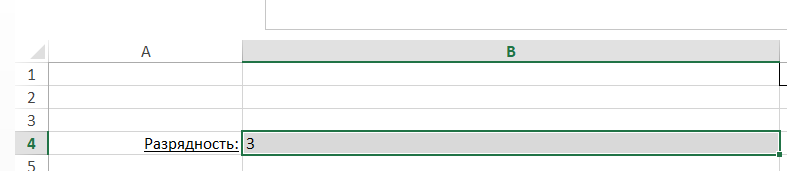
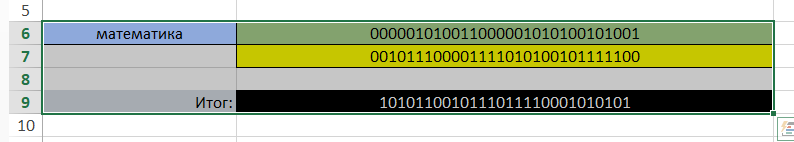
**ШИФР ВЕРНАМА**

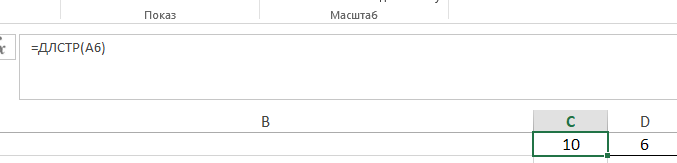
Чтобы выполнить процесс шифрования сначала подберем оптимальную минимальную разрядность кода для каждого уникального символа. Это выполняется с помощью неравенства Макмиллана. Выделяю поле для ввода пользователем разрядности.

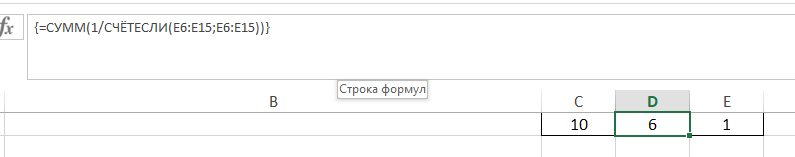


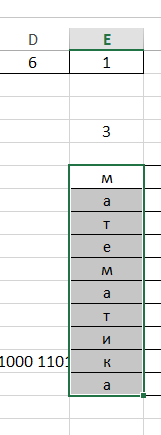
Выделяю поле для ввода открытого текста в 10 символов. Также зеленое поле, в котором будет зашифрованный текст, желтое – для ключа, и черное для итогового закрытого текста.



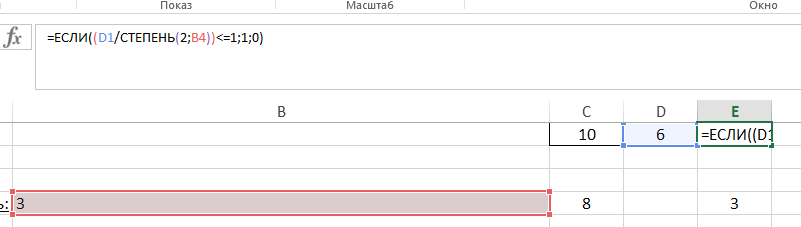
В ячейку C1 вписываю формулу =ДЛСТР(A6), чтобы найти размер открытого текста, в ячейку D1 вписываю формулу {=СУММ(1/СЧЁТЕСЛИ(E6:E15;E6:E15))}, чтобы найти количество уникальных символов в открытом тексте. В диапазоне E6:E15 находится массив символов открытого текста, который я получил с помощью функции ПСТР.



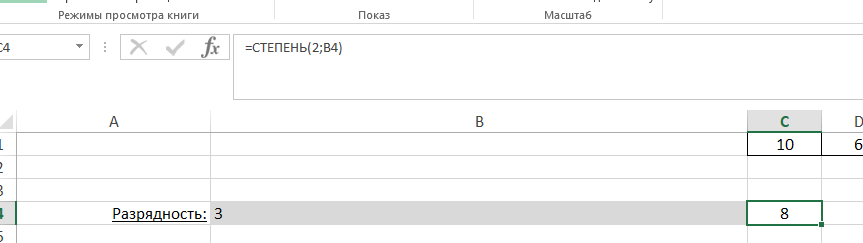




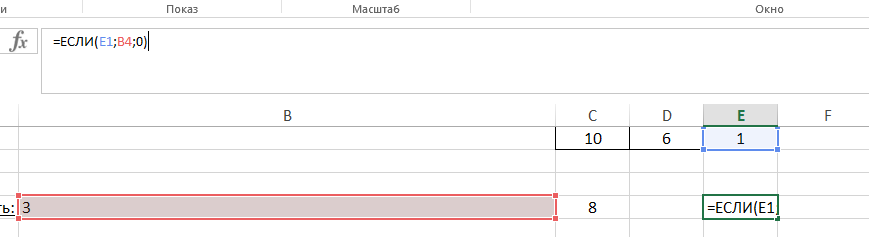
Теперь, имея разрядность, заданную пользователем, и количество уникальных символов проверяем эту разрядность на истинность с помощью неравенства Макмиллана и функции =ЕСЛИ((D1/СТЕПЕНЬ(2;B4))<=1;1;0) в ячейке E1.



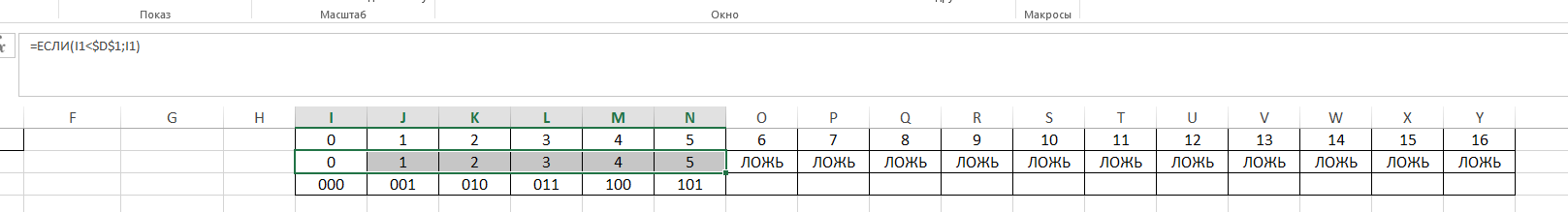
В ячейке C4 нахожу количество комбинаций битов с помощью формулы =СТЕПЕНЬ(2;B4).



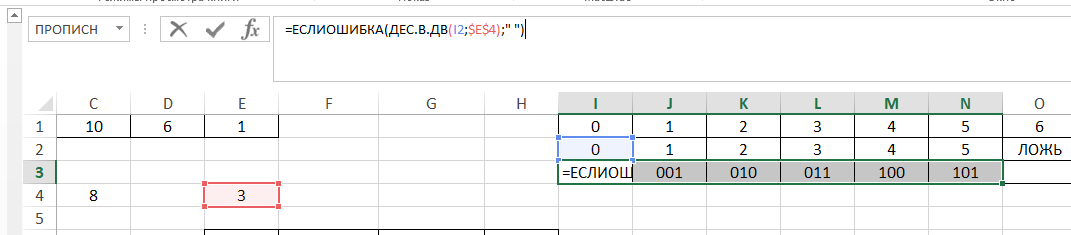
Теперь сделаем еще одну проверку рядом в ячейке E4 на то, что действительно разрядность, которую задал пользователь, подходит для кодирования текста.



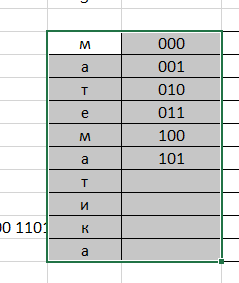
Теперь просто занимаю 16 ячеек для нумерации от 1 до 16. Чтобы каждый УНИКАЛЬНЫЙ символ имел свой код, выполняю функцию =ЕСЛИ(I1<$D$1;I1) в одной из ячеек и далее для всех ячеек последовательно. В моем случае, если количество уникальных символов будет 6, то и ячеек, в которых нет «ЛОЖЬ» будет 6, как раз-таки из-за этой проверки мы исключаем лишние вариации кодов.



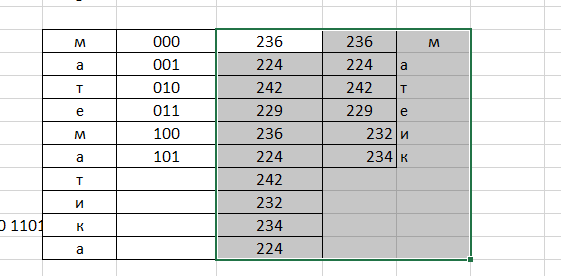
Потом просто переводим получившееся число в ячейках I2:Y2 в двоичный код с помощью функции =ЕСЛИОШИБКА(ДЕС.В.ДВ(I2;$E$4);" "). ЕСЛИОШИБКА исключает возникшие ошибки, ДЕС.В.ДВ переводит ячейки в двоичный код с разрядностью в ячейке E4.



Получив коды для уникальных символов, перечисляю их в столбец, рядом с буквами открытого текста.

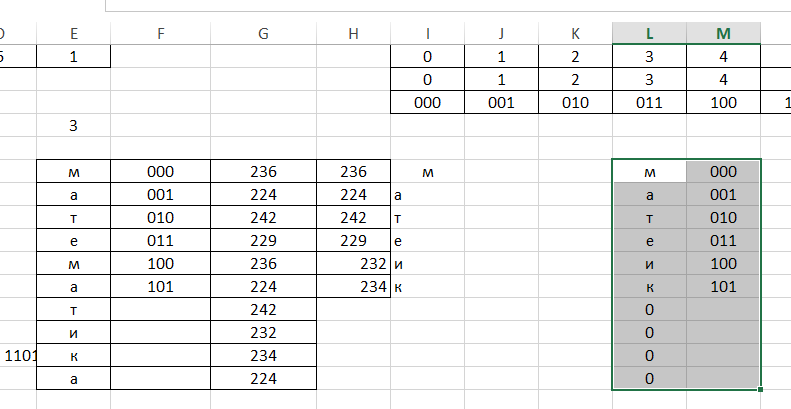


Дальше я делал различные идеи, пытался их реализовать, но это не понадобилось.

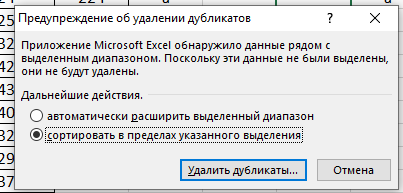


Я оставил это, чтобы не путать себя, что будет если я это уберу. Тем более там идет связь (на картинке самый правый столбец) столбца с другими столбцами далее.

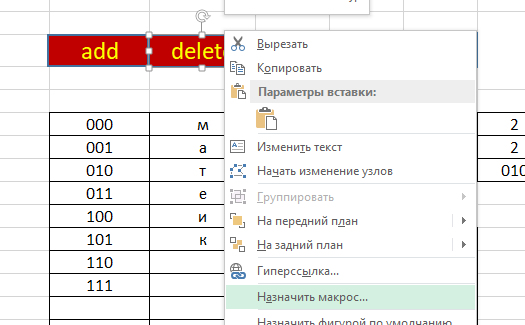
Далее перечислил все символы из столбца I6:I15, это уже используется активно. В I6:I15 с помощью макроса будут удаляться дубликаты символов. Без дубликатов все будет записываться в столбец L6:L15, где вместо отсутствующих символов будет 0. В столбец M6:M15 я перечислил коды из F6:F15.



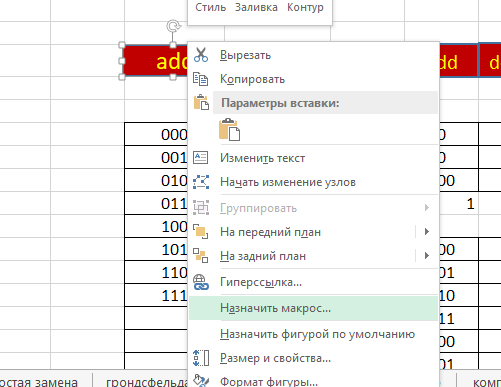
Сначала создадим макрос, который будет удалять дубликаты в столбце H6:H15. Макрос записал как обычно, действие при записи выполнял по пути Данные > Удалить дубликаты > Сортировать в пределах указанного выделения > Снял галочку мои данные содержат заголовки > Ок.



Этот макрос мы назначим первой кнопке delete.



Макрос для add будет просто добавлять данные ячеек из столбца G6:G15 в H6:H15.

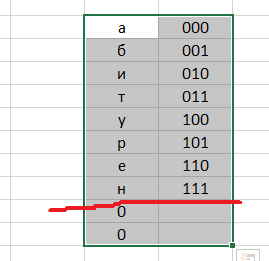


Эти две кнопки главное нажимать последовательно: сначала add, потом delete (по сути это не нужно, т.к. удаляются дубликаты в ненужных столбцах, но чтобы не путаться – лучше нажать).

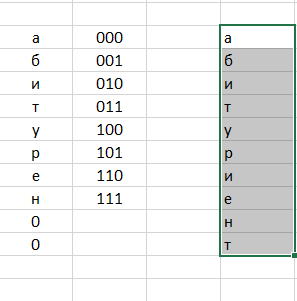
Далее создаем еще два макроса для второй пары кнопок add и delete, которые будут добавлять символы из столбца E6:E15 в столбец I6:I15 и удалять дубликаты в столбце I6:I15 соотвественно. Чтобы не было ошибок, в ячейках I6:I15 и H6:H15 привязка к ячейках происходит с символами $.

Нажимаем четыре кнопки последовательно: add, delete, add, delete. Это исключит ошибки в расчетах точно.

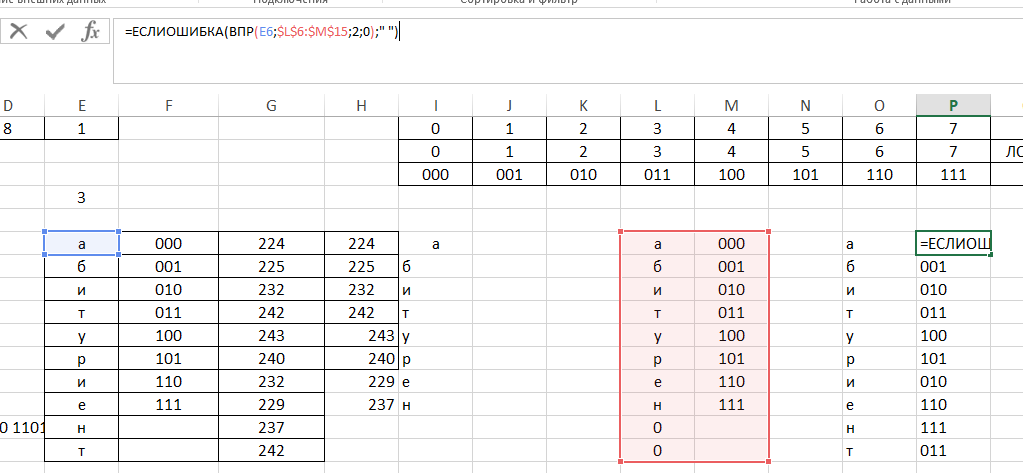
Так как дубликаты удалены, то в столбце L6:L15 не будет дубликатов, там связь, созданная ранее. Ранее мы создавали коды для каждого уникального символа, так вот теперь количество символов без дубликатов и коды символа совпали по количеству.



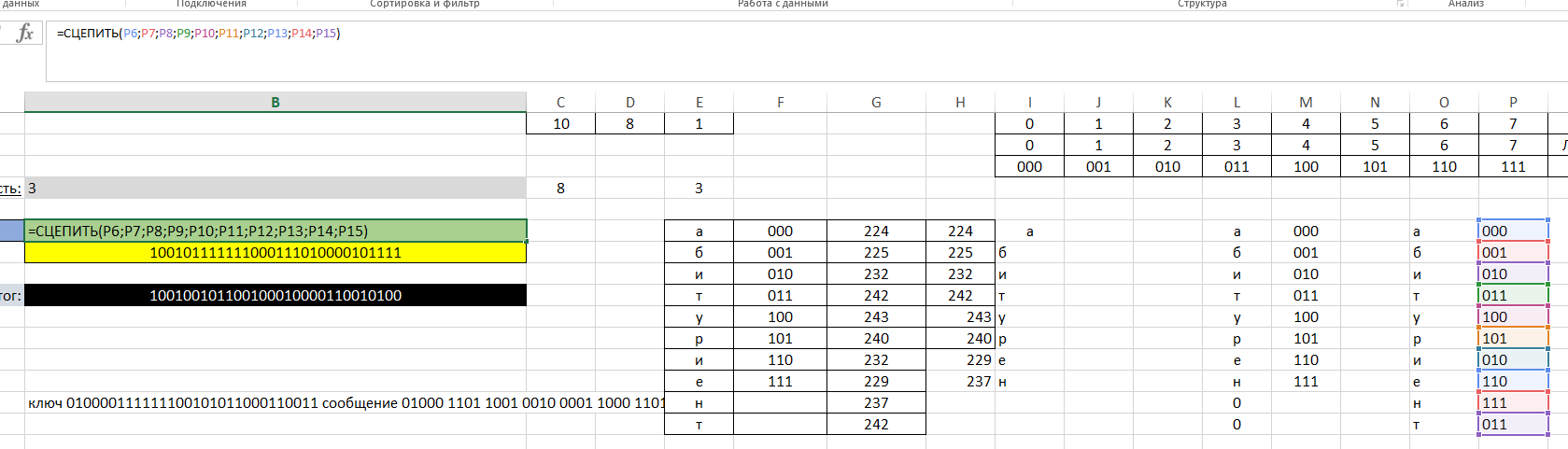
В соседнем столбце выписываю буквы из столбца E6:E15 для дальнейшнего использования в формуле ВПР.



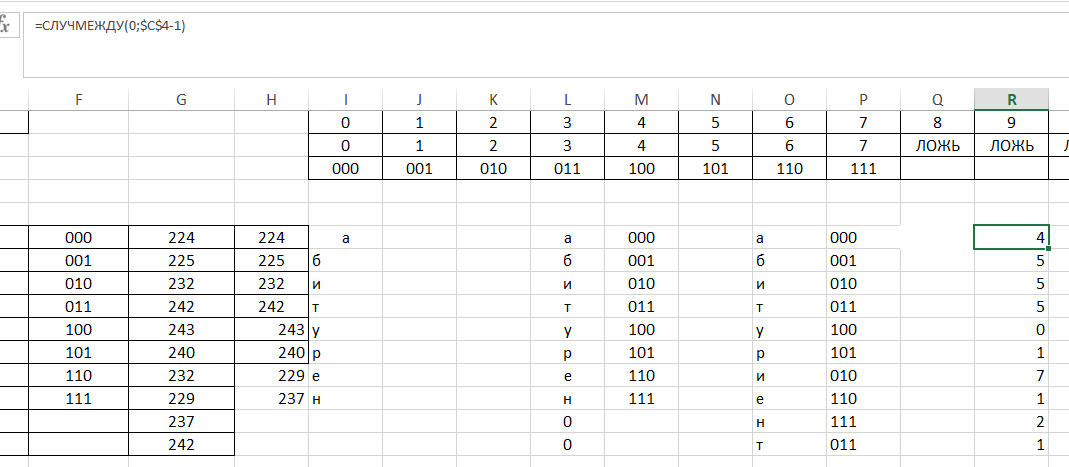
Правее в столбце вписываю формулу =ЕСЛИОШИБКА(ВПР(E6;$L$6:$M$15;2;0);" ") для каждого символа левого столбца. Если будет ошибка, то останется только пробел. Функция ищет значение ячейки E6 в диапазоне L6:M15 и вставляет в ячейку, где формула вписана, значение из второго столбца диапазона.



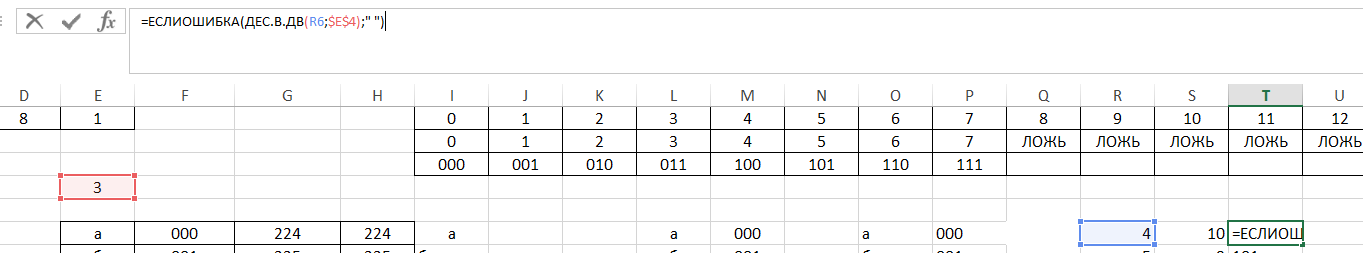
Теперь мы получили разделенный закодированный код, соединяем его с помощью функции СЦЕПИТЬ в зеленой ячейке B6.



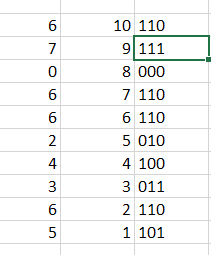
Чтобы получить рандомно сгенерированный ключ, в соседнем столбце R6:R15 вписываем функцию СЛУЧМЕЖДУ(0;$C$4-1), и размазываем ее по всему столбцу. Функция срандомит числа между 0 и 2^разрядность-1.



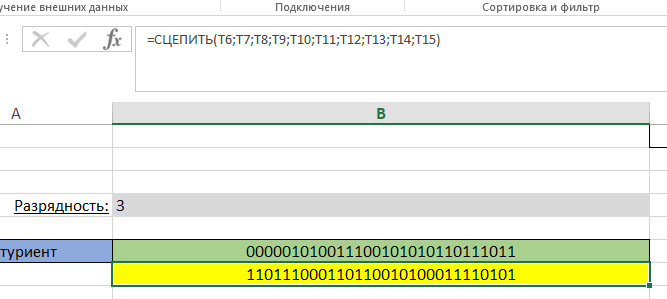
Переводим эти числа в двоичную систему счисления.



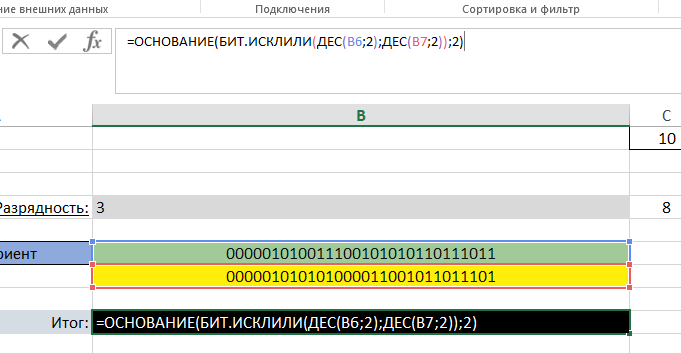
Получив двоичный код ключа,



скрепляем его с помощью функции СЦЕПИТЬ в ячейке B7.

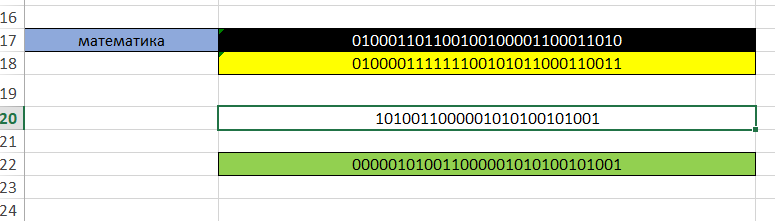


Вводим функцию =ОСНОВАНИЕ(БИТ.ИСКЛИЛИ(ДЕС(B6;2);ДЕС(B7;2));2) в ячейку B9 для выполнения операции XOR между двумя двоичными значениями ячеек B6 и B7, а потом для перевода числа в двоичную систему счисления, т.к. результат операции XOR будет в десятичной системе счисления.

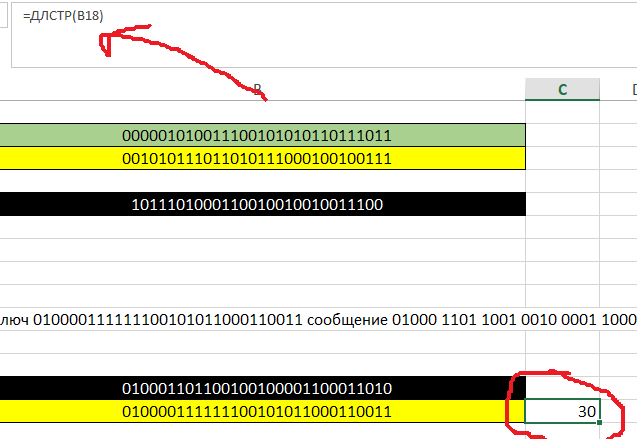


Процесс зашифрования готов. Остался процесс расшифровки.

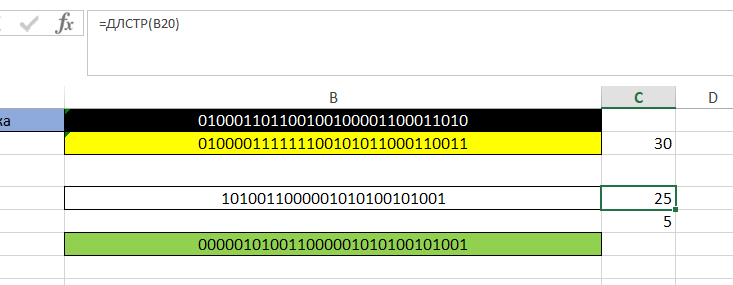
Выделяю 5 областей: синяя для расшифрованного сообщения, черная для закрытого текста, желтая для ключа, белая для промежуточной операции XOR, зеленая для итога операции XOR с добавлением пропущенных нулей.



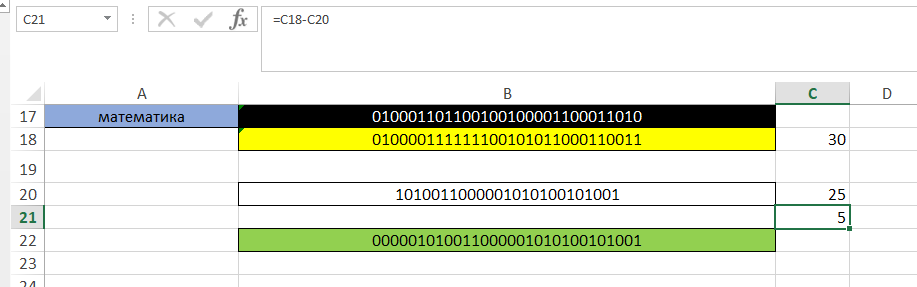
Чтобы потом добавить пропущенные нули в прозрачной области в зеленую область, выполним подсчет количества символов в ячейке B18.



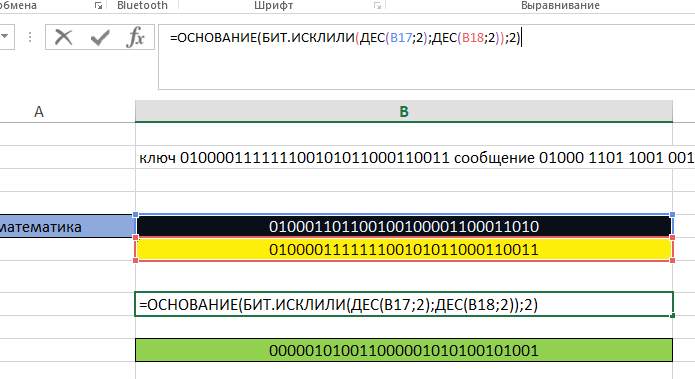
Далее находим размер данных в ячейке B20.



Находим разность ячейки C18 и C20, чтобы узнать количество пропущенных нулей в прозрачной области.

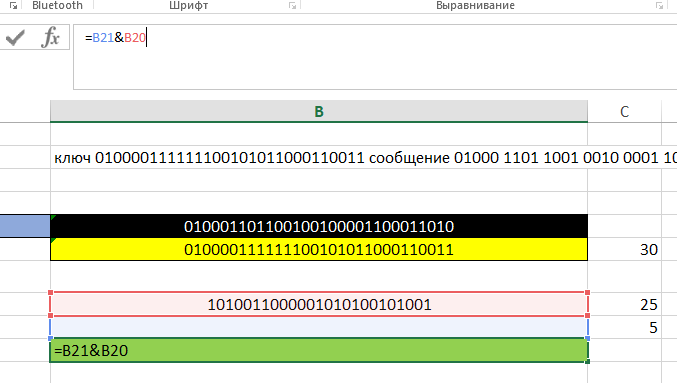


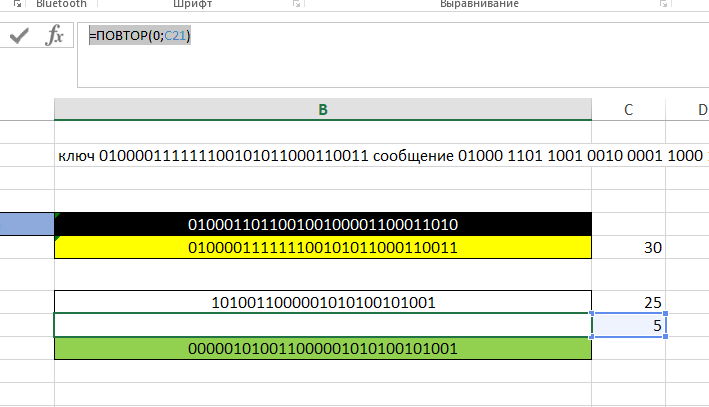
Сначала выполняем в прозрачной области операцию XOR между данными черной ячейки и желтой, чтобы получить исходное зашифрованное сообщение.



БИТ.ИСКЛИЛИ выполнить операцию XOR, ДЕС переведет значение в черной и желтой областях в двоичную систему счисления, т.к. это текстовое поле. Результат будет переведет с помощью ОСНОВАНИЕ в двоичную систему счиления, т.к. результат в десятичной системе счисления.

Сейчас пригодится ранее выделения область с количеством пропущенных бит. В зеленой области вписываем функцию =B21&B20, которая соединит две строки. В ячейку B21 вписываем функцию =ПОВТОР(0;C21), где C21 – количество пропущенных бит. Получаем совокупность нулей и далее в зеленой области получаем исходный текст.

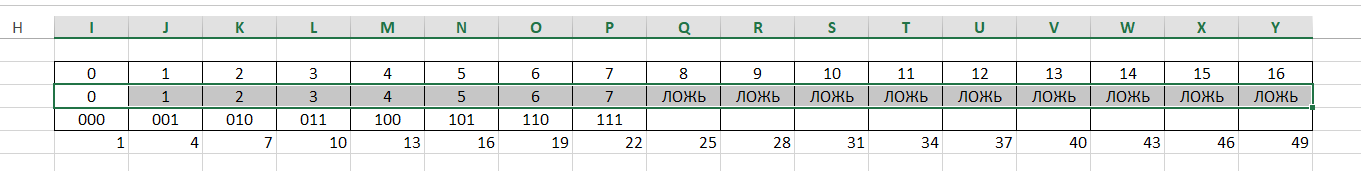




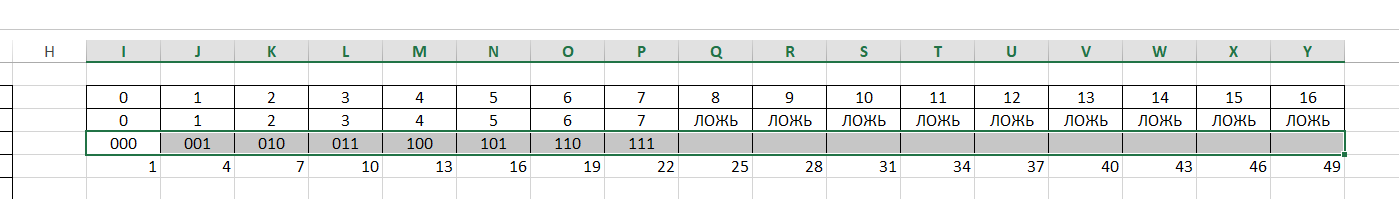
Текст ячейки B21 белый, чтобы скрыть эту информацию от пользователя.

Ячейки I22:Y22 нумеруем как и в ячейках I1:Y1, это пригодится далее.

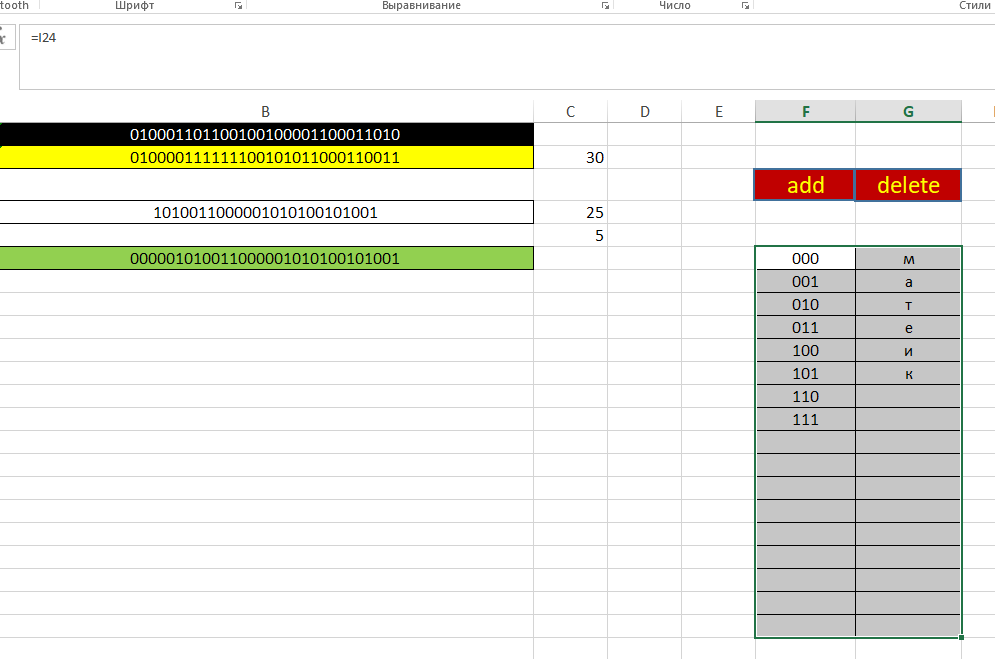
Также оставляем столько двоичных кодов для уникальных символов, сколько у нас самих уникальных символов.



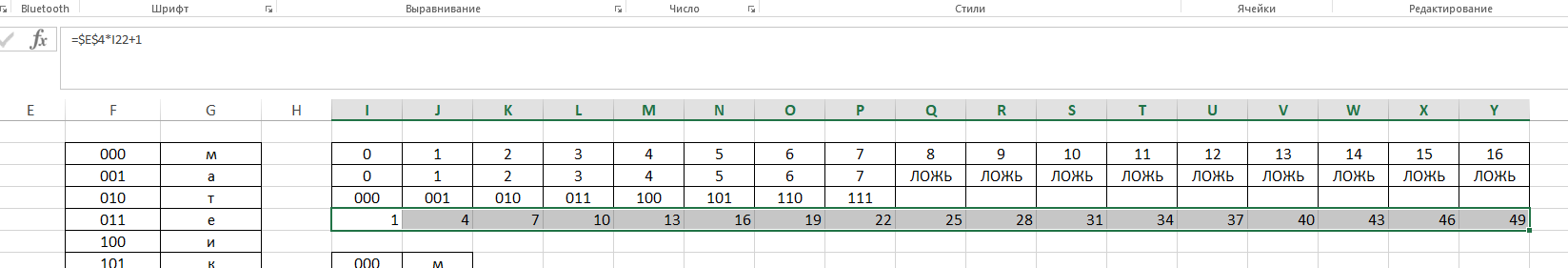
Ниже переводим, аналогично предыдущему разу, получившиеся символы в двоичный код.



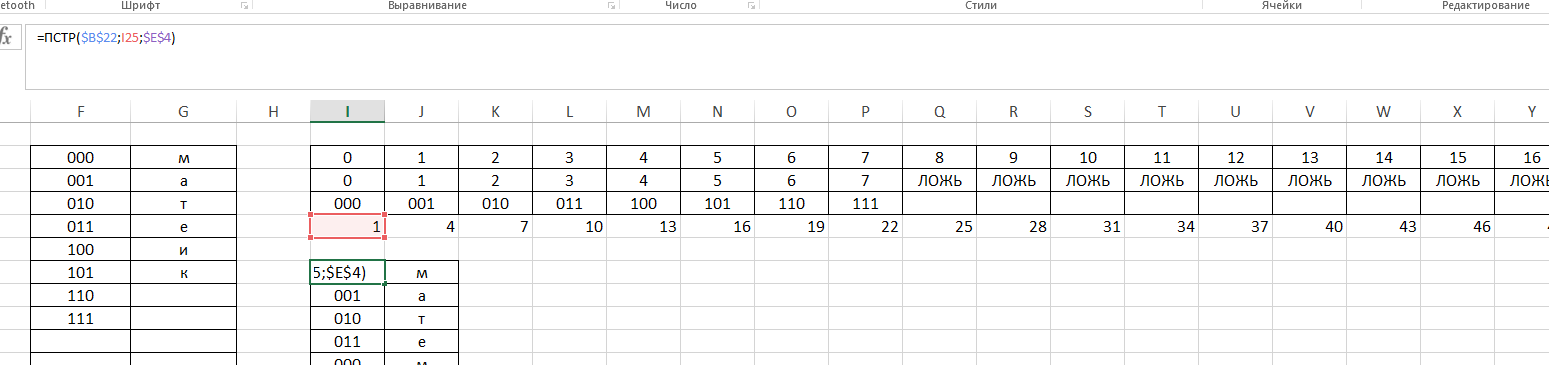
В столбец F22:F38 заносим данные ячеек I24:Y24 – это коды уникальных символов. Правее заносим статично уникальные символы, которые находятся в открытом тексте. Это делает сам пользователь.



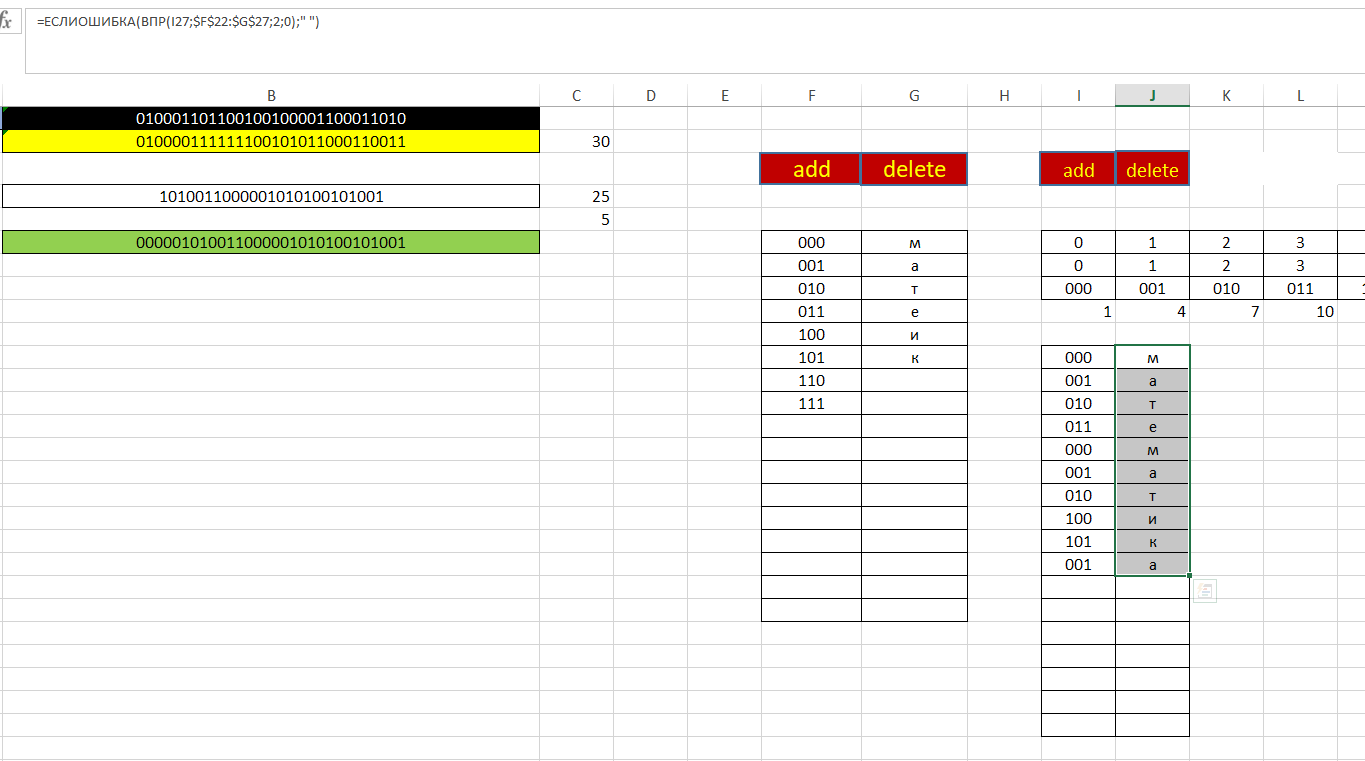
В ячейки I25:Y25 вводим формулу =$E$4\*I22+1, где E4 – разрядность кода для одного символа, I22 – нумерация одной тройки битов. Это значение понадобится, когда будем вырезать по N битов из зеленой ячейки. Т.е. если разрядность 3, то каждые три бита будем брать последовательно из зеленой области.



В столбец I27:I43 помещаем формулу =ПСТР($B$22;I25;$E$4), где B22 – кодированное сообщение, I25 – позиция вырезаемых бит, E4 – количество вырезаемых бит, для вырезки N битов из M позиции.



В ячейки J27:J43 записываю формулу ВПР (в одной из ячеек =ЕСЛИОШИБКА(ВПР(I27;$F$22:$G$27;2;0);" ") к примеру), чтобы каждой вырезанной тройке бит сопоставить свой символ.



Получив буквы открытого текста, соединяем их в ячейке A17 с помощью функции =СЖПРОБЕЛЫ(СЦЕПИТЬ(J27;J28;J29;J30;J31;J32;J33;J34;J35;J36;J37;J38;J39;J40;J41;J42;J43)), где функция СЖПРОБЕЛЫ уберет пробелы возникаемые в результате некоторых ошибок вычислений, если таковы будут.

Процесс расшифровки готов.