**Министерство науки и высшего образования Российской Федерации**

**Федеральное государственное автономное образовательное учреждение высшего образования**

**«Национальный исследовательский университет ИТМО»**

***Факультет Программной инженерии и компьютерной техники***

**Учебно-исследовательская работа №1**

**«Кодирование данных в телекоммуникационных сетях»**

**по «Телекоммуникационным системам»**

**Выполнил: Группа P33312 Хайкин О. И.**

**Преподаватель:**

**Алиев Т.И.**

**Санкт-Петербург, 2023**

Содержание

[Этап 1 - Формирование сообщения 3](#_Toc148645759)

[Кодируемое сообщение 3](#_Toc148645760)

[Представление в шестнадцатеричной системе 3](#_Toc148645761)

[Представление в двоичной системе 3](#_Toc148645762)

[Получившееся сообщение 3](#_Toc148645763)

[Этап 2 - Физическое кодирование 3](#_Toc148645764)

[Манчестер 4](#_Toc148645765)

[NRZ 5](#_Toc148645766)

[RZ 6](#_Toc148645767)

[AMI 7](#_Toc148645768)

[Дифференциальный манчестер 8](#_Toc148645769)

[Итоги 8](#_Toc148645770)

[Этап 3 - Избыточное кодирование - 4B/5B 9](#_Toc148645771)

[AMI (Избыточное кодирование - 4B/5B) 10](#_Toc148645772)

[Этап 4 - Скремблирование – Bi = Ai ⊕ Bi-5 ⊕ Bi-7 11](#_Toc148645773)

[Процесс скремблирования 11](#_Toc148645774)

[Результат скремблирования 13](#_Toc148645775)

[NRZ (Скремблирование - Bi = Ai ⊕ Bi-5 ⊕ Bi-7) 13](#_Toc148645776)

[Этап 5 - Сравнительный анализ результатов 14](#_Toc148645777)

# Этап 1 - Формирование сообщения

## Кодируемое сообщение

“Хайкин О.И.”

Преобразуем сообщение в шестнадцатеричный код. Покажем «новое» значение каждого символа:

|  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Символ | Х | а | й | к | и | н |  | О | . | И | . |
| 16x-представление | D5 | E0 | E9 | EA | E8 | ED | 20 | CE | 2E | C8 | 2E |

Теперь представим это же сообщение в двоичной системе:

|  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 16x-представление | D | 5 | E | 0 | E | 9 | E | A | E | 8 | E |
| 2x-представление | 1101 | 0101 | 1110 | 0000 | 1110 | 1001 | 1110 | 1010 | 1110 | 1000 | 1110 |
| 16x-представление  (продолжение) | D | 2 | 0 | C | E | 2 | E | C | 8 | 2 | E |
| 2x-представление  (продолжение) | 1101 | 0010 | 0000 | 1100 | 1110 | 0010 | 1110 | 1100 | 1000 | 0010 | 1110 |

## Получившееся сообщение

|  |  |
| --- | --- |
| Тип | Сообщение |
| Исходное | Хайкин О.И. |
| 16-ная система | D5 E0 E9 EA E8 ED 20 CE 2E C8 2E |
| 2-ная система | 1101 0101 1110 0000 1110 1001 1110 1010 1110 1000 1110 1101 0010 0000 1100 1110 0010 1110 1100 1000 0010 1110 |

Длина сообщения: 11 байт (88 бит)

# Этап 2 - Физическое кодирование

Будем кодировать первые 4 байта сообщения, т.е.

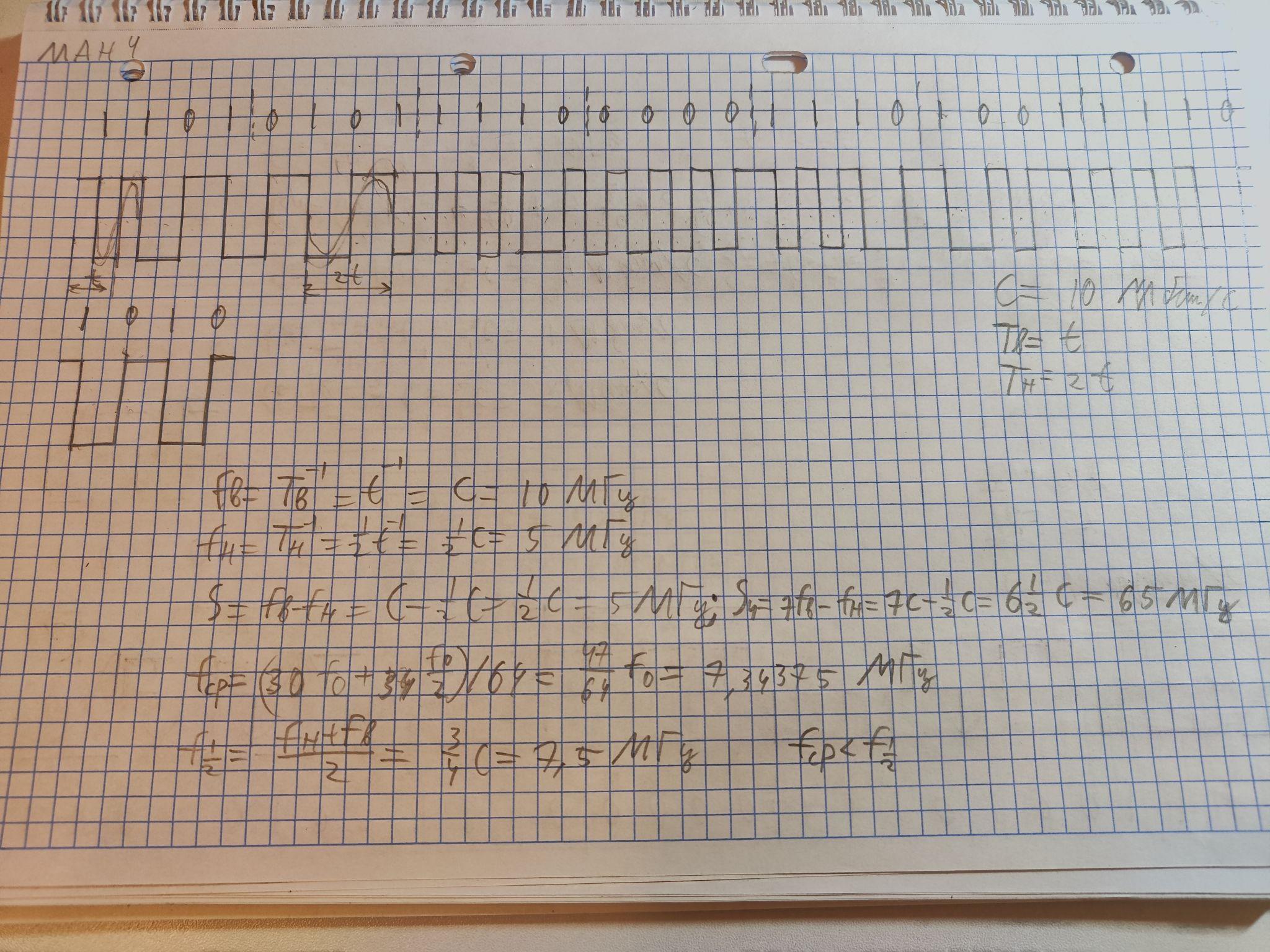
M = "Хайк” = D5 E0 E9 EA = 1101 0101 1110 0000 1110 1001 1110 1010

Для каждого метода кодирования приведём временную диаграмму, а также рассчитаем следующие значения:

* fв: самая высокая частота сигнала
* fн: самая низкая частота сигнала
* S: спектр частот сигнала
* S4: спектр частот для качественной передачи сигнала (4 гармоники).
* F: полоса пропускания, необходимая для качественной передачи сигнала. Определяется на основе значение S4 с возможным округлением в большую сторону. Обусловим это тем, что в реальности никто не станет настраивать полосу пропускания с точностью до десятых/сотых/тысячных и т.д.
* fср: средняя частота сигнала

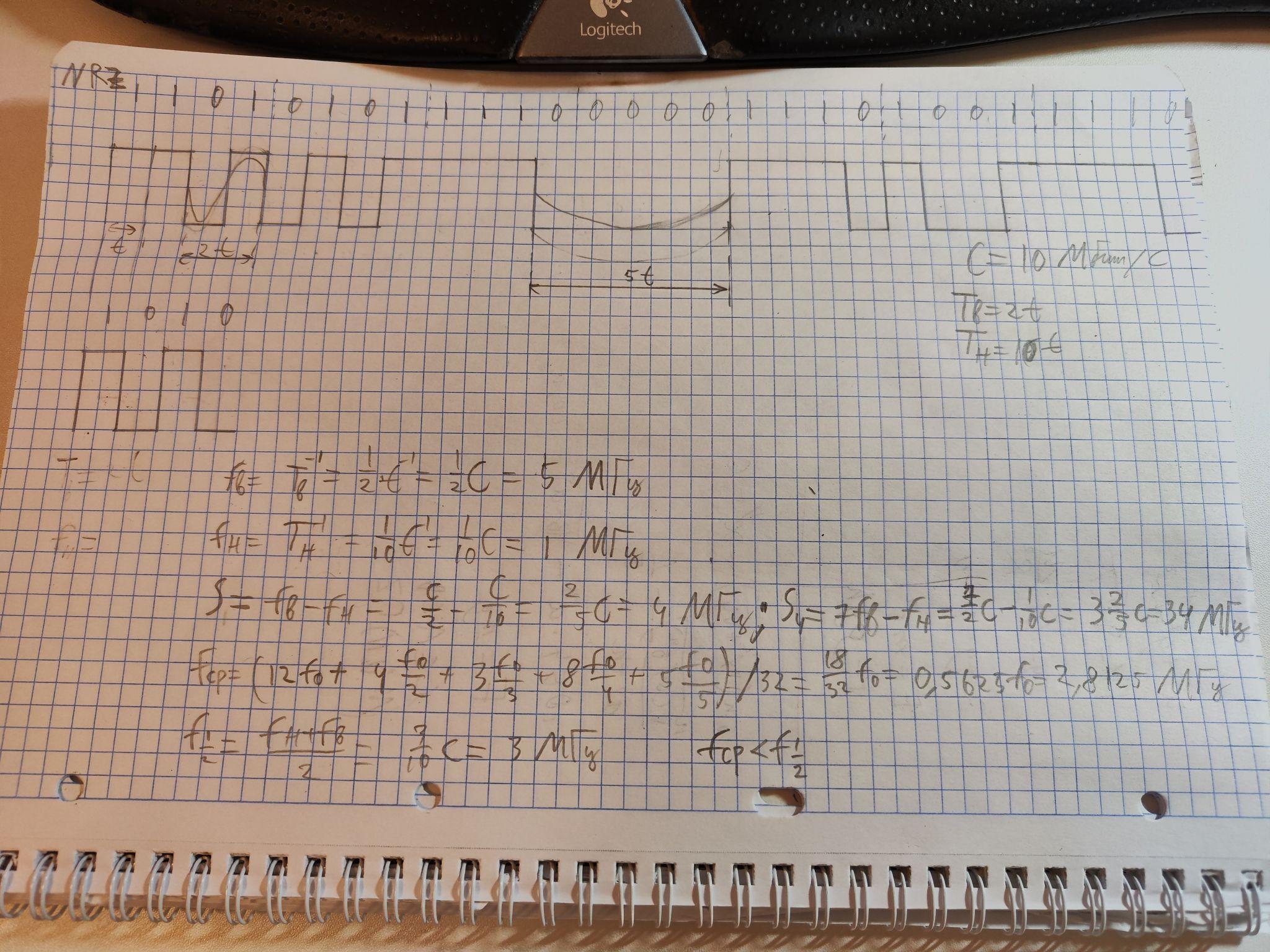
Будем считать, что пропускная способность канала связи для всех методов равна 10 Мбит/с.

## Манчестерский код



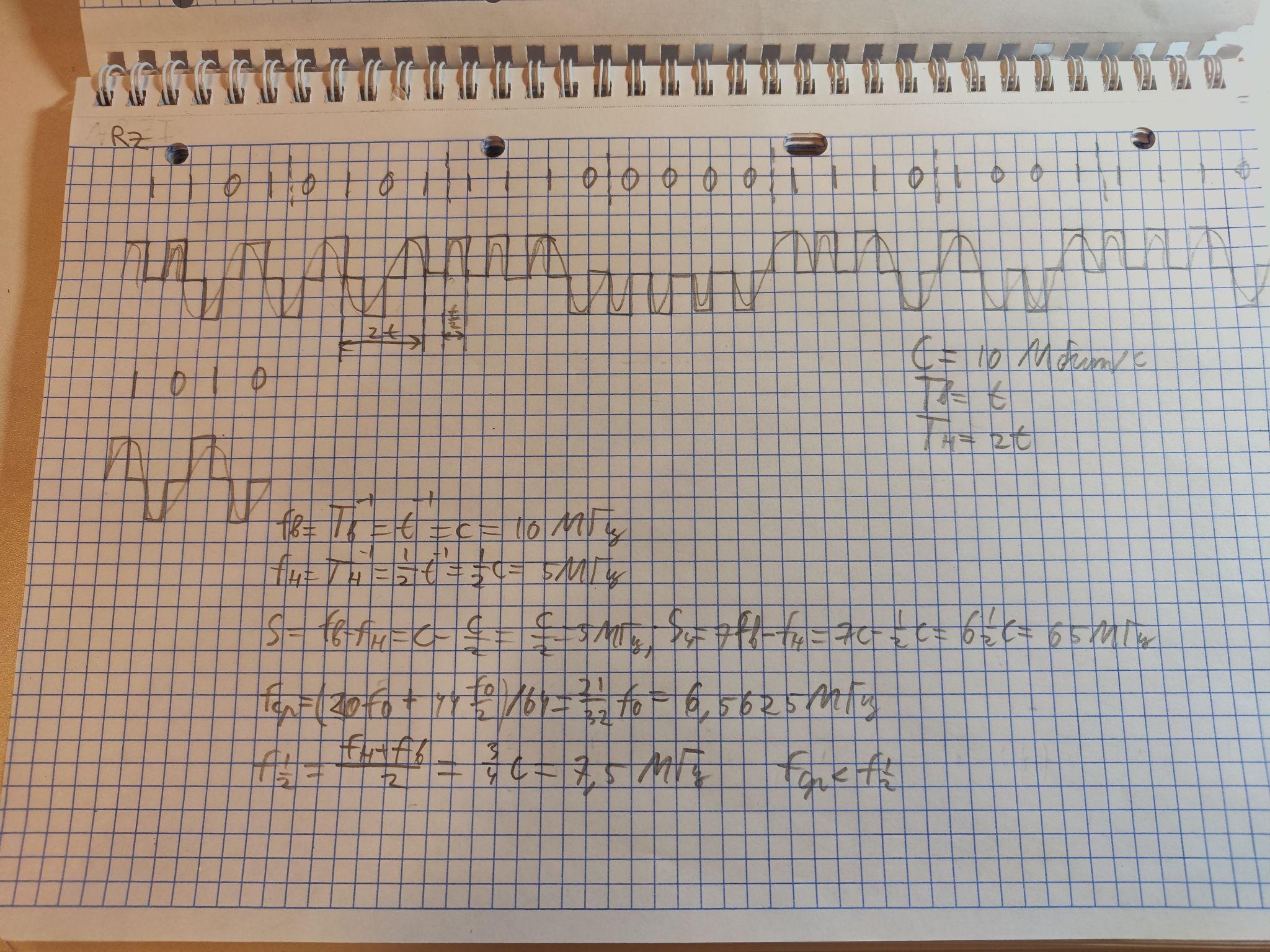
|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Метод | fв (МГц) | fн (МГц) | S (МГц) | S4 (МГц) | F (МГц) | fср (МГц) |
| Манчестер | 10 | 5 | 5 | 65 | 65 | 7.34375 |

## NRZ (Потенциальный код без возврата к нулю)



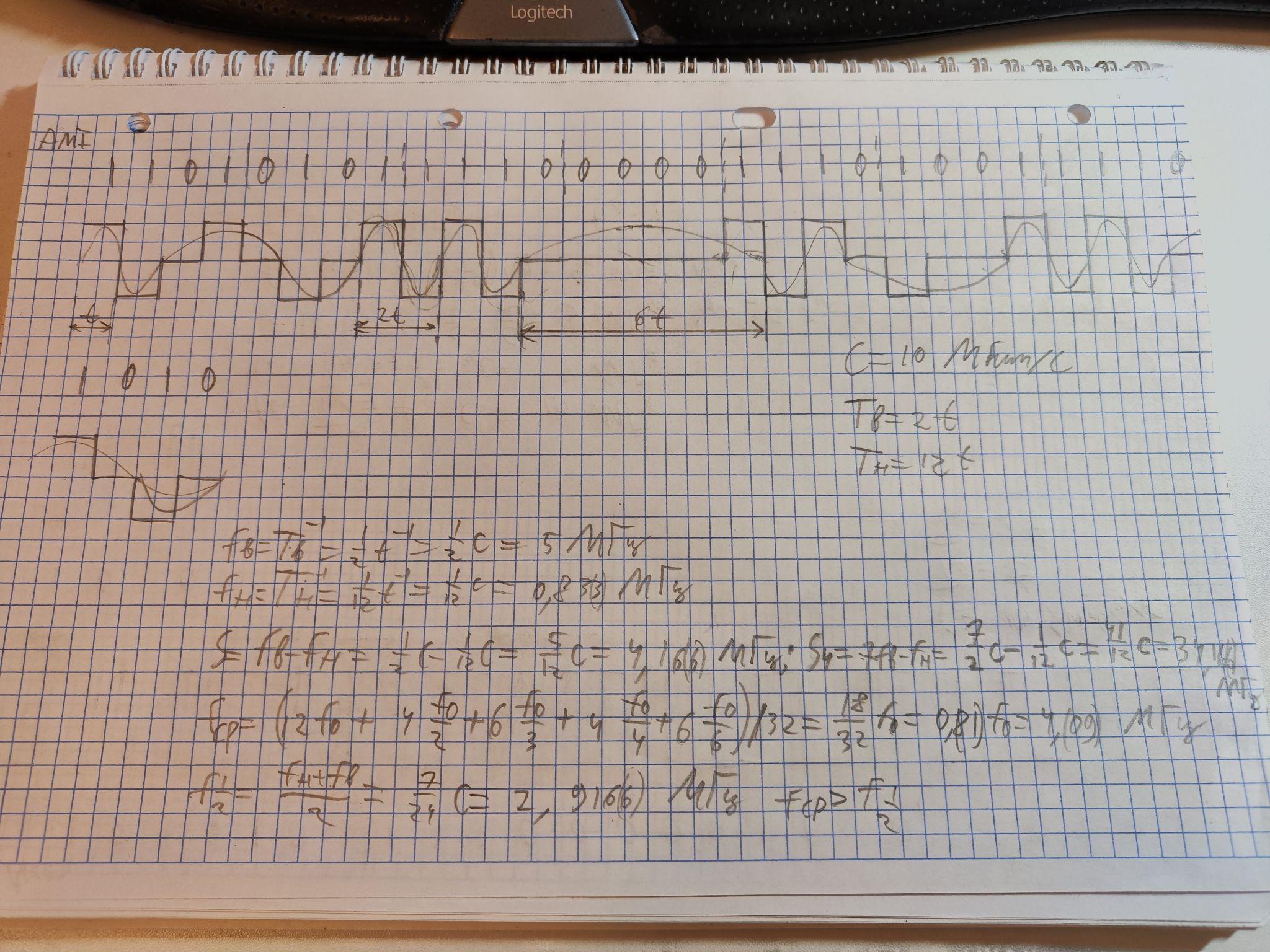
|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Метод | fв (МГц) | fн (МГц) | S (МГц) | S4 (МГц) | F (МГц) | fср (МГц) |
| NRZ | 5 | 1 | 4 | 34 | 35 | 2.8125 |

## RZ (Биполярный импульсный код)



|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Метод | fв (МГц) | fн (МГц) | S (МГц) | S4 (МГц) | F (МГц) | fср (МГц) |
| RZ | 10 | 5 | 5 | 65 | 65 | 6.5625 |

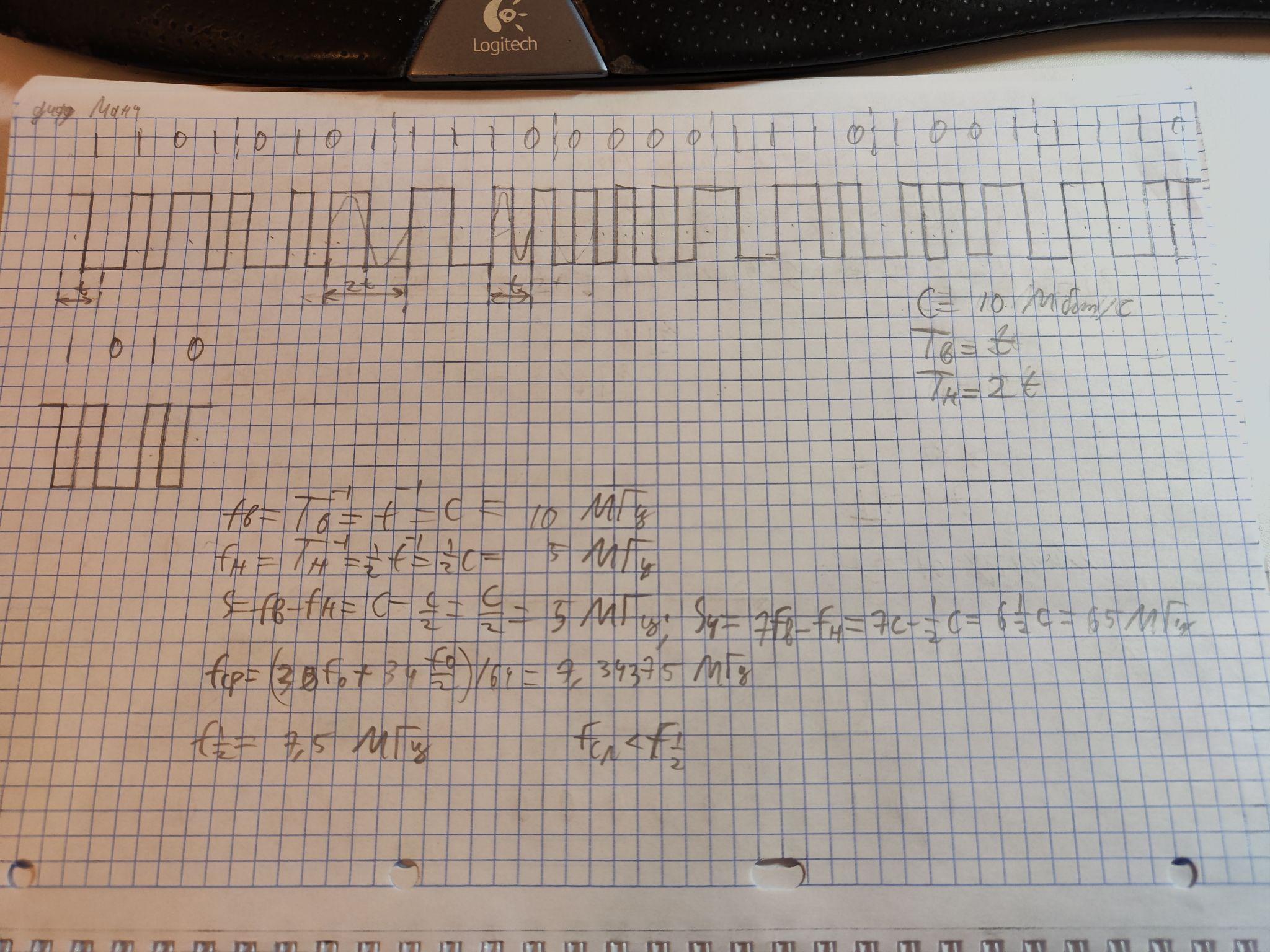
## AMI (Биполярное кодирование с альтернативной инверсией)



|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Метод | fв (МГц) | fн (МГц) | S (МГц) | S4 (МГц) | F (МГц) | fср (МГц) |
| AMI | 5 | 0.8(3) | 4.1(6) | 34.1(6) | 35 | 4.(09) |

Заметим, что в этом методе мы получили наименьшее значение низкой частоты. Это вызвано последовательностью из пяти нулей в сообщении, которая представляется в AMI-методе постоянным сигналом.

## Дифференциальный манчестерский код



|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Метод | fв (МГц) | fн (МГц) | S (МГц) | S4 (МГц) | F (МГц) | fср (МГц) |
| Дифф. Манчестер | 10 | 5 | 5 | 65 | 65 | 7.34375 |

## Итоги

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Метод | fв (МГц) | fн (МГц) | S (МГц) | S4 (МГц) | F (МГц) | fср (МГц) |
| Манчестер | 10 | 5 | 5 | 65 | 65 | 7.34375 |
| NRZ | 5 | 1 | 4 | 34 | 35 | 2.8125 |
| RZ | 10 | 5 | 5 | 65 | 65 | 6.5625 |
| AMI | 5 | 0.8(3) | 4.1(6) | 34.1(6) | 35 | 4.(09) |
| Дифф. Манчестер | 10 | 5 | 5 | 65 | 65 | 7.34375 |

NRZ и AMI обладают наименьшими спектрами ~~(и необходимыми полосами пропускания)~~ (и, соответственно, наименьшими необходимыми полосам пропускания) и преобладают в низких частотах. Один из основных недостатков этих методов это проблема с постоянным уровнем сигнала для длинных последовательностей нулей (для NRZ - ещё и единиц). В данном сообщении присутствует одна длинная последовательность нулей длиной 5 бит, что оставляет значение низкой частоты ~~приемлимым~~ приемлемым - 1 МГц для NRZ и 0.8(3) МГц для AMI.

Как и ожидалось, Манчестер-методы обладают большим спектром, чем NRZ/AMI, но позволяют достичь самосинхронизации без необходимости переходить на 3 уровня сигналов, подобно RZ. ~~Сам метод RZ для данного сообщения обладает спектром, равным манчестерским.~~ Также заметим, что для данного сообщения Манчестер-методы и RZ-метод обладают одинаковым значением спектра.

Для дальнейшей работы выберем NRZ и AMI, ориентируясь на их относительно небольшой спектр. Кроме этого, оба этих метода имеют проблему с длинными последовательностями нулей - посмотрим, поможет ли избыточное кодирование нашему сообщению на их примере.

# Этап 3 - Избыточное кодирование - 4B/5B

Проведём избыточное кодирование методом 4B/5B. Для каждых 4-ёх битов сообщения приведём их «новое» значение, длиной в 5 бит:

|  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 16x | D | 5 | E | 0 | E | 9 | E | A | E | 8 | E |
| 4B | 1101 | 0101 | 1110 | 0000 | 1110 | 1001 | 1110 | 1010 | 1110 | 1000 | 1110 |
| 5B | 11011 | 01011 | 11100 | 11110 | 11100 | 10011 | 11100 | 10110 | 11100 | 10010 | 11100 |
| 16x | D | 2 | 0 | C | E | 2 | E | C | 8 | 2 | E |
| 4B | 1101 | 0010 | 0000 | 1100 | 1110 | 0010 | 1110 | 1100 | 1000 | 0010 | 1110 |
| 5B | 11011 | 10100 | 11110 | 11010 | 11100 | 10100 | 11100 | 11010 | 10010 | 10100 | 11100 |

Итог (x2): 11011 01011 11100 11110 11100 10011 11100 10110 11100 10010 11100 11011 10100 11110 11010 11100 10100 11100 11010 10010 10100 11100

Получившаяся длина: 110 бит (13.75 байт)

Чтобы представить результат в шестнадцатеричной системе, дополним его 2-мя нулями в конце. Тогда получим:

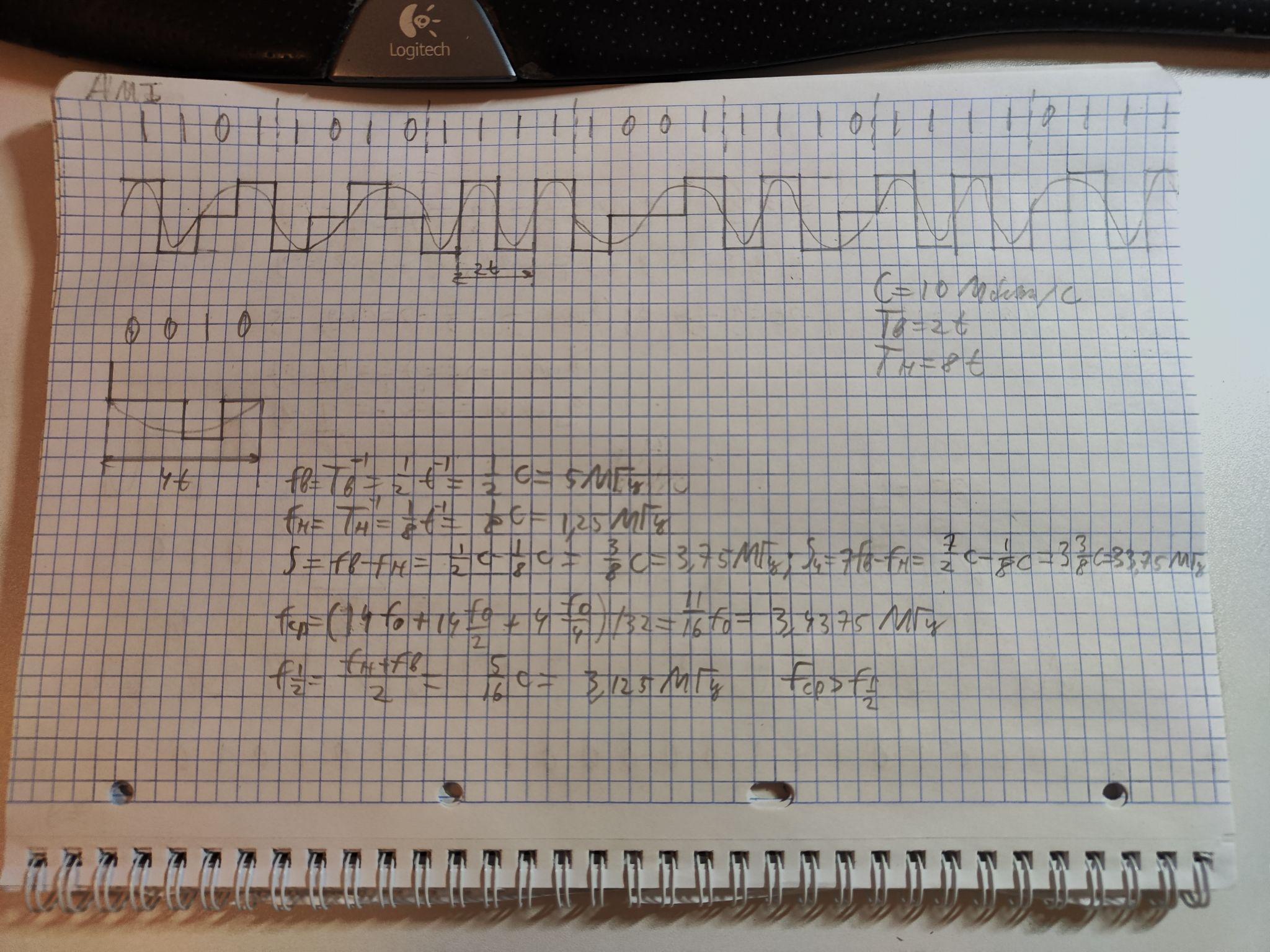
Итог (x16): DA F9 EE 4F 96 E4 B9 BA 7B 5C A7 35 2A 70

Избыточность: 25% (1 “лишний” бит на 4 “полезных”).

Посмотрим, как изменятся характеристики для AMI-метода, после избыточного кодирования.

Будем кодировать первые 4 байта нового сообщения - т.е. 1101 1010 1111 1001 1110 1111 0111 0010

## AMI (Избыточное кодирование - 4B/5B)



|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Сообщение | fв (МГц) | fн (МГц) | S (МГц) | S4 (МГц) | F (МГц) | fср (МГц) |
| Исходное | 5 | 0.8(3) | 4.1(6) | 34.1(6) | 35 | 4.(09) |
| 4B/5B | 5 | 1.25 | 3.75 | 33.75 | 35 | 3.4375 |

Заметим, что самая низкая частота стала выше, что вызвало уменьшение спектра. Необходимая полоса пропускания стала меньше на ~0.4 МГц, что не повлияло на итоговое округлённое значение.

Отметим, что в получившемся сообщении нет последовательностей нулей, длиной больше двух. Это позволяет избавиться от недостатка AMI-метода - постоянных промежутков, можно сказать, нет.

Для NRZ ситуация немного сложнее: в сообщении появились более длинные последовательности единиц. В NRZ-методе такие последовательности будут представляться постоянным значением сигнала, т.е. исправлено ничего не будет.

# Этап 4 - Скремблирование – Bi = Ai ⊕ Bi-5 ⊕ Bi-7

Для скремблирования выберем полином .

Чем больше разности индексов в полиноме, тем меньше реальных преобразований нужно будет делать на этапе скремблирования/дескремблирования (например, в данном полиноме первые 5 бит остаются неизменными, а 2 следующих требуют выполнения только одной XOR-операции). Т.к. Наше сообщение не содержит “проблемных” последовательностей в начальных битах, можем “позволить” себе полином с более высокими индексами.

Исходное сообщение:

1101 0101 1110 0000 1110 1001 1110 1010 1110 1000 1110 1101 0010 0000 1100 1110 0010 1110 1100 1000 0010 1110

## Процесс скремблирования

|  |  |  |  |
| --- | --- | --- | --- |
| Индекс нового бита | Формула нового бита | Расчёт нового бита | Значение нового бита |
| B1 | A1 | 1 | 1 |
| B2 | A2 | 1 | 1 |
| B3 | A3 | 0 | 0 |
| B4 | A4 | 1 | 1 |
| B5 | A5 | 0 | 0 |
| B6 | A6 ⊕ B1 | 1 ⊕ 1 | 0 |
| B7 | A7 ⊕ B2 | 0 ⊕ 1 | 1 |
| B8 | A8 ⊕ B3 ⊕ B1 | 1 ⊕ 0 ⊕ 1 | 0 |
| B9 | A9 ⊕ B4 ⊕ B2 | 1 ⊕ 1 ⊕ 1 | 1 |
| B10 | A10 ⊕ B5 ⊕ B3 | 1 ⊕ 0 ⊕ 0 | 1 |
| B11 | A11 ⊕ B6 ⊕ B4 | 1 ⊕ 0 ⊕ 1 | 0 |
| B12 | A12 ⊕ B7 ⊕ B5 | 0 ⊕ 1 ⊕ 0 | 1 |
| B13 | A13 ⊕ B8 ⊕ B6 | 0 ⊕ 0 ⊕ 0 | 0 |
| B14 | A14 ⊕ B9 ⊕ B7 | 0 ⊕ 1 ⊕ 1 | 0 |
| B15 | A15 ⊕ B10 ⊕ B8 | 0 ⊕ 1 ⊕ 0 | 1 |
| B16 | A16 ⊕ B11 ⊕ B9 | 0 ⊕ 0 ⊕ 1 | 1 |
| B17 | A17 ⊕ B12 ⊕ B10 | 1 ⊕ 1 ⊕ 1 | 1 |
| B18 | A18 ⊕ B13 ⊕ B11 | 1 ⊕ 0 ⊕ 0 | 1 |
| B19 | A19 ⊕ B14 ⊕ B12 | 1 ⊕ 0 ⊕ 1 | 0 |
| B20 | A20 ⊕ B15 ⊕ B13 | 0 ⊕ 1 ⊕ 0 | 1 |
| B21 | A21 ⊕ B16 ⊕ B14 | 1 ⊕ 1 ⊕ 0 | 0 |
| B22 | A22 ⊕ B17 ⊕ B15 | 0 ⊕ 1 ⊕ 1 | 0 |
| B23 | A23 ⊕ B18 ⊕ B16 | 0 ⊕ 1 ⊕ 1 | 0 |
| B24 | A24 ⊕ B19 ⊕ B17 | 1 ⊕ 0 ⊕ 1 | 0 |
| B25 | A25 ⊕ B20 ⊕ B18 | 1 ⊕ 1 ⊕ 1 | 1 |
| B26 | A26 ⊕ B21 ⊕ B19 | 1 ⊕ 0 ⊕ 0 | 1 |
| B27 | A27 ⊕ B22 ⊕ B20 | 1 ⊕ 0 ⊕ 1 | 0 |
| B28 | A28 ⊕ B23 ⊕ B21 | 0 ⊕ 0 ⊕ 0 | 0 |
| B29 | A29 ⊕ B24 ⊕ B22 | 1 ⊕ 0 ⊕ 0 | 1 |
| B30 | A30 ⊕ B25 ⊕ B23 | 0 ⊕ 1 ⊕ 0 | 1 |
| B31 | A31 ⊕ B26 ⊕ B24 | 1 ⊕ 1 ⊕ 0 | 0 |
| B32 = | A32 ⊕ B27 ⊕ B25 | 0 ⊕ 0 ⊕ 1 | 1 |
| B33 = | A33 ⊕ B28 ⊕ B26 | 1 ⊕ 0 ⊕ 1 | 0 |
| B34 = | A34 ⊕ B29 ⊕ B27 | 1 ⊕ 1 ⊕ 0 | 0 |
| B35 | A35 ⊕ B30 ⊕ B28 | 1 ⊕ 1 ⊕ 0 | 0 |
| B36 | A36 ⊕ B31 ⊕ B29 | 0 ⊕ 0 ⊕ 1 | 1 |
| B37 | A37 ⊕ B32 ⊕ B30 | 1 ⊕ 1 ⊕ 1 | 1 |
| B38 | A38 ⊕ B33 ⊕ B31 | 0 ⊕ 0 ⊕ 0 | 0 |
| B39 | A39 ⊕ B34 ⊕ B32 | 0 ⊕ 0 ⊕ 1 | 1 |
| B40 | A40 ⊕ B35 ⊕ B33 | 0 ⊕ 0 ⊕ 0 | 0 |
| B41 | A41 ⊕ B36 ⊕ B34 | 1 ⊕ 1 ⊕ 0 | 0 |
| B42 | A42 ⊕ B37 ⊕ B35 | 1 ⊕ 1 ⊕ 0 | 0 |
| B43 | A43 ⊕ B38 ⊕ B36 | 1 ⊕ 0 ⊕ 1 | 0 |
| B44 | A44 ⊕ B39 ⊕ B37 | 0 ⊕ 1 ⊕ 1 | 0 |
| B45 | A45 ⊕ B40 ⊕ B38 | 1 ⊕ 0 ⊕ 0 | 1 |
| B46 | A46 ⊕ B41 ⊕ B39 | 1 ⊕ 0 ⊕ 1 | 0 |
| B47 | A47 ⊕ B42 ⊕ B40 | 0 ⊕ 0 ⊕ 0 | 0 |
| B48 | A48 ⊕ B43 ⊕ B41 | 1 ⊕ 0 ⊕ 0 | 1 |
| B49 | A49 ⊕ B44 ⊕ B42 | 0 ⊕ 0 ⊕ 0 | 0 |
| B50 | A50 ⊕ B45 ⊕ B43 | 0 ⊕ 1 ⊕ 0 | 1 |
| B51 | A51 ⊕ B46 ⊕ B44 | 1 ⊕ 0 ⊕ 0 | 1 |
| B52 | A52 ⊕ B47 ⊕ B45 | 0 ⊕ 0 ⊕ 1 | 1 |
| B53 | A53 ⊕ B48 ⊕ B46 | 0 ⊕ 1 ⊕ 0 | 1 |
| B54 | A54 ⊕ B49 ⊕ B47 | 0 ⊕ 0 ⊕ 0 | 0 |
| B55 | A55 ⊕ B50 ⊕ B48 | 0 ⊕ 1 ⊕ 1 | 0 |
| B56 | A56 ⊕ B51 ⊕ B49 | 0 ⊕ 1 ⊕ 0 | 1 |
| B57 | A57 ⊕ B52 ⊕ B50 | 1 ⊕ 1 ⊕ 1 | 1 |
| B58 | A58 ⊕ B53 ⊕ B51 | 1 ⊕ 1 ⊕ 1 | 1 |
| B59 | A59 ⊕ B54 ⊕ B52 | 0 ⊕ 0 ⊕ 1 | 1 |
| B60 | A60 ⊕ B55 ⊕ B53 | 0 ⊕ 0 ⊕ 1 | 1 |
| B61 | A61 ⊕ B56 ⊕ B54 | 1 ⊕ 1 ⊕ 0 | 0 |
| B62 | A62 ⊕ B57 ⊕ B55 | 1 ⊕ 1 ⊕ 0 | 0 |
| B63 | A63 ⊕ B58 ⊕ B56 | 1 ⊕ 1 ⊕ 1 | 1 |
| B64 | A64 ⊕ B59 ⊕ B57 | 0 ⊕ 1 ⊕ 1 | 0 |
| B65 | A65 ⊕ B60 ⊕ B58 | 0 ⊕ 1 ⊕ 1 | 0 |
| B66 | A66 ⊕ B61 ⊕ B59 | 0 ⊕ 0 ⊕ 1 | 1 |
| B67 | A67 ⊕ B62 ⊕ B60 | 1 ⊕ 0 ⊕ 1 | 0 |
| B68 | A68 ⊕ B63 ⊕ B61 | 0 ⊕ 1 ⊕ 0 | 1 |
| B69 | A69 ⊕ B64 ⊕ B62 | 1 ⊕ 0 ⊕ 0 | 1 |
| B70 | A70 ⊕ B65 ⊕ B63 | 1 ⊕ 0 ⊕ 1 | 0 |
| B71 | A71 ⊕ B66 ⊕ B64 | 1 ⊕ 1 ⊕ 0 | 0 |
| B72 | A72 ⊕ B67 ⊕ B65 | 0 ⊕ 0 ⊕ 0 | 0 |
| B73 | A73 ⊕ B68 ⊕ B66 | 1 ⊕ 1 ⊕ 1 | 1 |
| B74 | A74 ⊕ B69 ⊕ B67 | 1 ⊕ 1 ⊕ 0 | 0 |
| B75 | A75 ⊕ B70 ⊕ B68 | 0 ⊕ 0 ⊕ 1 | 1 |
| B76 | A76 ⊕ B71 ⊕ B69 | 0 ⊕ 0 ⊕ 1 | 1 |
| B77 | A77 ⊕ B72 ⊕ B70 | 1 ⊕ 0 ⊕ 0 | 1 |
| B78 | A78 ⊕ B73 ⊕ B71 | 0 ⊕ 1 ⊕ 0 | 1 |
| B79 | A79 ⊕ B74 ⊕ B72 | 0 ⊕ 0 ⊕ 0 | 0 |
| B80 | A80 ⊕ B75 ⊕ B73 | 0 ⊕ 1 ⊕ 1 | 0 |
| B81 | A81 ⊕ B76 ⊕ B74 | 0 ⊕ 1 ⊕ 0 | 1 |
| B82 | A82 ⊕ B77 ⊕ B75 | 0 ⊕ 1 ⊕ 1 | 0 |
| B83 | A83 ⊕ B78 ⊕ B76 | 1 ⊕ 1 ⊕ 1 | 1 |
| B84 | A84 ⊕ B79 ⊕ B77 | 0 ⊕ 0 ⊕ 1 | 1 |
| B85 | A85 ⊕ B80 ⊕ B78 | 1 ⊕ 0 ⊕ 1 | 0 |
| B86 | A86 ⊕ B81 ⊕ B79 | 1 ⊕ 1 ⊕ 0 | 0 |
| B87 | A87 ⊕ B82 ⊕ B80 | 1 ⊕ 0 ⊕ 0 | 1 |
| B88 | A88 ⊕ B83 ⊕ B81 | 0 ⊕ 1 ⊕ 1 | 0 |

## Результат скремблирования

После скремблирования получаем следующее сообщение:

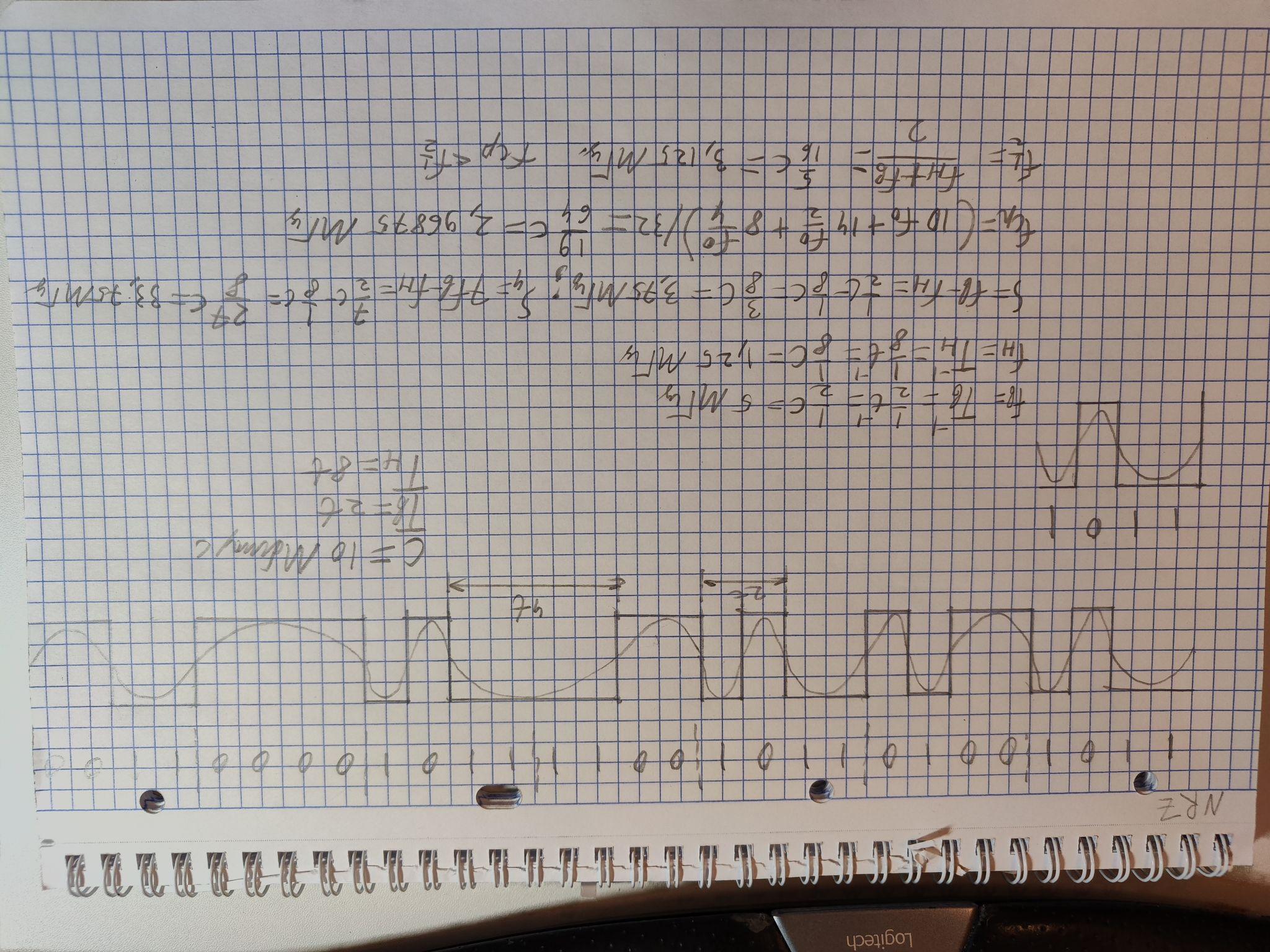
1101 0010 1101 0011 1101 0000 1100 1101 0001 1010 0000 1001 0111 1001 1111 0010 0101 1000 1011 1100 1011 0010

В 16-ной системе:

D2 D3 D0 CD 1A 09 79 F2 58 BC B2

Посмотрим, как изменятся характеристики для NRZ-метода, после скремблирования сообщения:

## NRZ (Скремблирование - Bi = Ai ⊕ Bi-5 ⊕ Bi-7)



|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Сообщение | fв (МГц) | fн (МГц) | S (МГц) | S4 (МГц) | F (МГц) | fср (МГц) |
| Исходное | 5 | 1 | 4 | 34 | 35 | 2.8125 |
| После скремблирования | 5 | 1.25 | 3.75 | 33.75 | 35 | 2.96875 |

Наблюдаем увеличение самой низкой частоты - подобно результату избыточного кодирования для AMI-метода.

Стоит отметить, что для рассматриваемых 4-ёх байт была удалена последовательность из 5 одинаковых бит (нулей). Однако, если рассматривать всё сообщение после скремблирования, то можно обнаружить новую последовательность из 5 нулей - просто так получилось, что она находится на байтах с бОльшим порядковым номером. Это подтверждает, что скремблирование, вообще говоря, не гарантирует, что любое сообщение сможет избавиться от длинных последовательностей нулей/единиц.

# Этап 5 - Сравнительный анализ результатов

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| Метод | Сообщение | fв (МГц) | fн (МГц) | S (МГц) | S4 (МГц) | F (МГц) | fср (МГц) |
| Манчестер | Исходное | 10 | 5 | 5 | 65 | 65 | 7.34375 |
| Дифф. Манчестер | Исходное | 10 | 5 | 5 | 65 | 65 | 7.34375 |
| RZ | Исходное | 10 | 5 | 5 | 65 | 65 | 6.5625 |
| AMI | Исходное | 5 | 0.8(3) | 4.1(6) | 34.1(6) | 35 | 4.(09) |
| AMI | 4B/5B | 5 | 1.25 | 3.75 | 33.75 | 35 | 3.4375 |
| NRZ | Исходное | 5 | 1 | 4 | 34 | 35 | 2.8125 |
| NRZ | После скремблирования | 5 | 1.25 | 3.75 | 33.75 | 35 | 2.96875 |

Методы AMI и NRZ обладают наименьшим спектром среди рассмотренных методов. Использование избыточного кодирования (4B/5B) и скремблирования (Bi = Ai ⊕ Bi-5 ⊕ Bi-7) соответственно позволяет снизить это значение ещё больше. Наименьший спектр означает наименьшую необходимую полосу пропускания. ~~Кроме этого, меньший спектр означает большую скорость передачи (для постоянного значения пропускной способности канала).~~

В то же время, использование избыточного кодирования/скремблирования не “бесплатно” - избыточное кодирование требует передачи “лишних” (избыточных) битов, а скремблирование требует дополнительной работы передатчика/приёмника (необходимость проводить само скремблирование и дескремблирование сообщения).

Кроме того, NRZ не обладает способностью к самосинхронизации, а AMI ещё и требует использования 3-ёх уровней сигнала. Манчестер-коды не обладают подобными проблемами, но обладают большим спектром и, соответственно, большей необходимой полосой пропускания.

Подводя итоги, я бы посчитал наиболее успешной комбинацией использование AMI с 4B/5B избыточным кодированием. Это позволяет достичь минимального спектра, находить некоторые ошибочные сигналы (благодаря правилу о смене полярности сигнала в AMI) и гарантирует отсутствие длинных промежутков с постоянным значением сигнала. Из минусов стоит отметить необходимость использования 3-ёх уровней сигнала и отсутствие самосинхронизации.