

Санкт-Петербургский Политехнический Университет Петра  
Великого  
Институт компьютерных наук и технологий  
Кафедра компьютерных систем и программных технологий

Телекоммуникационные технологии

Отчет по лабораторной работе №7

Помехоустойчивое кодирование

**Работу выполнил:**  
Вашуров А., гр 33501.4  
**Преподаватель:**  
Богач Н.В.

Санкт-Петербург  
2017

## Содержание

1. Цель работы.....	3
2. Постановка задачи.....	3
3. Теоретический раздел.....	3
4. Ход работы.....	5
5. Вывод.....	8
6. Приложение.....	8

## **1. Цель работы**

Изучение методов помехоустойчивого кодирования и сравнение их свойств.

## **2. Постановка задачи**

1) Провести кодирование/декодирование сигнала, полученного с помощью функции `randerr` кодом Хэмминга 2-мя способами: с помощью встроенных функций `encode/decode`, а также через создание проверочной и генераторной матриц и вычисление синдрома. Оценить корректирующую способность кода.

2) Выполнить кодирование/декодирование циклическим кодом, кодом БЧХ, кодом Рида-Соломона. Оценить корректирующую способность кода.

## **3. Теоретический раздел**

### **3.1 Кодирование**

Физическое кодирование — преобразование двоичных данных, осуществляемое для их передачи по физическому каналу. Физическое кодирование может менять форму, ширину полосы частот и гармонический состав сигнала в целях осуществления синхронизации приёмника и передатчика, устранения постоянной составляющей или уменьшения аппаратных затрат передачи сигнала.

Обнаружение ошибок в технике связи — действие, направленное на контроль целостности данных при записи/воспроизведении информации или при её передаче по линиям связи. Исправление ошибок — процедура восстановления информации после чтения её из устройства хранения или канала связи.

Для обнаружения ошибок используют коды обнаружения ошибок, для исправления — корректирующие коды (коды, исправляющие ошибки, коды с коррекцией ошибок, помехоустойчивые коды).

#### **3.1.1 Кодирование Хэмминга**

Коды Хемминга — линейные коды с минимальным расстоянием 3, то есть способные исправить одну ошибку. При кодировании используется порождающая матрица  $G$

$\text{code} = \text{msg} * G$  (1) При декодировании используется проверочная матрица  $H$ , которая позволяет определить синдром  $S$ .

$$S = \text{code} * H^T \quad (2)$$

Синдром позволяет определить в какой позиции произошла ошибка. Коды Хэмминга являются самокорректирующимися кодами, то есть кодами, позволяющими автоматически обнаруживать ошибки при передаче данных и исправлять их.

Для построения самокорректирующегося кода, рассчитанного на исправление одиночных ошибок, одного контрольного разряда недостаточно. Как видно из дальнейшего, количество контрольных разрядов  $k$  должно быть выбрано так, чтобы удовлетворялось неравенство  $2^k \geq k+m+1$  (3)

или  $k \geq \log_2(k + m + 1)$  (4), где  $m$  — количество основных двоичных разрядов кодового слова.

Построение кодов Хэмминга основано на принципе проверки на четность числа единичных символов: к последовательности добавляется такой элемент, чтобы число единичных символов в получившейся последовательности было четным.

$$r_1 = i_1 \oplus i_2 \oplus \dots \oplus i_k \quad (5)$$

$$S = i_1 \oplus i_2 \oplus \dots \oplus i_n \oplus r_1 \quad (6)$$

Тогда если  $S = 0$  - ошибки нет, иначе есть однократная ошибка. Такой код называется  $(k + 1, k)$ . Первое число — количество элементов последовательности, второе — количество информационных символов. Получение кодового слова выглядит следующим образом:

$$(i_1 \ i_2 \ i_3 \ i_4) \begin{pmatrix} 1 & 0 & 0 & 0 & 1 & 0 & 1 \\ 0 & 1 & 0 & 0 & 1 & 1 & 1 \\ 0 & 0 & 1 & 0 & 1 & 1 & 0 \\ 0 & 0 & 0 & 1 & 0 & 1 & 1 \end{pmatrix} = (i_1 \ i_2 \ i_3 \ i_4 \ r_1 \ r_2 \ r_3) \quad (7)$$

Получение синдрома выглядит следующим образом:

$$(i_1 \ i_2 \ i_3 \ i_4 \ r_1 \ r_2 \ r_3) \begin{pmatrix} 1 & 0 & 1 \\ 1 & 1 & 1 \\ 1 & 1 & 0 \\ 0 & 1 & 1 \\ 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix} = (S_1 \ S_2 \ S_3) \quad (8)$$

### 3.1.2. Циклические коды

Циклический код — линейный код, обладающий свойством цикличности, то есть каждая циклическая перестановка кодового слова также является кодовым словом. Используется для преобразования информации для защиты её от ошибок.

### 3.1.3. Коды БЧХ

Коды Боуза — Чоудхури — Хоквингема (БЧХ-коды) — в теории кодирования это широкий класс циклических кодов, применяемых для защиты информации от ошибок. Отличается возможностью построения кода с заранее определёнными корректирующими свойствами, а именно, минимальным кодовым расстоянием. Частным случаем БЧХ-кодов является код Рида — Соломона.

### 3.2.3 Коды Рида-Соломона

Коды Рида—Соломона (англ. Reed–Solomon codes) — не двоичные циклические коды, позволяющие исправлять ошибки в блоках данных. Элементами кодового вектора являются не биты, а группы битов (блоки). Код Рида—Соломона является частным случаем БЧХ-кода.

## 4 Ход работы

Листинг программы представлен в Приложении

### 4.1 Коды Хэмминга

Сообщение и код, полученный с помощью encode (кодирование Хэмминга (7,4)):

Исходное сообщение 1100

Кодированное сообщение 1011100

Декодированное сообщение 1100

## 4.2 Циклические коды

Исходное сообщение 1100

Кодированное сообщение 0101100

Декодированное сообщение 1100

При кодировании сообщений с кодовым расстоянием 1, получаются закодированные сообщения с кодовым расстоянием 3.

## 4.3 Коды БЧХ

Исходное сообщение

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

Закодированное сообщение

Columns 1 through 6

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

Columns 7 through 12

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

Columns 13 through 15

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

Декодированное сообщение

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

#### 4.4. Коды Рида-Соломона

Код Рида-Соломона позволяет кодировать не только двоичные сообщения. При кодировании сообщений с кодовым расстоянием 1, получаются закодированные сообщения с кодовым расстоянием 3 или 4.

Исходное сообщение

2	7	3
4	0	6
5	1	1

Закодированное сообщение

Columns 1 through 6

2	7	3	3	6	7
4	0	6	4	2	2
5	1	1	4	5	4

Column 7

6
0
0

Декодированное сообщение

2	7	3
4	0	6
0	7	6

Число исправленных ошибок

1  
2  
-1

## 5 Выводы

В ходе работы было исследовано несколько способов кодирования информации (коды Хэмминга, Рида-Соломона, циклические коды, коды Боуза-Чоудхури-Хоквингема).

Кодирование используется не только для обеспечения безопасности при передаче сигнала, но и для защиты его от помех и улучшения качества передачи сигнала (наряду с модуляцией).

Рассмотренные коды являются корректирующимися, то есть способны предохранять от ошибок и исправлять их.

## 6 Приложение

```
function
lab7()

    close all
    format loose
    clc
    out = randerr(1,4);
    disp('Hamming')
    fprintf('Исходное сообщение\n');
    disp(out);
    code = encode (out, 7, 4, 'hamming/binary');
    fprintf('Кодированное сообщение\n');
    disp(code);
    dcode = decode (code, 7, 4, 'hamming/binary');
    fprintf('Декодированное сообщение\n');
    disp(dcode)

    msg = out;
    [h, gen] = hamngen(3);
    % gen = gen2par(h);
    code = rem(msg * gen, 2);
    trt = syndtable(h);
    syndrome = rem(code * h', 2);

    % error location:
    err = bi2de(fliplr(syndrome));
```



```
err_loc = trt(err + 1, :);
```

```
% corrected code
```

```
ccode = rem(err_loc + code, 2);
```

```
dcode = ccode(7 - 4 + 1:7);
```

```
fprintf('Порождающая матрица\n');  
disp(gen)  
fprintf('Проверочная матрица\n');  
disp(h)
```

```
fprintf('Исходное сообщение\n');  
disp(msg)  
fprintf('Кодированное сообщение\n');  
disp(code)  
fprintf('Синдром\n');  
disp(syndrome)  
fprintf('Скорректированное сообщение\n');  
disp(ccode)  
fprintf('Декодированное сообщение\n');  
disp(dcode)
```

```
disp('Cyclic')  
out = randerr(1,4);  
fprintf('Исходное сообщение\n');  
disp(out);  
code = encode (out, 7, 4, 'cyclic/binary');  
fprintf('Кодированное сообщение\n');  
disp (code);  
dcode = decode (code, 7, 4, 'cyclic/binary');  
fprintf('Декодированное сообщение\n');  
disp(dcode)
```

```
m = 4;  
n = 2^m-1;  
k = 5;  
nwords = 5;
```

```

msg = randi([0 1],nwords,k);
code = gf(msg);
[poly,t] = bchgenpoly(n,k);
enc = bchenc(code,n,k);
noisycode = enc + randerr(nwords,n,1:t);
dcode = bchdec(noisycode,n,k);

```

```

fprintf('Исходное сообщение\n');
disp(msg)
fprintf('Полином\n');
disp(poly.x)
fprintf('Кодированное сообщение\n');
disp(enc.x)
fprintf('Декодированное сообщение\n');
disp(dcode.x)
disp('Rid - Salomon')

```

```

m = 3;
n = 2^m - 1;
k = 3;

```

```

msg = gf([2 7 3; 4 0 6; 5 1 1],m);
code = rsenc(msg,n,k);

```

```

errors = gf([2 0 0 0 0 0; 3 4 0 0 0 0; 5 6 7 0 0 0],m);
noisycode = code + errors;

```

```

[dcode,cnumerr] = rsdec(noisycode,n,k);

```

```

fprintf('Исходное сообщение\n');
disp(msg.x)
fprintf('Кодированное сообщение\n');
disp(code.x)
fprintf('Декодированное сообщение\n');
disp(dcode.x)
fprintf('Число исправленных ошибок\n');
disp(cnumerr)
end

```