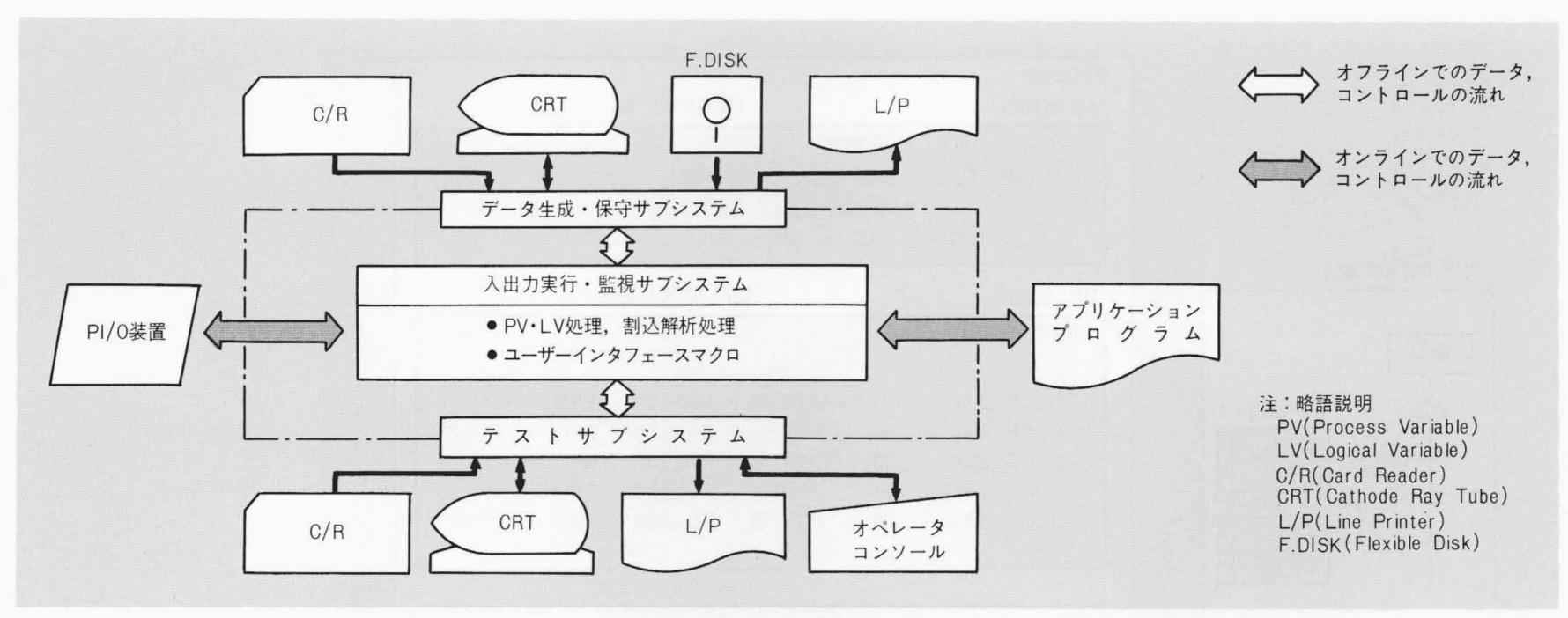


図3 PDMSの構成 PDMSは、「プロセスデータ処理の環境仕様定義データの生成・保守」、「プロセス入出力実行・監視」及び「プロセスシミュレーション テスト」を行なう各サブシステムから構成される。

おり、テストシステムの各種支援機能を利用して、プロセス いう性能 シミュレーションテストが容易にかつ効率良く行なえるよう になっている。

4 PDMSの機能

4.1 プロセスデータ処理記述言語PDDL

PDDLは、プロセスデータ処理の環境仕様を定義するため の問題向き言語である。PDDLの構成及びその定義情報を図 4に示す。ユーザーは、これらPDDLの定義機能により、仕 様データの定義・生成や変更・追加を効率良くかつ信頼性高 く行なうことができる。

(1) システム仕様定義機能

アプリケーションシステムに要求される性能仕様を定義す る機能である。要求性能としては,一般に次の二つのタイプ がある。

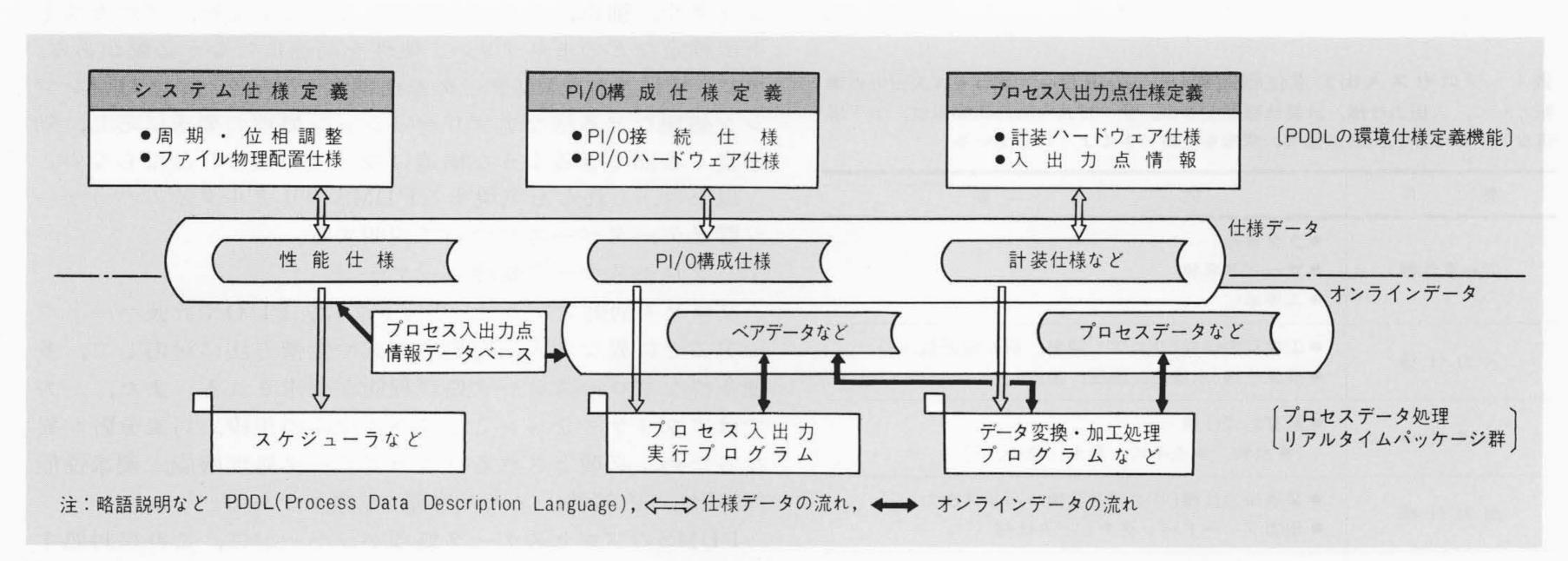
- (a) 処理性:決められた周期内に必要な量のプロセスデー タ処理を行なわねばならないという性能
- (b) 応答性:プロセスのイベントに対応して決められた時 間内に必要なプロセスデータ処理を行なわねばならないと

これらに対してシステム仕様定義では, 処理性確保のため に,(i)プロセスの時定数に応じた周期の定義機能と,負荷平 滑化のための処理位相調整機能を,また,応答性確保及び処 理性向上のために(ii) プロセスデータ処理で使用されるファイ ルを高速メモリ(主記憶装置),低速メモリ(補助記憶装置)の いずれに配置するかを選択定義することにより, 性能調整を 可能とするファイル定義機能を設け、システムの特性に応じ た性能の最適化を実現できるようにしている。

(2) PI/O構成仕樣定義機能

PI/O構成仕様定義では、プロセスとのインタフェースに 使用されるPI/Oのハードウェア仕様や、計算機との接続仕 様を規定する。

従来これらは、アプリケーションプログラムの中に散在し て埋め込まれており、接続方法の変更などによる影響を局所 化できないという問題があった。PI/O構成仕様定義では, これに対処するため、「PI/O構成仕様定義言語」により、計 算機システムに詳しいシステムエンジニアが一括して間違い のない形で仕様を規定する。これにより、 ハードウェアとの



PDDLの環境仕様定義機能とリアルタイムパッケージ群の構成 PDDL(Process Data Description Language)により定義・生成された仕様デ ータは、プロセス入出力点情報データベースに格納され、リアルタイムパッケージ群はこれをもとにプロセスデータ処理を実行する。

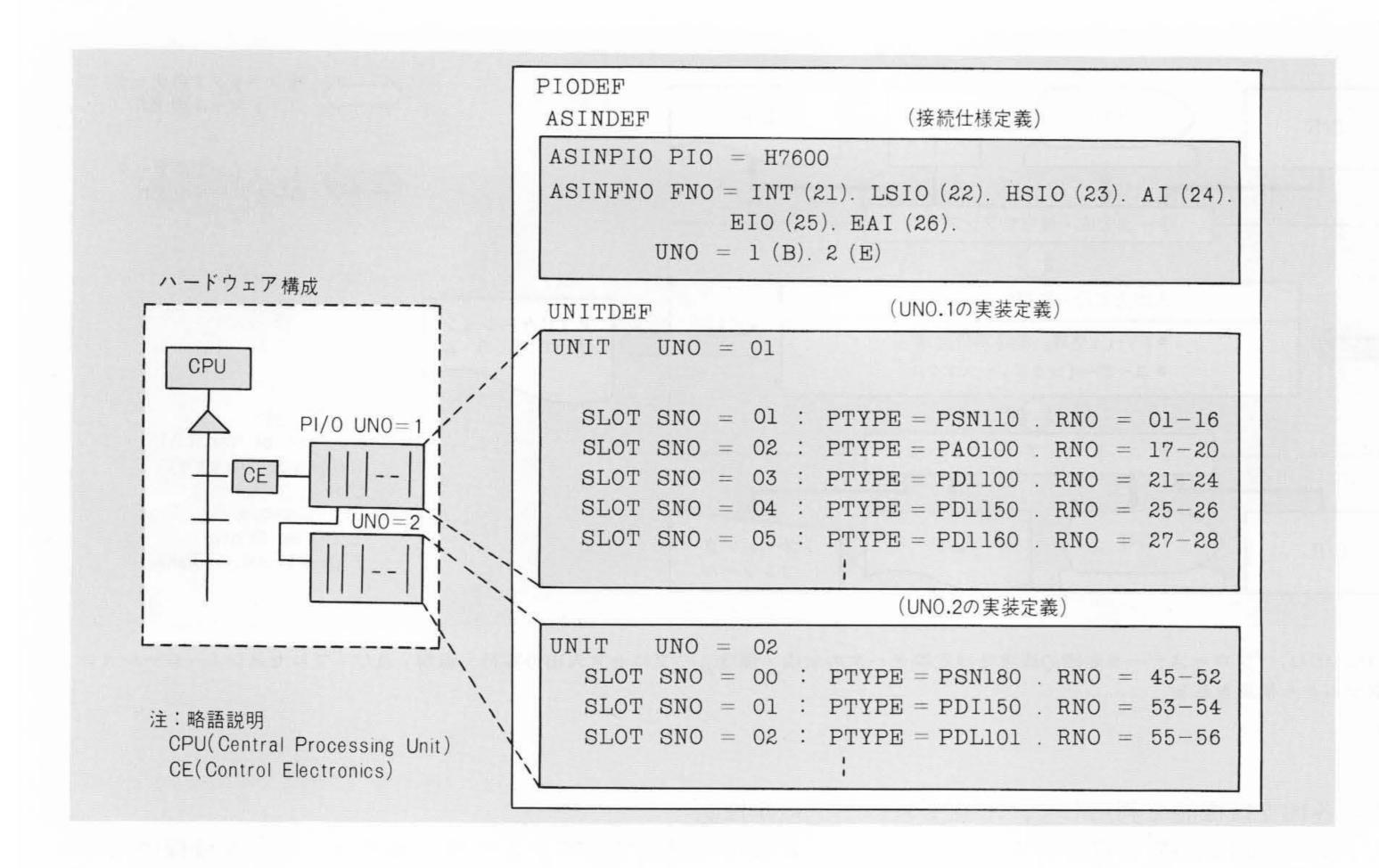


図5 PI/O構成仕様定 義例 PI/O構成仕様定 義は、ハードウェア構成・ カード実装イメージをその ままコーディングすればよ い形になっている。

インタフェースを吸収し、以降のプログラムやプロセスデータ処理の仕様定義では、PI/Oハードウェアの影響を一切排除するようにしている。

「PI/O構成仕様定義言語」は、図5に示すようにPI/Oの実装イメージをそのまま記述すればよい形となっており、これを用いることにより、システムエンジニアであれば簡単にかつ間違いなくPI/Oハードウェア仕様や接続仕様の定義が可能となる。

(3) プロセス入出力点仕様定義機能

プロセス入出力点数は数百点からプラントによっては数千点に及ぶ場合もあり、この大量入出力点のプロセスデータ処理の記述をいかに効率良く、かつ間違いなく行なえるかが重要である。プロセス入出力点仕様定義は、このプロセス入出力点ごとのプロセスデータ処理仕様や、計装ハードウェア仕様を規定するものである(表1)。

「プロセス入出力点仕様定義言語」の特長は,以下に述べるとおりである。

表 I プロセス入出力点仕様定義(PV)の項目 プロセス入出力点情報として、入出力仕様、計装仕様とともに、サービス名称、工業単位、上下限値などプロセス入出力点固有の情報を定義できるようにしている。

項目	内	容
データ仕様	● タグ番号● サービス名称● 工業単位	
入力仕様	■工学変換仕様(リニア,開平,折れ線近似,…●補正仕様(冷接点,温圧,密度)	
処理仕様	●計算処理仕様 (多点和,多点平均,最大,最小,…)	
出力仕様	●変換出力仕様(リニア逆変換,パルス出力,…) ●出力フィードバックチェック仕様	
モニタリング仕様	プロセス上下限・上上下下限,不感帯プロセス変化率検定仕様	

- (a) 記述量が少なく、簡単にコーディングできるようにFIF (Fill In the Form:空欄記述)形式を採用し、かつ記述のガイドと保守容易化のためにキーワード形式も併用している。
- (b) 複数のプロセス入出力点間で共通な仕様は、あらかじめマクロ定義として一括定義可とし、記述量の削減、仕様の一元管理を行なえるようにしている。

図6に、プロセス入出力点仕様定義FIFシートの記述例を示す。

4.2 プロセスデータ処理リアルタイムパッケージ群と データベース

PDDLで定義された環境仕様データに基づき、プロセス入出力、データ変換・加工などのプロセスデータ処理を実行する部分である(図4)。すなわち、PI/Oから計算機へ入力されたアナログデータなどを、PI/Oハードウェア仕様、プロセス入出力点仕様をもとに、PI/Oや計装ハードウェア仕様から独立したPVデータへ変換する。この変換のためには、入力チェックや、補正、スケール変換などの工学変換、プロセス上下限検定などのモニタリング処理を高速に行なう必要がある。更に、変換されたPVデータを格納するデータベースはマンマシン処理など多様なアプリケーション処理の要求に応じ、効率良く参照できるような構造になっていなければならない。

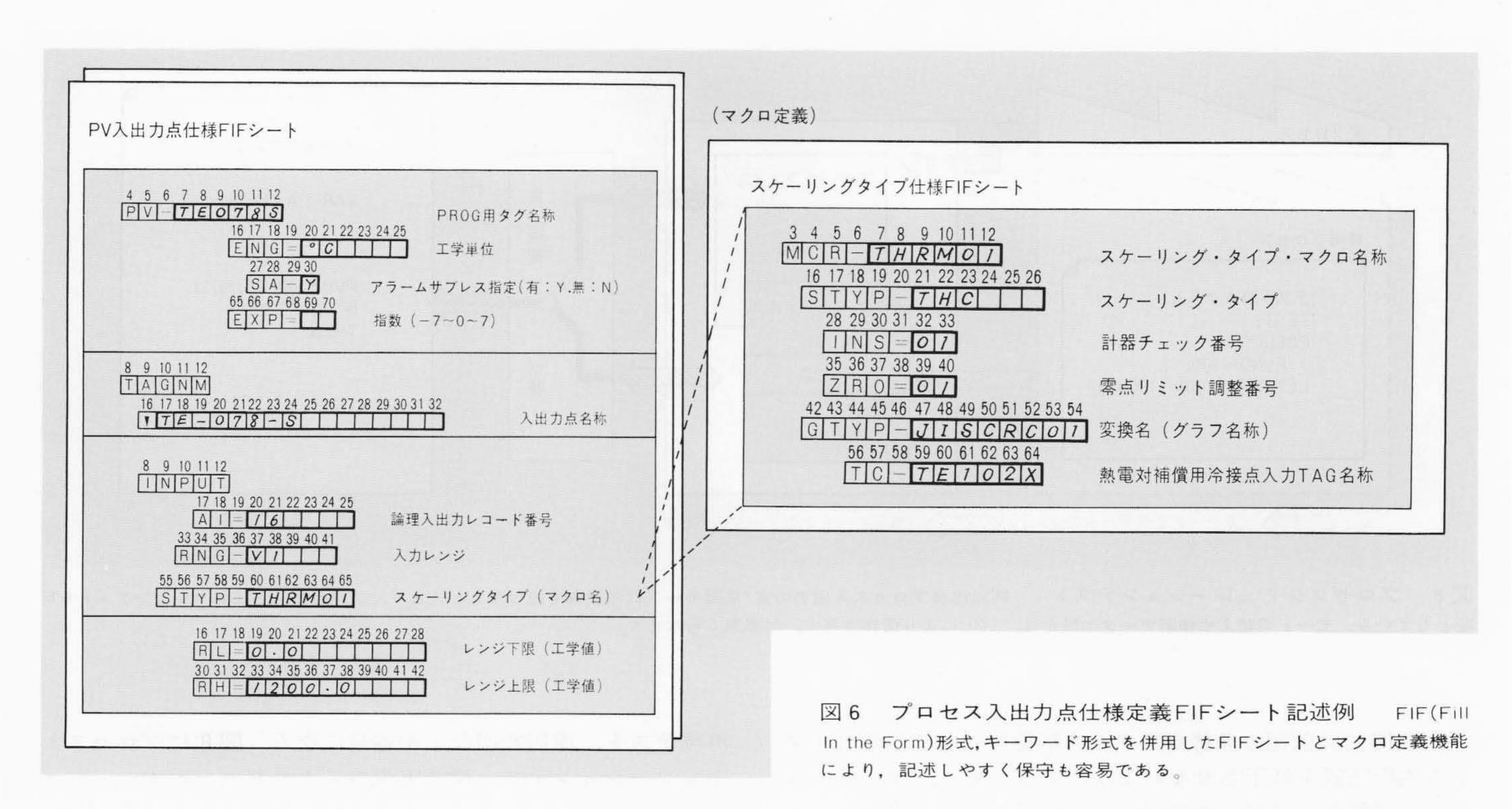
以下に、これらを実現するPDMSのリアルタイムパッケージ群とデータベースについて説明する。

(1) プロセスデータ処理パッケージ

プロセス制御アプリケーションでは、PI/Oや計装ハードウェアごとに異なる入出力処理方式や変換方法に対応して、多種多様なプロセスデータ処理機能が要求される。また、一方ではアプリケーションごとにシステムの規模、対象分野が異なるため、必要とされるプロセスデータ処理機能、要求性能(処理性、応答性、メモリ容量)も様々である。

PDMSのプロセスデータ処理パッケージは,これに対処するため,以下に述べるような特長をもっている。

(a) 入出力実行, 工学変換, モニタリング, 各種計算処理



など、豊富な機能バリエーションを標準サポートしている。

- (b) これらは機能単位にパッケージ化され、アプリケーションごとにユーザーが必要機能を選択できる。
- (c) 特殊なデータ処理に対しては、ユーザー作成プログラムをプロセスデータ処理システムの一部として有機的に組み込むことができる(ユーザーオウンコーディング)。
- (d) リアルタイム処理で使用される仕様データに対しては、あらかじめ(定義時に)、(i)リアルタイム処理に都合のよいデータ形式に変換しておく、(ii)同一周期の処理グループ単位に一括処理できるように編集しておく、などにより、リアルタイム処理の高速化を図っている。
- (2) ユーザーインタフェースマクロ ユーザーインタフェースマクロは, ユーザープログラムと PDMSとの間で, 以下に述べる三つの見方(データビュー)に

従ってデータの受け渡しを行なうためのものである。

- (a) マンマシン処理プログラムなどから, プロセス入出力点の関連情報(PV・LVデータ, ステータス情報, 原始データなど)をアクセスするデータビュー
- (b) 積算処理プログラムなどから、プロセスデータの時系 列データをアクセスするデータビュー
- (c) データ変換・加工,集計処理などで使用する仕様データ,プロセスデータを,同一周期の処理グループ単位にアクセスするデータビュー

PDMSでは、これら三つのデータビューに対応したユーザーインタフェースマクロを用意しており、ユーザープログラムからは統一的な形式で、物理ファイル構造を意識することなくデータをアクセスできるようにしている(図7)。

(3) プロセス入出力点情報データベース

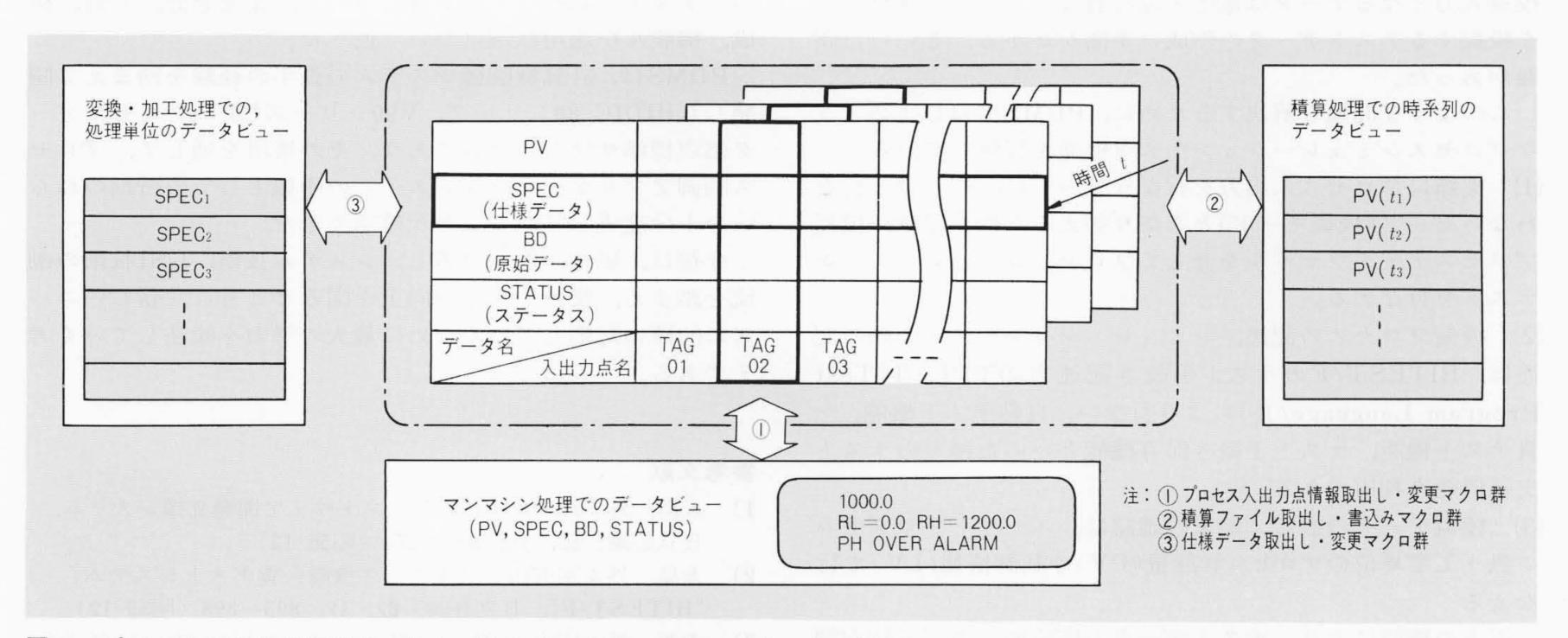


図 7 プロセス入出力点情報データベースのデータビューとユーザーインタフェースマクロ ユーザーインタフェースマクロによりリアルタイム性能を低下させることなく、アプリケーションプログラムから多様なデータビューでデータをアクセスできる。

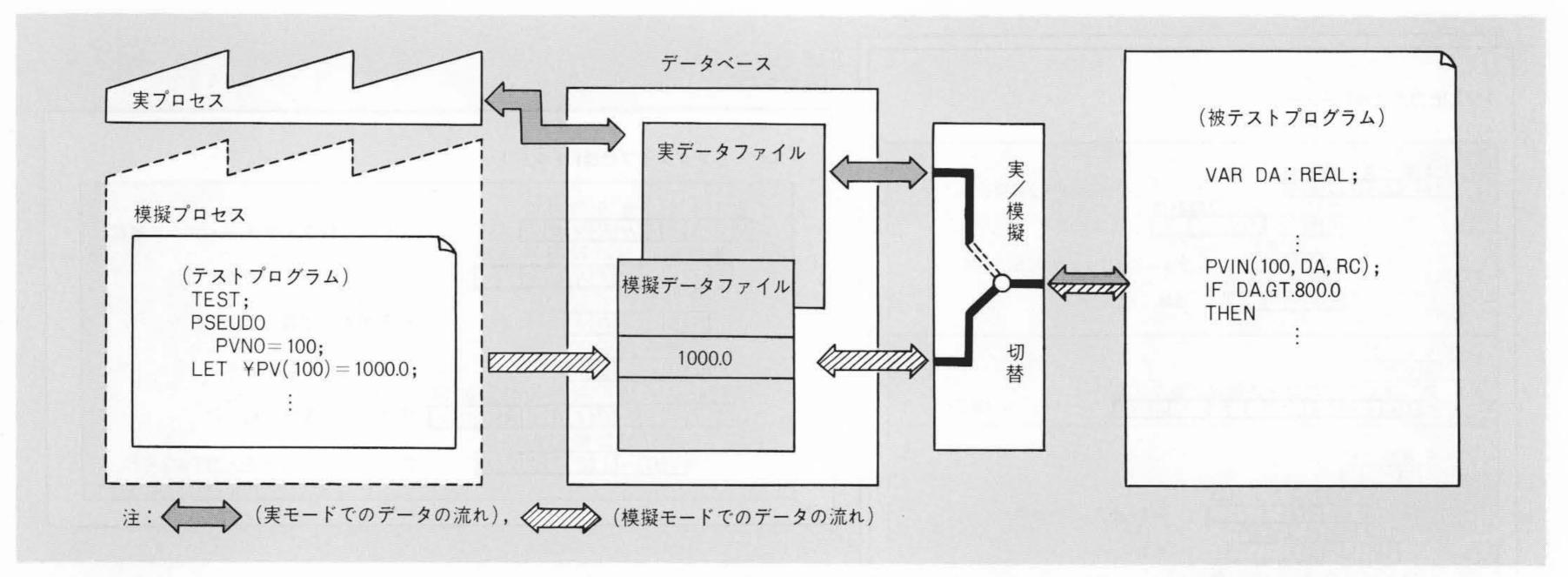


図 8 プロセスシミュレーションテスト PDMSはプロセス入出力の実/模擬モード切替機構を設けることにより、プロセスシミュレーションテストを可能としている。モード切替えや模擬データの設定は、TPLにより信頼性高く、効率良く行なえる。

PDMSでは前記の多様なアクセス形態に対して、リアルタイム処理性能を低下させないようにするために、プロセス入出力点情報データベースに対して、以下に示すようなファイル構成をとっている。

- (a) プロセスへの追従性を重視するため、オンラインでの変換・加工に適した処理グループ単位、データ種別単位に分割されたファイル構成。
- (b) 頻度高く参照されるデータは、ファイル単位、又は処理グループ単位で主記憶装置に配置できる。この定義は、 先に述べた「ファイル定義機能」により可能。

以上,(1),(2)及び(3)により,ユーザーはPDMSを用いてリアルタイム性能を確保しながら,目的に対応した論理的なデータビューで,効率良くデータのアクセスができるようになる。

4.3 プロセスシミュレーションテスト機能

従来、プロセスシミュレーションテストは、電圧・電流発生器やスイッチ、ランプをもったシミュレーション用パネルなどのテスト用のハードウェアを用いて行なっていた。この方法は、テスト結果が残らない、再現テストが困難、また、模擬入力となるデータは電圧・電流値などであり、プロセスを模擬するテストデータの作成に手間がかかる、といった問題があった。

このような問題を解決するために、PDMSでは以下のようなプロセスシミュレーションテスト機能を提供している。

- (1) 実際にプロセス入出力を行なうモード(実モード)と行なわないモード(模擬モード)とを切り替える機構を設け、模擬プロセスデータファイルを介してプロセスシミュレーションテストを行なえる。
- (2) 模擬プロセスの記述、シミュレーションテスト手順の記述は、HITEST/Fのテスト手続き記述言語TPL/II(Test Program Language/II)により行ない、自動テスト機能、一貫テスト機能、テスト手続き保存機能といった種々のテスト支援機能を利用できる。
- (3) 模擬データの設定, 結果の確認は, ユーザープログラムで扱う工学単位のプロセス状態量(PV)や制御情報(LV)で行なえる。

以上の機能により、テストデータの作成やテスト操作が簡単になり、また、トラブル発生時の不具合箇所の切り分け、

再現テスト,原因究明などが容易になる。**図8**にプロセスシ ミュレーションテスト機能実現のメカニズムを示す。

5 PDMSの適用状況とその結果

PDMSは昭和55年3月にその最終形態が完成し、その後現在までに数多くのシステムに適用を行なってきた。これらの適用を通じて得られた成果は、次に述べるとおりである。

- (1) システムエンジニア、プログラマなどシステム開発での各担当者の役割分担が明確になり、ソフトウェアの製作が容易になった。
- (2) 擬似オンラインテストにより、出荷前の擬似オンラインテストによる加速試験や、現地でのプロセスとのつなぎ込み試験時のオンラインと擬似オンラインとの即時切換えが可能となり、テスト期間の短縮及び品質の向上が実現できた。
- (3) システム構築後のシステム性能(応答性・負荷)の調整をプログラムを変更することなく、容易にできるようになった。

6 結 言

PDMSを構成するプロセスデータ処理記述言語PDDLとそのリアルタイムパッケージ群について、開発思想、方針、構成、機能及び適用状況について述べた。

PDMSは、計算機制御システムの長年の経験を踏まえて開発したHIDIC 80シリーズ、V90シリーズ³⁾用のプロセスデータ処理標準サブシステムであり、その適用を通して、プロセス制御アプリケーションシステムの中核としての所期のねらいを十分達成していることが確認できた。

今後は、更に、進歩の著しいシステム技術、LSI技術の潮流を踏まえ、使いやすさの向上を図るとともに、新しいニーズに的確に対応していくために最大の努力を傾注していく考えである。

参考文献

- 1) 大島,外:HIDIC V90/50ソフトウェア開発支援システム, 日立評論,63,12,869~874(昭56-12)
- 2) 大島,外:制御用ソフトウェア機能一貫テストシステム "HITEST/F",日立評論,62,12,893~898(昭55-12)
- 3) 中西,外:HIDIC V90/50基本制御ソフトウェア,日立評論, 63,12,863~868(昭56-12)