

(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2004-5280

(P2004-5280A)

(43) 公開日 平成16年1月8日(2004.1.8)

(51) Int. Cl. ⁷	F I	テーマコード (参考)
G05B 19/048	G05B 19/05	5B054
G05B 9/03	G05B 9/03	5H209
G06F 1/24	G06F 1/00 350B	5H220

審査請求 未請求 請求項の数 4 O L (全 10 頁)

(21) 出願番号	特願2002-160807 (P2002-160807)	(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
(22) 出願日	平成14年5月31日 (2002.5.31)	(74) 代理人	100092598 弁理士 松井 伸一
		(72) 発明者	永野 英輔 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	原田 衛 京都府京都市下京区西洞院木津屋橋通東入 オムロンソフトウェア株式会社内
		(72) 発明者	楠田 英男 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

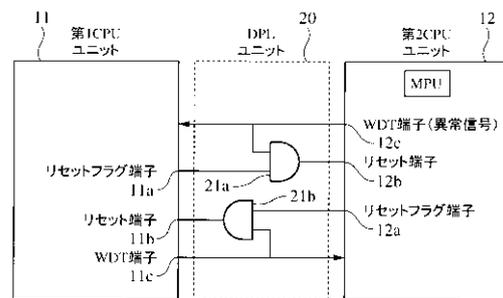
(54) 【発明の名称】 プログラマブルコントローラ及びCPUユニット

(57) 【要約】

【課題】非二重化状態になった場合でも、自動的に二重化システムに復帰することができるプログラマブルコントローラを提供すること

【解決手段】二重化された第1, 第2CPUユニット11, 12と、両CPUユニット間のバス制御や、制御系/待機系の切り替えなどを行うデュプレックスユニット20とを備えている。デュプレックスユニットは、第1, 第2アンド回路21a, 21bを備え、各アンド回路には、一方のCPUユニットのリセットフラグ端子11a, 12aと、他方のCPUユニットのWDT端子12c, 11cが入力される。一方のCPUユニットが異常となりWDTがONの時に、他方のCPUユニットのリセットフラグがONになると、アンド回路の出力もONになり、異常状態のCPUユニットのリセット端子にリセット信号が入力され、リセット処理されて正常に復帰する。

【選択図】 図2



【特許請求の範囲】**【請求項 1】**

二重化された 2 つの CPU ユニットを備え、前記 2 つの CPU ユニットは、相手の状態を確認可能とし、一方が制御系となるとともに他方が待機系となり、前記制御系の CPU ユニットがダウンした場合には、前記待機系の CPU ユニットが制御系に切り替わって運転を継続するプログラマブルコントローラにおいて、

前記 2 つの CPU ユニットは、それぞれ相手の CPU ユニットに向けてリセット信号を出力するリセット信号出力手段と、相手方の CPU ユニットから送られてきたリセット信号を受信すると、初期処理その他のリセット処理を行って起動するリセット処理手段とを備え、

10

前記 2 つの CPU ユニットのうちの一方が異常発生した際に、他方の CPU ユニットに備えた前記リセット信号出力手段がリセット信号を出力するようにしたことを特徴とするプログラマブルコントローラ。

【請求項 2】

前記リセット信号出力手段からリセット信号が出力された場合、前記相手の CPU ユニットが異常状態にあることを条件にその相手の CPU ユニットの前記リセット処理手段が動作してリセット処理を行うようにしたことを特徴とする請求項 1 に記載のプログラマブルコントローラ。

【請求項 3】

前記リセット信号出力手段から前記リセット信号は、相手が起動されるまで繰り返し出力するようにし、

20

かつ、前記リセット信号が出力された回数を計数し、その回数が予め設定した基準値に達するか、または越えた場合には、リセット信号を出力しないようにしたことを特徴とする請求項 1 または 2 に記載のプログラマブルコントローラ。

【請求項 4】

二重化対応の CPU ユニットであって、ユーザプログラムに基づく制御を実行する制御系と、前記制御を実行しない待機系の 2 つのモードを取り、相手の CPU ユニットの状態を確認する機能を持ち、前記待機系の時に相手の CPU ユニットがダウンしたことを検知すると、制御系に切り替わって運転を継続する CPU ユニットにおいて、

30

相手の CPU ユニットに向けてリセット信号を出力するリセット信号出力手段と、リセット信号を出力した相手の CPU ユニットから応答を受けた際に二重化のために必要なデータを前記相手の CPU ユニットに向けて送る手段と、

相手方の CPU ユニットから送られてきたリセット信号を受信すると、初期処理その他のリセット処理を行って起動するリセット処理手段とを備え、

前記相手の CPU ユニットの状態を確認する機能により相手が異常発生したことを検知した際に、前記リセット信号出力手段がリセット信号を出力するようにしたことを特徴とする CPU ユニット。

【発明の詳細な説明】**【0001】****【発明の属する技術分野】**

40

この発明は、プログラマブルコントローラ及び CPU ユニットに関するもので、より具体的には、2 台の CPU ユニットの二重化を図ったプログラマブルコントローラに関する。

【0002】**【従来の技術】**

ファクトリーオートメーション (FA) の制御装置として、プログラマブルコントローラ (PLC) が用いられている。この PLC は、複数のユニットから構成される。すなわち、電源供給源の電源ユニット、PLC 全体の制御を統率する CPU ユニット、FA の生産装置や設備装置の適所に取り付けられたスイッチやセンサの信号を入力する入力ユニット、アクチュエータなどに制御出力を出す出力ユニット、通信ネットワークに接続するための通

50

信ユニットなどの各種のユニットを適宜組み合わせる構成される。

【0003】

PLCのCPUユニットにおける制御は、入力ユニットで入力した信号をCPUユニットのI/Oメモリに取り込み（INリフレッシュ）、予め登録されたユーザプログラム記述言語（例えばラダー言語）で組まれたユーザプログラムに基づき論理演算をし（演算実行）、その演算実行結果をI/Oメモリに書き込んで出力ユニットに送り出し（OUTリフレッシュ）、その後、いわゆる周辺処理を行うと言うことをサイクリックに繰り返し処理するようになる。

【0004】

ところで、システムの安全性、信頼性を向上するため、PLCを構成する各ユニットを二重化することが行われる。例えばCPUユニットを二重化する場合、従来のCPUユニットを2個設ける場合とともに、それら2つのCPUユニットをCPU間バスにより接続する。そして、それら2個のCPUユニットは、基本的に同一の機能を持ったものであり、同じユーザプログラムを実行する。

10

【0005】

そして、2台のCPUユニットのうち、一方が動作系のCPUユニットとなっており、実際にサイクリックな処理を行い、メモリに対して読み書きを行ったり、外部のI/O機器等との間で制御データ（I/Oデータ）の送受を行い、FAネットワークシステムの制御を司る。また、他方のCPUユニットは待機系のCPUユニットとなり、待機中は、動作系のユーザプログラムと同一のユーザプログラムを実行するものの、演算実行結果を出力・反映することはない。そして、制御系PLCから処理結果等を受信し、待機系のCPUユニットのメモリの更新を行う。これにより、待機系のCPUユニットと制御系のCPUユニットのメモリの同一性が確保される。

20

【0006】

そして、制御系のCPUユニットが故障した場合には、待機系のCPUユニットが制御系のCPUユニットに切り替わって実際の制御等の動作を行う。従って、CPUユニットに故障が発生しても、すぐにシステムが停止することなく、継続して運転できるので信頼性が向上する。

【0007】

【発明が解決しようとする課題】

しかしながら、上記した従来のシステムでは、以下に示す問題がある。すなわち、制御系と待機系のCPUユニットがともに正常に動作している状態では二重化システムが保証されているので、その状態で制御系が故障・異常によりダウンした場合でも、待機系のCPUユニットが制御系に切り替わることにより、連続してシステムの運転を継続できる。しかし、係る切り替わりにより一旦待機系のCPUユニットが制御系として動作を開始すると、それまでの制御系はダウンしているため、非二重化システムとして運転を継続することになり、二重化システムの高信頼性が維持できない。その結果、その非二重化システムの状態ですら動作中のCPUユニットに故障・異常が発生すると動作を停止しなければならない。

30

【0008】

また、待機系のCPUユニットが故障・異常によりダウンすると、制御系がそのまま動作するので、制御は継続して行えるものの、非二重化システムとなるため、この状態でさらに動作中のCPUユニットに故障・異常が発生すると動作を停止しなければならない。

40

【0009】

そのため、ユーザにとっては、現在の状態が二重化であるか非二重化であるかを常に監視し、非二重化システムになった場合には、現場まで足を運び、二重化システムを立ち上げる必要がある。つまり、ノイズなどが原因によりダウンした場合には、ハード的には問題がないので、電源を再投入し、初期化処理、自己診断を経て立ち上げ、待機系のCPUユニットとして動作させる。

50

【0010】

また、ハードの故障により切り替わった場合には、電源を再投入してもCPUユニットを正常に復帰することができない。そして、非二重化システムになったのが、二重化システムへ復旧可能なノイズ等が原因なのか、復旧できないハードの故障なのかは、現場に実際に行かないとわからず、ハードの故障の場合には、現場に行き行ってハードの故障とわかると、原因を特定後、後日対策する必要が生じることがある。

【0011】

このように、二重化システムへ復旧するためには、常時監視し、非二重化システムになった場合に、必ず現場へ足を運ぶ必要があるので煩雑であり、しかも、実際に現場に行かなければ非二重化になった原因を特定できないので、ハードの故障の場合に、再度現場に行くといった二度手間となるという問題がある。

10

【0012】

この発明は、CPUユニットを二重化したプログラマブルコントローラにおいて、一方のCPUユニットがダウンして非二重化状態になった場合でも、自動的に二重化システムに復帰することができるプログラマブルコントローラ及びCPUユニットを提供することを目的とする。

【0013】

【課題を解決するための手段】

この発明によるプログラマブルコントローラは、二重化された2つのCPUユニットを備え、前記2つのCPUユニットは、相手の状態を確認可能とし、一方が制御系となるとともにも他方が待機系となり、前記制御系のCPUユニットがダウンした場合には、前記待機系のCPUユニットが制御系に切り替わって運転を継続するプログラマブルコントローラを前提とする。係る前提において、前記2つのCPUユニットは、それぞれ相手のCPUユニットに向けてリセット信号を出力するリセット信号出力手段と、相手方のCPUユニットから送られてきたリセット信号を受信すると、初期処理その他のリセット処理を行って起動するリセット処理手段とを備えた。そして、前記2つのCPUユニットのうち一方が異常発生した際に、他方のCPUユニットに備えた前記リセット信号出力手段がリセット信号を出力するようにした。

20

【0014】

ここで、リセット信号出力手段は、実施の形態では、図4のステップ14を実行する機能に対応する。また、リセット処理手段は、実施の形態では、図3のステップ3から7を実施する機能に対応する。

30

【0015】

このようにすると、一方のCPUユニットが異常を発生してダウンし、非二重化システム状態となった場合に、正常に動作している他方のCPUユニットからダウンした一方のCPUユニットに対してリセット信号を出力する。このリセット信号を受けて一方のCPUユニットはリセットされて再立ち上げされ、その後、二重化に必要なデータを前記他方のCPUユニットから取得することにより、二重化システムが稼働する。このように、自動的に二重化システムに復帰させることができる。

【0016】

なお、二重化を図るためには、リセット信号を出力した相手のCPUユニットに対し、正常に動作しているCPUユニットが持つメモリに格納されたデータを、相手のCPUユニットに対して送り、2つのCPUユニットにおいてメモリ内容の同一性を図る必要がある。係る処理は、例えば、正常に起動した相手方から正常に起動した旨の通知を受けた後で、所定のタイミングで正常に動作していたCPUユニットが必要なデータを送るようにしても良いし、係る相手方のCPUユニットが正常に動作しているCPUユニットのメモリにアクセスし、取得するようにしても良い。また、共有メモリを用いた転送処理を利用し、正常に動作しているCPUユニットが所定のメモリアreaに必要なデータを書き込むことで、自動的に(CPUユニットが関与しない)復帰したCPUユニットのメモリに書き込むようにしても良い。

40

50

【0017】

また、前記リセット信号出力手段からリセット信号が出力された場合、前記相手のCPUユニットが異常状態にあることを条件にその相手のCPUユニットの前記リセット処理手段が動作してリセット処理を行うようにするとよい。この機能は、実施の形態では、デュプレックスユニット20により実現されているが、CPUユニットに組み込んでよい。もちろん、係る機能は必須ではない。係る構成を採ると、正常に動作している相手に対して過ってリセット信号を出力した場合には、係る正常に動作している相手方はリセット処理をしない。

【0018】

さらに、前記リセット信号出力手段から前記リセット信号は、相手が起動されるまで繰り返し出力するようにし、かつ、前記リセット信号が出力された回数を計数し、その回数が予め設定した基準値に達するか、または越えた場合には、リセット信号を出力しないような機能を設けると良い。 10

【0019】

もちろん、係る機能は必須ではなく、相手が起動されるまで何回も繰り返しリセット信号を出力し続けるようにしても良いし、リセット信号を1回だけ出力するようにしてもよい。そして、上記のように基準値を設定した場合には、基準値を超えてもリセットできない場合には、ハードの故障があったと判断できる。よって、ハードの故障でなければ、自動的に二重化に復旧することができ、また、ハードの故障の場合には、現在非二重化状態であることが分かるとハードの故障であると推定できるので、現場に行くことなく、知ることができ。 20

【0020】

さらに、本発明に係るCPUユニットは、二重化対応のCPUユニットであって、ユーザプログラムに基づく制御を実行する制御系と、前記制御を実行しない待機系の2つのモードを取り、相手のCPUユニットの状態を確認する機能を持ち、前記待機系の時に相手のCPUユニットがダウンしたことを検知すると、制御系に切り替わって運転を継続するCPUユニットを前提とする。そして、相手のCPUユニットに向けてリセット信号を出力するリセット信号出力手段と、リセット信号を出力した相手のCPUユニットから応答を受けた際に二重化のために必要なデータを前記相手のCPUユニットに向けて送る手段と、相手方のCPUユニットから送られてきたリセット信号を受信すると、初期処理その他のリセット処理を行って起動するリセット処理手段とを備え、前記相手のCPUユニットの状態を確認する機能により相手が異常発生したことを検知した際に、前記リセット信号出力手段がリセット信号を出力するようにした。 30

【0021】

【発明の実施の形態】

図1は、本発明の好適な一実施の形態を示している。良く知られているように、PLC10は、複数のユニットから構成され、本発明が対象とする二重化システムに対応する場合には、CPUユニットを2台設けている。すなわち、ユーザプログラムを演算実行したり、I/Oリフレッシュや周辺処理をサイクリックに実行する機能を持つ第1,第2CPUユニット11,12や、電源ユニット13や、入出力機器を接続するI/Oユニット14 40
や、上記ネットワークに接続し、他の装置と通信を行う通信ユニット15等がある。これらのユニットは、バスを介して接続される。もちろん、これらの他にもユニットは存在し、必要に応じて連結するユニットを増減する。

【0022】

第1,第2CPUユニット11,12は、同一のユーザプログラムを実行するもので、実際のシステム稼動時には、何れか一方が制御系となり、他方が待機系となる。そして、制御系となったCPUユニットが行ったユーザプログラムの演算結果が、制御系に反映される。また、待機系のCPUユニット側でも同一のユーザプログラムは実行されるが、その結果が出力されない。さらに、待機系のCPUユニットは、制御系のCPUユニットから実行結果を受け取り、自己のメモリの更新などの所定の処理を実行する。これにより、待機 50

系のCPUユニットは、制御系のCPUユニットと同じ状態を維持する。従って、制御系のCPUユニットが異常によりダウンした場合には、直ぐに待機系であったCPUユニットが制御系に切り替わり、制御動作を実行することができる。

【0023】

ここで本発明では、第1, 第2CPUユニット11, 12間のバス制御や、運転中の表示、制御系/待機系の切り替えなどを行う二重化コントロールユニットたるデュプレックスユニット20を設けている。このデュプレックスユニット20には、相手方(ダウン中)のCPUユニットに対してリセット信号を送る機能を有している。これにより、リセット信号を受信したCPUユニットは、初期化処理並びに自己診断処理を行い、稼動する。

【0024】

そして、本発明の要部となるデュプレックスユニット20の内部構造と第1, 第2CPUユニット11, 12の関係は、図2に示すようになっている。すなわち、デュプレックスユニット20は、本発明との関係で言うと、一方のCPUユニットから送られるリセット信号を受け、他方のCPUユニットに実際にリセット信号を与える機能を持ち、2つのアンド回路21a, 21bを備えている。

【0025】

そして、第1アンド回路21aは、第1CPUユニット11から第2CPUユニット12に向けてリセット信号を送るためのもので、第1CPUユニット11のリセットフラグ端子(リセット信号の出力端子)11aと、第2CPUユニット12のWDT(ウォッチドックタイマ)端子12cを入力とし、出力を第2CPUユニット12のリセット端子12

10

20

【0026】

同様に、第2アンド回路21bは、第2CPUユニット12から第1CPUユニット11に向けてリセット信号を送るためのもので、第2CPUユニット12のリセットフラグ端子(リセット信号の出力端子)12aと、第1CPUユニット11のWDT端子11cを入力とし、出力を第1CPUユニット11のリセット端子11bに与えるようになっている。

【0027】

ところで、WDT端子11c, 12cからは、各CPUユニット11, 12が異常状態のときにONとなる異常信号が出力され、この異常信号は相手側のPCUユニット12, 11にも与えられ、これにより各CPUユニット11, 12は相手方が異常状態にあるか否かを監視・判断する。

30

【0028】

そして、この異常信号を第1, 第2アンド回路21a, 21bに入力するようにしたことにより、相手先が異常状態(WDT端子がON)のときに、リセットフラグ端子12aからリセット信号が出力された場合に、相手方のCPUユニットに対してリセット信号が与えられる。これにより、相手方が正常に動作している際に誤ってリセット信号が与えられてリセット処理されるのが防止できる。

【0029】

次に、二重化システムに自動復帰させるための各第1, 第2CPUユニット11, 12の機能を説明する。第1, 第2CPUユニット11, 12は、制御系と実行系のいずれにもなり得る。そして、現在の状況に応じて、図3, 図4に示す何れかのフローチャートを実施する機能を有する。

40

【0030】

すなわち、現在制御系のCPUユニットの場合には、通常のサイクリックな処理を実行し、外部機器との通信等の運転を行っているが、このとき異常が発生すると(ST1)、待機系に切り替わる(ST2)。一方、待機系のCPUユニットの場合には、相手に異常があるか否かを常時監視(ST11)し、異常があった場合には制御系に切り替わる(ST12)。これにより、以後制御系としての上記のサイクリックな処理を実行し、FAシステムを継続して運転を続行する。なお、この場合の待機系は、実際には例えば暴走したり

50

、ハードの故障などで動作できない状態となっているので、非二重化システムの状態となる。つまり、この状態でさらに現在の制御系のCPUユニットが異常によりダウンするとシステムは停止することになる。

【0031】

そこで、サイクリックな処理とは別に、制御系になったCPUユニットは、 $n = 1$ をセット後、待機系のCPUユニットに対し、リセット信号を出力する(ST13, ST14)。なお、WDT端子がONにならない異常で、かつ、二重化運転を継続できない異常がある場合、このリセット信号に先立ち、相手方のCPUユニットに対してCPU間通信などによりWDTをONにする命令を送る。

【0032】

これにより、リセットフラグ端子(11a あるいは 12a)がONになり、対応するアンド回路の一方の入力端子がONになる。そして、相手方(待機系)のCPUユニットのWDT端子(11c あるいは 12c)がONになっていると、アンド回路(21a あるいは 21b)がONになり、待機系のCPUユニットのリセット端子(11b あるいは 12b)がONになり、その待機系のCPUユニットに対してリセット信号が与えられる。

10

【0033】

これを受けて、待機系のCPUユニットは、メモリクリアその他の初期処理を行った後(ST3)、自己チェック(自己診断)を行い(ST4)、不良箇所の有無を判断する(ST5)。そして、不良箇所があった場合には、WDTエラーになり(ST6)、不良箇所がない場合には正常復帰するので、復帰したことをCPU間通信などにより相手に応答する(ST7)。

20

【0034】

一方、リセット信号を送った制御系のCPUユニットは、相手からの応答を待ち(ST15)、応答がない場合にはステップ14のリセット信号を送ってから一定時間経過するまで応答を待つ(ST16)。そして、一定時間経過しても応答がない場合(ステップ16でNo)には、 n を1インクリメントし、 n が設定数(例えば10)を越えたか否かを判断する(ST17, ST18)。

【0035】

設定数を越えていない場合には、ステップ14に戻り、再びリセット信号信号を出力する。このようにして、所定回数リセット信号を出力しても応答がない場合(ステップ18でYes)には、復帰できないハード的な故障が生じていると判断し、自動復帰処理を終了する。

30

【0036】

一方、相手からの応答があると、ステップ15の分岐判断がYesとなり、ステップ19に飛び、必要なデータ(自己が保有するI/Oデータ等)を待機系のCPUユニットへ転送する。これにより、待機系のCPUユニットは、転送されてきたデータを取得することにより、制御系のCPUユニットの状態と同じになり、二重化システムが復旧される。

【0037】

なお、上記したフローチャートは、制御系のCPUユニットが異常発生した場合の処理機能を説明したが、二重化システムで運転中に待機系のCPUユニットが異常になることももちろんある。この場合には、制御系のCPUユニットも相手に異常があるか否かを常時監視しているので、待機系のCPUユニットが異常によりダウンしたことを検知すると、ステップ13からの処理を実行するようになる。

40

【0038】

これにより、待機系、制御系のいずれのCPUユニットが異常によりダウンした場合でも、正常に動作しているCPUユニットからのリセット信号にしたがい、リセット処理されるため、たとえCPUが暴走などしていても確実に復帰することができる。もちろん、偶発的なノイズによるメモリ化けによるメモリ異常、プログラムエラーによる異常などに対しても、自動的に復帰することができる。しかも、この復帰処理の際に、初期化処理に続いて自己チェック(自己診断)が行われるので、故障の有無がその都度判断されるので、

50

待機系として稼働するCPUユニットの信頼性が保証される。

【0039】

なお、上記した実施の形態では、デュプレックスユニットを設けたが、本発明は必ずしも係るユニットを設ける必要はなく、デュプレックスユニットと同様の機能を一方のCPUユニットに組み込んでも良いし、各アンド回路をそれぞれのCPUユニットに組み込んでも良い。さらには、アンド回路を設けることなく、互いのリセットフラグ端子を相手方のリセット端子に接続するようにしても良く、各種の変形実施が可能である。

【0040】

【発明の効果】

以上のように、この発明では、CPUユニットを二重化したプログラマブルコントローラにおいて、一方のCPUユニットがダウンして非二重化状態になった場合でも、他方のCPUユニットからダウンしたCPUユニットに向けてリセット信号が送られるので、それに応答してリセット処理されて正規に起動し、自動的に二重化システムに復帰することができる。

【図面の簡単な説明】

【図1】本発明の好適な一実施の形態を示す図である。

【図2】本発明の好適な一実施の形態の要部であるデュプレックスユニットの内部構造を示す図である。

【図3】第1の実施の形態の機能を説明するフローチャートである。

【図4】第1の実施の形態の機能を説明するフローチャートである。

【符号の説明】

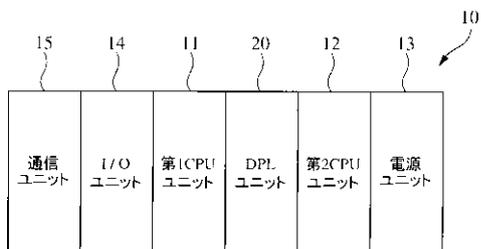
- 10 PLC
- 11 第1CPUユニット
- 11a リセットフラグ端子
- 11b リセット端子
- 11c WDT端子
- 12 第2CPUユニット
- 12a リセットフラグ端子
- 12b リセット端子
- 12c WDT端子
- 13 電源ユニット
- 14 I/Oユニット
- 15 通信ユニット
- 20 デュプレックスユニット
- 21a 第1アンド回路
- 21b 第2アンド回路

10

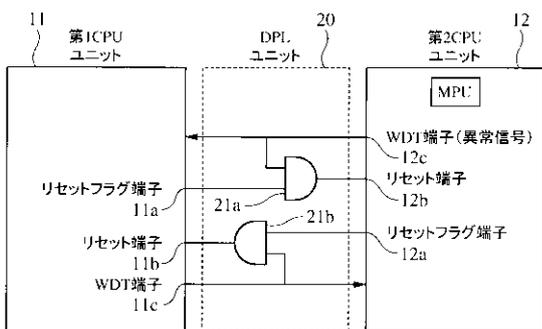
20

30

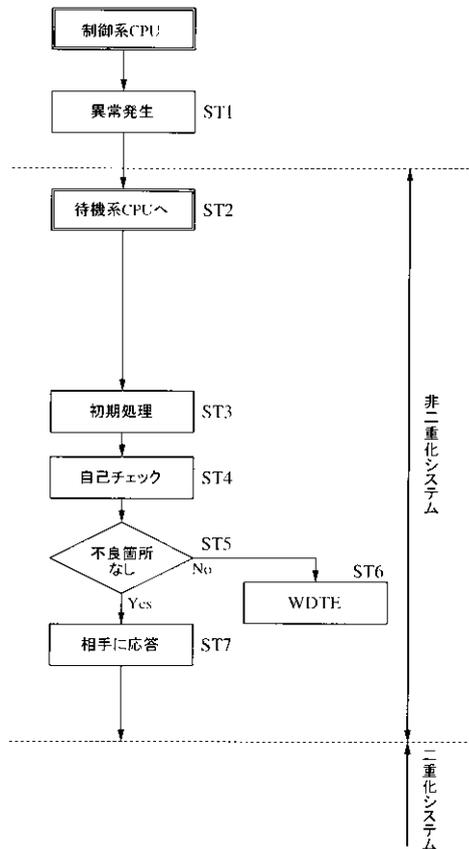
【図1】



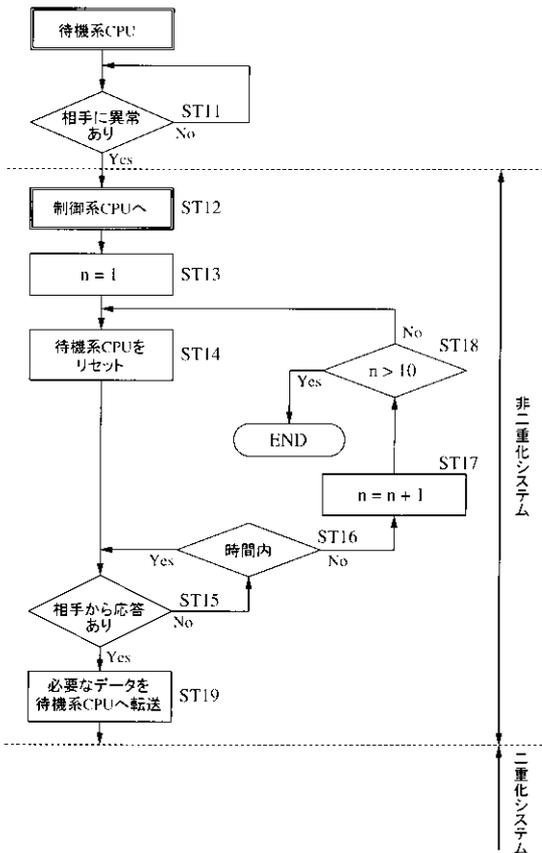
【図2】



【図3】



【図4】



フロントページの続き

Fターム(参考) 5B054 BB05 CC01
5H209 AA01 AA05 CC03 DD04 EE11 GG04 HH06 HH13 SS01 SS04
SS08 TT01
5H220 AA01 AA04 AA08 BB09 BB13 CC02 CC03 CC09 CX09 HH04
MM08