

Санкт-Петербургский политехнический  
университет Петра Великого  
Институт компьютерных наук и технологий  
Кафедра компьютерных систем и программных технологий

**Отчёт по лабораторной работе**

**Дисциплина:** Телекоммуникационные технологии

**Тема:** Помехоустойчивые коды

Выполнила студентка группы 33501/3

\_\_\_\_\_  
(подпись) Ивашкевич О.А.

Преподаватель

\_\_\_\_\_  
(подпись) Богач Н.В.

Санкт-Петербург  
2017

# 1 Цель работы

Изучение методов помехоустойчивого кодирования и сравнение их свойств

## 2 Постановка задачи

Провести кодирование/декодирование сигнала кодом Хэмминга 2мя способами с помощью встроенных функций encode/decode, а также через создание проверочной и генераторной матриц и вычисление синдрома. Провести кодирование/декодирование с помощью циклических кодов. Выполнить кодирование/-декодирование: циклическим кодом, кодом БЧХ, кодом Рида-Соломона.

## 3 Теоретические сведения

Кодирование информации необходимо для проверки полученного сообщения на наличие ошибок, так же некоторые коды могут исправлять ошибки. Данная функция достигается добавлением в сообщение дополнительной информации (информационной избыточности). Значительную долю кодов составляют блочные коды. При их применении передаваемое сообщение разбивается на блоки одинаковой длины, после чего каждому блоку сопоставляется код в соответствии с выбранным способом кодирования. Другая характеристика, позволяющая выделить коды в отдельный класс - цикличность. Цикличность — свойство, суть которого в том, что каждая циклическая перестановка кодового слова также является кодовым словом.

### 3.1 Код Хэмминга

Помимо информационных бит в сообщении передается набор контрольных бит, которые вычисляются как сумма по модулю 2 всех информационных бит, кроме одного. Для  $m$  контрольных бит максимальное число информационных бит составляет  $2^m - n - 1$ . Код Хэмминга позволяет обнаружить до двух ошибок при передаче и исправить инверсную передачу одного двоичного разряда.

### 3.2 Циклические коды

При рассмотрении циклических кодов двоичные числа представляют в виде многочлена, степень которого  $(n-1)$ , где  $n$  - длина кодовой комбинации. При таком представлении действия над кодовыми комбинациями сводятся к действиям над многочленами. Эти действия производятся в соответствии с обычной алгеброй, за исключением того, что приведение подобных членов осуществляется по модулю 2.

### 3.3 Коды БЧХ

Позволяют при необходимости исправлять большее число ошибок в разрядах за счет внесения дополнительной избыточности. Они принадлежат к категории блочных кодов.

### 3.4 Коды Рида-Соломона

Являются частным случаем кодов БЧХ, работают с не двоичными данными.

## 4 Ход работы

### 4.1 Код Хэмминга

Листинг кода приведен ниже:

```
%By Hamming
msg = [1 0 0 0] %signal
m=3; n = 2^m-1; k = n-m;
code = encode(msg,n,k,'hamming/ binary ') %coding
decoding = decode(code,n,k,'hamming/binary ') %decoding
%By matrix
[parmat,g,n,k] = hammgen(m); % matrix.
trt = syndtable(parmat); % table.
%recd = [1 1 0 1 0 0 0] % Suppose this is the received vector.
recd = msg*g % Suppose this is the received vector.
recd(1)=0; % some error
syndrome = rem(recd * parmat',2);
syndrome_de = bi2de(syndrome,'left-msb');
disp(['Syndrome = ',num2str(syndrome_de),...
      ' (decimal), ',num2str(syndrome),' (binary)'])
corrvect = trt(1+syndrome_de,:) % Correction vector
correctedcode = rem(corrvect+recd,2)
decoding = decode(correctedcode,n,k,'hamming/binary ') %decoding
```

Листинг 1: Кодирование Хэмминга

```
msg = 1      0      0      0
code = 1      1      0      1      0      0      0
decoding = 1      0      0      0
```

Листинг 2: Результат работы encode/decode

```
recd = 1      1      0      1      0      0      0
Syndrome = 4 (decimal), 1 0 0 (binary)
corrvect = 1      0      0      0      0      0      0
correctedcode = 1      1      0      1      0      0      0
decoding = 1      0      0      0
```

Листинг 3: Результат, полученный с использованием матрицы с исправлением ошибки

### 4.2 Циклический код

Листинг кода приведен ниже:

```
% Cycle code
message = [1 0 0 0];
clen=6;
mlen=4;
pol = cyclpoly(clen, mlen);
[h_cycle, g_cycle] = cyclgen(clen, pol);
codecycle = message*g_cycle;
codecycle = rem(codecycle, ones(1,clen).*2)
decode_message = [code(4), code(5), code(6), code(7)]
syndrome = rem(codecycle*h_cycle', ones(1,clen-mlen).*2)
```

Листинг 4: Циклический код

```
codecycle = 1      0      1      0      0      0
decode_message = 1      0      0      0
syndrome = 0      0
```

Листинг 5: Результат циклического кодирования

### 4.3 Код БЧХ

Листинг кода приведен ниже:

```
msg = [1 0 0 0]
code_msg = comm.BCHEncoder(7,4)
decode_msg = comm.BCHDecoder(7,4)
temp = msg'; % transpose
code = step (code_msg , temp(:))'
code(2) = ~ code(4) % some error
decode = step (decode_msg , code')'
```

Листинг 6: Код БЧХ

```
code = 1      0      0      0      1      0      1
code = 1      1      0      0      1      0      1
decode = 1      0      0      0
```

Листинг 7: Результат работы кода БЧХ

### 4.4 Код Рида-Соломона

Листинг кода приведен ниже:

```
fprintf('RS code ');
code_rs = comm.RSEncoder(6,4);
dec_rs = comm.RSDecoder(6,4);
temp = message';
code = step(code_rs, temp(:))'
decode = step(dec_rs, code')'
```

Листинг 8: Код Рида-Соломона

```
code = 1      0      0      0      1      1
decode = 1      0      0      0
```

Листинг 9: Результат работы кода Рида-Соломона

## 5 Выводы

Методы помехоустойчивого кодирования используют избыточную информацию, которая необходима для проверки наличия ошибок в передаваемой информации, что обеспечивает надежность передачи сообщений. Код Хэмминга является самым простым по реализации из рассмотренных способов кодирования. Другие коды имеют более сложную реализацию, но обладают улучшенными корректирующими свойствами и большей надежностью.