РОССИЙСКАЯ ФЕДЕРАЦИЯ

ФЕДЕРАЛЬНАЯ СЛУЖБА
ПО ИНТЕЛЛЕКТУАЛЬНОЙ СОБСТВЕННОСТИ

(19) **RU** (11) **2 598 781** (13) **C1**

(51) МПК
H04L 9/06 (2006.01)
G06F 7/76 (2006.01)

(12) ОПИСАНИЕ ИЗОБРЕТЕНИЯ К ПАТЕНТУ

(21)(22) Заявка: **2015131963/08, 31.07.2015**

(24) Дата начала отсчета срока действия патента:
31.07.2015

Приоритет(ы):
(22) Дата подачи заявки: **31.07.2015**

(45) Опубликовано: **27.09.2016** Бюл. № **27**

(56) Список документов, цитированных в отчете о поиске: **US 5946473 A, 31.08.1999. EP 1514174 A1, 16.03.2005. WO 2010132895 A1, 18.11.2010. US 2014079215 A1, 20.03.2014. RU 2296427 C1, 27.03.2007.**

Адрес для переписки:
127287, Москва, Старый Петровско-Разумовский пр-д, 1/23, стр. 1, Открытое акционерное общество "Информационные технологии и коммуникационные системы"

(72) Автор(ы):
Борисенко Николай Павлович (RU),
Уривский Алексей Викторович (RU)

(73) Патентообладатель(и):
Открытое акционерное общество "Информационные технологии и коммуникационные системы" (RU)

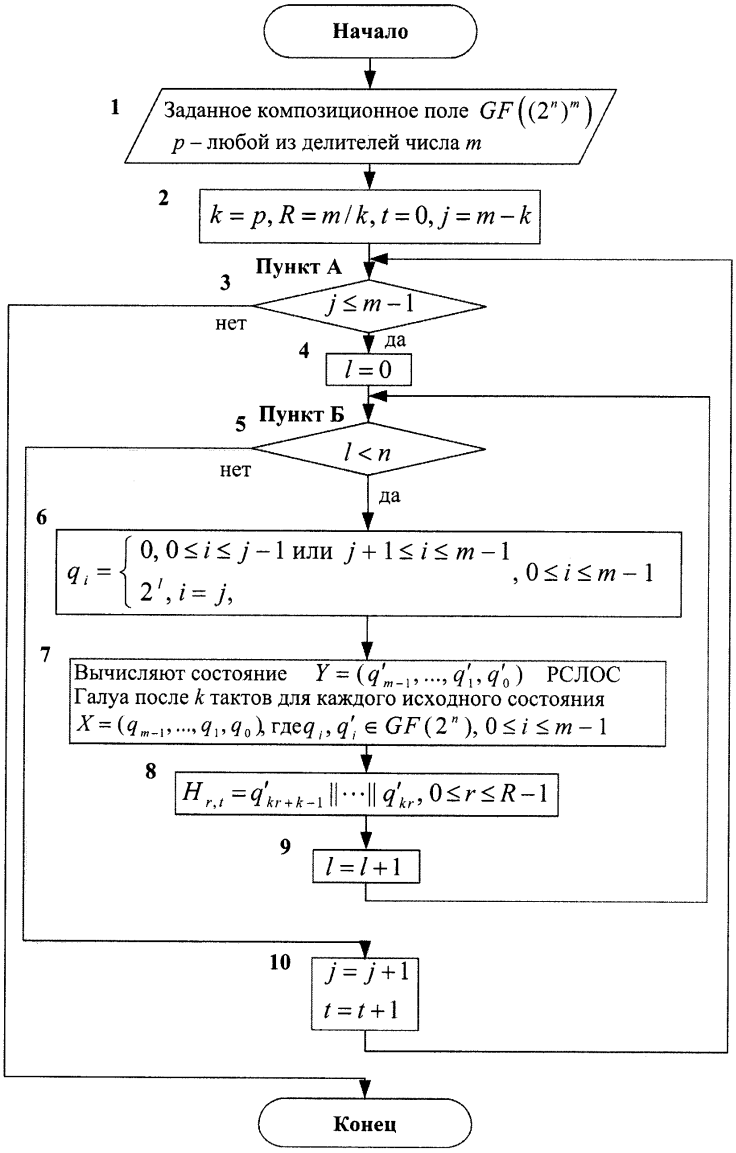
(54) СПОСОБ ЛИНЕЙНОГО ПРЕОБРАЗОВАНИЯ (ВАРИАНТЫ)

(57) Реферат:

Группа изобретений относится к области вычислительной техники и может быть использована в устройствах защиты данных. Техническим результатом является уменьшение объема памяти при заданной разрядности процессоров. Способ содержит этапы, на которых задают разрядность W процессора вычислительной системы, равную целочисленной степени числа 2, задают доступный объем памяти вычислительной системы M бит, задают размер s сообщения S, причем s кратно W, задают значение разрядности n регистра сдвига с линейной обратной связью (РСЛОС) по схеме Галуа, формируют РСЛОС по схеме Галуа, модифицируют РСЛОС, осуществляют R тактов работы модифицированного РСЛОС, вычисляют выходное состояние ячеек модифицированного РСЛОС, получают после R тактов работы РСЛОС линейное преобразование блоков s сообщения S, считывают из ячеек модифицированного РСЛОС блоки s линейно преобразованного сообщения S, объединяют блоки и получают линейно преобразованное сообщение S. 2 н.п. ф-лы, 14 ил., 3 табл.

RU 2598781 C1

RU 2598781 C1



Фиг. 6

RU 2598781 C1

RU 2598781 C1

RUSSIAN FEDERATION

FEDERAL SERVICE
FOR INTELLECTUAL PROPERTY

(19) **RU** (11) **2 598 781** (13) **C1**

(51) Int. Cl.
H04L 9/06 (2006.01)
G06F 7/76 (2006.01)

(12) ABSTRACT OF INVENTION

(21)(22) Application: **2015131963/08, 31.07.2015**

(24) Effective date for property rights:
31.07.2015

Priority:
(22) Date of filing: **31.07.2015**

(45) Date of publication: **27.09.2016** Bull. № **27**

Mail address:
127287, Moskva, Staryj Petrovsko-Razumovskij pr-d, 1/23, str. 1, Otkrytoe aktsionernoe obshchestvo "Informatsionnye tekhnologii i kommunikatsionnye sistemy"

(72) Inventor(s):
Borisenko Nikolaj Pavlovich (RU),
Urivskij Aleksej Viktorovich (RU)

(73) Proprietor(s):
Otkrytoe aktsionernoe obshshestvo "Informatsionnye tekhnologii i kommunikatsionnye sistemy" (RU)

(54) METHOD OF LINEAR CONVERSION (VERSIONS)

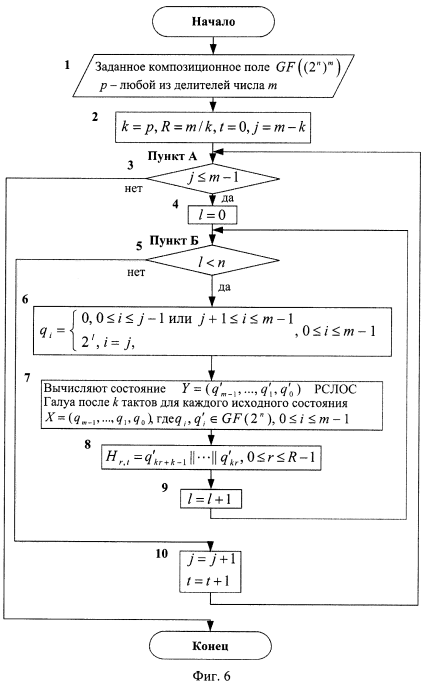
(57) Abstract:

FIELD: computer engineering.

SUBSTANCE: group of inventions relates to computer engineering and can be used in devices of data protection. Method includes steps of setting length W of the computer system processor equal to an integer degree of number 2, setting available memory volume of the computer system M bits, setting size s of message S, herewith s is multiple of W, setting capacity value n of a linear feedback shift register (LFSR) as per Galois, forming the LFSR as per Galois, modifying the LFSR, performing R cycles of the modified LFSR operation, calculating output state of the modified LFSR cells, obtaining after R cycles of the LFSR operation a linear conversion of s blocks of message S, reading from the modified LFSR cells s blocks of the linearly converted message S, combining the blocks and receiving the linearly converted message S.

EFFECT: technical result is reduced memory volume at the specified capacity of the processors.

2 cl, 14 dwg, 3 tbl



Фиг. 6

RU 2598781 C1

RU 2598781 C1

Область техники, к которой относится изобретение

Предлагаемое изобретение относится к области вычислительной техники и криптографии и, в частности, к использованию регистров сдвига для реализации линейного преобразования большой размерности и последующего применения в устройствах криптографической защиты данных.

Уровень техники

Для криптографической защиты данных используются различные способы реализации линейных преобразований.

Так, известен способ, улучающий и программную, и аппаратную реализацию фиксированного линейного преобразования шифра AES, основанный на использовании специфического вида матрицы линейного преобразования. Известный способ относится к криптографической области и также может быть использован для программной или аппаратной реализации в системах защиты информации [1].

Известны также и другие способы линейных преобразований [2-4].

Недостатками известных способов являются невозможность их применения для реализации произвольных линейных преобразований, в том числе больших размерностей, и неэффективное использование ресурсов для ряда вычислительных платформ.

Перспективным для реализации линейного преобразования является использование регистров сдвига с линейной обратной связью (РСЛОС) [5]. Такие регистры, выполняемые программно или аппаратно и способные работать как в прямом, так и в обратном направлении, могут быть реализованы на различных вычислительных платформах (фиг. 1-4).

Опубликовано большое количество научных работ, где предложено осуществление линейных преобразований на основе различных РСЛОС, включая РСЛОС типа Галуа и Фибоначчи.

Но такие линейные преобразования обычно имеют малую размерность. При построении рассеивающего слоя криптографического преобразования, например блочного шифра или хэш-функции, они не позволяют обработать целый блок большой размерности и требуют дополнительного линейного преобразования для повышения уровня защищенности, например, в стандарте AES - это функция ShiftRows(), в блочном шифре LED - функция ShiftCells(), в хэш-функции ГОСТ Р 34.11-2012 - функция перестановки байт. Обычно использование линейных преобразований малой размерности компенсируется увеличением числа раундов криптографического преобразования для достижения высокой стойкости, что ведет к снижению быстродействия.

Наиболее близким по своей технической сущности к заявляемому является способ [2], позволяющий эффективно реализовать РСЛОС, который выполняет линейную операцию и может быть применен для линейного преобразования. Способ основан на использовании разделимых таблиц и предложен для реализации РСЛОС только в двоичном поле.

Этот способ принимается за прототип.

Другой известный способ [3] позволяет реализовать РСЛОС большого размера, требует мало памяти, но работает медленно, а способ [4] работает быстро, но требует очень много памяти.

Недостатком прототипа и перечисленных известных способов является невозможность выбора параметров вычислительной системы для эффективного использования ее ресурсов, что не позволяет сократить количество необходимых тактов работы используемых в системе процессоров для вычисления результата

преобразования.

Раскрытие изобретения

Техническим результатом является обеспечение возможности выбора взаимосвязанных характеристик (быстродействие и объем необходимой памяти) для конкретной вычислительной системы при реализации линейного преобразования большой размерности.

Для этого предлагается способ, позволяющий осуществить линейное преобразование исходного сообщения с использованием РСЛОС типа Галуа фиг. 1, 2 или типа Фибоначчи фиг. 3, 4.

При этом, зная разрядность процессора и объем выделенной для реализации способа памяти, можно заранее определить, сколько тактов работы РСЛОС необходимо для вычисления линейного преобразования исходного сообщения.

Вариант предлагаемого способа, предусматривающий построение РСЛОС типа Галуа и линейного преобразования сообщения S, представленного в двоичном виде, заключающийся в том, что

• задают разрядность W процессора вычислительной системы (размер машинного слова), равную целочисленной степени числа 2;

• задают доступный объем памяти вычислительной системы M бит;

• задают размер s сообщения S, причем s кратно W;

• задают значение разрядности n регистра сдвига с линейной обратной связью (РСЛОС) по схеме Галуа (фиг. 1, 2), причем выполняется соотношение

$$n = \frac{W}{2^N},$$

где N∈0, 1, 2, …,

• формируют РСЛОС по схеме Галуа со следующими параметрами:

внутренний примитивный полином

$$f(x) = x^n \oplus \sum_{i=0}^{n-1} a_i x^i,$$

$a_i \in GF(2)$

внешний полином

$$h(y) = y^m \oplus \sum_{i=0}^{m-1} h_i y^i,$$

где $m = \frac{s}{n}$ - количество ячеек РСЛОС,

причем $h_i \in GF(2^n)$,

исходное состояние ячеек РСЛОС q_i образует вектор данных

$X=(q_{m-1}, q_{m-2}, \ldots, q_2, q_1, q_0)$,

причем $q_i \in GF(2^n)$, $0 \le i \le m-1$

выходное состояние ячеек РСЛОС q'_i за один такт работы образует вектор

$$Y = \left(q'_{m-1}, q'_{m-2}, \ldots, q'_2, q'_1, q'_0 \right),$$

причем $q'_i \in GF\left(2^n\right)$, $0 \le i \le m-1$,

где $q'_i = h_i \cdot q_{m-1} \oplus q_{i-1}$, для 1≤i≤m-1,

$q'_0 = h_0 \cdot q_{m-1}$

• определяют все делители числа m в виде значений p_0, p_1, …, p_d, причем $p_0<p_1<\ldots p_d$;

• выбирают максимально возможный делитель p из соотношения

$$p \le \frac{M}{sn};$$

• модифицируют РСЛОС, выполняя следующие действия:

○ вычисляют R матриц H_r, причем r=(R-1), …, 0. размерностью n×k строк, каждая из которых имеет длину n×k бит, выполняя следующие действия:

▪ вычисляют

k=p,

▪ вычисляют

$$R = \frac{m}{k},$$

где R - количество матриц H;

▪ вычисляют

j=m-k;

▪ вычисляют

t=0;

▪ (A1) если не выполняется соотношение

j≤m-1,

то переходят к выполнению этапа A3;

▪ вычисляют

l=0,

▪ (A2) если не выполняется соотношение

l<n,

то вычисляют

j=j+1,

t=t+1,

переходят к выполнению этапа A1;

▪ устанавливают исходное состояние РСЛОС

$X=(q_{m-1}, \ldots, q_1, q_0)$,

$$q_i = \begin{cases} 0, 0 \le i \le j-1 \text{ или } j+1 \le i \le m-1 \\ 2^l, i = j, \end{cases}, 0 \le i \le m-1,$$

где $q_i \in GF(2^n)$,

▪ вычисляют после k тактов работы для каждого исходного состояния новое состояние РСЛОС

$$Y = \left(q'_{m-1}, \ldots, q'_1, q'_0\right),$$

где $q'_i \in GF(2^n)$, 0≤i≤m-1,

▪ вычисляют t-е значения для всех матриц H_i, i=r-1…0 путем конкатенации k значений ячеек q′

$$H_{r,l} = q'_{kr+k-1} \| \dots \| q'_{kr},$$

причем 0≤r≤R-1,

▪ вычисляют

l=l+1,

переходят к выполнению этапа A2;

• (A3) записывают в ячейки модифицированного РСЛОС блоки s исходного сообщения S, причем исходное состояние ячеек модифицированного РСЛОС q_i образует вектор

$X'=(Q_{R-1}, \dots, Q_1, Q_0)$,

где Q_r - это содержимое ячеек $q_{kr+k-1} \| \cdots \| q_{kr}$,

причем 0≤r≤R-1

• осуществляют R тактов работы модифицированного РСЛОС, выполняя на каждом такте следующие действия:

▪ вычисляют выходное состояние ячеек модифицированного РСЛОС Q'_i за один такт работы, образующие вектор

$$Y' = (Q'_{R-1}, \dots, Q'_1, Q'_0),$$

каждое значение Q'_i которого вычисляется по формуле

$$Q'_i = f(H_i) \oplus Q_{i-1}$$

для каждого i=R-1, …, 1,

причем

$$Q'_0 = f(H_0),$$

где $f(H_r) = \bigoplus_{j=0}^{W-1} z_{R-1,j} \cdot H_{r,j}$,

где $z_{R-1,j}$ - значение j-го бита вектора Q_{R-1},

причем r=R-1, …, 1, 0,

j=0, 1, …, W-1,

$z_{R-1,j} \in GF(2)$;

• получают после R тактов работы РСЛОС линейное преобразование блоков s сообщения S;

• считывают из ячеек модифицированного РСЛОС блоки линейно преобразованного сообщения s;

• объединяют блоки и получают линейно преобразованное сообщение S.

Вариант предлагаемого способа, предусматривающий построение РСЛОС типа Фибоначчи и линейного преобразования сообщения S, представленного в двоичном виде, заключающийся в том, что

• задают разрядность W процессора вычислительной системы (размер машинного слова), равную целочисленной степени числа 2;

• задают доступный объем памяти вычислительной системы M бит;

• задают размер s сообщения S, причем s кратно W;

• задают значение разрядности n регистра сдвига с линейной обратной связью (РСЛОС) по схеме Фибоначчи (фиг. 3, 4), причем выполняется соотношение

$$n = \frac{W}{2^N},$$

где N∈0, 1, 2, …,

• формируют РСЛОС по схеме Фибоначчи со следующими параметрами:

внутренний примитивный полином

$$f(x) = x^n \oplus \sum_{i=0}^{n-1} a_i x^i,$$

$a_i \in GF(2)$

внешний полином

$$h(y) = y^m \oplus \sum_{i=0}^{m-1} h_i y^i,$$

где $m = \frac{s}{n}$ - количество ячеек РСЛОС,

причем $h_i \in GF(2^n)$

исходное состояние ячеек РСЛОС q_i образует вектор

$X=(q_{m-1}, q_{m-2}, \ldots, q_2, q_1, q_0)$,

причем $q_i \in GF(2^n)$, $0 \le i \le m-1$

выходное состояние ячеек РСЛОС q'_i за один такт работы образует вектор

$$Y = (q'_{m-1}, q'_{m-2}, \ldots, q'_2, q'_1, q'_0),$$

причем $q'_i \in GF(2^n)$, $0 \le i \le m-1$,

где $q'_i = q_{i+1}$,

для каждого i=0, …, m-2

$$q'_{m-1} = \bigoplus_{i=0}^{m-1} h_i \cdot q_i$$

• определяют все делители числа m в виде значений $p_0, p_1, \ldots, p_d$, причем $p_0 < p_1 < \ldots p_d$;

• выбирают максимально возможный делитель p из соотношения

$$p \le \frac{M}{sn};$$

• модифицируют РСЛОС, выполняя следующие действия:

○ вычисляют R матриц H_r, причем r=(R-1), …, 0, размерностью n×k строк, каждая из которых имеет длину n×k бит, выполняя следующие действия:

- вычисляют

k=p,

▪ вычисляют

$$R = \frac{m}{k},$$

где R - количество матриц H_r;

▪ вычисляют

r=0;

▪ (A5) если не выполняется соотношение

r<R,

то переходят к выполнению этапа A7;

▪ вычисляют

j=0,

▪ (A6) если не выполняется соотношение

j<k,

то вычисляют

r=r+1,

переходят к выполнению этапа A5;

▪ вычисляют

l=0,

▪ если не выполняется соотношение

l<n,

то вычисляют

j=j+1,

переходят к выполнению этапа A6;

▪ устанавливают исходное состояние РСЛОС

$X=(q_{m-1}, q_{m-2}, \ldots, q_1, q_0)$,

$$q_i = \begin{cases} 2^l, i = rk + j, \\ 0, \text{ иначе} \end{cases}, 0 \le i \le m-1$$

где $q_i \in GF(2^n)$;

▪ вычисляют после k тактов работы для каждого исходного состояния новое состояние РСЛОС

$$Y = (q'_{m-1}, q'_{m-2}, \ldots, q'_1, q'_0),$$

где $q'_i \in GF(2^n)$, $0 \le i \le m-1$;

▪ вычисляют (jk+l)-е значение для матрицы H_r путем конкатенации k значений ячеек

$q'_{m-1}, q'_{m-2}, \ldots, q'_{m-k}$

$$H_{r,t} = q'_{kr+k-1} \| \ldots \| q'_{kr},$$

причем $0 \le r \le R-1$;

▪ вычисляют

l=l+1,

переходят к выполнению этапа A6;

• (A7) записывают в ячейки модифицированного РСЛОС блоки s исходного сообщения S, причем исходное состояние ячеек модифицированного РСЛОС q_i образует вектор

$X'=(Q_{R-1}, \ldots, Q_1, Q_0)$,

где $Q_r = q_{kr+k-1} \| \cdots \| q_{kr}$,

причем $0 \le r \le R-1$;

• осуществляют R тактов работы модифицированного РСЛОС, выполняя на каждом такте следующие действия:

▪ вычисляют выходное состояние ячеек модифицированного РСЛОС Q'_i за один такт работы, образующие вектор

$$Y' = (Q'_{R-1}, ..., Q'_1, Q'_0),$$

каждое значение Q'_i которого вычисляется по формуле

$$Q'_i = Q_{i+1}$$

для каждого i=0, …, R-2,

и

$$Q'_{R-1} = Q'_{m-1},$$

а значение Q'_{m-1} вычисляется по соотношению

$$Q'_{m-1} = \bigoplus_{r=0}^{R-1} f(H_r),$$

где $f(H_r) = \bigoplus_{j=0}^{W-1} z_{r,j} \cdot H_{r,j}$,

где $z_{r,j}$ - значение j-го бита вектора Q_r,

причем r=R-1, …, 1, 0,

j=0, 1, …, W-1,

$z_{r,j} \in GF(2)$;

• получают после R тактов работы РСЛОС линейное преобразование блоков s сообщения S;

• считывают из ячеек модифицированного РСЛОС блоки s линейно преобразованного сообщения S;

• объединяют блоки и получают линейно преобразованное сообщение S.

Для реализации предложенного способа с использованием РСЛОС типа Галуа модифицируют РСЛОС.

Основное отличие модифицированного РСЛОС Галуа - способ вычисления значения функции обратной связи. В модифицированных РСЛОС Галуа значения функции обратной связи регистра вычисляются по таблицам, в зависимости от значений бит старшей ячейки регистра.

Исходное линейное преобразование $L: V_s \mapsto V_s$. Преобразование L задается на основе РСЛОС Галуа над композиционным полем $GF((2^n)^m)$, где s=m×n, с помощью внутреннего примитивного полинома

$$f(x) = x^n \oplus \sum_{i=0}^{n-1} a_i x^i,$$

где $a_i \in GF(2)$,

и внешнего неприводимого полинома

$$h(y) = y^m \oplus \sum_{i=0}^{m-1} h_i y^i,$$

где $h_i \in GF(2^n)$ и $h_0=1$.

Исходное состояние ячеек РСЛОС Галуа q_i образует вектор данных

$X=(q_{m-1}, q_{m-2}, …, q_2, q_1, q_0)$,

где $q_i \in GF(2^n)$, $0 \le i \le m-1$.

Элементы композиционного поля $GF((2^n)^m)$ также вычисляются с помощью следующего регистра сдвига с линейной обратной связи типа Галуа (далее РСЛОС) на основе полиномов f(x) и h(y) [4].

Под линейным преобразованием L исходного вектора данных $X=(q_{m-1}, q_{m-2}, \ldots, q_2, q_1, q_0)$ будем понимать результат m тактов работы РСЛОС.

Выходное состояние ячеек РСЛОС Галуа q'_i за один такт работы образует вектор

$$Y = (q'_{m-1}, q'_{m-2}, \ldots, q'_2, q'_1, q'_0),$$

где $q'_i \in GF(2^n)$, $0 \le i \le m-1$,

и каждое значение q'_i вычисляется по формуле

$$q'_i = h_i \cdot q_{m-1} \oplus q_{i-1}$$

для каждого i=m-1, …, 1 и $q'_0 = h_0 \cdot q_{m-1}$.

Операции сложения и умножения двух n-разрядных чисел в РСЛОС Галуа осуществляются в поле $GF(2^n)$. Линейное преобразование исходного вектора данных осуществляется за m тактов работы РСЛОС типа Галуа.

Итогом преобразования является новое состояние регистра на m-ом такте. А обратное линейное преобразование L^{-1} осуществляется за m тактов работы РСЛОС в обратном направлении.

Пусть $p_0, p_1, \ldots, p_d$, - все делители числа m, причем $p_0 < p_x < \ldots p_d$. Обозначаются значения $k=p_i$, $R = \frac{m}{k}$ и W=nk, где W - разрядность процессора, на котором реализуется исходное линейное преобразование, p_i выбирается исходя из размера доступной памяти M. При этом общая схема модифицированного РСЛОС Галуа имеет вид, показанный на фиг. 5.

Пусть исходное состояние ячеек модифицированного РСЛОС Галуа образует вектор $X'=(Q_{R-1}, \ldots, Q_1, Q_0)$,

где Q_r равно содержимому ячеек $q_{kr+k-1} \| \cdots \| q_{kr}$,

причем $0 \le r \le R-1$.

Выходное состояние ячеек модифицированного РСЛОС Галуа Q'_i за один такт работы образует вектор $Y' = (Q'_{R-1}, \ldots, Q'_1, Q'_0)$, и каждое значение Q'_i для каждого r=R-1, …, 1 вычисляется по формуле

$$Q'_r = f(H_r) \oplus Q_{r-1}$$

причем

$$Q'_0 = f(H_0),$$

а функция определяется в виде

$$f(H_r) = \bigoplus_{j=0}^{W-1} z_{R-1,j} \cdot H_{r,j},$$

где r=R-1, …, 1, 0,

$z_{R-1,j} \in GF(2)$,

j=0, 1, …, W-1 - биты ячейки Q_{R-1} модифицированного РСЛОС Галуа.

Если состояние на m-ом такте есть результат линейного преобразования L по схеме РСЛОС Галуа (фиг. 1), то такое же состояние будет получено на R-ом такте работы модифицированного РСЛОС Галуа (фиг. 5). Причем R тактов работы модифицированного РСЛОС требуют

$$R \cdot W = \frac{m}{k} \cdot n \cdot k = m \cdot n$$

операций проверки "true - false" для всех бит ячейки Q_{R-1}. Количество сложений по модулю два W-разрядных чисел для каждого вычисления значения $f(H_r)$ по каждой таблице равно W - 1. Следовательно, каждый такт работы модифицированного РСЛОС Галуа требует следующего количества сложений

$$R(W-1) + R - 1 = RW - 1 = \frac{m}{k} nk - 1 = mn - 1$$

В итоге необходимое количество сложений по модулю два W-разрядных чисел для R тактов работы модифицированного РСЛОС Галуа равно

$$R(mn-1) = \frac{m(mn-1)}{k}$$

Объем необходимой памяти равен

$$M = R \cdot W \cdot W = \frac{m}{k} \cdot (n \cdot k)^2 = m \cdot n^2 \cdot k \text{ (бит)}$$

для сохранения R таблиц H_r, r=15, …, 0.

Для правильного функционирования схемы фиг. 5 по правилу схемы фиг. 1 (получения одинакового выхода при одинаковых входных данных) необходимо определить R таблиц H_r, r=(R-1), …, 0. Блок-схема процесса их вычисления представлена на фиг. 6.

Последовательность вычисления R таблиц H_r, r=(R-1), …, 0 базируется на принципе суперпозиции линейных преобразований. Входные данные алгоритма - линейное преобразование над заданным композиционным полем $GF((2^n)^m)$, и p - любой из делителей числа m. А выходные - R необходимых таблиц H_r, r=(R-1), …, 0.

Рассмотрим каждый шаг алгоритма (фиг. 6).

Шаг 1 [Блок 2]: Присваивают значения k=p, $R = \frac{m}{k}$, j=m-k и t=0;

Шаг 2 [Пункт А - блок 3]: проверяют условие

j≤m-1

• если условие выполняется, то присваивают l=0 (блок 4) и переходят к шагу 3 [Пункту Б];

• если условие не выполняются, то завершают процесс;

Шаг 3 [Пункт Б - блок 5] проверяют условие

l<n

• если условие выполняется, то

○ определяют исходное состояние РСЛОС Галуа (блок 6):

$$q_i = \begin{cases} 0, 0 \le i \le j-1 \text{ или } j+1 \le i \le m-1 \\ 2^l, i = j, \end{cases}, 0 \le i \le m-1,$$

○ вычисляют новое состояние $Y = (q'_{m-1}, ..., q'_1, q'_0)$ РСЛОС Галуа после k тактов работы для каждого исходного состояния $X=(q_{m-1}, \ldots, q_1, q_0)$, где $q'_i \in GF(2^n)$, $0 \le i \le m-1$ (блок 7),

○ вычисляют t-е значения для всех таблиц H путем конкатенации k значений ячеек q′ (блок 8):

$$H_{r,t} = q'_{kr+k-1} \| \cdots \| q'_{kr}, \quad 0 \le r \le R-1,$$

○ увеличивают значение l=l+1 (блок 9), и переходят к шагу 3 [Пункту Б];

• если условие не выполняются, то увеличивают значения j=j+1, t=t+1 (блок 10), и переходят к шагу 2 [Пункту А].

Порядок вычисления необходимых таблиц для обратного линейного преобразования L^{-1} выполняется аналогичным образом. Но при этом полученный модифицированный РСЛОС типа Галуа будет работать в противоположном направлении с проверкой на "true - false" для всех бит ячейки Q_0 вместо Q_{R-1}.

Если линейное преобразование $L: V_s \mapsto V_s$ задается на основе РСЛОС Фибоначчи над композиционным полем $GF((2^n)^m)$, где s=m×n, с помощью внутреннего примитивного полинома

$$f(x) = x^n \oplus \sum_{i=0}^{n-1} a_i x^i,$$

где $a_i \in GF(2)$,

и внешнего неприводимого полинома

$$h(y) = y^m \oplus \sum_{i=0}^{m-1} h_i y^i,$$

где $h_i \in GF(2^n)$ и $h_0=1$,

то можно его реализовать по схеме модифицированного РСЛОС Фибоначчи.

Исходное состояние ячеек РСЛОС Фибоначчи q_i образует вектор данных

$X=(q_{m-1}, q_{m-2}, \ldots, q_2, q_1, q_0)$,

где $q_i \in GF(2^n)$, $0 \le i \le m-1$

Выходное состояние ячеек РСЛОС Фибоначчи q'_i за один такт работы образует вектор

$$Y = \left(q'_{m-1}, q'_{m-2}, ..., q'_2, q'_1, q'_0 \right),$$

где $q'_i \in GF(2^n)$, $0 \le i \le m-1$

и каждое значение q'_i вычисляется по формуле

$$q'_i = q_{i+1}$$

для каждого i=0, …, m-2 и

$$q'_{m-1} = \bigoplus_{i=0}^{m-1} h_i \cdot q_i .$$

Операции сложения и умножения двух n-разрядных чисел в РСЛОС Фибоначчи осуществляются в поле $GF(2^n)$. Линейное преобразование исходного вектора данных - m тактов работы РСЛОС Фибоначчи (фиг. 3). Итогом преобразования является новое состояние регистра на m-ом такте. А обратное линейное преобразование L^{-1} достигается через m тактов работы РСЛОС Фибоначчи в обратном направлении (фиг. 4).

В этом случае общая схема модифицированного РСЛОС Фибоначчи имеет вид, представленный на фиг. 7.

Пусть исходное состояние ячеек модифицированного РСЛОС Фибоначчи образует вектор

$X'=(Q_{R-1}, \ldots, Q_1, Q_0)$,

где $Q_r = q_{kr+k-1} \| \cdots \| q_{kr}$, $0 \le r \le R-1$.

Выходное состояние ячеек Q'_i за один такт работы образует вектор

$$Y' = (Q'_{R-1}, \ldots, Q'_1, Q'_0),$$

и каждое значение Q'_i вычисляется по формуле $Q'_r = Q_{r+1}$ для каждого r=0, …, R-2 и

$$Q'_{R-1} = \bigoplus_{r=0}^{R-1} f(H_r),$$

где $f(H_r) = \bigoplus_{j=0}^{W-1} z_{r,j} \cdot H_{r,j}$,

r=R-1, …, 1, 0

$z_{r,j} \in GF(2)$,

j=0, 1, …, W-1 - биты ячейки модифицированного РСЛОС Фибоначчи.

Если состояние на m-ом такте есть результат линейного отображения L по схеме РСЛОС Фибоначчи на фиг. 3, то состояние на R-ом такте соответствует его результату по схеме модифицированного РСЛОС Фибоначчи на фиг. 7. Причем R тактов его работы требуют

$$R \cdot R \cdot W = \left(\frac{m}{k}\right)^2 \cdot n \cdot k = \frac{m^2 \cdot n}{k}$$

операций проверки "true - false". Количество сложений по модулю два поразрядных чисел для вычисления каждого значения $f(H_r)$ по каждой таблице равно W-1. Следовательно, каждый такт работы модифицированного РСЛОС Фибоначчи требует

$$R(W-1) + R - 1 = RW - 1 = \frac{m}{k} nk - 1 = mn - 1$$

операций сложения по модулю два W-разрядных чисел. В итоге необходимое количество сложений по модулю два W-разрядных чисел для R тактов работы регистра равно

$$R(mn-1)=\frac{m(mn-1)}{k}$$

Объем необходимой памяти равен

$$R\cdot W\cdot W=\frac{m}{k}\cdot(n\cdot k)^2=m\cdot n^2\cdot k \text{ (бит)}$$

для сохранения R таблиц H_r, r=15, …, 0.

Для корректной работы модифицированной схемы необходимо определить R таблиц H_r, r=(R-1), …, 0. Блок-схема процесса их вычисления представлена на фиг. 8.

Алгоритм вычисления R таблиц H_r, r=(R-1), …, 0 также базируется на принципе суперпозиции линейных преобразований. Входные данные алгоритма - линейное преобразование над заданным композиционным полем $GF((2^n)^m)$, построенное по схеме РСЛОС Фибоначчи, и p - любой из делителей числа m. А выходные - R необходимых таблиц H_r, r=(R-1), …, 0.

Рассмотрим каждый шаг алгоритма.

Шаг 1 [Блок 2]: Присваивают значения k=p, $R=\frac{m}{k}$ и r=0;

Шаг 2 [Пункт А - блок 3]: проверяют условие

r<R

• если условие выполняется, то присваивают j=0 (блок 4) и переходят к шагу 3 [Пункту Б];

• если условие не выполняется, то завершают процесс; Шаг 3 [Пункт Б - блок 5] проверяют условие

j<k,

• если условие выполняется, то присваивают l=0 (блок 6) и переходят к шагу 4 [Пункту В];

• если условие не выполняется, то увеличивают значение r=r+1 (блок 12), и переходят к шагу 2 [Пункту А];

Шаг 4 [Пункт В - блок 7] проверяют условие

l<n,

• если условие выполняется, то

○ определяют исходное состояние РСЛОС Фибоначчи (блок 8):

$$q_i=\begin{cases}2^l, i=rk+j,\\ 0, \text{ иначе}\end{cases}, 0\le i\le m-1,$$

○ вычисляют новое состояние $Y=(q'_{m-1},\ldots,q'_1,q'_0)$ РСЛОС Фибоначчи после k тактов работы для каждого исходного состояния $X=(q_{m-1},\ldots,q_1,q_0)$, где $q'_i\in GF(2^n)$, $0\le i\le m-1$ (блок 9),

○ вычисляют (jk+l)-е значение для таблицы H_r путем конкатенации k значений ячеек $q'_{m-1}, q'_{m-2}, \ldots, q'_{m-k}$ (блок 10)

$$H_{r,jk+l}=q'_{m-1}\|\cdots\|q'_{m-k}$$

○ увеличивают значение l=l+1 (блок 10) и переходят к шагу 4 [Пункту В];

○ если условие не выполняется, то увеличивают значение j=j+1 (блок 11), и переходят к шагу 3 [Пункту Б].

Порядок вычисления необходимых таблиц для обратного линейного отображения L^{-1} выполняется аналогичным образом. Но при этом необходимо использовать схему РСЛОС Фибоначчи (фиг. 4) для вычисления его состояния (блок 9, фиг. 8). Полученный модифицированный РСЛОС сдвигается в противоположном направлении.

Краткое описание чертежей

На фиг. 1 показана схема работы линейного регистра сдвига с линейной обратной связью типа Галуа в прямом направлении.

На фиг. 2 показана схема работы линейного регистра сдвига с линейной обратной связью типа Галуа в обратном направлении.

На фиг. 3 показана схема работы линейного регистра сдвига с линейной обратной связью типа Фибоначчи в прямом направлении.

На фиг. 4 показана схема работы линейного регистра сдвига с линейной обратной связью типа Фибоначчи в обратном направлении.

На фиг. 5 показана схема работы модифицированного линейного регистра сдвига с линейной обратной связью типа Галуа.

На фиг. 6 показана блок-схема алгоритма вычисления таблиц функции обратной связи модифицированного линейного регистра сдвига с линейной обратной связью типа Галуа.

На фиг. 7 показана схема работы модифицированного линейного регистра сдвига с линейной обратной связью типа Фибоначчи.

На фиг. 8 показана блок-схема алгоритма вычисления таблиц функции обратной связи модифицированного линейного регистра сдвига с линейной обратной связью типа Фибоначчи.

На фиг. 9 показана схема линейного регистра сдвига с линейной обратной связью типа Галуа для прямого линейного преобразования для примера реализации способа.

На фиг. 10 показана схема линейного регистра сдвига с линейной обратной связью типа Галуа для обратного линейного преобразования для примера реализации способа.

На фиг. 11 показана схема работы модифицированного линейного регистра сдвига с линейной обратной связью типа Галуа для прямого линейного преобразования для примера реализации способа.

На фиг. 12 показана схема работы модифицированного 16-разрядного линейного регистра сдвига с линейной обратной связью типа Галуа для прямого линейного преобразования для примера реализации способа.

На фиг. 13 показана схема работы модифицированного 32-разрядного линейного регистра сдвига с линейной обратной связью типа Галуа для прямого линейного преобразования для примера реализации способа.

На фиг. 14 показана схема работы модифицированного 64-разрядного линейного регистра сдвига с линейной обратной связью типа Галуа для прямого линейного преобразования для примера реализации способа.

Осуществление изобретения

Рассмотрим пример реализации предложенного способа с использованием модифицированного РСЛОС типа Галуа.

Предложенный способ может быть реализован в прикладной программе для вычислительной системы, в качестве которой может быть использован компьютер с одним процессором с разрядностью 8 и выше, работающий под управлением операционной системы (например, Microsoft Windows 7).

Прикладная программа, реализующая работу РСЛОС типа Галуа (или типа Фибоначчи), может быть составлена специалистом по программированию (программистом) на основе знания известных принципов и структуры РСЛОС соответствующего типа и действий предложенного способа.

Для удобства при анализе и синтезе, в описании изобретения рассматривается линейное преобразование L с конкретными параметрами, типичными для большого класса криптографических алгоритмов:

• исходное линейное преобразование $L: V_{128} \mapsto V_{128}$;

• композиционное поле $GF((2^8)^{16})$ (m=16, n=8);

• внутренний примитивный полином

$$f(x) = x^8 \oplus x^7 \oplus x^6 \oplus x \oplus 1$$

для построения поля $GF(2^8)$;

• внешний неприводимый полином h(y) для построения композиционного поля $GF((2^8)^{16})$

$$h(y) = y^{16} + 148y^{15} + 32y^{14} + 133y^{13} + 16y^{12} + 194y^{11} + 192y^{10} + y^9 + \\ + 251y^8 + y^7 + 192y^6 + 194y^5 + 16y^4 + 133y^3 + 32y^2 + 148y + 1,$$

где

$$(h_{15}, h_{14}, ..., h_0) = (148, 32, 133, 16, 194, 192, 1, 251, 1, 192, 194, 16, 133, 32, 148, 1)$$

$h_i \in GF(2^8)$

Схема РСЛОС для прямого преобразования L и РСЛОС для обратного преобразования L^{-1} изображены на фиг. 9 и 10 соответственно.

Ранее для линейного преобразования было выявлено полезное в области криптографии свойство: если в ячейки РСЛОС записать любую последовательность символов и «сдвинуть» регистр 16 раз влево в регистре останутся проверочные символы кода с максимальным расстоянием (МДР кода) С(32,16,17) [6]. Минимальное расстояние между любыми кодовыми словами данного кода равно 17. Если взять такой код в качестве линейного преобразования блочного шифра, то оно будет обладать максимальным свойством рассеивания (d=17).

Последовательность работы одного такта РСЛОС:

○ исходное состояние - вектор $X=(q_{15}, q_{14}, \ldots, q_2, q_1, q_0)$, где $q_i \in GF(2^8)$, $0 \le i \le 15$. Вектор X имеет 16 координат, расположенных слева направо в 16 ячеек РСЛОС, начиная с координаты индексом i=15;

○ в работе РСЛОС Галуа только значение q_{15} в самой старшей ячейке участвует в выработке значения функции обратной связи;

○ выходное состояние - вектор $Y = (q'_{15}, q'_{14}, ..., q'_2, q'_1, q'_0)$, где $q'_i \in GF(2^8)$, $0 \le i \le 15$. Значения q'_i для каждого i=15, …, 1 вычисляются

$$q'_i = h_i \cdot q_{15} \oplus q_{i-1},$$

при этом $q_0=h_0 \cdot q_{15}$

Под линейным преобразованием L исходного вектора данных

$X=(q_{15}, q_{14}, \ldots, q_2, q_1, q_0)$

будем понимать 16 тактов работы РСЛОС.

Итогом преобразования является новое состояние регистра на m-ом такте, которое можно записать следующим образом

$$Y = (q'_{15},\ q'_{14}, \ldots, q'_2,\ q'_1, q'_0),$$

где $q'_i \in GF(2^8)$, $0 \le i \le 15$ - значения ячеек РСЛОС.

Обратное преобразование L^{-1} - 16 тактов работы РСЛОС в обратном направлении.

Обозначим через k какой-нибудь делитель числа m=16 (его выбор определяется имеющейся разрядностью процессора W и допустимым объемом памяти M).

Сущность предлагаемого способа реализации на соответствующей платформе зависит от значения k и основывается на применении принципа суперпозиции при рассмотрении влияния каждого бита текущего состояния РСЛОС на последующее. В соответствии с тем, что для каждого k имеется способ реализации преобразования L на (nk)-разрядном процессоре.

Рассмотрим следующие случаи.

Расчетный случай 1: k=1. В этом случае рассматривается способ реализации преобразования L на 8-разрядных процессорах ($n \cdot k = 8 \cdot 1 = 8$). Для данного случая выполняются следующие действия:

• вычисляют количество r необходимых вычисляемых таблиц H_j, j=15, …, 0

$$r = \frac{m}{k} = \frac{16}{1} = 16$$

• вычисляют 16 таблиц H_j, j=15, …, 0, каждая из них имеет $nk = 8 \cdot 1 = 8$ элементов, а элементы представляют собой 8-разрядные числа (элементы поля $GF(2^8)$), выполняя следующие действия:

○ вычисляют соответствующее состояние РСЛОС (фиг. 1) после k=1 тактов по каждому исходному состоянию

$$X = (q_{m-1}, q_{m-2}, \ldots, q_2, q_1, q_0) = (q_{15}, \underbrace{0, \ldots, 0}_{15\ \text{раз}})$$

для всех $q_{15}=2^l$, l=0, …, 7, т.е. рассматривают влияние каждого бита числа q_{15} на состояние РСЛОС (фиг. 1) после k=1 тактов, в результате получают nk=8 состояний;

○ составляют массив A из nk=8 полученных состояний таким, что последняя его строка соответствует состоянию при $j=m-k=16-1=15$ и l=0, предпоследняя его строка соответствует состоянию при $j=m-k=16-1=15$ и l=1, и т.д. В результате чего массив A имеет nk=8 строк и m=16 столбцов. Первый столбец массива A соответствует значениям ячейки q_{15}, второй - q_{14} и т.д. (табл. 1);

○ в случае k=1 таблицы H_j, j=15, …, 0 равны каждому столбцу массива A в соответствии с индексами.

Таблица 1

Результаты работы РСЛОС после одного такта

Исходные состояния $q_{15}, \ldots, q_0$	Состояния РСЛОС															
	H15	H14	H13	H12	H11	H10	H9	H8	H7	H6	H5	H4	H3	H2	H1	H0
2^7.0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0	E5	6D	B2	D7	6E	AD	80	DE	80	AD	6E	D7	B2	6D	E5	80
2^6.0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0	93	D7	59	8A	37	B7	40	6F	40	B7	37	8A	59	D7	93	40
2^5.0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0	A8	8A	CD	45	FA	BA	20	D6	20	BA	FA	45	CD	8A	A8	20
2^4.0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0	54	45	87	C3	7D	5D	10	6B	10	5D	7D	C3	87	45	54	10
2^3.0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0	2A	C3	A2	80	DF	CF	08	D4	08	CF	DF	80	A2	C3	2A	08
2^2.0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0	15	80	51	40	8E	86	04	6A	04	86	8E	40	51	80	15	04
2^1.0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0	EB	40	C9	20	47	43	02	35	02	43	47	20	C9	40	EB	02
2^0.0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0	94	20	85	10	C2	C0	01	FB	01	C0	C2	10	85	20	94	01

• строят расширенную схему РСЛОС (фиг. 5), который имеет r=16 ячеек, значение каждой из которых представляет собой 8-разрядное число;

• обозначают через (Q_{15}, Q_{14}, …, Q_2, Q_1, Q_0) состояние расширенного РСЛОС, где $Q_i \in GF(2^8)$, i=15, 14, …, 0;

• определяют значение f (H_j) функции обратной связи для каждой H_j по формуле

$$f(H_j) = \bigoplus_{u=0}^{7} w_{15,u} \cdot H_{j,u},$$

где j=15, …, 1, 0,

$w_{15,u} \in GF(2)$

u=0, 1, …, 7 - биты ячейки Q_{15} расширенного РСЛОС.

Это значит, что если u-й бит ячейки Q_{15} равен единице, то соответствующая строка $H_{j,u}$ участвует в процессе выработки значения функций обратной связи.

• прямое линейное преобразование L есть r=16 тактов работы расширенного РСЛОС;

• 16 тактов работы расширенного РСЛОС требуют

$r \cdot n \cdot k = \frac{m}{k} \cdot n \cdot k = m \cdot n = 16 \cdot 8 = 128$ операций проверки "true - false" для всех битов ячейки Q_{15} и

$r \cdot r \cdot n \cdot k = \frac{m}{k} \cdot \frac{m}{k} \cdot n \cdot k = \frac{m^2 \cdot n}{k} = \frac{16^2 \cdot 8}{1} = 2048$ операций сложения по модулю два двух n×k-разрядных чисел Q_l и $H_{l,j}$, где l=0, 1, …, r-1 и j=0, 1, …, nk.

Объем необходимой памяти равен

$$r \cdot (n \cdot k) \cdot (n \cdot k) = \frac{m}{k} \cdot (n \cdot k)^2 = m \cdot n^2 \cdot k = 16 \cdot 8^2 \cdot 1 = 1024 \text{ бит} = 128 \text{ байт}$$

для сохранения 16 таблиц H_j, j=15, …, 0.

Порядок реализации обратного линейного преобразования L^{-1} на 8-разрядных процессорах выполняется аналогичным образом с использованием РСЛОС (фиг. 2) для вычисления 16 таблиц H_j, j=15, …, 0. За счет симметричности внешнего

неприводимого полинома h(y) для рассмотренного линейного преобразования можно использовать те же 16 таблиц H_j, j=15, …, 0 и для реализации прямого преобразования, и для обратного ему.

Расчетный случай 2: k=2. В этом случае рассматривается способ реализации преобразования L на 16-разрядных процессорах. Данный способ включает следующие действия:

• вычисляют количество r необходимых вычисляемых таблиц H_j, j=7, …, 0

$$r = \frac{m}{k} = \frac{16}{2} = 8$$

• вычисляют r=8 таблиц H_j, j=7, …, 0, каждая из них имеет nk=8·2=16 элементов, а элементы представляют собой 16-разрядные числа, выполняя следующие действия:

○ вычисляют соответствующее состояние РСЛОС (фиг. 1) после k=2 тактов по каждому исходному состоянию

$$X = (q_{m-1}, q_{m-2}, ..., q_2, q_1, q_0) = (q_{15}, q_{14}, \underbrace{0, ..., 0}_{14 \text{ раз}})$$

для всех

$q_j=2^l$, j=14, 15 и l=0, …, 7,

начиная с ячейки на позиции j=m-k=16-2=14, т.е. рассматривают влияние каждого бита числа $q_{15} \| q_{14}$ на состояние РСЛОС (фиг. 1) после k=2 тактов. В результате получают nk=16 состояний;

○ составляют массив A из nk=16 полученных состояний таким, что последняя его строка соответствует состоянию при j=m-k=16-2=14 и l=0, предпоследняя его строка соответствует состоянию при j=m-k=16-2=14 и l=1, и т.д., и первая строка массива A соответствует состоянию при j=m-1=16-1=15 и l=n-1=8-1=7. В результате чего массив A имеет nk=16 строк и m=16 столбцов;

○ формируют r=8 таблиц H_j, j=7, …, 0, начиная со значения j=(r-1)=8-1=7, путем конкатенации k=2 значений в соседних столбцах массива A, начиная с первого его столбца, причем нумеруют таким образом, например, для таблицы H_7, чтобы значение в первой строке первого столбца после конкатенации соответствует $H_{7,15}$, а значение в последней строке того же столбца соответствует $H_{7,0}$ (табл. 2).

Таблица 2

Состояния РСЛОС после двух тактов его работы

	Исходные состояния $q_{15}, \ldots, q_0$	Состояния РСЛОС после конкатенации 2-х соседних ячеек							
		H_7	H6	H5	H4	H3	H2	H1	H0
15	2^7,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0	3222	2526	4266	3B76	4888	38FA	9F75	DFE5
14	2^6,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0	1911	F313	2133	FC3B	2444	1C7D	AEDB	8E93
13	2^5,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0	EDE9	98E8	F1F8	7EFC	1222	0EDF	578C	47A8
12	2^4,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0	9795	4C74	997C	3F7E	0911	078E	CA46	C254
11	2^3,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0	AAAB	263A	AD3E	FE3F	E5E9	E247	6523	612A
10	2^2.0.0.0,0,0,0,0,0,0,0,0,0,0,0,0	55B4	131D	B71F	7FFE	9395	71C2	D3F0	D115
9	2^1,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0	CB5A	E8EF	BAEE	DE7F	A8AB	D961	8878	89EB
8	2^0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0	842D	7496	5D77	6FDE	54B4	8DD1	443C	A594
7	$0,2^7$.0.0,0,0,0,0,0,0,0,0,0,0,0,0	E56D	B2D7	6EAD	80DE	80AD	6ED7	B26D	E580
6	$0,2^6$.0,0,0,0,0,0,0,0,0,0,0,0,0,0	93D7	598A	37B7	406F	40B7	378A	59D7	9340
5	$0,2^5$.0,0,0,0,0,0,0,0,0,0,0,0,0,0	A88A	CD45	FABA	20D6	20BA	FA45	CD8A	A820
4	$0,2^4$,0,0,0,0,0,0,0,0,0,0,0,0,0,0	5445	87C3	7D5D	106B	105D	7DC3	8745	5410
3	$0,2^3$,0,0,0,0,0,0,0,0,0,0,0,0,0,0	2AC3	A280	DFCF	08D4	08CF	DF80	A2C3	2A08
2	$0,2^2$,0,0,0,0,0,0,0,0,0,0,0,0,0,0	1580	5140	8E86	046A	0486	8E40	5180	1504
1	$0,2^1$,0,0,0,0,0,0,0,0,0,0,0,0,0,0	EB40	C920	4743	0235	0243	4720	C940	EB02
0	$0,2^0$,0,0,0,0,0,0,0,0,0,0,0,0,0,0	9420	8510	C2C0	01FB	01C0	C210	8520	9401

• строят расширенную схему РСЛОС, причем расширенный РСЛОС имеет r=8 ячеек, значение каждой представляет собой 16-разрядное число;

• состояние расширенного РСЛОС можно представить как
(Q_7, Q_6, …, Q_2, Q_1, Q_0)

• определяют значение $f(H_j)$ функции обратной связи для каждой H_j по формуле

$$f(H_j) = \bigoplus_{u=0}^{15} w_{7,u} \cdot H_{j,u},$$

где j=7, …, 1, 0,

$w_{7,u} \in GF(2)$

u=0, 1, …, 15 - биты ячейки Q_7 расширенного РСЛОС.

Это значит, что если u-й бит ячейки Q_{15} равен единице, то соответствующая строка $H_{j,u}$ участвует в процессе выработки значения функций обратной связи.

Прямое линейное преобразование L есть r=8 тактов работы расширенного РСЛОС, при этом 8 тактов работы расширенного РСЛОС требуют

$r \cdot n \cdot k = \frac{m}{k} \cdot n \cdot k = m \cdot n = 16 \cdot 8 = 128$ операций проверки "true - false"

и

$r \cdot r \cdot n \cdot k = \frac{m}{k} \cdot \frac{m}{k} \cdot n \cdot k = \frac{m^2 \cdot n}{k} = \frac{16^2 \cdot 8}{2} = 1024$ операций сложения по модулю 2 двух 16-разрядных чисел.

Объем необходимой памяти равен

$$r \cdot (n \cdot k) \cdot (n \cdot k) = \frac{m}{k} \cdot (n \cdot k)^2 = m \cdot n^2 \cdot k = 16 \cdot 8^2 \cdot 2 = 2048 \text{ бит} = 256 \text{ байт}$$

для сохранения 8 таблиц H_j, j=7, …, 0.

Полученная схема представлена на фиг. 12.

Порядок реализации обратного линейного преобразования L^{-1} на 16-разрядных процессорах выполняется аналогичным образом с использованием РСЛОС (фиг. 2) для формирования 8 таблиц H_j, j=7, …, 0. За счет симметричности внешнего неприводимого полинома h(y) для рассмотренного линейного преобразования можно использовать те же 8 таблиц H_j, j=7, …, 0 и для реализации прямого преобразования, и для обратного ему.

Аналогично, при k=4 и k=8 имеется возможность реализации прямого линейного преобразования L и обратного ему на 32- и 64-разрядных процессорах (фиг. 13 и 14 соответственно).

Результаты расчетов численных значений, характерных для реализации предложенного способа с использованием РСЛОС типа Галуа, представлены в табл. 3.

Таблица 3

Результаты расчетов численных значений, характерных для реализации предложенного способа с использованием РСЛОС типа Галуа

Разрядность используемого процессора	Количество необходимых проверок “true-false”	Размер требуемой памяти, байт	Количество операций сложения по модулю 2 двух ячеек памяти	Требуемое число тактов процессора
8	128	128	2048	16
16	128	256	1024	8
32	128	512	512	4
64	128	1024	256	2

Сравнительный анализ значений, приведенных в табл. 3, показывает, что предложенный способ позволяет осуществлять выбор взаимосвязанных характеристик вычислительной системы.

Так, если имеется 8-разрядный процессор, то для реализации заданного линейного преобразования потребуется минимальный объем памяти 128 байт и 16 операций сдвига шестнадцати байт.

Если же в распоряжении имеется более мощный 64-разрядный процессор, то для реализации заданного линейного преобразования потребуется 1024 байт памяти и всего 2 операции сдвига двух 64-разрядных слов.

В результате, использование предложенного способа позволяет также предоставить разработчику дополнительные возможности при проектировании прикладной программы или аппаратного узла вычислительной системы, реализующей линейное преобразование, и учитывать возникающие в практике требования.

Источники информации

1. Европейская заявка №1514174, приоритет от 04.06.2003 г.

2. Патент США №5946473, приоритет от 17.06.1997 г.

3. Nicolay Borisenko, Nguyen Van Long, Alexey Bulygin. Algorithm design software and hardware implementation of large size linear mapping. 2nd Workshop on Current Trends in Cryptology (CTCrypt 2013) June 23-25, 2013, Ekaterinburg, Russia. Pre-proceedings, pp. 192-205;

4. Mikhail Borodin, Andrey Rybkin, Alexey Urivskiy. High-Speed Software Implementation of the Prospective 128-bit Block Cipher and Streebog Hash-Function, 3rd Workshop on Current Trends in Cryptology (CTCrypt 2014) June 5-6, 2014, Moscow, Russia. Pre-proceedings, pp. 189-197;

5. Кузьмин А.С., Нечаев А.А., Линейные рекуррентные последовательности над кольцами Галуа, Алгебра и логика, 3:2 (1995), с. 169-189.

6. Коусело Е., Гонсалес С., Марков В.Т., Нечаев А.А. Рекурсивные МДР-коды и рекурсивно дифференцируемые квазигруппы. Дискретная математика, том 10, выпуск 2, 1998, с. 3-29.

Формула изобретения

1. Способ линейного преобразования сообщения S, представленного в двоичном виде, заключающийся в том, что

- задают разрядность W процессора вычислительной системы (размер машинного слова), равную целочисленной степени числа 2;
- задают доступный объем памяти вычислительной системы M бит;
- задают размер s сообщения S, причем s кратно W;
- задают значение разрядности n регистра сдвига с линейной обратной связью (РСЛОС) по схеме Галуа, причем выполняется соотношение

$$n = \frac{W}{2^N},$$

где $N \in 0, 1, 2, \ldots$;

- формируют РСЛОС по схеме Галуа со следующими параметрами:

внутренний примитивный полином

$$f(x) = x^n \oplus \sum_{i=0}^{n-1} a_i x^i,$$

$$a_i \in GF(2)$$

внешний полином

$$h(y) = y^m \oplus \sum_{i=0}^{m-1} h_i y^i,$$

где $m = \frac{s}{n}$ - количество ячеек РСЛОС,

причем $h_i \in GF(2^n)$,

исходное состояние ячеек РСЛОС q_i образует вектор

$$X = (q_{m-1}, q_{m-2}, \ldots, q_2, q_1, q_0),$$

причем $q_i \in GF(2^n),\ 0 \le i \le m-1$

выходное состояние ячеек РСЛОС q'_i за один такт работы образует вектор

$$Y = \left(q'_{m-1}, q'_{m-2}, \ldots, q'_2, q'_1, q'_0 \right),$$

причем $q'_i \in GF\left(2^n\right)$, $0 \le i \le m-1$,

где $q'_i = h_i \cdot q_{m-1} \oplus q_{i-1}$,

$q'_0 = h_0 \cdot q_{m-1}$;

- определяют все делители числа m в виде значений p_0, p_1, …, p_d, причем $p_0<p_1<\ldots p_d$;

- выбирают максимально возможный делитель p из соотношения

$$p \le \frac{M}{sn};$$

- модифицируют РСЛОС, выполняя следующие действия:

- вычисляют R матриц H_r, причем r=(R-1), …, 0, размерностью n×k строк, каждая из которых имеет длину n×k бит, выполняя следующие действия:

- вычисляют

k=p;

- вычисляют

$$R = \frac{m}{k};$$

где R - количество матриц H;

- вычисляют

j=m-k;

- вычисляют

t=0;

- (A1) если не выполняется соотношение

j≤m-1,

то переходят к выполнению этапа A3;

- вычисляют

l=0;

- (A2) если не выполняется соотношение

l<n,

то вычисляют

j=j+1,

t=t+1,

переходят к выполнению этапа A1;

- устанавливают исходное состояние РСЛОС

$X=(q_{m-1}, \ldots, q_1, q_0)$,

$$q_i = \begin{cases} 0, 0 \le i \le j-1 \text{ или } j+1 \le i \le m-1 \\ 2^l, i = j, \end{cases}, \quad 0 \le i \le m-1,$$

где $q_i \in GF(2^n)$;

- вычисляют после k тактов работы для каждого исходного состояния новое состояние РСЛОС

$$Y = \left(q'_{m-1}, \ldots, q'_1, q'_0\right),$$

где $q'_i \in GF(2^n)$, $0 \le i \le m-1$;

- вычисляют t-е значения для всех матриц H_i, i=r-1…0 путем конкатенации k значений ячеек q′

$$H_{r,t} = q'_{kr+k-1} \| \dots \| q'_{kr},$$

причем $0 \le r \le R-1$;

- вычисляют

l=l+1,

переходят к выполнению этапа А2;

- (А3) записывают в ячейки модифицированного РСЛОС блоки исходного сообщения S, причем исходное состояние ячеек модифицированного РСЛОС q_i образует вектор

$$X' = (Q_{R-1}, \dots, Q_1, Q_0),$$

где $Q_r = q_{kr+k-1} \| \dots \| q_{kr},$

причем $0 \le r \le R-1$,

- осуществляют R тактов работы модифицированного РСЛОС, выполняя на каждом такте следующие действия:

- вычисляют выходное состояние ячеек модифицированного РСЛОС Q'_i за один такт работы, образующие вектор

$$Y' = (Q'_{R-1}, \dots, Q'_1, Q'_0),$$

каждое значение Q'_i которого вычисляется по формуле

$$Q'_i = f(H_i) \oplus Q_{i-1}$$

для каждого $i = R-1, \dots, 1$,

причем

$$Q'_0 = f(H_0),$$

где $f(H_r) = \bigoplus_{j=0}^{W-1} z_{R-1,j} \cdot H_{r,j}$,

где $z_{R-1,j}$ - значение j-го бита вектора Q_{R-1},

причем $r = R-1, \dots, 1, 0$,

$j = 0, 1, \dots, W-1$,

$z_{R-1,j} \in GF(2)$;

- получают после R тактов работы РСЛОС линейное преобразование блоков s сообщения S;

- считывают из ячеек модифицированного РСЛОС блоки s линейно преобразованного сообщения S;

- объединяют блоки и получают линейно преобразованное сообщение S.

2. Способ линейного преобразования сообщения S, представленного в двоичном виде, заключающийся в том, что

- задают разрядность W процессора вычислительной системы (размер машинного слова), равную целочисленной степени числа 2;

- задают доступный объем памяти вычислительной системы M бит;

- задают размер s сообщения S, причем s кратно W;

- задают значение разрядности n регистра сдвига с линейной обратной связью (РСЛОС) по схеме Фибоначчи, причем выполняется соотношение

$$n = \frac{W}{2^N},$$

где $N \in 0, 1, 2, \ldots$;

- формируют РСЛОС по схеме Фибоначчи со следующими параметрами:

внутренний примитивный полином

$$f(x) = x^n \oplus \sum_{i=0}^{n-1} a_i x^i,$$

$$a_i \in GF(2)$$

внешний полином

$$h(y) = y^m \oplus \sum_{i=0}^{m-1} h_i y^i,$$

где $m = \frac{s}{n}$ - количество ячеек РСЛОС,

причем $h_i \in GF(2^n)$,

исходное состояние ячеек РСЛОС q_i образует вектор

$$X = (q_{m-1}, q_{m-2}, \ldots, q_2, q_1, q_0),$$

причем $q_i \in GF(2^n)$, $0 \le i \le m-1$,

выходное состояние ячеек РСЛОС q'_i за один такт работы образует вектор

$$Y = (q'_{m-1}, q'_{m-2}, \ldots, q'_2, q'_1, q'_0),$$

причем $q'_i \in GF(2^n)$, $0 \le i \le m-1$,

где $q'_i = q_{i+1}$,

для каждого $i = 0, \ldots, m-2$

$$q'_{m-1} = \bigoplus_{i=0}^{m-1} h_i \cdot q_i,$$

определяют все делители числа m в виде значений $p_0, p_1, \ldots, p_d$, причем $p_0<p_1<\ldots p_d$,

выбирают максимально возможный делитель p из соотношения

$$p \le \frac{M}{sn};$$

- модифицируют РСЛОС, выполняя следующие действия:

- вычисляют R матриц H_r, причем r = (R-1), …, 0, размерностью n×k строк, каждая из которых имеет длину n×k бит, выполняя следующие действия:

- вычисляют

k=p;

- вычисляют

$$R = \frac{m}{k},$$

где R - количество матриц H_r;

- вычисляют

r = 0;

- (A5) если не выполняется соотношение

r<R,

то переходят к выполнению этапа A7;

- вычисляют

j=0;

- (A6) если не выполняется соотношение

j<k,

то вычисляют

r = r+1,

переходят к выполнению этапа A5;

- вычисляют

l=0;

- если не выполняется соотношение

l<n,

то вычисляют

j=j+1,

переходят к выполнению этапа A6;

- устанавливают исходное состояние РСЛОС

$$X = (q_{m-1}, q_{m-2}, ..., q_1, q_0),$$

где $q_i, q'_i \in GF(2^n), 0 \le i \le m-1,$

$$q_i = \begin{cases} 2^l, i = rk + j, \\ 0, \text{ иначе} \end{cases}, 0 \le i \le m-1;$$

- вычисляют после k тактов работы для каждого исходного состояния новое состояние РСЛОС

$$Y = (q'_{m-1}, q'_{m-2}, ..., q'_1, q'_0);$$

- вычисляют (jk+l)-е значение для матрицы H_r путем конкатенации k значений ячеек $q'_{m-1}, q'_{m-2}, \cdots, q'_{m-k}$

$$H_{r,jk+l} = q'_{m-1} \| \cdots \| q'_{m-k};$$

- вычисляют

l=l+1,

переходят к выполнению этапа A6;

- (A7) записывают в ячейки модифицированного РСЛОС блоки s исходного сообщения S, причем исходное состояние ячеек модифицированного РСЛОС q_i образует вектор

$$X' = (Q_{R-1}, ..., Q_1, Q_0),$$

где $Q_r = q_{kr+k-1} \| \cdots \| q_{kr}$,

причем $0 \le r \le R-1$;

- осуществляют R тактов работы модифицированного РСЛОС, выполняя на каждом такте следующие действия:

- вычисляют выходное состояние ячеек модифицированного РСЛОС Q'_i за один такт работы, образующие вектор

$$Y' = (Q'_{R-1}, ..., Q'_1, Q'_0),$$

каждое значение Q'_i которого вычисляется по формуле

$Q'_i = Q_{i+1}$

для каждого $i = 0, ..., R-2$,

а значение Q'_{R-1} вычисляется по соотношению

$$Q'_{R-1} = \bigoplus_{r=0}^{R-1} f(H_r),$$

где $f(H_r) = \bigoplus_{j=0}^{W-1} z_{r,j} \cdot H_{r,j}$,

где $z_{r,j}$ - значение j-го бита ячейки Q_r,

причем $r = R-1, ..., 1, 0$,

$j = 0, 1, ..., W-1$,

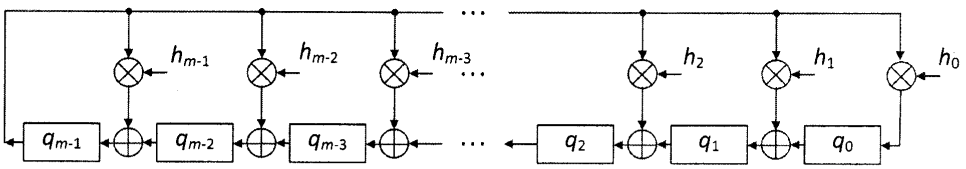
$z_{r,j} \in GF(2)$;

- получают после R тактов работы РСЛОС линейное преобразование блоков s сообщения S;

- считывают из ячеек модифицированного РСЛОС блоки s линейно преобразованного сообщения S;

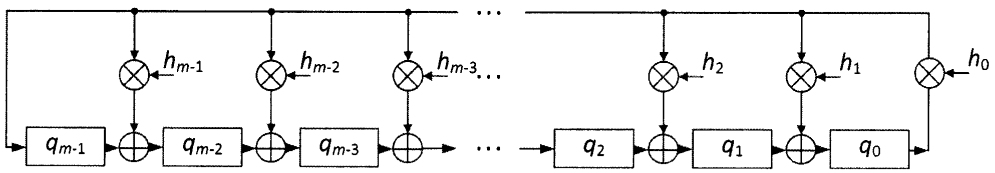
- объединяют блоки и получают линейно преобразованное сообщение S.

Способ линейного преобразования (варианты)



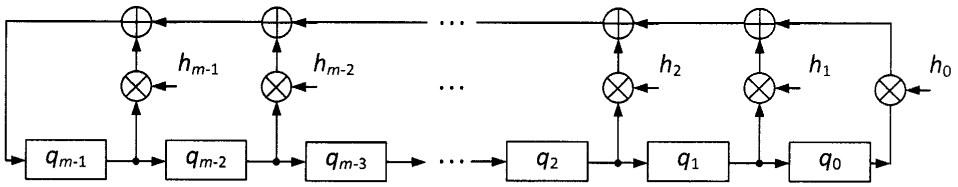
Фиг. 1

Способ линейного преобразования (варианты)



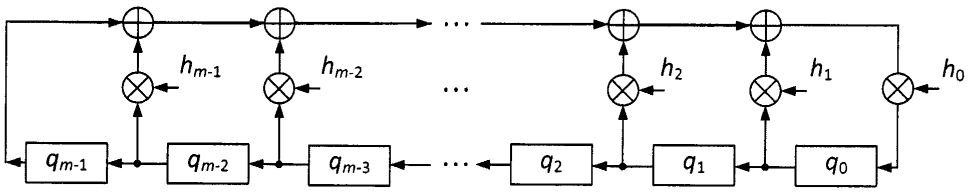
Фиг. 2

Способ линейного преобразования (варианты)



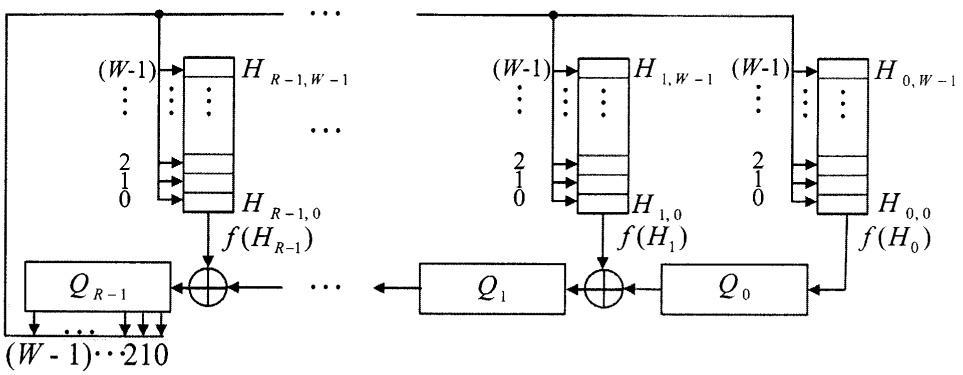
Фиг. 3

Способ линейного преобразования (варианты)



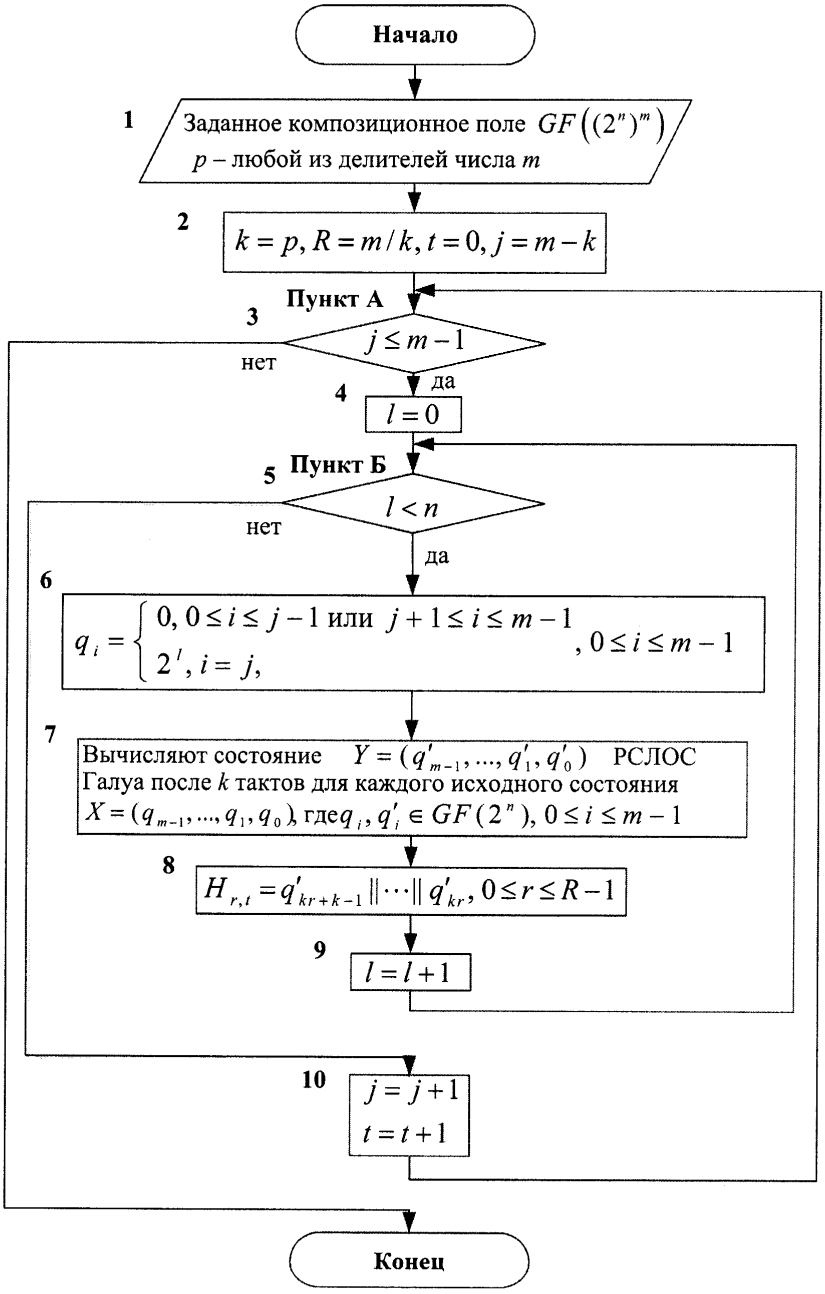
Фиг. 4

Способ линейного преобразования (варианты)



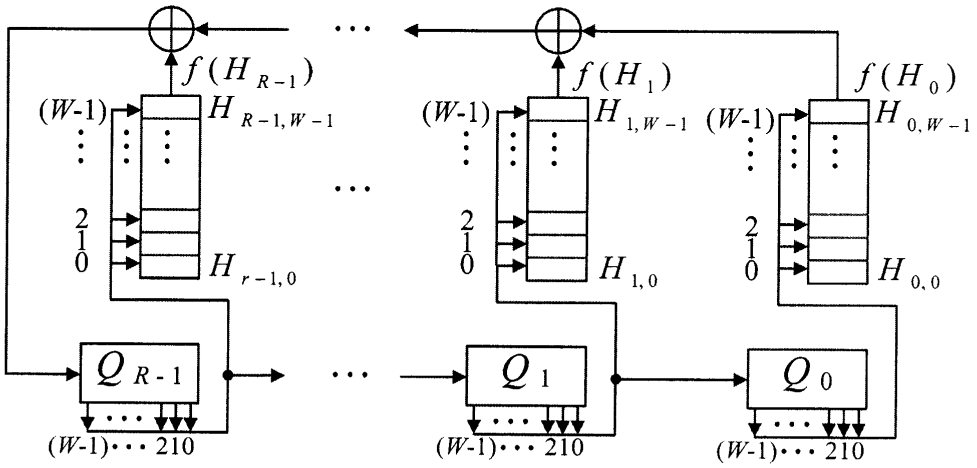
Фиг. 5

Способ линейного преобразования (варианты)



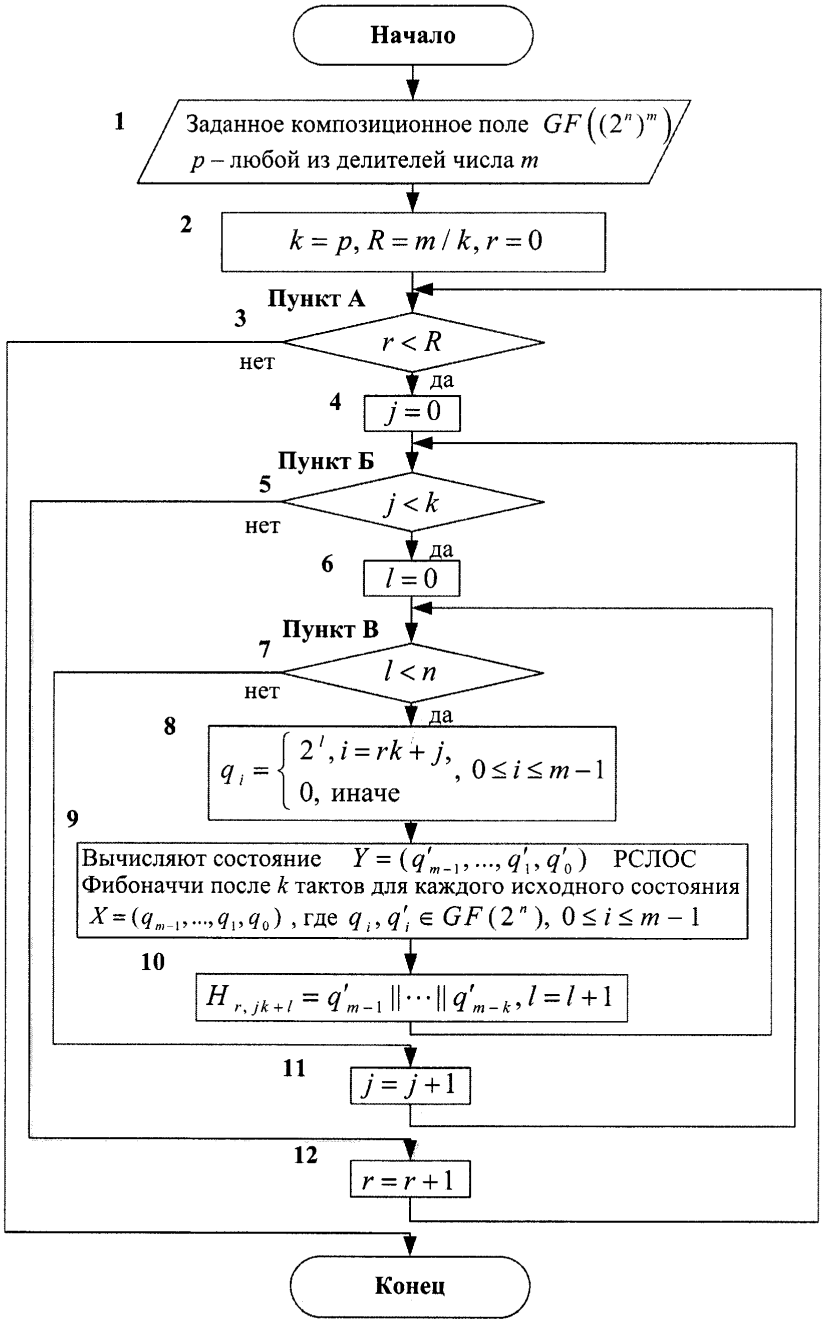
Фиг. 6

Способ линейного преобразования (варианты)



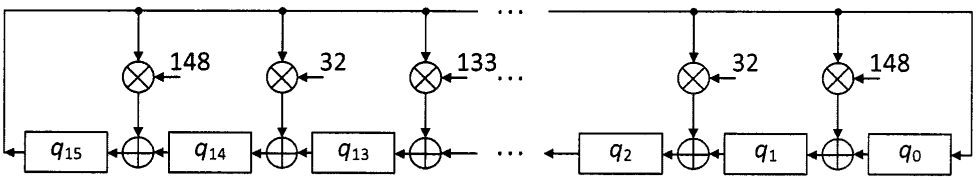
Фиг. 7

Способ линейного преобразования (варианты)



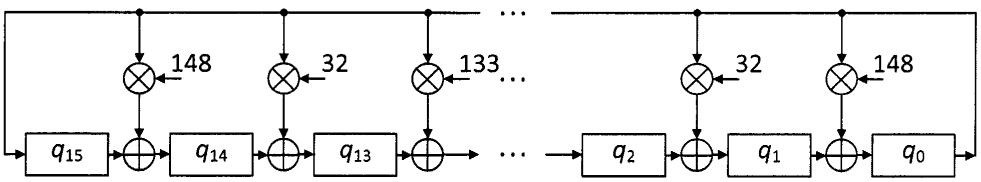
Фиг. 8

Способ линейного преобразования (варианты)



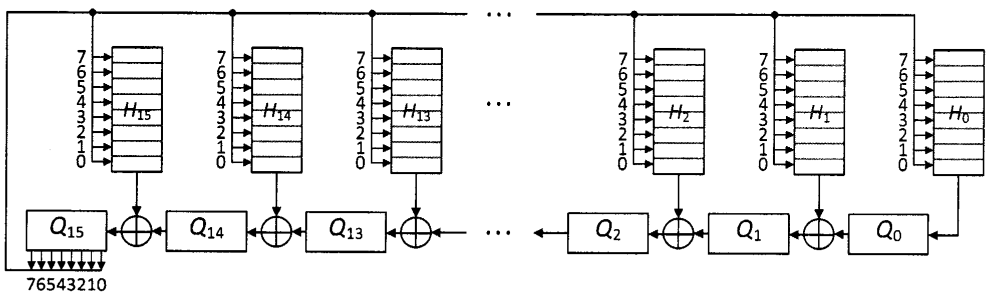
Фиг. 9

Способ линейного преобразования (варианты)



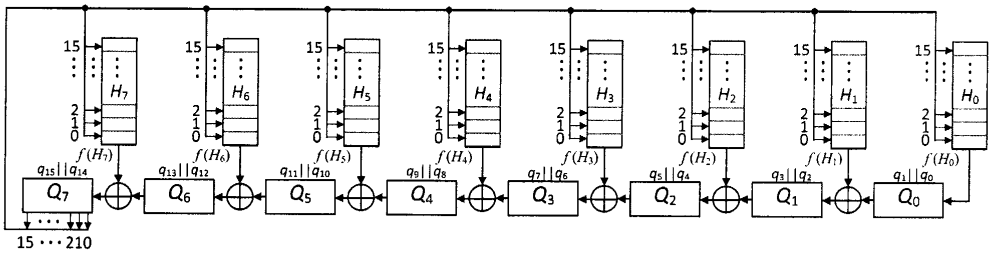
Фиг. 10

Способ линейного преобразования (варианты)



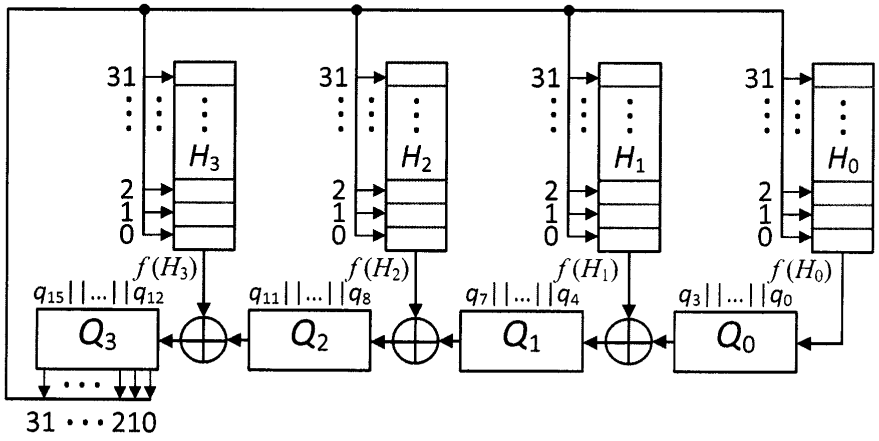
Фиг. 11

Способ линейного преобразования (варианты)



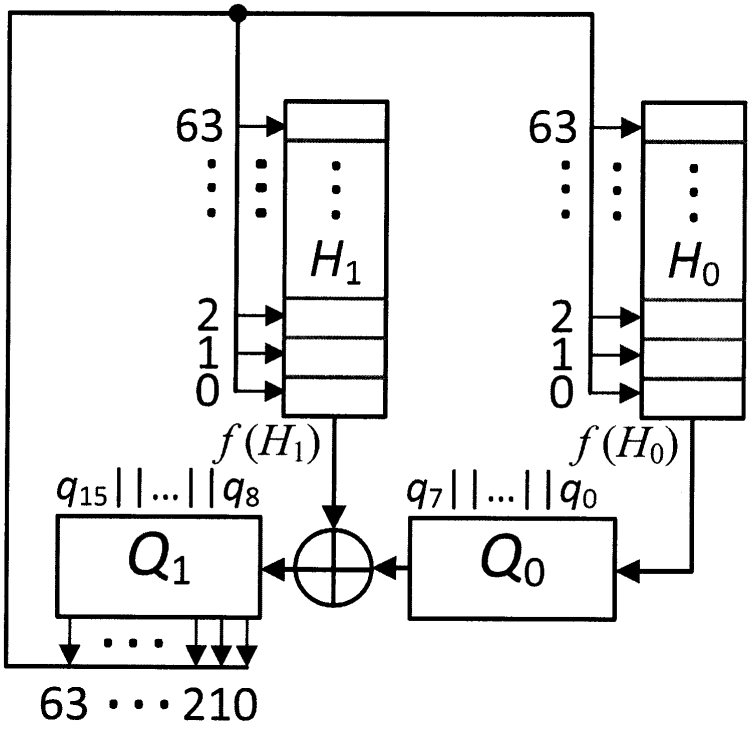
Фиг. 12

Способ линейного преобразования (варианты)



Фиг. 13

Способ линейного преобразования (варианты)



Фиг. 14