МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ

РОССИЙСКОЙ ФЕДЕРАЦИИ

ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ АВТОНОМНОЕ ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ

«Национальная научно-образовательная корпорация ИТМО»

Факультет Программной инженерии и компьютерной техники

Лабораторная работа №2

По дисциплине

«Информатика»

Вариант №75

**Выполнил:**

Студент группы P3114

Щербинин Эдуард Павлович

**Преподаватель:**

Машина Екатерина Алексеевна

**Содержание**

[Задание 3](#_Toc115197652)

[Основные этапы вычислений 4](#_Toc115197653)

[Вывод 8](#_Toc115197654)

[Список литературы 9](#_Toc115197655)

Задание

Построить схему декодирования классического кода Хэмминга (7;4) и (15;11), которые представить в отчёте в виде изображения.

Показать, исходя из выбранных вариантов сообщений (4 (7;4) и 1 (15;11)), имеются ли в принятом сообщении ошибки, и если имеются, то какие. Подробно прокомментироватьи записать правильное сообщение.

Занесем все наборы сообщений из варианта 75 в таблицу:

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| № | r1 | r2 | i1 | r3 | i2 | i3 | i4 | r4 | i5 | i6 | i7 | i8 | i9 | i10 | i11 |
| 57 | 0 | 0 | 0 | 0 | 1 | 0 | 0 |  |  |  |  |  |  |  |  |
| 94 | 1 | 0 | 1 | 0 | 1 | 1 | 0 |  |  |  |  |  |  |  |  |
| 19 | 0 | 1 | 0 | 1 | 0 | 0 | 1 |  |  |  |  |  |  |  |  |
| 10 | 1 | 0 | 1 | 0 | 0 | 0 | 0 |  |  |  |  |  |  |  |  |
| 74 | 0 | 0 | 1 | 1 | 1 | 0 | 0 | 1 | 0 | 1 | 0 | 0 | 1 | 0 | 0 |

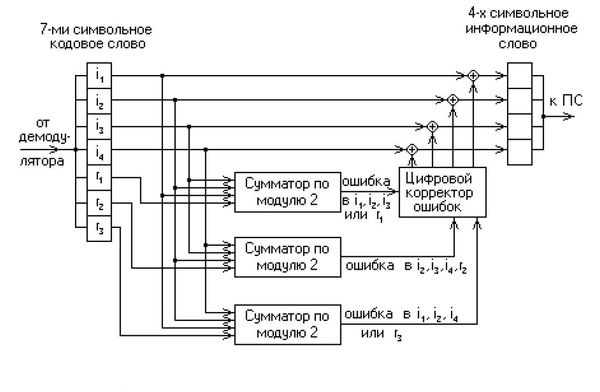
Сложить номера всех 5 вариантов заданий. Умножить полученное число на 4. Принять данное число как число информационных разрядов в передаваемом сообщении. Вычислить для данного числа минимальное число проверочных разрядов и коэффициент избыточности.

57+94+19+10+74 = 254\*4 = 1016

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.

Основные этапы вычислений

Схема декодирования классического кода Хэмминга (7;4):



Декодирование сообщений по коду Хэмминга 1-4:

1) 0000100

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Сообщение | 0 | 0 | 0 | 0 | 1 | 0 | 0 |  |
| 2x | r1 | r2 | i1 | r3 | i2 | i3 | i4 | S |
| 1 | x |  | x |  | x |  | x | s1 |
| 2 |  | x | x |  |  | x | x | s2 |
| 4 |  |  |  | x | x | x | x | s3 |

Расставляем корректирующие биты на позициях степени двойки по порядку (r1 на 20=1, r2 на 21=2 и т.д.), остальные места занимают информационные биты по порядку. Расставим “x” в таблице на тех битах, за которые отвечает корректирующий бит на позиции 2x. Просуммируем соответствующие иксам биты по модулю 2 и получим результирующий синдром S:

s1 = r1 ⊕ i1 ⊕ i2 ⊕ i4 = 0 ⊕ 0 ⊕ 1 ⊕ 0 = 1

s2 = r2 ⊕ i1 ⊕ i3 ⊕ i4 = 0 ⊕ 0 ⊕ 0 ⊕ 0 = 0

s3 = r3 ⊕ i2 ⊕ i3 ⊕ i4 = 0 ⊕ 1 ⊕ 0 ⊕ 0 = 1

S = 101

Также можно просто посчитать количество единиц и если оно чётное s будет равным 0, если нечётное – 1.

Столбик 1 0 1 соответствует x – x в таблице, по которой понимаем, что ошибочный бит i2.

Также можно высчитать свои корректирующие биты и сравнить их с полученными с одинаковыми индексами (r1 и R1 и тд) и если они не совпадают сложить их позиции:

r1 != R1 r2 = R2 r3 != R3

1 + … + 4 = 5 поз. (соотв. i2)

Следовательно, правильное сообщение из инф-х битов:

0000

2) 1010110

Делаем всё аналогично 1 номеру:

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Сообщение | 1 | 0 | 1 | 0 | 1 | 1 | 0 |  |
| 2x | r1 | r2 | i1 | r3 | i2 | i3 | i4 | S |
| 1 | x |  | x |  | x |  | x | s1 |
| 2 |  | x | x |  |  | x | x | s2 |
| 4 |  |  |  | x | x | x | x | s3 |

s1 = r1 ⊕ i1 ⊕ i2 ⊕ i4 = 1 ⊕ 1 ⊕ 1 ⊕ 0 = 1

s2 = r2 ⊕ i1 ⊕ i3 ⊕ i4 = 0 ⊕ 1 ⊕ 1 ⊕ 0 = 0

s3 = r3 ⊕ i2 ⊕ i3 ⊕ i4 = 0 ⊕ 1 ⊕ 1 ⊕ 0 = 0

S = 100 => ошибка в корректирующем бите r1, а значит инф-е биты без ошибок и полученное сообщение –

1110

3)0101001

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Сообщение | 0 | 1 | 0 | 1 | 0 | 0 | 1 |  |
| 2x | r1 | r2 | i1 | r3 | i2 | i3 | i4 | S |
| 1 | x |  | x |  | x |  | x | s1 |
| 2 |  | x | x |  |  | x | x | s2 |
| 4 |  |  |  | x | x | x | x | s3 |

s1 = r1 ⊕ i1 ⊕ i2 ⊕ i4 = 0 ⊕ 0 ⊕ 0 ⊕ 1 = 1

s2 = r2 ⊕ i1 ⊕ i3 ⊕ i4 = 1 ⊕ 0 ⊕ 0 ⊕ 1 = 0

s3 = r3 ⊕ i2 ⊕ i3 ⊕ i4 = 1 ⊕ 0 ⊕ 0 ⊕ 1 = 0

S = 100 => ошибка в корректирующем бите r1, а значит инф-е биты без ошибок и полученное сообщение –

0001

4)1010000

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Сообщение | 1 | 0 | 1 | 0 | 0 | 0 | 0 |  |
| 2x | r1 | r2 | i1 | r3 | i2 | i3 | i4 | S |
| 1 | x |  | x |  | x |  | x | s1 |
| 2 |  | x | x |  |  | x | x | s2 |
| 4 |  |  |  | x | x | x | x | s3 |

s1 = r1 ⊕ i1 ⊕ i2 ⊕ i4 = 1 ⊕ 1 ⊕ 0 ⊕ 0 = 0

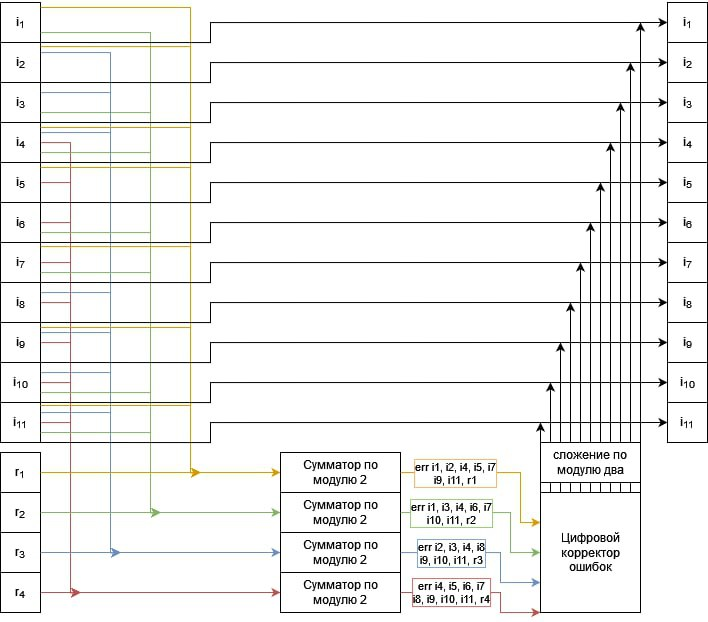
s2 = r2 ⊕ i1 ⊕ i3 ⊕ i4 = 0 ⊕ 1 ⊕ 0 ⊕ 0 = 1

s3 = r3 ⊕ i2 ⊕ i3 ⊕ i4 = 0 ⊕ 0 ⊕ 0 ⊕ 0 = 0

S = 100 => ошибка в корректирующем бите r2, а значит инф-е биты без ошибок и полученное сообщение –

1000

Схема декодирования классического кода Хэмминга (15;11):



Декодирование сообщений по коду Хэмминга 5:

5) 001110010100100

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Сообщение | 0 | 0 | 1 | 1 | 1 | 0 | 0 | 1 | 0 | 1 | 0 | 0 | 1 | 0 | 0 |  |
| 2x | r1 | r2 | i1 | r3 | i2 | i3 | i4 | r4 | i5 | i6 | i7 | i8 | i9 | i10 | i11 | S |
| 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 |

s1 = 0 ⊕ 1 ⊕ 1 ⊕ 0 ⊕ 0 ⊕ 0 ⊕ 1 ⊕ 0 = 1

s2 = 0 ⊕ 1 ⊕ 0 ⊕ 0 ⊕ 1 ⊕ 0 ⊕ 0 ⊕ 0 = 0

s3 = 1 ⊕ 1 ⊕ 0 ⊕ 0 ⊕ 0 ⊕ 1 ⊕ 0 ⊕ 0 = 1

s4 = 1 ⊕ 0 ⊕ 1 ⊕ 0 ⊕ 0 ⊕ 1 ⊕ 0 ⊕ 0 = 1

S = 1011 => Ошибка в бите i9, заменяем его и составляем сообщение –

11000100000

Вычислим для 1016 инф-х битов минимальное число проверочных разрядов и коэффициент избыточности.

i <= 2r – r – 1

r=10: 210-10-1 = 1013 < 1016 – не хватает битов

r=11: 211-11-1 = 2036 >= 1016 – достаточно

Минимальное число проверочных разрядов – 11

Коэф. изб-ти = r / (r+i) = 11 / (11+1016) = 11 / 1027 = 0,0107108

Программа для нахождения ошибки в коде Хэмминга и вывод правильного сообщения:

string = input()  
bitlen = len(string)  
#Если длина строки + 1 является степенью двойки и состоит только из 1 и 0 она подходит  
if ((bitlen+1) & bitlen == 0) and all(x in ['0', '1'] for x in string):  
 bit = [int(x) for x in string] #в массив битов  
 r = []  
 for i in range(1,bitlen+1):  
 if(i & (i-1) == 0): #Является ли i степенью двойки  
 r.append(bit[i-1]) #если да то это корректирующий бит  
  
 s = [0]\*len(r) #длина синдрома = длине r  
 for i in range(len(r)):  
 k = 2\*\*i-1 #начинаем с позиции r (степени двойки)  
 while k < bitlen: #идем до конца строки  
 for j in range(k, k + 2\*\*i): #начиная с k проходим количество бит равное степени двойки позиции r (как в таблице)  
 s[i] = s[i] ^ bit[j] #и суммируем их по модулю 2  
 k += 2\*\*(i+1) #перескакиваем на начало следующей сумм  
 print('Синдром:', s)  
  
 mistakePos = 0  
 for i in range(len(s)): #переводим синдром из двоичной системы в десятичную  
 mistakePos += s[i] \* 2\*\*i  
  
 if(mistakePos == 0):  
 print('Ошибок в бите нет')  
 else:  
 print('Ошибка в бите номер', mistakePos)  
 bit[mistakePos-1] = bit[mistakePos-1] ^ 1 #Инвертируем бит  
  
 correctStr = ''  
 correctMsgStr = ''  
 for i in range(bitlen): #собираем строчки  
 correctStr += str(bit[i])  
 if ((i+1) & i != 0):  
 correctMsgStr += str(bit[i])  
 print('Правильная строка:', correctStr)  
 print('Сообщение:', correctMsgStr)  
  
else:  
 print('Syntax error')

Вывод

В результате выполнения работы было получено понимание работы кода Хэмминга и корректирующих кодов в целом, а также умение декодировать и исправлять ошибки в полученном сообщении, закодированному по коду Хэмминга, определять необходимое количество корректирующих битов и коэффициент избыточности кода.

Список литературы

1. Хабр. Код Хэмминга. Пример работы алгоритма. [Электронный ресурс] Режим доступа: <https://habr.com/ru/post/140611/> (дата обращения: 15.10.2022)
2. Википедия. Код Хэмминга [Электронный ресурс] Режим доступа: <https://ru.wikipedia.org/wiki/%D0%9A%D0%BE%D0%B4_%D0%A5%D1%8D%D0%BC%D0%BC%D0%B8%D0%BD%D0%B3%D0%B0> (дата обращения: 15.10.2022)