[19] 中华人民共和国国家知识产权局

[51] Int. Cl.
H04L 9/22 (2006.01)

[12] 发明专利申请公布说明书

[21] 申请号 200680023071.7

[43] 公开日 2008 年 6 月 25 日

[11] 公开号 CN 101208902A

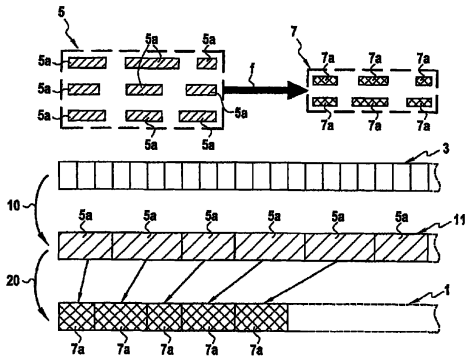
[22] 申请日 2006.5.23
[21] 申请号 200680023071.7
[30] 优先权
[32] 2005. 5.26 [33] FR [31] 0505296
[86] 国际申请 PCT/FR2006/050472 2006.5.23
[87] 国际公布 WO2007/000549 法 2007.1.4
[85] 进入国家阶段日期 2007.12.26
[71] 申请人 法国电信公司
地址 法国巴黎
[72] 发明人 A·古热 H·赛伯特
C·贝尔拜恩
[74] 专利代理机构 北京市中咨律师事务所
代理人 杨晓光 李 峥
权利要求书 3 页 说明书 14 页 附图 3 页

[54] 发明名称

用于生成伪随机数据序列的方法、系统和设备

[57] 摘要

本发明涉及一种从初始数据流(3)生成伪随机数据序列(1)的设备和方法，所述方法特征在于包括以下步骤：定义形成完全前缀码(5)的码字的集合；定义输出字(7)的集合；定义去同步函数(f)，所述去同步函数将来自所述输出字(7)的集合的输出字与所述完全前缀码(5)中的任意码字关联；将初始数据流(3)分解为根据所述完全前缀码(5)编码的字的系列(11)；依照所述去同步函数(f)将所述已编码的字的系列(11)中的字与对应输出字关联，以形成所述伪随机数据序列(1)。



1. 一种从初始数据流（3）生成伪随机数据序列（1）的方法，其特征在于包括以下步骤:

.定义形成完全前缀码（5）的码字的集合;

.定义输出字（7）的集合;

.定义去同步函数（f），所述去同步函数将来自所述输出字（7）的集合的输出字与所述完全前缀码（5）中的任意码字关联;

.将初始数据流（3）分解为根据所述完全前缀码（5）编码的字（11）的系列;

.根据所述去同步函数（f）将所述已编码的字（11）的系列中的字与对应输出字关联，以形成所述伪随机数据序列（1）。

2. 如权利要求1所述的方法，其特征在于，所述去同步函数（f）是取决于预定参数的参数化函数，可以在生成所述伪随机数据序列（1）期间修改所述预定参数的值。

3. 如权利要求所述的方法1，其特征在于，所述输出字（7）的集合包括每一输出字的补码输出字。

4. 如权利要求1至3中的任意一项所述的方法，其特征在于，所述完全前缀码（5）的码字具有受上限h限制的长度，并且其特征在于，如果任意输出字x具有与另一个输出字y相同的长度，则按照输出字x的所述函数的前项的数量与按照输出字y的所述函数的前项的数量相同。

5. 如权利要求4所述的方法，其特征在于，所述完全前缀码（5）的字具有受下限m限制的长度，并且其特征在于，存在所述完全前缀码（5）的字的上限长度l，从而对于小于或等于上限长度l的任意长度k，所述完全前缀码包含长度k的2^{m-1}个字。

6. 如权利要求4或5所述的方法，其特征在于:

.由码字的集合C_1来定义完全前缀码（5），所述码字具有以下形式，其中，$h \geqslant 2$:

$C_1=\{01^n0;0\leqslant n\leqslant h-2\}\cup\{10^n1;0\leqslant n\leqslant h-2\}\cup\{01^{h-1}\}\cup\{10^{h-1}\}$

·由来自二进制集合 $E_1=\{0,1\}$的字来定义输出字（7）的集合；

·参数化去同步函数（f）的预定参数 u 是矢量 $u=(u_0,\ldots,u_{h-1})$，其分量属于集合{0,1}；以及

·按以下方式来定义参数化去同步函数 f_u：

·$f_u(01^n0)=u_n$，对于 $0\leqslant n\leqslant h-2$；

·$f_u(10^n1)=u_n\oplus 1$，对于 $0\leqslant n\leqslant h-2$；

·$f_u(01^{h-1})=u_{h-1}$；

·$f_u(10^{h-1})=u_{h-1}\oplus 1$。

7. 如权利要求 4 所述的方法，其特征在于，

·由以下述形式的两个字节构成的码字的集合 C_2 来定义完全前缀码（5）：$C_2=\{w_1w_2;\ w_i\in\{0,1\}^8,i=1,2\}$；

·由来自集合 $E_2=\{0,1\}^8\cup\{\varepsilon\}$的字定义输出字（7）的集合；

·参数化去同步函数（f）的预定参数 ν 是矢量 $\nu=(\nu_0,\ldots,\nu_8,\tilde{\nu}_9)$，其分量属于集合{0,1}，前面八个分量 $\nu_0,\ldots,\nu_8$ 的值是不变量，最后的分量 $\tilde{\nu}_9$ 的值是变量；以及

·参数化去同步函数 $f_\nu:C_2\rightarrow\{\varepsilon\}\cup\{0,1\}^8$ 与形式为 w_1w_2 的任意字关联：

·如果满足以下条件之一，则字 w_1：

• $4<wt(w_1\oplus w_2)\leqslant 8$ 并且 $\nu_{wt(w1\oplus w2)}=b$，wt 是 Hamming 重量，b 是{0,1}的预定元素；

• $0\leqslant wt(w_1\oplus w_2)<4$ 并且 $\nu_{wt(w1\oplus w2)}\neq b$；

• $wt(w_1\oplus w_2)=4$，$\tilde{\nu}_9=1$，$4<wt(w_1\oplus w_2\oplus e)\leqslant 8$ 并且 $\nu_4=b$，e 是来自 $\{0,1\}^8$ 的奇数 Hamming 重量字；

• $wt(w_1\oplus w_2)=4$，$\tilde{\nu}_9=1$，$0\leqslant wt(w_1\oplus w_2\oplus e)<4$ 并且 $\nu_4\neq b$；

·如果满足以下条件之一，则字 w_2：

• $4<wt(w_1\oplus w_2)\leqslant 8$ 并且 $\nu_{wt(w1\oplus w2)}\neq b$；

• $0\leqslant wt(w_1\oplus w_2)<4$ 并且 $\nu_{wt(w1\oplus w2)}=b$；

- $wt(w_1 \oplus w_2)=4$，$\tilde{v}_9=1$，$4<wt(w_1 \oplus w_2 \oplus e) \leqslant 8$ 并且 $v_4 \neq b$；
- $wt(w_1 \oplus w_2)=4$，$\tilde{v}_9=1$，$0 \leqslant wt(w_1 \oplus w_2 \oplus e)<4$ 并且 $v_4=b$；以及

·如果 $wt(w_1 \oplus w_2)=4$ 并且 $\tilde{v}_9=0$，则空字 ε。

8. 一种生成器，用于从初始数据流（3）生成伪随机数据序列（1），其特征在于包括：存储器（25）和处理单元（27），所述存储器（25）存储形成完全前缀码（5）的码字的集合和输出字（7）的集合，所述处理单元（27）能够读取所述初始数据流（3），并将其分解为根据所述完全前缀码（5）编码的字的系列（11），并根据去同步函数（f）将所述已编码的字（11）的系列中的字与对应输出字关联，以形成所述伪随机数据序列（1）。

9. 如权利要求 8 所述的生成器，其特征在于进一步包括：参数化装置（29），用于取决于预定参数来呈递所述去同步函数（9），并在生成伪随机数据序列（1）期间修改所述预定参数的值。

10. 一种加密/解密设备，包括：异或逻辑门（43），其特征在于，进一步包括根据权利要求 9 或权利要求 10 所述的生成器（21）。

11. 一种安全系统，包括经由网络（35）连接的至少两个实体（37a，37b），其特征在于，所述至少两个实体中的每一个包括：根据权利要求 10 所述的加密/解密设备（39a，39b）。

用于生成伪随机数据序列的方法、系统和设备

技术领域

本发明涉及加密/解密领域，并且本发明关注用于生成伪随机数据序列的系统和方法。

本发明找到一种在创建旨在对称加密的比特系列中高度有利的应用，所述对称加密是一种加密处理，其中，加密和解密使用相同秘密密钥。本发明的上下文首先描述将消息逐比特加到相同长度的伪随机数据序列的加密方法，其次描述方法，在所述方法中加密运算和解密运算是相同的。在诸如移动通信（GSM、UMTS 等）、互联网（SSL 等）、微芯片卡（银行卡）等的所有类型的通信中通常采用对称加密。

背景技术

最基本的对称加密技术是已知的流加密技术，其以明文将消息逐比特加到相同长度的随机系列。这种技术增加了产生长伪随机系列的必要和困难问题。

最常用的流加密方法使用独立于将通过采用线性反馈移位寄存器而加密的消息以节省硬件所生成的伪随机系列。

线性反馈移位寄存器的主要缺点在于它们是线性的。事实上，如果寄存器的输出比特的速率等于寄存器的长度并且已知与寄存器关联的反馈多项式，则可以确定寄存器的输出比特和所有后续状态。

因此，为了“破坏”线性反馈移位寄存器的线性，标准的做法是例如使用非线性布尔函数来组合来自多个寄存器的输出和其可能的内部状态。

图 5 示出这种生成器 21，其被称为互缩生成器（shrinking generator），在欧洲专利申请 EP 0 619 659 中描述了这种生成器，其包括第一线性反馈

移位寄存器 123a、第二线性反馈移位寄存器 123b 以及用于选择生成器 121 的输出的装置 125。

在每次移位时，两个寄存器 123a 和 123b 被同时移位，并且如果第一寄存器 123a 的输出是“1”，则设备 121 的输出等于第二寄存器 123b 的输出；否则不输出比特。

互缩生成器允许不仅组合两个线性反馈移位寄存器的输出，而且更通常地，组合任意一对比特系列。互缩生成器属于流加密系统的分类，其中，一个线性反馈移位寄存器控制另一线性反馈移位寄存器。这个构思在于通过既改变所采用的各个寄存器之间的移位数量又改变两个连续比特之间的移位数量来破坏寄存器的线性。

被称为自缩生成器的互缩生成器的变化是基于相同原理的，但仅使用一个寄存器。逐两个比特地读取寄存器的输出比特，并且第一比特控制第二比特的输出，从而如果第一比特是“1”，则系统的输出是第二比特；否则不输出比特。

仅使用线性反馈移位寄存器具有许多缺点。主要缺点是来自设备的线性的弱点。如果通过布尔函数来组合寄存器，则也是不利的。在硬件级别，这些不利之处来自于实现功能的复杂度。此外，这种函数是固定的，并且易受攻击。

此外，如果在伪随机系列生成器使用的移位寄存器的反馈是常规的并且容易预测的，则该生成器容易受到代数攻击。

更进一步地，统计方法已经示出互缩生成器的某些弱点。具体地说，在互缩生成器中，由两个输出比特之间的两个寄存器所影响的移位数量改变，并且对于两个寄存器均具有相同的值。

发明内容

本发明提供一种从初始数据流生成伪随机数据序列的方法，其包括以下步骤：

· 定义形成完全前缀码的码字的集合；

·定义输出字的集合;

·定义去同步函数，其将来自所述输出字的集合的输出字与所述完全前缀码中的任意码字关联;

·将初始数据流分解为根据所述完全前缀码所编码的字的系列;

· 根据去同步函数将所述已编码的字的系列中的字与对应输出字关联，以形成所述伪随机数据序列。

因此，完全前缀码使得初始数据流能够被独特地分解，并且可以由有限自动机（finite automaton）容易地起作用。此外，该方法实现简单，并且使用初始数据流去同步函数来生成伪随机数据序列。事实上，完全前缀码和“去同步分量”的使用防止了代数攻击，或者，在产生所确保的最小数量的比特的同时实质上无效地呈递它们。与之对比，即使将互缩生成器用作伪随机系列生成器中的去同步分量，最小的所确保的速率实际上也是零。

所述去同步函数有利地是取决于预定参数的参数化函数，可以在生成伪随机数据序列期间修改所述预定参数的值。

去同步取决于可修改的初始化参数的事实增加了初始数据流和伪随机数据序列之间的关系的复杂度，使得更难以预测伪随机数据序列。

根据本发明的特征，所述输出字的集合包括用于每一输出字的补码输出字。

例如，对于给定的输出字长度，这样平衡了所述输出字的集合中的“0”和“1”的数量。

所述完全前缀码的码字优选地具有受上限 h 限制的长度，并且所述码如下：如果任意输出字 x 具有与输出字 y 相同的长度，则按照输出字 x 的所述去同步函数的前项（antecedent）的数量与按照输出字 y 的所述去同步函数的前项的数量相同。

这样确保了在允许伪随机数据序列具有良好统计特性的同时不取决于初始数据流的特定特性的最小速率。特别地，这样确保了所生成的输出伪随机数据序列的 Hamming 重量不提供关于所述初始数据序列的信息。换

句话说，在所述输出序列中的具有值“0”的比特和具有值“1”的比特包含关于所述初始数据序列的相同数量的信息。

所述完全前缀码的字有利地具有受下限m限制的长度，并且有利的是，存在所述完全前缀码的字的上限长度l，从而对于小于或等于上限长度l的任意长度k，完全前缀码包含长度k的2^{m-1}个字。

该特征通过优化所述伪随机数据序列的概率分布而增强了上述统计特性。

所述方法的第一实施例的特征在于：

·由码字的集合C_1定义完全前缀码，所述码字具有以下形式，其中，$h \geqslant 2$：

$C_1=\{01^n0;0 \leqslant n \leqslant h-2\} \cup \{10^n1;0 \leqslant n \leqslant h-2\} \cup \{01^{h-1}\} \cup \{10^{h-1}\}$

·从二进制集合$E_1=\{0,1\}$由字定义输出字的集合；

·参数化去同步函数的预定参数u是矢量$u=(u_0,\ldots,u_{h-1})$，其分量属于集合{0,1}；以及

·按以下方式定义参数化去同步函数f_u：

·$f_u(01^n0)=u_n$，对于$0 \leqslant n \leqslant h-2$；

·$f_u(01^n1)=u_n \oplus 1$，对于$0 \leqslant n \leqslant h-2$；

·$f_u(01^{h-1})=u_{h-1}$；

·$f_u(10^{h-1})=u_{h-1} \oplus 1$。

该实施例相对廉价地实现，并且可以有利地用于提供硬件类型加密。此外，比特输出的数量（即伪随机数据序列的比特数量）和输入比特的数量（即初始数据流的比特数量）之间的比率严格大于1/3，并且长度h的输入上的比特输出的所确保的最小数量是1。

所述方法的另一实施例的特征在于：

·由以下述形式的两个字节构成的码字的集合C_2定义完全前缀码

$C_2=\{w_1w_2;\ w_i \in \{0,1\}^8,\ i=1,2\}$；

·由来自集合$E_2=\{0,1\}^8 \cup \{\varepsilon\}$的字定义输出字的集合；

·参数化去同步函数的预定参数ν是矢量$\nu=(\nu_0,\ldots,\nu_8,\ \tilde{\nu}_9)$，其分

量属于集合{0,1}，前面八个分量 $\nu_0,\ldots,\nu_8$ 的值是不变量，最后的分量 $\tilde{\nu}_9$ 的值是变量；以及

· 参数化去同步函数 $f_\nu:C_2\rightarrow\{\varepsilon\}\cup\{0,1\}^8$ 与形式 w_1w_2 的任意字关联：

· 如果满足以下条件之一，则字 w_1：

• $4<wt(w_1\oplus w_2)\leqslant 8$ 并且 $\nu_{wt(w1\oplus w2)}=b$，wt 是 Hamming 重量，b 是来自{0,1}的预定元素；

• $0\leqslant wt(w_1\oplus w_2)<4$ 并且 $\nu_{wt(w1\oplus w2)}\neq b$；

• $wt(w_1\oplus w_2)=4$，$\tilde{\nu}_9=1$，$4<wt(w_1\oplus w_2\oplus e)\leqslant 8$ 并且 $\nu_4=b$，e 是来自 $\{0,1\}^8$ 的奇数 Hamming 重量字；

• $wt(w_1\oplus w_2)=4$，$\tilde{\nu}_9=1$，$0\leqslant wt(w_1\oplus w_2\oplus e)<4$ 并且 $\nu_4\neq b$；

· 如果满足以下条件之一，则字 w2：

• $4<wt(w_1\oplus w_2)\leqslant 8$ 并且 $\nu_{wt(w1\oplus w2)}\neq b$；

• $0\leqslant wt(w_1\oplus w_2)<4$ 并且 $\nu_{wt(w1\oplus w2)}=b$；

• $wt(w_1\oplus w_2)=4$，$\tilde{\nu}_9=1$，$4<wt(w_1\oplus w_2\oplus e)\leqslant 8$ 并且 $\nu_4\neq b$；

• $wt(w_1\oplus w_2)=4$，$\tilde{\nu}_9=1$，$0\leqslant wt(w_1\oplus w_2\oplus e)<4$ 并且 $\nu_4=b$；以及

· 如果 $wt(w_1\oplus w_2)=4$ 并且 $\tilde{\nu}_9=0$，则空字 ε 。

该实现方式可以有利地用于软件类型加密。此外，字节输出数量和输入字节数量之间的比率严格大于 1/3，并且长度 h 字节的输入上的字节输出的所确保的最小数量是 1。

本发明还提供一种生成器，用于从初始数据流生成伪随机数据序列，其特征在于包括：存储器和处理单元，所述存储器存储形成完全前缀码的码字的集合和输出字的集合，并且所述处理单元能够读取初始数据流，并将其分解为根据所述完全前缀码所编码的字的系列，并根据去同步函数将所述已编码的字的系列中的字与对应输出字关联，以生成所述伪随机数据序列。

因此，所述生成器示出具有不取决于初始数据流的特定特性的最小速率的伪随机数据序列。此外，所述生成器易于实现、高效，并具有相对低的成本。

所述生成器有利地进一步包括：参数化装置（parameterizing means），用于取决于预定参数来呈递去同步函数，并在生成伪随机数据序列期间修改所述预定参数的值。

因此，不知道初始化参数的值或其被修改的时间就使得更加难以预测所述伪随机数据序列。

本发明还提供一种加密/解密设备，包括：异或逻辑门和具有上述特征的生成器。

所述设备提供了一种简单的方式来将所述伪随机数据序列的每一比特与将通过模 2 加以形成极大线性复杂度的已加密数据序列的消息的数据序列的对应比特进行组合。

本发明还提供一种安全系统，包括：经由网络所连接的至少两个实体，其中，所述至少两个实体中的每一个包括具有上述特征的加密/解密设备。

因此，所述安全系统包括易于实现同时具有作为固有复杂度的机制的结构。

附图说明

当阅读以下通过非限定示例所给出的描述并参照附图时，本发明的其它特点和优点会变得明显，其中：

图 1 是用于生成伪随机数据序列的本发明的方法的高度示意性示图；

图 2A 和图 2B 概略示出将初始数据流分解为根据本发明所编码的字的序列的有限自动机的示意性示例；

图 3A 和图 3B 示出本发明的伪随机数据序列生成器的示意性示例；

图 4 示出包括图 3A 和图 3B 的生成器的安全系统；以及

图 5 是现有技术生成器的示意性示图。

具体实施方式

图 1 示出根据本发明的方法从初始数据流 3 生成伪随机数据序列 1 的示意性示例。

术语“字”（或图案）以下指的是来自字母表－例如仅包括 0 和 1 的二进制集合的字母表－的任意有限的字母的系列。每个字于是具有给定的长度。例如，1、11、000、1010、00111 是分别具有长度 1、2、3、4、5 的字。此外，“空”字 ε 是零长度的字（即字不包含任意字母）。

初始数据流 3 与高“线性复杂度”的数据序列对应。例如，可以由初始产生装置 23（见图 3A）来生成初始数据流 3，初始产生装置 23 包括最大周期线性反馈移位寄存器。

事实上，可以由无限数量的线性反馈移位寄存器来生成任意周期系列。这些寄存器中的一个比所有其它寄存器短。这个最短寄存器的长度被称为流的“线性复杂度”。如果初始数据序列的线性复杂度是 L，则可以使用 Berlekamp-Massey 算法来从长度 2L 的初始数据序列的子系列来重构寄存器的初始状态（并且事实上是所有序列）。

因此，为了生成安全伪随机数据序列，推荐初始数据流 3 具有较高线性复杂度。事实上，当前，如果初始数据流的线性复杂度 L 小于 160，则最佳已知算法可以通过少于 2^{80} 次的运算来重建初始数据流 3。因此，有利的是，得到具有线性复杂度 L 大于或等于 160 的初始数据流 3。

根据本发明，定义码字的集合 5a，其形成“完全前缀码”5。

通用术语“码”指的是特定字母表上的字的任意集合。于是将所述码的特性看作与正在讨论的字母表有关。

假设 A 是任意固定字母表。考虑以来自 A 的字母所构成的字，并假设 C 指定 A 上的码。于是:

1）如果存在 y，w=xy，则字 x 被称为字 w 的“前缀”，该记法表示 x 与 y 连结（concatenation）。如果在码 C 中没有作为码的另一字的前缀的码的字，则 C 被称为“前缀码”；

2）如果字 w 与 C 的字连结，换句话说，如果其可以用 C 的字 w_1、w_2、……、w_n 写为 $w=w_1w_2...w_n$，则字 w 被称为由 C 来编码；于是，对于任意字 w，如果存在字 w’从而 ww’由 C 来编码，则 C 被称为“完全的”。可以看出，对于任一字 w，当且仅当存在字 w’以及属于 C 的码字 u 使得 u

是字 ww'的前缀时，也就是如果存在字 w'、属于 C 的码字 u、以及字 u'，使得 ww'等于 uu'，则码 C 是完全的。

如果 C 既是前缀码又是完全码，则 C 被称为完全前缀码。

更进一步地，定义由 E 所指定的输出字 7a 的集合 7，从而 E 被包含在 $\{0,1\}^k$ 和{ ε }的并集中，并且通过将来自输出字的集合 7 的输出字 7a 与完全前缀码 5 的任意码字 5a 关联来定义去同步函数 f。

例如，输出字的集合 7 是集合 E={ ε ,0,1}或 E={0,1}或 E={ ε }和 $\{,0,1\}^{8k}$ 的并集（其中，k⩾1，即多个字节）。具体地说，对于输出字的给定长度，为了平衡输出字的集合中的“0”和“1”的数量，有利的是，选择输出字的集合 7，其包括对于表示为 s 的任一输出字 7a 的表示为 $\bar{s}$ 的补码输出字 7a。

此外，去同步函数 f 是基本运算（例如逐比特模 2 加），可以以相对低的成本来实现其估算。

因此，本发明的方法将完全前缀码 C5、输出字 7 的集合 E 以及去同步函数 f 作为输入。

更进一步地，根据本发明的方法包括：分解运算 10，用于将初始数据流 3 分解为根据所述完全前缀码 5 所编码的字的序列 11。事实上，完全前缀码 5 的使用提供了初始数据流 3 的独特分解。

可以由有限自动机来有利地识别完全前缀码 5，被给定字 w，所述有限自动机确定字 w 是否处于完全前缀码 5 中。

事实上，图 2A 和图 2B 示意性示出有限自动操作 13a 和 13b 的两个示例，其用于将初始数据流 3 分解为已编码的字的序列 11。有限自动机包括由路径或箭头 17 所连接的节点或状态 15 的集合，从而可以由以 I 表示的初始状态 15 和以 F 表示的最终状态 15 之间的路径 17 的集合来定义完全前缀码 5 的每一字。因此，当读取比特时，取得与所述比特的值对应的路径 17，并且当到达最终状态 F 时，已知的是，对完全前缀码 5 的字的读取恰好已经完成，并且返回到初始状态 I 使得能够读取下一字。

图 2A 示例识别由 C={10*1,01*0}所定义的完全前缀码 5 的码字，其中，

10*1（分别地 01*0）指定形式 10^k1（分别地 01^k0）的字的集合，其中，k 是整数。应注意，$10^0$1 与字 11 对应，而 $01^0$0 与字 00 对应。因此，这种有限自动机 13a 将初始数据流 3 分解为已编码的字 11 的系列，其包括形式为 10...01、01...10、11 和 00 的字。

此外，图 2B 示例识别具有由码 $C=\{10^{h+1},01^{h+1},10^k1,01^k0,k \leqslant h\}$所定义的完全前缀码 5 的上限 h 所限制的长度的码字。

此外，根据本发明的方法包括关联运算 20，其用于根据去同步函数 f 将已编码的字的序列 11 的字与对应输出字 7a 关联，以形成伪随机数据序列 1。

因此，本发明的机制的应用关注将输入系列（即初始数据流 3）分解为完全前缀码 5 的字的系列 11。每当识别完全前缀码 5 的字，该字的图像通过去同步函数 f 产生输出字。重复这种机制，直到达到初始数据流 3 的输入系列的最后比特，或满足由用户所确定的停止条件。

注意，在密码学应用的环境中，所述机制（码 5，函数 f）的一些数据或全部数据可以有利地保留密码学系统的秘密数据。

去同步函数 f 有利地是取决于在生成伪随机数据序列 1 期间可以修改的值的预定参数的参数化函数。这种参数化函数于是具有形式 $v,x \mapsto f_v(x)$，其将来自输出字 7 的集合 E 的元素与长度 m 的任意二进制矢量 v（即 $\{0,1\}^m$ 的任意元素 v）以及完全前缀码 C5 的任意字 x 关联。

本发明的方法于是取得初始数据流 3 和二进制矢量 v 的输入。可以由于将（非线性）运算应用于初始化矢量而导致所述矢量 v 的选择。当然，矢量 v 可以取得预定值（例如 v 是空矢量）。矢量 v 可以有利地在处理期间被修改，并且因此取决于伪随机数据序列 1 的生成。

因此，这个参数增强了初始数据流 3 和伪随机数据序列 1 之间的关系的复杂度，使得更难以预测伪随机数据序列 1。

此外，为了确保不取决于初始数据流 3 的特定特性的最小速率，有利的是，对于完全前缀码 5 的码字，具有由上限 h 所限制的长度，并且从而如果任意输出字 s_1 具有与另一输出字 s_2 相同的长度，则按照输出字 s_1 的去

同步函数 f 的前项的数量与按照输出字 s_2 的去同步函数 f 的前项的数量相同。

换句话说，完全前缀码 5 具有以下有利特性:

· 存在整数 h，使得完全前缀码 C5 的所有字具有小于或等于 h 的长度；以及

· 对于任意整数 k，对于来自集合 $E\backslash\{\varepsilon\}$ 的任意对（s_1,s_2），集合 $f_v^{-1}\{s_1\}$ 和 $f_v^{-1}\{s_2\}$ 中的长度 k 的元素的数量分别相等。

此外，这样确保了所生成的输出伪随机数据序列 1 的 Hamming 重量（即值“1”的比特的数量）不提供关于初始数据序列 3 的任意信息。换句话说，输出伪随机数据序列 1 中具有值“0”的比特和具有值“1”的比特包含关于初始数据序列 3 的相同数量的信息。

完全前缀码 5 的字有利地具有受下限 m 所限制的长度，并且有利的是存在所述完全前缀码的字的较大长度 l，从而对于小于或等于所述较大长度 l 的任意长度 k，完全前缀码包含长度 k 的 2^{m-1} 个码。换句话说，关联函数 f_v 满足以下特性：存在整数 m，从而所考虑的码 C 的所有字具有大于或等于 m 的长度，并且，此外，存在整数 $l \geqslant m$，从而对于大于或等于 m 并且小于或等于 l 的任意整数 k，码 $C\backslash\{f_v^{-1}(\varepsilon)\}$ 确切地包含长度 k 的 2^{m-1} 个字。

因此，在初始数据流 3 的均匀分布的情况下，平均速率具有级（order）l/(m+1)。具体地说，对于伪随机数据序列的给定比特，并且无论给定比特是什么（0 或 1），生成所述比特的码字具有长度 k 的概率都与 k 成反比。概率分布的这种选择关于伪随机数据序列 1 必须满足的统计特性是最佳的

在本发明的第一特定实施例中，考虑由集合 $C_1=\{01^n0;0 \leqslant n \leqslant h-2\} \cup \{10^n1;0 \leqslant n \leqslant h-2\} \cup \{01^{h-1}\} \cup \{10^{h-1}\}$（其中，∪标志并集运算符），h 是固定整数（$h \geqslant 2$），并且由二进制集合 $E_1=\{0,1\}$ 的字来定义输出字的集合，例如。

此外，定义矢量 $u=(u_0,\ldots,u_{h-1})$，其分量属于集合={0,1}。按以下方式来定义参数化函数 f_u:

· $f_u(01^n0)=u_n$，对于 $0 \leqslant n \leqslant h-2$;

· $f_u(10^n1)=u_n\oplus 1$，对于 $0\leqslant n\leqslant h-2$（$\oplus$表示模2加）；

· $f_u(01^{h-1})=u_{h-1}$;

· $f_u(10^{h-1})=u_{h-1}\oplus 1$。

在该实施例中，在知道字 x 的第一比特以及值 n（对于形式$\bar{b}b^n\bar{b}$的字）或固定值 h-1（对于形式$\bar{b}b^{h-1}$的字）的情况下，对于码 C_1 的任意字 x，逐比特读取初始数据流3，并计算值 $f_u(x)$。

为了描述所述第一实施例的机制，使 $f_u'(b,n)=f_u(b\bar{b}^nb)=b\oplus u_n$，其中，$0\leqslant n\leqslant h-2$，并使 $f_u'(b,h-1)=f_u(\bar{b}b^{h-1})=b\oplus u_{h-1}$。此外，将标志初始化为值 0，并仅在识别码 C_1 的字之后取值 1。在每一字识别之后重置标记的值。对于每一字识别，当前字的第一比特被临时存储在变量 c 中，并且以变量“length”来更新字。输出比特的值被存储在变量 s 中。

因此，所述第一实施例包括以下机制的重复:

· flag←0; length←0

· 读取初始数据流3的系列 S 中的下一比特 b

· c←b

· 执行:

· 读取系列 S 中的下一比特 b

• 如果 b=c，则 flag←1，否则 length=length+1

直到 (flag=0) 并且 length<h-1

· 输出 $c\oplus u_{length}$. //$c\oplus u_{length}=f_u'(c,length)$

该实施例相对廉价地实现，并且可以有利地用于以硬件中的电子电路来实现加密。此外，平均速率，即比特输出的数量（伪随机数据序列 1 的比特数量）和输入比特的数量（初始数据流 3 的比特数量）之间的比率，严格大于 1/3。此外，长度 h 的输入上的比特输出的所确保的最小数量是 1。

在本发明第二特定实施例中，由以下面形式中的两个字节所构成的码字的集合 C_2 来定义完全前缀码 5，$C_2=\{w_1w_2\};w_i\in\{0,1\}^8$，i=1,2}并且由来自集合 $E_2=\{0,1\}^8\cup\{\varepsilon\}$的字来定义输出字的集合。

此外，定义了矢量 $v=(v_0,\ldots,v_8,\tilde{v}_9)$，其分量属于集合{0,1}，前面八

个分量 $\nu_0,\ldots,\nu_8$ 的值是不变量，最后的分量 $\tilde{\nu}_9$ 的值是变量。

此外，参数化函数 $f_v:C_2\rightarrow\{\varepsilon\}\cup\{0,1\}^8$ 与形式 w_1w_2 的任意字关联：

·如果满足以下条件之一，则字 w_1：

- $4<wt(w_1\oplus w_2)\leqslant 8$ 并且 $\nu_{wt(w1\oplus w2)}=b$，wt 是 Hamming 重量，b 是{0,1}的预定元素；
- $0\leqslant wt(w_1\oplus w_2)<4$ 并且 $\nu_{wt(w1\oplus w2)}\neq b$；
- $wt(w_1\oplus w_2)=4$，$\tilde{\nu}_9=1$，$4<wt(w_1\oplus w_2\oplus e)\leqslant 8$ 并且 $\nu_4=b$，e 是来自 $\{0,1\}^8$ 的奇数 Hamming 重量字；
- $wt(w_1\oplus w_2)=4$，$\tilde{\nu}_9=1$，$0\leqslant wt(w_1\oplus w_2\oplus e)<4$ 并且 $\nu_4\neq b$；

·如果满足以下条件之一，则字 w_2：

- $4<wt(w_1\oplus w_2)\leqslant 8$ 并且 $\nu_{wt(w1\oplus w2)}\neq b$；
- $0\leqslant wt(w_1\oplus w_2)<4$ 并且 $\nu_{wt(w1\oplus w2)}=b$；
- $wt(w_1\oplus w_2)=4$，$\tilde{\nu}_9=1$，$4<wt(w_1\oplus w_2\oplus e)\leqslant 8$ 并且 $\nu_4\neq b$；
- $wt(w_1\oplus w_2)=4$，$\tilde{\nu}_9=1$，$0\leqslant wt(w_1\oplus w_2\oplus e)<4$ 并且 $\nu_4=b$；以及

·如果 $wt(w_1\oplus w_2)=4$ 并且 $\tilde{\nu}_9=0$，则空字 ε。

为了描述所述第二实施例的机制，设置整数 h，从而 $\geqslant 1$，并且标记被用值 0 初始化，并仅在码 C_2 的字的识别之后取值 1。对于每一字识别，所读取的最后比特被临时存储在变量 c 中，并且以变量“length”来更新字。输出比特的值被存储在变量 s 中。两个字节 w_1 和 w_2 的连结被表示为 $w_1|w_2$。

因此，所述第二实施例包括以下机制的重复：

·flag←0；length←0

·$\tilde{\nu}_9\leftarrow 0$

·读取系列 S 中的下一字节 oct

·c←oct

·执行：

- 读取系列 S 中的下一比特 oct
- $s\leftarrow f_v(c/oct)$
- 如果 $s\neq\varepsilon$ 则 flag←1，否则：

- c←oct
  - length=length+1
  - 如果 length=h-3, 则$\tilde{v}_9$←1

直到 (flag=0)

·输出 s。

该第二实施例可以有利地用于软件类型的加密。此外，平均速率（即字节输出的数量和输字节的数量之间的比率）严格大于 1/3，并且长度 h 字节的输入上的字节输出的所确保的最小数量是 1。

图 3A 非常概略地示出用于生成伪随机数据序列 1 的生成器 21 的示例。

生成器 21 包括初始产生装置 23，其包括至少一个最大周期线性反馈移位寄存器，用于生成初始数据流 3。已知的是，在最大周期 T 的系列中，长度 k 的所有字或图案（其中，$T=2^k-1$）至少出现一次。线性反馈移位寄存器是配备了由称为反馈多项式所表示的线性组合的有限长度的比特表。在每一移位上，具有最高索引（index）的比特被移出，所有其它比特被移位一个索引，并且具有最低索引的比特在移位之前取得所述线性组合的值。

例如，反馈多项式可以有利地是与产生最大周期系列的线性反馈移位寄存器对应的本原多项式，或形式为 $Q=(x^2+1)P$ 的多项式，其中，P 是本原多项式。

此外，生成器 21 包括存储器 25 和处理单元 27。存储器 25 存储形成完全前缀码 5 的码字的集合以及输出字 7 的集合。

处理单元 27 读取初始数据流 3，并将其分解为根据完全前缀码 5 而编码的字 11 的序列，并根据去同步函数 f 将字 11 的所述已编码系列的字与对应输出字关联，以生成伪随机数据序列。去同步函数 f 将来自输出字 7 的集合的输出字与完全前缀码 5 的任意码字关联。

注意，处理单元 27 可以同时进行分解运算 10 和关联运算 20。因此，处理单元 27 逐比特读取初始数据流 3，并且每当找到完全前缀码 5 的字，就由去同步函数 9 来计算所述字的图像。

所述生成器因此简单地实现，并且包括：分解装置（存储器 25 和处理

单元 27），其按独特方式分解输入初始数据流 3；以及去同步装置（存储器 25 和处理单元 27），其生成具有最小速率的伪随机数据序列，所述最小速率不取决于初始数据流 3 的特定特性。

图 3B 示出用于生成伪随机数据序列 1 的生成器 21 的另一示例；其与来自图 3A 的生成器不同之处在于：进一步包括参数设置装置 29。参数设置装置 29 取决于预定参数制成去同步函数 f，并在伪随机数据序列 1 的生成期间修改所述参数的值。

图 4 示出安全系统 30，其包括经由互联网、GSM、UMTS、WiFi、超宽带等类型通信网络 35 而互连的至少两个实体。

该图示出第一实体 33a，其经由通信网络 35 连接到第二实体 33b。

第一实体 33a（分别地，第二实体 33b）包括第一终端 37a（分别地，第二终端 37b）、第一加密/解密设备 39a（分别地，第二加密/解密设备 39b）、以及第一调制解调器 41a（分别地，第二调制解调器 41b），调制解调器 41a 和 41b 是能够与通信网络 35 进行接口的任意设备。

第一加密/解密设备 39a 和第二加密/解密设备 39b 中的每一个包括：生成器 21，其用于生成上述伪随机数据序列 1；以及异或逻辑门 43。

每一加密/解密设备 39a、39b 通过逐比特对消息进行加密或解密来对流加密或流解密起作用。

根据该示例，第一加密/解密设备 39a 实现加密运算。因此，由异或逻辑门 43 来组合被称为加密系列的伪随机数据序列 1，其中，在消息 45 的对应位置处的每一比特由第一终端 37a 以明文来发送；这样产生已加密文本 47，第一调制解调器 41a 于是将已加密文本 47 发送给第二实体 33b。因此，加密运算通过逐比特将加密系列 1 加到消息 45 的明文的文本来获得已加密文本 47。

第二加密/解密设备 39b 通过将相同的加密系列 1 逐比特加到由第一实体 33a 所发送的已加密文本 47 来实现解密运算。

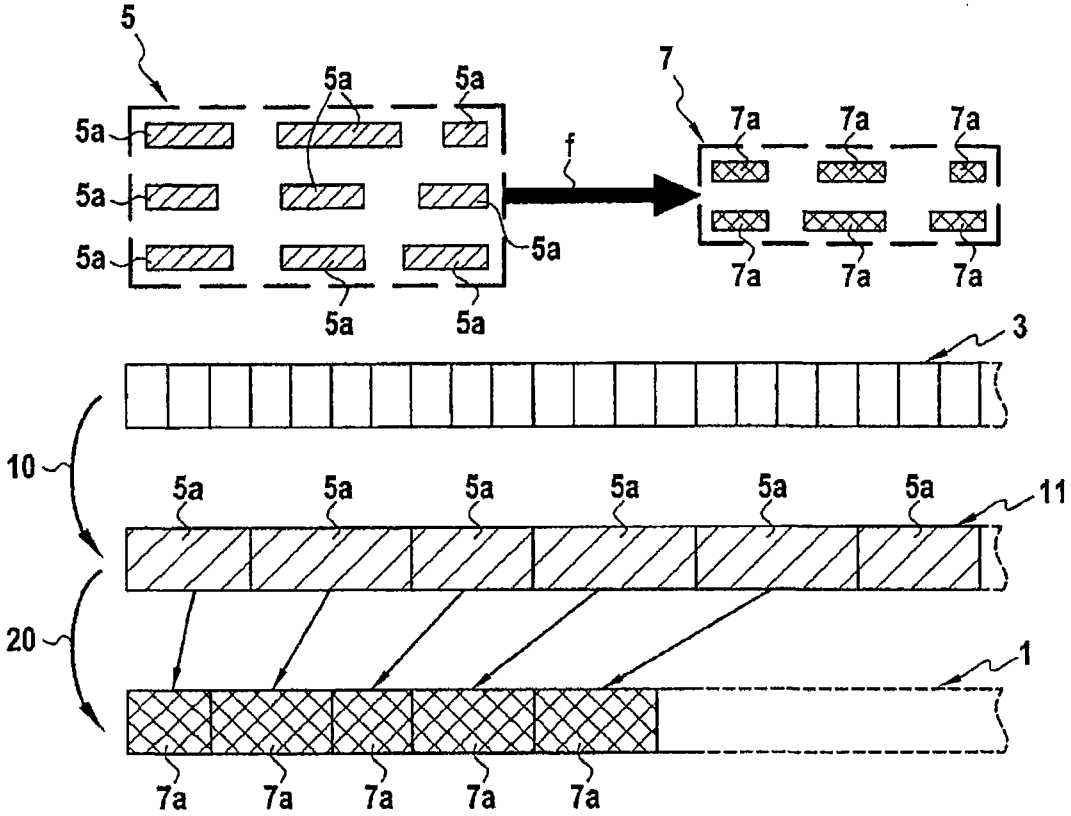


图 1

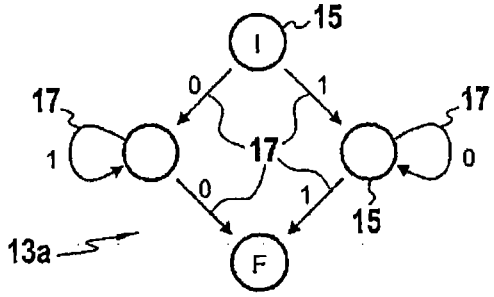


图 2A

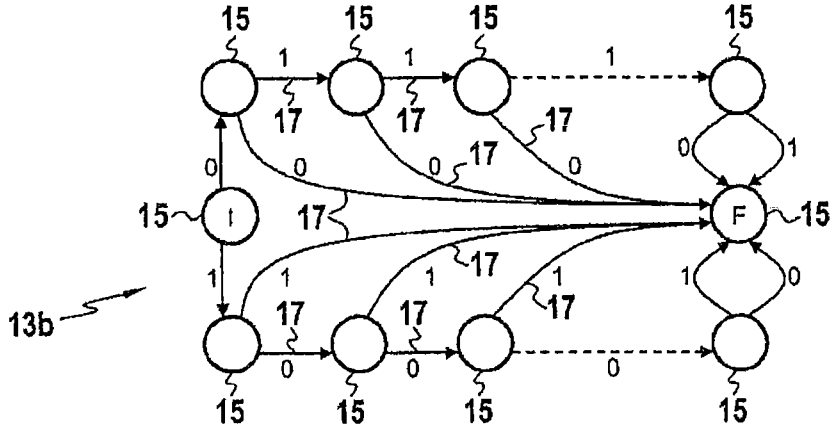


图 2B

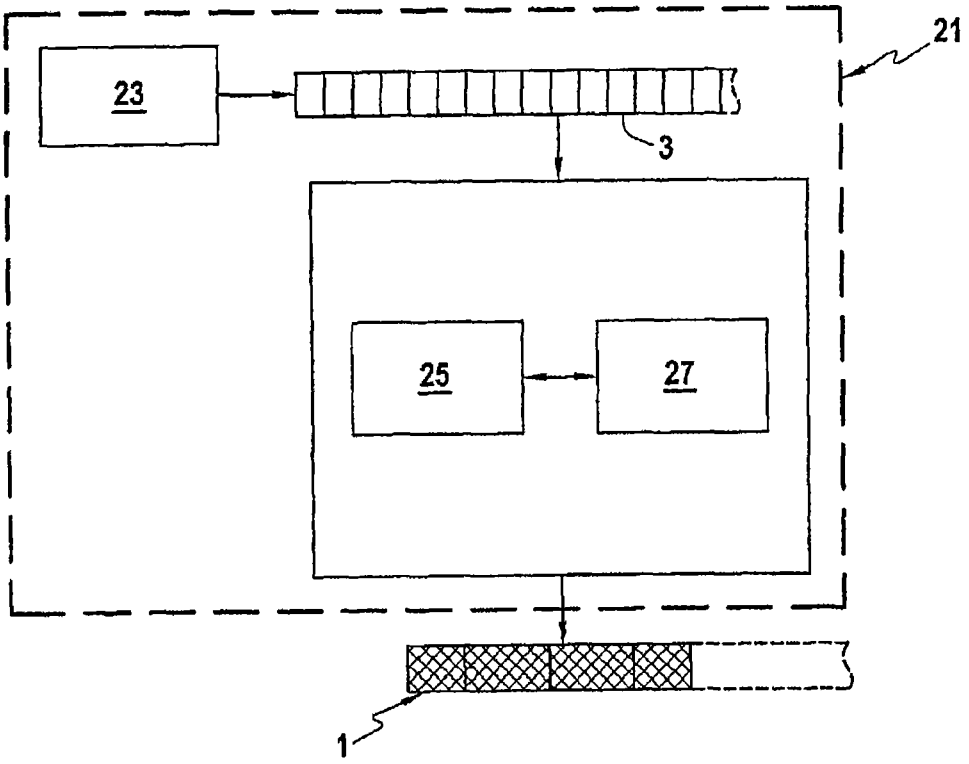


图 3A

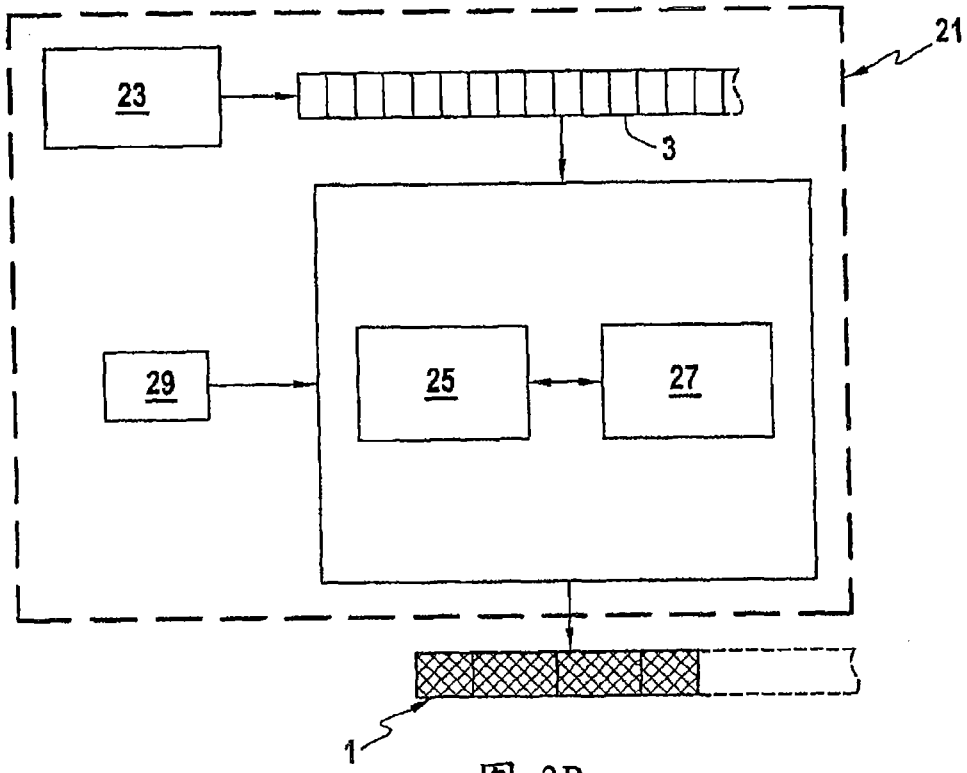


图 3B

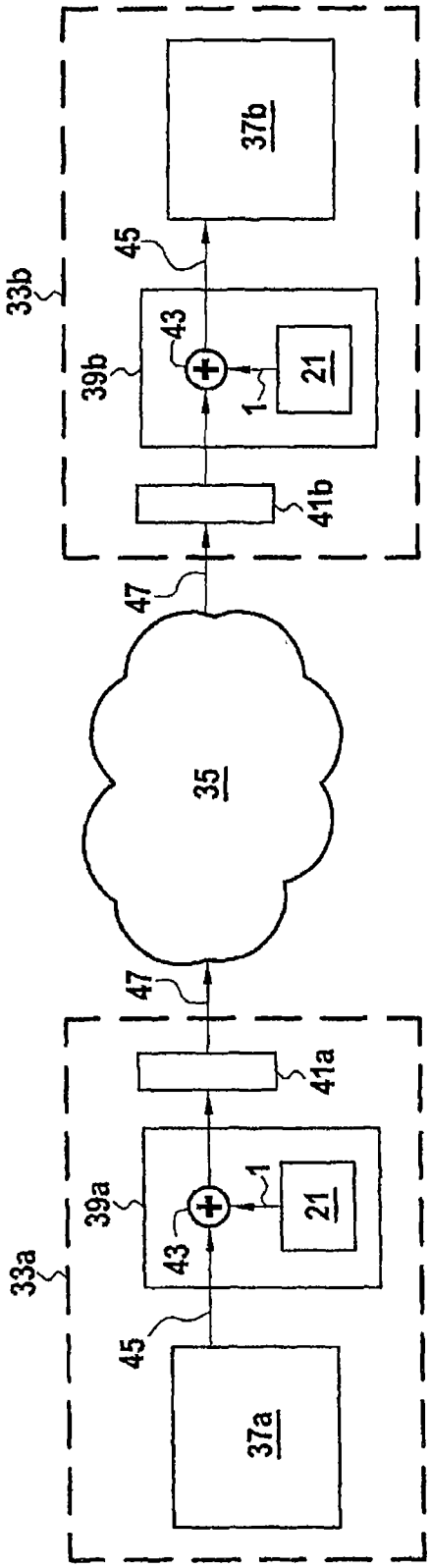

33a
33b
37a
37b
45
45
39a
39b
43
43
1
1
21
21
41a
41b
47
47
35
30

图 4

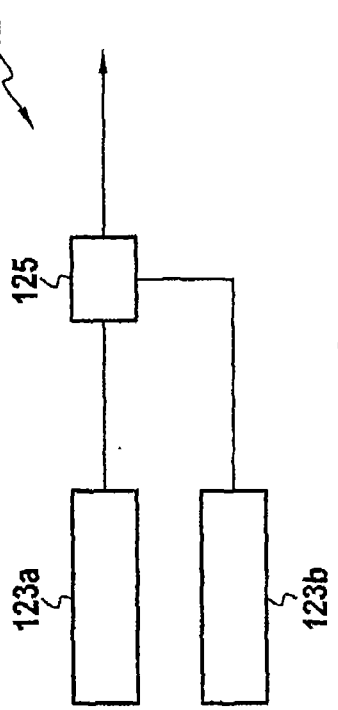

121
125
123a
123b

图 5
现有技术