

НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ УНИВЕРСИТЕТ ИТМО

Факультет программной инженерии и компьютерной техники

Дисциплина

Лабораторная работа № 2

Выполнил студент

Григорьев Никита Александрович

Группа № Р3124

Преподаватель: Болдырева Елена Александровна

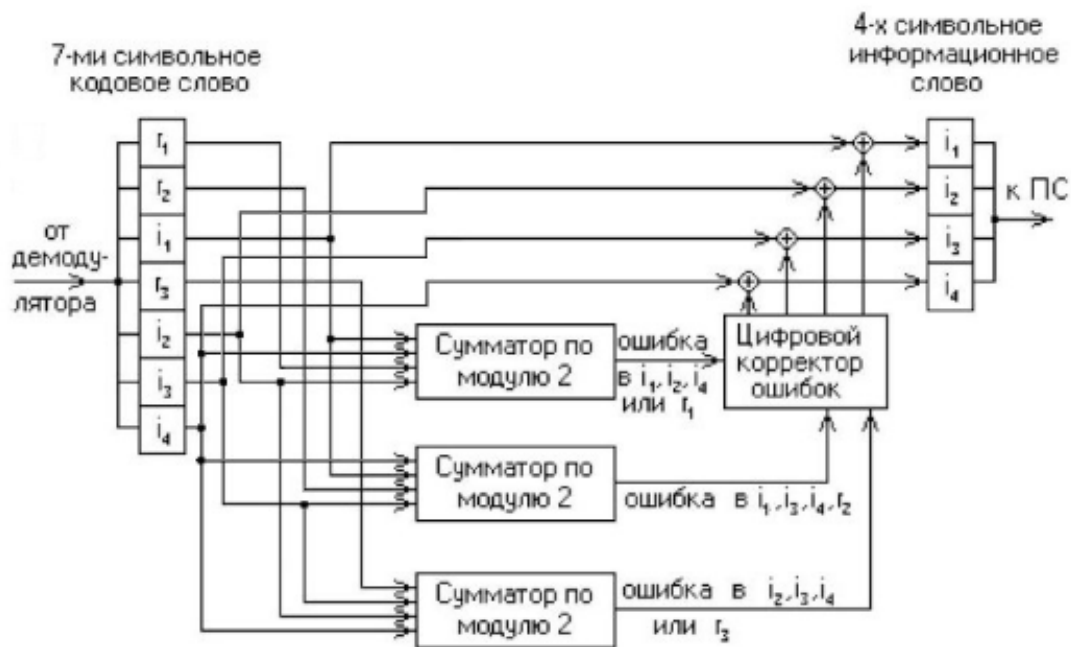
Санкт-Петербург

2023

Вариант: 89

Задание 1:

Построить схему декодирования классического кода Хэмминга (7;4), которую представить в отчёте в виде изображения.



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.

1

	r1	r2	i1	r3	i2	i3	i4
73	0	0	1	0	1	0	1

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

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

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

$S(1, 0, 0) \rightarrow$ ошибка в r_1

Верный вариант: 1010101

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

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

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

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

②

	r1	r2	i1	r3	i2	i3	i4
3	0	0	1	1	0	0	0

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

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

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

$S(1,1,1) \rightarrow$ ошибка в i_4

✓: 0011001

$$s_1 = r_1 \oplus i_1 \oplus i_2 \oplus i_4 :$$

$$s_2 = r_2 \oplus i_1 \oplus i_3 \oplus i_4 :$$

$$s_3 = r_3 \oplus i_2 \oplus i_3 \oplus i_4 :$$

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

③

	r1	r2	i1	r3	i2	i3	i4
45	0	0	1	0	0	1	1

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

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

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

$S(0,1,0) \rightarrow$ ошибка в r_2

✓: 0110011

$$s_1 = r_1 \oplus i_1 \oplus i_2 \oplus i_4 :$$

$$s_2 = r_2 \oplus i_1 \oplus i_3 \oplus i_4 :$$

$$s_3 = r_3 \oplus i_2 \oplus i_3 \oplus i_4 :$$

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

④

	r1	r2	i1	r3	i2	i3	i4
30	0	0	0	1	0	1	0

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

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

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

$S(0,1,0) \rightarrow$ ошибка в r_2

✓: 0101010

$$s_1 = r_1 \oplus i_1 \oplus i_2 \oplus i_4 :$$

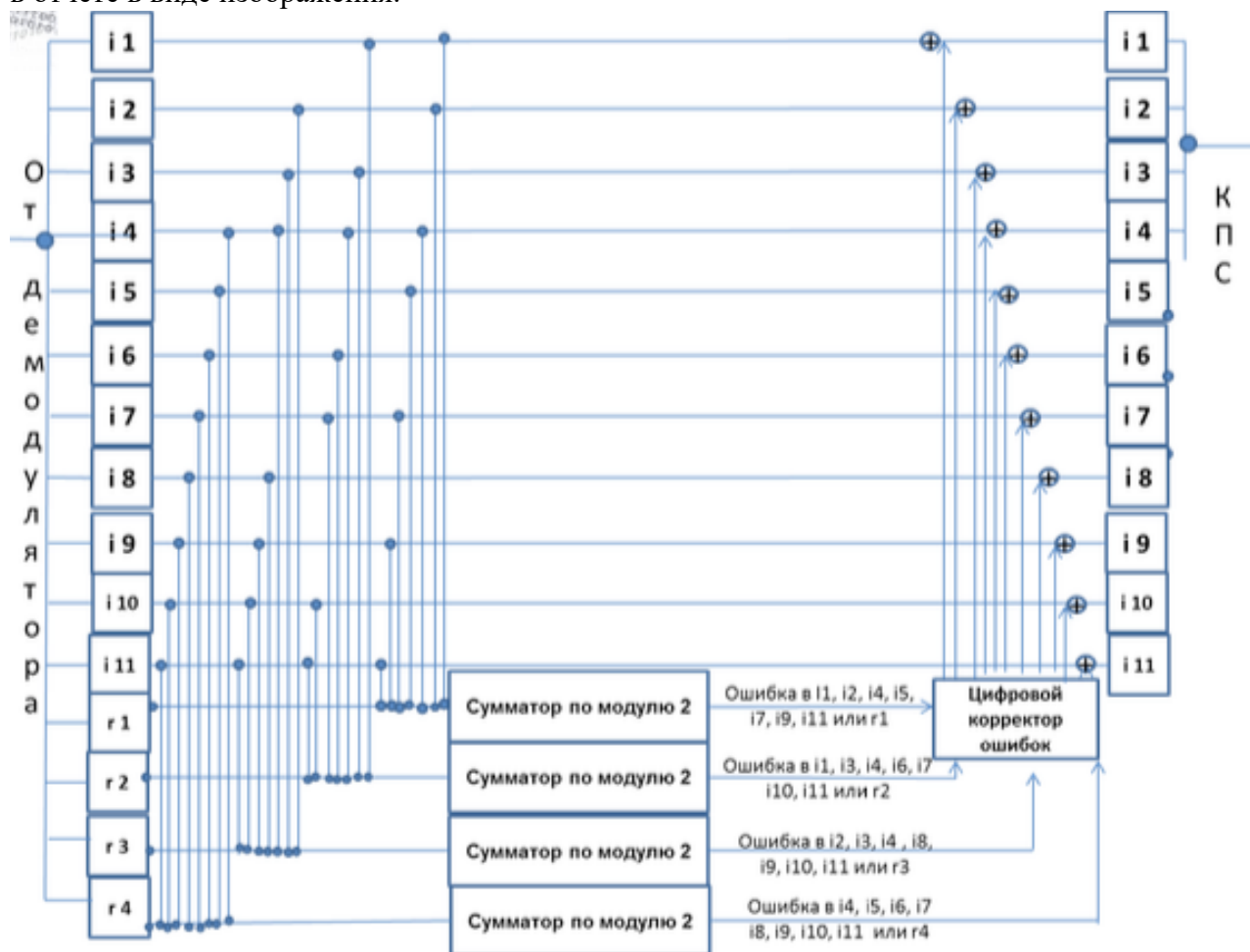
$$s_2 = r_2 \oplus i_1 \oplus i_3 \oplus i_4 :$$

$$s_3 = r_3 \oplus i_2 \oplus i_3 \oplus i_4 :$$

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

Задание 3:

Построить схему декодирования классического кода Хэмминга (15;11), которую представить в отчёте в виде изображения.



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

5

	r1	r2	i1	r3	i2	i3	i4	r4	i5	i6	i7	i8	i9	i10	i11
87	0	0	1	0	1	0	1	0	1	1	1	0	1	0	1

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

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

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

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

$$S(1,1,0,1) \rightarrow \text{ошибка в } i7$$

$$\checkmark: 001010101100101$$

$s_1 = r_1 \oplus i_1 \oplus i_2 \oplus i_4 \oplus i_5 \oplus i_7 \oplus i_9 \oplus i_{11};$
 $s_2 = r_2 \oplus i_1 \oplus i_3 \oplus i_4 \oplus i_6 \oplus i_7 \oplus i_{10} \oplus i_{11};$
 $s_3 = r_3 \oplus i_2 \oplus i_3 \oplus i_4 \oplus i_8 \oplus i_9 \oplus i_{10} \oplus i_{11};$
 $s_4 = r_4 \oplus i_5 \oplus i_6 \oplus i_7 \oplus i_8 \oplus i_9 \oplus i_{10} \oplus i_{11};$

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}

правильное сообщение.

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.

$$\textcircled{6} \quad 73 + 3 + 45 + 30 + 87 = 238 \cdot 4 = 952$$

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

$$2^r \geq r + i + 1$$

$$2^{10} \geq 10 + 953$$

$$1024 \geq 963$$

$$r = 10$$

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

$$\frac{10}{962} \approx 0,01$$

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.

```
1  def output(array):
2      result = ""
3      for i in array:
4          result += str(i)
5      return result
6
7
8  msg = input('Код: ')
9  bits = []
10
11  for i in msg:
12      bits.append(int(i))
13
14  s1 = bits[0] ^ bits[2] ^ bits[4] ^ bits[6]
15  s2 = bits[1] ^ bits[2] ^ bits[5] ^ bits[6]
16  s3 = bits[3] ^ bits[4] ^ bits[5] ^ bits[6]
17
18  print('Синдром: ', s1, s2, s3)
19
20  if (s1, s2, s3) == (0, 0, 0):
21      print('Ошибка нет', output(bits))
22  elif (s1, s2, s3) == (0, 0, 1):
23      bits[3] = (bits[3] + 1) % 2
24      print('Ошибка в r3.\nОшибочный бит - 4', '\nИсправленная последовательность: ', output(bits))
25  elif (s1, s2, s3) == (0, 1, 0):
26      bits[1] = (bits[1] + 1) % 2
27      print('Ошибка в r2.\nОшибочный бит - 2', '\nИсправленная последовательность: ', output(bits))
28  elif (s1, s2, s3) == (0, 1, 1):
29      bits[5] = (bits[5] + 1) % 2
30      print('Ошибка в i3.\nОшибочный бит - 6', '\nИсправленная последовательность: ', output(bits))
31  elif (s1, s2, s3) == (1, 0, 0):
32      bits[0] = (bits[0] + 1) % 2
33      print('Ошибка в r1.\nОшибочный бит - 1', '\nИсправленная последовательность: ', output(bits))
34  elif (s1, s2, s3) == (1, 0, 1):
35      bits[4] = (bits[4] + 1) % 2
36      print('Ошибка в i2.\nОшибочный бит - 5', '\nИсправленная последовательность: ', output(bits))
37  elif (s1, s2, s3) == (1, 1, 0):
38      bits[2] = (bits[2] + 1) % 2
39      print('Ошибка в i1.\nОшибочный бит - 3', '\nИсправленная последовательность: ', output(bits))
40  elif (s1, s2, s3) == (1, 1, 1):
41      bits[6] = (bits[6] + 1) % 2
42      print('Ошибка в i4.\nОшибочный бит - 7', '\nИсправленная последовательность: ', output(bits))
```

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) Методичка: Информатика П.В. Балакшин, В.В. Соснин, Е.А. Машина, 2020