

МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РОССИЙСКОЙ ФЕДЕРАЦИИ

ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ АВТОНОМНОЕ ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ  
ВЫСШЕГО ОБРАЗОВАНИЯ

«Санкт-Петербургский национальный исследовательский университет информационных  
технологий, механики и оптики»

ФАКУЛЬТЕТ ПРОГРАММНОЙ ИНЖЕНЕРИИ И КОМПЬЮТЕРНОЙ  
ТЕХНИКИ

**ЛАБОРАТОРНАЯ РАБОТА №2**  
**Синтез помехоустойчивого кода**  
по дисциплине  
**«ИНФОРМАТИКА»**

Вариант №19

***Выполнил:***

Студент группы Р3131  
Варфоломеева  
Марина Николаевна

***Проверил:***

Преподаватель практики  
Марухленко  
Даниил Сергеевич

# Оглавление

Оглавление .....	2
Задание .....	3
Основные этапы вычисления .....	4
Дополнительное задание .....	7
Вывод.....	7
Список литературы .....	9

# Задание

1. На основании номера варианта задания выбрать набор из 4 полученных сообщений в виде последовательности 7-символьного кода.
2. Построить схему декодирования классического кода Хэмминга (7;4), которую представить в отчёте в виде изображения.
3. Показать, исходя из выбранных вариантов сообщений (по 4 у каждого часть №1 в варианте), имеются ли в принятом сообщении ошибки, и если имеются, то какие. Подробно прокомментировать и записать правильное сообщение.
4. На основании номера варианта задания выбрать 1 полученное сообщение в виде последовательности 15-символьного кода.
5. Построить схему декодирования классического кода Хэмминга (15;11), которую представить в отчёте в виде изображения.
6. Показать, исходя из выбранного варианта сообщений (по 1 у каждого часть №2 в варианте), имеются ли в принятом сообщении ошибки, и если имеются, то какие. Подробно прокомментировать и записать правильное сообщение.
7. Сложить номера всех 5 вариантов заданий. Умножить полученное число на 4. Принять данное число как число информационных разрядов в передаваемом сообщении. Вычислить для данного числа минимальное число проверочных разрядов и коэффициент избыточности.
8. Дополнительное задание №1 (позволяет набрать от 86 до 100 процентов от максимального числа баллов БаРС за данную лабораторную). Сделать себе учётную запись на <https://gitlab.se.ifmo.ru/>.
9. Написать программу на любом языке программирования, которая на вход получает набор из 7 цифр «0» и «1», записанных подряд, анализирует это сообщение на основе классического кода Хэмминга (7,4), а затем выдает правильное сообщение (только информационные биты) и указывает бит с ошибкой при его наличии

# Основные этапы вычисления

1. №32:

r1	r2	i1	r3	i2	i3	i4
0	0	1	1	0	1	0

1	2	3	4	5	6	7	
r1	r2	i1	r3	i2	i3	i4	S
X		X		X		X	S1
	X	X			X	X	S2
			X	X	X	X	S3

$$S_1 = r1 \oplus i1 \oplus i2 \oplus i4 = 0 \oplus 1 \oplus 0 \oplus 0 = 1$$

$$S_2 = r2 \oplus i1 \oplus i3 \oplus i4 = 0 \oplus 1 \oplus 1 \oplus 0 = 0$$

$$S_3 = r3 \oplus i2 \oplus i3 \oplus i4 = 1 \oplus 0 \oplus 1 \oplus 0 = 0$$

$$S = (S_1, S_2, S_3) = 001 \Rightarrow \text{ошибка в символе } r1$$

Правильное сообщение 1010

2. №54:

r1	r2	i1	r3	i2	i3	i4
1	1	0	1	0	1	1

1	2	3	4	5	6	7	
r1	r2	i1	r3	i2	i3	i4	S
X		X		X		X	S1
	X	X			X	X	S2
			X	X	X	X	S3

$$S_1 = r1 \oplus i1 \oplus i2 \oplus i4 = 1 \oplus 0 \oplus 0 \oplus 1 = 0$$

$$S_2 = r2 \oplus i1 \oplus i3 \oplus i4 = 1 \oplus 0 \oplus 1 \oplus 1 = 1$$

$$S_3 = r3 \oplus i2 \oplus i3 \oplus i4 = 1 \oplus 0 \oplus 1 \oplus 1 = 1$$

$$S = (S_1, S_2, S_3) = 110 \Rightarrow \text{ошибка в символе } i3$$

Правильное сообщение 0001

3. №76:

r1	r2	i1	r3	i2	i3	i4
0	1	1	0	1	0	1

1	2	3	4	5	6	7	
r1	r2	i1	r3	i2	i3	i4	S
X		X		X		X	S1
	X	X			X	X	S2
			X	X	X	X	S3

$S_1 = r1 \oplus i1 \oplus i2 \oplus i4 = 0 \oplus 1 \oplus 1 \oplus 1 = 1$   
 $S_2 = r2 \oplus i1 \oplus i3 \oplus i4 = 1 \oplus 1 \oplus 0 \oplus 1 = 1$   
 $S_3 = r3 \oplus i2 \oplus i3 \oplus i4 = 0 \oplus 1 \oplus 0 \oplus 1 = 0$   
 $S = (S_1, S_2, S_3) = 011 \Rightarrow$  ошибка в символе  $i1$   
 Правильное сообщение 0101

4. №61:

r1	r2	i1	r3	i2	i3	i4
0	1	0	0	1	0	0

1	2	3	4	5	6	7	
r1	r2	i1	r3	i2	i3	i4	S
X		X		X		X	S1
	X	X			X	X	S2
			X	X	X	X	S3

$S_1 = r1 \oplus i1 \oplus i2 \oplus i4 = 0 \oplus 0 \oplus 1 \oplus 0 = 1$   
 $S_2 = r2 \oplus i1 \oplus i3 \oplus i4 = 1 \oplus 0 \oplus 0 \oplus 0 = 1$   
 $S_3 = r3 \oplus i2 \oplus i3 \oplus i4 = 0 \oplus 1 \oplus 0 \oplus 0 = 1$   
 $S = (S_1, S_2, S_3) = 111 \Rightarrow$  ошибка в символе  $i4$   
 Правильное сообщение 0101

5. №20:

r1	r2	i1	r3	i2	i3	i4	r4	i5	i6	i7	i8	i9	i10	i11
0	1	1	0	0	0	1	0	1	0	0	0	0	0	1

1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	
r1	r2	i1	r3	i2	i3	i4	r4	i5	i6	i7	i8	i9	i10	i11	S
X		X		X		X		X		X		X		X	S1
	X	X			X	X			X	X			X	X	S2
			X	X	X	X					X	X	X	X	S3
							X	X	X	X	X	X	X	X	S4

$S_1 = r1 \oplus i1 \oplus i2 \oplus i4 \oplus i5 \oplus i7 \oplus i9 \oplus i11 = 0 \oplus 1 \oplus 0 \oplus 1 \oplus 1 \oplus 0 \oplus 0 \oplus 1 = 0$   
 $S_2 = r2 \oplus i1 \oplus i3 \oplus i4 \oplus i6 \oplus i7 \oplus i10 \oplus i11 = 1 \oplus 1 \oplus 0 \oplus 1 \oplus 0 \oplus 0 \oplus 0 \oplus 1 = 0$   
 $S_3 = r3 \oplus i2 \oplus i3 \oplus i4 \oplus i8 \oplus i9 \oplus i10 \oplus i11 = 0 \oplus 0 \oplus 0 \oplus 1 \oplus 0 \oplus 0 \oplus 0 \oplus 1 = 0$   
 $S_4 = r4 \oplus i5 \oplus i6 \oplus i7 \oplus i8 \oplus i9 \oplus i10 \oplus i11 = 0 \oplus 1 \oplus 0 \oplus 0 \oplus 0 \oplus 0 \oplus 0 \oplus 1 = 0$   
 $S = (S_1, S_2, S_3) = 0000 \Rightarrow$  ошибок нет  
 Правильное сообщение 10011000001

6.  $32+54+76+61+20 = 243 * 4 = 972$ :

Информационных разрядов в передаваемом сообщении: 972

Пусть будет  $r$  проверочных разрядов, тогда всего бит в сообщении:  $2^r \geq 1 + r + i$

Путем нехитрых вычислений (или банального перебора) получаем, что  $r = 10$ .  
Проверка:  $2^{10} \geq 1 + 10 + 972 \cdot 1024 \geq 983$ .

В таком случае коэффициент избыточности равен:

$$\frac{r}{n} = \frac{r}{i+r} = \frac{10}{982} \approx 0,0101833$$

Ответ:  $r = 10$ ,  $r/n \approx 0,0101833$

# Дополнительное задание

Написать программу на любом языке программирования, которая на вход получает набор из 7 цифр «0» и «1», записанных подряд, анализирует это сообщение на основе классического кода Хэмминга (7,4), а затем выдает правильное сообщение (только информационные биты) и указывает бит с ошибкой при его наличии

Листинг программы:

```
# =====#
#/      получим код Хэмминга (7, 4) и проверим ввод на наличие ошибок      /#
# =====#
message = input("Введите классический код Хэмминга (7, 4): ")
if len(message) != 7 or set(message) not in [{0, 1}, {1}, {0}]:
    raise ValueError('Код должен состоять ровно из 7 цифр в формате "0" и "1", без пробелов и запятых')

message = [int(i) for i in map(int, message)]
orig_message = ''.join(map(str, message))

# =====#
#/      Вычислим синдром для каждого контрольного бита      /#
# =====#

S1 = sum(message[::2])%2
S2 = sum([message[1], message[2], message[5], message[6]])%2
S3 = sum(message[3:])%2
S = int(str(S3)+str(S2)+str(S1), 2) # Номер цифры в десятичной системе счисления, в которой допущена ошибка

# =====#
#/      В зависимости от обнаружения ошибки выводим одно из этих сообщений      /#
# =====#

if S == 0:
    i = ''.join([message[2], message[4], message[5], message[6]])
    print(f"В коде {orig_message} нет ошибок. Исходное сообщение: {i}")
else:
    bits = ["r1", "r2", "i1", "r3", "i2", "i3", "i4"]
    message[S-1] = int(not(message[S-1]))
    message = ''.join(map(str, message))
    i = ''.join([message[2], message[4], message[5], message[6]])
    print(f"В коде {orig_message} есть ошибка в бите {bits[S-1]}. Исходное сообщение: {i}")
```

# Вывод

В процессе выполнения этой лабораторной работы я научилась работать с кодом Хэмминга, определять количество контрольных битов по количеству информационных, и высчитывать коэффициент избыточности.

# Список литературы

1. [https://www.decmtdzol.ru/articles/mtd\\_handbook.pdf](https://www.decmtdzol.ru/articles/mtd_handbook.pdf)
2. <https://libarch.nmu.org.ua/handle/GenofondUA/81131>
3. <https://fat-crocodile.livejournal.com/214584.html>