

МИНИСТЕРСТВО ОБРАЗОВАНИЯ И НАУКИ РОССИЙСКОЙ ФЕДЕРАЦИИ  
Федеральное государственное автономное образовательное учреждение  
высшего профессионального образования  
«САНКТ-ПЕТЕРБУРГСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ  
АЭРОКОСМИЧЕСКОГО ПРИБОРОСТРОЕНИЯ»  
КАФЕДРА №52

ОТЧЕТ  
ЗАЩИЩЕН С ОЦЕНКОЙ  
ПРЕПОДАВАТЕЛЬ

ассистент

\_\_\_\_\_  
должность, уч. степень, звание

\_\_\_\_\_  
подпись, дата

А.А. Бурков

\_\_\_\_\_  
инициалы, фамилия

ОТЧЕТ ПО ЛАБОРАТОРНОЙ РАБОТЕ № 1

Использование циклических кодов для обнаружения ошибок в сетях передачи  
данных

по курсу: ОСНОВЫ ПОСТРОЕНИЯ ИНФОКОМУНИКАЦИОННЫХ  
СИСТЕМ И СЕТЕЙ

РАБОТУ ВЫПОЛНИЛ

СТУДЕНТ ГР.

5723

\_\_\_\_\_  
подпись, дата

А. Ю. Глушенкова

\_\_\_\_\_  
инициалы, фамилия

**Цель работы:** исследование типового алгоритма формирования контрольной суммы с использованием циклических кодов, использование численного расчета и имитационного моделирования для оценки вероятности того, что декодер не обнаружит ошибки.

**Модель системы:** в большинстве современных систем передачи данных для обнаружения ошибок применяется следующий подход. К передаваемым данным добавляют контрольную сумму, которая вычисляется на основе этих же данных. По каналу передается сообщение, состоящее из данных и контрольной суммы. Использование контрольной суммы позволяет определить, по принятому сообщению, возникли ли ошибки при передаче данного сообщения по каналу.

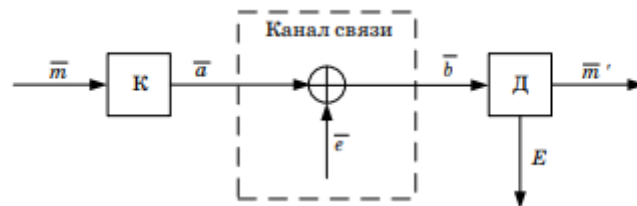


Рис. 1. Структурная схема системы передачи данных:  
 $\bar{m}$  – информационное сообщение, К – блок кодера,  
 $\bar{a}$  – закодированное сообщение,  $\bar{e}$  – вектор ошибок,  
 $\bar{b}$  – сообщение на выходе канала, Д – блок декодера,  
 $E$  – принятое решение,  $\bar{m}'$  – сообщение на выходе декодера

Декодер по некоторому алгоритму проверяет контрольную сумму в принятом сообщении и принимает одно из следующих решений

$$E = \begin{cases} 1, & \text{если были ошибки} \\ 0, & \text{если не было ошибок} \end{cases}.$$

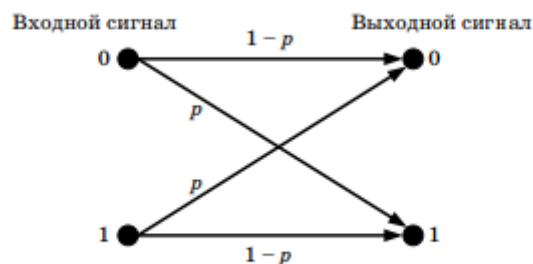


Рис. 2. Модель двоично-симметричного канала

Канал является двоичным, поэтому возможны только два значения битов на входе и выходе канала:  $\{0,1\}$ . Канал называется симметричным ввиду того, что вероятность ошибки для обоих значений битов одинакова.

Поскольку используются только двоичные коды, то можно сделать вывод, что используются многочлены с коэффициентами GF(2).

### Описание работы кодера:

Кодер хранит порождающий многочлен  $g(x)$ ,  $\deg(g(x)) = r$  и определяет количество бит контрольной суммы в кодовом слове.  $k$  – длина информационного сообщения  $m$ .

1. На основе вектора  $m$  формируется многочлен  $m(x)$ , степень которой меньше или равна  $k - 1$
2. Вычисляется многочлен  $c(x) = m(x) * x^r \bmod g(x)$ . Степень многочлена  $c(x)$  меньше или равна  $r - 1$
3. Вычисляется многочлен  $a(x) = m(x) * x^r + c(x)$
4. На основе многочлена  $a(x)$  формируется вектор  $a$ , длина которого равна  $n = k + r$

Допустим, что в канале связи произошли ошибки согласно вектору ошибок  $e$ . Тогда закодированное сообщение будет  $b = a + e$ .

### Описание работы декодера:

Декодер хранит порождающий многочлен  $g(x)$ ,  $\deg(g(x)) = r$  и  $n$  – длина кодового слова.

1. Принятый вектор  $b$  переводится в многочлен  $b(x)$
2. Вычисляется синдром  $s(x) = b(x) \bmod g(x)$
3. Если  $s(x) = 0$ , то декодер выносит решение, что ошибок не произошло  $E = 0$ , иначе ошибки произошли и  $E = 1$ .

На вход программы подаются вектора  $g$ ,  $m$ , длина  $m$  и вектор ошибок  $e$ , который можно задать в двух форматах – вектором, но при этом нужно рассчитать длину закодированного сообщения или вероятностью  $p$ , с которой будет формироваться 0 в векторе ошибок,  $(1 - p)$  – вероятность формирования 1.

Также предусмотрены исключения, если формат введенных данных некорректен.

Примеры:

В каждом примере представлен порождающий вектор  $g(x)$ , вектор информационного сообщения  $m(x)$ , промежуточные шаги формирования вектора  $a(x)$ .

Если формирование вектора ошибки задано через вероятность, то выводится случайная вероятность для каждого бита, по которой определяется бит вектора 0 или 1.

Выводится сформированный или входной вектор ошибок  $e(x)$ , вектор после прохождения через канал связи  $b(x)$ , т.е. с добавлением вектора ошибок. Также выводится синдром  $s(x)$ , по которому выводится решение декодера  $E$ .

В примере 1 решение декодера  $E = 1$ . Ошибки были обнаружены.

```
C:\Users\alng\Desktop\Cети\lab1>main.exe 1011 1000 4 0.5
g(x) = 1011    deg = 3    length = 4
m(x) = 1000    deg = 3    length = 4
m(x) * x^3 = 1000000    deg = 6    length = 7
c(x) = 101    deg = 2    length = 3
a(x) = 1000101    deg = 6    length = 7
current p(0) = 0.13
current p(1) = 0.61
current p(2) = 0.77
current p(3) = 0.95
current p(4) = 0.27
current p(5) = 0.39
current p(6) = 0.67
e(x) = 0111001    deg = 3    length = 7
b(x) = 1111100    deg = 6    length = 7
s(x) = 11    deg = 1    length = 2
E = 1
```

В примере 2 была допущена ошибка в входных данных, декодер не начал свою работу.

```
C:\Users\alng\Desktop\Сети\lab1>main.exe 1011 1010 4 00000000
g(x) = 1011    deg = 3    length = 4
m(x) = 1010    deg = 3    length = 4
m(x) * x^3 = 1010000    deg = 6    length = 7
c(x) = 11      deg = 1    length = 2
a(x) = 1010011    deg = 6    length = 7
e(x) = 00000000    deg = 0    length = 8
e has incorrect size
```

В примере 3 на вход был подан нулевой вектор ошибок, декодер выдает решение  $E = 0$ , что является верным результатом, т.к.  $a(x) = b(x)$  и по аксиму  $a(x) \bmod g(x) = 0$ , т.е.  $b(x) \bmod g(x)$  также равно 0.

```
C:\Users\alng\Desktop\Сети\lab1>main.exe 1011 1010 4 00000000
g(x) = 1011    deg = 3    length = 4
m(x) = 1010    deg = 3    length = 4
m(x) * x^3 = 1010000    deg = 6    length = 7
c(x) = 11      deg = 1    length = 2
a(x) = 1010011    deg = 6    length = 7
e(x) = 00000000    deg = 0    length = 7
b(x) = 1010011    deg = 6    length = 7
s(x) = 0
E = 0
```

В примере 4 решение декодера  $E = 1$ . В канале были допущены ошибки.

```
C:\Users\alng\Desktop\Сети\lab1>main.exe 10111 111110 6 0111000010
g(x) = 10111    deg = 4    length = 5
m(x) = 111110    deg = 5    length = 6
m(x) * x^4 = 1111100000    deg = 9    length = 10
c(x) = 10      deg = 1    length = 2
a(x) = 1111100010    deg = 9    length = 10
e(x) = 0111000010    deg = 8    length = 10
b(x) = 1000100000    deg = 9    length = 10
s(x) = 1010    deg = 3    length = 4
E = 1
```

В примере 5 формат входных данных некорректен.

```
C:\Users\alng\Desktop\Сети\lab1>main.exe 10111 1014353453453 73 0000002211213213
Error *create_polynom*: bits[i] != 0 != 1
```

**Описание дополнительного задания:** пусть  $f(x) = x^3 + x + 1$  или  $f(x) = x^3 + x^2 + 1$ , а порождающий многочлен  $g(x) = f(x) * (x + 1)$ . Привести примеры, когда не обнаруживается нечетное число ошибок. Если таких примеров нет, то обосновать почему.

В примерах в начале выводятся сформированные по входным данным вектора. Вектора  $s(x)$  для проверки формирования закодированного информационного сообщения  $a(x)$ . Сформированные вектора закодированного сообщения  $a(x)$ , вектора ошибок  $e(x)$ , при которых декодер не обнаруживает допущенные в канале связи ошибки, а также количество таких векторов ошибок. Синдромы  $s(x)$ , если он равен 0, то ошибки не обнаружены.

Пример 1.

```

m(x) = 1010    deg = 3    length = 4
f(x) = 1011    deg = 3    length = 4
g(x) = 11101   deg = 4    length = 5

for f(x): c(x) = 11    deg = 1    length = 2
for g(x): c(x) = 110   deg = 2    length = 3

for f(x): a(x) = 1010011    deg = 6    length = 7
for g(x): a(x) = 10100110   deg = 7    length = 8

```

Количество необнаруженных ошибок для  $f(x)$  в данном случае равно 8.

```

FOR f(x)
ERROR IS NULL 8
e(x) = 0001011    deg = 3    length = 7
s(x) = 0

e(x) = 0010110    deg = 4    length = 7
s(x) = 0

e(x) = 0101100    deg = 5    length = 7
s(x) = 0

e(x) = 0110001    deg = 5    length = 7
s(x) = 0

e(x) = 1000101    deg = 6    length = 7
s(x) = 0

e(x) = 1011000    deg = 6    length = 7
s(x) = 0

e(x) = 1100010    deg = 6    length = 7
s(x) = 0

e(x) = 1111111    deg = 6    length = 7
s(x) = 0

```

В то время как для  $g(x)$  все ошибки обнаружены

```

FOR g(x)
ERROR IS NULL 0
g(x) = 11101    deg = 4    length = 5
a(x) = 10100110    deg = 7    length = 8

```

Пример 2.

```

m(x) = 10101    deg = 4    length = 5
f(x) = 1011     deg = 3    length = 4
g(x) = 11101    deg = 4    length = 5

for f(x): c(x) = 101    deg = 2    length = 3
for g(x): c(x) = 1      deg = 0    length = 1

for f(x): a(x) = 10101101    deg = 7    length = 8
for g(x): a(x) = 101010001   deg = 8    length = 9

```

```
FOR f(x)
ERROR IS NULL 16
e(x) = 00001011    deg = 3    length = 8
s(x) = 0

e(x) = 00010110    deg = 4    length = 8
s(x) = 0

e(x) = 00101100    deg = 5    length = 8
s(x) = 0

e(x) = 00110001    deg = 5    length = 8
s(x) = 0

e(x) = 01000101    deg = 6    length = 8
s(x) = 0

e(x) = 01011000    deg = 6    length = 8
s(x) = 0

e(x) = 01100010    deg = 6    length = 8
s(x) = 0

e(x) = 01111111    deg = 6    length = 8
s(x) = 0

e(x) = 10001010    deg = 7    length = 8
s(x) = 0

e(x) = 10010111    deg = 7    length = 8
s(x) = 0

e(x) = 10101101    deg = 7    length = 8
s(x) = 0

e(x) = 10110000    deg = 7    length = 8
s(x) = 0

e(x) = 11000100    deg = 7    length = 8
s(x) = 0

e(x) = 11011001    deg = 7    length = 8
s(x) = 0

e(x) = 11100011    deg = 7    length = 8
s(x) = 0

e(x) = 11111110    deg = 7    length = 8
s(x) = 0

FOR g(x)
ERROR IS NULL 0
g(x) = 11101    deg = 4    length = 5
a(x) = 101010001    deg = 8    length = 9
```

Пример 3.

```
m(x) = 10    deg = 1  length = 2
f(x) = 1011   deg = 3  length = 4
g(x) = 11101  deg = 4  length = 5

for f(x): c(x) = 110    deg = 2  length = 3
for g(x): c(x) = 111    deg = 2  length = 3

for f(x): a(x) = 10110  deg = 4  length = 5
for g(x): a(x) = 100111 deg = 5  length = 6

FOR f(x)
ERROR IS NULL 2
e(x) = 01011   deg = 3  length = 5
s(x) = 0

e(x) = 10110   deg = 4  length = 5
s(x) = 0

FOR g(x)
ERROR IS NULL 0
g(x) = 11101   deg = 4  length = 5
a(x) = 100111  deg = 5  length = 6
```

Пример 4.

```
m(x) = 101    deg = 2  length = 3
f(x) = 1011   deg = 3  length = 4
g(x) = 11101  deg = 4  length = 5

for f(x): c(x) = 100    deg = 2  length = 3
for g(x): c(x) = 11     deg = 1  length = 2

for f(x): a(x) = 101100  deg = 5  length = 6
for g(x): a(x) = 1010011 deg = 6  length = 7

FOR f(x)
ERROR IS NULL 4
e(x) = 001011  deg = 3  length = 6
s(x) = 0

e(x) = 010110  deg = 4  length = 6
s(x) = 0

e(x) = 101100  deg = 5  length = 6
s(x) = 0

e(x) = 110001  deg = 5  length = 6
s(x) = 0

FOR g(x)
ERROR IS NULL 0
g(x) = 11101   deg = 4  length = 5
a(x) = 1010011 deg = 6  length = 7
```

### Пример 5.

```
m(x) = 011 deg = 1 length = 3
f(x) = 1011 deg = 3 length = 4
g(x) = 11101 deg = 4 length = 5

for f(x): c(x) = 101 deg = 2 length = 3
for g(x): c(x) = 1010 deg = 3 length = 4

for f(x): a(x) = 011101 deg = 4 length = 6
for g(x): a(x) = 0111010 deg = 5 length = 7

FOR f(x)
ERROR IS NULL 4
e(x) = 001011 deg = 3 length = 6
s(x) = 0

e(x) = 010110 deg = 4 length = 6
s(x) = 0

e(x) = 101100 deg = 5 length = 6
s(x) = 0

e(x) = 110001 deg = 5 length = 6
s(x) = 0

FOR g(x)
ERROR IS NULL 0
g(x) = 11101 deg = 4 length = 5
a(x) = 0111010 deg = 5 length = 7
```

### Более подробно:

```
m(x) = 011 deg = 1 length = 3
f(x) = 1011 deg = 3 length = 4
g(x) = 11101 deg = 4 length = 5

for f(x): c(x) = 101 deg = 2 length = 3
for g(x): c(x) = 1010 deg = 3 length = 4

for f(x): a(x) = 011101 deg = 4 length = 6
for g(x): a(x) = 0111010 deg = 5 length = 7

FOR f(x)
ERROR IS NULL 4
a(x) = 011101 deg = 4 length = 6
e(x) = 001011 deg = 3 length = 6
b(x) = 010110 deg = 4 length = 6
s(x) = 0

a(x) = 011101 deg = 4 length = 6
e(x) = 010110 deg = 4 length = 6
b(x) = 001011 deg = 4 length = 6
s(x) = 0

a(x) = 011101 deg = 4 length = 6
e(x) = 101100 deg = 5 length = 6
b(x) = 110001 deg = 5 length = 6
s(x) = 0

a(x) = 011101 deg = 4 length = 6
e(x) = 110001 deg = 5 length = 6
b(x) = 101100 deg = 5 length = 6
s(x) = 0

FOR g(x)
ERROR IS NULL 0
g(x) = 11101 deg = 4 length = 5
a(x) = 0111010 deg = 5 length = 7
```



### Другой $f(x)$

```
m(x) = 011 deg = 1 length = 3
f(x) = 1101 deg = 3 length = 4
g(x) = 10111 deg = 4 length = 5

for f(x): c(x) = 10 deg = 1 length = 2
for g(x): c(x) = 1001 deg = 3 length = 4

for f(x): a(x) = 011010 deg = 4 length = 6
for g(x): a(x) = 0111001 deg = 5 length = 7

FOR f(x)
ERROR IS NULL 4
a(x) = 011010 deg = 4 length = 6
e(x) = 001101 deg = 3 length = 6
b(x) = 010111 deg = 4 length = 6
s(x) = 0

a(x) = 011010 deg = 4 length = 6
e(x) = 011010 deg = 4 length = 6
b(x) = 000000 deg = 4 length = 6
s(x) = 0

a(x) = 011010 deg = 4 length = 6
e(x) = 100011 deg = 5 length = 6
b(x) = 111001 deg = 5 length = 6
s(x) = 0

a(x) = 011010 deg = 4 length = 6
e(x) = 110100 deg = 5 length = 6
b(x) = 101110 deg = 5 length = 6
s(x) = 0

FOR g(x)
ERROR IS NULL 0
g(x) = 10111 deg = 4 length = 5
a(x) = 0111001 deg = 5 length = 7
```

Можно заметить, что для многочлена  $g(x)$  ошибки обнаруживаются, а для многочлена  $f(x)$  число необнаруженных ошибок всегда равно  $2^{(k-1)}$ , где  $k$  — длина информационного сообщения  $m$ .

Также можно заметить, что в векторе ошибок  $f(x)$  циклически сдвигается влево, тем самым складывая с  $a$ , получаем  $b(x)$ , кратное  $f(x)$  и ошибки не обнаруживаются.

Для  $g(x)$  все ошибки обнаруживаются, можно предположить, что не найдено такого вектора  $e$ , при котором в векторе  $b$  1 расположены таким образом, что можно было сказать, что  $g$  это арифметический и циклический сдвиг в векторе  $b$ , при котором  $b$  кратно  $g$ .

**Вывод:** в ходе лабораторной работы был построен алгоритм работы кодера и декодера, т.е. промоделирована работа кодирования сообщения, которое в свою очередь проходит по каналу связи с некоторыми ошибками, декодер, по принятому сообщению, должен обнаружить ошибки. Декодер может выдавать неверное решение, поскольку закодированное сообщение после прохождения канала связи с допущением ошибок, может принадлежать множеству возможных кодовых слов, но вектор  $a$  не может быть равен вектору  $b$  при допущении ошибок в канале связи.