Лабараторная работа №6

Обнаружение и исправление однократной ошибки в сообщении

Цель работы: изучить основные принципы помехоустойчивого кодирования; изучить способ определение позиции и значения корректирующих бит кода Хемминга; получить практические навыки построения кода Хемминга, корректирующего однократные ошибки; изучить способ построения линейногруппового кода и возможность коррекции однократной ошибки с помощью линейно-группового кода.

Задания части №1

Выполнить кодирование текстового сообщения М1 по буквам, используя русский или латинский алфавит, размер сообщения не менее 4 букв. Определить размер n в битах закодированного сообщения М. Например, в качестве кода можно использовать порядковый номер буквы в алфавите. Если М1 = “АБ”, то М = 000001000010 и размер сообщения n = 12.

2. Определить количество k контрольных разрядов кода Хемминга, необходимых для кодирования сообщения М размер n бит.

3. Определить позиции и значения k контрольных разрядов кода Хемминга: двумя способами:

1) подсчёт количества единиц в контролируемых контрольным битом разрядах сообщения;

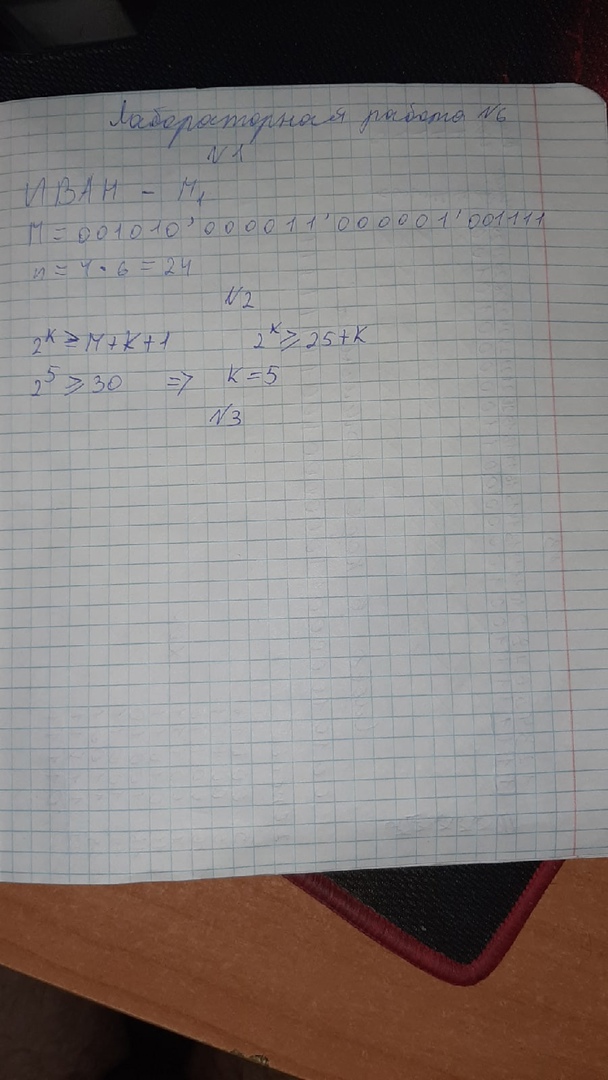
2) использование двоичного представления номеров разрядов сообщения.

4. Записать полученное сообщение размера (n +k) в коде Хемминга.

5. Смоделировать коррекцию ошибки: внести однократную, двукратную и k-кратную ошибки в произвольные биты сообщения и найти эти ошибки с помощью кода Хемминга, используя:

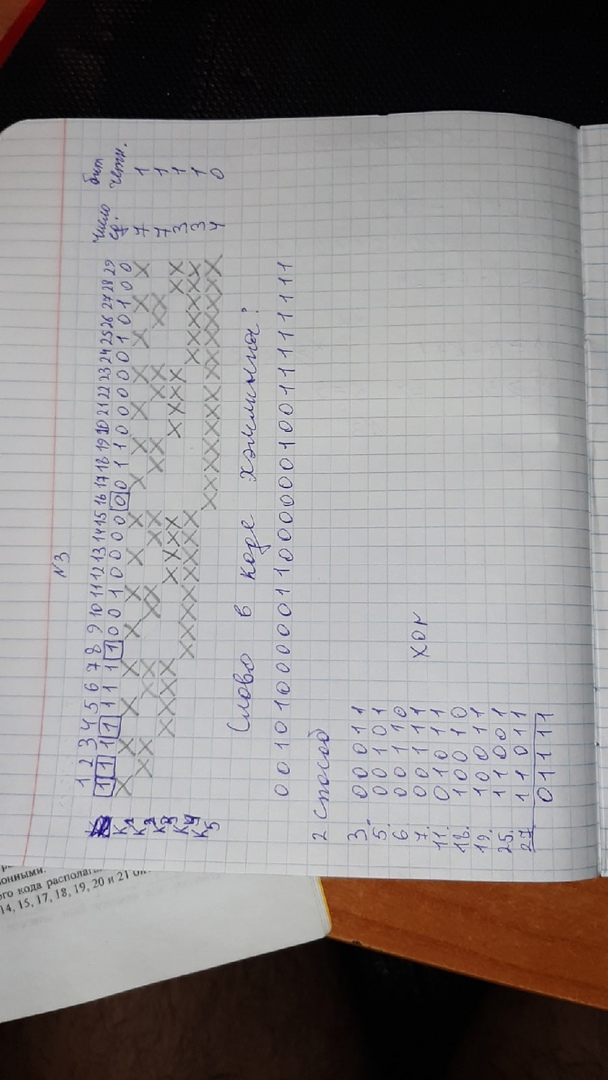
1) подсчёт количества единиц в контролируемых контрольным битом разрядах сообщения;

2) двоичное представление номеров разрядов сообщения.

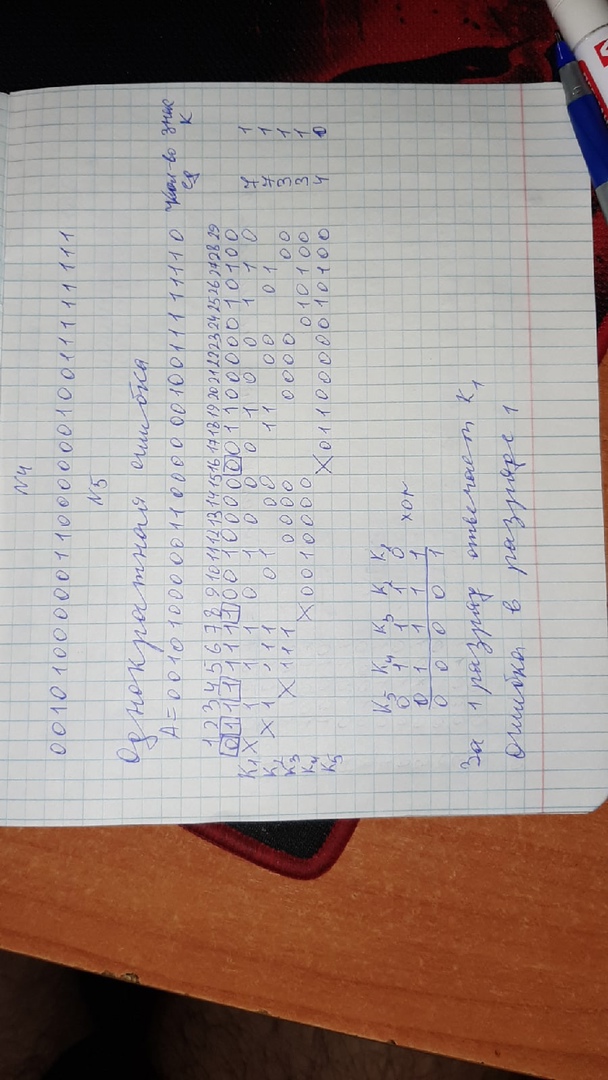


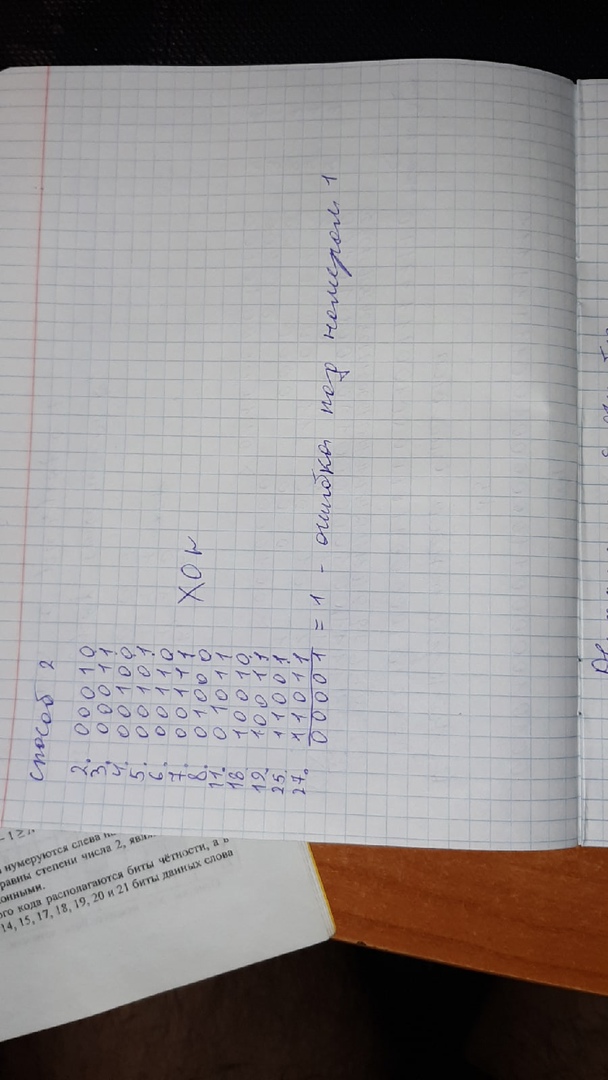
Задание№1 перевод слова в двоичный код

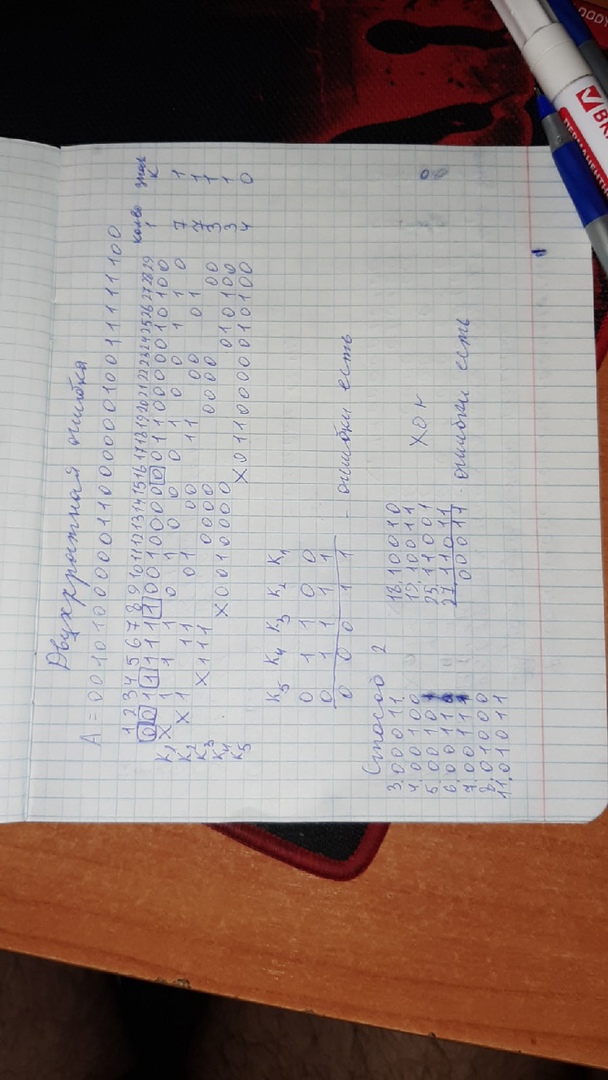
Задание№2 по формуле 2 𝑘 ≥ 𝑀 + 𝑘 + 1 определил К- количество контрольных разрядов

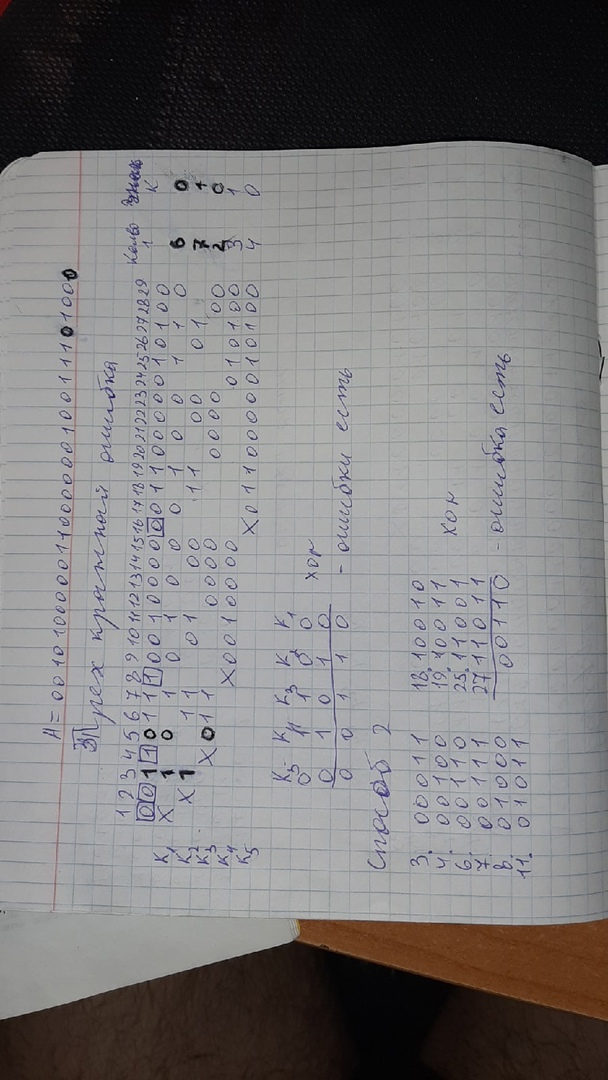


Задание№3 построил таблицу с кодом Хемминга, каждую степень двойки обозначил 0, расставил контрольные разряды (к1 – один разряд через разряд, к2 – два разряда через два и тд) и по количеству единиц узнал биты контрольных разрядов (колво единиц четное – бит 0). Далее построил таблицу со значениями разрядов в которых есть 1 и перевел их в двоичный код, далее применил операцию xor к каждому столбцу и получил те же контрольные разряды (в обратном порядке)









Задание№5

1)смоделировал однократную и обнаружил ее

2) смоделировал двукратную ошибку

3) смоделировал 3 кратную ошибку

Задание части №2

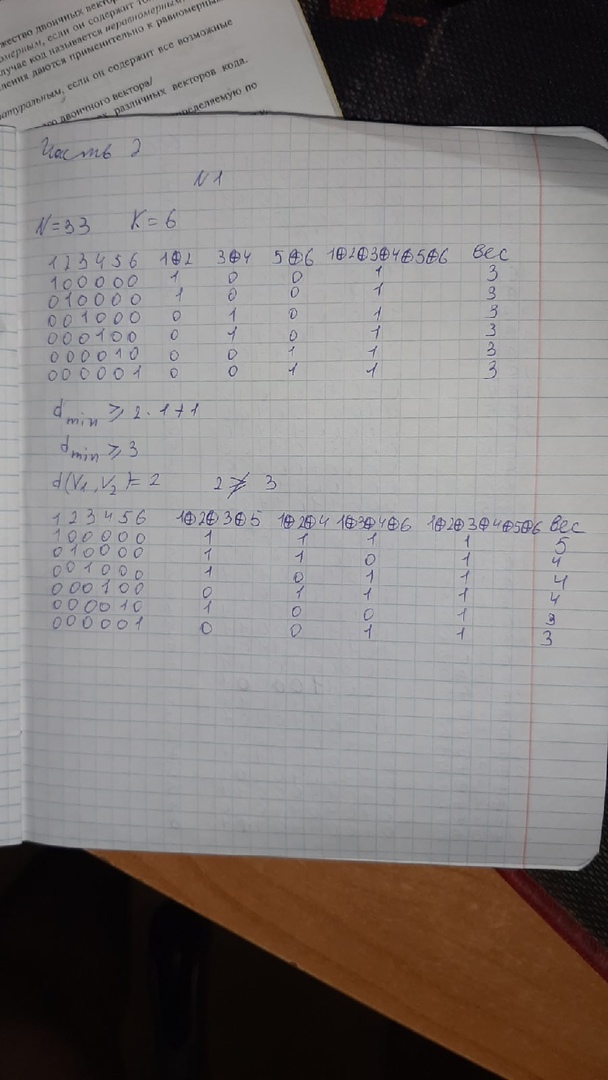
Выполнить построение порождающей матрицы G линейно-группового кода, необходимой для кодирования сообщения М1 по буквам. Определить необходимое число информационных и проверочных столбцов матрицы G.

Вычислить значение проверочных столбцов и доказать соответствие полученной порождающей матрицы G требованиям.

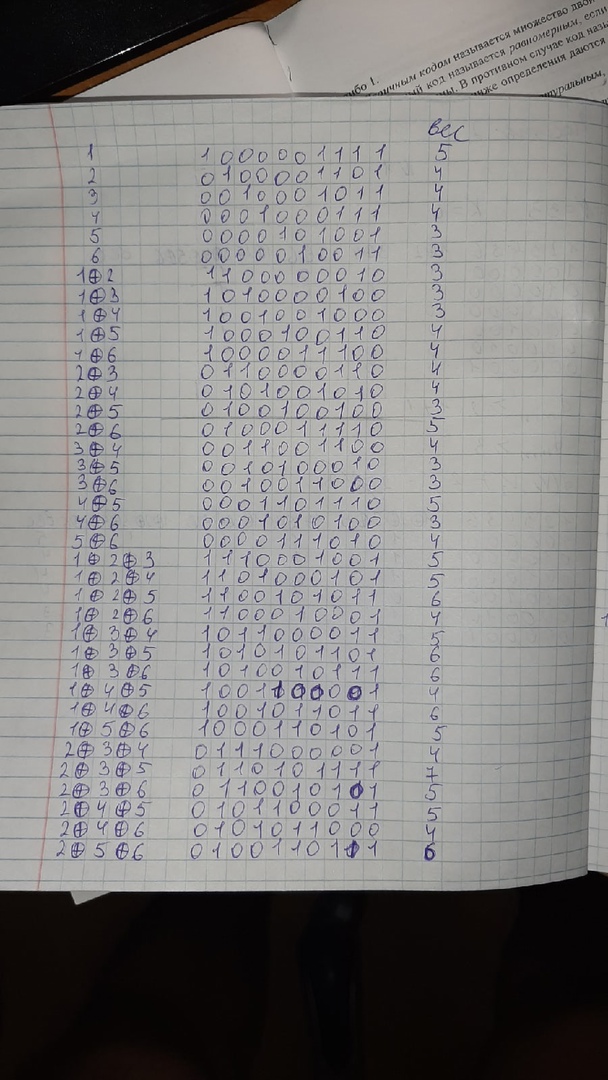
2. Выполнить кодирование сообщения М1 по буквам с помощью порождающей матрицы G.

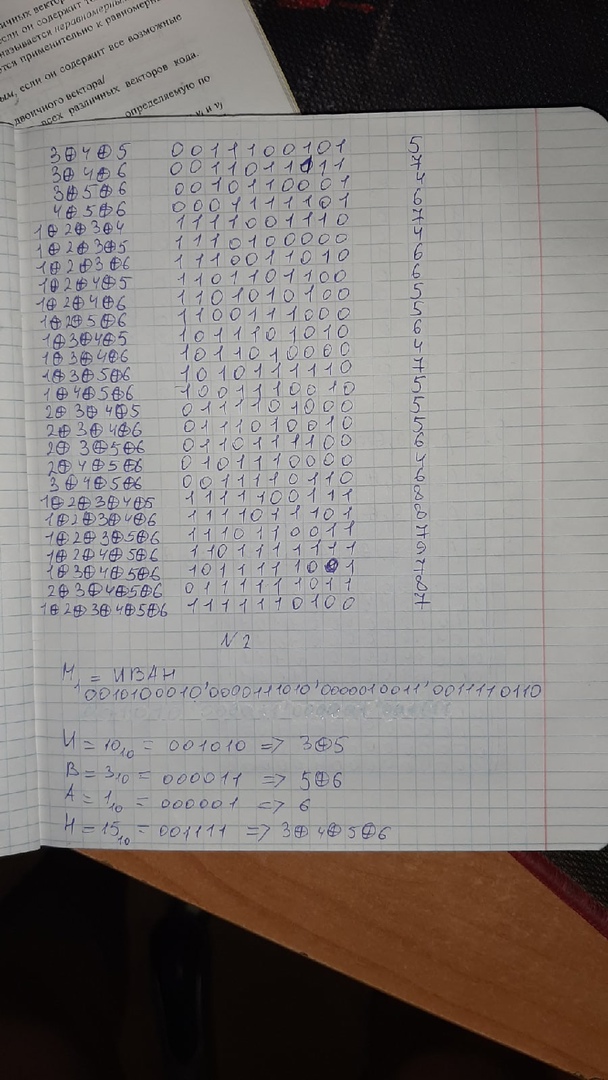
3. Смоделировать коррекцию ошибки: внести в линейно-групповой код одной из букв сообщения М1 однократную ошибку, выполнить проверку сообщения на наличие ошибки и найти бит с ошибкой в сообщении.

Провести аналогичную проверку для двукратной ошибки.

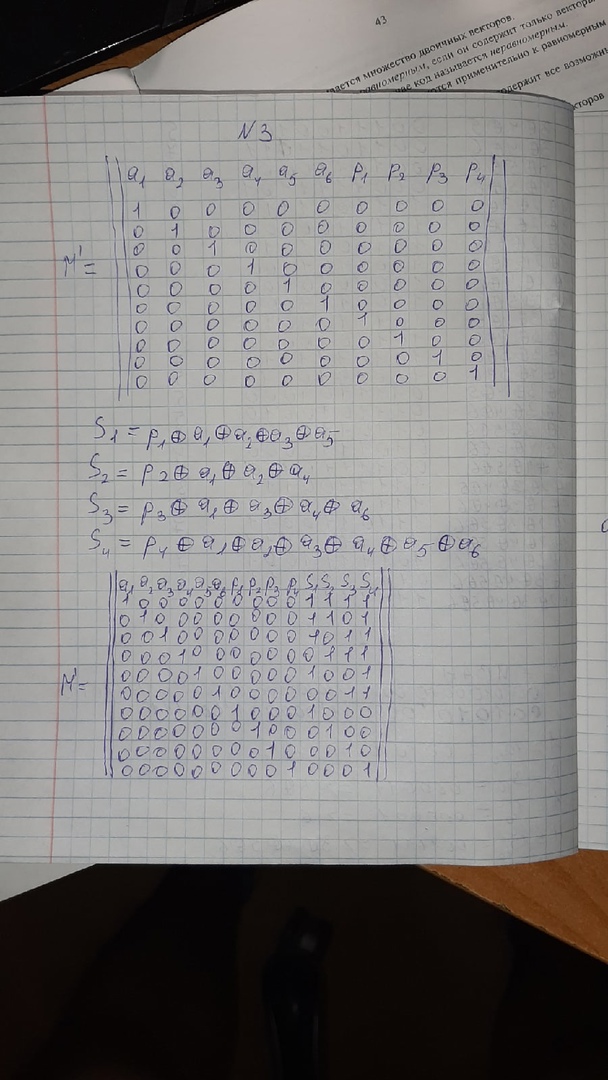


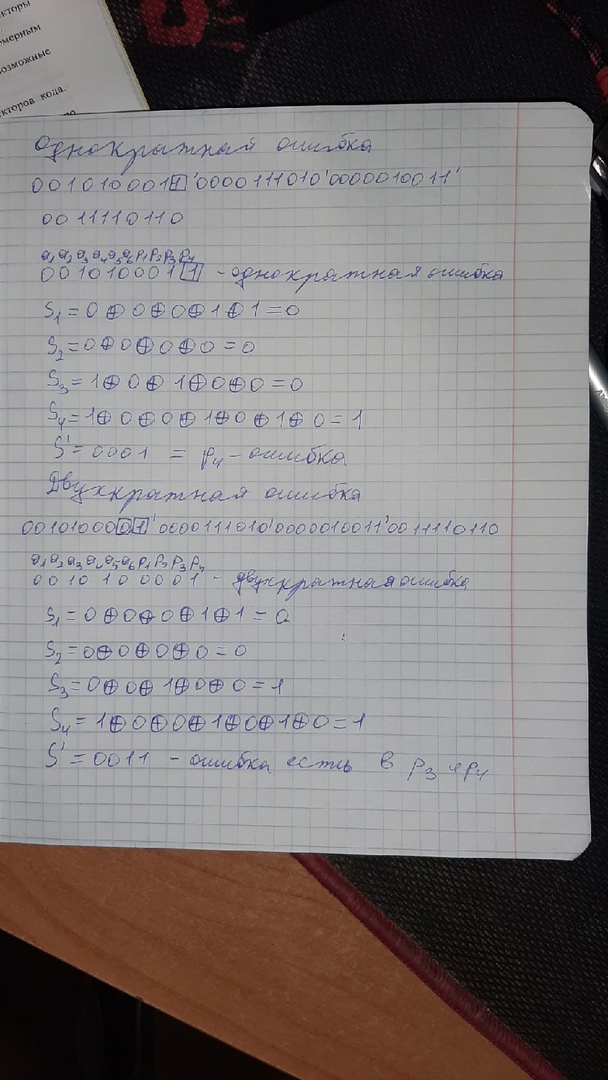
1)Построил порождающую матрицу применим операцию xor к строкам квадратной матрицы, но операцию применял так что бы колво единиц в проверочной части матрицы было больше или равно минимального веса(3) и выполнялось условие по расстоянию(что бы вектора были различны не менее чем в 3 разрядах)





2) проверил вес векторов сделав сумму по модулю 2, и получил так что везде вес ≥ 3 и выполнили кодировку букв с помощью порождающей матрицы





3) построил квадратную матрицу с добавление проверочных разрядов p1,p2,p3,p4,p5 и нашел вектора синдромы (смотрел на вектора к которым применял xor в порождающей матрице и добавил еще p1 xor)

Нахождение однократной ошибки

1)Выписываем букву в которой допущена ошибка, из ее чисел находим вектора синдромы, каждый вектор синдром как бы за координату и находим местоположение ошибки и теперь в матрице ищем строку S содержащую эти координаты(в моем случае p4)

Нахождение двукратной ошибки

1)строим матрицу со всеми возможными комбинациями пар единиц(a1-a2,a1- a3 и тд) так же находим все векторы синдромы

