ГОСТ 28147-89

Группа П85

ГОСУДАРСТВЕННЫЙ СТАНДАРТ СОЮЗА ССР

СИСТЕМЫ ОБРАБОТКИ ИНФОРМАЦИИ. ЗАЩИТА КРИПТОГРАФИЧЕСКАЯ

Алгоритм криптографического преобразования

ОКП 40 4000

Дата введения 1990-07-01

ИНФОРМАЦИОННЫЕ ДАННЫЕ

1. УТВЕРЖДЕН И ВВЕДЕН В ДЕЙСТВИЕ Постановлением Государственного комитета СССР по стандартам от 02.06.89 N 1409

2. ВВЕДЕН ВПЕРВЫЕ

3. ССЫЛОЧНЫЕ НОРМАТИВНО-ТЕХНИЧЕСКИЕ ДОКУМЕНТЫ

Обозначение НТД, на который дана ссылка	Номер пункта
ГОСТ 15971-90	Приложение 1
ГОСТ 17657-79	Приложение 1
ГОСТ 19781-90	Приложение 1

4. ПЕРЕИЗДАНИЕ, апрель 1996 г.

Настоящий стандарт устанавливает единый алгоритм криптографического преобразования для систем обработки информации в сетях электронных вычислительных машин (ЭВМ), отдельных вычислительных комплексах и ЭВМ, который определяет правила шифрования данных и выработки имитовставки.

Алгоритм криптографического преобразования предназначен для аппаратной или программной реализации, удовлетворяет криптографическим требованиям и по своим возможностям не накладывает ограничений на степень секретности защищаемой информации.

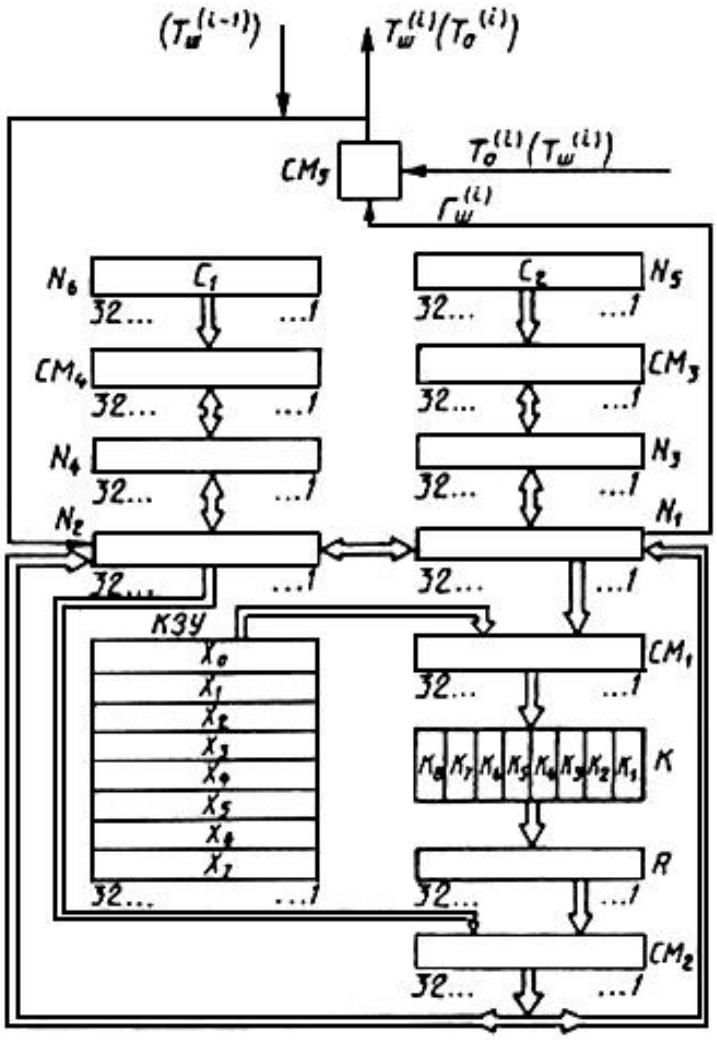
Стандарт обязателен для организаций, предприятий и учреждений, применяющих криптографическую защиту данных, хранимых и передаваемых в сетях ЭВМ, в отдельных вычислительных комплексах или в ЭВМ.

Термины, применяемые в настоящем стандарте, и их определения приведены в приложении 1.

1. СТРУКТУРНАЯ СХЕМА АЛГОРИТМА КРИПТОГРАФИЧЕСКОГО ПРЕОБРАЗОВАНИЯ

1.1. Структурная схема алгоритма криптографического преобразования (криптосхема) содержит (см. черт.1):

Черт.1. Структурная схема алгоритма криптографического преобразования



Черт.1

ключевое запоминающее устройство (КЗУ) на 256 бит, состоящее из восьми 32-разрядных накопителей (X_0, X_1, X_2, X_3, X_4, X_5, X_6, X_7);

четыре 32-разрядных накопителя (N_1, N_2, N_3, N_4);

два 32-разрядных накопителя (N_5, N_6) с записанными в них постоянными заполнениями C_2, C_1;

два 32-разрядных сумматора по модулю 2^{32} (CM_1, CM_3);

32-разрядный сумматор поразрядного суммирования по модулю 2 (CM_2);

32-разрядный сумматор по модулю (2^{32}-1) (CM_4);

сумматор по модулю 2(CM_5), ограничение на разрядность сумматора CM_5 не накладывается;

блок подстановки (K);

регистр циклического сдвига на одиннадцать шагов в сторону с

таршего разряда (R).

1.2. Блок подстановки K состоит из восьми узлов замены K_1, K_2, K_3, K_4, K_5, K_6, K_7, K_8 с памятью на 64 бита каждый. Поступающий на блок подстановки 32-разрядный вектор разбивается на восемь последовательно идущих 4-разрядных векторов, каждый из которых преобразуется в 4-разрядный вектор соответствующим узлом замены, представляющим собой таблицу из шестнадцати строк, содержащих по четыре бита заполнения в строке. Входной вектор определяет адрес строки в таблице, заполнение данной строки является выходным вектором. Затем 4-разрядные выходные векторы последовательно объединяются в 32-разрядный вект

1.3. При сложении и циклическом сдвиге двоичных векторов старшими разрядами считаются разряды накопителей с большими номерами.

1.4. При записи ключа (W_1, W_2 ..., W_{256}), $W_q \in \{0,1\}$, $q = 1 \div 256$, в КЗУ значение W_1 вводится в 1-й разряд накопителя X_0, значение W_2 вводится во 2-й разряд накопителя X_0, ... , значение W_{32} вводится в 32-й разряд накопителя X_0; значение W_{33} вводится в 1-й разряд накопителя X_1, значение W_{34} вводится во 2-й разряд накопителя X_1, ... , значение W_{64} вводится в 32-й разряд накопителя X_1; значение W_{65} вводится в 1-й разряд накопителя X_2 и т.д., значение W_{256} вводится в 32-й разр

яд накопителя X_7.

1.5. При перезаписи информации содержимое p-го разряда одного накопителя (сумматора) переписывается в p-й разряд другого накопителя (сумматора).

1.6. Значения постоянных заполнений C_1, C_2 (констант) накопителей N_6, N_5 приведены в приложении 2.

1.7. Ключи, определяющие заполнения КЗУ и таблиц блока подстановки K, являются секретными элементами и поставляются в установленном порядке.

Заполнение таблиц блока подстановки K является долговременным ключевым элементом, общим для сети ЭВМ.

Организация различных видов связи достигается построением соответствующей ключевой системы. При этом может быть использована возможность выработки ключей (заполнений КЗУ) в режиме простой замены и зашифрования их в режиме простой замены с обеспечением имитозащиты для передачи по каналам связи или хранения в памяти ЭВМ.

1.8. В криптосхеме предусмотрены четыре вида работы:

зашифрование (расшифрование) данных в режиме простой замены;

зашифрование (расшифрование) данных в режиме гаммирования;

зашифрование (расшифрование) данных в режиме гаммирования с обратной связью;

режим выработки имитовставки.

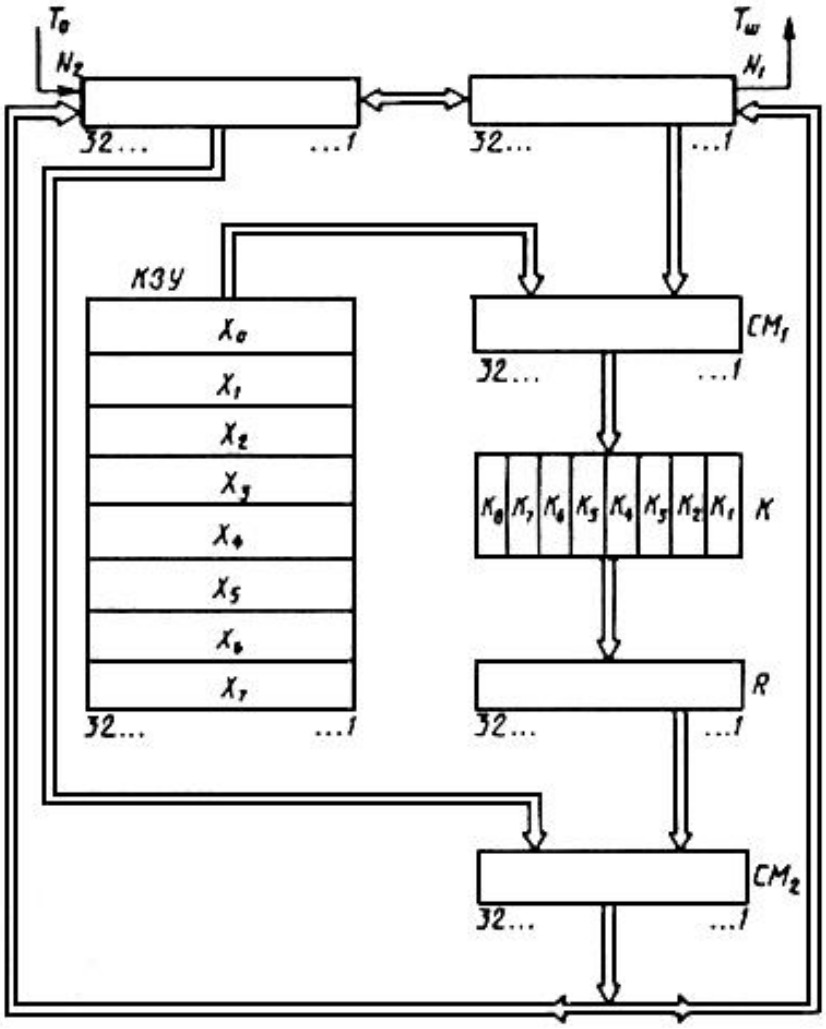
Схемы программной реализации алгоритма криптографического преобразования приведены в приложении 3.

2. РЕЖИМ ПРОСТОЙ ЗАМЕНЫ

2.1. Зашифрование открытых данных в режиме простой замены

2.1.1. Криптосхема, реализующая алгоритм зашифрования в режиме простой замены, должна иметь вид, указанный на черт.2.

Черт.2. Криптосхема, реализующая алгоритм зашифрования в режиме простой замены



Черт.2

Открытые данные, подлежащие зашифрованию, разбивают на блоки по 64 бита в каждом. Ввод любого блока T_0 = (a_1(0), a_2 (0). …, a_{31} (0), a_{32} (0), b_1 (0), b_2(0) …, b_{32} (0)) двоичной информации в накопители N_1 и N_2 производится так, что значение a_1 (0) вводится в 1-й разряд N_1, значение a_2 (0) вводится во 2-й разряд N_1 и т.д., значение a_{32}(0) вводится в 32-й разряд N_1; значение b_1 (0) вводится в 1-й разряд N_2, значение b_2 (0) вводится во 2-й разряд N_2 и т.д., значение b_{32}(0) вводится в 32-й разряд N_2. В результате получают состояние (a_{32} (0), a_{31} (0), …, a_2(0), a_1(0)) накопителя N_1 и состояние (b_{32} (0), b_{31}(0),

..., b_1 (0)) накопителя N_2.

2.1.2. В КЗУ вводятся 256 бит ключа. Содержимое восьми 32-разрядных накопителей X_0, X_1 ... , X_7 имеет вид:

$$X_0 = (W_{32}, W_{31}, \ldots, W_2, W_1)$$

$X_1 = (W_{64}, W_{63}, \ldots, W_{34}, W_{33})$

$X_7 = (W_{256}, W_{255}, \ldots, W_{226}, W_{225})$

2.1.3. Алгоритм зашифрования 64-разрядного блока открытых данных в режиме простой замены состоит из 32 циклов.

В первом цикле начальное заполнение накопителя N_1 суммируется по модулю 2^{32} в сумматоре CM_1 с заполнением накопителя X_0, при этом заполнение накопителя N_1 сохраняется.

Результат суммирования преобразуется в блоке подстановки K и полученный вектор поступает на вход регистра R, где циклически сдвигается на одиннадцать шагов в сторону старших разрядов. Результат сдвига суммируется поразрядно по модулю 2 в сумматоре CM_2 с 32-разрядным заполнением накопителя N_2. Полученный в CM_2 результат записывается в N_1, при этом старое заполнение N_1 переписывается в N_2. Первый цикл заканчивается.

Последующие циклы осуществляются аналогично, при этом во 2-м цикле из КЗУ считывается заполнение X_1 в 3-м цикле из КЗУ считывается заполнение X_2 и т.д., в 8-м цикле из КЗУ считывается заполнение X_7. В циклах с 9-го по 16-й, а также в циклах с 17-го по 24-й заполнения из КЗУ считываются в том же порядке:

$X_0, X_1, X_2, X_3, X_4, X_5, X_6, X_7.$

В последних восьми циклах с 25-го по 32-й порядок считывания заполнений КЗУ обратный:

$X_7, X_6, X_5, X_4, X_3, X_2, X_1, X_0.$

Таким образом, при зашифровании в 32 циклах осуществляется следующий порядок выбора заполнений накопителей:

$X_0, X_1, X_2, X_3, X_4, X_5, X_6, X_7, \quad X_0, X_1, X_2, X_3, X_4, X_5, X_6, X_7,$

$X_0, X_1, X_2, X_3, X_4, X_5, X_6, X_7, \quad X_7, X_6, X_5, X_4, X_3, X_2, X_1, X_0.$

В 32 цикле результат из сумматора CM_2 вводится в накопитель N_2, а в накопителе N_1 сохраняется старое заполнение.

Полученные после 32-го цикла зашифрования заполнения накопителей N_1 и N_2 являются блоком зашифрованных данных, соответствующим блоку открытых данных.

2.1.4. Уравнения зашифрования в режиме простой замены имеют вид:

$$\begin{cases} a(j) = (a(j-1) \boxplus X_{(j-1)\ (\mathrm{mod}8)})KR \oplus b(j-1) \\ b(j) = a(j-1) \end{cases}$$

при j = 1÷24;

$$\begin{cases} a(j) = (a(j-1) \boxplus X_{(32-j)})KR \oplus b(j-1) \\ b(j) = a(j-1) \end{cases}$$

при j =25 ÷31;

$$a(32) = a(31)$$

$$b(32) = (a(31) \boxplus X_0)KR \oplus b(31)$$

при j = 32,

где $a(0)$ = ($a_{32}(0)$, $a_{31}(0)$, … , $a_1(0)$) - начальное заполнение N_1 перед первым циклом зашифрования;

$b(0)$ = ($b_{32}(0)$, $b_{31}(0)$, … , $b_1(0)$) - начальное заполнение N_2 перед первым циклом зашифрования;

$a(j)$ = ($a_{32}(j)$, $a_{31}(j)$, … , $a_1(j)$) - заполнение N_1 после j-го цикла зашифрования;

$b(j)$ = ($b_{32}(j)$, $b_{31}(j)$, … , $b_1(j)$) - заполнение N_2 после j-го цикла зашифрования, j = 1÷32.

Знак $\oplus$ означает поразрядное суммирование 32-разрядных векторов по модулю 2.

Знак $\boxplus$ означает суммирование 32-разрядных векторов по модулю 2^{32}. Правила суммирования по модулю 2^{32} приведены в приложении 4;

R - операция циклического сдвига на одиннадцать шагов в сторону старших разрядов, т.е.

$$R(r_{32}, r_{31}, r_{30}, r_{29}, r_{28}, r_{27}, r_{26}, r_{25}, r_{24}, r_{23}, r_{22}, r_{21}, r_{20}, \ldots, r_2, r_1) =$$

$$= (r_{21}, r_{20}, \ldots, r_2, r_1, r_{32}, r_{31}, r_{30}, r_{29}, r_{28}, r_{27}, r_{26}, r_{25}, r_{24}, r_{23}, r_{22}).$$

2.1.5. 64-разрядный блок зашифрованных данных $T_{ш}$ выводится из накопителей N_1, N_2 в следующем порядке: из 1-го, 2-го, ... , 32-го разрядов накопителя N_1 затем из 1-го, 2-го, ... , 32-го разрядов накопителя N_2, т.е.

$$T_{ш} = (a_1(32), a_2(32), \ldots, a_{32}(32), b_1(32), b_2(32), \ldots, b_{32}(32)).$$

Остальные блоки открытых данных в режиме простой замены зашифровываются аналогично.

2.2. Расшифрование зашифрованных данных в режиме простой замены

2.2.1. Криптосхема, реализующая алгоритм расшифрования в режиме простой замены, имеет тот же вид (см. черт.2), что и при зашифровании. В КЗУ вводятся 256 бит того же ключа, на котором осуществлялось зашифрование. Зашифрованные данные, подлежащие расшифрованию, разбиты на блоки по 64 бита в каждом. Ввод любого блока

$$T_{\text{ш}} = (a_1(32), a_2(32), \ldots, a_{32}(32), b_1(32), b_2(32), \ldots, b_{32}(32))$$

в накопители N_1 и N_2 производятся так, что значение $a_1(32)$ вводится в 1-й разряд N_1, значение $a_2(32)$ вводится во 2-й разряд N_1 и т.д., значение $a_{32}(32)$ вводится в 32-й разряд N_1; значение $b_1(32)$ вводится в 1-й разряд N_2 и т.д., значение $b_{32}(32)$ вводится в 32-й раз

ряд N_2.

2.2.2. Расшифрование осуществляется по тому же алгоритму, что и зашифрование открытых данных, с тем изменением, что заполнения накопителей X_0, X_1, ... , X_7 считываются из КЗУ в циклах расшифрования в следующем порядке:

$$X_0, X_1, X_2, X_3, X_4, X_5, X_6, X_7, \quad X_7, X_6, X_5, X_4, X_3, X_2, X_1, X_0,$$

$$X_7, X_6, X_5, X_4, X_3, X_2, X_1, X_0, \quad X_7, X_6, X_5, X_4, X_3, X_2, X_1, X_0.$$

2.2.3. Уравнения расшифрования имеют вид:

$$\begin{cases} a(32-j) = (a(32-j+1) \boxplus X_{j-1}) KR \oplus b(32-j+1) \\ b(32-j) = a(32-j+1) \end{cases}$$

при $j = 1 \div 8$;

$$\begin{cases} a(32-j) = (a(32-j+1) \boxplus X_{(32-j)(\mathrm{mod}8)}) KR \oplus b(32-j+1) \\ b(32-j) = a(32-j+1) \end{cases}$$

при $j = 9 \div 31$;

$$a(0) = a(1)$$

$$b(0) = (a(1) \boxplus X_0) KR \oplus b(1)$$

при j = 32.

2.2.4. Полученные после 32 циклов работы заполнения накопителей N_1 и N_2 составляют блок открытых данных.

$T_0 = (a_1(0), a_2(0), \dots, a_{32}(0), b_1(0), b_2(0), \dots, b_{32}(0))$, соответствующий блоку зашифрованных данных, при этом значение $a_1(0)$ блока T_0 соответствует содержимому 1-го разряда N_1, значение $a_2(0)$ соответствует содержимому 2-го разряда N_1 и т.д., значение $a_{32}(0)$ соответствует содержимому 32-го разряда N_1; значение $b_1(0)$ соответствует содержимому 1-го разряда N_2, значение $b_2(0)$ соответствует содержимому 2-го разряда N_2 и т.д., значение $b_{32}(0)$ соответствует содержимому 32-го разряда N_2.

Аналогично расшифровываются остальные блоки зашифр

ованных данных.

2.3. Алгоритм зашифрования в режиме простой замены 64-битового блока T_0 обозначается через A, т.е.

$$A(T_0) = A(a(0), b(0)) = (a(32), b(32)) = T_{\text{ш}}.$$

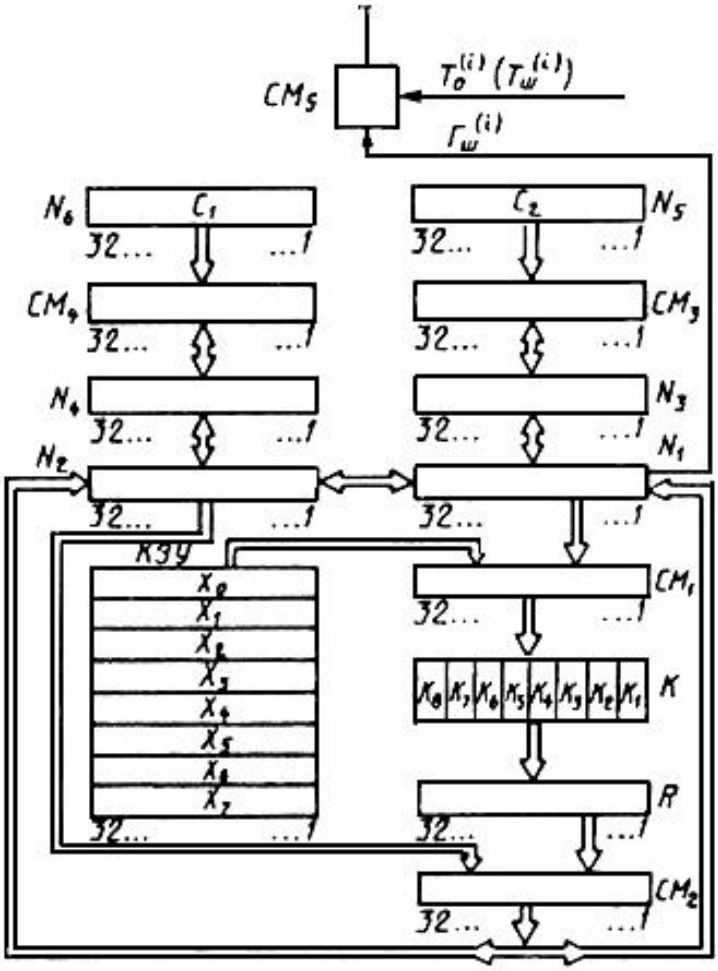
2.4. Режим простой замены допускается использовать для зашифрования (расшифрования) данных только в случаях, приведенных в п.1.7.

3. РЕЖИМ ГАММИРОВАНИЯ

3.1. Зашифрование открытых данных в режиме гаммирования

3.1.1. Криптосхема, реализующая алгоритм зашифрования в режиме гаммирования, имеет вид, указанный на черт.3.

Черт.3. Криптосхема, реализующая алгоритм зашифрования в режиме гаммирования



Черт.3

Открытые данные, разбитые на 64-разрядные блоки $T_0^{(1)}$, $T_0^{(2)}$ ... , $T_0^{(M-1)}$, $T_0^{(M)}$, зашифровываются в режиме гаммирования путем поразрядного суммирования по модулю 2 в сумматоре $СМ_5$ с гаммой шифра $Г_ш$, которая вырабатывается блоками по 64 бита, т.е.

$$Г_ш = (Г_ш^{(1)}, Г_ш^{(2)}, \dots, Г_ш^{(м-1)}, \dots, Г_ш^{(м)}),$$

где M - определяется объемом шифруемых данных.

$Г_ш^{(i)}$ - i-й 64-разрядный блок, $i = 1 \div M$, число двоичных разрядов в блоке $T_0^{(M)}$ может быть меньше 64, при этом неиспользованная для зашифрования часть гаммы шифра из блока $Г_ш^{(M)}$ отбрасывается

3.1.2. В КЗУ вводятся 256 бит ключа. В накопители N_1, N_2 вводится 64-разрядная двоичная последовательность (синхропосылка) $S = (S_1, S_2, \ldots, S_{64})$, являющаяся исходным заполнением этих накопителей для последующей выработки M блоков гаммы шифра. Синхропосылка вводится в N_1 и N_2 так, что значение S_1 вводится в 1-й разряд N_1, значение S_2 вводится во 2-й разряд N_1 и т.д., значение S_{32} вводится в 32-й разряд N_1; значение S_{33} вводится в 1-й разряд N_2, значение S_{34} вводится во 2-й разряд N_2 и т.д., значение S_{64} вводится в 32

-й разряд N_2.

3.1.3. Исходное заполнение накопителей N_1 и N_2 (синхропосылка S) зашифровывается в режиме простой замены в соответствии с требованиями п.2.1. Результат зашифрования $A(S) = Y_0$, Z_0) переписывается в 32-разрядные накопители N_3 и N_4, так, что заполнение N_1 переписывается в N_3, а заполнение N_2 переписывается

в N_4.

3.1.4. Заполнение накопителя N_4 суммируется по модулю (2^{32}-1) в сумматоре CM_4 с 32-разрядной константой C_1 из накопителя N_6, результат записывается в N_4. Правила суммирования по модулю (2^{32}-1) приведены в приложении 4. Заполнение накопителя N_3 суммируется по модулю 2^{32} в сумматоре CM_3 в 32-разрядной константой C_2 из накопителя N_5, результат записывается в N_3.

Заполнение N_3 переписывается в N_1, а заполнение N_4 переписывается в N_2, при этом заполнение N_3, N_4 сохраняется.

Заполнение N_1 и N_2 зашифровывается в режиме простой замены в соответствии с требованиями п.2.1. Полученное в результате зашифрования заполнение N_1, N_2 образует первый 64-разрядный блок гаммы шифра $T_{\text{ш}}^{(1)}$, который суммируется поразрядно по модулю 2 в сумматоре CM_5 с первым 64-разрядным блоком открытых данных

$$T_0^{(1)} = (t_1^{(1)}, t_2^{(1)}, \ldots, t_{63}^{(1)}, t_{64}^{(1)}).$$

В результате суммирования получается 64-разрядный блок зашифрованных данных

$$T_{\text{ш}}^{(1)} = (\tau_1^{(1)}, \tau_2^{(1)}, \ldots, \tau_{63}^{(1)}, \tau_{64}^{(1)}).$$

Значение $\tau_1^{(1)}$ блока $T_{\text{ш}}^{(1)}$ является результатом суммирования по модулю 2 в CM_5 значения $t_1^{(1)}$ из блока $T_0^{(1)}$ со значением 1-го разряда N_1, значение $\tau_2^{(1)}$ блока $T_{\text{ш}}^{(1)}$ является результатом суммирования по модулю 2 в CM_5 значения $t_2^{(1)}$ из блока $T_0^{(1)}$ со значением 2-го разряда N_1 и т.д., значение $\tau_{64}^{(1)}$ блока $T_{\text{ш}}^{(1)}$ является результатом суммирования по модулю 2 в CM_5 значения $t_{64}^{(1)}$ из блока $T_0^{(1)}$ со значением 32-г

о разряда N_2.

3.1.5. Для получения следующего 64-разрядного блока гаммы шифра $\Gamma_{\text{ш}}^{(2)}$ заполнение N_4 суммируется по модулю (2^{32}-1) в сумматоре CM_4 с константой C_1 из N_6, заполнение N_3 суммируется по модулю 2^{32} в сумматоре CM_3 с константой C_2 из N_5. Новое заполнение N_3 переписывается в N_1, а новое заполнение N_4 переписывается в N_2, при этом заполнение N_3 и N_4 сохраняется.

Заполнение N_1 и N_2 зашифровывается в режиме простой замены в соответствии с требованиями п.2.1. Полученное в результате зашифрования заполнение N_1, N_2 образует второй 64-разрядный блок гаммы шифра $\Gamma_{\text{ш}}^{(2)}$, который суммируется поразрядно по модулю 2 в сумматоре CM_5 со вторым блоком открытых данных $T_0^{(2)}$. Аналогично вырабатываются блоки гаммы шифра $\Gamma_{\text{ш}}^{(3)}$, $\Gamma_{\text{ш}}^{(4)}$ ... , $\Gamma_{\text{ш}}^{(\text{м})}$ и зашифровываются блоки открытых данных $T_0^{(3)}$, $T_0^{(4)}$ ... , $T_0^{(\text{м})}$. Если длина последнего M-го блока открытых данных $T_0^{(\text{м})}$ меньше 64 бит, то из последнего M-го блока гаммы шифра $\Gamma_{\text{ш}}^{(\text{м})}$ для зашифрования используется только соответствующее число разрядов гаммы шифра, оста

льные разряды отбрасываются.

3.1.6. В канал связи или память ЭВМ передаются синхропосылка S и блоки зашифрованных данных $T_{\text{ш}}^{(1)}$, $T_{\text{ш}}^{(2)}$, ... , $T_{\text{ш}}^{(\text{м})}$.

3.1.7. Уравнение зашифрования имеет вид:

$$T_{\text{ш}}^{(i)} = A(Y_{i-1} \boxplus C_2, Z_{i-1} \boxplus' C_1) \oplus T_0^{(i)} = \Gamma_{\text{ш}}^{i} \oplus T_0^{(i)},$$

$$i = 1 \ldots M$$

где $\boxplus'$ - означает суммирование 32-разрядных заполнений по модулю (2^{32}-1);

$\boxplus$ - поразрядное суммирование по модулю 2 двух заполнений;

Y_i - содержимое накопителя N_3 после зашифрования i-го блока открытых данных $T_0^{(i)}$;

Z_i - содержимое накопителя N_4 после зашифрования i-го блока открытых данных $T_0^{(i)}$;

$(Y_0, Z_0) = A(S)$.

3.2. Расшифрование зашифрованных данных в режиме гаммирования

3.2.1. При расшифровании криптосхема имеет тот же вид, что и при зашифровании (см. черт.3). В КЗУ вводятся 256 бит ключа, с помощью которого осуществлялось зашифрование данных $T_0^{(1)}$, $T_0^{(2)}$, ..., $T_0^{(м)}$. Синхропосылка S вводится в накопители N_1 и N_2 и аналогично пп.3.1.2-3.1.5 осуществляется процесс выработки M блоков гаммы шифра $Г_ш^{(1)}$, $Г_ш^{(2)}$, ... , $Г_ш^{(м)}$. Блоки зашифрованных данных $T_ш^{(1)}$, $T_ш^{(2)}$, ... , $T_ш^{(м)}$ суммируются поразрядно по модулю 2 в сумматоре $СМ_5$ с блоками гаммы шифра, в результате получаются блоки открытых данных $T_0^{(1)}$, $T_0^{(2)}$, ..., $T_0^{(м)}$, при этом $T_0^{(м)}$ может содержать меньше

64 разрядов.

3.2.2. Уравнение расшифрования имеет вид:

$$T_0^{(i)} = A(Y_{i-1} \boxplus C_2, Z_{i-1} \boxplus' C_1) \oplus T_ш^{(i)} = Г_ш^{(i)} \oplus T_ш^{(i)},$$

$$i = 1 \div M$$

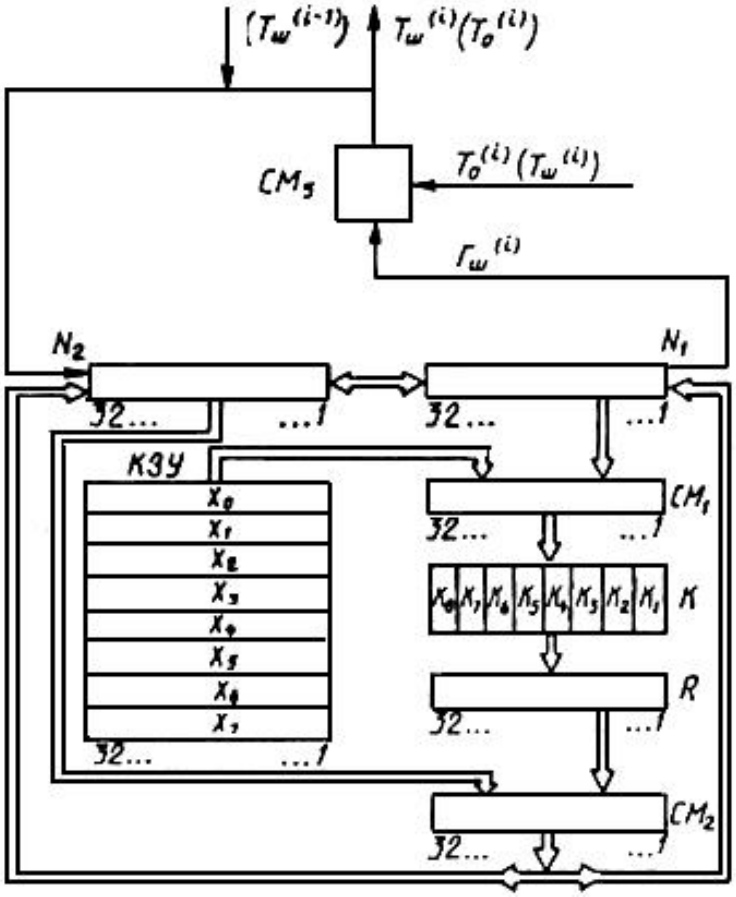
4. РЕЖИМ ГАММИРОВАНИЯ С ОБРАТНОЙ СВЯЗЬЮ

4.1. Зашифрование открытых данных в режиме гаммирования с обратной связью

4.1.1. Крипстосхема, реализующая алгоритм зашифрования в режиме гаммирования с обратной связью, имеет вид, указанный на черт.4.

Черт.4. Крипстосхема, реализующая алгоритм зашифрования в режиме гаммирования с обратной

связью



Черт.4

Открытые данные, разбитые на 64-разрядные блоки $T_0^{(1)}$, ..., $T_0^{(m)}$, зашифровываются в режиме гаммирования с обратной связью путем поразрядного суммирования по модулю 2 в сумматоре CM_5 с гаммой шифра $\Gamma_{ш}$, которая вырабатывается блоками по 64 бита, т.е. $\Gamma_{ш} = (\Gamma_{ш}^{(1)}, \Gamma_{ш}^{(2)}, \ldots, \Gamma_{ш}^{(m)})$, где M определяется объемом открытых данных, $\Gamma_{ш}^{(i)}$ - i-й 64-разрядный блок, $i = 1 \div M$. Число двоичных разрядов в блоке $T_0^{(m)}$ может быть м

еньше 64.

4.1.2. В КЗУ вводятся 256 бит ключа. Синхропосылка $S = (S_1, S_2, \ldots, S_{64})$ из 64 бит вводится в N_1 и N_2 аналогично п.3.1.2.

4.1.3. Исходное заполнение N_1 и N_2 зашифровывается в режиме простой замены в соответствии с требованиями п.2.1. Полученное в результате зашифрования заполнение N_1 и N_2 образует первый 64-разрядный блок гаммы шифра $\Gamma_{\text{ш}}^{(1)} = A(S)$, который суммируется поразрядно по модулю 2 в сумматоре CM_5 с первым 64-разрядным блоком открытых данных $T_0^{(1)} = (t_1^{(1)}, t_2^{(1)}, \ldots, t_{64}^{(1)})$.

В результате получается 64-разрядный блок зашифрованных данных

$$T_{\text{ш}}^{(1)} = \tau_1^{(1)}, \tau_2^{(1)}, \ldots, \tau_{64}^{(1)}).$$

4.1.4. Блок зашифрованных данных $T_{\text{ш}}^{(1)}$ одновременно является также исходным состоянием N_1, N_2 для выработки второго блока гаммы шифра $\Gamma_{\text{ш}}^{(2)}$ и по обратной связи записывается в указанные накопители. При этом значение $\tau_1^{(1)}$ вводится в 1-й раздел N 1, значение $\tau_2^{(1)}$ вводится во 2-й разряд N 1 и т.д., значение $\tau_{32}^{(1)}$ вводится в 32-й разряд N 1; значение $\tau_{33}^{(1)}$ вводится в 1-й разряд N 2, значение $\tau_{34}^{(1)}$ вводится во 2-й разряд N 2 и т.д., значение $\tau_{64}^{(1)}$ вводится в 32-й разряд N 2.

Заполнение N_1, N_2 зашифровывается в режиме простой замены в соответствии с требованиями п.2.1. Полученное в результате зашифрования заполнение N_1, N_2 образует второй 64-разрядный блок гаммы шифра $\Gamma_{\text{ш}}^{(2)}$, который суммируется поразрядно по модулю 2 в сумматоре CM_5 со вторым блоком открытых данных $T_0^{(2)}$.

Выработка последующих блоков гаммы шифра $\Gamma_{\text{ш}}^{(i)}$ и зашифрование соответствующих блоков открытых данных $T_0^{(i)}$ ($i = 3 \div M$) производится аналогично. Если длина последнего M-го блока открытых данных $T_0^{(\text{м})}$ меньше 64 разрядов, то из $\Gamma_{\text{ш}}^{(\text{м})}$ используется только соответствующее число разрядов гаммы шифра, остальные разр

яды отбрасываются.

4.1.5. Уравнения зашифрования в режиме гаммирования с обратной связью имеют вид:

$$\begin{cases} T_{\text{ш}}^{(1)} = A(S) \oplus T_0^{(1)} = \Gamma_{\text{ш}}^{(1)} \oplus T_0^{(1)} \\ T_{\text{ш}}^{(i)} = A(T_{\text{ш}}^{(i-1)}) \oplus T_0^{(i)} = \Gamma_{\text{ш}}^{(i)} \oplus T_0^{(i)}, \; i = 2 \div M. \end{cases}$$

4.1.6. В канал связи или память ЭВМ передаются синхропосылка S и блоки зашифрованных данных $T_{\text{ш}}^{(1)}$, $T_{\text{ш}}^{(2)}$, ... , $T_{\text{ш}}^{(\text{м})}$.

4.2. Расшифрование зашифрованных данных в режиме гаммирования с обратной связью

4.2.1. При расшифровании криптосхема имеет тот же вид (см. черт.4), что и при зашифровании.

В КЗУ вводятся 256 бит того же ключа, на котором осуществлялось зашифрование $T_0^{(1)}$, $T_0^{(2)}$, ... , $T_0^{(м)}$. Синхропосылка S вводится в N_1 и N_2 аналогично п.3.1.2.

4.2.2. Исходное заполнение N_1, N_2 (синхропосылка S) зашифровывается в режиме простой замены согласно подразделу 2.1. Полученное в результате зашифрования заполнение N_1, N_2 образует первый блок гаммы шифра $Г_ш^{(1)} = A(S)$, который суммируется поразрядно по модулю 2 в сумматоре $СМ_5$ с блоком зашифрованных данных $T_ш^{(1)}$. В результате получается первый блок открытых данных

$T_0^{(1)}$.

4.2.3. Блок зашифрованных данных $T_ш^{(1)}$ является исходным заполнением N_1, N_2 для выработки второго блока гаммы шифра $Г_ш^{(2)}$. Блок $T_ш^{(1)}$ записывается в N_1, N_2 в соответствии с требованиями п.4.1.4. Полученное заполнение N_1, N_2 зашифровывается в режиме простой замены в соответствии с требованиями п.2.1, полученный в результате блок $Г_ш^{(2)}$ суммируется поразрядно по модулю 2 в сумматоре $СМ_5$ со вторым блоком зашифрованных данных $T_ш^{(2)}$. В результате получается блок открытых данных $T_ш^{(2)}$.

Аналогично в N_1, N_2 последовательно записываются блоки зашифрованных данных $T_ш^{(2)}$, $T_ш^{(3)}$, … , $T_ш^{(м-1)}$, из которых в режиме простой замены вырабатываются блоки гаммы шифра $Г_ш^{(3)}$, $Г_ш^{(4)}$, … , $Г_ш^{(м)}$. Блоки гаммы шифра суммируются поразрядно по модулю 2 в сумматоре $СМ_5$ с блоками зашифрованных данных $T_ш^{(3)}$, $T_ш^{(4)}$, … , $T_ш^{(м)}$, в результате получаются блоки открытых данных $Г_0^{(3)}$, $Г_0^{(4)}$ ... , $Г_0^{(м)}$, при этом длина последнего блока открытых данных $T_0^{(м)}$ может содерж

ать меньше 64 разрядов.

4.2.4. Уравнения расшифрования в режиме гаммирования с обратной связью имеют вид:

$$T_o^{(1)} = A(S) \oplus T_ш^{(1)} = I_ш^{(1)} \oplus T_ш^{(1)}$$

$$T_o^{(i)} = A(T_ш^{(i-1)}) \oplus T_ш^{(i)} = I_ш^{(i)} \oplus T_ш^{(i)}, \quad i = 2 \div M$$

5. РЕЖИМ ВЫРАБОТКИ ИМИТОВСТАВКИ

5.1. Для обеспечения имитозашиты открытых данных, состоящих из M 64-разрядных блоков $T_0^{(1)}$, $T_0^{(2)}$, ... , $T_0^{(м)}$, $M \geq 2$, вырабатывается дополнительный блок из l бит (имитовставка $И_l$). Процесс выработки имитовставки единообразен для всех режимов шифрования

5.2. Первый блок открытых данных

$$T_0^{(1)} = (t_1^{(1)}, t_2^{(1)}, \ldots, t_{64}^{(1)}) = (a_1^{(1)}(0), a_2^{(1)}(0), \ldots, a_{32}^{(1)}(0), b_1^{(1)}(0), b_2^{(1)}(0), \ldots, b_{32}^{(1)}(0))$$

записывается в накопители N_1 и N_2, при этом значение $t_1^{(1)} = a_1^{(1)}(0)$ вводится в 1-й разряд N_1, значение $t_2^{(1)} = a_2^{(1)}(0)$ вводится во 2-й разряд N_1 и т.д., значение $t_{32}^{(1)} = a_{32}^{(1)}(0)$ вводится в 32-й разряд N_1; значение $t_{33}^{(1)} = b_1^{(1)}(0)$ вводится в 1-й разряд N_2 и т.д., значение $t_{64}^{(1)} = b_{32}^{(1)}(0)$ вводится в 32-й раз

ряд N_2.

5.3. Заполнение N_1 и N_2 подвергается преобразованию, соответствующему первым 16 циклам алгоритма зашифрования в режиме простой замены (см.подраздел 2.1). В КЗУ при этом находится тот же ключ, которым зашифровываются блоки открытых данных $T_0^{(1)}$, $T_0^{(2)}$, ... , $T_ш^{(м)}$ в соответствующие блоки зашифрованных данных $T_ш^{(1)}$, $T_ш^{(2)}$, ... , $T_ш^{(м)}$.

Полученное после 16 циклов работы заполнение N_1 и N_2, имеющее вид ($a_1^{(1)}(16)$, $a_2^{(1)}(16)$, ... , $a_{32}^{(1)}(16)$, $b_1^{(1)}(16)$, $b_2^{(1)}(16)$, ... , $b_{32}^{(1)}(16)$), суммируется в CM_5 по модулю 2 со вторым блоком $T_0^{(2)} = (t_1^{(2)}, t_2^{(2)}, \ldots, t_{64}^{(2)})$.

Результат суммирования

$$(a_1^{(1)}(16) \oplus t_1^{(2)}, a_2^{(1)}(16) \oplus t_2^{(2)}, \ldots, a_{32}^{(1)}(16) \oplus t_{32}^{(2)},$$

$$b_1^{(1)}(16) \oplus t_{33}^{(2)}, b_2^{(1)}(16) \oplus t_{34}^{(2)}, \ldots, b_{32}^{(1)}(16) \oplus t_{64}^{(2)} =$$

$$= (a_1^{(2)}(0), a_2^{(2)}(0), \ldots, a_{32}^{(2)}(0), b_1^{(2)}(0), b_2^{(2)}(0), \ldots, b_{32}^{(2)}(0))$$

заносится в N_1 и N_2 и подвергается преобразованию, соответствующему первым 16 циклам алгоритма зашифрования в режиме простой замены.

Полученное заполнение N_1 и N_2 суммируется по CM_5 по модулю 2 с третьим блоком $T_0^{(3)}$ и т.д., последний блок $T_0^{(м)} = (t_1^{(м)}, t_2^{(м)}, \dots, t_{64}^{(м)})$, при необходимости дополненный до полного 64-разрядного блока нулями, суммируется в CM_5 по модулю 2 с заполнением N_1, N_2

($a_1^{(м-1)}(16)$, $a_2^{(м-1)}(16)$, ... , $a_{32}^{(м-1)}(16)$, $b_1^{(м-1)}(16)$, $b_2^{(м-1)}(16)$, ... , $b_{32}^{(м-1)}(16)$).

Результат суммирования

$$(a_1^{(м-1)}(16) \oplus t_1^{(м)}, a_2^{(м-1)}(16) \oplus t_2^{(м)}, \dots, a_{32}^{(м-1)}(16) \oplus t_{32}^{(м)},$$

$$b_1^{(м-1)}(16) \oplus t_{33}^{(м)}, b_2^{(м-1)}(16) \oplus t_{34}^{(м)}, \dots, b_{32}^{(м-1)}(16) \oplus t_{64}^{(м)} =$$

$$= (a_1^{(м)}(0), a_2^{(м)}(0), \dots, a_{32}^{(м)}(0), b_1^{(м)}(0), b_2^{(м)}(0), \dots, b_{32}^{(м)}(0))$$

заносится в N_1, N_2 и зашифровывается в режиме простой замены по первым 16 циклам работы алгоритма. Из полученного заполнения накопителей N_1 и N_2

($a_1^{(м)}(16)$, $a_2^{(м)}(16)$, ... , $a_{32}^{(м)}(16)$, $b_1^{(м)}(16)$, $b_2^{(м)}(16)$, ... , $b_{32}^{(м)}(16)$)

выбирается отрезок $И_l$ (имитовставка) длиной l бит:

$$И_l = [a_{32-l+1}^{(м)}(16), a_{32-l+2}^{(м)}(16), \dots, a_{32}^{(м)}(16)].$$

Имитовставка $И_l$ передается по каналу связи или в память ЭВМ в конце зашифрованных данных, т.е. $T_{ш}^{(1)}$, $T_{ш}^{(2)}$, … , $T_{ш}^{(м)}$, $И_l$.

5.4. Поступившие зашифрованные данные $T_{ш}^{(1)}$, $T_{ш}^{(2)}$, ... , $T_{ш}^{(м)}$ расшифровываются, из полученных блоков открытых данных $T_0^{(1)}$, $T_0^{(2)}$, ... , $T_0^{(м)}$ аналогично п.5.3 вырабатывается имитовставка $И_l$, которая затем сравнивается с имитовставкой $И_l$, полученной вместе с зашифрованными данными из канала связи или из памяти ЭВМ. В случае несовпадения имитовставок полученные блоки открытых данных $T_0^{(1)}$, $T_0^{(2)}$, ... , $T_0^{(м)}$ считают ложными.

Выработка имитовставки $И_l$ ($И_l$') может производиться или перед зашифрованием (после расшифрования) всего сообщения, или параллельно с зашифрованием (расшифрованием) по блокам. Первые блоки открытых данных, которые участвуют в выработке имитовставки, могут содержать служебную информацию (адресную часть, отметку времени, синхропосылку и др.) и не зашифровываться.

Значение параметра l (число двоичных разрядов в имитовставке) определяется действующими криптографическими требованиями, при этом учитывается, что вероятность навязывания ложных данных равна 2^{-l}.

ПРИЛОЖЕНИЕ 1 (справочное). ТЕРМИНЫ, ПРИМЕНЯЕМЫЕ В НАСТОЯЩЕМ СТАНДАРТЕ, И ИХ ОПРЕДЕЛЕНИЯ

ПРИЛОЖЕНИЕ 1
Справочное

Термин	Определение
Алгоритм	По ГОСТ 19781
Гаммирование	Процесс наложения по определенному закону гаммы шифра на открытые данные
Гамма шифра	Псевдослучайная двоичная последовательность, вырабатываемая по заданному алгоритму для зашифрования открытых данных и расшифрования зашифрованных данных
Данные	По ГОСТ 15971
Зашифрование данных	Процесс преобразования открытых данных в зашифрованные при помощи шифра
Имитозащита	Защита системы шифрованной связи от навязывания ложных данных
Имитовставка	Отрезок информации фиксированной длины, полученной по определенному правилу из открытых данных и ключа и добавленный к зашифрованным данным для обеспечения имитозащиты
Канал связи	По ГОСТ 17657
Ключ	Конкретное секретное состояние некоторых параметров алгоритма криптографического преобразования данных, обеспечивающее выбор одного преобразования из совокупности всевозможных для данного алгоритма преобразований
Криптографическая защита	Защита данных при помощи криптографического преобразования данных

Криптографическое преобразование	Преобразование данных при помощи шифрования и (или) выработки имитовставки
Расшифровка данных	Процесс преобразования зашифрованных данных в открытые при помощи шифра
Синхропосылка	Значения исходных открытых параметров алгоритма криптографического преобразования
Уравнение зашифрования	Соотношение, выражающее процесс образования зашифрованных данных из открытых данных в результате преобразований, заданных алгоритмом криптографического преобразования
Уравнение расшифрования	Соотношение, выражающее процесс образования открытых данных из зашифрованных данных в результате преобразований, заданных алгоритмом криптографического преобразования
Шифр	Совокупность обратимых преобразований множества возможных открытых данных на множество возможных зашифрованных данных, осуществляемых по определенным правилам с применением ключей
Шифрование	Процесс зашифрования или расшифрования

ПРИЛОЖЕНИЕ 2 (обязательное). ЗНАЧЕНИЯ КОНСТАНТ C1, C2

ПРИЛОЖЕНИЕ 2
Обязательное

1. Константа C1 имеет вид:

Разряд накопителя N_6	32	31	30	29	28	27	26	25	24	23	22	21	20	19	18
Значение разряда	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0

Разряд накопителя N_6	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1
Значение разряда	1	0	0	0	0	0	0	0	1	0	0	0	0	0	1	0	0

2. Константа C2 имеет вид:

Разряд накопителя N_5	32	31	30	29	28	27	26	25	24	23	22	21	20	19	18
Значение разряда	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0

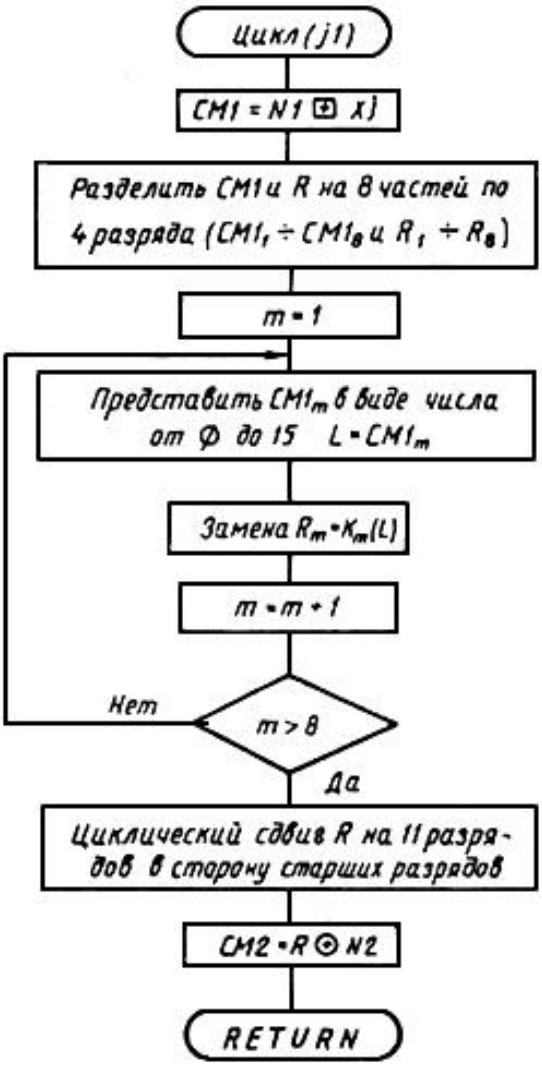
Разряд накопителя N_5	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1
Значение разряда	1	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	1

ПРИЛОЖЕНИЕ 3 (справочное). СХЕМЫ ПРОГРАММНОЙ РЕАЛИЗАЦИИ АЛГОРИТМА КРИПТОГРАФИЧЕСКОГО ПРЕОБРАЗОВАНИЯ

ПРИЛОЖЕНИЕ 3
Справочное

Черт.5. Схема одного цикла шифрования

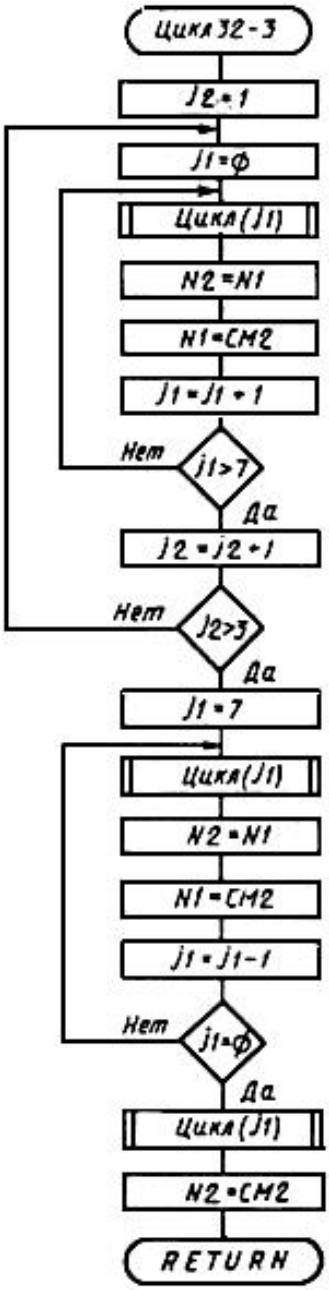
1. Схема одного цикла шифрования



Черт.5

2. Схема 32-х циклов зашифрования

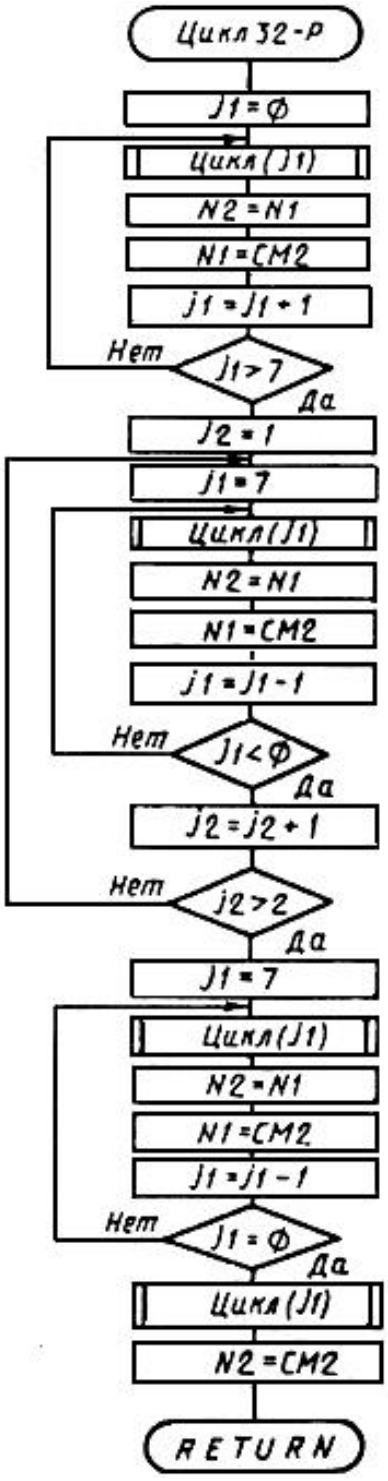
Черт.6. Схема 32-х циклов зашифрования



Черт.6

3. Схема 32-х циклов расшифрования

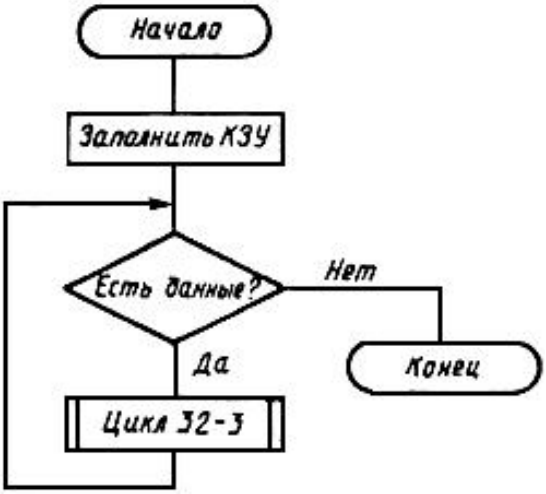
Черт.7. Схема 32-х циклов расшифрования



Черт.7

4. Схема алгоритма зашифрования в режиме простой замены

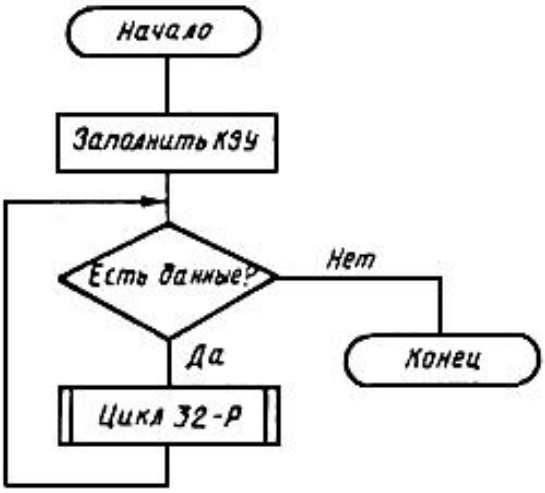
Черт.8. Схема алгоритма зашифрования в режиме простой замены



Черт.8

5. Схема алгоритма расшифрования в режиме простой замены

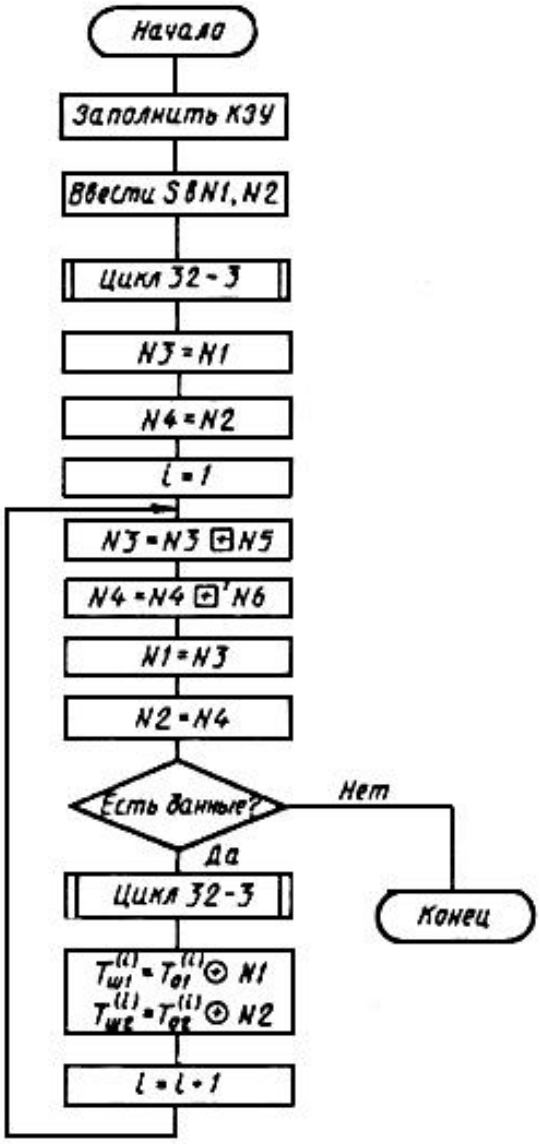
Черт.9. Схема алгоритма расшифрования в режиме простой замены



Черт.9

6. Схема алгоритма зашифрования в режиме гаммирования

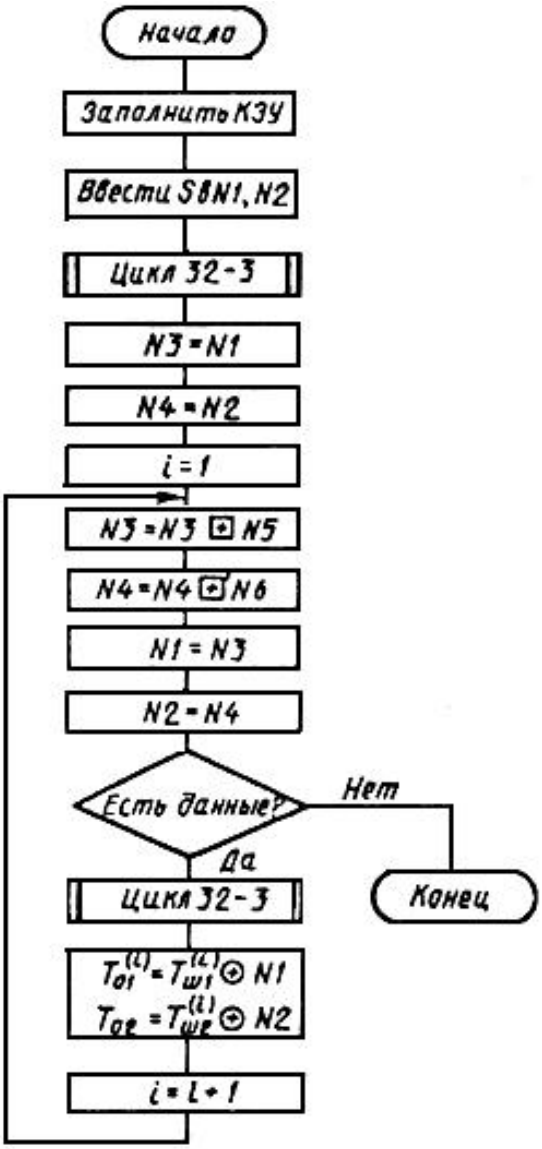
Черт.10. Схема алгоритма зашифрования в режиме гаммирования



Черт.10

7. Схема алгоритма расшифрования в режиме гаммирования

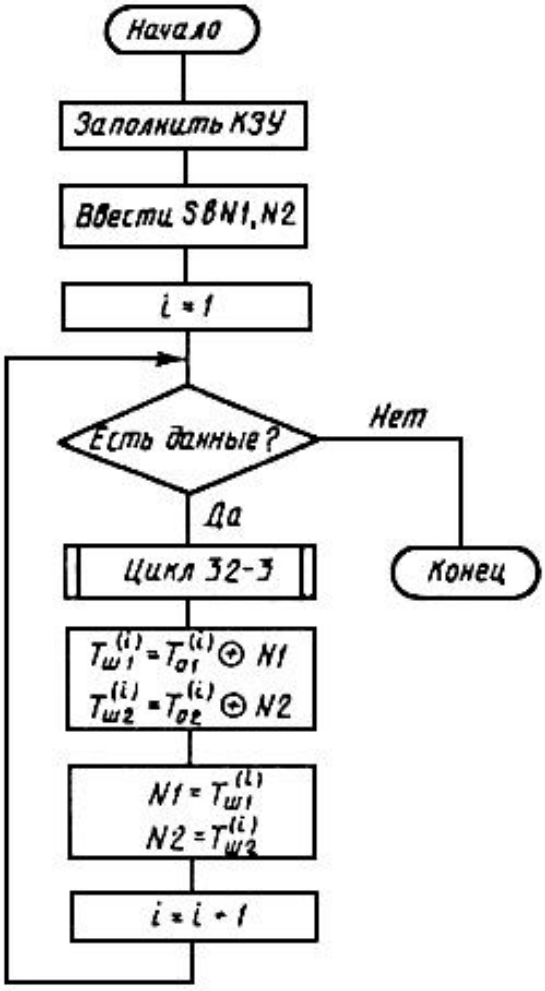
Черт.11. Схема алгоритма расшифрования в режиме гаммирования



Черт.11

8. Схема алгоритма зашифрования в режиме гаммирования с обратной связью

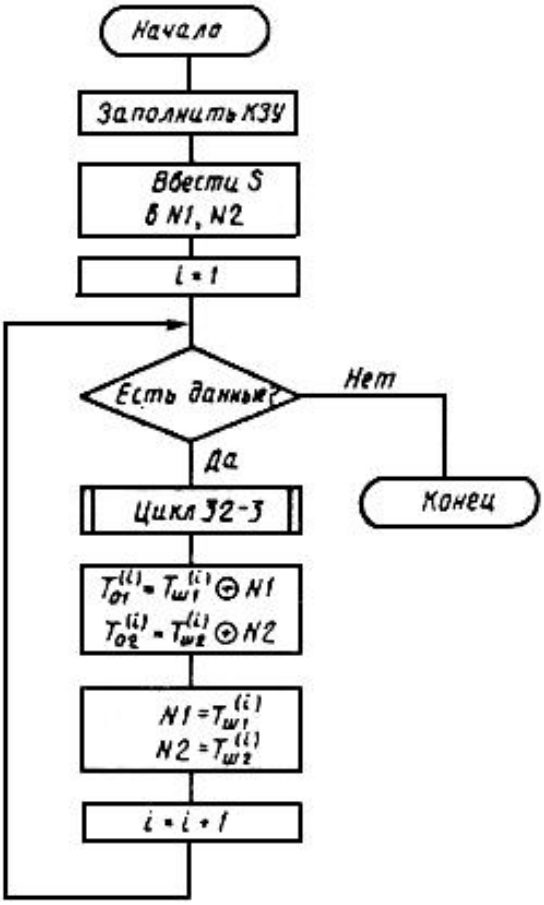
Черт.12. Схема алгоритма зашифрования в режиме гаммирования с обратной связью



Черт.12

9. Схема алгоритма расшифрования в режиме гаммирования с обратной связью

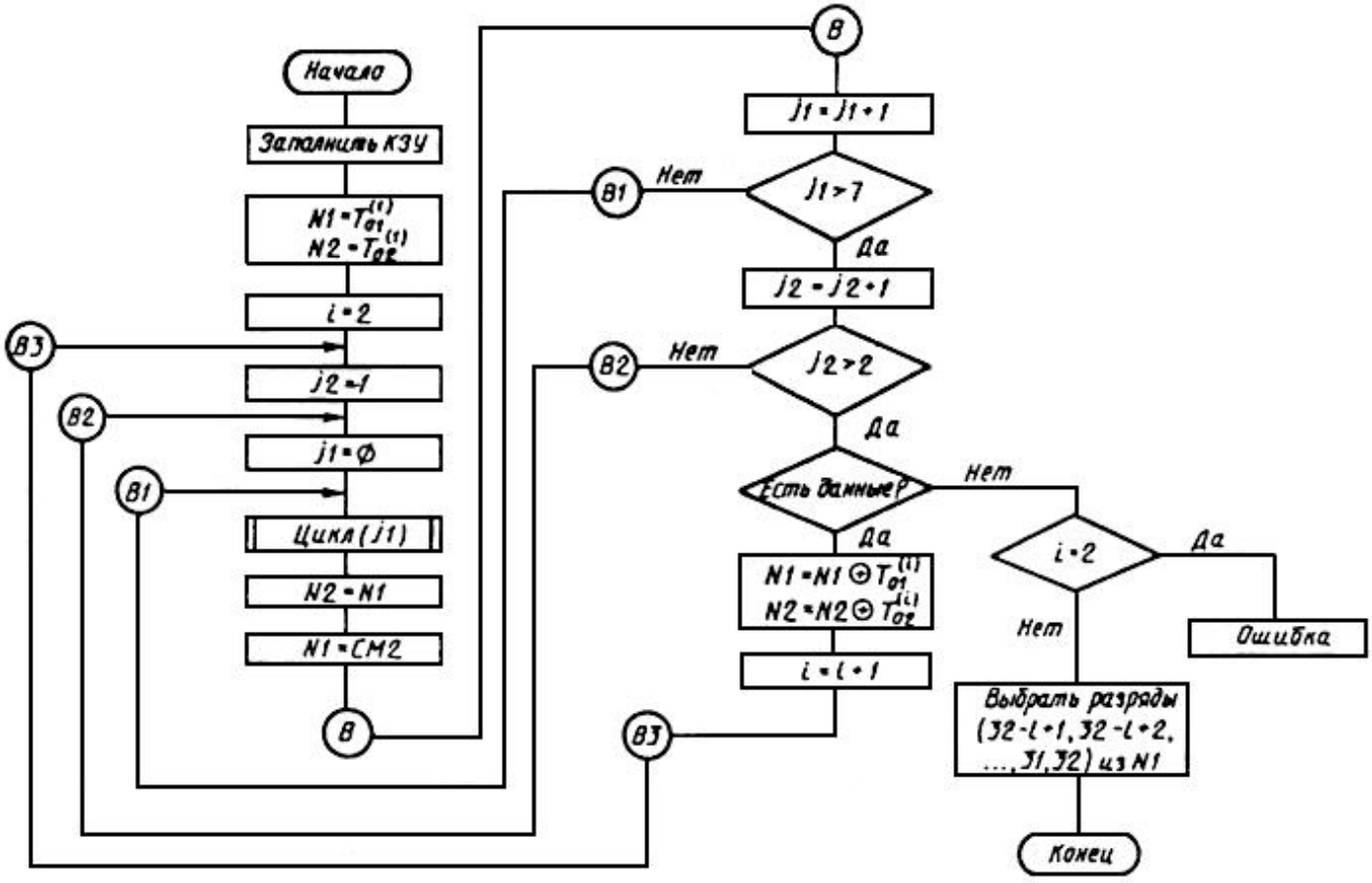
Черт.13. Схема алгоритма расшифрования в режиме гаммирования с обратной связью



Черт.13

10. Схема алгоритма криптографического преобразования в режиме выработки имитовставки

Черт.14. Схема алгоритма криптографического преобразования в режиме выработки имитовставки



Черт.14

ПРИЛОЖЕНИЕ 4 (справочное). ПРАВИЛА СУММИРОВАНИЯ ПО МОДУЛЮ 2_32 И ПО МОДУЛЮ (2_32-1)

ПРИЛОЖЕНИЕ 4
Справочное

ПРАВИЛА СУММИРОВАНИЯ ПО МОДУЛЮ 2^{32} И ПО МОДУЛЮ (2^{32} -1)

1. Два целых числа a, b, где 0 $\leq a$, $b \leq$ <, 2^{32} -1, представленные в двоичном виде

$a = (a_{32}, a_{31}, ..., a_2, a_1)$, $b = (b_{32}, и_{31}, ..., b_2, b_1)$,

т.е. $a = a_{32} \cdot 2^{31} + a_{31} \cdot 2^{30} + ... + a_2 \cdot 2 + a_1$, $b = b_{32} \cdot 2^{31} + b_{31} \cdot 2^{30} + ... + b_2 \cdot 2 + b_1$,

суммируются по модулю 2^{32} (операция $\boxplus$) по следующему правилу:

$a \boxplus b = a + b$, если $a + b < 2^{32}$,

$a \boxplus b = a + b - 2^{32}$, если $a + b \geq 2^{32}$,

где операция +(-) есть арифметическая сумма (разность) двух целых чисел.

2. Два целых числа a, b, где $0 \leq a$, $b \leq 2^{32}$ -1, представленные в двоичном виде

$a = (a_{32}, a_{31}, \ldots, a_2, a_1)$, $b = (b_{32}, b_{31}, \ldots, b_2, b_1)$,

суммируются по модулю (2^{32} -1) (операция $\boxplus'$) по следующему правилу:

$a \boxplus' b = a + b$, если $a + b < 2^{32}$,

$a \boxplus' b = a + b - 2^{32} + 1$, если $a + b \geq 2^{32}$.

Текст документа сверен по:
официальное издание
М.: ИПК Издательство стандартов, 1996