**Министерство науки и высшего образования Российской Федерации** **Федеральное государственное бюджетное образовательное учреждение высшего образования** **«Московский государственный технический университет** **имени Н.Э. Баумана**

**(национальный исследовательский университет)»**

**(МГТУ им. Н.Э. Баумана)**

**Факультет «Информатика и системы управления»**

**Кафедра ИУ5 «Системы обработки информации и управления»**

Курс «Сети и телекоммуникации»

Отчет по домашнему заданию №2

«Разработка и реализация алгоритмов кодирования, декодирования и определения корректирующей способности кода в линейных протоколах»

Вариант №22

|  |  |  |
| --- | --- | --- |
| Выполнил: |  | Проверил: |
| студент группы ИУ5-53Б |  | Галкин В. А. |
| Ювенский Лев |  |  |
|  |  |  |

Москва, 2024 г.

**Постановка и метод решения задачи для варианта задания**

Имеется дискретный канал связи, на вход которого подается закодированная в соответствии с вариантом задания кодовая последовательность. В канале возможны ошибки любой кратности. Вектор ошибки может принимать значения от единицы в младшем разряде до единицы во всех разрядах кодового вектора. Для каждого значения вектора ошибки на выходе канала после декодирования определяется факт наличия ошибки и предпринимается попытка ее исправления.

Таблица 1. Индивидуальный вариант домашнего задания.

|  |  |  |  |
| --- | --- | --- | --- |
| № варианта | Информационный вектор | Код | Способность кода |
| 22 | 11111010011 | Ц [15,11] | Ck |

Требуется, используя кодирование циклическим кодом Ц [15,11], определить обнаруживающую способность этого кода Ck:  
Обозначения:

* Ц [n, k] – циклический код с образующим полиномом.
* n – число разрядов в закодированной записи. n = 15
* k – число разрядов в информационной части. k = 11
* z – число дополнительных битов для обнаружения ошибки. z=n−k=4.
* r – число разрядов в образующем векторе. r=z+1=5.
* Ck – корректирующая способность кода.
* g(x) - образующий полином степени z = 4. х4+ х + 1
* Vi - информационный вектор. По заданию 11111010011
* Vc – циклический вектор – результат кодирования Vi.
* I – входящий вектор. Циклический вектор, возможно содержащий ошибки

**Алгоритмы кодирования, декодирования, вычисления корректирующей способности кода для ошибок всех возможных кратностей.**

1. **Алгоритм кодирования циклическим кодом**
   1. Умножить информационный полином Vi(x) на xz . Т. е. сдвинуть информационный вектор на z разрядов в сторону старших, заполнив освободившиеся нулями.
   2. Разделить полученный полином на порождающий полином g(x), после получить остаток p(x).
   3. Сложить p(x) с Vi(x), чтобы получить Vс(x), то есть объединить остаток р(х) в векторной форме и исходный полином xz×Vi(x) для получения кодового слова.
2. **Алгоритм декодирования циклическим кодом**
   1. Разделить принятый полином I(x) на порождающий полином g(x) и проверить остаток от деления S(x) – он является синдромом ошибки.
   2. Если S(x) = 0, то ошибки нет или она не была обнаружена
   3. Если S(x) ≠ 0, то ошибка есть. По виду вектора синдрома определить место ошибки и исправить ее.
   4. Поделить информационный полином Vi(x) на xz для получения исходного полинома, то есть сдвинуть информационный вектор на z разрядов в сторону младших, чтобы получить первоначальный вектор.
3. **Алгоритм вычисления корректирующей способности кода Сk для ошибок всех возможных кратностей**

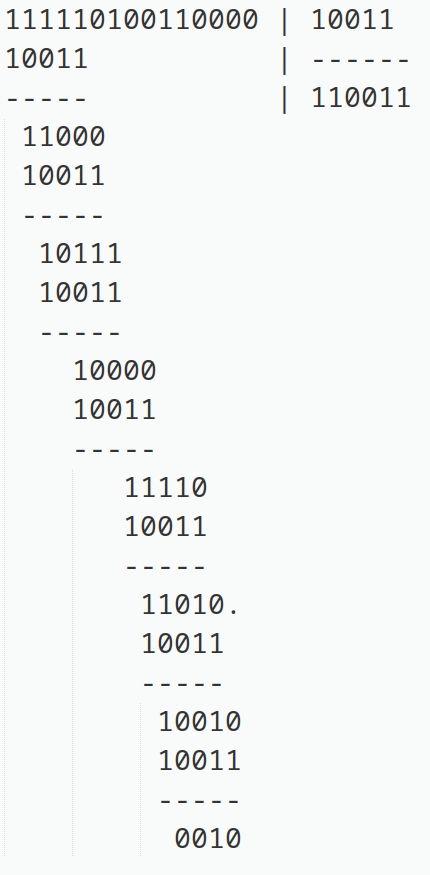
Корректирующая способность кода Ck определяется как отношение числа исправленных ошибок Nk к общему числу ошибок данной кратности, которое определяется как число сочетаний из n (длина кодовой комбинации) по i (кратность ошибки – число единиц в векторе ошибок) -

Корректирующая способность кода вычисляется как

Для подсчета корректирующей способности нужно перебрать все возможные вектора ошибок. Их число

Ошибки нужно сгруппировать по кратности.

* 1. Закодировать данный информационный вектор
  2. Для каждой группы ошибок по кратности
     1. Ввести счетчик , который по умолчанию будет равен 0.
     2. Для каждой ошибки в группе
        1. Наложить на исходный закодированный вектор текущий вектор ошибки e из текущей группы
        2. Воспользовавшись алгоритмом коррекции, декодировать получившуюся последовательность.
        3. Сравнить декодированную последовательность с заданным информационным вектором.
           1. Если информационный и декодированный вектора равны, то коррекция ошибки успешна. Увеличить счетчик на 1
           2. Иначе коррекция ошибки неудачна. Необходимо продолжить выполнение алгоритма, не изменяя счетчик.
  3. Получим n значений для каждой группы. Составить результирующую таблицу корректирующей способности, где номеру группы i будет соответствовать кратность ошибки i и значение , выраженное в процентах.

1. **Алгоритм коррекции ошибки**
   1. Составить таблицу соответствия синдрома ошибки Se ошибке e. Для этого для каждой ошибки e∈[0...1,1...1] рассчитать ее синдром Se, сгруппировать синдромы по кратности ошибки.
   2. Найти в таблице синдром-ошибка синдром Sv, полученный при декодировании входной последовательности V, и определить таким образом вектор ошибки.
   3. Инвертировать те разряды во входной последовательности, которые отмечены в векторе ошибки как 1.
   4. Декодировать измененную входную последовательность еще раз.
2. **Получение циклического кода по варианту задания**
   1. vi = 111.1101.0011. g(x) = 10011.
   2. Сдвиг на 4 бита влево: 111.1101.0011.0000
   3. Деление на образующий полином (см. рисунок 1)  
        
      Рисунок 1 — деление на образующий полином  
      Остаток: 0010
   4. Конкатенация информационного вектора с остатком: 111.1101.0011.0010
3. **Программная реализация алгоритмов**

Для программной реализации модели канала связи, алгоритмов кодирования, декодирования и вычисления корректирующей способности кода для ошибок всех возможных кратностей выберем язык Go. В нём реализованы необходимые побитовые операции, а так же он позволяет отобразить результаты в удобочитаемом формате с использованием HTML шаблонов.

Исходный код разработанного решения можно скачать по ссылке: <https://github.com/Yu-Leo/bmstu-networks-corrective-ability>

1. **Таблица ошибок всех возможных кратностей для n = 15**

Для n = 15 число возможных ошибок составляет 2n - 1 = 32767.

Все ошибки можно разделить на 15 классов: от ошибок кратности 1 до ошибок кратности 15.

Число ошибок в каждом разряде определяется по формуле: , где i – кратность ошибки.

Таблица, отображающая количество ошибок в каждом классе, приведена на рисунке 2. Суммарное количество ошибок равно 32767. Фрагмент таблицы ошибок, сгруппированных по классам, приведён на рисунке 3.

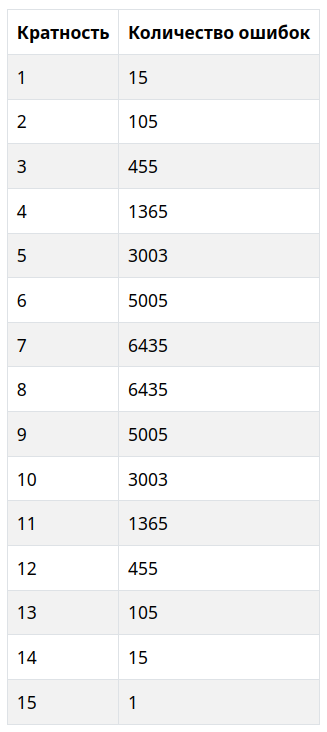
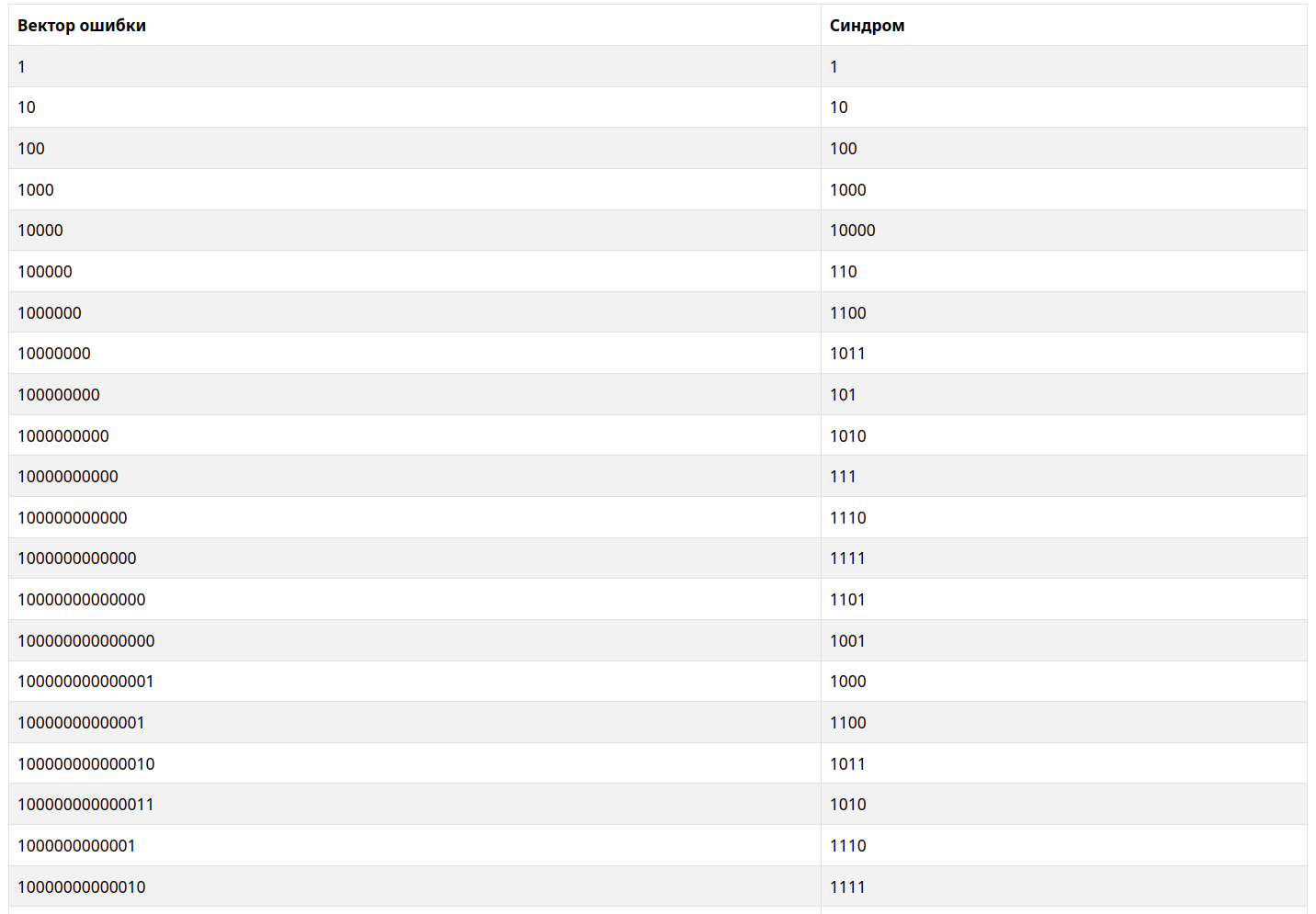
Рисунок 2 — количество ошибок всех возможных кратностей

Рисунок 3 — фрагмент таблицы ошибок всех возможных кратностей

1. **Таблица синдромов для всех ошибок**

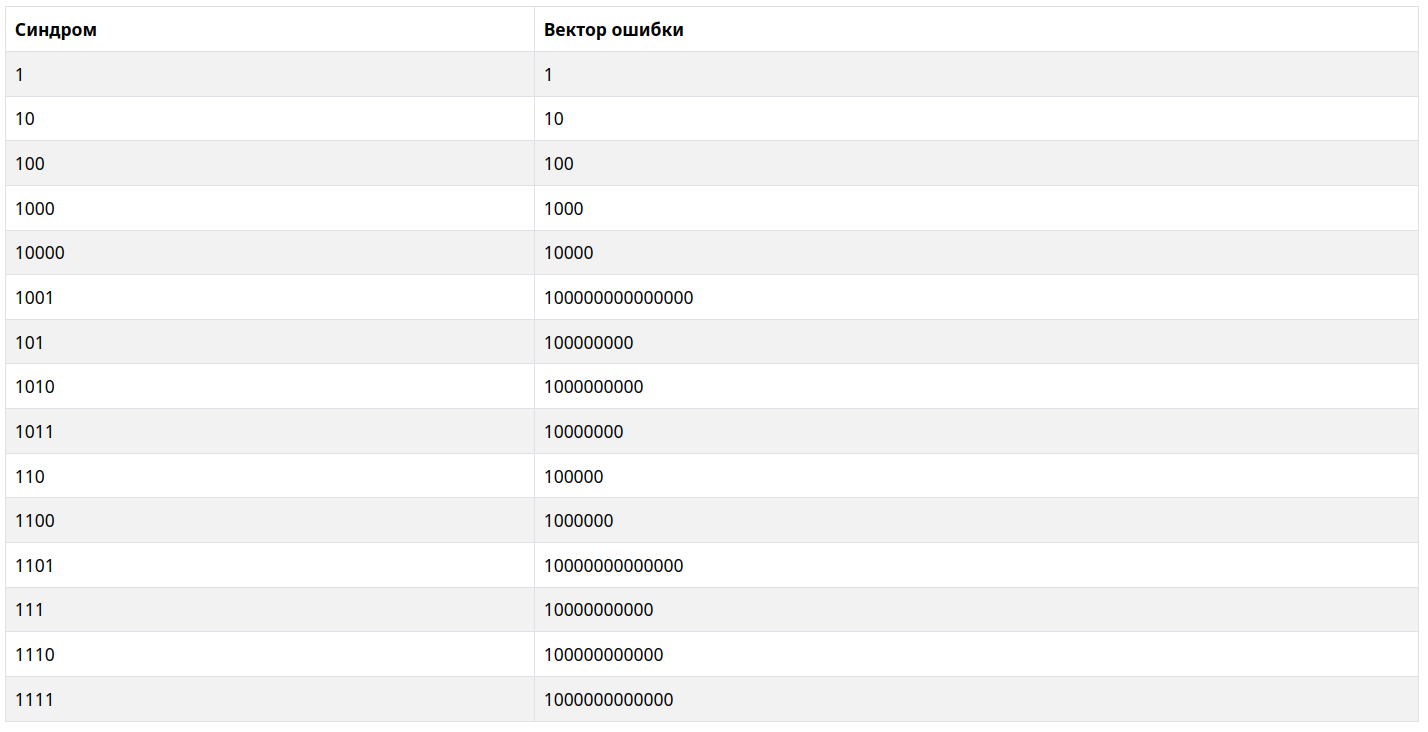
Реализуем алгоритм операции О. Посчитаем синдром для каждой ошибки. Фрагмент таблицы приведён на рисунке 4.

Рисунок 4 — фрагмент таблицы синдромов всех ошибок

1. **Таблица синдромов**

Заметим, что у некоторых ошибок совпадают синдромы. Это происходим из-за того, что вектор симптома имеет слишком мало бит, чтобы покрыть все 32767 ошибок, и с образующим полиномом 100112 (19) может предоставить только 18 уникальных ненулевых значений.

Сгруппируем ошибки по синдромам. Получим таблицу, приведённую на рисунке 5. Будем использовать её для попыток исправления ошибок в полученной последовательности по синдромам. Конечно, однозначно определить ошибку невозможно. Именно из-за этого корректирующая способность не стопроцентная.

Рисунок 5 — ошибки по синдромам для ошибок кратности 1

1. **Результирующая таблица**

Построим результирующую таблицу. Она приведена на рисунке 6.

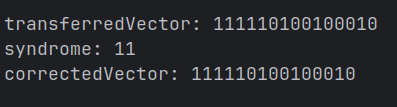
Рисунок 6 — результирующая таблица

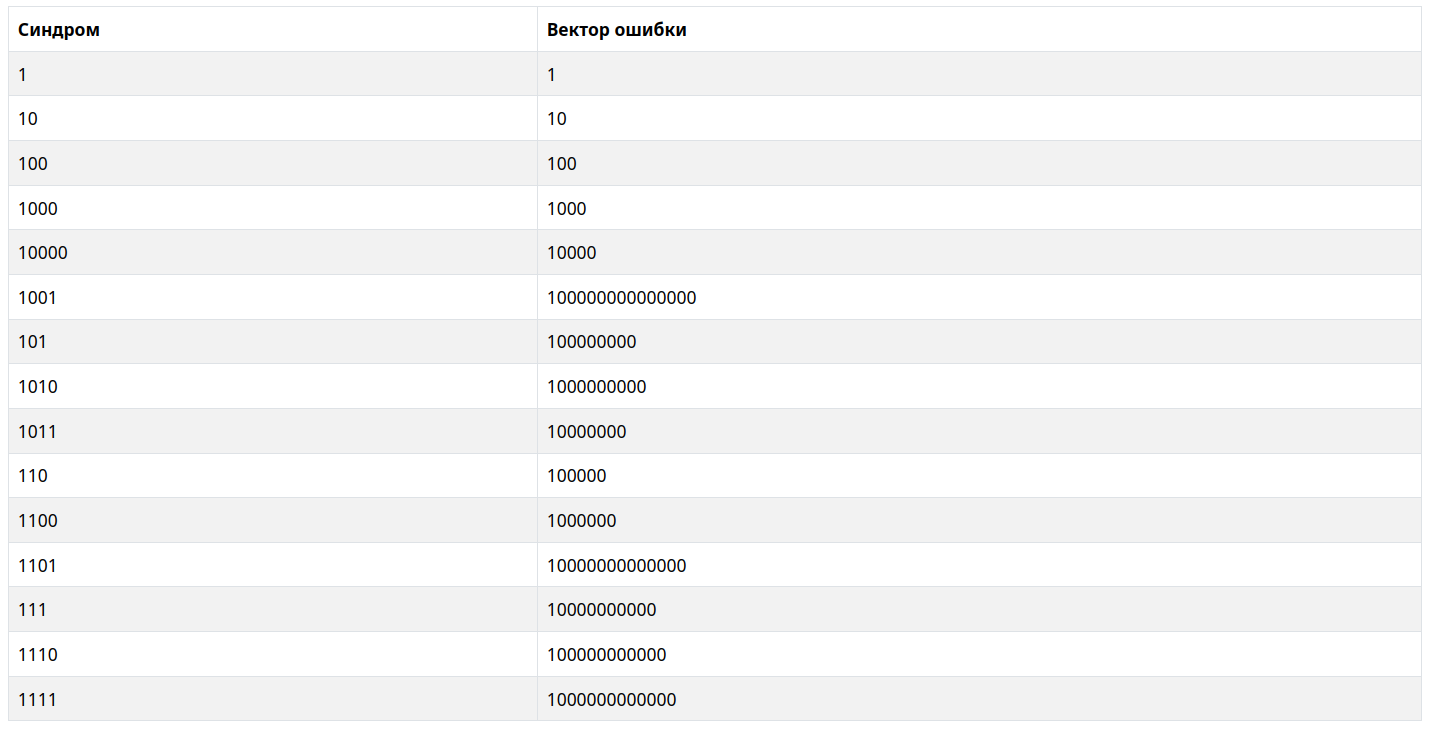
1. **Выводы**

В коде Ц[15,11] под вектор симптома выделено 4 дополнительных бита. Их не хватает, чтобы покрыть все 32767 ошибок. С образующим полиномом 100112 (19) возможно предоставить только 18 уникальных ненулевых значений. Коллизии неизбежны в тех классах ошибок, в которых их число превышает 18. Именно из-за этого корректирующая способность не стопроцентная.

Невозможно завести одну таблицу симптомов для ошибок разных кратностей, чтобы иметь возможность распознавать несколько поврежденных бит.

**Анализ результатов для ошибок кратности 1.** Корректирующая способность: 93,33% — одна ошибка из 15. Число 111.1101.0010.0010 (наложенная на исходный код ошибка e = 100002) дает остаток 11 при делении на образующий полином (см. рисунок 7). Этого остатка нет в таблице симптомов, т.к. ни одна ошибка с кодами от 1, 10, 100, …, 100.0000.0000.0000 не дает такого остатка (см. рисунок 8).

Рисунок 7 — единственная неисправленная ошибка кратности 1

Рисунок 8 — таблица ошибок по синдромам по ошибкам кратности 1

**Анализ результатов для ошибок кратности 14.** Корректирующая способность: 100%. Достигнута за счет полного совпадения таблицы симптомов с полученными симптомами.

**Анализ результатов для ошибок кратности 15.** Набор состоит из единственного вектора ошибки. Вектор ошибки 111.1111.1111.1111 имеет симптом 0, т.е. данный вектор ошибки нельзя обнаружить в при использовании циклических кодов.

Итак, циклический код применим каналам связи, у которых статистическая вероятность ошибки кратности 2 и более минимальна.

**Список используемой литературы**

1. Галкин В.А. Методическое пособие по выполнению домашнего задания по дисциплине «Сети и телекоммуникации», 2024 г.
2. Галкин В.А., Григорьев Ю.А. Телекоммуникации и сети: Учеб. Пособие для вузов.-М.: Изд-во МГТУ им. Н.Э. Баумана, 2003 г.
3. С. М. Рацеев, А. М. Иванцов, П. А. Булдаковский. Об алгоритмах декодирования циклических кодов, 2021, с.87–94
4. Telecommunication technologies - телекоммуникационные технологии (v2.1) [Электронный ресурс] URL: https://www.opennet.ru/docs/RUS/inet\_book/ (дата обращения: 09.10.2024)