[19]中华人民共和国专利局

[51]Int.Cl[6]
G11B 11/18

[12] 发明专利申请公开说明书

[21] 申请号 96105671.1

[43]公开日 1997 年 1 月 8 日

[11] 公开号 CN 1139802A

[22]申请日 96.5.9

[30]优先权

[32]95.5.9 [33]JP[31]110505/95

[71]申请人 株式会社日立制作所

地址 日本东京

[72]发明人 永次由英 能見诚

[74]专利代理机构 上海专利商标事务所

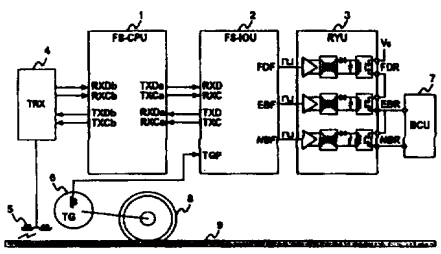
代理人 孙敬国

权利要求书 2 页 说明书 12 页 附图页数 12 页

[54]发明名称 提供多个检验码序列的差错检验方法和实施该方法的系统

[57]摘要

为了通过采用普通的非防障理论而不采用用作检验理论电路的特殊理论来实现包括检验电路自身故障检测的防障数据检验，用两个不同的检验码产生装置将检验码加至计算处理结果，而具有检验码的两个计算处理结果被交替输入到与检验码相应的两个检验装置中，从而使被检验结果交替变化。



权 利 要 求 书

1. 一种差错检验方法，其特征在于，它包含下述步骤：

用不同的方法对同样的数据进行具有相同处理内容的计算处理，

分别对所述两个计算处理结果，产生两个独立的检验码序列，以及

在所述序列之间交换了所述检验码序列以后，按照各序列的相应方法，检验计算处理结果。

2. 一种差错检验系统，其特征在于，它包含：

两个计算处理装置，用来对同样的数据进行相同内容的处理，

两个检验码产生装置，用来分别对所述两个计算处理装置的两个计算处理结果独立地产生两个检验码，以及

两个与所述检验码产生装置对应的检验装置，计算处理结果是在序列之间交换所述检验码序列之后通过提供所述检验装置来检验的。

3. 如权利要求 2 所述的差错检验系统，其特征在于，

所述两个检验码产生装置对两个计算处理结果产生互不相同的检验码，

所述检验装置分别按照所述检验码进行检验，以及

在每一计算循环的两序列之间交换了检验码以后，把两个计算处理结果提供至两个检验装置。

4. 一种差错检验系统，其特征在于，它包含：

两个计算处理装置，用来对同样的数据进行相同内容的处理，

对于计算处理结果的检验码产生装置，

与所述检验码对应的检验装置，

产生两个不同检验码的两个检验码产生装置，以及

与所述检验码对应的两个检验装置，在每一计算循环内把加入了由所述检验码产生装置之一所产生的一个检验码的计算处理结果以及加入了由另一个检验码产生装置所产生的另一个检验码的计算处理结果，交替地提供至所述两个检验装置。

5. 一种差错检验系统，其特征在于，它包含：

两个计算处理装置，用来对同样的数据进行相同内容的处理，

两个检验码产生装置，用来分别对所述两个计算处理装置的两个计算处理结果产生两个检验码，

第一计算处理器，它具有用来在两序列之间进行所述检验码交换的检验码交换装置，以及

具有与所述检验码对应的检验装置的第二计算处理器，在所述第一计算处理器用作发射机而所述第二计算处理器用作接收机的条件下，通过进行通信来检测第一计算处理器中的处理差错和通信差错。

说　明　书

提供多个检验码序列的
差错检验方法和实施该方法的系统

本发明涉及在采用微型计算机的控制器中使用的差错检验方法及其系统，尤其涉及一种差错检验方法和系统，这种系统和方法适用于需要防障能力的控制器，如用于轨道动车(rail car)的控制器。

在轨道动车控制中对有差错的数据进行差错控制时，有可能会引起威胁人们生命安全的事故。所以，要求用作轨道动车的控制器具有防障功能，并且要求确保输入—输出数据和计算处理的正确性。

至于数据传输过程中确保输入—输出数据的正确性的装置，有一种广泛采用的检测传输差错的方法，其中，在发射机侧将检验码加入到传输数据，而在接收机侧检验检验码和传输数据。至于检验码，通常采用奇偶检验位或CRC(循环冗余检验)。

另外，至于确保处理本身正确性的装置，大多数要求具有防障功能的系统采用这样一种方法，其中的处理系统是用多个系统构成的，并通过检验多个系统的处理结果以及在被检验结果中找出不一致性来检测处理过程中的异常情况。

至于上述例子中的数据检验方法，有以总线检验方法为代表的硬件逼近方法和日本专利申请公开号为60—233734(1985)的专利文献中揭示的软件逼近方法，在这种方法中，通过检验加入到每一序列中的被处理结果的CRC码来检测处理过程中的不一致性。

在这样一种通常技术中，由于检验手段自身出现的异常性，有时检验结果始终固定在“无差错”上。因此，这不可能是一种完善的防障功能。所以，在某一双系统中进行检验时，特别使用一种能够采用特

殊的防障理论来进行故障(包括检验电路自身故障)检测的方法。然而,由于这种理论很特别,电路变得规模较大而且复杂,因而会出现可靠性和处理速度的问题。

另一方面,当采用软件进行检验时,可以使系统简单,且处理速度加快。然而却存在软件失控下的防障能力的问题。

在数据差错不是通过构筑多个系统而是通过检验数据和检验码来检测的数据传输中,是不考虑检验器自身的故障的。

本发明的目的是采用普通的非防障理论而不采用用于检验理论电路的特殊理论,来实现包括检验电路自身的故障检测在内的防障数据检验。

为了实现上述目的,按照本发明,用两个不同的检验码产生装置,将检验码加入到计算处理结果,并且将具有检验码的两个计算处理结果交替地输入到与该检验码对应的两个检验装置中去。因此,被检验结果是交替变化的。

上述结构中,数据正常时,被检验结果是一个以"无差错"和"差错"交替出现的交替信号。当某一被检验结果是"无差错"时,另一被检验结果就是"差错"。

数据异常时,两个检验装置的被检验结果均为"差错",且不是交替信号。当检验装置本身为异常时,被判断结果固定在"无差错"或"差错"上。因此两个被检验结果中的至少有一个结果停止交替变化。从而可以检测处理过程中的异常性或检验装置中的异常性。

图 1 是按照本发明的轨道动车控制器的结构实施例方框图。

图 2 是数据处理器(FS—CPU)和速度控制器(FS—IOU)的结构实施例方框图。

图 3 是按照本发明的轨道动车功能结构实施例的方框图。

图 4 是描述数据处理器(FS—CPU)详细结构的方框图。

图 5 是按照本发明的数据处理器(FS—CPU)中的处理流程图。

图 6 是按照本发明的 CRC 码交换过程的示意图。

图 7 是按照本发明的速度控制器(FS－IOU)的详细结构方框图。

图 8 是按照本发明的速度控制器(FS-IOU)中的处理流程图。

图 9 是按照本发明的数据检验电路的结构图。

图 10 是按照本发明的输出寄存器运行的时序图。

图 11 是按照本发明，当出现处理过程中的不一致或数据传输中的差错时，交替“与”电路运行的时序图。

图 12 是按照本发明，在检验程序中出现异常时，交替“与”电路运行的时序图。

图 13 是按照本发明，当来自数据处理器(FS－CPU)的输出数据固定时，交替“与”电路运行的时序图。

下面参见附图详细描述本发明的实施例。

图 1 是装在轨道动车上的轨道动车控制器的结构方框图，按照本发明的数据差错检验系统施加在该轨道动车上。图 2 是数据处理器(FS－CPU)1 和速度控制器(FS－IOU)2 的内部结构方框图。图 3 是本实施例轨道动车控制器的功能结构方框图。

数据处理器(FS－CPU)1 在其输入端 RXDb、RXCb 中接收来自地面的有关前一辆轨道动车的位置、在轨道动车前的一段内的极限速度的信息，并将极限速度信息从数据处理输出端口 TXDa、TX-Ca 输出到速度控制器(FS－IOU)2 的输入端口 RXD、RXC。

有关前一辆轨道动车的位置信息以及在轨道动车前一段内的极限速度等数据被传送到数据处理器 1 的输入端口 RXDb，而传输同步时钟信号被传送到输入端口 RXCb。另外，极限速度被传送到速度控制器的输入端口 RXD，传输时钟信号被传送到输入端口 RXC。

继电器单元(RYU)3 使与速度控制器 2 输出的故障检测信号 FDF、紧急刹车信号 EBF 以及正常刹车信号 NBF 对应的故障检测

继电器FDR、紧急刹车继电器EBR以及正常刹车继电器NBR。

信号通信发射机/接收机(TRX)4通过一接收天线5将从地面发送的信息输入到数据控制器1内。转速传感器(TG)6将有关轮子转速的信号输入到速度控制器2的输入端口TGP，以计算轨道动车的运行距离和速度。刹车控制单元(BCU)7根据紧急刹车继电器EBR或正常刹车继电器NBR的输出，用与轨道9之间的摩擦力刹住轮子8，以停住轨道动车。

前一辆轨道动车的位置信息(通常铁路电路)从安装在地面上的信号通信系统(未图示)通过轨道9定期地发送出来，由接收天线5和发射机/接收机4接收数据，输入到数据控制器1的输入端口RXDb和RXCb。有关前一辆轨道动车位置信息、在轨道动车前的一段极限速度等数据被发送到输入端口RXDb，而传输同步时钟信号被发送到输入端口RXCb。

另一方面，由安装在轮子8的轮轴上的转速传感器检测的距离脉冲被输入到速度控制器2的输入端口TGP。脉冲由转速传感器输入轮子直径校正单元来校正，并随后在距离计算单元中累加，从而计算轨道动车的自身当前位置，如图3所示。

轨道动车自身的位置信息被传送到极限速度内插单元，并通过输出端口TXD和TXC还传送到数据处理器(FS—CPU)1的输入端口RXDa和RXCa。数据输入到输入端口RXAa，而传输同步时钟信号输入到输入端口RXCa。有关位置的信息通过输出端口TXDb发送到下一辆轨道车，传输同步时钟信号通过输出端口TXCb和信号通信发射机/接收机(TRX)发送到下一辆轨道动车。

极限速度控制器2的极限速度内插单元用发送到输入端口RXC的传输同步时钟和提供给输入端口RXD的速度极限信息以及轨道车自身位置，计算一极限速度。速度检测单元用距离脉冲和时间间隔计时器的输出计算一当前运行速度。刹车指令计算控制单元将

轨道动车自身的速度与该极限速度比较。如果它超过该极限速度，则刹车指令计算控制单元通过继电器单元3输出一刹车指令，并控制刹车单元7，从而使轨道动车安全运行。

如图3所示，在接收到前一辆轨道动车的位置、在轨道动车前的一段内的极限速度等信息以后，速度控制器(FS—CPU)1用数据接收差错检验单元来检验这些信息，并根据轨道动车Xf自身的位置信息、铁路数据等共享极限速度方式的计算。

速度控制器(FS—IOU)2根据极限速度方式，共享即时刹车控制指令的计算。这里，参考字符NBF表示正常刹车指令，而参考字符EBF表示紧急刹车指令。当控制器出现故障时，产生用于控制器的故障检测指令FDR。

速度控制器2通过输出端口TXD和TXC、数据处理器1的输入端RXDa和RXCa以及输出端口TXDa和TXCa，将有关本轨道动车的位置信息传送到下一辆轨道动车。

继电器单元(RYU)3共享交变刹车指令的交流放大和交变刹车指令的整流。这里，刹车指令是交变的原因是，为了防止控制器或继电器驱动电路由于故障而不安全地固定在刹车释放一侧，通过将交变电流信号判定为刹车释放指令，而交变电流的停止判定为刹车动作指令，从而在故障时该指令变成刹车动作侧的交变电流停止，来保持安全运行功能。通常采用这一方法。

在从地面接收到前一辆轨道动车的位置、在轨道动车前一段内的极限速度等信息以后，速度控制器(FS—CPU)1用数据接收差错检验单元来检验信息中有无差错存在。如果不存在差错，则数据处理器参照存储有在前一辆轨道动车Xp的位置和本轨道动车Xf当前位置之间的一段轨道斜坡信息的铁路数据档案和速度极限信息(如弯道、分叉)有关的数据，随后根据这些数据，计算驱动该轨道动车既不撞上前一辆轨道动车但又不超过该极限速度的极限速度方式Vp

(Xi)。

这里，坡度信息用来计算轨道动车的有效减速度。Xi 是由于坡度变化、速度极限等产生的方式的拐点(inflection point of the pattern)。这样算得的极限速度方式序列(Vp(X0)，Vp(X1)，...，Vp(Xn))通过输出端口 TXDa 和 TXCa 被传送到速度控制器(FS—IOU)2。在数据处理器(FS—CPU)1 中重复方式数据计算过程，从而无差错地进行处理，而这一点在图中未示出。

另一方面，由安装在轮子 8 的轮轴中的转速传感器 6 检测的距离脉冲被输入到速度控制器 2 的输入端口 TGP。这些脉冲由转速传感器输入轮子直径校正单元来校正，随后在距离计算单元中累加，以计算本轨道动车的当前位置。速度控制器(FS—IOU)2 通过累加上述脉冲来计算本轨道动车的当前位置 Xf(t)，并且，同时计算轨道动车的自身速度 Vf(t)，它是本轨道动车当前位置的时间微分。

根据从数据处理器(FS—CPU)1 接收的本轨道动车当前位置 Xf(t)以及极限速度方式序列(Vp(X0)，Vp(X1)，...Vp(Xn))，速度控制器(FS—IOU)2 选择一对极限速度方式 Vp(Xi)，Vp(Xi—1)，Xi＜Xf＜Xi—1，即，处在包含轨道动车自身位置的那一段，随后通过内插这一段来计算 Xf 处的极限速度 Vp(Xf)。

在刹车指令计算控制单元中，将极限速度 Vp(Xf)与本轨道动车的当前速度 Vf(t)作比较。从而，如果 Vf＞Vp，则将正常刹车指令输出到输出端口 NBR。然而，如果 Vf＞Vp＋α(这里，α 是常数：例如，5km/h)，则将紧急刹车指令输出到输出端口 EBR。这时，最后输出的刹车指令是以交变电流信号的形式输出的。通过将交变电流信号判定为刹车释放指令，而将交变电流的停止判定为刹车动作指令，从而在故障时在刹车动作一侧的交变电流停止时出现该指令，由此保持安全运行功能。

在将铁路条件存储为在车(on—car)单元中的数据并根据上述

数据进行控制的系统中，当使用了某一错误速度方式进行控制的最坏情况下，存在轨道动车相撞的可能性。因为方式数据中一个位的差错会使数据变成另一完全不同的数据，所以，必须进行方式数据的计算处理和数据处理器(FS—CPU)1与速度控制器(FS—IOU)2之间的通信。

图4是数据处理器FS—CPU的结构。中央处理单元(CPU)11由时钟发生器CPG10控制。串行通信控制电路(下文称为CCUa和CCUb)12a和12b与速度控制器(FS—IOU)2、诸如用于地面单元的发射机/接收机等的电路相连。

尽管图中的结构有两个端口，但个数是任意的，并且还可以提供一个外部总线接口(本实施例中被省去)。

图5是图2中所示数据处理器(FS—CPU)中进行处理的处理流程图。步骤110a中，由程序A计算方式数据A 1100a。与此类似，步骤111a中的CRC码A 1110a也是由程序A计算的。这里计算的CRC码序列称作为“序列0”。

步骤110b中，由程序B计算方式数据B。程序B与程序A具有相同的功能，但与程序A不同。这样做，就防止了两个程序同时出现相同的差错。与此类似，CRC码B 1110b由程序B计算。这里算得CRC码B的序列(下文中称为“序列1”)与由程序A算得的CRC码A的序列不同。

步骤112中，对程序A和B的算得结果之间的CRC码进行交换处理。这里，如图6中所示，将方式数据A 1100a与CRC码B1110b组合在一起的数据称作为被转移数据，而将方式数据B 1100b与CRC码A 1110a组合在一起的数据称作为转移数据B 1120b。

顺便指出，图6是步骤112中的处理轮廓图。步骤113中，判断数据交换标志FLAG。如果FLAG＝1，则在步骤114a中执行对被发

送数据 A 的发送处理，然后在步骤 115a 中将标志重新设定为 FLAG＝0。如果 FLAG＝0，则在步骤 114b 中执行对被发送数据 B 的发送处理，然后在步骤 115b 将标志重新设定为 FLAG＝1。所以，对于每一个计算周期，从数据处理器(FS－CPU)交替地发送被转移数据 A 和被转移数据 B。

图 7 是速度控制器(FS－IOU)2 的详细结构方框图。速度控制器 FS－IOU 包含：中央处理单元(CPU)21、用来驱动 CPU21 的时钟脉冲发生器 20、用来连接数据处理器 FS－CPU1 和其他控制单元和电路的串行通信控制电路 CCU22、用来对来自转速传感器的距离脉冲进行计数从而计算轨道动车当前位置和速度的脉冲计数器(CNT)、用来对 CPU 计算的刹车指令进行信号转换从而提供给继电器单元 RYU 的输出接口电路(IFU)24 以及数据检验电路(CHK)25。

单元中的每一个具有读信号 RD、写信号端子 WR、暂停信号端子、中断信号端子 WAI、地址信号端子 A15 至 A0，以及数据总线端子 D0 至 D7。在速度控制器(FS－IOU)2 的中央处理单元 CPU21 中进行刹车指令计算之前，检验从数据处理器(FS－CPU)1 得到的被转移数据。

图 8 是检验被发送数据的流程图。在步骤 210 中进行对来自数据处理器(FS－CPU)1 的被发送数据(A 或 B)的接收。

在步骤 211 中，用检验程序 A 对具有“序列 0”CRC 码的被发送数据进行 CRC 检验。这里，如果检验结果是“无差错”，则在步骤 212a 中将 CRC 寄存器 a 设定成“1”。如果检验结果是“差错”，则在步骤 213a 中将 CRC 寄存器 a 设定成“0”。

本实施例中，把 CPU 寄存器中特定的一个字节指派为 CRC 寄存器，最低(第 0)位用作 CRC 寄存器 a。在步骤 211b 中，用检验程序 B 对具有“序列 1”CRC 码的被发送数据进行 CRC 检验。其中，如果

检验结果是“无差错”，则在步骤212b中将CRC寄存器b设定为“1”。如果检验结果是“差错”，则在步骤213b中，将CRC寄存器b设定成“0”。

与CRC寄存器b一样，采用上述CRC寄存器的第一位。步骤214中，将CRC寄存器a和b的内容发送到数据检验电路(CHK)25中的一个输出寄存器(REG)252。

在完成这些步骤以后，CPU21根据方式数据执行刹车指令计算，并将结果传送到输出接口电路(IFU)24。

如果由数据处理器(FS—CPU)1中的两个程序算得的方式数据计算结果相同，则当具有CRC码B的被发送数据由检验程序B检验时，检验结果变成“无差错”，且CRC寄存器b被设定成“1”。

另一方面，当具有CRC码A的被发送数据B由检验程序A进行CRC检验时，检验结果变成“无差错”，且CRC寄存器a被设定成“1”。这是因为，由于CRC码a和b是对相同数据产生的，所以，即使交换CRC码，CRC码也与方式数据(a或b)相匹配。

相反，当具有CRC码B的被发送数据A由检验程序A进行CRC检验时，则不管被发送数据的内容如何，CRC序列都不同，检验结果总是“差错”，而CRC寄存器a都被设定成“0”。

另一方面，当具有CRC码A的被发送数据由检验程序B进行CRC检验时，由于不管被发送数据的内容如何，CRC序列都不同，检验结果总是“差错”，且CRC寄存器a(b)被设定为“0”。因为被发送数据a和被发送数据b是从数据处理器(FS—CPU)1被交替发送的，则在每个计算周期内将“1”和“0”交替地写入CRC寄存器a和b。另外，CRC寄存器a和b的内容是互相相反的，因而二者不可能同时为“1”或“0”。

下面描述数据处理器(FS—CPU)中的两个处理结果是不一致的情形。这种情况下，由于数据不与CRC码对应，则由程序B(或A)

对被发送数据A(或B)的检验结果变成“差错”,且CRC寄存器b(或a)被设定成“0”。另一方面,在这种情况下,如上所述,CRC寄存器b(或a)的内容为“0”。所以,两个CRC寄存器的内容同时为“0”。

当数据发送过程中出现差错时,由于数据中存在差错,所以,由程序B(或A)对被发送数据A(或B)的检验结果变成“差错”,且CRC寄存器b(或a)被设定成“0”。所以,在这种情况下,两个CRC寄存器a和b的内容也同时变成“0”。

当CPU 21中或CPU中的检验程序中出现异常时,CRC的检验结果被固定在“无差错”或“差错”,或者随机出现这些结果。

当由于数据处理器(FS—CPU)1中的中央处理单元(CPU)21或CPU中的程序出现异常而使速度控制器(FS—IOU)2的输出固定于被发送数据a或b时,CRC寄存器b的内容固定在“1”(或“0”)。

图9是数据检验电路(CHK)25的内部结构图。“与”门251对地址译码器(DEC)250的输出和CPU21的写信号WR作“与”。输出寄存器REG 252由两个触发器FF0和FF1组成,在触发器内分别写入CRC寄存器a和b的内容。图10示出正常情况下FF0和FF1的输出时序图。

如图所示,交变“与”电路(FSAND)253是将触发器(FFa)2530和触发器(FFb)2531组合在一起构成的。仅当输入信号Ta和Tb是交变信号并且相位互相相反,被检验的输出FDF才变成交变信号。图11至图13是FSAND的运行时序图。

图11是当数据处理器(FS—CPU)中的计算处理过程中出现不一致或者在数据传输过程中出现差错时,交变“与”电路的运行时序图。因为当出现不一致或差错时,输入Ta和Tb同时变成0,所以检验输出FDF的交替变化停止。

图12是在检验程序中出现异常时交变“与”电路的运行时序图。

这种情况下，在出现异常后，在输入 Ta 和 Tb 同时变成“1”或“0”的时刻，检验输出 FDF 的交替变化停止。图中示出了程序 A 中出现异常且输出固定在“1”时的情况，但这种情况同样出现在输出固定在“0”或程序 B 中出现异常的时候。

图 13 是当数据处理器(FS—CPU)的输出固定在被发送数据 a 时交变“与”电路的运行时序图。在此情形下，当被发送数据固定时，检验输出 FDF 的交替变化同时停止。同样，当数据处理器(FS—CPU)的输出固定在被发送数据 b 时也会出现这种情况。

如上所述，当数据处理器(FS—CPU)中的处理过程中出现不一致，或者在数据传输过程中出现差错，或者在检验程序中出现差错，或者数据处理器(FS—CPU)的输出数据被固定的时候，FSAND 的检验输出 FDF 的交替变化停止，所以一定可以检测出这种异常。

继电器单元 RYU 中的故障检测继电器 FDR 是由 FSAND 的检验输出 FDF 驱动的，并且当 FDF 的交替变化停止后，FDR 的线圈断电，紧急刹车指令通过打开触点而输出至刹车单元 BCU。所以，当数据传输过程中出现故障，或者 CRC 检验器中出现故障，或者数据处理器(FS—CPU)的输出数据被固定的时候，就操纵紧急刹车，从而可以获得防障能力。

本实施例中采用的是 CRC 码，实际上也可以采用其他的编码。本实施例中 CRC 检验是用软件进行的，当然 CRC 检验也可以采用硬件进行。可以在单独的一个硬件中完成两种处理过程，也可以由另一个硬件来完成。

如果仅仅为了找出在数据传输过程中的差错，则用一个程序来进行计算处理，而两个检验码产生程序是由切换每一计算循环用的。这样，就可以将检验结果改变成一交变信号。

从上面的描述中可知，按照本发明，通过采用两个与不同检验码对应的检验装置，以及通过将检验码交替地输入到检验装置来使检

验结果交替变化，可以不使用任何特别防障理论而实现防障检验。从而可以使检验理论简化，并且采用防障检验的系统尺寸可以做得更小而可靠性更高。

说　明　书　附　图

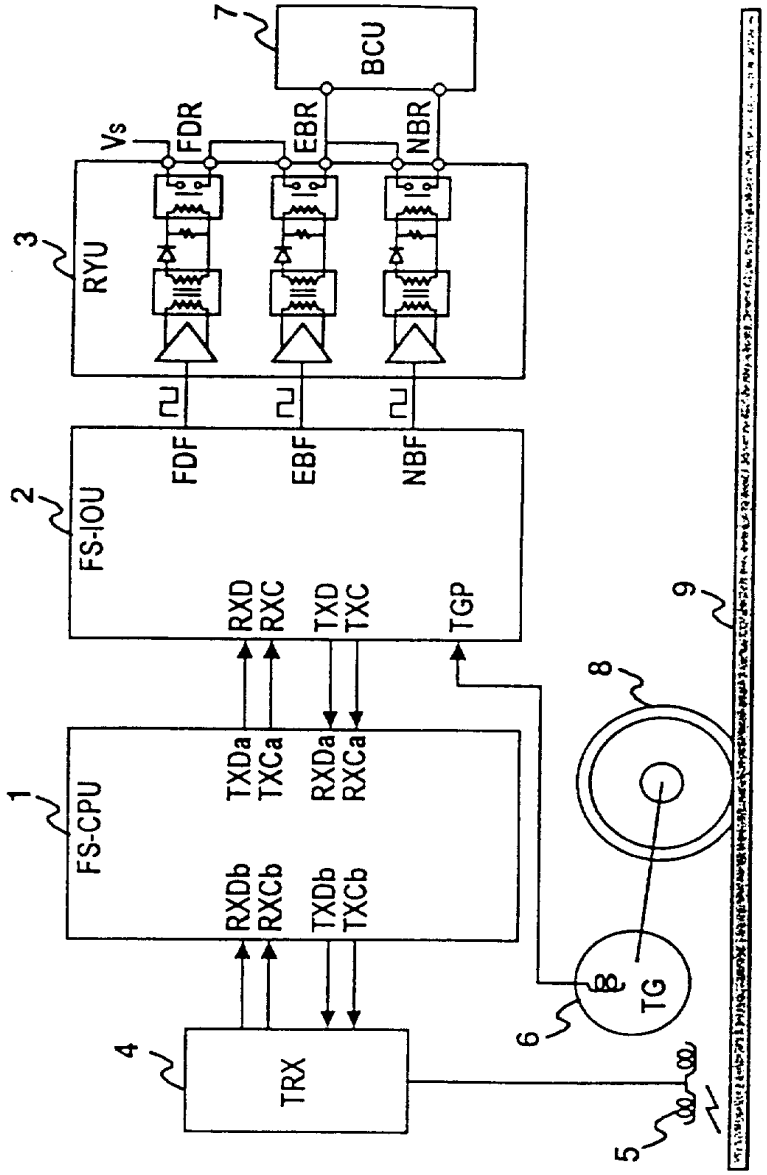

7
BCU
Vs
FDR
EBR
NBR
3
RYU
FDF
EBF
NBF
2
FS-IOU
RXD
RXC
TXD
TXC
TGP
1
FS-CPU
TXDa
TXCa
RXDa
RXCa
RXDb
RXCb
TXDb
TXCb
9
8
TG
6
4
TRX
5

图 1

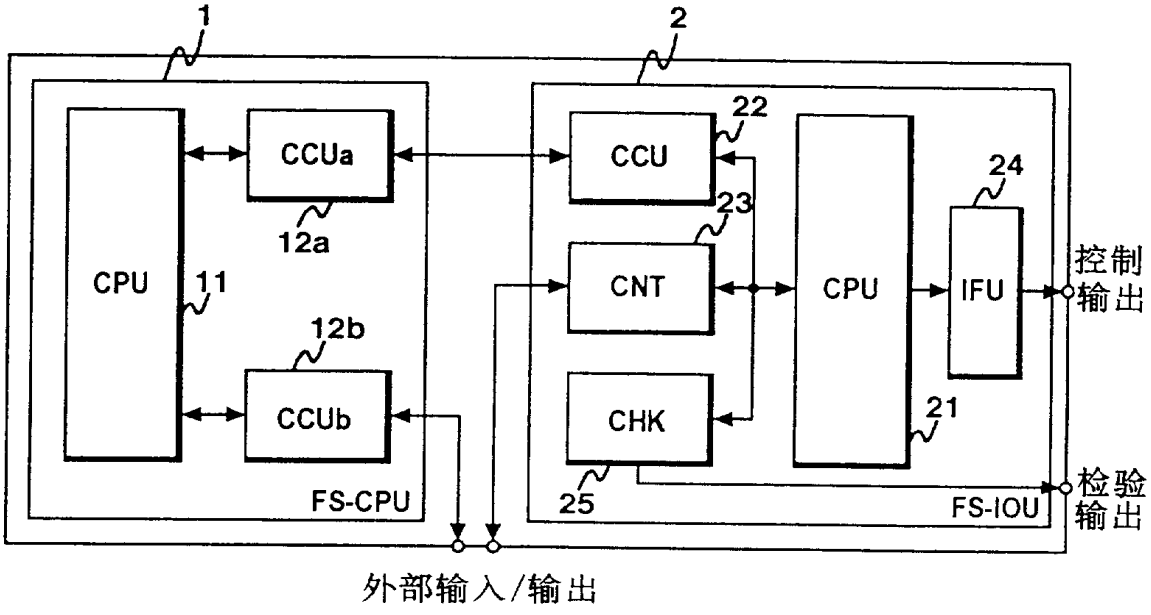
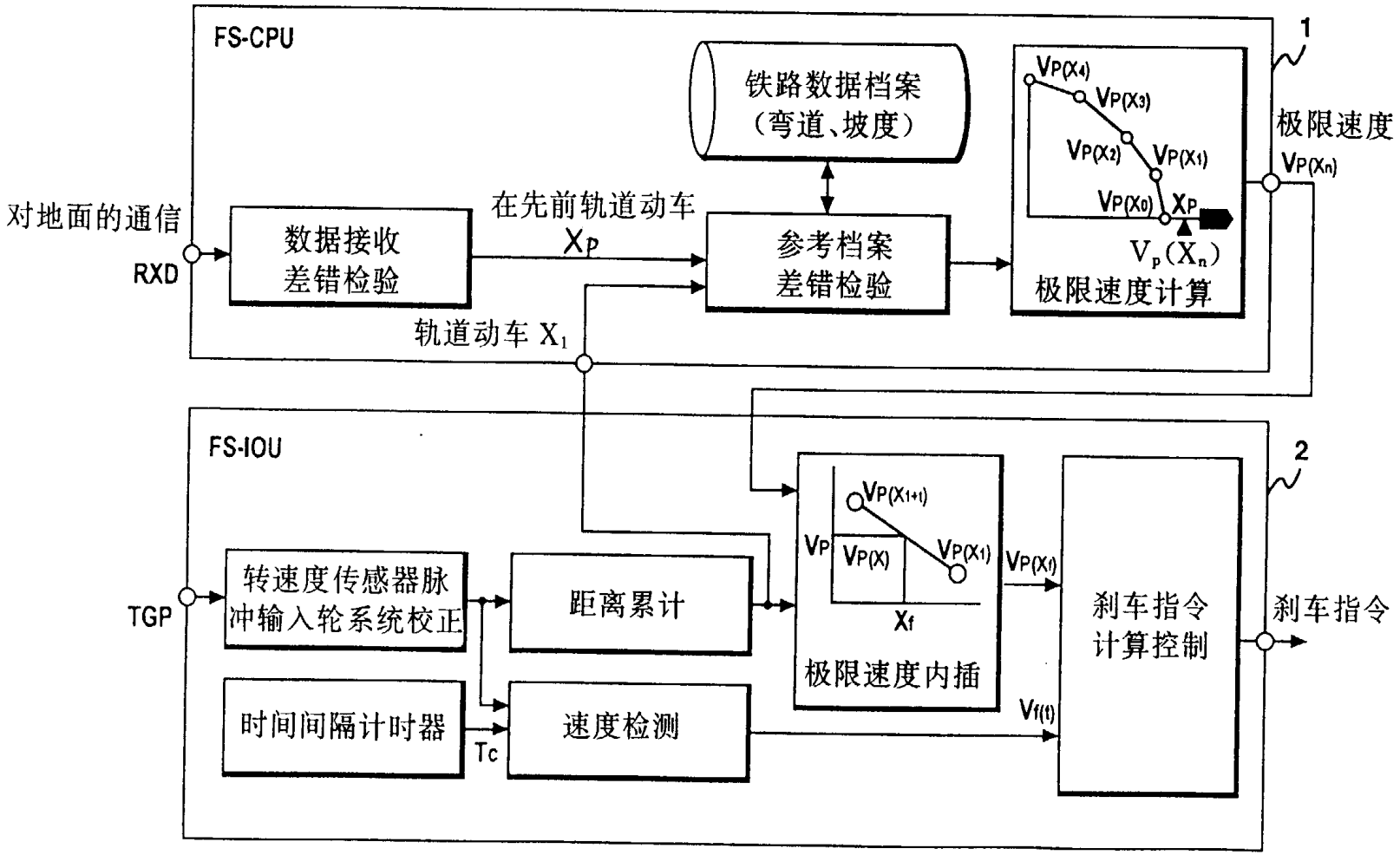

1
2
CPU
11
CCUa
12a
12b
CCUb
FS-CPU
CCU
22
CNT
23
CHK
25
CPU
21
24
IFU
控制输出
检验输出
FS-IOU
外部输入/输出

图 2

图 3


FS-CPU
铁路数据档案
（弯道、坡度）
1
极限速度
VP(Xn)
VP(X4)
VP(X3)
VP(X2)
VP(X1)
VP(X0)
XP
Vp(Xn)
极限速度计算
对地面的通信
RXD
数据接收
差错检验
在先前轨道动车
XP
参考档案
差错检验
轨道动车 X1
FS-IOU
2
VP(X1+1)
VP
VP(X)
VP(X1)
Xf
VP(Xf)
极限速度内插
转速度传感器脉
冲输入轮系统校正
TGP
距离累计
刹车指令
计算控制
刹车指令
时间间隔计时器
Tc
速度检测
Vf(1)

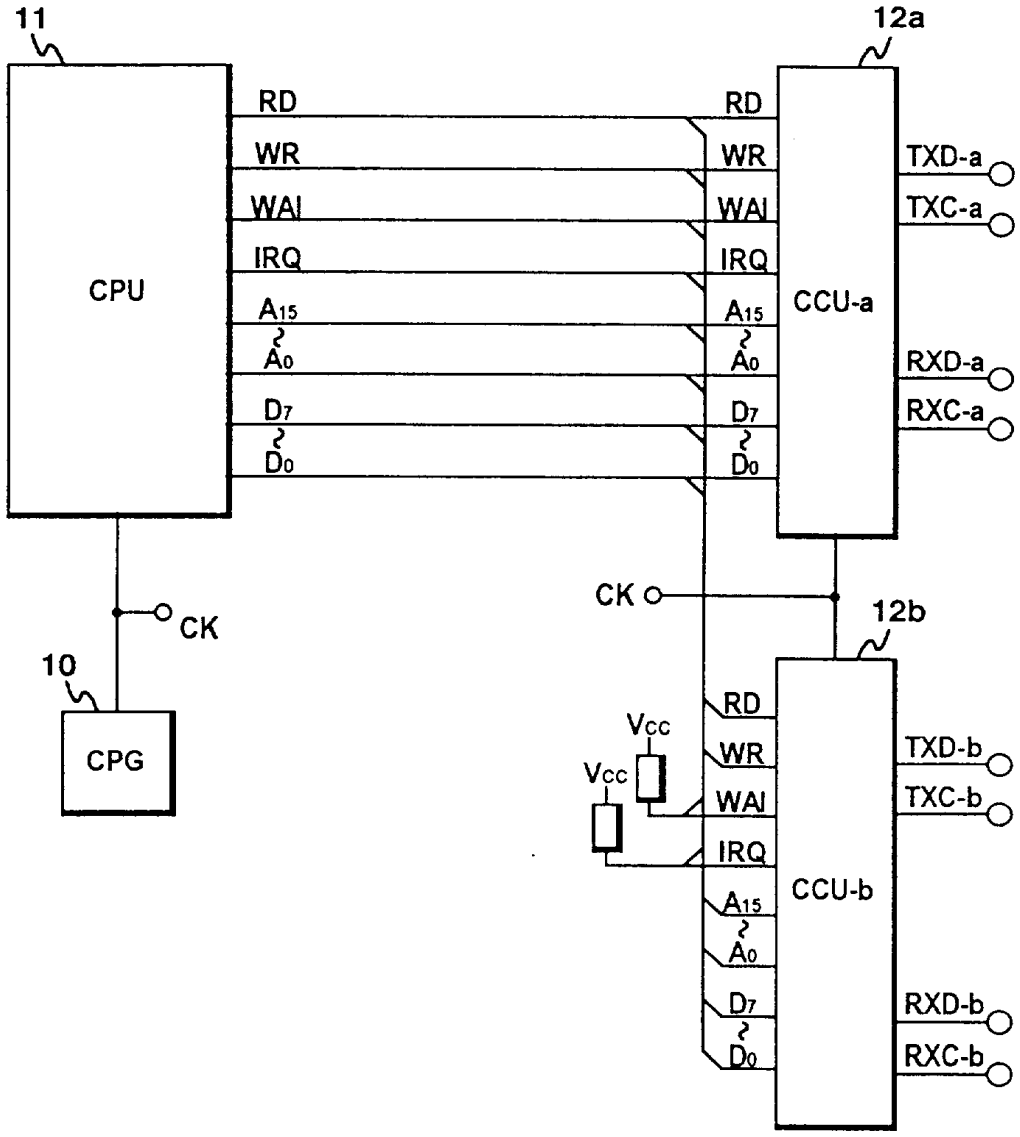

11
12a
CPU
RD
WR
WAI
IRQ
A15
A0
D7
D0
CCU-a
TXD-a
TXC-a
RXD-a
RXC-a
CK
10
CPG
12b
Vcc
Vcc
CCU-b
TXD-b
TXC-b
RXD-b
RXC-b

図 4

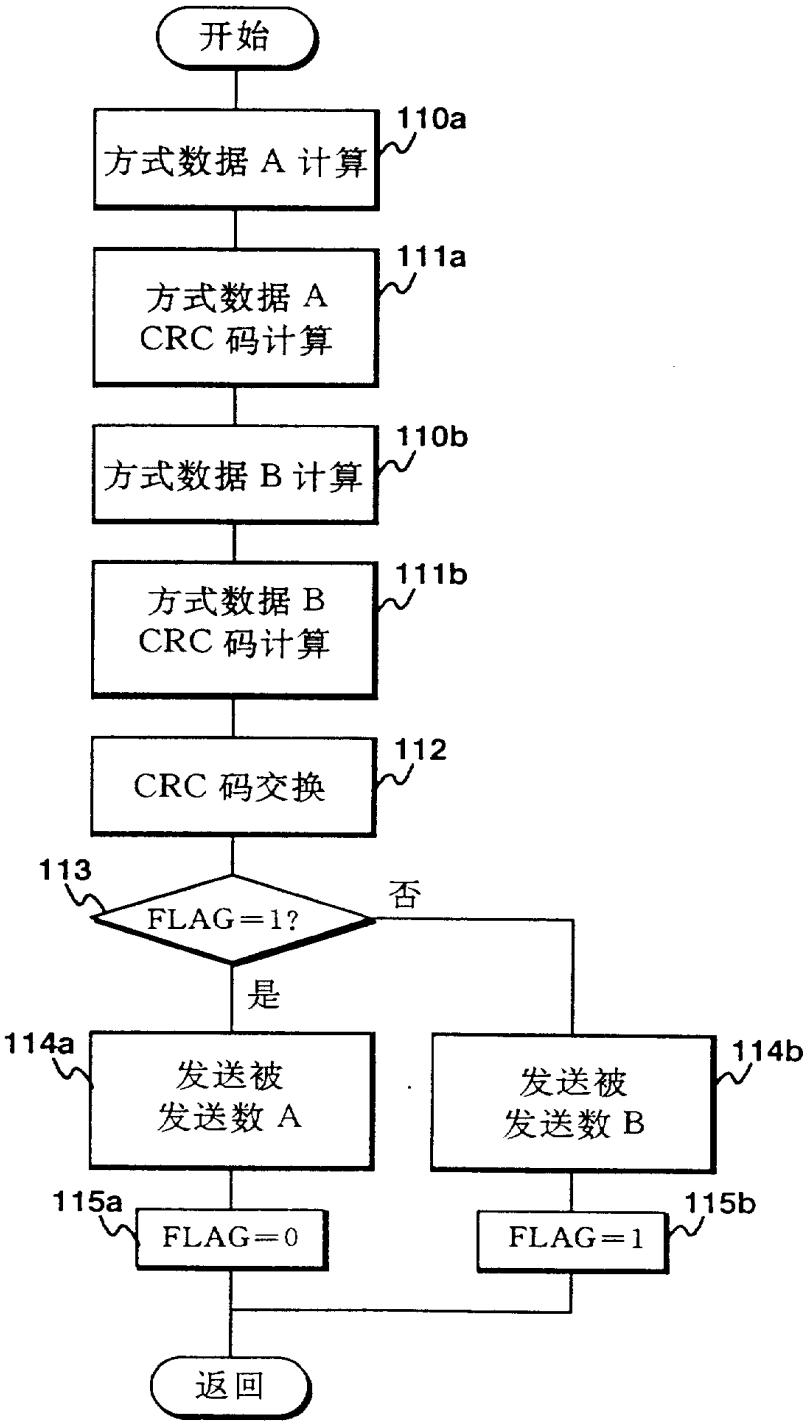

开始
方式数据 A 计算
110a
方式数据 A
CRC 码计算
111a
方式数据 B 计算
110b
方式数据 B
CRC 码计算
111b
CRC 码交换
112
113
FLAG=1?
否
是
114a
发送被
发送数 A
发送被
发送数 B
114b
115a
FLAG=0
FLAG=1
115b
返回

图 5

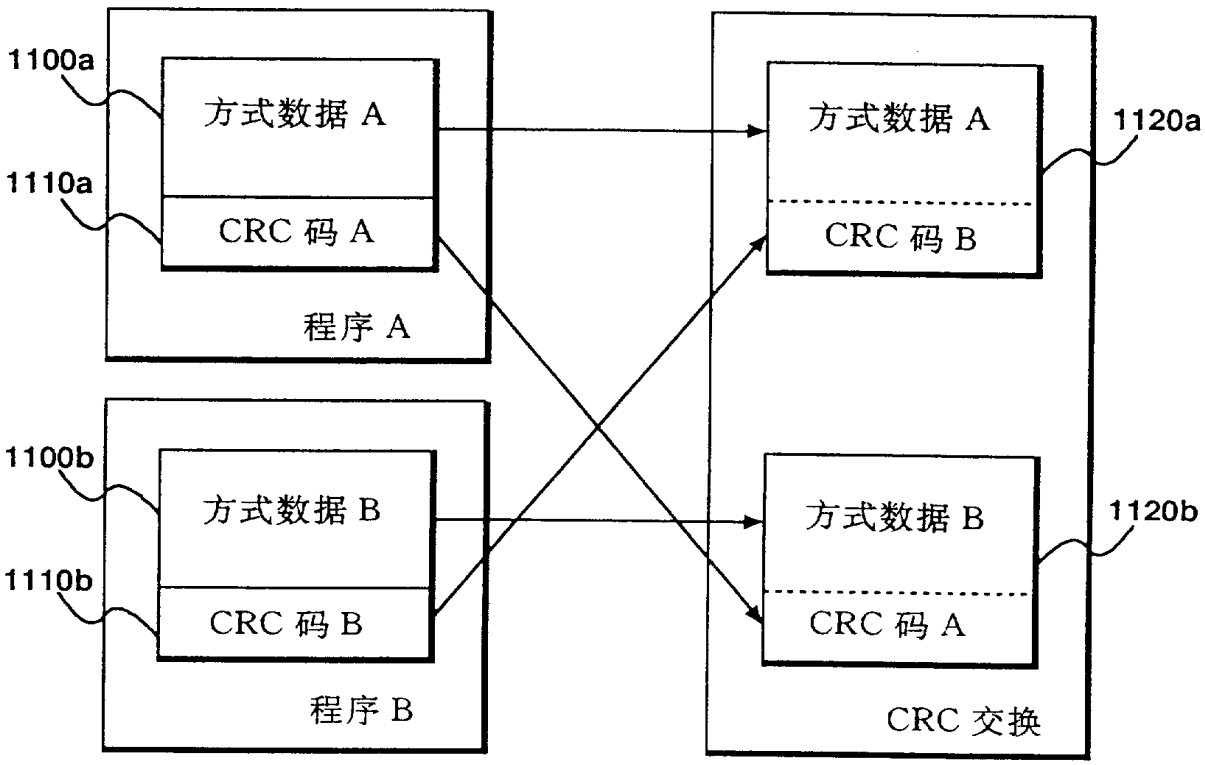

1100a
方式数据 A
1110a
CRC 码 A
程序 A
1100b
方式数据 B
1110b
CRC 码 B
程序 B
方式数据 A
1120a
CRC 码 B
方式数据 B
1120b
CRC 码 A
CRC 交换

图 6

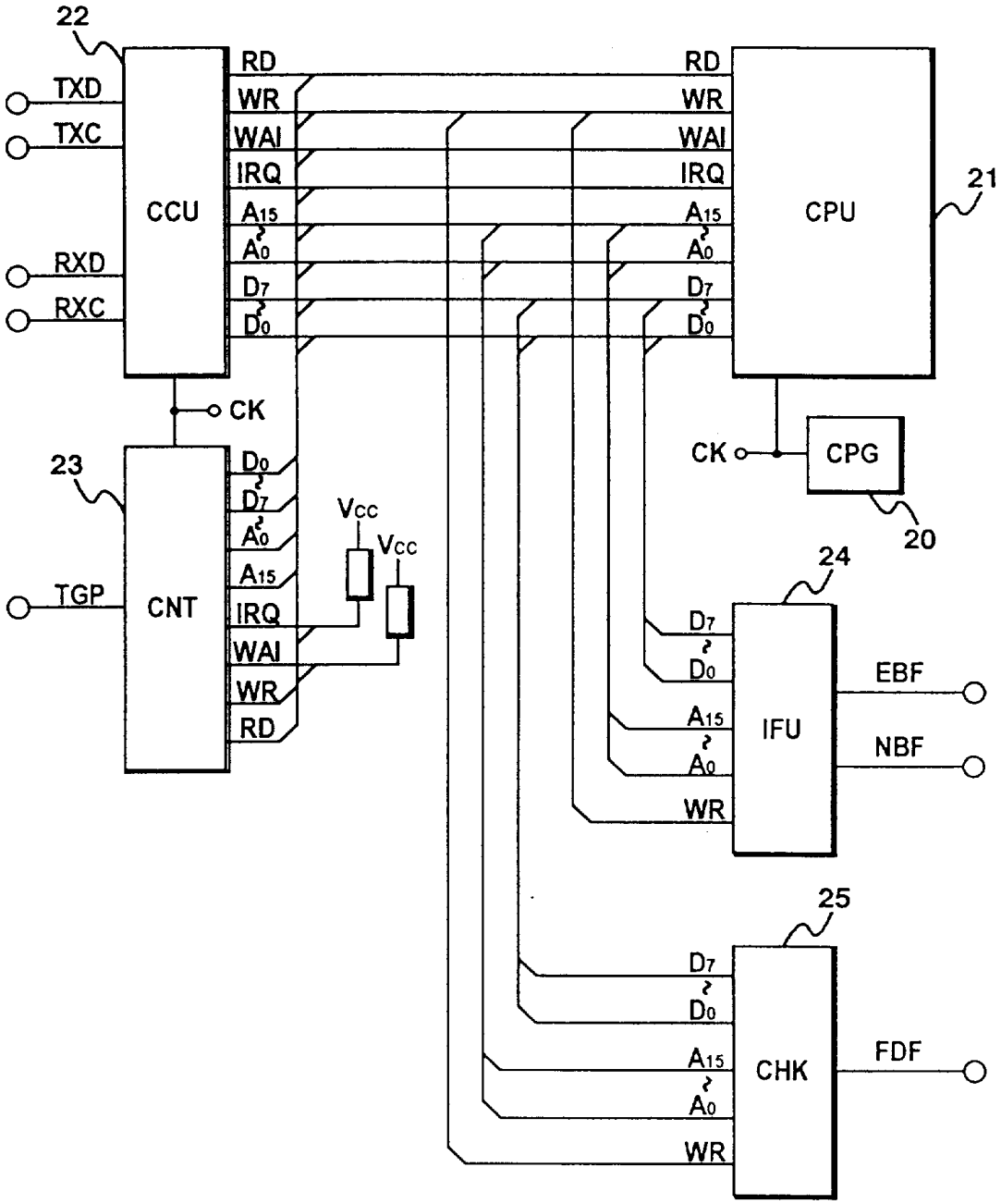

22
TXD
TXC
RXD
RXC
CCU
RD
WR
WAI
IRQ
A15
A0
D7
D0
CK
23
TGP
CNT
Vcc
Vcc
RD
WR
WAI
IRQ
A15
A0
D7
D0
CPU
21
CK
CPG
20
24
IFU
EBF
NBF
25
CHK
FDF

図 7

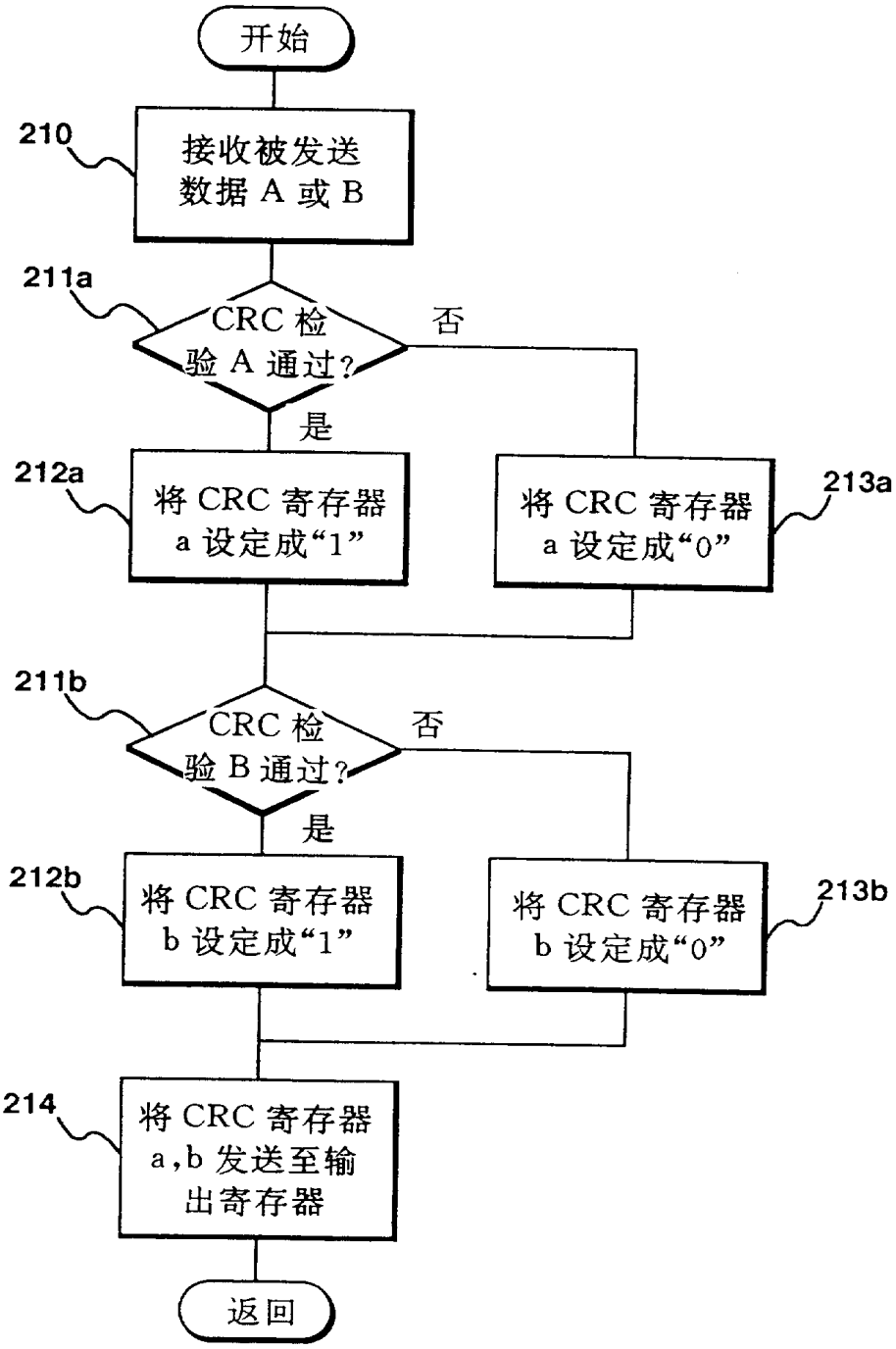

开始
210
接收被发送数据 A 或 B
211a
CRC 检验 A 通过?
否
是
212a
将 CRC 寄存器 a 设定成“1”
将 CRC 寄存器 a 设定成“0”
213a
211b
CRC 检验 B 通过?
否
是
212b
将 CRC 寄存器 b 设定成“1”
将 CRC 寄存器 b 设定成“0”
213b
214
将 CRC 寄存器 a,b 发送至输出寄存器
返回

图 8

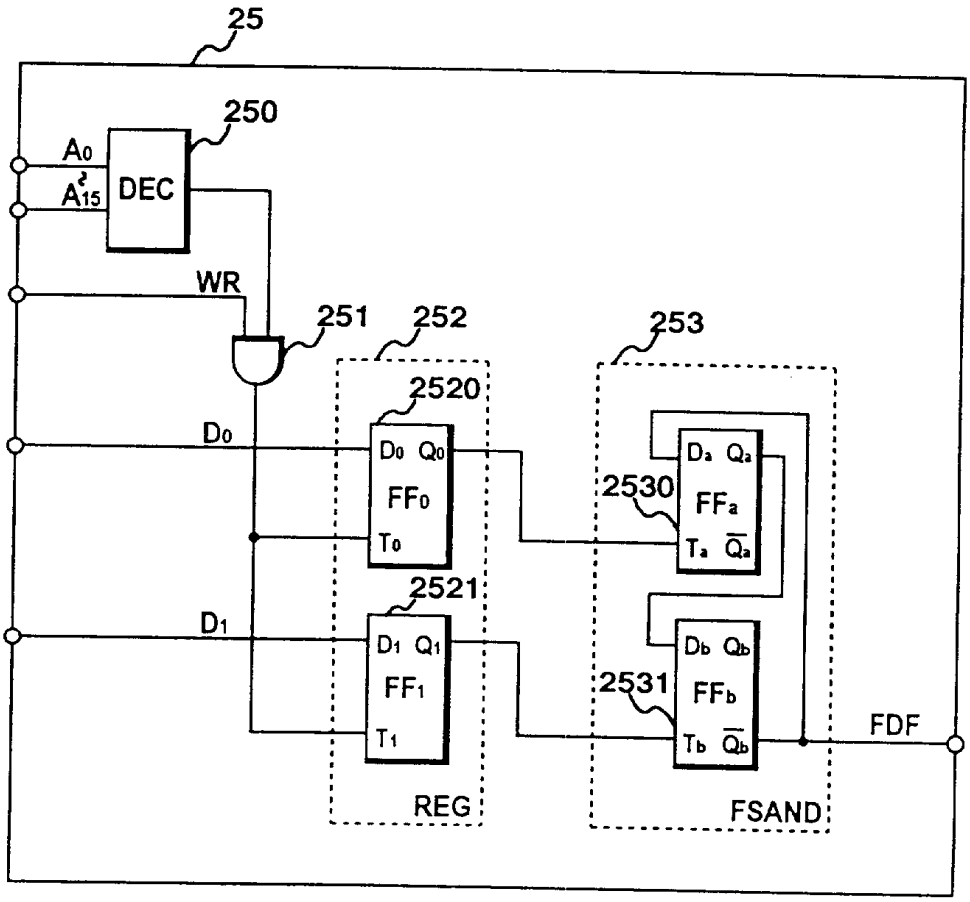

25
250
A0
A15
DEC
WR
251
252
253
2520
D0
D0 Q0
FF0
T0
2530
Da Qa
FFa
Ta Qa
2521
D1
D1 Q1
FF1
T1
2531
Db Qb
FFb
Tb Qb
FDF
REG
FSAND

图 9

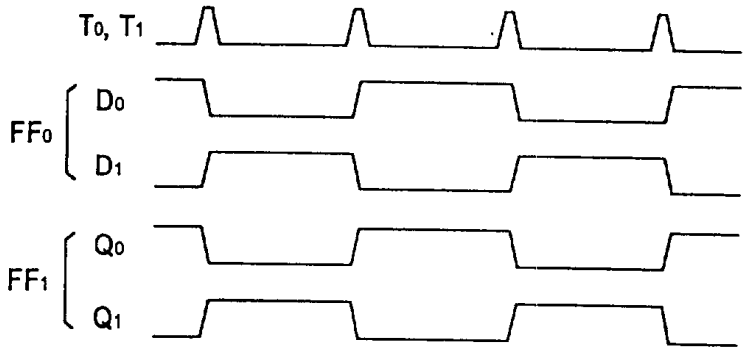

T0, T1
FF0
D0
D1
FF1
Q0
Q1

图 10

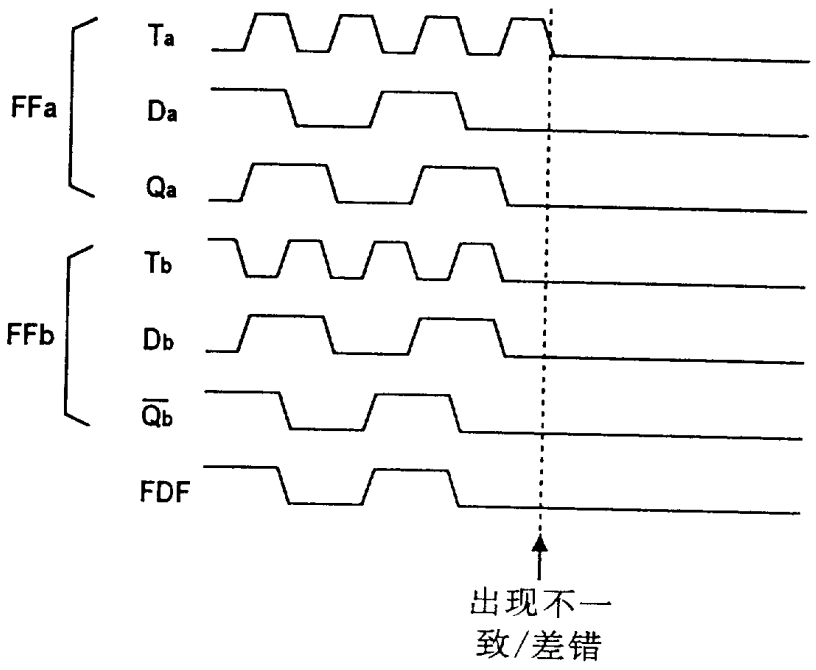


图 11

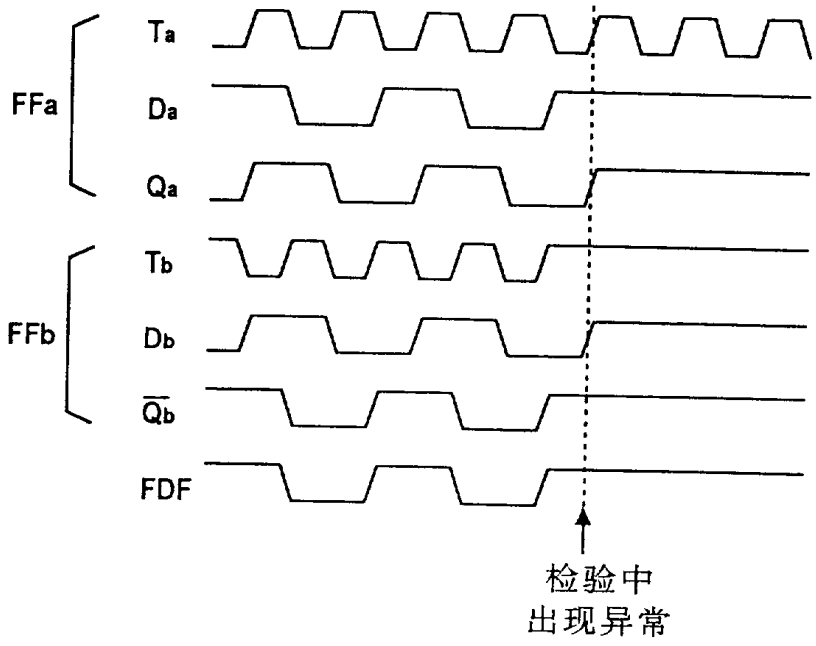


图 12

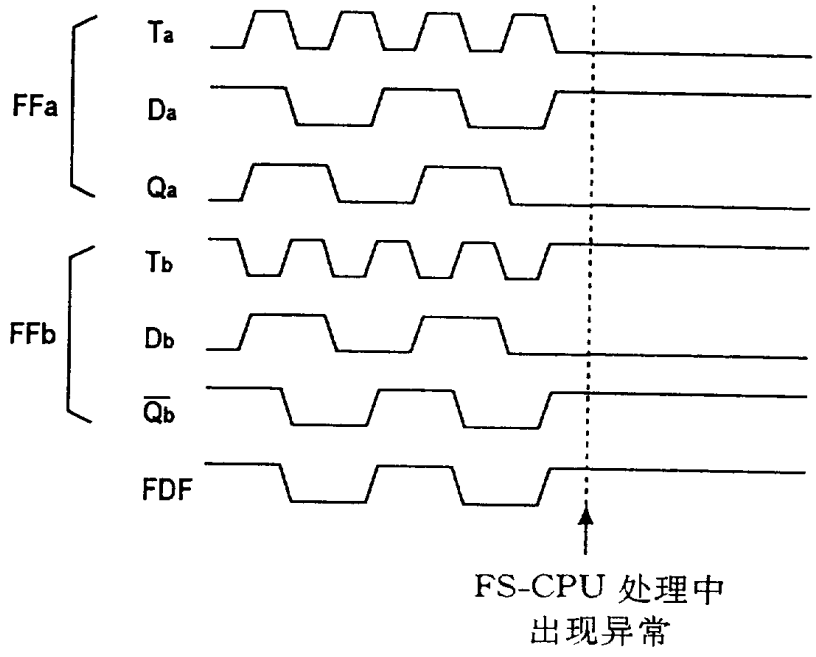


图 13