



(12)发明专利

(10)授权公告号 CN 103809502 B

(45)授权公告日 2018.04.20

(21)申请号 201310566385.3

(22)申请日 2013.11.14

(65)同一申请的已公布的文献号
申请公布号 CN 103809502 A

(43)申请公布日 2014.05.21

(30)优先权数据
2012-250658 2012.11.14 JP

(73)专利权人 欧姆龙株式会社
地址 日本京都府京都市

(72)发明人 冈村弘太郎 西山佳秀 太田政则
矢尾板宏心

(74)专利代理机构 隆天知识产权代理有限公司
72003
代理人 金相允 浦柏明

(51)Int.Cl.

G05B 19/05(2006.01)

G06F 17/30(2006.01)

(56)对比文件

JP 特開2012-108642 A,2012.06.07,
CN 102456052 A,2012.05.16,

审查员 赵磊

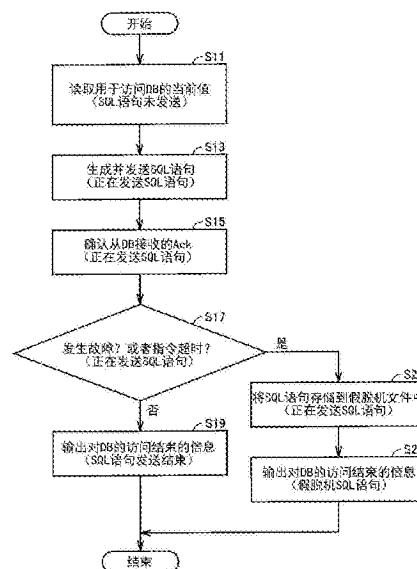
权利要求书2页 说明书9页 附图9页

(54)发明名称

控制器及记录介质

(57)摘要

提供一种提高了PLC等控制器访问数据库系统的可靠性的技术。能够访问数据库的PLC通过用户程序调出DB访问处理程序来访问DB服务器。DB访问处理程序按照用户程序的指令,来将表示访问DB服务器的访问状况的状态信息存储至非易失性存储器。在发生断电时,PLC能够通过参照状态信息,来获取在发生断电的时刻的访问DB服务器的访问状况,由此根据访问状况来进行SQL语句的再次发送等的处理。



1. 一种控制器,其特征在于,具有:

指令执行部,其执行用于对控制对象进行控制的用户程序,

通信接口,其用于连接数据库系统,

非易失性存储器,

访问处理部,其通过执行包含用于访问所述数据库系统的访问指令在内的所述用户程序,来生成与所述访问指令相对应的指令语句,基于所生成的指令语句来控制对数据库系统的访问,

状态输出部,其在从所述访问处理部开始基于与所述访问指令相对应地生成的所述指令语句来执行用于访问所述数据库系统的访问处理起,至基于该指令语句的处理结束为止的期间内,生成表示所述指令语句的执行状况的状态信息,并输出所生成的状态信息;所述访问处理部包括指令输出部,该指令输出部在到基于所述指令语句的处理结束为止的期间内,将所述指令语句存储至所述非易失性存储器;

所述状态信息是表示用于访问所述数据库系统的SQL语句的发送状况的信息,并保持以下五个状态中的任意状态,该五个状态包括:

初始状态,表示访问所述数据库系统的初始状态,

未发送状态,表示向所述数据库系统发送所述指令语句之前的状态,

正在发送状态,表示正在向所述数据库系统发送所述指令语句并且尚未发送结束的状态,

假脱机状态,表示向所述数据库系统发送的所述指令语句已被假脱机,而所述指令语句保持在所述非易失性存储器中的状态,

发送结束状态,表示向所述数据库系统发送所述指令语句的处理结束的状态;

在发生断电时生成的所述状态信息中保持的状态是所述未发送状态的情况下,所述访问处理部再次向所述数据库系统发送所述指令语句,

在发生断电时生成的所述状态信息中保持的状态是所述正在发送状态的情况下,所述访问处理部再次向所述数据库系统发送所述指令语句,

在发生断电时生成的所述状态信息中保持的状态是所述假脱机状态的情况下,所述访问处理部向所述数据库系统发送被保持于所述非易失性存储器中的所述指令语句,

在发生断电时生成的所述状态信息中保持的状态是所述发送结束状态的情况下,不进行处理。

2. 如权利要求1所述的控制器,其特征在于,

该控制器还具有保持部,该保持部保持恢复程序,该恢复程序用于参照所述状态信息,根据状态信息来恢复对数据库系统的访问处理;

该控制器在处于规定的动作状态时,执行所述恢复程序。

3. 如权利要求2所述的控制器,其特征在于,

所述恢复程序,根据从所述控制器向所述数据库系统发送的SQL语句的发送状况来进行恢复。

4. 一种计算机可读的记录介质,所述记录介质中保存有用于对控制器的动作进行控制的程序,所述控制器具有处理器、用于连接数据库系统的通信接口和非易失性存储器,该程序的特征在于,

所述程序使所述处理器执行以下步骤：

通过执行包含用于访问所述数据库系统的访问指令在内的用户程序，来生成与所述访问指令相对应的指令语句，并基于所生成的指令语句来控制对数据库系统的访问，

在开始基于与所述访问指令相对应地生成的所述指令语句来执行用于访问所述数据库系统的访问处理起，至基于该指令语句的处理结束为止的期间内，生成表示所述指令语句的执行状况的状态信息，并输出所生成的状态信息；

在到基于所述指令语句的处理结束为止的期间内，将所述指令语句存储至所述非易失性存储器；

所述状态信息是表示用于访问所述数据库系统的SQL语句的发送状况的信息，并保持以下五个状态中的任意状态，该五个状态包括：

初始状态，表示访问所述数据库系统的初始状态，

未发送状态，表示向所述数据库系统发送所述指令语句之前的状态，

正在发送状态，表示正在向所述数据库系统发送所述指令语句并且尚未发送结束的状态，

假脱机状态，表示向所述数据库系统发送的所述指令语句已被假脱机，而所述指令语句保持在所述非易失性存储器中的状态，

发送结束状态，表示向所述数据库系统发送所述指令语句的处理结束的状态；

在发生断电时生成的所述状态信息中保持的状态是所述未发送状态的情况下，再次向所述数据库系统发送所述指令语句，

在发生断电时生成的所述状态信息中保持的状态是所述正在发送状态的情况下，再次向所述数据库系统发送所述指令语句，

在发生断电时生成的所述状态信息中保持的状态是所述假脱机状态的情况下，向所述数据库系统发送被保持于所述非易失性存储器中的所述指令语句，

在发生断电时生成的所述状态信息中保持的状态是所述发送结束状态的情况下，不进行处理。

控制器及记录介质

技术领域

[0001] 本发明涉及可编程逻辑控制器(PLC:Programmable Logic Controller)等控制器及程序,特别涉及向数据库系统发送数据的控制器及程序。

背景技术

[0002] PLC等控制器用于对工厂等的自动设备的控制等中,具有多个输入输出功能,例如依次读取从传感器等输出的值并保持(保存)数据。

[0003] 另外,近年来PLC通过连接数据库系统来将保持在PLC中的数据储存至数据库系统。由此,能够使得连接至数据库的信息处理装置之间的数据的共享变得容易,并且能够通过收集各种数据来使得数据的统计及解析等变得容易。例如,日本特开2012-108642号公报(专利文献1)公开了具有如下功能的PLC,该功能是指将包含多个用于访问数据库的SQL语句的SQL处理组发送至数据库的功能。

[0004] 现有技术文献

[0005] 专利文献

[0006] 专利文献1:日本特开2012-108642号公报

[0007] 但是,若对PLC等控制器的供电因某种原因而不稳定,则对数据库系统的访问会被中断。例如,以往的控制器将数据作为“Spool”(Simultaneous Peripheral Operation On Line:外部设备联机并行操作,以下称为“假脱机”)信息保持在非易失性存储器中,并将数据定期性地移动(保存、备份)至外部存储装置。控制器具有如下结构,即,在断电发生时,对于未能移动至外部存储装置的数据,在从断电开始到恢复供电之后,将未能移动的量该“Spool”信息再次移动至外部存储装置。此时,若在控制器正在将数据移动至外部存储装置的过程中发生断电,则有可能导致已经存在的已移动的数据发生破损。因发生这样的断电,会导致从控制器发送至数据库系统的数据消失,从而导致控制器的可靠性下降。

发明内容

[0008] 因此,本发明的目的在于,提供一种能够提高PLC等控制器的访问数据库系统的可靠性的技术。

[0009] 一个实施方式的控制器的具有:指令执行部,其执行用于对控制对象进行控制的用户程序;通信接口,其用于连接数据库系统;访问处理部,其通过执行包含用于访问数据库系统的访问指令在内的用户程序,来生成与访问指令相对应的指令语句,基于所生成的指令语句来控制对数据库系统的访问;状态输出部,其在从访问处理部开始基于与访问指令相对应地生成的指令语句来执行用于访问数据库系统的访问处理起,至基于该指令语句的处理结束为止的期间内,生成表示指令语句的执行状况的状态信息,并输出所生成的状态信息。

[0010] 优选地,控制器,还包括非易失性存储器;访问处理部,包括指令输出部,该指令输出部在到基于指令语句的处理结束为止的期间内,将指令语句存储至存储器。

[0011] 优选地,状态信息,也可以包括用于访问数据库系统的SQL语句的发送状况的信息。

[0012] 优选地,状态信息,也可以包括表示是否将用于访问数据库系统的SQL语句保留在存储部中的信息。

[0013] 优选地,控制器也可以还具有保持部,该保持部保持恢复程序,该恢复程序参照状态信息,根据状态信息所表示的访问的状况来恢复对数据库系统的访问处理;也可以在处于规定的动作状态时,执行恢复程序。

[0014] 优选地,恢复程序,也可以根据从控制器向数据库系统发送的SQL语句的发送状况来进行恢复。

[0015] 根据另一实施方式,提供用于对控制器的动作进行控制的程序。控制器具有处理器和用于连接数据库系统的通信接口,该程序的特征在于,程序使处理器执行以下步骤:通过执行包含用于访问数据库系统的访问指令在内的用户程序,来生成与访问指令相对应的指令语句,并基于所生成的指令语句来控制对数据库系统的访问;在访问处理部开始基于与访问指令相对应地生成的指令语句来执行用于访问数据库系统的访问处理起,至基于该指令语句的处理结束为止的期间内,生成表示指令语句的执行状况的状态信息,并输出所生成的状态信息。

[0016] 根据上述的一个实施方式,将控制器访问数据库系统时的访问状况保持为状态信息,因而即使在断电发生的情况下也能够恢复时参照访问状况,从而提高了访问数据库系统的可靠性。

[0017] 另一个实施方式的控制器具有:指令执行部,其执行用于对控制对象进行控制的用户程序,通信接口,其用于连接数据库系统,非易失性存储器,访问处理部,其通过执行包含用于访问所述数据库系统的访问指令在内的所述用户程序,来生成与所述访问指令相对应的指令语句,基于所生成的指令语句来控制对数据库系统的访问,状态输出部,其在从所述访问处理部开始基于与所述访问指令相对应地生成的所述指令语句来执行用于访问所述数据库系统的访问处理起,至基于该指令语句的处理结束为止的期间内,生成表示所述指令语句的执行状况的状态信息,并输出所生成的状态信息;所述访问处理部包括指令输出部,该指令输出部在到基于所述指令语句的处理结束为止的期间内,将所述指令语句存储至所述非易失性存储器;所述状态信息是表示用于访问所述数据库系统的SQL语句的发送状况的信息,并保持以下五个状态中的任意状态,该五个状态包括:初始状态,表示访问所述数据库系统的初始状态,未发送状态,表示向所述数据库系统发送所述指令语句之前的状态,正在发送状态,表示正在向所述数据库系统发送所述指令语句并且尚未发送结束的状态,假脱机状态,表示向所述数据库系统发送的所述指令语句已被假脱机,而所述指令语句保持在所述非易失性存储器中的状态,发送结束状态,表示向所述数据库系统发送所述指令语句的处理结束的状态;在发生断电时生成的所述状态信息中保持的状态是所述未发送状态的情况下,所述访问处理部再次向所述数据库系统发送所述指令语句,在发生断电时生成的所述状态信息中保持的状态是所述正在发送状态的情况下,所述访问处理部再次向所述数据库系统发送所述指令语句,在发生断电时生成的所述状态信息中保持的状态是所述假脱机状态的情况下,所述访问处理部向所述数据库系统发送被保持于所述非易失性存储器中的所述指令语句,在发生断电时生成的所述状态信息中保持的状态是所述发送

结束状态的情况下,不进行处理。

[0018] 另一个实施方式的计算机可读取的记录介质,所述记录介质中保存有用于对控制器的动作进行控制的程序,所述控制器具有处理器、用于连接数据库系统的通信接口和非易失性存储器,所述程序使所述处理器执行以下步骤:通过执行包含用于访问所述数据库系统的访问指令在内的用户程序,来生成与所述访问指令相对应的指令语句,并基于所生成的指令语句来控制对数据库系统的访问,在开始基于与所述访问指令相对应地生成的所述指令语句来执行用于访问所述数据库系统的访问处理起,至基于该指令语句的处理结束为止的期间内,生成表示所述指令语句的执行状况的状态信息,并输出所生成的状态信息;在到基于所述指令语句的处理结束为止的期间内,将所述指令语句存储至所述非易失性存储器;所述状态信息是表示用于访问所述数据库系统的SQL语句的发送状况的信息,并保持以下五个状态中的任意状态,该五个状态包括:初始状态,表示访问所述数据库系统的初始状态,未发送状态,表示向所述数据库系统发送所述指令语句之前的状态,正在发送状态,表示正在向所述数据库系统发送所述指令语句并且尚未发送结束的状态,假脱机状态,表示向所述数据库系统发送的所述指令语句已被假脱机,而所述指令语句保持在所述非易失性存储器中的状态,发送结束状态,表示向所述数据库系统发送所述指令语句的处理结束的状态;在发生断电时生成的所述状态信息中保持的状态是所述未发送状态的情况下,再次向所述数据库系统发送所述指令语句,在发生断电时生成的所述状态信息中保持的状态是所述正在发送状态的情况下,再次向所述数据库系统发送所述指令语句,在发生断电时生成的所述状态信息中保持的状态是所述假脱机状态的情况下,向所述数据库系统发送被保持于所述非易失性存储器中的所述指令语句,在发生断电时生成的所述状态信息中保持的状态是所述发送结束状态的情况下,不进行处理。

[0019] 本发明的上述及其他目的、特征、方式及优点,根据与附图相关联地理解的本发明的下面的详细说明来明确了解。

附图说明

[0020] 图1是示出了本实施方式的控制系统的整体的结构的图。

[0021] 图2是示出了PC10的功能的框图。

[0022] 图3是示出了DB(数据库)服务器20的功能的框图。

[0023] 图4是示出了PLC30的功能的框图。

[0024] 图5A、5B是示出了本实施方式的硬件结构的图。

[0025] 图6是示出了在状态信息51中保持的数据的图。

[0026] 图7是示出了通过DB访问处理程序35来访问DB服务器20的访问处理和输出用于表示访问DB服务器20的状态的输出变量State的图。

[0027] 图8是示出了按照用户程序32的指令通过DB访问处理程序35来开始访问DB服务器20并输出输出变量State的处理的图。

[0028] 图9是示出了断电时刻、输出变量State以及在发生断电时所采取的措施的图。

[0029] 图10是示出了用于通过DB访问处理程序35来输出状态信息51的用户程序的一个例子的图。

[0030] 图11是示出了在发生断电时用于恢复的用户程序的一个例子的图。

- [0031] 附图标记的说明
- [0032] 101 CPU
- [0033] 102 ROM
- [0034] 103 RAM
- [0035] 104 通信接口
- [0036] 105 显示器
- [0037] 106 输入输出接口
- [0038] 10 PC
- [0039] 12 编程工具
- [0040] 14 指令执行状态获取部
- [0041] 16 编程功能
- [0042] 20 DB服务器
- [0043] 22 DBMS
- [0044] 24 DB
- [0045] 30 PLC
- [0046] 31 OS
- [0047] 32 用户程序
- [0048] 33 系统程序
- [0049] 34 序列指令程序
- [0050] 35 DB访问处理程序
- [0051] 36 输入输出处理程序
- [0052] 37 工具接口处理程序
- [0053] 38 排程器
- [0054] 39 控制程序
- [0055] 40 PLC
- [0056] 51 连接设定信息(状态信息)

具体实施方式

[0057] 下面,参照附图,说明本发明的实施方式。在下面的说明中,对相同的部件标注有相同的附图标记。它们的名称及功能也相同。因此,不重复对它们的详细说明。

[0058] <第一实施方式>

[0059] 图1是示出了本实施方式的控制系统整体的结构的图。

[0060] 首先说明概要。参照图1,控制系统中包括PC(Personal Computer:个人计算机)10、DB(Data base:数据库)服务器20、PLC30及PLC40。PC10是用户用于作成在PLC30中进行动作的用户程序的信息处理装置。DB服务器20是为了管理数据库而作为DBMS(data base management system:数据库管理系统)来发挥功能的服务器。

[0061] PLC30与PC10、DB服务器20及PLC40相连接。PLC30保持用于与DB服务器20连接的设定(信息),并按照设定(信息)来访问DB服务器20。

[0062] 另外,PLC30从外部接收传感器的输出信息等并蓄积储存数据,并通过后述的DB访

问处理程序35,将所蓄积的数据发送至DB服务器20。PLC30接收由其他的PLC(在图1的例子中是PLC40)发送至DB服务器20的数据,并将所接收的数据发送至DB服务器20。PLC40从外部接收传感器的输出信息等来蓄积储存数据,并将所蓄积的数据经由PLC30发送至DB服务器20。

[0063] <各设备的详细结构>

[0064] 接着,说明各设备的详细结构。

[0065] 图2是示出了PC10的功能的框图。

[0066] PC10是具备有MCU(Micro Control Unit:微控制单元)、ROM(Read Only Memory:只读存储器)、RAM(Random Access Memory:随机存取存储器)等的计算机系统(信息处理装置)。参照图2,PC10发挥用户用于作成用户程序的编程工具12的功能,该用户程序用于使PLC30进行动作。编程工具12包括指令执行状态获取部14和编程功能16的功能。将用户通过编程工具12作成的用户程序发送至PLC30,并在PLC30中执行该用户程序。

[0067] 指令执行状态获取部14是用于对PLC30进行如下指示的软件,该指示是指,使PLC30将保持在PLC30的存储器中的状态信息51的内容发送至PC10的指示。

[0068] 编程功能16是辅助用户用于作成程序的软件。这些软件在PC10中被执行,由此发挥指令执行状态获取部14和编程功能16等的功能。

[0069] 图3是示出了DB服务器20的功能的框图。DB服务器20包括大容量的存储装置等。

[0070] DB服务器20发挥DBMS22的功能,管理读取及写入数据库。DBMS22控制向发挥数据库的功能的DB24写入数据以及从该DB24读取数据。DB24是用于保持多条数据记录的存储区域,按照DBMS22的控制来进行数据的保持等。

[0071] 图4是示出了PLC30的功能的框图。PLC30周期性执行用户程序。PLC30通过对PLC30中的存储器区域的信息进行读写,来依次执行用户程序所表示的指令。PLC30具有输入输出功能,从PLC30的外部接收传感器输出的数据并蓄积储存数据。PLC30将所蓄积的数据发送至DB服务器20。

[0072] 参照图4,PLC30的各功能是通过由OS(操作系统)31、系统程序33及用户程序32等进行动作来发挥的。PLC30保持状态信息51。

[0073] OS31是用于管理PLC30的资源的操作系统,使PLC30发挥工作排程(task schedule)以及对DB服务器20进行访问处理等的功能。

[0074] 系统程序33是用于提供PLC30的基本功能的软件组。具体而言,系统程序33包括序列指令程序34、DB访问处理程序35、输入输出处理程序36、工具接口处理程序37及排程器(scheduler)38。

[0075] 序列指令程序34对PLC30内的存储器区域的信息进行读写,并且使PLC30发挥依次执行用户程序32的指令的功能。

[0076] 随着PLC30通过序列指令程序34来执行用户程序32,而调出DB访问处理程序35,使PLC30执行用于使用户程序32访问DB服务器20的处理,例如发挥执行SQL语句生成等的处理的功能。

[0077] 输入输出处理程序36使PLC30发挥如下功能,即,控制通过PLC30所具备的多个输入输出功能来向PLC30输入数据以及从PLC30输出数据的处理。各输入输出功能分别分配有存储器的地址。与各输入输出功能相对应地保持在存储器中的数据,周期性地被一并替换

为PLC30外部的数据。工具接口处理程序37使PLC30发挥如下功能,即,通过在PC10中进行动作的编程工具12的功能来接收从PC10通知给PLC30的指示,并通过使DB访问处理程序35等进行处理,来将处理的结果等应答至PC10。排程器38使PLC30发挥基于由PLC30所执行的工作的优先级来进行工作排程的功能。

[0078] 用户程序32是用PC10来作成的,保持在PLC30的存储器中。参照图4,由用户程序32和序列指令程序34构成控制程序39。即,序列指令程序34从存储器中读取用户程序32并依次执行指令,由此使PLC30进行动作。

[0079] 通过DB访问处理程序35来输出状态信息51,该状态信息51保持有DB访问处理程序35访问DB服务器20的访问状况。在后面详细叙述。

[0080] 图5A、5B是示出了本实施方式的硬件结构的图。图5A示出了作为计算机系统的PC10的结构。参照图5A,PC10具有CPU101、ROM102、RAM103、通信接口104、显示器105及输入输出接口106。CPU101通过从ROM102等读取程序并执行该程序来使PC10发挥各种功能。ROM102是非易失性的存储区域,存储有系统数据等。RAM103是易失性存储器,保存有程序及数据。通信接口104是PC10用于与外部设备进行通信的接口。显示器105用于将CPU101等的计算结果通过图像来输出的显示设备。输入输出接口106是使PC10通过接收用户的输入操作等来输出PC10的动作状态的接口。

[0081] 图5B是示出了PLC30的硬件结构的图。PLC30具有CPU101、ROM102、RAM103、通信接口104及总线107等。PLC40也具有与PLC30相同的硬件结构。

[0082] <数据>

[0083] 接着,说明在本实施方式中所使用的数据。

[0084] 图6是示出了保持在状态信息51中的数据的图。状态信息51含有用于通过DB访问处理程序35来输出的输出变量State,该状态信息51保持在非易失性存储器中。

[0085] 在本实施方式中,在用户程序中指定输出变量State的输出,由此,PLC30通过DB访问处理程序35来输出输出变量State,并将其保持到非易失性存储器中。状态信息51,保持通过DB访问处理程序35来输出的输出变量State所表示的五个状态中的任意状态。输出变量State的数据类型是枚举类型(enumerated type),枚举数如图6所示。例如,枚举数具有:“_DBC_SEND_INIT”(初始状态),表示通过DB访问处理程序35来访问DB服务器20的初始状态;“_DBC_SEND_UNSENT”(SQL语句未发送),表示通过DB访问处理程序35向DB服务器20发送SQL语句之前(未发送)的状态;“_DBC_SEND_SENDING”(正在发送SQL语句),表示正在通过DB访问处理程序35来向DB服务器20发送SQL语句并且尚未发送结束的状态;“_DBC_SEND_SPOOLED”(SQL语句已被假脱机),表示通过DB访问处理程序35向DB服务器20发送的SQL语句已被假脱机(SPOOL:Simultaneous Peripheral Operation On Line)而SQL语句保持在非易失性存储器中的状态;“_DBC_SEND_COMPLETE”(SQL语句发送结束),表示通过DB访问处理程序35向DB服务器20发送SQL语句的处理结束的状态。

[0086] <动作>

[0087] 接着,利用附图,详细说明本实施方式的PLC30的动作。第一实施方式的动作的概要如下。PC10基于用户的操作输入来作成在PLC30中执行的用户程序。PLC30从PC10接收用户程序并存储该用户程序。PLC30伴随由序列指令程序34执行用户程序32而调出DB访问处理程序35。PLC30执行用于使用户程序32访问DB服务器20的处理,例如通过DB访问处理程序

35来进行SQL语句的生成和发送等。

[0088] 图7是示出了通过DB访问处理程序35来访问DB服务器20的访问处理和输出用于表示访问DB服务器20的状态的输出变量State的图。

[0089] 在步骤S11中,PLC30通过DB访问处理程序35来读取用于访问DB服务器20的值(既是作为向DB服务器20写入的对象的数据,又是存储在存储器中的数据的当前值)。此时输出的输出变量State是表示未发送SQL语句的“_DBC_SEND_UNSENT”。

[0090] 在步骤S13中,PLC30通过DB访问处理程序35来生成用于访问DB服务器20的SQL语句并将其发送至DB服务器20。此时输出的输出变量State是表示正在发送SQL语句的状态“_DBC_SEND_SENDING”。

[0091] 在步骤S15中,PLC30在超时之前一直待机,并且通过DB访问处理程序35,对于向DB服务器20发送的SQL语句,来确认是否收到来自DB服务器20的Ack响应(发送Ack)。此时输出的输出变量State是表示正在发送SQL语句的“_DBC_SEND_SENDING”。

[0092] 在步骤S17中,PLC30通过DB访问处理程序35来判断出以下两种情况中的至少任意一种情况,这两种情况是指:在与DB服务器20之间的通信中接收到了错误信息(error);接收Ack响应的处理超时。此时输出的输出变量State是表示正在发送SQL语句的“_DBC_SEND_SENDING”。在步骤S17中得到肯定的判断结果的情况下,PLC30通过DB访问处理程序35来进行步骤S21的处理,在判断结果为否定的情况下,通过DB访问处理程序35来进行步骤S19的处理。

[0093] 在步骤S19(通过DB访问处理程序35来向DB服务器20发送SQL语句的处理正常结束)中,PLC30通过DB访问处理程序35,来向调出了DB访问处理程序35的程序输出表示对DB服务器20的访问结束的响应。此时输出的输出变量State是表示SQL语句的发送结束的“_DBC_SEND_COMPLETE”。

[0094] 在步骤S21中,PLC30通过DB访问处理程序35,将向DB服务器20的发送未正常结束的SQL语句被假脱机至非易失性存储器。在到假脱机处理结束为止的期间内输出的输出变量State是表示正在发送SQL语句的“_DBC_SEND_SENDING”。

[0095] 在步骤S23中,PLC30将表示通过DB访问处理程序35对DB服务器20的访问已结束的响应,输出至调出了DB访问处理程序35的程序。此时输出的输出变量State是表示SQL语句已被假脱机的“_DBC_SEND_SPOOLED”。

[0096] 图8是示出了按照用户程序32的指令通过DB访问处理程序35来开始访问DB服务器20并输出输出变量State的处理的图。

[0097] (1) 开始

[0098] 就包含在用户程序32中的DB指令60而言,是用于调出DB访问处理程序35来访问DB服务器20的指令,包括表示向DB服务器20进行连接的连接指令的DB_Connect指令61以及向DB服务器20插入数据的DB_Insert指令62等。通过执行包含将输出变量State输出至DB访问处理程序35的DB指令60,PLC30调出DB访问处理程序35并开始输出输出变量State。

[0099] (2) 读取当前值

[0100] PLC30通过DB访问处理程序35来执行步骤S11的处理。

[0101] (3) 发送

[0102] PLC30通过DB访问处理程序35来执行步骤S13的处理。

[0103] (4) 发送Ack

[0104] PLC30通过DB访问处理程序35来执行步骤S15及步骤S17的处理。

[0105] (5-1) 发生故障时, (5-2) 指令超时时

[0106] PLC30通过DB访问处理程序35来执行步骤S21的处理, 并将SQL语句假脱机至非易失性存储器66。

[0107] (6-1) 结束, (6-2) 结束(假脱机)

[0108] PLC30通过DB访问处理程序35来执行步骤S19或步骤S23的处理。

[0109] 图9是示出了断电时刻、输出变量State以及在发生断电时所采取的措施的图。

[0110] 图9中的断电时刻(1)开始、(2)当前值的读取、(3)发送、(4)发送Ack、(5-1)发生故障时、(5-2)指令超时时、(6-1)结束以及(6-2)结束(假脱机), 与在图8中说明的各时刻相对应。参照图9, 在断电时刻为“(1)开始”和“(2)当前值的读取”的情况下, 未通过DB访问处理程序35来向DB服务器20发送SQL语句。因此, 发生断电时所采取的措施如下: 用户程序被执行而通过再次调出DB访问处理程序35来再次发送SQL语句。

[0111] 在断电时刻为“(3)发送、(4)发送Ack、(5-1)发生故障时以及(5-2)指令超时时”的情况下, 由于未通过DB访问处理程序35来发送SQL语句, 因而发生断电时所采取的措施如下: 确认DB服务器20一侧有无记录(record), 并在DB服务器20中未存储记录的情况下再次发送SQL语句。

[0112] 在断电时刻为“(6-1)结束”的情况下, 由于通过DB访问处理程序35向DB服务器20发送SQL语句的处理已结束, 因而即使在发生断电时也不需要特别的措施。

[0113] 在断电时刻为“(6-2)结束(假脱机)”的情况下, 由于SQL语句已被假脱机, 因而发生断电时所采取的措施如下: 通过DB访问处理程序35来再次发送已假脱机的SQL语句。

[0114] <用户程序的例子>

[0115] 如上所述, 对PLC30通过DB访问处理程序35来输出访问DB服务器20时将访问状况输出至状态信息51的动作进行了说明。在此, 对用于将状态信息51输出至DB访问处理程序35的用户程序的例子和在发生断电时用于恢复的用户程序的例子进行说明。

[0116] 图10是示出了用于通过DB访问处理程序35来输出状态信息51的用户程序的一个例子的图。

[0117] 参照图10, 例如, 在由用户程序32用于调出DB访问处理程序35的指令(DB_Insert指令62等)中, 包含有用于保存从DB访问处理程序35输出的访问DB服务器20的访问状况的指示。

[0118] 图11是示出了在发生断电时用于恢复的用户程序的一个例子的图。此外, 图11与用户程序的一个例子一同示出了功能块“DB_Insert”的一个例子。参照图11, 在用户程序中包含有如下的指令, 该指令使得在对PLC30的供电后的第一周期(“P_First_RunMode”)内参照保存在非易失性存储器中的状态信息51。在供电后的第一周期的状态信息51中表示为“_DBC_SEND_UNSENT”(SQL语句未发送)的情况下, 再次发送SQL语句。另外, 在状态信息51中表示为“_DBC_SEND_SENDING”(正在发送SQL语句)的情况下, PLC30确认DB服务器20有无记录, 并根据确认结果来再次发送SQL语句。这样, 恢复程序也可以使PLC30执行与SQL语句的发送状况相对应的处理。

[0119] 这样对各实施方式进行了说明, 但显然也可以组合这些实施方式。以上, 基于实施

方式具体说明了本发明人员们做出的发明,但本发明并不限于所述实施方式,显然能够在不脱离其宗旨的范围内进行各种变更。

[0120] 应当认为本公开的实施方式是在全部点的例示而非限制。本发明的范围并不由无的说明来表示,而是由权利要求书来表示,意在包括在与权利要求书均等的意思和范围内的全部变更。

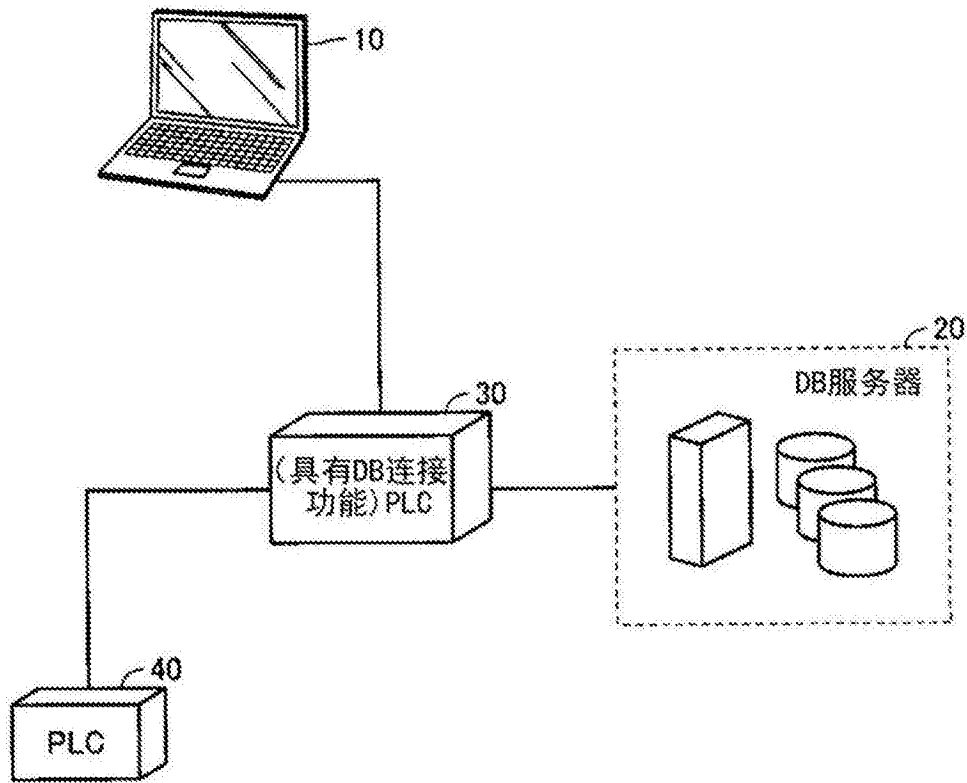


图1

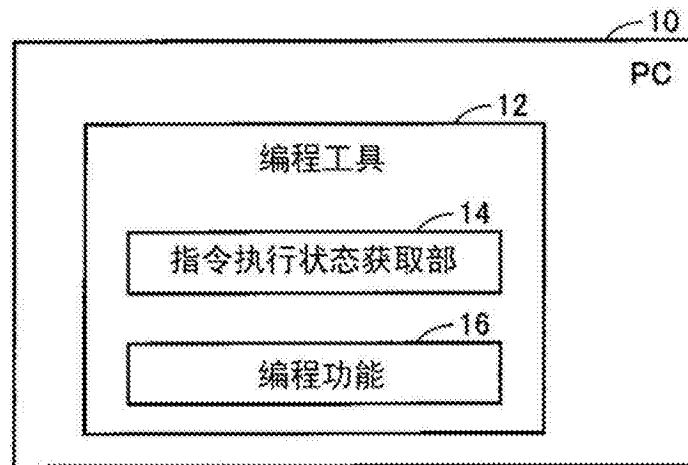


图2

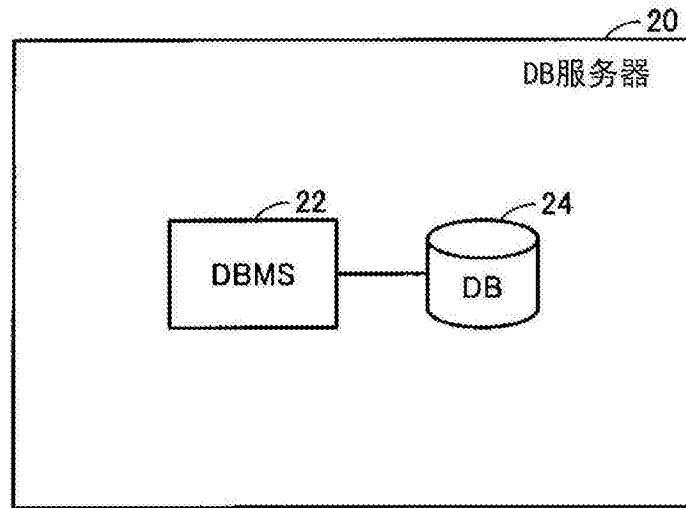


图3

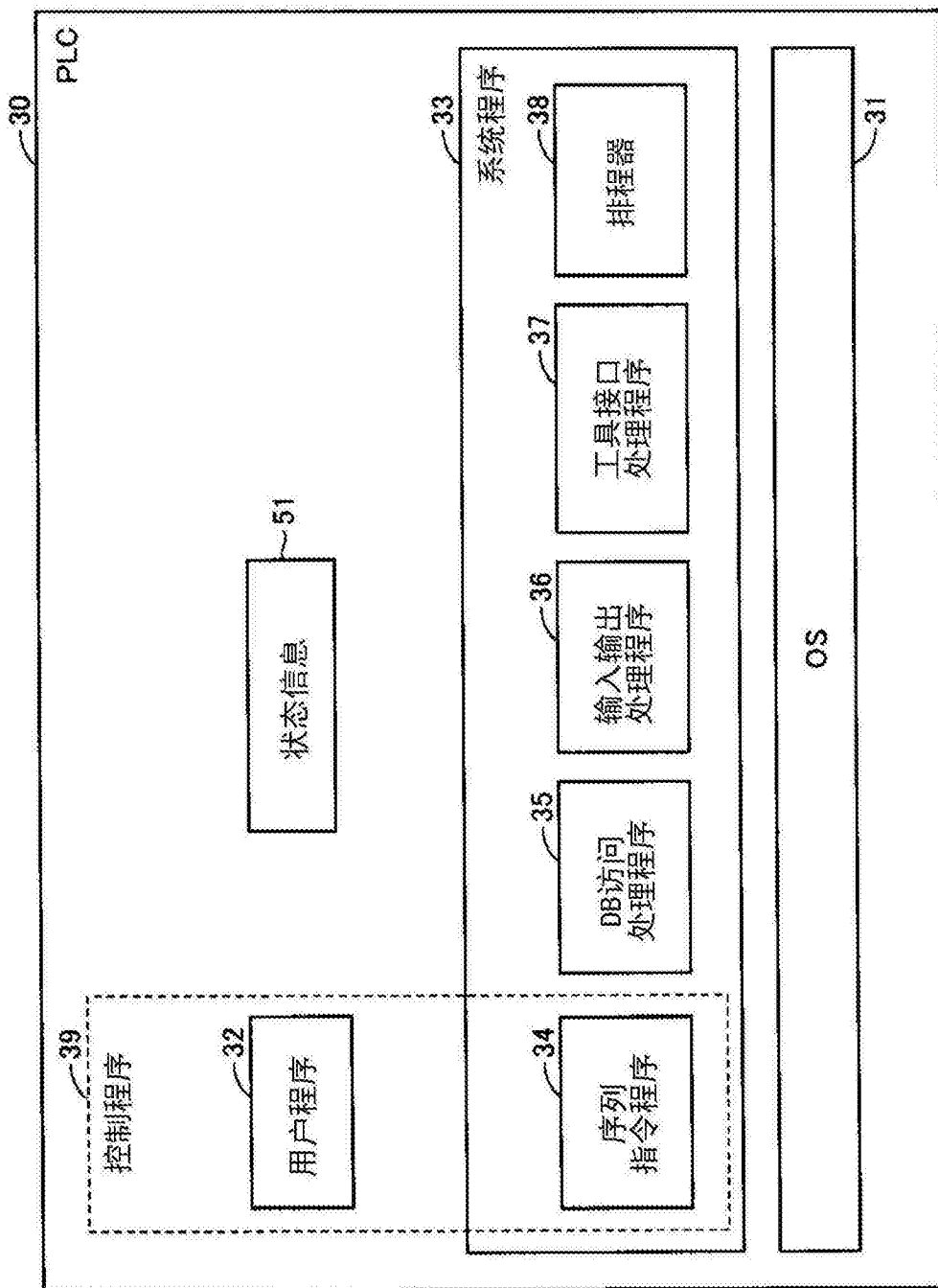


图4

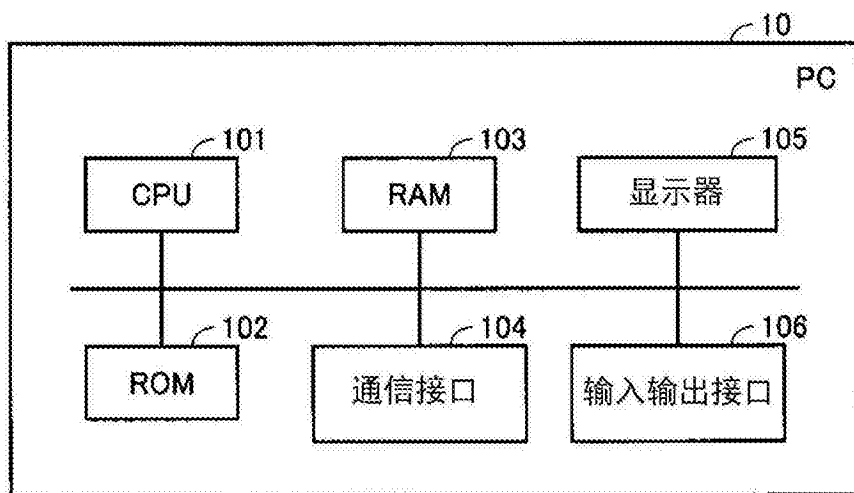


图5A

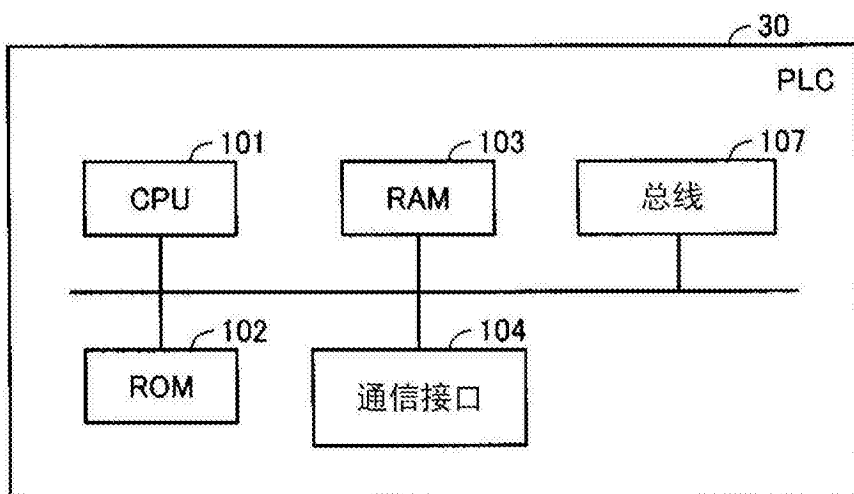


图5B

枚举类型	枚举数
_eDBC_SEND_STATUS	_DBC_SEND_INIT (初始状态)
	_DBC_SEND_UNSENT (SQL语句未发送)
	_DBC_SEND_SENDING (正在发送SQL语句)
	_DBC_SEND_SPOOLED (已导出SQL语句)
	_DBC_SEND_COMPLETE(SQL语句发送结束)

图6

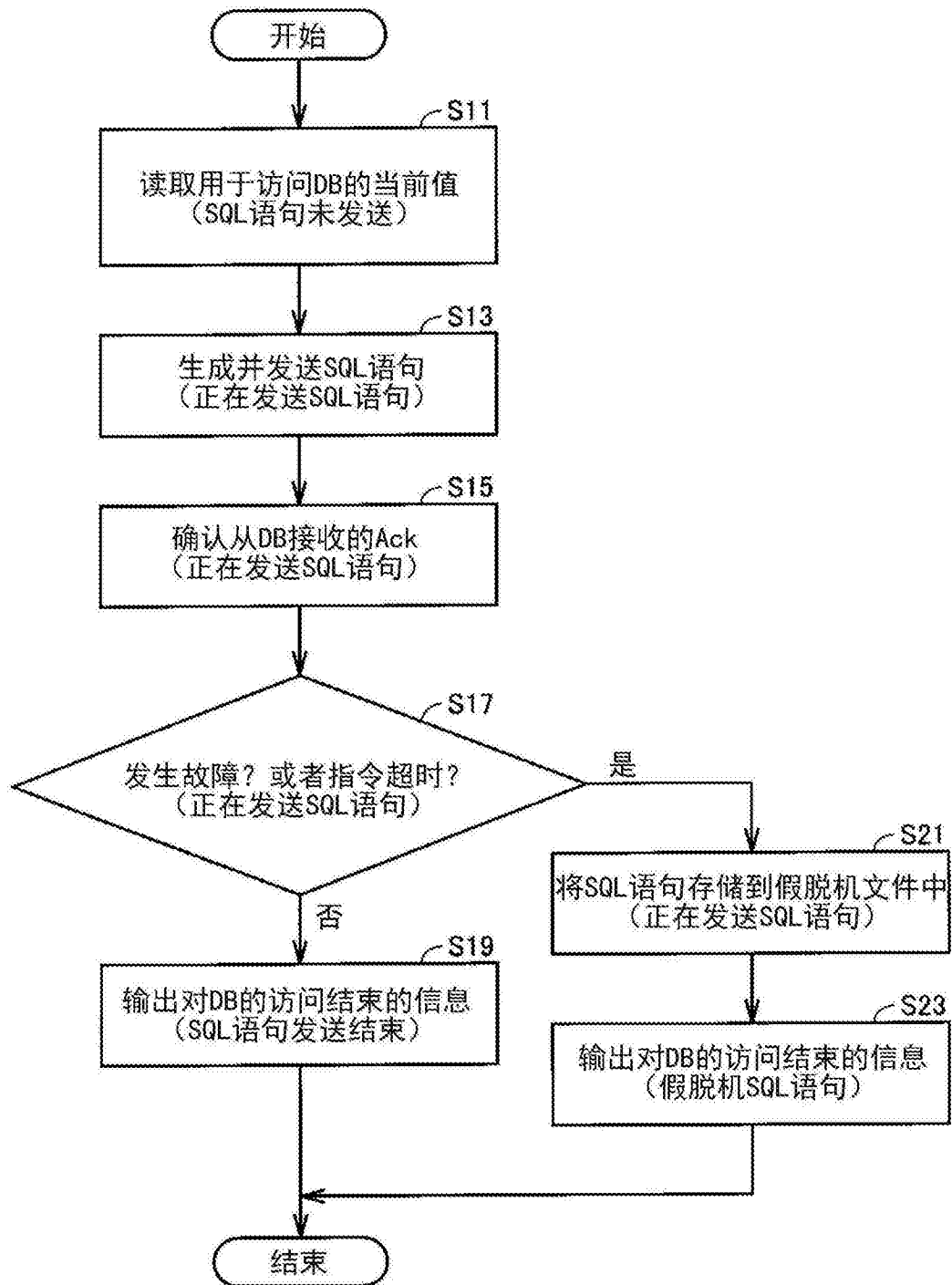


图7

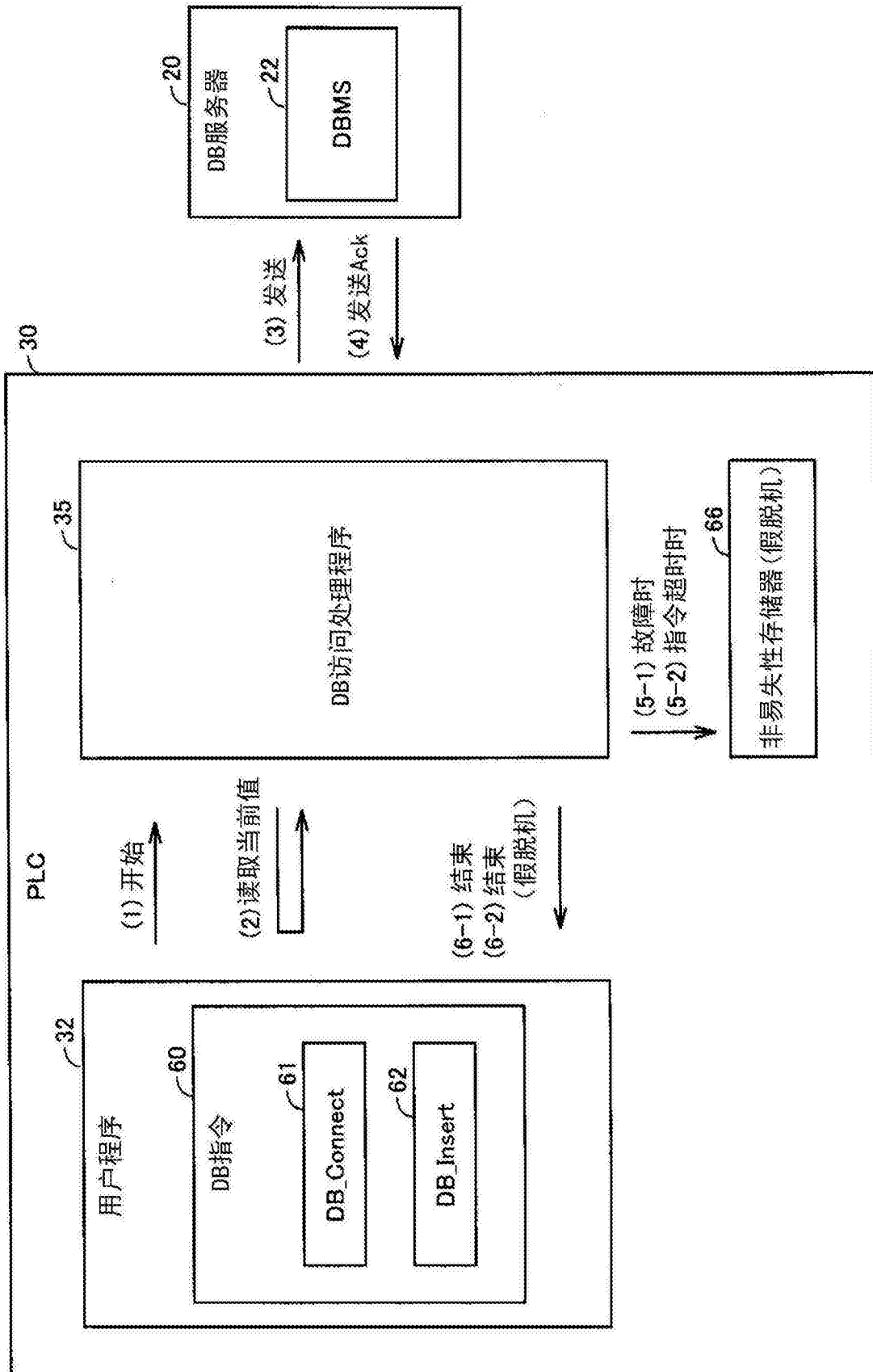


图8

断电时刻	(1) 开始	(2) 读取 当前值	(3) 发送	(4) 发送 Ack	(5-1) 发生故障时 (5-2) 指令超时	(6-1) 结束	(6-2) 结束 (假脱机)
输出变量 State	DBC_SEND_UNSENT		DBC_SEND_SENDING			DBC_SEND_COMPLETE	DBC_SEND_SPOOLED
用户所采取的措施的内容	通过用户程序来再次发送		在确认DB服务器20一侧有无记录之后， 通过用户程序来再次发送			不需要 采取措施	再次发送 假脱机文件

图9

```
VAR RETAIN
    MyMapVar : sMyMap ;
    MyState : _eDBC_SEND_STATUS ;
END VAR
```

图10

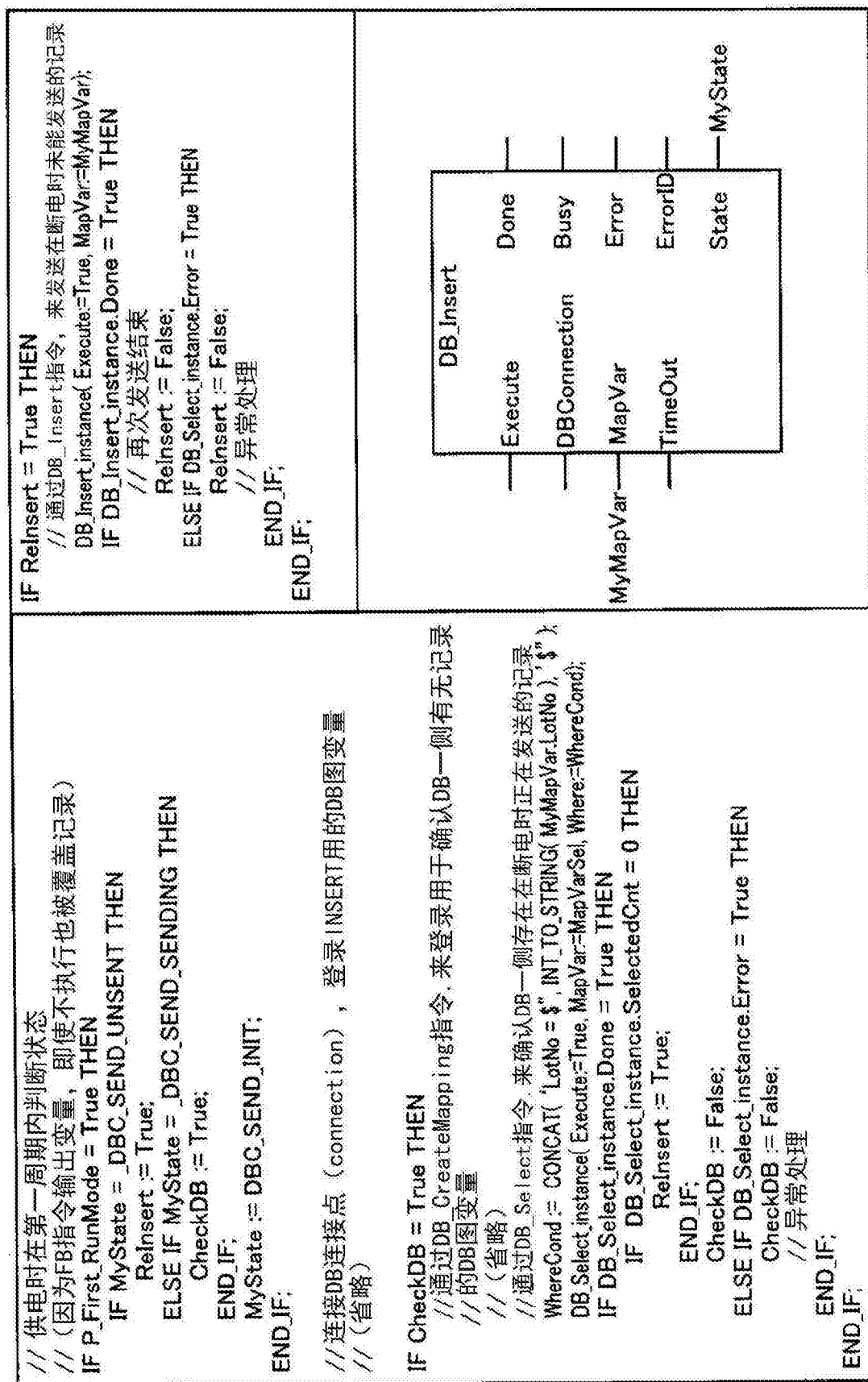


图11