

**МИНОБРНАУКИ РОССИИ**  
**САНКТ-ПЕТЕРБУРГСКИЙ ГОСУДАРСТВЕННЫЙ**  
**ЭЛЕКТРОТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ**  
**«ЛЭТИ» ИМ. В.И. УЛЬЯНОВА (ЛЕНИНА)**  
**Кафедра Информационной безопасности**

**ОТЧЕТ**  
**по лабораторной работе №4**  
**по дисциплине «Криптография и защита информации»**  
**Тема: Изучение шифра DES**

Студент гр.8382

\_\_\_\_\_

Нечепуренко Н.А.

Преподаватель

\_\_\_\_\_

Племянников А.К.

Санкт-Петербург

2021

## **Цели работы.**

Исследовать шифры DES, 3DES, а также другие модификации шифра DES: DESX, DESL, DESXL и получить практические навыки работы с ними, в том числе с использованием приложения Cryptool 1 и 2

## **Исследование преобразований DES.**

### ***Задание***

1. Изучить преобразования шифра DES с помощью демонстрационного приложения из Cryptool 1.
  - Indiv.Procedures-> Visualization...-> DES...
2. Выполнить вручную преобразования первых двух раундов и вычисление раундовых ключей при следующих исходных данных:
  - Открытый текст (не более 64 бит) – фамилия\_имя (транслитерация латиницей)
  - Ключ (56 бит) – номер зачетной книжки и инициал отчества (всего 7 символов)
3. Выполнить вручную обратное преобразование зашифрованного сообщения
4. Убедиться в совпадении результатов

### ***Описание преобразований DES.***

Стандарт шифрования данных (DES) — блочный симметричный шифр, разработанный Национальным Институтом Стандартов и Технологии (NIST – National Institute of Standards and Technology).

Шифр DES основан на сети Фейстеля.

DES шифрует информацию блоками по 64 бита с помощью 64-битного ключа шифрования. Шифрование выполняется следующим образом (рис. 1):

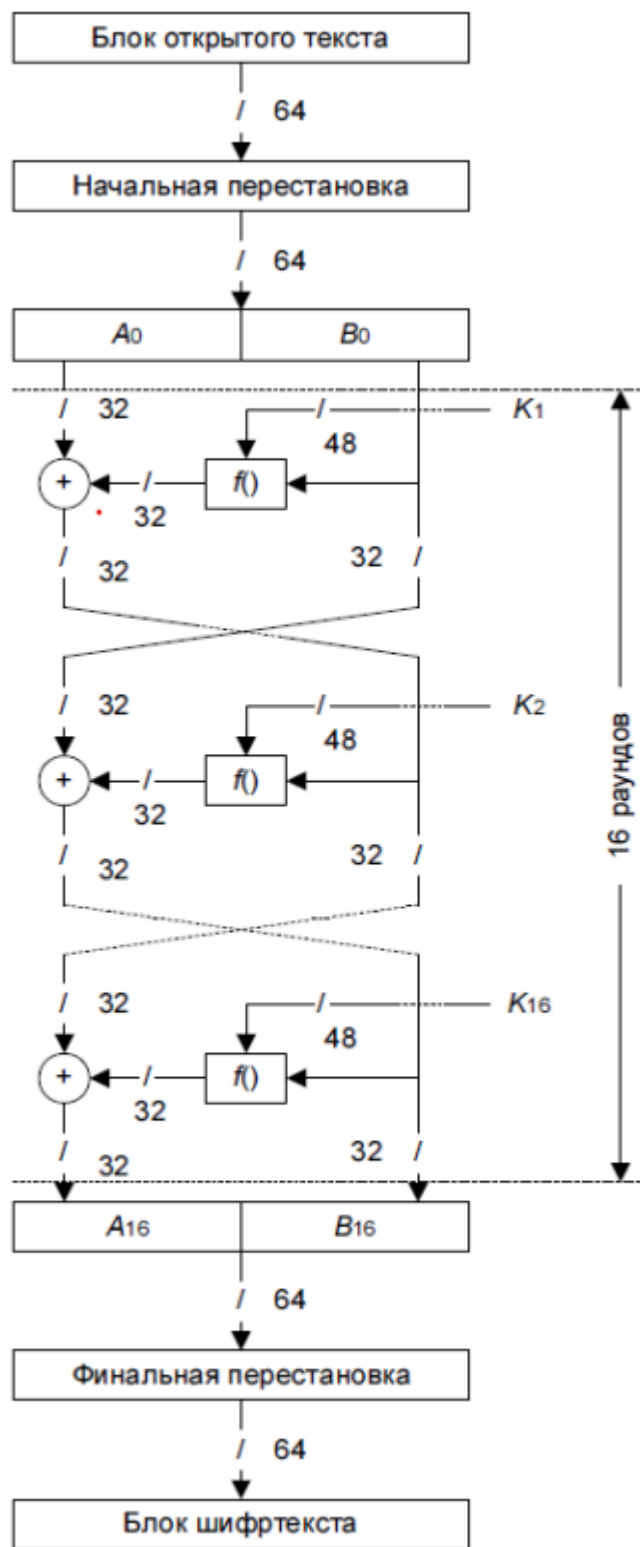


Рисунок 1

1. Над 64-битными блоками производится начальная перестановка, зада-

ваемая таблично

- После начальной перестановки блок делится на 2 субблока по 32 бита ( $A_0$  и  $B_0$ ), над которыми производятся 16 раундов преобразований:

$$A_i = B_{i-1}$$

$$B_i = A_{i-1} \oplus f(B_{i-1}, K_i)$$

где  $i$  – номер текущего раунда,  $K_i$  – ключ раунда,  $\oplus$  – логическая операция XOR.

Схема работы функции раунда  $f()$  представлена на рисунке 2.

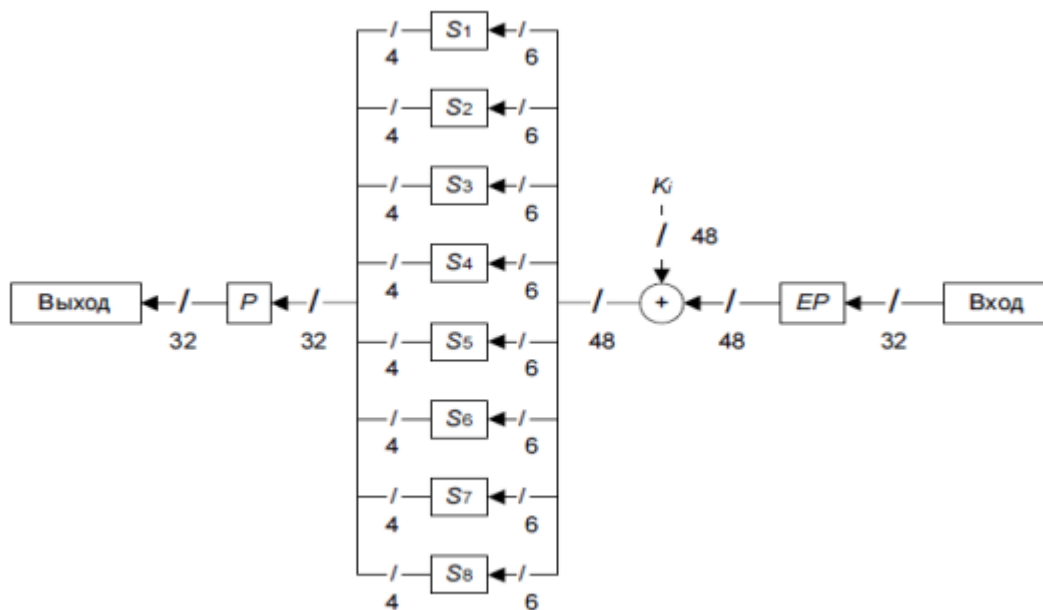


Рисунок 2

Этапы раундового преобразования следующие:

- Расширяющая перестановка EP, которая преобразует входные 32 бита в 48 бит (рис. 3).

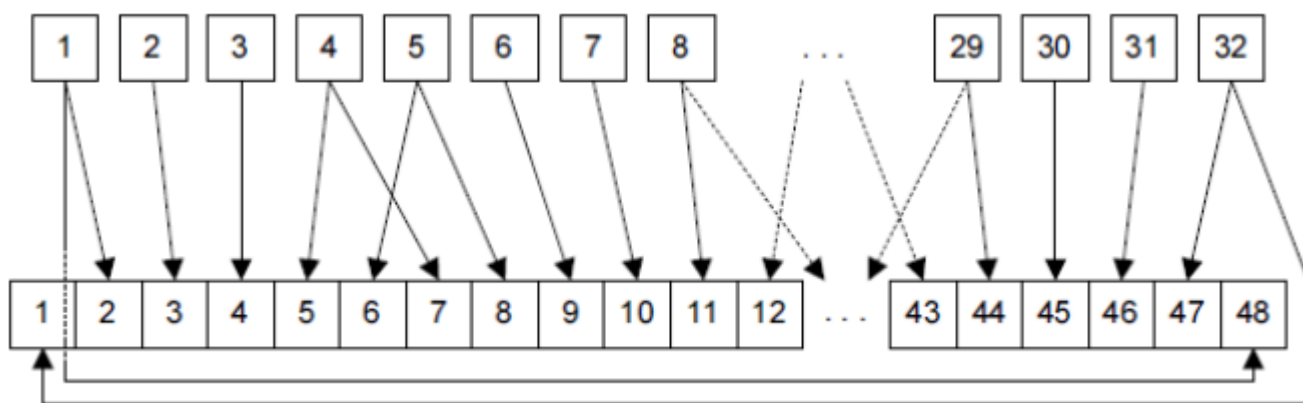


Рисунок 3

- (b) Полученные 48 бит складываются с  $K_i$  операцией хог.
- (c) Результат сложения разбивается на 8 блоков по 6 битов. Каждый блок обрабатывается соответствующей таблицей замен.
- (d) Над полученными 32 битами, после выполнения замен, выполняется перестановка (на рисунке 2 обозначена P).

На последнем раунде алгоритма субблоки местами не меняются.

3. Полученные в итоге субблоки  $A_{16}$  и  $B_{16}$  образуют 64-битный блок, над которым производится конечная перестановка и в итоге получается результирующий блок шифротекста.

Процедура генерации раундовых ключей представлена на рисунке 4. Из 64-битного ключа шифрования используется только 56 бит, каждый 8-й бит исключается. На рисунке 4 операция сжатия ключа и перестановка обозначена как E.

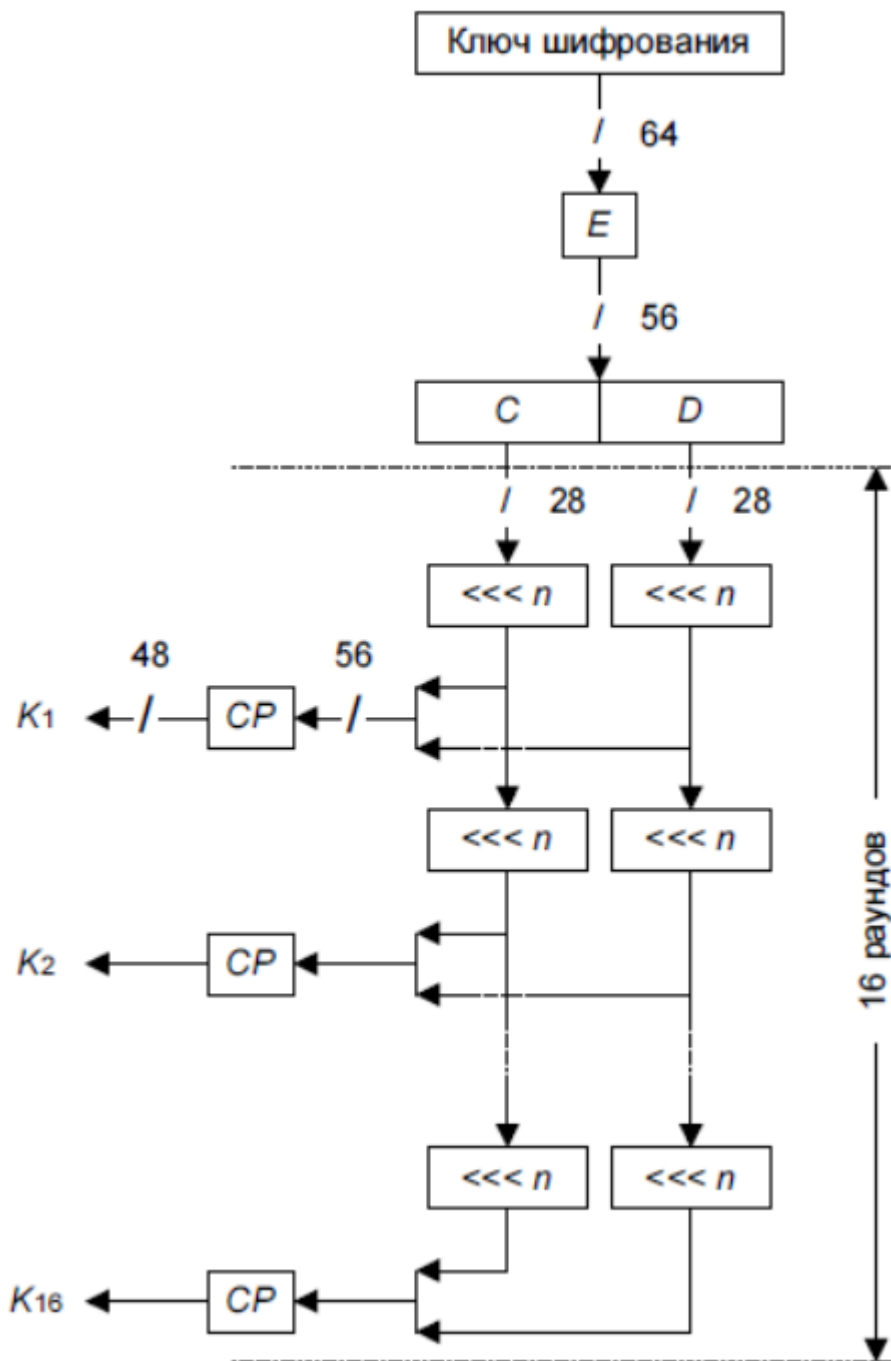


Рисунок 4

После перестановки блок в 56 бит делится на два 28-битных блока (C и D). Затем выполняются 16 раундов преобразований:

- (а) Текущие C и D циклически сдвигаются влево на определенное количество бит.

(b) С и D объединяются в 56-битное значение, к которому применяется сжимающая перестановка. На выходе получаем 48-битный раундовый ключ.

Расшифровывание данных алгоритмом DES происходит при прохождении всех шагов алгоритма в обратном порядке.

### ***Визуализация преобразований шифра DES в Cryptool 1.***

В верхнем меню Cryptool 1 выберем *Indiv.Procedures-> Visualization...-> DES...*

После этого откроется окно визуализации преобразований шифра DES (см. рис. 5).

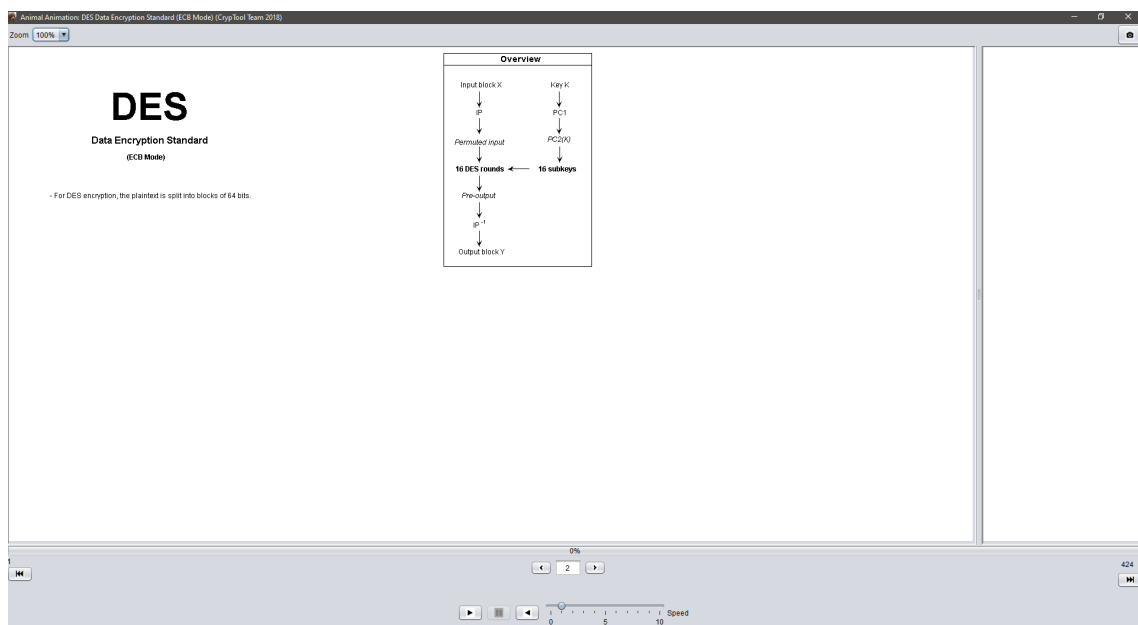


Рисунок 5 – Интерфейс окна визуализации работы шифра DES

Нажмем кнопку «Play» в нижней панели с кнопками. Далее начинается анимация первоначальной перестановки (IP) блоков исходного текста.

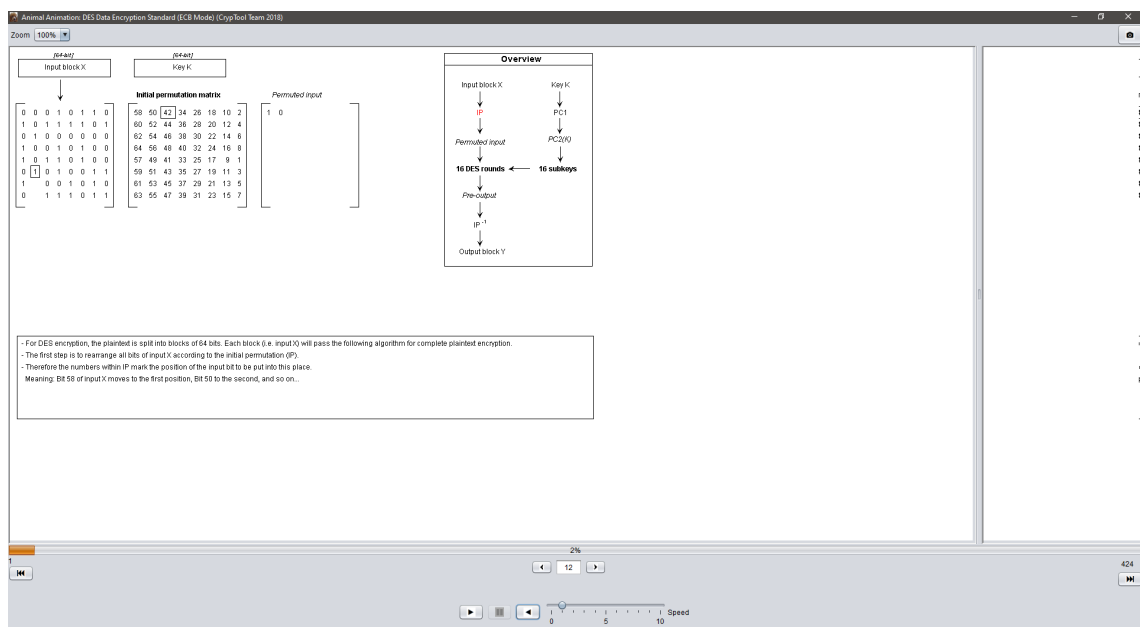


Рисунок 6 – Первоначальная перестановка блоков исходного текста

Затем происходит процесс генерации ключа раунда. Отбрасывается последний столбец исходного ключа, состоящий из битов четности, и производится перестановка  $C_0D_0$  для получения  $k_0$ .



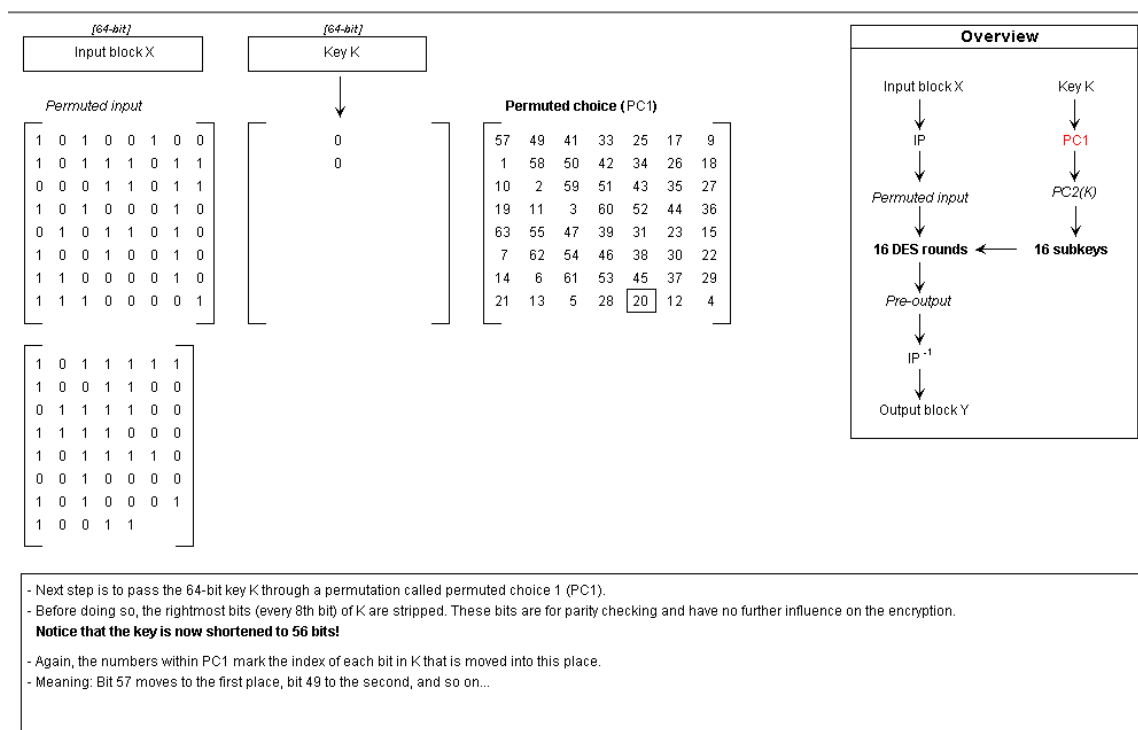


Рисунок 7 – Первоначальное преобразование ключа шифрования

Затем демонстрируется процесс генерации 16 ключей для каждого из раундов.

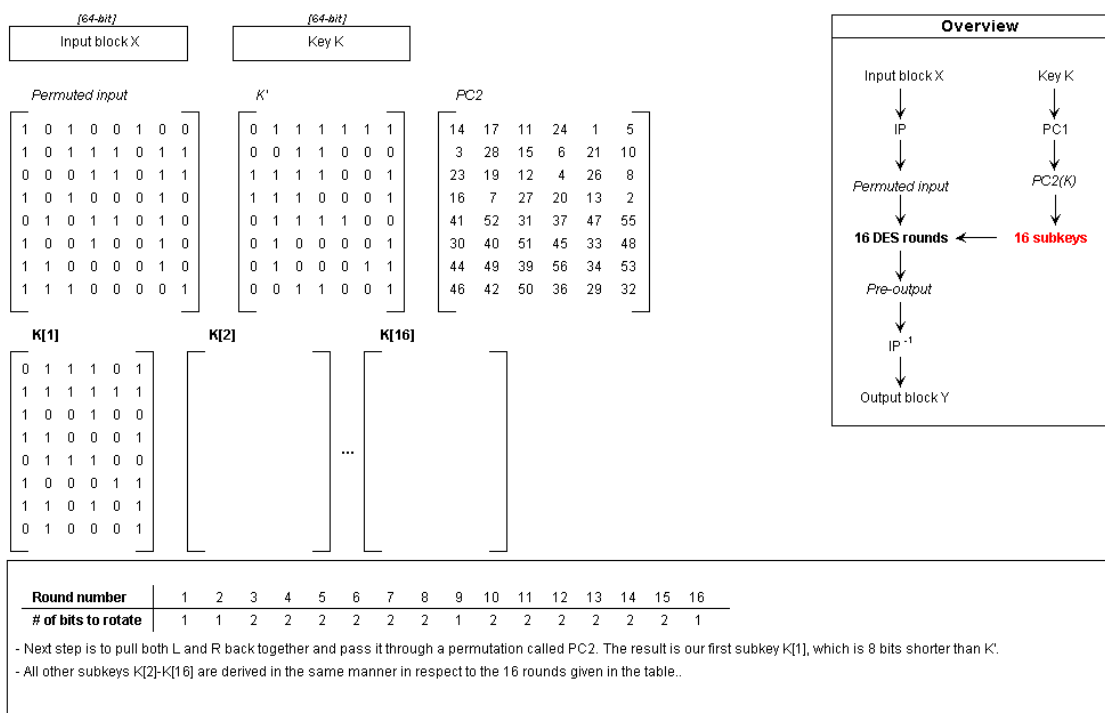


Рисунок 8 – Генерация 16 ключей для каждого из раундов

Далее приводится схема прямого хода алгоритма.

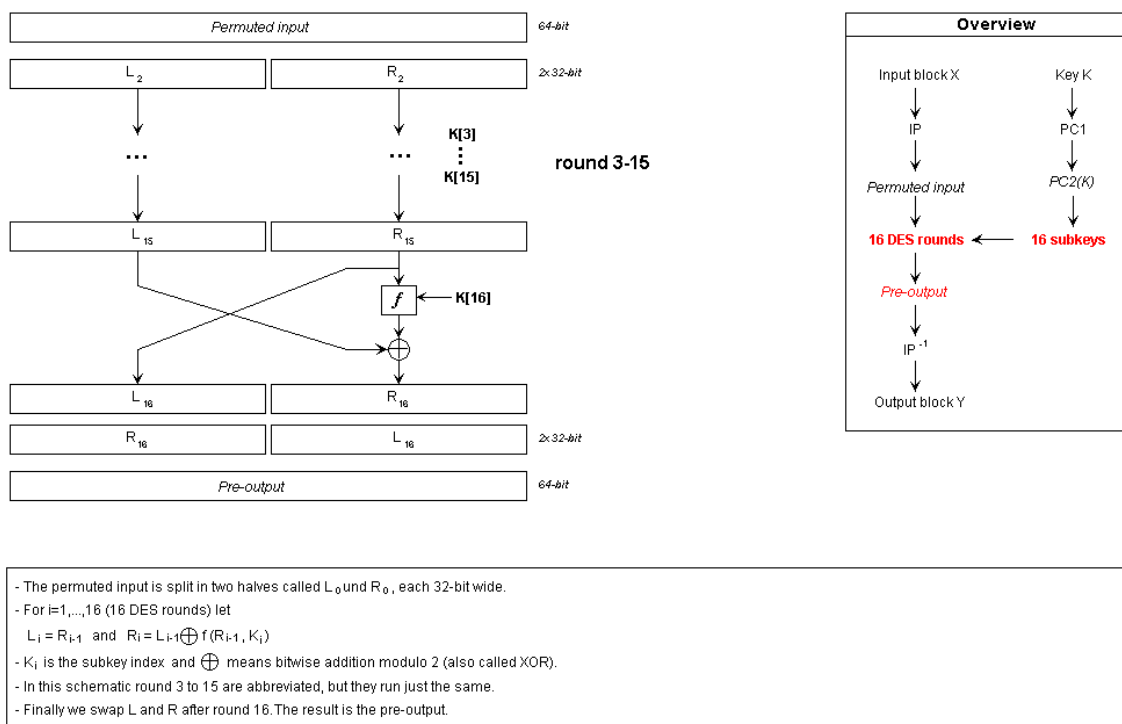


Рисунок 9 – 16 раундов прямого преобразования сетью Фейстеля

В следующей сцене происходит демонстрация работы функции расширения  $E$  для правого субблока (см. рис. 10).

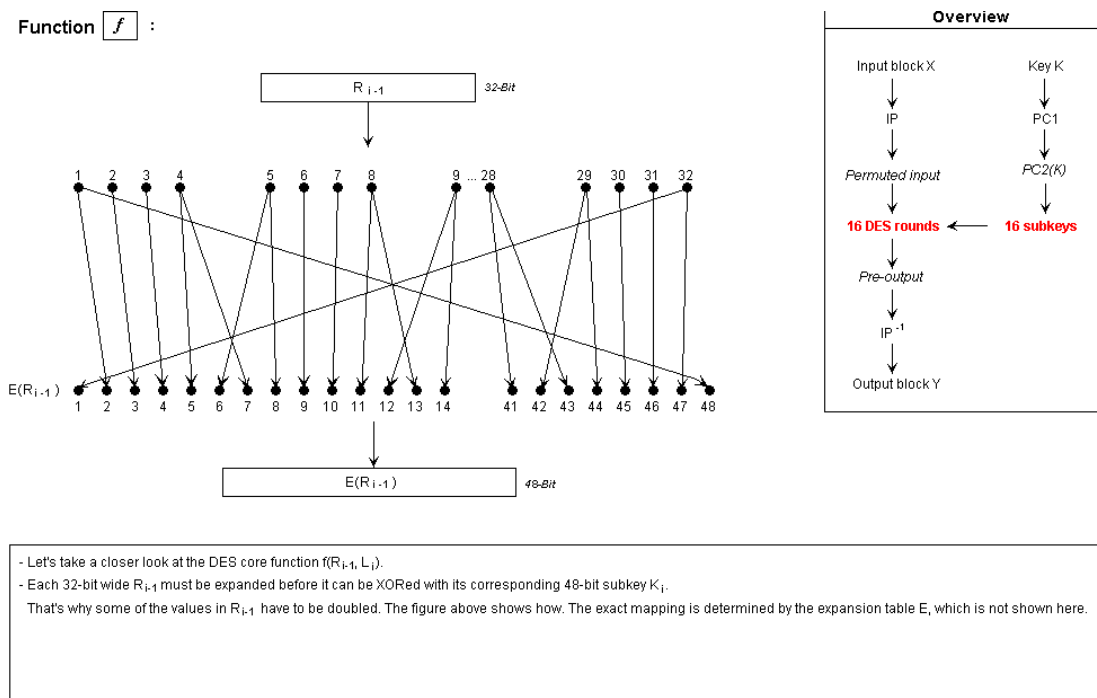


Рисунок 10 – Демонстрация работы функции расширения

Затем происходит вычисление операции XOR между значением функции расширения и ключом раунда.

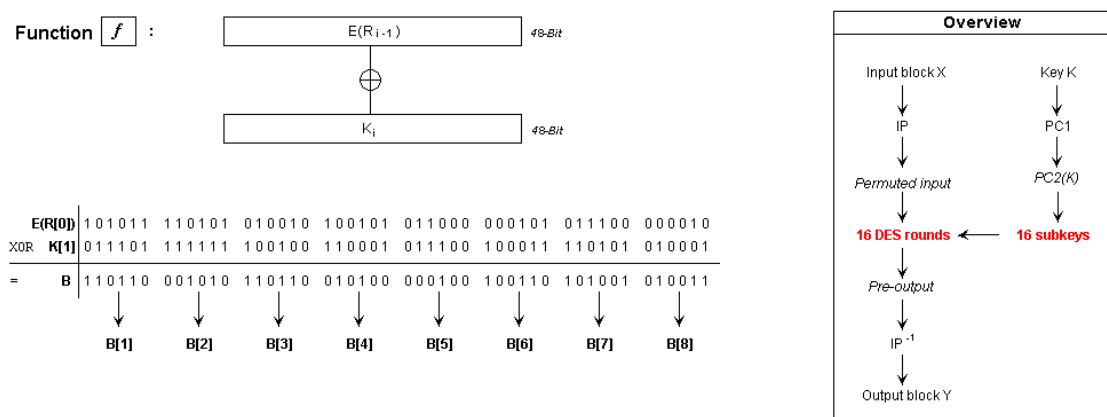


Рисунок 11 – Вычисление XOR между значением функции расширения и ключом раунда

Затем происходит преобразование S-блоков (см. рис. 12)

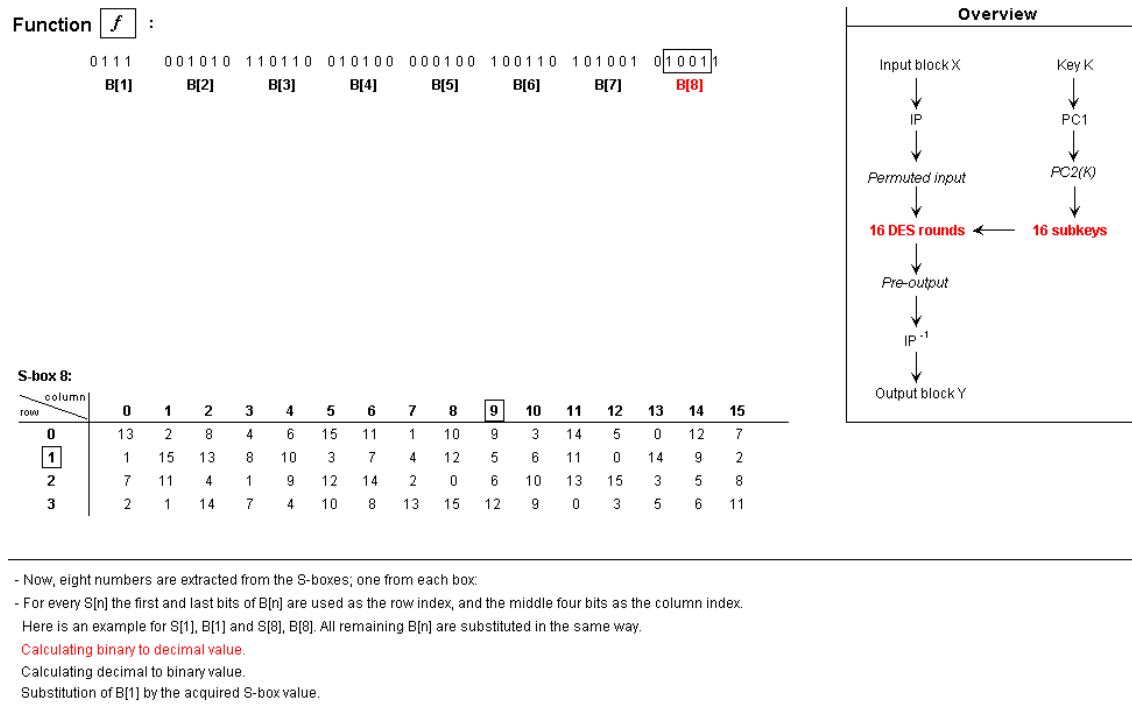
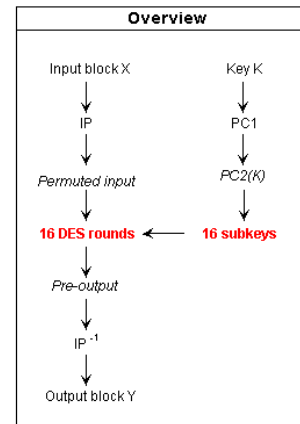


Рисунок 12 – Преобразование S-блоков

Затем происходит Р-перестановка полученного битового вектора.

Function  $f$  :

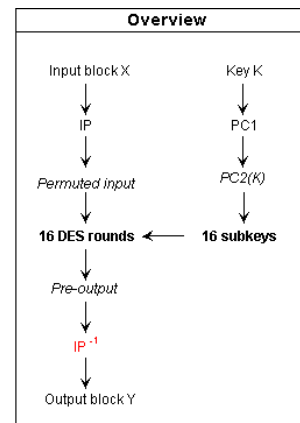
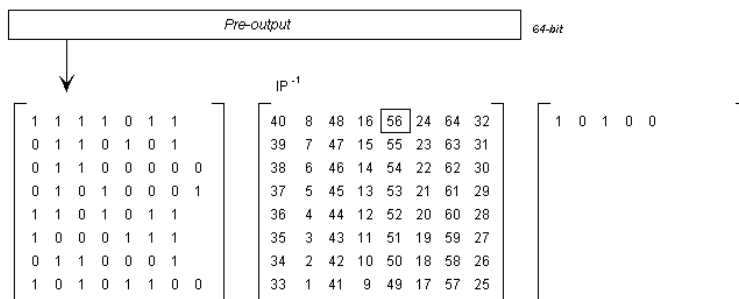
R	P	
1 1 1	16 7 20 21	0 1 0 0
1 0 1	29 12 28 17	0 0 1 0
1 1 0	1 15 23 26	0
1 0 0	5 18 31 10	
1 0	2 8 24 14	
1 0 1	32 27 3 9	
0 0 0	19 13 30 6	
1 0 1	22 11 4 25	



- Now, eight numbers are extracted from the S-boxes; one from each box.
- For every  $S[n]$  the first and last bits of  $B[n]$  are used as the row index, and the middle four bits as the column index.
- $B[2]$  to  $B[7]$  are substituted the same way.
- The result  $R$  is the concatenation of  $B[1]$  to  $B[8]$ , shown in the matrix above.
- Finally,  $R$  is passed through the permutation  $P$ . Just like the other S-boxes, this matrix is constant.

Рисунок 13 – P-перестановка

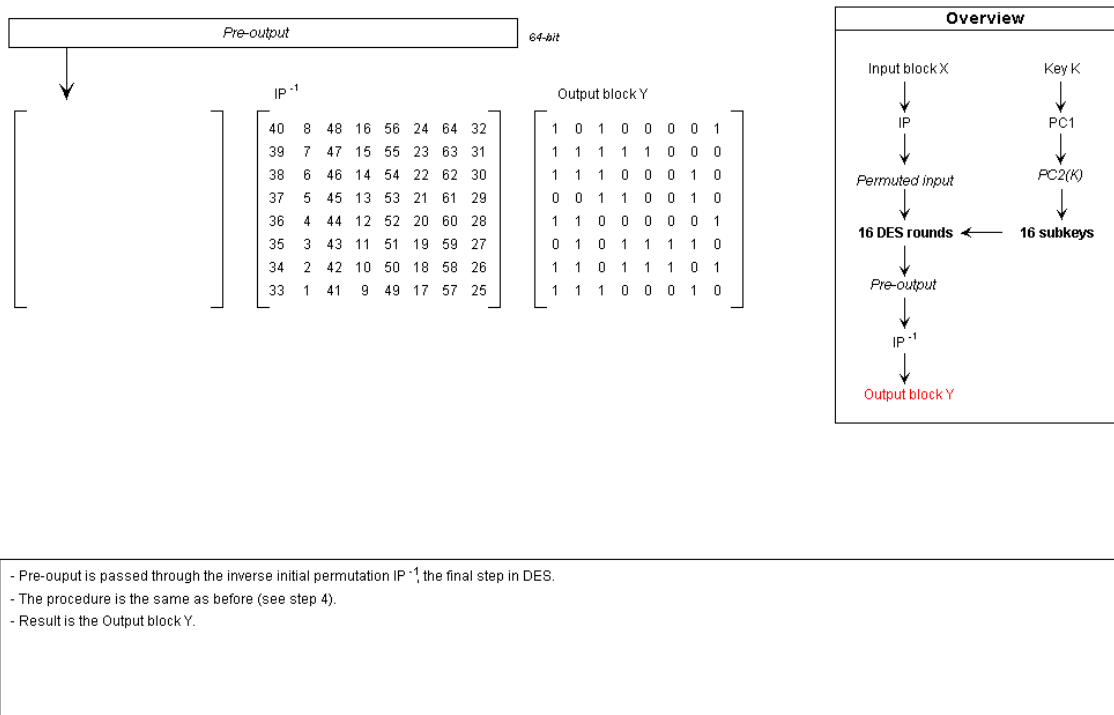
После 16 раундов полученный блок подвергается перестановке  $IP^{-1}$ .



- Pre-output is passed through the inverse initial permutation  $IP^{-1}$ , the final step in DES.
- The procedure is the same as before (see step 4).

Рисунок 14 –  $IP^{-1}$ -перестановка

Заканчивается анимация получением шифротекста (см. рис. 15).



- Pre-output is passed through the inverse initial permutation IP<sup>-1</sup>, the final step in DES.
- The procedure is the same as before (see step 4).
- Result is the Output block Y.

Рисунок 15 – Результат шифрования DES

### *Ручное шифрование блока текста.*

Вычислим раундовые ключи и субблоки для первых двух раундов шифрования текста NECHEPUR шифром DES с ключом 838209A.

$$\text{NECHEPUR} = \begin{pmatrix} 0 & 1 & 0 & 0 & 1 & 1 & 1 & 0 \\ 0 & 1 & 0 & 0 & 0 & 1 & 0 & 1 \\ 0 & 1 & 0 & 0 & 0 & 0 & 1 & 1 \\ 0 & 1 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 1 & 0 & 1 \\ 0 & 1 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 1 & 0 & 1 & 0 & 1 \\ 0 & 1 & 0 & 1 & 0 & 0 & 1 & 0 \end{pmatrix}$$

$$838209A = \begin{pmatrix} 0 & 0 & 1 & 1 & 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 1 & 0 & 0 & 1 & 1 \\ 0 & 0 & 1 & 1 & 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 1 & 0 & 0 & 1 & 0 \\ 0 & 0 & 1 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 1 & 1 & 0 & 0 & 1 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 & 1 \end{pmatrix}$$

Сначала вычислим раундовые ключи для двух раундов (см. рис. 15а и 15б).





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
1	1	0	0	1	0	0	0	1	0	0	0	1	0	0	0	1	0	1	0	0	0	1	0	0
26	27	28	29	30	31	32	33	34	35	36	37	38	39	40	41	42	43	44	45	46	47	48	49	50
0	0	1	0	0	1	1	1	0	0	0	1	0	0	1	1	0	0	0	0	0	0	0	1	1
51	52	53	54	55	56																			
1	0	0	1	1	1																			

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
0	1	0	0	1	1	0	1	0	0	0	0	1	1	0	0	0	0	0	0	0	0	1	1	1
26	27	28	29	30	31	32	33	34	35	36	37	38	39	40	41	42	43	44	45	46	47	48	49	50
0	1	1	0	1	0	1	1	0	1	0	0	1	0	1	0	0	0	0	1	0	0	1	-	-
51	52	53	54	55	56																			
-	-	-	-	-	-																			

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
0	0	0	0	1	0	0	0	0	0	0	0	0	0	1	1	0	1	1	0	1	0	0	0	0
26	27	28	29	30	31	32	33	34	35	36	37	38	39	40	41	42	43	44	45	46	47	48	49	50
0	1	0	0	1	1	1	0	0	0	1	0	1	1	0	1	0	0	1	1	0	1	-	-	-
51	52	-	-	-	-																			
-	-																							

K<sub>1</sub>

K<sub>2</sub>

(сдвиг на 1  
6 умнож.)

Рисунок 15b – Вычисление ключей для первых двух раундов

На рисунках 15c, 15d и 15e продемонстрирован процесс расчета первых двух раундов алгоритма

IP (схематический)

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
1	1	1	1	1	1	1	1	1	1	0	0	0	0	0	0	0	1	0	1	0	0	1	1	0
26	27	28	29	30	31	32	33	34	35	36	37	38	39	40	41	42	43	44	45	46	47	48	49	50
1	0	1	0	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
51	52	53	54	55	56	57	58	59	60	61	62	63	64											
0	0	1	0	0	1	1	0	0	0	0	1	0	1											

$$L_1 = R_0$$

$$R_1 = L_0 \oplus f(R_0, k_1)$$

$$f(R_0, k_1)$$

$E(R_0)$ :

1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24
1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	1	0	0	1	1	0	1	0	0	0	0	1	1	0	0	0	0	0	0	0	0	1	1
1	1	0	0	1	1	0	1	0	0	0	0	1	1	0	0	0	0	0	0	0	0	1	1

25	26	27	28	29	30	31	32	33	34	35	36	37	38	39	40	41	42	43	44	45	46	47	48
0	0	0	0	0	1	0	1	0	0	1	1	1	1	0	0	0	0	0	0	1	0	1	0
1	0	1	1	0	1	0	1	1	0	1	0	0	1	0	1	0	0	0	0	1	0	0	1
1	0	1	1	0	0	0	1	0	0	1	1	0	0	1	0	0	0	0	0	0	0	1	1

S1	S2	S3	S4	S5	S6	S7	S8
11	9	11	8	13	15	4	1

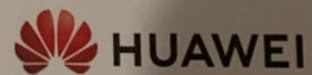


Рисунок 15с



	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	
B'	1	0	1	1	1	0	0	1	1	0	1	1	1	0	0	0	1	1	0	1	1	1	1	1	0	
	26	27	28	29	30	31	32																			
	1	0	0	0	0	0	1																			
	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	
P(B)	0	0	1	1	0	1	0	1	1	0	1	1	1	1	0	0	0	1	1	0	1	0	1	1	0	
L <sub>0</sub>	1	1	1	1	1	1	1	1	1	1	0	0	0	0	0	0	0	1	0	1	0	0	1	1	0	
XOR	1	1	0	0	1	0	1	0	0	1	0	1	1	1	0	0	0	0	1	1	0	0	0	0	0	
	26	27	28	29	30	31	32																			
	1	0	0	1	1	1	0																			
	1	0	1	0	1	1	0																			
XOR	0	0	1	1	0	0	0																			
	R <sub>1</sub> = 11 00 10 10 00 10 11 00 00 11 10 00 00 11 10 00																									
	00 00 11 00 0																									
	L <sub>2</sub> = R <sub>1</sub>																									
	R <sub>2</sub> = L <sub>1</sub> ⊕ f(R <sub>1</sub> , K <sub>2</sub> )																									
	⊗ f(R <sub>1</sub> , K <sub>2</sub> ), ⊗ E(R <sub>1</sub> )																									
	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	
	1	1	0	0	1	0	1	0	0	1	0	1	1	1	0	0	0	0	1	1	1	0	0	0	0	
	26	27	28	29	30	31	32																			
	0	0	1	1	0	0	0																			



**HUAWEI**

Рисунок 15d

	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24
$E(R_1)$	0	1	1	0	0	1	0	1	0	1	0	0	0	0	1	0	1	1	1	1	1	0	0	0
$V_2$	0	0	0	0	1	0	0	0	0	0	0	0	0	0	1	1	0	1	1	0	1	0	0	0
$XOR$	0	1	1	0	1	1	0	1	0	1	0	0	0	0	0	1	1	0	0	1	0	0	0	0

25	26	27	28	29	30	31	32	33	34	35	36	37	38	39	40	41	42	43	44	45	46	47	48
0	0	0	1	1	1	1	0	0	0	0	0	0	0	0	0	1	1	1	1	0	0	0	1
0	0	1	0	0	1	1	0	0	0	1	0	1	1	1	0	1	0	0	1	1	0	1	0
0	0	1	1	1	0	0	0	0	0	1	0	1	1	1	0	1	1	1	1	0	0	0	0

S1	S2	S3	S4	S5	S6	S7	S8
5	2	14	1	6	12	1	5

	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24
$B'$	0	1	0	1	0	0	1	0	1	1	0	0	0	0	1	0	1	1	0	1	1	0	0	0
$R(B')$	1	1	0	1	0	0	1	0	0	0	0	0	0	1	0	1	0	0	0	1	0	0	1	0
$L_1$	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	1	0
$XOR$	1	1	0	1	0	0	1	0	0	0	0	0	0	1	0	1	1	0	0	0	0	0	0	0

25	26	27	28	29	30	31	32
0	0	0	1	0	1	0	1
1	0	1	0	1	1	1	0
1	0	0	0	0	1	0	1
0	0	1	0	1	0	1	1

$= R_2$

$R_2 L_2 =$

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
1	1	0	1	0	0	1	0	0	0	0	0	0	1	0	1	1	0	0	0	0	0	0	0	0

26	27	28	29	30	31	32	33	34	35	36	37	38	39	40	41	42	43	44	45	46	47	48	49	50
0	1	0	1	0	1	1	1	1	0	0	1	0	1	0	0	1	0	1	1	1	0	0	0	0
51	52	53	54	55	56	57	58	59	60	61	62	63	64											
1	1	1	0	0	0	0	0	0	1	1	0	0	0											

Рисунок 15е

В результате после двух раундов получили цепочку  $R_2 L_2$ .

Расшифровка выполнялась по схеме

$$R_{i-1} = L_i$$

$$L_{i-1} = R_i \oplus f(L_i, k_i)$$

Ключи раундов остаются прежними, как и используемые матрицы перестановок.

Результат дешифровки совпал с исходным текстом.

### ***Вывод.***

Была изучена интерактивная демонстрация работы алгоритма DES в программе Cryptool 1. Было выполнено ручное преобразование первых двух раундов и вычисление раундовых ключей для части фамилии NECHEPUR и ключа 838209A.

## **Исследование DES в режимах ECB и CBC.**

### ***Задание.***

1. Создать картинку со своими ФИО (формат bmp).
2. Зашифровать картинку шифром DES в режиме ECB.
3. Зашифровать картинку шифром DES в режиме CBC с тем же ключом.
4. Сохранить скриншоты картинок для отчета.
5. Сжать исходную и 2 зашифрованных картинки средствами Cryptool.  
Зафиксировать размеры полученных файлов в таблице.
6. Выбрать случайный текст на английском языке (не менее 1000 знаков) и зашифровать его DES в режиме ECB.
7. Для одного и того же шифротекста оценить время проведения атаки «грубой силы» в случаях, когда известно n-4, n-6, n-8,..., 2 байт секретного ключа. Зафиксировать результаты измерений в таблице.

8. Повторить подобные измерения для DES в режиме CBC.

### ***Описание преобразований DES в режимах ECB и CBC.***

В режиме ECB шифра DES используется независимо для каждого 64-битного блока шифруемых данных. Схема использования шифра в режиме ECB представлена на рисунке 16.

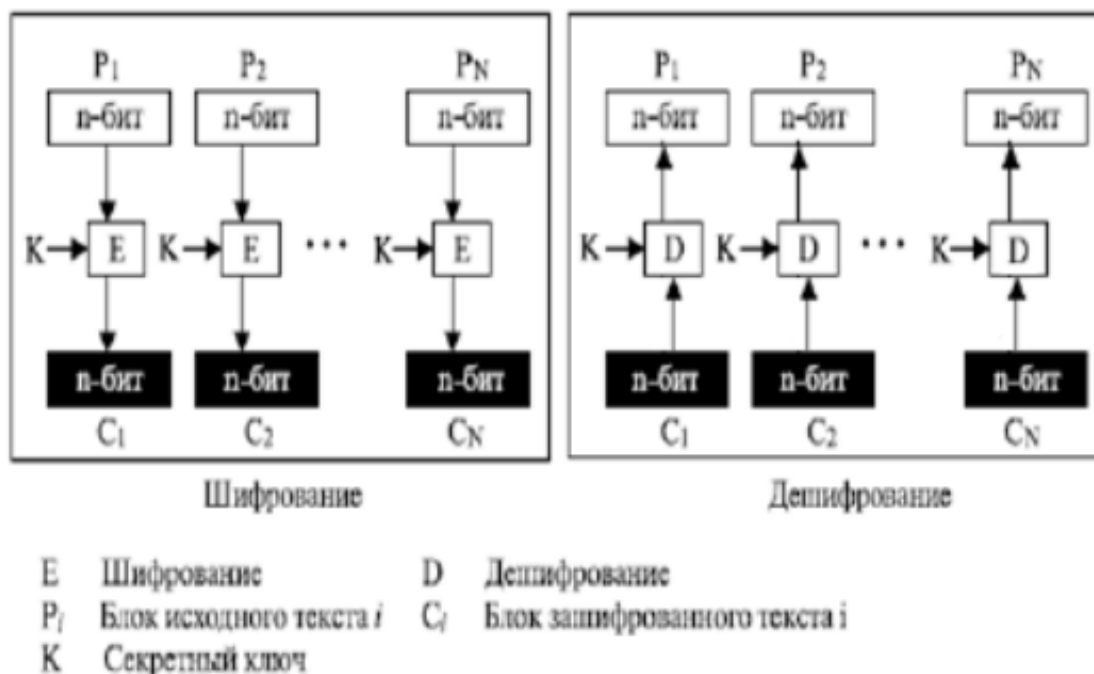


Рисунок 16 – Схема использования шифра в режиме ECB

Достоинства:

- Шифрование может быть параллельным
- Ошибка в передаче блока не имеет никакого воздействия на другие блоки

Недостатки:

- Одинаковые блоки открытого текста будут преобразовываться в одинаковые блоки шифротекста
- Независимость блоков создает возможность для замены некоторых блоков зашифрованного текста без знания ключа

В режиме CBC перед запуском DES для зашифрования каждого очередного блока открытого текста происходит побитовое XOR-сложение этого блока с блоком зашифрованного текста из предыдущего шага.

Схема использования шифра в режиме CBC представлена на рисунке 17.

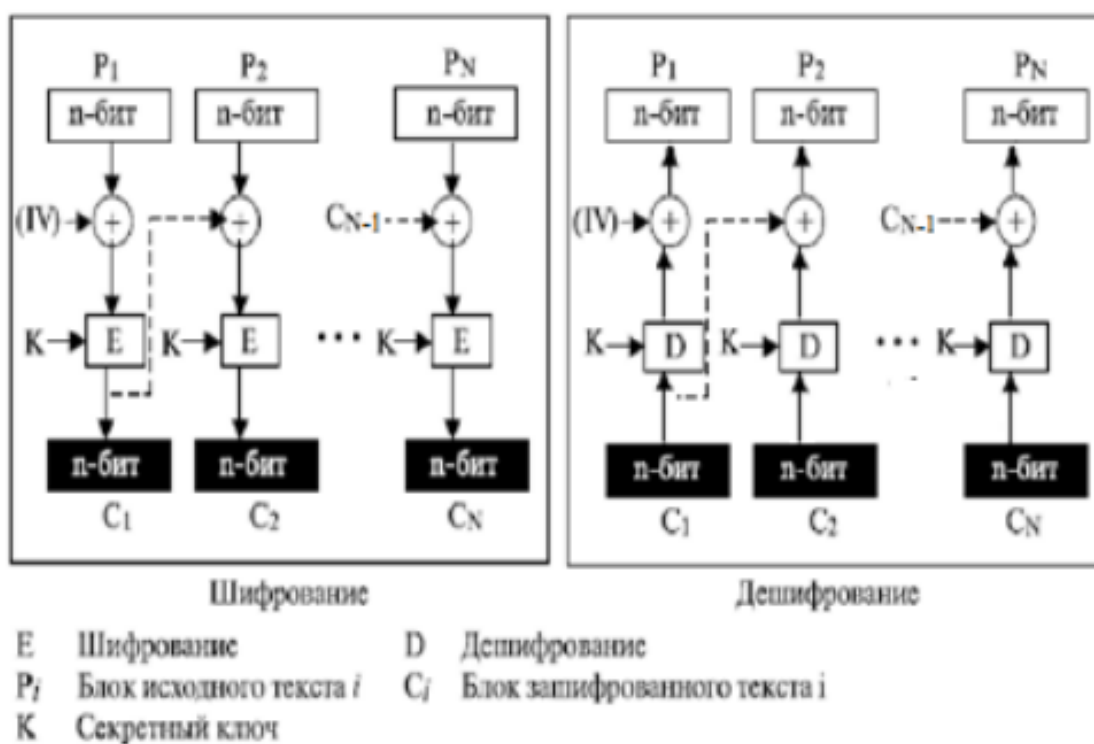


Рисунок 17 – Схема использования шифра в режиме CBC

Достоинства:

- Одинаковые блоки исходного текста преобразуются в различные блоки шифротекста
- Если при передаче произойдёт изменение одного бита шифротекста, данная ошибка распространится только на следующий блок (самовосстановление)
- Последний блок шифротекста зависит от всех бит открытого текста открытого текста сообщения и может использоваться для контроля це-

лостности

Недостатки:

- Шифрование сообщения не поддаётся распараллеливанию

### *Использования шифра DES в режимах ECB и CBC в Cryptool 1*

Создадим картинку в формате bmp с фамилией NECHEPURENKO (см. рис. 18).

---



Рисунок 18 – Исходная картинка с фамилией.

Зашифруем изображение с ключом 12 34 56 78 9A BC DE F0. Отделим шапку bmp файла, чтобы она не подверглась шифрованию.

```
42 4D B2 1F 00 00 00 00 00 00 00 3E 00 00 00 28 00
00 00 43 01 00 00 B7 00 00 00 01 00 01 00 00 00
00 00 74 1F 00 00 00 00 00 00 00 00 00 00 00 00
00 00 00 00 00 00 00 00 00 00 FF FF FF 00 FF FF
FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF
```

Результаты шифрования изображений приведены на рисунках 19 и 20



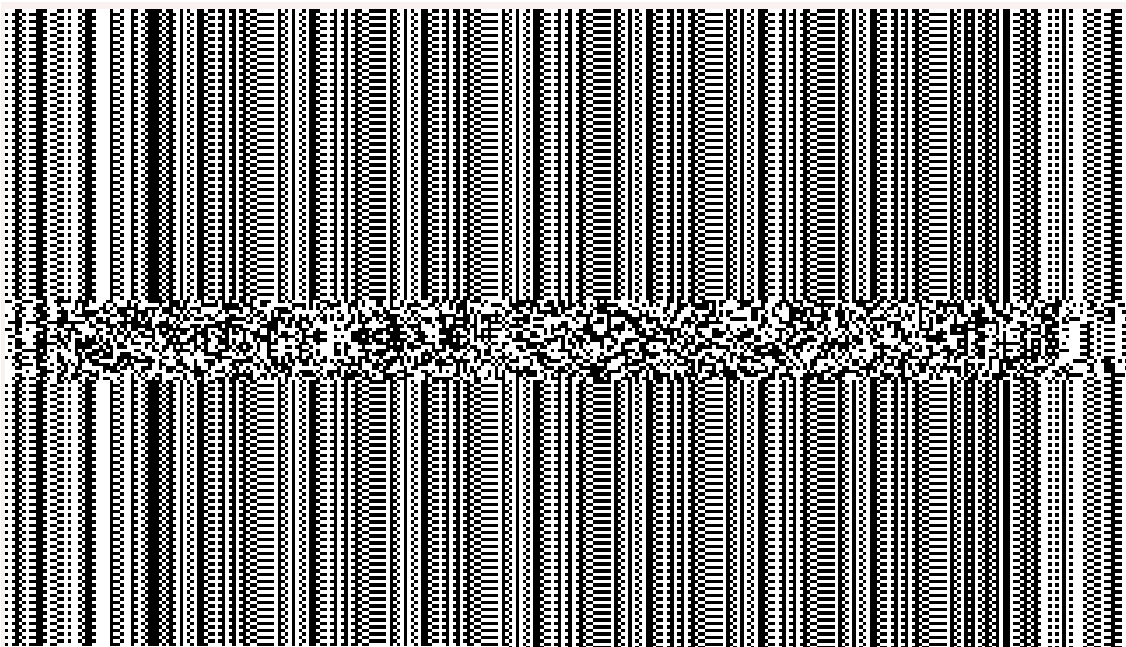


Рисунок 19 – Шифрование картинки в режиме ECB

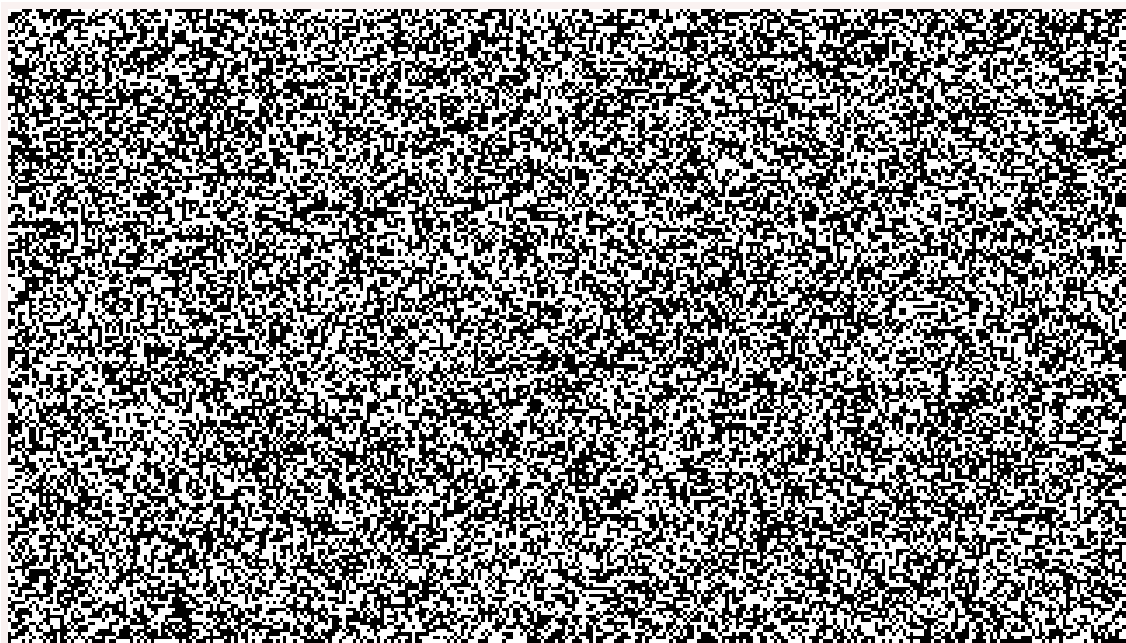


Рисунок 20 – Шифрование картинки в режиме CBC

Произведем сжатие полученных изображений средствами Cryptool 1. Результаты приведены в таблице 1.

*Таблица 1 – Сравнение размеров изображений*

Режим	Начальный размер, Кб	Размер после сжатия, Кб
Исходная картинка	7.92	0.44
ECB	7.92	1.04
CBC	7.92	7.92

Проведем анализ скорости расшифровки текста с помощью атаки грубой силы, если нам известна часть ключа. Результаты для алгоритмов ECB и CBC приведены в таблице 2.

*Таблица 2 – Сравнение времени взлома*

Известно байт	ECB	CBC
0	50000 лет	70000 лет
2	3.1 года	4.6 года
4	80 минут	120 минут
6	меньше секунды	меньше секунды

### ***Вывод.***

Исходное изображение было зашифровано с помощью DES ECB и DES CBC. Метод CBC является более криптостойким по сравнению с обычным ECB. На примере с шифрованием bmp файла было установлено, что алгоритм CBC дает более случайные данные, процент энтропии которых выше, чем при использовании ECB.

### **Исследование 3-DES.**

#### ***Задание.***

1. Выбрать случайный текст на английском языке (не менее 1000 знаков).

2. Создать бинарный файл с этим текстом, зашифровав и расшифровав его DES на 0-м ключе.
3. Снять и сохранить частотную и автокорреляционную характеристику этого файла.
4. Зашифровать бинарный файл шифром 3-DES в режиме ECB.
5. Снять и сохранить частотную и автокорреляционную характеристику файла с шифровкой.
6. Зашифровать исходный бинарный файл 3-DES в режиме CBC с тем же ключом.
7. Снять и сохранить частотную и автокорреляционную характеристику файла с шифровкой.
8. Определить экспериментальным путем по какой схеме работает реализация 3-DES в CrypTool. Сохранить подтверждающие скриншоты.

### ***Описание разновидностей 3-DES.***

Шифр 3-DES состоит из 3 последовательных раундов обычного DES, но из-за увеличения количества раундов удалось увеличить размер ключа до 112 или 168 бит, в зависимости от реализации. Существует 4 основные версии данного шифра:

1. DES-EEE3 – шифрование происходит 3 раза независимыми ключами
2. DES-EDE3 – операции шифровка-расшифровка-шифровка с тремя разными ключами
3. DES-EEE2 – то же что и DES-EEE3, но на первом и последнем шаге одинаковый ключ
4. DES-EDE2 – то же что и DES-EDE3, но на первом и последнем шаге одинаковый ключ

На текущий момент самыми популярными разновидностями шифра яв-

ляются DES-EDE3 и DES-EDE2. Данный шифр реализован во многих приложениях, ориентированных на работу с сетью Интернет, в том числе в PGP и S/mime.

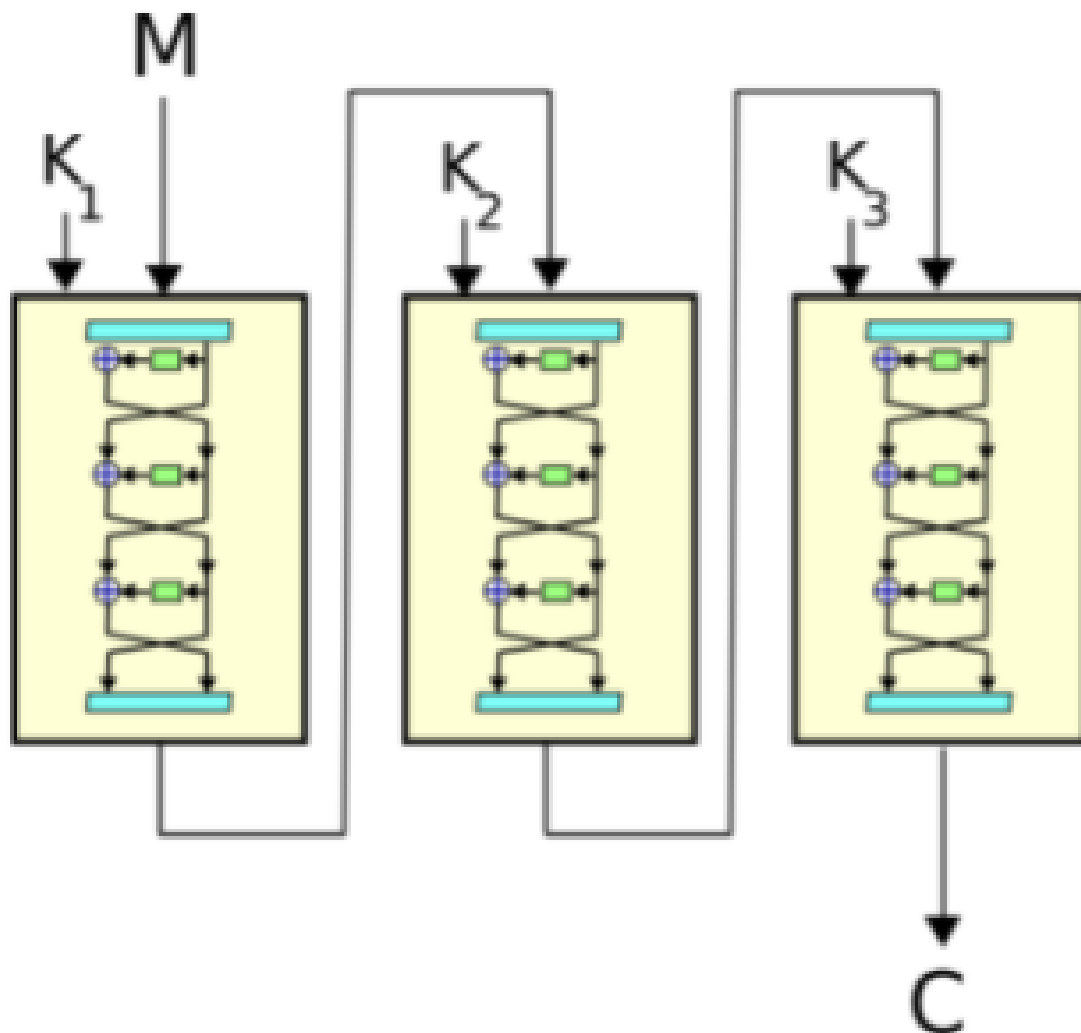


Рисунок 21 – Схема трехэтапного шифрования

### ***Исследование 3-DES в Cryptool 1.***

Из открытых источников возьмем текст примерно в 1000 символов (см. рис. 22).

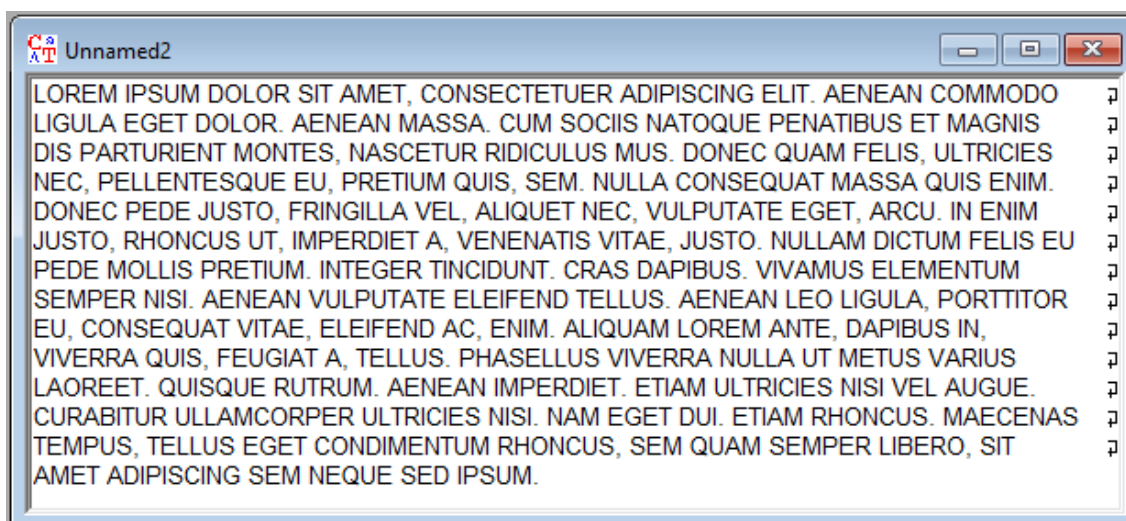


Рисунок 22 – Исходный текст

Получим бинарный файл, путем шифрации и дешифрации DES с нуле-  
 ВЫМ КЛЮЧОМ.

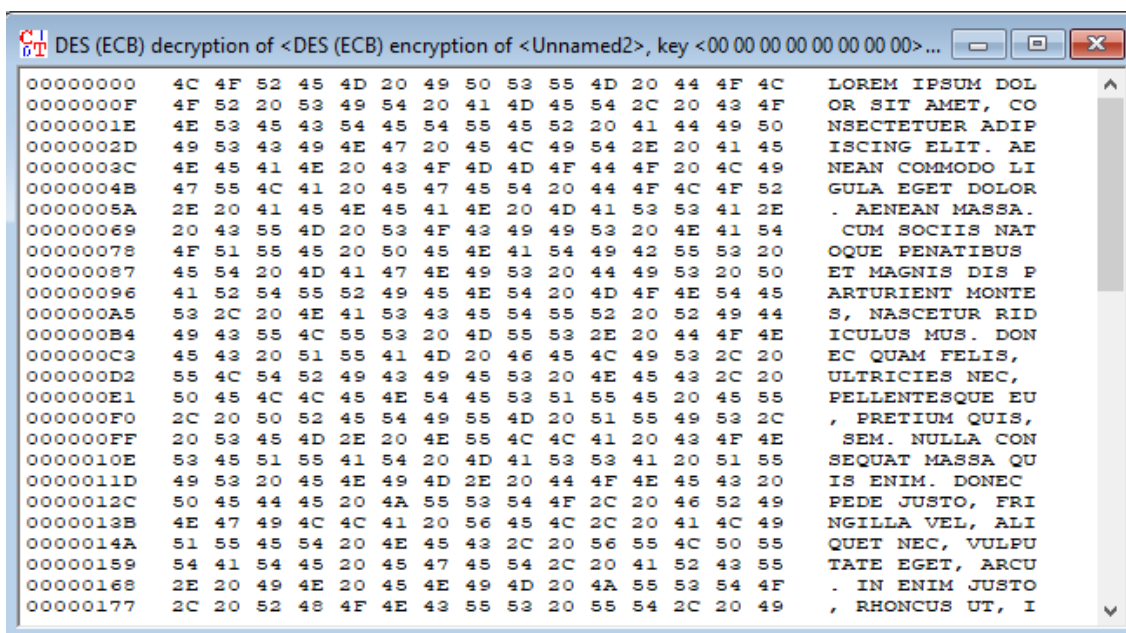


Рисунок 23 – Бинарный файл

Графики частотной и автокорреляционной характеристик открытого тек-  
 ста приведены на рисунках 24 и 25.

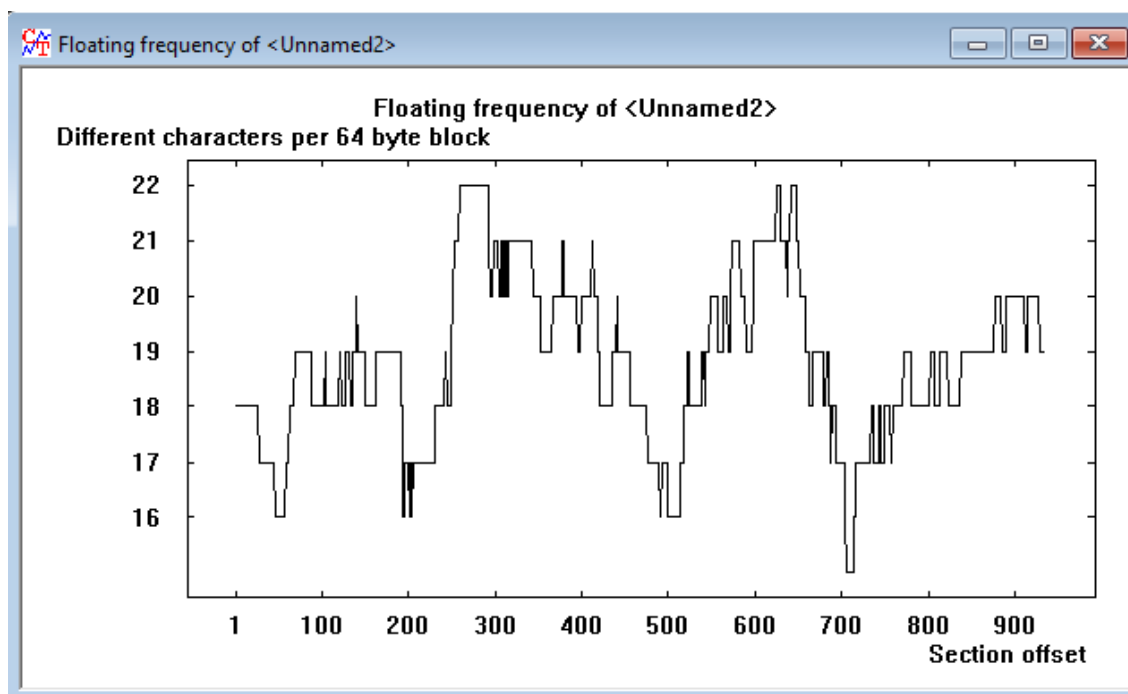


Рисунок 24 – График частотной характеристики исходного текста

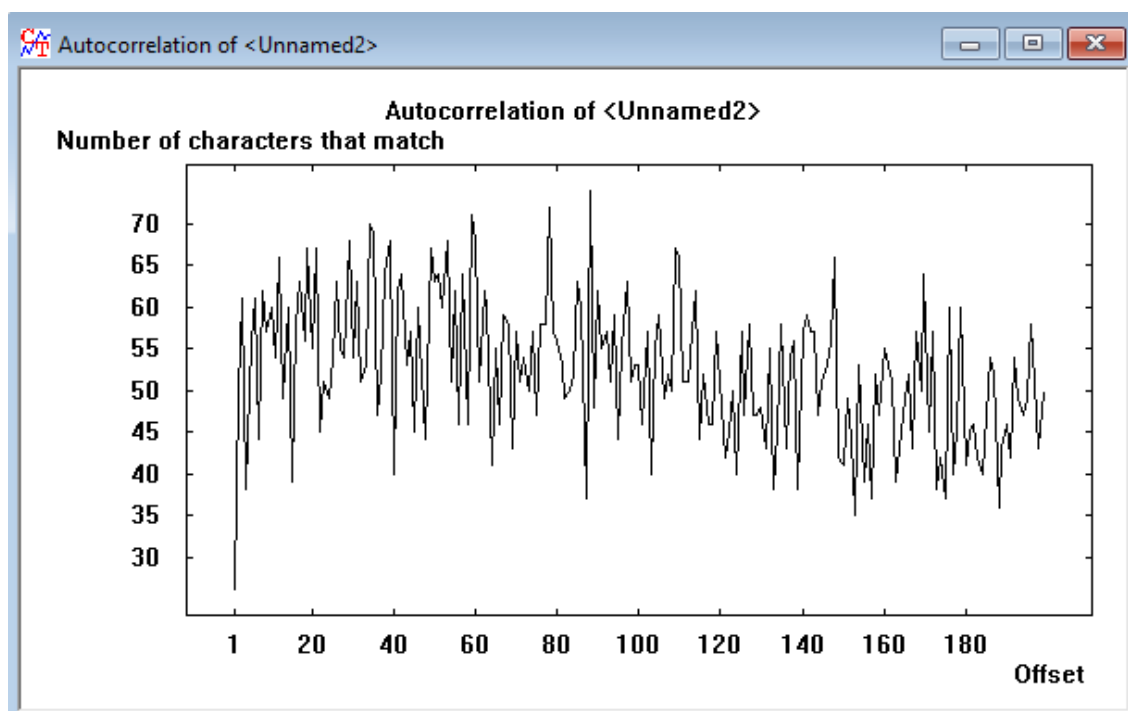


Рисунок 25 – График автокорреляционной характеристики исходного текста

Для шифрования 3-DES выберем ключ 55 66 77 88 99 13 37 13 37 16 13 16 13 11 22 33.



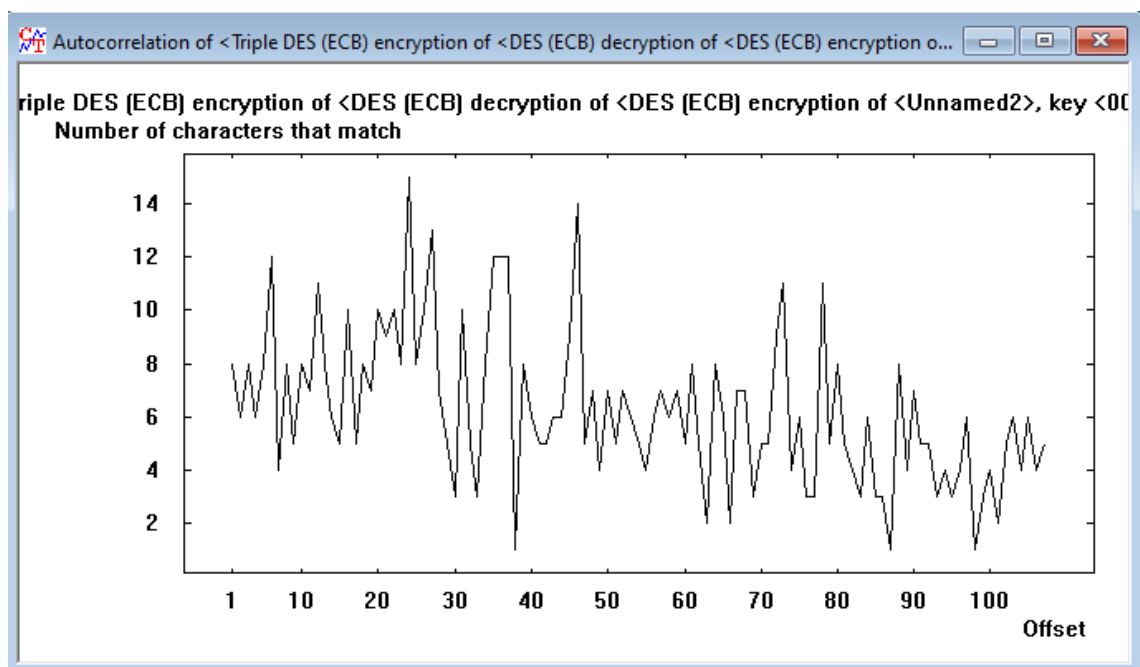
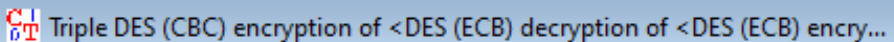


Рисунок 28 – График автокорреляционной характеристики 3-DES(ECB)

Повторим действия в режиме CBC с тем же ключом (см. рис. 29, 30 и 31).





00000000	6A 18 05 FA 33 1A 72 18 44 3B F8 A6	j...3.r.Dr...
0000000C	B6 79 F5 F4 EC 4C D9 AD 10 07 98 06	.y...L.....
00000018	5D 4D 21 0E 09 16 9F 82 63 BA B6 0D	]M!.....c...
00000024	3C 54 30 69 25 90 A6 B2 16 46 6C 42	<T0i%....FlB
00000030	C2 C7 6A 4D 88 FA 3A 1E 82 16 F9 A9	..jM.....
0000003C	34 6F 4D D7 D5 FE E5 31 ED 6B 87 3C	4oM....l.k.<
00000048	23 9A 60 89 9A B1 1B EB 59 6B B9 FE	#.'.....Yk..
00000054	26 35 66 39 0B 2D 9D C8 B6 F0 09 E1	\$5f9.-.....
00000060	95 DA 00 56 78 24 4E 2B F3 FC 91 DF	...Vx\$N+....
0000006C	18 E8 10 B1 F6 A6 27 46 49 5B 40 A8	.....'FI[ @.
00000078	B3 1B 96 6D 53 81 47 56 01 14 15 85	...mS.GV....
00000084	B5 F5 F6 6F AD 5A 06 66 94 F3 CC 77	...o.Z.f...w
00000090	A6 F9 0E 2B CB 70 19 B4 8B 56 03 9C	...+p...V..
0000009C	84 5C 56 60 07 79 D7 05 FD 6F 6E 25	.\V`.y...on%
000000A8	6F 01 32 69 D1 80 E1 64 F2 13 56 76	o.2i...d..Vv
000000B4	27 4A 83 EE CF 4E 05 41 85 C3 15 5D	'J...N.A...]
000000C0	DD 64 4A 52 71 72 E4 53 92 C8 4D B9	.dJRqr.S...M.
000000CC	BA 38 A5 41 CE 2E 4C 1B 34 22 23 BE	.8.A..L.4"#.
000000D8	50 C2 00 AC 9D 14 44 83 EB 61 D4 A7	P.....D...a..
000000E4	75 09 D2 D7 02 56 8E 89 F5 9B 14 FA	u....V.....
000000F0	82 67 FE D3 11 CF C0 3A 2A 13 5B 77	.g.....*: [w
000000FC	DC B8 89 28 33 4A 35 4B F7 96 DC B4	... (3J5K...
00000108	1C 9B 68 1D 2D E8 A8 E6 0A 26 73 AB	..h.-....\$s.
00000114	69 18 22 B5 28 34 7F 76 08 5E 59 49	i.". (4.v.^YI
00000120	39 48 BC 32 15 CA 61 81 A6 49 CC 7A	9H.2...a...I.s
0000012C	5B 38 E6 21 6E DF 27 49 82 3C 1C 09	[8.!n.'I.<...
00000138	A3 3B 55 1E 20 D3 1E C9 A9 DB 8F 9B	..:U. ....
.....	-- -- -- -- -- -- -- -- -- -- -- --	

Рисунок 29 – 3-DES(CBC) шифровка

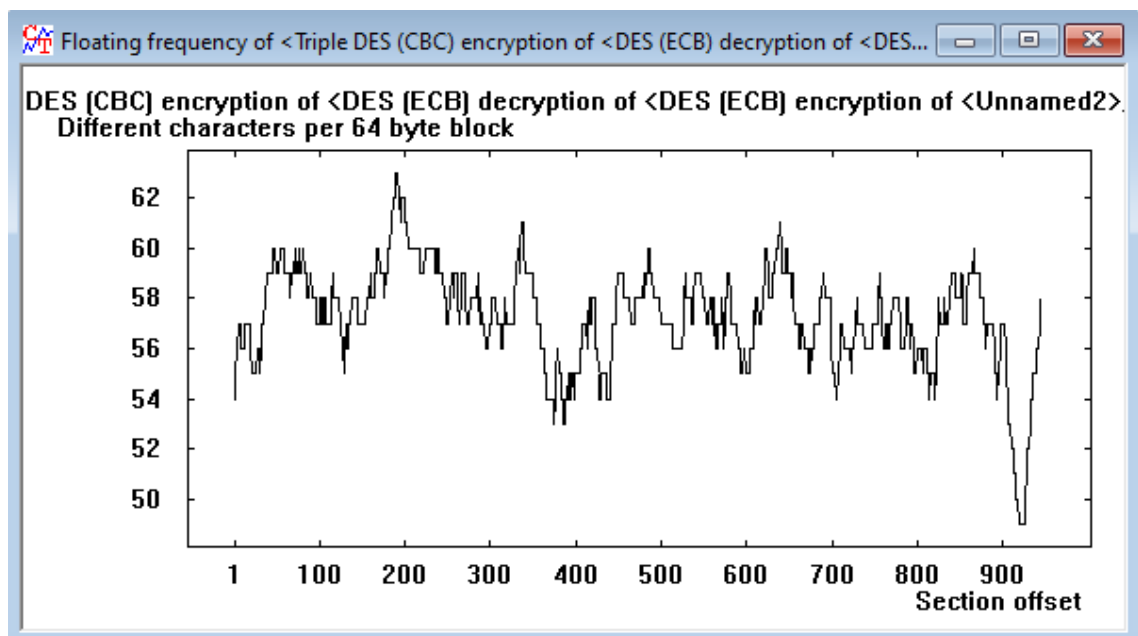


Рисунок 30 – График частотной характеристики 3-DES(CBC)

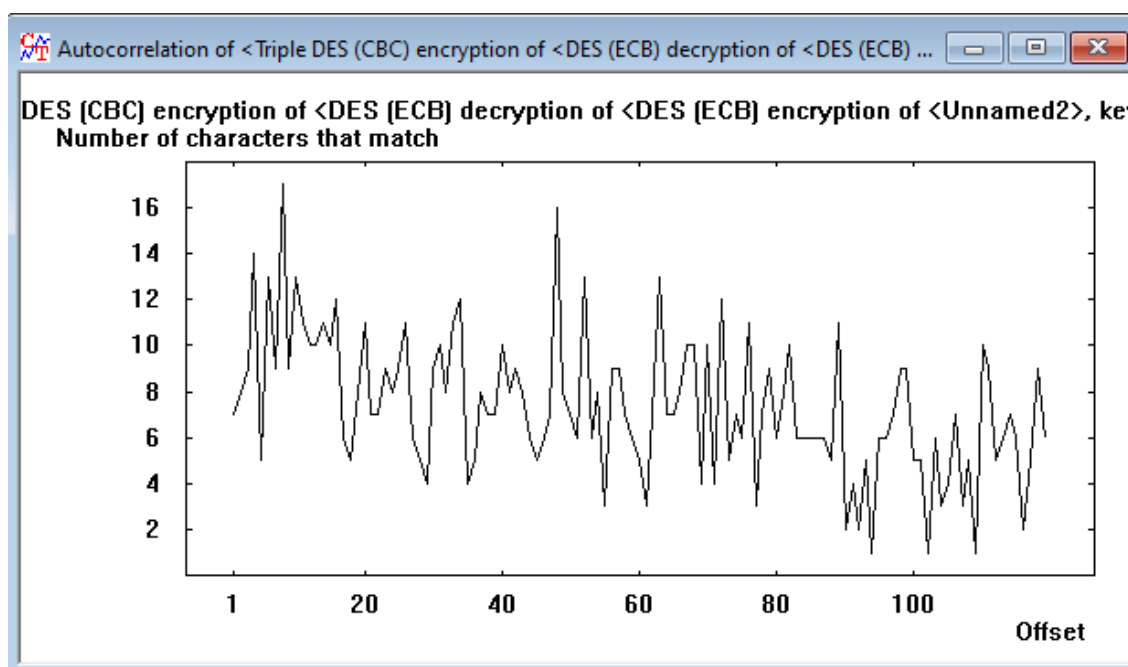


Рисунок 31 – График автокорреляционной характеристики 3-DES(CBC)

Таблица зависимости времени атаки грубой силы от размера известной части ключа приведена в таблице 3.

Таблица 3 – Сравнение времени взлома

Известно байт	3-DES ECB	3-DES CBC
0	2.8e21 лет	3e21 лет
4	1e13 лет	1.5e13 лет
8	48000 лет	60000 лет
14	около секунды	около секунды

Зашифруем исходный текст алгоритмами DES и 3-DES в Cryptool 1 с нулевым ключом. Сравним полученные тексты (см. рис. 32).

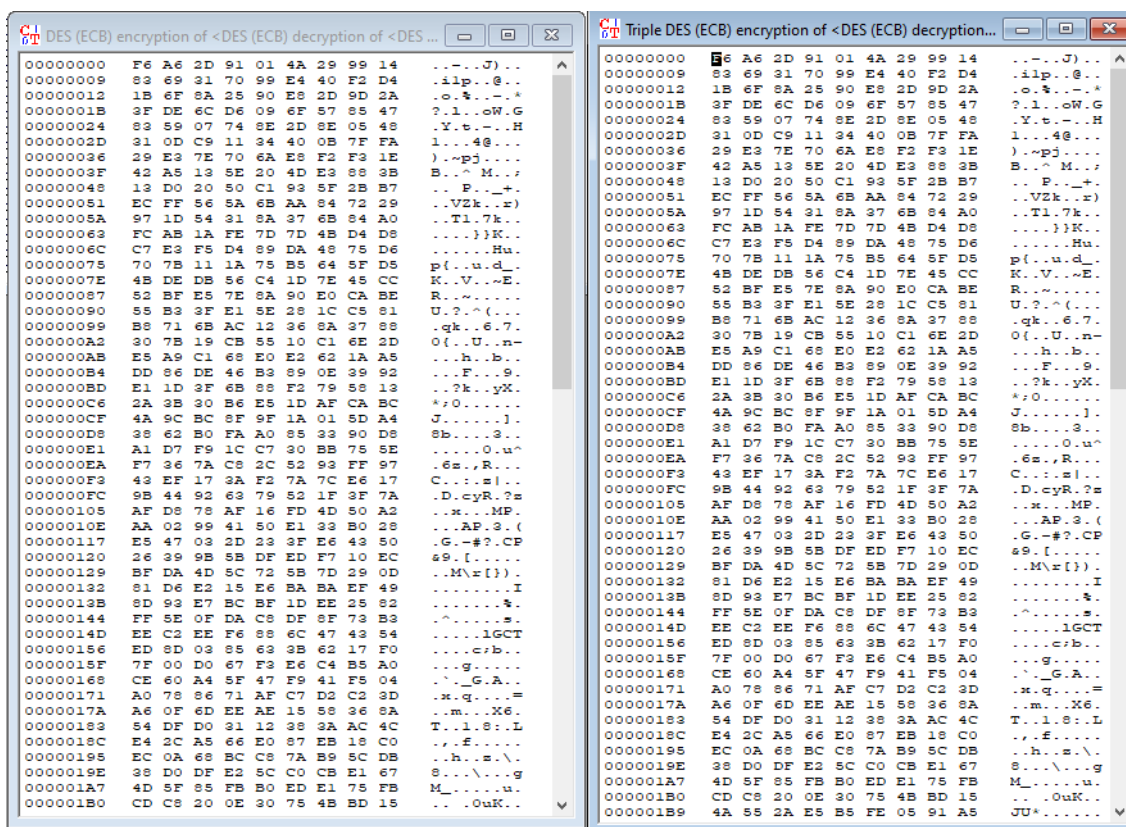


Рисунок 32 – Шифротексты DES и 3-DES с нулевым ключом

Так как тексты совпали, можно сделать вывод, что в Cryptool реализован алгоритм DES-EDE2.

### **Вывод.**

Был зашифрован исходный текст в 1000 символов. Для алгоритмов 3-DES CBC и 3-DES ECB были построены графики частотности и автокорреляции. Алгоритм 3-DES CBC оказался более криптостойким, чем 3-DES ECB. В Cryptool 1 применяется модификация DES-EDE2.

## **Исследование модификаций DESX, DESL, DESXL шифра DES.**

### **Задание.**

1. Выбрать случайный текст на английском языке (не менее 1000 знаков).
2. Создать бинарный файл с этим текстом, зашифровав и расшифровав его

DES на 0-м ключе.

3. С помощью CrypTool зашифровать текст с использованием шифров DESX, DESL, DESXL.
4. Средствами CrypTool вычислить энтропию исходного текста и шифротекстов, полученных в итоге. Зафиксировать результаты измерений в таблице.
5. Средствами CrypTool оцените время проведения атаки «грубой силы» при полном отсутствии информации о секретном ключе

### ***Описание модификаций DESX, DESL, DESXL шифра DES.***

Алгоритм DESX использует метод «отбеливания» ключа с целью усиления устойчивости к атакам на основе полного перебора. Для достижения поставленной цели алгоритм использует на входе ключ длиной 184 бита, который делится на 3 56-битные части. Процесс шифрования происходит по следующей схеме:

$$\text{DESX}(M) = K_2 \oplus \text{DES}_K(M \oplus K_1)$$

Если  $K_1 = K_2 = 0$ , то данный алгоритм сводится к стандартному DES. Возможно использовать ключ переменной длины, предварительно применив к нему хеширование SHA-1. Так же возможен вариант шифрования когда  $K_1 = K_2$ , тогда длина ключа будет составлять 128 бит. Еще один вариант шифрования предполагает использование 128-битного ключа, если  $K_2$  будет являться функцией от всех 16 байт ключа. Так же возможен случай, когда сложение происходит не по модулю 2, а по модулю 264.

Скорость алгоритма примерна равна скорости алгоритма DES.

Эффективная длина ключа составляет  $56 + 64 - 1 - \log M = 119$ , где  $M$  – число известных пар открытый текст/шифротекст. Также длина ключа падает до 88 бит при 232 известных открытых текстов и при использовании

атаки на основе адаптивно подобранного открытого текста. По этой причине иногда используется реализация с ключом  $K_2$ , являющимся односторонней функцией от  $K_1$  и  $K$ .

Алгоритм DESL является облегченной версией алгоритма DES. Данный алгоритм был создан в 2006 году для RFID-меток. Алгоритм предполагает отказ от входной и выходной перестановки блока текста, т.к. они не несут криптографической сложности, а также 8 S-блоков заменяется на 1, но многократно более стойкие чем все 8 стандартных блока DES. Благодаря указанным изменениям достигается увеличение скорости обработки данных и уменьшение места для хранения шифра без потери надежности шифрования – это необходимо из-за крайне ограниченных ресурсов системы.

Алгоритм DESXL использует те же оптимизации что и DESL, но производит шифрование по алгоритму DESX. Алгоритм так же используется для RFID-меток.

### ***Исследование модификаций DESX, DESL, DESXL шифра DES в Cryptool 1.***

В качестве исходного текста оставим текст из прошлого раздела.

Зашифруем исходный текст алгоритмом DESX с ключом 13 37 13 37 16 13 16 13 AA BB AA BB 00 11 22 33 44 55 66 77 88 99 13 37, алгоритмом DESL с ключом 12 34 56 78 9A BC DE F0 и алгоритмом DESXL с ключом как у DESX. Результаты шифрования приведены на рисунках 33, 34 и 35.

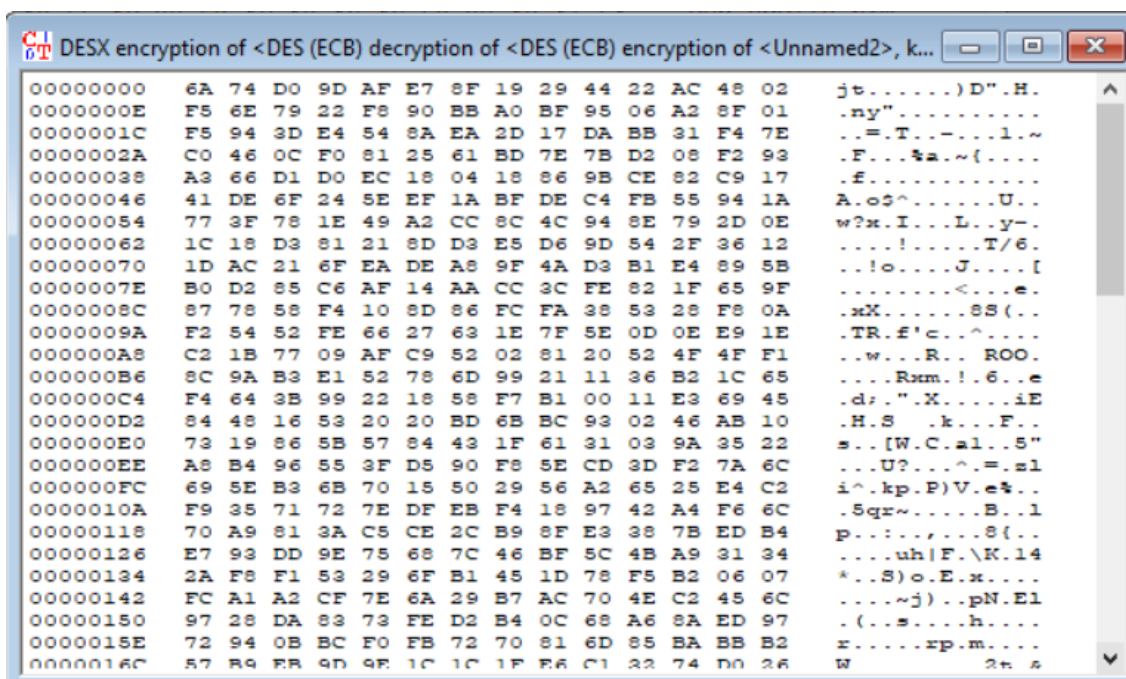


Рисунок 33 – Шифротекст DESX

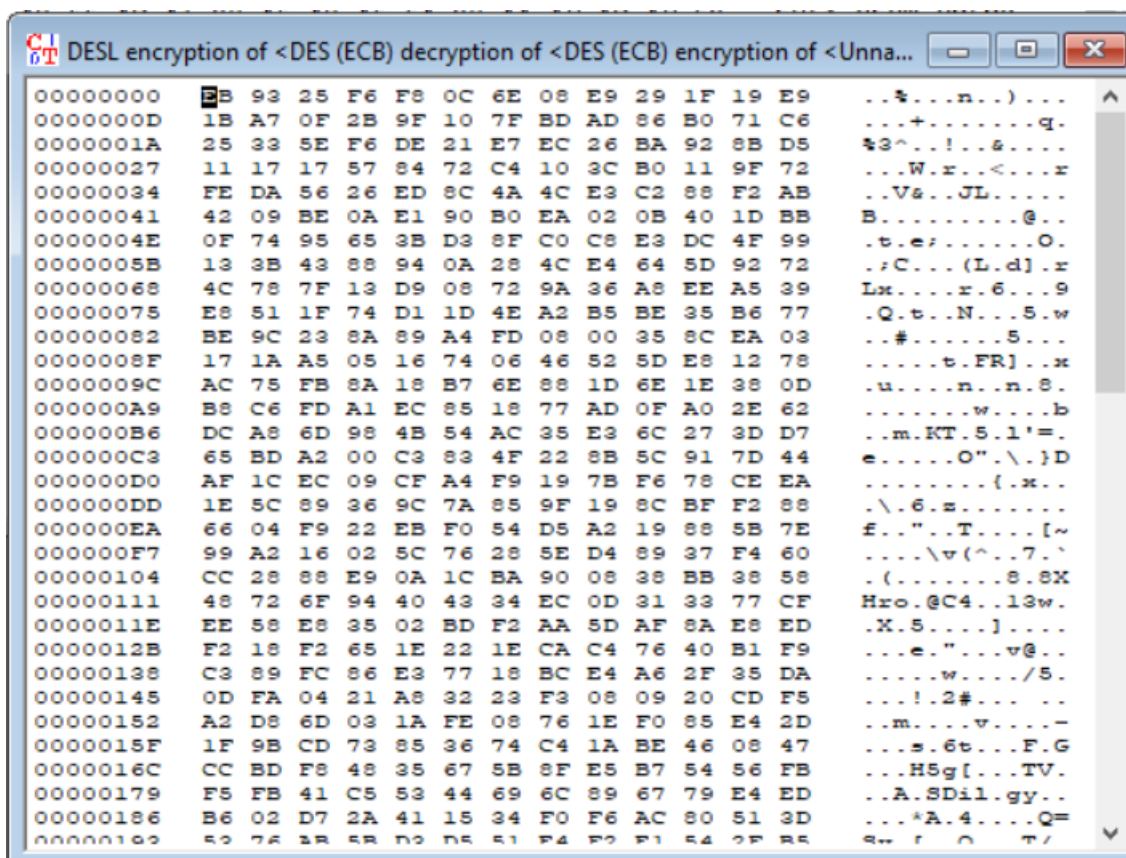


Рисунок 34 – Шифротекст DESL





ло достигнуто с использованием алгоритма DESX. Наиболее криптостойким (согласно эстимации времени атаки) оказался алгоритм DESX, хотя порядок времени атаки на шифротекст алгоритма DESXL имеет тот же порядок.



## **Заключение.**

В результате выполнения лабораторной работы было исследовано семейство шифров DES (DES, 3-DES, DESX, DESL, DESXL) и были получены практические навыки работы с ними в Cryptool 1. В работе шифры применялись для текста размером примерно в 1000 символов и для монохромного bmp изображения. Для каждого алгоритма средствами Cryptool 1 была получена оценка сложности атаки грубой силы.

Были рассмотрены 2 режима: ECB и CBC. Преимущество первого заключается в возможности параллелизации вычислений, второго – увеличенная энтропия шифротекста и лучшая криптостойкость.

Были рассмотрены модификации 3-DES, которые позволяют увеличить стойкость ключа. Принцип работы заключается в троекратном применении DES с разными ключами.

Были исследованы модификации DESX, DESL и DESXL. Последние две позволяют уменьшить объем вычислений, сохраняя достаточную криптостойкость.