|  |
| --- |
| Лабораторная Работа №1 |
| Кодирование данных в телекоммуникационных сетях |
| Сети ЭВМ и телекоммуникации |

|  |
| --- |
| Айдар Шайхиев[Выберите дату]**Группа 3121** |

*Исходное сообщение:* Шайхиев А.А.

*В шестнадцатеричном коде:* D8 E0 E9 F5 E8 E5 E2 20 C0 2E C0 2E

*В двоичном:* 11011000 11100000 11101001 11110101 11101000 11100101 11100010 00100000 11000000 00101110 11000000 00101110

*Длина сообщения*: 12 байт (96 бит)

**Физическое кодирование**

*(результаты кодирования для первых четырех байтов)*

*полагаем, что с = 100 Мбит/с*

**NRZ**



f0 = 1/2tb = c/2 = 50 МГц

fн = 1/24tb = с/24 = 6,25 МГц,

fв = 7f0 = 350 МГц

F = (350 – 4,16) = 343,75 МГц

fср = (1/2\*7 + 1/4\*6 + 1/6\*9+1/10\*10)\*с / 32 = 23,44 МГц

**RZ**



f0 = 1/tb = c/1 = 100 МГц

fн = 1/2tb = с/2 = 50 МГц

fв = 7f0 = 700 МГц

F = (700 – 50) = 650 МГц

fср = (1/2\*14+18)\*с / 32 = 78,13 МГц

**Manchester**



f0 = 1/tb = c/1 = 100 МГц

fн = 1/2tb = с/2 = 50 МГц,

fв = 7f0 = 700 МГц

F = (700 – 50) = 650 МГц

fср = (18+14\*1/2)\*с / 32 = 78,13 МГц

|  |  |  |  |
| --- | --- | --- | --- |
|  | NRZ | RZ | M |
| Минимизация спектра | + | - | - |
| Самосинхронизация | - | + | + |
| Постоянная составляющая | + | - | - |
| Обнаружение ошибок и их исправление | - | + | + |
| Низкая стоимость реализации | + | - | + |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
|  | f0,МГц | fн,МГц | fв,МГц | F,МГц | fср,МГц |
| NRZ | 50 | 6,25 | 350 | 343,75 | 23,44 |
| RZ | 100 | 50 | 700 | 650 | 78,13 |
| M | 100 | 50 | 700 | 650 | 78,13 |

На основе полученных данных можно сделать вывод, что лучшим для передачи сообщения является Манчестерский, следом за ним идет RZ. Несмотря на меньшие значения основной гармоники и полосы пропускания NRZ метода, он не может быть выбран для передачи данного сообщения. Поскольку в данном сообщении присутствует большая часть постоянной составляющей, что вызовет проблемы для передачи, а так же потребует дополнительной линии для сигналов синхронизации.

**Логическое кодирование**

*(метод 4B/5B)*

*В шестнадцатеричном коде:* DC B9 EE 4F AB E4 B8 BE 52 9E D7 A9 CD 7A 9C

*В двоичном коде:* 11011100 10111001 11101110 01001111 10101011 11100100 10111000 10111110 01010010 10011110 11010111 10101001 11001101 01111010 10011100

*Длина сообщения:* 15 байт (120 бит)

*Избыточность:* 3/12=24/96=0,25 (25%)

**RZ**

****

f0 = 1/tb = c/1 = 100 МГц

fн = 1/2tb = с/2 = 50 МГц

fв = 7f0 = 700 МГц

F = (700 – 50) = 650 МГц

fср = (12+20\*1/2)\*с / 32 = 68,75 МГц

**Manchester**

****

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
|  | f0,МГц | fн,МГц | fв,МГц | F,МГц | fср,МГц |
| RZ | 100 | 50 | 700 | 650 | 68,75 |
| M | 100 | 50 | 700 | 650 | 68,75 |

f0 = 1/tb = c/1 = 100 МГц

fн = 1/2tb = с/2 = 50 МГц

fв = 7f0 = 700 МГц

F = (700 – 50) = 650 МГц

fср = (12+20\*1/2)\*с / 32 = 68,75 МГц

Из последней таблицы видно, что оба метода показывают одинаковые результаты. Само логическое кодирование дает лишь небольшое преимущество в значении fср. И поскольку данные методы самосинхронизирующиеся и не имеют постоянную составляющую, то логический метод кодирования для них не эффективен. Лучшим остается Манчестерский код, за счет более низкой стоимости.

**Скремблирование исходного сообщения**

*В 16-ом коде:* C6 12 3F F4 D4 D8 27 E3 A8 61 F1 90

*В 2-ом коде:* 11000110 00010010 00111111 1111 0100 1101 0100 1101 1000 0010 0111 1110 0011 1010 1000 0110 0001 1111 0001 1001 0000

Т.к. мы кодируем не более 32 битов, то достаточно выбрать полином Bi = Ai ⊕ Bi-3 ⊕ Bi-5;

|  |  |  |  |
| --- | --- | --- | --- |
| B1 = А1 = 1  | В9 = А9 ⊕ В6 ⊕ В4 = 0 | В17 = А17 ⊕ В14 ⊕ В12 = 1 | В25 = А25 ⊕ В22 ⊕ В20 = 0 |
| В2 = А2 = 1 | В10 = А10 ⊕ В7 ⊕ В5 = 0 | В18 = А18 ⊕ В15 ⊕ В13 = 0 | В26 = А26 ⊕ В23 ⊕ В21 = 1 |
| В3 = А3 = 0 | В11 = А11 ⊕ В8 ⊕ В6 = 0 | В19 = А19 ⊕ В16 ⊕ В14 = 1 | В27 = А27 ⊕ В24 ⊕ В22 = 1 |
| В4 = А4 ⊕ В1 = 0 | В12 = А12 ⊕ В9 ⊕ В7 = 1 | В20 = А20 ⊕ В17 ⊕ В15 = 0 | В28 = А28 ⊕ В25 ⊕ В23 = 0 |
| В5 = А5 ⊕ В2 = 0 | В13 = А13 ⊕ В10 ⊕ В8 = 0 | В21 = А21 ⊕ В18 ⊕ В16 = 1 | В29 = А29 ⊕ В26 ⊕ В24 = 0 |
| В6 = А6 ⊕ В3 ⊕ В1 = 0 | В14 = А14 ⊕ В11 ⊕ В9 = 0 | В22 = А22 ⊕ В19 ⊕ В17 = 1 | В30 = А30 ⊕ В27 ⊕ В25 = 0 |
| В7 = А7 ⊕ В4 ⊕ В2 = 0 | В15 = А15 ⊕ В12 ⊕ В10 = 0 | В23 = А23 ⊕ В20 ⊕ В18 = 1 | В31 = А31 ⊕ В28 ⊕ В26 = 1 |
| В8 = А8 ⊕ В5 ⊕ В3 = 0 | В16 = А16 ⊕ В13 ⊕ В11 = 0 | В24 = А24 ⊕ В21 ⊕ В19 = 1 | В32 = А32 ⊕ В29 ⊕ В27 = 1 |

**RZ**

****

f0 = 1/tb = c/1 = 100 МГц

fн = 1/2tb = с/2 = 50 МГц

fв = 7f0 = 700 МГц

F = (700 – 50) = 650 МГц

fср = (21+11\*1/2)\*с / 32 = 82,81 МГц

**Manchester**

****

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
|  | f0,МГц | fн,МГц | fв,МГц | F,МГц | fср,МГц |
| RZ | 100 | 50 | 700 | 650 | 82,81 |
| M | 100 | 50 | 700 | 650 | 82,81 |

f0 = 1/tb = c/1 = 100 МГц

fн = 1/2tb = с/2 = 50 МГц,

fв = 7f0 = 700 МГц

F = (700 – 50) = 650 МГц

fср = (21+11\*1/2)\*с / 31 = 82,81 МГц

Из третей таблицы видно, что методы вновь идентичны по своим показателям, и мы получили чуть меньшие значения для fср. Скремблирование сделало код более изменчивым, но это не критично для рассматриваемых методов. Поэтому лидирует Манчестерский код.

**Вывод**

**Сравнительная таблица**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
|  | **f0, МГц** | **fн, МГц** | **fв, МГц** | **F, МГц** | **fср, МГц** |
| Без лог.код. | 4B/5B | Скрем-блиро-вание | Без лог.код. | 4B/5B | Скрем-блиро-вание | Без лог.код. | 4B/5B | Скрем-блиро-вание | Без лог.код. | 4B/5B | Скрем-блиро-вание | Без лог.код. | 4B/5B | Скрем-блиро-вание |
| **RZ** | 100 | 100 | 100 | 50 | 50 | 50 | 700 | 700 | 700 | 650 | 650 | 650 | 78,13 | 68,75 | 82,81 |
| **M** | 100 | 100 | 100 | 50 | 50 | 50 | 700 | 700 | 700 | 650 | 650 | 650 | 78,13 | 68,75 | 82,81 |
| **NRZ** | 50 | - | - | 6,25 | - | - | 350 | - | - | 343,75 | - | - | 23,44 | - | - |

По данным из сравнительной таблицы можно сделать вывод, что логическое кодирование не эффективно для таких методов физического кодирования как Манчестерский и RZ (т. к. используется для уменьшения постоянной составляющей, несвойственной рассматриваемым методам). Поэтому Манчестерский — лучший метод физического кодирования, т. к. обладает более низкой стоимостью реализации по сравнению с RZ.

Что касается логического кодирования, то избыточное кодирование позволяет избавиться от длинных последовательностей нулей и единиц, а так же улучшит распознаваемость кода, но взамен потребуется передача большего трафика. Скремблирование приводит информацию к выводу, похожему на случайный набор нулей и единиц, что улучшает спектральные и статистические характеристики. Однако за счет этого увеличивается стоимость реализации.