

Изменение № 1 O'z DSt 1105:2009

Информационная технология.
Криптографическая защита
информации. Алгоритм
шифрования данных

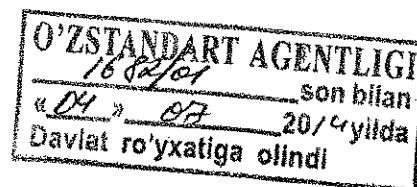
Утверждено и введено в действие постановлением Узбекского
агентства стандартизации, метрологии и сертификации (агентство
«Узстандарт») от 04.07.2014 № 05-556

Дата введения 14.07.2014

Срок действия государственного стандарта продлить до 15.10.2019 г.

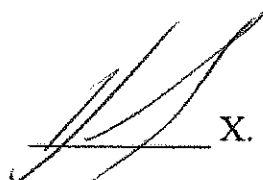
Раздел 2

В обозначении государственного стандарта O'z DSt 1109 год
утверждения «2006» заменить на «2013».



Изменение № 1 O'z DSt 1105:2009

Вр.и.о. директора
ГУП «UNICON.UZ»

 Х. Хасанов

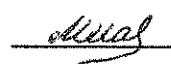
Начальник
научно-исследовательского отдела
криптографии

 О. Ахмедова

Младший научный сотрудник
научно-исследовательского отдела
криптографии

 Д. Джаматова

Нормоконтроль

 Л. Шаймарданова

СОГЛАСОВАНО

Начальник отдела
информационной безопасности
Государственного комитета связи,
информатизации и информационных
технологий Республики Узбекистан

А.Гафуров
письмо от 25.03.2014 г.
№ 14-8/1679

СОГЛАСОВАНО

Службой национальной
безопасности Республики Узбекистан

письмо от 24.06.2014 г.
№ 39/2199

ЎЗБЕКИСТОН РЕСПУБЛИКАСИ
STANDARTLASHTIRISH, DAVLAT
NAZORATINI MUHOFAZALASHTIRISH VA
AKBOROT TEXNOLOGIYALARINI
JONLIY ETISH BOSHQARMASI

ГОСУДАРСТВЕННЫЙ СТАНДАРТ УЗБЕКИСТАНА

**Информационная технология
КРИПТОГРАФИЧЕСКАЯ ЗАЩИТА ИНФОРМАЦИИ
Алгоритм шифрования данных**

Издание официальное

Узбекское агентство стандартизации, метрологии и сертификации
Ташкент

Предисловие

1 РАЗРАБОТАН И ВНЕСЕН Государственным унитарным предприятием Центр научно-технических и маркетинговых исследований - «UNICON.UZ» (ГУП «UNICON.UZ») Узбекского агентства связи и информатизации

2 УТВЕРЖДЕН И ВВЕДЕН В ДЕЙСТВИЕ постановлением Узбекского агентства стандартизации, метрологии и сертификации (агентство «Узстандарт») от 28.09.09 № 05-163

3 В настоящем стандарте реализованы нормы законов Республики Узбекистан «Об электронной цифровой подписи» и «Об электронном документообороте»

4 ВЗАМЕН O'z DSt 1105:2006

Информация о введении в действие (прекращении действия) настоящего стандарта и изменений к нему на территории Узбекистана публикуется в указателе, издаваемом агентством «Узстандарт». В случае пересмотра или отмены настоящего стандарта соответствующая информация будет опубликована в информационном указателе, издаваемом агентством «Узстандарт».

Исключительное право официального опубликования настоящего стандарта на территории Узбекистана принадлежит агентству «Узстандарт»

Содержание

1	Область применения	1
2	Нормативная ссылка	1
3	Термины, определения и обозначения	2
3.1	Термины и определения	2
3.2	Обозначения	2
4	Общие положения	3
5	Математические соглашения	4
5.1	Математические определения и предпосылки	4
5.2	Формы представления и соглашения	6
5.3	Параметры и функции алгоритма шифрования	9
6	Основные процессы	11
6.1	Вводные замечания	11
6.2	Процедура зашифрования	11
6.3	Процедура расшифрования	14
6.4	Вопросы реализации	20
	Приложение А (справочное) Контрольный пример	22

O'ZSTANDART AGENTLIGI
 STANDARTLASHTIRISH,
 KO'YBEIQLASHTIRISH
 VA DAVLAT NAZORATI
 BESHQARVASI

ГОСУДАРСТВЕННЫЙ СТАНДАРТ УЗБЕКИСТАНА

Ахборот технологияси
АХБОРОТНИНГ КРИПТОГРАФИК МУҲОФАЗАСИ
Маълумотларни шифрлаш алгоритми

Информационная технология
КРИПТОГРАФИЧЕСКАЯ ЗАЩИТА ИНФОРМАЦИИ
Алгоритм шифрования данных

Information technology
CRYPTOGRAPHIC DATA SECURITY
Algorithm of data encryption

Дата введения 15.10.2009
15.10.2014

1 Область применения

Настоящий стандарт «Алгоритм шифрования данных» (АШД) представляет собой криптографический алгоритм, предназначенный для защиты электронных данных. АШД - симметричный блочный шифр, который используется для зашифрования и расшифрования информации. АШД может использовать криптографические ключи длиной 256 или 512 bit для зашифрования и расшифрования блоков данных длиной 256 bit.

Стандарт устанавливает единый алгоритм шифрования информации для систем обработки информации в сетях электронных вычислительных машин (ЭВМ), телекоммуникаций, отдельных вычислительных комплексах и ЭВМ и определяет правила шифрования данных.

Алгоритм шифрования данных предназначен для программной, аппаратной, аппаратно-программной реализации.

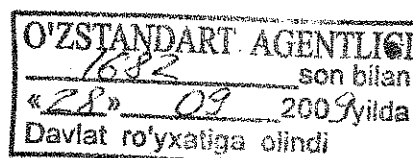
Стандарт может быть использован для криптографической защиты данных, хранимых и передаваемых в сетях ЭВМ, телекоммуникаций, в отдельных вычислительных комплексах или в ЭВМ предприятий, организаций и учреждений.

2 Нормативная ссылка

В настоящем стандарте использованы ссылки на следующие стандарты:

O'z DSt 1047:2003 Информационные технологии. Термины и определения.

Издание официальное



O'z DSt 1109:2006 Информационная технология. Криптографическая защита информации. Термины и определения.

Примечание – При использовании настоящим стандартом целесообразно проверить действие ссылочного стандартов (классификаторов) на территории Узбекистана по соответствующему указателю стандартов, составленному по состоянию на 1 января текущего года, и по соответствующим информационным указателям, опубликованным в текущем году. Если ссылочный документ заменен (изменен), то при использовании настоящим стандартом следует руководствоваться замененным (измененным) стандартом. Если ссылочный документ отменен без замены, то положение, в котором дана ссылка на него, применяется в части, не затрагивающей эту ссылку.

3 Термины, определения и обозначения

3.1 Термины и определения

В настоящем стандарте применяются термины по O'z DSt 1047, O'z DSt 1109, а также следующие термины с соответствующими определениями:

3.1.1 вектор инициализации: вектор, используемый для определения исходной точки криптографического процесса в рамках криптографического алгоритма.

3.1.2 сеансовый ключ: двумерный массив секретных ключей, формируемый на основе ключа шифрования и функционального ключа.

3.1.3 средства шифрования: аппаратные, программные и аппаратно-программные средства, реализующие криптографические алгоритмы преобразования информации и предназначенные для защиты информации от несанкционированного доступа при ее обработке, хранении и передаче по каналам связи.

3.1.4 режим сцепления блоков: режим шифрования, в котором каждый зашифрованный (расшифрованный) блок зависит от предыдущего зашифрованного (расшифрованного) блока. Для первого блока в качестве предыдущего блока используется вектор инициализации. В случае, если последний блок текста является не полным, он дополняется до необходимой длины.

3.1.5 режим электронной кодовой книги: режим шифрования, в котором все блоки открытого текста шифруются независимо друг от друга на одном ключе, в соответствии с алгоритмом шифрования данных.

3.2 Обозначения

В настоящем стандарте использованы следующие обозначения:

M – исходные (открытые) данные (сообщение пользователя);

C – зашифрованный текст;

m – режим шифрования;

sh – индекс, обозначающий режим преобразования открытого текста в зашифрованный текст;

dsh – индекс, обозначающий режим преобразования зашифрованного текста в открытый текст;

$Holat$ – двумерный массив состояния;

H_{ch}, H_o – левая (верхняя) и правая (нижняя) половина (часть) двумерного массива состояния;

k – ключ шифрования;

k_f – функциональный ключ;

k_{se} – сеансово - этапный ключ;

B_a – линейный массив преобразования элементов на байтовом уровне;

K_{ss} – линейный массив элементов сеансового ключа шифрования;

K_s – двумерный массив сеансового ключа шифрования;

K_{sch}, K_{so} – левая (верхняя) и правая (нижняя) половина (часть) массива сеансового ключа шифрования;

K_e – двумерные массивы этапного ключа;

$p, (p+1)$ – модуль, $p = 256$;

e – число этапов процедуры зашифрования или расшифрования;

$bosqich$ – порядковый номер этапа процедуры зашифрования или расшифрования;

R – параметр;

$\oplus$ – символ операции XOR (операции сложения по модулю 2);

$\otimes$ – символ операции умножения чисел с коэффициентом R по модулю p ;

$\otimes_2$ – символ операции диаметричного умножения по модулю p ;

$\overset{-1}{\otimes}$ – символ операции обращения с коэффициентом R по модулю p ;

$\overset{d}{\otimes}$ – символ операции возведения в степень d с параметром R по модулю p ;

$\overset{-1}{\otimes}$ – символ операции обращения по модулю p или $p+1$;

IV – вектор инициализации.

O'ZSTANDART AGENTLIGI
STANDARTLASHTIRISH,
KUVVATLASHTIRISH
VA DAVLAT NAZORATI
TASHKENT

4 Общие положения

4.1 В симметричных криптосистемах обмен сообщениями происходит в трёх этапах:

- отправитель сообщения передаёт получателю ключ шифрования (или/и функциональный ключ) по защищенному каналу, который никому кроме их самих неизвестен;

- отправитель с помощью ключа шифрования и функционального ключа преобразует исходные данные в зашифрованные данные и отправляет их получателю по каналу связи;

- получатель, получив зашифрованные данные, расшифровывает их с помощью ключа шифрования и функционального ключа. Обе стороны могут воспользоваться этими ключами несколько раз.

4.2 Описываемый АШД основан на использовании преобразований перемешивания столбцов и по байтовой замене, и характеризуется следующими признаками:

- ключи этапа формируются на основе ключа шифрования и функционального ключа, периодически обновляемого через определенное количество сеансов, в зависимости от требуемого уровня защиты информации. Функциональный ключ обновляется на каждом сеансе, когда требуется высокий уровень защиты;

- в перемешивании столбцов участвуют секретные параметры.

Перечисленные признаки обеспечивают повышение криптостойкости алгоритма шифрования данных.

5 Математические соглашения

5.1 Математические определения и предпосылки

5.1.1 Для определения АШД необходимо описать базовые математические объекты, используемые в процессах зашифрования и расшифрования. В данном разделе установлены основные математические определения и требования, накладываемые на параметры алгоритма шифрования данных.

5.1.2 В АШД используется алгебра диаматриц модульной арифметики, вычисления в которой осуществляются на том же уровне трудоемкости, что и в алгебре матриц.

5.1.3 Основной операцией алгебры диаматриц, используемой в процедурах зашифрования и расшифрования, является операция обращения диаматрицы в матрицу по модулю p . В этих операциях принимают участие левая или правая половина двумерного массива сеансового ключа, отображаемая квадратной диаматрицей порядка 4×4 со специальной структурой, где все диагональные элементы идентичны, и идентичны недиагональные элементы в строке 1, а также элементы в начале и конце строки 2. Диаматрица специальной структуры порядка 4×4 формируется на основе десяти элементов на байтовом уровне. На рисунке 1 приведена диаматрица, сформированная на основе элементов $d_0, d_1, d_2, d_3, d_4, d_5, d_6, d_7, d_8, d_9$.

d_7	d_0	d_1	d_2
d_8	d_7	d_8	d_8
d_9	d_3	d_7	d_9
d_4	d_5	d_6	d_7

Рисунок 1 - Диаматрица специальной структуры

Важным свойством диаматрицы специальной структуры является простота формулы для вычисления диаопределителя диаматрицы, что

упрощает процедуру проверки условий обратимости диаматрицы. Обратная диаматрица диаматрицы специальной структуры сохраняет исходную структуру.

5.1.4 Диаопределитель диаматрицы специальной структуры порядка 4×4 находится как произведение диагонального элемента на три сомножителя, каждый из которых представляет собой сумму элементов столбца с элементом, расположенным на одной строке с диагональным по соседству справа.

Диаопределитель d для рассматриваемой диаматрицы находится как:

$$d \equiv d_7 \times (d_7 + d_0 + d_8 + d_3 + d_5) \times (d_7 + d_1 + d_8 + d_9 + d_6) \times (d_7 + d_2 + d_8 + d_9 + d_4) \pmod{p}. \quad (1)$$

Проверка условий обратимости диаматрицы специальной структуры является основным требованием, предъявляемым к параметрам АШД. Она сводится к сравнению с нулем значений диагонального элемента и упомянутых сомножителей по модулю 2. Это позволяет из любого ключа шифрования и функционального ключа формировать обратимую диаматрицу.

5.1.5 В АИД для перемешивания элементов используется операция умножения над матрицами порядка 4×4 :

$$H' \equiv H \otimes_2 K \pmod{p},$$

здесь: $\otimes_2$ – символ операции диаметричного умножения по модулю, H', H, K – диаметрицы порядка 4×4 , H' – результирующая диаметрица, H, K – заданные диаметрицы.

Операция диаматричного умножения $\otimes_2$ выполняется на основе следующих выражений.

Для $s, u \in \{0, 1, 2, 3\}$:

$$h'[u, u] \equiv h[u, u] * \sum_{i=0}^3 k[i, u] - \sum_{i=0, i \neq u}^3 h[i, i] * k[i, u] \pmod{p},$$

$$h'[s,u]_{s \neq u} \equiv h[s,u] * \sum_{i=0}^3 k[i,u] + k[s,u] * \sum_{i=0}^3 h[i,u] - \sum_{i=0; i \neq s,u}^3 h[s,i] * k[i,u] \pmod{p},$$

здесь: $h'[u,u]$ — диагональные элементы результирующей диаграммы, $h'[s,u]$ — недиагональные элементы результирующей диаграммы, $h[s,u]$, $h[i,u]$, $h[s,i]$, $h[i,i]$, $k[i,u]$, $k[s,u]$ — элементы H , K .

5.1.6 Также в АШД используются операции алгебры с параметром: умножение, обращение и возведение в степень целых чисел с параметром, алгебры с параметром.

Умножение с параметром R по модулю p числа X на Y обозначается как $X \circledast Y \pmod{p}$ и определяется следующим образом:

$$X \circledast Y \pmod{p} \equiv X + Y(1 + RX) \pmod{p}. \quad (2)$$

Эта операция является коммутативной и ассоциативной операцией.

Операция обращения с параметром R переменной по модулю p числа X обозначается как $X^{+1} \pmod{p}$ и определяется следующим образом:

$$X^{i-1} \pmod{p} \equiv -X(1+RX)^{-1} \pmod{p}, \quad (3)$$

здесь $X^{i-1} \otimes X \equiv 0 \pmod{p}$, 0 — единичный элемент группы параметров.

ЭЛЕМЕНТ ГРУППЫ
STANDARTLASHTIRISH,
MUVOFIQLASHTIRISH
VA DAVLAT NAZORATI
BIRLASHMASI

Операция возведения в степень d с параметром R по модулю p основания X выражается следующим образом: $X^{1d} \pmod{p}$. Например, если $d = 37$ возведение в степень с параметром R по модулю p основания X вычисляется следующим образом:

$$X^{137} \Rightarrow X^{132+4+1} \pmod{p} \equiv (((X^{12})^{12})^{12})^{12} \otimes (X^{12})^{12} \otimes X \pmod{p},$$

здесь: $X^{12} \pmod{p} \equiv X(2 + XR) \pmod{p}$.

Используемое в АШД возведение чисел побайтно в степень с параметром выполняется с учетом требования обратимости преобразований.

5.2 Формы представления и соглашения

5.2.1 Входы и выходы

5.2.1.1 В АШД как вход, так и выход представляют собой последовательности длиной 256 bit, которые иногда будут называться блоками. В АШД используются два ключа - ключ шифрования и функциональный ключ, каждый из которых представляет собой последовательность из 256 bit. Исходный функциональный ключ генерируется в двух режимах: 1) как функция от ключа шифрования; 2) независимо от ключа шифрования, т.е. в виде псевдослучайного числа. Обновление функционального ключа осуществляется использованием функции хэширования, где ключом хэширования служит ключ шифрования, входом - функциональный ключ, использованный на предыдущем сеансе. Генерация функционального ключа в зависимости от ключа шифрования длиной 256 bit равносильно использованию в АШД ключа шифрования с длиной 512 bit. Другие длины входа, выхода, ключа шифрования и функционального ключа указанным стандартом не допускаются.

5.2.1.2 Биты в последовательностях нумеруются, начиная с нуля и заканчивая номером на единицу меньшим, чем длина последовательности. При этом i - порядковый номер бита, известный как его индекс, будет находиться в диапазоне $0 \leq i < 256$ для входных блоков длиной 256 bit.

5.2.1.3 В процедурах зашифрования и расшифрования данных алгоритма АШД используются двумерные массивы сеансового и этапных ключей, которые формируются на основе ключа шифрования и функционального ключа.

5.2.1.4 В АШД для обработки входных блоков основной единицей считается байт - последовательность состоящая из восьми bit. Битовые последовательности входных, выходных, сеансовых и этапных ключей обрабатываются как массивы, состоящие из байтов, при этом определенные битовые последовательности формируются посредством разделения на группы, построенные из соседних битов.

5.2.1.5 Байты результирующего массива входного, выходного, сеансово-этапного или этапного ключей, обозначенных буквой a ,

применяются с использованием одного из обозначений a_n или $a[n]$, здесь n принимает значения $0 \leq n < 32$.

5.2.1.6 Все значения байта представляются, как конкатенация его индивидуальных битовых значений (0 или 1) между скобками в порядке $\{b_7, b_6, b_5, b_4, b_3, b_2, b_1, b_0\}$.

Например, для значения байта $\{01100011\}$, битовые значения определяются как $b_7=0, b_6=1, b_5=1, b_4=0, b_3=0, b_2=0, b_1=1, b_0=1$.

5.2.2 Массивы байтов

5.2.2.1 Для входного блока массивы байтов выражаются следующим образом:

$$a_0, a_1, a_2, a_3, \dots, a_{31}.$$

Байты и расположение битов в байтах определяется 256 битовой последовательностью $kir_0, kir_1, kir_2, kir_3, \dots, kir_{254}, kir_{255}$ следующим образом:

$$a_0 = \{kir_0, kir_1, \dots, kir_7\};$$

$$a_1 = \{kir_8, kir_9, \dots, kir_{15}\};$$

:

$$a_{31} = \{kir_{248}, kir_{249}, \dots, kir_{255}\}.$$

Учитывая вышеизложенное на рисунке 2 приведена нумерация битов в байте.

Последовательности битов	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19		23
Номер байта	0								1								2					
Номер бита в байте	7	6	5	4	3	2	1	0	7	6	5	4	3	2	1	0	7	6	5	4		0

Рисунок 2 - Нумерация байтов и битов

5.2.3 Массив Holat

5.2.3.1 Процесс обработки данных для входных блоков в АШД можно интерпретировать как выполнение операций алгоритма над двумерным массивом байтов *Holat*.

Массив *Holat* состоит из четырех (или восьми) строк и восьми (или четырех) столбцов байтов, причем каждая строка содержит 32 (64) bit.

В массиве *Holat*, обозначаемом h , каждый отдельный байт имеет два индекса s и u , где s - номер его строки в диапазоне $0 \leq s < 4$ (8); u - номер его столбца в диапазоне $0 \leq u < 8$ (4). Эта индексация позволяет ссылаться на конкретный байт массива *Holat* как на $h[s, u]$. Следующие разъяснения в основном приведены на примере массива размером 4×8 . Интерпретация относительно массивов размера 8×4 аналогична.

5.2.3.2 В самом начале процессов зашифрования и расшифрования, которые будут подробно разбираться позже, входной массив *kir* - массив полубайтов $kir_0, kir_1, \dots, kir_{31}$ - копируется в массив *Holat*, как показано на рисунке 3. Затем в массиве *Holat* выполняются необходимые операции зашифрования или расшифрования, после чего окончательное значение элементов массива *Holat* копируется в выходной массив *chiq* - массив полубайтов (байтов) $chiq_0, chiq_1, \dots, chiq_{31}$ (на рисунках 3-7 массивы имеют размер 4×8).

Байты массива *kir*

kir_0	kir_4	kir_8	kir_{12}	kir_{16}	kir_{20}	kir_{24}	kir_{28}
kir_1	kir_5	kir_9	kir_{13}	kir_{17}	kir_{21}	kir_{25}	kir_{29}
kir_2	kir_6	kir_{10}	kir_{14}	kir_{18}	kir_{22}	kir_{26}	kir_{30}
kir_3	kir_7	kir_{11}	kir_{15}	kir_{19}	kir_{23}	kir_{27}	kir_{31}

↓ Массив *Holat*

$h[0,0]$	$h[0,1]$	$h[0,2]$	$h[0,3]$	$h[0,4]$	$h[0,5]$	$h[0,6]$	$h[0,7]$
$h[1,0]$	$h[1,1]$	$h[1,2]$	$h[1,3]$	$h[1,4]$	$h[1,5]$	$h[1,6]$	$h[1,7]$
$h[2,0]$	$h[2,1]$	$h[2,2]$	$h[2,3]$	$h[2,4]$	$h[2,5]$	$h[2,6]$	$h[2,7]$
$h[3,0]$	$h[3,1]$	$h[3,2]$	$h[3,3]$	$h[3,4]$	$h[3,5]$	$h[3,6]$	$h[3,7]$

↓ Байты массива *chiq*

$chiq_0$	$chiq_4$	$chiq_8$	$chiq_{12}$	$chiq_{16}$	$chiq_{20}$	$chiq_{24}$	$chiq_{28}$
$chiq_1$	$chiq_5$	$chiq_9$	$chiq_{13}$	$chiq_{17}$	$chiq_{21}$	$chiq_{25}$	$chiq_{29}$
$chiq_2$	$chiq_6$	$chiq_{10}$	$chiq_{14}$	$chiq_{18}$	$chiq_{22}$	$chiq_{26}$	$chiq_{30}$
$chiq_3$	$chiq_7$	$chiq_{11}$	$chiq_{15}$	$chiq_{19}$	$chiq_{23}$	$chiq_{27}$	$chiq_{31}$

Рисунок 3 - Вход и выход массива *Holat*

Следует отметить, что перед преобразованиями, основанными на использовании операций над матрицами, двумерный массив *Holat* $[4,8]$ следует разбивать на левую и правую половины (части): $H_{ch} [4,4]$ и $H_o [4,4]$, как это иллюстрировано на рисунке 4.

Массив H_{ch}

$h_{ch}[0,0]$	$h_{ch}[0,1]$	$h_{ch}[0,2]$	$h_{ch}[0,3]$
$h_{ch}[1,0]$	$h_{ch}[1,1]$	$h_{ch}[1,2]$	$h_{ch}[1,3]$
$h_{ch}[2,0]$	$h_{ch}[2,1]$	$h_{ch}[2,2]$	$h_{ch}[2,3]$
$h_{ch}[3,0]$	$h_{ch}[3,1]$	$h_{ch}[3,2]$	$h_{ch}[3,3]$

Массив H_o

$h_o[0,0]$	$h_o[0,1]$	$h_o[0,2]$	$h_o[0,3]$
$h_o[1,0]$	$h_o[1,1]$	$h_o[1,2]$	$h_o[1,3]$
$h_o[2,0]$	$h_o[2,1]$	$h_o[2,2]$	$h_o[2,3]$
$h_o[3,0]$	$h_o[3,1]$	$h_o[3,2]$	$h_o[3,3]$

Рисунок 4 - Массивы $H_{ch} [4,4]$ и $H_o [4,4]$

По окончании преобразования эти половины вновь копируются в массив *Holat* [4,8] (рисунок 5).

Массив *Holat*

$h[0,0]$	$h[0,1]$	$h[0,2]$	$h[0,3]$	$h[0,4]$	$h[0,5]$	$h[0,6]$	$h[0,7]$
$h[1,0]$	$h[1,1]$	$h[1,2]$	$h[1,3]$	$h[1,4]$	$h[1,5]$	$h[1,6]$	$h[1,7]$
$h[2,0]$	$h[2,1]$	$h[2,2]$	$h[2,3]$	$h[2,4]$	$h[2,5]$	$h[2,6]$	$h[2,7]$
$h[3,0]$	$h[3,1]$	$h[3,2]$	$h[3,3]$	$h[3,4]$	$h[3,5]$	$h[3,6]$	$h[3,7]$

Рисунок 5 - Массив *Holat*

5.2.4 Массив сеансового ключа K_s

Процедура формирования массива сеансового ключа K_s начинается с формирования двумерных массивов $K_1[4, 4]$ и $K_2[4, 4]$ из 20 левых элементов массива $K_{st}[32]$, состоящего из байтовых элементов, а также включает в себя операцию обращения диаматрицы в диаматрицу. Процедура формирования сеансового ключа приведена в 6.3.2.

5.2.5 Массив этапного ключа K_e

Процедуры зашифрования и расшифрования начинаются с формирования массива сеансового ключа K_s , используемого при побайтовой замене на каждом этапе. Этот массив формируется для каждого этапа на основе сдвига массива линейного этапно-сеансового ключа k_{se} , заданного на битовом уровне.

Массив K_e состоит из четырех (восьми) строк и восьми (четыре) столбцов байтов и к каждому байту массива можно обращаться также, как в массивах *Holat* и K_s , т.е. как $K_e[s, u]$.

Результат формирования этапного ключа выражается в виде двумерного массива, как показано на рисунке 6.

Массив K_e

$k_e[0,0]$	$k_e[0,1]$	$k_e[0,2]$	$k_e[0,3]$	$k_e[0,4]$	$k_e[0,5]$	$k_e[0,6]$	$k_e[0,7]$
$k_e[1,0]$	$k_e[1,1]$	$k_e[1,2]$	$k_e[1,3]$	$k_e[1,4]$	$k_e[1,5]$	$k_e[1,6]$	$k_e[1,7]$
$k_e[2,0]$	$k_e[2,1]$	$k_e[2,2]$	$k_e[2,3]$	$k_e[2,4]$	$k_e[2,5]$	$k_e[2,6]$	$k_e[2,7]$
$k_e[3,0]$	$k_e[3,1]$	$k_e[3,2]$	$k_e[3,3]$	$k_e[3,4]$	$k_e[3,5]$	$k_e[3,6]$	$k_e[3,7]$

Рисунок 6 - Массивы этапного ключа K_e

Процедура формирования этапных ключей приведена в 6.3.5.

5.3 Параметры и функции алгоритма шифрования данных

АШД использует следующие параметры и функции:

- k – ключ шифрования длиной 256 или 512 bit;
- k_f – функциональный ключ длиной 256 bit;
- K_e – двумерный массив этапного ключ порядка 8×4 (или 4×8);

d) b – количество входных блоков длиной 256 bit;

e) e – количество этапов, $e=8$;

f) $p, (p+1)$ – модуль, $p=256$;

g) *Aralash()* – является криптографическим преобразованием и выполняется над диаметричными частями при зашифровании и расшифровании; входными данными при зашифровании являются диаметричные части массива *Holat*, а также массивы K_1 и K_2 , выходными данными является массив *Holat*.

h) *BaytAlmash()* – является криптографическим преобразованием и используется для замены на байтовом уровне элементов массива *Holat* с элементами массива замен; входными данными данного криптопреобразования являются массив *Holat*, линейный массив замен B_{sA} [256] или B_{sAD} [256] на байтовом уровне, а выходными данными является массив *Holat* на байтовом уровне;

i) *Sur()* – используется при зашифровании и расшифровании для тщательного перемешивания элементов массива *Holat*; входными данными данного преобразования является массив *Holat*, при шифровании выходными данными – массив *Holat* с циклически сдвинутыми вниз столбцами и с циклически сдвинутыми вправо строками; при расшифровании выходными данными является массив *Holat* с циклически сдвинутыми вверх столбцами и с циклически сдвинутыми влево строками;

j) *ShaklSeansKalitBayt()* – используется для формирования ключа для каждого сеанса и для выполнения преобразования *BaytAlmash()* при зашифровании и расшифровании; в данном преобразовании входными данными являются ключ шифрования k и функциональный ключ k_f , выходными данными являются массивы B_{sA} [256] и B_{sAD} [256] на байтовом уровне;

k) *ShaklSeansKalit()* – используется для формирования ключа для каждого сеанса и для выполнения преобразования *Aralash()* при зашифровании и расшифровании; входными данными данного преобразования является массив $K_{st}=[32]$ на байтовом уровне; выходными данными являются пара массивов (K_{1t}, K_{2t}) или (K_1, K_2) , состоящих из диаграмм специальной структуры;

l) *ShaklBosqichKalit()* – используется для формирования из сеансово-этапного ключа этапного ключа и для выполнения преобразования *Qo'shBosqichKalit()* при зашифровании и расшифровании; входными данными данного преобразования является линейный массив сеансово-этапного ключа k_{se} , выходными данными является заданный на байтовом уровне двумерный массив $K_d[8,4]$;

m) *Qo'shBosqichKalit()* – является простым криптопреобразованием и заключается в выполнении операции исключающего ИЛИ (побитовое сложение по модулю 2) при зашифровании и расшифровании массивов *Holat* и массива этапного ключа K_e ; входными данными данного преобразования являются массивы *Holat* и K_e на байтовом уровне, выходными данными является массив *Holat* на байтовом уровне.

n) *Qo'shHolat()* — является простым криптопреобразованием и выполняется над блоками шифруемых блоков с использованием операции XOR при зашифровании и расшифровании во всех режимах кроме режима электронной кодовой книги.

6 Основные процессы

6.1 Вводные замечания

6.1.1 В АШД длина входного и выходного блоков, а также длина элементов массива *Holat* равны 256 bit. Длина ключа шифрования и функционального ключа в АШД также равны 256 bit. Для АШД число этапов установлено $e=8$.

6.1.2 Как в режиме зашифрования, так и в режиме расшифрования алгоритм АШД использует однократное преобразование - формирование массива сеансового ключа и следующих четырех байт-ориентированных и одного бит — ориентированного преобразования на каждом этапе, приведенных в 6.3. К этим преобразованиям относятся:

- a) формирование массивов этапных ключей;
- b) смешивание данных на основе массива сеансового ключа;
- d) циклические сдвиги строк и столбцов массива *Holat* на различные значения смещений;
- e) побайтовая замена байтов массива *Holat* на основе массивов линейного массива;
- f) операция сложения по модулю 2 массивов *Holat* и массива этапного ключа K_e ;
- g) циклические сдвиги линейного массива сеансово - этапного ключа на одно и то же значение битов на каждом этапе.

6.2 Процедура зашифрования

6.2.1 Псевдокод процедуры зашифрования

При инициализации шифрующего криптографического модуля (рисунок 7) сначала в криптографический модуль загружаются ключ шифрования k и функциональный ключ k_f , количество этапов e , а также вектор инициализации IV для режима $m=ShBil$. Также, при зашифровании в массив *Holat* криптографического модуля загружается открытый текст, при зашифровании шифртекст. В начале процедуры зашифрования инициализируются криптопреобразования *ShaklSeansKalitBayt(k,k_f)*, *ShaklSeansKalit(K_{se})* и *ShaklBosqichKalit(k_{se})*. На выходах криптопреобразований *ShaklSeansKalitBayt(k,k_f)*, *ShaklSeansKalit(K_{se})* на байтовом уровне формируются массивы замен и сеансовый ключ, состоящий из диаматричных частей. Эти массивы используются в следующих сеансах до тех пор, пока k , k_f остаются постоянными. На

выходе криптопреобразования $ShaklBosqichKalit(k_{se})$ формируется начальный ключ и набор этапных ключей, сформированный для каждого этапа.

Псевдокод для режима электронной кодовой книги (**Elektron kod kitobi**) $m=Ekk$ и режима сцепления блоков (**ShifrBloklarni ilaktirish**) $m=ShBil$ приведен на рисунке 7.

6.2.2 Простые криптопреобразования $Aralash(Holat, K_s)$, $BaytAlmash(Holat, B_s)$, $Qo'shBosqichKalit(Holat, K_s)$, $Sur(Holat)$ и преобразования $ShaklSeansKalitBayt(k, k_p)$, $ShaklSeansKalit(K_{sd})$, $ShaklBosqichKalit(k_{se})$ и $Qo'shHolat(Holatm, Holat)$ приведены в следующем подразделе.

Обновление функционального ключа в аппаратно-программном модуле целесообразно выполнять в совокупности с преобразованиями $ShaklSeansKalitBayt(k, k_p)$, $ShaklSeansKalit(K_{sd})$, $ShaklBosqichKalit(k_{se})$. В этом случае в псевдокод шифрования необходимо ввести результаты $ShaklSeansKalitBayt(k, k_p)$, $ShaklSeansKalit(K_{sd})$, $ShaklBosqichKalit(k_{se})$.

В приложении А в качестве примера приведен псевдокод процедуры зашифрования (sh) и расшифрования (dsh) для базовых режимов **Elektron kod kitobi** и **ShifrBloklarni ilaktirish**, здесь приведены значения массива **Holat** после применения в процессе каждого этапа вышеупомянутых преобразований.

```

Shifr (int blok_soni, byte IV[32], byte kirish [blok_soni][32],
byte chiqish [blok_soni][32], byte k[32], byte k_f[32], byte e)
begin
    byte k_e [8,4], K_s [8,4], K_e [8,4]
    Holat [8,4], Holatn[8,4]
    if(m=Sh)
        ShaklSeansKalitBayt (k, k_f)
        ShaklSeansKalit(K_st)
        ShaklBosqichKalit(k_se)
        for blok =1 step 1 to blok_soni
            Holat=kirish[blok]
            if(m=ShBil)
                if(blok=1)
                    Holatn=IV
                else
                    Holatn=chiqish[blok-1]
                end if
                Qo'shHolat(Holat, Holatn)
            end if
            for bosqich =1 step 1 to e
                Qo'shBosqichKalit(Holat, K_e)
                Aralash(Holat, K_s)
                Sur(Holat)
                BaytAlmash(Holat, B_a)
            end for
            Qo'shBosqichKalit(Holat, K_e)
            Aralash(Holat, K_s)
            chiqish[blok]= Holat
        end for
    else
        ShaklSeansKalitBayt (k, k_f)
        ShaklSeansKalit(K_st)
        ShaklBosqichKalit(k_se)
        for blok =1 step 1 to blok_soni
            Holat=kirish[blok]
            Aralash(Holat, K_s)
            Qo'shBosqichKalit(Holat, K_e)
            for bosqich =1 step 1 to e
                BaytAlmash(Holat, B_a)
                Sur(Holat)
                Aralash(Holat, K_s)
                Qo'shBosqichKalit(Holat, K_e)
            end for
            if(m=ShBil)
                if(blok=1)
                    Holatn=IV
                else
                    Holatn = kirish[blok-1]
                end if
                Qo'shHolat(Holat, Holatn)
            end if
            chiqish[blok]= Holat
        end for
    end if
end

```

O'ZSTANDART AGENTLIGI
 STANDARTLASHTIRISH,
 MUVOFIQLASHTIRISH
 VA DAVLAT NAZORATI
 QO'RG'ONASI

Рисунок 7 – Псевдокод процедуры шифрования

6.3 Криптографические преобразования

6.3.1 Преобразование *ShaklSeansKalitBayt(k,k_p)*

Преобразование *ShaklSeansKalitBayt(k,k_p)* заключается в выполнении следующих операций.

6.3.1.1 Вычислить $k_{se} = k + k' * (1 + k_{fp} * k)$ и оставить 672 bit слева, здесь k' - это 192 bit k_{fp} справа.

Примечание - При генерации ключа шифрования, набора функциональных ключей, обновляемых с использованием ключа шифрования, и при формировании на их основе ключа k_{se} они обязательно должны быть проверены на основе критериев случайности и с левой части результирующего k_{se} , начиная с первого левого бита выделяются 672 bit.

6.3.1.2 Выделить с правой части k_{se} 256+64 bit, из левой 256 битной части сформировать линейный массив $K_{st} = [0, 1, 2, 3, \dots, 31]$, состоящий из байтовых элементов, из остальной 64 битной части сформировать линейный массив $B = [0, 1, 2, 3, 4, 5, 6, 7]$, состоящий из элементов на байтовом уровне.

6.3.1.3 Из элементов линейного массива B сформировать пару массивов $B_1 = [0, 1, 2, 3]$ и $B_2 = [4, 5, 6, 7]$ и на основе нижеперечисленных правил сформировать тройку параметров (d_1, R_1, L_1) и (d_2, R_2, L_2) :

1) для $j=0$, если $b[j] < 3$, принять $d_1=3$, в противном случае принять $d_1 = b[0]$; для $j=4$, если $b[j] < 3$, принять $d_2=3$, в противном случае принять $d_2 = b[4]$.

2) для $j=1$, если $b[j]=0$, принять $R_1=1$, в противном случае принять $R_1 = b[1]$; для $j=5$, если $b[j]=0$, принять $R_2=1$, в противном случае $R_2 = b[5]$.

3) для $j=2$, если $b[j]=0$, принять $L_1=1$, в противном случае принять $L_1 = b[2]$; для $j=6$, если $b[j]=0$, принять $L_2=1$, в противном случае принять $L_2 = b[6]$.

4) для $d_s \pmod{2}=0$, если $d_s \pmod{4}=0$, принять $d_s = d_s - 1$, в противном случае принять $d_s = d_s + 1$, здесь $s \in \{1, 2\}$.

5) для $d_s \pmod{2}=1$, если $d_s - 1 \pmod{4}=0$, тогда принять $d_s = d_s - 2$.

6) для $j=3$, если $b[j]=0$, принять $b[j]=1$; для $j=7$, если $b[j]=0$, принять $b[j]=1$.

6.3.1.4 Для выполнения преобразований на байтовом уровне и получения шифртекста сформировать пару массивов $(B_{1A} [256], B_{2A} [256])$. Смысл формирования этих массивов разъясняется в нижеследующих указаниях:

- для $s \in \{1, 2\}$ возвести в степень d_s с параметром R_s по модулю 257 значение $((i + L_s) \pmod{256}) + 1$, соответствующее каждому адресу $i \in \{0, 1, 2, \dots, 255\}$, результат привести по модулю 256 и в каждом шаге текущее значение сравнить со значением предыдущего шага и i . Если значения равны или, если текущее значение близко ($|b_{sA}[i-1] - b_{sA}[i]| \geq 8$) к предыдущему значению, то это текущее значение заменить на $(i - b[3])$ при $s=1$ и на $(i - b[7])$ при $s=2$.

Примечание - В псевдокоде в нечетных этапах используется массив $B_{1A} [256]$, а в четных этапах массив $B_{2A} [256]$.

Алгоритм вычислений включает в себя следующие операции:

1) вычислить $b_{sA}[i] = (((i+L) \bmod 256) + 1)^{ds} \bmod 257 \bmod 256$ для $i=0 \div 255$.

Начиная с $i=1$ на каждом шаге проверить условие $i-b_{sA}[i] \neq 0$ и $|b_{sA}[i-1]-b_{sA}[i]| \geq 8$ и если оба условия выполняются, то принять значение, в противном случае при $s=1$ поменять местами $b_{sA}[i]$ и элемент, находящийся по адресу $(i-b[3]) \bmod 256$ или при $s=2$ поменять местами $b_{sA}[i]$ и элемент, находящийся по адресу $(i-b[7]) \bmod 256$ и принять $b[3] = b[3]-5 \bmod 256$ для $s=1$ или $b[7] = b[7]-5 \bmod 256$ для $s=2$.

2) из элементов $b_{sA}[i, i=0,1,2,\dots,255]$, чтобы получить шифртекст на байтовом уровне, т.е. для использования в режиме *sh* сформировать пару линейных массивов ($B_{1A}[256], B_{2A}[256]$).

6.3.1.5 Для расшифрования на байтовом уровне, т.е. для использования в режиме *dsh* сформировать пару линейных массивов ($B_{1AD}[256], B_{2AD}[256]$).

Для этого, чтобы получить шифртекст (*dsh*) на байтовом уровне, достаточно заменить каждый элемент $b_{sA}[i]$ на значение в линейном массиве $B_{sA}[256]$ с индексом, равным его значению, здесь $s \in \{1,2\}$ и расположить элементы сформированного массива в порядке возрастания адреса.

Примечание - В псевдокоде в нечетных этапах используется массив $B_{2AD}[256]$, в четных этапах массив $B_{1AD}[256]$.

6.3.2 Преобразование *ShaklSeansKalit* (K_{ss})

Преобразование *ShaklSeansKalit* (K_{ss}) заключается в выполнении следующих операций.

6.3.2.1 Сформировать линейный массив $K_{ss}=[0,1,2,3,\dots,19]$, состоящий из 20 байтовых элементов слева линейного массива $K_{ss}=[0,1,2,3,\dots,31]$, состоящего из байтовых элементов.

Если $k_{ss}[i] = 0$ для $i = 0 - 19$, заменить $k_{ss}[i]$ на $k_{ss}[i]-1 \bmod p$.

Если $k_{ss}[6] \bmod 2 = 0$, заменить $k_{ss}[6]$ на $k_{ss}[6]-1 \bmod p$.

Если $k_{ss}[16] \bmod 2 = 0$, заменить $k_{ss}[16]$ на $k_{ss}[16]-1 \bmod p$.

Если $k_{ss}[6]+k_{ss}[0]+k_{ss}[8]+k_{ss}[3]+k_{ss}[5] \bmod 2 = 0$, заменить $k_{ss}[8]$ на $k_{ss}[8]-1 \bmod p$.

Если $k_{ss}[16]+k_{ss}[10]+k_{ss}[18]+k_{ss}[13]+k_{ss}[15] \bmod 2 = 0$, заменить $k_{ss}[18]$ на $k_{ss}[18]-1 \bmod p$.

Если $k_{ss}[6]+k_{ss}[1]+k_{ss}[3]+k_{ss}[9]+k_{ss}[4] \bmod 2 = 0$, заменить $k_{ss}[9]$ на $k_{ss}[9]-1 \bmod p$.

Если $k_{ss}[16]+k_{ss}[11]+k_{ss}[13]+k_{ss}[19]+k_{ss}[14] \bmod 2 = 0$, заменить $k_{ss}[19]$ на $k_{ss}[19]-1 \bmod p$.

Если $k_{ss}[6]+k_{ss}[2]+k_{ss}[3]+k_{ss}[4]+k_{ss}[7] \bmod 2 = 0$, заменить $k_{ss}[7]$ на $k_{ss}[7]-1 \bmod p$.

Если $k_{ss}[16]+k_{ss}[12]+k_{ss}[13]+k_{ss}[14]+k_{ss}[17] \bmod 2 = 0$, заменить $k_{ss}[17]$ на $k_{ss}[17]-1 \bmod p$.

UZSTATISTIK AGENTLIGI
STANDARTLASHTIRISH
MUVOFIDLASHTIRISH
VA BAYLAT NAZORATI

6.3.2.2 Из элементов линейного массива K_{ss} в следующем порядке сформировать двумерные массивы $K_1[4, 4]$ и $K_2[4, 4]$:

1) разделить линейный массив $K_{ss}=[0,1,2,3,...19]$ на два линейных массива $K_{ss1}=[0,1,2,3,...9]$ и $K_{ss2}=[10,11,12,13,...,19]$ и каждый из них однозначно отобразить в упорядоченный набор $\{k_{s1}[0,1], k_{s1}[0,2], k_{s1}[0,3], k_{s1}[1,0], k_{s1}[2,0], k_{s1}[2,1], k_{s1}[2,2], k_{s1}[3,0], k_{s1}[3,1], k_{s1}[3,2]\}$ и $\{k_{s2}[0,1], k_{s2}[0,2], k_{s2}[0,3], k_{s2}[1,0], k_{s2}[2,0], k_{s2}[2,1], k_{s2}[2,2], k_{s2}[3,0], k_{s2}[3,1], k_{s2}[3,2]\}$ и соответственно из каждого из них сформировать элементы $k_1[i,j]$, $k_2[i,j]$ массивов $K_1[4, 4]$ и $K_2[4, 4]$;

2) сформировать остальные элементы массива $K_s[4, 4]$, $s \in \{1,2\}$ на основе следующего правила:

- для $j \in \{0,1,2,3\}$, если $i=j$, то соответствующие элементы равны по значению $k_s[2,2]$;

- для $i=1, j=0,2,3$ соответствующие элементы равны по значению $k_s[1,0]$;

- для $i=2, j=0,3$ соответствующие элементы равны по значению $k_s[2,0]$.

В результате для использования в режиме шифрования в качестве $K_s[8, 4]$ формируется пара диаграмм специальной структуры $K_1[4, 4]$ и $K_2[4, 4]$.

В результате для использования в режиме *sh* в качестве $K_s[8, 4]$ формируется две диаграмм специальной структуры $K_1[4, 4]$ и $K_2[4, 4]$.

$k_{ss}[6]$	$k_{ss}[0]$	$k_{ss}[1]$	$k_{ss}[2]$
$k_{ss}[3]$	$k_{ss}[6]$	$k_{ss}[3]$	$k_{ss}[3]$
$k_{ss}[4]$	$k_{ss}[5]$	$k_{ss}[6]$	$k_{ss}[4]$
$k_{ss}[7]$	$k_{ss}[8]$	$k_{ss}[9]$	$k_{ss}[6]$

$k_{ss}[16]$	$k_{ss}[10]$	$k_{ss}[11]$	$k_{ss}[12]$
$k_{ss}[13]$	$k_{ss}[16]$	$k_{ss}[13]$	$k_{ss}[13]$
$k_{ss}[14]$	$k_{ss}[15]$	$k_{ss}[16]$	$k_{ss}[14]$
$k_{ss}[17]$	$k_{ss}[18]$	$k_{ss}[19]$	$k_{ss}[16]$



Массив K_1

$k_1[0,0]$	$k_1[0,1]$	$k_1[0,2]$	$k_1[0,3]$
$k_1[1,0]$	$k_1[1,1]$	$k_1[1,2]$	$k_1[1,3]$
$k_1[2,0]$	$k_1[2,1]$	$k_1[2,2]$	$k_1[2,3]$
$k_1[3,0]$	$k_1[3,1]$	$k_1[3,2]$	$k_1[3,3]$

Массив K_2

$k_2[0,0]$	$k_2[0,1]$	$k_2[0,2]$	$k_2[0,3]$
$k_2[1,0]$	$k_2[1,1]$	$k_2[1,2]$	$k_2[1,3]$
$k_2[2,0]$	$k_2[2,1]$	$k_2[2,2]$	$k_2[2,3]$
$k_2[3,0]$	$k_2[3,1]$	$k_2[3,2]$	$k_2[3,3]$

6.3.2.3 Для матрицы специальной структуры $K_{1i}[4, 4]$ вычислить обратную матрицу специальной структуры $K_{1i}[4, 4]$ для использования в режиме *sh*.

Для матрицы специальной структуры $K_{2i}[4, 4]$ вычислить обратную матрицу специальной структуры $K_{2i}[4, 4]$ для использования в режиме *dsh*.

Получение обратной матрицы для диаматрицы специальной структуры $K_{1i}[4, 4]$ (здесь $i=\{1,2\}$), у которой диаопределитель не равен нулю, заключается в вычислении обратной матрицы для матрицы, полученной в результате выполнения диапреобразования над ним и в результате выполнения диапреобразования над полученной обратной матрицей.

В результате обращения диаматриц специальной структуры $K_{1i}[4, 4]$ и $K_{2i}[4, 4]$ образуются диаматрицы специальной структуры $K_{1i}[4, 4]$ и $K_{2i}[4, 4]$, имеющие следующий вид (K_{1i} и K_{2i}).

Массив K_{1i}

$k_{1i}[0,0]$	$k_{1i}[0,1]$	$k_{1i}[0,2]$	$k_{1i}[0,3]$
$k_{1i}[1,0]$	$k_{1i}[1,1]$	$k_{1i}[1,2]$	$k_{1i}[1,3]$
$k_{1i}[2,0]$	$k_{1i}[2,1]$	$k_{1i}[2,2]$	$k_{1i}[2,3]$
$k_{1i}[3,0]$	$k_{1i}[3,1]$	$k_{1i}[3,2]$	$k_{1i}[3,3]$

Массив K_{2i}

$k_{2i}[0,0]$	$k_{2i}[0,1]$	$k_{2i}[0,2]$	$k_{2i}[0,3]$
$k_{2i}[1,0]$	$k_{2i}[1,1]$	$k_{2i}[1,2]$	$k_{2i}[1,3]$
$k_{2i}[2,0]$	$k_{2i}[2,1]$	$k_{2i}[2,2]$	$k_{2i}[2,3]$
$k_{2i}[3,0]$	$k_{2i}[3,1]$	$k_{2i}[3,2]$	$k_{2i}[3,3]$

В режиме *sh* на вход подается пара, состоящая из (K_{1i} , K_{2i}), в режиме *dsh* на вход подается пара состоящая из (K_1 , K_2).

6.3.3 Преобразование *Aralash* ($Holat$, K_s)

Преобразование *Aralash* ($Holat$, K_s) заключается в выполнении следующих операций.

Если $m=sh$, то принять $K_1=K_{1i}$, $K_2=K_{2i}$, вычислить $H_1 \otimes_2 K_1 \pmod{p}$, $H_2 \otimes_2 K_2 \pmod{p}$, результат записать в массивы H_1 , H_2 и копировать в массив *Holat*, в противном случае, т.е. если $m=dsh$, то принять $K_1=K_1$, $K_2=K_2$, вычислить $H_1 \otimes_2 K_1 \pmod{p}$, $H_2 \otimes_2 K_2 \pmod{p}$, результат записать в массивы H_1 , H_2 и копировать в массив *Holat*.

Операция диаматричного умножения $\otimes_2$ выполняется на основе нижеследующих выражений, здесь индекс, используемый в выражениях, принимает значения $s \in \{1,2\}$.

Выражения для $i=j \in \{0,1,2,3\}$:

$$h'_s[0,0] = h_s[0,0](k_s[0,0] + k_s[1,0] + k_s[2,0] + k_s[3,0]) - h_s[1,1]k_s[1,0] - h_s[2,2]k_s[2,0] - h_s[3,3]k_s[3,0] \pmod{p},$$

$$h'_s[1,1] = h_s[1,1](k_s[0,1] + k_s[1,1] + k_s[2,1] + k_s[3,1]) - h_s[0,0]k_s[0,1] - h_s[2,2]k_s[2,1] - h_s[3,3]k_s[3,1] \pmod{p},$$

$$h'_s[2,2] = h_s[2,2](k_s[0,2] + k_s[1,2] + k_s[2,2] + k_s[3,2]) - h_s[0,0]k_s[0,2] - h_s[1,1]k_s[1,2] - h_s[3,3]k_s[3,2] \pmod{p},$$

$$h'_s[3,3] = h_s[3,3](k_s[0,3] + k_s[1,3] + k_s[2,3] + k_s[3,3]) - h_s[0,0]k_s[0,3] - h_s[1,1]k_s[1,3] - h_s[2,2]k_s[2,3] \pmod{p}.$$

UZSTAFBAIT AGENTLIGI
STANDARTLASHTIRISH
MUHOFIGLASHTIRISH
VA DAVLAT NAZORATI

Выражения для $i \neq j \in \{0,1,2,3\}$:

$$h'_s[0,1]=h_s[0,1](k_s[0,1]+k_s[1,1]+k_s[2,1]+k_s[3,1])+(h_s[0,0]+h_s[1,0]+h_s[2,0]+h_s[3,0])k_s[0,1]-h_s[0,2]k_s[2,1]-h_s[0,3]k_s[3,1] \pmod{p},$$

$$h'_s[0,2]=h_s[0,2](k_s[0,2]+k_s[1,2]+k_s[2,2]+k_s[3,2])+(h_s[0,0]+h_s[1,0]+h_s[2,0]+h_s[3,0])k_s[0,2]-h_s[0,1]k_s[1,2]-h_s[0,3]k_s[3,2] \pmod{p},$$

$$h'_s[0,3]=h_s[0,3](k_s[0,3]+k_s[1,3]+k_s[2,3]+k_s[3,3])+(h_s[0,0]+h_s[1,0]+h_s[2,0]+h_s[3,0])k_s[0,3]-h_s[0,1]k_s[1,3]-h_s[0,2]k_s[2,3] \pmod{p},$$

$$h'_s[1,0]=h_s[1,0](k_s[0,0]+k_s[1,0]+k_s[2,0]+k_s[3,0])+(h_s[0,1]+h_s[1,1]+h_s[2,1]+h_s[3,1])k_s[1,0]-h_s[1,2]k_s[2,0]-h_s[1,3]k_s[3,0] \pmod{p},$$

$$h'_s[1,2]=h_s[1,2](k_s[0,2]+k_s[1,2]+k_s[2,2]+k_s[3,2])+(h_s[0,1]+h_s[1,1]+h_s[2,1]+h_s[3,1])k_s[1,2]-h_s[1,0]k_s[0,2]-h_s[1,3]k_s[3,2] \pmod{p},$$

$$h'_s[1,3]=h_s[1,3](k_s[0,3]+k_s[1,3]+k_s[2,3]+k_s[3,3])+(h_s[0,1]+h_s[1,1]+h_s[2,1]+h_s[3,1])k_s[1,3]-h_s[1,0]k_s[0,3]-h_s[1,2]k_s[2,3] \pmod{p},$$

$$h'_s[2,0]=h_s[2,0](k_s[0,0]+k_s[1,0]+k_s[2,0]+k_s[3,0])+(h_s[0,2]+h_s[1,2]+h_s[2,2]+h_s[3,2])k_s[2,0]-h_s[2,1]k_s[1,0]-h_s[2,3]k_s[3,0] \pmod{p},$$

$$h'_s[2,2]=h_s[2,2](k_s[0,2]+k_s[1,2]+k_s[2,2]+k_s[3,2])+(h_s[0,2]+h_s[1,2]+h_s[2,2]+h_s[3,2])k_s[2,2]-h_s[2,0]k_s[0,2]-h_s[2,3]k_s[3,2] \pmod{p},$$

$$h'_s[2,3]=h_s[2,3](k_s[0,3]+k_s[1,3]+k_s[2,3]+k_s[3,3])+(h_s[0,2]+h_s[1,2]+h_s[2,2]+h_s[3,2])k_s[2,3]-h_s[2,0]k_s[0,3]-h_s[2,1]k_s[1,3] \pmod{p},$$

$$h'_s[3,0]=h_s[3,0](k_s[0,0]+k_s[1,0]+k_s[2,0]+k_s[3,0])+(h_s[0,3]+h_s[1,3]+h_s[2,3]+h_s[3,3])k_s[3,0]-h_s[3,1]k_s[1,0]-h_s[3,2]k_s[2,0] \pmod{p},$$

$$h'_s[3,1]=h_s[3,1](k_s[0,1]+k_s[1,1]+k_s[2,1]+k_s[3,1])+(h_s[0,3]+h_s[1,3]+h_s[2,3]+h_s[3,3])k_s[3,1]-h_s[3,0]k_s[0,1]-h_s[3,2]k_s[2,1] \pmod{p},$$

$$h'_s[3,2]=h_s[3,2](k_s[0,2]+k_s[1,2]+k_s[2,2]+k_s[3,2])+(h_s[0,3]+h_s[1,3]+h_s[2,3]+h_s[3,3])k_s[3,2]-h_s[3,0]k_s[0,2]-h_s[3,1]k_s[1,2] \pmod{p}.$$

Данное преобразование эффективнее по сравнению с матричным преобразованием. Здесь, при изменении одного элемента в исходном массиве *Holat*, в зависимости от адреса изменившегося элемента, изменяется 6 или 7 элементов.

6.3.4 Преобразование *BaytAlmash(Holat, B_a)*

Преобразование *BaytAlmash(Holat, B_a)* заключается в выполнении следующих операций.

1) Назвать массив *Holat[8,4]*, заданный на байтовом уровне, как массив *Holatb[8, 4]* на байтовом уровне.

2) Если $m=sh$, то принять $B_a[256] = B_{sA}[256]$; заменить каждый элемент массива *Holatb[8, 4]* элементом массива *B_a*, находящегося по адресу, равному значению элемента массива *Holatb[8, 4]* и присвоить результирующий массив *Holatb[8, 4]* массиву *Holat[8, 4]*, заданного на байтовом уровне, в противном случае, если $m=dsh$, принять $B_a[256] = B_{sAD}[256]$, заменить каждый элемент массива *Holatb[8, 4]* элементом массива *B_a*, находящимся по адресу равному значению элемента массива *Holatb[8, 4]* и присвоить результирующий массив *Holatb[8, 4]* массиву *Holat[8, 4]*, заданному на байтовом уровне, здесь $s \in \{1,2\}$.

6.3.5 Преобразование *ShaklBosqichKalit* (k_{se})

Преобразование *ShaklBosqichKalit* (k_{se}) (формирование линейного сеансово-этапного ключа) происходит следующим образом:

1) если $bosqich=1$ и $m=sh$, то оставить неизменным массив k_{se} линейного сеансово-этапного ключа, если $bosqich=0$ и $m=dsh$, то сдвинуть массив k_{se} на $672-(e \times 83) \bmod 672$ bit вправо.

Отделить левую 256 битовую часть массива линейного сеансово-этапного ключа и сформировать из неё на байтовом уровне массив $K_d[8,4]$. Это преобразование выполняется для всех этапов до начала процедуры шифрования.

2) если $bosqich>1$ и $m=sh$, то массив k_{se} циклически сдвинуть на 83 bit влево; если $bosqich\geq 1$ и $m=dsh$, то массив k_{se} циклически сдвинуть на 83 bit вправо.

Отделить левую 256 битовую часть массива линейного сеансово-этапного ключа и сформировать из неё на байтовом уровне массив $K_d[8,4]$.

Это преобразование выполняется для всех этапов до начала процедуры расшифрования.

6.3.6 Преобразование *Qo'shBosqichKalit* (*Holat*, K_d)

Преобразование *Qo'shBosqichKalit* (*Holat*, K_d) заключается в выполнении операции исключающего ИЛИ (побитовое сложение по модулю 2) над одноименными элементами на байтовом уровне массивов *Holat* и $K_d[8,4]$.

Для $0 \leq c < 8$:

$$[h'[c,0], h'[c,1], h'[c,2], h'[c,3]] = [h[c,0], h[c,1], h[c,2], h[c,3]] \oplus [k_d[c,0], k_d[c,1], k_d[c,2], k_d[c,3]].$$

Результат скопировать в массив *Holat*.

6.3.7 Преобразование *Sur* (*Holat*)

Если $m=sh$, то сначала циклически сдвинуть j -столбец массива *Holat* на $(j+1) \bmod 8$ байтов вниз, затем сдвинуть i -строку результирующего массива на $(i+1) \bmod 4$ байтов вправо, в противном случае, т.е. если $m=dsh$, то сначала циклически сдвинуть i -строку массива *Holat* на $(i+1) \bmod 4$ байтов влево, затем сдвинуть j -столбец результирующего массива на $(j+1) \bmod 8$ байтов вверх. Здесь, $0 \leq i < 4$, $0 \leq j < 8$.

6.3.8 Преобразование *Qo'shHolat*(*Holatin*, *Holat*)

В преобразовании *Qo'shHolat*(*Holatin*, *Holat*) к каждому байту массива *Holatin* прибавляется побитово с помощью операции сложения XOR (сложение по модулю 2) одноименный байт массива *Holat*. Массив *Holatin* состоит восьми слов. Когда эти слова в диапазоне $0 \leq s < 8$, то элементы массива *Holat*, находящиеся, в столбцах складываются по отдельности следующим образом:

O'ZSTATBART AGENTLIGI
STANBARILASHTIRISH,
MUVOFIQLASHTIRISH
VA DAVLAT NAZORATI

$$[h'[s,0], h'[s,1], h'[s,2], h'[s,3]] = [h[s,0], h[s,1], h[s,2], h[s,3]] \oplus [hn[s,0], hn[s,1], hn[s,2], hn[s,3]],$$

здесь: hn – элементы массива *Holatn*, h' – элементы результирующего массива.

Результат преобразования копируется в массив *Holat*.

6.4 Вопросы реализации

АШД реализуется с использованием ключей длиной в 256 и 512 bit.

В первом случае в криптографический модуль, который используется для шифрования, вводится 256 битный ключ. Этот ключ полностью используется как ключ шифрования k , а функциональный ключ k_f начального сеанса вычисляется как значение хэш-функции.

Во втором случае в криптографический модуль, который используется для шифрования, вводится 512 битный ключ. Первая 256 битная часть этого ключа используется как ключ шифрования k , вторая часть используется как функциональный ключ k_f первого сеанса.

В третьем случае в криптографический модуль, который используется для шифрования, не вводится никакой ключ. В качестве ключа шифрования используется ключ шифрования k , использованный на предыдущем этапе, а в качестве функционального ключа k_f используется хэшированное значение с использованием ключа шифрования функционального ключа k_{f-1} , использованного на предыдущем этапе.

В первом и втором случае, рассмотренных выше, обновленный функциональный ключ k_f очередного сеанса вычисляется как хэш-функция функционального ключа k_{f-1} предыдущего сеанса. В качестве ключа хэширования по правилу используется ключ шифрования, а программа вычисления функции хэширования включается в программное (аппаратное) обеспечение АШД. Период обновления функционального ключа определяется протоколом, использующим АШД в зависимости от режима использования АШД и от уровня конфиденциальности шифруемой информации.

Стандарт АШД может применяться в любых режимах блочных алгоритмов шифрования, установленных в международных стандартах. В данном стандарте описаны два базовых режима:

- электронная кодовая книга (**Elektron kod kitobi**);
- сцепление блоков (**ShifrBloklarni ilaktirish**).

В режиме «электронная кодовая книга» все блоки открытого текста шифруются независимо друг от друга посредством замены каждого блока открытого текста блоком шифртекста (каждого блока шифртекста блоком открытого текста).

Режим «сцепление блоков» использует механизм **обратной связи** и имеет специальный вход для вектора инициализации **IV**, обновляемого на каждом сеансе: в начале псевдокода зашифрования над текущим блоком открытого текста и над предыдущим блоком шифртекста выполняется

преобразование *Qo'shHolat(Holatn, Holat)*; в конце псевдокода
расшифрования над текущим блоком зашифрованного текста и
предыдущим блоком расшифрованного текста выполняется
преобразование *Qo'shHolat(Holatn, Holat)*.

Приложение А (справочное)

Контрольный пример

В данном приложении приведены процедуры зашифрования и расшифрования информации, состоящей из одного блока, представленного в виде чисел. Здесь числа приведены в шестнадцатеричной системе счисления.

Входной блок (Открытый текст):

30 31 32 33 34 35 36 37 38 39 41 42 43 44 45 46 30 31 32 33 34 35 36 37 38 39 41 42 43 44 45 46

Ключ шифрования K:

37 B6 0B BA 0A B1 60 CF DC 18 F5 0C DE E8 E0 45 30 B3 F8 AF 14 32 FE 51 1F BB 20 29 11 2F 21 43

Функциональный ключ Kf:

47 E7 69 46 69 C5 46 B6 FE 16 3A 89 B0 D8 96 D6 23 8B 23 15 32 C4 04 34 9C B0 C7 AA 81 3D F9 6D

Вектор инициализации IV:

26 54 BB 5F A3 75 D8 98 54 EA 48 9F 9A A8 84 16 FD 4D EB BD 9B 3B 40 33 48 29 F9 EE 52 34 C3 7A

*****Процесс шифрования*****

ShaklSeansKalitByte Kse

F8 7E 98 FF C6 66 0C 64 06 65 8B E7 29 A8 1D 65 B9 A6 26 D9 22 77 40 C3 44 68 45 38 FC 23 7C 28 B7 86
8B D0 CD 19 8F EA 31 AF BC 1B FD C9 2A 38 79 B1 6B C3 29 8D EB D5 F0 20 12 64 8C 4D 44 2F AE 7A
61 F6 91 E0 86 ED 8A 4C 75 AF EF 7A DF 6F 13 01 F6 98

Массив замены байтов Bsa1

B3 A5 19 29 7A AD 21 13 CB 37 1B A6 8F 36 BC 53 C7 09 46 26 E6 7F 5F 8D 2E 17 D6 B6 E8 8C 99 03 D7
82 56 8A 74 0E FD 65 34 06 D9 A7 A4 76 DA 78 C1 30 41 55 28 72 11 BE FB 5D 2D DD 16 83 0D 4D A0 02
D0 91 E0 32 D4 BB 14 2A 4C 68 93 FA 48 6C CF 4A 3E B1 BA 67 AA EB 89 5A FE 47 E2 18 86 6A A3 58
A2 BF 01 12 F2 6D 90 75 31 FF 6B 61 B7 2B CE 7D DE 1A C2 77 87 9A 81 F4 2F 5B 8B D1 84 40 A1 1D D8
63 EF A9 88 57 42 F6 3B F5 97 D3 6F 2C 3F 4F 04 AC 79 49 E4 9B 0F 54 D2 C8 95 51 3D 59 D5 B4 C5 07
24 6E 23 70 45 AE E5 F9 C9 39 8E DC 1C 7B 9C 15 25 7C F8 66 7E CD 33 5E 9D 0C B2 0B F3 E9 E7 35 F7
C4 EC 94 B9 43 B0 EA 69 64 CC 08 B5 A8 9E 71 85 96 05 4E ED 50 1E 22 EE C0 92 1F CA B8 52 3C 5C 44
C6 F0 C3 62 3A DF 10 F1 AF 98 60 4B E3 20 DB 80 BD 00 73 0A 38 27 E1 AB 9F FC

Массив замены байтов Bsa2

75 C8 2A B9 90 C0 D7 F8 F4 00 E6 28 3F 6F 46 D5 FE 8A 7C 1F 35 E3 A9 DE F0 E2 E5 D8 E4 23 16 7A AE
CF 18 2C 4A 67 F1 C5 80 95 10 42 9D D0 52 5F 20 59 1E B4 13 93 D4 7D 72 31 79 36 C7 4D 66 40 55 9E 3C
8F 6E 8C 29 BE 03 5C AB 74 87 9F 15 4C EB F5 81 2E 09 24 FD C2 06 2D CD 44 F7 ED A2 97 BA 47 9B
DA A1 96 EC 91 7F 01 8E EE 77 17 B6 D9 39 B0 6D 89 34 FF 05 57 AD 83 F3 9A 07 50 7B 04 43 C4 4B 5B
4E A5 F2 1A 22 E0 53 0D 5A B1 A4 9C 3B BC FB 84 AF 19 65 0C 58 C1 A8 FA 0B CB 76 92 4F C6 26 49
E8 33 11 71 37 6B 0A B7 69 5E 25 64 B8 45 68 5D 12 08 BB 32 D2 F9 3D 02 DB F6 D1 7E 88 14 B3 EA 60
78 8B 54 A3 FC 41 D6 73 CC 70 C3 61 AA BF 99 B2 38 C9 86 CE 8D 82 2B 6C 48 63 E1 A6 DF A0 3E 2F 62
BD EF 6A 94 3A 0E 98 B5 D3 E7 30 51 85 E9 DC 1B 27 AC 1D 0F 21 56 1C CA DD A7

Начальный этапный ключ Ke

F8 7E 98 FF C6 66 0C 64 06 65 8B E7 29 A8 1D 65 B9 A6 26 D9 22 77 40 C3 44 68 45 38 FC 23 7C 28

Qo'shHolat (Holat, Holatn)

16 65 89 6C 97 40 EE AF 6C D3 09 DD D9 EC C1 50 CD 7C D9 8E AF 0E 76 04 70 10 B8 AC 11 70 86 3C

1-этап

Этапный ключ Ke[1]:

F8 7E 98 FF C6 66 0C 64 06 65 8B E7 29 A8 1D 65 B9 A6 26 D9 22 77 40 C3 44 68 45 38 FC 23 7C 28

Qo'shBosqichKalit (Holat, Ke)

EE 1B 11 93 51 26 E2 CB 6A B6 82 3A F0 44 DC 35 74 DA FF 57 8D 79 36 C7 34 78 FD 94 ED 53 FA 14
Aralash(Holat, Ks)

A2 33 69 FF 46 85 01 70 A5 39 ED F5 40 EB 1F 71 3A 87 62 7D D5 29 D0 3F FC 50 A3 5C 29 02 AF DA
Sur(Holat)

7D 29 50 D0 A3 3F A2 02 33 AF 5C 46 A5 85 69 DA FF 40 39 01 ED 70 3A EB 87 1F F5 D5 FC 29 62 71
BaytAlmash(Holat, Ba)

D1 06 CF B5 07 4D C5 19 55 DC E2 D4 6E A9 75 1E FC A0 5D A5 F1 CE 2D DF 57 03 80 96 E1 06 A2 7D

2-этап**Этапный ключ Ke[2]:**

5F 39 4D 40 EB 2D CD 31 36 C9 13 BA 06 1A 23 42 29 C7 E1 1B E1 45 BC 34 5E 86 68 CC 7F 51 8D 7D

Qo'shBosqichKalit (Holat, Ke)

8E 3F 82 F5 EC 60 08 28 63 15 F1 6E 68 B3 56 5C D5 67 BC BE 10 8B 91 EB 09 85 E8 5A 9E 57 2F 00

Aralash(Holat, Ks)

63 B4 47 20 F0 03 C8 4C 49 6A A4 BC 2E 2E FD 41 CD 05 3E 8E 5A 98 6A 5D 01 50 FF CA 44 3E 58 A0

Sur(Holat)

8E 44 50 6A FF 5D 63 3E B4 58 CA F0 49 03 47 A0 20 2E 6A C8 A4 4C CD 2E 05 FD BC 5A 01 98 3E 41

BaytAlmash(Holat, Ba)

A4 6E EB 8E A7 ED DA 66 12 06 41 30 5C B9 BE 4F AE 52 8E A3 E8 87 CC 52 C0 CA DB CD C8 58 66 9E

3-этап**Этапный ключ Ke[3]:**

9D D0 30 D1 1A 11 4E 3F 08 DF 0A 2D E1 A2 F4 33 46 63 FA 8C 6B EF 06 FF 72 4A 8E 1E 6C 5A F0 CA

Qo'shBosqichKalit (Holat, Ke)

39 BE DB 5F BD FC 94 59 1A D9 4B 1D BD 1B 4A 7C E8 31 74 2F 83 68 CA AD B2 80 55 D3 A4 02 96 54

Aralash(Holat, Ks)

F5 18 4D 05 9C 97 BF 58 28 E6 5A D3 0E 5E 7D E0 3E 1B 20 6B 6B A0 B4 45 80 E5 A7 99 DC DF 63 4A

Sur(Holat)

6B DC E5 B4 A7 45 F5 DF 18 63 99 9C 28 97 4D 4A 05 0E E6 BF 5A 58 3E 5E 1B 7D D3 6B 80 A0 20 E0

BaytAlmash(Holat, Ba)

FF EE 44 25 70 32 80 1F 2E BF 54 95 34 9B FA 4C AD BC C6 0B FE 89 0D 86 B6 D1 71 FF A1 D5 D7 CA

4-этап**Этапный ключ Ke[4]:**

51 6F 0D 17 A1 9A 33 1F D4 63 5F 78 37 FB 92 54 70 F3 62 D7 86 53 1B D7 AB E0 40 24 C9 18 9A 88

Qo'shBosqichKalit (Holat, Ke)

AE 81 49 32 D1 A8 B3 00 FA DC 0B ED 03 60 68 18 DD 4F A4 DC 78 DA 16 51 1D 31 31 DB 68 CD 4D 42

Aralash(Holat, Ks)

51 46 36 51 35 35 16 AC D7 18 88 F2 10 DB EB EB 7B 6D 3A A4 80 69 33 91 D1 7D 40 9F 96 0B 11 B8

Sur(Holat)

A4 96 7D 33 40 91 51 0B 46 11 9F 35 D7 35 36 B8 51 10 18 16 88 AC 7B DB 6D EB F2 80 D1 69 3A EB

BytAlmash(Holat, Ba)

E8 65 50 B4 55 BC F5 28 29 8A 92 93 86 93 D4 D2 F5 FE F0 A9 22 69 9A 2B 17 0E 85 43 AA 01 79 0E

5-этап**Этапный ключ Ke[5]:**

FB C1 BF DC 92 A3 87 9B 16 BC 32 98 DE BD 5F 02 01 26 48 C4 D4 42 FA E7 A6 1F 69 1E 08 6E D8 A4

Qo'shBosqichKalit (Holat, Ke)

13 A4 EF 68 C7 1F 72 B3 3F 36 A0 0B 58 2E 8B D0 F4 D8 B8 6D F6 2B 60 CC B1 11 EC 5D A2 6F A1 AA

Aralash(Holat, Ks)

CC AB 22 6A 1E 39 A5 AA 1C FF 76 88 E1 F9 16 E1 24 10 86 E9 B6 35 DE DC 17 8F BA 63 80 B9 25 2A

Sur(Holat)

E9 80 8F DE BA DC CC B9 AB 25 63 1E 1C 39 22 2A 6A E1 FF A5 76 AA 24 F9 10 16 88 B6 17 35 86 E1

BaytAlmash(Holat, Ba)

62 A1 2C 92 33 EE 69 CD F9 0E BF 99 E8 5D 56 D9 31 B8 FC 6E 87 E5 74 0A C7 5F 42 F8 8D 72 88 B8

6-этап**Этапный ключ Ke[6]:**

94 C6 F5 EA F8 10 09 32 46 26 A2 17 D7 3D 30 FB 48 F0 43 76 C5 26 9A D7 F7 BD 6E B7 89 80 FB 4C

Qo'shBosqichKalit (Holat, Ke)

F6 67 D9 78 CB FE 60 FF BF 28 1D 8E 3F 60 66 22 79 48 BF 18 42 C3 4E DD 30 E2 2D 4F 04 F2 73 F4

75 STANDART AGENTLIGI
 STANDARTLASHTIRISH
 VA DAVLAT NAZORATI
 BOSHQARUVCHI

O'z DSt 1105:2009

Aralash(Holat, Ks)

6F 08 B6 76 E8 FB C1 7C C8 84 72 DF AA 4D 7B 3B 47 A9 04 DA 1A 94 B5 DD 26 71 EA 4D 6E 7A 51 EA
Sur(Holat)

DA 6E 71 B5 EA DD 6F 7A 08 51 4D E8 C8 FB B6 EA 76 AA 84 C1 72 7C 47 4D A9 7B DF 1A 26 94 04 3B

BaytAlmash(Holat, Ba)

82 B6 B0 08 3A 48 D9 F3 F4 F5 9F 6A A3 56 BB 3A 05 0A 4E 14 6D 07 BE 9F 6B 9A E1 E5 F1 AF 90 36

7-этап

Этапный ключ Ke[7]:

10 BE B9 E9 87 DA 47 82 1B B6 29 31 D6 BF BD EB 7D BC 4C 07 DA 63 E1 FA 63 FF 19 98 31 90 19 96

Qo'shBosqichKalit (Holat, Ke)

92 08 09 E1 BD 92 9E 71 EF 43 B6 5B 75 E9 06 D1 78 B6 02 13 B7 64 5F 65 08 65 F8 7D C0 3F 89 A0

Aralash(Holat, Ks)

DE FA 93 37 4A 61 AF 5E 37 F7 31 43 5E 63 9C E5 C8 9A A4 83 85 6C E3 C3 70 AA D0 3D 7A 5E 62 B0

Sur(Holat)

83 7A AA E3 D0 C3 DE 5E FA 62 3D 4A 37 61 93 B0 37 5E F7 AF 31 5E C8 63 9A 9C 43 85 70 6C A4 E5

BaytAlmash(Holat, Ba)

63 2F E5 3C B5 35 92 86 38 A2 83 4C BE 58 AC 1C BE 86 00 DC 30 86 B9 BF D2 95 91 A9 CE 6B 24 44

8-этап

Этапный ключ Ke[8]:

49 8E B5 FD EF 5B ED E2 60 3E D3 1F 0F D3 1F F8 CC C1 8C 80 CC B1 7C E5 35 03 AC B7 34 C4 DB 24

Qo'shBosqichKalit (Holat, Ke)

2A A1 50 C1 5A 6E 7F 64 58 9C 50 53 B1 8B B3 E4 72 47 8C 5C FC 37 C5 5A E7 96 3D 1E FA AF FF 60

Aralash(Holat, Ks)

C8 1C 6B EE DD 0E FC 17 86 29 28 69 45 66 4A 32 58 CC AB 84 AA A1 D9 18 61 2B 59 E0 6C 55 FB D6

Sur(Holat)

84 6C 2B D9 59 18 C8 55 1C FB E0 DD 86 0E 6B D6 EE 45 29 FC 28 17 58 66 CC 4A 69 AA 61 A1 AB 32

BaytAlmash(Holat, Ba)

4E 77 42 8D 2D F0 A3 24 E4 56 A6 48 F2 46 EE C9 D3 8C 95 1C 80 DE 06 EC 73 AB 01 0A 47 C6 B7 1E

Ke на последнем шаге

98 F8 7E 98 FF C6 66 0C 64 06 65 8B E7 29 A8 1D 65 B9 A6 26 D9 22 77 40 C3 44 68 45 38 FC 23 7C

Qo'shBosqichKalit (Holat, Ke)

D6 8F 3C 15 D2 36 C5 28 80 50 C3 C3 15 6F 46 D4 B6 35 33 3A 59 FC 71 AC B0 EF 69 4F 7F 3A 94 62

Aralash(Holat, Ks)

39 E2 2F F6 3F 77 18 65 DE 95 D4 C9 F6 B3 FE C7 6C D1 2D 1C 5F D0 DE 1A 2E BA DB D5 A3 B2 FA B8

***** **Результат шифрования** *****

39 E2 2F F6 3F 77 18 65 DE 95 D4 C9 F6 B3 FE C7 6C D1 2D 1C 5F D0 DE 1A 2E BA DB D5 A3 B2 FA B8

Входной блок (Зашифрованный текст):

39 E2 2F F6 3F 77 18 65 DE 95 D4 C9 F6 B3 FE C7 6C D1 2D 1C 5F D0 DE 1A 2E BA DB D5 A3 B2 FA B8

Ключ шифрования K:

37 B6 0B BA 0A B1 60 CF DC 18 F5 0C DE E8 E0 45 30 B3 F8 AF 14 32 FE 51 1F BB 20 29 11 2F 21 43

Функциональный ключ Kf:

47 E7 69 46 69 C5 46 B6 FE 16 3A 89 B0 D8 96 D6 23 8B 23 15 32 C4 04 34 9C B0 C7 AA 81 3D F9 6D

Вектор инициализации IV:

26 54 BB 5F A3 75 D8 98 54 EA 48 9F 9A A8 84 16 FD 4D EB BD 9B 3B 40 33 48 29 F9 EE 52 34 C3 7A

***** **Процесс расшифрования** *****

ShaklSeansKalitByte Kse

F8 7E 98 FF C6 66 0C 64 06 65 8B E7 29 A8 1D 65 B9 A6 26 D9 22 77 40 C3 44 68 45 38 FC 23 7C 28 B7 86

8B D0 CD 19 8F EA 31 AF BC 1B FD C9 2A 38 79 B1 6B C3 29 8D EB D5 F0 20 12 64 8C 4D 44 2F AE 7A

61 F6 91 E0 86 ED 8A 4C 75 AF EF 7A DF 6F 13 01 F6 98

Массив замены байтов Bsa1

F7 64 41 1F 92 D6 29 A3 CF 11 F9 BF BD 3E 25 98 EC 36 65 07 48 B3 3C 19 5D 02 73 0A B0 81 DA DF F3
 06 DB A6 A4 B4 13 FB 34 03 49 6F 8F 3A 18 7A 31 6A 45 BA 28 C3 0D 09 FA AD EA 8A E3 9E 52 90 7F 32
 88 C9 E5 A8 12 5B 4E 95 51 F1 4A 3F D7 91 D9 9D E2 0F 99 33 22 87 61 9F 59 7B E4 39 BB 16 F0 6D E9 83
 CD 27 B7 55 4B CC 5F 6C 4F 67 A5 8E A7 D3 35 F8 24 69 2D 75 2F 94 04 B1 B5 71 B8 15 F5 78 21 3D 7E
 D4 5E 76 86 58 23 7C 1D 17 AE 0C 68 43 DE 4C C7 9C D5 8C EF 1E 77 97 B2 BC D2 FE 40 80 62 60 2C 01
 0B 2B D1 85 56 FD 93 05 A9 EE CA 53 BE 00 A1 D0 1B 6E E1 C8 54 47 0E F6 37 63 DD 30 74 E8 C5 A2 E6
 10 9B AC E0 08 CE B9 70 50 42 7D 9A 8D 46 A0 1A 20 82 2A 2E F4 AF 3B 72 EB 44 FC 5C F2 96 AA 14
 C2 1C C1 CB 57 C6 D8 DC 84 E7 ED 66 C0 79 8B 89 C4 B6 AB 4D 38 FF 26 5A 6B

Массив замены байтов Bsa2

09 69 BB 48 7F 76 58 7C B5 54 AA 9C 97 8B EB F9 2A A6 B4 34 C1 4E 1E 6D 22 95 87 F5 FC F8 32 13 30
 FA 88 1D 55 AE A2 F6 0B 46 02 DB 23 59 53 E4 F0 39 B7 A5 74 14 3B A8 D5 70 EA 90 42 BA E3 0C 3F CA
 2B 80 5B B1 0E 61 DD A3 24 82 4F 3D 84 A0 7D F1 2E 8A C7 40 FB 77 98 31 8C 83 49 B3 AD 2F C4 D0 E5
 DE AF 96 3E 25 B2 AC E8 A9 DC 72 44 0D CE A7 38 CC 4B 00 9E 6C C5 3A 1F 7E 12 37 BF 68 28 52 DA
 79 93 F2 D7 4C C0 73 11 C6 45 D9 6A 43 04 67 9F 35 E9 29 65 5F EC D3 7B 62 8F 2C 41 4D E2 64 5E C8 8E
 85 E0 FF 9A 16 D1 4A F7 78 20 94 71 8D D4 C2 33 ED 6E AB B0 03 60 B6 91 E6 47 D2 05 99 57 CF 81 27
 A1 3C 01 D6 FD 9D CD 5A D8 21 2D BE B8 EE 36 0F CB 06 1B 6F 63 BC F4 FE 17 E1 89 DF 19 15 1C 1A
 0A EF A4 F3 C3 50 66 5D 6B E7 18 26 86 7A 08 51 BD 5C 07 B9 9B 92 C9 56 10 75

Начальный этапный ключ Ke

98 F8 7E 98 FF C6 66 0C 64 06 65 8B E7 29 A8 1D 65 B9 A6 26 D9 22 77 40 C3 44 68 45 38 FC 23 7C

Aralash(Holat, Ks)

D6 8F 3C 15 D2 36 C5 28 80 50 C3 C3 15 6F 46 D4 B6 35 33 3A 59 FC 71 AC B0 EF 69 4F 7F 3A 94 62

Qo'shBosqichKalit(Holat, Ke)

4E 77 42 8D 2D F0 A3 24 E4 56 A6 48 F2 46 EE C9 D3 8C 95 1C 80 DE 06 EC 73 AB 01 0A 47 C6 B7 1E

1-этан**Этапный ключ Ke[1]:**

49 8E B5 FD EF 5B ED E2 60 3E D3 1F 0F D3 1F F8 CC C1 8C 80 CC B1 7C E5 35 03 AC B7 34 C4 DB 24

BaytAlmash(Holat, Ba)

84 6C 2B D9 59 18 C8 55 1C FB E0 DD 86 0E 6B D6 EE 45 29 FC 28 17 58 66 CC 4A 69 AA 61 A1 AB 32

Sur(Holat)

C8 1C 6B EE DD 0E FC 17 86 29 28 69 45 66 4A 32 58 CC AB 84 AA A1 D9 18 61 2B 59 E0 6C 55 FB D6

Aralash(Holat, Ks)

2A A1 50 C1 5A 6E 7F 64 58 9C 50 53 B1 8B B3 E4 72 47 8C 5C FC 37 C5 5A E7 96 3D 1E FA AF FF 60

Qo'shBosqichKalit(Holat, Ke)

63 2F E5 3C B5 35 92 86 38 A2 83 4C BE 58 AC 1C BE 86 00 DC 30 86 B9 BF D2 95 91 A9 CE 6B 24 44

2-этан**Этапный ключ Ke[2]:**

10 BE B9 E9 87 DA 47 82 1B B6 29 31 D6 BF BD EB 7D BC 4C 07 DA 63 E1 FA 63 FF 19 98 31 90 19 96

BaytAlmash(Holat, Ba)

83 7A AA E3 D0 C3 DE 5E FA 62 3D 4A 37 61 93 B0 37 5E F7 AF 31 5E C8 63 9A 9C 43 85 70 6C A4 E5

Sur(Holat)

DE FA 93 37 4A 61 AF 5E 37 F7 31 43 5E 63 9C E5 C8 9A A4 83 85 6C E3 C3 70 AA D0 3D 7A 5E 62 B0

Aralash(Holat, Ks)

92 08 09 E1 BD 92 9E 71 EF 43 B6 5B 75 E9 06 D1 78 B6 02 13 B7 64 5F 65 08 65 F8 7D C0 3F 89 A0

Qo'shBosqichKalit(Holat, Ke)

82 B6 B0 08 3A 48 D9 F3 F4 F5 9F 6A A3 56 BB 3A 05 0A 4E 14 6D 07 BE 9F 6B 9A E1 E5 F1 AF 90 36

3-этан**Этапный ключ Ke[3]:**

94 C6 F5 EA F8 10 09 32 46 26 A2 17 D7 3D 30 FB 48 F0 43 76 C5 26 3A D7 F7 BD 6F B7 89 80 FB 4C

BaytAlmash(Holat, Ba)

DA 6E 71 B5 EA DD 6F 7A 08 51 4D E8 C8 FB B6 EA 76 AA 84 C1 72 7C 47 4D A9 7B DF 1A 26 94 04 3B

Sur(Holat)

6F 08 B6 76 E8 FB C1 7C C8 84 72 DF AA 4D 7B 3B 47 A9 04 DA 1A 94 B5 DD 26 71 EA 4D 6E 7A 51 EA

Aralash(Holat, Ks)

F6 67 D9 78 CB FE 60 FF BF 28 1D 8E 3F 60 66 22 79 48 BF 18 42 C3 4E DD 30 E2 2D 4F 04 F2 73 F4

STANDARD AGENTLIGI
 TANCARTLASHTIRISH
 KUYOVIQLASHTIRISH
 O'ZBOK AVLAT NAZORATI
 QOSHWARMAHI

O'z DSt 1105:2009

Qo'shBosqichKalit(Holat, Ke)

62 A1 2C 92 33 EE 69 CD F9 0E BF 99 E8 5D 56 D9 31 B8 FC 6E 87 E5 74 0A C7 5F 42 F8 8D 72 88 B8

4-этап

Этапный ключ Ke[4]:

FB C1 BF DC 92 A3 87 9B 16 BC 32 98 DE BD 5F 02 01 26 48 C4 D4 42 FA E7 A6 1F 69 1E 08 6E D8 A4

BaytAlmash(Holat, Ba)

E9 80 8F DE BA DC CC B9 AB 25 63 1E 1C 39 22 2A 6A E1 FF A5 76 AA 24 F9 10 16 88 B6 17 35 86 E1

Sur(Holat)

CC AB 22 6A 1E 39 A5 AA 1C FF 76 88 E1 F9 16 E1 24 10 86 E9 B6 35 DE DC 17 8F BA 63 80 B9 25 2A

Aralash(Holat, Ks)

13 A4 EF 68 C7 1F 72 B3 3F 36 A0 0B 58 2E 8B D0 F4 D8 B8 6D F6 2B 60 CC B1 11 EC 5D A2 6F A1 AA

Qo'shBosqichKalit(Holat, Ke)

E8 65 50 B4 55 BC F5 28 29 8A 92 93 86 93 D4 D2 F5 FE F0 A9 22 69 9A 2B 17 0E 85 43 AA 01 79 0E

5-этап

Этапный ключ Ke[5]:

51 6F 0D 17 A1 9A 33 1F D4 63 5F 78 37 FB 92 54 70 F3 62 D7 86 53 1B D7 AB E0 40 24 C9 18 9A 88

BaytAlmash(Holat, Ba)

A4 96 7D 33 40 91 51 0B 46 11 9F 35 D7 35 36 B8 51 10 18 16 88 AC 7B DB 6D EB F2 80 D1 69 3A EB

Sur(Holat)

51 46 36 51 35 35 16 AC D7 18 88 F2 10 DB EB EB 7B 6D 3A A4 80 69 33 91 D1 7D 40 9F 96 0B 11 B8

Aralash(Holat, Ks)

AE 81 49 32 D1 A8 B3 00 FA DC 0B ED 03 60 68 18 DD 4F A4 DC 78 DA 16 51 1D 31 31 DB 68 CD 4D 42

Qo'shBosqichKalit(Holat, Ke)

FF EE 44 25 70 32 80 1F 2E BF 54 95 34 9B FA 4C AD BC C6 0B FE 89 0D 86 B6 D1 71 FF A1 D5 D7 CA

6-этап

Этапный ключ Ke[6]:

9D D0 30 D1 1A 11 4E 3F 08 DF 0A 2D E1 A2 F4 33 46 63 FA 8C 6B EF 06 FF 72 4A 8E 1E 6C 5A F0 CA

BaytAlmash(Holat, Ba)

6B DC E5 B4 A7 45 F5 DF 18 63 99 9C 28 97 4D 4A 05 0E E6 BF 5A 58 3E 5E 1B 7D D3 6B 80 A0 20 E0

Sur(Holat)

F5 18 4D 05 9C 97 BF 58 28 E6 5A D3 0E 5E 7D E0 3E 1B 20 6B 6B A0 B4 45 80 E5 A7 99 DC DF 63 4A

Aralash(Holat, Ks)

39 BE DB 5F BD FC 94 59 1A D9 4B 1D BD 1B 4A 7C E8 31 74 2F 83 68 CA AD B2 80 55 D3 A4 02 96 54

Qo'shBosqichKalit(Holat, Ke)

A4 6E EB 8E A7 ED DA 66 12 06 41 30 5C B9 BE 4F AE 52 8E A3 E8 87 CC 52 C0 CA DB CD C8 58 66 9E

7-этап

Этапный ключ Ke[7]:

5F 39 4D 40 EB 2D CD 31 36 C9 13 BA 06 1A 23 42 29 C7 E1 1B E1 45 BC 34 5E 86 68 CC 7F 51 8D 7D

BaytAlmash(Holat, Ba)

8E 44 50 6A FF 5D 63 3E B4 58 CA F0 49 03 47 A0 20 2E 6A C8 A4 4C CD 2E 05 FD BC 5A 01 98 3E 41

Sur(Holat)

63 B4 47 20 F0 03 C8 4C 49 6A A4 BC 2E 2E FD 41 CD 05 3E 8E 5A 98 6A 5D 01 50 FF CA 44 3E 58 A0

Aralash(Holat, Ks)

8E 3F 82 F5 EC 60 08 28 63 15 F1 6E 68 B3 56 5C D5 67 BC BE 10 8B 91 EB 09 85 E8 5A 9E 57 2F 00

Qo'shBosqichKalit(Holat, Ke)

D1 06 CF B5 07 4D C5 19 55 DC E2 D4 6E A9 75 1E FC A0 5D A5 F1 CE 2D DF 57 03 80 96 E1 06 A2 7D

8-этап

Этапный ключ Ke[8]:

F8 7E 98 FF C6 66 0C 64 06 65 8B E7 29 A8 1D 65 B9 A6 26 D9 22 77 40 C3 44 68 45 38 FC 23 7C 28

BaytAlmash(Holat, Ba)

7D 29 50 D0 A3 3F A2 02 33 AF 5C 46 A5 85 69 DA FF 40 39 01 ED 70 3A EB 87 1F F5 D5 FC 29 62 71

Sur(Holat)

A2 33 69 FF 46 85 01 70 A5 39 ED F5 40 EB 1F 71 3A 87 62 7D D5 29 D0 3F FC 50 A3 5C 29 02 AF DA

Aralash(Holat, Ks)

EE 1B 11 93 51 26 E2 CB 6A B6 82 3A F0 44 DC 35 74 DA FF 57 8D 79 36 C7 34 78 FD 94 ED 53 FA 14

Qo'shBosqichKalit(Holat, Ke)

16 65 89 6C 97 40 EE AF 6C D3 09 DD D9 EC C1 50 CD 7C D9 8E AF 0E 76 04 70 10 B8 AC 11 70 86 3C

Holatn

26 54 BB 5F A3 75 D8 98 54 EA 48 9F 9A A8 84 16 FD 4D EB BD 9B 3B 40 33 48 29 F9 EE 52 34 C3 7A

Qo'shHolat(Holat, Holatn)

30 31 32 33 34 35 36 37 38 39 41 42 43 44 45 46 30 31 32 33 34 35 36 37 38 39 41 42 43 44 45 46

******* Результат расшифрования *******

30 31 32 33 34 35 36 37 38 39 41 42 43 44 45 46 30 31 32 33 34 35 36 37 38 39 41 42 43 44 45 46

ЎЗСТАНДАРТ АГЕНТЛИГИ
СТАНДАРТИЛАСHTИРИШ,
МУВОФИҚЛАСHTИРИШ
ВА ДАВЛАТ НАЗОРАТИ
ЎЎЎЎЎЎЎЎ

УДК:

ОКС 35.040

Группа П85

Ключевые слова: алгоритм шифрования данных, шифртекст, зашифрования, расшифрование, ключ, сеансовый ключ, функциональный ключ.

Заместитель директора ГУП
«UNICON.UZ»


Х.П. Хасанов

Главный научный сотрудник
ГУП «UNICON.UZ» д.т.н., проф.


П.Ф. Хасанов

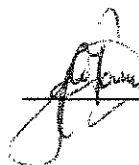
Начальник Научно-
исследовательского отдела
средств защиты информации


О.Х. Расулов

Нальник Научно-
исследовательского отдела
криптографии к.т.н.


О.П. Ахмедова

Заместитель начальника Научно-
исследовательского отдела
криптографии


Ж.Д. Мукимов

Начальник лаборатории
криптоанализа


А.Б. Давлатов

Нормоконтроль


Н. К. Травина

СОГЛАСОВАНО

Служба национальной
безопасности Республики Узбекистан

Письмо от 27 марта 2007 года
№39/657

O'ZSTANDART AGENTLIGI
STANDARTLASHTIRISH,
MUVOFIQLASHTIRISH
VA DAVLAT KAZORATI
BESHARMASI

11/11/2024 10:11:11 AM

11/11/2024 10:11:11 AM