(19)中华人民共和国国家知识产权局

(12)发明专利

(10)授权公告号 CN 101536062 B
(45)授权公告日 2013.09.18

(21)申请号 200780040806.1
(22)申请日 2007.08.29
(30)优先权数据
238225/2006 2006.09.01 JP
(85)PCT申请进入国家阶段日
2009.04.30
(86)PCT申请的申请数据
PCT/JP2007/066730 2007.08.29
(87)PCT申请的公布数据
WO2008/026622 JA 2008.03.06
(73)专利权人 索尼株式会社
地址 日本东京都
(72)发明人 白井太三 涩谷香士 秋下彻 盛合志帆
(74)专利代理机构 北京林达刘知识产权代理事务所(普通合伙) 11277
代理人 刘新宇
(51)Int.Cl.
H04L 9/06(2006.01)
G09C 1/00(2006.01)
(56)对比文件
US 5745577 A,1998.04.28, 说明书第3栏第28行至第33行,第4栏第29行至第49行,第1栏第43-45行,附图5,6.
US 5003597 A,1991.03.26, 全文.
US 3962539 A,1976.06.08, 全文.
CN 1364284 A,2002.08.14, 全文.
审查员 张飞弦

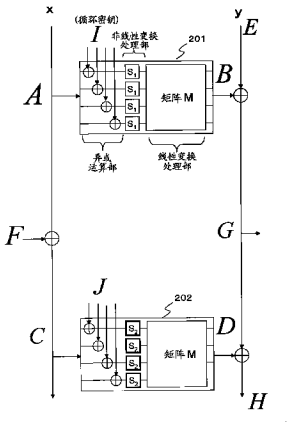
权利要求书2页 说明书24页 附图19页

(54)发明名称

密码处理装置和密码处理方法

(57)摘要

实现一种提高了对饱和攻击、代数攻击(XSL攻击)等攻击的抵抗性的共用密钥块密码处理结构。在执行共用密钥块密码处理的密码处理装置中,设为在设定于循环函数执行部中的作为非线性变换处理部的S盒中至少利用了两种以上多个不同的S盒的结构。根据本结构,能够提高对饱和攻击的抵抗性。另外,作为S盒的类型使不同类型的S盒共存。根据本结构,能够提高对代数攻击(XSL攻击)的抵抗性,实现安全性较高的密码处理装置。



CN 101536062 B

1. 一种密码处理装置，其特征在于，

具有密码处理部，该密码处理部将利用包含应用于非线性变换处理的 S 盒的 F 函数进行的数据变换处理作为循环函数，对利用 2 以上的数分割输入数据而得到的各数据序列分别执行上述数据变换处理，

上述密码处理部是如下结构：

输入数据序列和输出数据序列相同的 F 函数利用了至少两种以上多个不同的 S 盒，以及

作为应用于上述非线性变换处理的具有 s 比特的输入输出的不同的 S 盒，利用如下 (1) ～ (3) 三种类型的 S 盒中至少两种以上不同类型的 S 盒：

(1) 类型 1：利用了扩展域 GF(2^s) 上的逆映射：$Y = X^{-1}$ 或者幂函数 $Y = X^q$ 的 S 盒；

(2) 类型 2：组合多个 t-bit 的较小的 S-box 而制作出的 S 盒，其中，设为 $t < s$；以及

(3) 类型 3：随机选择的 S 盒。

2. 根据权利要求 1 所述的密码处理装置，其特征在于，

上述密码处理部是如下结构：

执行应用了 Feistel 结构或者扩展 Feistel 结构的密码处理，其中，上述 Feistel 结构将数据序列数设为 2，上述扩展 Feistel 结构将数据序列数设为 2 以上，

进行应用了 F 函数的密码处理，该 F 函数将在输入数据序列和输出数据序列相同并且位于上下相邻位置的、作为循环函数执行部的 F 函数中的非线性变换处理设定为执行不同的非线性变换处理的不同的 S 盒。

3. 根据权利要求 2 所述的密码处理装置，其特征在于，

作为上述循环函数执行部的各 F 函数具有多个 S 盒，该多个 S 盒对分割处理对象数据得到的各分割数据执行非线性变换处理，

该多个 S 盒由至少两种以上不同的 S 盒构成。

4. 根据权利要求 2 所述的密码处理装置，其特征在于，

作为上述循环函数执行部的各 F 函数具有多个 S 盒，该多个 S 盒对分割处理对象数据得到的各分割数据执行非线性变换处理，

上述密码处理部是如下结构：以各分割数据为单位，将输入数据序列和输出数据序列相同并且位于上下相邻位置的、作为循环函数执行部的 F 函数中的非线性变换处理设定为执行不同的非线性变换处理的不同的 S 盒。

5. 根据权利要求 2 所述的密码处理装置，其特征在于，

包含在作为上述循环函数执行部的各 F 函数中的 S 盒的种类和各 S 盒的数量在各 F 函数中是相同的设定。

6. 根据权利要求 1 所述的密码处理装置，其特征在于，

上述密码处理部是应用于上述循环函数的执行的 S 盒具有如下 (a) ～ (d) 中的任一个的结构：

(a) 将一部分设为类型 1 的 S 盒，将其它设为类型 2 的 S 盒的结构；

(b) 将一部分设为类型 1 的 S 盒，将其它设为类型 3 的 S 盒的结构；

(c) 将一部分设为类型 2 的 S 盒，将其它设为类型 3 的 S 盒的结构；以及

(d) 将一部分设为类型 1 的 S 盒，将另一部分设为类型 2 的 S 盒，将其它设为类型 3 的

S 盒的结构。

7. 根据权利要求 1 所述的密码处理装置，其特征在于，

上述密码处理部是如下结构：

在循环函数执行部内具有多个 S 盒，该多个 S 盒对分割处理对象数据得到的各分割数据执行非线性变换处理，

在一个循环中利用一种类型的 S 盒，以循环为单位执行利用了不同类型的 S 盒的处理。

8. 根据权利要求 1 所述的密码处理装置，其特征在于，

上述密码处理部是如下结构：

在循环函数执行部内具有多个 S 盒，该多个 S 盒对分割处理对象数据得到的各分割数据执行非线性变换处理，

在一个循环内利用多个类型的 S 盒。

9. 根据权利要求 8 所述的密码处理装置，其特征在于，

包含在各上述循环函数执行部中的 S 盒的种类和各 S 盒的数量在各 F 函数中是相同的设定。

10. 根据权利要求 1 ～ 9 中的任一项所述的密码处理装置，其特征在于，

上述密码处理部是执行按照共用密钥密码方式的密码处理的结构。

11. 根据权利要求 1 ～ 9 中的任一项所述的密码处理装置，其特征在于，

上述密码处理部是执行按照共用密钥块密码方式的密码处理的结构。

12. 根据权利要求 1 所述的密码处理装置，其特征在于，对输入数据进行 4 分割，从而得到 4 个数据序列。

13. 一种密码处理方法，在密码处理装置中执行密码处理，该密码处理方法的特征在于，

具有密码处理步骤，该密码处理步骤在密码处理部中将利用包含应用于非线性变换处理的 S 盒的 F 函数进行的数据变换处理作为循环函数，对利用 2 以上的数分割输入数据而得到的各数据序列分别执行上述数据变换处理，

在上述密码处理步骤中：

输入数据序列和输出数据序列相同的 F 函数利用了至少两种以上多个不同的 S 盒；

其中，作为应用于上述非线性变换处理的具有 s 比特的输入输出的不同的 S 盒，利用如下 (1) ～ (3) 三种类型的 S 盒中至少两种以上不同类型的 S 盒：

(1) 类型 1：利用了扩展域 GF(2^s) 上的逆映射：$Y = X^{-1}$ 或者幂函数 $Y = X^q$ 的 S 盒；

(2) 类型 2：组合多个 t-bit 的较小的 S-box 而制作出的 S 盒，其中，设为 $t < s$；以及

(3) 类型 3：随机选择的 S 盒。

14. 根据权利要求 13 所述的密码处理方法，其特征在于，

上述密码处理步骤执行按照共用密钥密码方式或者共用密钥块密码方式的密码处理。

密码处理装置和密码处理方法

技术领域

[0001] 本发明涉及一种密码处理装置和密码处理方法、以及计算机程序。更详细地说，涉及一种执行共用密钥块密码处理的密码处理装置和密码处理方法、以及计算机程序。

背景技术

[0002] 最近，随着网络通信、电子商务的发展，确保通信中的安全成为重要的问题。确保安全的方法之一是密码技术，当前实际进行着使用了各种加密方法的通信。

[0003] 例如实际应用有如下的系统：在IC卡等小型装置中嵌入密码处理模块，在IC卡与作为数据读取写入装置的读写器之间进行数据的发送接收，进行认证处理或者发送接收数据的加密、解密。

[0004] 在密码处理算法中存在各种算法，大致分类为公开密钥密码方式和共用密钥密码方式，其中，上述公开密钥密码方式将加密密钥和解密密钥设定为不同的密钥、例如公开密钥和私人密钥，上述共用密钥密码方式将加密密钥和解密密钥设定为共用的密钥。

[0005] 在共用密钥密码方式中也存在各种算法，其中之一是如下的方式：以共用密钥为基础生成多个密钥，使用所生成的多个密钥来反复执行块单位（64比特、128比特等）的数据变换处理。应用了这种密钥生成方式和数据变换处理的算法的代表方式是共用密钥块密码方式。

[0006] 作为代表的共用密钥块密码的算法，例如已知过去作为美国标准密码的DES(Data Encryption Standard：数据加密标准）算 法、现在作为美国标准密码的AES(Advanced Encryption Standard：高级加密标准）算法等。

[0007] 这种共用密钥块密码的算法主要由密码处理部和密钥调度部构成，其中，上述密码处理部具有反复执行输入数据的变换的循环函数执行部，上述密钥调度部生成在循环函数部的各循环中应用的循环密钥。密钥调度部根据作为私人密钥的主密钥(Master key)首先生成增加了比特数的扩展密钥，根据所生成的扩展密钥生成在密码处理部的各循环函数部中应用的循环密钥（副密钥）。

[0008] 作为执行这种算法的具体结构，已知反复执行具有线性变换部和非线性变换部的循环函数的结构。例如代表的结构有Feistel结构。Feistel结构具有如下结构：通过作为数据变换函数的循环函数（F函数）的简单的反复，将明文变换为密文。在循环函数（F函数）中执行线性变换处理和非线性变换处理。此外，作为记载有应用了Feistel结构的密码处理的文献，例如存在非专利文献1、非专利文献2。

[0009] 但是，作为该共用密钥块密码处理的问题点，存在由密码分析引起的密钥泄漏。利用密码分析容易进行密钥的分析导致该密码处理的安全性较低，实用上成为较大的问题。

[0010] 非专利文献1：K.Nyberg，‘Generalized Feistel networks’，ASIACRYPT’96，Springer Verlag，1996，pp.91--104.

[0011] 非专利文献2：Yuliang Zheng，Tsutomu Matsumoto，Hideki Imai：On the Construction of Block Ciphers Provably Secure and Not Relying on Any Unproved

Hypotheses. CRYPTO1989 :461-480

发明内容

[0012] 发明要解决的问题

[0013] 本发明是鉴于上述问题点而完成的，目的在于提供一种提高密码分析的难度、实现安全性较高的共用密钥块密码算法的密码处理装置和密码处理方法、以及计算机程序。

[0014] 用于解决问题的方案

[0015] 本发明的第一侧面在于一种密码处理装置，执行共用密钥块密码处理，其特征在于，

[0016] 具有密码处理部，该密码处理部进行将循环函数反复多个循环的数据变换处理，

[0017] 上述密码处理部是如下结构 ：

[0018] 在各循环中的循环函数执行处理中进行应用了 S 盒的非线性变换处理，

[0019] 进行作为应用于非线性变换处理的 S 盒而利用了至少两种以上的多个不同的 S 盒的处理。

[0020] 并且，在本发明的密码处理装置的一个实施方式中，其特征在于，上述密码处理部是如下结构 :执行应用了将数据序列数（分割数）设为 2 的 Feistel 结构或者将数据序列数（分割数）设为 2 以上的扩展 Feistel 结构的密码处理，进行应用了 F 函数的密码处理，该 F 函数将在输入数据序列和输出数据序列相同并且位于上下相邻位置的、作为循环函数执行部的 F 函数中的非线性变换处理设定为执行不同的非线性变换处理的不同的 S 盒。

[0021] 并且，在本发明的密码处理装置的一个实施方式中，其特征在于，作为上述循环函数执行部的各 F 函数具有多个 S 盒，该多个 S 盒对分割处理对象数据得到的各分割数据执行非线性变换处理，该多个 S 盒由至少两种以上不同的 S 盒构成。

[0022] 并且，在本发明的密码处理装置的一个实施方式中，其特征在于，作为上述循环函数执行部的各 F 函数具有多个 S 盒，该 多个 S 盒对分割处理对象数据得到的各分割数据执行非线性变换处理，上述密码处理部是如下结构 :以各分割数据为单位，将在输入数据序列和输出数据序列相同并且位于上下相邻位置的、作为循环函数执行部的 F 函数中的非线性变换处理设定为执行不同的非线性变换处理的不同的 S 盒。

[0023] 并且，在本发明的密码处理装置的一个实施方式中，其特征在于，包含在作为上述循环函数执行部的各 F 函数中的 S 盒的种类和各 S 盒的数量在各 F 函数中是相同的设定。

[0024] 并且，在本发明的密码处理装置的一个实施方式中，其特征在于，上述密码处理部是如下结构 :作为应用于上述非线性变换处理的不同的具有 s 比特输入输出的 S 盒，利用如下 (1) ～ (3) 三种类型的 S 盒中至少两种以上不同类型的 S 盒 ：

[0025] (1) 类型 1 :利用了扩展域 $GF(2^s)$ 上的逆映射 $:Y = X^{-1}$ 或者幂函数 $Y = X^p$ 的 S 盒 ；

[0026] (2) 类型 2 :组合多个 t-bit 的较小的 S-box 而制作出的 S 盒，其中，设为 $t < s$ ；以及

[0027] (3) 类型 3 :随机选择的 S 盒。

[0028] 并且，在本发明的密码处理装置的一个实施方式中，其特征在于，上述密码处理部具有将应用于上述循环函数的执行的 S 盒设为如下 (a) ～ (d) 中的任一个的结构 ：

[0029] (a) 将一部分设为类型 1 的 S 盒，将其它设为类型 2 的 S 盒的结构 ；

[0030] (b) 将一部分设为类型 1 的 S 盒，将其它设为类型 3 的 S 盒的结构；

[0031] (c) 将一部分设为类型 2 的 S 盒，将其它设为类型 3 的 S 盒的结构；以及

[0032] (d) 将一部分设为类型 1 的 S 盒，将另一部分设为类型 2 的 S 盒，将其它设为类型 3 的 S 盒的结构。

[0033] 并且，在本发明的密码处理装置的一个实施方式中，其特征在于，上述密码处理部是如下结构：在循环函数执行部内具有多个 S 盒，该多个 S 盒对分割处理对象数据得到的各分割数据执行非线性变换处理，在一个循环内利用一种类型的 S 盒，以循环为单位执行利用了不同类型的 S 盒的处理。

[0034] 并且，在本发明的密码处理装置的一个实施方式中，其特征在于，上述密码处理部是如下结构：在循环函数执行部内具有多个 S 盒，该多个 S 盒对分割处理对象数据得到的各分割数据执行非线性变换处理，在一个循环内利用多个类型的 S 盒。

[0035] 并且，在本发明的密码处理装置的一个实施方式中，其特征在于，包含在各上述循环函数执行部中的 S 盒的种类和各 S 盒的数量在各 F 函数中是相同的设定。

[0036] 并且，在本发明的密码处理装置的一个实施方式中，其特征在于，上述密码处理部是执行按照共用密钥密码方式的密码处理的结构。

[0037] 并且，在本发明的密码处理装置的一个实施方式中，其特征在于，上述密码处理部是执行按照共用密钥块密码方式的密码处理的结构。

[0038] 并且，本发明的第二侧面在于一种密码处理方法，在密码处理装置中执行共用密钥块密码处理，该密码处理方法的特征在于，

[0039] 具有密码处理步骤，该密码处理步骤在密码处理部中进行将循环函数反复多个循环的数据变换处理，

[0040] 上述密码处理步骤是如下步骤：

[0041] 在各循环中的循环函数执行处理中进行应用了 S 盒的非线性变换处理，进行作为应用于该非线性变换处理的 S 盒而利用 了至少两种以上多个不同的 S 盒的处理。

[0042] 并且，在本发明的密码处理方法的一个实施方式中，其特征在于，上述密码处理步骤执行按照共用密钥密码方式或者共用密钥块密码方式的密码处理。

[0043] 并且，本发明的第三侧面在于一种计算机程序，使密码处理装置执行共用密钥块密码处理，其特征在于，

[0044] 具有密码处理步骤，该密码处理步骤使密码处理部进行将循环函数反复多个循环的数据变换处理，

[0045] 上述密码处理步骤是如下步骤：

[0046] 在各循环中的循环函数执行处理中进行应用了 S 盒的非线性变换处理，进行作为应用于该非线性变换处理的 S 盒而利用了至少两种以上多个不同的 S 盒的处理。

[0047] 并且，在本发明的计算机程序的一个实施方式中，其特征在于，上述密码处理步骤是如下步骤：执行按照共用密钥密码方式或者共用密钥块密码方式的密码处理。

[0048] 此外，本发明的计算机程序例如是能够通过以计算机可读形式提供的存储介质、通信介质、例如 CD、FD、MO 等记录介质或者网络等通信介质对可执行各种程序代码的计算机系统提供的计算机程序。通过以计算机可读形式提供这种程序，在计算机系统上实现与程序相应的处理。

[0049] 通过基于后述的本发明的实施例、附图的更详细的说明，本发明的进一步的其它目的、特征、优点会更清楚。此外，在本说明书中系统是多个装置的逻辑集合结构，并不限于各结构的装置在同一壳体内。

[0050] 发明的效果

[0051] 根据本发明的一个实施例的结构，在执行共用密钥块密码处理的密码处理装置中，设为在设定于循环函数执行部的作为 非线性变换处理部的 S 盒中利用至少两种以上多个不同的 S 盒的结构。根据本结构，能够提高对饱和攻击的抵抗性。另外，根据作为 S 盒的类型而使不同类型的 S 盒共存的本发明的一个实施例的结构，能够提高对代数攻击（XSL 攻击）的抵抗性，实现安全性较高的密码处理装置。

附图说明

[0052] 图 1 是表示共用密钥块密码算法的基本结构的图。

[0053] 图 2 是说明图 1 所示的共用密钥块密码处理部 E10 的内部结构的图。

[0054] 图 3 是说明图 2 所示的密码处理部 12 的详细结构的图。

[0055] 图 4 是说明作为循环函数执行部的一个结构例的 SPN 结构循环函数的图。

[0056] 图 5 是说明作为循环函数执行部的一个结构例的 Feistel（菲斯特尔）结构的图。

[0057] 图 6 是说明作为循环函数执行部的一个结构例的扩展 Feistel 结构的图。

[0058] 图 7 是说明非线性变换处理部的具体例的图。

[0059] 图 8 是说明线性变换处理部的具体例的图。

[0060] 图 9 是说明 Feistel 结构或者扩展 Feistel 结构的一般结构例的图。

[0061] 图 10 是说明在 Feistel 结构或者扩展 Feistel 结构中配置了不同的 S-box 的结构例的图。

[0062] 图 11 是说明通过配置不同的 S-box 来实现提高对饱和攻击的抵抗性的结构例的图。

[0063] 图 12 是说明通过配置不同的 S-box 来实现提高对饱和攻击的抵抗性的结构例的图。

[0064] 图 13 是说明通过配置不同的 S-box 来实现提高对饱和攻击的抵抗性的结构例的图。

[0065] 图 14 是说明通过配置不同类型的 S-box 来实现提高对代数攻击（XSL 攻击）的抵抗性的结构例的图。

[0066] 图 15 是说明通过配置不同类型的 S-box 来实现提高对代数攻击（XSL 攻击）的抵抗性的结构例的图。

[0067] 图 16 是说明通过配置不同类型的 S-box 来实现提高对代数攻击（XSL 攻击）的抵抗性的结构例的图。

[0068] 图 17 是说明通过配置不同类型的 S-box 来实现提高对代数攻击（XSL 攻击）的抵抗性的结构例的图。

[0069] 图 18 是说明通过配置不同类型的 S-box 来实现提高对代数攻击（XSL 攻击）的抵抗性的结构例的图。

[0070] 图 19 是表示作为本发明所涉及的执行密码处理的密码处理装置的 IC 模块的结构

例的图。

具体实施方式

[0071] 下面，详细说明本发明的密码处理装置和密码处理方法、以及计算机程序。按照下面的项目进行说明。

[0072] 1. 共用密钥块密码的概要

[0073] 2. 通过配置多个不同的 S 盒来提高抵抗性的结构

[0074] (2A) 通过在使用了 S-box(S 盒）的 Feistel 或者扩展 Feistel 型密码中配置两种以上不同的 S-box 来提高对饱和攻击的抵抗性的结构

[0075] (2B) 通过在使用了 S-box 的块密码中使两种以上不同的 S-box 共存来提高对代数攻击 (XSL 攻击）的抵抗性的结构

[0076] (2C) 在使用了 S-box 的 Feistel 密码或者扩展 Feistel 型密码中同时实现上述 (2A)、(2B) 的结构

[0077] 3. 密码处理装置的结构例

[0078] [1. 共用密钥块密码的概要]

[0079] 首先，说明能够应用本发明的共用密钥块密码的概要。在本说明书中，共用密钥块密码（下面为块密码）是指如下定义的密码。

[0080] 块密码输入明文 P 和密钥 K，输出密文 C。将明文和密文的比特长度称为块大小，在此用 n 来表示。n 可取任意的整数值，但是通常在每个块密码算法中是预先决定为一个的值。有时也将块长度为 n 的块密码称为 n 比特块密码。

[0081] 用 k 表示密钥的比特长度。密钥可取任意的整数值。共用密钥块密码算法与一个或多个密钥大小对应。例如，某个块密码算法 A 也能是如下结构 :块大小 n ＝ 128，与比特长度 k ＝ 128、或者 k ＝ 192 或者 k ＝ 256 的各种密钥大小对应。

[0082] 如下这样表示明文 [P]、密文 [C]、密钥 [K] 的各比特大小。

[0083] 明文 P :n 比特

[0084] 密文 C :n 比特

[0085] 密钥 K :k 比特

[0086] 图 1 示出与 k 比特的密钥长度对应的 n 比特共用密钥块密码算法 E 的图。如图 1 所示，共用密钥块密码处理部 E10 输入 n 比特的明文 P 和 k 比特的密钥 K，执行预先决定的密码算法，输出 n 比特的密文 C。此外，图 1 仅示出从明文生成密文的加密处理。从密文生成明文的解密处理一般使用 E10 的逆函数。但是，根据加密处理部 E10 的结构，在解密处理中也能够应用相同的共用密钥块密码处理部 E10，通过变更密钥的输入顺序等顺序能够进行解密处理。

[0087] 参照图 2 说明图 1 所示的共用密钥块密码处理部 E10 的内部结构。能够考虑将块密码分成两个部分。是密钥调度部 11 和密 码处理部 12，其中，上述密钥调度部 11 将密钥 K 作为输入，通过某个决定的步骤来扩展输入密钥 K 的比特长度，输出扩展密钥 K’(比特长度 k’)，上述密码处理部 12 接受明文 P 和从密钥调度部 11 输入的扩展密钥 K’，输入明文 P，执行应用了扩展密钥 K’ 的密码处理，执行用于生成密文 C 的数据变换。此外，如之前说明的那样，根据密码处理部 12 的结构，有时在将密文还原为明文的数据解密处理中也能够

应用密码处理部 12。

[0088] 接着，参照图 3 说明图 2 所示的密码处理部 12 的详细结构。如图 3 所示，密码处理部 12 具有反复执行应用了循环函数执行部 20 的数据变换的结构。即，能够将密码处理部 12 分割为循环函数执行部 20 这种处理单位。循环函数执行部 20 作为输入接受前级的循环函数执行部的输出 X_i 和根据扩展密钥生成的循环密钥 RK_i 这两个数据，在内部执行数据变换处理，将输出数据 X_{i+1} 输出到下一个循环函数执行部。此外，在第一循环中，输入是明文或者对明文的初始化处理数据。另外，最终循环的输出为密文。

[0089] 在图 3 所示的例子中，密码处理部 12 具有 r 个循环函数执行部 20，为反复进行循环函数执行部中的数据变换 r 次来生成密文的结构。将循环函数的反复次数称为循环数。在图示的例子中，循环数为 r。

[0090] 各循环函数执行部的输入数据 X_i 是加密过程中的 n 比特数据，提供某个循环中的循环函数的输出 X_{i+1} 作为下一个循环的输入。各循环函数执行部的另一个输入数据使用基于从密钥调度部输出的扩展密钥 K’的数据。将该被输入到各循环函数执行部、应用于循环函数的执行的密钥称为循环密钥。在图中，将应用于 i 循环的循环密钥表示为 RK_i。扩展密钥 K’例如构成为 r 个循环的循环密钥 $RK_1 \sim RK_r$ 的连接数据。

[0091] 图 3 所示的结构是从密码处理部 12 的输入侧观察、将第一循环的输入数据设为 X_0、将从第 i 个循环函数输出的数据表示为 X_i、将循环密钥表示为 RK_i 的密码处理部 12 的结构。此外，根据该密码处理部 12 的结构，例如能够设为如下结构：将所应用的循环密钥的应用顺序设定为与加密处理相反，通过将密文输入到密码处理部 12 来输出明文。

[0092] 图 3 所示的密码处理部 12 的循环函数执行部 20 可取各种方式。能够根据其密码算法所采用的结构 (structure) 来分类循环函数。作为代表的结构存在如下结构：

[0093] (a)SPN(Substitution Permutation Network：置换组合网路）结构、

[0094] (b)Feistel 结构、

[0095] (c) 扩展 Feistel 结构。

[0096] 下面，参照图 4 ～图 6 说明这些具体结构。

[0097] (a)SPN 结构循环函数

[0098] 首先，参照图 4 说明作为循环函数执行部 20 的一个结构例的 SPN 结构循环函数。SPN 结构循环函数执行部 20a 具有连接了非线性变换层 (S 层）和线性变换层 (P 层）的所谓 SP 型结构。如图 4 所示，SPN 结构循环函数执行部 20a 由如下等部分构成：异或运算部 21，其对所有 n 比特的输入数据执行与循环密钥之间的异或 (EXOR) 运算；非线性变换处理部 22，其输入异或运算部 21 的运算结果，执行输入数据的非线性变换；以及线性变换处理部 23，其输入非线性变换处理部 22 中的非线性变换处理结果，对输入数据执行线性变换处理。线性变换处理部 23 的线性变换处理结果被输出到下一个循环。在最终循环中成为密文。此外，在图 4 示出的例子中示出了异或运算部 21、非线性变换处理部 22、线性变换处理部 23 的处理顺序，但是并不限定这些处理部的顺序， 也可以设为以其它顺序进行处理的结构。

[0099] (b)Feistel 结构

[0100] 接着，参照图 5 说明作为循环函数执行部 20 的一个结构例的 Feistel（菲斯特尔）结构。如图 5 所示，Feistel 结构将来自前一循环的作为输入（在第一循环中为输入文）的

n 比特输入数据分割为 n/2 比特的两个数据，在各循环中进行替换并且执行处理。

[0101] 如图所示，在应用了具有 Feistel 结构的循环函数执行部 20b 的处理中，对 F 函数部 30 输入一方的 n/2 比特数据和循环密钥。F 函数部 30 与上述的 SPN 结构相同，具有连接了非线性变换层（S 层）和线性变换层（P 层）的所谓 SP 型结构。

[0102] 来自前一循环的 n/2 比特数据和循环密钥被输入到 F 函数部 30 的异或运算部 31 来进行异或（EXOR）处理。并且，将该结果数据输入到非线性变换处理部 32 来执行非线性变换，并且，该非线性变换结果被输入到线性变换处理部 33 来执行线性变换。输出该线性变换结果作为 F 函数处理结果数据。

[0103] 并且，将该 F 函数输出和从前一循环输入的另一个 n/2 比特输入输入到异或运算部 34，执行异或运算（EXOR），将执行结果设定为下一个循环中的 F 函数的输入。此外，图示的被设定为第 i 循环中的 F 函数输入的 n/2 比特应用于与下一个循环的 F 函数输出之间的异或运算。这样，Feistel 结构在各循环中相互替换输入并且执行应用了 F 函数的数据变换处理。

[0104] (c) 扩展 Feistel 结构

[0105] 接着，参照图 6 说明作为循环函数执行部 20 的一个结构例的扩展 Feistel 结构。首先，参照图 5 说明的 Feistel 结构将 n 比特的明文分割为两个，划分为各 n/2 比特来执行处理。即，是设为分割数 :d ＝ 2 的处理。此外，该分割数也被称为数据序列数。

[0106] 在扩展 Feistel 结构中，是将该数据序列数（分割数）d 设为 2 以上的任意整数的设定。能够定义与数据序列数（分割数）d 的值相应的各种扩展 Feistel 结构。在图 6 所示的例子中，数据序列数（分割数）d ＝ 4，对各序列输入 n/4 比特的数据。在各循环中执行一个以上的作为循环函数的 F 函数。图示的例子是在一个循环中进行两个 F 函数部的循环运算的结构例。

[0107] F 函数部 41、42 的结构与之前参照图 5 说明的 F 函数部 30 的结构相同，是执行循环密钥与输入值之间的异或运算、非线性变换处理、线性变换处理的结构。此外，调整输入到各 F 函数部的循环密钥使其比特数与输入比特一致。在图示的例子中，输入到各 F 函数部 41、42 的循环密钥为 n/4 比特。对构成扩展密钥的循环密钥进一步进行比特分割来生成这些密钥。此外，在将数据序列数（分割数）设为 d 时，输入到各序列的数据是 n/d 比特，输入到各 F 函数的密钥的比特数也被调整为 n/d 比特。

[0108] 此外，在图 6 所示的扩展 Feistel 结构中，是将数据序列数（分割数）设为 d、在各循环中并列地执行 d/2 个 F 函数的结构例，但是扩展 Feistel 结构可以是在各循环中执行一个以上 d/2 个以下 F 函数的结构。

[0109] 如参照图 4 ～图 6 说明的那样，共用密钥块密码中的密码处理部 12 的循环函数执行部 20 可取

[0110] (a)SPN(Substitution Permutation Network) 结构、

[0111] (b)Feistel 结构、

[0112] (c) 扩展 Feistel 结构、

[0113] 这些结构。这些循环函数执行部都具有连接了非线性变换层（S 层）和线性变换层（P 层）的所谓 SP 型结构。即，具有执行非线性变换处理的非线性变换处理部和执行线性变换处理的线性变换处理部。下面，说明这些变换处理结构。

[0114] （非线性变换处理部）

[0115] 参照图 7 说明非线性变换处理部的具体例。如图 7 所示，具体地说，排列 m 个被称为 S 盒 (S-box)51 的 s 比特输入 s 比特输出的非线性变换表而得到非线性变换处理部 50，ms 比特的输入数据以每 s 比特进行分割并分别输入到对应的 S 盒 (S-box)51 来变换数据。在各 S 盒 51 中，例如执行应用了变换表的非线性变换处理。

[0116] 当所输入的数据大小变大时存在安装上的成本变高的趋势。如图 7 所示，为了避免这种情况，多采用将处理对象数据 X 分割为多个单位、对每个单位实施非线性变换的结构。例如在将输入大小设为 ms 比特时，分割为各 s 比特的 m 个数据，分别对 m 个 S 盒 (S-box)51 输入 s 比特，例如执行应用了变换表的非线性变换处理，合成这些各 S 比特输出 m 个来得到 ms 比特的非线性变换结果。

[0117] （线性变换处理部）

[0118] 参照图 8 说明线性变换处理部的具体例。线性变换处理部输入输入值、例如输入来自 S 盒的作为输出数据的 ms 比特的输出值作为输入值 X，对该输入实施线性变换并输出 ms 比特的结果。线性变换处理例如执行输入比特位置的替换处理等线性变换处理，输出 ms 比特的输出值 Y。线性变换处理例如对输入应用线性变换矩阵来进行输入比特位置的替换处理。该矩阵的一例是图 8 所示的线性变换矩阵。

[0119] 在线性变换处理部中应用的线性变换矩阵的要素为扩展域 :GF(2^8) 域的要素、GF(2) 的要素等，一般能够构成为应用了各种表现的矩阵。图 8 是表示具有 ms 比特输入输出、通过在 GF(2^s) 上定义的 m×m 矩阵来定义的线性变换处理部的一个结构例的图。

[0120] [2. 通过配置多个不同的 S 盒来提高抵抗性的结构]

[0121] 如上所述，共用密钥块密码是通过循环函数的反复来进行密码处理的结构。作为该共用密钥块密码处理的问题点，存在由密码分析引起的密钥泄漏。利用密码分析容易进行密钥的分析导致该密码处理的安全性较低，实用上成为较大的问题。下面，说明通过配置多个不同的 S-box(S 盒) 来提高抵抗性的密码处理结构。

[0122] 如参照图 7 说明的那样，包括在循环函数执行部中的非线性变换处理部由执行非线性变换处理的多个 S-box(S 盒) 构成。以往，这些 S-box 都是如下结构 :应用共用的非线性变换处理用的表，在各 S 盒中进行共用的非线性变换处理。

[0123] 在本发明中着眼于由该 S-box 的共用性引起的弱点、即对于密钥分析等作为密码分析的攻击较薄弱，提出通过配置多个不同的 S-box 来提高抵抗性的结构。

[0124] 下面，作为本发明的实施例，依次说明下面三个实施例。

[0125] (2A) 通过在使用了 S-box 的 Feistel 或者扩展 Feistel 型密码中配置两种以上不同的 S-box 来提高对饱和攻击的抵抗性的结构

[0126] (2B) 通过在使用了 S-box 的块密码中使两种以上不同的 S-box 共存来提高对代数攻击 (XSL 攻击) 的抵抗性的结构

[0127] (2C) 在使用了 S-box 的 Feistel 密码或者扩展 Feistel 型密码中同时实现上述 (2A)、(2B) 的结构

[0128] (2A) 通过在使用了 S-box 的 Feistel 或者扩展 Feistel 型密码中配置两种以上不同的 S-box 来提高对饱和攻击的抵抗性的结构

[0129] 首先，说明通过在使用了 S-box 的 Feistel 或者扩展 Feistel 型密码中配置两种

以上不同的 S-box 来提高对饱和攻击的抵抗性的结构。

[0130] (2A-1. 饱和攻击的概要)

[0131] 首先,说明作为对块密码的攻击而第一个被知道的饱和攻 击。饱和攻击存在多个类型。第一类型是利用了如下性质的攻击法 :在对明文的特定的字节位置改变 256 种值并进行输入的情况下,在数个循环的循环变换处理之后,在输出值的特定的字节位置上出现所有 256 种值。

[0132] 另外,作为饱和攻击的其它方式的攻击,是利用了如下性质的攻击法 :当对数个循环的循环变换后出现在特定的字节位置上的值计算总和时,必然为 0。

[0133] 例如,作为对执行循环函数的共用密钥块密码处理装置输入的 256 种明文 P_0 ～ P_{255} 而依次输入如下这些明文 P_0 ～ P_{255}。

[0134] P_0 = (0,0,0,0,0,0,0,0)

[0135] P_1 = (0,0,0,0,0,0,0,1)

[0136] : :

[0137] P_{255} = (0,0,0,0,0,0,0,255)

[0138] 此外,在上述记载中,各 [0] 表示一个字节数据的 0。

[0139] 在依次输入这些明文 P_0 ～ P_{255} 的情况下,将某个特定的循环数的数据变换处理结束之后的输出值如下那样设为 C_0 ～ C_{255}。

[0140] C_0 = (c_0, ? , ? , ? , ? , ? , ? , ?)

[0141] C_1 = (c_1, ? , ? , ? , ? , ? , ? , ?)

[0142] : :

[0143] C_{255} = (c_{255}, ? , ? , ? , ? , ? , ? , ?)

[0144] 上述输出中的 [?] 可以是任意的比特值。

[0145] 如上所述,在这些输出 C_0 ～ C_{255} 中具有如下性质 :在特定的字节位置 (在上述例子中为第一字节位置) 上出现所有 256 种值 c_0 ～ c_{255}。这样,在预先已知 0 至 255 的值与顺序无关地必然分别出现各一次的情况下,能够进行利用了该性质的攻击。已知通过依次变更输入值地分析输出值能够估计循环密钥。

[0146] 并且,在对包括在输出 C_0 ～ C_{255} 中的特定的字节位置的值 : c_0 ～ c_{255} 进行求和 (EXOR) 时,在为 0 的情况下,能够进行利用了该性质的攻击 (密码分析)。这样,通过依次输入 256 种明文 P_0 ～ P_{255} 地分析特定字节位置的输出能够估计密钥。

[0147] 饱和攻击是如下攻击 (分析) 方法 :在这样循环函数部的变换结果中产生带来上述那样的特定的规则的输出、即具有

[0148] 出现所有 256 种值 c_0 ～ c_{255} 或者

[0149] 对特定的字节位置的值 :c_0 ～ c_{255} 进行求和 (EXOR) 时为 0

[0150] 这种规则性的输出的情况下,根据这些规则性来执行攻击。

[0151] 因而,在密码的设计阶段设为不产生这种特异的输出作为循环函数部的输出的结构,对设为对饱和攻击较强的密码很有效。此外,该饱和攻击不限于仅字节单位 (8 比特) 的分析,能够对任意的比特长度进行利用了相同性质的攻击。

[0152] (2A-2) 应用了 Feistel 结构或者扩展 Feistel 结构的密码处理中的问题点

[0153] 接着,研究应用了 Feistel 结构或者扩展 Feistel 结构的密码处理中的问题点。

[0154] 关于Feistel结构或者扩展Feistel结构，如之前参照图5、图6说明的那样，都是反复执行循环运算的结构，该循环运算应用了SP型的具有非线性变换处理部和线性变换处理部的F函数部。Feistel结构的数据序列数（分割数）限定为2、但是在扩展Feistel结构中数据序列数（分割数）被设定为2以上的任意数这一点不同。

[0155] 下面，假设在应用了Feistel结构或者扩展Feistel结构的密码处理中作为循环函数执行部的F函数内的非线性变换处理部中利用了S-box的结构。如之前参照图7说明的那样，S-box对m分割输入到非线性变换处理部的ms比特数据得到的各s比特数据例如执行应用了非线性变换用表的非线性变换处理。

[0156] 如之前说明的那样，以往的应用于块密码中的循环函数的执行的F函数在各循环中反复利用相同的函数。在各循环中设定了这种相同的F函数的Feistel结构或者扩展Feistel结构中，容易受到上述的饱和攻击。参照图9说明其理由。

[0157] 图9是表示切出Feistel结构或者扩展Feistel结构的一部分的结构的图。即，图9示出包括在具有Feistel结构或者扩展Feistel结构的密码结构中的两个循环函数执行部、即F函数101、102。该两个F函数101、102是输入数据序列(x)和输出数据序列(y)相同并且位于上下相邻的位置的F函数。

[0158] 两个F函数101、102由与循环密钥之间的异或运算部、非线性变换处理部、线性变换处理部构成。在本处理例中F函数101、102是进行32比特输入输出处理的结构，非线性变换处理部由四个S-box构成，各S-box进行8比特输入输出。

[0159] 图9所示的A～J表示各种数据。即，表示如下这些各数据。

[0160] A：对先行F函数101的输入

[0161] B：先行F函数101的输出

[0162] C：后续F函数102的输入

[0163] D：后续F函数102的输出

[0164] E：对先行F函数101的输出B的异或运算数据

[0165] F：对数据A的异或运算数据

[0166] G：数据B与数据E之间的异或运算结果

[0167] H：数据D与数据G之间的异或运算结果

[0168] I：对先行F函数101输入的循环密钥

[0169] J：对后续F函数102输入的循环密钥

[0170] 在之后的说明中，在用字节单位(8比特)来分割表示要在各F函数101、102中处理的32比特数据的情况下，例如在数据A是32比特数据的情况下，作为各一个字节(8比特)数据A[0]、A[1]、A[2]、A[3]的连接数据，如下这样表现。

[0171] $A = A[0]|A[1]|A[2]|A[3]$

[0172] 在此，假设作为对图9所示的加密处理结构输入的明文而依次输入256种数据、例如

[0173] $P_0 = (0,0,0,0)$

[0174] $P_1 = (1,0,0,0)$

[0175] ：：

[0176] $P_{255} = (255,0,0,0)$

[0177] 这些明文 $P_0 \sim P_{255}$。此外，在上述记载中，各 [0]、[1] ～ [255] 表示一个字节数据。

[0178] 将该输入值设为对图 9 所示的先行 F 函数 101 的输入数据 A。如上所述，将数据 A 设为如下数据 :在观察 256 种数据时，在第一个字节 A[0] 中出现 0 至 255 的全部 256 种值，除此以外的字节位置总是被相同的值固定。（这是由于尝试饱和攻击的攻击者有时控制明文输入来作出这种状况，因此进行这种假设。）

[0179] 并且，当假设为在上述 256 种数据 A 的依次输入处理期间对数据 A 的异或运算数据 F 的值总是固定时，保证在后续 F 函数 102 的输入数据 C 的第一个字节 C[0] 中也出现 0 至 255 的全部 256 种值、除此以外的字节位置总是被相同的值固定。

[0180] 此时，根据

[0181] I :对先行 F 函数 101 输入的循环密钥

[0182] J :对后续 F 函数 102 输入的循环密钥

[0183] 以及

[0184] F :对数据 A 的异或运算数据

[0185] 这些各数据的值的组合，能够产生下式总是成立的情况。即，

[0186] A[0](EXOR)I[0] ＝ C[0](EXOR)J[0]

[0187] 上式能够成立。

[0188] 此外，(EXOR) 表示异或运算，

[0189] A[0](EXOR)I[0] 表示数据 A[0] 与数据 I[0] 之间的异或运算，

[0190] C[0](EXOR)J[0] 表示数据 C[0] 与数据 J[0] 之间的异或运算。

[0191] 式 :A[0](EXOR)I[0] ＝ C[0](EXOR)J[0]

[0192] 该式无非意味着输入到两个 F 函数 101、102 中的两个 S-box 的值总是为相同的值。这些 S-box 都执行相同的非线性变换处理，对相同的输入值输出相同的输出值。因而，两个 F 函数 101、102 中的两个 S-box 的输出总是相同。在各 F 函数 101、102 中，由线性变换处理部的矩阵对该相同的 S-box 输出进行线性变换，输出到右侧的数据序列 (y) 的异或运算部。是图示的异或运算部 111、112。

[0193] 从该两个 F 函数 101、102 输出到异或运算部 111、112 的值 B、D 具有特定的差分值 Δ。即，成为

[0194] B(EXOR) Δ ＝ D。

[0195] 此时，在异或运算部 111 中，根据

[0196] G ＝ B(EXOR)E

[0197] 的运算来算出数据 G，

[0198] 在异或运算部 112 中执行

[0199] H ＝ G(EXOR)D。

[0200] 由于 G ＝ B(EXOR)E、B(EXOR) Δ ＝ D，因此上述式 H ＝ G(EXOR)D 成为

[0201] H ＝ B(EXOR)E(EXOR)B(EXOR) Δ

[0202] ＝ E(EXOR) Δ。

[0203] 即，由于具有固定差分值的值之间的异或运算结果为固定值 Δ，因此作为结果成为

[0204] H = B(EXOR)E(EXOR)B(EXOR) Δ

[0205] = Δ (EXOR)E

[0206] = E(EXOR) Δ 。

[0207] 即，异或运算部 112 的输出 H 为仅是对数据 E 进行与固定的值 Δ 的异或，尽管分两级执行循环函数 (F 函数)，也为不进行数据的搅拌这种结果。当利用该性质时，变得容易估计后续循环的循环密钥。即，即使存在后续循环也能够在使用临时设定的密钥对该循环进行解密直至解密出 H 的数据为止时，通过检查是否出现该性质来概率地判断临时使用的密钥是否正确。也就是说能够估计循环密钥，导致能够利用饱和攻击进行分析。

[0208] 作为其对策能够根据各 F 函数的位置而变更应用于线性变换处理的矩阵部分，但是在各 F 函数的 S-box 相同的情况下，如果产生与上述相同的条件，则由于线性变换矩阵的要素的关联性，在后续 F 函数 102 的输出 D 与数据 G 进行异或的时刻，会产生一部分字节相互消去，有时产生对攻击者有利的状况。

[0209] 这样，将在至少对相同序列进行输出的多个 F 函数中应用的非线性变换处理设为相同结构，存在能够利用饱和攻击来进行密钥估计的可能性。并且，根据 S-box，作为其运算 (EXOR) 结果、即

[0210] S(A[0](EXOR)I[0])(EXOR)S(C[0](EXOR)J[0])

[0211] 它们的结果，不能说都符合 0 至 255 的所有 256 种值都出现这种情况。严密地说，即使 A[0] 和 C[0] 两者都分别输出 256 种不同的值，也未必保证它们的运算 (EXOR) 结果得到所有 256 种输出值，但是根据 S-box 能够产生这种状况。当产生该未预期的情况时，能够利用于攻击的信息 (所有值不同的信息) 被保存到下一级，导致产生对于攻击者来说有利的状况。

[0212] (2A-3) 通过利用多种 S-box 来提高抵抗性的方法

[0213] 下面，说明用于使利用饱和攻击的密钥估计困难化的结构 例。即，构成各 F 函数的非线性变换处理部、即 S-box 使得即使万一具备上述那样的条件也通过数据的消去使循环函数执行前的数据与执行后的数据成为不相等的数据。

[0214] 参照图 10 说明其具体例。图 10 所示的结构也与图 9 相同，示出切出 Feistel 结构或者扩展 Feistel 结构的一部分的结构，示出输入数据序列 (x) 和输出数据序列 (y) 相同并且位于上下相邻的位置的 F 函数 201、202。

[0215] 两个 F 函数 201、202 由与循环密钥之间的异或运算部、非线性变换处理部、线性变换处理部构成。F 函数 201、202 是进行 32 比特输入输出处理的结构，非线性变换处理部由四个 S-box 构成，各 S-box 进行 8 比特输入输出。

[0216] 图 10 所示的 A ～ J 与图 9 相同，表示如下这些各数据。

[0217] A :对先行 F 函数 201 的输入

[0218] B :先行 F 函数 201 的输出

[0219] C :后续 F 函数 202 的输入

[0220] D :后续 F 函数 202 的输出

[0221] E :对先行 F 函数 201 的输出 B 的异或运算数据

[0222] F :对数据 A 的异或运算数据

[0223] G :数据 B 与数据 E 之间的异或运算结果

[0224] H：数据 D 与数据 G 之间的异或运算结果

[0225] I：对先行 F 函数 201 输入的循环密钥

[0226] J：对后续 F 函数 202 输入的循环密钥

[0227] 在图 10 示出的结构中，设为设定在先行 F 函数 201 和后续 F 函数 202 各自中的非线性变换处理部的 S-box 利用了不同的 S-box[S1]、[S2] 的结构。

[0228] 即，在先行 F 函数 201 中执行非线性变换处理的 S-box[S1] 和在后续 F 函数 202 中执行非线性变换处理的 S-box[S2] 执行不 同的非线性变换处理。具体地说，例如执行应用了不同的非线性变换表的非线性变换处理，而对相同输入进行相同输出不被限定。

[0229] 在此，将各 S-box ：S1、S2 设为满足以下条件的两个不同的 S-box。

[0230] 当各 S-box ：S1、S2 是执行 n 比特输入输出的非线性变换处理的 S-box 时，

[0231] （条件 1）

[0232] 在对任意 s 比特数据 c 依次输入所有 s 比特数据即 2^s 个 x 的情况下，

[0233] 与输入数据 [x] 对应的第一 S-box[S1] 的输出 S1(x) 和

[0234] 与输入数据 [x(EXOR)c] 对应的 S-box[S2] 的输出 S2(x(EXOR)c) 最少一个具有不同的值。即，

[0235] S1(x)(EXOR)S2(x(EXOR)c)

[0236] 上述式不是固定值。

[0237] 并且，

[0238] （条件 2）

[0239] 在对任意 s 比特数据 c 依次输入所有 s 比特数据即 2^s 个 x 的情况下，

[0240] 与输入数据 [x] 对应的第一 S-box[S1] 的输出 S1(x) 和

[0241] 与输入数据 [x(EXOR)c] 对应的 S-box[S2] 的输出 S2(x(EXOR)c) 最少一个具有重复值。即，

[0242] S1(x)(EXOR)S2(x(EXOR)c)

[0243] 上述式不是所有 2^s 各出现一次的形式。

[0244] 其在图 10 中表示如下条件：

[0245] 在将数据 A 假设为 [x]

[0246] 将数据 F 假设为 [c] 的情况下，

[0247] 先行 F 函数 201 的 S-box[S1] 的输出 S1(x) 与

[0248] 后续 F 函数 202 的 S-box[S2] 的输出 S2(x(EXOR)c)

[0249] 这两个输出不会成为完全相同或者进行异或的结果为所有值。

[0250] 如图 10 所示那样设定满足该条件的两个 S-box[S1]、[S2]。

[0251] 即，在某个 F 函数中设为仅使用 S-box[S1] 的非线性变换处理部，在下一个 F 函数中设为仅使用 S-box[S2] 的非线性变换处理部。在此之后存在循环的情况下，同样地，在各 F 函数的非线性变换处理部中按 S-box[S1]、[S2] 的顺序进行设定。

[0252] 这样，通过将输入数据序列和输出数据序列相同并且位于上下相邻的位置的 F 函数中的非线性变换处理设为执行不同的非线性变换处理的结构、即配置不同的 S-box，能够使出现在输出序列的数据与出现在循环函数执行前的相同输出序列的数据具有较强相关的可能性显著降低。

[0253] 即，通过利用满足上述的（条件1）的S-box，即使在两个S-box的输入关系具有固定值的差的情况下，其输出的异或也最少一次具有不同值，因此能够保证不会完全消去。

[0254] 另外，通过利用满足上述的（条件2）的S-box，即使在两个S-box的输入关系具有固定值的差的情况下，其输出的异或也最少一次具有重复值，因此可以说破坏了能够利用于攻击的性质。因而，通过上述那样配置两个S-box，对饱和攻击的攻击者有利的条件减少，因此可期望能够提高对攻击的抵抗性。

[0255] 即，在图10中，在各F函数201、202中的S-box的输入值相等的情况下，即使成为

[0256] A[0](EXOR)I[0] ＝ C[0](EXOR)J[0]，

[0257] 各F函数的S-box的输出值，即

[0258] S1(A[0](EXOR)I[0])

[0259] S2(C[0](EXOR)J[0])

[0260] 这些输出值不会在所有情况下都完全一致，结果是各F函数201、202的F函数输出B、D不完全一致，能够消除之前参照图9说明那样的现象、即

[0261] E ＝ H(EXOR) Δ

[0262] 这种在一个数据序列中循环函数（F函数）执行前后的数据仅具有固定值的差分这种可能性。

[0263] 这样，通过将输入数据序列和输出数据序列相同并且位于上下相邻位置的F函数中的非线性变换处理设为执行不同的非线性变换处理的不同S-box，能够大幅提高饱和攻击的难度，提高对攻击的抵抗性。

[0264] （发展系-1）

[0265] 参照图10说明的上述结构仅着眼于两个F函数的关系导出将设定在这两个F函数中的S-box设为不同的S-box这种条件，但是在三个以上的F函数中也可以说是相同的情形。例如，如图11所示，通过对每个F函数配置不同的S-box能够期望提高对饱和攻击的抵抗性。

[0266] 图11示出切出Feistel结构或者扩展Feistel结构的一部分的结构，示出输入数据序列(x)和输出数据序列(y)相同并且位于上下相邻位置的三个F函数211～213。

[0267] 在F函数211的非线性变换处理部中设定S-box[S1]，

[0268] 在F函数212的非线性变换处理部中设定S-box[S2]，

[0269] 在F函数213的非线性变换处理部中设定S-box[S3]，

[0270] S1 ≠ S2 ≠ S3。

[0271] 这样，作为要求多个S-box具有的条件，

[0272] （条件1）

[0273] 对由k个(k ＞ 2)S-box构成的集合S1，S2，..，Sk的相互不同的两 个S-box的组Si和Sj(i ≠ j)、对任意的c提供所有可能的2^s个x作为输入时，

[0274] Si(x)和

[0275] Sj(x(EXOR)c)

[0276] 这些S盒输出不完全相同，最少一个输出不同的值。即，

[0277] Si(x)和Sj(x(EXOR)c)的异或结果不是固定值。

[0278] 并且，

[0279] （条件 2）

[0280] 对由 k 个 (k ＞ 2)S-box 构成的集合 S1，S2，..，Sk 的相互不同的两个 S-box 的组 Si 和 Sj(i ≠ j)、对任意的 c 提供所有可能的 2^n 个 x 作为输入时，

[0281] Si(x) 和

[0282] Sj(x(EXOR)c)

[0283] 这些 S 盒输出不是所有 2^n 个值各出现一次的形式。即，最少出现一个重复值。

[0284] 设定满足该条件的由 S-box 构成的集合 S1，S2，..，Sk，在输入数据序列 (x) 和输出数据序列 (y) 相同并且位于上下按序排列的位置的多个 F 函数中配置这些 F 函数，由此能够显著降低在输出序列上出现的数据与在循环函数执行前的相同输出序列上出现的数据一致的可能性，结果是能够大幅提高饱和攻击的难度，提高对攻击的抵抗性。

[0285] （发展系 -2）

[0286] 当考虑实际安装时，存在即使包含在各 F 函数中的 S-box 的种类为多个也希望包含在各 F 函数中的 S-box 的组合相同的情况。

[0287] 即，例如在通过硬件、软件来进行相当于 F 函数的数据变换的情况下，如果包含在各 F 函数中的 S-box 的组合相同，则将 作为 F 函数的硬件、软件设为相同，在各循环中仅适当地变更输入输出就能够实现各循环中的 F 函数的数据变换。

[0288] 参照图 12 说明具体例。图 12 也与图 10 相同，示出切出 Feistel 结构或者扩展 Feistel 结构的一部分的结构，示出输入数据序列 (x) 和输出数据序列 (y) 相同并且位于上下相邻位置的 F 函数 221、222。

[0289] 包含在先行 F 函数 221 中的四个 S-box 从上起按 S1、S2、S1、S2 的顺序进行配置，在下一个循环的后续 F 函数 222 中从上起设为 S2、S1、S2、S1 的顺序。

[0290] 此外，S1 ≠ S2。

[0291] 如果设为这种设定，进行设为能够并列执行两个 S1、两个 S2 的结构的安装，则能够应用该结构执行 F 函数 221、222，还能够降低安装上的成本并且使装置小型化。

[0292] 在图 12 所示的结构中，在各 F 函数 221、222 中应用于对应的比特列的非线性变换处理为

[0293] S1 → S2 或者

[0294] S2 → S1，

[0295] 作为对应的比特数据（例如各字节单位）的处理，成为与参照图 10 说明的处理相同的处理，作为结果，能够得到相同的效果、即显著降低在输出序列中出现的数据与在循环函数执行前的相同输出序列中出现的数据一致的可能性，结果是能够大幅提高饱和攻击的难度，提高对攻击的抵抗性。

[0296] 图 13 示出另一个具体例。图 13 与图 11 相同，示出切出 Feistel 结构或者扩展 Feistel 结构的一部分的结构，示出输入数据序列 (x) 和输出数据序列 (y) 相同并且位于上下相邻位置的三个 F 函数 231 ～ 233。

[0297] 包含在先行 F 函数 231 中的四个 S-box 从上起按 S1、S2、S3、 S4 的顺序进行配置，在下一个循环的中间 F 函数 232 中从上起设定为 S2、S3、S4、S1 的顺序，在再下一个循环的中间 F 函数 233 中从上起设定为 S3、S4、S1、S2 的顺序。

[0298] 此外，S1 ≠ S2 ≠ S3 ≠ S4。

[0299] 如果设为这种设定，进行设为能够并列执行 S1 ～ S4 各一个的结构的安装，则能够应用该结构执行所有 F 函数 231 ～ 233，能够降低安装上的成本并且使装置小型化。

[0300] 在图 13 所示的结构中，在各 F 函数 231 ～ 233 中应用于对应的比特列的非线性变换处理成为如下顺序，

[0301] S1 → S2 → S3 → S4 → S1 → S2…

[0302] 作为对应的比特数据（例如各字节单位）的处理，成为与参照图 10、图 11 说明的处理相同的处理，作为结果，能够得到相同的效果、即显著降低在输出序列中出现的数据与在循环函数执行前的相同输出序列中出现的数据一致的可能性，结果是能够大幅提高饱和攻击的难度，提高对攻击的抵抗性。

[0303] (2B) 在使用了 S-box 的块密码中通过使两种以上不同的 S-box 共存来提高对代数攻击 (XSL 攻击) 的抵抗性的结构

[0304] 接着，说明在使用了 S-box 的块密码中通过使种类不同的 S-box 共存来提高对代数攻击 (XSL 攻击) 的抵抗性的结构。

[0305] (2B-1) 代数攻击 (XSL 攻击) 的概要

[0306] 首先说明作为对块密码的攻击而已知的代数攻击 (XSL 攻击)。对块密码的代数攻击 (XSL 攻击) 是利用了 S-box 的代数表现的攻击。关于 S-box 的输入输出，在表现为代数式时能够导出多个表达式，攻击所需的计算量根据该表达式的最大次数、所包含的项数而发生变化。

[0307] 作为代数攻击 (XSL 攻击) 的实施例之一存在使用了布尔表达式的方式。例如设为存在某个安装了多个具有 8-bit 输入输出 的 S-box 的块密码。此时，在 8 比特输入输出 S-box 中，当将输入侧和输出侧的比特分别设为

[0308] 输入 X :(x1, x2, x3, x4, x5, x6, x7, x8)

[0309] 输出 Y :(y1, y2, y3, y4, y5, y6, y7, y8)

[0310] 时，评价以两次以下的布尔表达式表达出来的表达式的数量。

[0311] 更具体地说，评价包含在以布尔表达式表示上述输入输出 X、Y 得到的结果中的项能够以

[0312] (1, xi, yi, xixj, yiyj, xiyj)

[0313] 这种两次以下的任一种形式表示的多项式的数量。

[0314] 在将最大次数限定为两次等低次数进行取出时，如果这样表示的全部布尔表达式之中独立表达式的数量更多、项目数较少，则视为有利于攻击。即，在这样将最大次数限定为两次等时，如果独立表达式的数更多、项目数较少，则视为有利于攻击，对攻击的抵抗性差。

[0315] 另外，不仅是布尔表达式，在扩展域 GF(2^8) 等定义域上能够以较低次数进行代数表现的情况下，也容易使用相同的方法进行代数攻击 (XSL 攻击)，视为攻击抵抗性较弱。

[0316] (2B-2) 使用了一种 S-box 时的问题点

[0317] 接着，说明在使用了 S-box 的块密码中仅利用了一种 S-box 的结构的问题点、即容易进行代数攻击 (XSL 攻击) 这种问题点。

[0318] 在执行 n-bit 的输入输出的非线性变换的 s 比特 S-box 中存在下面三种代表类型。

[0319] 类型 1 :利用扩展域 GF(2^s) 上的逆映射 :$Y = X^{-1}$、幂函数 $Y = X^p$ 的 S-box

[0320] 类型 2 :组合多个例如 4-bit 输入输出等小于 s 比特的 S-box 而制作出的 S-box

[0321] 类型 3 :随机选择的 S-box

[0322] 这三种类型是代表。

[0323] 特别是关于类型 1 和类型 2,由于在硬件 (H/W) 安装时能够以低成本进行安装,因此是经常利用的 S-box。

[0324] 下面,对上述各类型 1 ～ 3 说明仅利用了一种 S-box 的结构的问题点、即容易进行代数攻击 (XSL 攻击) 这种问题点。

[0325] < 类型 1 的问题点 >

[0326] 说明类型 1、即利用了 GF(2^s) 上的逆映射 :$Y = X^{-1}$、幂函数 $Y = X^p$ 的 S-box 的情况下的问题点。

[0327] 例如,已知在使用了 GF(2^8) 上的逆映射的 S-box 的情况下以布尔表达式表示时存在如下表现 :是 20 几个独立的二次表达式并且项数为 80 几个。关于幂函数也存在相同的简单关系。另外,不仅在 GF(2^8) 上,在 GF(2^s) 上定义的 S-box 也期待相同的关系。

[0328] 当使用这些多项式表现时,能够估计针对代数攻击 (XSL 攻击) 的计算量,为确保安全性需要在密码的设计阶段利用充分数量的 S-box 使得能够得到充分的计算量。并且在使用了 GF(2^s) 上的逆映射的 S-box 的情况下,在 GF(2^s) 上能够进行 XY = 1 这种代数表现,也能够导出次数较低的多项式。已知也存在利用了这些性质的攻击方法。关于幂函数也能够应用相同的结果。

[0329] 这样,在仅使用利用了 GF(2^s) 上的逆映射、幂映射的 S-box 的密码中,由于利用两种代数性质,因此必须考虑其两者来进行设计。

[0330] 此外,关于在逆映射、幂函数的前后追加仿射变换而制作出的 S-box,也可以说与上述相同。

[0331] < 类型 2 的问题点 >

[0332] 接着,说明类型 2、即组合多个更小的 (例如 4-bit)S-box 而制作出的 S-box 的情况下的问题点。

[0333] 考虑使用了组合多个输入输出例如 4-bit 的较小的 S-box 而 制作出的 8 比特 S-box 的情况。已知利用 3 ～ 5 个 4 比特 S-box 能够制作出 8 比特 S-box。为了代数攻击 (XSL 攻击) 而对 4 比特 S-box 的输入输出比特导出两次以下的布尔表达式多项式,说起来输入输出比特的合计仅 8 个,因此已知以这种以较低次数表现的独立表达式至少存在二十几个。因此,能够利用它们构成攻击。在为了制作较大输入输出大小的 S-box 而使用更小大小的 S-box 来构成的情况下,可以说具有这种趋势。

[0334] 但是,作为该方式的优点,使用利用了 GF(2^8) 上的逆映射的 S-box 时那样的在 GF(2^s) 域上存在简单的代数关系的概率极低,由此可知攻击所需的计算量变多,因此根据上述情况,与上述 S-box 相比从代数攻击 (XSL 攻击) 的观点出发可以说优点和缺点两者共存。

[0335] < 类型 3 的问题点 >

[0336] 接着,说明类型 3、即随机选择的 S-box 的情况下的问题点。随机选择的 S-box 不能期望具有上述那样的在代数上看较弱的性质,能够期望对代数攻击 (XSL 攻击) 的较高的

安全性，但是 H/W 安装成本极高，因此存在不希望将所有 S-box 都设为随机选择的 S-box 这种问题。

[0337] (2B-3) 通过利用代数性质不同的多种 S-box 来提高抵抗性的结构

[0338] 下面说明如下结构：鉴于上述问题，通过利用代数性质不同的两种以上的 S-box 来提高对使用了布尔多项式的代数攻击 (XSL 攻击)、以及利用了 GF(2^s) 域的代数攻击 (XSL 攻击) 两者的抵抗性，并且与将所有 S-box 都设为随机选择的 S-box 的情况相比，提高了硬件 (H/W) 安装效率。

[0339] 如之前所说明那样，在执行 s-bit 输入输出的非线性变换的 s 比特 S-box 中存在下面三种代表类型。

[0340] 类型 1：利用了扩展域 GF(2^s) 上的逆映射：$Y = X^{-1}$、幂函数 $Y = X^p$ 的 S-box

[0341] 类型 2：组合多个 t-bit 的较小的 S-box 而制作出的 S-box(其中设为 $t < s$)

[0342] 类型 3：随机选择的 S-box

[0343] 这三种类型是代表。

[0344] 在本实施例中，通过组合利用这些不同类型的 S-box 来实现提高对代数攻击 (XSL 攻击) 的抵抗性、并且提高了硬件 (H/W) 安装效率的结构。即，是在使用了 S-box 的块密码中通过使两种以上不同的 S-box 共存来提高对代数攻击 (XSL 攻击) 的抵抗性的结构。此外，能够应用本实施例的密码处理结构只要是具有执行非线性变换处理的 S-box 的密码处理结构即可，例如，能够应用于之前说明的下面各种密码处理结构、即

[0345] (a)SPN(Substitution Permutation Network) 结构

[0346] (b)Feistel 结构

[0347] (c) 扩展 Feistel 结构

[0348] 这些结构中的任一个。

[0349] 在本处理例中，将包含在执行数据变换处理的循环函数中的、作为非线性变换处理部的 S-box 设定为下面 (a) ～ (d) 中的任一个。

[0350] (a) 将一部分设为类型 1 的 S-box、将其它设为类型 2 的 S-box 的结构

[0351] (b) 将一部分设为类型 1 的 S-box、将其它设为类型 3 的 S-box 的结构

[0352] (c) 将一部分设为类型 2 的 S-box、将其它设为类型 3 的 S-box 的结构

[0353] (d) 将一部分设为类型 1 的 S-box、将另一部分设为类型 2 的 S-box、将其它设为类型 3 的 S-box 的结构

[0354] 例如，在设定为上述 (a) 的情况下，设为如下结构：

[0355] 将包含在执行数据变换处理的循环函数中的作为非线性变换处理部的 S-box 中一半的 S-box 设为类型 1、即使用了 GF(2^8) 上的逆映射的 S-box，考虑除去了此以外的 S-box 的虚构的密码，根据其对利用了布尔表达式的代数攻击 (XSL 攻击) 的计算量进行估计，如果估计出充分的计算量，则将剩余一半的 S-box 设为类型 2、即组合多个 4-bit 的较小的 S-box 而制作出的 8 比特 S-box。

[0356] 这样，通过设定为上述 (a) 的类型 1 与类型 2 共存的密码处理结构，如果在对 GF(2^8) 上的计算量进行估计时能够具有充分的抵抗性，则与单独利用各 S-box 时相比，能够制作出综合抵抗性提高的块密码。

[0357] 并不限于上述设定，在上述 (a) ～ (d) 中的任一个的情况下都相同，进行设定使得

即使仅限定一部分 S-box 对代数攻击 (XSL 攻击) 的抵抗性也充分高，剩余 S-box 考虑安装效率等来决定即可。

[0358] 参照图 14 ～图 18 说明上述 (a) ～ (d) 那样配置了不同类型的 S-box 的具体的密码处理结构例。图 14 ～图 18 所示的例子都示出具有六个循环的循环函数执行部的密码处理结构，在各循环函数执行部中具有由多个 S-box 构成的非线性变换处理部和线性变换处理部。

[0359] 图 14 示出由六个循环构成、在每一个循环中包含十个 S-box 的 SPN 块密码的例子。SPN 块密码在各循环中执行具有非线性变换层 (S 层)、线性变换层 (P 层) 的数据变换。各循环的十个 S-box 例如输入将输入数据进行十分割得到的分割数据，执行非线性变换处理，将非线性变换结果数据输出到线性变换层 (P 层)，线性变换处理结果被输出到下一个循环函数执行部。最终级循环函数执行部的输出为密文。

[0360] 图示的各循环函数执行部 301 ～ 306 的 $[S_1]$、$[S_2]$ 分别表示类型 1 的 S-box、类型 2 的 S-box，是上述的不同类型的作为非线性变换处理部的 S-box。

[0361] 图 14 所示的例子是如下结构例 ：在先行的三个循环的循环函数执行部 301 ～ 303 中配置

[0362] 类型 1 ：利用了扩展域 $GF(2^s)$ 上的逆映射 ：$Y = X^{-1}$、幂函数 $Y = X^p$ 的 S-box

[0363] 该类型 1 的 S-box$[S_1]$，

[0364] 在后续的三个循环的循环函数执行部 304 ～ 306 中配置

[0365] 类型 2 ：组合多个 4-bit 那样较小的 S-box 而制作出的 S-box

[0366] 该类型 2 的 S-box$[S_2]$。

[0367] 在图 14 的结构中，非线性变换处理在前半循环中执行为应用了类型 1 的 S-box 的处理，在后半循环中执行为应用了类型 2 的 S-box 的处理。通常，在全部假设为相同类型的 S-box 之后执行代数攻击 (XSL 攻击)，在这种不同类型的 S-box 共存的设定的情况下，攻击、即分析变得困难。结果是实现对代数攻击 (XSL 攻击) 等密码分析的抵抗性较高的密码处理结构。

[0368] 图 15 与图 14 相同，示出由六个循环构成的、在每一个循环中包含十个 S-box 的 SPN 块密码的例子。

[0369] 图 15 所示的例子是如下结构例 ：

[0370] 在第一、三、五奇数循环的循环函数执行部 321、323、325 中配置

[0371] 类型 1 ：利用了扩展域 $GF(2^s)$ 上的逆映射 ：$Y = X^{-1}$、幂函数 $Y = X^p$ 的 S-box

[0372] 该类型 1 的 S-box$[S_1]$，

[0373] 在第二、四、六偶数循环的循环函数执行部 322、324、326 中配置

[0374] 类型 2 ：组合多个 4-bit 那样较小的 S-box 而制作出的 S-box

[0375] 该类型 2 的 S-box$[S_2]$。

[0376] 在图 15 的结构中，非线性变换处理在奇数循环中执行为应用了类型 1 的 S-box 的处理，在偶数循环中执行为应用了类型 2 的 S-box 的处理。在本结构中也与图 14 的结构相同，是不同类型的 S-box 共存的设定，实现对代数攻击 (XSL 攻击) 等密码分析的抵抗性较高的密码处理结构。

[0377] 图 16 与图 14、图 15 相同，示出由六个循环构成的、在每一个循环中包含十个

S-box 的 SPN 块密码的例子。

[0378] 图 16 所示的例子是如下结构例 :

[0379] 在所有循环的循环函数执行部 341 ～ 346 中配置

[0380] 类型 1 :利用了扩展域 GF(2^s) 上的逆映射 :$Y = X^{-1}$、幂函数 $Y = X^p$ 的 S-box

[0381] 该类型 1 的 S-box[S_1] 和

[0382] 类型 2 :组合多个 4-bit 那样较小的 S-box 而制作出的 S-box

[0383] 该类型 2 的 S-box[S_2]

[0384] 各一半、即各五个。

[0385] 输入到各循环函数执行部 341 ～ 346 的数据被进行十分割后输入到各 S 盒。进行十分割得到的分割数据 d_1 ～ d_{10} 的前半的数据 d_1 ～ d_5 输入到类型 1 的 S-box,执行应用了类型 1 的 S-box 的非线性变换处理,后半的数据 d_6 ～ d_{10} 输入到类型 2 的 S-box,执行应用了类型 2 的 S-box 的非线性变换处理。

[0386] 在图 16 的结构中也与图 14、图 15 的结构相同,是不同类型的 S-box 共存的设定,实现对代数攻击 (XSL 攻击) 等密码分析的抵抗性较高的密码处理结构。

[0387] 图 17 与图 14 ～图 16 相同,示出由六个循环构成的、在每一个循环中包含十个 S-box 的 SPN 块密码的例子。

[0388] 图 17 所示的例子与图 16 所示的例子相同,是如下结构例 :

[0389] 在所有循环的循环函数执行部 361 ～ 366 中配置

[0390] 类型 1 :利用了扩展域 GF(2^s) 上的逆映射 :$Y = X^{-1}$、幂函数 $Y = X^p$ 的 S-box

[0391] 该类型 1 的 S-box[S_1] 和

[0392] 类型 2 :组合多个 4-bit 那样较小的 S-box 而制作出的 S-box

[0393] 该类型 2 的 S-box[S_2]

[0394] 各一半、即各五个。

[0395] 输入到各循环函数执行部 361 ～ 366 的数据被进行十分割后输入到各 S 盒。进行十分割得到的分割数据 d_1 ～ d_{10} 的第奇数个分割数据 d_1、d_3、d_5、d_7、d_9 输入到类型 1 的 S-box,执行应用了类型 1 的 S-box 的非线性变换处理,第偶数个分割数据 d_2、d_4、d_6、d_8、d_{10} 输入到类型 2 的 S-box,执行应用了类型 2 的 S-box 的非线性变换处理。

[0396] 在图 17 的结构中也与图 14 ～图 16 的结构相同,是不同类型的 S-box 共存的设定,实现对代数攻击 (XSL 攻击) 等密码分析的抵抗性较高的密码处理结构。

[0397] 在图 16、图 17 所示的结构中,在各循环中并列执行的 S-box 是类型 1 的 S-box 和类型 2 的 S-box 各五个,其在所有循环中共用。因而,例如作为安装如果设为能够并列执行类型 1 和类型 2 的 S-box 各五个的结构,则能够反复应用该结构来执行所有循环的循环函数,在安装上还存在降低成本、实现小型化的优点。

[0398] 图 18 示出在 Feistel 结构中的各循环函数执行部 381 ～ 386 中配置多个不同类型的 S-box 的例子。

[0399] 图 18 所示的例子是如下结构例 :

[0400] 在所有循环的循环函数执行部 381 ～ 386 中配置

[0401] 类型 1 :利用了扩展域 GF(2^s) 上的逆映射 :$Y = X^{-1}$、幂函数 $Y = X^p$ 的 S-box

[0402] 该类型 1 的 S-box[S_1] 和

[0403] 类型 2 :组合多个 4-bit 那样较小的 S-box 而制作出的 S-box

[0404] 该类型 2 的 S-box[S_2]

[0405] 各一半、即各两个。

[0406] 输入到各循环函数执行部 381 ～ 386 中的数据被进行四分割后输入到各 S 盒。进行四分割得到的分割数据 d_1 ～ d_4 的第奇数个分割数据 d_1、d_3 输入到类型 1 的 S-box，执行应用了类型 1 的 S-box 的非线性变换处理，第偶数个分割数据 d_2、d_4 输入到类型 2 的 S-box，执行应用了类型 2 的 S-box 的非线性变换处理。

[0407] 在图 18 的结构中也与图 14 ～图 17 的结构相同，是不同类型的 S-box 共存的设定，实现对代数攻击 (XSL 攻击) 等密码分析的抵抗性较高的密码处理结构。

[0408] 此外，在图 14 ～图 18 所示的例子中示出使类型 1 和类型 2 这两种类型的 S-box 共存来进行利用的结构例，但是作为不同类型的 S-box 的共存结构，如之前说明的那样，可以是

[0409] (a) 将一部分设为类型 1 的 S-box、将其它设为类型 2 的 S-box 的结构

[0410] (b) 将一部分设为类型 1 的 S-box、将其它设为类型 3 的 S-box 的结构

[0411] (c) 将一部分设为类型 2 的 S-box、将其它设为类型 3 的 S-box 的结构

[0412] (d) 将一部分设为类型 1 的 S-box、将另一部分设为类型 2 的 S-box、将其它设为类型 3 的 S-box 的结构

[0413] 这些各种共存结构，在任一种情况下都实现提高对代数攻 击 (XSL 攻击) 的抵抗性。

[0414] (2C) 在使用了 S-box 的 Feistel 密码或者扩展 Feistel 型密码中同时实现上述 (2A)、(2B) 的结构

[0415] 接着，说明在使用了 S-box 的 Feistel 密码或者扩展 Feistel 型密码中同时实现上述 (2A)、(2B)、即

[0416] (2A) 通过在使用了 S-box 的 Feistel 或者扩展 Feistel 型密码中配置两种以上不同的 S-box 来提高对饱和攻击的抵抗性的结构

[0417] (2B) 通过在使用了 S-box 的块密码中使两种以上不同的 S-box 共存来提高对代数攻击 (XSL 攻击) 的抵抗性的结构

[0418] 这些结构的结构例。

[0419] 上述的 (2A) 的结构是通过在 Feistel 结构或者扩展 Feistel 结构中应用两种以上的 S-box 来提高对饱和攻击的抵抗性的结构，上述的 (2B) 的结构是通过对具有 S-box 的任意的块密码使用两种以上的 S-box 来提高对代数攻击 (XSL 攻击) 的抵抗性的结构。

[0420] 能够同时实现这些 (2A)、(2B) 的结构。即，能够利用满足 (2A)、(2B) 所需的性质的两种以上的 S-box 来构成具有同时提高对两个攻击的抵抗性的 Feistel 或者扩展 Feistel 结构的块密码。

[0421] 具体地说，对各 S-box[S1]、[S2]、[S3]、[S4]・・对应地设定三个类型的 S-box，其中，

[0422] 上述各 S-box[S1]、[S2]、[S3]、[S4]・・为在上述的

[0423] (2A) 通过在使用了 S-box 的 Feistel 或者扩展 Feistel 型密码中配置两种以上不同的 S-box 来提高对饱和攻击的抵抗性的结构

[0424] 中说明的、在图 10 ～图 13 的各结构中应用的执行不同的非线性变换处理的 S-box，

[0425] 上述三个类型的 S-box 为在上述的

[0426] (2B) 通过在使用了 S-box 的块密码中使两种以上不同的 S-box 共存来提高对代数攻击 (XSL 攻击) 的抵抗性的结构

[0427] 中说明的不同类型的 S-box、即

[0428] 类型 1 :利用了扩展域 GF(2^s) 上的逆映射 :$Y = X^{-1}$、幂函数 $Y = X^p$ 的 S-box

[0429] 类型 2 :组合多个 4-bit 那样较小的 S-box 而制作出的 S-box

[0430] 类型 3 :随机选择的 S-box。

[0431] 例如，在图 10 所示的结构中，

[0432] 通过将 S-box[S1] 和 S-box[S2] 设定为在 (2B) 中说明的不同类型的 S-box 来实现对饱和攻击、对代数攻击 (XSL 攻击) 抵抗性都较高的结构。

[0433] 在图 11 ～图 13 所示的结构中也相同，

[0434] 通过将 S-box[S1]、[S2]・・设定为在 (2B) 中说明的不同类型的 S-box 来实现对饱和攻击、对代数攻击 (XSL 攻击) 抵抗性都较高的结构。

[0435] [3. 密码处理装置的结构例]

[0436] 最后，图 19 示出按照上述的实施例执行密码处理的作为密码处理装置的 IC 模块 700 的结构例。例如能够在 PC、IC 卡、读写器、其它各种信息处理装置中执行上述处理，图 19 所示的 IC 模块 700 能够构成在这些各种设备中。

[0437] 图19所示的CPU(Central processing Unit :中央处理单元)701是执行密码处理的开始、结束、数据的发送接收的控制、各结构部之间的数据传送控制、其它各种程序的处理器。存储器 702 由 ROM(Read-Only-Memory :只读存储器)、RAM(Random Access Memory :随机存取存储器) 等构成，其中，上述 ROM 保存 CPU701 所执行的程序或者运算参数等固定数据，上述 RAM 作为在 CPU701 的处理中执行的程序以及在程序处理中适当变化的参数的保存区域、工作区域而使用。另外，存储器 702 能够 作为密码处理所需的密钥数据、在密码处理中应用的变换表 (置换表)、在变换矩阵中应用的数据等的保存区域而使用。此外，数据保存区域优选构成为具有耐损害结构的存储器。

[0438] 密码处理部 703 例如按照共用密钥块密码处理算法来执行密码处理、解密处理，上述共用密钥块密码处理算法应用了上述的各种密码处理结构、即

[0439] (a)SPN(Substitution Permutation Network) 结构

[0440] (b)Feistel 结构

[0441] (c) 扩展 Feistel 结构

[0442] 这些各结构中的任一个结构。

[0443] 另外，密码处理部 703 具有作为非线性变换处理部的 S-box，该非线性变换处理部具有与上述的各实施例对应的结构、即与

[0444] (2A) 在使用了 S-box 的 Feistel 或者扩展 Feistel 型密码中配置两种以上不同的 S-box 的结构

[0445] (2B) 在使用了 S-box 的块密码中使两种以上不同类型的 S-box 共存的结构

[0446] (2C) 在使用了 S-box 的 Feistel 密码或者扩展 Feistel 型密码中同时实现上述

(2A)、(2B) 的结构

[0447] 这些结构中的任一个对应的结构。

[0448] 此外，在此示出了将密码处理单元设为单独模块的例子，但是也可以构成为不设置这种独立的密码处理模块，例如将密码处理程序保存在 ROM 中，由 CPU701 读取 ROM 保存程序来执行。

[0449] 随机数产生器 704 执行在密码处理所需的密钥的生成等中成为必需的随机数的产生处理。

[0450] 发送接收部 705 是执行与外部的数据通信的数据通信处理 部，例如执行与读写器等 IC 模块之间的数据通信，执行在 IC 模块内生成的密文的输出、或者来自外部读写器等设备的数据输入等。

[0451] 该 IC 模块 700 具有按照上述实施例的作为非线性变换处理部的 S-box 的排列，结果是具有

[0452] (2A) 通过在使用了 S-box 的 Feistel 或者扩展 Feistel 型密码中配置两种以上的不同的 S-box 来提高对饱和攻击的抵抗性的结构

[0453] (2B) 通过在使用了 S-box 的块密码中使两种以上不同的 S-box 共存来提高对代数攻击 (XSL 攻击) 的抵抗性的结构

[0454] (2C) 在使用了 S-box 的 Feistel 密码或者扩展 Feistel 型密码中同时实现上述 (2A)、(2B) 的结构

[0455] 这些结构中的某一个结构，具有提高了对饱和攻击、代数攻击 (XSL 攻击) 的抵抗性的结构。

[0456] 以上，参照特定的实施例详细说明了本发明。然而，在不脱离本发明的要旨的范围内，本领域技术人员可进行该实施例的修改、代用是显而易见的。即，以例示的方式公开了本发明，并不是限定性地解释本发明。为了判断本发明的要旨，应该参考权利要求书的范围一栏。

[0457] 此外，在说明书中说明的一系列处理能够通过硬件、或者软件、或者两者的复合结构来执行。在通过软件执行处理的情况下，能够将记录了处理顺序的程序安装到嵌入在专用硬件中的计算机内的存储器中来执行、或者将程序安装到可执行各种处理的通用计算机中来执行。

[0458] 例如，能够将程序预先记录在作为记录介质的硬盘、ROM(Read Only Memory) 中。或者，能够将程序暂时或永久性地保存 (记录) 在软盘、CD-ROM(Compact Disc Read Only Memory ：光盘只读存储器)、MO(Magneto optical ：磁光) 盘、DVD(Digital Versatile Disc ：数字多功能光盘)、磁盘、半导体存储器等可移动记录介质中。能够将这种可移动记录介质作为所谓的封装软件进行提供。

[0459] 此外，除了将程序从如上所述的可移动记录介质安装到计算机以外，还能够从下载站点无线传送到计算机、或者通过称为 LAN(Local Area Network ：局域网)、因特网这种网络来有线传送到计算机，在计算机中接收这样传送过来的程序并安装到内置的硬盘等记录介质中。

[0460] 此外，不仅可以按照记载以时间序列执行，也可以根据执行处理的装置的处理能力或需要来并行或单独地执行说明书中记载的各种处理。另外，在本说明书中系统是多个

装置的逻辑集合结构，并不限于各结构的装置在同一壳体内。

[0461] 产业上的可利用性

[0462] 如上所述，根据本发明的一个实施例的结构，在执行共用密钥块密码处理的密码处理装置中，设为在设定于循环函数执行部中的作为非线性变换处理部的 S 盒中至少利用两种以上多个不同的 S 盒的结构。根据本结构，能够提高对饱和攻击的抵抗性。另外，根据作为 S 盒的类型而使不同类型的 S 盒共存的本发明的一个实施例的结构，能够提高对代数攻击（XSL 攻击）的抵抗性，从而实现安全性较高的密码处理装置。

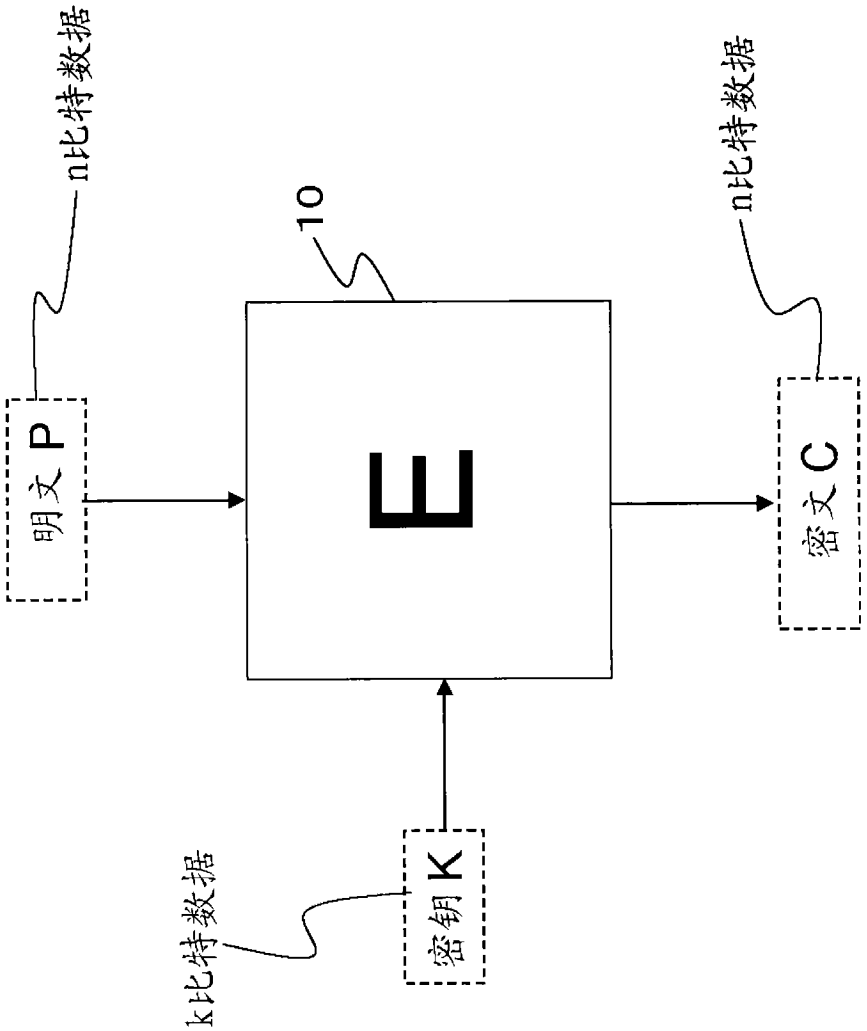


图 1

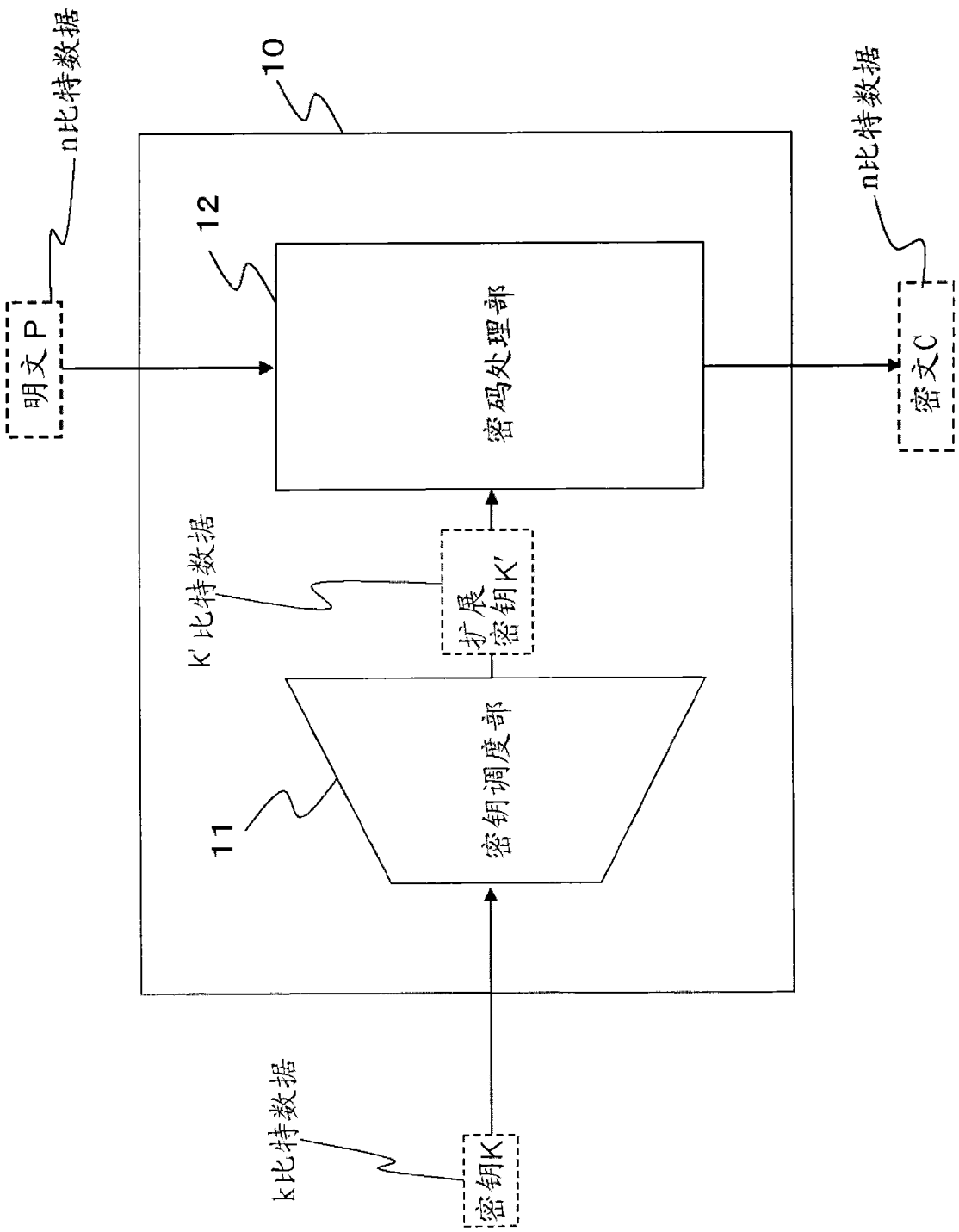

n比特数据
明文P
10
12
密码处理部
n比特数据
密文C
K'比特数据
扩展密钥K'
密钥调度部
11
K比特数据
密钥K

图 2

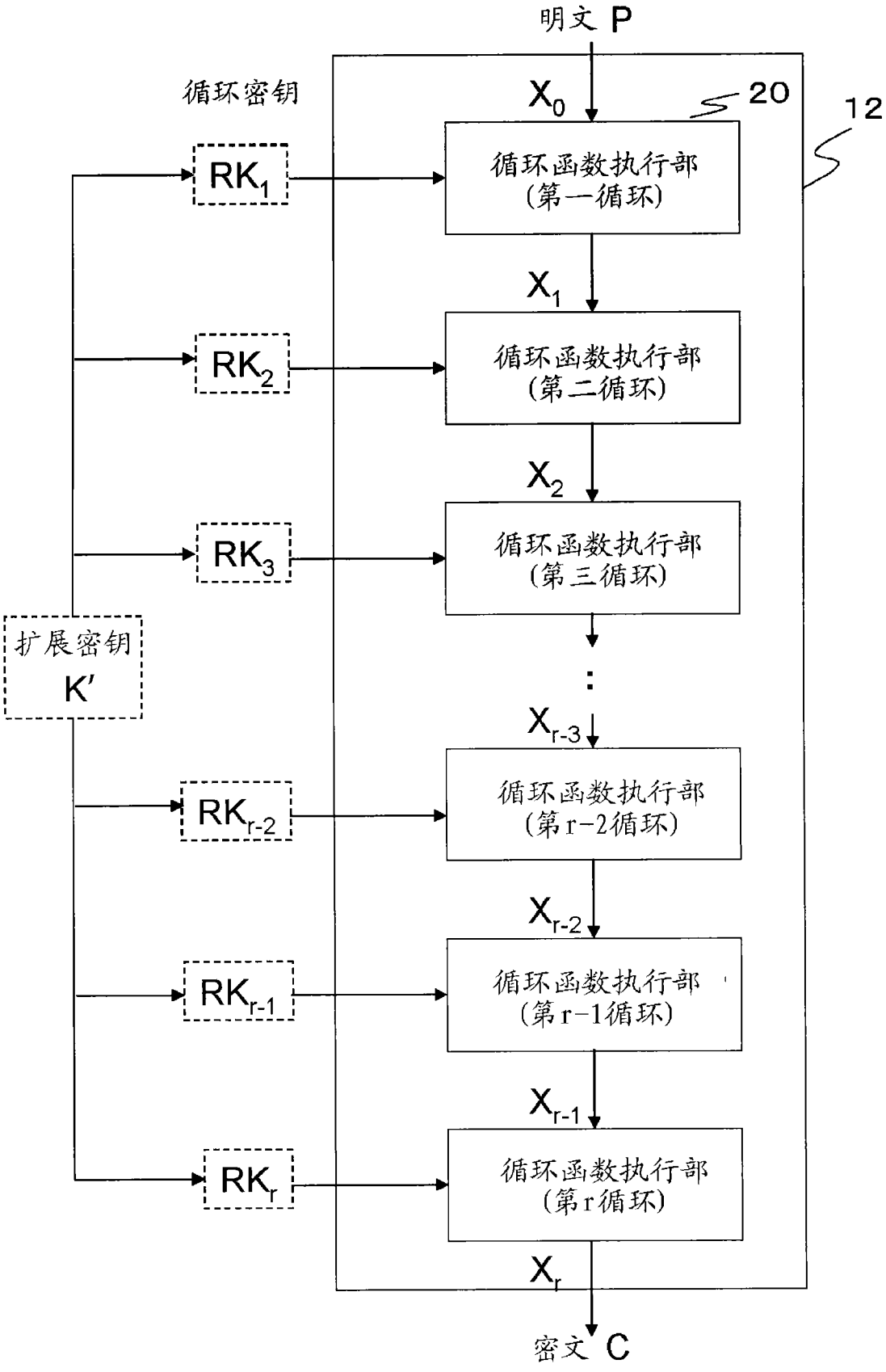

明文 P
循环密钥
X_0
20
12
RK_1
循环函数执行部
(第一循环)
X_1
RK_2
循环函数执行部
(第二循环)
X_2
RK_3
循环函数执行部
(第三循环)
扩展密钥
K'
X_{r-3}
RK_{r-2}
循环函数执行部
(第r-2循环)
X_{r-2}
RK_{r-1}
循环函数执行部
(第r-1循环)
X_{r-1}
RK_r
循环函数执行部
(第r循环)
X_r
密文 C

图 3

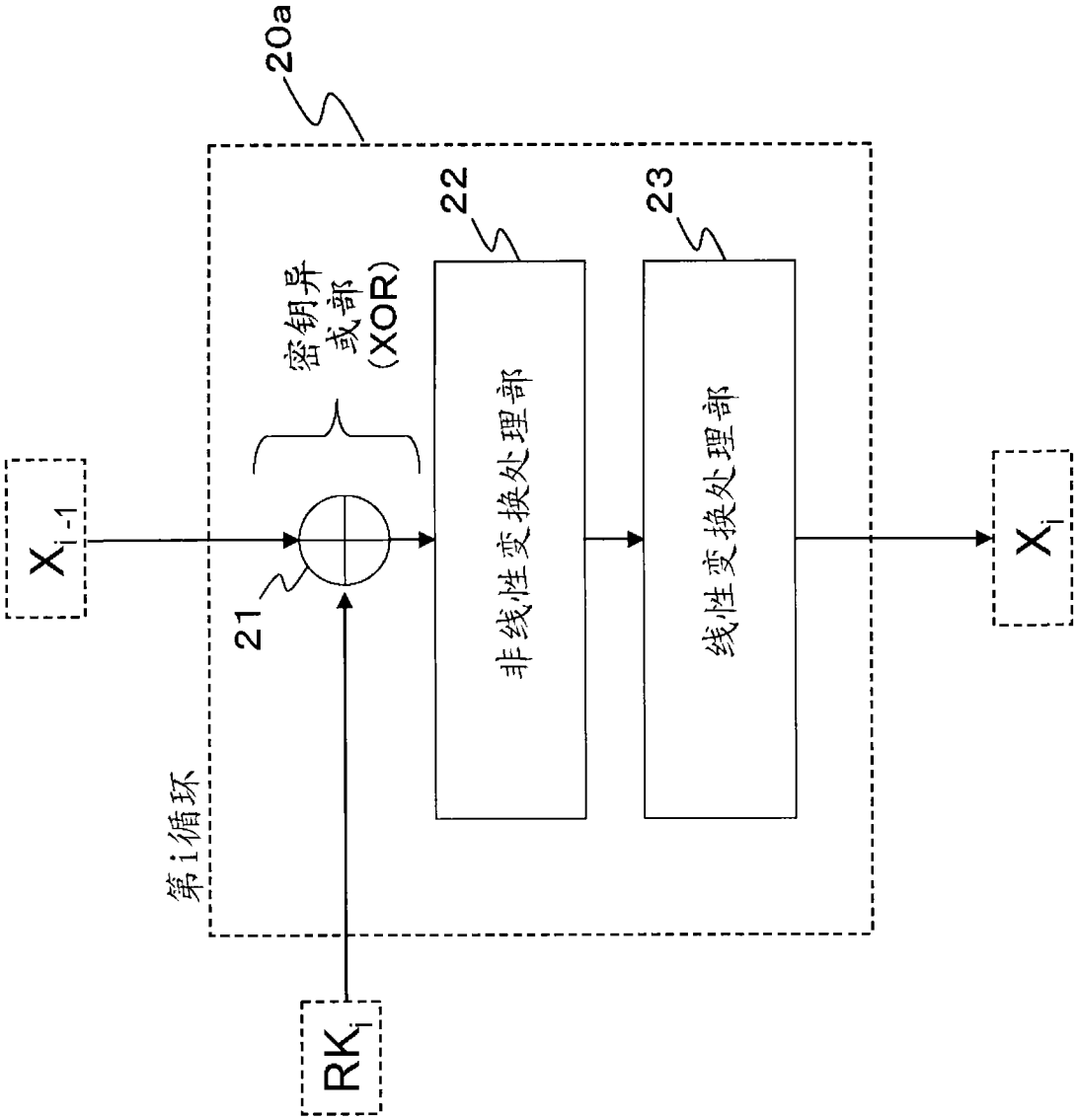


图 4

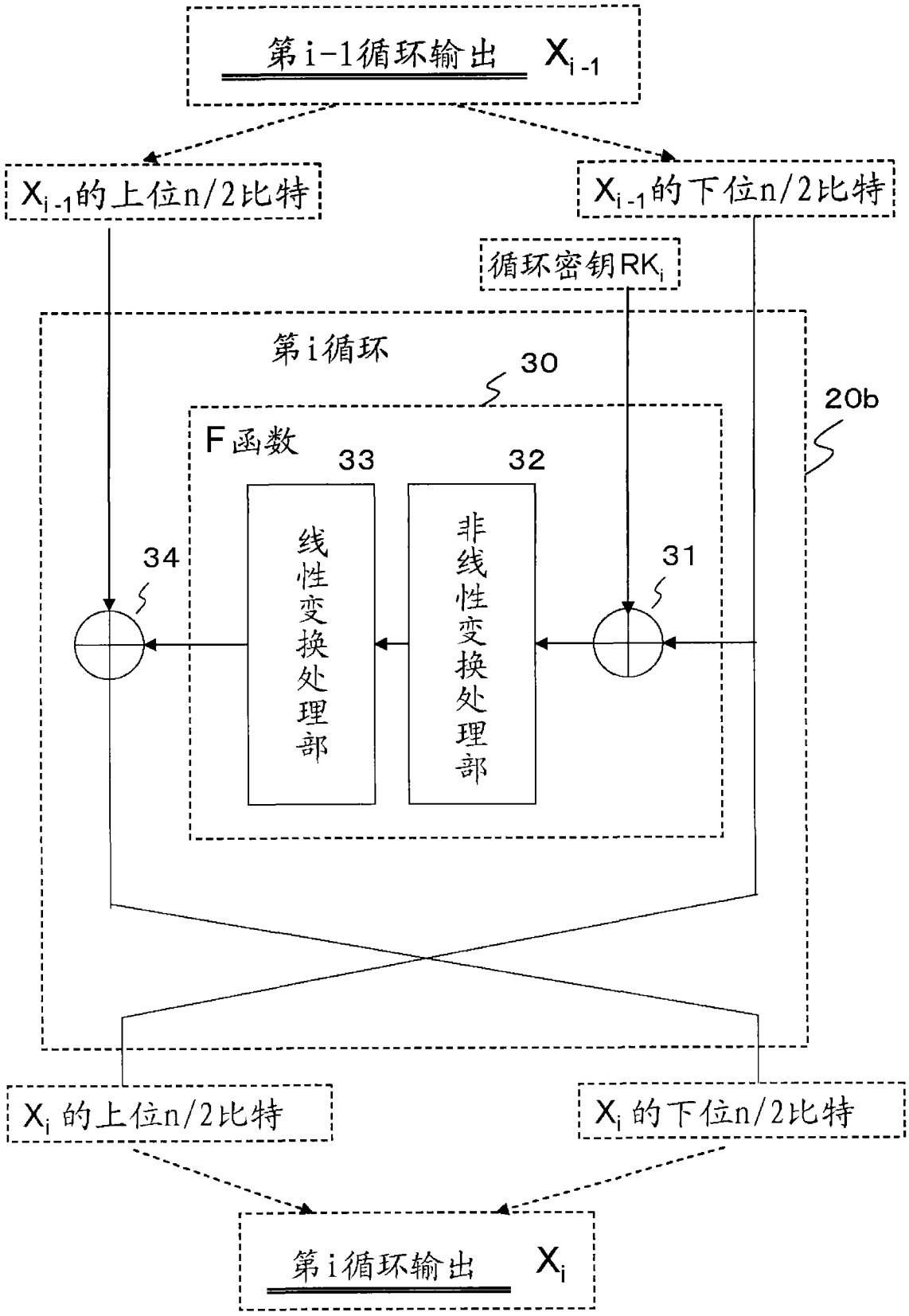

第i-1循环输出 X_{i-1}
X_{i-1}的上位n/2比特
X_{i-1}的下位n/2比特
循环密钥RK_i
第i循环
30
20b
F函数
33
32
线性变换处理部
非线性变换处理部
34
31
X_i的上位n/2比特
X_i的下位n/2比特
第i循环输出 X_i

图5

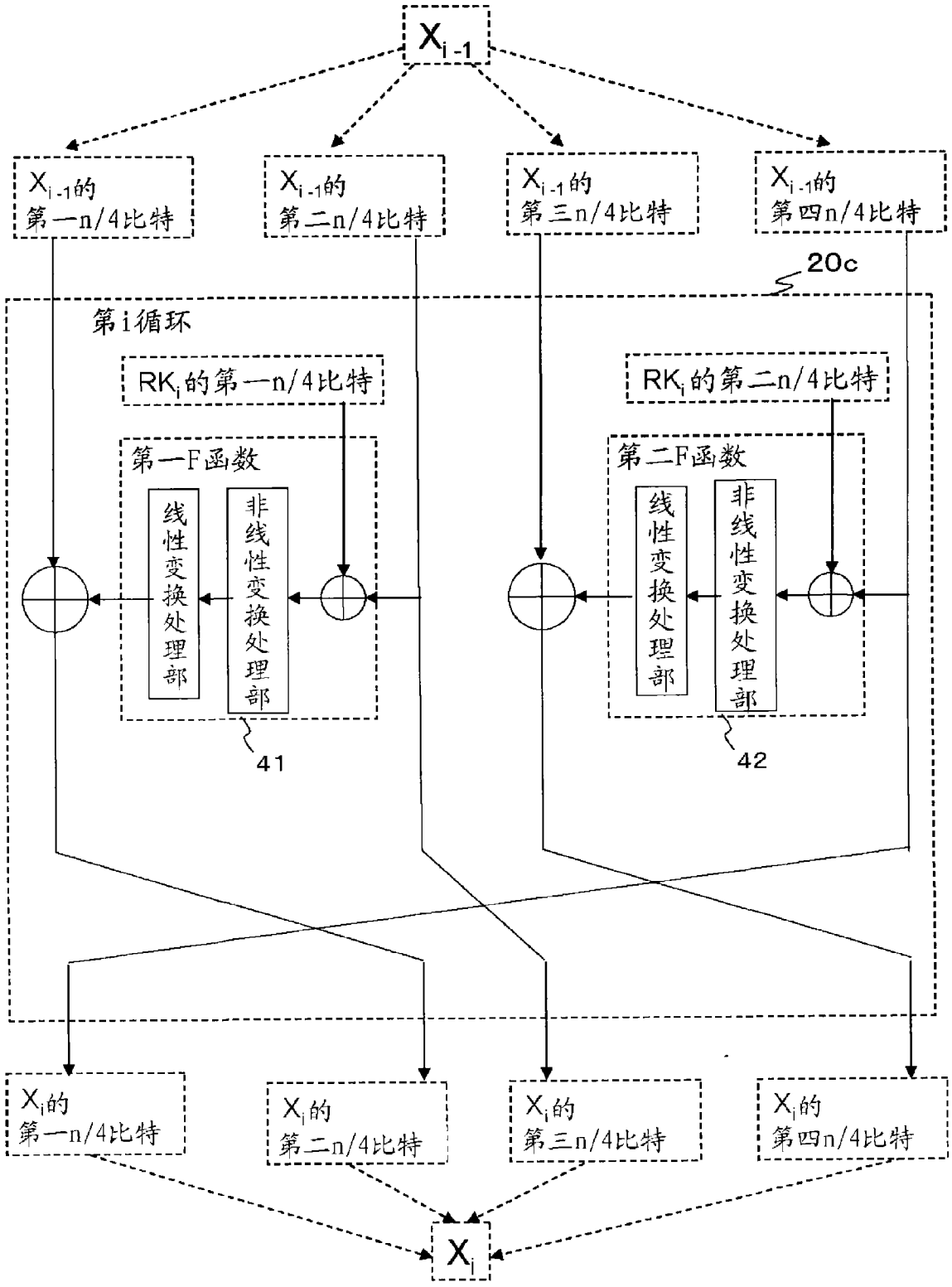

X_{i-1}
X_{i-1}的第一n/4比特
X_{i-1}的第二n/4比特
X_{i-1}的第三n/4比特
X_{i-1}的第四n/4比特
20c
第i循环
RK_i的第一n/4比特
RK_i的第二n/4比特
第一F函数
第二F函数
线性变换处理部
非线性变换处理部
线性变换处理部
非线性变换处理部
41
42
X_i的第一n/4比特
X_i的第二n/4比特
X_i的第三n/4比特
X_i的第四n/4比特
X_i

图 6

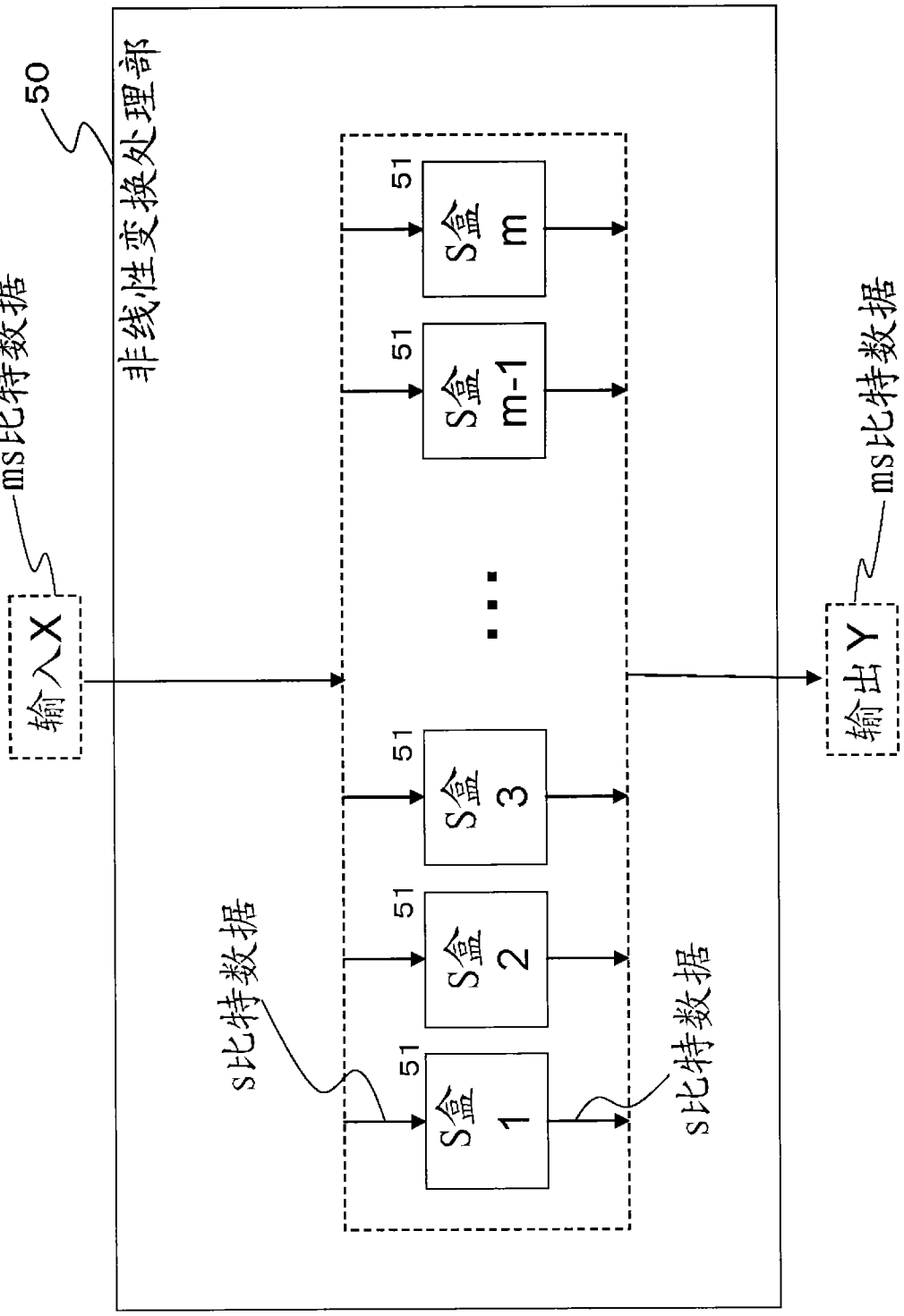

50
非线性变换处理部
ms比特数据
输入X
S盒 1
S盒 2
S盒 3
S盒 m-1
S盒 m
51
S比特数据
S比特数据
ms比特数据
输出Y

图 7

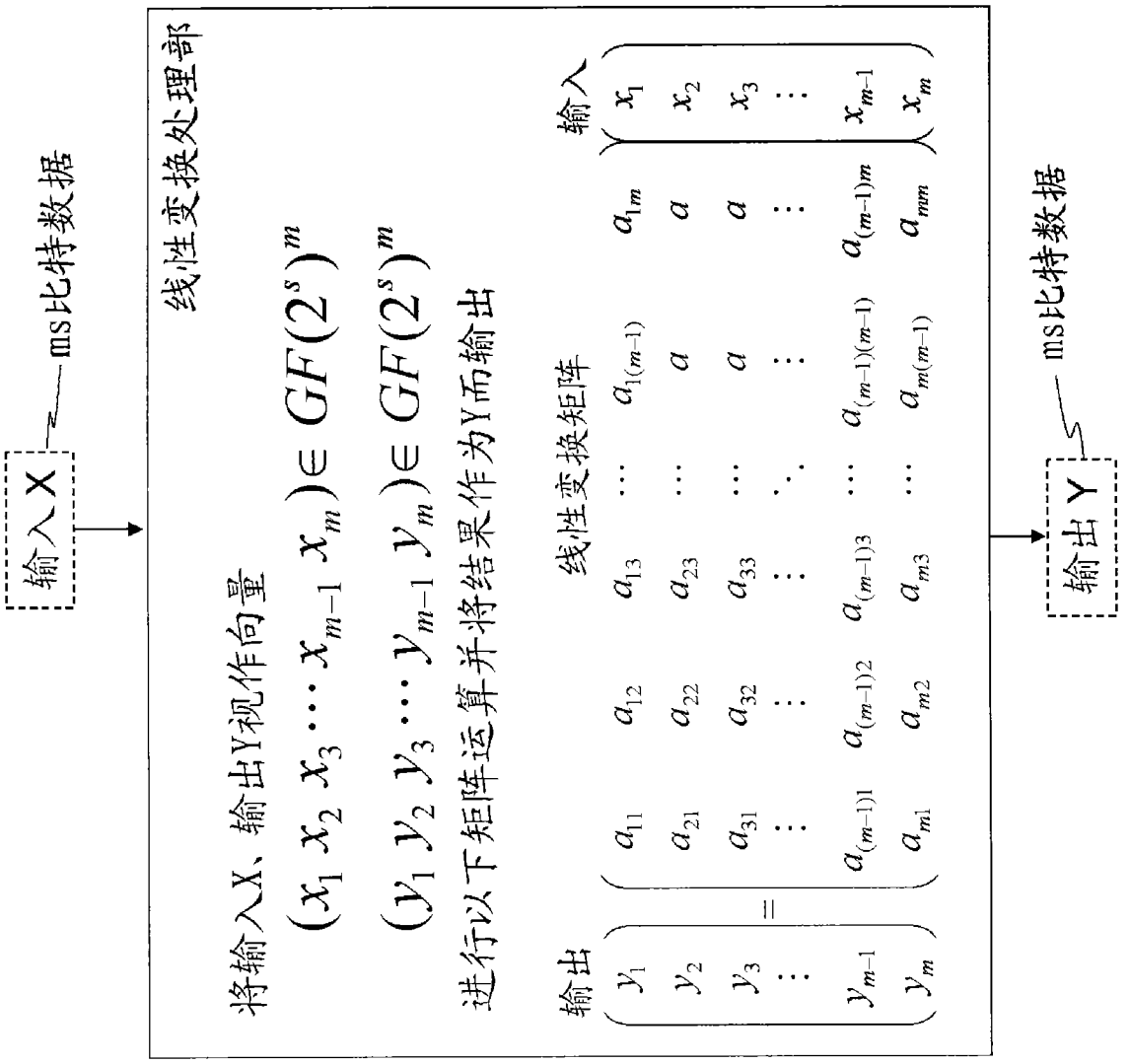

输入X
ms比特数据
线性变换处理部
将输入X、输出Y视作向量
$(x_1\ x_2\ x_3 \cdots x_{m-1}\ x_m) \in GF(2^s)^m$
$(y_1\ y_2\ y_3 \cdots y_{m-1}\ y_m) \in GF(2^s)^m$
进行以下矩阵运算并将结果作为Y而输出
$\begin{pmatrix} y_1 \\ y_2 \\ y_3 \\ \vdots \\ y_{m-1} \\ y_m \end{pmatrix} = \begin{pmatrix} a_{11} & a_{12} & a_{13} & \cdots & a_{1(m-1)} & a_{1m} \\ a_{21} & a_{22} & a_{23} & \cdots & a & a \\ a_{31} & a_{32} & a_{33} & \cdots & a & a \\ \vdots & \vdots & \vdots & \ddots & \vdots & \vdots \\ a_{(m-1)1} & a_{(m-1)2} & a_{(m-1)3} & \cdots & a_{(m-1)(m-1)} & a_{(m-1)m} \\ a_{m1} & a_{m2} & a_{m3} & \cdots & a_{m(m-1)} & a_{mm} \end{pmatrix} \begin{pmatrix} x_1 \\ x_2 \\ x_3 \\ \vdots \\ x_{m-1} \\ x_m \end{pmatrix}$
输出
线性变换矩阵
输入
输出Y
ms比特数据

图 8

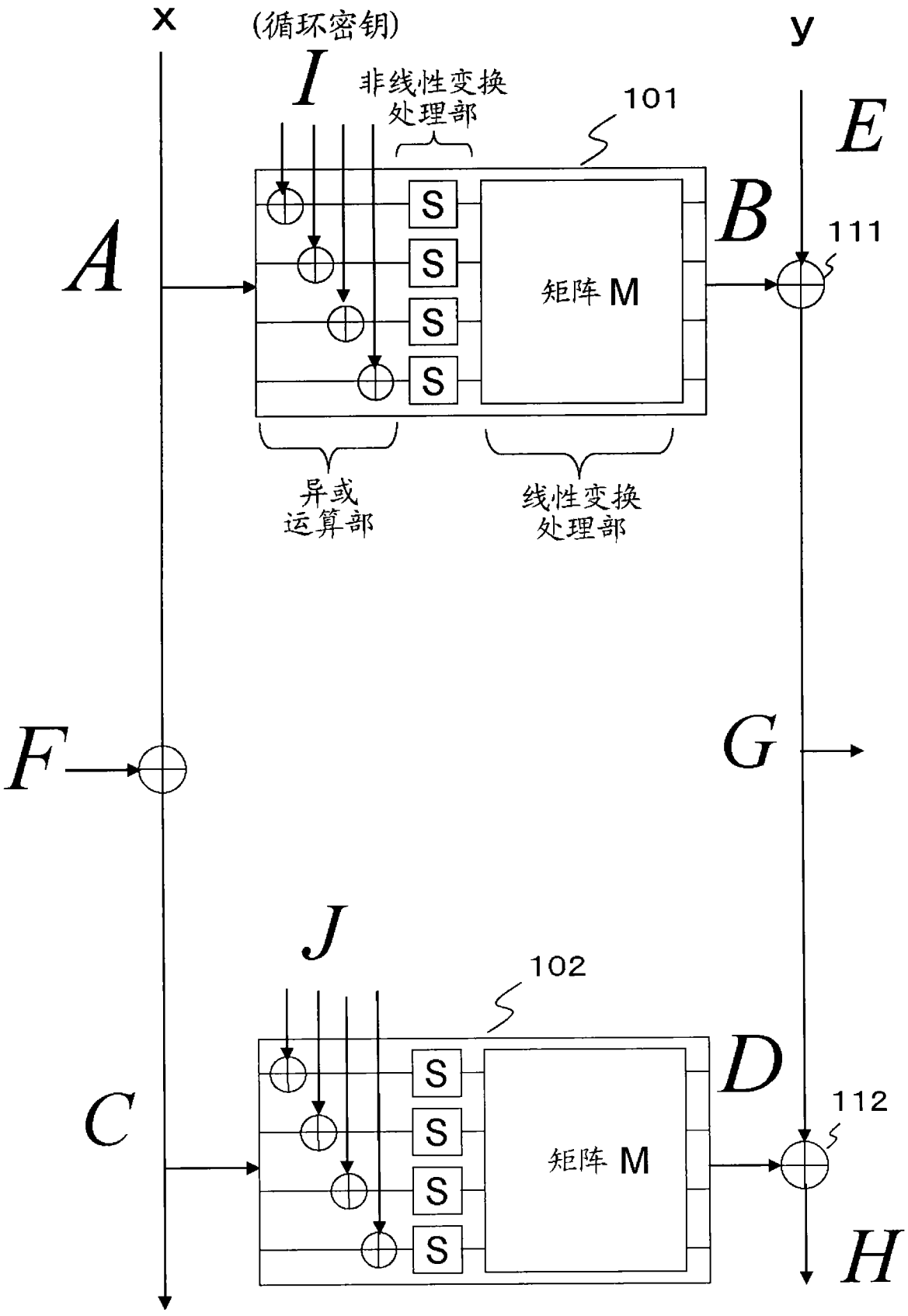

x
(循环密钥)
y
I
非线性变换
处理部
101
E
A
B
111
S
S
S
S
矩阵 M
异或
运算部
线性变换
处理部
F
G
J
102
C
D
112
矩阵 M
H

图 9

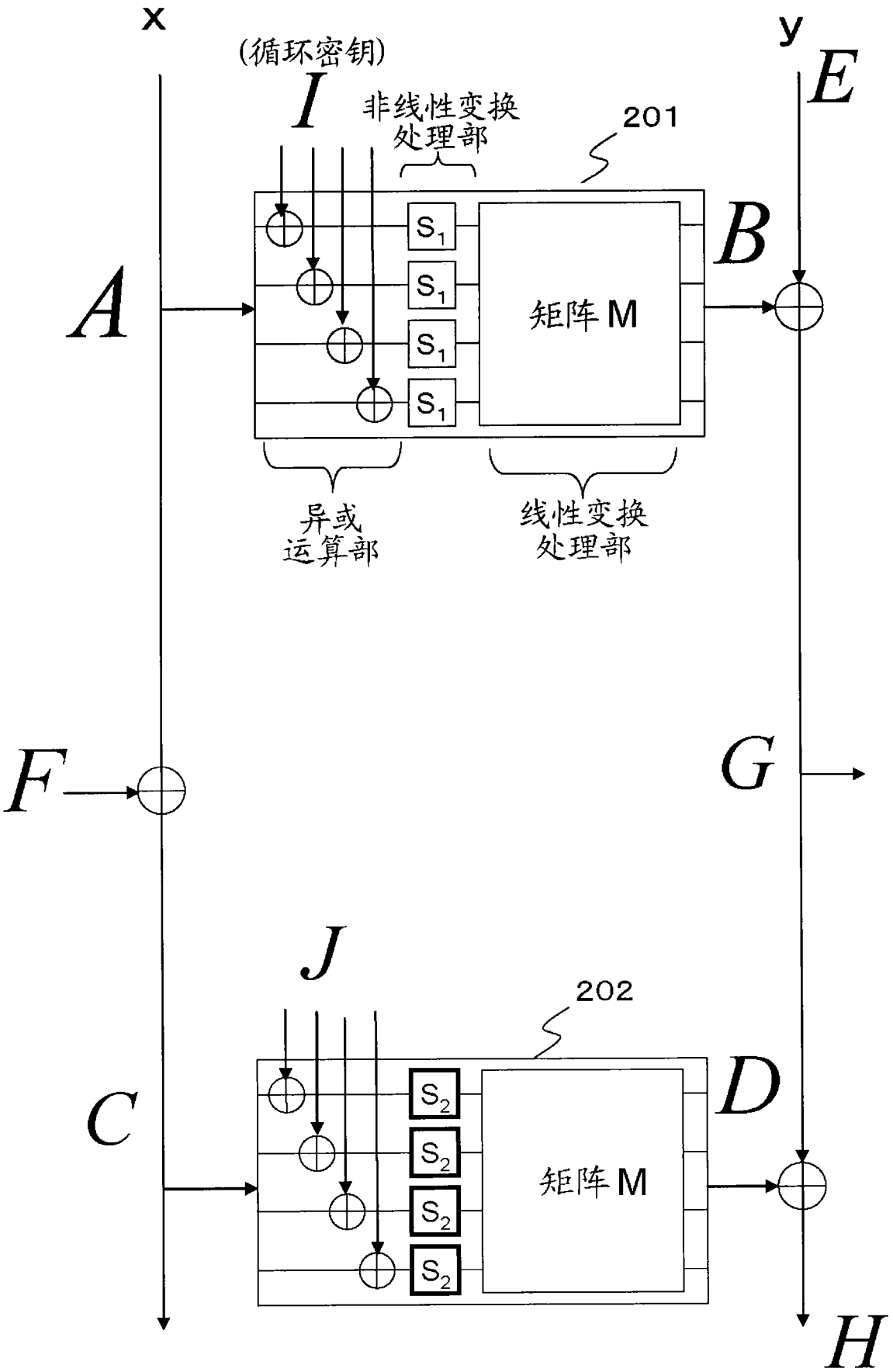

x
(循环密钥)
I
非线性变换
处理部
201
y
E
A
B
S1
S1
S1
S1
矩阵 M
异或
运算部
线性变换
处理部
F
G
J
202
C
D
S2
S2
S2
S2
矩阵 M
H

图 10

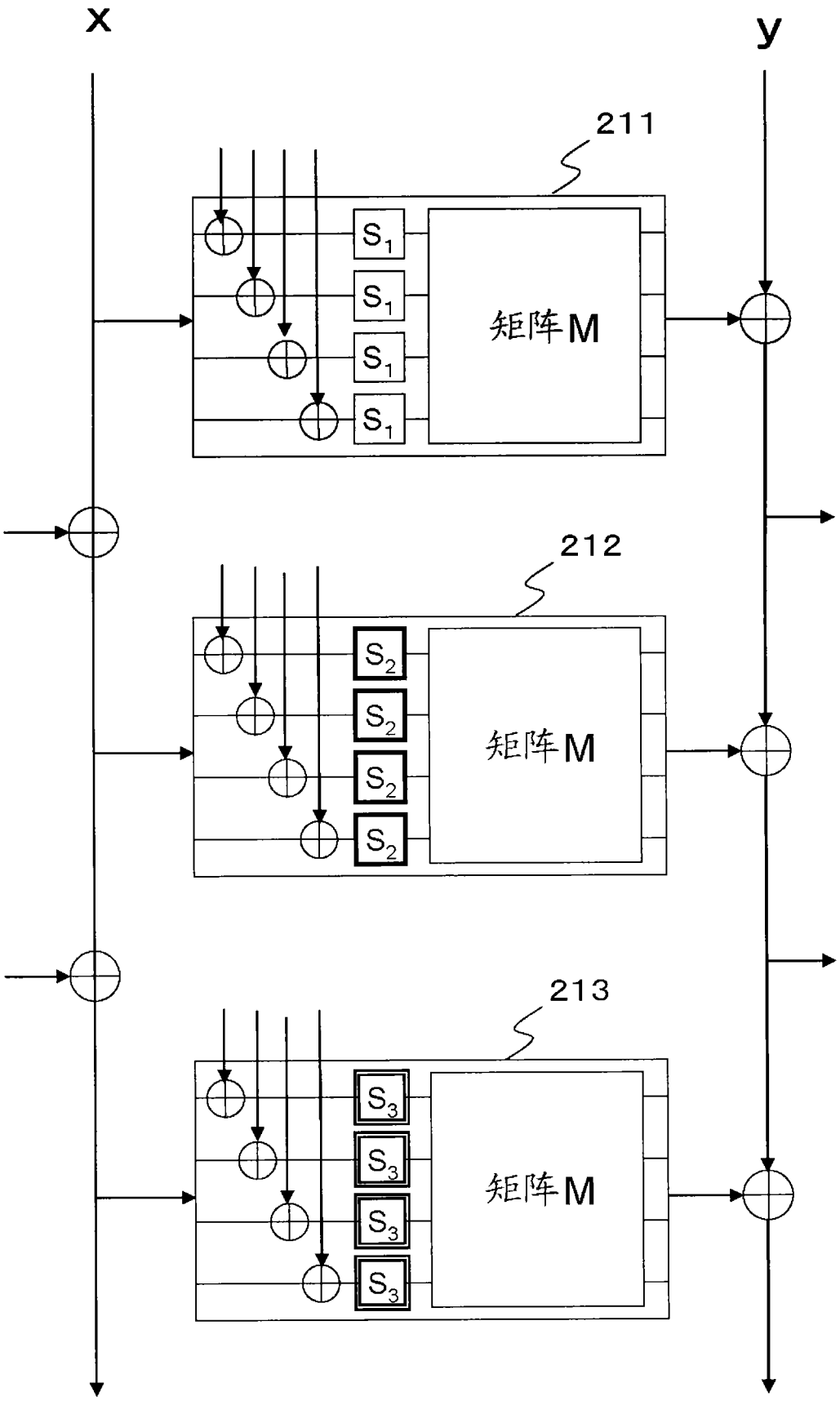


图 11

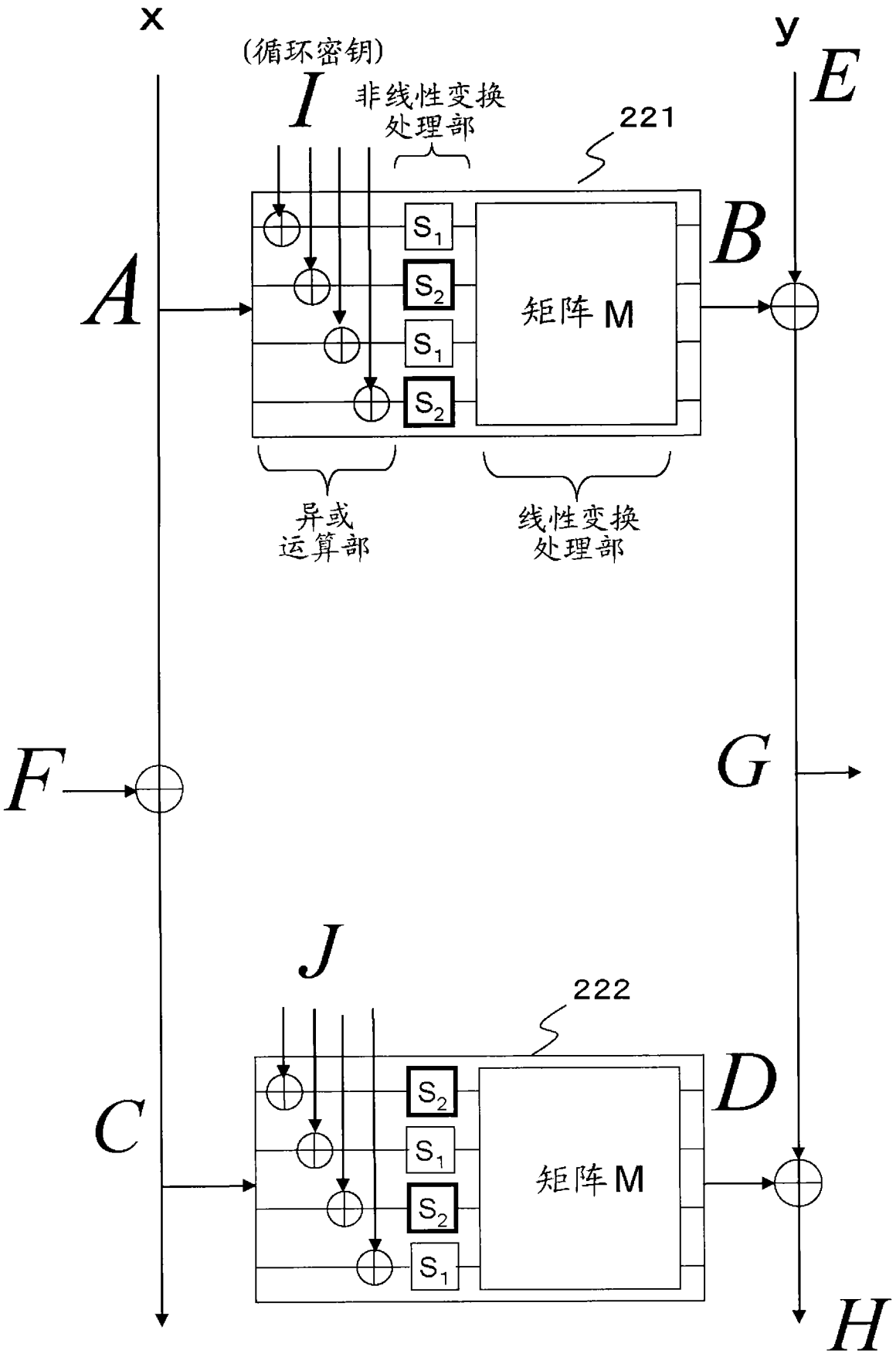

x
(循环密钥)
I
非线性变换
处理部
221
y
E
A
B
S1
S2
S1
S2
矩阵 M
异或
运算部
线性变换
处理部
F
G
J
222
C
D
S2
S1
S2
S1
矩阵 M
H

图 12

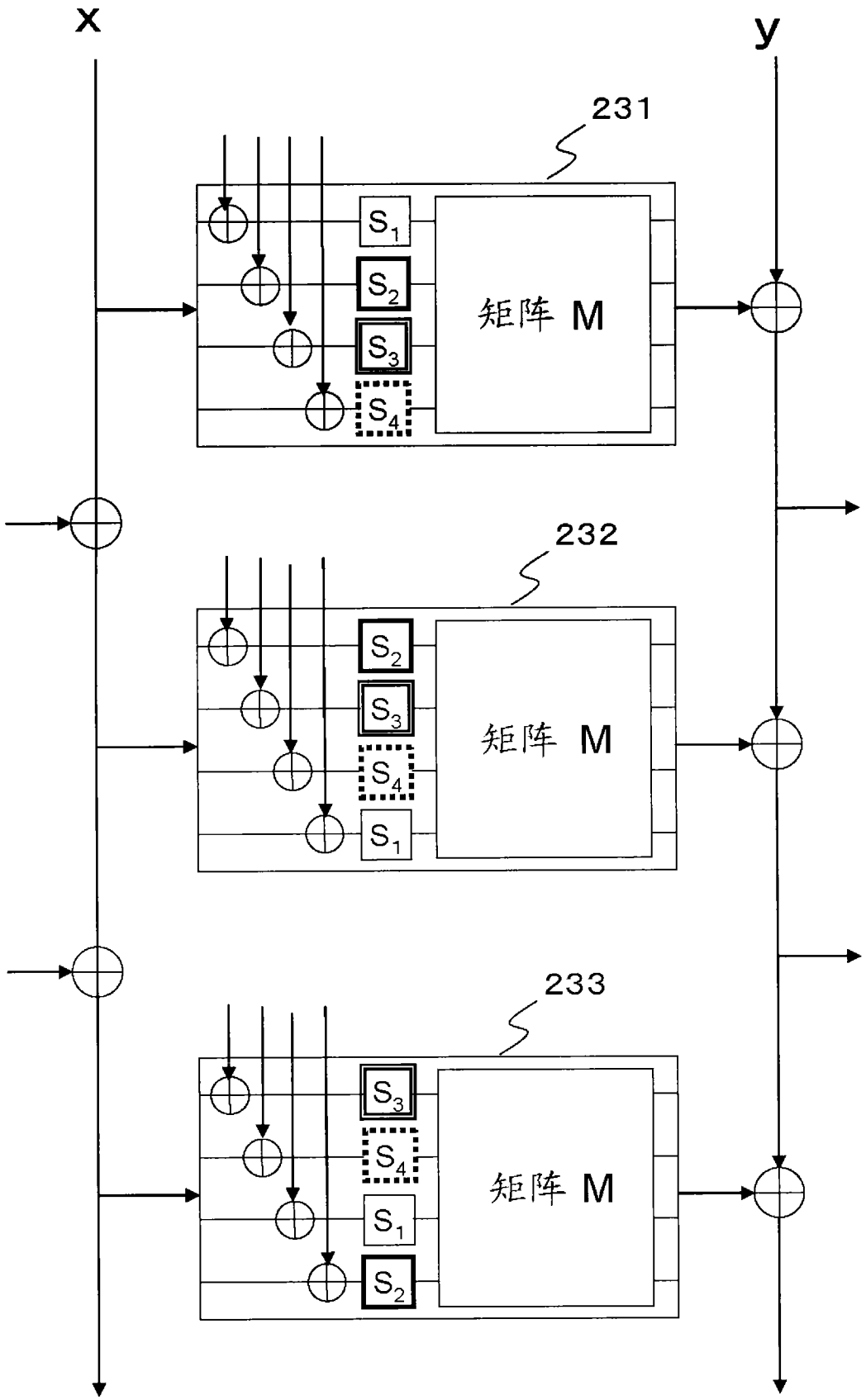


图 13

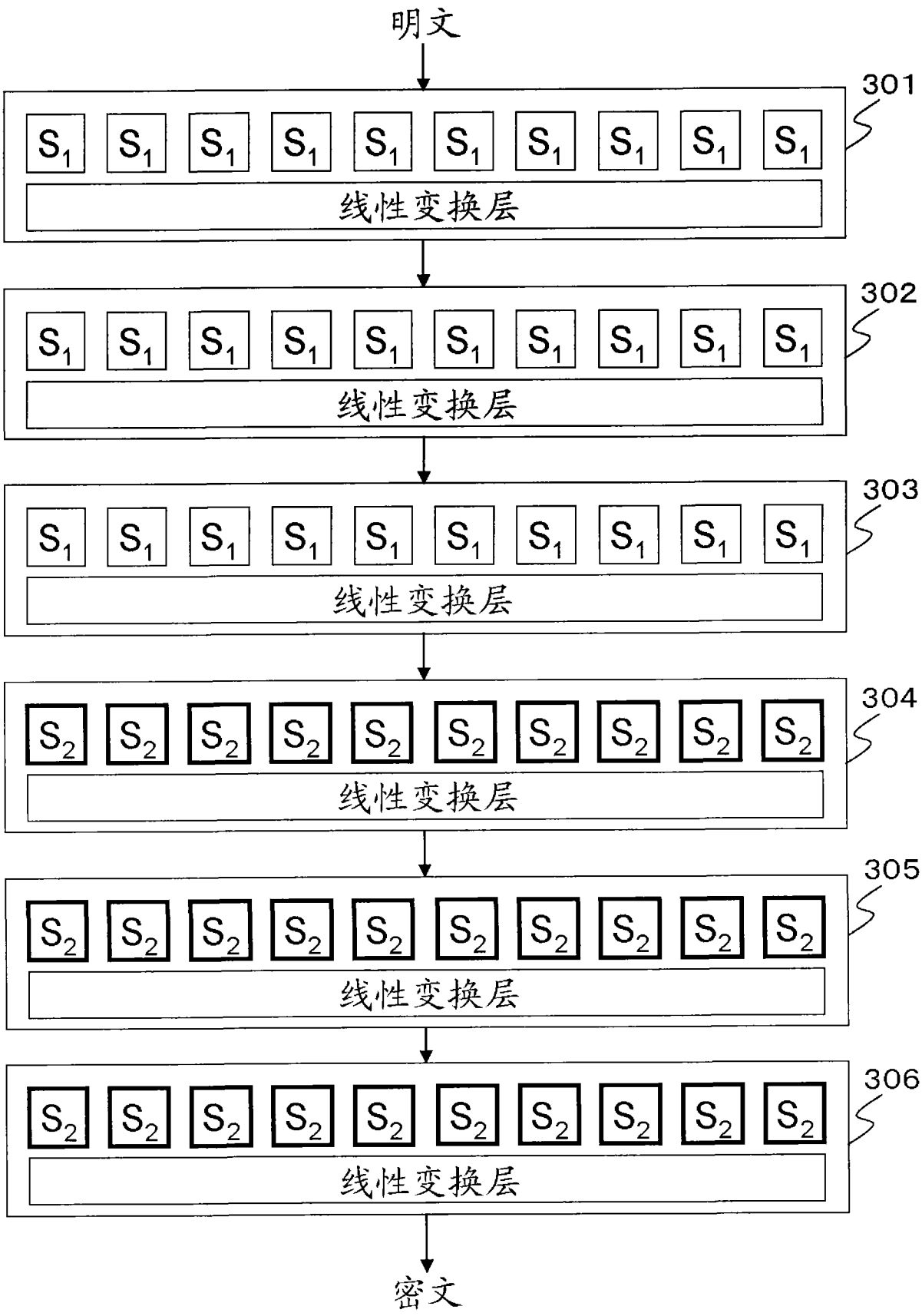

明文
301
S1 S1 S1 S1 S1 S1 S1 S1 S1 S1
线性变换层
302
S1 S1 S1 S1 S1 S1 S1 S1 S1 S1
线性变换层
303
S1 S1 S1 S1 S1 S1 S1 S1 S1 S1
线性变换层
304
S2 S2 S2 S2 S2 S2 S2 S2 S2 S2
线性变换层
305
S2 S2 S2 S2 S2 S2 S2 S2 S2 S2
线性变换层
306
S2 S2 S2 S2 S2 S2 S2 S2 S2 S2
线性变换层
密文

图 14

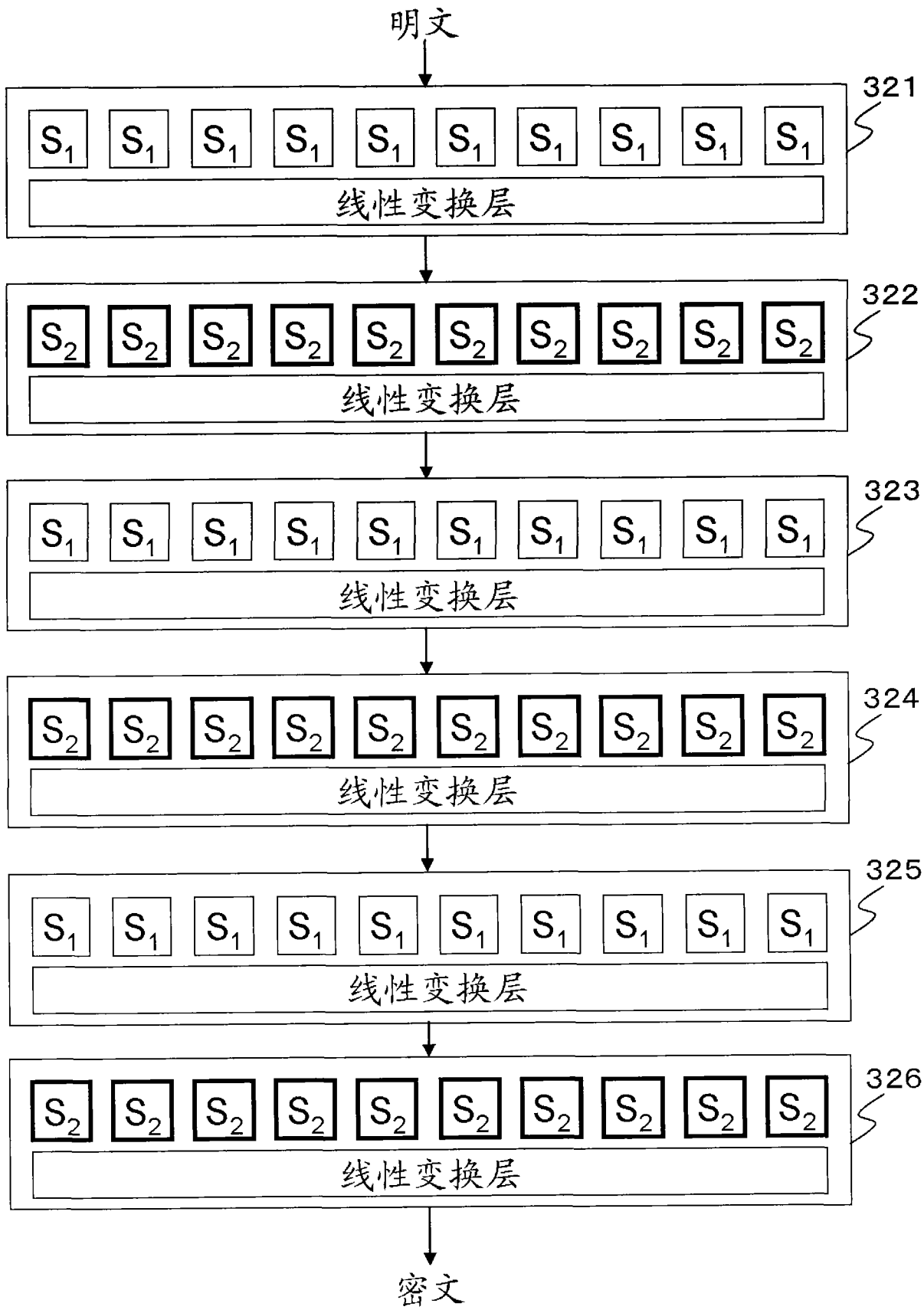

明文
321
S1 S1 S1 S1 S1 S1 S1 S1 S1 S1
线性变换层
322
S2 S2 S2 S2 S2 S2 S2 S2 S2 S2
线性变换层
323
S1 S1 S1 S1 S1 S1 S1 S1 S1 S1
线性变换层
324
S2 S2 S2 S2 S2 S2 S2 S2 S2 S2
线性变换层
325
S1 S1 S1 S1 S1 S1 S1 S1 S1 S1
线性变换层
326
S2 S2 S2 S2 S2 S2 S2 S2 S2 S2
线性变换层
密文

图 15

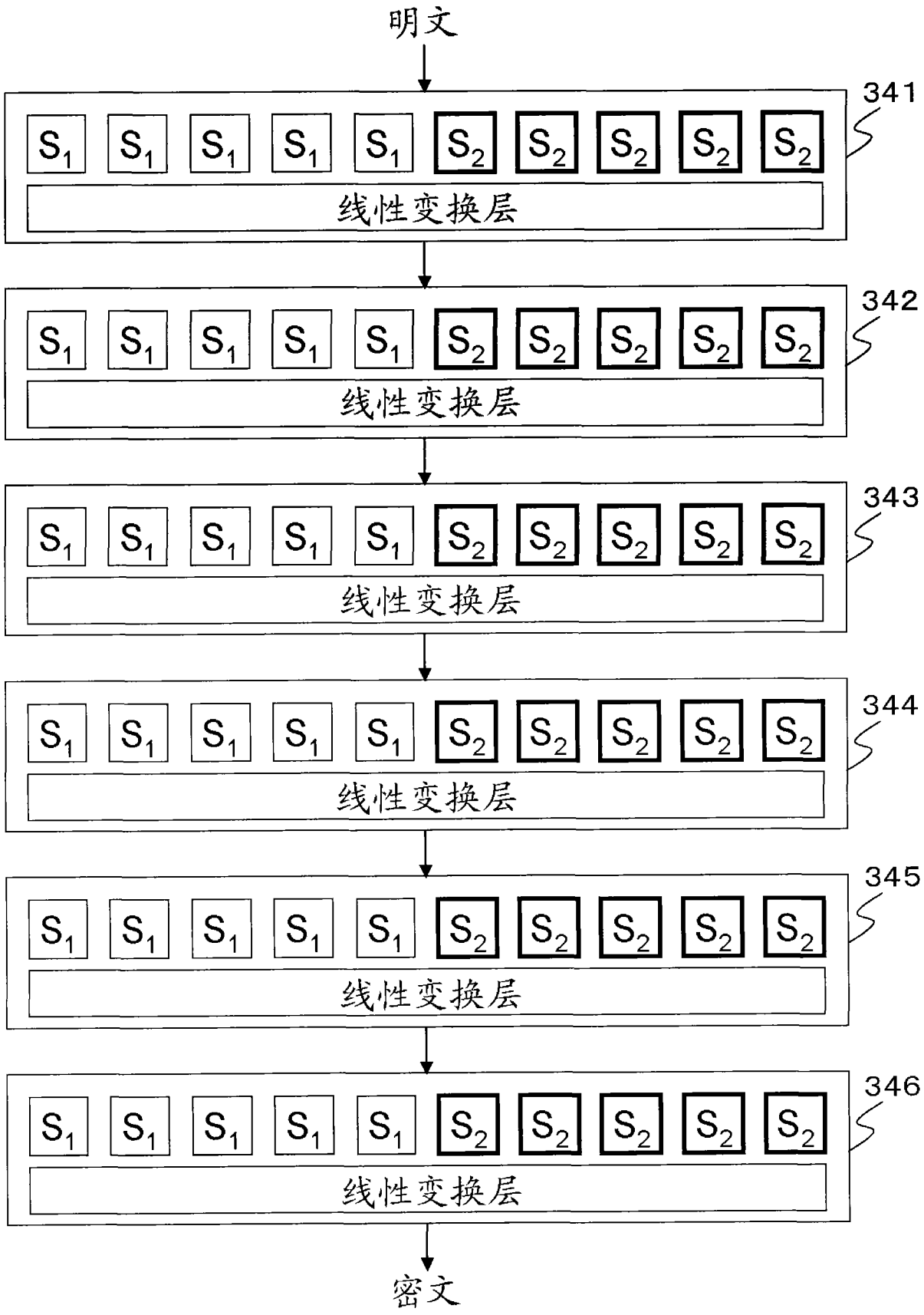

明文
341
S1 S1 S1 S1 S1 S2 S2 S2 S2 S2
线性变换层
342
S1 S1 S1 S1 S1 S2 S2 S2 S2 S2
线性变换层
343
S1 S1 S1 S1 S1 S2 S2 S2 S2 S2
线性变换层
344
S1 S1 S1 S1 S1 S2 S2 S2 S2 S2
线性变换层
345
S1 S1 S1 S1 S1 S2 S2 S2 S2 S2
线性变换层
346
S1 S1 S1 S1 S1 S2 S2 S2 S2 S2
线性变换层
密文

图 16

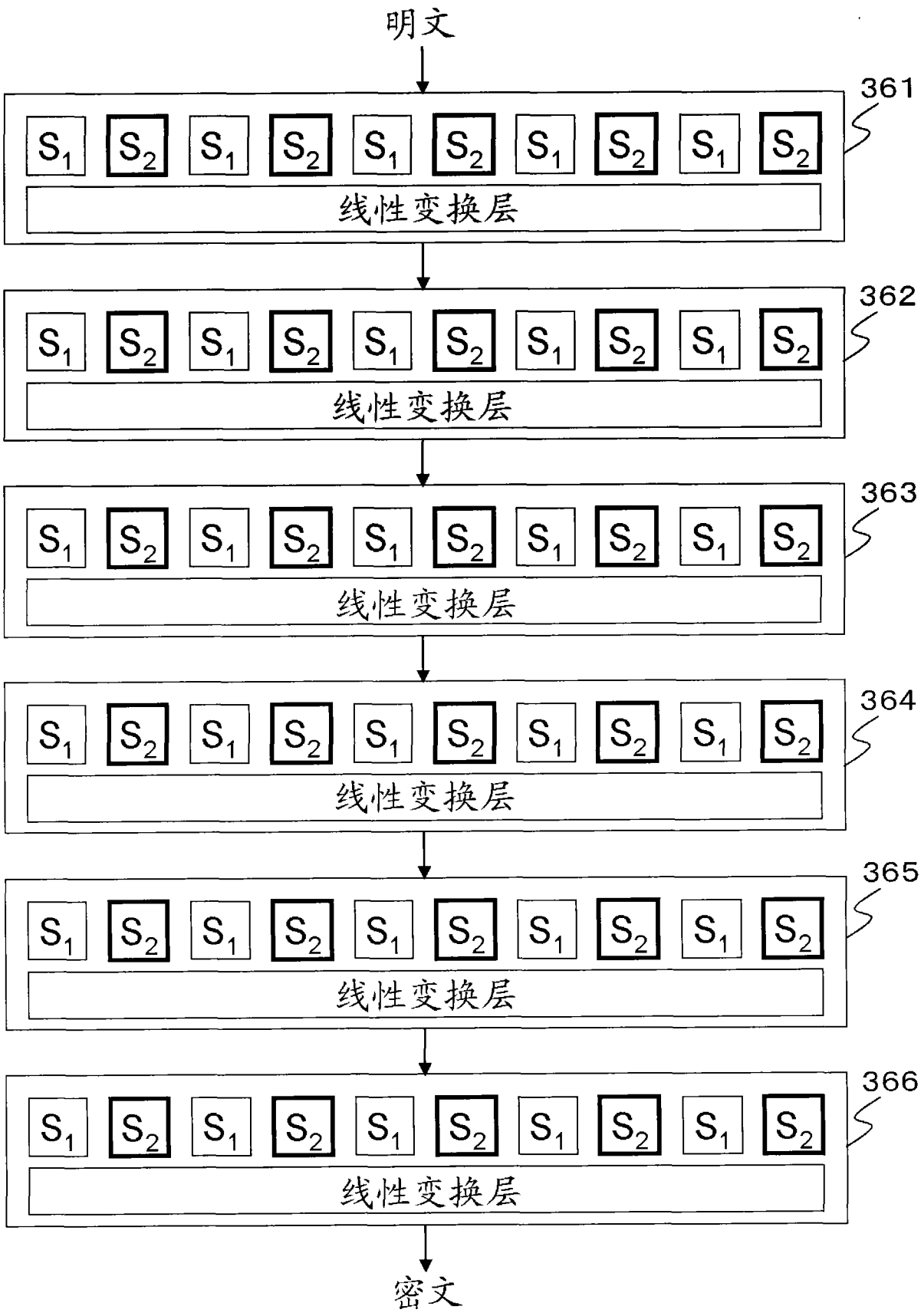

明文
361
S1 S2 S1 S2 S1 S2 S1 S2 S1 S2
线性变换层
362
S1 S2 S1 S2 S1 S2 S1 S2 S1 S2
线性变换层
363
S1 S2 S1 S2 S1 S2 S1 S2 S1 S2
线性变换层
364
S1 S2 S1 S2 S1 S2 S1 S2 S1 S2
线性变换层
365
S1 S2 S1 S2 S1 S2 S1 S2 S1 S2
线性变换层
366
S1 S2 S1 S2 S1 S2 S1 S2 S1 S2
线性变换层
密文

图 17

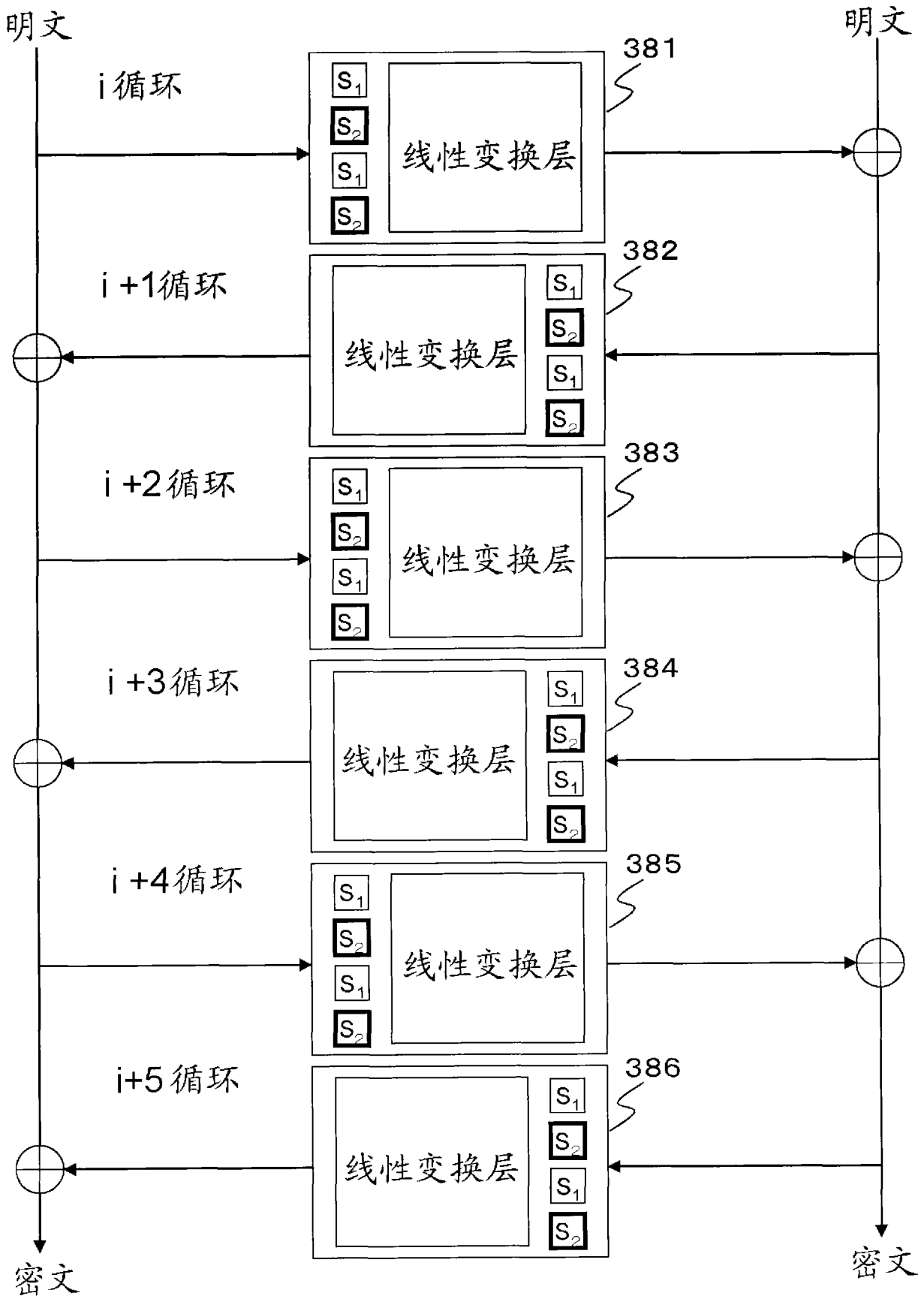

明文
明文
i循环
381
S1
S2
S1
S2
线性变换层
i +1循环
382
线性变换层
i +2循环
383
线性变换层
i +3循环
384
线性变换层
i +4循环
385
线性变换层
i+5循环
386
线性变换层
密文
密文

图 18

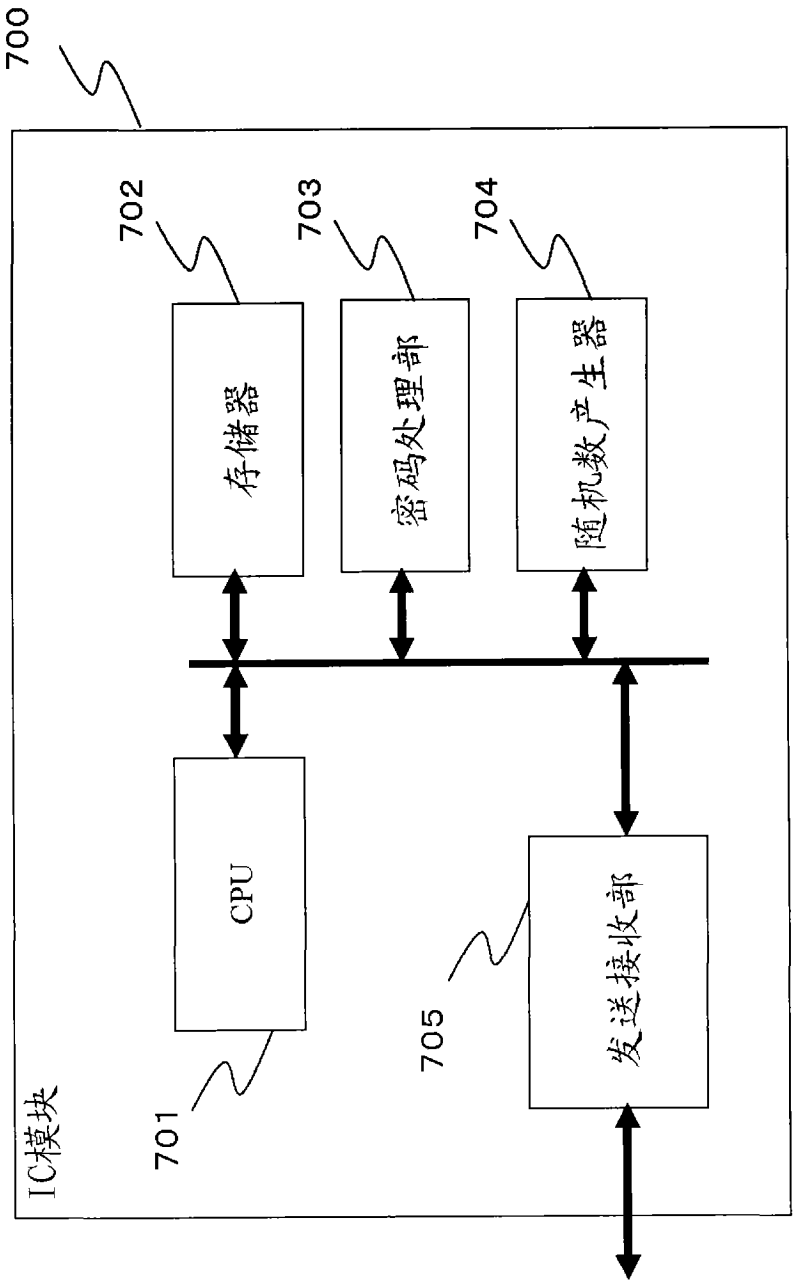


图 19