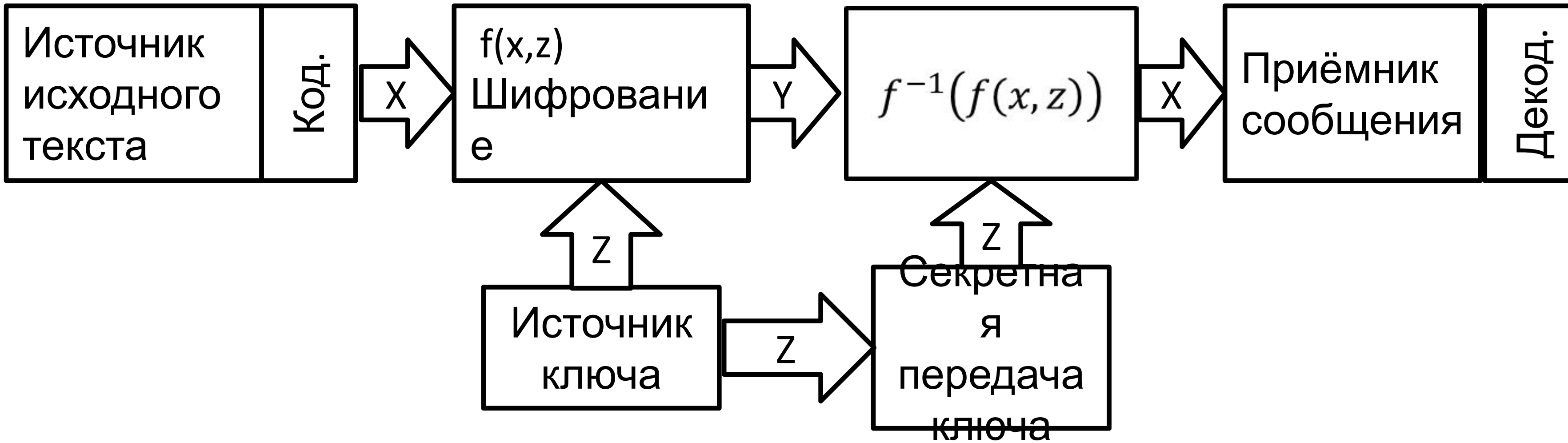
Классическая криптография

1. Криптографическая система с одним ключом (общим для шифрования и расшифрования)



X — числовое представление (код) исходного текста

Y — шифрограмма

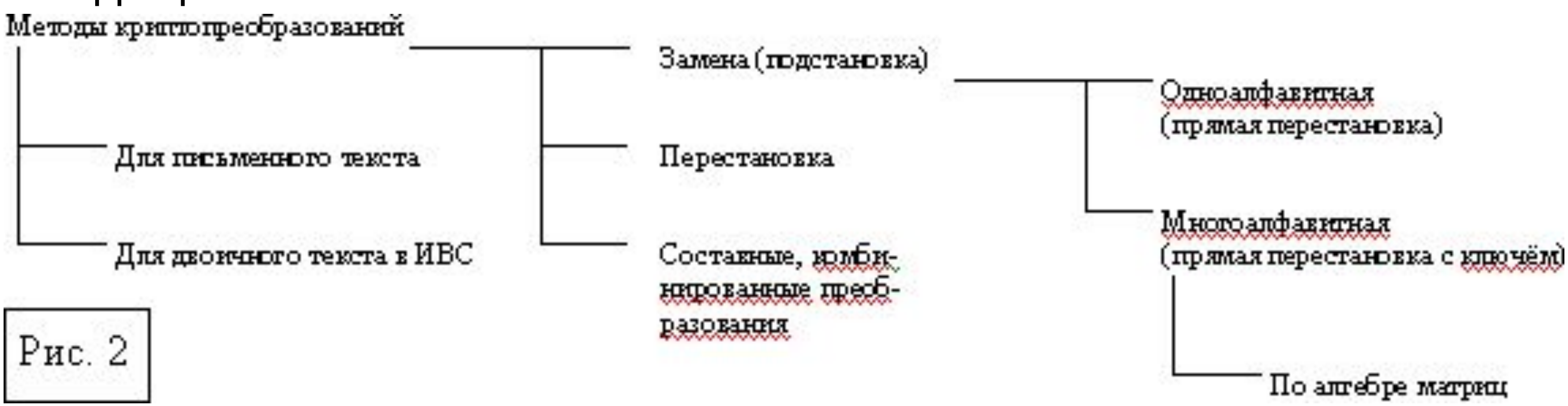


Рис. 2

Шифрование заменой (подстановками)
Моно(одно)алфавитная замена — самый простой способ прямой замены. Составляется таблица прямой замены букв шифруемого текста другими буквами данного алфавита.

Таблица замены
Знаки в таблице шифрования не должны повторяться, т.е. таблица замены должна представлять полную перестановку алфавита (когда все буквы подверглись перестановке). После замены шифротекст для удобства работы с ним разбивается на равновеликие группы. В шифре Цезаря таблица замены есть алфавит сдвинутый в кольцо на 3 позиции.
Одноалфавитный шифр имеет низкую стойкость. Сравнительно легко взламывается, т.к. имеет те же статистические характеристики частости букв в шифрограмме, что и в исходном (открытом) тексте. При достаточной длине шифротекста он раскрывается статистическим криптоанализом.

Многотабличная замена. Буквенная ключевая последовательность.
Многоалфавитный шифр более стойкий. Например, таблица Вижинера. Это квадратная матрица N*N, где N — количество символов алфавита.
Первая строка матрицы — исходный алфавит. Следующие — кольцевой сдвиг алфавита на одну букву. Для шифрования задаётся слово из K букв (буквенный ключ). Из таблицы Вижинера выписывается рабочая подтаблица (K+1)*N. Первая строка — исходный алфавит.
Следующие строки — алфавиты, начинающиеся с очередных букв ключа.
Процедура шифрования:

- под каждой буквой шифруемого текста записываются буквы ключа, повторяя его необходимое число раз;
- Замена букв производится по подматрице и затем шифротекст разбивается на группы, например по 5 знаков.

Расшифрование шифротекста происходит в обратной последовательности. Ключ следует периодически или для каждого файла менять.

Заменив буквы числами, получим цифровую шифрограмму. Статистические характеристики букв шифротекста уже иные, чем у исходного текста, т.к. в разных местах текста данная буква будет шифроваться разными буквами.

Проблемы ключа.

При коротком ключе шифрование не надёжно (злоумышленнику для раскрытия по крайней мере надо перехватить количество знаков в шифровке равное 20 длинам ключа). Длинный же ключ запомнить трудно (если он ещё и не имеет лингвосмысла), а запись его на бумаге может быть похищена. Ключ может вводиться пользователем с терминала или храниться в ЗУ в зашифрованном виде.

Одноалфавитные и многоалфавитные подстановки можно представить общей формулой, рассматривая её как задачу современной алгебры, т.к. между *N* знаками алфавита и набором положительных целых чисел 0, 1, 2, ..., (*N* - 1) устанавливается произвольное однозначное соответствие, то при сложении и вычитании по модулю *N* эти числа формируют алгебраическое кольцо и однозначное обратное преобразование.

Шифрование: $y_{i=}(x_i + z_i)$ modN

Расшифрование: x_i=($y_i - z_i$)

Если $z_i = const$, то имеем одноалфавитную подстановку. Для нее общую форму моржно расширить:

$y_i = (a * x_i + z) \; mod \; N$, при $z_i = const$

Где:

$y_i -$ числовой код букв шифр

$x_i -$ числовой код букв исходного текста

N — размер алфавита

a — десятичный коэффициент

z — коэффициент сдвиг

При *a* = 1, *z* = 3(4), *N* = 27 получаем код Цезаря с алфавитом, например:

A B C D E F G H I J K L M N O P Q R S T U V W X Y Z в (пробел) 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25 26 27 0 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25 26

Отметим, что две одноалфавитные замены подряд не увеличивают стойкости шифра, т.к. эквивалентны одной (суммарной) замене. Например если первая замена была с *z* = 3 (формула 2), а вторая с *z* = 5, то получим результирующую одну замену с *z* = 8 .

Числовая ключевая последовательность

Если z выбирается из последовательности z_1, z_2, K, z_n то имеем многоалфавитную подстановку с периодом ключа $\{z = z_1, z_2, K, z_K\}$ равным К.

Если в многоалфавитной подстановке:
Число знаков в ключе больше (или равно) числу шифруемых (исходных) знаков текста и знаки в ключе распределены случайно
Ключ используется только один раз
Исходный текст (или его часть) неизвестен злоумышленнику (криптоаналитику), то зашифрованный текст будет нераскрываем и называется *системой (схемой) Вернама.*
Именно для этих условий Шеннон Э. и доказал нераскрываемость шифра.
Если криптоаналитику известен (или предполагается известным) отрезок исходного текста заведомо в несколько раз длиннее ключа, то ключ будет раскрыт вычитанием из шифрограммы известного отрезка текста
z = {y - x)mo6.N
перебором знакоместа шифрограммы для начала серии вычитаний. Появление периодической структуры результата и есть признак вскрытия ключа.
С этой позиции рассмотрим известное усовершенствование таблицы Вижинера. Во всех строках, кроме первой буквы алфавита располагаются в произвольном порядке (а не сдвигаются), т.е. используется множество перестановок букв алфавита. Число перестановок *P(N) = N\,* P(27) = 1.088 -10^{28}. Однако, из этого множества не так много подходящих, нужны только «полные» перестановки, т.е. такие которые затронули все буквы алфавита. Вот из этого множества и выбираем 10 (не считая первой) перестановок.

Нумеруем их натуральными числами 0, 1, ..., 9.
В качестве ключа берём случайный (практически псевдослучайный) ряд чисел бесконечной длины или длины не меньшей, чем количество букв исходном тексте.
Например*: π* =3.14159265358979323846..., е =2.71828182845904523536...
При длине ключа равной длине текста статистическая закономерность букв исходного алфавита, по - видимому, полностью маскируется.
Однако это всё таки всего 10-алфавитный ключ, правда алфавиты чередуются на всём протяжении текста в «случайном» порядке, а не повторяются группами по слову текстового ключа. Стойкость шифра несколько усиливается.
Формула (1) даст ещё лучшую стойкость, если в ней в качестве последовательности ключа взять «случайные» (например, по таблице случайных чисел 2-хразрядных десятичных) из множества 0, 1, 2, ..., *(N* -1).
В этом случае получим 27-алфавитную подстановку со «случайным» чередованием алфавитов на всём протяжении исходного текста.

Шифрование с использованием алгебры матриц (частный случай перестановок).
Считается, что этим методом можно получить надёжное закрытие информации.
Например, применим правило умножения матрицы на вектор

$$\begin{pmatrix} a_{11} & a_{12} & a_{13} \\ a_{21} & a_{22} & a_{23} \\ a_{31} & a_{32} & a_{33} \end{pmatrix} * \begin{pmatrix} b_1 \\ b_2 \\ b_3 \end{pmatrix} = \begin{pmatrix} a_{11} * b_1 + & a_{12} * b_2 + & a_{13} * b_3 \\ a_{21} * b_1 + & a_{22} * b_2 + & a_{23} * b_3 \\ a_{31} * b_{1+} & a_{32} * b_2 + & a_{33} * b_3 \end{pmatrix} = \begin{pmatrix} c_1 \\ c_2 \\ c_3 \end{pmatrix}$$

Здесь матрицу $[a_{ii}]$ будем брать за основу (ключ) шифрования. Матрицу [b_i] — как символы исходного текста. Матрицу столбец $[c_i\ ]$ — как символы шифрованног текста.
Пример. Представляем ключ матрицы, например, 3-го порядка

$$\begin{pmatrix} 14 & 8 & 3 \\ 8 & 5 & 2 \\ 3 & 2 & 1 \end{pmatrix}$$

A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z
0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25

Рассмотрим пример на слове Data.

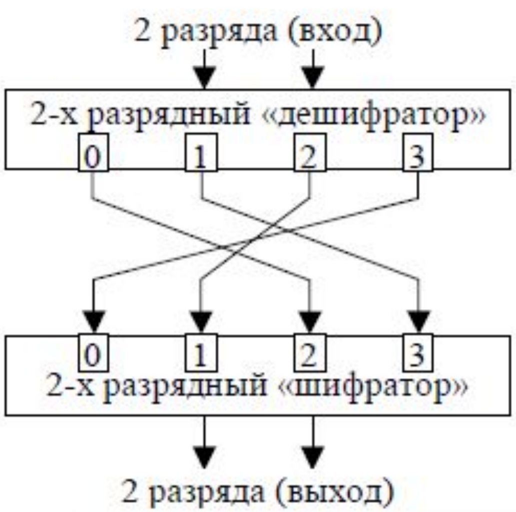
$$\begin{pmatrix} 14 & 8 & 3 \\ 8 & 5 & 2 \\ 3 & 2 & 1 \end{pmatrix} \begin{pmatrix} 3 \\ 0 \\ 19 \end{pmatrix} \begin{matrix} D \\ A \\ T \end{matrix} = \begin{pmatrix} 14 * 3 + & 8 * 0 + & 3 * 19 \\ 8 * 3 + & 5 * 0 + & 2 * 19 \\ 3 * 3 + & 2 * 0 + & 1 * 19 \end{pmatrix} = \begin{pmatrix} 99 \\ 62 \\ 28 \end{pmatrix} \text{ и т.д.}$$

Получим шифрованный текст: 99, 62, 28, 96, 60, 24, и т.д.

Дешифрование производится по тому же правилу умножения, но в качестве ключа берём обратную матрицу $(a_{ij})^{-1}$ и умножаем её на вектор столбец из соответствующего количества чисел шифрограммы. Числа вектора результата дадут эквиваленты знаков исходного текста.
Т.к. процедуры шифрования и дешифрования строго формализованы, то они сравнительно легко программируются. Недостаток — много арифметических действий для матрицы выше 3-го порядка.
Достоинство — фактически длина ключа (здесь 9 чисел) длиннее групп (здесь 3 числа) циклического шифрования/дешифрования символов текста, что, по-видимому, и увеличивает стойкость шифра.

Возьмём исходный («человеческий») текст информации, представленный языком, содержащим *к* символов. Закодируем каждый символ языка каким-либо исходным кодом (m бит/символ), например нормированным по длине кодом Морзе (точка - 0, тире - \) или стандартным телеграфным кодом, или байтами кода ASCII, и т.п. При простейшем кодировании только 32 букв русского алфавита 5-ю битами получим уже известные виды буквенных замен.
Но теперь рассматриваем всё сообщение как *сплошной поток* бит. Разбиваем его на блоки из *n* разрядов: $n > m$
Замену (шифрование) производим поблочно, рассматривая каждый блок как единое целое, заменяющий его блок шифрограммы должен содержать не меньшее количество бит.
Число различных *n* - разрядных блоков равно 2^n. Все такие различные подстановки можно рассматривать как отображение внутри этого множества блоков.
Если отображение для 2^n различных блоков обратимо, то говорят, что оно несингулярно, т.е. существует взаимнооднозначное соответствие между каждым блоком исходного текста и некоторым блоком этого же множества, рассматриваемого как шифротекст (Таблица 1).

X			Y		
D	X_1	X_0		Y_1	Y_0
0	0	0	2	1	0
1	0	1	3	1	1
2	1	0	1	0	1
3	1	1	0	0	0



На Рис. 1 показан пример устройства обратимого (несингулярного) преобразования. Здесь «шифратор» и «дешифратор» — это термины схемотехники, а не криптографии. Преобразование *n* входных разрядов в *n* выходных представляетсобой подсоединение (перестановку) 2^n выходов «дешифратора» в 2^n входов «шифратора».

На рис. 4 показана реализация таблицы 1 на микросхеме КП12. Количество таких обратимых (несингулярных) преобразований (перестановок) равно $(2n^2)!$.Любое из этих преобразований реализуется соответствующими соединениями. Эти соединения называют ключом шифра, а преобразования *n* разрядов в *n* разрядов называют S-преобразованиями.

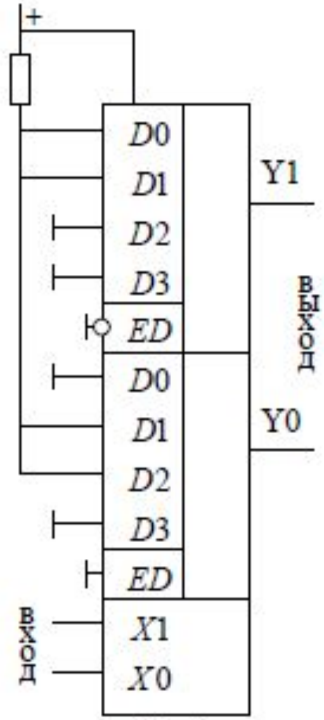


Рис.4
Реализация рис.3
с помощью МС ххКП12
(2-х разрядный селектор-
мультиплексор).
Для 4-х разрядного входа
надо взять две МС и т.д.

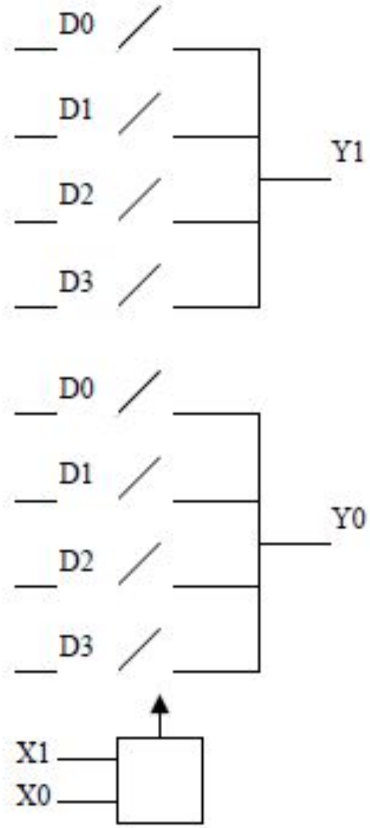


Рис.5 Схема функционирования
МС КП12

5.5. Свойства S-преобразований.

Имеется множество n-разрядных двоичных слов. S-преобразование есть отображение этого множества на самое себя. Отображение (S-преобразование) можно задавать либо правилами, либо таблично. Например, для 2-х разрядных слов:

Обратимое отображение		Необратимое отображение	
Слова исходного текста	Слова шифротекста	Слова исходного текста	Слова шифротекста
A	S(A)	A	S(A)
00	11	00	11
01	10	01	10
10	00	10	01
11	01	11	01

Всего во множестве имеется 2^n n -разрядных слов, а различных отображений в этом множестве $(2^n)^{(2n)}$

Однако, все отображения, содержащие сингулярные множества, нежелательны, т.к. приводят к неоднозначности дешифрования шифротекста. Поэтому применяют только обратимые (несингулярные) S-преобразования. Количество таких S-преобразований равно $(2^n)!$. Фактически — это перестановки слов в таблице обратимого S-преобразования, которое называют аффинным преобразованием.

Аффинным называют преобразование S, обладающее свойством: если A и B два двоичных вектора, одинаковой размерности; если S есть преобразование пространства этих векторов в себя, и если Z, вычисляемое как:

Z = $S(A \oplus B) \oplus S(A) \oplus S(B)$

Оказывается постоянным для все x A и все x B, то S является аффинным преобразованием.

Проверим аффинность для приведённой выше таблицы обратимого преобразования.

	A=00	A=00		A=01	…
	$B=00$	$B=01$		$B=11$	…и т.д. для всех пар
$A \oplus B$	00	01		10	…
$S(A \oplus B)$	S(00)=11	S(01)=10		S(10)=00	
по	S(A)=11	S(A)=11		S(A)=10	
таблице	$S(B)=11$	$S(B)=10$		$S(B)=01$	
	Z=11	Z=11	и т.д	Z=11	и т.д. Z=const

Метод перестановок (шифрование перестановками)

Исходный текст разбивается на ключевые группы с равными количествами букв в группах. В каждой группе по заданному правилу производится перестановка букв.

Табличный вариант

Записываем исходный текст по строкам в матрицу из N столбцов. Затем шифруем текст переставляя столбцы матрицы в заданном порядке перестановок. Этотпорядок перестановок есть ключ (и операция) перестановок. Заданный порядок перестановок можно выразить осмысленным словом (ключом) с неповторяющимися буквами и

производить шифрование, т.е. перестановку колонок таблицы в той последовательности, в которой располагаются в алфавите буквы ключевого слова.

Ключ	Д	Е	З	А	В	И
	5	6	8	1	3	9
Текст	Ш	И	Ф	Р	У	Й
	Т	Е	Ё	П	Е	Р
	Е	С	Т	А	Н	О
	В	К	А	М	И	Ь

— порядок букв ключа в алфавите

Пробелы между словами исходного текста и конец текста заполняем для полноты матрицы произвольными буквами.

Получаем, читая по столбцам в порядке перестановок следующую шифровку:
РПАМУЕНИШТЕВИЕСКФЁТАЙРОЬ или группами по 6 букв:
РПАМУЕ НИШТЕВ ИЕСКФЁ ТАЙРОЬ

Расшифровка

Определяем число колонок, деля количество знаков в шифрограмме на число букв в ключе 30/6 = 5.

Выписываем ключевое слово с обозначением последовательности букв ключа в алфавите и под ними в *колонки* с указанной последовательностью выписываем текст шифровки. Открытый текст читаем по строкам.

Усложнение табличного варианта.

Шифруемый текст вписываем в таблицу выбранной размерности по некоторому маршруту, например по спирали. Затем колонки выписываем либо подряд, либо переставляя по ключу. Расшифровываем в обратной последовательности.

Перестановка по маршрутам Гамильтона.

Такая сравнительно простая перестановка является по оценкам американских специалистов достаточно стойким шифром.

Исходный текст разбивается на группы по 8 букв. 1-ая операция — вписывание исходного текста в шаблон с 8-ю знакоместами с указанным на них порядком вписывания. Например текст «ШИФРУЙТЕ ПЕРЕСТАНОВКАМИ» вписываем без пробелов, а конец текст дополним до полноты шаблона буквами «А».

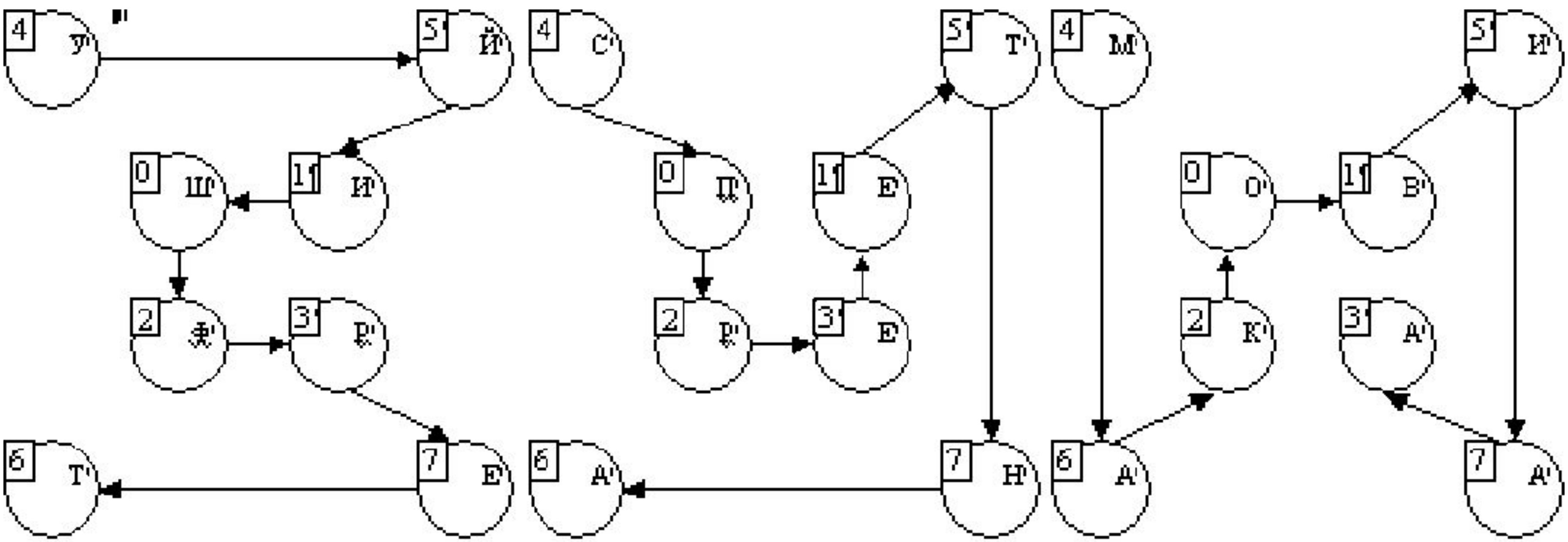


Рис.6'

2-ая операция — последовательное повторение 5-ти разных маршрутов Гамильтона. На рисунках нам хватило 3-х маршрутов. Выписываем по этим маршрутам шифрограмму:

УЙИШФРЕТ	СПРЕЕТНА	МАКОВИАА
1-я перестановка	2-я перестановка	3-я перестановка

Для перестановки букв в группах по 8 количество разных перестановок (маршрутов) *M*= P(8) = 8!=40320. Количество возможных перестановок быстро увеличивается с ростом длины группы перестановок.
Если злоумышленник *угадает* длину группы, то он может перебрать последовательно все возможные перестановки пока не найдёт осмысленную. Для малой длины группы это легко особенно с помощью ЭВМ. Посмотрим как усложняется этот пример с ростом длины группы.

Длина группы	Количество перестановок	Время просмотра их на ЭВМ со скоростью 1 перестановка/ сек.
8	40320	11.2 часа
10	3628800	42 суток
12	* $479 \cdot 10^6$	5544 суток « 15 лет

Количество M перестановок для группы из N букв равно: $M = P(N) = N!$ Перестановки удобно задавать числовыми ключами (гаммами) Так перестановки Гамильтона будутиметь вид:

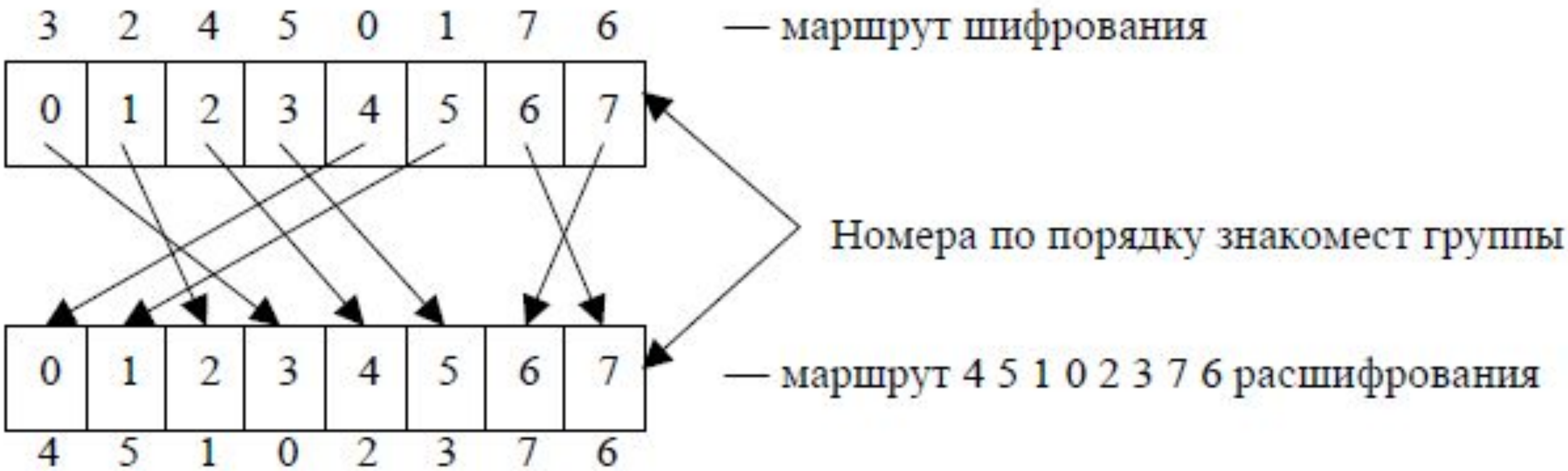
	0	1	2	3	4	5	6	7	0	1	2	3	4	5	6	7	0	1	2	3	4	5	6	7
исх. текст	Ш	И	Ф	Р	У	Й	Т	Е	П	Е	Р	Е	С	Т	А	Н	О	В	К	А	М	И	А	А
ключи шифрования	3	2	4	5	0	1	7	6	1	4	2	3	0	5	7	6	3	4	2	7	0	5	1	6
шифротекст	У	Й	И	Ш	Ф	Р	Е	Т	С	П	Р	Е	Е	Т	Н	А	М	А	К	О	В	И	А	А
	0	1	2	3	4	5	6	7	0	1	2	3	4	5	6	7	0	1	2	3	4	5	6	7

Расшифрование производится в обратном порядке (двигаться в направлении обратном стрелке перестановки), т.е. ключи перестановки для расшифрования будут: Перепишем ключи шифрования в виде

Ключ шифрования:	03	12	24	35	40	51	67	76
Ключ расшифр.:	30	21	42	53	04	15	76	67
Ключ шифрования:	04	15	23	30	42	53	67	76
Ключ расшифр.:	4	5	3	0	2	3	7	6

03 – 0-е место исх. текста переставляется на 3-е место шифрограммы

- ключ шифрования, упорядоченный по 1-му знаку



Очевидно, что две (разные) перестановки подряд не увеличивают стойкость шифра, т.к. эквивалентны некоторой одной.
Статистика букв шифротекста перестановки такая же как и у исходного текста. Но знание её не помогает взломать шифр, т.к. буквы поменялись местами, однако в рассмотренных вариантах оказывается проявляются статистические закономерности *букв ключа*, что может позволить раскрыть его.