(19) **RU** (11) **2 226 041** (13) **C2**

(51) МПК[7] **H 04 L 9/08**

РОССИЙСКОЕ АГЕНТСТВО ПО ПАТЕНТАМ И ТОВАРНЫМ ЗНАКАМ

(12) ОПИСАНИЕ ИЗОБРЕТЕНИЯ К ПАТЕНТУ РОССИЙСКОЙ ФЕДЕРАЦИИ

(21), (22) Заявка: 2001129345/09 , 01.11.2001

(24) Дата начала действия патента: 01.11.2001

(46) Дата публикации: 20.03.2004

(56) Ссылки: RU 2130641 C1, 20.05.1999. RU 2099890 C1, 20.12.1997. RU 2103829 C1, 27.01.1998. US 5778011 A, 07.07.1998. US 6046744 A, 04.04.2000.

(98) Адрес для переписки:
344007, г.Ростов-на-Дону, пер. Газетный, 51, КБ "СПЕЦВУЗАВТОМАТИКА", Ю.Н. Власову

(72) Изобретатель: **Аграновский А.В., Хади Р.А., Балакин А.В., Фомченко В.Н., Мартынов А.П.**

(73) Патентообладатель:
Государственное предприятие конструкторское бюро "СПЕЦВУЗАВТОМАТИКА"

(54) СПОСОБ КРИПТОГРАФИЧЕСКОГО ПРЕОБРАЗОВАНИЯ ДВОИЧНЫХ ДАННЫХ

(57)

Изобретение относится к области электросвязи и вычислительной техники и может найти использование в системах связи, вычислительных и информационных системах для криптографического закрытия двоичной информации при обмене данными. Технический результат заключается в повышении сложности криптоанализа и несанкционированного дешифрования данных. При этом входные данные разбивают на блоки фиксированной длины и кодируют, подвергая их арифметическому кодированию с криптографическими псевдослучайными функциями F_1, F_2 и F_3, зависящими от указанной ключевой последовательности, на каждом шаге для арифметического кодирования вводят данные посимвольно и кодируют их с использованием таблицы символов, реализующих преобразование F_2, и таблицы интервалов вероятности появления символов, реализующих преобразование F_3, обновляют таблицу символов, переставляя местами символы в таблице символов, затем выполняют обмен интервалами вероятности появления в указанной таблице, после чего преобразованные входные данные передают в блок выходных данных. 2 з.п. ф-лы,1 ил.

RU 2226041 C2

RU 2226041 C2

(19) **RU** (11) **2 226 041** (13) **C2**

(51) Int. Cl.[7] **H 04 L 9/08**

RUSSIAN AGENCY
FOR PATENTS AND TRADEMARKS

(12) ABSTRACT OF INVENTION

(21), (22) Application: 2001129345/09 , 01 .11 .2001

(24) Effective date for property rights: 01 .11 .2001

(46) Date of publication: 20 .03 .2004

(98) Mail address:
344007, g.Rostov-na-Donu, per. Gazetnyj, 51, KB "SPETsVUZAVTOMATIKA", Ju.N. Vlasovu

(72) Inventor: Agranovskij A.V., Khadi R.A., Balakin A.V., Fomchenko V.N. , Martynov A.P.

(73) Proprietor:
Gosudarstvennoe predprijatie konstruktorskoe bjuro "SPETsVUZAVTOMATIKA"

(54) METHOD FOR BINARY DATA CRYPTOGRAPHIC CONVERSION

(57) Abstract:

FIELD: electrical communications and converter engineering. SUBSTANCE: proposed method can be used in communication systems, computing and data-processing systems during data exchange and includes following procedures: input data are divided into fixed-length blocks and coded by means of arithmetic operation with cryptographic pseudorandom functions F1, F2, and F3 dependent of mentioned key sequence; data are entered during each step for arithmetic coding in character-by-character manner and coded with aid of F2 conversion implementing character table and occurrence probability interval table of F3 conversion implementing characters; character table is updated by interchanging characters in character table; then occurrence probability intervals are exchanged in mentioned table, whereupon converted input data are transferred to input data block. EFFECT: enhanced reliability of protection against cryptographic analysis and unauthorized data decoding. 3 cl, 1 dwg

RU 2226041 C2

RU 2226041 C2

Изобретение относится к области электросвязи и вычислительной техники, а конкретнее к области способов и устройств для криптографического преобразования данных, и может быть использовано в связных, вычислительных и информационных системах для криптографического закрытия двоичной информации при обмене данными правительственными, правоохранительными, оборонными, банковскими и промышленными учреждениями, когда возникает необходимость хранения и передачи конфиденциальной информации.

Известен способ шифрования двоичной информации и устройство для осуществления способа - "Албер" [1], в котором для осуществления криптографического преобразования текста используется набор из информационного и ключевого p-разрядных регистров. Входные данные разбиваются на блоки переменной длины из m ячеек, включающей по p разрядов в каждой. Далее, содержимое этих ячеек преобразуется с помощью n-кратного сложения по модулю 2 или по модулю 2^p с содержимым второй p-разрядной ячейки информационного регистра, к полученной сумме прибавляют по модулю 2 или по модулю 2^p содержимое очередной p-разрядной ячейки ключевого регистра, результат преобразуют блоком p-разрядного функционального преобразования f, к полученному результату прибавляют по модулю 2 или по модулю 2^p содержимое r-й ($3 \le r \le m-1$) ячейки информационного регистра.

Однако, данный способ обладает недостатками, которые связаны с возможностью анализа зашифрованных данных путем применения метода дифференциального криптоанализа. Это происходит благодаря тому, что сложение по модулю 2 или по модулю 2^p p-разрядной ячейки с нулем не приводит к изменению содержимого данной ячейки, а в случае изменения может быть выявлена зависимость изменения содержимого данной ячейки от текущего на момент зашифрования содержания информационного и ключевого регистров.

Известен также способ шифрования блоков данных [2], который ориентирован на шифрование входных блоков данных с помощью присоединения к входным блокам псевдослучайных двоичных блоков той же длины и использования неповторяющихся комбинаций генерируемых подключей. Способ включает формирование ключа шифрования, формирование $K \ge 1$ блоков данных, содержащих $P \ge 1$ участков двоичной информации и преобразование блоков данных под управлением ключа шифрования. Дополнительно генерируют $D \ge 1$ двоичных векторов, а блоки данных формируют путем присоединения двоичных векторов к участкам двоичного кода информации. Двоичные вектора генерируют по случайному или псевдослучайному закону. Двоичные вектора присоединяют к участкам двоичного кода информации в зависимости от ключа шифрования или по фиксированному правилу.

Однако способ имеет недостаток, заключающийся в том, что входные данные обрабатываются блоками большого размера, а не посимвольно; размер двоичного вектора выходных данных превосходит в два раза размер входного вектора, приводит к снижению быстродействия выполнения как прямой задачи по шифрованию информации, так и обратной по ее дешифрованию, также снижает надежность работы и увеличивает энергетические затраты на дальнейшую обработку, передачу и хранение зашифрованной информации.

Наиболее близким по технической сущности к предлагаемому является способ защиты информации от несанкционированного доступа [3], принятый за прототип, заключающийся в том, что на передающей стороне исходная информация искажается с помощью ключевой последовательности, а на приемной стороне искаженная информация восстанавливается с помощью того же ключа, при этом исходные данные обрабатываются посимвольно. Существо способа прототипа заключается в том, что исходный файл информации разбивается на информационные блоки переменной длины i (i=1, 2, 3,...), после чего в каждом блоке осуществляется варьируемый сдвиг ASCII-кода каждого символа блока на t (t=1, 2, 3,...) разрядов. Сдвиг может осуществляться как влево, так и вправо, при этом для предотвращения потерь информации сдвиг осуществляется по замкнутому кольцу в блоке, результат преобразования фиксируют в памяти по адресу исходного файла.

Ключ, определяемый пользователем, является w (w=1, 2, 3,...) разрядным целым беззнаковым числом. Преобразованный файл можно оперативно восстановить путем обратного сдвига символов блоков файла.

Недостатками прототипа являются низкая стойкость к несанкционированному расшифрованию зашифрованной с его помощью информации; необходимость хранить ключевую последовательность большого объема, определяющую длину каждого шифруемого блока; отсутствие процедуры избавления от информационной избыточности во входных данных, а в следствие этого - повышенные энергетические затраты на дальнейшую обработку, передачу и хранение избыточной информации.

Техническим результатом, получаемым от внедрения изобретения, является устранение указанных недостатков, то есть повышение сложности криптоанализа и несанкционированного дешифрования зашифрованной с его помощью информации, а также снижение энергетических затрат на обработку, передачу и хранение информации.

Данный технический результат достигается за счет того, что в известном способе криптографического преобразования двоичных данных, заключающемся в том, что на передающей стороне исходная информация искажается с помощью ключевой последовательности, а на приемной стороне искаженная информация восстанавливается с помощью той же ключевой последовательности, при этом исходные данные обрабатываются посимвольно, входные данные подвергаются преобразованию методом арифметического кодирования с введенными криптографическими преобразованиями, зависящими от ключевой последовательности, и при кодировании

RU 2226041 C2

RU 2226041 C2

преобразуются с помощью кодирующей функции E (в качестве функции E выбирается функция арифметического кодирования согласно [5]) совместно с криптографическими преобразованиями F_1, F_2 и F_3 в вещественное число из интервала [0,1), которое записывается двоичной последовательностью конечной переменной длины и вместе с ключевой последовательностью однозначно определяет входные данные.

При этом в качестве функции преобразования F_2 используют функцию линейного конгруэнтного преобразования вида F(x)=ax+b mod c, с начальным значением x, зависящим от ключевой последовательности (т.е. x представляет собой последовательность значений ключевой последовательности), где константы a и b выбираются так, чтобы вычисленная последовательность не обладала периодом максимальной длины, а константа c выбирается равной M, где M - это количество всевозможных значений формируемых блоков из входной последовательности данных.

А в качестве составляющей функций преобразования F_1 и F_3 используют нормированную функцию линейного конгруэнтного преобразования вида

$$F(x) = \frac{ax + b \bmod c}{N},$$

с начальным значением x (т.е. x представляет собой последовательность значений ключевой последовательности), зависящим от ключевой последовательности и разным для обеих F_1 и F_3, где константы a и b выбираются так, чтобы вычисленная последовательность не обладала периодом максимальной длины, а константа c для F_1 и F_3 выбирается равной целой части 0.25N, где N - фиксированное простое целое число большого порядка, выбранное произвольно до начала кодирования.

Другими словами, данный технический результат достигается по существу за счет введения посимвольных криптографических преобразований, зависящих от данной ключевой последовательности, с использованием метода арифметического кодирования двоичной информации, который состоит в том, что при кодировании входные данные преобразуются с помощью кодирующей функции E в вещественное число из интервала [0,1), который называют отображающим интервалом. Результат функции E (см. [4, 5]) является вещественным числом из отображающего интервала, которое записывается двоичной последовательностью конечной длины и однозначно определяет входные данные. Таким образом, входные данные и выходные данные представляют собой двоичную последовательность переменной длины. Перед началом процесса кодирования входные данные разбиваются на блоки фиксированной длины, и кодируются поочередно, как и в прототипе. Значением блока является двоичное представление битов блока. Совокупность значений всех блоков образует алфавит, а сами значения блоков являются символами алфавита. Количество возможных значений формируемых блоков из входной последовательности данных обозначим через M. По мере кодирования входных данных, отображающий интервал уменьшается, а количество бит для его представления возрастает.

Каждому символу алфавита ставят в соответствие априорную вероятность его появления в потоке входных данных. Дополнительно, каждому символу присваивают интервал вероятности на промежутке [0, 1), такой что его длина равна вероятности появления данного символа. Нижнюю границу вероятности обозначают через L, а верхнюю границу вероятности через H. В таком случае, совокупность пар [$L_{(символ)}$, $H_{(символ)}$) для всех символов входных данных однозначно определяет вероятностную модель появления каждого символа в кодируемом тексте.

Очередные кодируемые символы входных данных сокращают величину отображающего интервала исходя из значений их априорных вероятностей, определяемых имеющейся моделью. Более вероятные символы делают это в меньшей степени, чем менее вероятные, добавляя меньше бит к выходным данным. В процессе кодирования, вероятностная модель изменяется так, чтобы учитывать фактические появления символов во входных данных. Данный эффект достигается за счет увеличения вероятности появления текущего символа и соответственного уменьшения всех остальных. Таким образом, достигается оптимальное сжатие входных данных [4,5]. Данный метод сжатия называют адаптивным сжатием. Для того, чтобы сигнализировать окончание кодирования используют специальный символ, который обозначают EOF.

Процесс кодирования текста можно описать с помощью формального алгоритмического языка следующим образом (см. [5]):

1. Начальное построение вероятностной модели (вероятности также могут быть выбраны произвольно),

L=0,0, H=1,0;

2. (Символ)=Очередной символ входного текста,

Интервал=H-L,

$H=L+Интервал \cdot H_{(символ)}$,

$L=L+Интервал \cdot L_{(символ)}$,

Изменить вероятностную модель в зависимости от (Символ);

3. Если (Символ) не равен EOF перейти к пункту 2;

4. Выходные данные=L.

Тогда процесс декодирования входного потока данных можно описать с помощью формального алгоритмического языка следующим образом (также см. [5]):

1. Начальное построение вероятностной модели,

(Число)=Прочитать двоичные данные;

2. (Символ)=Найти символ, в интервал которого попадает (Число),

Добавить к выходным данным (Символ),

$Интервал=H_{(символ)}-L_{(символ)}$,

$Число=Число-L_{(символ)}$,

Число=Число/Интервал,

Изменить вероятностную модель в зависимости от (Символ);

3. Если Символ не равен EOF, перейти к пункту 2.

Начальное построение вероятностной модели и ее изменение при обработке очередного символа должны совпадать для процессов кодирования и декодирования, иначе входные данные не будут должным

RU 2226041 C2

RU 2226041 C2

образом декодированы.

Согласно предлагаемому способу, начальное построение вероятностной модели, а также ее изменение при обработке очередного символа модифицируется с использованием ключевой последовательности К, называемой также ключом. Таким образом, не имея должного ключа, невозможно адекватно восстановить исходные входные данные, преобразованные с помощью данного способа.

Для этого предлагается вводить три дополнительных вида криптографических преобразований в алгоритм кодирования и, в частности, в вероятностную модель символов входных данных.

Первый вид дополнительного криптографического преобразования изменяет начальную таблицу априорных вероятностей увеличением вероятности появления одних символов и уменьшением других, с помощью некоторой функции F_1 в зависимости от ключевой последовательности К. Данный вид дополнительного криптографического преобразования применяется в начале процесса кодирования и декодирования один раз.

Второй и третий виды дополнительного криптографического преобразования действуют на каждом шаге кодирования, то есть один раз при кодировании каждого символа.

Второй вид дополнительного криптографического преобразования изменяет таблицу интервалов вероятности всех символов, переставляя интервалы вероятности на отрезке [0, 1) с помощью преобразования F_2 в зависимости от ключевой последовательности К и сквозного номера текущего символа. Данный вид дополнительного криптографического преобразования не влияет на длину выходных данных, а влияет лишь на скорость обработки закодированной последовательности при декодировании и потому является наиболее предпочтительным в случае, когда используются не все из предлагаемых видов дополнительного криптографического преобразования вероятностной модели.

Третий вид дополнительного криптографического преобразования изменяет таблицу интервалов вероятности всех символов, переназначая каждому символу новый интервал вероятности на отрезке [0, 1) с помощью преобразования F_3 в зависимости от ключевой последовательности К и сквозного номера текущего символа. Данный вид дополнительного криптографического преобразования, в отличие от второго и первого видов, влияет на длину выходных данных.

Согласно предлагаемому способу, в качестве криптографического преобразования данных используются все три вида дополнительных криптографических преобразований алгоритма арифметического кодирования. В качестве преобразования F_1 выбирается псевдослучайная функция, зависящая от ключевой последовательности К, с вещественным результатом на отрезке [0,0,25]. Вероятность P_i каждого i-го (i=1, 2,3,...,M) символа увеличивается на результат F_1:

$$P_i = P_i + F_1(K, i)$$

После чего все P_i нормируются так, чтобы соответствовать условию полной системы i несовместных событий [7], так что

$$\sum_i P_i = 1$$

Для этого все P_i умножаются на нормирующий множитель $0<\mu\leq 1$, который вычисляется по формуле

$$\mu = \frac{1}{\sum_i P_i}$$

В качестве преобразования F_2 выбирается псевдослучайная функция, зависящая от ключевой последовательности К, с целочисленным результатом на отрезке [1, M]. Множество ее 2M первых значений разбивается на пары (a_q, b_q), где q=1,...,M. Затем выполняется перестановка местами символа a_q с символом b_q в таблице символов.

В качестве преобразования F_3 выбирается псевдослучайная функция, зависящая от ключевой последовательности К, с целочисленным результатом на отрезке [1, M]. Множество ее 2M первых значений разбивается на пары (a_q, b_q), где q=1,...,M. Затем выполняется обмен интервалов вероятностей символа a_q и символа b_q в таблице интервалов вероятностей, то есть:

La_q заменяется на Lb_q и Lb_q заменяется на La_q;

Ha_q заменяется на Hb_q и Hb_q заменяется на Ha_q.

При обратном криптографическом преобразовании последовательность действий сохраняется и все виды дополнительных криптографических преобразований производятся аналогично.

Изобретение поясняется фиг.1,на которой представлена блок-схема устройства для реализации способа с помощью ЭВМ или вычислительного устройства.

Устройство для реализации способа состоит из блока 1 ввода ключевой последовательности; блока 2 ввода входных данных; блока 3 хранения таблицы обрабатываемых символов; блока 4 хранения таблицы интервалов вероятности для каждого символа; операционного блока 5, реализующего преобразование F_1; операционного блока 6, реализующего преобразование F_2; операционного блока 7, реализующего преобразование F_3; операционного блока 8, реализующего арифметическое сжатие с использованием блока 2, блока 3 и блока 4; блока 9 выходных данных.

Согласно предлагаемому способу, устройство работает следующим образом. Используя блок 1 (алфавитно-цифровое устройство ввода информации, как, например, описанное в [6]), в устройство вводят секретную ключевую последовательность, которая передается на вход блока 5, блока 6 и блока 7 (операционные блоки, организованные согласно [8]), а выходные данные блока 5 в блок 4, где преобразовывается таблица вероятностей появления символов. Вероятность P_i каждого i-го (i=1, 2, 3,...,M) символа увеличивается на результат преобразования F_1:

RU 2226041 C2

RU 2226041 C2

$P_i=P_i+F_1(K, i)$

После чего все P_i умножаются на нормирующий множитель $0<\mu\leq 1$, который вычисляется по формуле

$$\mu=\frac{1}{\sum_i P_i}$$

Таким образом, содержимое блока 3 и 4 определяется один раз в начале кодирования с помощью блока 5 и ключевой последовательности. С помощью блока 2 (аналогичного по устройству блоку 1) в устройство по одному символу вводят данные, предназначенные для зашифрования, после чего данные помещаются в блок 8, где посимвольно кодируются с использованием таблиц из блока 3 и блока 4 [4]. Затем на каждом шаге кодирования содержимое блока 3 обновляется с помощью блока 6 следующим образом: из блока 6 извлекается 2M первых значений, которые разбиваются на M пар (a_q, b_q), где q=1,...,M. Затем с помощью блока 6 выполняется перестановка местами символа a_q с символом b_q в таблице символов блока 3. Также, на каждом шаге кодирования содержимое блока 4 обновляется с помощью блока 7 следующим образом: из блока 7 извлекается 2M первых значений, которые разбиваются на M пар (a_q, b_q), где q=1,...,M. Затем выполняется переприсвоение интервалов вероятностей символа a_q и символа b_q в таблице интервалов вероятностей, то есть:

La_q заменяется на Lb_q и Lb_q заменяется на La_q;

Ha_q заменяется на Hb_q и Hb_q заменяется на Ha_q.

Затем входные данные, преобразованные описанным выше способом, посимвольно передаются в блок 9 (алфавитно-цифровое устройство вывода, описанное, например, в [8]), откуда могут быть непосредственно извлечены. Процесс зашифрования завершается, как только обработан последний символ входных данных. После чего зашифрованные данные извлекаются из блока 9.

При обратном криптографическом преобразовании для расшифрования данных используется устройство, идентичное данному. Расшифрование данных происходит в том же порядке.

Таким образом, в данном способе повышается сложность криптоанализа и несанкционированного дешифрования зашифрованной информации, снижаются энергетические затраты на обработку, передачу и хранение зашифрованной информации, чем достигается поставленный технический результат.

Источники информации:

1. Патент РФ №2099890 C1, кл. H 04 L 9/00.

2. Патент РФ №2103829 C1, кл. H 04 L 9/20.

3. Патент РФ №2130641 C1, кл. G 06 F 13/00, G 09 C 1/00, H 04 L 9/00 - прототип.

4. Witten Ian H., Neal Radford M., Cleary John G. Arithmetic coding for data compression. Communications of the ACM. - June 1987, Vol.30, N6.

5. Мастрюков Д. Алгоритмы сжатия информации. Арифметическое кодирование. - М.: Монитор, N1, 1994.

6. Клингман Э. Проектирование специализированных микропроцессорных систем. - М.: Мир, 1985.

7. Гмурман В. Теория вероятностей и математическая статистика. - М.: Высшая школа, 2000.

8. Зангер Г. Электронные системы. Теория и применение. - М.: Мир, 1980.

Формула изобретения:

1. Способ криптографического преобразования двоичных данных, заключающийся в том, что на передающей стороне исходная информация искажается с помощью ключевой последовательности, а на приемной стороне искаженная информация восстанавливается с помощью ключевой последовательности с посимвольной обработкой, отличающийся тем, что входные двоичные данные разбивают на блоки фиксированной длины и кодируют поочередно, подвергая их преобразованию методом арифметического кодирования с криптографическими функциями F_1, F_2 и F_3, зависящими от указанной ключевой последовательности, в качестве криптографического преобразования F_2 и F_3 выбирают псевдослучайные функции, на каждом шаге кодирования вводят данные для арифметического кодирования посимвольно и кодируют их с использованием таблицы символов, реализующих преобразование F_2 и таблицы интервалов вероятности появления символов, реализующих преобразование F_3, на каждом шаге кодирования обновляют таблицу символов путем извлечения из нее N ее первых значений, разбивают их на пары, переставляют местами извлеченные символы в таблице символов, а также на каждом шаге кодирования из таблицы интервалов вероятности появления каждого символа извлекают N ее первых значений, разбивают их на пары, затем выполняют обмен интервалами вероятности появления в указанной таблице, после чего преобразованные входные данные передают в блок выходных данных.

2. Способ по п.1, отличающийся тем, что в качестве функции преобразования F_2 используют функцию линейного конгруэнтного преобразования вида $F(x)=ax+b \bmod c$, с начальным значением, зависящим от заданной ключевой последовательности, где a и b выбираются так, чтобы генерируемая последовательность не обладала периодом максимальной длины, а константа c выбирается равной M, где M - количество символов.

3. Способ по п.1, отличающийся тем, что в качестве составляющей функций преобразования F_1 и F_3, используют нормированную функцию линейного конгруэнтного преобразования вида

$$F(x)=\frac{ax+b \bmod c}{N},$$

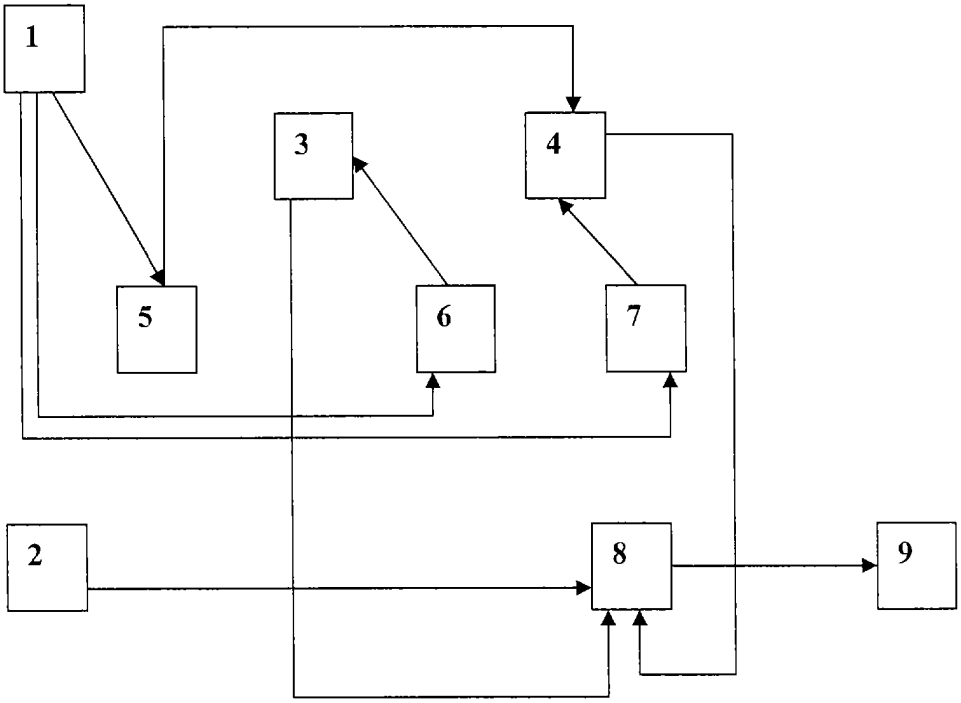
с начальным значением, зависящим от заданной ключевой последовательности и разным для обеих F_1 и F_3, где константы a и b выбираются так, чтобы генерируемая последовательность не обладала периодом максимальной длины, константа c для F_1 и F_3 выбирается равной целой части 0,25N, где N - фиксированное простое целое число

RU 2226041 C2

RU 2226041 C2

большого порядка, выбранное произвольно до начала кодирования.

RU 2226041 C2



RU 2226041 C2

RU 2226041 C2