

(19)日本国特許庁(JP)

(12)特許公報(B2)

(11)特許番号
特許第7003951号
(P7003951)

(45)発行日 令和4年1月21日(2022.1.21)

(24)登録日 令和4年1月6日(2022.1.6)

(51)国際特許分類		F I			
G 0 5 B	19/05 (2006.01)	G 0 5 B	19/05	Z	
G 0 6 F	15/173 (2006.01)	G 0 6 F	15/173	6 7 5	
G 0 6 F	11/20 (2006.01)	G 0 6 F	11/20	6 2 0	

請求項の数 11 (全36頁)

(21)出願番号	特願2019-46796(P2019-46796)	(73)特許権者	000002945 オムロン株式会社 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
(22)出願日	平成31年3月14日(2019.3.14)	(74)代理人	110001195 特許業務法人深見特許事務所
(65)公開番号	特開2020-149402(P2020-149402 A)	(72)発明者	米田 光宏 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
(43)公開日	令和2年9月17日(2020.9.17)	(72)発明者	阪谷 信幸 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
審査請求日	令和2年12月15日(2020.12.15)	(72)発明者	赤井 亮太 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

(57)【特許請求の範囲】

【請求項1】

1以上の制御装置および情報処理装置を備える制御システムであって、前記情報処理装置と前記1以上の制御装置は、通信可能に接続されて互いに時刻同期され、前記情報処理装置は、各前記1以上の制御装置から受信するデータに基づく制御演算に従う制御データを算出する制御演算処理を実行する処理部を、含み、前記各制御装置は、フィールド装置からの入来データを前記情報処理装置へ転送し、前記情報処理装置から受信する前記制御データを前記フィールド装置に出力する入出力処理を実行する制御部を、含み、前記制御演算処理および前記入出力処理は、前記時刻同期に基づく時間であって前記各制御装置と前記情報処理装置との間で共有される時間内で実行される、制御システム。

【請求項2】

前記処理部が実行する処理は、さらに、前記共有される時間内で、前記制御演算処理を実行するための時間が不足するとき、前記共有される時間を長くする処理を含む、請求項1に記載の制御システム。

【請求項3】

前記共有される時間は制御周期を含み、前記入出力処理は、

制御周期において、当該制御周期以前の複数の制御周期に対応する前記入力データを前記情報処理装置へ転送し、前記情報処理装置から受信する前記複数の制御周期に対応する前記制御データを前記フィールド装置に出力する処理を含む、請求項 1 または 2 に記載の制御システム。

【請求項 4】

前記共有される時間は制御周期を含み、

前記入出力処理は、

1 の制御周期に対応する前記入力データを前記情報処理装置へ転送し、前記情報処理装置から受信する当該入力データに基づく複数の制御周期に対応する制御データを、前記フィールド装置に出力する処理を含む、請求項 1 または 2 に記載の制御システム。

10

【請求項 5】

前記入出力処理は、

制御周期において、前記情報処理装置から前記複数の制御周期に対応する制御データを受信したとき、当該複数の制御周期に対応する制御データを、それぞれ、当該制御周期よりも後の複数の制御周期において、前記フィールド装置に出力する処理を含む、請求項 3 または 4 に記載の制御システム。

【請求項 6】

前記情報処理装置と前記各制御装置は、前記共有される時間内で受付ける共通のトリガに応じて処理を開始し、

前記情報処理装置は、

前記トリガの受け付け時間と前記入出力処理にかかる時間とに基づき予測される前記データの受信開始時間から遡る一定時間内において、前記制御演算処理のための準備処理を実行する、請求項 1 から 5 のいずれか 1 項に記載の制御システム。

20

【請求項 7】

前記準備処理は、前記制御演算処理を除く情報処理の実行を禁止する処理を含む、請求項 6 に記載の制御システム。

【請求項 8】

前記情報処理装置に複数の前記制御装置が通信可能に接続されて、

前記制御演算処理は、それぞれが各前記複数の制御装置に対応する複数の制御演算処理を含み、

前記処理部は、

前記複数の制御装置からデータを受信したとき前記複数の制御演算処理を実行開始する第 1 モード、または、前記複数の制御装置のうちの 1 つからデータを受信したとき当該 1 の制御装置に対応する前記制御演算を実行開始する第 2 モードを備える、請求項 1 から 7 のいずれか 1 項に記載の制御システム。

30

【請求項 9】

前記処理部は、前記第 1 モードおよび前記第 2 モードを備える場合、外部指示に従い前記第 1 モードおよび前記第 2 モードの一方を実施する、請求項 8 に記載の制御システム。

【請求項 10】

複数の前記フィールド装置がフィールドネットワークを介して前記制御装置に接続され、

前記制御部は、

前記入出力処理において、前記フィールドネットワークを介して、前記複数のフィールド装置から入力データを収集し、および前記情報処理装置から受信する前記制御データを前記複数のフィールド装置へ転送する、請求項 1 から 9 のいずれか 1 項に記載の制御システム。

40

【請求項 11】

前記情報処理装置に複数の前記制御装置が通信可能に接続されて、

前記複数の制御装置は稼働系として動作する制御装置と、待機系として動作する制御装置とを含み、

前記待機系の制御装置は、所定タイミングに応じて稼働を開始し、

50

前記情報処理装置は、前記所定タイミングに応じて、前記稼働系の制御装置とのデータ通信を、前記待機系との制御装置とのデータ通信に切替える、請求項 1 から 10 のいずれか 1 項に記載の制御システム。

【発明の詳細な説明】

【技術分野】

【0001】

本開示は、制御対象を制御する制御システムに関する。

【背景技術】

【0002】

生産ラインに備えられる F A (Factory Automation) には、P L C (プログラマブルロジックコントローラ) などの制御装置が用いられる。このような制御装置は、生産ラインに設けられた 1 または複数のフィールド装置である制御対象との間でバスまたはネットワークを介してデータを送受信する。制御装置は、フィールド装置から受信したデータをユーザプログラム (User Program、以下 U P G と略する場合がある) に従う制御演算処理を実行し、処理結果に基づく制御データをフィールド装置に送信する。フィールド装置は制御データに従い制御されて、制御結果によるデータを制御装置に出力する。

10

【0003】

近年は、F A システムの高機能化に伴い、効率的なシステムの運用が望まれている。例えば、特開 2018 - 73436 号公報 (特許文献 1) は、制御機器毎のパラメータ設定を、制御機器とネットワークを介して通信可能に接続されるサーバにより実施することで、パラメータの簡単な管理を可能にしている。

20

【先行技術文献】

【特許文献】

【0004】

【文献】特開 2018 - 73436 号公報

【発明の概要】

【発明が解決しようとする課題】

【0005】

また、F A システムの高機能化に伴い、U P G に従う制御演算処理にかかる処理量または処理時間が増加している。したがって、U P G の制御演算処理にかかる処理量または処理時間の増加に伴う効率的なシステム運用の対策が望まれている。しかし、特許文献 1 は、パラメータ設定に関する構成を開示するが、U P G の制御演算処理にかかる処理量または処理時間の増加に伴う効率的なシステム運用のための技術は提案しない。

30

【0006】

本開示の 1 つの目的は、制御演算処理に関して効率的なシステムの運用を可能にする制御システムを提供することである。

【課題を解決するための手段】

【0007】

本開示の一例は、1 以上の制御装置および情報処理装置を備える制御システムであって、情報処理装置と 1 以上の制御装置は、通信可能に接続されて互いに時刻同期され、情報処理装置は、各 1 以上の制御装置から受信するデータに基づく制御演算に従う制御データを算出する制御演算処理を実行する処理部を、含み、各制御装置は、フィールド装置からの入来データを情報処理装置へ転送し、情報処理装置から受信する制御データをフィールド装置に出力する入出力処理を実行する制御部を、含み、制御演算処理および入出力処理は、時刻同期に基づく時間であって各制御装置と情報処理装置との間で共有される時間内で実行される。

40

【0008】

上述の開示によれば、互いに時刻同期される情報処理装置と制御装置を備える制御システムにおいて、情報処理装置の制御演算処理と、制御装置の入出力処理とを共有される時間内で協調して実行する環境が提供される。ここでは、制御装置は制御演算処理の実行環境

50

を備える必要はないので、制御装置は制御演算処理に要求される処理量または処理時間の制約を受けずに（独立に）フィールド装置を制御することができる。

【 0 0 0 9 】

上述の開示において処理部が実行する処理は、さらに、共有される時間内で、制御演算処理を実行するための時間が不足するとき、共有される時間を長くする処理を含む。

【 0 0 1 0 】

上述の開示によれば、共有される時間内において制御演算処理を実行するための時間を確保することができる。

【 0 0 1 1 】

上述の開示において、共有される時間は制御周期を含み、入出力処理は、制御周期において、当該制御周期以前の複数の制御周期に対応する入来データを情報処理装置へ転送し、情報処理装置から受信する複数の制御周期に対応する制御データをフィールド装置に出力する処理を含む。

10

【 0 0 1 2 】

上述の開示によれば、制御装置は、複数の制御周期の入来データをまとめて、情報処理装置に送信することで、制御周期毎に送信する場合に比較して、情報処理装置と制御装置との間の通信時間を節約できる。また、複数制御周期分のデータが通信されるので、ある制御周期のデータ落ち（喪失）があっても、他の制御周期のデータで補完することが可能となる。

【 0 0 1 3 】

上述の開示において、共有される時間は制御周期を含み、入出力処理は、1の制御周期に対応する入来データを前記情報処理装置へ転送し、情報処理装置から受信する当該入来データに基づく複数の制御周期に対応する制御データを、フィールド装置に出力する処理を含む。

20

【 0 0 1 4 】

上述の開示によれば、制御装置は、1の制御周期の入来データを情報処理装置に送信することで、複数の制御周期に対応する制御データを取得することができる。

【 0 0 1 5 】

上述の開示において、入出力処理は、制御周期において、情報処理装置から複数の制御周期に対応する制御データを受信したとき、当該複数の制御周期に対応する制御データを、それぞれ、当該制御周期よりも後の複数の制御周期において、フィールド装置に出力する処理を含む。

30

【 0 0 1 6 】

上述の開示によれば、情報処理装置は、その後の複数の制御周期の制御データをまとめて、制御装置に送信することで、制御周期毎に送信する場合に比較して、情報処理装置と制御装置との間の通信時間を節約できる。また、複数制御周期分のデータが通信されるので、ある制御周期のデータ落ち（喪失）があっても、他の制御周期のデータで補完することが可能となる。

【 0 0 1 7 】

上述の開示において、情報処理装置と各制御装置は、共有される時間内で受付ける共通のトリガに応じて処理を開始し、情報処理装置は、トリガの受付け時間と入出力処理にかかる時間とに基づき予測されるデータの受信開始時間から遡る一定時間内において、制御演算処理のための準備処理を実行する。

40

【 0 0 1 8 】

上述の開示によれば、制御演算処理が開始される前に、準備処理を実行する時間を確保することができる。

【 0 0 1 9 】

上述の開示において、準備処理は、制御演算処理を除く情報処理の実行を禁止する処理を含む。

【 0 0 2 0 】

50

上述の開示によれば、制御演算処理が開始される前は他の情報処理の実行を禁止することで、その後の制御演算処理のためのリソースを確実に準備することができる。

【 0 0 2 1 】

上述の開示において、情報処理装置に複数の制御装置が通信可能に接続されて、制御演算処理は、それぞれが各複数の制御装置に対応する複数の制御演算処理を含み、処理部は、複数の制御装置からデータを受信したとき複数の制御演算処理を実行開始する第 1 モード、または、複数の制御装置のうちの 1 つからデータを受信したとき当該 1 の制御装置に対応する制御演算を実行開始する第 2 モードを備える。

【 0 0 2 2 】

上述の開示によれば、情報処理装置において、複数の制御演算処理を実行開始するモードとして、同期して開始させるような上記の第 1 モード、または、同期せずに非同期に（対応の制御装置からデータを受信した時に）開始させるような上記の第 2 モードが提供される。

10

【 0 0 2 3 】

上述の開示において、処理部は、第 1 モードおよび第 2 モードを備える場合、外部指示に従い第 1 モードおよび第 2 モードの一方を実施する。

【 0 0 2 4 】

上述の開示によれば、外部からの指示により、情報処理装置に第 1 モードおよび第 2 モードのいずれかを実行させることができる。

【 0 0 2 5 】

上述の開示において、複数のフィールド装置がフィールドネットワークを介して制御装置に接続され、制御部は、入出力処理において、フィールドネットワークを介して、複数のフィールド装置から入来データを収集し、および情報処理装置から受信する制御データを複数のフィールド装置へ転送する。

20

【 0 0 2 6 】

上述の開示によれば、制御装置は、入出力処理において、フィールド装置からの入来データを収集し、および情報処理装置から受信する制御データを複数のフィールド装置へ転送することができる。

【 0 0 2 7 】

上述の開示において、情報処理装置に複数の制御装置が通信可能に接続されて、複数の制御装置は稼働系として動作する制御装置と、待機系として動作する制御装置とを含み、待機系の制御装置は、所定タイミングに応じて稼働を開始し、情報処理装置は、所定タイミングに応じて、稼働系の制御装置とのデータ通信を、待機系との制御装置とのデータ通信に切替える。

30

【 0 0 2 8 】

上述の開示によれば、制御システムが備える複数の制御装置を、稼働系または待機系として動作させて、所定タイミングに応じて待機系の制御装置を稼働系に切替えることができる。所定タイミングが故障発生を含む場合は、故障発生後であっても制御システムを継続して運転することが可能となる。

【 発明の効果 】

40

【 0 0 2 9 】

本開示によれば、制御演算処理に関して効率的なシステムの運用を可能にする制御システムを提供できる。

【 図面の簡単な説明 】

【 0 0 3 0 】

【 図 1 】 本実施の形態にかかる制御システム 1 の構成の一例を模式的に示す図である。

【 図 2 】 本実施の形態に従う制御システム 1 のネットワーク構成例を示す模式図である。

【 図 3 】 本実施の形態にかかる P L C 1 0 0 の構成例を示すブロック図である。

【 図 4 】 本実施の形態にかかるサーバ 6 0 0 の構成例を示す図である。

【 図 5 】 本実施の形態にかかる P L C 1 0 0 のソフトウェア構成例を示すブロック図であ

50

る。

【図 6】本実施の形態にかかるサーバ 600 のソフトウェア構成例を示す図である。

【図 7】本実施の形態にかかる制御システム 1 の基本系を模式的に示す図である。

【図 8】本実施の形態にかかる基本系のタイミングチャートの一例を模式的に示す図である。

【図 9】本実施の形態にかかる基本系における処理のフローチャートの一例を示す図である。

【図 10】本実施の形態にかかる基本系における処理のフローチャートの変形例を示す図である。

【図 11】本実施の形態にかかる制御システム 1 の多重化系を模式的に示す図である。

10

【図 12】本実施の形態にかかる多重化系のタイミングチャートの一例を模式的に示す図である。

【図 13】本実施の形態にかかる多重化系のタイミングチャートの他の例を模式的に示す図である。

【図 14】本実施の形態にかかる多重化系における処理のフローチャートの一例を示す図である。

【図 15】本実施の形態にかかる多重化系の応用例を模式的に示す図である。

【図 16】本実施の形態にかかる多重化系の他の応用例を模式的に示す図である。

【図 17】本実施の形態にかかるオーバサンプリングの一例を模式的に説明する図である。

【図 18】図 17 のオーバサンプリングにより格納されるデータの一例を模式的に示す図である。

20

【図 19】本実施の形態にかかるオーバサンプリングの他の例を模式的に説明する図である。

【図 20】本実施の形態にかかる基本系における処理のフローチャートの変形例を示す図である。

【図 21】本実施の形態にかかる多重化系の処理のフローチャートの変形例を示す図である。

【発明を実施するための形態】

【0031】

本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。

30

【0032】

< A . 適用例 >

まず、本発明が適用される場面の一例について説明する。図 1 は、本実施の形態にかかる制御システム 1 の構成の一例を模式的に示す図である。図 1 を参照して、FA に適用され得る制御システム 1 では、例えば「制御装置」の一実施例である PLC 100A, 100B および 100C および「情報処理装置」の一実施例であるサーバ 600 を備える。PLC 100A, 100B および 100C は同様の機能を備えており、これらを区別しない場合は PLC 100 と総称する。サーバ 600 には、有線または無線のネットワーク 11 を介して PLC 100 が接続されている。また、ネットワーク 11 にはサポート装置 500 を接続することができる。

40

【0033】

図 1 では、制御システム 1 は、PLC 100 とサーバ 600 を別体として構成されているが、これに限定されない。例えば、制御システム 1 は、PLC 100 の機能とサーバ 600 の機能を内蔵して備える産業用のコンピュータ（所謂 IPC : Industrial Personal Computer）を備えて構成されてもよい。

【0034】

ネットワーク 11 には、PLC 100A, 100B および 100C と、サーバ 600 とが接続されており、ネットワーク 11 では装置間でデータを遣り取りできる。また、サーバ 600 には、FA システム内のサーバ、またはクラウドサーバ等のような FA システム外

50

のサーバを適用することもできる。サーバ600が外部サーバである場合、ネットワーク11はインターネットなどの外部ネットワークを含む。

【0035】

PLC100には、センサ、アクチュエータといった各種のフィールド装置90が接続される。フィールド装置90は、「制御対象」の一例である。これらのフィールド装置90は、PLC100に装着される入出力ユニットを介してPLC100に直接接続される場合もあるが、フィールド側のネットワーク110を介してPLC100に接続されることもある。図1では、PLC100は、1または複数のネットワーク110に接続される。各ネットワーク110には、1または複数のフィールド装置90が接続される。1または複数のフィールド装置90の各々は、製造装置または生産ラインなど（以下、「フィールド」とも総称する。）に対して何らかの物理的な作用を与えるアクチュエータ、およびフィールドとの間で情報を遣り取りする入出力装置などを含む。

10

【0036】

ネットワーク110を介して、PLC100とフィールド装置90との間で遣り取りされるデータは、数100 μ secオーダー～数10msecオーダーのごく短い周期で更新されることになる。なお、このような遣り取りされるデータの更新処理は、I/Oリフレッシュ処理72により実現される。I/Oリフレッシュ処理72は、「入出力処理」の一実施例である。

【0037】

具体的には、本実施の形態では、PLC100は、指定された観測値を含むフレームを生成する機能を有する。PLC100は、生成されるフレームを、ネットワーク11を介してサーバ600に転送（送信）する。観測値の概念は、任意の制御対象についてのデータ（観測値）の時間的な変化を連続的（あるいは、一定間隔をおいて不連続）に観測して得られる一連の値も含み得る。

20

【0038】

本明細書において「観測値」は、PLC100での制御演算において利用可能な値（実値）を総称する概念であり、典型的には、制御対象から取得されて制御演算に入力される値（フィールドから取得された測定値など）などを含み得る。

【0039】

制御システム1では、サーバ600のタイマ601とPLC100Aのタイマ101A、PLC100Bのタイマ101BおよびPLC100Cのタイマ101Cは、互いに時刻同期されている。サーバ600は、PLC100A、100Bおよび100Cから受信する観測値に相当する入力データ（送信）182を用いてUPG（User Program）69を実行することにより、入力データ（送信）182に基づいた制御演算処理を実行する。サーバ600は、制御演算処理による算出値である出力データ（受信）183をPLC100に送信する。算出値は、フィールド装置90の「制御データ」の一実施例に相当し、例えば指令値、制御量などを示す。各PLC100は、フィールド装置90からの観測値に相当の入来データを収集しサーバ600へ転送し、サーバ600から受信する制御データをフィールド装置90に出力するI/Oリフレッシュ処理72を実行する。サーバ600におけるUPG69の実行による制御演算処理および各PLC100におけるI/Oリフレッシュ処理72は、上記した時刻同期に基づく時間であって各PLC100とサーバ600との間で共有される時間内で実行される。

30

40

【0040】

図1では、時刻同期に基づく共有される時間内においてサーバ600とPLC100がデータを遣り取りしながら協調制御が実施される。具体的には、この共有される時間内において、(i) PLC100はI/Oリフレッシュ処理72によりフィールド装置90からの出力データ（観測値）を収集し、(ii) サーバ600はI/Oリフレッシュ処理72により収集されたフィールド装置90からの出力データに基づく制御演算処理を実施し、(iii) PLC100はI/Oリフレッシュ処理72によりサーバ600からの算出値（指令値、制御量）を効果）をフィールド装置90に出力する、一連の処理が実施される。制御システム

50

1では、UPG69の実行環境はPLC100ではなくサーバ600に備えられる。したがって、協調制御では、PLC100の処理能力(所謂スペック)を超える演算量、演算速度が要求されるUPG69を実行し、その実行結果を用いてフィールド装置90を制御することが可能となる。

【0041】

以下、本実施の形態のより具体的な応用例について説明する。

< B . 制御システム1のネットワーク構成 >

再び図1を参照して、制御システム1のネットワーク11に接続されるPLC100A、100Bおよび100Cは、それぞれ、工程3A、3Bおよび3Cに備えられる。限定されないが、工程3A、3Bおよび3Cは、同じ工程(例えば、ワークを製品に組立てる工程等)であってもよく、または異なる工程(ワークの組立工程、組立てられた製品の塗装工程、塗装された製品の検査工程等)であってもよい。

10

【0042】

ネットワーク11のプロトコルとしては、限定されないが、例えば汎用的なEthernet(登録商標)上に制御用プロトコルを実装した産業用オープンネットワークであるEthernet/IP(登録商標)を用いてもよい。また、ネットワーク110のプロトコルとしては、限定されないが、マシンコントロール用ネットワークの一例であるEtherCAT(登録商標)を採用してもよい。なお、ネットワーク11のプロトコルとネットワーク110のプロトコルとは、同じであってもよいし、異なってもよい。また、ネットワーク110は、データバスで構成することもできる。

20

【0043】

また、サポート装置500は、ユーザが制御システム1を運用するのを支援する支援ツールを提供する。支援ツールは、サーバ600におけるUPG69の実行環境、PLC100との通信環境等の準備の設定等を支援するツールを含む。サポート装置500は、ネットワーク11に接続され得るが、サーバ600に脱着自在に接続されてもよく、またはサーバ600に支援ツールが内蔵されてもよい。また、支援ツールは、PLC100がフィールド装置90を制御するために必要な準備の設定、サーバ600との通信環境等の準備の設定等を支援するツールを含んでもよく、その場合、サポート装置500は、PLC100に脱着自在に接続され得る。なお、支援ツールは、例えばUI(User Interface)によりユーザに提供される。

30

【0044】

< C . 制御システム1における時刻同期 >

図1の制御システム1では、ネットワーク11に接続される複数のPLC100A、100Bおよび100Cは、それぞれ、ネットワーク110を介して接続される1または複数のフィールド装置90とのデータを送受信する。具体的には、PLC100は、フィールド装置90から入来するデータをサーバ600へ転送し、サーバ600から受信する制御データをフィールド装置90に出力する入出力処理(IOリフレッシュ処理72)を実行する。

【0045】

また、ネットワーク110では、データの到着時間が保証される必要がある。そのため、PLC100は、データ伝送のタイミングを規定する、データが送受信される主体(すなわち、1または複数のフィールド装置90およびサーバ600)の間で互いに時刻同期されたタイマを有している。

40

【0046】

また、本実施の形態に従う制御システム1においては、複数のPLC100の各々が有するタイマを互いに時刻同期させる。これにより、互いに異なるPLC100に接続される、すなわち異なる工程間において複数のフィールド装置90の協調制御が実現され得る。

【0047】

以下、本実施の形態にかかる制御システム1が提供する時刻同期機能について説明する。

【0048】

50

(c 1 . 時刻同期)

図 2 は、本実施の形態に従う制御システム 1 のネットワーク構成例を示す模式図である。

図 2 を参照して、本実施の形態に従う制御システム 1 における時刻同期を説明する。

【 0 0 4 9 】

図 2 に示す制御システム 1 は、複数の P L C 1 0 0 A , 1 0 0 B および 1 0 0 C と、複数のフィールド装置 9 0 A ~ 9 0 I とを備える。制御システム 1 では、P L C 1 0 0 A、1 0 0 B および 1 0 0 C は、それぞれ、対応するネットワーク 1 1 0 内のデータ伝送を管理するマスタとして機能する。フィールド装置 9 0 A ~ 9 0 I は、対応するマスタからの指令に従ってデータ伝送を行なうスレーブ (Slave) として機能する。

【 0 0 5 0 】

本実施の形態に従う制御システム 1 において、ネットワーク 1 1 0 に接続される複数の装置、すなわち P L C 1 0 0 および 1 または複数のフィールド装置 9 0 の間では、送受信タイミングが同期されている (図中の時刻同期 (3) に相当) 。具体的には、P L C 1 0 0 およびフィールド装置 9 0 の各々は、互いに時刻同期されたタイマを備える。このタイマは、同期してインクリメントまたはデクリメントされるカウンタに相当する。P L C 1 0 0 およびフィールド装置 9 0 の各々は、それらの時刻同期されたタイマに従って、データの送信または受信のタイミングを決定する。

【 0 0 5 1 】

なお、本実施の形態では「タイミング」は何らかの事象が生じる時期、時間または時刻の概念を表す。また、「時刻同期」とはお互いの装置が有するタイマ、時間データ等を同期させることを示す。本実施の形態では、この時刻同期は、例えば、ネットワーク 1 1、1 1 0 において T S N (Time-Sensitive Networking) に従うデータ通信が実施されることで実現され得る。

【 0 0 5 2 】

図 2 に示す例においては、P L C 1 0 0 A はタイマ 1 0 2 A を備え、フィールド装置 9 0 A ~ 9 0 C はそれぞれタイマ 9 1 A ~ 9 1 C を備える。P L C 1 0 0 A のタイマ 1 0 2 A がマスタとして機能し、フィールド装置 9 0 A ~ 9 0 C のタイマ 9 1 A ~ 9 1 C がこのマスタを基準として、タイマ 1 0 2 A のタイマ値に基づく値が、タイマ 9 1 A ~ 9 1 C に設定される (時刻同期がとられる) 。

【 0 0 5 3 】

P L C 1 0 0 A と同様に、P L C 1 0 0 B が備えるタイマ 1 0 2 B と、フィールド装置 9 0 D ~ 9 0 F が備えるタイマ 9 1 D ~ 9 1 F との間で時刻同期がとられる。また、P L C 1 0 0 A と同様に、P L C 1 0 0 C が備えるタイマ 1 0 2 C と、フィールド装置 9 0 G ~ 9 0 I が備えるタイマ 9 1 G ~ 9 1 I との間で時刻同期がとられる。

【 0 0 5 4 】

これにより、ネットワーク 1 1 0 を構成する P L C 1 0 0 とフィールド装置 9 0 との間でデータの伝送タイミングなどを互いに一致させることができる。

【 0 0 5 5 】

図 2 に示す例において、P L C 1 0 0 A は、タイマ 1 0 2 A と時刻同期されたタイマ 1 0 1 A を備え、また、P L C 1 0 0 B は、タイマ 1 0 2 B と時刻同期されたタイマ 1 0 1 B を備え、また、P L C 1 0 0 C は、タイマ 1 0 2 C と時刻同期されたタイマ 1 0 1 C を有している (図中の時刻同期 (2) に相当) 。

【 0 0 5 6 】

また、制御システム 1 においては、サーバ 6 0 0 のタイマ 6 0 1 がマスタとして機能し、P L C 1 0 0 A のタイマ 1 0 1 A、P L C 1 0 0 B のタイマ 1 0 1 B および P L C 1 0 0 C のタイマ 1 0 1 C は、それぞれ、マスタのタイマ 6 0 1 と時刻同期する。これにより、各 P L C 1 0 0 はサーバ 6 0 0 と相互に時刻同期する (図中の時刻同期 (1) に相当) 。

【 0 0 5 7 】

また、P L C 1 0 0 A のタイマ 1 0 1 A は、サーバ 6 0 0 のタイマ 6 0 1 と時刻同期し、他の P L C 1 0 0 B , 1 0 0 C のタイマ 1 0 1 B , 1 0 1 C が、タイマ 1 0 1 A と時刻同

10

20

30

40

50

期してもよい。

【 0 0 5 8 】

これにより、制御システム 1 では、各 P L C 1 0 0 とサーバ 6 0 0 とは互いに時刻同期され、さらに各 P L C 1 0 0 と、自 P L C 1 0 0 に接続されるフィールド装置 9 0 とは互いに時刻同期される。

【 0 0 5 9 】

< D . 各装置の構成 >

制御システム 1 が備える各装置の構成を説明する。

【 0 0 6 0 】

(d 1 . P L C 1 0 0 の構成)

図 3 は、本実施の形態にかかる P L C 1 0 0 の構成例を示すブロック図である。図 3 を参照して、P L C 1 0 0 は、プロセッサ 1 0 2 と、チップセット 1 0 4 と、主記憶装置 1 0 6 と、二次記憶装置 1 0 8 と、N I C (Network Interface Card) 等のネットワークコントローラ 1 0 5 と、U S B (Universal Serial Bus) コントローラ 1 0 7 と、メモリカードインターフェイス 1 1 4 と、ローカルバスコントローラ 1 2 0 と、フィールドネットワークコントローラ 1 1 8 と、カウンタ 1 2 6 とを含む。

10

【 0 0 6 1 】

プロセッサ 1 0 2 は、C P U (Central Processing Unit)、M P U (microprocessor unit)、U P G (Graphics Processing Unit) など構成される。プロセッサ 1 0 2 は、二次記憶装置 1 0 8 に格納された各種プログラムを読み出して、読み出されたプログラムを主記憶装置 1 0 6 に展開して実行することで、制御対象に応じた制御、および、後述するような各種処理を実現する。二次記憶装置 1 0 8 は、例えば、H D D (Hard Disk Drive) または S S D (Solid State Drive) などの不揮発性記憶装置などで構成される。主記憶装置 1 0 6 は、D R A M (Dynamic Random Access Memory) や S R A M (Static Random Access Memory) などの揮発性記憶装置などで構成される。

20

【 0 0 6 2 】

チップセット 1 0 4 は、プロセッサ 1 0 2 と各デバイスを制御することで、P L C 1 0 0 全体としての処理を実現する。

【 0 0 6 3 】

二次記憶装置 1 0 8 には、基本的な機能を実現するためのシステムプログラムに加えて、制御対象の製造装置や設備に応じて作成されるユーザプログラムが格納される。

30

【 0 0 6 4 】

ネットワークコントローラ 1 0 5 は、ネットワーク 1 1 を介して、サーバ 6 0 0 との間のデータを遣り取りする。U S B コントローラ 1 0 7 は、U S B 接続を介してサポート装置 5 0 0 との間のデータの遣り取りを制御する。

【 0 0 6 5 】

メモリカードインターフェイス 1 1 4 は、メモリカード 1 1 6 を脱着可能 (detachable) に構成されており、メモリカード 1 1 6 に対してデータを書込み、メモリカード 1 1 6 から各種データ (ユーザプログラムやトレースデータなど) を読み出すことが可能になっている。

40

【 0 0 6 6 】

カウンタ 1 2 6 は、一種のタイマである。カウンタ 1 2 6 は、P L C 1 0 0 における各種処理の実行タイミングを管理するための時刻基準として用いられる。カウンタ 1 2 6 は、典型的には、所定周期毎にカウンタ値をインクリメントまたはデクリメントする。カウンタ 1 2 6 として、プロセッサ 1 0 2 を駆動するシステムバス上に配置された、ハードウェアタイマが実装されてもよく、あるいは、A S I C (Application Specific Integrated Circuit) や F P G A (Field-Programmable Gate Array) などの専用回路を用いて実装されてもよい。図 2 で示したタイマ 1 0 1 A、1 0 1 B および 1 0 1 C は、それぞれ、カウンタ 1 2 6 に対応する。

【 0 0 6 7 】

50

フィールドネットワークコントローラ 118 は、ネットワーク 110 を介したフィールド装置 90 を含む他のデバイスとの間のデータの遣り取りを制御する。フィールドネットワークコントローラ 118 は、他のデバイスとの間でタイミングを管理するための時刻基準として用いられるカウンタ 119 を有している。図 2 に示したタイマ 102A、102B および 102C は、それぞれ、カウンタ 119 に対応する。

【0068】

ローカルバスコントローラ 120 は、PLC 100 に接続され得る図示しない機能ユニットとの間でデータを遣り取りするインターフェイスである。

【0069】

また、ネットワーク 110 上の各フィールド装置 90 も、フィールドネットワークコントローラ 118 との間でタイミングを管理するための時刻基準として用いられるカウンタ 92 を有している。図 2 のタイマ 91A ~ 91I は、それぞれ、カウンタ 92 に対応する。

10

【0070】

カウンタ 119 および各フィールド装置 90 が有するカウンタについては、上述のカウンタ 126 と同様の構成を採用できる。

【0071】

フィールドネットワークコントローラ 118 は、ネットワーク 110 を介した定周期通信を行うための通信マスタとして動作する。フィールドネットワークコントローラ 118 は、フィールド装置 90 のカウンタ 91 が示すカウンタ値をカウンタ 119 が示すカウンタ値と一致させるよう同期管理を実施する。

20

【0072】

図 3 の PLC 100 では、上記に述べた時刻同期により、カウンタ 119 は、カウンタ 126 と同期する。

【0073】

図 3 には、プロセッサ 102 がプログラムを実行することで必要な機能が提供される構成例を示したが、これらの提供される機能の一部または全部を、専用のハードウェア回路（例えば、ASIC (Application Specific Integrated Circuit) または FPGA (Field-Programmable Gate Array) など）を用いて実装してもよい。あるいは、PLC 100 の主要部を、汎用的なアーキテクチャに従うハードウェア（例えば、汎用パソコンをベースとした産業用パソコン）を用いて実現してもよい。この場合には、マルチプロセッサを備えて並列実行可能な環境が提供される。また、仮想化技術を用いて、用途の異なる複数の OS (Operating System) を並列的に実行させるとともに、各 OS 上で必要なアプリケーションを実行させるようにしてもよい。

30

【0074】

(d2. サーバ 600 の構成)

図 4 は、本実施の形態にかかるサーバ 600 の構成例を示す図である。サーバ 600 は、汎用のコンピュータの構成を備える。図 4 を参照して、サーバ 600 は、タイマ 601、CPU または MPU などのプロセッサ 603、記憶装置、PLC 100 を含む他の装置との間でデータを遣り取りするためのネットワークコントローラ 620 を備える。ネットワークコントローラ 620 は、例えば NIC を備えて構成されてもよい。これらのコンポーネントは、内部バス 602 を介して互いにデータ通信可能に接続されている。この記憶装置は、ROM (Read Only Memory) 604、RAM (Random Access Memory) 605 および HDD (Hard Disk Drive) 606 を含む。プロセッサ 603 は、マルチプロセッサを備えて、UPG 69 を並列に実行できるよう構成されてもよい。

40

【0075】

HDD 606 は、OS (Operating System) を含むシステムプログラム 71、サーバプログラム 68、制御演算処理のための UPG 69A、69B、異常を監視するための監視プログラム 65、異常を処理する異常処理プログラム 66、上記に述べた時刻同期を実施するための時刻同期プログラム 67、スケジューラ 63、通信プログラム 64、制御演算とは異なる他の処理を実施するための他のプログラム 611、および後述する「準備時間

50

」を確保する処理に関するリザーブプログラム 612 を格納する。プロセッサ 603 は、これらのプログラムを RAM 605 に読出して実行する。

【0076】

本実施の形態では、HDD 606 は、ユーザプログラムとして、PLC 100A の UPG 69A および PLC 100B の UPG 69B の 2 種類を格納するが、UPG 69A, 69B に加えて、PLC 100C の UPG を格納してもよい。また、ユーザプログラムが PLC 100A, 100B および 100C に共通する場合には、HDD 606 に格納されるユーザプログラムは 1 種類であってもよい。UPG 69A, 69B は区別しない場合は UPG 69 と呼ぶ。UPG 69 は、MCE (Motion Control Engine) 70 を含んでも良い。MCE 70 は、ロボットを目標軌跡に従う移動 (駆動) させるための指令値を生成 (算出) するプログラムである。

10

【0077】

RAM 605 は、後述する入力データ (送信) 182 を格納する受信用データ領域 61 と、後述する出力データ (受信) 183 を格納する送信用データ領域 62 を含む。入力データ (送信) 182 と出力データ (受信) 183 は、データの送信元または宛先である PLC 100 の識別子、変数名および当該変数に対応するデータとからなる。プロセッサ 603 は、ネットワークコントローラ 620 を介して PLC 100 から受信する入力データ (送信) 182 を受信用データ領域 61 に格納する。また、プロセッサ 603 は、受信用データ領域 61 の入力データ (送信) 182 を用いた制御演算処理を実行し、算出値である出力データ (受信) 183 を、送信用データ領域 62 に格納する。入力データ (送信) 182 の受信と出力データ (受信) 183 の送信は、プロセッサ 603 が、通信プログラム 64 を実行することにより実現される。

20

【0078】

また、制御演算処理は、プロセッサ 603 が、UPG 69 を実行することにより実現される。UPG 69 は、限定されないが、例えば、ファンクションブロックなどを利用したラダーロジックなどで規定することができる。

【0079】

(d3. PLC 100 のソフトウェア構成)

次に、本実施の形態にかかる PLC 100 のソフトウェア構成例について説明する。図 5 は、本実施の形態にかかる PLC 100 のソフトウェア構成例を示すブロック図である。図 5 を参照して、PLC 100 は、制御エンジン 150、バッファ 180 と、ゲートウェイプログラム 194 とを含む。バッファ 180 は、典型的には、例えば主記憶装置 106 または二次記憶装置 108 内に、または、ネットワークコントローラ 105 が備えるメモリ内に配置され得る。また、バッファ 180 は、入力データ (送信) 182 と、出力データ (受信) 183 を格納する。

30

【0080】

制御エンジン 150 は、各種プログラムの実行環境を提供する。具体的には、PLC 100 のプロセッサ 102 が、二次記憶装置 108 に格納されているシステムプログラム 151 を読出して主記憶装置 106 に展開して実行することで制御エンジン 150 が実現される。

40

【0081】

制御エンジン 150 は、制御プログラム 152 と、変数管理プログラム 160 と、スケジューラプログラム 170 と、入力プログラム 172 と、出力プログラム 174 と、時刻同期プログラム 177 を実行する。変数管理プログラム 160 と、スケジューラプログラム 170 と、入力プログラム 172 と、出力プログラム 174 とについては、システムプログラム 151 の一部として実装されてもよい。この場合には、これらのプログラムが提供するそれぞれの機能を単一のシステムプログラム 151 が提供するようによい。

【0082】

制御プログラム 152 は、ユーザが提供するプログラムの一例である周辺処理プログラム 154、I/O リフレッシュ 156、転送プログラム 158、異常処理プログラム 178 お

50

よび監視プログラム179を含む。周辺処理プログラム154は、UPG69により実施される制御演算処理とは異なる処理であって、制御演算処理の周辺で実施される処理であり、例えば表示処理、ロギング処理等を含む。なお、周辺処理は、UPG69が実施する制御演算処理と関連した処理も含み得る概念である。

【0083】

変数管理プログラム160は、制御エンジン150で利用可能な値を変数の形で、例えば主記憶装置106の所定記憶領域において管理する。より具体的には、変数管理プログラム160は、PLC100の状態などを示すシステム変数と、PLC100とネットワーク110を介して接続されるフィールド装置90が保持する値を示すデバイス変数161と、PLC100で実行される周辺処理プログラム154が保持する値を示すユーザ変数とを管理する。デバイス変数161は、フィールド装置90から入来するデータである入力データ162と、フィールド装置90に出力するべきデータである出力データ163の変数を含む。出力データ163の出力変数には、サーバ600から受信した出力データ(受信)183が設定される。入力データ162の入力変数には、出力データ163の制御量、指令値に従い制御されたフィールド装置90からの観測値が設定される。

10

【0084】

入力プログラム172は、ネットワーク110を介してフィールド装置90から入来する入力データ162を受付けて、変数管理プログラム160に渡す。また、出力プログラム174は、変数管理プログラム160から渡される出力データ163を、ネットワーク110を介してフィールド装置90へ出力する。

20

【0085】

転送プログラム158は、I/Oリフレッシュ156からの入力データ162を送信可能な形式に変換することにより入力データ(送信)182を生成し、バッファ180に格納する。また、転送プログラム158は、サーバ600から受信されて格納されたバッファ180の出力データ(受信)183を、処理可能な形式の出力データ163に変換し、変換後の出力データ163をI/Oリフレッシュ156に出力する。例えば、限定されないが、入力データ(送信)182は、フィールド装置90での時系列順の観測値を含むデータを示し、また、出力データ(受信)183は、フィールド装置90に出力すべき時系列順の制御データを示す。

【0086】

I/Oリフレッシュ156は、変数管理プログラム160と連携することによりI/Oリフレッシュ処理72を実施する。具体的には、I/Oリフレッシュ156は、サーバ600から受信した出力データ(受信)183を、転送プログラム158を介してバッファ180から読み出し、デバイス変数プログラムに161に出力する。また、I/Oリフレッシュ156は、変数管理プログラム160から、デバイス変数161のうち入力変数の値である入力データ162を受信し、受信した入力データ162を、転送プログラム158を介してバッファ180に入力データ(送信)182として格納する。

30

【0087】

変数管理プログラム160は、入力プログラム172から渡されたデータを、デバイス変数161のうち入力変数に設定するとともに、入力変数の入力データ162を、I/Oリフレッシュ156に出力する。また、変数管理プログラム160は、I/Oリフレッシュ156からの出力データ(受信)183を、デバイス変数161のうち出力データ163の出力変数に設定するとともに、出力変数の出力データ163を出力プログラム174に渡す。

40

【0088】

上述のI/Oリフレッシュ156においては、デバイス変数161のうち入力データ162の変数値を、フィールド装置90から入来するデータ(観測値)で更新するとともに、デバイス変数161のうちフィールド装置90への指令値、制御量となる変数値に、出力データ163を反映(設定)する。したがって、I/Oリフレッシュ156により、PLC100とフィールド装置90との間で、より特定的にはサーバ600とフィールド装置90との間で入力データ162(入力データ(送信)182)と出力データ163(出力デー

50

タ（受信）183）のデータ遣り取りが実現される。

【0089】

ゲートウェイプログラム194は、サーバ600と通信する。例えば、ゲートウェイプログラム194は、サーバ600にバッファ180の入力データ（送信）182からフレームを生成し、生成されたフレームをサーバ600に送信する。また、ゲートウェイプログラム194は、サーバ600からフレームを受信し、受信したフレームから制御演算処理の算出値（指令値、制御量）を抽出し、抽出値をバッファ180に出力データ（受信）183として格納する。バッファ180およびゲートウェイプログラム194は、ネットワークコントローラ105内に備えられてもよい。

【0090】

時刻同期プログラム177は、上記に述べたサーバ600との時刻同期およびフィールド装置90との時刻同期を実現するとともに、PLC100内の制御プログラム152の実行周期を、時刻同期に基づき管理する。

【0091】

スケジューラプログラム170は、PLC100のプロセスまたはタスクなどに対して、リソース割当てや実行タイミングなどを管理する。このような、プロセスまたはタスクは、制御プログラム152、変数管理プログラム160、入力プログラム172、出力プログラム174、時刻同期プログラム177、異常処理プログラム178および監視プログラム179等がPLC100により実行されることにより生成され得るプロセスまたはタスクが含まれる。スケジューラプログラム170は、例えば予め定められた制御周期に基づく周期で各プログラムの実行タイミングを調整する。

【0092】

これにより、PLC100は、制御プログラム152による周辺処理およびI/Oリフレッシュ処理を、サーバ600およびフィールド装置90との間で同期した時刻に基づく期で実施することができる。

【0093】

（d4．通信マスタ処理）

PLC100は、通信マスタ処理を実施する。通信マスタ処理では、PLC100が通信マスタとして動作し、I/Oリフレッシュ処理72において、ネットワーク110を介した、複数のフィールド装置90からの入来データの収集、およびサーバ600から受信する制御データの複数のフィールド装置90への転送を制御する。

【0094】

具体的には、PLC100は、通信マスタ処理において、ネットワーク110を介したPLC100とフィールド装置90との間のデータ通信を制御する。通信マスタ処理では、PLC100は通信フレームを制御周期10に同期した周期でサイクリック（例えば、数～10数ms）にフィールド装置90間を一巡するよう制御する。本実施の形態においては、このような通信フレームをサイクリック伝送させるためのプロトコルの一例として、EthernetCATが採用される。

【0095】

フィールド装置90は、ネットワーク110を介して例えばダイジーチェーンで順次に接続されている。通信フレームには、フィールド装置90毎にデータ領域が割り当てられている。例えば、各フィールド装置90は、通信フレームを受信すると、当該受信した通信フレーム内の自装置に割り当てられたデータ領域に現在値（観測値）を書込む、またはデータ領域からPLC100が設定した制御データ（指令値、制御量）を読み出し、その後、通信フレームを次段のフィールド装置90に送出する。

【0096】

各フィールド装置90が通信フレームに予め設定されたデータの現在値を書込むことで、ネットワーク110を一巡して通信マスタ（PLC100）に戻る通信フレームには、各フィールド装置90により観測された最新の値が含まれることになる。

【0097】

10

20

30

40

50

PLC100による通信マスタ処理は、変数管理プログラム160と、スケジューラプログラム170と、入力プログラム172と、出力プログラム174と、システムプログラム151との少なくとも一部が互いに協働して実行されることで実現され得る。

【0098】

(d5.サーバ600のソフトウェア構成)

図6は、本実施の形態にかかるサーバ600のソフトウェア構成例を示す図である。サーバ600は、処理エンジン610を備える。処理エンジン610は、システムプログラム71のOSの元で、サーバプログラム68が実行されることで実現される。

【0099】

処理エンジン610は、上記に述べたPLC100との時刻同期を実施する時刻同期プログラム67およびスケジューラ63を実行するとともに、スケジューラ63の元で実行のタイミングを管理しながらUPG69A, 69B、通信プログラム64、異常処理プログラム66、監視プログラム65、他のプログラム611およびリザーブプログラム612を実行する。スケジューラ63は、システムプログラム71の一部として実装されてもよい。

10

【0100】

通信プログラム64は、各PLC100とデータの遣り取りを実施する。具体的には、通信プログラム64は、PLC100からの入力データ(送信)182を領域61に格納し、また、UPG69の制御演算処理の算出値である出力データ(受信)183を領域62に格納する。通信プログラム64は、ネットワークコントローラ620が内蔵する通信制御プログラムと協同してデータの遣り取りを実施してもよい。

20

【0101】

時刻同期プログラム67は、上記に述べた各PLC100との時刻同期を実現するとともに、サーバ600内のプログラムの実行周期を、時刻同期に基づき管理する。これにより、本実施の形態では、サーバ600は、UPG69A, 69Bの制御演算処理を、各PLC100との間で同期した時刻に基づいた周期で実施することができる。スケジューラ63は、サーバ600において生成されるプロセスまたはタスクなどに対して、リソース割当てや実行タイミングなどを管理する。このような、プロセスまたはタスクは、異常処理プログラム66、監視プログラム65および時刻同期プログラム67等がサーバ600により実行されることにより生成され得るプロセスまたはタスクが含まれる。スケジューラ63は、例えば予め定められた制御周期に基づく周期で各プログラムの実行タイミングを調整する。

30

【0102】

< E . 基本系 >

まず、制御システム1の基本系における、PLC100とサーバ600との間のデータの遣り取りを説明する。図7は、本実施の形態にかかる制御システム1の基本系を模式的に示す図である。図7を参照して、基本系ではサーバ600には1台のPLC100(例えばPLC100A)が接続されている。

【0103】

(e1.基本系のタイミングチャート)

図8は、本実施の形態にかかる基本系のタイミングチャートの一例を模式的に示す図である。タイミングチャートの横軸に示す時間経過に従い、サーバ600とPLC100との間での通信を含む処理の開始および終了のタイミングと処理の実施時間の長さが示される。処理の実施時間の長さは、時間軸が延びる方向に延びる帯状の長さで示される。図8では、PLC100に3台のフィールド装置90(Slave1, Slave2およびSlave3)が接続されるが、接続されるフィールド装置90は3台に限定されない。

40

【0104】

また、本実施の形態では、PLC100では、スケジューラプログラム170により、I/Oリフレッシュ156、PLC100がサーバ600と通信するための転送プログラム158、および周辺処理プログラム154が、この順番に実行されるようにスケジューリン

50

グされる。これにより、P L C 1 0 0では、図 8 に示すように、同期した時刻に基づく制御周期 1 0 毎に、I O リフレッシュ処理 7 2、サーバ通信 7 3 および周辺処理 7 4 がこの順番で実施される。制御周期 1 0 は、互いに時刻同期する P L C 1 0 0 とサーバ 6 0 0 が共有する時間の一例であって、P L C 1 0 0 とサーバ 6 0 0 は、共通のトリガ 8 が与えられたとき、制御周期 1 0 の開始と終了のタイミングを検知する。

【 0 1 0 5 】

フィールド装置 9 0 では、制御周期 1 0 において、観測値である入力データ 1 6 2 を P L C 1 0 0 へ送信する処理「Snd」、P L C 1 0 0 から指令値、制御量を含む出力データ 1 6 3 を受信する処理「Rcv」、および出力データ 1 6 3 に従う対象の「制御」が、この順番で実行される。フィールド装置 9 0 における「制御」は、共通のトリガ 7 の入力にตอบสนองして開始される処理であって、この処理では、直前に受信した出力データ 1 6 3 に従い対象が制御される。本実施の形態では、全てのフィールド装置 9 0 が出力データ 1 6 3 を受信完了したタイミングでトリガ 7 が各フィールド装置 9 0 に与えられる。これにより、フィールド装置 9 0 は、互いに同期したタイミングで制御処理を開始することができる。なお、トリガ 7 とトリガ 8 は制御システム 1 が備える図示しない信号発生回路から供給することができる。

10

【 0 1 0 6 】

図 8 を参照して、各フィールド装置 9 0 が処理「Snd」を実施すると、(n) 回目(ただし、 $n = 1, 2, 3, \dots$)の制御周期 1 0 は終了し、(n + 1) 回目の制御周期 1 0 が開始される。

20

【 0 1 0 7 】

(n + 1) 回目の制御周期 1 0 が開始するとき、I O リフレッシュ処理 7 2 が実施されて、P L C 1 0 0 は直前(n) 回目の制御周期 1 0 でフィールド装置 9 0 から受信したデータを処理する。具体的には、I O リフレッシュ処理 7 2 では、デバイス変数 1 6 1 のうち入力データ 1 6 2 の変数値が、フィールド装置 9 0 から入来するデータで更新される。これにより、デバイス変数 1 6 1 のうち入力データ 1 6 2 の変数値が収集される。

【 0 1 0 8 】

続くサーバ通信 7 3 では、直前に収集された入力データ 1 6 2 が、入力データ(送信) 1 8 2 に変換されて、ゲートウェイプログラム 1 9 4 を介してサーバ 6 0 0 に送信される。また、このとき、フィールド装置 9 0 は、P L C 1 0 0 の出力プログラム 1 7 4 によって送信された出力データ 1 6 3 を受信する処理「Rcv」を実施し、その後、トリガ 7 にตอบสนองして「制御」を開始する。フィールド装置 9 0 では、「制御」を終了すると、処理「Snd」を実施する。

30

【 0 1 0 9 】

また、(n + 1) 回目の制御周期 1 0 では、P L C 1 0 0 とサーバ 6 0 0 との間で互いに同期した処理が実施される。サーバ 6 0 0 は、P L C 1 0 0 からの入力データ(送信) 1 8 2 を受信し、P L C 1 0 0 と同期した時刻に基づくタイミングにおいて、U P G プログラム 6 9 を実行する。サーバ 6 0 0 は、U P G 6 9 による制御演算処理の算出値(すなわち出力データ(受信) 1 8 3 に対応)を P L C 1 0 0 へ送信する。P L C 1 0 0 は、サーバ 6 0 0 からの算出値を受信し、バッファ 1 8 0 の出力データ(受信) 1 8 3 として格納する。その後、P L C 1 0 0 では、周辺処理 7 4 が実施される。その後、(n + 2) 回目の制御周期 1 0 では、(n + 1) 回目の制御周期 1 0 と同様に処理が実施される。

40

【 0 1 1 0 】

また、制御周期 1 0 において、P L C 1 0 0 では、入力データ(送信) 1 8 2 をサーバ 6 0 0 に送信後、サーバ 6 0 0 から出力データ(受信) 1 8 3 を受信するまでの間は、周辺処理 7 4 を実施してもよい。

【 0 1 1 1 】

基本系においては、サーバ 6 0 0 と P L C 1 0 0 は互いに時刻同期されて、制御周期 1 0 内において P L C 1 0 0 による I O リフレッシュ処理 7 2 とサーバ 6 0 0 により U P G 6 9 の実行(制御演算処理の実行)が実施(開始)される。

50

【 0 1 1 2 】

(e 2 . 基本系のフローチャート)

図 9 は、本実施の形態にかかる基本系における処理のフローチャートの一例を示す図である。図 9 では、制御周期 1 0 において P L C 1 0 0 で実施される処理とサーバ 6 0 0 で実施される処理とが、関連付けて示されている。図 9 では、制御周期 1 0 の開始を指示するにトリガ 8 が P L C 1 0 0 およびサーバ 6 0 0 に供給される。したがって、P L C 1 0 0 およびサーバ 6 0 0 は、共通のトリガ 8 を受信したとき、当該制御周期 1 0 で実施すべき処理を開始する。

【 0 1 1 3 】

まず、P L C 1 0 0 は、トリガ 8 に応じて、I O リフレッシュ処理 7 2 を実施する (ステップ S 3)。P L C 1 0 0 は、I O リフレッシュ処理 7 2 により収集した入力データ (送信) 1 8 2 をサーバ 6 0 0 へ送信するサーバ通信 7 3 を実施する (ステップ S 5)。

10

【 0 1 1 4 】

P L C 1 0 0 は、その後、周辺処理 7 4 を実施する (ステップ S 7)。P L C 1 0 0 は、サーバ通信 7 3 によりサーバ 6 0 0 から出力データ (受信) 1 8 3 を受信するか否か (受信完了するか否か) を判断する (ステップ S 9)。未受信と判断する、または受信完了していないと判断すると (ステップ S 9 で N O)、処理はステップ S 7 に戻り、周辺処理 7 4 を実施する。

【 0 1 1 5 】

一方、P L C 1 0 0 が出力データ (受信) 1 8 3 は受信完了したと判断すると (ステップ S 9 で Y E S)、次の制御周期 1 0 のための準備を実施する (ステップ S 1 1)。具体的には、フィールド装置 9 0 を制御するために、受信された出力データ (受信) 1 8 3 が、デバイス変数 1 6 1 の出力変数に設定 (コピー) される。これにより、次の制御周期 1 0 において、フィールド装置 9 0 に、サーバ 6 0 0 による制御演算の算出値 (指令値、制御量) を出力することができる。その後、次の制御周期 1 0 の処理に移行する。

20

【 0 1 1 6 】

また、サーバ 6 0 0 は、トリガ 8 を受けると、通信プログラム 6 4 により P L C 1 0 0 から入力データ (送信) 1 8 2 の受信処理を実施する。サーバ 6 0 0 は、入力データ (送信) 1 8 2 の受信が完了したか否かを判断する (ステップ R 3)。この受信完了の判定は、例えば、トリガ 8 を受け付けてから所定時間が経過したことに基き実施される。入力データ (送信) 1 8 2 の受信完了が判定されないときは (ステップ R 3 で N O)、ステップ R 3 が繰り返し実施される。

30

【 0 1 1 7 】

一方、サーバ 6 0 0 は、所定時間が経過した、すなわち入力データ (送信) 1 8 2 の受信が完了したと判定すると (ステップ R 3 で Y E S)、サーバ 6 0 0 は U P G 6 9 を実行する (ステップ R 5)。これにより、受信した入力データ (送信) 1 8 2 を用いた制御演算処理が実施される。サーバ 6 0 0 は、制御演算の算出値を出力データ (受信) 1 8 3 に反映 (設定) し、出力データ (受信) 1 8 3 を、通信プログラム 6 4 により、P L C 1 0 0 へ送信する (ステップ R 7)。その後、次の制御周期 1 0 の処理に移行する。

【 0 1 1 8 】

図 9 では、互いに時刻同期された P L C 1 0 0 とサーバ 6 0 0 との間において、処理は、制御周期 1 0 の開始時に供給されるトリガ 8 に応じて開始されることで、両者の処理を、時刻同期に基づく共有された時間 (制御周期 1 0 内の時間) において開始することが可能となる。

40

【 0 1 1 9 】

(e 3 . 基本系のフローチャートの変形例)

基本系の図 9 の処理は、図 1 0 に示すように異常に対処する異常処理を追加するように変形することもできる。図 1 0 は、本実施の形態にかかる基本系における処理のフローチャートの変形例を示す図である。図 1 0 の処理は、図 9 の処理に異常処理 (ステップ S 1 3、S 1 5、R 9、R 1 1) を追加して構成される。図 1 0 の他の処理は、図 9 と同様であ

50

るので説明は繰返さない。

【 0 1 2 0 】

図 1 0 を参照して、 P L C 1 0 0 は、サーバ 6 0 0 から出力データ（受信） 1 8 3 を受信完了したと判定したとき（ステップ S 9 で Y E S ）、監視プログラム 1 7 9 により異常の有無を判定させる（ステップ S 1 3 ）。監視プログラム 1 7 9 により、ステップ S 9 における受信完了が、トリガ 8 の入力から一定時間内であったか否かに基き、異常の有無が判定される。監視プログラム 1 7 9 により、この一定時間内に受信完了がなされたらと判定されたとき（ステップ S 1 3 で N O : 異常無し）、ステップ S 1 1 の処理に移行する。また、監視プログラム 1 7 9 により、この一定時間内に受信完了がなされていないと判定されたとき（ステップ S 1 3 で Y E S : 異常有り）、 P L C 1 0 0 は、異常処理プログラム 1 7 8 により、異常処理を実施させる（ステップ S 1 5 ）。その後、この制御周期 1 0 の処理は終了する。

10

【 0 1 2 1 】

なお、監視プログラム 1 7 9 は、サーバ 6 0 0 から送信された異常通知の受信の有無判定も含み得る。また、異常処理（ステップ S 1 5 ）では、異常処理プログラム 1 7 8 により、例えば異常検知のログデータを二次記憶装置 1 0 8 に格納する処理、またはサポート装置 5 0 0 に異常通知を出力する処理が含まれ得る。また、異常処理（ステップ S 1 5 ）に続いて、ステップ S 1 1 の処理が上述と同様に実施されてもよい。

【 0 1 2 2 】

また、異常有りが検知された場合（ステップ S 1 3 で Y E S ）は、サーバ 6 0 0 からの出力データ（受信） 1 8 3 を出力データ 1 6 3 に設定する処理（ステップ S 1 1 の処理）はスキップされる。この場合、次の制御周期 1 0 では、直前の制御周期 1 0 の出力データ 1 6 3 を再度用いてフィールド装置 9 0 が制御される。

20

【 0 1 2 3 】

また、サーバ 6 0 0 では、 P L C 1 0 0 から入力データ（送信） 1 8 2 を受信完了したと判定したとき（ステップ R 3 で Y E S ）、監視プログラム 6 5 により異常の有無を判定させる（ステップ R 1 1 ）。監視プログラム 6 5 により、ステップ R 3 における受信完了が、トリガ 8 の入力から一定時間内であったか否かに基き、異常の有無が判定される。監視プログラム 6 5 により、この一定時間内に受信完了がなされたらと判定されたとき（ステップ R 1 1 で N O : 異常無し）、ステップ R 5 の処理に移行する。また、監視プログラム 1 7 9 により、この一定時間内に受信完了がなされていないと判定されたとき（ステップ R 1 1 で Y E S : 異常有り）、サーバ 6 0 0 は、異常処理プログラム 6 6 により、異常処理を実施させる（ステップ R 9 ）。その後、この制御周期 1 0 の処理は終了する。

30

【 0 1 2 4 】

ステップ R 9 の異常処理は、異常検知の通知を、 P L C 1 0 0 へ送信する処理を含む。また、図 1 0 では、異常処理（ステップ R 9 ）を実施後は、この制御周期の処理を終了したが、ステップ R 7 において U P G 6 9 による制御演算処理を実施してもよい。この場合は、先の制御周期 1 0 で受信した入力データ（送信） 1 8 2 に基づく制御演算処理が実施されて、その算出値である出力データ（受信） 1 8 3 が P L C 1 0 0 に送信される（ステップ R 7 ）。

40

【 0 1 2 5 】

（ e 4 . 基本系のフローチャートのさらなる変形例 ）

図 9 または図 1 0 の処理においては、宛先で確実にデータを受信できるように、入力データ（送信） 1 8 2 または出力データ（受信） 1 8 3 は、複数回送信されてもよい。

【 0 1 2 6 】

具体的には、 P L C 1 0 0 の処理（ステップ S 5 ）では、入力データ（送信） 1 8 2 を格納した同一のフレームが複数回送信される。同一の入力データ（送信） 1 8 2 を格納したフレームには、同一の識別子（ I D ）を割当てられる。サーバ 6 0 0 は、受信するフレームの I D に基き、 P L C 1 0 0 から同一の入力データ（送信） 1 8 2 が送信されていることを検知することができる。

50

【 0 1 2 7 】

また、サーバ 6 0 0 の処理（ステップ R 7）では、出力データ（受信）1 8 3 を格納した同一のフレームが複数回送信される。同一の出力データ（受信）1 8 3 を格納したフレームには、同一の識別子（ID）を割当てられる。PLC 1 0 0 は、受信するフレームの ID に基づき、サーバ 6 0 0 から同一の出力データ（受信）1 8 3 が送信されていることを検知することができる。なお、ステップ R 7 では、サーバ 6 0 0 は、例えば、同一の出力データ（受信）1 8 3 を格納したフレームの送信回数を、出力データ（受信）1 8 3 を送信開始タイミングから次の制御周期 1 0 の開始タイミングまでの時間の長さに基づき決定することができる。

【 0 1 2 8 】

このように、入力データ（送信）1 8 2 および出力データ（受信）1 8 3 は複数回送信されることで、ネットワーク 1 1 上でデータ落ち（喪失）があっても、受信側においてデータを回復させることができる。また、ネットワーク 1 1 上のジッタにより受信側に到着するデータの順番が乱れたとしても、受信側は、データの ID に基づきデータを本来の順番で受信することができる。

【 0 1 2 9 】

（ F . 多重化系 ）

制御システム 1 は、多重化系で構成されてもよい。多重化系は、サーバ 6 0 0 と、サーバ 6 0 0 にネットワーク 1 1 を介して接続された複数の PLC 1 0 0 を備えるよう構成される。図 1 1 は、本実施の形態にかかる制御システム 1 の多重化系を模式的に示す図である。図 1 1 を参照して、多重化系ではサーバ 6 0 0 には 2 台の PLC 1 0 0（例えば PLC 1 0 0 A と PLC 1 0 0 B）が接続されている。なお、サーバ 6 0 0 に接続される PLC 1 0 0 の台数は、2 台に限定されず、3 台以上であってもよい。

【 0 1 3 0 】

（ f 1 . 多重化系のタイミングチャート ）

図 1 2 は、本実施の形態にかかる多重化系のタイミングチャートの一例を模式的に示す図である。なお、図 1 2 では、フィールド装置 9 0 の処理は略されている。図 1 2 では、PLC 1 0 0 A からの入力データ（送信）1 8 2 に基づく制御演算処理のための「UPG 1」と、PLC 1 0 0 B からの入力データ（送信）1 8 2 に基づく制御演算処理のための「UPG 2」が実行される。「UPG 1」と「UPG 2」は、共通のトリガ 8 に応じて同時に実行開始可能とされる。

【 0 1 3 1 】

図 1 2 を参照して、制御周期 1 0 の開始に同期して PLC 1 0 0 A , 1 0 0 B は I O リフレッシュ処理 7 2 を開始するが、各 PLC 1 0 0 に接続されるフィールド装置 9 0 の台数、または収集するべき入力変数の数、またはネットワーク 1 1 の負荷等の差異に起因して、サーバ通信 7 3 を開始できる時間（すなわち、I O リフレッシュ処理 7 2 により入力データ（送信）1 8 2 の収集を完了する時間）に差が生じ得る。このように、サーバ通信 7 3 を開始できる時間に差が生じることに伴い、サーバ 6 0 0 では、PLC 1 0 0 A から入力データ（送信）1 8 2 を受信し領域 6 1 に格納するデータ格納時間 1 2 と、PLC 1 0 0 B から入力データ（送信）1 8 2 を受信し領域 6 1 に格納するためのデータ格納時間 1 2 とは一致せず、ずれることになる。サーバ 6 0 0 では、この「ずれ」は待ち時間 1 4 で調整される。例えば、サーバ 6 0 0 は、PLC 1 0 0 B の入力データ（送信）1 8 2 のデータ格納完了後の待ち時間 1 4 が経過したときの時間 1 5 において、サーバ 6 0 0 は「UPG 1」と「UPG 2」を同時（並列）に実行開始する。

【 0 1 3 2 】

「UPG 1」と「UPG 2」の並列実行においては、「UPG 2」は、「UPG 1」の算出値を用いて実行されてもよい。これにより、例えば、PLC 1 0 0 A に接続されたサーボ装置（フィールド装置 9 0）からの位置データ（出力データ）に基づく算出値を、「UPG 2」の制御演算（例えば、PLC 1 0 0 B に接続されたサーボ装置（フィールド装置 9 0）の指令値を決定するために用いられる。

10

20

30

40

50

【 0 1 3 3 】

並列実行された「UPG1」と「UPG2」の実行結果である制御演算の算出値（出力データ（受信）183）は領域62にデータ格納時間13で格納される。領域62の出力データ（受信）183は、PLC100A、100Bにそれぞれ送信される。PLC100A、100Bは、それぞれ、次の制御周期10が開始される前に、サーバ600から出力データ（受信）183を受信完了する。

【 0 1 3 4 】

図13は、本実施の形態にかかる多重化系のタイミングチャートの他の例を模式的に示す図である。図12では、サーバ600で実行されるUPG69を異ならせることで、例えばPLC100Aの「UPG1」による制御演算およびPLC100Bの「UPG2」による制御演算を並列に実行する環境を提供した。これに対して、図13に示すように、サーバ600で実行されるUPG69は1種類であってもよい。これにより、各PLC100からの入力データ（送信）182に基づく制御演算を、PLC100の間で共通化することができる。

10

【 0 1 3 5 】

なお、図12と図13においても、PLC100は、入力データ（送信）182をサーバ600へ送信後からサーバ600から出力データ（受信）183を受信するまでの時間は、周辺処理74を実施してもよい。

【 0 1 3 6 】

（f2．多重化系のフローチャート）

図14は、本実施の形態にかかる多重化系における処理のフローチャートの一例を示す図である。図14では、図12または図13のタイミングチャートに従い実施される処理であって、PLC100A、PLC100Bおよびサーバ600の各装置で実施される処理が相互に関連付けて示される。PLC100Aでは、ステップS3、S5、S7、S9およびS11の処理が図9のそれらと同様に実施される。また、PLC100Bでも、ステップSS3、SS5、SS7、SS9およびSS11の処理が、上記のステップS3～S11と同様に実施される。また、サーバ600ではステップRR3、RR5およびRR7の処理が実施される。多重化系では、ステップRR3では、PLC100Aおよび100Bのそれぞれから、入力データ（送信）182を受信する。また、ステップRR7では、PLC100Aおよび100Bのそれぞれに、出力データ（受信）183が送信される。なお、図14の処理においても、図10に示した監視処理および異常処理が実施されてもよい。これら処理は、図9または図10に示す処理と同様であるので、説明は繰返さない。

20

30

【 0 1 3 7 】

（f3．多重化系の応用例）

上記に述べた多重化系は、制御システム1のイベント発生時の対応に適用できる。限定されないが、例えば障害発生時の対策に適用することができる。図15は、本実施の形態にかかる多重化系の応用例を模式的に示す図である。図15の制御システム1では、サーバ600に接続される複数のPLC100のうちの1つを稼働系として動作させ、および他をバックアップのための待機系として動作させる。待機系のPLC100は、稼働系のPLC100に障害が発生した場合に当該稼働系のPLC100による制御を引き継ぐように構成される。例えば、PLC100Aが稼働系として動作し、PLC100Bは待機系として動作する。PLC100AとPLC100Bは、切替器200を介して、フィールド装置90のネットワーク110に接続されている。障害発生時、切替器200は、指令信号185に応じて、PLC100Aおよび100Bのいずれか一方をネットワーク110に接続する。

40

【 0 1 3 8 】

図15の（A）は制御システム1の正常稼働時を示し、図15の（B）は障害発生時を示す。指令信号185は、「正常稼働」または「障害発生」のいずれかを示し、切替器200は、指令信号185が「正常稼働」を示すときは、ネットワーク110にPLC100Aを接続し、「障害発生」を示すときは、ネットワーク110にPLC100Bを接続す

50

るように構成される。

【0139】

図15の(A)の正常稼働時は、待機系であるPLC100Bは、稼働していないスタンバイ状態にある。サーバ600は、稼働中のPLC100Aの異常停止を検知すると、「障害発生」を示す指令信号185を切替器200に出力するとともに、PLC100Bに稼働指令を出力する。これにより、切替器200は、フィールド装置90に接続されるPLC100を、PLC100AからPLC100Bに切替える。

【0140】

障害が発生すると、図15の(B)に示すように、PLC100Aは停止するが、待機系のPLC100Bは稼働指令に従い待機モードから稼働モードに切り替わる。PLC100Bが稼働すると、サーバ600は入力データ(送信)182と出力データ(受信)183を遣り取りするPLC100をPLC100AからPLC100Bに切替える。これにより、障害発生時においても、フィールド装置90の制御を継続することができる。

10

【0141】

なお、待機系のPLC100Bがサーバ600およびフィールド装置90との間で時刻同期をとるタイミングは、PLC100Bが稼働されるときであってよいが、時刻同期をとるタイミングは稼働する時に限定されない。例えば、PLC100Bは、待機中において、サーバ600およびフィールド装置90と時刻同期をとるとしてもよい。

【0142】

図15では、サーバ600においてUPG69を実行する環境が実現されているので、待機系のPLC100Bが稼働されるときであっても、PLC100BにおいてUPG69の実行環境の準備(UPG69のインストール(コピー)など)等を行なう必要はない。したがって、待機系のPLC100Bを稼働状態へ速やかに移行させることができる。

20

【0143】

(f4. 多重化系の他の応用例)

図16は、本実施の形態にかかる多重化系の他の応用例を模式的に示す図である。上記に述べた多重化系を制御システム1の障害対策に応用する他のケースを、図16を参照して説明する。図15の構成では、PLC100Aを稼働系、およびPLC100BをPLC100Aに専用の待機系と設定したが、図16に示すように、待機系となるPLC100は、稼働系である複数のPLC100間で共有することができる。

30

【0144】

図16では、サーバ600に、PLC100A、PLC100BおよびPLC100Cが接続される。これら3台のPLC100のうち、PLC100Aと100Bは稼働系として動作し、PLC100CはPLC100Aと100Bで共有される待機系として動作する。このように、図16では、1の待機系のPLC100を、複数台の稼働系のPLC100で共有することができる。図16では、サーバ600はPLC100Aの制御演算処理のための「UPG1」とPLC100Bの制御演算処理のための「UPG2」とを実行する。

【0145】

例えば、PLC100Aに障害が発生した場合は、待機系のPLC100Cは稼働を開始し、切替器200は、指令信号185に応じて、工程3Aのネットワーク110に接続されるPLC100を、PLC100AからPLC100Cに切替える。また、サーバ600は、「UPG1」の制御演算のための入力データ(送信)182と出力データ(受信)183を遣り取りするPLC100を、PLC100AからPLC100Cに切替える。これにより、PLC100Aに障害が発生したとしても、PLC100Cにより工程3Aのフィールド装置90の制御を継続することができる。

40

【0146】

同様に、PLC100Bに障害が発生した場合は、待機系のPLC100Cは稼働を開始し、切替器200は、指令信号185に応じて、工程3Bのネットワーク110に接続されるPLC100を、PLC100BからPLC100Cに切替える。また、サーバ60

50

0は、「UPG2」の制御演算のための入力データ(送信)182と出力データ(受信)183を遣り取りするPLC100を、PLC100BからPLC100Cに切替える。これにより、PLC100Bに障害が発生したとしても、PLC100Cにより工程3Bのフィールド装置90の制御を継続することができる。

【0147】

なお、図15と図16では、稼働系のPLC100に障害が発生したときに、待機系のPLC100を稼働させたが、待機系のPLC100を稼働させる所定のタイミングは、障害発生時に限定されない。

【0148】

(G. オーバサンプリング)

図17は、本実施の形態にかかるオーバサンプリングの一例を模式的に説明する図である。図18は、図17のオーバサンプリングにより格納されるデータの一例を模式的に示す図である。上述した実施の形態では、制御周期10毎に、PLC100とサーバ600との間で入力データ(送信)182と出力データ(受信)183を遣り取りするためのサーバ通信73が実施された。これにより、サーバ600は、制御周期10毎に入力データ(送信)182を受信し、且つ出力データ(受信)183を送信する。

【0149】

本実施の形態では、サーバ通信73で遣り取りする入力データ(送信)182および出力データ(受信)183は1回の制御周期10のデータに限定されず、図17に示すようにN(ただし $N > 1$)回分の制御周期10のデータであってもよい。この場合は、I/Oリフレッシュ処理72によりN回分の制御周期10の入力データ(送信)182が収集された後に、収集データがサーバ600へ送信される。以下では、PLC100が、サーバ600に1回の通信で送信すべきデータを、制御周期10よりも長い周期で収集することを「オーバサンプリング」と称する場合がある。

【0150】

(g1. オーバサンプリングの一例)

図17を参照して、限定されないが、例えば $N = 3$ である場合の「オーバサンプリング」を説明する。図18では、バッファ180には、3回分の制御周期10収集した入力データ(送信)182を格納するための領域と、サーバ600から受信する3回分の制御周期10の出力データ(受信)183を格納するための領域とが模式的に示される。

【0151】

PLC100は、 n 番目、 $n+1$ 番目および $n+2$ 番目の各制御周期10においてI/Oリフレッシュ処理72により収集した3回分の制御周期10の入力データ(送信)182をバッファ180に格納する。PLC100は、($n+2$)番目の制御周期10のサーバ通信73により、バッファ180の3回分の制御周期10の入力データ(送信)182を、サーバ600に送信する(ステップS81)。続く、 $n+3$ 番目、 $n+4$ 番目および $n+5$ 番目の各制御周期10においてもI/Oリフレッシュ処理72により収集した3回分の制御周期10の入力データ(送信)182がバッファ180に格納されて、PLC100は、($n+5$)番目の制御周期10のサーバ通信73により、バッファ180の3回分の制御周期10の入力データ(送信)182を、サーバ600に送信する(ステップS81)。このように、PLC100は、3回分の制御周期10毎に、当該3回分の制御周期10において収集した入力データ(送信)182をサーバ600へ送信する。

【0152】

一方、サーバ600は、3回分の制御周期10毎に、PLC100から当該3回分の制御周期10の入力データ(送信)182を受信し、受信した各回の制御周期10の入力データ(送信)182に基づく制御演算を実施する(ステップS82)。これにより、サーバ600は、 $n+2$ 番目、 $n+3$ 番目および $n+4$ 番目の制御周期10の各回において、制御演算を実行することで、3回分の制御周期10に対応の出力データ(受信)183を算出し、PLC100に送信する。

【0153】

10

20

30

40

50

PLC100は、 $n+4$ 番目の制御周期10において、サーバ600から、 n 番目、 $n+1$ 番目および $n+2$ 番目の3回分の制御周期10の入力データ(送信)182に基づく出力データ(受信)183を受信する。受信された出力データ(受信)183は、バッファ180に格納される。PLC100は、バッファ180の3回分の制御周期10の出力データ(受信)183を、それぞれ、 $n+5$ 番目、 $n+6$ 番目および $n+7$ 番目の制御周期10の各回でフィールド装置90に出力されるように、I/Oリフレッシュ処理72を実施する(ステップS83)。

【0154】

このように、図17によれば、PLC100は、 n 番目(例えば、 $n+2$ 番目)の制御周期10において、その前の N 回分の制御周期(すなわち、 n 番目、 $n+1$ 番目および $n+2$ 番目の3回分の制御周期)10の入力データ(送信)182を収集して、サーバ600に送信すると、サーバ600から、その後の N 回分の制御周期(すなわち、 $n+5$ 番目、 $n+6$ 番目および $n+7$ 番目の3回分の制御周期)10の出力データ(受信)183を取得することができる。

10

【0155】

図17のオーバサンプリングによれば、限定されないが、例えばPLC100とフィールド装置90間のネットワーク110の通信時間が長い、またはPLC100とサーバ600間のネットワーク11の通信時間が長い場合に、利点をもたらす。つまり、このような通信時間が長い場合、待ち時間14が長くなりUPG69の制御演算を制御周期10内に完了できない事態(異常)が頻繁となる。これに対して、オーバサンプリングが実施される場合は、各制御周期10でサーバ通信73を実施する必要はなくなり、各制御周期10で待ち時間14が生じるのを回避することができる。

20

【0156】

例えば、サーバ600(またはPLC100)はネットワーク11に通信時間(または通信速度)を計測し、またPLC100はネットワーク110の通信時間を計測し、計測した通信時間が閾値を超える場合は、サーバ600およびPLC100は、オーバサンプリングを実施するモードが設定される。

【0157】

(g2. オーバサンプリングの他の例)

図19は、本実施の形態にかかるオーバサンプリングの他の例を模式的に説明する図である。図19では、PLC100は、各制御周期10のサーバ通信73において、前の N 回分の制御周期10で収集した入力データ(送信)182をサーバ600に送信する(ステップS84、S85)。したがって、サーバ600は、各制御周期10において、前の N 回分の制御周期10の入力データ(送信)182を受信する。サーバ600は、各制御周期10において、PLC100から受信した N 回分の制御周期10の入力データ(送信)182に基き、制御演算を実施することにより、将来の N 回分の制御周期10の出力データ(受信)183を算出し、PLC100に送信する。例えば、サーバ600はマルチプロセスにより、各制御周期10で受信する入力データ(送信)182を用いたUPG69を並列に実行することにより、制御周期10毎に、将来の N 回分の制御周期10の出力データ(受信)183を生成する。

30

40

【0158】

図19のオーバサンプリングによれば、PLC100は、同一の制御周期10の入力データ(送信)182をサーバ600に繰返し送信し、また、サーバ600も、同一の制御周期10のための出力データ(受信)183をPLC100に繰返し送信する。したがって、例えばある制御周期10において、ネットワーク11またはネットワーク110を介したデータ通信にエラー(例えば、通信におけるデータ喪失等)が発生したとしても、上記の同一データ(入力データ(送信)182または出力データ(受信)183)の繰返し送信により、当該エラーに対処(エラーをカバー、例えばデータ補完)することができる。

【0159】

< H . 制御周期の出力データの予測 >

50

上記のオーバサンプリングとは異なり、1の制御周期10の入力データ(送信)182に基づき将来の複数の制御周期10の出力データ(受信)183が予測(算出)されてもよい。

【0160】

このケースでは、PLC100は、サーバ通信73において、1の制御周期10で収集した入力データ(送信)182をサーバ600に送信し、サーバ600は、PLC100から受信した当該1の制御周期10の入力データ(送信)182に基づき、制御演算を実施することにより、将来のN回分の制御周期10の出力データ(受信)183を算出(予測)し、PLC100に送信する。このケースでも、上述のマルチプロセッシングによるUPG69の並列により、制御周期10毎に将来のN回分の制御周期10の出力データ(受信)183を生成することができる。PLC100は、制御周期10毎に、将来のN回分の制御周期10の出力データ(受信)183を取得することができる。これにより、例えば、ある制御周期10においてデータ通信エラーが発生しても、前の制御周期10で受信済みの予測された出力データ(受信)183に基づく出力データ163の制御量、指令値をフィールド装置90に出力することができる。したがって、このケースであっても、当該通信エラーに対処することができる。

10

【0161】

< I . 準備時間の確保 >

本実施の形態では、サーバ600は、UPG69の制御演算処理とは異なる他の処理を実施するための他のプログラム611を実行することができる。他の処理は、限定されないが、例えばログデータの管理等を含む。サーバ600では、制御周期10内において、制御演算処理を実施する時間外において他のプログラム611を実行することにより他の処理を実施する。

20

【0162】

本実施の形態では、サーバ600は、PLC100でI/Oリフレッシュ処理72が実施される時間において、他の処理を実施する。また、サーバ600は、I/Oリフレッシュ処理72が終了するとき、PLC100からの入力データ(送信)182を用いてUPG69を実行するためのリソースを確保する準備処理を実施する。ここでは、この準備処理に必要な時間を「準備時間」と称する。なお、準備時間は、予め計測されて設定(格納)されていると想定する。

30

【0163】

図20は、本実施の形態にかかる基本系における処理のフローチャートの変形例を示す図である。図20のフローチャートは、図9のフローチャートの処理に、ステップR1、R1a、R2、R4およびR4aの処理を追加したものである。ここでは、追加された処理を説明し、他の処理は図9に示した処理と同様であるので詳細は繰返さない。

【0164】

図20を参照して、PLC100では、図9と同様の処理が実施される。サーバ600では、制御周期10の開始時に受付けるトリガ8に応答して、リザーブプログラム612により、「準備時間9」を確保できるか否かを判断する(ステップR1)。具体的には、リザーブプログラム612により、トリガ8を受付けた時間からI/Oリフレッシュ処理72が終了するまでの所要時間のうちの残り時間が計測されて、計測された残り時間と上記の準備時間とが比較され、その比較結果に基づき(残り時間>準備時間)の条件が満たされるか否かが判断される。本実施の形態では、I/Oリフレッシュ処理72の所要時間は予め設定(格納)されている。

40

【0165】

リザーブプログラム612により、上記の条件が満たされると判断される(ステップR1でNO)間は、すなわちI/Oリフレッシュ処理72が完了するまで十分に余裕があるときは、他のプログラム611は実行可能である(ステップR1a)。一方、リザーブプログラム612により、上記の条件は満たされないと判断されると(ステップR1でNO)、すなわちI/Oリフレッシュ処理72が終了に近づいていると、サーバ600は準備処理を

50

実施する（ステップR2）。したがって、サーバ600は、この準備処理を、準備時間9において実施することができる。

【0166】

このように、サーバ600では、リザーブプログラム612は、トリガ8の受付け時間とI/Oリフレッシュ処理72にかかる時間とに基づき入力データ（送信）182の受信開始時間（すなわちI/Oリフレッシュ処理72の終了時間）を算出（予測）し、この予測される受信開始時間から遡る一定時間内（準備時間9）において、制御演算処理のための準備処理を実行する。

【0167】

準備処理では、具体的には、処理エンジン610は、他の処理の実施が禁止されるように他のプログラム611の実行を停止し、通信プログラム64を起動する。これにより、確保された「準備時間」において、サーバ600のリソースを、その後にUPG69を実行するために確保しておくことが可能となる。

10

【0168】

起動された通信プログラム64により、PLC100からの入力データ（送信）182を受信し、受信が完了するまで（ステップR3でNO）、ステップR2が実施される。

【0169】

通信プログラム64により、入力データ（送信）182の受信が完了したと判断されると（ステップR3でYES）、処理エンジン610は、当該制御周期10においてUPG69を実行することが可能な時間が残っているか否かを判断する（ステップR4）。処理エンジン610は、この実行可能残り時間を、 $(A - (B - C) - D - E)$ の演算式で算出する。なお、変数Aは制御周期10の時間、変数Bはサーバ600が入力データ（送信）182を受信完了した時間、変数Cは制御周期10の開始時間、変数Dはサーバ600からPLC100への出力データ（受信）183の通信時間、および変数EはPLC100の周辺処理74の所用時間を、それぞれ示す。なお、変数Dの通信時間は、予め実験等で算出した値を示す。

20

【0170】

処理エンジン610は、 $(\text{実行可能残り時間} > \text{UPGによる制御演算処理の時間})$ の条件が満たされるか否かを判断する（ステップR4）。なお、UPGによる制御演算処理の時間は、予め実験で算出した値を示す。

30

【0171】

処理エンジン610は、上記の条件が満たされると判断すると（ステップR4でYES）UPG69を実行する（ステップR5）が、条件は満たされないと判断すると（ステップR4でNO）、処理エンジン610は、サーバ600の制御周期10をM倍となるように変更する（ステップR4a）。例えば、処理エンジン610は、図17に示すように、3回分（ $M = 3$ のケース）の制御周期10の入力データ（送信）182に基づき、UPG69が1回実行されるように、サーバ600の制御周期を長くする。サーバ600は、制御周期10をM倍に変更する場合は、PLC100に制御周期をM倍にする旨を通知する。PLC100は、受信した通知に応じて、例えば、次の制御周期10から、M回分（ただし $M > 1$ ）の制御周期10の入力データ（送信）182を収集し、サーバ600に送信する。この値Mは、算出された実行可能残り時間に基き決定されてもよい。なお、図20において、ステップR4およびR4aの処理はオプションとして備えることができる。

40

【0172】

このように、サーバ600は、PLC100から入力データ（送信）182が到着する時間を予測できるので、予測到着時間から遡る一定時間である準備時間において、他の処理を実行禁止して、受信する入力データ（送信）182を用いてUPG69を実行するためのリソースを準備しておくことができる。サーバ600は、UPG69を実行するために、「準備時間」を確保するスケジューリングを実施する。したがって、サーバ600でUPG69が実行されるとしても、PLC100が、各制御周期10内で出力データ（受信）183を確実に取得できる可能性を高めることができる。

50

【 0 1 7 3 】

< J . 多重系の処理の変形例 >

図 2 1 は、本実施の形態にかかる多重化系の処理のフローチャートの変形例を示す図である。上記に述べた図 1 2 ~ 図 1 4 では、サーバ 6 0 0 は、「UPG 1」および「UPG 2」の両方の UPG 6 9 のための入力データ（送信）1 8 2 を受信完了したときに、UPG 6 9（「UPG 1」および「UPG 2」）を実行開始した。これに対して、この変形例に示すように、各 UPG 6 9 についての入力データ（送信）1 8 2 が受信完了した時点で、当該 UPG 6 9 を実行開始することもできる。例えば、「UPG 1」と「UPG 2」が、互いにデータを遣り取りしないよう構成されている場合は、サーバ 6 0 0 は、「UPG 1」と「UPG 2」を、それぞれ、独立して実行することができる。

10

【 0 1 7 4 】

図 2 1 では、PLC 1 0 0 A と PLC 1 0 0 B の処理は、図 1 3 に示された処理と同様であるので説明は繰返さない。サーバ 6 0 0 では、PLC 1 0 0 A の制御演算処理を実施するための「UPG 1」と、PLC 1 0 0 B の制御演算処理を実施するための「UPG 2」とが並列に実行される。なお、図 2 1 の処理でも、図 2 0 に示す「準備時間」を確保する処理または制御周期 1 0 を長くする処理が実施されてもよい。

【 0 1 7 5 】

まず、「UPG 1」を実行するために、処理エンジン 6 1 0 は、通信プログラム 6 4 により、PLC 1 0 0 A から入力データ（送信）1 8 2 を受信し（ステップ R 9）、処理エンジン 6 1 0 は、「UPG 1」の変数のための入力データ（送信）1 8 2 が受信完了しているかを判定する（ステップ R 1 1）。具体的には、処理エンジン 6 1 0 は、「UPG 1」から入力変数を抽出し、抽出した全ての入力変数に対応するデータを受信完了しているかを判定する。処理エンジン 6 1 0 は、入力データ（送信）1 8 2 を受信完了していないと判定すると（ステップ R 1 1 で NO）、ステップ R 9 を繰返す。

20

【 0 1 7 6 】

一方、処理エンジン 6 1 0 は、「UPG 1」のための入力データ（送信）1 8 2 を受信完了したと判定すると（ステップ R 1 1 で YES）、「UPG 1」を実行する（ステップ R 1 3）。処理エンジン 6 1 0 は、PLC 1 0 0 A に必要な出力データ（受信）1 8 3 が算出されたか否かを判定する（ステップ R 1 5）。具体的には、処理エンジン 6 1 0 は、「UPG 1」から全ての出力変数を抽出し、抽出された全ての出力変数に対応する値が「UPG 1」の実行により算出されたか否かを判定する。処理エンジン 6 1 0 は、PLC 1 0 0 A に必要な出力データ（受信）1 8 3 が算出されていないと判定すると（ステップ R 1 5 で NO）、「UPG 1」の実行が継続する（ステップ R 1 3）。

30

【 0 1 7 7 】

一方、処理エンジン 6 1 0 は、PLC 1 0 0 A に必要な出力データ（受信）1 8 3 が算出されたと判定すると（ステップ R 1 5 で YES）、算出された出力データ（受信）1 8 3 を、PLC 1 0 0 A に送信する（ステップ R 1 7）。

【 0 1 7 8 】

同様に、PLC 1 0 0 B の「UPG 2」を実行するために、処理エンジン 6 1 0 は、ステップ R 1 9、R 2 1、R 2 3、R 2 5 および R 2 7 の処理を実施する。これらの処理は、ステップ R 9、R 1 1、R 1 3、R 1 5 および R 1 7 の処理と同様であるので説明は繰返さない。

40

【 0 1 7 9 】

図 2 1 の処理によれば、「UPG 1」および「UPG 2」は、他方のプログラムのための入力データ（送信）1 8 2 の受信完了を待つことなく、自己の入力データ（送信）1 8 2 を受信完了した時点で実行開始される。この場合は、図 1 2 に示す待ち時間 1 4 をなくすることができるから、サーバ 6 0 0 は、PLC 1 0 0 の制御演算処理を制御周期 1 0 内で完了させることが可能となる。

【 0 1 8 0 】

（ j 1 . モードの設定 ）

50

本実施の形態は、制御システム1は、PLC100Aの「UPG1」とPLC100Bの「UPG2」の実行に関して、図12のように両UPGの入力データ（送信）182が受信完了したときに両方のUPGを実行開始するモード（第1モードという）と、図10のように、「UPG1」と「UPG2」は、それぞれ、対応のPLC100から入力データ（送信）182が受信完了したときには実行を開始するモード（第2モードという）を備える。サーバ600は、サポート装置500を介した外部指示（ユーザ指示）に従い第1モードおよび第2モードの一方を実施するように設定される。

【0181】

具体的には、例えば、「UPG1」はPLC100AとPLC100Bからの入力データ（送信）182を使用するのであれば「第1モード」がサーバ600に設定されて、「UPG1」はPLC100Aからの入力データ（送信）182のみを使用するのであれば「第2モード」がサーバ600に設定される。ユーザは、例えば、サポート装置500が提供するUIを介して「第1モード」または「第2モード」を設定することができる。モードの設定に関連して、サポート装置500は、「UPG1」が使用する変数名の一覧と、「UPG2」が使用する変数名の一覧とを比較する。サポート装置500は、比較の結果に基き、一覧に両者に共通する変数名が有ると判定するときは、選択肢として「第1モード」をユーザに提示し、一覧に両者に共通する変数名は無いと判定するときは選択肢として「第1モード」または「第2モード」を提示する。ユーザは、上記の外部指示をサーバ600に設定するために、サポート装置500を操作する。

【0182】

< K . 利点 >

本実施の形態では、UPG69はPLC100ではなくサーバ600において実行されるので、PLC100がUPG69を実行する場合に比較して、PLC100においてUPG69を実行するためのリソース（メモリ）を節約できる。また、サーバ600にUPG69を実行する環境が備えられるので、PLC100のプロセッサは、UPG69を実行可能なスペックを要求されない。したがって、PLC100にかかるコストを低くしながら、PLC100は自己のプロセッサのスペックを超える制御演算処理の精度（処理量、処理速度等）を得ることができる。

【0183】

また、UPG69の実行環境はサーバ600において構成されるので、例えば各PLC100に対応のUPG69の協調実行（互いにデータを遣り取りする、または互いに同期をとる等）を、PLC100同士のデータ通信を必要とすることなく、実現することができる。

【0184】

また、サーバ600とPLC100とは互いに時刻同期されて、共有される時間である、例えば制御周期内でIオリフレッシュ処理72およびUPG69の制御演算処理が実施されるように、自動的に制御周期10を長くする処理（ステップR4a）が実行される。これにより、制御周期内でIオリフレッシュ処理72およびUPG69の制御演算処理が完了することが保証され得る。

【0185】

また、制御システム1では、サーバ600が、各PLC100と入力データ（送信）182および出力データ（受信）183を遣り取りすることで、UPG69は、それぞれ、複数のPLC100に自動的に対応付けされる。

【0186】

< L . 付記 >

本開示は、以下のような技術思想を含む。

【0187】

（構成1）

1以上の制御装置（100）および情報処理装置（600）を備える制御システム（1）であって、

10

20

30

40

50

前記情報処理装置と前記 1 以上の制御装置は、通信可能に接続されて互いに時刻同期され、前記情報処理装置は、

各前記 1 以上の制御装置から受信するデータ (1 8 2) に基づく制御演算に従う制御データ (1 8 3) を算出する制御演算処理 (6 9) を実行する処理部 (6 0 3) を、含み、前記各制御装置は、

フィールド装置 (9 0) からの入来データを前記情報処理装置へ転送し、前記情報処理装置から受信する前記制御データを前記フィールド装置に出力する入出力処理 (7 2) を実行する制御部 (1 0 2) を、含み、

前記制御演算処理および前記入出力処理は、前記時刻同期に基づく時間であって前記各制御装置と前記情報処理装置との間で共有される時間 (1 0) 内で実行される、制御システム。

10

【 0 1 8 8 】

(構成 2)

前記処理部が実行する処理は、さらに、

前記共有される時間内で、前記制御演算処理を実行するための時間が不足するとき、前記共有される時間を長くする処理 (R 4 a) を含む、構成 1 に記載の制御システム。

【 0 1 8 9 】

(構成 3)

前記共有される時間は制御周期 (1 0) を含み、

前記入出力処理は、

制御周期において、当該制御周期以前の複数の制御周期に対応する前記入来データを前記情報処理装置へ転送し、前記情報処理装置から受信する前記複数の制御周期に対応する前記制御データを前記フィールド装置に出力する処理 (S 8 1、S 8 3) を含む、構成 1 または 2 に記載の制御システム。

20

【 0 1 9 0 】

(構成 4)

前記入出力処理は、

制御周期において、前記情報処理装置から前記複数の制御周期に対応する制御データを受信したとき、当該複数の制御周期に対応する制御データを、それぞれ、当該制御周期よりも後の複数の制御周期において、前記フィールド装置に出力する処理 (S 8 3) を含む、構成 3 に記載の制御システム。

30

【 0 1 9 1 】

(構成 5)

前記共有される時間は制御周期 (1 0) を含み、

前記入出力処理は、

1 の制御周期に対応する前記入来データを前記情報処理装置へ転送し、前記情報処理装置から受信する当該入来データに基づく複数の制御周期に対応する制御データを、前記フィールド装置に出力する処理を含む、構成 1 または 2 に記載の制御システム。

【 0 1 9 2 】

(構成 6)

前記情報処理装置と前記各制御装置は、前記共有される時間内で受付ける共通のトリガ (8) に応じて処理を開始し、

前記情報処理装置は、

前記トリガの受け付け時間と前記入出力処理にかかる時間とに基づき予測される前記データの受信開始時間から遡る一定時間内において、前記制御演算処理のための準備処理 (R 2) を実行する、構成 1 から 5 のいずれか 1 に記載の制御システム。

40

【 0 1 9 3 】

(構成 7)

前記準備処理は、前記制御演算処理を除く情報処理の実行を禁止する処理を含む、構成 6 に記載の制御システム。

50

【 0 1 9 4 】

(構成 8)

前記情報処理装置に複数の前記制御装置が通信可能に接続されて、
前記制御演算処理は、それぞれが各前記複数の制御装置に対応する複数の制御演算処理を含み、

前記処理部は、

前記複数の制御装置からデータを受信したとき前記複数の制御演算処理を実行開始する第 1 モード、または、前記複数の制御装置のうちの 1 つからデータを受信したとき当該 1 の制御装置に対応する前記制御演算を実行開始する第 2 モードを備える、構成 1 から 7 のいずれか 1 に記載の制御システム。

10

【 0 1 9 5 】

(構成 9)

前記処理部は、前記第 1 モードおよび前記第 2 モードを備える場合、外部指示に従い前記第 1 モードおよび前記第 2 モードの一方を実施する、構成 8 に記載の制御システム。

【 0 1 9 6 】

(構成 10)

複数の前記フィールド装置がフィールドネットワーク (1 1 0) を介して前記制御装置に接続され、

前記制御部は、

前記入出力処理において、前記フィールドネットワークを介した、前記複数のフィールド装置から入来データを収集し、および前記情報処理装置から受信する前記制御データを前記複数のフィールド装置へ転送する、構成 1 から 9 のいずれか 1 に記載の制御システム。

20

【 0 1 9 7 】

(構成 11)

前記情報処理装置に複数の前記制御装置が通信可能に接続されて、

前記複数の制御装置は稼働系として動作する制御装置と、待機系として動作する制御装置とを含み、

前記待機系の制御装置は、所定タイミングに応じて稼働を開始し、

前記情報処理装置は、前記所定タイミングに応じて、前記稼働系の制御装置とのデータ通信を、前記待機系との制御装置とのデータ通信に切替える、構成 1 から 9 のいずれか 1 に記載の制御システム。

30

【 0 1 9 8 】

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】

【 0 1 9 9 】

1 制御システム、7, 8 トリガ、9 準備時間、10 制御周期、11, 110 ネットワーク、12, 13 データ格納時間、14 待ち時間、15 時間、61 受信用データ領域、62 送信用データ領域、63 スケジューラ、64 通信プログラム、65, 179 監視プログラム、66, 178 異常処理プログラム、151 システムプログラム、67, 177 時刻同期プログラム、68 サーバプログラム、72 リフレッシュ処理、73 サーバ通信、74 周辺処理、90, 90A~90I フィールド装置、91A~91I, 101A, 101B, 101C, 102A, 102B, 102C, 601 タイマ、150 制御エンジン、152 制御プログラム、154 周辺処理プログラム、156 I/Oリフレッシュ、158 転送プログラム、160 変数管理プログラム、161 デバイス変数、162 入力データ、163 出力データ、170 スケジューラプログラム、172 入力プログラム、174 出力プログラム、180 バッファ、185 指令信号、194 ゲートウェイプログラム、200 切替器、500 サポート装置、600

40

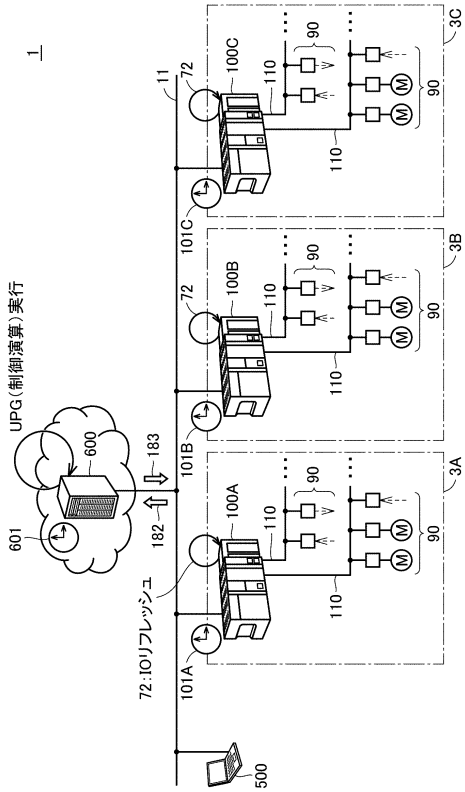
50

サーバ、610 処理エンジン、612 リザーブプログラム。

【図面】

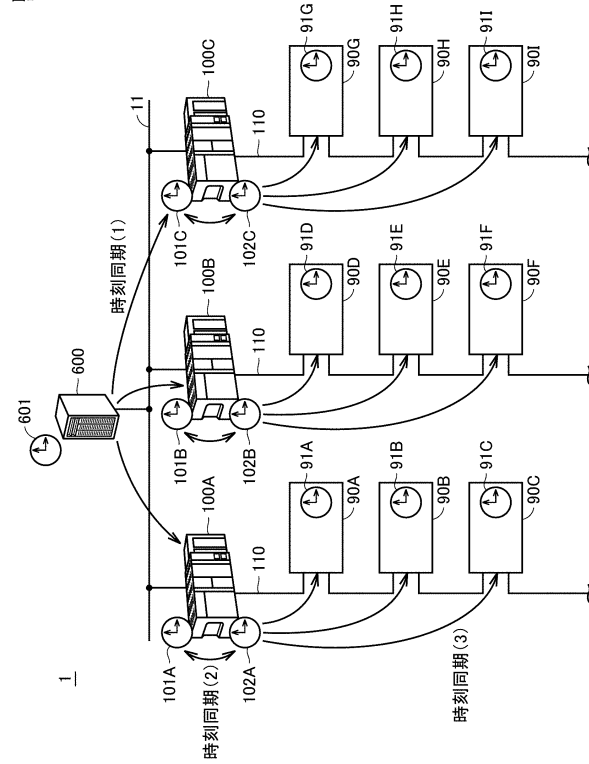
【図1】

図1



【図2】

図2

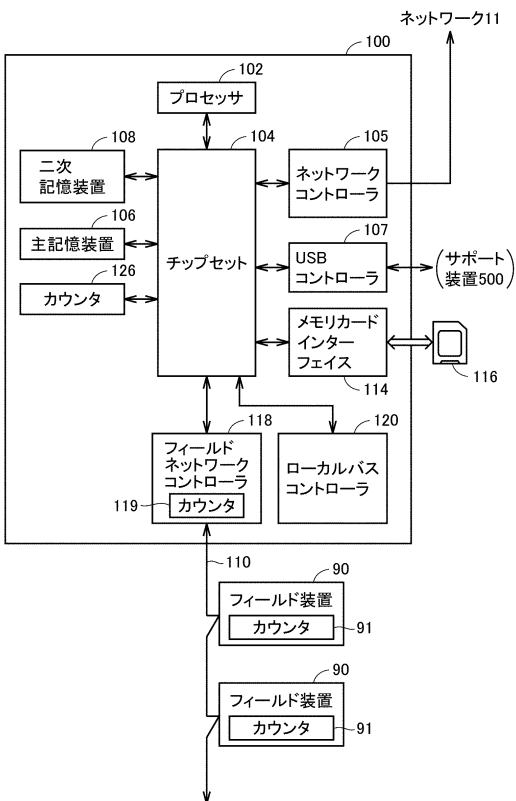


10

20

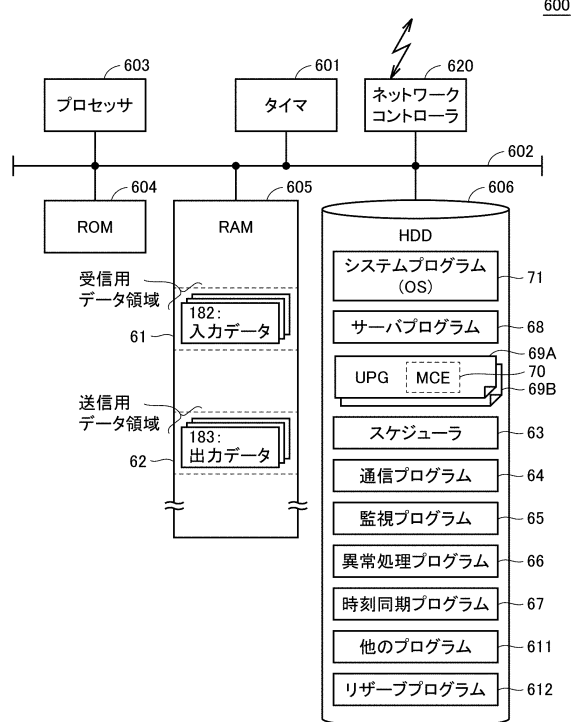
【図3】

図3



【図4】

図4

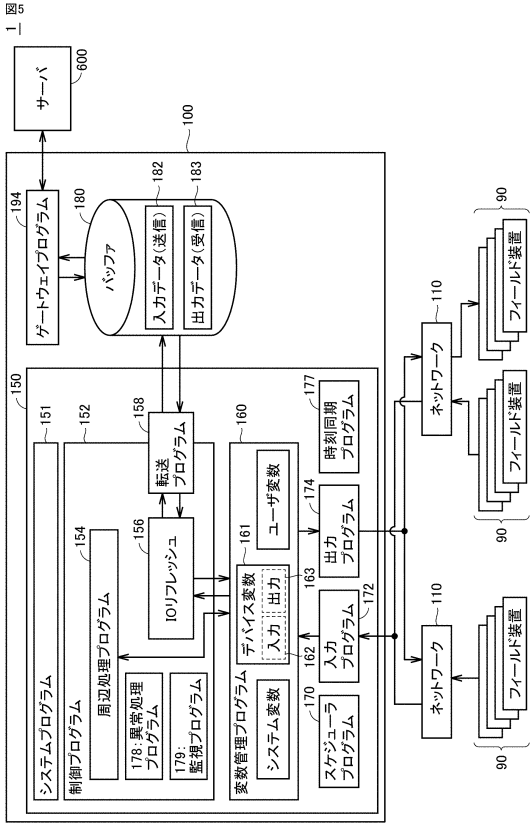


30

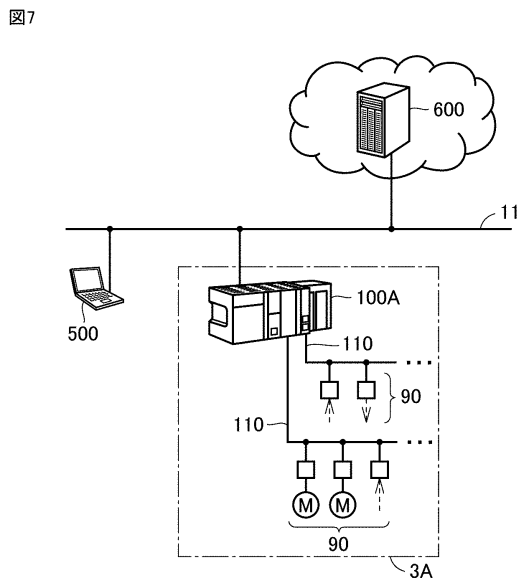
40

50

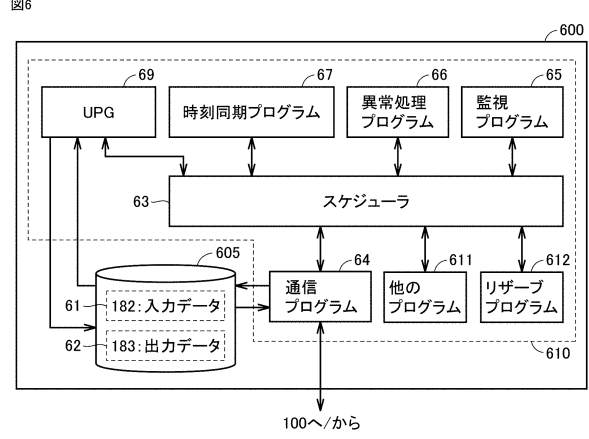
【図5】



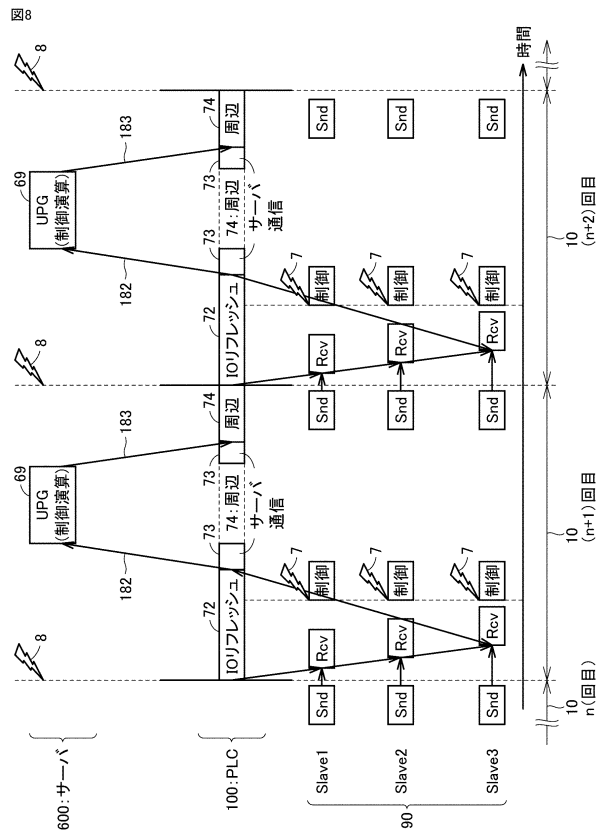
【図7】



【図6】



【図8】



10

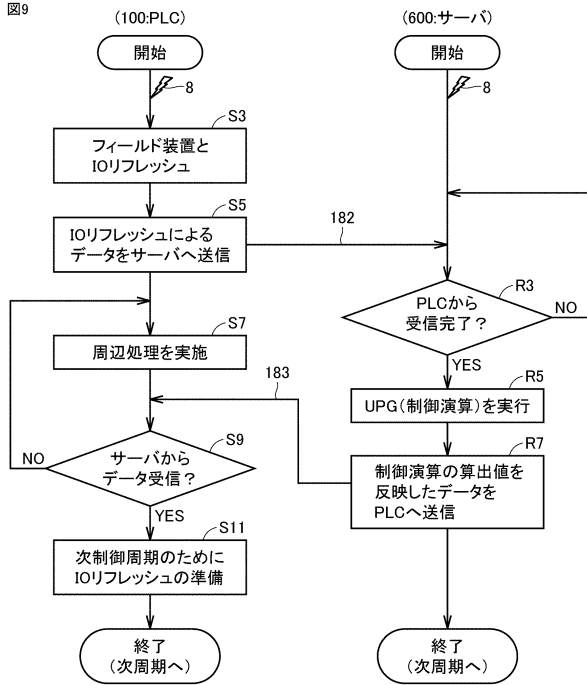
20

30

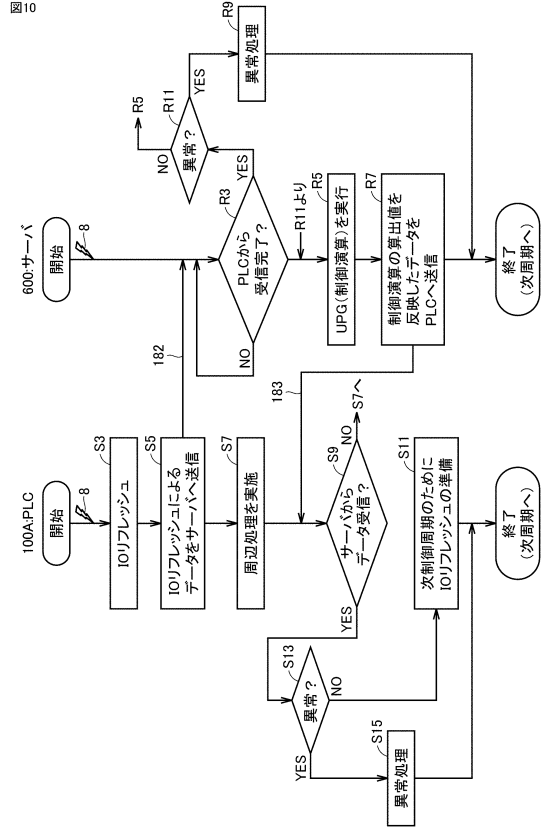
40

50

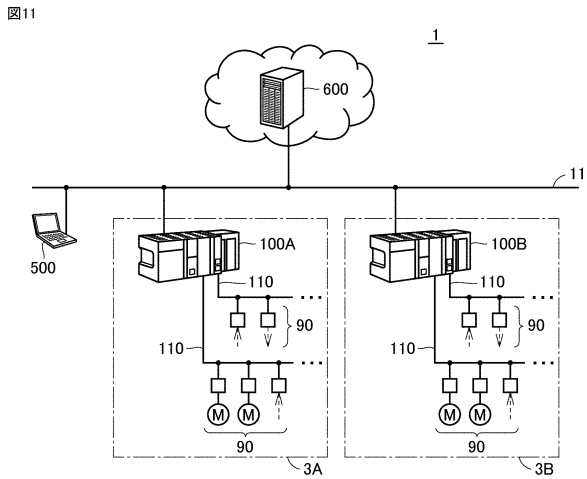
【図 9】



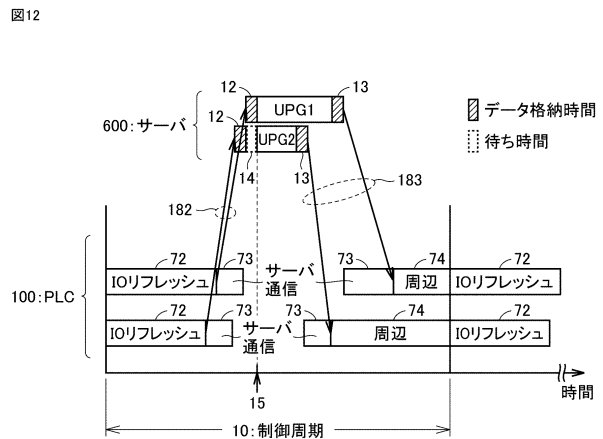
【図 10】



【図 11】



【図 12】



10

20

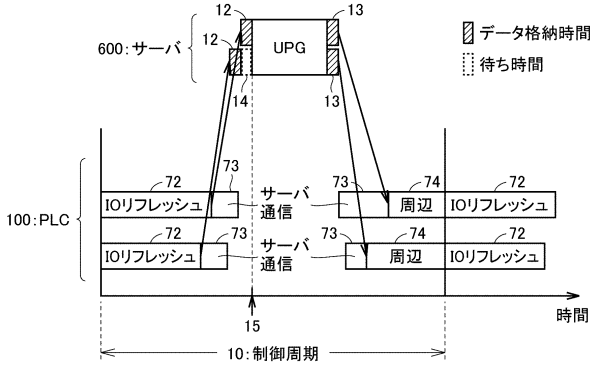
30

40

50

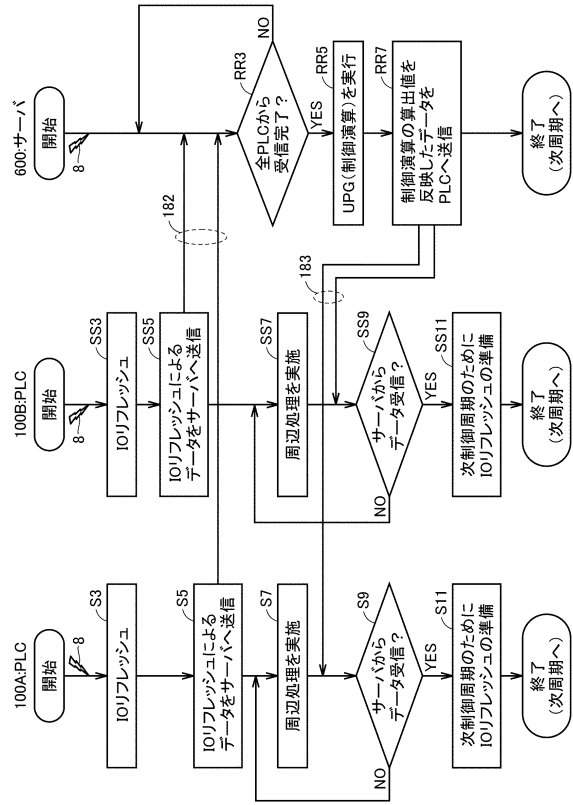
【図 13】

図13



【図 14】

図14

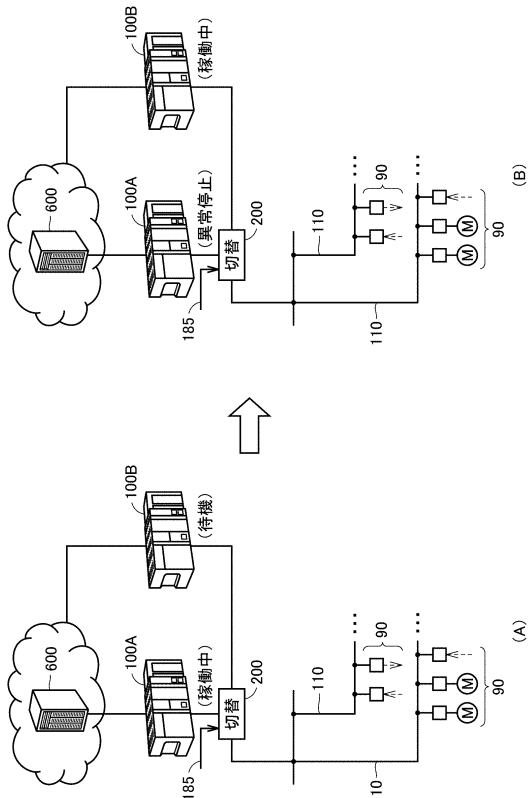


10

20

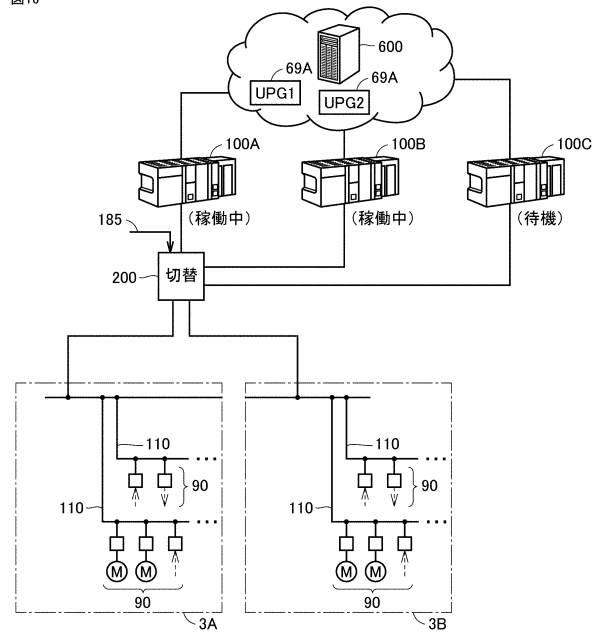
【図 15】

図15



【図 16】

図16



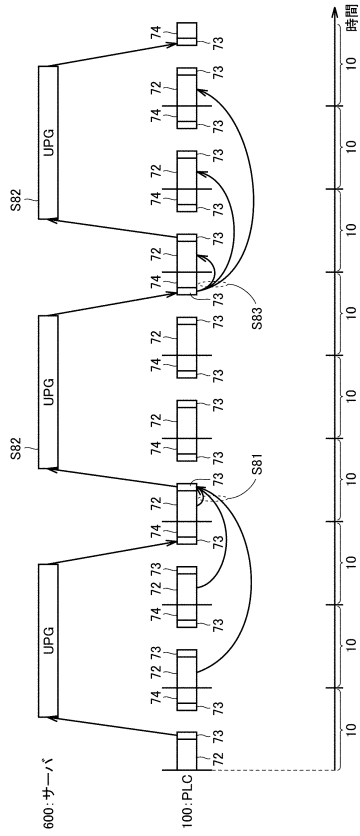
30

40

50

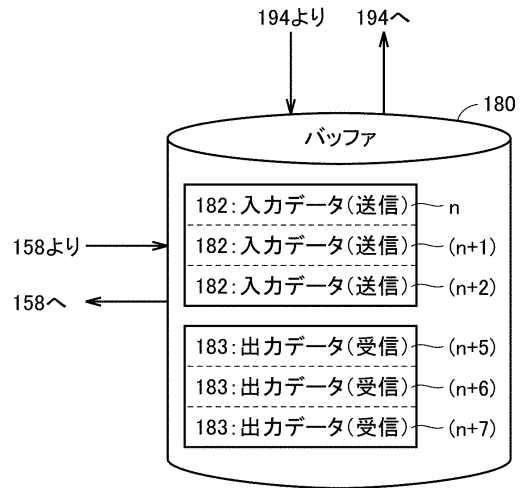
【図 17】

図17



【図 18】

図18

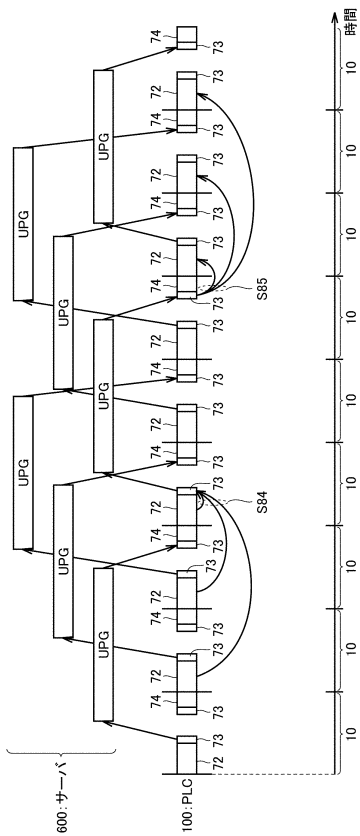


10

20

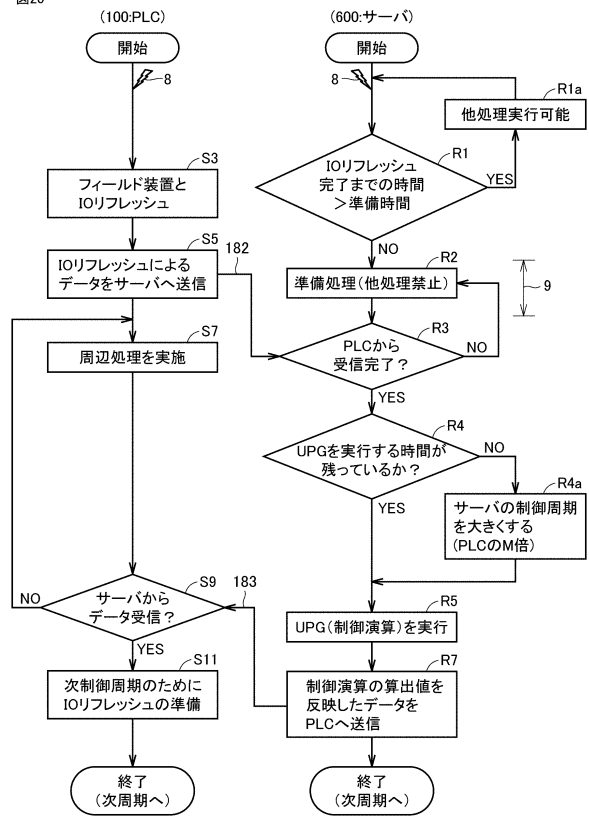
【図 19】

図19



【図 20】

図20



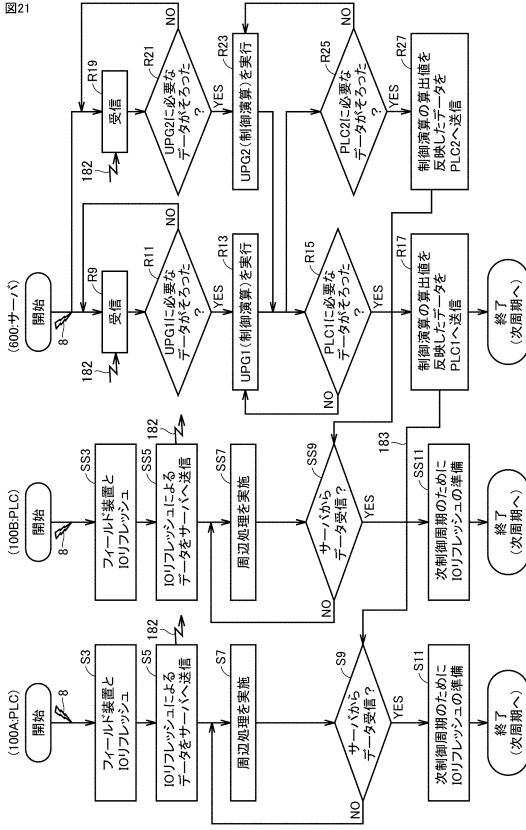
30

40

50

【 図 2 1 】

図 21



10

20

30

40

50

フロントページの続き

(72)発明者 新實 太雅
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 澤田 成憲
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

審査官 黒田 暁子

(56)参考文献 特開2004-326531(JP,A)
特開2010-182101(JP,A)
特開2018-151726(JP,A)
特開昭63-310004(JP,A)

(58)調査した分野 (Int.Cl., DB名)
G05B 19/05
G06F 15/173
G06F 11/20