

(19) 日本国特許庁(JP)

(12) 公開特許公報(A)

(11) 特許出願公開番号

特開2014-99058

(P2014-99058A)

(43) 公開日 平成26年5月29日(2014.5.29)

(51) Int.Cl.
G05B 19/05 (2006.01)

F I
G05B 19/05

テーマコード(参考)
5H220

審査請求 未請求 請求項の数 6 O L (全 18 頁)

(21) 出願番号 特願2012-250615 (P2012-250615)
(22) 出願日 平成24年11月14日(2012.11.14)

(71) 出願人 000002945
オムロン株式会社
京都市下京区塩小路通堀川東入南不動堂町
801番地
(74) 代理人 110001195
特許業務法人深見特許事務所
(72) 発明者 岡村 弘太郎
京都府京都市下京区塩小路通堀川東入南不
動堂町801番地 オムロン株式会社内
(72) 発明者 西山 佳秀
京都府京都市下京区塩小路通堀川東入南不
動堂町801番地 オムロン株式会社内
(72) 発明者 太田 政則
京都府京都市下京区塩小路通堀川東入南不
動堂町801番地 オムロン株式会社内
最終頁に続く

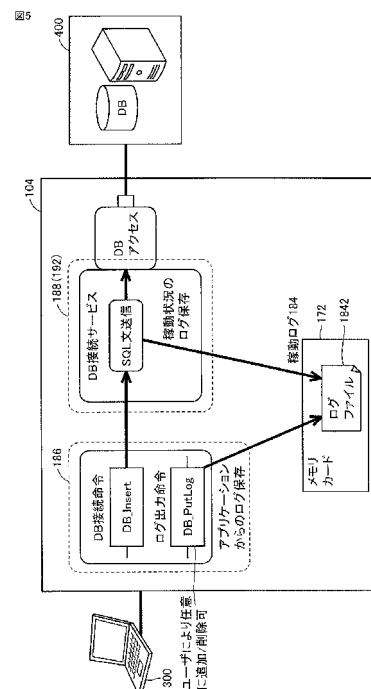
(54) 【発明の名称】 制御システムおよび制御装置

(57) 【要約】 (修正有)

【課題】プログラムのデバッグや健全性のチェックを効率的に行うことのできる制御システムおよび制御装置を提供する。

【解決手段】制御システムは、ユーザプログラムを周期的またはイベント的に実行する制御装置104と、制御装置に接続可能な情報処理装置300とを含む。制御装置104は、予め定められた事象の発生にตอบสนองして、当該発生した事象の情報を時刻情報と関連付けて記憶部に書き込む第1の書込手段と、ユーザプログラムに含まれる所定の命令の実行にตอบสนองして、当該命令によって指定された情報を時刻情報と関連付けて記憶部に書き込む第2の書込手段とを含む。情報処理装置300は、制御装置104で実行されるユーザプログラムの任意の位置に所定の命令を追加する処理、および、ユーザプログラムに含まれる所定の命令を任意に削除する処理の少なくとも一方を実行するためのユーザインターフェイスを提供する。

【選択図】 図5



【特許請求の範囲】

【請求項 1】

制御システムであって、
ユーザプログラムを周期的またはイベント的に実行する制御装置と、
前記制御装置に接続可能な情報処理装置とを備え、
前記制御装置は、

予め定められた事象の発生にตอบสนองして、当該発生した事象の情報を時刻情報と関連付けて記憶部に書き込む第 1 の書込手段と、

前記ユーザプログラムに含まれる所定の命令の実行にตอบสนองして、当該命令によって指定された情報を時刻情報と関連付けて前記記憶部に書き込む第 2 の書込手段とを備え、

前記情報処理装置は、前記制御装置で実行される前記ユーザプログラムの任意の位置に前記所定の命令を追加する処理、および、前記ユーザプログラムに含まれる前記所定の命令を任意に削除する処理の少なくとも一方を実行するためのユーザインターフェイスを提供する、制御システム。

【請求項 2】

前記第 1 の書込手段および前記第 2 の書込手段は、前記記憶部の共通の記憶領域へ情報を発生順に書き込む、請求項 1 に記載の制御システム。

【請求項 3】

前記制御装置は、予め格納されたシステムプログラムを実行することで、前記予め定められた事象を発生させる、請求項 1 または 2 に記載の制御システム。

【請求項 4】

前記情報処理装置は、前記制御装置が前記ユーザプログラムを実行中に、前記所定の命令の追加または削除が可能である、請求項 1 ~ 3 のいずれか 1 項に記載の制御システム。

【請求項 5】

前記情報処理装置は、前記制御装置の前記記憶部に格納される情報を時系列に表示するユーザインターフェイスを提供する、請求項 1 ~ 4 のいずれか 1 項に記載の制御システム。

【請求項 6】

ユーザプログラムを周期的またはイベント的に実行する制御装置であって、

予め定められた事象の発生にตอบสนองして、当該発生した事象の情報を時刻情報と関連付けて記憶部に書き込む第 1 の書込手段と、

前記ユーザプログラムに含まれる所定の命令の実行にตอบสนองして、当該命令によって指定された情報を時刻情報と関連付けて前記記憶部に書き込む第 2 の書込手段と、

外部からの指令に従って、前記ユーザプログラムの任意の位置に前記所定の命令を追加する処理、または、前記ユーザプログラムに含まれる前記所定の命令を任意に削除する処理の少なくとも一方を実行するための変更手段とを備える、制御装置。

【発明の詳細な説明】

【技術分野】

【0001】

本発明は、ユーザプログラムを周期的またはイベント的に実行する制御装置を含む制御システムおよびその制御装置に関する。

【背景技術】

【0002】

多くの生産現場で使用される機械や設備は、典型的には、プログラマブルコントローラ (Programmable Logic Controller ; 以下「PLC」とも称す) などの制御装置を主たる構成とした制御システムによって制御される。このような制御システムでは、システム構成時や運転時に発生した不具合を事後的に検証できるように、各種のログ出力機能が搭載されている。

【0003】

このような機能を搭載した制御装置に関する先行技術文献として、特開 2012 - 20

10

20

30

40

50

8932号公報(特許文献1)は、機器の動作を制御するPLCと、PLCと通信回線を介して接続されるプログラマブル表示器とを備えるPLCシステムを開示する。このPLCシステムにおいて、PLCは、機器の所定の動作状態を検出した際に、情報記憶手段により記憶した情報の中から、所定の動作状態に該当する該当状態情報を所定の格納部へ格納する該当情報格納手段を含む。

【先行技術文献】

【特許文献】

【0004】

【特許文献1】特開2012-208932号公報

【発明の概要】

10

【発明が解決しようとする課題】

【0005】

近年の情報処理技術の進歩に伴って、PLCなどの制御装置の処理能力も向上し、多機能化および処理高速化が進みつつある。この結果、1台の制御装置に多種多様なプログラムがインストールされて実行される状況が生じている。このような状況下では、各アプリケーションが正しくプログラムされているかといったテストや、各アプリケーションが正しく動作しているかといったテストを効率的に行う必要がある。

【0006】

特許文献1に記載のPLCシステムは、操作性を向上させることができるが、このようなプログラムのデバッグや健全性のチェックを効率的に行うという目的からは十分ではない場合がある。

20

【0007】

本発明は、プログラムのデバッグや健全性のチェックを効率的に行うことのできる制御システムおよび制御装置を提供することを目的とする。

【課題を解決するための手段】

【0008】

本発明のある局面に係る制御システムは、ユーザプログラムを周期的またはイベント的に実行する制御装置と、制御装置に接続可能な情報処理装置とを含む。制御装置は、予め定められた事象の発生に应答して、当該発生した事象の情報を時刻情報と関連付けて記憶部に書き込む第1の書込手段と、ユーザプログラムに含まれる所定の命令の実行に应答して、当該命令によって指定された情報を時刻情報と関連付けて記憶部に書き込む第2の書込手段とを含む。情報処理装置は、制御装置で実行されるユーザプログラムの任意の位置に所定の命令を追加する処理、および、ユーザプログラムに含まれる所定の命令を任意に削除する処理の少なくとも一方を実行するためのユーザインターフェイスを提供する。

30

【0009】

好ましくは、第1の書込手段および第2の書込手段は、記憶部の共通の記憶領域へ情報を発生順に書き込む。

【0010】

好ましくは、制御装置は、予め格納されたシステムプログラムを実行することで、予め定められた事象を発生させる。

40

【0011】

好ましくは、情報処理装置は、制御装置がユーザプログラムを実行中に、所定の命令の追加または削除が可能である。

【0012】

好ましくは、情報処理装置は、制御装置の記憶部に格納される情報を時系列に表示するユーザインターフェイスを提供する。

【0013】

本発明の別の局面に係るユーザプログラムを周期的またはイベント的に実行する制御装置は、予め定められた事象の発生に应答して、当該発生した事象の情報を時刻情報と関連付けて記憶部に書き込む第1の書込手段と、ユーザプログラムに含まれる所定の命令の実

50

行に応答して、当該命令によって指定された情報を時刻情報と関連付けて記憶部に書き込む第2の書込手段と、外部からの指令に従って、ユーザプログラムの任意の位置に所定の命令を追加する処理、または、ユーザプログラムに含まれる所定の命令を任意に削除する処理の少なくとも一方を実行するための変更手段とを含む。

【発明の効果】

【0014】

本発明によれば、プログラムのデバッグや健全性のチェックを効率的に行うことができる。

【図面の簡単な説明】

【0015】

【図1】本実施の形態に係る制御システムのシステム構成を示す模式図である。

【図2】本実施の形態に係るPLCの主要部を示すハードウェア構成を示す模式図である。

【図3】本実施の形態に係るPLCのソフトウェア構成を示す模式図である。

【図4】本実施の形態に係るPLCに接続して用いられるサポート装置のハードウェア構成を示す模式図である。

【図5】本実施の形態に係る制御システムにより提供されるログ出力機能の概要を説明するための模式図である。

【図6】本実施の形態に係るサポート装置において提供されるユーザプログラムの生成に係るユーザインターフェイスの一例を示す図である。

【図7】本実施の形態に係るサポート装置において提供されるユーザプログラムの生成に係るユーザインターフェイスの一例を示す図である。

【図8】本実施の形態に係るサポート装置において提供される稼動ログの取得および表示に係るユーザインターフェイスの一例を示す図である。

【図9】本実施の形態に係るPLCにおける処理手順を示すフローチャートである。

【図10】本実施の形態の変形例に係る稼動ログの格納方法を説明するための図である。

【発明を実施するための形態】

【0016】

本発明の実施の形態について、図面を参照しながら詳細に説明する。なお、図中の同一または相当部分については、同一符号を付してその説明は繰り返さない。

【0017】

< A . システム構成 >

まず、本実施の形態に係る制御システムのシステム構成について説明する。本実施の形態においては、機械や設備などの制御対象を制御するプログラマブルコントローラ(PLC)を制御装置の典型例として説明を行う。但し、本発明に係る制御装置は、PLCに限られることなく、各種の制御装置へ適用可能である。

【0018】

図1は、本実施の形態に係る制御システム1のシステム構成を示す模式図である。図1を参照して、制御システム1は、PLC100と、PLC100に接続されるサポート装置300と、PLC100からのアクセスを受けるデータベース装置400とを含む。PLC100は、後述するようなユーザプログラムを周期的またはイベント的に実行する。このユーザプログラムにアクセス命令を含めることができ、PLC100は、このアクセスに従って、データベース装置400にアクセスすることができる。すなわち、PLC100およびデータベース装置400は、イーサネット(登録商標)などのネットワーク112を介して、データを互いに遣り取りできるように構成される。

【0019】

サポート装置300は、PLC100に接続可能な情報処理装置の典型例である。サポート装置300は、接続ケーブル114を介してPLC100に接続され、PLC100との間で各種パラメータの設定、プログラミング、モニタ、デバッグなどの機能を提供する。PLC100とサポート装置300との間は、典型的には、USB(Universal Seri

10

20

30

40

50

a) Bus) 規格に従って通信可能に構成される。

【0020】

PLC100は、制御演算を実行するCPUユニット104と、1つ以上のI/O (Input/Output) ユニット106とを含む。これらのユニットは、PLCシステムバス108を介して、データを互いに遣り取りできるように構成される。また、これらのユニットには、電源ユニット102によって適切な電圧の電源が供給される。

【0021】

制御システム1において、PLC100は、(PLCシステムバス108を介して接続される) I/Oユニット106を介して、および/または、フィールドバス110を介して、各種のフィールド機器との間でデータを遣り取りする。これらのフィールド機器は、制御対象に対して何らかの処理を行うためのアクチュエータや、制御対象から各種情報を取得するためのセンサなどを含む。図1には、このようなフィールド機器の一例として、検出スイッチ10、リレー20およびモータ32を駆動するサーボモータドライバ30を含む。また、PLC100は、フィールドバス110を介してリモートI/Oターミナル200も接続されている。リモートI/Oターミナル200は、基本的には、I/Oユニット106と同様に、一般的な入出力処理に関する処理を行う。より具体的には、リモートI/Oターミナル200は、フィールドバス110でのデータ伝送に係る処理を行うための通信ケーブル202と、1つ以上のI/Oユニット204とを含む。これらのユニットは、リモートI/Oターミナルバス208を介して、データを互いに遣り取りできるように構成される。

【0022】

< B . PLC100の構成 >

次に、本実施の形態に係るPLC100の構成について説明する。図2は、本実施の形態に係るPLC100の主要部を示すハードウェア構成を示す模式図である。図3は、本実施の形態に係るPLC100のソフトウェア構成を示す模式図である。

【0023】

図2を参照して、PLC100のCPUユニット104のハードウェア構成について説明する。CPUユニット104は、プロセッサ120と、チップセット122と、システムクロック124と、主メモリ126と、不揮発性メモリ128と、USBコネクタ130と、PLCシステムバスコントローラ140と、フィールドバスコントローラ150と、上位通信コントローラ160と、メモリカードインターフェイス170とを含む。チップセット122と他のコンポーネントとの間は、各種のバスを介してそれぞれ結合されている。

【0024】

プロセッサ120およびチップセット122は、典型的には、汎用的なコンピュータアーキテクチャに準じて構成される。すなわち、プロセッサ120は、チップセット122から内部クロックに従って順次供給される命令コードを解釈して実行する。チップセット122は、接続されている各種コンポーネントとの間で内部的なデータを遣り取りするとともに、プロセッサ120に必要な命令コードを生成する。システムクロック124は、予め定められた周期のシステムクロックを発生してプロセッサ120に提供する。チップセット122は、プロセッサ120での演算処理の実行の結果得られたデータなどをキャッシュする機能を有する。

【0025】

CPUユニット104は、記憶手段として、主メモリ126および不揮発性メモリ128を有する。主メモリ126は、揮発性の記憶領域であり、プロセッサ120で実行されるべき各種プログラムを保持するとともに、各種プログラムの実行時の作業用メモリとしても使用される。不揮発性メモリ128は、OS (Operating System)、システムプログラム、ユーザプログラム、データ定義情報、ログ情報などを不揮発的に保持する。

【0026】

USBコネクタ130は、サポート装置300とCPUユニット104とを接続するためのインターフェイスである。典型的には、サポート装置300から転送される実行可能

10

20

30

40

50

なプログラムなどは、USBコネクタ130を介してCPUユニット104に取込まれる。

【0027】

CPUユニット104は、通信手段として、PLCシステムバスコントローラ140、フィールドバスコントローラ150、および上位通信コントローラ160を有する。これらの通信回路は、データの送信および受信を行う。

【0028】

PLCシステムバスコントローラ140は、PLCシステムバス108を介したデータの遣り取りを制御する。より具体的には、PLCシステムバスコントローラ140は、バッファメモリ142と、PLCシステムバス制御回路144と、DMA (Dynamic Memory Access) 制御回路146とを含む。PLCシステムバスコントローラ140は、PLCシステムバスコネクタ148を介してPLCシステムバス108と接続される。

10

【0029】

フィールドバスコントローラ150は、バッファメモリ152と、フィールドバス制御回路154と、DMA制御回路156とを含む。フィールドバスコントローラ150は、フィールドバスコネクタ158を介してフィールドバス110と接続される。上位通信コントローラ160は、バッファメモリ162と、上位通信制御回路164と、DMA制御回路166とを含む。上位通信コントローラ160は、上位通信コネクタ168を介してネットワーク112と接続される。

20

【0030】

メモリカードインターフェイス170は、CPUユニット104に対して着脱可能なメモリカード172とプロセッサ120とを接続する。

【0031】

次に、図3を参照して、本実施の形態に係るPLC100が提供する各種機能を実現するためのソフトウェア構成について説明する。これらのソフトウェアに含まれる命令コードは、適切なタイミングで読み出され、CPUユニット104のプロセッサ120によって実行される。

【0032】

図3を参照して、CPUユニット104で実行されるソフトウェアとしては、OS180と、システムプログラム188と、ユーザプログラム186との3階層になっている。

30

【0033】

OS180は、プロセッサ120がシステムプログラム188およびユーザプログラム186を実行するための基本的な実行環境を提供する。

【0034】

システムプログラム188は、PLC100としての基本的な機能を提供するためのソフトウェア群である。具体的には、システムプログラム188は、シーケンス命令プログラム190と、DB (データベース) アクセス処理プログラム192と、入出力処理プログラム194と、Toolインターフェイス処理プログラム196と、スケジューラ198とを含む。

40

【0035】

これに対して、ユーザプログラム186は、制御対象に対する制御目的に応じて任意に作成されたプログラムである。すなわち、ユーザプログラム186は、制御システム1を用いて制御する対象のライン (プロセス) などに応じて、任意に設計される。

【0036】

ユーザプログラム186は、シーケンス命令プログラム190と協働して、ユーザにおける制御目的を実現する。すなわち、ユーザプログラム186は、シーケンス命令プログラム190によって提供される命令、関数、機能モジュールなどを利用することで、プログラムされた動作を実現する。そのため、ユーザプログラム186およびシーケンス命令プログラム190を「制御プログラム」と総称する場合もある。

【0037】

50

データ定義情報 182 は、ユーザプログラム 186 などが実行される際に、参照されるデータ（入力データ、出力データ、内部データ）をユニークな変数として扱うための定義を含む。稼働ログ 184 には、システムプログラム 188 およびユーザプログラム 186 の実行に伴って、予め定められた事象が発生した際に、当該発生した事象の情報が時刻情報と関連付けて格納される。すなわち、稼働ログ 184 には、システムプログラム 188 および / またはユーザプログラム 186 の実行に伴う各種情報がログ（履歴情報）として格納される。

【0038】

以下、各プログラムについてより詳細に説明する。

シーケンス命令プログラム 190 は、ユーザプログラム 186 の実行に伴って、ユーザプログラム 186 内で指定されているシーケンス命令の実体を呼び出して、その命令の内容を実現するための命令コード群を含む。

10

【0039】

DB アクセス処理プログラム 192 は、ユーザプログラム 186 の実行に伴って、データベース装置 400 へアクセスするために必要な処理を実現するための命令コード群を含む。DB アクセス処理プログラム 192 は、ユーザプログラム 186 において指定可能な命令に対応する実行コードを含む。

【0040】

入出力処理プログラム 194 は、I/O ユニット 106 や各種のフィールド機器との間で、入力データの取得および出力データの送信を管理するためのプログラムである。

20

【0041】

Tool インターフェイス処理プログラム 196 は、サポート装置 300 との間でデータを遣り取りするためのインターフェイスを提供する。

【0042】

スケジューラ 198 は、予め定められた優先度やシステムタイマの値などに従って、制御プログラムを実行するためのスレッドやプロシージャを生成する。

【0043】

ユーザプログラム 186 は、上述したように、ユーザにおける制御目的（たとえば、対象のラインやプロセス）に応じて作成される。ユーザプログラム 186 は、典型的には、CPU ユニット 104 のプロセッサ 120 で実行可能なオブジェクトプログラム形式になっている。ユーザプログラム 186 は、サポート装置 300 などにおいて、ラダー形式やファンクションブロック形式で記述されたソースプログラムがコンパイルされることで生成される。そして、生成されたオブジェクトプログラム形式のユーザプログラムは、サポート装置 300 から CPU ユニット 104 へ転送され、不揮発性メモリ 128 などに格納される。

30

【0044】

< C . サポート装置 300 の構成 >

次に、本実施の形態に係るサポート装置 300 について説明する。サポート装置 300 は、PLC 100 の CPU ユニット 104 の使用を支援するためのものであり、PLC 100 との間で各種パラメータの設定、プログラミング、モニタ、デバッグなどの機能を提供する。

40

【0045】

図 4 は、本実施の形態に係る PLC 100 に接続して用いられるサポート装置 300 のハードウェア構成を示す模式図である。サポート装置 300 は、典型的には、汎用のコンピュータで構成される。

【0046】

図 4 を参照して、サポート装置 300 は、OS を含む各種プログラムを実行する CPU 302 と、BIOS や各種データを格納する ROM (Read Only Memory) 304 と、CPU 302 でのプログラムの実行に必要なデータを格納するための作業領域を提供するメモリ RAM 306 と、CPU 302 で実行されるプログラムなどを不揮発的に格納するハー

50

ドディスク（HDD）308とを含む。より具体的には、ハードディスク308には、サポート装置300が提供する機能を実現するためのサポートプログラム330が格納されている。

【0047】

サポート装置300は、さらに、ユーザからの操作を受け付けるキーボード310およびマウス312と、情報をユーザに提示するためのモニタ314とを含む。さらに、サポート装置300は、PLC100（CPUユニット104）などと通信するための通信インターフェイス（IF）318を含む。

【0048】

サポート装置300で実行されるサポートプログラム330などは、光学記録媒体332に格納されて流通する。光学記録媒体332に格納されたプログラムは、光学ディスク読取装置316によって読み取られ、ハードディスク308などへ格納される。あるいは、上位のホストコンピュータなどからネットワークを通じてプログラムをダウンロードするように構成してもよい。

【0049】

< D . データベース装置400の構成 >

次に、本実施の形態に係るデータベース装置400について説明する。データベース装置400としては、データベースを提供する公知の構成を採用することができる。このようなデータベースとしては、リレーショナルデータ型やオブジェクトデータ型といった任意の構成を採用できる。データベース装置400は、汎用的なコンピュータアーキテクチャに従って構成されるため、ここではその詳細な説明は繰り返さない。

【0050】

データベース装置400は、PLC100からの接続要求やアクセス要求（リレーショナルデータ型では、SQL文）を受けて、必要な処理を実行し、その処理結果などをPLC100へ応答する。

【0051】

< E . ログ出力機能の概要 >

次に、本実施の形態に係る制御システム1により提供されるログ出力機能の概要について説明する。本実施の形態に係るログ出力機能は、通常プログラムの実行に伴って出力されるログ（以下「実行ログ」とも称す。）や、プログラムの実行失敗やハードウェアの不具合の発生に伴って出力されるログ（以下「イベントログ」とも称す。）に加えて、ユーザプログラム186において任意に指定された位置および内容のログ（以下「デバッグログ」とも称す。）を出力することが可能になっている。稼働ログ184は、これらのすべてのログを含む。

【0052】

実行ログは、予め定められた種類の処理が実行された際に、その処理内容を時刻情報と関連付けて記録した情報である。この実行ログによって、ユーザプログラム186において指定された処理が実行されたことを確認することができる。実行ログは、基本的には、PLC100が稼働中（プログラムを実行している状態）において、常にロギングを行う。イベントログは、指定された処理の実行が失敗したような場合に、その処理内容を時刻情報と関連付けて記録した情報である。典型的には、指定された処理がエラーなどで中断したような場合に、その内容がロギングされる。

【0053】

デバッグログは、プログラムのデバッグや健全性のチェックを効率的に行うための情報であり、基本的には、システムの調整中に利用される。すなわち、ユーザが利用したいときだけ、ユーザプログラム186に追加することで、プログラム実行中の所望する情報を所望するタイミング（実行位置）で出力させることができる。

【0054】

本実施の形態に係るログ出力機能を用いることで、ユーザは、ユーザプログラム186の任意の位置にデバッグログを出力するための命令を追加および削除することができ、こ

10

20

30

40

50

の命令によって出力されるデバッグログを用いて、プログラムのデバッグや健全性のチェックを効率的に行うことができる。すなわち、本実施の形態に係る制御システム 1 は、ユーザが目的とする実行位置において、目的とするログを容易に取得するための機能を提供する。

【 0 0 5 5 】

図 5 は、本実施の形態に係る制御システム 1 により提供されるログ出力機能の概要を説明するための模式図である。図 5 には、ユーザプログラム 1 8 6 においてデータベース装置 4 0 0 への接続命令が定義されている例を示す。但し、ユーザプログラム 1 8 6 に定義される命令は任意に定義でき、この定義される命令の種類は、本発明の範囲に何ら影響を与えない。

10

【 0 0 5 6 】

より具体的には、ユーザがサポート装置 3 0 0 を操作して、ユーザプログラム 1 8 6 内に DB 接続命令 (DB_Connect) を指定したとする。このユーザプログラム 1 8 6 が実行され、DB 接続命令の実行タイミングになると、DB アクセス処理プログラム 1 9 2 (図 3) の対応する命令コードが呼び出されて、データベース装置 4 0 0 への接続要求を含む SQL 文が生成および送信される。すなわち、システムプログラム 1 8 8 に含まれる DB アクセス処理プログラム 1 9 2 によって実現される DB 接続サービスが提供されており、ユーザプログラム 1 8 6 内の DB 接続命令の発行によって、DB 接続サービスがトリガーされる。そして、DB 接続サービスは、SQL 文を生成し、生成した SQL 文を DB アクセス機能へ渡す。DB アクセス機能は、上位通信コントローラ 1 6 0 (図 2) および DB アクセス処理プログラム 1 9 2 (図 3) によって実現される機能であり、DB 接続サービスから受取った SQL 文を、ネットワーク 1 1 2 を介してデータベース装置 4 0 0 へ送信する。

20

【 0 0 5 7 】

データベース装置 4 0 0 は、PLC 1 0 0 からの SQL 文を受信すると、それに従って処理を実行し、必要に応じて、その結果を PLC 1 0 0 へ応答する。

【 0 0 5 8 】

DB 接続サービスは、この DB 接続命令の実行 (SQL 文の作成および送信) に応答し、当該命令によって指定された情報を時刻情報と関連付けて (つまり、実行ログとして) 稼動ログ 1 8 4 (図 5 に示す例では、メモリカード 1 7 2 に格納されるログファイル 1 8 4 2) に書き込む。つまり、PLC 1 0 0 の稼動状況のログが保存される。このとき、ユーザは、稼動ログ 1 8 4 への実行ログの書き込みを明示的に指示する必要はなく、DB 接続命令が実行されると、DB 接続サービスが自動的に稼動状況を稼動ログ 1 8 4 へ書き込むようなシステム環境が構築されている。

30

【 0 0 5 9 】

すなわち、PLC 1 0 0 の CPU ユニット 1 0 4 は、DB アクセス処理プログラム 1 9 2 (システムプログラム 1 8 8) を実行することで、予め定められた事象の発生に応答して、当該発生した事象の情報 (実行ログおよび / または イベントログ) を時刻情報と関連付けて記憶部であるメモリカード 1 7 2 に書き込む書込手段として機能する。このとき、不揮発性メモリ 1 2 8 などに予め格納されたシステムプログラム 1 8 8 を PLC 1 0 0 の CPU ユニット 1 0 4 が実行することで、予め定められた事象 (例えば、DB 接続命令の実行) を発生させる。

40

【 0 0 6 0 】

なお、稼動ログ 1 8 4 の書き込み先は、メモリカード 1 7 2 に限らず、主メモリ 1 2 6 や不揮発性メモリ 1 2 8 (図 2) であってもよいし、あるいは、PLC 1 0 0 の外部にある記憶装置であってもよい。但し、PLC 1 0 0 の電源が遮断された場合であっても、これらの稼動ログ 1 8 4 を保持できる記憶装置が好ましい。

【 0 0 6 1 】

このようなシステム環境において、ユーザは、ユーザプログラム 1 8 6 の任意の位置にログ出力命令を追加することができるとともに、ユーザプログラム 1 8 6 に含まれるログ

50

出力命令を任意に削除することもできる。例えば、ユーザが、図5に示すようにユーザプログラム186にログ出力命令(D B _ P u t L o g)を指定したとする。このログ出力命令は、D B 接続命令の後に定義されたとする。すると、D B 接続命令の実行に伴って出力される実行ログに続いて、ログ出力命令の実行に伴って、デバッグログが生成され、稼働ログ184へ書き込まれる。このデバッグログは、ログ出力命令によって指定された情報と、それに関連付けられた時刻情報とを含む。

【0062】

なわち、P L C 1 0 0 の C P U ユニット104は、ユーザプログラム186に含まれる所定の命令(ログ出力命令)の実行に応答して、当該命令によって指定された情報(デバッグログ)を時刻情報と関連付けて記憶部であるメモリカード172に書き込む書込手段として機能する。なお、図5に示す例では、P L C 1 0 0 の C P U ユニット104がD B アクセス処理プログラム192(システムプログラム188)を実行することで出力される情報(実行ログおよび/またはイベントログ)と、P L C 1 0 0 の C P U ユニット104がユーザプログラム186に含まれるログ出力命令を実行することで出力される情報(デバッグログ)とは、記憶部であるメモリカード172の共通の記憶領域(ログファイル1842)へ発生順に書き込まれる。

10

【0063】

< F . ユーザインターフェイス(ログ出力命令の追加/削除) >

次に、本実施の形態に係るサポート装置300により提供されるログ出力機能に係るユーザインターフェイスについて説明する。図5に示すようなログ出力機能に向けて、サポート装置300は、P L C 1 0 0 で実行されるユーザプログラム186の任意の位置に所定の命令(ログ出力命令)を追加する処理、および、ユーザプログラム186に含まれる所定の命令(ログ出力命令)を任意に削除する処理の少なくとも一方を実行するためのユーザインターフェイスを提供する。

20

【0064】

図6および図7は、本実施の形態に係るサポート装置300において提供されるユーザプログラムの生成に係るユーザインターフェイスの一例を示す図である。より詳しくは、図6は、D B 接続命令を含むユーザプログラムを生成した状態を示し、図7は、図6に示すユーザプログラムに対してログ出力命令を追加した状態を示す。

30

【0065】

図6に示すユーザインターフェイス画面350において作成されるユーザプログラム186では、実行開始条件として、動作中フラグ352および実行開始フラグ354が定義されており、これらの両フラグがオンになると、D B 接続処理が実行される。このD B 接続処理として、D B 接続命令ブロック356と、D B 接続用マップ選択ブロック358と、I N S E R T 用 / S E L E C T 用 D B マップ生成ブロック360とが定義されている。

【0066】

そして、D B 接続用マップ選択ブロック358の実行が完了したことをもってオンされるマップ選択完了フラグ362と、動作中フラグ352とによって、D B 接続完了を示すフラグ364がオンされるようにプログラムされている。

40

【0067】

図7に示すユーザプログラム186では、図6と比較して、ログ出力命令ブロック370が追加されている。ログ出力命令ブロック370は、ユーザが所定の操作を行うことで追加される。また、図7に示すログ出力命令ブロック370を任意のタイミングで削除することもできる。すなわち、ユーザプログラム186に対するログ出力命令の追加および削除は、サポート装置300においてユーザプログラム186のソースコードを編集する際に行うこともできるし、P L C 1 0 0 がユーザプログラム186を実行中に(つまり、オンラインで)ログ出力命令を追加または編集できる(オンラインエディット)ようにしてもよい。すなわち、サポート装置300では、P L C 1 0 0 がユーザプログラム186を実行中に、ログ出力命令の追加または削除が可能であることが好ましい。このようなオンラインエディットの機能を提供することで、ユーザプログラム186に対するデバッグ

50

をより効率的に行うことができる。

【 0 0 6 8 】

このように、サポート装置 3 0 0 は、P L C 1 0 0 で実行されるユーザプログラム 1 8 6 の任意の位置にログ出力命令を追加する処理、および、ユーザプログラム 1 8 6 に含まれるログ出力命令を任意に削除する処理の少なくとも一方を実行するためのユーザインターフェイスを提供する。

【 0 0 6 9 】

< G . ユーザインターフェイス (デバッグログの表示) >

次に、本実施の形態に係るサポート装置 3 0 0 により提供される稼動ログの取得および表示に係るユーザインターフェイスについて説明する。上述のようにログ出力命令が追加されることで出力された稼動ログ 1 8 4 について、サポート装置 3 0 0 にて確認することが可能である。すなわち、サポート装置 3 0 0 は、P L C 1 0 0 の記憶部 (メモリカード 1 7 2 など) に格納される情報を時系列に表示するユーザインターフェイスを提供する。

【 0 0 7 0 】

図 8 は、本実施の形態に係るサポート装置 3 0 0 において提供される稼動ログ 1 8 4 の取得および表示に係るユーザインターフェイスの一例を示す図である。

【 0 0 7 1 】

図 8 に示すユーザインターフェイス画面 3 8 0 において、ユーザが取得ボタン 3 9 0 を押下すると、サポート装置 3 0 0 と接続された P L C 1 0 0 から稼動ログ 1 8 4 が取得される。この稼動ログ 1 8 4 は、基本的には、U S B コネクタ 1 3 0 を介して、P L C 1 0 0 からサポート装置 3 0 0 へ送信される。但しこれに限られるものではなく、P L C 1 0 0 が提供する F T P (File Transfer Protocol) サーバ機能や H T T P (Hyper Text Transfer Protocol) サーバ機能などを用いて、ネットワーク経由で送信してもよい。

【 0 0 7 2 】

図 8 に示すユーザインターフェイス画面 3 8 0 には、稼動ログが時系列に表示されている。一例として、図 8 に示す稼動ログのうち、エントリ「 2 」のログ 3 8 6 がログ出力命令によって生成されたものである。

【 0 0 7 3 】

具体的には、各ログは、日時、カテゴリ、ログコード、ログ名、結果、接続名、シリアル I D といった事項を含む。日時は、対応するイベントが発生した時刻を示す。カテゴリおよびログコードは、対応するログの属性を示し、対応するログを生成したアプリケーションなどを特定することができる。ログ名は、対応するログの内容を一見して把握できる情報を含み、ログ出力命令によって生成されたログでは、ユーザが任意にログ名を設定することもできる。結果は、対応するログの情報を示すものであり、エラーなどが発生した場合には、そのエラーコードが結果として出力される場合もある。接続名およびシリアル I D は、主として、D B 接続に関する情報であり、予め設定されている接続の設定のうち、いずれの設定が使用されたかといった情報を含む。

【 0 0 7 4 】

なお、図 8 に示すユーザインターフェイス画面 3 8 0 のタブに表示の「 S Q L 失敗ログ」は上述の「イベントログ」に対応する。

【 0 0 7 5 】

図 8 に示す稼動ログによれば、まず、D B 接続サービス (図 5) が開始されると、その事象の情報を含むログ 3 8 2 が生成され、続いて、データベース装置 4 0 0 への接続要求が実行されると、その事象の情報を含むログ 3 8 4 が生成されたことがわかる。このとき、図 7 に示すようなログ出力命令が追加されていると、そのログ出力命令によって指定された情報を含むログ 3 8 6 が生成される。その後、接続済みのデータベース装置 4 0 0 に対して、何らかのデータを追加するような要求が実行されると、その事象の情報を含むログ 3 8 8 が生成される。

【 0 0 7 6 】

ログ出力命令ブロック 3 7 0 (図 6) の設定内容とログ 3 8 6 の内容との対応関係の一

10

20

30

40

50

例を示す。ログ出力命令ブロック370では、出力するログの内容についてもユーザが任意に指定することもできる。例えば、ログ出力命令ブロック370のLog Code、Log Name、Error IDは、それぞれログコード、ログ名、結果に対応する。これらの値をユーザプログラム186においてユーザが予め設定しておくことで、不具合の特定などをより容易に行うことができる。さらに、ログ出力命令ブロック370のLog Msgには、通常のログとして格納される情報量に加えて、より多くの情報量をもつ内容を出力させることができる。このより詳細な内容は、ユーザインターフェイス画面380の詳細情報の欄に表示される。

【0077】

< H . 処理手順 >

次に、本実施の形態に係るPLC100におけるログ出力機能に係る処理手順について説明する。図9は、本実施の形態に係るPLC100における処理手順を示すフローチャートである。図9に示す各ステップは、CPUユニット104のプロセッサ120がユーザプログラム186およびシステムプログラム188をそれぞれ実行することで実現される。なお、図9には、ユーザプログラム186およびシステムプログラム188が互いに独立して実行される例を示すが、両プログラムを包含する単一のプログラムを実行するようにしてもよい。

10

【0078】

図9を参照して、ユーザプログラム186の実行に関して、プロセッサ120は、予め格納されているユーザプログラム186をロードし、ロードしたユーザプログラム186を予め定められた周期で繰り返し実行する(ステップS100)。なお、このロードされるユーザプログラム186は、コンパイルされて実行可能なオブジェクト形式になっているものとする。但し、ユーザプログラム186をソースコードのままロードする、あるいは、ユーザプログラム186を中間コードにコンパイルしたものをロードするようにしてもよい。この場合には、ロードされたコードを逐次コンパイルしつつ、プロセッサ120は、処理を実行することになる。

20

【0079】

すなわち、プロセッサ120は、ユーザプログラム186に含まれる命令に従って、必要に応じてシステムプログラム188を呼び出しつつ、指定された処理を実行する(ステップS102)。

30

【0080】

この実行にあたって、実行ログの出力対象になっている処理を実行すると(ステップS104においてYES)、プロセッサ120は、当該処理を実行した処理結果などの情報を時刻情報と関連付けて実行ログとして出力する(ステップS106)。

【0081】

また、この実行にあたって、エラーなどの予め定められた事象が発生すると(ステップS108においてYES)、プロセッサ120は、当該エラーなどの情報を時刻情報と関連付けてイベントログとして出力する(ステップS110)。

【0082】

また、この実行にあたって、ログ出力命令が要求されると(ステップS112においてYESの場合)、当該ログ出力命令によって指定された情報を時刻情報と関連付けてデバッグログとして出力する(ステップS114)。

40

【0083】

一連のユーザプログラム186の実行が完了すると、ステップS102以下の処理が繰り返される。

【0084】

図9を参照して、システムプログラム188の実行に関して、プロセッサ120は、予め格納されているシステムプログラム188をロードし、ロードしたシステムプログラム188を予め定められた周期で繰り返し実行する(ステップS200)。なお、このロードされるシステムプログラム188は、コンパイルされて実行可能なオブジェクト形式に

50

なっているものとする。但し、ユーザプログラム 186 と同様に、システムプログラム 188 をソースコードのままロードする、あるいは、中間コードにコンパイルしたものをロードするようにしてもよい。

【0085】

プロセッサ 120 は、ログ出力命令の追加がサポート装置 300 から要求されたか否かを判断する（ステップ S202）。ログ出力命令の追加がサポート装置 300 から要求されていれば（ステップ S202 において YES の場合）、プロセッサ 120 は、ユーザプログラム 186 の指定された位置に指定された内容のログ出力命令を追加して更新する（ステップ S204）。

【0086】

また、プロセッサ 120 は、ログ出力命令の削除がサポート装置 300 から要求されたか否かを判断する（ステップ S206）。ログ出力命令の削除がサポート装置 300 から要求されていれば（ステップ S206 において YES の場合）、プロセッサ 120 は、ユーザプログラム 186 の指定されたログ出力命令を削除して更新する（ステップ S208）。

【0087】

すなわち、ステップ S202 ~ S208 は、ユーザプログラム 186 を実行中にログ出力命令を追加または編集するための機能（オンラインエディット）に係る処理である。

【0088】

その後、プロセッサ 120 は、稼働ログ 184 の送信がサポート装置 300 から要求されたか否かを判断する（ステップ S210）。稼働ログ 184 の送信がサポート装置 300 から要求されていれば（ステップ S210 において YES の場合）、プロセッサ 120 は、格納している稼働ログ 184 を読み出してサポート装置 300 へ送信する（ステップ S212）。

【0089】

すなわち、ステップ S210 および S212 の処理は、サポート装置 300 から稼働ログ 184 の取得が要求された場合の処理である。

【0090】

一連のシステムプログラム 188 の実行が完了すると、ステップ S202 以下の処理が繰り返される。

【0091】

< I . 変形例 >

上述の実施の形態においては、1つのログファイル 1842 に、すべての稼働ログ（実行ログ、イベントログ、デバッグログ）を時系列に格納する構成について例示したが、当該発生した事象の情報が時刻情報と関連付けて格納されれば、1つのログファイル 1842 にこれらの情報を格納する必要はない。

【0092】

図 10 は、本実施の形態の変形例に係る稼働ログ 184 の格納方法を説明するための図である。図 10 に示すように、例えば、PLC 100 において、ログの種類別に 3 つのファイル 1844 , 1846 , 1848 を用意しておき、各ファイルに対応する種類のログを書き込むようにしてもよい。

【0093】

この場合、PLC 100 またはサポート装置 300 において、これらのファイルの内容をマージするとともに、時系列にソートすることで、図 8 に示すような稼働ログ 184 を取得することができる。このようなマージおよびソートの処理は、公知の技術を採用することで実現できる。

【0094】

< J . 利点 >

本実施の形態に係る制御システム 1 によれば、ユーザは、ログ出力命令をユーザプログラム 186 に対して任意に追加することができる。PLC 100 は、ユーザプログラム 1

10

20

30

40

50

86の実行に伴って、予め定められた処理に係るログ(実行ログ)および処理が失敗した場合などに発生するログ(イベントログ)を自動的に出力するようになっている。ユーザが追加するログ出力命令によって発生するデバッグログは、これらの実行ログおよびイベントログとともに、時系列に格納される。そして、このように時系列に格納された稼動ログをユーザは事後的に参照することができる。

【0095】

このような構成を採用することで、ユーザは、ユーザプログラム186が適切に実行されているか、および、ユーザプログラム186の実行時に不具合が生じたときに、何が原因であるかを容易に特定できる。これによって、プログラムのデバッグや健全性のチェックを効率的に行うことができる。

10

【0096】

今回開示された実施の形態はすべての点で例示であって制限的なものではないと考えられるべきである。本発明の範囲は、上記した説明ではなく、特許請求の範囲によって示され、特許請求の範囲と均等の意味および範囲内でのすべての変更が含まれることが意図される。

【符号の説明】

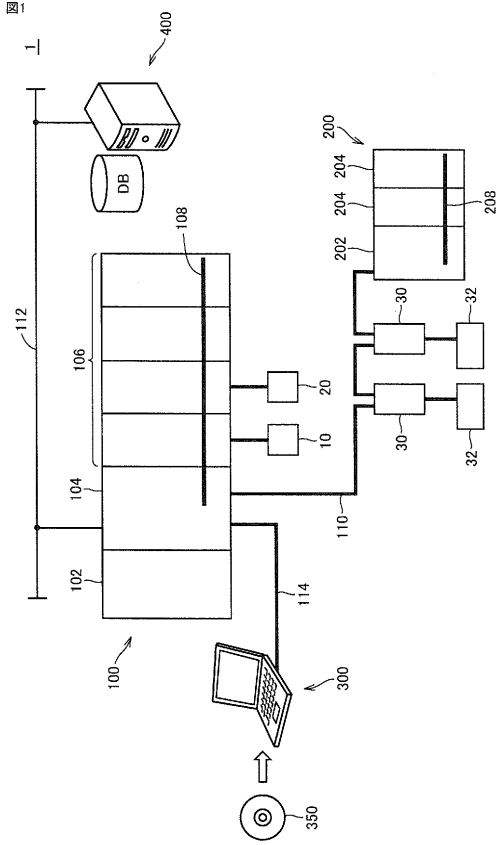
【0097】

1 制御システム、10 検出スイッチ、20 リレー、30 サーボモータドライバ、32 モータ、100 PLC、102 電源ユニット、104 CPUユニット、106 IOユニット、108 PLCシステムバス、110 フィールドバス、112 ネットワーク、114 接続ケーブル、120 プロセッサ、122 チップセット、124 システムクロック、126 主メモリ、128 不揮発性メモリ、130 USBコネクタ、140 PLCシステムバスコントローラ、142, 152, 162 バッファメモリ、144 PLCシステムバス制御回路、146, 156, 166 DMA制御回路、148 PLCシステムバスコネクタ、150 フィールドバスコントローラ、154 フィールドバス制御回路、158 フィールドバスコネクタ、160 上位通信コントローラ、164 上位通信制御回路、168 上位通信コネクタ、170 メモリカードインターフェイス、172 メモリカード、180 OS、182 データ定義情報、184 稼動ログ、1842 ログファイル、186 ユーザプログラム、188 システムプログラム、190 シーケンス命令プログラム、192 アクセス処理プログラム、194 入出力処理プログラム、196 インターフェイス処理プログラム、198 スケジューラ、200 リモートIOターミナル、202 通信カブラ、208 ターミナルバス、300 サポート装置、302 CPU、304 ROM、306 RAM、308 ハードディスク、310 キーボード、312 マウス、314 モニタ、316 光学ディスク読取装置、330 サポートプログラム、332 光学記録媒体、400 データベース装置。

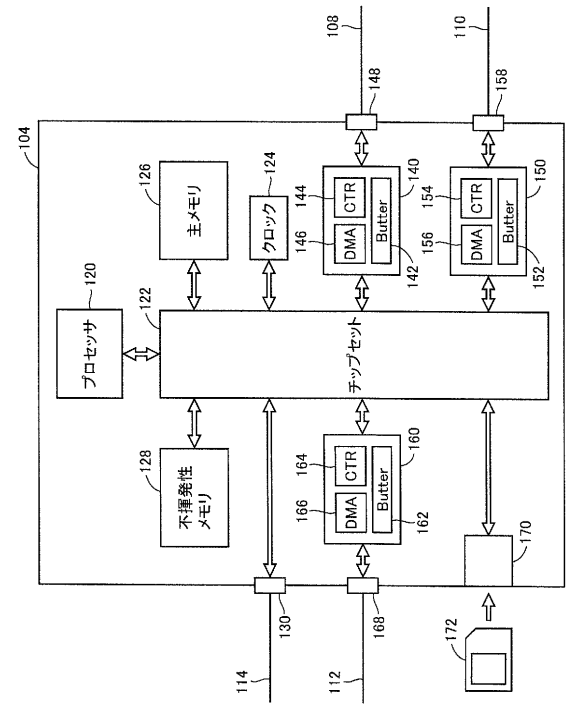
20

30

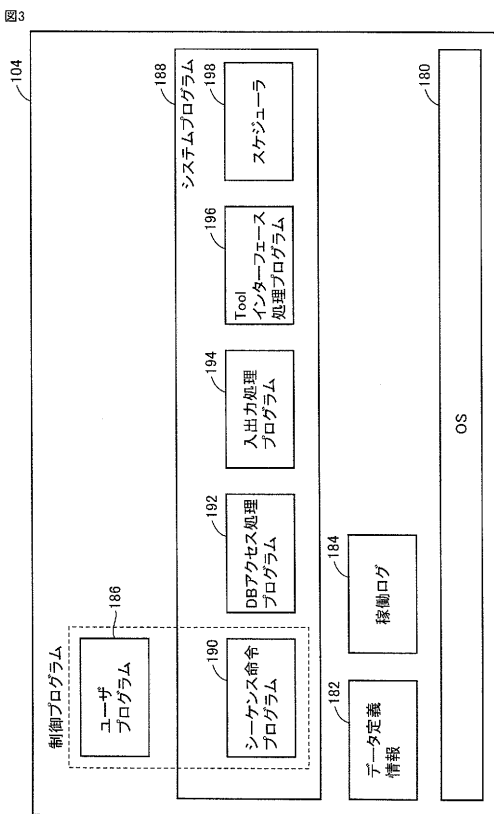
【図 1】



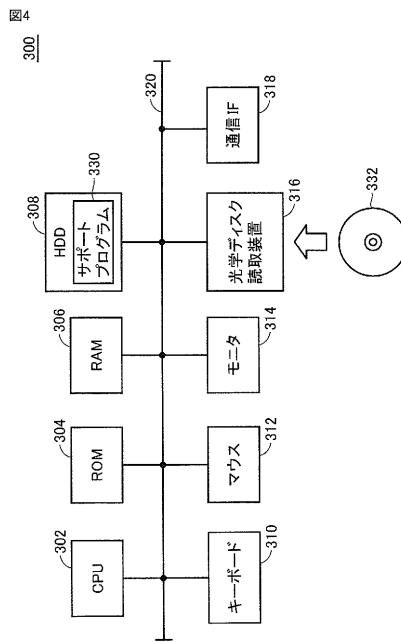
【図 2】



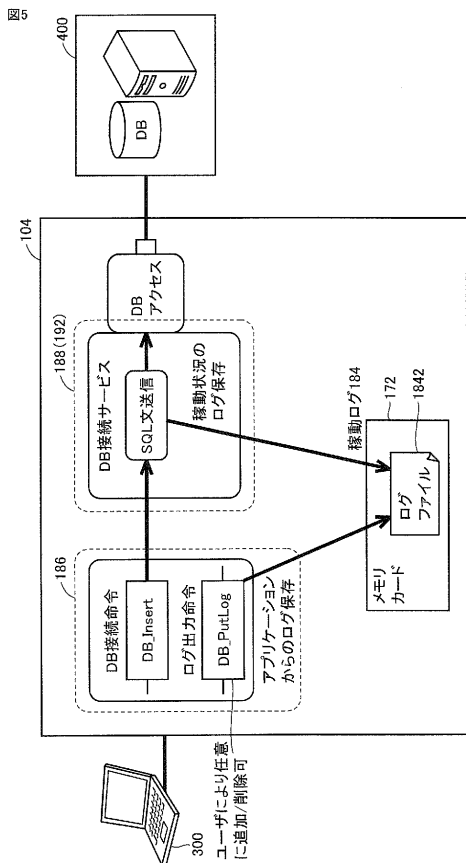
【図 3】



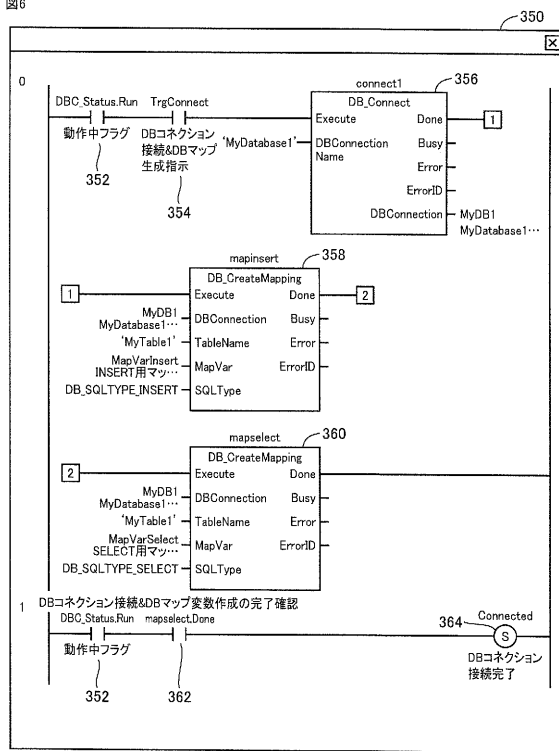
【図 4】



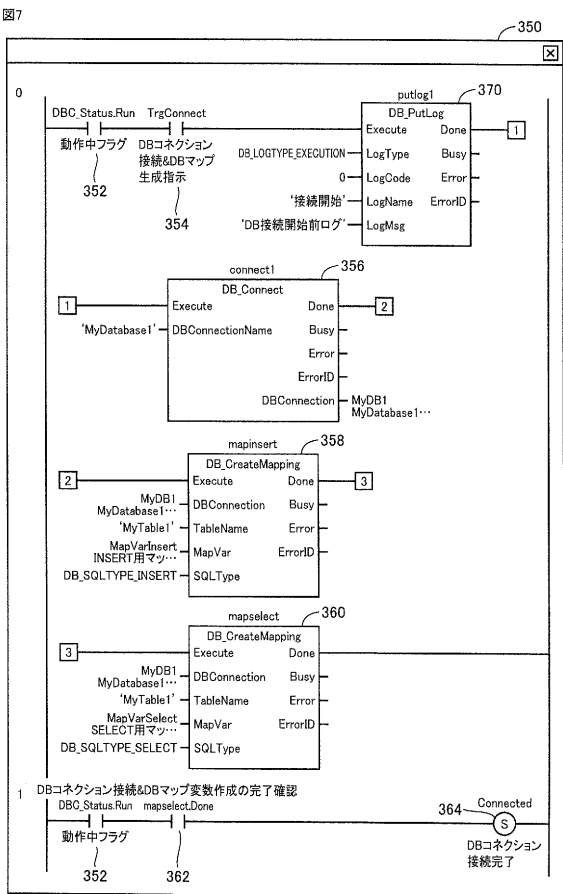
【図5】



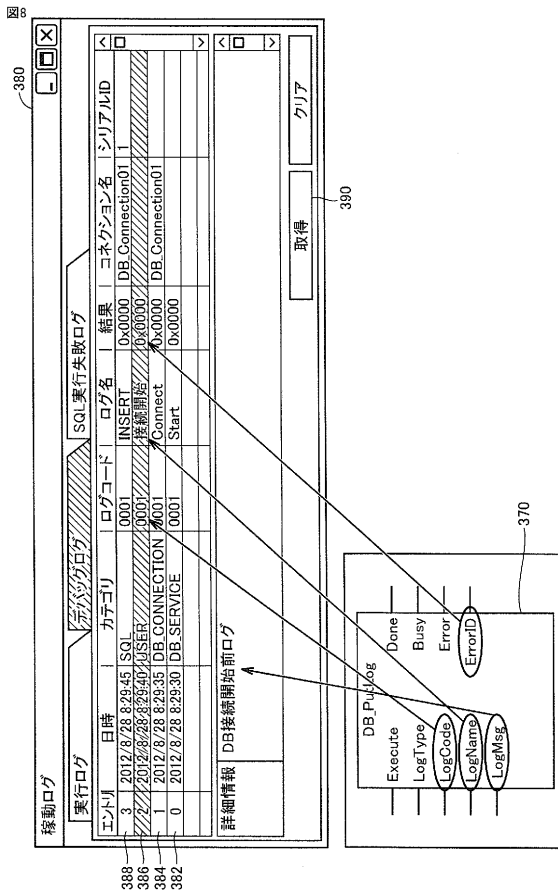
【図6】



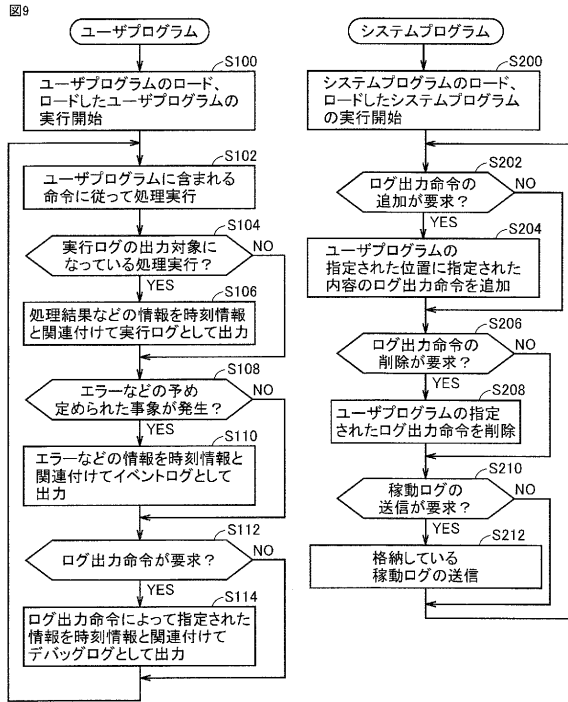
【図7】



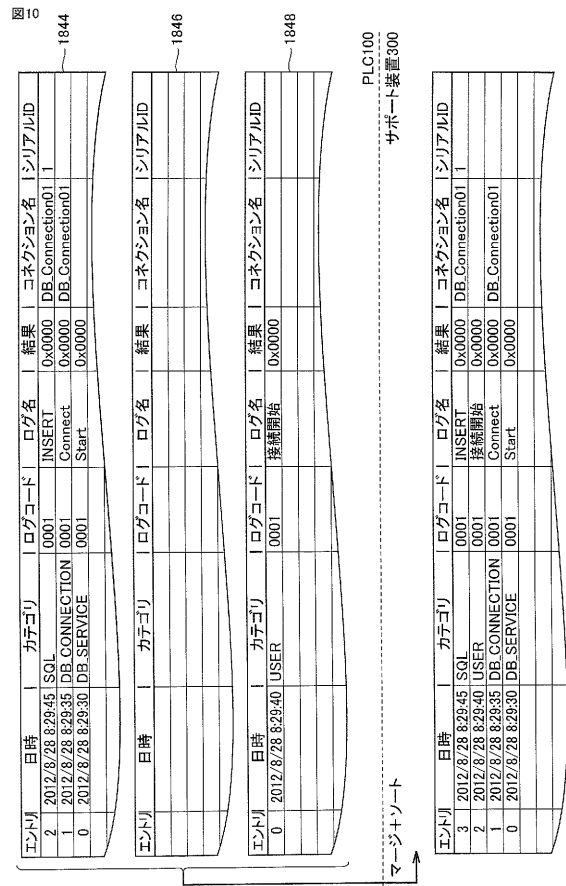
【図8】



【 図 9 】



【 図 10 】



フロントページの続き

(72)発明者 矢尾板 宏心

京都府京都市下京区塩小路通堀川東入南不動堂町 8 0 1 番地 オムロンソフトウェア株式会社内

Fターム(参考) 5H220 AA04 BB10 CC06 CX01 DD04 FF01 FF03 FF05 FF10 GG03

GG05 GG13 HH03 JJ12 JJ16 JJ28 JJ42