МИНИСТЕРСТВО ОБРАЗОВАНИЯ РЕСПУБЛИКИ БЕЛАРУСЬ

Учреждение образования «БЕЛОРУССКИЙ ГОСУДАРСТВЕННЫЙ

ТЕХНОЛОГИЧЕСКИЙ УНИВЕРСИТЕТ»

Факультет информационных технологий

Кафедра информационных систем и технологий

Специальность Информационные системы и технологии

**ОТЧЕТ ПО ЛАБОРАТОРНОЙ РАБОТЕ №7 НА ТЕМУ:**

Перемежение и деперемежение данных в информационно-вычислительных системах

Выполнил студент 3 курса 1 группы

Кашперко Василиса Сергеевна

Минск 2022

**Цель:** приобретение практических навыков использования методов перемежения/деперемежения двоичных данных в информационных системах.

**Задачи:**

1. Закрепить теоретические знания по алгебраическому описанию и использованию методов перемежения/деперемежения двоичных данных в информационных системах.

2. Разработать приложение для реализации метода перемежения/деперемежения символов в сообщениях на основе двоичного алфавита.

3. Результаты выполнения лабораторной работы оформить в виде описания разработанного приложения, методики выполнения экспериментов с использованием приложения и результатов эксперимента.

**Теоретические сведения**

Ранее изученные коды, как и большинство других кодов, которые были разработаны для увеличения надежности каналов передачи и хранения информации, наиболее эффективны, когда возникающие ошибки статистически независимы, т.е. вероятность передачи (хранения в памяти) любого символа одинакова. Однако довольно часто распределение ошибок носит взаимозависимый характер. В таких случаях говорят о группах (или пакетах) ошибок. Такие ошибки характерны и для каналов передачи, и для устройств памяти (главным образом магнитной и полупроводниковой.

Существуют специальные коды, корректирующие пакетные ошибки, однако на практике чаще используют перемежение/деперемежение совместно с традиционными кодами.

Идея перемежения/деперемежения состоит в следующем. Если биты каждого кодового слова *Хn* передаются не в обычной последовательности, а через интервалы, превышающие ожидаемую длину пакета ошибок (в промежутки между битами одного слова вставляются биты других кодовых слов), то при возникновении такого типа ошибки обратная перемежению операция – деперемежение – разнесет («размажет») группу ошибок по всей совокупности кодовых слов, составляющих данное сообщение.

Предложено много алгоритмов перемежения/деперемежения. Наиболее простыми являются блочные.

При блочном перемежении входные биты делятся на блоки, которые последовательно записываются в строки некоторой таблицы.

Передаваемая последовательность делится на блоки по n битов. Каждый блок записывается в отдельную строку таблицы по порядку. Сообщение для передачи или хранения формируется при считывании символов из таблицы по столбцам.

Деперемежение производится в обратной последовательности. Глубина перемежения – это разница между позициями одного и того же символа до и после перемежения. Особенностью является неизменная позиция первого символа.

В общем случае выбор глубины перемежения зависит от двух факторов: с одной стороны, чем больше расстояние между соседними символами, тем большей длины пакет ошибок может быть исправлен.

с другой стороны, чем больше глубина перемежения, тем сложнее аппаратно-программная реализация оборудования и больше задержка сигнала.

Для борьбы с длинными пакетами ошибок желательно увеличивать размеры таблицы. Однако это приводит к увеличению задержки в отправке и декодировании сообщения.

**Ход работы**

Вариант выполненной работы – 4. Рассмотрим процесс передачи информации с использованием кода Хемминга и блокового перемежителя (рис.1).

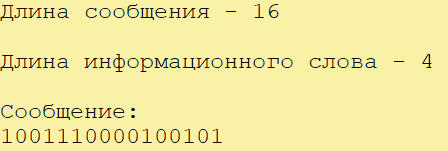


Рисунок 1 – Начальное сообщение

Начальное сообщение после добавления проверочных битов (рис. 2).



Рисунок 2 – Добавлены проверочные биты

Далее составлена матрица перемежения с 6 столбцами в соответствии с вариантом. После заполнения она будет иметь вид, как изображена на рисунке 3.

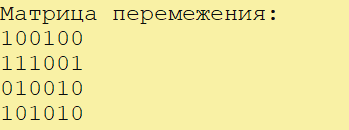


Рисунок 3 – Изображение матрицы перемежения

После перемежения сообщение соответствует последовательности, изображенной на рисунке 4.



Рисунок 4 – Закодированное сообщение после перемежения

В соответствии с вариантом длина пакета ошибок может быть 3, 4, 5 и начало отправки пакета ошибок может быть разное. Введем с 3 позиции и длину пакета равную 3 (рис. 5).

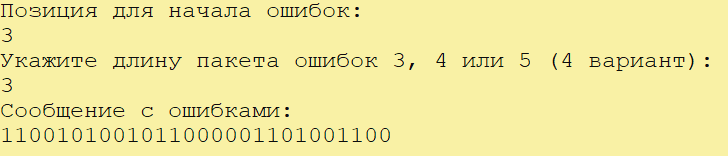


Рисунок 5 – Полученное сообщение с ошибками

Сообщение на выходе канала записывается по столбцам в матрицу тех же размеров (в нашем случае равна 6) деперемежения (рис. 6).

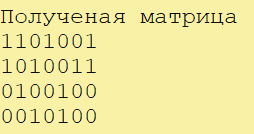


Рисунок 6 – Получаем матрицу деперемежения

Из матрицы деперемежения двоичные символы сообщения считываются по строкам и поступают на декодер кода Хемминга (рис. 7).



Рисунок 7 – Ошибки переставлен по всему сообщению

После деперемежения пакет ошибок преобразован в одиночные (формально – независимые) ошибки кратности 1 для каждой из кодовых комбинаций кода Хемминга. Как помним, такие ошибки код в состоянии исправить (рис. 8).

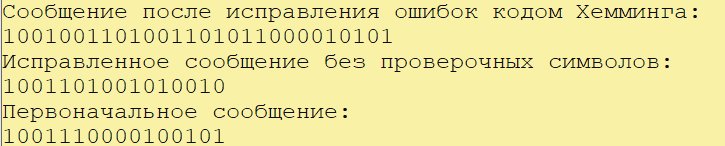


Рисунок 8 – Исправление сообщения

После использования перемежений и деперемежений в ситуациях с сообщениями разной длины было определено, что в сообщении можно исправить все ошибки только при условии, что длинна пакета не будет превышать число количество комбинаций кода, на которое поделено потоковое сообщение.

**Вывод:** в ходе выполнения лабораторной работы я приобрела практические навыки использования методов перемежения и деперемежения двоичных данных в информационных системах. В ходе анализа работы программы было практически выяснено, что из 10 раз запусков одинаковой программы, но в различных позициях отправленного пакетов ошибок входящее сообщение будет исправлено не всегда (сообщение было исправлено 6 раз из 10). Код Хэмминга позволяет гарантированно исправить только одну ошибку, однако, если, в строке после перемежения возникает 2 ошибки (пакет ошибок достаточно длинный), сообщение не может быть исправлено.