



Аннотация

Программа RS-Codec предназначена для изучения возможностей обнаружения и исправления искажения информации при применении кодов Рида-Соломона, анализа исходных и вспомогательных данных, используемых при кодировании и декодировании, и вероятностного анализа искажений и решений, принимаемых алгоритмом декодирования.

Программа не рассчитана на широкий круг пользователей. Для работы с программой требуется определенный уровень знаний и квалификации пользователя в области современной алгебры, дискретной математики, теории чисел, теории кодирования и теории вероятностей. Кроме того, знание кодов Рида-Соломона и наличие опыта их применения значительно облегчает освоение интерфейса и функций программы и позволяет максимально использовать все ее возможности. Программа в первую очередь предназначена для студентов, преподавателей и специалистов в области кодирования, передачи и защиты информации в современных системах хранения и сетях передачи данных, а также в области цифрового телевидения и спутниковой связи.



Условия применения

Программа поставляется в том виде как есть, и автор не несет ответственности в случае, если использование или неправильное использование программы повлекло умышленное или неумышленное нарушение функционирования каких-либо технических систем, либо искажение или потерю каких-либо программ или данных, либо иные виды ущерба, либо упущение какой-либо выгоды. Пользователи программы используют ее на свой страх и риск. Никаких гарантий к программе не прилагается и не предусматривается.

Программа поставляется для бесплатного использования, и не допускается создавать новые версии, сдавать внаем или аренду, продавать, модифицировать, дизассемблировать, запускать программу в отладчиках или изучать код программы другими способами.

Программа поставляется без какой-либо технической поддержки. Также автор программы не несет никаких обязательств по дальнейшему развитию программы, консультированию по вопросам внедрения и эксплуатации программы, а также по вопросам о деталях разработки программы и возможности расширения ее функциональности.

Данное руководство пользователя поставляется в том виде, как есть. Не допускается публикация данного руководства в каких-либо печатных изданиях или в Интернет без ведома и согласия автора. Руководство поставляется для бесплатного использования и не допускается создавать новые версии, модифицировать, продавать, сдавать внаем или аренду.

Все права авторства принадлежат разработчику (автору) программы и данного руководства и охраняются согласно действующему законодательству.

Microsoft, Windows, Windows NT являются зарегистрированными торговыми марками фирмы Microsoft Corporation.

Intel, Pentium, Pentium Pro, Itanium являются зарегистрированными торговыми марками фирмы Intel Corporation.

Borland, Delphi, Borland Pascal являются зарегистрированными торговыми марками фирмы Borland Software Corporation.



Системные требования

- Процессор: Intel Pentium или выше. Оперативная память: 1,5 Мбайт. Диск: 640 Кбайт.
- Операционная система: ОС Microsoft Windows 98 / NT 4.0 или более новые версии.



Установка и настройка

Программа поставляется в виде одного единственного исполняемого файла RSCODEC.EXE и не требует никаких специальных действий по установке или настройке. Для начала работы с программой достаточно запустить исполняемый файл.



Краткое описание программы и ее ограничений

Программное обеспечение включает в себя следующие компоненты:

- Графический интерфейс пользователя, включающий в себя главное окно и множество дополнительных окон с необходимыми элементами управления теми или иными функциями программы, элементами для ввода данных и отображения результатов.
- Кодер, позволяющий закодировать текстовое сообщение длиной в пределах от 1 до 253 символов с применением кодов Рида-Соломона для заданного четного количества контрольных байтов в пределах от 2 до 254, и передать сформированный кадр в среду искажения. Минимальное кодовое расстояние пространства кадров для подмножества допустимых кадров (под «допустимым кадром» понимается термин «кодовое слово», используемый в теории кодирования) равно количеству контрольных байтов плюс один. Соответственно, кратность исправляемых ошибок равна половине количества контрольных байтов. Полный размер кадра должен находиться в пределах от 3 до 255. При кодировании также формируется порождающий полином, при помощи которого выполняется кодирование текстового сообщения.
- Среда искажения, позволяющая исказить заданное количество байтов кадра, выбираемых случайным образом и искажаемых при помощи операции XOR на случайную величину от 1 до 255. Количество искажаемых байтов не может превышать полный размер кадра. Кроме того, в среде искажения также имеется возможность редактировать байты кадра в шестнадцатеричном формате.
- Декодер, позволяющий принять кадр из среды искажения и вычислить для него синдром ошибок, полином локаторов ошибок, сами локаторы ошибок, полином величин ошибок, полином формальной производной, сами величины ошибок, и сформировать итоговый полином ошибок, и преобразовать его вектор ошибок. В случае если на каком-либо этапе декодирования не удастся продолжить вычисления в силу неисправимого характера искажений, выводится соответствующее сообщение. В случае отсутствия ошибок также выводится соответствующее сообщение. В случае наличия ошибок и успешного декодирования выводится вектор ошибок и информационная часть исправленного кадра.
- Калькулятор для поля Галуа, позволяющий выполнять вычисления арифметических выражений с использованием правил и элементов в поле Галуа $GF(2^8)$. В арифметических выражениях можно использовать скобки и операцию возведения в степень. Арифметические операции и скобки для степеней не поддерживаются.
- Модуль для расчета вероятностей искажений для заданной длины кадра (от 3 до 255) и кратности исправляемых ошибок (от 1 до 127, причем кратность должна быть меньше половины длины кадра) и вероятности искажения бита для биномиального распределения искажений в канале передаче данных при условии, что вероятность искажения бита постоянна во времени, и не зависит от того, были ли искажены другие биты или нет («канал передачи без памяти»).
- Модуль статистических испытаний, позволяющий генерировать заданное количество кадров (от 10^3 до 10^9) заданной длины (от 3 до 255) при заданной кратности исправляемых ошибок (от 1 до 127, причем кратность должна быть меньше половины длины кадра), вносить в них случайные искажения, декодировать их и собирать статистическую информацию по результатам искажения и декодирования кадров.



Описание графического интерфейса

Графический интерфейс программы включает 1 главное окно и 9 дополнительных окон:

- Главное окно для ввода текстового информационного блока, кодирования, генерации и внесения случайных искажений в кадр и декодирования кадра, а также для запуска дополнительных окон для детального изучения кодирования и декодирования, расчета вероятности искажений, проведения статистических испытаний.
- Дополнительное окно для отображения таблицы элементов поля Галуа, таблицы логарифмов элементов и выполнения арифметических операций с элементами поля.
- Дополнительное окно для отображения порождающего полинома и полинома-кадра, сформированного кодером при последней операции кодирования.
- Дополнительное окно для отображения полинома-кадра, принятого декодером, и вычисленного декодером синдрома ошибок, при последней операции декодирования.
- Дополнительное окно для отображения полинома локаторов ошибок и локаторов ошибок, найденных декодером при последней операции декодирования.
- Дополнительное окно для отображения полинома величин ошибок, полинома формальной производной полинома локаторов ошибок, а также величин ошибок, вычисленных декодером при последней операции декодирования.
- Дополнительное окно для отображения полинома ошибок, вычисленного декодером при последней операции декодирования.
- Дополнительное окно для расчета вероятностей основных и особых случаев искажения для биномиального распределения искажений.
- Дополнительное окно для проведения статистических испытаний, заключающихся в генерации заданного количества случайных кадров заданной длины с заданным количеством контрольных байтов, внесения случайных искажений в кадры, декодирования кадров, а также отображения статистической информации.
- Дополнительное окно для отображения информации о программе и ее разработчике.

Главное окно программы

Главное окно состоит из 4 секций, содержащих те или иные элементы интерфейса:

- *Сторона кодера* – секция для ввода и кодирования текстовой информации.
- *Среда искажения* – секция для просмотра, редактирования и искажения кадра.
- *Сторона декодера* – секция для декодирования кадра и просмотра вектора ошибок.
- *Дополнительные функции и информация* – секция, содержащая кнопки для запуска дополнительных функциональных и информационных окон программы.

Секция «сторона кодера»:

- *Информационный блок на стороне кодера* – поле для ввода исходной текстовой информации, используемой для кодирования. Длина информационного блока должна находиться в пределах от **1** до **255 – r** , где **r** это количество контрольных байтов.
- *Размер контрольного блока кадра* – ползунок для выбора количества контрольных байтов, используемого для кодирования. Количество контрольных байтов: от **2** до **254** (соответственно, кратность исправляемых ошибок: от **1** до **127**, а полная длина кадра: от **3** до **255** байтов).
- *Закодировать и передать кадр в среду* – кнопка для кодирования исходной текстовой информации и формирования кадра с необходимым количеством контрольных байтов, указанных в поле размера контрольного блока кадра. Если поле текстовой информации пустое, то выводится ошибка.
- *Параметр b* – поле для выбора параметра b , используемого при формировании порождающего полинома. Диапазон выбора: от **0** до **254**. По умолчанию **$b = 1$** .

Секция «среда искажения»:

- *Кадр в среде искажения* – поле для отображения кадра, сформированного кодером после успешной операции кодирования, в виде последовательности байтов в шестнадцатеричном представлении. Для компактности отображения байты кадра располагаются построчно по 32 байта в строке в матрице размером 32 столбца на 8 строк. Ячейки матрицы доступны не только для просмотра, но и для редактирования в шестнадцатеричном формате. Корректность вводимой информации контролируется. Кроме того, также контролируется доступность ячейки для редактирования: если ячейка соответствует байту, который не входит в пределы кадра, то она недоступна для редактирования. Двумерные координаты i и j ячейки матрицы и смещение q байта внутри кадра связаны следующими соотношениями:

$$\begin{cases} i = (n - 1 - q) \div 32, & i \in [0, 7] \\ j = (n - 1 - q) \bmod 32, & j \in [0, 31] \\ q = (n - 1) - (32 \cdot i + j), & q \in [0, n - 1] \end{cases}$$

- *Информационный блок в среде искажения* – в данном поле в текстовом виде отображается информационная часть сформированного кадра.
- *Количество искажаемых байтов в кадре* – ползунок для выбора количества одновременно искажаемых байтов в кадре с помощью генератора случайных искажений. Количество искажаемых байтов: от **1** до **255**. Количество искажаемых байтов не должно превышать полный размер кадра (сумму числа информационных и числа контрольных байтов), в случае нарушения данного условия выводится ошибка.
- *Внести искажение в кадр* – кнопка для искажения случайным образом заданного количества байтов в кадре. Кнопка по умолчанию недоступна и становится доступной только после выполнения операции кодирования.

Секция «сторона декодера»:

- *Вычисленный вектор ошибок на стороне декодера* – поле для отображения предполагаемого вектора ошибки, получаемого при успешном декодировании кадра. Так же, как и в вышеописанном случае, для компактности отображения, байты вектора размещаются построчно по 32 байта в строке в матрице размером 32 столбцов на 8 строк. Байты в векторе ошибок соответствуют байтам кадра, и находятся по тем же смещениям в матрице, что и байты в матрице для кадра в среде искажения.
- *Информационный блок на стороне декодера* – в данном поле выводится информационная часть принятого и исправленного декодером кадра.
- *Принять кадр из среды и декодировать* – кнопка для декодирования кадра. При декодировании программа считывает данные из поля «кадр в среде искажения», пытается декодировать кадр, обнаружить в нем искажения и исправить их по возможности. В зависимости от ситуации выводятся сообщения об отсутствии искажений в кадре, о его успешном исправлении или о невозможности исправления.

Секция «дополнительные функции и информация»:

- *Калькулятор для поля Галуа* – кнопка для отображения соответствующего окна.
- *Порождающий полином (кодер)* – кнопка для отображения соответствующего окна.
- *Синдром ошибок (декодер)* – кнопка для отображения соответствующего окна.
- *Локаторы ошибок (декодер)* – кнопка для отображения соответствующего окна.
- *Величины ошибок (декодер)* – кнопка для отображения соответствующего окна.
- *Полином ошибок (декодер)* – кнопка для отображения соответствующего окна.
- *Расчет вероятностей искажений* – кнопка для отображения соответствующего окна.
- *Статистические испытания* – кнопка для отображения соответствующего окна.
- *О программе* – кнопка для отображения соответствующего окна.

Декодер: В кадре успешно выявлены и исправлены ошибки.

Окно «калькулятор для поля Галуа»

Окно состоит из трех секций:

- *Калькулятор для поля Галуа* – секция для выполнения арифметических вычислений с элементами поля Галуа $GF(2^8)$. Секция содержит текстовое поле для ввода арифметического выражения и поле для вывода результата вычислений. В случае наличия синтаксических ошибок в выражении или возникновения ошибок при вычислении выражения в строке состояния окна выводится сообщение об ошибке.
- *Таблица степеней* – секция для отображения ненулевых элементов поля Галуа. Секция содержит ненулевые элементы поля Галуа $GF(2^8)$, расположенные построчно в ячейках матрицы размером 16 x 16. Элементы упорядочены в порядке возведения примитивного элемента 2 поля последовательно в 0-ю, 1-ю, ..., 254-ю степень. Последняя 255-я ячейка матрицы содержит значение нулевой ячейки (2^0).
- *Таблица логарифмов* – секция для отображения логарифмов ненулевых элементов поля Галуа. Секция содержит логарифмы ненулевых элементов поля Галуа $GF(2^8)$, расположенные построчно в ячейках матрицы размером 16 x 16. Нулевая ячейка матрицы содержит значение «неприменимо» (N/A). Далее, начиная с 1-й ячейки матрицы, заканчивая последней 255-й, располагаются значения логарифмов, упорядоченные в порядке логарифмов по основанию примитивного элемента 2 поля последовательно для всех ненулевых элементов поля Галуа: 1, 2, ..., 255.

Окно «порождающий полином (кодер)»

Окно состоит из двух секций:

- *Порождающий полином кодера* – секция для отображения коэффициентов порождающего полинома, сформированного кодером при последней операции кодирования для заданного количества контрольных байтов. Секция содержит матрицу размером 16 x 16, в которой построчно выводятся коэффициенты порождающего полинома, начиная с младшего 0-го и заканчивая старшим r -м, где r это количество контрольных байтов. В секции также имеется поле для ввода в качестве аргумента любого из элементов поля Галуа и поле для вывода значения порождающего полинома при заданном аргументе.
- *Переданный кодером кадр* – секция для отображения коэффициентов полинома-кадра, сформированного кодером при последней операции кодирования, для заданного информационного блока и количества контрольных байтов. Секция содержит матрицу размером 16 x 16, в которой построчно выводятся коэффициенты полинома-кадра, начиная с младшего 0-го и заканчивая старшим $(n - 1)$ -м, где n это длина кадра. В секции также имеется поле для ввода в качестве аргумента любого из элементов поля Галуа и поле для вывода значения полинома-кадра при заданном аргументе.

Окно «синдром ошибок (декодер)»

Окно состоит из двух секций:

- *Принятый декодером кадр* – секция для отображения коэффициентов полинома-кадра, принятого декодером из среды искажения при последней операции декодирования. Секция содержит матрицу размером 16 x 16, в которой построчно выводятся коэффициенты полинома-кадра, начиная с младшего 0-го и заканчивая старшим $(n - 1)$ -м, где n это длина кадра. В секции также имеется поле для ввода в качестве аргумента любого из элементов поля Галуа и поле для вывода значения полинома-кадра при заданном аргументе.

- *Синдром ошибок* – секция для отображения компонент синдрома ошибок, вычисленных декодером для принятого из среды искажения кадра при последней операции декодирования. Секция содержит матрицу размером 16×16 , в которой построчно выводятся компоненты синдрома, начиная с младшего 1-го и заканчивая старшим r -м, где r – это количество контрольных байтов.

Окно «локаторы ошибок (декодер)»

Окно состоит из двух секций:

- *Полином локаторов ошибок* – секция для отображения коэффициентов полинома локаторов ошибок, найденного декодером для принятого из среды искажения кадра при последней операции декодирования, при котором полином локаторов ошибок был успешно найден. Секция содержит матрицу размером 8×16 , в которой построчно выводятся коэффициенты полинома локаторов, начиная с младшего 0-го и заканчивая старшим t -м, где t – предполагаемое декодером количество ошибок. В секции также имеется поле для ввода в качестве аргумента любого из элементов поля Галуа и поле для вывода значения полинома локаторов ошибок при заданном аргументе.
- *Локаторы ошибок* – секция для отображения локаторов, найденных декодером для принятого из среды искажения кадра при последней операции декодирования, при котором локаторы ошибок были успешно найдены. Секция содержит матрицу размером 8×16 , в которой построчно выводятся локаторы ошибок, начиная с первого и заканчивая последним, отсортированные в порядке возрастания.

Окно «величины ошибок (декодер)»

Окно состоит из двух секций:

- *Полином величин ошибок* – секция для отображения коэффициентов полинома величин ошибок, вычисленного декодером для принятого из среды искажения кадра при последней операции декодирования, при котором полином величин ошибок был успешно вычислен. Секция содержит матрицу размером 8×16 , в которой построчно выводятся коэффициенты полинома величин, начиная с младшего 0-го и заканчивая старшим $(t - 1)$ -м, где t – предполагаемое декодером количество ошибок. В секции также имеется поле для ввода в качестве аргумента любого из элементов поля Галуа и поле для вывода значения полинома величин ошибок при заданном аргументе.
- *Полином формальной производной полинома локаторов ошибок* – секция для отображения коэффициентов полинома формальной производной, вычисленной декодером для принятого из среды искажения кадра при последней операции декодирования, при котором полином формальной производной был успешно вычислен. Секция содержит матрицу размером 8×16 , в которой построчно выводятся коэффициенты полинома формальной производной, начиная с младшего 0-го и заканчивая старшим $((t - 1) - (t - 1) \bmod 2)$ -м, где t – предполагаемое декодером количество ошибок. В секции также имеется поле для ввода в качестве аргумента любого из элементов поля Галуа и поле для вывода значения полинома формальной производной при заданном аргументе.
- *Величины ошибок* – секция для отображения величин ошибок, вычисленных декодером для принятого из среды искажения кадра при последней операции декодирования, при котором величины были успешно вычислены. Секция содержит матрицу размером 8×16 , в которой построчно выводятся величины ошибок, начиная с первого и заканчивая последним, в порядке соответствующих им локаторов ошибок (расположение величин ошибок в матрице величин ошибок в точности соответствует расположению соответствующих им локаторов ошибок в матрице локаторов ошибок).

Окно «полином ошибок (декодер)»

Окно состоит из одной секции. *Вычисленный полином ошибок* – секция для отображения коэффициентов полинома ошибок, сформированного декодером для принятого из среды искажения кадра при последней операции декодирования, при котором полином ошибок был успешно сформирован. Секция содержит матрицу размером 16×16 , в которой построчно выводятся коэффициенты полинома ошибок, начиная с младшего 0-го и заканчивая последним $(n - 1)$ -м, где n это длина кадра. В секции также имеется поле для ввода в качестве аргумента любого из элементов поля Галуа и поле для вывода значения полинома ошибок при заданном аргументе.

Вид окна «калькулятор для поля Галуа»

Калькулятор для поля Галуа $GF(2^8)$:

$(3 \cdot 12 + 33^{13}) / 56 = 149$

Таблица степеней примитивного элемента ($\alpha = 2$) поля Галуа $GF(2^8)$

	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
0	1	2	4	8	16	32	64	128	29	58	116	232	205	135	19	38
16	76	152	45	90	180	117	234	201	143	3	6	12	24	48	96	192
32	157	39	78	156	37	74	148	53	106	212	181	119	238	193	159	35
48	70	140	5	10	20	40	80	160	93	186	105	210	185	111	222	161
64	95	190	97	194	153	47	94	188	101	202	137	15	30	60	120	240
80	253	231	211	187	107	214	177	127	254	225	223	163	91	182	113	226
96	217	175	67	134	17	34	68	136	13	26	52	104	208	189	103	206
112	129	31	62	124	248	237	199	147	59	118	236	197	151	51	102	204
128	133	23	46	92	184	109	218	169	79	158	33	66	132	21	42	84
144	168	77	154	41	82	164	85	170	73	146	57	114	228	213	183	115
160	230	209	191	99	198	145	63	126	252	229	215	179	123	246	241	255
176	227	219	171	75	150	49	98	196	149	55	110	220	165	87	174	65
192	130	25	50	100	200	141	7	14	28	56	112	224	221	167	83	166
208	81	162	89	178	121	242	249	239	195	155	43	86	172	69	138	9
224	18	36	72	144	61	122	244	245	247	243	251	235	203	139	11	22
240	44	88	176	125	250	233	207	131	27	54	108	216	173	71	142	1

Таблица логарифмов по основанию примитивного элемента поля Галуа $GF(2^8)$

	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
0	N/A	0	1	25	2	50	26	198	3	223	51	238	27	104	199	75
16	4	100	224	14	52	141	239	129	28	193	105	248	200	8	76	113
32	5	138	101	47	225	36	15	33	53	147	142	218	240	18	130	69
48	29	181	194	125	106	39	249	185	201	154	9	120	77	228	114	166
64	6	191	139	98	102	221	48	253	226	152	37	179	16	145	34	136
80	54	208	148	206	143	150	219	189	241	210	19	92	131	56	70	64
96	30	66	182	163	195	72	126	110	107	58	40	84	250	133	186	61
112	202	94	155	159	10	21	121	43	78	212	229	172	115	243	167	87
128	7	112	192	247	140	128	99	13	103	74	222	237	49	197	254	24
144	227	165	153	119	38	184	180	124	17	68	146	217	35	32	137	46
160	55	63	209	91	149	188	207	205	144	135	151	178	220	252	190	97
176	242	86	211	171	20	42	93	158	132	60	57	83	71	109	65	162
192	31	45	67	216	183	123	164	118	196	23	73	236	127	12	111	246
208	108	161	59	82	41	157	85	170	251	96	134	177	187	204	62	90
224	203	89	95	176	156	169	160	81	11	245	22	235	122	117	44	215
240	79	174	213	233	230	231	173	232	116	214	244	234	168	80	88	175

GT[0] = 1 LT[1] = 0 Готово

[illegible]

[illegible]

Вид окна «локаторы ошибок (декодер)»

Decoder Error-Locators

Полином локаторов ошибок

Коэффициенты полинома локаторов: $L[0]...L[2]$

1	93	19																	

Вычисление значения полинома при заданном аргументе

X: L(X):

Локаторы ошибок

Локаторы в порядке возрастания: $LOC[1]...LOC[2]$

6	8																		

$L[0] = 1$ $LOC[1] = 6$

Вид окна «полином ошибок (декодер)»

Decoder Error-Polynom

Вычисленный полином ошибок

Коэффициенты полинома ошибок: $E[0]...E[8]$

0	0	0	0	0	0	6	0	30											

Вычисление значения полинома при заданном аргументе

X: E(X):

$E[0] = 0$

Важное примечание. В дополнительных окнах для отображения порождающего полинома и полинома-кадра, сформированных кодером, полинома-кадра, принятого декодером, синдрома ошибок, полинома локаторов ошибок и самих локаторов ошибок, полинома величин ошибок, полинома формальной производной и самих величин ошибок, вычисленных и вычисленных декодером, а также полинома ошибок, сформированного декодером, используется десятичное представление данных. При этом коэффициенты полиномов всегда выводятся в обратном порядке, начиная с младшего коэффициента и заканчивая старшим. В то же время в главном окне программы байты кадра и байты вектора ошибок для наглядности и компактности выводятся в шестнадцатеричном представлении, начиная с младшего байта и заканчивая старшим.

Окно «расчет вероятностей искажений»

Error Probability Calculator

Расчет вероятностей искажений при биномиальном распределении:

Вероятность искажения бита (p):

Размер кадра в байтах (n): 36

Кратность исправляемых ошибок (t): 1

Вероятности основных случаев при биномиальном распределении:

0,763926491363159	Вероятность искажения хотя бы 1 байта: $P(T > 0)$
0,236073508636841	Вероятность отсутствия искажений: $P(T = 0)$
0,34772388792167	Вероятность искажения от 1 до t байтов: $P(1 \leq T \leq t)$
0,583797396558511	Вероятность искажения не более t байтов: $P(0 \leq T \leq t)$
0,416202603441489	Вероятность искажения более t байтов: $P(T > t)$
0,16722668284947	Вероятность искажения более 2 * t байтов: $P(T > 2 * t)$

Вероятности особых случаев невозможных искажений ($T > t$):

0,359574286267783	Вероятность такого искажения, при котором оказывается невозможным исправление искаженного кадра.
0,0566257516805346	Вероятность такого искажения, при котором исправление приводит к некоторому допустимому кадру, не являющемуся исходным - т.е. смежное исправление.
2,56549317217962E-6	Вероятность такого искажения, при котором искаженный кадр оказывается допустимым, не являющимся исходным - т.е. происходит маскирование искажения.

Готово

Окно состоит из трех секций:

- *Расчет вероятностей искажений при биномиальном распределении* – секция для задания вероятности искажения бита p , размера кадра n , кратности исправляемых ошибок t , а также для запуска и останова процедуры расчета вероятностей, и наблюдения за прогрессом (доли выполнения) расчетов. Секция содержит следующие элементы графического интерфейса:
 - Поле для ввода вероятности искажения бита в формате действительного числа в пределах от 0 до 1. Корректность вводимых данных контролируется.
 - Ползунок для выбора размера кадра в пределах от 3 до 255.
 - Ползунок для выбора кратности исправляемой ошибки в пределах от 1 до 127. Кратность исправляемой ошибки должна быть меньше половины длины кадра.
 - Кнопка «пуск» для запуска процедуры расчета вероятностей.
 - Кнопка «стоп» для принудительного останова процедуры расчетов. При принудительной остановке расчетов часть результатов, которые не были до конца вычислены, не отображается.
 - Индикатор прогресса частичных расчетов внутри общего расчета.
 - Индикатор общего прогресса расчета вероятностей.

- *Вероятности основных случаев при биномиальном распределении* – секция для отображения вычисленных вероятностей для типовых случаев искажения. Секция содержит шесть текстовых полей для отображения:
 - Вероятности искажения хотя бы одного байта.
 - Вероятности отсутствия искажений.
 - Вероятности искажения от 1 байта до t байтов (восстановимое искажение).
 - Вероятности искажения не более t байтов.
 - Вероятности искажения более t байтов (невосстановимое искажение).
 - Вероятности искажения более $2t$ байтов.
- *Вероятности особых случаев невосстановимых искажений* – секция для расчета вероятностей трех частных случаев невосстановимых искажений, когда искажается более t байтов. Секция содержит три текстовых поля для отображения:
 - Вероятности неисправимого искажения.
 - Вероятности смежно-исправимого искажения.
 - Вероятности маскируемого искажения.

Примечание. Отметим, что рассчитанные вероятности искажений справедливы при допущении, что вероятность искажения бита одинакова для всех битов всех байтов кадра и не зависит от расположения конкретного бита в конкретном байте внутри кадра.

Окно «статистические испытания»

Reed-Solomon Codec Stats

Генератор случайных кадров и искажений:

Размер кадра в байтах (n): 36

Кратность исправляемых ошибок (t): 1

Количество искажаемых байтов (w): 0

Количество генерируемых кадров: 1000000

☐ Искажать заданное количество байтов (w)
☐ Искажать равномерно распределенное случайное количество байтов от 0 до w
☐ Искажать равномерно распределенное случайное количество байтов от w до n
☒ Искажать биномиально распределенное случайное количество байтов от 0 до n при заданной вероятности искажения бита (p):

Пуск Стоп 0.005

Общая статистика декодера:

Количество кадров, распознанных как неискаженные, причем исходный кадр действительно не был искажен: 236332

Количество исправленных кадров, причем исправленный кадр совпал с исходным неискаженным кадром: 346725

Количество кадров, которые были распознаны как неисправимые: 360178

Количество исправленных кадров, причем исправленный кадр не совпал с исходным неискаженным кадром: 56762

Количество кадров, распознанных как неискаженные, причем исходный кадр на самом деле был искажен: 3

Статистика декодера по неисправным кадрам

Количество кадров, для которых не найден полином локаторов ошибок: 1631

Кол-во кадров, для которых найдено некорректное количество локаторов: 1669

Кол-во кадров, для которых найдены некорректные локаторы (вне кадра): 356878

Кол-во кадров, после исправления которых получен ненулевой синдром: 0

Статистика искажения:

Общее количество кадров: 1000000

Средняя скорость обработки кадров (за секунду процессорного времени): 62496

Кол-во искаженных кадров: 763668

Среднее кол-во искажаемых байтов: 1,416

Окно состоит из четырех секций:

- *Генератор случайных кадров и искажений* – секция для задания размера кадра n , кратности исправляемых ошибок t , количества искажаемых байтов w , количества генерируемых байтов, вероятности искажения бита p , выбора одного из четырех режимов искажения, а также для запуска и останова генератора кадров и искажений кадра. Секция содержит следующие элементы графического интерфейса:
 - Ползунок для выбора размера кадра в пределах от **3** до **255**.
 - Ползунок для выбора кратности исправляемой ошибки в пределах от **1** до **127**. Кратность исправляемой ошибки должна быть меньше половины длины кадра.
 - Ползунок для выбора количества искажаемых байтов от **0** до **255** для первых трех режимов искажения (одного режима фиксированного числа и двух режимов равномерно распределенного случайного числа искажаемых байтов).
 - Ползунок для выбора числа генерируемых кадров в пределах от 10^3 до 10^9 .
 - Группа из 4-х радио-кнопок для выбора режима искажения:
 - Ровно заданного числа w байтов. Данный режим полезен для анализа работы декодера при конкретном количестве искажаемых байтов.
 - Равномерно распределенного случайного числа байтов от **0** до w . Данный режим полезен для анализа работы декодера в области «восстановимых» искажений (при $w \leq t$).
 - Равномерно распределенного случайного числа байтов от w до n . Данный режим полезен для анализа работы декодера в области «невосстановимых» искажений (при $w > t$).
 - Биномиально распределенного случайного числа байтов при заданной вероятности искажения бита p . Данный режим полезен для анализа работы декодера в условиях, приближенных к искажению кадров в канале передачи данных с постоянной во времени вероятностью искажения бита, причем вероятность искажения бита не зависит от того, были предыдущие биты или нет (канал передачи «без памяти»).
 - Кнопка «пуск» для запуска генератора кадров и искажений, а также декодера.
 - Кнопка «стоп» для останова генератора кадров и искажений, а также декодера.
 - Поле для ввода вероятности искажения бита в формате действительного числа в пределах от **0** до **1** для четвертого режима искажения (биномиального распределения искажений).
- *Статистика искажения* – секция для отображения общего количества сгенерированных кадров, количества искаженных кадров, средней скорости обработки кадров (за секунду процессорного времени), среднего количества искажаемых байтов при заданном режиме искажения кадров.
- *Общая статистика декодера* – секция для отображения количества:
 - Неискаженных кадров, распознанных как неискаженные кадры.
 - Искаженных кадров, исправление которых дало исходный кадр.
 - Искаженных кадров, которые оказались неисправимыми.
 - Искаженных кадров, исправление которых дало другой «допустимый кадр» (кодовое слово), не являющийся исходным кадром – смежное исправление.
 - Искаженных кадров, оказавшихся «допустимыми кадрами», не являющимися исходными, и распознанных как неискаженные – маскирование искажения.
- *Статистика декодера по неисправимым кадрам* – секция для отображения распределения неисправимых кадров по причинам невозможности их исправления:
 - Число кадров, для которых не найден полином локаторов ошибок.
 - Число кадров, для которых найдено некорректное количество локаторов.
 - Число кадров, для которых найдены некорректные локаторы (вне кадра).
 - Число кадров, исправление которых привело к недопустимому кадру (не являющемуся кодовым словом). Равно нулю, если декодер работает правильно.



Примеры работы с программой

Пример 1. Кодирование, искажение и восстановление информационного сообщения.

Введем некоторое исходное информационное сообщение в соответствующее поле редактирования информационного блока на стороне кодера.

Информационный блок на стороне кодера (размер блока: 78)

Координаты цели: 68° 52' 23" северной широты и 125° 32' 44" восточной долготы.

Выберем количество контрольных байтов, равное 50. Соответственно, мы сможем гарантированно исправлять искажения кратностью вплоть до 25 байтов.

Размер контрольного блока кадра в байтах: 50

Закодируем теперь сообщение, нажав кнопку «закодировать и передать кадр в среду», и получим следующий кадр:

Кадр в среде искажения (размер кадра: 128)

CA	EE	EE	F0	E4	E8	ED	E0	F2	FB	20	F6	E5	EB	E8	3A	20	36	38	B0	20	35	32	27	20	32	33	22	20	F1	E5	E2
E5	F0	ED	EE	E9	20	F8	E8	F0	EE	F2	FB	20	E8	20	31	32	35	B0	20	33	32	27	20	34	34	22	20	E2	EE	F1	F2
EE	F7	ED	EE	E9	20	E4	EE	EB	E3	EE	F2	FB	2E	A3	78	7E	05	F1	B4	64	27	58	A6	B4	F7	CE	42	CC	22	0E	75
E9	B2	05	FA	2A	43	75	8C	26	F5	2E	59	A6	BD	D2	5F	C6	5B	F1	9C	67	B8	D0	B5	DB	71	F4	62	60	A4	9D	B1

Кроме того, в секции дополнительных функций мы также можем запустить окно для отображения порождающего полинома, который был сформирован кодером и использован для кодирования информационного сообщения:

Коэффициенты порождающего полинома: G[0]...G[50]

1	98	3	128	195	229	86	201	228	19	189	99	136	85	153	18
30	103	104	45	233	76	165	213	235	170	149	160	156	194	233	138
32	121	204	18	152	70	167	249	189	43	80	51	38	229	145	230
204	243	1													

Попробуем декодировать кадр, пока не внося в него никаких искажений. Для этого сразу после кодирования нажимаем кнопку «принять кадр из среды и декодировать». В силу отсутствия искажений в кадре, в строке состояния программы получаем сообщение:

Декодер: Синдром ошибок равен нулю. Кадр не содержит ошибок.

Декодер сообщает, что синдром ошибок для принятого кадра равен нулю.

Теперь попробуем использовать генератор искажений. Для начала введем количество байтов, которые будут случайным образом искажены. Выберем количество искажаемых байтов: 25 – это предельное количество искажений, которые еще могут быть гарантированно исправлены при 50 контрольных байтах.

Количество искажаемых байтов в кадре: 25

После этого запустим генератор случайных искажений, нажав кнопку «внести искажение в кадр». В результате кадр искажается и принимает следующий вид:

Кадр в среде искажения (размер кадра: 128)

01	FB	EE	F0	E4	E8	ED	E0	F2	FB	79	F6	E5	E3	E8	3A	20	30	AD	B0	20	35	32	E3	F4	32	09	22	20	F1	E5	E2
E5	F0	ED	EE	E9	20	F8	E8	F0	EE	68	FB	F3	E8	20	31	55	35	B0	20	33	32	EF	20	34	34	22	20	11	EE	25	F2
EE	F7	ED	EE	E9	20	D0	EE	EB	E3	EE	F2	FB	2E	A3	78	7E	05	F1	B4	64	27	58	7E	B4	F7	A8	42	CC	22	0E	75
E9	2C	05	FA	2A	43	75	8C	26	F5	2E	59	A6	BD	D2	5F	C6	5B	41	3B	67	B8	D0	B5	DB	D2	90	C5	60	A4	9D	A2

Кроме того, также имеем информационный блок кадра в искаженном виде:

Информационный блок в среде искажения

Координаты цели: 0° 52' 21" северной широты и 105° 32' 44" восточной долготы.

Теперь попробуем декодировать кадр, нажав кнопку «принять кадр из среды и декодировать». Получаем предполагаемый вектор искажений следующего вида:

Вычисленный вектор ошибок на стороне декодера (вычисленное количество искаженных байт: 25)

CB	15	00	00	00	00	00	00	00	00	59	00	00	08	00	00	00	06	95	00	00	00	00	C4	D4	00	3A	00	00	00	00	00
00	00	00	00	00	00	00	00	00	00	9A	00	D3	00	00	00	67	00	00	00	00	00	C8	00	00	00	00	00	F3	00	D4	00
00	00	00	00	00	00	00	34	00	00	00	00	00	00	00	00	00	00	00	00	00	00	D8	00	00	66	00	00	00	00	00	00
00	9E	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	B0	A7	00	00	00	00	A3	64	A7	00	00	00	13	

Как видим, декодер правильно распознал количество искаженных байтов, точно определил их местонахождение и характер искажений. В итоге имеем успешно восстановленное информационное сообщение после исправления кадра декодером:

Информационный блок на стороне декодера

Координаты цели: 68° 52' 23" северной широты и 125° 32' 44" восточной долготы.

Кроме того, для детального анализа всех этапов декодирования кадра, в секции дополнительных функций мы можем запустить окна для отображения вычисленного синдрома ошибок, найденного полинома локаторов ошибок, найденных локаторов ошибок, вычисленных полиномов величин ошибок и формальной производной, вычисленных величин ошибок, а также сформированного полинома ошибок.

Имеем синдром ошибок и видим, что он не равен нулю, что свидетельствует о том, что принятый кадр содержит ошибки (подвергнут искажениям):

Компоненты синдрома: S[1]...S[50]															
10	94	81	156	192	254	74	1	77	60	19	150	2	106	189	195
210	88	4	27	103	171	66	46	193	18	41	130	44	52	152	236
34	41	20	111	250	134	151	114	136	144	28	102	67	10	94	114
181	133														

Также имеем найденный декодером полином локаторов ошибок:

Коэффициенты полинома локаторов: L[0]...L[25]															
1	238	176	67	110	144	34	73	190	51	133	219	24	214	42	16
225	67	181	17	123	208	46	219	104	158						

Также имеем вычисленные локаторы ошибок, заметим, что декодер сделал правильное предположение о количестве искаженных байтов (25):

Локаторы в порядке возрастания: LOC[1]...LOC[25]															
0	4	5	6	12	13	30	37	40	57	65	67	73	79	83	85
101	103	104	109	110	114	117	126	127							

Также имеем вычисленные полиномы величин ошибок и формальной производной:

Коэффициенты полинома величин: O[0]...O[24]															
10	188	251	47	35	218	12	126	123	63	72	73	197	230	104	44
16	223	244	193	63	83	28	139	85							

Коэффициенты полинома-производной: DL[0]...DL[24]															
238	0	67	0	144	0	73	0	51	0	219	0	214	0	16	0
67	0	17	0	208	0	219	0	158							

Также имеем вычисленные величины ошибок:

Величины в порядке соответствующих им локаторов: VAL[1]...VAL[25]															
19	167	100	163	167	176	158	102	216	52	212	243	200	103	211	154
58	212	196	149	6	8	89	21	203							

Наконец, также имеем сформированный полином ошибок:

Коэффициенты полинома ошибок: E[0]...E[127]															
19	0	0	0	167	100	163	0	0	0	0	0	167	176	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	158	0
0	0	0	0	0	102	0	0	216	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	52	0	0	0	0	0	0
0	212	0	243	0	0	0	0	0	200	0	0	0	0	0	103
0	0	0	211	0	154	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	58	0	212	196	0	0	0	0	149	6	0
0	0	8	0	0	89	0	0	0	0	0	0	0	21	203	

Примечание. Если же мы бы внесли большее количество искажений (26 и выше), декодер бы вывел в строке состояния программы одно из сообщений о невозможности исправления кадра, например:

Декодер: Не удалось найти полином локаторов ошибок.

Пример 2. Допустим, что при проектировании некоторой информационной системы, включающей в себя подсистему аутентификации пользователей, ее разработчики легкомысленно использовали коды Рида-Соломона в чистом виде для вычисления значений «хэш» для паролей пользователей, не подозревая о корректирующей способности кодов Рида-Соломона. Допустим, что злоумышленнику удалось похитить базу данных, хранящую хэш паролей или перехватить их по сети, причем длина хэш ему известна и составляет 32 байта, и ему также известно, что используются коды Рида-Соломона в чистом виде.

Тогда, учитывая, что 32 контрольных байта позволяют исправлять искажения кратностью вплоть до 16 байтов, можно говорить о том, что если пароль имеет длину не более 16 символов, то он может быть однозначно декодирован с помощью программы.

Таким образом, можно положиться наудачу (предположить, что пароль не более 16 символов) и попробовать декодировать его. Делается это с помощью программы следующим образом, выбирается некоторая предполагаемая длина пароля в пределах от 1 до 16, и в поле информационного блока вводится соответствующее количество произвольных символов. Устанавливается количество контрольных байтов равным 32, далее выполняется кодирование, после этого 32 байта контрольных байтов кадра (они всегда расположены в конце кадра) заменяются 32 байтами «хэш» исследуемого пароля. После этого делается попытка декодирования. Если не удалось успешно декодировать, то выбирается другая предполагаемая длина пароля и вся процедура повторяется, пока либо пароль не будет успешной декодирован, либо все варианты длин будут перебраны, и можно будет сделать вывод о том, что пароль имеет длину больше, чем 16 символов.

Допустим, для некоторого неизвестного пароля имеется его «хэш»: **54 81 9A 7D 14 7F D3 A9 8E 29 5D 1A B0 3F 3B 27 FD 98 C1 18 6A 4B 3B 20 74 53 DD 66 B4 9D B5 34.**

Учитывая, что хэш 32-байтный, при 32 контрольных байтах декодер может исправлять до 16 искаженных байтов, начнем сразу с длины 16 с целью сокращения поиска.

Вводим в поле информационного блока на стороне кодера 16 произвольных символов, например «звездочки»:

Информационный блок на стороне кодера (размер блока: 16)

XXXXXXXXXXXXXXXX

Выбираем количество контрольных байтов, равное 32.

Размер контрольного блока кадра в байтах: 32

Кодируем сообщение, используя кнопку «закодировать». Получаем следующий кадр:

Кадр в среде искажения (размер кадра: 48)																																			
2A	2A	2A	2A	2A	2A	2A	2A	2A	2A	2A	2A	2A	2A	2A	2A	FC	0A	4D	CE	02	54	A8	69	63	E1	77	C3	24	A8	37	79				
72	DD	68	DD	EC	8B	15	B4	40	26	C6	05	E7	77	5F	6C																				

Теперь непосредственное редактируя поле кадра, заменяем контрольные байты кадра (последние 32 байта) на вышеуказанные байты «хэша» исследуемого пароля:

Кадр в среде искажения (размер кадра: 48)																																
2A	2A	2A	2A	2A	2A	2A	2A	2A	2A	2A	2A	2A	2A	2A	2A	54	81	9A	7D	14	7F	D3	A9	8E	29	5D	1A	B0	3F	3B	27	
FD	98	C1	18	6A	4B	3B	20	74	53	DD	66	B4	9D	B5	34																	

Таким образом, заменив все контрольные байты, мы тем самым исказили последние 32 байта кадра, и, такое искажение, конечно же, не может быть корректно исправлено, но все зависит от точки зрения. Мы попытаемся представить ситуацию в ином свете, что наоборот,

