



(12) 发明专利

(10) 授权公告号 CN 101334746 B

(45) 授权公告日 2011. 11. 16

(21) 申请号 200810131805. 4

(22) 申请日 2008. 06. 24

(30) 优先权数据

2007-166383 2007. 06. 25 JP

(73) 专利权人 株式会社日立产机系统

地址 日本东京都

(72) 发明人 柳原德久 木原一 山田勉

苗村万纪子 清野宪二

(74) 专利代理机构 北京银龙知识产权代理有限公司

公司 11243

代理人 许静

(51) Int. Cl.

G06F 11/00 (2006. 01)

审查员 李强

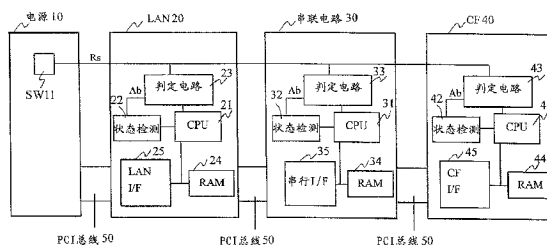
权利要求书 2 页 说明书 6 页 附图 4 页

(54) 发明名称

多组件系统

(57) 摘要

本发明提供一种多组件系统,在该多组件系统中,不会对处于正常状态的组件的 CPU 产生影响,只可以对处于异常状态的组件的 CPU 进行复位。在通过公共总线连接多个至少具有 CPU 的组件而构成的多组件系统中,设置根据开关操作生成复位信号,并将其提供给各个组件的第一复位信号生成单元,并且对于该每个组件设置判定可否进行 CPU 复位的判定单元,判定单元在 CPU 处于正常状态时,禁止基于来自第一复位信号生成单元的复位信号的 CPU 的复位,在 CPU 处于异常状态时,根据来自第一复位信号生成单元的复位信号对 CPU 进行复位。



1. 一种通过公共总线连接多个至少具有 CPU 的组件而构成的多组件系统,其特征在
于,

具备根据开关操作生成复位信号,并将其提供给各个所述组件的第一复位信号生成单
元,以及

对于每个所述组件,具备判定可否进行所述 CPU 的复位的判定单元,

所述判定单元在所述 CPU 处于生成中断信号的正常状态时,禁止基于来自所述第一复
位信号生成单元的所述复位信号的所述 CPU 的复位,

在所述 CPU 处于不生成所述中断信号的异常状态时,根据来自所述第一复位信号生成
单元的所述复位信号对所述 CPU 进行复位。

2. 根据权利要求 1 所述的多组件系统,其特征在在于,

对于每个所述组件,具备检测所述 CPU 的状态的状态检测单元,

所述判定单元根据所述状态检测单元的检测结果,判定可否进行所述 CPU 的复位。

3. 根据权利要求 2 所述的多组件系统,其特征在在于,

所述状态检测单元是通过所述 CPU 生成的所述中断信号被复位,进行从初始值开始的
计数的看门狗定时器,在所述计数溢出时检测出所述 CPU 的状态为异常。

4. 一种通过公共总线连接多个至少具有 CPU 的组件而构成的多组件系统,其特征在
于,

具备根据开关操作生成复位信号,并将其提供给各个所述组件的第一复位信号生成单
元,以及

对于每个所述组件,具备判定可否进行所述 CPU 的复位的判定单元,

所述判定单元在所述 CPU 处于生成中断信号的正常状态时,禁止基于来自所述第一复
位信号生成单元的所述复位信号的所述 CPU 的复位,

在所述 CPU 处于不生成所述中断信号的异常状态时,根据来自所述第一复位信号生成
单元的所述复位信号对所述 CPU 进行复位,

所述组件各自的所述 CPU 在处于所述正常状态时,输出表示为正常状态 的生存信息,
并通过所述公共总线提供给其他的所述组件,并且取得来自其他的所述组件的所述生存信
息,由此判定其他的所述组件中的所述 CPU 是否处于所述正常状态,

具备对于每个所述组件设置的第二复位信号生成单元,其在所述 CPU 将其他所述组件
中的所述 CPU 判定为处于所述异常状态时,生成复位信号并提供给各个所述组件的所述判
定单元,对处于所述异常状态的 CPU 进行复位。

5. 根据权利要求 4 所述的多组件系统,其特征在在于,

对于每个所述组件具备检测所述 CPU 的状态的状态检测单元,

所述判定单元根据所述状态检测单元的检测结果判定可否进行所述 CPU 的复位。

6. 根据权利要求 5 所述的多组件系统,其特征在在于,

所述状态检测单元是通过所述 CPU 生成的所述中断信号被复位,进行从初始值开始的
计数的看门狗定时器,在所述计数溢出时检测出所述 CPU 的状态为异常。

7. 一种通过公共总线连接多个至少具有 CPU 的组件而构成的多组件系统,其特征在
于,

具备根据开关操作生成复位信号,并将其提供给各个所述组件的第一复位信号生成单

元,以及

对于每个所述组件,具备判定可否进行所述 CPU 的复位的判定单元,

所述判定单元在所述 CPU 处于生成中断信号的正常状态时,禁止基于来自所述第一复位信号生成单元的所述复位信号的所述 CPU 的复位,

在所述 CPU 处于不生成所述中断信号的异常状态时,根据来自所述第一复位信号生成单元的所述复位信号对所述 CPU 进行复位,

各个所述组件的所述 CPU 在处于所述正常状态时,输出表示为正常状态的生存信息,并通过所述公共总线提供给其他的所述组件,并且取得来自其他的所述组件的所述生存信息,由此判定其他的所述组件中的所述 CPU 是否处于所述正常状态,

具备对于每个所述组件设置的第二复位信号生成单元,其在所述 CPU 将其他所述组件中的所述 CPU 判定为处于所述异常状态时,生成复位信号并提供给各个所述组件的所述判定单元,对处于所述异常状态的 CPU 进行复位,

所述组件的所述 CPU 各自在处于正常状态时,经由所述公共总线取得所述 CPU 成为异常状态的所述组件中的所述 CPU 的处理结果数据,然后从第二复位信号生成单元生成所述复位信号。

8. 根据权利要求 7 所述的多组件系统,其特征在于,

对于每个所述组件,具备检测所述 CPU 的状态的状态检测单元,

所述判定单元根据所述状态检测单元的检测结果判定可否进行所述 CPU 的复位。

9. 根据权利要求 8 所述的多组件系统,其特征在于,

所述状态检测单元是通过所述 CPU 生成的所述中断信号被复位,进行从初始值开始的计数的看门狗定时器,在所述计数溢出时检测出所述 CPU 的状态为异常。

多组件系统

技术领域

[0001] 本发明涉及一种通过公共总线连接多个 LAN 等至少具有 CPU (Central Processing Unit: 中央处理装置) 的组件而构成的多组件系统, 特别是涉及该组件中的成为异常状态的 CPU 的复位机构。

背景技术

[0002] 目前, 已知通过包含复位线的公共总线连接多个 CPU 单元, 由各个 CPU 单元分担整体的处理的多 CPU 系统, 例如在特开平 5-290008 号公报中提出了在这样的多 CPU 系统中, 根据执行中的应用程序的指定, 使 CPU 单元复位的技术。

[0003] 该技术是由至少包含 CPU 单元的多个子架 (组件) 构成的系统, 在该系统的使用过程中, 在某个子架的 CPU 单元中, 根据执行中的应用程序的指定, 生成系统复位请求, 当具有该请求时, 该 CPU 单元获得复位权, 通过复位线对其他子架的 CPU 单元进行复位。

[0004] 但是, 在特开平 5-290008 号公报公开的系统中, 构成该系统的多个子架, 即组件的 CPU 单元各自分担整体的处理, 所以根据系统复位请求, 所有组件的 CPU 单元被同时复位, 但在由多个组件构成的, 并且各个组件单独进行独立的处理的系统中存在如下问题: 如上述专利文献 1 中记载的系统那样, 根据复位信号所有组件的 CPU 单元被同时复位, 正在正常进行处理的 CPU 单元也被复位, 到此为止的处理数据丢失, 到此为止的处理变为无用。

[0005] 为了解决这样的问题, 例如考虑在某一组件中的 CPU 单元中发生了异常时, 仅对该 CPU 单元进行复位, 但是为此, 在每次某个 CPU 单元成为异常状态时, 需要由操作员找出 CPU 单元成为异常状态的组件, 这对于操作员来说非常费工夫。

[0006] 而且, 还已知在 CPU 中设置 WDT (Watch Dog Timer: 看门狗定时器), 该 WDT 始终监视 CPU 的状态, 当检测到 CPU 成为异常状态时对该 CPU 进行复位的技术, 在由多个具有 CPU 单元的组件构成的系统中, 还考虑对每个组件设置这样的 WDT, 当该组件的 CPU 成为异常状态时, 通过该 WDT 对 CPU 单元进行复位, 由此, 可以只对成为异常状态的 CPU 单元进行复位。

[0007] 然而, 根据该方法, 与操作员的意图无关, 成为异常状态的 CPU 被自动复位。但是在系统中发现异常时, 作为操作员例如有时希望充分确认该异常, 或者根据情况找出成为该异常原因的组件, 采取确保该 CPU 到此为止的处理数据等措施, 但是根据该方法, 无论是否具有这样的操作员要求, 异常状态的 CPU 都被自动地复位。

发明内容

[0008] 本发明的目的在于提供一种多组件系统, 其可以解决这样的问题, 并且能够在操作员所希望的定时仅对成为异常状态的组件的 CPU 进行复位。

[0009] 本发明的另一目的还在于提供一种能够在不丢失处理数据的定时, 仅对成为异常状态的组件的 CPU 进行复位的多组件系统。

[0010] 为了达到上述目的, 本发明是一种通过公共总线连接多个至少具有 CPU 的组件而构成的多组件系统, 其特征为: 设置根据开关操作生成复位信号, 并将其提供给各个组件的

第一复位信号生成单元,并且对于每个组件设置判定可否进行该 CPU 复位的判定单元,该判定单元在 CPU 处于正常状态时,禁止基于来自第一复位信号生成单元的复位信号的 CPU 的复位,在 CPU 处于异常状态时,根据来自第一复位信号生成单元的复位信号对 CPU 进行复位。

[0011] 另外,本发明的特征为:组件各自的 CPU 在处于正常状态时,输出生存信息,通过公共总线提供给其他组件,同时取得来自其他组件的生存信息,由此判定其他组件中的 CPU 是否处于正常状态,对每个组件设置第二复位信号生成单元,其在 CPU 将其他组件中的 CPU 判定为处于异常状态时,生成复位信号然后提供给其他组件的判定单元。

[0012] 另外,本发明的特征为:各个组件的 CPU 在处于正常状态时,经由公共总线取得 CPU 成为异常状态的组件中的 CPU 的处理结果数据,然后从第二复位信号生成单元生成复位信号。

[0013] 另外,本发明的特征为:对于每个组件设置检测 CPU 的状态的状态检测单元,判定单元根据状态检测单元的检测结果判定可否进行 CPU 的复位。

[0014] 另外,本发明的特征为:状态检测单元是根据由 CPU 按照预定的周期反复进行清除,然后该反复清除消失,来检测 CPU 的状态为异常的看门狗定时器。

[0015] 根据本发明,可以在操作员需要的定时仅对成为异常状态的组件的 CPU 进行复位,并且可以使处于正常状态的组件的 CPU 原样不变地继续进行处理。

[0016] 另外,根据本发明,对于 CPU 成为异常状态的组件,正常组件的 CPU 可以访问,在进行了用于确保 CPU 成为异常状态的组件中的到目前为止的处理结果的处理之后,能够对该成为异常状态的 CPU 自动地进行复位,即使进行该复位,也不会使之前的处理变得无用。

附图说明

[0017] 通过参照以下的附图进行说明,本发明的这些以及其他的特征、目标以及优点将会变得更加明确,其中:

[0018] 图 1 是表示本发明的多组件系统的第一实施方式的整体立体图。

[0019] 图 2 是表示图 1 中的组件的主要部分的具体结构的电路图。

[0020] 图 3 是表示图 1 中的判定电路的一个具体例子的电路图。

[0021] 图 4 是表示本发明的多组件系统的第二实施方式的整体立体图。

[0022] 图 5 是表示图 4 中的组件的主要部分的具体结构的电路图。

具体实施方式

[0023] 虽然我们示出并描述了几个基于本发明的实施例,但本领域的技术人员应当可以理解,在不脱离本发明范围的情况下这些公开的实施例可以进行变形和修改。因此,本发明并不限于我们所示出并描述的细节,而是覆盖了全部的包含在后附的权利要求的保护范围内的变形以及修改。

[0024] 下面,使用附图对本发明的实施方式进行说明。

[0025] 此外,在以下的实施方式中,以用于机械等设备控制的 PLC(可编程逻辑控制器)为例进行说明,但是本发明并不限于此,可适用于由多个至少具有 CPU 的组件构成的任意系统。

[0026] 图 1 是表示本发明的多组件系统的第一实施方式的主要部分的概要结构图,10 是电源;11 是复位开关;20 是 LAN(Local Area Network);30 是串联电路(serial);40 是 CF(Compact Flash 紧凑式快闪(注册商标));21、31、41 是 CPU;22、32、42 是状态检测单元;23、33、43 是判定电路;24、34、44 是存储器;25 是 LAN I/F(接口);35 是串行 I/F;45 是 CF I/F;50 是 PCI(Peripheral Components Interconnect:外围组件互连)总线。

[0027] 在该图中,该第一实施方式由 LAN20、串联电路 30、CF40 等组件以及对它们提供电源电压的电源 10 构成,这些组件 20、30、40 通过作为公共总线的 PCI 总线 50 相互连接。

[0028] LAN20 具有 CPU21、存储 CPU21 执行的程序和 CPU21 运算的数据的存储器 24 以及 LAN I/F25,而且,设置了用于对 CPU21 进行复位的状态检测单元 22 和判定电路 23,是通过 LAN I/F25 与因特网进行通信的组件。另外,串联电路 30 同样具有 CPU31、存储 CPU31 执行的程序和 CPU31 运算的数据的存储器 34 以及串行 I/F35,而且,还设置了用于对 CPU31 进行复位的状态检测单元 32 和判定电路 33,是通过串行 I/F35 根据 RS232C 等标准与外围设备进行通信的组件。而且,CF40 同样具有 CPU41、存储 CPU41 执行的程序和 CPU41 运算的数据的存储器 44 以及 CF I/F45,而且,还设置了用于对 CPU41 进行复位的状态检测单元 42 和判定电路 43,是通过 CF I/F45 与 CF 卡等记录介质进行通信的组件。如此,这些组件 20、30、40 不是分担一个整体处理,而是分别单独进行独立的运算处理。

[0029] 电源 10 中设置有未图示的电源开关等各种操作开关,同时设置有复位开关 11,当该复位开关 11 被操作员操作时,通过未图示的复位信号生成单元生成高电平的复位信号 Rs,并提供给各组件(LAN20、串联电路 30、CF40)。

[0030] 图 2 是表示作为图 1 中的组件之一的 LAN20 的主要部分的具体结构的电路图,对于与图 1 对应的部分标注相同的符号。

[0031] 在该图中,CPU21、RAM24、LAN I/F25 通过 PCI 总线 50 相连接。CPU21 通过 PCI 总线 50 在 RAM24 中进行数据读出或数据写入来进行运算处理,同时进行 LAN I/F25 等各部的控制。另外,在 CPU21 上连接有状态检测单元 22 和判定电路 23,由此,如后所述,在 CPU21 成为异常状态时,根据在电源 10 中通过复位开关 11 的操作而提供的复位信号 Rs,对该 CPU21 进行复位。

[0032] 该结构在图 1 中的串联电路 30、CF40 中也相同,PCI 总线 50 在这些 LAN20、串联电路 30、CF40 不同的组件之间相互连接。

[0033] 在 LAN20 中,状态检测单元 22 例如是 WDT(Watch Dog Timer:看门狗定时器)。WDT 由根据来自 CPU21 的中断信号 Ir 进行复位的计数器构成。CPU21 在处于正常状态时以一定的周期产生该中断信号 Ir,在每次提供该中断信号 Ir 时,WDT 被复位反复从初始值开始进行计数。当 CPU21 成为异常状态时,由于该 CPU21 无法提供中断信号 Ir,因此 WDT 继续计数,当溢出时,产生异常信号 Ab,并将其提供给判定电路 23。

[0034] 当没有从异常状态检测单元 22 提供异常信号 Ab 时,判定电路 23 认为 CPU21 处于正常状态,判定为不能将该 CPU21 复位。当从异常状态检测单元 22 提供了异常信号 Ab 时,认为 CPU21 处于异常状态,判定为可以将该 CPU21 复位,当提供了复位信号 Rs 时,对该 CPU21 进行复位。

[0035] 图 3 是表示判定电路 23 的一具体例的电路结构图,23a 是 S/R MM(置位/复位多谐振荡器),23b 是与门。

[0036] 在该图中, S/R MM23a 通过操作电源 10(图 1)的复位开关 11(图 1)而提供的高电平的复位信号 Rs 的终端沿(下降沿)被复位,并且通过来自状态检测单元 22(图 1)的异常信号 Ab 的开端沿被复位,当被复位时,输出高电平的 Q 输出。该 Q 输出成为与门 23b 一方的输入,复位信号 Rs 作为另一方输入被提供给与门。当根据来自状态检测单元 22 的异常信号 Ab, S/RMM23a 处于置位状态时,操作复位开关 11 而产生的复位信号 Rs 通过与门 23b。该复位信号 Rs 被提供给 CPU21(图 1、图 2),对该 CPU21 进行复位。

[0037] 然后,使用图 1 对该第一实施方式的动作进行说明。

[0038] 在 LAN20、串联电路 30、CF40 等组件的 CPU21、31、41 正常动作时,该系统未图示的、例如控制对象的设备正常进行动作,但是当这些组件的某个 CPU 成为异常状态时,控制对象设备的动作受到该 CPU 的异常状态的影响。由此,操作员可以识别出系统的某个组件发生了异常,当操作员识别出该异常并对复位开关 11 进行了操作时,产生复位信号 Rs 并提供给 LAN20、串联电路 30、CF40。

[0039] 另一方面,在 CPU 成为异常状态的组件中,当假设该组件例如为 LAN20 时,如上所述,状态检测单元 22 检测到 CPU21 成为异常状态向判定电路 23 提供异常信号 Ab,判定电路 23 认为 CPU21 处于异常状态,判定为可以对该 CPU21 进行复位。在该状态下从电源 10 提供了复位信号 Rs 时,该判定电路 23 将被提供的复位信号 Rs 提供给 CPU21。由此,该 CPU21 被复位。

[0040] 在此时作为 CPU 处于正常状态的组件的串联电路 30 和 CF40 中,由于状态检测单元 32、42 分别检测到 CPU31、41 的正常状态,所以不产生异常信号 Ab,判定电路 33、43 分别认为 CPU31、41 处于正常状态,判定为不能将它们复位。因此,即使分别向 CPU31、41 提供复位信号 Rs, CPU31、41 也不会被复位,继续进行运算处理动作。

[0041] 如此,即使在某个组件的 CPU 成为异常状态从而在系统中发生了异常时,即使操作员没有识别或者无法识别出在哪个组件中发生了异常,也可以在识别出系统成为异常时仅仅通过操作复位开关 11,由此切实地只对成为异常状态的组件的 CPU 进行复位,可以避免将正常动作的组件的 CPU 也复位的不理想的状况。

[0042] 而且,可以进行如下的处理:根据操作员对复位开关 11 的操作,即,因为是按照操作员的要求对成为异常状态的组件的 CPU 进行复位,因此该 CPU 的复位定时也可以由操作员来决定,例如,在对控制对象设备进行了规定的处理之后,或者如果可以确保成为异常状态的 CPU 的到目前为止的处理数据,则在确保了处理数据之后,操作复位开关 11 对成为异常状态的组件的 CPU 进行复位。这样,在可以确保处理数据的情况下,在启动复位后的 CPU 时,可以利用该确保的数据,成为异常状态之前的处理不会变得无用。

[0043] 图 4 是表示本发明的多组件系统的第二实施方式的结构框图,26、36、46 是复位信号生成电路,对于与图 1 对应的部分标注相同的符号并省略重复的说明。

[0044] 在该图中,该第二实施方式在图 1 所示的第一实施方式中,在各组件,即 LAN20、串联电路 30、CF40 中分别设置复位信号生成电路 26、36、46,由此生成与操作复位开关 11 时的复位信号 Rs 相同格式的复位信号 Rs'。生成复位信号 Rs' 的组件是在 LAN20、串联电路 30、CF40 中的某一个组件的 CPU 成为异常时 CPU 处于正常状态的组件,该复位信号 Rs' 从判定电路被读入到 CPU 处于异常状态的组件中,对该处于异常状态的 CPU 进行复位。

[0045] 这里,如图 5 所示,CPU21、RAM24、LAN I/F25 被连接到 PCI 总线上,CPU21 通过 PCI

总线 50 在 RAM24 中进行数据读出、数据写入来进行运算处理,并且进行 LAN I/F25 等各部的控制。另外,在 CPU21 上连接有状态检测单元 22、判定电路 23 以及复位信号生成电路 26,由此,如先前的第一实施方式相同,在 CPU21 成为异常状态时,根据在电源 10 中操作复位开关 11 而提供的复位信号 Rs,该 CPU21 被复位,同时可以从复位信号生成电路 26 产生与该复位信号 Rs 相同格式的复位信号 Rs'。另外,在该第二实施方式中,CPU21 可以通过 PCI 总线 50 与其他的组件,即此时为串联电路 30、CF40 进行通信。

[0046] 该结构在图 1 中的串联电路 30、CF40 中也相同,PCI 总线 50 在这些 LAN20、串联电路 30、CF40 不同的组件之间相互连接,在这些 LAN20、串联电路 30、CF30 中进行后述的必要数据的通信。

[0047] 在各组件的 CPU21、31、41 分别处于正常状态时,通过 PCI 总线 50 将表示处于正常状态的生存信号按照一定的时间间隔发送给其他的组件。因此,CPU21、31、41 在分别处于正常状态时,始终识别其他组件的 CPU 是否处于正常状态。

[0048] 因此,如果当前串联电路 30 的 CPU31 成为异常状态时,与第一实施方式相同,成为判定电路 33 判定为可以对 CPU31 进行复位的状态,与此同时,CPU31 成为不对 PCI 总线 50 输出生存信息的状态。因此,在 LAN20 和 CF40 中,成为 CPU21、41 无法从串联电路 30 取得生存信号的状态。由此,这些 CPU21、41 识别出串联电路 30 的 CPU31 成为异常状态。

[0049] 因此,CPU21、41 中的某一方,例如 CPU21 对串联电路 30 进行规定的处理,在该处理结束后,控制复位信号生成电路 26 使其产生复位信号 Rs'。该复位信号 Rs' 被提供给串联电路 30 和 CF40,在串联电路 30 中,其判定电路 33 判定为可以对 CPU31 进行复位,并且使与门 23b(图 3)成为 ON 状态(即,判定电路 33 处于 ON 状态),所以,在提供了在 LAN20 的复位信号生成电路 26 中产生的复位信号 Rs' 时,根据该复位信号 Rs' 对判定电路 33 进行复位。

[0050] 此外,在复位信号生成电路 26 产生复位信号 Rs' 之前 CPU21 对串联电路 30 进行的上述规定的处理是指:例如,CPU21 当检测到 CPU31 成为异常状态时,通过 PCI 总线 50 读入串联电路 30 的 RAM34 中的处理程序和数据,存储在 RAM24 中的处理,由此,可以确保直到成为异常状态为止的在 CPU31 中进行的处理的结果(处理数据),而且,还可以通过其他的 CPU 或者在该 CPU31 启动时,继续进行从 CPU31 成为异常状态前不久开始的处理。由此,不会使到目前为止在 CPU31 中进行的处理结果无用,另外,还可以通过 PCI 总线 50 控制串行 I/F35,可以继续执行串联电路 30 的处理。

[0051] 另外,在 CF40 正常动作的情况下,成为其判定电路 43 认为 CPU41 处于正常状态从而判定为不能复位的状态。因此,在 CF40 中,由 LAN20 的复位信号生成电路 26 产生的复位信号 Rs' 被该判定电路 43 阻止,由此,CPU40 不会被复位。

[0052] 而且,在某个组件的 CPU 成为异常状态时,其余的所有组件的 CPU 可以从复位信号生成电路产生复位信号 Rs',但还可以对组件设定优先顺序,当某个组件的 CPU 成为异常状态时,剩余组件中的顺序最上位的组件的 CPU 从复位信号生成电路产生复位信号 Rs'。

[0053] 另外,在 PCI 总线 50 状况不良无法进行通信时,即使各个 CPU21、31、41 向该 PCI 总线 50 输出生存信息,但其中任何一个生存信息都不会被发送到其他组件的 CPU。因此,在图 4 中,例如在串联电路 30 和 CF40 之间 PCI 总线状况不良时,LAN20 的 CPU21 无法接收来自 CF40 的生存信息,认为该 CF40 的 CPU41 处于异常状态,从复位信号生成电路 26 产生

复位信号 Rs' 。该复位信号被提供给 CF40, 但该 CF40 中的 CPU41 正在输出生存信号处于正常状态, 因此, 判定电路 43 判定为不能对 CPU41 进行复位, 即使从 LAN20 提供了复位信号 Rs' , 也被判定电路 43 拒绝, CPU41 不会被复位。

[0054] 这样, 即使在 PCI50 中发生不良状况, 无法传递生存信息, 处于正常状态的 CPU 也不会被复位, 继续原样地进行处理。此外, 此时, 持续输出复位信号 Rs' , 由此可以容易地检测到 PCI 总线 50 的异常。

[0055] 如上所述, 在该第二实施方式中, 可以得到与通过操作复位开关 11 的第一实施方式相同的效果, 而且, 即使操作员不操作开关, 也可对于成为异常状态的组件的 CPU, 在不损失通过该 CPU 到目前为止的处理所得到的数据的状况下, 自动使该 CPU 复位。

[0056] 而且, 在以上各实施方式中, 状态检测单元 22、32、42 检测 CPU21、31、41 硬件方面的异常状态, 但是, 也可以设置例如作为程序中存在的不足, 对软件方向的异常进行检测的单元, 将其检测结果也提供给判定电路 23、33、43, 在成为 CPU21、31、41 软件方面的异常状态时, 同样进行复位。

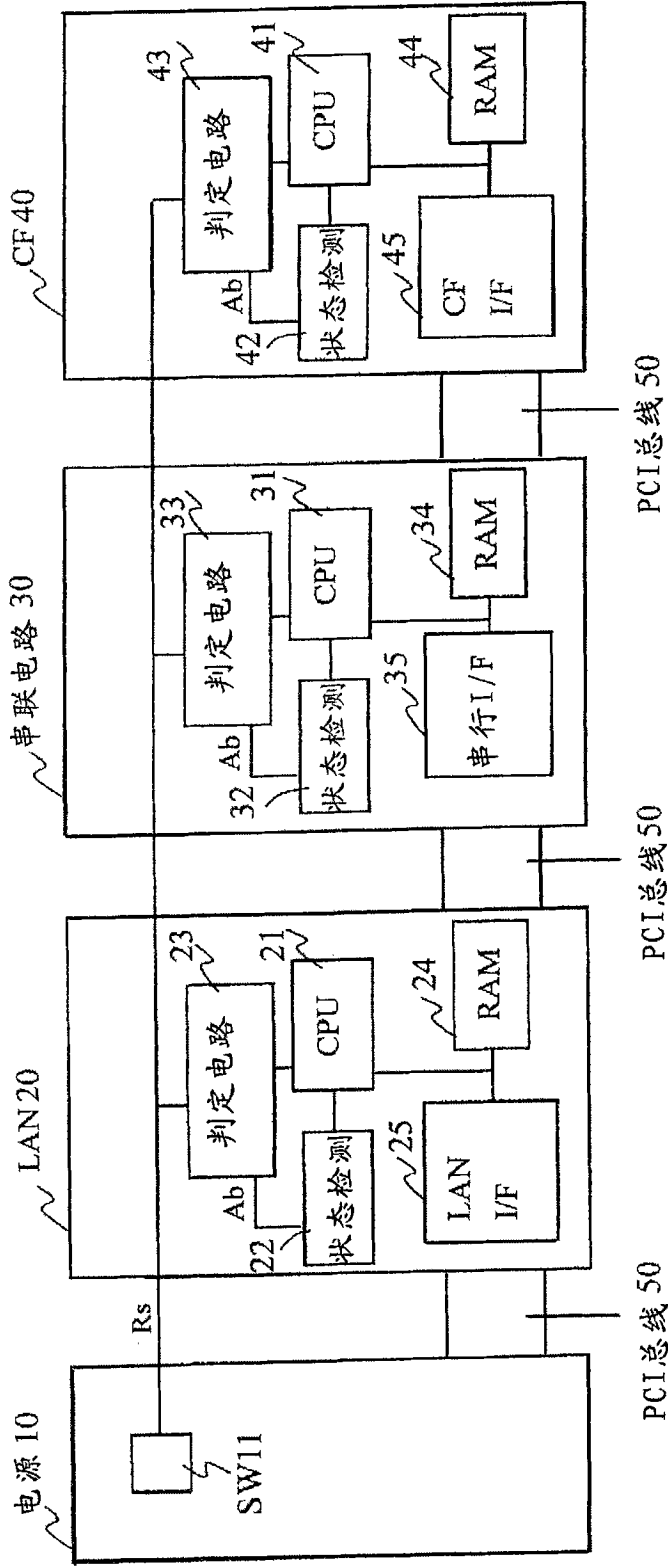


图 1

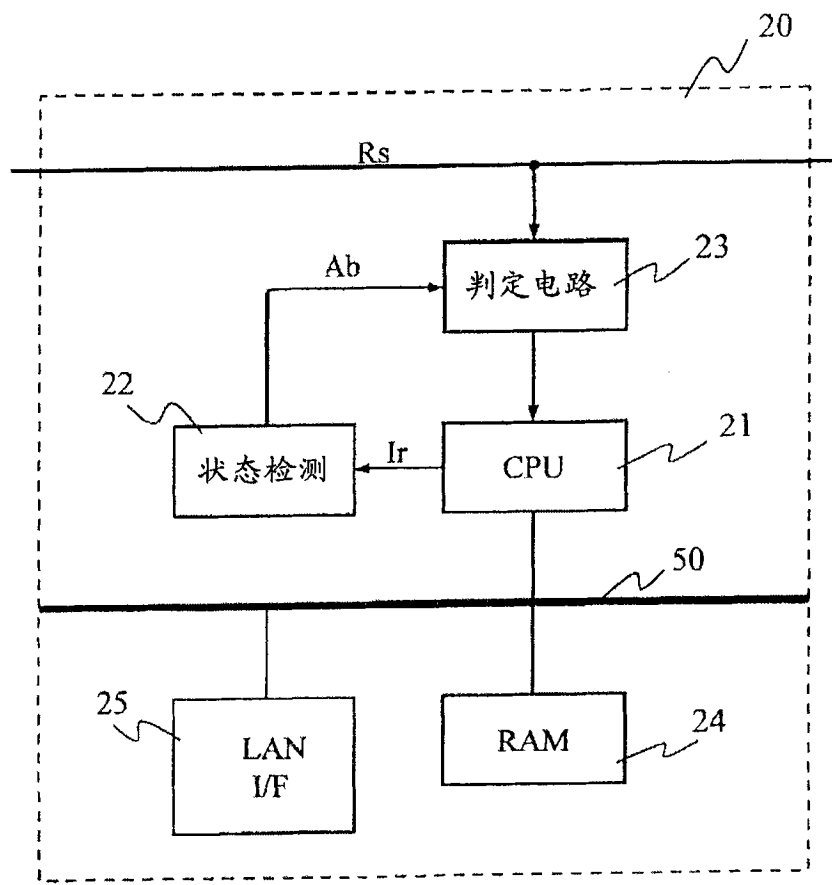


图 2

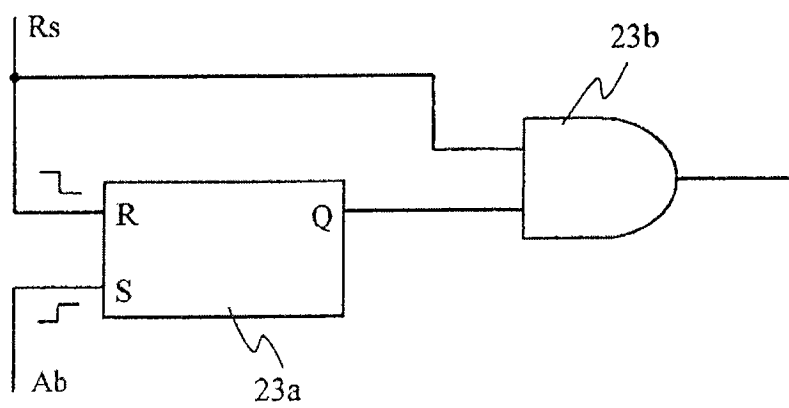


图 3

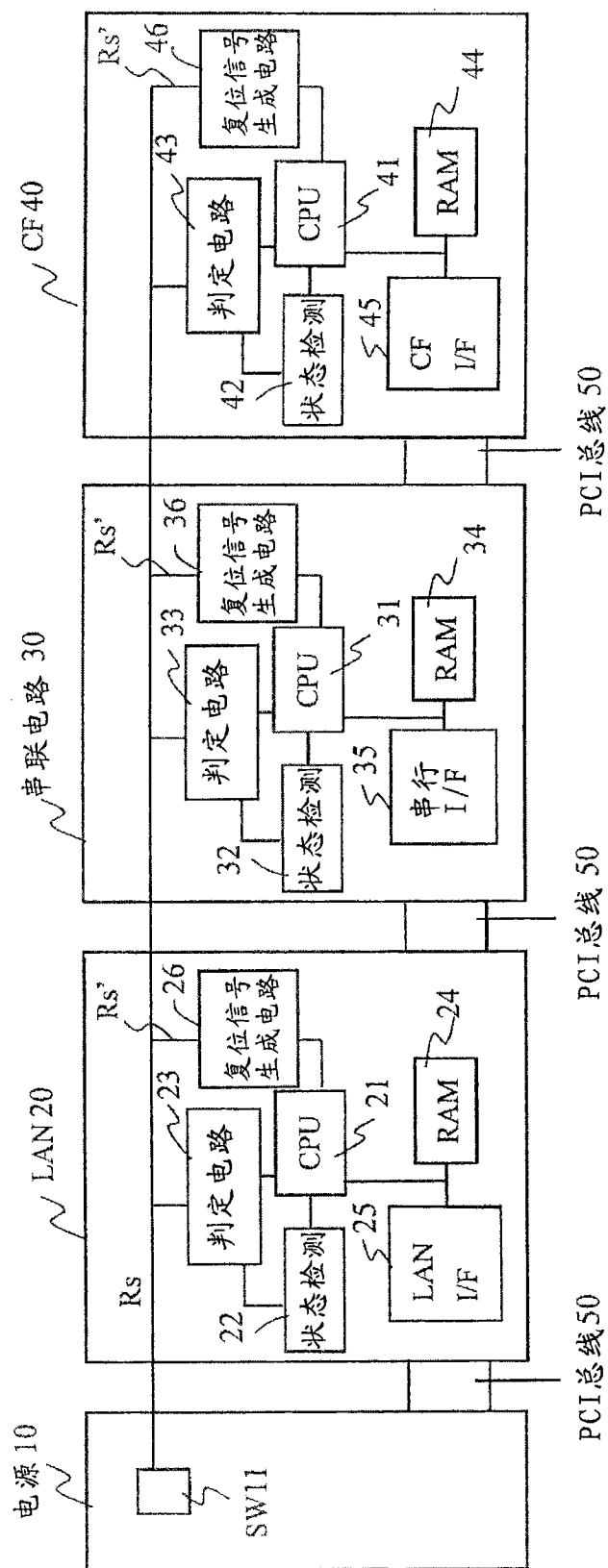


图 4

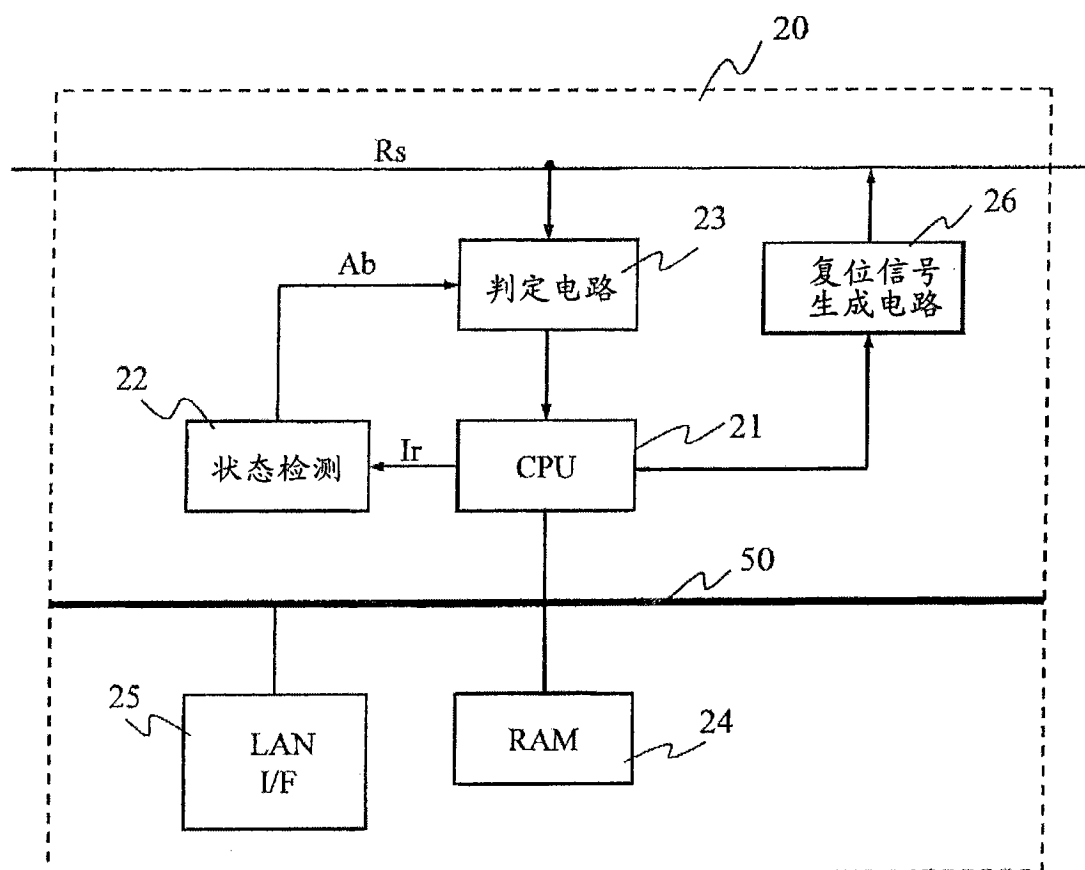


图 5