

НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ УНИВЕРСИТЕТ ИТМО

Факультет программной инженерии и компьютерной техники

Дисциплина информатика

Лабораторная работа № 2

Выполнил студент

Суджян Эдуард Эдуардович

Группа № Р3121

Преподаватель: Болдырева Елена Александровна

Санкт-Петербург

2023

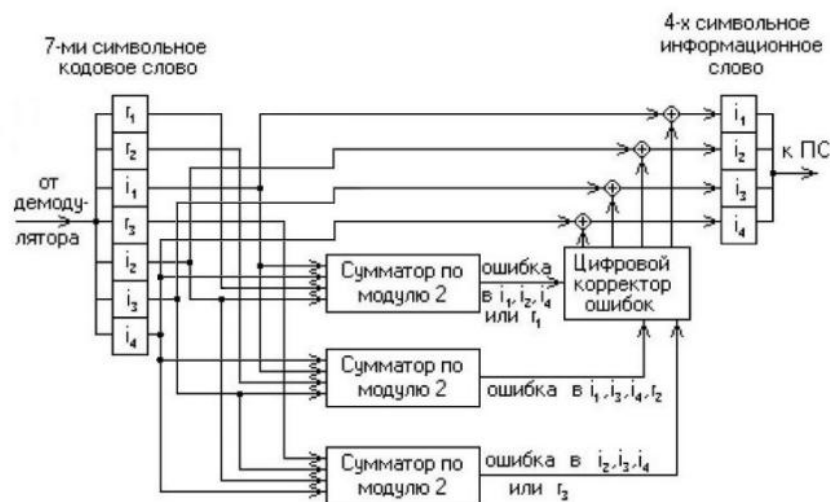
Вариант: 94

Оглавление

Задание №1.....	3
Задание №2.....	3
Задание №3.....	4
Задание №4.....	5
Задание №5.....	5
Дополнительное задание	6
Вывод	7
Список литературы	7

Задание №1

Построить схему декодирования классического кода Хэмминга (7;4)



Задание №2

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.

	r1	r2	i1	r3	i2	i3	i4
1. 105	0	1	1	1	1	1	1

$$S_1 = 0 \oplus 1 \oplus 1 \oplus 1 = 1$$

$$S_2 = 1 \oplus 1 \oplus 1 \oplus 1 = 0$$

$$S_3 = 1 \oplus 1 \oplus 1 \oplus 1 = 0$$

Ошибка в r_1

Правильно: 1111111

	r1	r2	i1	r3	i2	i3	i4
1. 26	1	1	0	0	0	0	1

$$S_1 = 1 \oplus 0 \oplus 0 \oplus 1 = 0$$

$$S_2 = 1 \oplus 0 \oplus 0 \oplus 1 = 0$$

$$S_3 = 0 \oplus 0 \oplus 0 \oplus 1 = 1$$

Ошибка в r_3

Правильно: 1101001

$S(1; 0; 0)$

$$\begin{aligned} s_1 &= r_1 \oplus i_1 \oplus i_2 \oplus i_4 \\ s_2 &= r_2 \oplus i_1 \oplus i_3 \oplus i_4 \\ s_3 &= r_3 \oplus i_2 \oplus i_3 \oplus i_4 \end{aligned}$$

Синдром ($s_1; s_2; s_3$)	Конфигурация ошибок	Ошибочный символ
000	НЕТ	НЕТ
001	0001000	r_3
010	0100000	r_2
011	0000010	i_3
100	1000000	r_1
101	0000100	i_2
110	0010000	i_1
111	0000001	i_4

$S(0; 0; 1)$

	r1	r2	i1	r3	i2	i3	i4	
1	64	1	0	0	0	1	0	0

$S_1 = 1 \oplus 0 \oplus 1 \oplus 0 = 0$
 $S_2 = 0 \oplus 0 \oplus 0 \oplus 0 = 0$
 $S_3 = 0 \oplus 1 \oplus 0 \oplus 0 = 1$

Ошибка в r_3
 Исправлено: 1001100

$s_1 = r_1 \oplus i_1 \oplus i_2 \oplus i_4$
 $s_2 = r_2 \oplus i_1 \oplus i_3 \oplus i_4$
 $s_3 = r_3 \oplus i_2 \oplus i_3 \oplus i_4$

Синдром ($s_1; s_2; s_3$)	Конфигурация ошибок	Ошибочный символ
000	НЕТ	НЕТ
001	0001000	r_3
010	0100000	r_2
011	0000010	i_3
100	1000000	r_1
101	0000100	i_2
110	0010000	i_1
111	0000001	i_4

	r1	r2	i1	r3	i2	i3	i4	
1.	10	1	0	1	0	0	0	0

$S_1 = 1 \oplus 1 \oplus 0 \oplus 0 = 0$
 $S_2 = 0 \oplus 1 \oplus 0 \oplus 0 = 1$
 $S_3 = 0 \oplus 0 \oplus 0 \oplus 0 = 0$

Ошибка в r_2
 Исправлено: 1110000

Задание №3

Построить схему декодирования классического кода Хэмминга (15;11)



Задание №4

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.

ALT	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
2.	84	0	0	1	0	1	0	1	0	0	0	0	1	0	1

$$S_1 = 0 \oplus 1 \oplus 1 \oplus 1 \oplus 1 \oplus 0 \oplus 1 \oplus 1 = 0$$

$$S_2 = 1 \oplus 1 \oplus 0 \oplus 1 \oplus 0 \oplus 0 \oplus 0 \oplus 1 = 1$$

$$S_3 = 0 \oplus 1 \oplus 0 \oplus 1 \oplus 0 \oplus 1 \oplus 0 \oplus 1 = 0$$

$$S_4 = 0 \oplus 1 \oplus 0 \oplus 0 \oplus 0 \oplus 0 \oplus 1 \oplus 0 \oplus 1 = 1$$

Ошибка в i_6

Правильно: 001010101100101

$S(0; 1; 0; 1)$

$$s_1 = r_1 \oplus i_1 \oplus i_2 \oplus i_4 \oplus i_5 \oplus i_7 \oplus i_9 \oplus i_{11};$$

$$s_2 = r_2 \oplus i_1 \oplus i_3 \oplus i_4 \oplus i_6 \oplus i_7 \oplus i_{10} \oplus i_{11};$$

$$s_3 = r_3 \oplus i_2 \oplus i_3 \oplus i_4 \oplus i_8 \oplus i_9 \oplus i_{10} \oplus i_{11};$$

$$s_4 = r_4 \oplus i_5 \oplus i_6 \oplus i_7 \oplus i_8 \oplus i_9 \oplus i_{10} \oplus i_{11};$$

Синдром ($s_1; s_2; s_3; s_4$)	Конфигурация ошибок	Ошибочный символ
0000	НЕТ	НЕТ
0001	000 0000 1000 0000	r_4
0010	000 1000 0000 0000	r_3
0011	000 0000 0000 1000	i_8
0100	010 0000 0000 0000	r_2
0101	000 0000 0010 0000	i_6
0110	000 0010 0000 0000	i_3
0111	000 0000 0000 0010	i_{10}
1000	100 0000 0000 0000	r_1
1001	000 0000 0100 0000	i_5
1010	000 0100 0000 0000	i_2
1011	000 0000 0000 0100	i_9
1100	001 0000 0000 0000	i_1
1101	000 0000 0001 0000	i_7
1110	000 0001 0000 0000	i_4
1111	000 0000 0000 0001	i_{11}

Задание №5

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.

$$i = (105 + 26 + 64 + 10 + 84) \cdot 4 = 1156$$

Определение минимального числа контрольных разрядов: $2^r \geq r + i + 1$.

$$2^r \geq r + 1157$$

$$2^{11} \geq 1168$$

$$2048 \geq 1168$$

$$r = 11$$

Коэффициент избыточности — отношение числа проверочных разрядов (r) к общему числу разрядов ($n = i + r$).

$$\frac{11}{1167} \approx 0,0094$$

Дополнительное задание

Написать программу на любом языке программирования, которая на вход из командной строки 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.

```
from typing import Union, List

def process_errors(l: List[int]) -> Union[None, List[str]]:
    """
    check a code for errors

    Args:
        l (List[int])

    Returns:
        Union[None, List[str]]
    """

    names = ["r1", "r2", "i1", "r3", "i2", "i3", "i4"]

    table = {
        "001": 3,
        "010": 1,
        "011": 5,
        "100": 0,
        "101": 4,
        "110": 2,
        "111": 6
    }

    syndromes = [l[0] ^ l[2] ^ l[4] ^ l[6],
                  l[1] ^ l[2] ^ l[5] ^ l[6],
                  l[3] ^ l[4] ^ l[5] ^ l[6]]

    if not any(syndromes):
        return None

    syndromes = "".join(list(map(str, syndromes)))

    place = names[table[syndromes]]
    bit = l[table[syndromes]]
    fixed = l.copy()
    fixed[table[syndromes]] = (bit + 1) % 2
    fixed = "".join(list(map(str, fixed)))

    return place, bit, fixed

if __name__ == "__main__":
    s = input("Enter your code:\n\n>> ")

    try:
        l = list(map(int, s.replace(" ", "")))

    except ValueError as e:
        print(f"{e}\nInput must contain only integers")

    else:
        if len(l) != 7:
            print("length != 7")

        else:
            errors = process_errors(l)

            if errors is None:
                print("Correct!")

            else:
                place, bit, fixed = errors
                print(f"Error detected!\n\nPlace: {place}\nBit: {bit}\nFixed input: {fixed}")
```

[Ссылка на код](#)

Вывод

В ходе выполнения второй лабораторной работы по информатике я узнал, что такое помехоустойчивое кодирование, самокорректирующиеся коды и код Хэмминга. А также из каких разрядов состоят помехоустойчивые коды (информационные и проверочные) и для чего нужен синдром последовательности. Помимо этого в результате выполнения дополнительного задания я научился реализовывать проделанную в основной части лабораторной работу в виде программного кода.

Список литературы

1. Балакшин П.В., Соснин В.В., Машина Е.А. Информатика