

---

## Лабораторная работа № 3

### Помехоустойчивый код Хэмминга

#### 1. Цель работы

Изучить принципы помехоустойчивого кодирования, получить навыки моделирования помехоустойчивых кодов с помощью Electronics Workbench (EWB) или Multisim.

#### 2. Общие сведения

При работе устройств вычислительной техники и телекоммуникационной аппаратуры возможно появление ошибок в обрабатываемых цифровых данных. Причинами сбоев могут быть мощные электромагнитные помехи, резкое изменение напряжения питания, старение радиоэлементов, ненадёжный контакт разъёмов, радиоактивное излучение естественных и искусственных источников и т.п. Сбои проявляются в виде случайного изменения одного или нескольких битов машинного слова (вместо единицы в отдельных разрядах передаётся ноль или наоборот).

Автоматическое обнаружение и исправление ошибок сопровождается введением избыточности в передаваемые или хранимые данные. Для этих целей разработаны специальные коды, в которые помимо информационных битов  $b_1b_2\dots b_n$  дополнительно вводят контрольные (проверочные) биты  $k_1k_2\dots k_m$ . Контрольные биты позволяют проверять целостность (не искажённость) информационных битов машинного слова, а наиболее сложные коды могут не только обнаружить, но и исправить неверно принятые биты.

Разработанные помехоустойчивые коды позволяют решать разные задачи: обнаружить одиночную ошибку, обнаружить и исправить единственную ошибку, обнаружить и исправить несколько ошибок. Первые коды называются обнаруживающими, а вторые – корректирующими кодами.

На передающей стороне формирование помехоустойчивого кода осуществляют с помощью специального устройства – **кодера**. Обнаружение и исправление ошибок на приёмной стороне производят с помощью устройств, которые называются **декодерами**. В данной лабораторной работе кодер и декодер строят с помощью комбинационных цифровых устройств (логические элементы, которые выполняют функцию неравнозначности).

### 3. Задания на выполнение лабораторной работы

#### 3.1. Задание 1. Формирование бита чётности

Сформировать бит чётности (бит паритета) для заданного байта передаваемых данных. При выполнении задания нужно в правую крайнюю колонку таблицы 3.1.1 записать единицу или ноль.

Таблица 3.1.1

Вариант	Байт	Бит паритета
1	10101011	
2	10101100	
3	10101101	
4	10101110	
5	10101111	
6	10110001	
7	10110010	
8	10110011	
9	10110100	
10	10110101	
11	10110110	
12	10110111	
13	10111000	
14	10111001	
15	10111010	
16	10111011	

#### 3.2. Задание 2. Исследование помехоустойчивого кода с формированием бита чётности

Выполнить моделирование процесса передачи информации (одной тетрады - половины байта). Исходные данные приведены в таблицах 3.2.1 и 3.2.2. Для моделирования использовать программы Electronics Workbench (EWB) или Multisim.

Таблица 3.2.1

Вариант	Тетрада
1	0000
2	0001
3	0010
4	0011
5	0100
6	0101
7	0110
8	0111
9	1000
10	1001
11	1010
12	1011
13	1100
14	1101
15	1110
16	1111

Моделирование следует выполнить четырежды при заданных значениях помех (табл.3.2.2). Результаты моделирования в виде таблицы поместите в отчёт. Прокомментируйте полученные результаты.

Таблица 3.2.2

Вариант	$S_8S_7S_6S_5$	$S_8S_7S_6S_5$	$S_8S_7S_6S_5$	$S_8S_7S_6S_5$
1	0000	1000	0011	1110
2	0000	0100	0110	0111
3	0000	0010	1100	1011
4	0000	0001	1001	1101
5	0000	1000	0011	1110
6	0000	0100	0110	0111
7	0000	0010	1100	1011
8	0000	0001	1001	1101
9	0000	1000	0011	1110
10	0000	0100	0110	0111
11	0000	0010	1100	1011
12	0000	0001	1001	1101
13	0000	1000	0011	1110
14	0000	0100	0110	0111
15	0000	0010	1100	1011
16	0000	0001	1001	1101

### 3.3. Задание 3. Исправление ошибки с помощью кода Хэмминга

Расчётным путём (вручную) определить, в каком разряде принятого кода Хэмминга произошло искажение. Исходные данные для разных вариантов приведены в таблице 3.3.1. Процесс вычисления искажённого бита следует подробно описать в отчёте.

Таблица 3.3.1

Вар.	$b_8$	$b_7$	$b_6$	$b_5$	$k_8$	$b_4$	$b_3$	$b_2$	$k_4$	$b_1$	$k_2$	$k_1$
1	1	0	0	1	1	0	0	1	0	0	1	1
2	1	0	0	0	0	0	1	0	0	1	1	1
3	1	1	1	0	1	0	1	1	0	0	1	1
4	1	1	0	0	0	1	1	0	0	1	1	0
5	1	1	0	1	0	0	0	1	0	1	0	1
6	1	1	1	0	0	1	0	0	0	1	1	1
7	1	0	0	1	1	0	1	0	0	0	0	0
8	0	0	0	0	1	0	1	1	1	1	0	0
9	1	1	1	0	1	0	0	0	1	0	1	0
10	1	0	1	0	0	0	0	0	0	1	0	0
11	1	0	1	0	0	1	1	1	1	0	0	0
12	1	0	0	0	0	0	1	1	1	0	0	0
13	1	1	0	0	0	1	1	0	1	0	1	0
14	1	1	0	0	1	0	0	1	0	0	0	0
15	1	0	1	0	1	1	0	0	0	0	1	0
16	0	0	1	0	0	1	1	0	1	0	1	1

### 3.4. Задание 4. Моделирование работы кода Хэмминга

С помощью программ Electronics Workbench или Multisim собрать схему для моделирования процесса передачи информации с использованием помехоустойчивого кодирования. Провести моделирование процесса передачи данных, приведённых в табл. 3.3.1 (для своего варианта). В отчёт следует поместить схему, соответствующую заданному варианту (в том числе с указанием положения ключей и содержимого генератора слов).

## 4. Порядок выполнения лабораторной работы

### 4.1. Методические указания к заданию 3.1

Простейший код, предназначенный для обнаружения одной ошибки (точнее – для обнаружения нечётного числа ошибок), основан на добавлении к информационным битам одного контрольного бита. При этом контрольный бит должен быть таким, чтобы суммарное число единиц в образованном машинном слове было чётным. Добавляемый бит называется **битом паритета**.

Проверочный бит  $k$  для  $n$ -битного двоичного слова  $b_1b_2\dots b_n$  вычисляется по формуле:

$$k = \begin{cases} 1, & \text{если } b_1 \oplus b_2 \oplus \dots \oplus b_n = 1 \\ 0, & \text{если } b_1 \oplus b_2 \oplus \dots \oplus b_n = 0 \end{cases}$$

В результате такого преобразования формируется  $(n+1)$  – битное слово  $b_1b_2\dots b_nk$ , число единиц в котором будет чётное.

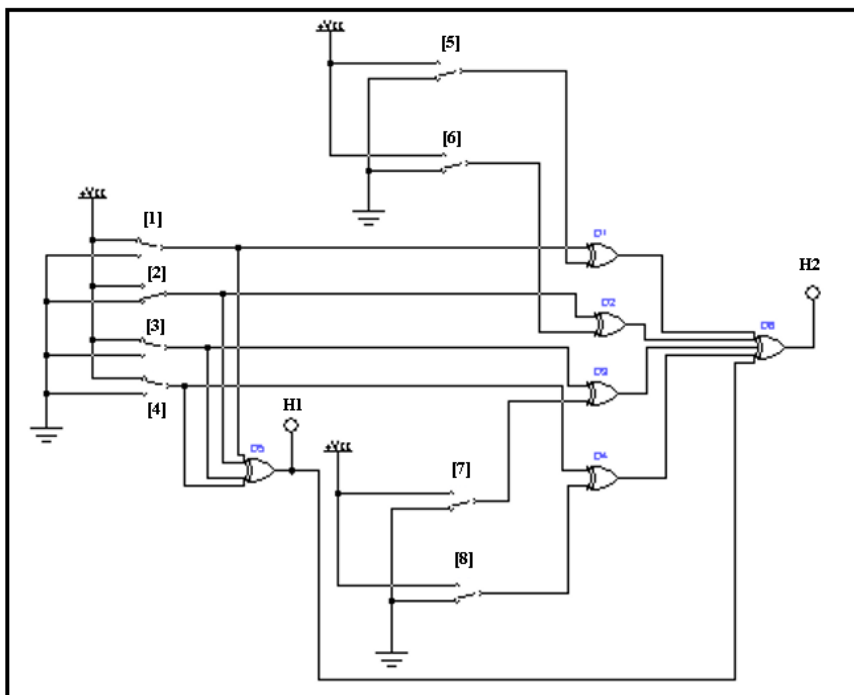
В задании 3.1 требуется для своего варианта определить необходимый бит паритета, который нужно добавить к информационным битам.

*Пример.*

Пусть дан байт 10111100. Число информационных единиц в этом байте нечётное, поэтому бит паритета нужно установить равным единице. В результате этого получается машинное слово 101111001.

## 4.2. Методические указания к заданию 3.2

Схема исследований кода с формированием бита паритета показана на рисунке.



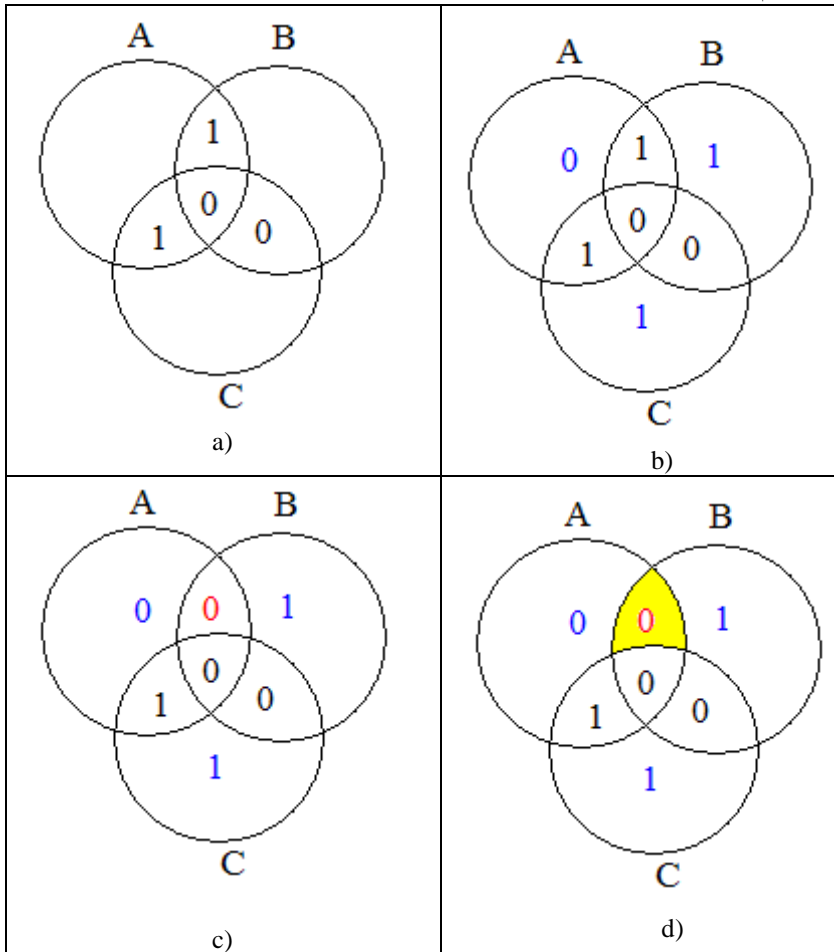
Переключатели S1...S4 имитируют передаваемую тетраду (четыре информационных бита). Переключатели S5...S8 симулируют помехи, возникающие при передаче или хранении цифровых данных. Нижнее положение переключателей S5...S8 соответствует отсутствию искажений в соответствующем разряде. Схема D5 формирует бит чётности (если на вход подаётся чётное число единиц, то выходной сигнал равен нулю). При этом светодиод H1 не горит.

Логическая схема Иключающее ИЛИ D6 на приёме осуществляет контроль искажений. Если светодиод H2 не горит, то считается, что искажений нет. Фактически это не так. Появление чётного числа искажений в машинном слове не фиксируется (не обнаруживается) данным устройством. Это является существенным недостатком данного метода контроля искажений. Такой код способен лишь сигнализировать о наличии редко появляющихся одиночных сбоев (точнее, нечётного числа сбоев).

**4.3. Методические указания к заданию 3.3**

Проиллюстрируем идею обнаружения и определения неверно принятого бита с помощью кода Хэмминга. Для этого используем диаграммы Вена [1]. Предположим, что передаётся тетрада 1010. Запишем эти четыре бита в сегменты трёх пересекающихся окружностей.

Таблица 4.3.1



Окружности A, B и C дают семь сегментов. В четыре внутренних сегмента поместим информационные биты числа 1010 (табл. 4.3.1 а). В оставшиеся три сегмента поместим контрольные биты (табл. 4.3.1 б).

Правило формирования контрольных битов такое: в каждой ок-

окружности должно быть чётное число единиц. В данном случае в каждой окружности получилось по две единицы. Пусть в процессе передачи информации один информационный бит будет искажён (табл. 4.3.1 с). На приёмной стороне осуществляется анализ принятой информации. Легко заметить, что в окружности С число единиц осталось чётным, а окружностях А и В число единиц стало нечётным. Это говорит о том, что искажённый бит находится в сегменте, который принадлежит окружностям А и В, но не принадлежит окружности С (табл. 4.3.1 d).

Рассмотрим пример нахождения искажённого бита с помощью кода Хэмминга.

Места расположения информационных битов (ИБ) и контрольных битов (КБ) в передаваемых данных указаны в следующей таблице. В верхней строке таблицы указан порядковый номер каждого бита в машинном слове.

Таблица 4.3.2.

№ раз.	12	11	10	9	8	7	6	5	4	3	2	1
ИБ	$b_8$	$b_7$	$b_6$	$b_5$		$b_4$	$b_3$	$b_2$		$b_1$		
КБ					$k_8$				$k_4$		$k_2$	$k_1$

Форма записи машинного слова, приведённая в предыдущей таблице, выбрана такой с целью повышения наглядности (из методических соображений). Фактически данные представляют машинным словом, состоящим из 12-ти бит:  $b_8b_7b_6b_5k_8b_4b_3b_2k_4b_1k_2k_1$ .

*Пример.*

Предположим, что в процессе передачи некоторых данных произошло искажение одного информационного бита и на приёме получены указанные в таблице 4.3.3 данные. Требуется найти и исправить искажённый информационный бит.

Таблица 4.3.3.

Раз-ряд	12	11	10	9	8	7	6	5	4	3	2	1
Слово	$b_8$	$b_7$	$b_6$	$b_5$	$k_8$	$b_4$	$b_3$	$b_2$	$k_4$	$b_1$	$k_2$	$k_1$
ИБ	1	0	0	0		1	1	0		1		
КБ					0				1		0	0



*Решение.*

Вычислим значения контрольных битов на приёме. Будем обозначать проверочные биты на приёме со штрихом (чтобы отличить их от контрольных битов, сформированных на передающей стороне). Расчёт производится по формулам [1]:

$$\begin{aligned}k_1^{\wedge} &= b_1 \oplus b_2 \oplus b_4 \oplus b_5 \oplus b_7; \\k_2^{\wedge} &= b_1 \oplus b_3 \oplus b_4 \oplus b_6 \oplus b_7; \\k_4^{\wedge} &= b_2 \oplus b_3 \oplus b_4 \oplus b_8; \\k_8^{\wedge} &= b_5 \oplus b_6 \oplus b_7 \oplus b_8.\end{aligned}\quad (4.3.1)$$

Используя формулу (4.3.1) и верхнюю строчку таблицы 4.3.2, получим конкретные значения контрольных битов для рассматриваемого примера на приёме:

$$\begin{aligned}k_1^{\wedge} &= 1 \oplus 0 \oplus 1 \oplus 0 \oplus 0 = 0; \\k_2^{\wedge} &= 1 \oplus 1 \oplus 1 \oplus 0 \oplus 0 = 1; \\k_4^{\wedge} &= 0 \oplus 1 \oplus 1 \oplus 1 = 1; \\k_8^{\wedge} &= 0 \oplus 0 \oplus 0 \oplus 1 = 1.\end{aligned}\quad (4.3.2)$$

Результаты расчётов (4.3.2) показывают, что контрольные биты, сформированные на передающей и приёмной сторонах, различаются:

$$k_1^{\wedge} = k_1; \quad k_2^{\wedge} \neq k_2; \quad k_4^{\wedge} = k_4; \quad k_8^{\wedge} \neq k_8.$$

Различие контрольных битов, сформированных на передающей и приёмной сторонах, говорит о том, что в процессе передачи произошло искажение машинного слова. Теперь необходимо определить, какой именно бит был принят неверно.

Для определения неверно принятого бита требуется вычислить так называемый синдром  $S = s_8 s_4 s_2 s_1$ , где

$$s_1 = k_1^{\wedge} \oplus k_1; \quad s_2 = k_2^{\wedge} \oplus k_2; \quad s_4 = k_4^{\wedge} \oplus k_4; \quad s_8 = k_8^{\wedge} \oplus k_8. \quad (4.3.3)$$

Используя результаты (4.3.2) и нижнюю строчку таблицы 4.3.3, вычислим для рассматриваемого примера четыре бита синдрома:

$$s_1 = 0 \oplus 0 = 0; \quad s_2 = 1 \oplus 0 = 1; \quad s_4 = 1 \oplus 1 = 0; \quad s_8 = 1 \oplus 0 = 1.$$

Переведём синдром  $S = 1010_2$  из двоичной системы счисления (СС) в десятичную СС  $S = 10_{10}$ . Десятичное число 10 говорит о том, что десятый разряд принятых данных ( $b_6$ ) искажён, и этот бит нужно исправить (проинвертировать). Таким образом, после корректировки принятые дан-

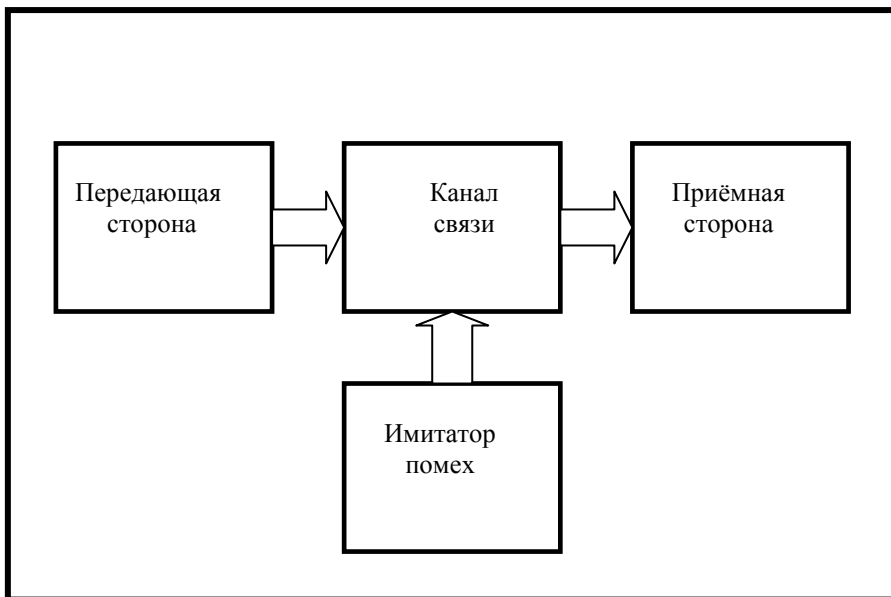
ные будут иметь вид, показанный в таблице 4.3.4. Напомним, что счёт разрядов ведётся справа налево.

Таблица 4.3.4.

Раз-ряд	12	11	<b>10</b>	9	8	7	6	5	4	3	2	1
Слово	$b_8$	$b_7$	$b_6$	$b_5$	$k_8$	$b_4$	$b_3$	$b_2$	$k_4$	$b_1$	$k_2$	$k_1$
ИБ	1	0	<b>1</b>	0		1	1	0		1		
КБ					0				1		0	0

#### 4.4. Методические указания к заданию 3.4

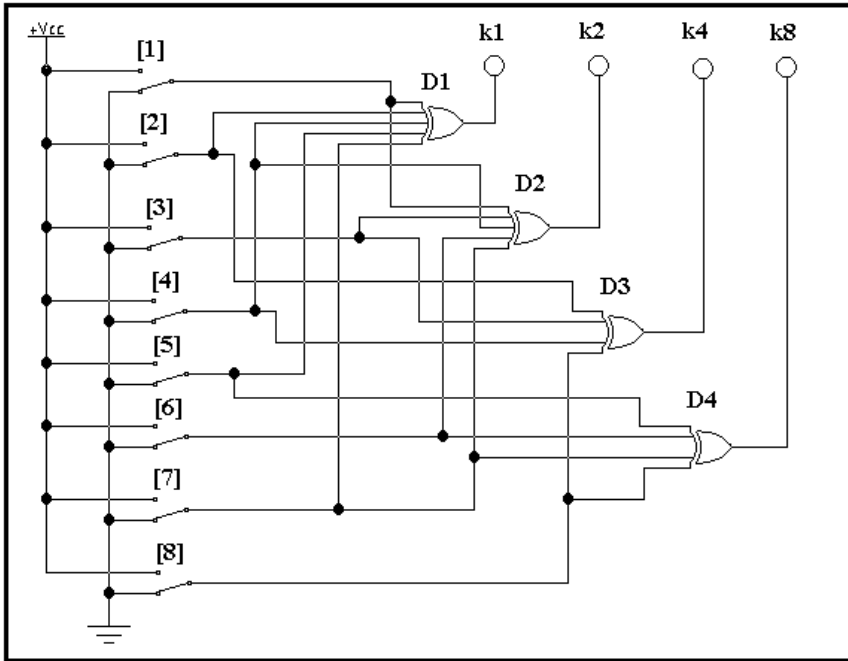
Структурная схема устройства для моделирования передачи данных с использованием кода Хэмминга показана на рисунке.



На передающей стороне формируются контрольные биты в соответствии с выражением (4.3.1). На приёмной стороне вычисляется синдром в соответствии с выражениями (4.3.1) и (4.3.3). Имитатор помех позволяет исказить любой бит данных, передаваемых по каналу связи.

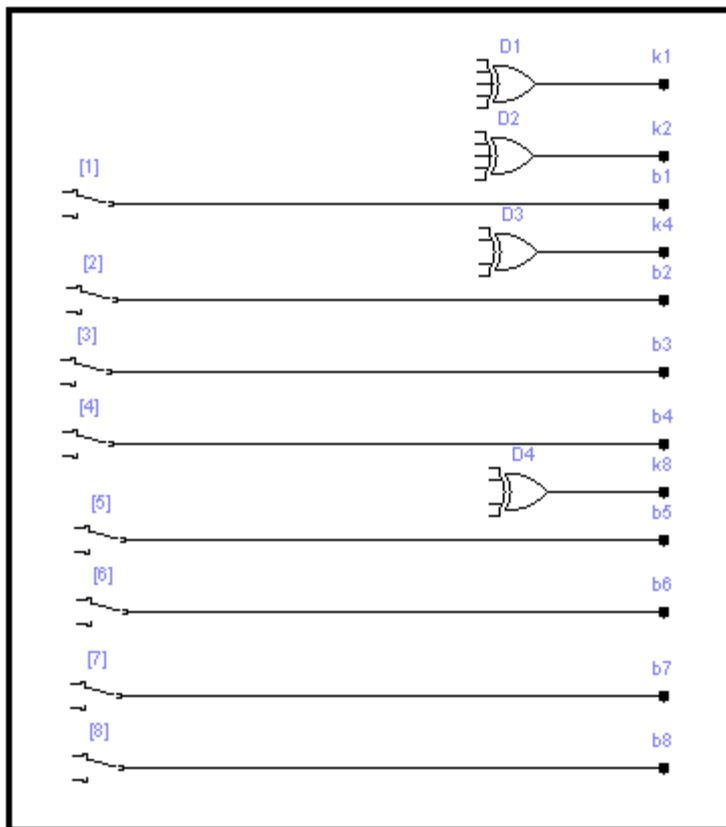
Рассмотрим поочерёдно конструкцию каждого блока, указанного на структурной схеме.

На следующем рисунке показана схема Передатчика (кодера).



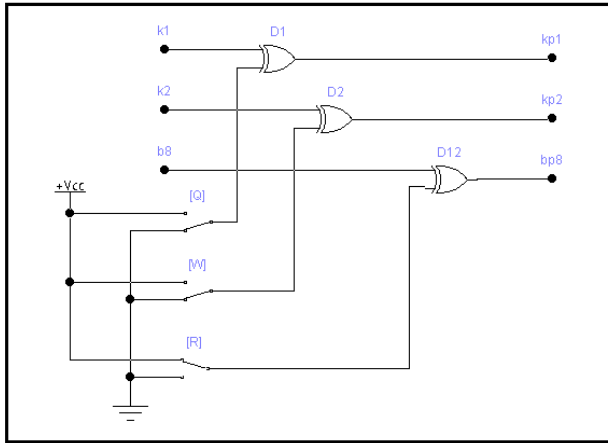
Переключатели 1...8 имитируют информационные биты передаваемых данных. С помощью четырёх логических элементов Иключающее ИЛИ D1...D4 формируются контрольные биты  $k_1, k_2, k_4, k_8$ . В канал связи передаётся 12 бит (восемь информационных и четыре контрольных бита). Схематично передаваемые по каналу связи данные показаны на следующем рисунке.

Заметим, что в устройствах телекоммуникаций информация передаётся не с помощью двенадцати проводной линии (параллельный код), а преобразуется в последовательный код, например, с помощью регистра сдвига.

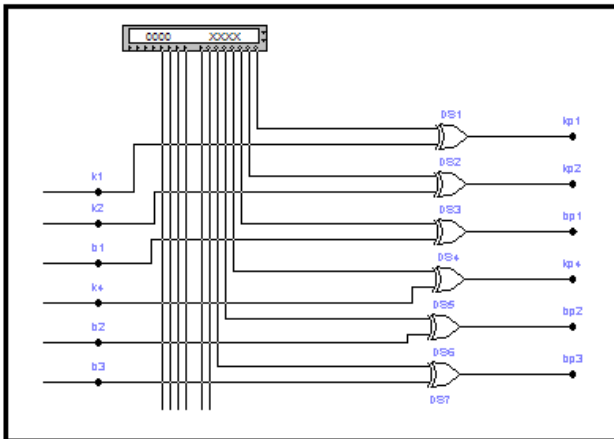


Имитатор помех позволяет исследователю моделировать возникновение сбоев в любом разряде передаваемых данных. Конструкция имитатора помех сходна с конструкцией блока формирования помех. В имитаторе помех содержится 12 логических схем Иключающее ИЛИ и 12 переключателей, с помощью которых можно изменить любой бит. Фрагмент этого блока показан на следующем рисунке. Переключатели Q, W, R могут изменять значения битов (соответственно  $k_1, k_2, b_8$ ). Нижнее положение переключателей не изменяет значения передаваемых битов, а верхнее положение приводит к инверсии соответствующего бита. Указанное положение переключателей Q, W и R приведёт к тому, что биты  $k_1, k_2$  будут переданы без искажений ( $k_{p1} = k_1, k_{p2} = k_2$ ), а бит  $k_8$  будет проинвертирован

$$(k_{p8} = \bar{k}_8).$$

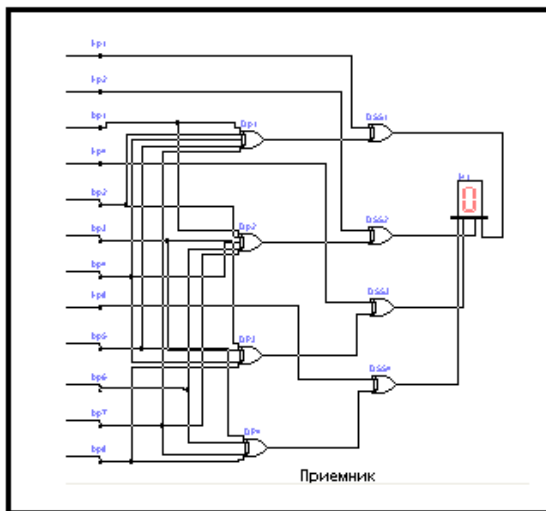


Рассмотренная конструкция имитатора помех не единственная. Этот блок можно построить иначе, например, с помощью генератора слов (см. следующий рисунок). Если генератор слов (Word Generator) формирует во всех разрядах логические нули, то искажений не происходит. Естественно, что логическая единица в каком-либо разряде управляющего слова приводит к искажению соответствующего бита (инверсии).



На приёмной стороне поступившие данные должны быть обработа-

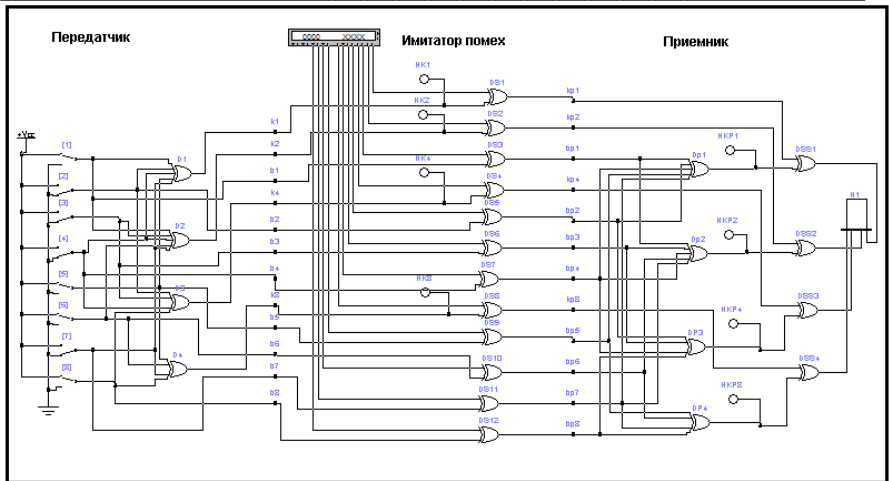
ны так, чтобы можно было автоматически исправить искажённый бит. На следующем рисунке показана принципиальная схема Приёмника (декодер).



Четыре логических элемента DP1...DP4 производят расчёт контрольных битов на приёме. Следующие четыре логические схемы Иключающее ИЛИ DSS1...DSS4 вычисляют синдром. Значение синдрома в шестнадцатеричной СС отображается на индикаторе И1. Если индикатор показывает нуль, то это означает, что передача данных произошла без искажений.

При наличии сбоев в канале связи показания индикатора изменяются от 1 до СН. Данная конструкция приёмника только указывает номер искажённого разряда, но коррекции принятых данных не производит. Естественно, что путём некоторого усложнения схемы эту опцию можно реализовать.

Полная принципиальная схема устройства моделирования показана на последнем рисунке. Восемь светодиодов позволяют анализировать работу устройства в местах формирования контрольных битов.



## 5. Требования к отчёту

Отчёт подготавливается в электронном виде. Он должен содержать постановки задач, результаты расчётов контрольных битов, схемы моделирования.

## 6. Контрольные вопросы

6.1. Можно ли исправить возникшую ошибку в передаваемых данных, если для помехоустойчивого кодирования используется лишь единственный бит паритета?

6.2. Можно ли исправить возникшую ошибку в передаваемых данных, если для помехоустойчивого кодирования используется лишь единственный бит паритета?

6.3. Перечислите программы, предназначенные для моделирования радиоэлектронных устройств.

6.4. Запишите формулы для формирования контрольных битов с помощью кода Хэмминга.

6.5. Нужно ли исправлять искажённые контрольные биты?

6.6. Как рассчитывается слово синдрома в коде Хэмминга?

6.7. К чему приведёт искажение контрольного бита при использовании кода Хэмминга?

6.8. Как составить схему цифрового устройства по заданному логическому выражению?

6.9. Какое действие выполняет декодер кода Хэмминга?

1.Цилькер, Б.Я. Организация ЭВМ и систем [Текст]: учеб. для вузов / Б. Я. Цилькер, С. А. Орлов, - СПб: Питер, 2011. - 687 с.