

Министерство науки и высшего образования Российской Федерации
Федеральное государственное автономное образовательное учреждение
высшего образования
«Национальный исследовательский университет ИТМО»
Факультет Программной инженерии и компьютерной техники

Лабораторная работа №2
“Синтез помехоустойчивого кода”

Вариант №36

Группа: Р3131
Выполнил: Друян Э.А.

Санкт-Петербург
2021 г.

Текст заданий

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 отчете в виде изображения.
4. Показать, исходя из выбранных вариантов сообщений (по 4 у каждого – часть No1 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 отчете в виде изображения.
7. Показать, исходя из выбранного варианта сообщений (по 1 у каждого – часть No2 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. Необязательное задания для получения оценки «5» (позволяет набрать от 86 до 100 процентов от максимального числа баллов БаРС за данную лабораторную). 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.

Выполнение заданий

1. Вариантом является комбинация 3-й и 5-й цифр номера в ISU = 283566, т.е. вариант = 36.
2. На основании номера варианта задания выбрать набор из 4 полученных сообщений в виде последовательности 7-символьного кода.

вар	часть 1				часть 2
36	32	59	86	1	37

Часть 1

#	1	2	3	4	5	6	7
	r1	r2	i1	r3	i2	i3	i4
32	0	0	1	1	0	1	0
59	0	0	1	0	1	0	0
86	0	0	0	1	1	1	0
1	0	0	0	1	0	0	0

3. Построить схему декодирования классического кода Хэмминга (7;4), которую представить в отчете 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.

#32	0	0	1	1	0	1	0	
	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

$$r1_{\text{рез}} = i1 \oplus i2 \oplus i4 = 1 \oplus 0 \oplus 0 = 1$$

$$r2_{\text{рез}} = i1 \oplus i3 \oplus i4 = 1 \oplus 1 \oplus 0 = 0$$

$$r3_{\text{рез}} = i2 \oplus i3 \oplus i4 = 0 \oplus 1 \oplus 0 = 1$$

$$s1 = r1_{\text{рез}} \oplus r1 = 1 \oplus 0 = 1$$

$$s2 = r2_{\text{рез}} \oplus r2 = 0 \oplus 0 = 0$$

$$s3 = r3_{\text{рез}} \oplus r3 = 1 \oplus 1 = 0$$

S(1;0;0). Какой бит содержится только в s1? Ответ (из таблицы): **r1 - ошибочный бит**. Инвертирую его и запишу **правильное сообщение**: 1011010.

#59	0	0	1	0	1	0	0	
	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

$$r1_{\text{рез}} = i1 \oplus i2 \oplus i4 = 1 \oplus 1 \oplus 0 = 0$$

$$r2_{\text{рез}} = i1 \oplus i3 \oplus i4 = 1 \oplus 0 \oplus 0 = 1$$

$$r3_{\text{рез}} = i2 \oplus i3 \oplus i4 = 1 \oplus 0 \oplus 0 = 1$$

$$s1 = r1_{\text{рез}} \oplus r1 = 0 \oplus 0 = 0$$

$$s2 = r2_{\text{рез}} \oplus r2 = 1 \oplus 0 = 1$$

$$s3 = r3_{\text{рез}} \oplus r3 = 1 \oplus 0 = 1$$

S(0;1;1). Какой бит содержится только в s2 и s3? Ответ (из таблицы): **i3 - ошибочный бит**. Инвертирую его и запишу **правильное сообщение**: 0010110.

#86	0	0	0	1	1	1	0	
	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
---	--	--	--	---	---	---	---	----

$$r1_{\text{рез}} = i1 \oplus i2 \oplus i4 = 0 \oplus 1 \oplus 0 = 1$$

$$r2_{\text{рез}} = i1 \oplus i3 \oplus i4 = 0 \oplus 1 \oplus 0 = 1$$

$$r3_{\text{рез}} = i2 \oplus i3 \oplus i4 = 1 \oplus 1 \oplus 0 = 0$$

$$s1 = r1_{\text{рез}} \oplus r1 = 1 \oplus 0 = 1$$

$$s2 = r2_{\text{рез}} \oplus r2 = 1 \oplus 0 = 1$$

$$s3 = r3_{\text{рез}} \oplus r3 = 0 \oplus 1 = 1$$

S(1;1;1). Какой бит содержится только в s1, s2 и s3? Ответ (из таблицы): **i4 - ошибочный бит**. Инвертирую его и запишу **правильное сообщение: 0001111**.

#1	0	0	0	1	0	0	0	
	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

$$r1_{\text{рез}} = i1 \oplus i2 \oplus i4 = 0 \oplus 0 \oplus 0 = 0$$

$$r2_{\text{рез}} = i1 \oplus i3 \oplus i4 = 0 \oplus 0 \oplus 0 = 0$$

$$r3_{\text{рез}} = i2 \oplus i3 \oplus i4 = 0 \oplus 0 \oplus 0 = 0$$

$$s1 = r1_{\text{рез}} \oplus r1 = 0 \oplus 0 = 0$$

$$s2 = r2_{\text{рез}} \oplus r2 = 0 \oplus 0 = 0$$

$$s3 = r3_{\text{рез}} \oplus r3 = 0 \oplus 1 = 1$$

S(0;0;1). Какой бит содержится только в s3? Ответ (из таблицы): **r3 - ошибочный бит**. Инвертирую его и запишу **правильное сообщение: 0000000**.

5. На основании номера варианта задания выбрать полученное сообщение в виде последовательности 11-символьного кода.
6. Построить схему декодирования классического кода Хэмминга (15;11), которую представить в отчете 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.

Часть 2

#37	0	1	0	1	0	1	0	0	1	0	1	0	0	1	0	
	r1	r2	i1	r3	i2	i3	i4	r4	i5	i6	i7	i8	i9	i10	i11	
1	X		X		X		X		X		X		X		X	s1
2		X	X			X	X			X	X			X	X	s2
4				X	X	X	X					X	X	X	X	s3
8								X	X	X	X	X	X	X	X	s4

$$r1_{\text{рез}} = i1 \oplus i2 \oplus i4 \oplus i5 \oplus i7 \oplus i9 \oplus i11 = [0001100] = 0$$

$$r2_{\text{рез}} = i1 \oplus i3 \oplus i4 \oplus i6 \oplus i7 \oplus i10 \oplus i11 = [0100110] = 1$$

$$r3_{\text{рез}} = i2 \oplus i3 \oplus i4 \oplus i8 \oplus i9 \oplus i10 \oplus i11 = [0100010] = 0$$

$$r4_{\text{рез}} = i5 \oplus i6 \oplus i7 \oplus i8 \oplus i9 \oplus i10 \oplus i11 = [1010010] = 1$$

$$s1 = r1_{\text{рез}} \oplus r1 = 0 \oplus 0 = 0$$

$$s2 = r2_{\text{рез}} \oplus r2 = 1 \oplus 1 = 0$$

$$s3 = r3_{\text{рез}} \oplus r3 = 0 \oplus 1 = 1$$

$$s4 = r4_{\text{рез}} \oplus r4 = 1 \oplus 0 = 1$$

S(0;0;1;1). Какой бит содержится только в s3 и s4? Ответ (из таблицы): **i8 - ошибочный бит**. Инвертирую его и запишу **правильное сообщение: 010101001011010**.

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.

$$m = (32 + 59 + 86 + 1 + 37) * 4 = 860,$$

m - число информационных разрядов в передаваемом сообщении.

Минимальное число проверочных разрядов находится из

неравенства:

$$2^k \geq k + m + 1,$$

где k - количество проверочных разрядов.

$$\{k=9; 2^9 = 512; k+m+1 = 870\};$$

$$\{k=10; 2^{10} = 1024; k+m+1 = 871\},$$
 - отсюда следует, что искомое

минимальное число проверочных разрядов $k_{\min} = 10$. Также легко

найти **коэффициент избыточности:**

$$C = k / (k + m) = 10 / 870 = 0.01149.$$

9. *Необязательное задания для получения оценки «5» (позволяет набрать от 86 до 100 процентов от максимального числа баллов БаРС за данную лабораторную). 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.*

Написал программу на языке C++. Ниже приведен код.

```
#include <iostream>
#include <string>
using namespace std;
// L - length (M + K),
// M - information bits, K - check bits
const int L = 7, M = 4, K = 3;

void make_bool(bool *a);
void show_right(bool *s, bool *a);

int main()
{
    /*
        input - 7 character string (4 information, 3 check):
        1100100
        output - 7 character string without mistakes:
        1100110

        #: 0  1  2  3  4  5  6
        name: r1 r2 i1 r3 i2 i3 i4
        exm: 1  1  0  0  1  0  0
    */
    bool a[L] {};
    make_bool(a);

    bool r1r = a[2] ^ a[4] ^ a[6]; // i1 i2 i4
```

```

    bool r2r = a[2] ^ a[5] ^ a[6]; // i1 i3 i4
    bool r3r = a[4] ^ a[5] ^ a[6]; // i1 i3 i4

    bool s[K] {};
    s[0] = r1r ^ a[0]; // r1r ^ r1
    s[1] = r2r ^ a[1]; // r2r ^ r2
    s[2] = r3r ^ a[3]; // r3r ^ r3

    show_right(s, a);

    return 0;
}

void make_bool(bool *a)
{
    // input
    string line;
    cin >> line;
    while (cin.get() != '\n')
    {}
    if (line.length() != L)
    { cout << "Wrong input. Stop the programm!\n"; return; }
    // string to bool
    for (int i = 0; i < L; ++i)
    {
        if (line[i] == '0')
        { a[i] = 0; }
        else if (line[i] == '1')
        { a[i] = 1; }
        else
        { cout << "Wrong input. Stop the programm!\n"; return; }
    }
}

void show_right(bool *s, bool *a)
{
    int s_val = 0;
    s[0] ? s_val += 4 : s_val += 0;
    s[1] ? s_val += 2 : s_val += 0;
    s[2] ? s_val += 1 : s_val += 0;
    //cout << "s_val: " << bitset<3>(s_val) << endl;
    if (s_val == 0)
    {
        for (int i = 0; i < L; ++i)
        { cout << a[i]; }
        cout << endl;
        return;
    }
    // table of correspondence
    const int IND_TBL[] = {-1, 3, 1, 5, 0, 4, 2, 6};
    a[IND_TBL[s_val]] = a[IND_TBL[s_val]] ? 0 : 1;

    cout << a[2] << a[4] << a[5] << a[6] << endl;
}

```


Вывод

Реализовал помехоустойчивый код Хэмминга в 5 примерах и написал соответствующий алгоритм на языке программирования c++. Также закрепил полученные знания о сжатии данных и помехоустойчивом кодировании.