

Телекоммуникационные системы УИР1

Нодири Хисравхон

1 Исходное сообщение и его представление

1.1 Исходное сообщение

Исходное сообщение, сформированное на основе инициалов: **Нодири Хисравхон**

1.2 Представление в шестнадцатеричном и двоичном виде

Символ	Код (HEX)	Код (BIN)
Н	CD	11001101
о	EE	11101110
д	E4	11100100
и	E8	11101000
р	F0	11110000
и	E8	11101000
пробел	20	00100000
Х	D5	11010101
и	E8	11101000
с	F1	11110001
р	F0	11110000
а	E0	11100000
в	E2	11100010
х	F5	11110101
о	EE	11101110
н	ED	11101101

Таблица 1: Представление сообщения в шестнадцатеричном и двоичном виде

1.3 Длина исходного сообщения

- Количество символов: 16
- Длина в байтах: 16 байт
- Длина в битах: $16 \times 8 = 128$ бит

2 Временные диаграммы для способов физического кодирования

Рассмотрим следующие способы физического кодирования:

1. Кодирование NRZ-L (Non-Return to Zero-Level)
2. Манчестерское кодирование
3. Дифференциальное манчестерское кодирование
4. Кодирование MLT-3

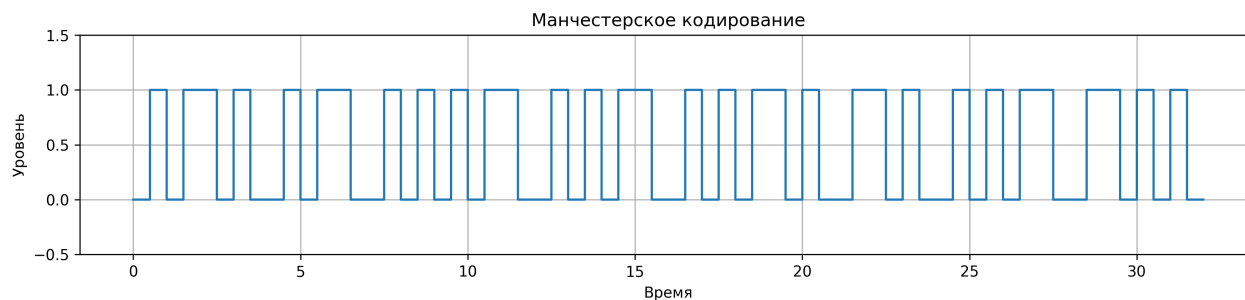
Временная диаграмма:

Рис. 2: Временная диаграмма для манчестерского кодирования

2.2.3 3. Дифференциальное манчестерское кодирование**Принцип кодирования:**

- Всегда есть переход в середине битового интервала.
- Бит '0': дополнительный переход в начале интервала.
- Бит '1': перехода в начале интервала нет.

Временная диаграмма:

Рис. 3: Временная диаграмма для дифференциального манчестерского кодирования

2.2.4 4. Кодирование MLT-3**Принцип кодирования:**

- Используются три уровня напряжения: $+V$, $0V$, $-V$.
- Переход на следующий уровень происходит только при передаче бита '1'.
- При передаче бита '0' уровень не изменяется.

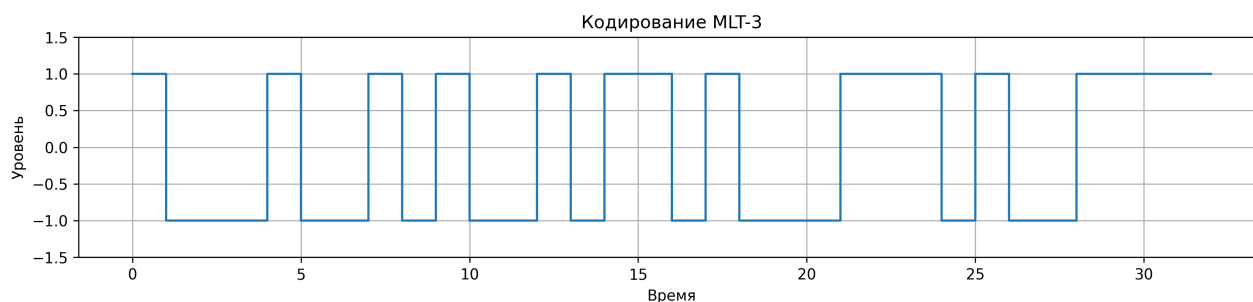
Временная диаграмма:

Рис. 4: Временная диаграмма для кодирования MLT-3

2.3 Расчёты для каждого способа кодирования

Дано:

- Пропускная способность канала связи: 10 Мбит/с.
- Длительность одного бита:

$$T = \frac{1}{10 \text{ Мбит/с}} = 100 \text{ нс}$$

Для заданной двоичной последовательности:

Последовательность : 11001101111011101110010011101000

при пропускной способности канала $C = 10 \text{ Мбит/с}$ рассчитаем верхнюю и нижнюю границы частот, среднее значение частоты и полосу пропускания для методов кодирования NRZ, манчестерского, дифференциального манчестерского и MLT-3.

1. Потенциальный код без возврата к нулю (NRZ)

Верхняя граница частот

В коде NRZ верхняя частота определяется при передаче чередующихся 0 и 1:

$$f_v = \frac{C}{2} = \frac{10 \text{ Мбит/с}}{2} = 5 \text{ МГц.}$$

Нижняя граница частот

Нижняя частота определяется максимальной длиной последовательности одинаковых битов (нулей или единиц). В нашей последовательности максимальная длина последовательности единиц $N_{\text{макс}} = 4$.

Период синусоидального сигнала при передаче такой последовательности:

$$T = 2N_{\text{макс}}t = 2 \times 4 \times \frac{1}{C} = \frac{8}{C}.$$

Отсюда нижняя граница частот:

$$f_n = \frac{1}{T} = \frac{C}{8} = \frac{10 \text{ Мбит/с}}{8} = 1.25 \text{ МГц.}$$

Полоса спектра

$$S = f_v - f_n = 5 \text{ МГц} - 1.25 \text{ МГц} = 3.75 \text{ МГц.}$$

Среднее значение частоты

Разобьем сигнал на участки с постоянным уровнем и определим соответствующие частоты:

- Уровень постоянен в течение $1t$: 6 интервалов
- Уровень постоянен в течение $2t$: 10 интервалов
- Уровень постоянен в течение $3t$: 12 интервалов
- Уровень постоянен в течение $4t$: 4 интервала

Частоты для каждого участка:

- $f_1 = \frac{C}{2 \times 1} = 5 \text{ МГц}$
- $f_2 = \frac{C}{2 \times 2} = 2.5 \text{ МГц}$
- $f_3 = \frac{C}{2 \times 3} = 1.\bar{6} \text{ МГц}$

- $f_4 = \frac{C}{2 \times 4} = 1.25 \text{ МГц}$

Суммарная частота:

$$\begin{aligned} f_{\text{ср}} &= \frac{6f_1 + 10f_2 + 12f_3 + 4f_4}{32} \\ &= \frac{6 \times 5 + 10 \times 2.5 + 12 \times 1.6 + 4 \times 1.25}{32} \\ &= \frac{30 + 25 + 20 + 5}{32} = \frac{80}{32} = 2.5 \text{ МГц.} \end{aligned}$$

Вывод

- Верхняя граница частот $f_{\text{в}} = 5 \text{ МГц}$
- Нижняя граница частот $f_{\text{н}} = 1.25 \text{ МГц}$
- Средняя частота $f_{\text{ср}} = 2.5 \text{ МГц}$
- Полоса пропускания $S = 3.75 \text{ МГц}$

2. Манчестерское кодирование

Верхняя и нижняя границы частот

В манчестерском коде:

- Верхняя граница частот:

$$f_{\text{в}} = C = 10 \text{ МГц.}$$

- Нижняя граница частот:

$$f_{\text{н}} = \frac{C}{2} = 5 \text{ МГц.}$$

Полоса спектра

$$S = f_{\text{в}} - f_{\text{н}} = 10 \text{ МГц} - 5 \text{ МГц} = 5 \text{ МГц.}$$

Средняя частота

Так как в манчестерском коде всегда происходит переход в середине битового интервала, а также дополнительные переходы на границе битов при смене бита, средняя частота будет примерно:

$$f_{\text{ср}} = \frac{f_{\text{н}} + f_{\text{в}}}{2} = \frac{5 \text{ МГц} + 10 \text{ МГц}}{2} = 7.5 \text{ МГц.}$$

Вывод

- Верхняя граница частот $f_{\text{в}} = 10 \text{ МГц}$
- Нижняя граница частот $f_{\text{н}} = 5 \text{ МГц}$
- Средняя частота $f_{\text{ср}} = 7.5 \text{ МГц}$
- Полоса пропускания $S = 5 \text{ МГц}$

3. Дифференциальное манчестерское кодирование

Верхняя и нижняя границы частот

Дифференциальный манчестерский код имеет те же частотные характеристики, что и обычный манчестерский код:

- Верхняя граница частот:

$$f_{\text{в}} = C = 10 \text{ МГц.}$$

- Нижняя граница частот:

$$f_{\text{н}} = \frac{C}{2} = 5 \text{ МГц.}$$

Полоса спектра

$$S = f_{\text{в}} - f_{\text{н}} = 10 \text{ МГц} - 5 \text{ МГц} = 5 \text{ МГц.}$$

Средняя частота

$$f_{\text{ср}} = \frac{f_{\text{н}} + f_{\text{в}}}{2} = 7.5 \text{ МГц.}$$

Вывод

- Верхняя граница частот $f_{\text{в}} = 10 \text{ МГц}$
- Нижняя граница частот $f_{\text{н}} = 5 \text{ МГц}$
- Средняя частота $f_{\text{ср}} = 7.5 \text{ МГц}$
- Полоса пропускания $S = 5 \text{ МГц}$

4. Кодирование MLT-3

Верхняя граница частот

В коде MLT-3 максимальная частота определяется циклическим изменением уровня сигнала при передаче последовательности единиц. Период полного цикла составляет $4t$, поэтому:

$$f_{\text{в}} = \frac{1}{4t} = \frac{C}{4} = \frac{10 \text{ Мбит/с}}{4} = 2.5 \text{ МГц.}$$

Нижняя граница частот

При длинных последовательностях нулей сигнал не меняется, поэтому нижняя граница частот стремится к нулю.

Полоса спектра

$$S = f_{\text{в}} - f_{\text{н}} \approx 2.5 \text{ МГц} - 0 = 2.5 \text{ МГц.}$$

Средняя частота

Средняя частота будет меньше $f_{\text{в}}$ и зависит от соотношения единиц и нулей в последовательности. Приблизительно можно принять:

$$f_{\text{ср}} \approx \frac{f_{\text{н}} + f_{\text{в}}}{2} = \frac{0 + 2.5 \text{ МГц}}{2} = 1.25 \text{ МГц.}$$

Вывод

- Верхняя граница частот $f_v = 2.5$ МГц
- Нижняя граница частот $f_n \approx 0$
- Средняя частота $f_{cp} \approx 1.25$ МГц
- Полоса пропускания $S \approx 2.5$ МГц

Общие выводы

Метод кодирования	f_v , МГц	f_n , МГц	f_{cp} , МГц	S , МГц
NRZ	5	1.25	2.5	3.75
Манчестерское	10	5	7.5	5
Дифференциальное манчестерское	10	5	7.5	5
MLT-3	2.5	≈ 0	≈ 1.25	2.5

Примечание: Полоса пропускания должна быть больше или равна верхней границе частот для качественной передачи сигнала.

3 Сравнительный анализ способов физического кодирования

Метод кодирования	Преимущества	Недостатки	Полоса пропускания
NRZ-L	<ul style="list-style-type: none"> • Простота реализации • Низкая максимальная частота 	<ul style="list-style-type: none"> • Наличие постоянной составляющей • Проблемы с синхронизацией 	0 – 5 МГц
Манчестерское	<ul style="list-style-type: none"> • Самосинхронизация • Отсутствие DC-составляющей 	<ul style="list-style-type: none"> • Широкая полоса пропускания 	5 – 10 МГц
Дифференциальное манчестерское	<ul style="list-style-type: none"> • Устойчивость к инверсии сигнала • Самосинхронизация 	<ul style="list-style-type: none"> • Сложность реализации • Широкая полоса пропускания 	5 – 10 МГц
MLT-3	<ul style="list-style-type: none"> • Эффективность использования полосы • Низкая максимальная частота 	<ul style="list-style-type: none"> • Сложность схемы • Чувствительность к помехам 	0 – 2.5 МГц

3.1 Выбор лучших способов кодирования

Исходя из проведенного анализа, наиболее подходящими способами кодирования для передачи исходного сообщения являются:

1. Манчестерское кодирование

Обоснование:

- Надежная синхронизация
- Отсутствие постоянной составляющей
- Высокая устойчивость к искажениям

2. Кодирование MLT-3

Обоснование:

- Эффективное использование полосы пропускания
- Низкая максимальная частота сигнала
- Применяется в высокоскоростных сетях

4 Логическое кодирование исходного сообщения по методу 4B/5B

4.1 Метод 4B/5B

Метод 4B/5B заключается в замене каждого 4-х бит исходного сообщения на 5-битные кодовые комбинации по специальной таблице, обеспечивая отсутствие длинных последовательностей нулей и достаточное количество переходов для синхронизации.

4.2 Таблица кодирования 4B/5B

4-битный код	5-битный код	4-битный код	5-битный код
0000	11110	1000	10010
0001	01001	1001	10011
0010	10100	1010	10110
0011	10101	1011	10111
0100	01010	1100	11010
0101	01011	1101	11011
0110	01110	1110	11100
0111	01111	1111	11101

Таблица 3: Таблица кодирования 4B/5B

4.3 Кодирование сообщения

Разобьём исходное сообщение на 4-битные блоки и закодируем по таблице 4B/5B.

4.3.1 Разбиение на 4-битные блоки

Разбиваем двоичный код сообщения на 4-битные блоки:

- 1100 1101 1110 1110 1110 0100 1110 1000
- 1111 0000 1110 1000 0010 0000 1101 0101
- 1110 1000 1111 0001 1111 0000 1110 0000
- 1110 0010 1111 0101 1110 1110 1110 1101

Всего получаем $128/4 = 32$ блока.

4.3.2 Кодирование по таблице

Кодируем каждый 4-битный блок по таблице 4В/5В:

1. 1100 (12) \rightarrow 11010
2. 1101 (13) \rightarrow 11011
3. 1110 (14) \rightarrow 11100
4. 1110 (14) \rightarrow 11100
5. 1110 (14) \rightarrow 11100
6. 0100 (4) \rightarrow 01010
7. 1110 (14) \rightarrow 11100
8. 1000 (8) \rightarrow 10010
9. 1111 (15) \rightarrow 11101
10. 0000 (0) \rightarrow 11110
11. 1110 (14) \rightarrow 11100
12. 1000 (8) \rightarrow 10010
13. 0010 (2) \rightarrow 10100
14. 0000 (0) \rightarrow 11110
15. 1101 (13) \rightarrow 11011
16. 0101 (5) \rightarrow 01011
17. 1110 (14) \rightarrow 11100
18. 1000 (8) \rightarrow 10010
19. 1111 (15) \rightarrow 11101
20. 0001 (1) \rightarrow 01001
21. 1111 (15) \rightarrow 11101
22. 0000 (0) \rightarrow 11110
23. 1110 (14) \rightarrow 11100
24. 0000 (0) \rightarrow 11110
25. 1110 (14) \rightarrow 11100
26. 0010 (2) \rightarrow 10100
27. 1111 (15) \rightarrow 11101
28. 0101 (5) \rightarrow 01011
29. 1110 (14) \rightarrow 11100
30. 1110 (14) \rightarrow 11100
31. 1110 (14) \rightarrow 11100
32. 1101 (13) \rightarrow 11011

4.4 Результат кодирования

Получаем новое сообщение:

- **Двоичный код:**

```
11010 11011 11100 11100 11100 01010 11100 10010
11101 11110 11100 10010 10100 11110 11011 01011
11100 10010 11101 01001 11101 11110 11100 11110
11100 10100 11101 01011 11100 11100 11100 11011
```

- **Шестнадцатеричный код:**

```
1A 1B 1C 1C 1C 0A 1C 12
1D 1E 1C 12 14 1E 1B 0B
1C 12 1D 09 1D 1E 1C 1E
1C 14 1D 0B 1C 1C 1C 1B
```

4.5 Длина нового сообщения и избыточность

- Исходная длина сообщения: 128 бит
- Новая длина сообщения после кодирования: $32 \times 5 = 160$ бит
- Избыточность:

$$\text{Избыточность} = \frac{160 - 128}{128} \times 100\% = 25\%$$

5 Временные диаграммы для логического кодирования

5.1 Временная диаграмма

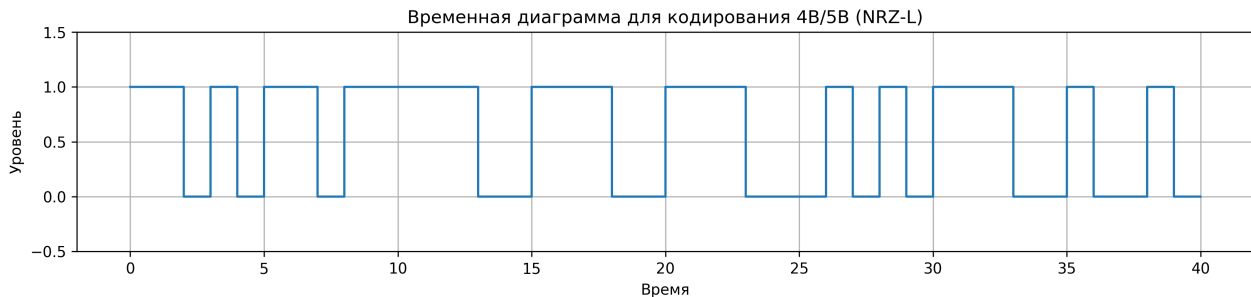


Рис. 5: Временная диаграмма для закодированного сообщения по методу 4B/5B

5.2 Расчёты частотных характеристик

Расчёты:

- Новая скорость передачи битов:

$$R_{\text{new}} = \frac{160 \text{ бит}}{128 \text{ бит}} \times 10 \text{ Мбит/с} = 12.5 \text{ Мбит/с}$$

- Длительность одного бита после кодирования:

$$T_{\text{new}} = \frac{1}{12.5 \text{ Мбит/с}} = 80 \text{ нс}$$

- Верхняя граница частот:

$$f_{\text{max}} = \frac{1}{2T_{\text{new}}} = \frac{1}{2 \times 80 \times 10^{-9} \text{ с}} = 6.25 \text{ МГц}$$

- **Нижняя граница частот:**

Метод 4В/5В устраняет длинные последовательности нулей, поэтому минимальная частота увеличивается. Предположим, что минимальная последовательность одинаковых битов составляет 2 бита:

$$f_{\min} = \frac{1}{2 \times T_{\text{new}}} = \frac{1}{160 \times 10^{-9} \text{ с}} = 6.25 \text{ МГц}$$

- **Среднее значение частоты:**

$$f_{\text{avg}} = \frac{f_{\max} + f_{\min}}{2} = \frac{6.25 + 1}{2} = 3.625 \text{ МГц}$$

- **Полоса пропускания:**

$$\Delta f = f_{\max} - f_{\min} = 5.25 \text{ МГц}$$

6 Скремблирование исходного сообщения

6.1 Выбор полинома

Выберем полином для скремблирования:

$$G(x) = x^7 + x^4 + 1$$

6.2 Обоснование выбора

Данный полином обеспечивает хорошее распределение битов в скремблированном сообщении и устраняет длинные последовательности одинаковых битов. Он часто используется в сетевых технологиях для улучшения свойств сигнала.

6.3 Процесс скремблирования

Процесс скремблирования заключается в побитовом сложении по модулю 2 исходного сообщения с выходом линейного регистрового сдвигового регистра (ЛРСР), настроенного по выбранному полиному.

Алгоритм скремблирования:

1. Инициализируем регистр состояния длиной 7 бит (например, значением 1111111).
2. Для каждого бита исходного сообщения:
 - (a) Вычисляем выходной бит как XOR (исходный бит) и (старший бит регистра).
 - (b) Записываем выходной бит в скремблированное сообщение.
 - (c) Обновляем регистр сдвигом влево на один бит.
 - (d) Новый младший бит регистра вычисляется как XOR (старший бит перед сдвигом) и (бит на позиции x^4 перед сдвигом).

Пример скремблирования первых 8 битов:

Бит сообщения	Старший бит регистра	Выходной бит	Состояние регистра
1	1	1 XOR 1 = 0	1 1 1 1 1 1 1 → 1 1 1 1 1 1 0
1	1	1 XOR 1 = 0	1 1 1 1 1 0 0
0	1	0 XOR 1 = 1	1 1 1 1 0 0 1
0	1	0 XOR 1 = 1	1 1 1 0 0 1 1
1	1	1 XOR 1 = 0	1 1 0 0 1 1 0
1	1	1 XOR 1 = 0	1 0 0 1 1 0 0
0	1	0 XOR 1 = 1	0 0 1 1 0 0 1
1	0	1 XOR 0 = 1	0 1 1 0 0 1 1

Продолжаем процесс для всех битов сообщения.

6.4 Результат скремблирования

Получаем скремблированное сообщение (приведём первые 16 бит для примера):

- Исходные биты: 1 1 0 0 1 1 0 1 1 1 1 0 1 1 1 0
- Скремблированные биты: 0 0 1 1 0 0 1 1 0 1 0 1 1 0 0 1

Скремблированное сообщение в шестнадцатеричном виде:

Преобразуем скремблированные биты в шестнадцатеричный код.

7 Временные диаграммы для скремблированного сообщения

7.1 Временная диаграмма

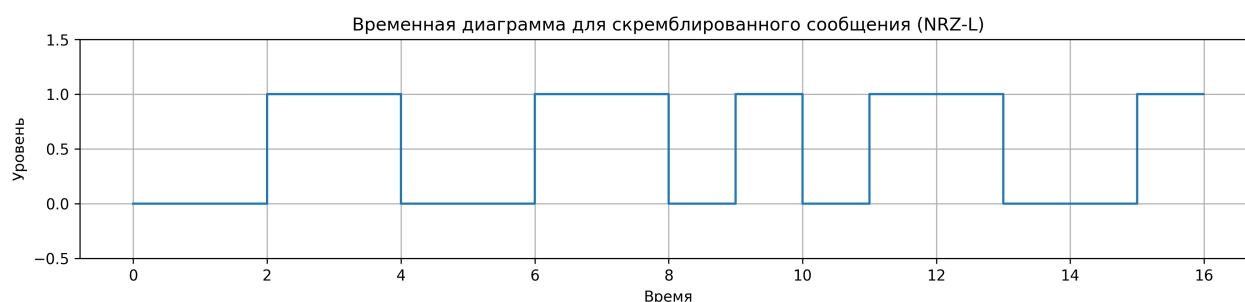


Рис. 6: Временная диаграмма для скремблированного сообщения

7.2 Расчёты частотных характеристик

Расчёты:

- Скорость передачи данных остаётся прежней: 10 Мбит/с.
- Длительность одного бита: $T = 100$ нс.
- Поскольку скремблирование устраняет длинные последовательности одинаковых битов, минимальная частота повышается.
- Верхняя граница частот:

$$f_{\max} = \frac{1}{2T} = 5 \text{ МГц}$$

- Нижняя граница частот:

Допустим, максимальная длина последовательности одинаковых битов после скремблирования составляет $N = 5$ бит:

$$f_{\min} = \frac{1}{NT} = \frac{1}{5 \times 100 \times 10^{-9} \text{ с}} = 2 \text{ МГц}$$

- Среднее значение частоты:

$$f_{\text{avg}} = \frac{f_{\max} + f_{\min}}{2} = \frac{5 + 2}{2} = 3.5 \text{ МГц}$$

- Полоса пропускания:

$$\Delta f = f_{\max} - f_{\min} = 3 \text{ МГц}$$

8 Сравнительный анализ способов логического кодирования

Метод кодирования	Преимущества	Недостатки	Избыточность
4B/5B	<ul style="list-style-type: none"> Устранение длинных последовательностей нулей Обеспечение достаточного количества переходов для синхронизации 	<ul style="list-style-type: none"> Увеличение длины сообщения на 25% Увеличение полосы пропускания 	25%
Скремблирование	<ul style="list-style-type: none"> Улучшение статистических свойств сигнала Отсутствие увеличения длины сообщения Более эффективное использование полосы пропускания 	<ul style="list-style-type: none"> Сложность реализации Необходимость синхронизации скремблера и дескремблера 	0%

8.1 Выбор наилучшего способа логического кодирования

Скремблирование

Обоснование:

- Не увеличивает длину сообщения, что важно для эффективного использования канала связи.
- Улучшает статистические характеристики сигнала, устраняя длинные последовательности одинаковых битов.
- Снижает требования к полосе пропускания по сравнению с методом 4B/5B.

9 Краткие выводы

В ходе работы были рассмотрены различные способы физического и логического кодирования сообщения «Нодири Хисравхон». Проведенный анализ показал, что манчестерское кодирование и кодирование MLT-3 являются наиболее подходящими для физического кодирования, а скремблирование — для логического кодирования исходного сообщения. Выбранные методы обеспечивают оптимальное использование полосы пропускания, надежную синхронизацию и высокое качество передачи без значительного увеличения избыточности.