

Телекоммуникации
Лабораторная работа №3
Исследование методов циклического кодирования

Цель работы: реализация метода побитного циклического кодирования для заданной битовой последовательности.

Задание на выполнение лабораторной работы

1. Подготовка к выполнению лабораторной работы.
 - 1.1. Создать скрипт `lab_checksum.m`, в котором реализовывать все последующие задания.
 - 1.2. Ознакомиться со следующими разделами документации MathWorks MATLAB:
 - `comm.CRCGenerator` System object и `comm.CRCDetector` System object;
 - Generate CRC code bits and append to input data.
2. Реализация побитного циклического кодирования.
 - 2.1. Задать кодовый полином CRC-4-ITU $x^4 + x + 1$ в побитном варианте (массив нулей и единиц), необходимом для `comm.CRCGenerator` и `comm.CRCDetector`. Сформировать объект кодировщика и декодировщика циклического кода для заданного кодового полинома. Установить параметра 'ChecksumsPerFrame' в значение 1 как кодировщика так и для декодировщика.
 - 2.2. Задать в десятичной системе число 46, осуществить его перевод в двоичную систему, используя функцию `de2bi()` в 8-битном представлении (вторым параметром функции `de2bi()` установить 8), полагая, что старший бит располагается слева (третьим параметром функции `de2bi()` установить 'left-msb').
 - 2.3. Осуществить кодирование заданного сообщения заданным кодовым полиномом, используя функцию `step()`. Вычислить длину полученного сообщения, длину закодированного сообщения и величину избыточности.
 - 2.4. Смоделировать ошибку в пятом младшем бите закодированного по каналу связи сообщения, используя функцию `bitxor()`. Реализовать декодирование полученного сообщения с помощью `step()` и определить наличие ошибки (второй выходной параметр функции).
 - Примечание. Ошибку задать как десятичное число, а затем перевести его в двоичный код через `de2bi()`. Количество бит в ошибке должно соответствовать количеству бит закодированного сообщения для того, чтобы можно было сделать побитное исключающее или `bitxor()`. Третьим параметром функции `de2bi()` установить 'left-msb'.
 - 2.5. Вывести в командную строку следующие величины (в двоичном представлении):
 - кодовый полином;
 - кодируемое сообщение (и дополнительно в десятичной);
 - остаток от деления кодируемого сообщения на кодовый полином;
 - закодированное сообщение;
 - модель ошибки;

- полученное приемником сообщение;
- декодированное сообщение.

Примечание. Вектор булевских значений для вывода вполне подойдет. Вывод векторов сделать в виде строк для удобства восприятия.

- 2.6. Реализовать вычисление синдрома ошибки. Вывести аналогично предыдущим пунктам синдром ошибки в командную строку.
- 2.7. Реализовать формирование таблицы синдромов ошибок для заданного кода и вычислить по таблице номер ошибочного бита. Вывести в командную строку таблицу, используя функцию `table()`, и номер ошибочного бита.

3. Реализация побитного “исключающего или”.

- 3.1. Используя в качестве исходного сообщения 7 младших бит десятичного числа 46, реализовать вычисление бита четности “исключающим или”, который записать в старший бит итогового 8-битного закодированного сообщения.
- 3.2. Смоделировать ошибку при передаче закодированного сообщения в пятом младшем бите. Осуществить проверку достоверности передачи (определить факт наличия ошибки на стороне получателя).
- 3.3. Смоделировать ошибку при передаче закодированного сообщения в пятом и третьем младшем бите. Осуществить проверку достоверности передачи (определить факт наличия ошибки на стороне получателя).
- 3.4. Сделать выводы о возможности использования побитного “исключающего или” в качестве алгоритмы вычисления контрольной суммы.

Вопросы для подготовки к защите лабораторной работы

1. В чем суть циклического кодирования?
2. Для каких задач могут применяться методы циклического кодирования?
3. В чем отличие побитного от побайтного метода циклического кодирования, в каких случаях какому методу необходимо отдавать приоритет?
4. Что такое порождающий полином?