[19] 中华人民共和国国家知识产权局

[51] Int. Cl.
G06F 12/14 (2006.01)

[12] 发明专利说明书

专利号 ZL 200710130039.5

[45] 授权公告日 2009 年 7 月 29 日

[11] 授权公告号 CN 100520741C

[22] 申请日 2007.7.25
[21] 申请号 200710130039.5
[30] 优先权
[32] 2006. 7.25 [33] JP [31] 2006 - 201505
[73] 专利权人 索尼株式会社
地址 日本东京
[72] 发明人 村冈如竹
[56] 参考文献
CN1355922A 2002.6.26
EP0837475A2 1998.4.22
EP0154252A2 1985.9.11
WO98/47060A2 1998.10.22
审查员 张 千

[74] 专利代理机构 北京康信知识产权代理有限责任公司
代理人 余 刚 吴孟秋

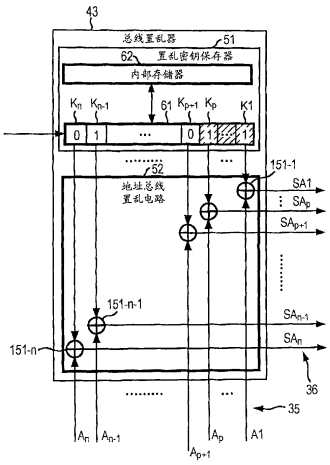
权利要求书 2 页 说明书 17 页 附图 8 页

[54] 发明名称

存储器存取控制装置和方法、以及通信装置

[57] 摘要

一种存储器存取控制装置，包括以下元件：置乱密钥生成器，被配置为生成包括为固定值的预定低阶位的二进制置乱密钥，其中，最低有效位的值是 1，剩余位是随机数或伪随机数；以及分配单元，被配置为使用置乱密钥来置乱逻辑地址，从而将物理地址分配给逻辑地址。



1. 一种存储器存取控制装置，包括:

置乱密钥生成装置，用于生成包括为固定值的预定低阶位的二进制置乱密钥，其中，最低有效位的值是1，剩余位是随机数或伪随机数；以及

分配装置，用于使用所述置乱密钥来置乱逻辑地址，从而将物理地址分配给所述逻辑地址。

2. 根据权利要求1所述的存储器存取控制装置，其中，所述置乱密钥生成装置生成所述固定值是只包括 1 的比特流的所述置乱密钥。

3. 根据权利要求1所述的存储器存取控制装置，进一步包括用于生成所述随机数或所述伪随机数的随机数生成装置。

4. 根据权利要求3所述的存储器存取控制装置，其中，所述随机数生成装置生成 Gold 序列伪随机数。

5. 根据权利要求3所述的存储器存取控制装置，其中，在所生成的随机数或所生成的伪随机数等于预定值的情况下，所述随机数生成装置生成新的随机数或新的伪随机数。

6. 一种存储器存取控制方法，包括以下步骤:

生成包括为固定值的预定低阶位的二进制置乱密钥，其中，最低有效位的值是1，剩余位是随机数或伪随机数；以及

使用所述置乱密钥来置乱逻辑地址，从而将物理地址分配给所述逻辑地址。

7. 一种用于与具有非接触集成电路卡功能的装置进行通信的通信装置，包括:

置乱密钥生成装置，用于生成包括为固定值的预定低阶位的二进制置乱密钥，其中，最低有效位的值是 1，剩余位是随机数或伪随机数；以及

分配装置，用于使用所述置乱密钥来置乱逻辑地址，从而将物理地址分配给所述逻辑地址，所述物理地址被用于存储从所述具有非接触集成电路卡功能的装置中读取的数据。

存储器存取控制装置和方法、以及通信装置

相关申请的交叉参考

本发明包含于2006年7月25日向日本专利局提交的日本专利申请JP 2006-201505的主题，其全部内容结合于此作为参考。

技术领域

本发明涉及存储器存取控制装置和方法以及通信装置，特别是，涉及一种易于提高存储器中的数据的安全性的存储器存取控制装置和方法，并涉及一种通信装置。

背景技术

例如在PCT日文翻译专利公开第2003-500786号中，已经提出了通过将被指定由处理器（例如中央处理单元（CPU）或类似物）访问的逻辑地址置乱，来把实际访问的物理地址分配给存储器，从而使存储器里的数据难于分析或篡改的提案。

发明内容

近年来，非授权的数据拦截和篡改已经变得越发巧妙，除了在PCT日文翻译专利公开第2003-500786号中描述的技术以外，还强烈要求加强存储器中数据的安全性。

期望能够很容易地加强存储器中的数据的安全性。

根据本发明的第一实施例，提供了一种存储器存取控制装置，包括以下元件：置乱密钥生成装置，用于生成包括为固定值的预定低阶位的二进制置乱密钥，其中，最低有效位的值是 1，剩余位是随机数或伪随机数；以及分配装置，用于使用置乱密钥来置乱逻辑地址，从而将物理地址分配给逻辑地址。

置乱密钥生成装置可生成其中的固定值是只包括 1 的比特流的置乱密钥。

该存储器存取控制装置可进一步包括用于生成随机数或伪随机数的随机数生成装置。

随机数生成装置可生成 Gold 序列伪随机数。

在生成的随机数或生成的伪随机数等于预定值的情况下，随机数生成装置可生成新的随机数或新的伪随机数。

根据本发明的第一实施例，提供了一种存储器存取控制方法，包括以下步骤：生成包括为固定值的预定低阶位的二进制置乱密钥，其中，最低有效位的值是 1，剩余位是随机数或伪随机数；以及使用置乱密钥来置乱逻辑地址，从而将物理地址分配给逻辑地址。

根据本发明的第二实施例，提供了一种通信装置，包括以下元件：置乱密钥生成装置，用于生成包括为固定值的预定低阶位的二进制置乱密钥，其中，最低有效位的值是 1，剩余位是随机数或伪随机数；以及分配装置，用于使用置乱密钥来置乱逻辑地址，从而将物理地址分配给逻辑地址，该物理地址被用于存储从具有非接触集成电路卡功能的装置中读取的数据。

根据本发明的第一实施例，生成包括为固定值的预定低阶位的二进制置乱密钥，其中，最低有效位的值是 1，剩余位是随机数或伪随机数，以及通过使用置乱密钥来置乱逻辑地址，从而将物理地址分配给逻辑地址。

根据本发明的第二实施例，生成包括为固定值的预定低阶位的二进制置乱密钥，其中，最低有效位的值是 1，剩余位是随机数或伪随机数；以及通过使用置乱密钥来置乱逻辑地址，从而将物理地址分配给逻辑地址，该物理地址被用于存储从具有非接触集成电路卡功能的装置中读取的数据。

根据本发明的第一或第二实施例，存储器里的数据变得难以分析或篡改。根据本发明的第一或第二实施例，能够很容易地提高存储器中数据的安全性。

附图说明

图 1 是根据本发明实施例的读/写器的框图；

图 2 是示出图 1 所示的控制模块的功能结构的框图；

图 3 是示出图 2 所示的随机数输出单元的功能结构的框图；

图 4 是示出图 2 所示的总线置乱器的详细功能结构的框图；

图 5 是描述图 2 所示的置乱密钥缓存器的内部寄存器中的值的序列的示图；

图 6 是描述由图 1 所示的读/写器执行的置乱密钥生成处理的流程图；

图7是描述由图1所示的读/写器执行的存储器存取控制处理的流程图；

图8是示出根据本发明第二实施例的图2所示的随机数输出单元的功能结构的框图；以及

图9是描述在读/写器具有图8所示的随机数输出单元的情况下，由图1所示的读/写器执行的置乱密钥生成处理的流程图。

具体实施方式

在描述本发明的实施例之前，以下描述了权利要求的特征与参照说明书或附图描述的本发明的实施例中公开的特定元素之间的对应关系。该描述的目的是确保在说明书或附图中描述了支持所要求的发明的实施例。因此，即使下面实施例中的元素在说明书或附图中没有相关于本发明的特定特征进行描述，这也不一定意味着该元素不与权利要求中的特征相关。相反，即使某一元素在本文中被描述为与权利要求中的特定特征相关，这并不一定意味着该元素不与权利要求的其他特征相关。

根据本发明的第一实施例，提供了一种存储器存取控制装置（例如，图2所示的总线置乱器**43**），包括以下元件：置乱密钥生成装置（例如，图2所示的置乱密钥缓存器**61**），用于生成包括为固定值的预定低阶位的二进制置乱密钥，其中，最低有效位的值是1，剩余位是随机数或伪随机数；以及分配装置（例如，图2所示的存储器**33**），通过使用置乱密钥来置乱逻辑地址，从而将物理地址分配给逻辑地址。

根据本发明第一实施例的存储器存取控制装置可进一步包括用于生成作为置乱密钥的随机数或伪随机数的随机数生成装置（例如，图 3 所示的随机数生成器 **101**）。

根据本发明的第一实施例，提供了一种存储器存取控制方法，包括以下步骤：生成包括为固定值的预定低阶位的二进制置乱密钥，其中，最低有效位的值是 1，剩余位是随机数或伪随机数（例如，图 6 所示的步骤 **S2** 或图 9 所示的步骤 **S105**）；以及使用置乱密钥来置乱逻辑地址，从而将物理地址分配给逻辑地址（例如，图 7 所示的步骤 **S38** 或 **S41**）。

根据本发明的第二实施例，提供了一种用于与具有非接触集成电路卡功能的装置（例如，图 1 所示的 IC 卡 **2**）进行通信的通信装置（例如，图 1 所示的读/写器 **1**），包括以下元件：置乱密钥生成装置（例如，图 2 所示的置乱密钥缓存器 **61**），用于生成包括为固定值的预定低阶位的二进制置乱密钥，其中，最低有效位的值是 1，剩余位是随机数或伪随机数；以及分配装置（例如，图 2 所示的地址总线置乱电路 **52**），用于使用置乱密钥来置乱逻辑地址，从而将物理地址分配给逻辑地址，该物理地址被用于存储从具有非接触集成电路卡功能的装置中读取的数据。

下面将参考附图描述本发明的实施例。

图 1 是根据本发明的实施例的读/写器的框图。根据本发明该实施例的读/写器 **1** 包括天线 **11**、射频（RF）驱动板 **12** 和控制模块 **13**。

RF 驱动板 **12** 经由天线 **11**，使用单频率载波与非接触集成电路（IC）卡 **2** 执行基于电磁感应的近距离通信。RF 驱动板 **12** 使用的载波的频率可以是例如在工业、科学和医学（ISM）频段中的

13.56MHz。近距离通信指的是当装置之间的距离在几十厘米以内时，装置能够相互通信，它还包括装置（的外壳）相互接触的通信。

控制模块 13 执行用于实施使用 IC 卡 2 的服务的处理。根据需要，控制模块 13 通过天线 11 和 RF 驱动板 12 从/向 IC 卡 2 读取/写入服务中使用的数据。控制模块 13 能够执行提供多种服务类型的并行处理。即，一个读/写器 1 能够提供使用非接触 IC 卡的多种服务，如电子货币服务、预付卡服务以及多种交通类型的票卡服务。

图 2 是示出图 1 所示的控制模块 13 的功能结构的框图。控制模块 13 包括 CPU 31、存储器存取控制器 32、存储器 33 以及复位电路 34。存储器存取控制器 32 包括置乱密钥变更指示单元 41、随机数输出单元 42 以及总线置乱器 43。总线置乱器 43 包括置乱密钥保存器 51 和地址总线置乱电路 52。置乱密钥保存器 51 包括置乱密钥缓存器 61 和内部存储器 62。

CPU 31 和地址总线置乱电路 52 通过设于其间的地址总线 35 相互连接，地址总线 35 的总线带宽为 n 位。地址总线置乱电路 52 和存储器 33 通过设于其间的地址总线 36 相互连接，地址总线 36 的总线带宽一样是 n 位。CPU 31 和存储器 33 通过设于其间的数据总线 37 相互连接，数据总线 37 的总线带宽是 m 位。

CPU 31 执行预定的程序来执行实施使用 IC 卡 2 的服务的处理。CPU 31 能够执行与相互并行的服务相关联的程序。换句话说，CPU 31 能够执行并行处理以提供多种服务。

CPU 31 从/向存储器 33 读取/写入各种服务中使用的数据。当向存储器 33 写数据时，CPU 31 经由地址总线 35 将表示逻辑数据写入位置的逻辑地址的逻辑地址信号提供给地址总线置乱电路 52，并经由数据总线 37 将包括要写入的数据和表示数据写入指令的写

入信号提供给存储器 33。当从存储器 33 中读数据时，CPU 31 经由地址总线 35 将表示逻辑数据读取位置的逻辑地址的逻辑地址信号提供给总线置乱电路 52，并经由数据总线 37 将表示数据读取指令的读取信号提供给存储器 33。

存储器存取控制器 32 控制 CPU 31 对存储器 33 的存取。

在包含于存储器存取控制器 32 的各个元件中，置乱密钥变更指示单元 41 包括，例如，按钮、开关、或类似物。为了改变置乱密钥保存器 51 中保存的置乱密钥，例如，用户经由置乱密钥变更指示单元 41 输入改变置乱密钥的指令。

在从置乱密钥变更指示单元 41 将表示改变置乱密钥的指令的信号提供给随机数输出单元 42 的情况下，随机数输出单元 42 生成包括 n-p 位的比特流的伪随机数，并将生成的伪随机数作为置乱密钥输出给置乱密钥缓存器 61。

总线置乱器 43 执行将从 CPU 31 提供的逻辑地址信号所表示的逻辑地址转换成用于实际访问存储器 33 的物理地址的处理。

在包含于总线置乱器 43 的各个元件中，置乱密钥保存器 51 使用从随机数输出单元 42 提供的伪随机数生成置乱密钥，并保存所生成的置乱密钥。更具体地，置乱密钥保存器 51 的置乱密钥缓存器 61 使用从随机数输出单元 42 提供的伪随机数生成置乱密钥，并保存所生成的置乱密钥。同时，置乱密钥缓存器 61 提供生成的置乱密钥并将其存储在内部存储器 62 中。内部存储器 62 是非易失性存储器，如闪存、或由电池或类似物支持的随机存取存储器（RAM）。即使在控制模块 13 的电源被关闭的情况下，内部存储器 62 仍持续地保存置乱密钥。当控制模块 13 从关闭的状态变为开启时，置乱密钥缓存器 61 读出存储在内部存储器 62 中的置乱密钥，并保存该

置乱密钥。然后，在从控制模块 13 开启到完成从内部存储器 62 读取置乱密钥的期间内，置乱密钥缓存器 61 向复位电路 34 提供复位指示信号。

利用置乱密钥缓存器 61 中保存的置乱密钥，地址总线置乱电路 52 把从 CPU 31 提供的逻辑地址信号表示的逻辑地址置乱，从而将逻辑地址转换成用于实际访问存储器 33 的物理地址。换句话说，地址总线置乱电路 52 置乱输入逻辑地址，从而给逻辑地址分配物理地址。地址总线置乱电路 52 通过地址总线 36 将表示经过转换的物理地址的物理地址信号提供给存储器 33。

存储器 33 是非易失性存储器，如闪存、电可擦除可编程只读存储器（EEPROM）、硬盘驱动（HDD）、磁阻 RAM（MRAM）、铁电 RAM（FeRAM）或 Ovonic 统一存储器（Ovonic Unified Memory）。在从 CPU 31 向存储器 33 提供写入信号的情况下，包含在该写入信号中的数据被写入到存储器 33 上的物理地址，该地址由从地址总线置乱电路 52 提供的物理地址信号所表示。在从 CPU 31 向存储器 33 提供读取信号的情况下，从存储器上的物理地址（该地址由从地址总线置乱电路 52 提供的物理地址信号所表示）读取数据，然后读出的数据经由数据总线 37 提供给 CPU 31。

当从置乱密钥缓存器 61 提供复位指示信号时，复位电路 34 提供复位信号给 CPU 31，从而初始化 CPU 31 的状态。

图 3 是示出随机数输出单元 42 的功能结构的框图。随机数输出单元 42 包括随机数字生成器 101 和开关 102。

随机数生成器 101 包括具有 L1 位移位寄存器的线性反馈移位寄存器（LFSR）随机数生成器 111、具有 L2 位移位寄存器的 LFSR 随机数生成器 112 和异或(EXOR)电路 113。

LFSR 随机数生成器 **111** 和 **112** 基于移位寄存器的预定位的值的异或作为反馈值输入到该移位寄存器的现有 LFSR 原理。随机数生成器 **101** 使用异或电路 **113**，通过一位一位地计算分别由 LFSR 随机数生成器 **111** 和 **112** 生成的两个不同的最大长度序列（M 序列）伪随机数的异或来生成 Gold 序列（Gold-sequence）伪随机数。随机数生成器 **101** 中包含的 LFSR 随机数生成器 **111** 和 **112** 的数目并不限于两个。随机数生成器 **101** 可以具有三个或更多个 LFSR 随机数生成器。

开关 **102** 响应于表示来自置乱密钥变更指示单元 **41** 的改变置乱密钥的指令的信号的输入而接通。表示由随机数生成器 **101** 生成的 Gold 序列伪随机数的比特流经由开关 **102** 输出至置乱密钥缓存器 **61**。

图 4 是示出总线置乱器 **43** 的详细功能结构的框图。

置乱密钥缓存器 **61** 包括具有串行及并行输入和并行输出的 n 位移位寄存器。如图 5 所示，在置乱密钥缓存器 **61** 的内部寄存器之中，低阶的 p 位（K1～K_p 位）是固定值，而作为串行信号从随机数输出单元 **42** 提供的伪随机数被设至剩余的高阶 n－p 位（K_{p+1}d～K_n 位）。即，置乱密钥缓存器 **61** 生成并保存为固定值的预定低阶 p 位和作为伪随机数的剩余 n－p 位的二进制置乱密钥。具有固定值的 p 位的最低有效位（LSB）总是设为 1。即，置乱密钥的 LSB 总被设为 1。

地址总线置乱电路 **52** 利用异或电路 **151-1～151-n**，一位接一位地计算包括经由地址总线 **35** 从 CPU **31** 提供的逻辑地址信号所表示的 A1～A_n 位的 n 位逻辑地址和包括置乱密钥缓存器 **61** 中保存的 K1～K_n 位的 n 位置乱密钥的异或，从而将逻辑地址转换为包括

$SA1 \sim SA_n$位的 n 位物理地址。地址总线置乱电路 **52** 经由地址总线 **36** 将表示转换后的物理地址的物理地址信号提供给存储器 **33**。

下面将参照图 6 和图 7 描述读/写器 **1** 执行的处理。

参照图 6 所示的流程图，将描述由读/写器 **1** 执行的置乱密钥生成处理。当在读/写器 **1** 开启的情况下，用户通过置乱密钥变更指示单元 **41** 输入改变置乱密钥的指令时该处理开始。

在步骤 **S1** 中，随机数输出单元 **42** 输出伪随机数。更具体地，置乱密钥变更指示单元 **41** 向开关 **102** 提供表示改变置乱密钥的指令的信号，从而开启开关 **102**。在读/写器 **1** 的电源打开的情况下，随机数生成器 **101** 始终生成伪随机数。通过接通开关 **102**，随机数生成器 **101** 开始经由开关 **102** 向置乱密钥缓存器 **61** 输出伪随机数。在随机数生成器 **101** 输出 n－p 位的伪随机数的情况下，开关 **102** 关闭。

在步骤 **S2** 中，总线置乱器 **43** 设置置乱密钥，并且置乱密钥生成处理结束。具体来说，置乱密钥缓存器 **61** 将包括从随机数输出单元 **42** 提供的 n－p 位的比特流的伪随机数设置为内部寄存器的高阶 n－p 位。从而，生成了包括 p 个低阶位固定值和 n－p 个高阶位伪随机数的 n 位置乱密钥。置乱密钥缓存器 **61** 在内部寄存器中保存生成的置乱密钥，并提供和在内部存储器 **62** 中存储置乱密钥。即，置乱密钥被备份在内部存储器 **62** 中。

从而，能够给每个控制模块 **13** 设置具有不同值并难于预测的置乱密钥。该置乱密钥设置处理例如在读/写器 **1** 从工厂出货之前执行。

接下来，参照图 7 的流程图，描述由读/写器 **1** 执行的存储器存取控制处理。该处理例如在读/写器 **1** 开启的情况下开始。

在步骤 **S31** 中，在读/写器 **1** 开启且控制模块 **13** 开启的情况下，置乱密钥缓存器 **61** 开始向复位电路 **34** 提供复位指示信号。

在步骤 **S32** 中，复位电路 **34** 开始向 CPU **31** 提供复位信号，从而复位 CPU **31**。因此，CPU **31** 的状态被初始化。

在步骤 **S33** 中，置乱密钥缓存器 **61** 读取内部寄存器 **62** 中保存的置乱密钥。置乱密钥缓存器 **61** 将读出的置乱密钥保存到内部寄存器中。

在步骤 **S34** 中，置乱密钥缓存器 **61** 停止向复位电路 **34** 提供复位指示信号。相应地，复位电路 **34** 停止向 CPU **31** 提供复位信号。CPU **31** 开始执行程序。

在步骤 **S35** 中，CPU **31** 确定是否写入数据。在由 CPU **31** 执行的程序中的下一处理不涉及写入数据的情况下，CPU **31** 确定不写入数据，流程进行到步骤 **S36**。

在步骤 **S36** 中，CPU **31** 确定是否读取数据。在由 CPU **31** 执行的程序中的下一处理不涉及读取数据的情况下，CPU **31** 确定不读取数据，流程返回到步骤 **S35**。

重复步骤 **S35** 和 **S36** 中的处理，直到确定在步骤 **S35** 中写入数据或在步骤 **S36** 中读取数据。

在步骤 **S35** 中由 CPU **31** 执行的程序中的下一处理涉及写入数据的情况下，CPU **31** 确定写入数据，流程进行到步骤 **S37**。

在步骤 S37 中，CPU 31 给出写入数据的指令。更具体地，CPU 31 经由地址总线 35 向地址总线置乱电路 52 提供表示逻辑数据写入位置的逻辑地址的逻辑地址信号，并经由数据总线 37 向存储器 33 提供包括将被写入的数据和表示写入数据的指令的写入信号。

在步骤 S38 中，地址总线置乱电路 52 将逻辑地址转换为物理地址。具体来说，地址总线置乱电路 52 一位接一位地计算由逻辑地址信号表示的逻辑地址与置乱密钥缓存器 61 中保存的置乱密钥的异或来置乱逻辑地址，从而将逻辑地址转换为物理地址。地址总线置乱电路 52 经由地址总线 36 向存储器 33 提供表示转换后的物理地址的物理地址信号。

在步骤 S39 中，存储器 33 写入数据。具体来说，存储器 33 将从 CPU 31 提供的写入信号中包含的数据写入到存储器 33 上的物理地址，该地址由物理地址信号表示。从而，即使在 CPU 31 给出将数据写入连续的逻辑地址的指令的情况下，数据也实际上被写入存储器 33 上随机安排的位置。从而，难于分析或篡改存储器 33 中存储的数据。

在置乱密钥的连续低阶位是零的情况下，与连续为零的位相对应的逻辑地址低阶位不转换为物理地址而进行分配。因此，在存储器 33 上低阶位没有转换的范围中，数据以与逻辑地址相同的序列排列。例如，在置乱密钥的三个连续的低阶位为零的情况下，逻辑地址的三个低阶位不转换为物理地址而进行分配，而且，在存储器 33 上低阶位没有转换的范围中，数据以与逻辑地址相同的序列排列。因此，数据更有可能被分析。与此相反，如上文所描述的，置乱密钥缓存器 61 中保存的置乱密钥的 LSB 被固定为 1，因此逻辑地址的 LSB 始终被置乱。因此，在存储器上，避免了数据以与逻辑地址相同的序列排列，从而能更可靠地使数据变得更难于分析。

通过将置乱密钥的固定值设置为只包括 1 的比特流，数据流能够被可靠地置乱，并以更详细的方式排列，从而数据变得更加难于分析。

之后，流程返回步骤 **S35**，执行步骤 **S35** 以下的处理。

在步骤 **S36** 中，在由 CPU **31** 执行的程序中的下一处理涉及读取数据的情况下，CPU **31** 确定读取数据，流程进行到步骤 **S40**。

在步骤 **S40** 中，CPU **31** 给出读取数据的指令。具体来说，CPU **31** 经由地址总线 **35** 向地址总线置乱电路 **52** 提供表示逻辑数据读取位置的逻辑地址的逻辑地址信号，并经由数据总线 **37** 向存储器 **33** 提供表示数据读取指令的读取信号。

在步骤 **S41** 中，与上文描述的步骤 **S38** 中的处理一样，逻辑地址被转换成物理地址，并经由地址总线 **36** 将表示转换后的物理地址的物理地址信号从地址总线置乱电路 **52** 提供给存储器 **33**。

在步骤 **S42** 中，存储器 **33** 读取数据。具体来说，存储器 **33** 读取存储在由物理地址信号表示的物理地址上的数据，并经由数据总线 **37** 向 CPU **31** 提供读出的数据。

之后，流程返回步骤 **S35**，执行步骤 **S35** 以下的处理。

如上文所描述的，能够容易地为不同的控制模块 **13** 设置不同的置乱密钥。即使在设置给一个控制模块 **13** 的置乱密钥被分析出来的情况下，也能防止使用该置乱密钥分析或篡改存储在另一控制模块 **13** 的存储器 **33** 中的数据。因此，由于数据泄漏或篡改造成的破坏能够被保持在最小。

在执行伪随机数生成方法和地址置乱方法中可采用现有的技术。由于不需要新的复杂电路，而且用户只需执行输入改变置乱密钥的指令的附加步骤，因而能够容易地提高存储器 **33** 中的数据安全性。

如上所述，防止了数据在存储器 **33** 中以与逻辑地址相同的序列排列，因此能更可靠地使数据变得更难于分析。

下面参照图 8 和图 9，描述根据本发明第二实施例的随机数输出单元 **42**。

图 8 是示出根据第二实施例的随机数输出单元 **42** 的功能结构的框图。图 8 所示的随机数输出单元 **42** 包括随机数生成器 **101**、比特流检查器 **201**、开关 **202**、包括 n－p 位移位寄存器的随机数存储单元 **203** 和开关 **204**。在图 8 中，对应于图 3 的部分使用相同的附图标号表示，为避免冗余而省略了执行相同处理的部分的描述。

比特流检查器 **201** 从置乱密钥变更指示单元 **41** 获得表示改变置乱密钥的指令的信号。在从置乱密钥变更指示单元 **41** 提供表示改变置乱密钥的指令的信号的情况下，比特流检查器 **201** 接通开关 **202**。因此，表示由随机数生成器 **101** 生成的 Gold 序列伪随机数的比特流经由开关 **202** 从随机数生成器 **101** 提供至随机数存储单元 **203**，并被存储在随机数存储单元 **203** 中。

比特流检查器 **201** 检查随机数存储单元 **203** 中存储的伪随机数是否与任意预定的禁用值一致。在随机数存储单元 **203** 中存储的伪随机数与一禁用值一致的情况下，比特流检查器 **201** 接通开关 **202**，并从随机数输出单元 **101** 向随机数存储单元 **203** 输出包括预定数目位的伪随机数，从而改变存储在随机数存储单元 **203** 中的伪随机数的值。在随机数存储单元 **203** 中存储的伪随机数与任何禁用值都不

一致的情况下，比特流检查器 **201** 接通开关 **204**。从而，经由开关 **204** 向置乱密钥缓存器 **61** 输出存储在随机数存储单元 **203** 中的包括 n－p 位比特流的伪随机数。即，在由随机数生成器 **101** 生成的伪随机数等于预定的禁用值的情况下，比特流检查器 **201** 控制随机数生成器 **101** 以生成新的随机数，并将这个不同于禁用值的随机值输出至置乱密钥缓存器 **61**。

接下来，参照图 9 的流程图，描述在读/写器 **1** 中设置了图 8 所示的随机数输出单元 **42** 的情况下，由读/写装置 **1** 执行的不同于图 6 的流程图的置乱密钥生成处理。当例如在读/写器 **1** 的电源打开的情况下用户通过置乱密钥变更指示单元 **41** 输入改变置乱密钥的指令时，该处理开始。

在步骤 **S101** 中，随机数输出单元 **42** 生成伪随机数。具体来说，置乱密钥变更指示单元 **41** 向比特流检查器 **201** 提供表示改变置乱密钥的指令的信号。比特流检查器 **201** 接通开关 **202**。当读/写器 **1** 的电源打开时，随机数生成器 **101** 不断地生成伪随机数。通过接通开关 **202**，随机数生成器 **101** 开始经由开关 **202** 向随机数存储单元 **203** 输出伪随机数。在随机数生成器 **101** 输出 n－p 位的伪随机数的情况下，比特流检查器 **201** 断开开关 **202**。

在步骤 **S102** 中，比特流检查器 **201** 确定伪随机数是否为禁用值。例如，比起其他值来说可能更容易预测的值，例如包括相同连续值的比特流（例如，111...111），或具有交替不同值的比特流（例如，0101...0101 或 1010...1010），被用户作为禁止用作置乱密钥的值而预先设置在比特流检查器 **201** 中。在通过从这些禁用值的每一个中除去置乱密钥的低阶固定值而获得的值与存储在随机数存储单元 **203** 中的伪随机数一致的情况下，比特流检查器 **201** 确定伪随机数是禁用值，流程进行到步骤 **S103**。

在步骤 **S103** 中，比特流检查器 **201** 生成新的伪随机数。具体来说，比特流检查器 **201** 接通开关 **202**，并从随机数生成器 **101** 向随机数存储单元 **203** 输出包括预定数目位的伪随机数。随机数存储单元 **203** 将存储的比特流向上移位新输入的伪随机数的位的数目，并将输入的伪随机数加到该比特流的末尾。即，由随机数生成器 **101** 生成的新的伪随机数被存储在随机数存储单元 **203** 中。

之后，流程返回到步骤 **S102**。重复步骤 **S102** 和 **S103** 中的处理，直到在步骤 **S102** 中确定伪随机数不是禁用值。

在步骤 **S102** 中确决定伪随机数不是禁用值的情况下，处理进行到步骤 **S104**。

在步骤 **S104** 中，随机数输出单元 **42** 输出伪随机数。具体来说，比特流检查器 **201** 接通开关 **204**。从而，存储在随机数存储单元 **203** 中的伪随机数经由开关 **204** 输出至置乱密钥缓存器 **61**。

在步骤 **S105** 中，与图 6 所示的步骤 **S2** 中的上述处理一样，设置置乱密钥，置乱密钥生成处理结束。

由于以上面描述的方式避免了容易预测的值被设置为置乱密钥，因而难于分析或篡改存储器 **33** 中存储的数据，从而加强了存储器 **33** 中数据的安全性。此外，通过在例如更换或初始化存储器 **33** 时改变置乱密钥，使得置乱密钥变得更加难于分析。

在上文的描述中，已经描述了 Gold 序列伪随机数被用作置乱密钥的情况。然而，用作置乱密钥的随机数或伪随机数并不限于上述举例。例如，可以使用通过仅使用一个 LFSR 生成的 M 序列伪随机数或利用热噪声的物理随机数。

置乱地址的方法并不限于上述举例。也可以采用使用通过随机数或伪随机数设置的置乱密钥的其他方法。

在上文的描述中，IC 卡 **2** 被描述为读/写器 **1** 的通信伙伴。勿庸置疑，读/写器 **1** 可与具有非接触 IC 卡功能的装置进行通信，例如移动电话、个人数字助理（PDA）、计时器以及具有非接触 IC 卡功能的计算机。

除读/写器外，图 2 所示的存储器存取控制器 **32** 还可以应用于从/向存储器读/写数据的其他装置。

在图 8 所示的随机数输出单元 **42** 中，除上文描述的禁止输出容易预测的值作为置乱密钥之外，还可以根据应用任意设置禁止被输出的值。

尽管在上文中描述了图 2 所示的存储器 **33** 是非易失性存储器的情况，但毫无疑问，存储器存取控制器 **32** 也可以用于控制易失性存储器。

可以允许用户设置置乱密钥的固定值的 LSB 以外的值。

进一步，可以允许用户设置置乱密钥的固定值以外的可变值。

本领域技术人员应该明白，根据设计要求和其他因素，在所附的权利要求或其等同范围以内，本发明可以进行各种修改、组合、自组合和变更。

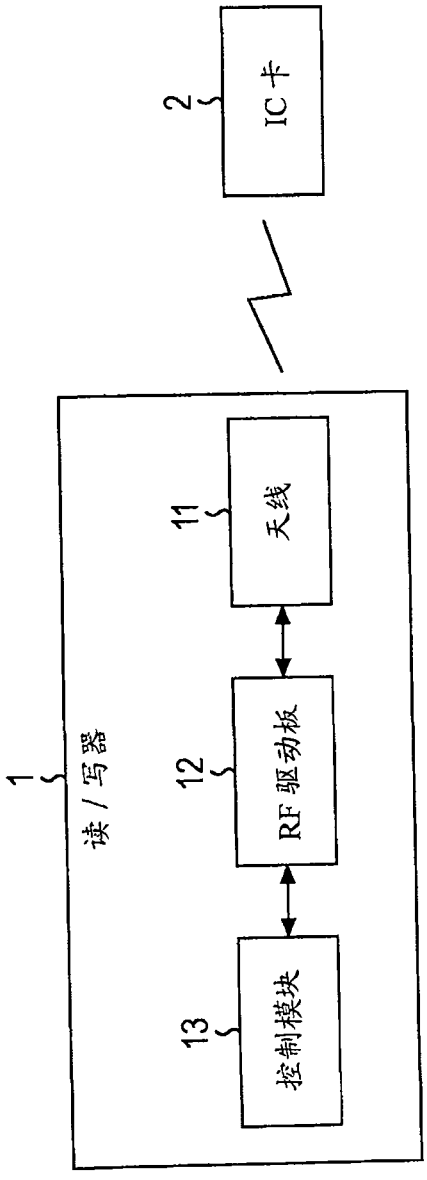


图 1

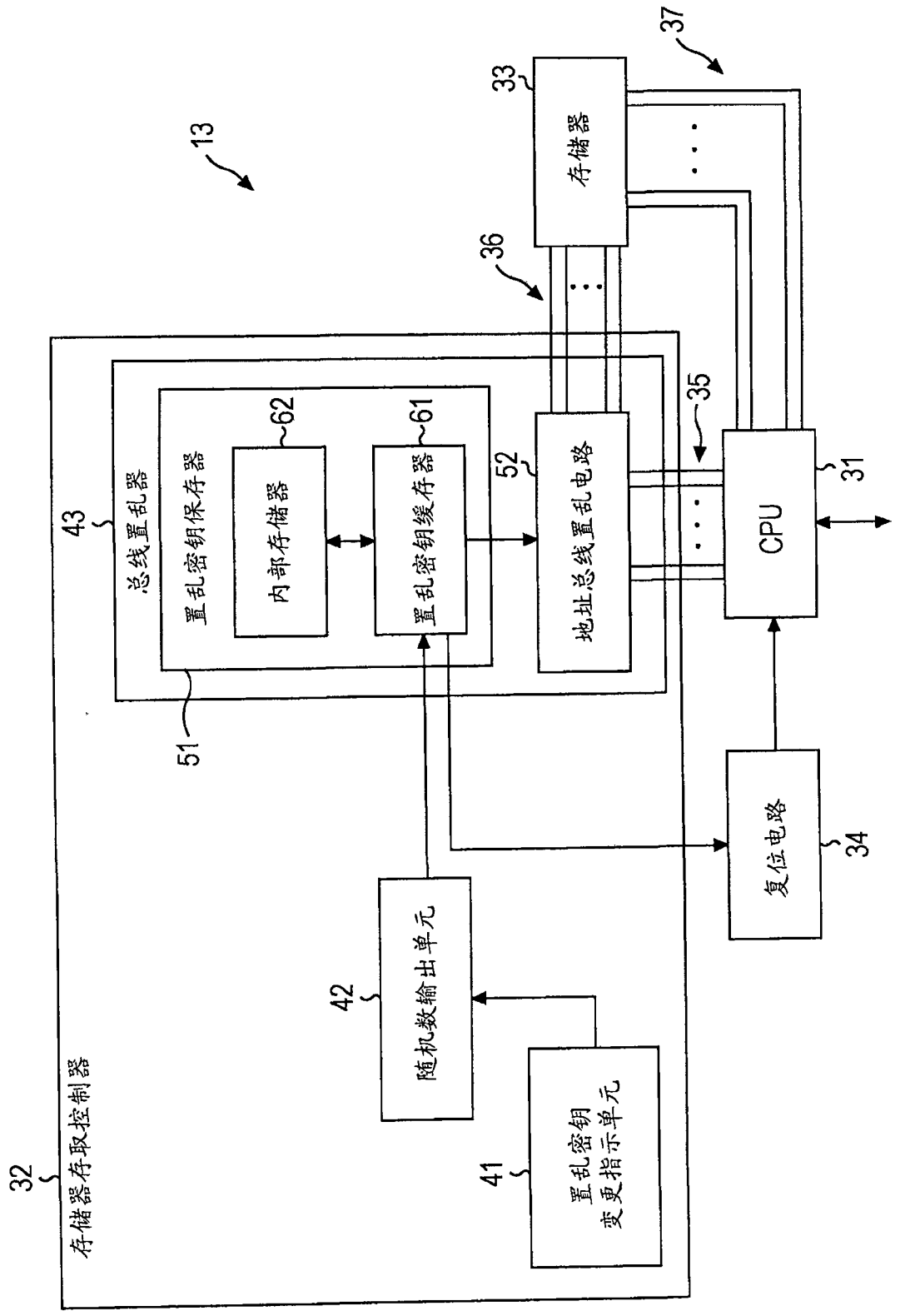

13
37
33
存储器
36
35
43
总线置乱器
置乱密钥保存器
62
内部存储器
61
置乱密钥缓存器
52
地址总线置乱电路
CPU
31
51
复位电路
34
42
随机数输出单元
41
置乱密钥
变更指示单元
32
存储器存取控制器

图 2

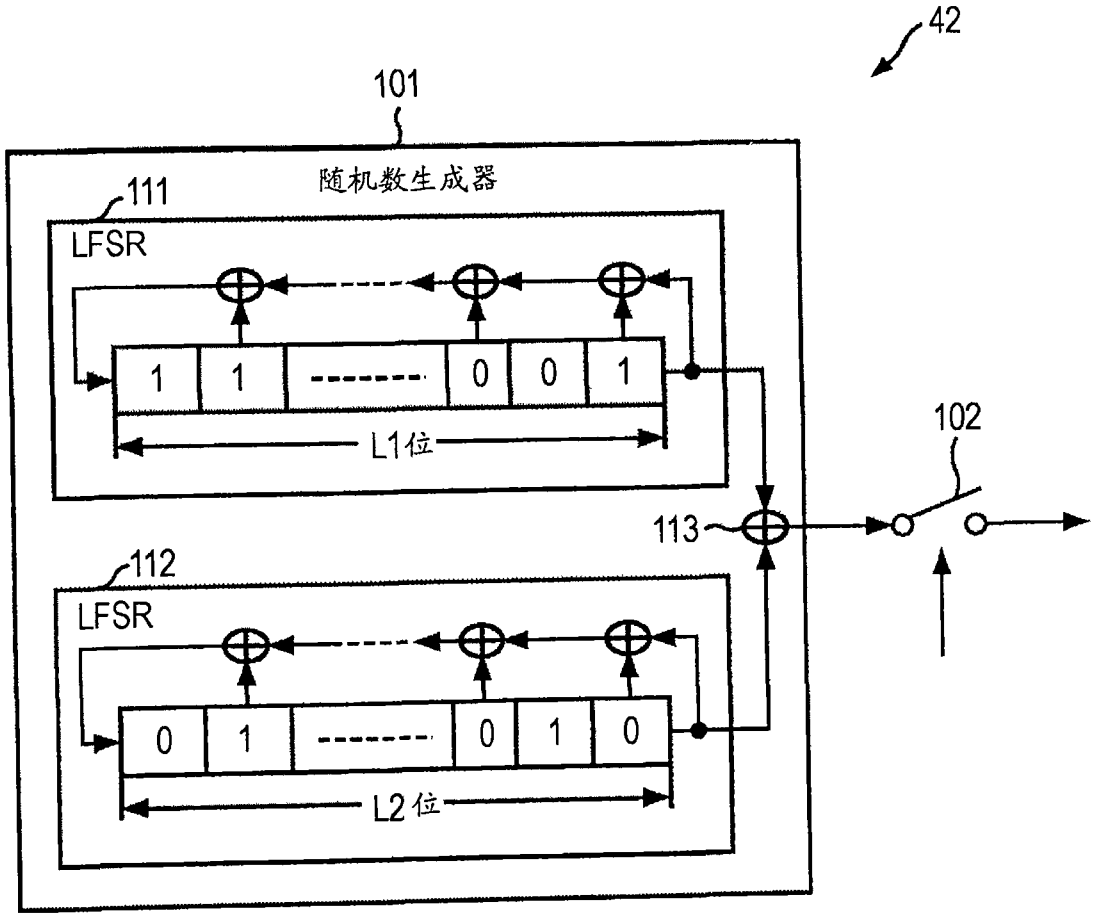


图 3

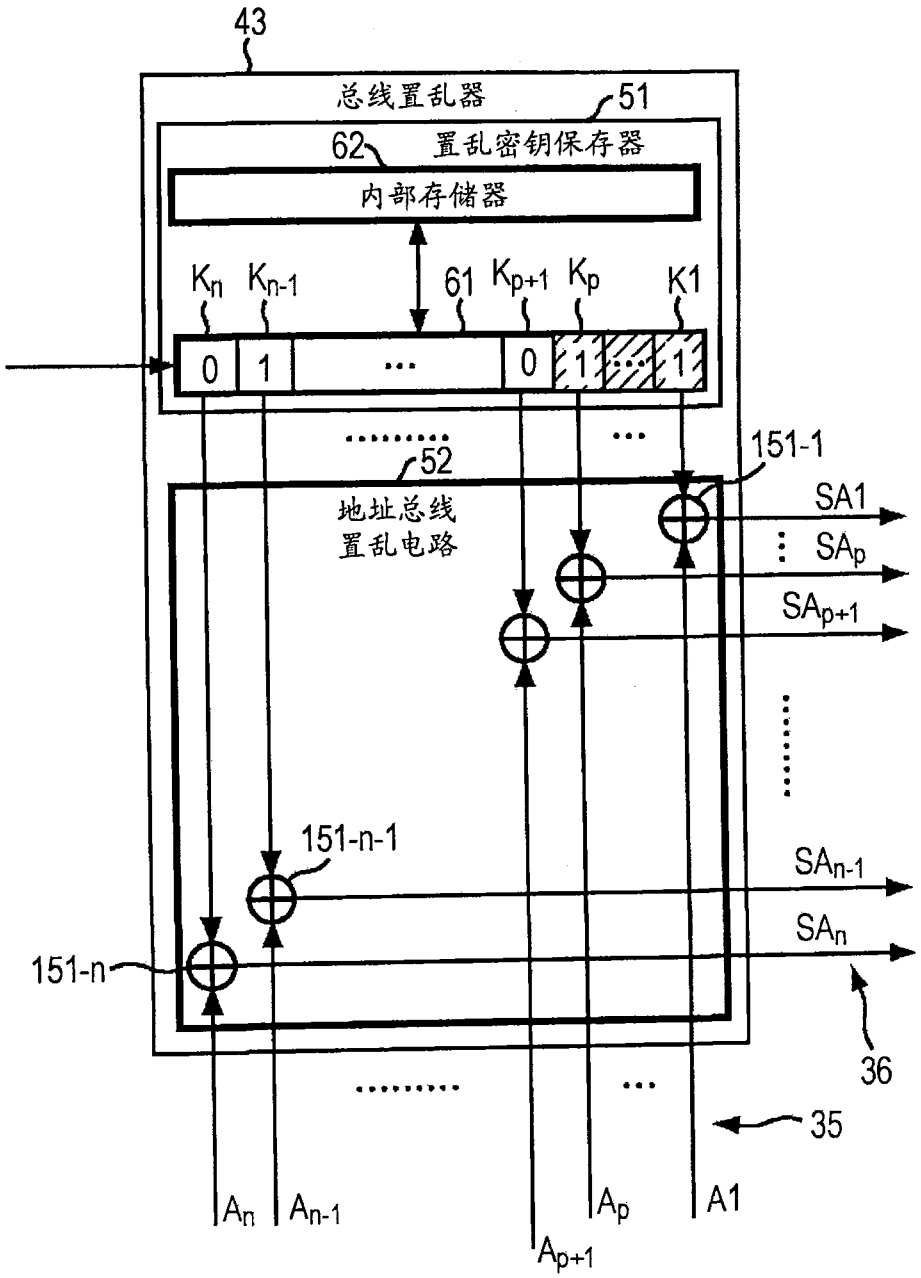

43
总线置乱器
51
62
置乱密钥保存器
内部存储器
Kn
Kn-1
61
Kp+1
Kp
K1
0
1
0
1
1
52
地址总线
置乱电路
151-1
SA1
SAp
SAp+1
151-n-1
SAn-1
SAn
151-n
36
35
An
An-1
Ap
A1
Ap+1

图 4

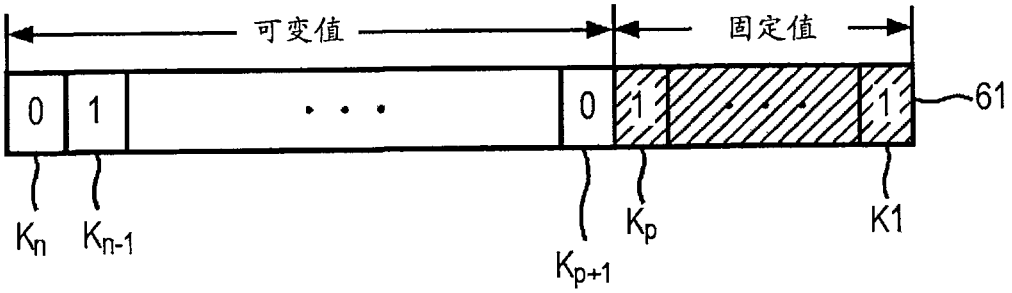

可变值
固定值
0
1
0
1
1
61
Kn
Kn-1
Kp+1
Kp
K1

图 5

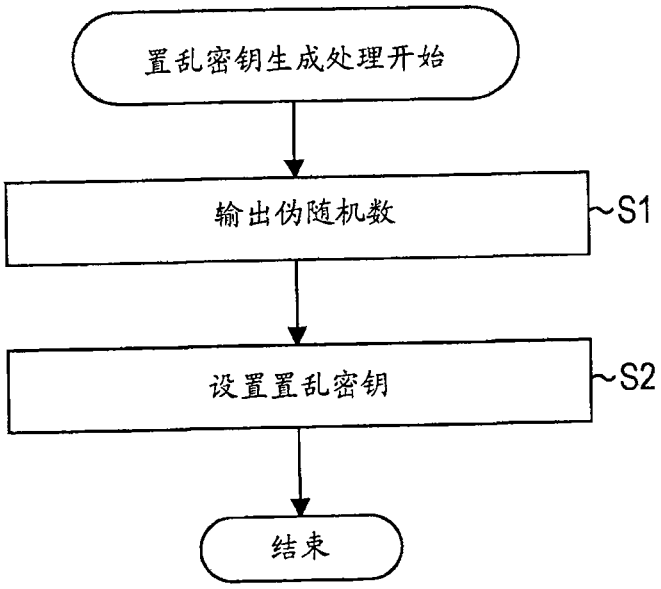

置乱密钥生成处理开始
输出伪随机数
S1
设置置乱密钥
S2
结束

图 6

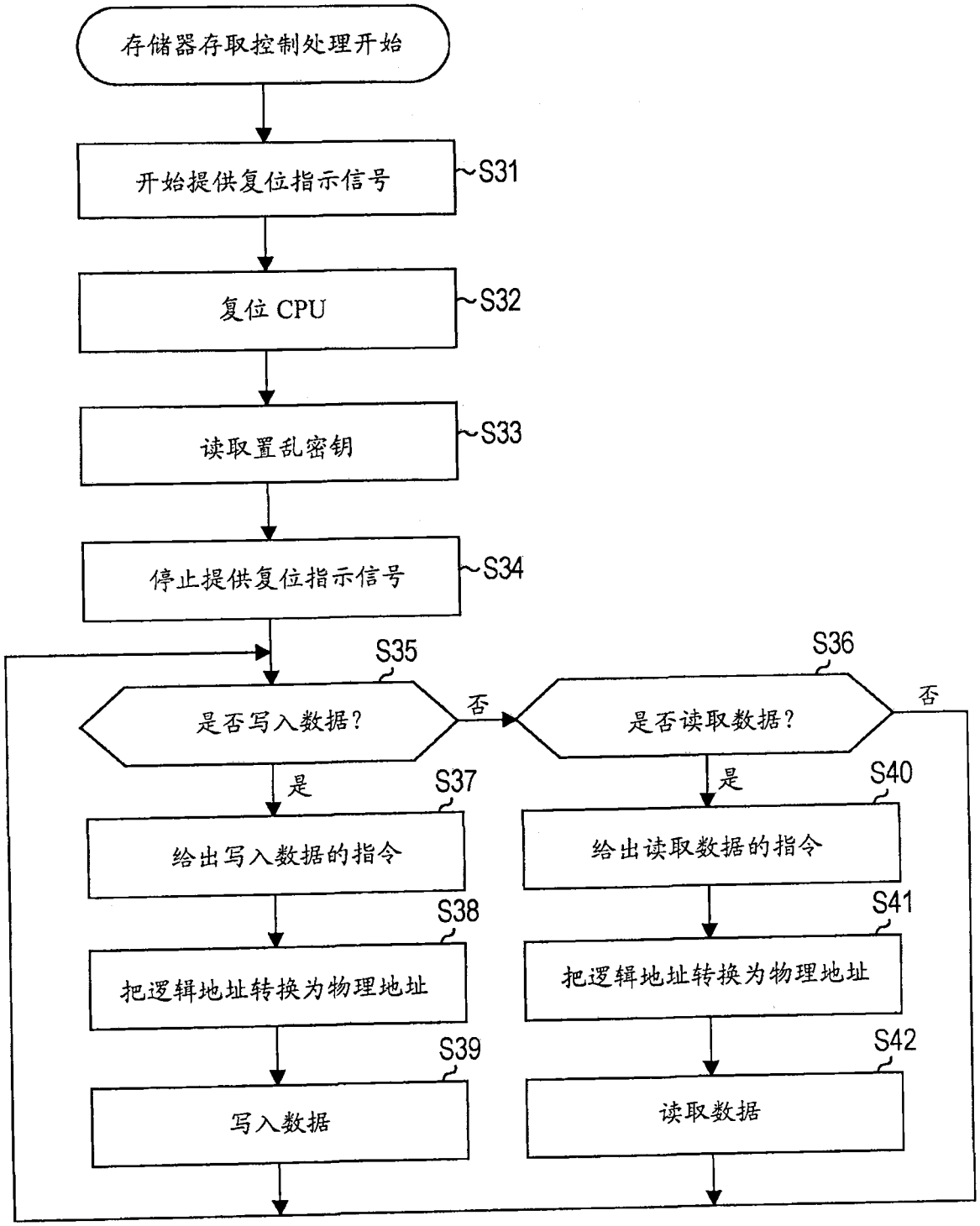

存储器存取控制处理开始
开始提供复位指示信号
S31
复位 CPU
S32
读取置乱密钥
S33
停止提供复位指示信号
S34
S35
是否写入数据？
否
S36
是否读取数据？
否
是
S37
给出写入数据的指令
S38
把逻辑地址转换为物理地址
S39
写入数据
是
S40
给出读取数据的指令
S41
把逻辑地址转换为物理地址
S42
读取数据

图 7

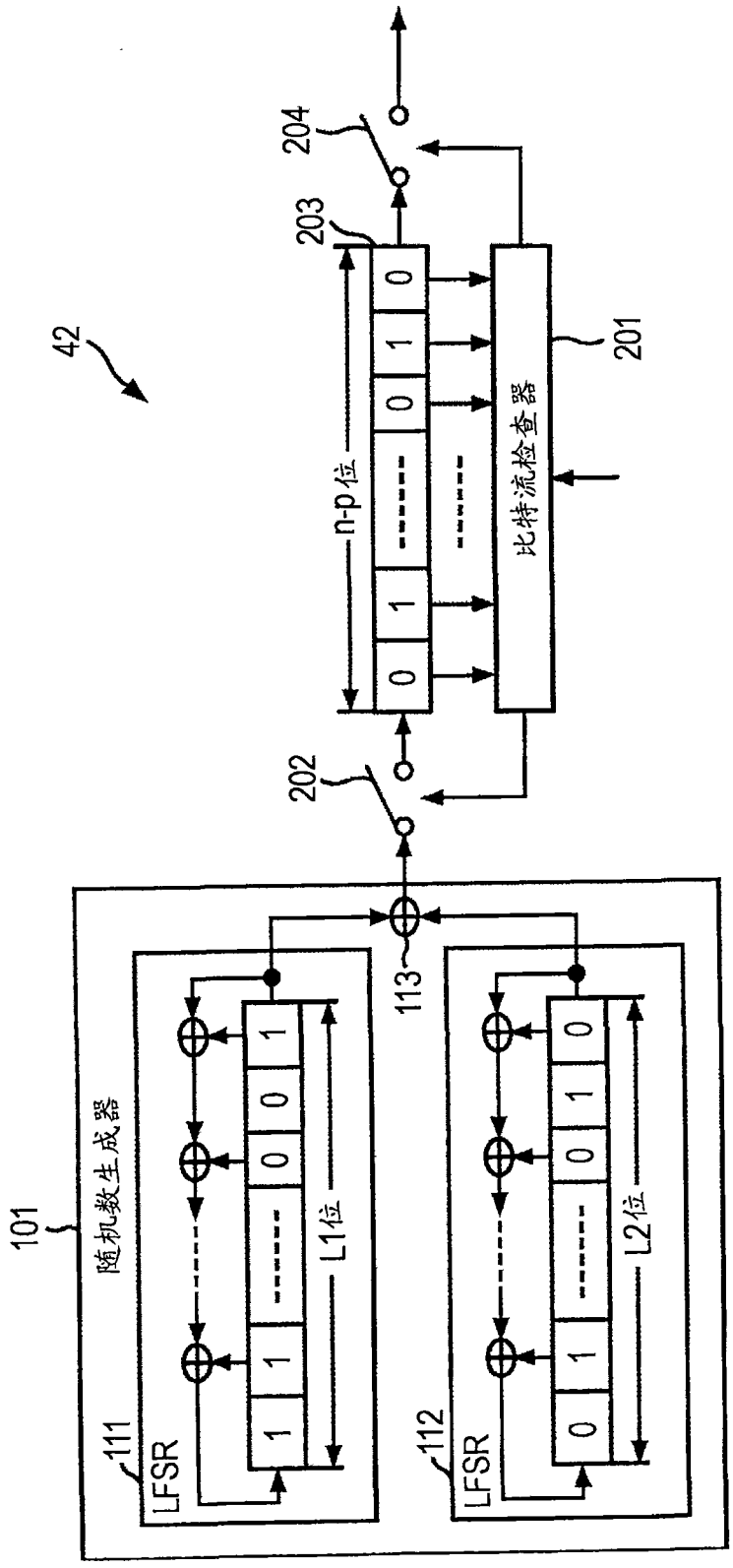

42
204
203
n-p位
201
比特流检查器
202
101
随机数生成器
113
111
LFSR
L1位
112
LFSR
L2位

图 8

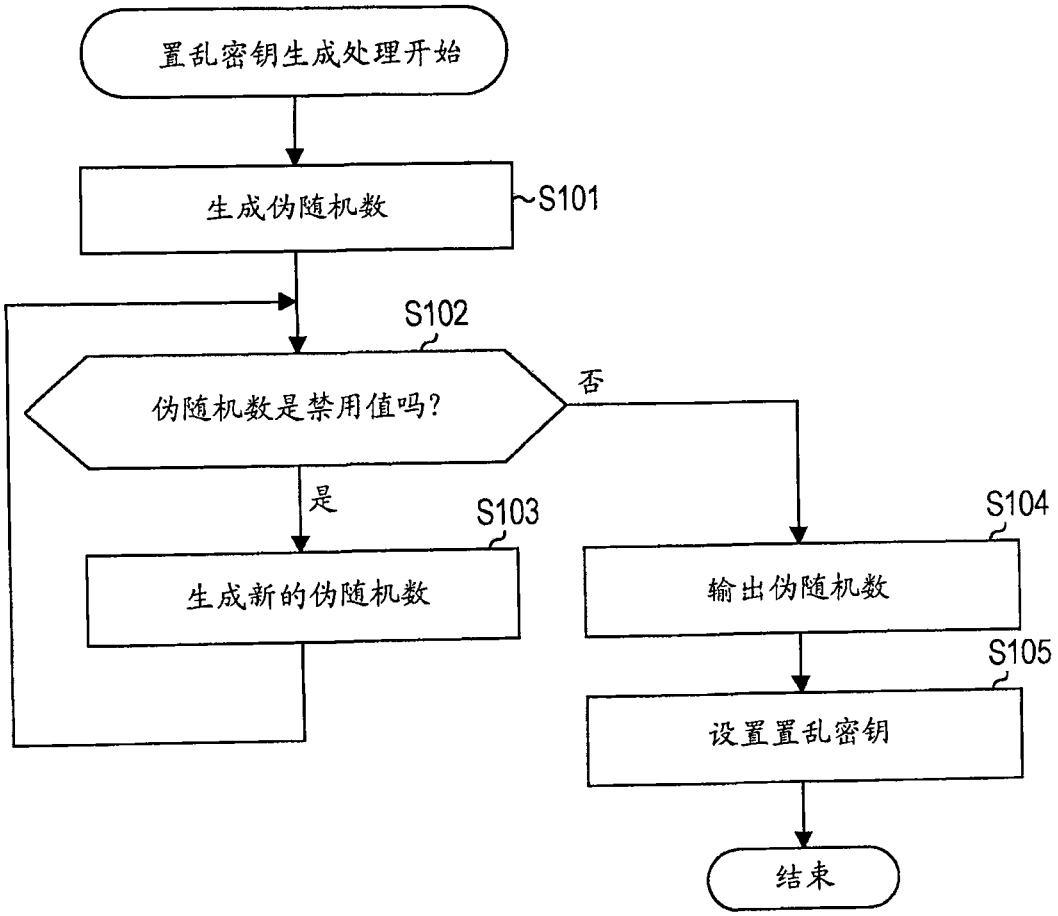

置乱密钥生成处理开始
生成伪随机数
S101
S102
伪随机数是禁用值吗?
否
是
S103
生成新的伪随机数
S104
输出伪随机数
S105
设置置乱密钥
结束

图 9