

Федеральное государственное автономное образовательное учреждение высшего образования
«Национальный исследовательский университет ИТМО»

Факультет программной инженерии и компьютерной техники

Лабораторная работа по информатике №2

Вариант №69

Выполнил:

Студент группы Р3106

Мельник Фёдор Александрович

Проверил:

Балакшин П.В.,

Кандидат технических наук, доцент ФПиКТ

Санкт-Петербург, 2024

Оглавление

Задание.....	3
Основные этапы вычисления	4
Часть №1	4
Часть №2.....	7
Часть №3.....	8
Дополнительное задание.....	9
Заключение.....	11
Список использованных источников	12

Задание

1. Определить свой вариант задания с помощью номера в ISU (он же номер студенческого билета). Вариантом является комбинация 3-й и 5-й цифр. Т.е. если номер в ISU = 123456, то вариант = 35.
2. На основании номера варианта задания выбрать набор из 4 полученных сообщений в виде последовательности 7-символьного кода.
3. Построить схему декодирования классического кода Хэмминга (7;4), которую представить в отчёте в виде изображения.
4. 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.
5. На основании номера варианта задания выбрать 1 полученное сообщение в виде последовательности 11-символьного кода.
6. Построить схему декодирования классического кода Хэмминга (15;11), которую представить в отчёте в виде изображения.
7. 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.
8. 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.
9. Дополнительное задание №1 (позволяет набрать от 86 до 100 процентов от максимального числа баллов БаРС за данную лабораторную). 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

Для решения задач использовались материалы изданий «Помехоустойчивое кодирование»^[1] и Помехоустойчивые коды^[2]

Часть №1

Схема декодирования и выполнения классического кода Хэмминга (7;4) представлена на [рисунке №1](#) и [рисунке №2](#) соответственно.

	1	2	3	4	5	6	7	
2^x	r_1	r_2	i_1	r_3	i_2	i_3	i_4	S
1	X		X		X		X	S_1
2		X	X			X	X	S_2
4				X	X	X	X	S_3

$$S_1 = r_1 \oplus i_1 \oplus i_2 \oplus i_4$$
$$S_2 = r_2 \oplus i_1 \oplus i_3 \oplus i_4$$
$$S_3 = r_3 \oplus i_2 \oplus i_3 \oplus i_4$$

Рисунок 1 - схема декодирования классического кода Хэмминга (7;4)

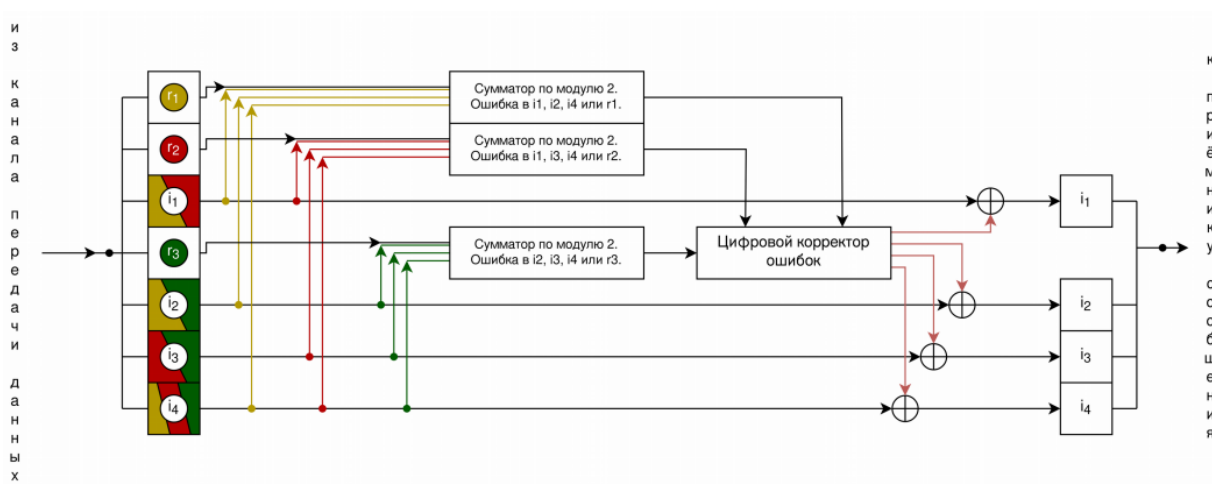


Рисунок 2 - схема выполнения классического кода Хэмминга (7;4)

Сообщение №1:

r_1	r_2	i_1	r_3	i_2	i_3	i_4
1	0	1	0	0	1	1

$$S_1 = r_1 \oplus i_1 \oplus i_2 \oplus i_4 = 1 \oplus 1 \oplus 0 \oplus 1 = 1$$

$$S_2 = r_2 \oplus i_1 \oplus i_3 \oplus i_4 = 0 \oplus 1 \oplus 1 \oplus 1 = 1$$

$$S_3 = r_3 \oplus i_2 \oplus i_3 \oplus i_4 = 0 \oplus 0 \oplus 1 \oplus 1 = 0$$

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

Синдром $S(S_1, S_2, S_3) = 110 \Rightarrow$ ошибка в i_1

Исправленный вариант: 1000011

Исходное сообщение без ошибки: 0011

Сообщение №2:

r_1	r_2	i_1	r_3	i_2	i_3	i_4
0	1	0	0	1	1	0

$$S_1 = r_1 \oplus i_1 \oplus i_2 \oplus i_4 = 0 \oplus 0 \oplus 1 \oplus 0 = 1$$

$$S_2 = r_2 \oplus i_1 \oplus i_3 \oplus i_4 = 1 \oplus 0 \oplus 1 \oplus 0 = 0$$

$$S_3 = r_3 \oplus i_2 \oplus i_3 \oplus i_4 = 0 \oplus 1 \oplus 1 \oplus 0 = 0$$

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

Синдром $S(S_1, S_2, S_3) = 100 \Rightarrow$ ошибка в r_1

Исправленный вариант: 1100110

Исходное сообщение без ошибки: 0110

Сообщение №3:

r_1	r_2	i_1	r_3	i_2	i_3	i_4
1	1	0	1	0	0	0

$$S_1 = r_1 \oplus i_1 \oplus i_2 \oplus i_4 = 1 \oplus 0 \oplus 0 \oplus 0 = 1$$

$$S_2 = r_2 \oplus i_1 \oplus i_3 \oplus i_4 = 1 \oplus 0 \oplus 0 \oplus 0 = 1$$

$$S_3 = r_3 \oplus i_2 \oplus i_3 \oplus i_4 = 1 \oplus 0 \oplus 0 \oplus 0 = 1$$

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

Синдром $S(S_1, S_2, S_3) = 111 \Rightarrow$ ошибка в i_4

Исправленный вариант: 110100**1**

Исходное сообщение без ошибки: 000**1**

Сообщение №4:

r_1	r_2	i_1	r_3	i_2	i_3	i_4
1	0	1	0	0	0	0

$$S_1 = r_1 \oplus i_1 \oplus i_2 \oplus i_4 = 1 \oplus 1 \oplus 0 \oplus 0 = 0$$

$$S_2 = r_2 \oplus i_1 \oplus i_3 \oplus i_4 = 0 \oplus 1 \oplus 0 \oplus 0 = 1$$

$$S_3 = r_3 \oplus i_2 \oplus i_3 \oplus i_4 = 0 \oplus 0 \oplus 0 \oplus 0 = 0$$

	1	2	3	4	5	6	7	
2^x	r_1	r_2	i_1	r_3	i_2	i_3	i_4	
1	X		X		X		X	S1
2		X	X			X	X	S2
4				X	X	X	X	S3

Синдром $S(S_1, S_2, S_3) = 010 \Rightarrow$ ошибка в r_2

Исправленный вариант: 1**1**10000

Исходное сообщение без ошибки: 1000

Часть №2

Схема декодирования и выполнения классического кода Хэмминга (15;11) представлена на [рисунке №3](#) и [рисунке №4](#) соответственно.

	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	
2^x	r_1	r_2	i_1	r_3	i_2	i_3	i_4	r_4	i_5	i_6	i_7	i_8	i_9	i_{10}	i_{11}	S
1	X		X		X		X		X		X		X		X	S_1
2		X	X			X	X			X	X			X	X	S_2
4				X	X	X	X					X	X	X	X	S_3
8								X	X	X	X	X	X	X	X	S_4

$$S_1 = r_1 \oplus i_1 \oplus i_2 \oplus i_4 \oplus i_5 \oplus i_7 \oplus i_9 \oplus i_{11}$$

$$S_2 = r_2 \oplus i_1 \oplus i_3 \oplus i_4 \oplus i_6 \oplus i_7 \oplus i_{10} \oplus i_{11}$$

$$S_3 = r_3 \oplus i_2 \oplus i_3 \oplus i_4 \oplus i_8 \oplus i_9 \oplus i_{10} \oplus i_{11}$$

$$S_4 = r_4 \oplus i_5 \oplus i_6 \oplus i_7 \oplus i_8 \oplus i_9 \oplus i_{10} \oplus i_{11}$$

Рисунок 3 - схема декодирования классического кода Хэмминга (15;11)

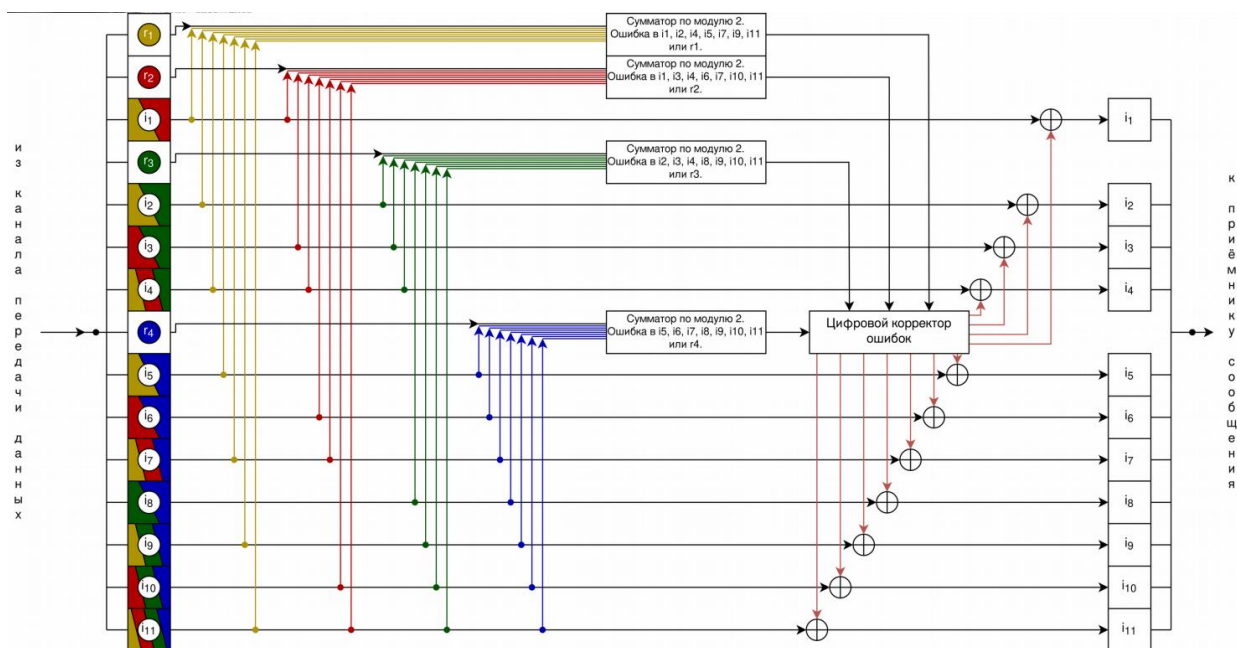


Рисунок 4 - Схема выполнения классического кода Хэмминга (15;11)

r_1	r_2	i_1	r_3	i_2	i_3	i_4	r_4	i_5	i_6	i_7	i_8	i_9	i_{10}	i_{11}
0	0	1	1	1	0	0	0	1	0	1	0	1	0	0

$$S_1 = r_1 \oplus i_1 \oplus i_2 \oplus i_4 \oplus i_5 \oplus i_7 \oplus i_9 \oplus i_{11} = 0 \oplus 1 \oplus 1 \oplus 0 \oplus 1 \oplus 1 \oplus 1 \oplus 0 = 1$$

$$S_2 = r_2 \oplus i_1 \oplus i_3 \oplus i_4 \oplus i_6 \oplus i_7 \oplus i_{10} \oplus i_{11} = 0 \oplus 1 \oplus 0 \oplus 0 \oplus 0 \oplus 1 \oplus 0 \oplus 0 = 0$$

$$S_3 = r_3 \oplus i_2 \oplus i_3 \oplus i_4 \oplus i_8 \oplus i_9 \oplus i_{10} \oplus i_{11} = 1 \oplus 1 \oplus 0 \oplus 0 \oplus 0 \oplus 1 \oplus 0 \oplus 0 = 1$$

$$S_4 = r_4 \oplus i_5 \oplus i_6 \oplus i_7 \oplus i_8 \oplus i_9 \oplus i_{10} \oplus i_{11} = 0 \oplus 1 \oplus 0 \oplus 1 \oplus 0 \oplus 1 \oplus 0 \oplus 0 = 1$$

Синдром $S(S_1, S_2, S_3, S_4) = 1011 \Rightarrow$ ошибка в i_9

Исправленный вариант: 001110001010000

Исходное сообщение без ошибки: 11001010000

Часть №3

Число информационных разрядов $(i) = (51 + 88 + 13 + 10 + 69) * 4 = 924$

$2^r \geq r + i + 1 \Rightarrow 1024 \geq 935 \Rightarrow$ Минимальное число проверочных разрядов $(r) = 10$

$$\text{Коэффициент избыточности} = \frac{r}{i+r} = \frac{10}{924+10} = \frac{5}{467} \approx 0,010707$$

Дополнительное задание

Решение задания представлено на языке программирования Python. Исходный код программы можно найти по ссылке:

<https://github.com/ldpst/itmo/blob/main/labs/sem1/inf/lab2/main.py>

```
from random import randint

def task(a):
    def get_s(s):
        if int(a[s[0]] ^ a[s[1]] ^ a[s[2]] ^ a[s[3]]) == 1:
            return s[0] + 1
        return 0

    error = get_s([0, 2, 4, 6]) + get_s([1, 2, 5, 6]) + get_s([3, 4, 5, 6]) -
1
    name = ["r1", "r2", "i1", "r3", "i2", "i3", "i4"]
    if error == -1:
        print("Ошибок нет")
        return
    a[error] = int(not a[error])
    print(f"Ошибка в бите {name[error]}. Сообщение без ошибок:
{"".join(list(map(str, [a[2], a[4], a[5], a[6]]))))}")

def test():
    print("1. Исходное сообщение: 1010011")
    task(list(map(int, list("1010011"))))
    print("2. Исходное сообщение: 0100110")
    task(list(map(int, list("0100110"))))
    print("3. Исходное сообщение: 1101000")
    task(list(map(int, list("1101000"))))
    print("4. Исходное сообщение: 1010000")
    task(list(map(int, list("1010000"))))
    for i in range(5, 11):
        s = [randint(0, 1), randint(0, 1), randint(0, 1), randint(0, 1),
randint(0, 1), randint(0, 1), randint(0, 1)]
        print(f"{i}. Исходное сообщение: {"".join(list(map(str, s)))}")
        task(s)

if __name__ == "__main__":
    test()
```

Результат выполнения программы на 4 числах из [части №1](#) и 6 случайных числах представлен на [рисунке №5](#).

```
1. Исходное сообщение: 1010011
Ошибка в бите i1. Сообщение без ошибок: 0011
2. Исходное сообщение: 0100110
Ошибка в бите r1. Сообщение без ошибок: 0110
3. Исходное сообщение: 1101000
Ошибка в бите i4. Сообщение без ошибок: 0001
4. Исходное сообщение: 1010000
Ошибка в бите r2. Сообщение без ошибок: 1000
5. Исходное сообщение: 1100110
Ошибок нет
6. Исходное сообщение: 1101000
Ошибка в бите i4. Сообщение без ошибок: 0001
7. Исходное сообщение: 0110101
Ошибка в бите i1. Сообщение без ошибок: 0101
8. Исходное сообщение: 0010111
Ошибка в бите i4. Сообщение без ошибок: 1110
9. Исходное сообщение: 1110111
Ошибка в бите r3. Сообщение без ошибок: 1111
10. Исходное сообщение: 0001100
Ошибка в бите r1. Сообщение без ошибок: 0100
```

Рисунок 5 - Результат выполнения программы

Заключение

В процессе выполнения лабораторной работы я познакомился с помехоустойчивым кодированием, узнал, какими преимуществами и недостатками оно обладает, научился строить таблицы декодирования классического хода Хэмминга и написал собственную программу на языке программирования Python для проверки сообщения на ошибки и исправления их.

Список использованных источников

1. Смирнов, В. А. Помехоустойчивое кодирование / В. А. Смирнов. — М.: БИНОМ. Лаборатория знаний, 2014. — 280 с.
2. Шевченко, А. А. Помехоустойчивые коды / А. А. Шевченко. — М.: Горячая линия — Телеком, 2018. — 350 с.