

## ИННОВАЦИОННАЯ ОБРАЗОВАТЕЛЬНАЯ ПРОГРАММА



**Проект 2:** индивидуальная траектория обучения и качество образования

**Цель:** ориентированное на требования рынка образовательных услуг улучшение качества подготовки и переподготовки специалистов

Федеральное агентство по образованию  
Государственное образовательное учреждение  
высшего профессионального образования  
Владимирский государственный университет  
Кафедра вычислительной техники

# МЕТОДИЧЕСКИЕ УКАЗАНИЯ К ПРАКТИЧЕСКИМ ЗАНЯТИЯМ ПО ДИСЦИПЛИНЕ «ТЕОРИЯ ЭЛЕКТРИЧЕСКОЙ СВЯЗИ»

Составитель  
Л.А. КАЛЫГИНА

Владимир 2008

УДК 621.39(07)  
ББК 32.88-01 я 2  
М54

Рецензент

Кандидат технических наук, доцент  
Владимирского государственного университета  
*С.Ю. Кириллова*

Печатается по решению редакционного совета  
Владимирского государственного университета

**Методические** указания к практическим занятиям по дисциплине «Теория электрической связи» / Владим. гос. ун-т ; сост. Л.А. Кальгина. – Владимир : Изд-во Владим. гос. ун-та, 2008. – 30 с.

Рассматриваются вопросы теории передачи и кодирования сообщений, алгоритмы преобразования сигналов в системах связи. В каждой теме приводятся примеры и задачи. Содержатся теоретические вопросы по курсу.

Предназначены для студентов специальности 210402 – Средства связи с подвижными объектами при изучении курса «Теория электрической связи», могут быть полезны студентам, аспирантам, специалистам, занимающимся вопросами проектирования и анализа систем телекоммуникаций.

Табл. 7. Ил. 8. Библиогр.: 2 назв.

УДК 621.39 (07)  
ББК 32.88-01 я 2

## ПРЕДИСЛОВИЕ

Курс «Теория электрической связи» входит в учебный план подготовки инженеров по специальности 210402 – Средства связи с подвижными объектами. Курс изучают в цикле общепрофессиональных дисциплин в четвертом и пятом учебных семестрах.

Второй семестр дисциплины посвящен изучению основных методов и алгоритмов обработки сигналов в цифровых системах связи, состоит из четырех разделов:

- кодирование источников;
- характеристики сигналов и систем связи;
- оптимальный прием для канала с аддитивным белым гауссовым шумом;
- помехоустойчивое кодирование.

Материалы практических занятий связаны с лекционным курсом, призваны помочь освоению основных понятий теории связи.

В первую часть практических занятий включены вопросы определения количества информации источника, алгоритмы кодирования для дискретных источников, понятия скорости и эффективности кодирования источников информации в системах связи.

Во второй части рассматриваются математическая модель сигнала, который может быть представлен в базисе ортонормированных функций, процедура нахождения ортонормированного набора функций для сигнала с ограниченной энергией. Освоение этого материала необходимо при изучении оптимального приема сигналов цифровой модуляции.

В третьей части рассматриваются алгоритмы помехоустойчивого кодирования и декодирования, основные характеристики помехоустойчивых кодов.

Заключительная часть методических указаний содержит вопросы к трем контрольным работам, проводимым в течение семестра. Эти вопросы отражают основное содержание лекционного курса.

# 1. КОДИРОВАНИЕ ИСТОЧНИКОВ

## 1.1. Логарифмическая мера информации

$X$  и  $Y$  – дискретные случайные величины. Возможные значения случайных величин  $x_i, i = 1, 2, \dots, n$  и  $y_j, j = 1, 2, \dots, m$ .

Взаимная информация  $I(x_i, y_j)$  между  $x_i$  и  $y_j$  – это количество информации, которую даёт наблюдение события  $Y = y_j$  относительно события  $X = x_i$ :

$$I(x_i, y_j) = \log \frac{P(x_i|y_j)}{P(x_i)},$$

где  $P(x_i|y_j)$  – условная вероятность  $x_i$  при наличии  $y_j$ ,  
 $P(x_i)$  – априорная вероятность  $x_i$ .

Собственная информация события

$$P(x_i) = \log P(x_i).$$

### Пример 1.1

$X$  и  $Y$  – двоичные  $\{0, 1\}$  случайные величины,  $X$  – вход,  $Y$  – выход канала с двоичным входом и двоичным выходом. Входные символы равновероятны  $P(X = 0) = P(X = 1) = 1/2$ , условные вероятности выходных символов при заданном входе задаются выражениями:

$$P(Y = 0 | X = 0) = 1 - p_0,$$

$$P(Y = 1 | X = 0) = p_0,$$

$$P(Y = 0 | X = 1) = 1 - p_1,$$

$$P(Y = 1 | X = 1) = p_1.$$

Необходимо определить, сколько информации о событии  $X = 0$  и  $X = 1$  содержится в событии  $Y = 0$ .

*Решение:* событие  $Y = 0$  наблюдается, если при входе 0, на выходе – 0 и при входе 1, на выходе – 0.

$$\begin{aligned} P(Y = 0) &= P(Y = 0 | X = 0) \cdot P(X = 0) + P(Y = 0 | X = 1) \cdot P(X = 1) = \\ &= (1 - p_0) \frac{1}{2} + (1 - p_1) \frac{1}{2} = \frac{2 - p_0 - p_1}{2}. \end{aligned}$$

Событие  $Y = 1$  наблюдается, если при входе 0, на выходе – 1 и при входе 1, на выходе – 1.

$$\begin{aligned} P(Y = 1) &= P(Y = 1 | X = 0) \cdot P(X = 0) + P(Y = 1 | X = 1) \cdot P(X = 1) = \\ &= p_0 \frac{1}{2} + p_1 \frac{1}{2} = \frac{p_0 + p_1}{2}. \end{aligned}$$

Тогда взаимная информация о символе  $X = 0$  при условии, что наблюдается  $Y = 0$  равна

$$I(0; 0) = \log_2 \frac{P(Y = 0 | X = 0)}{P(Y = 0)} = \log_2 \frac{2(1 - p_0)}{2 - p_0 - p_1}.$$

Аналогично взаимная информация о символе  $X = 1$  при условии, что наблюдается  $Y = 0$  равна

$$I(1; 0) = \log_2 \frac{P(Y = 0 | X = 1)}{P(Y = 0)} = \log_2 \frac{2(1 - p_1)}{2 - p_0 - p_1}.$$

Средняя взаимная информация между  $X$  и  $Y$

$$I(X, Y) = \sum_{i=1}^n \sum_{j=1}^m P(x_i, y_j) I(x_i, y_j) = I(Y, X).$$

Средняя собственная информация (энтропия источника):

$$H(X) = \sum_{i=1}^n p(x_i) I(x_i) = - \sum_{i=1}^n p(x_i) \log P(x_i).$$

Если символы источника равновероятны, то энтропия источника равна  $H(X) = \log_2 n$ .

### *Пример 1.2*

Найти энтропию двоичного источника, который выдает последовательность независимых символов, причем выходной кодовый символ «0» с вероятностью  $q$ , а символ «1» с вероятностью  $1 - q$ .

По формуле энтропии источника для двух возможных символов ( $n = 2$ ):

$$H(X) = -q \log q - (1 - q) \log (1 - q).$$

## **1.2. Кодирование кодовыми словами фиксированной длины**

Имеется  $L$  возможных символов источника.

Число двоичных символов кодера (длина кодового слова) при уникальном (однозначном) кодировании или скорость кодирования:

$$R = \log_2 L,$$

если  $L$  равно целой степени основания 2,

$$\text{и } R = [\log_2 L] + 1,$$

если  $L$  не равно целой степени основания 2.

Эффективность кодирования определяется отношением  $H(X)/R$ .

Соотношение между скоростью кодирования и энтропией источника  $R \geq H(X)$ .

### Пример 1.3

Рассчитать эффективность кодирования кодами фиксированной длины, если количество символов кодирования  $L$  равно 4 для:

а)  $p_1 = p_2 = p_3 = p_4 = 1/4$ ;

б)  $p_1 = 1/2, p_2 = 1/4, p_3 = p_4 = 1/8$ .

Решение:

а) Длина кодового слова  $R = \log_2 4 = 2$ .

Так как символы равновероятны, энтропия источника  $H(X) = \log_2 4 = 2$ .

Эффективность кодирования  $H(X)/R = 1$ ;

б) Длина кодового слова  $R = \log_2 4 = 2$ .

Энтропия источника

$$H(X) = - (p_1 \log_2 p_1 + p_2 \log_2 p_2 + p_3 \log_2 p_3 + p_4 \log_2 p_4) = \\ = 0,5 \cdot 1 + 0,25 \cdot 2 + 2 \cdot 0,125 \cdot 3 = 1,75.$$

Эффективность кодирования  $H(X)/R = 1,75/2 = 0,875$ .

### 1.3. Алгоритм кодирования Хаффмена

Это алгоритм кодирования кодовыми словами переменной длины, основанный на знании априорных вероятностей символов  $p(x_i), i = 1, 2, \dots, L$ .

Алгоритм:

1. Упорядочить символы источника в порядке убывания вероятностей

$$p(x_1) \geq p(x_2) \geq \dots \geq p(x_L).$$

2. Объединение двух символов (с наименьшими вероятностями)

$$x_L \text{ и } x_{L-1} \rightarrow x'_{L-1}; \quad p(x'_{L-1}) = p(x_L) + p(x_{L-1}).$$

3. Формируем новый алфавит, количество символов уменьшаем на единицу ( $L = L - 1$ ). Выполняются п. 1 и 2. Повторяем процедуру, пока не объединятся все символы ( $L = 1$ ).

Этот алгоритм проще представить в виде кодового дерева (рис. 1); одной ветви, соответствующей одному исходному символу ( $x_{L-1}$ ), присвоить код – «0», другой ветви (символ  $x_L$ ) – «1».

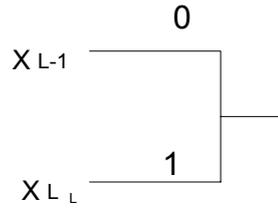


Рис. 1

4. Формируем код каждого символа путем движения от самого правого узла дерева к символу исходного алфавита.

*Пример 1.4*

Составить код переменной длины для источника с семью возможными символами, имеющими вероятность выбора по табл. 1.

Таблица 1

$x_i$	$p(x_i)$	Код	$I(x_i)$	$n_i$
$x_1$	0,35	0	1,514	1
$x_2$	0,3	10	1,737	2
$x_3$	0,2	110	2,322	3
$x_4$	0,1	1110	3,322	4
$x_5$	0,04	11110	4,644	5
$x_6$	0,005	111110	7,644	6
$x_7$	0,005	111111	7,644	6

*Решение:*

Кодовое дерево представлено на рис. 2.

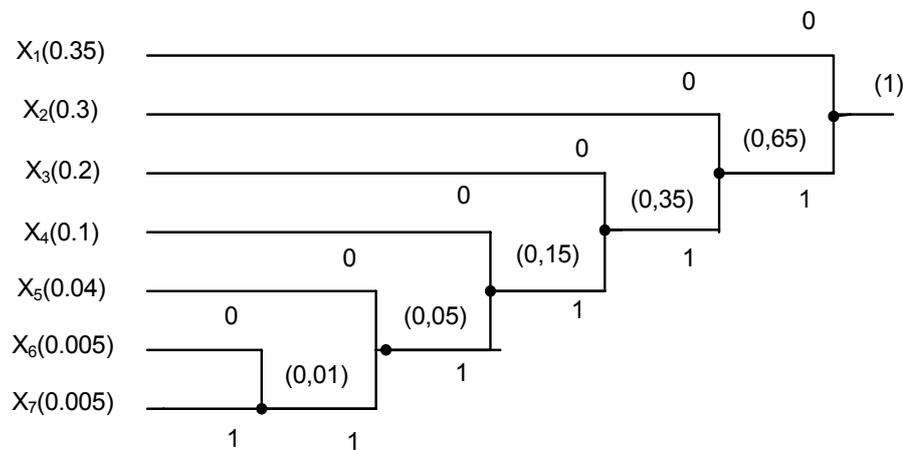


Рис. 2

В табл. 1 приведены коды символов источника, соответствующие дереву рис. 2. Рассчитаем собственную информацию символа по формуле

$$I(x_i) = -\log p(x_i).$$

Занесем значение  $I(x_i)$  и количество символов в кодовом слове  $n_i$  в табл. 1. Скорость кода и энтропия источника:

$$\bar{R} = \sum_{i=1}^l P(x_i) n_i = 2,21 \quad H(x) = \sum_{i=1}^l P(x_i) I(x_i) = 2,11$$

Эффективность кодирования  $H(X)/R = 2,11/2,21 = 0,95$ .

#### 1.4. Алгоритм кодирования источников Лемпела – Зива

Этот алгоритм не требует знания статистики источника, то есть может быть применен как для источников без памяти, так и для источников с памятью. Последовательность с выхода источника делится на фразы (блоки переменной длины). Каждая фраза – это последовательность символов источника, которые отличаются от уже записанной в словарь фразы последним символом. Кодирование выполняется после формирования словаря, когда известно число фраз в словаре. Для бесконечной последовательности размер словаря задается. Код фразы состоит из двух частей: двоичный номер (адрес) существующей в словаре фразы и новый символ. Для фраз «1» и «0» адрес существующей фразы – 00...0.

##### *Пример 1.5*

Задана бинарная последовательность:

10101101001001110101000011001110101100011011.

Закодировать последовательность, используя алгоритм Лемпела – Зива.

*Решение:*

Зададимся количеством фраз в словаре – 16, двоичный адрес содержит  $\log_2 16 = 4$  символа, кодовые слова содержат  $4 + 1 = 5$  символов. Первая фраза словаря – «1», ее код складывается из адреса «0000» пустой фразы и «1». Следующий символ последовательности – «0» образует фразу «0» с кодом «0000 0». Сформированные фразы и соответствующие им коды приведены в табл. 2.

Таблица 2

Номер фразы	Двоичный номер (адрес)	Фразы (имеющаяся фраза и новый символ)	Код
1	2	3	4
1	0001	1	0000 1
2	0010	0	0000 0
3	0011	1 0	0010 0
4	0100	1 1	0001 1
5	0101	0 1	0010 1
6	0110	0 0	0010 0
7	0111	10 0	0011 0
8	1000	11 1	0100 1
9	1001	01 0	0101 0
10	1001	100 0	0111 0
11	1010	01 1	0101 1
12	1011	00 1	0110 1
13	1101	11 0	0100 0
14	1110	10 1	0011 1
15	1111	1000 1	1010 1
16	10000	101 1	1110 1

В данном примере закодировано 44 бита 16 словами по 5 бит, то есть для передачи используется  $5 \cdot 16 = 80$  бит.

### 1.5. Задачи

#### Задача 1.1

Найти взаимную информацию о символе  $X = 0$  при условии, что наблюдается  $Y = 1$  и взаимную информацию о символе  $X = 1$  при условии, что наблюдается  $Y = 1$  для условий примера 1.1:

- а) канал без шумов ( $p_0 = p_1 = 0$ );
- б)  $p_0 = p_1 = 1/2$ ;
- в)  $p_0 = p_1 = 1/4$ .

#### Задача 1.2

Построить зависимость энтропии двоичного источника от вероятности  $q$  из примера 1.2.

### Задача 1.3

Используя алгоритм Хаффмена, построить код переменной длины для источника с восемью возможными символами, имеющими вероятность выбора  $p(x_1) = 0,36$ ,  $p(x_2) = 0,14$ ,  $p(x_3) = 0,13$ ,  $p(x_4) = 0,12$ ,  $p(x_5) = 0,1$ ,  $p(x_6) = 0,09$ ,  $p(x_7) = 0,04$ ,  $p(x_8) = 0,02$ . Рассчитать скорость кодирования, энтропию источника и эффективность кодирования.

### Задача 1.4

Для источника из задачи 1.3 рассчитать скорость кодирования, эффективность кодирования при кодировании кодами постоянной длины.

### Задача 1.5

Для примера 1.3 рассчитать скорость кодирования, эффективность кодирования при кодировании кодами переменной длины.

### Задача 1.6

Сравнить эффективность кодирования кодами постоянной и переменной длины для различных источников, используя результаты решения задач 1.3 – 1.5.

### Задача 1.7

Рассчитать эффективность кодирования для источника из  $L$  символов при использовании кодовых слов из  $R$  двоичных символов.

а)  $L = 7, R = 3$

$$p(x_1) = p(x_2) = p(x_3) = p(x_4) = 1/8, p(x_5) = p(x_6) = 1/5, p(x_7) = 1/10.$$

б)  $L = 7, R = 3$

$$p(x_1) = p(x_2) = p(x_3) = p(x_4) = p(x_5) = p(x_6) = p(x_7).$$

в)  $L = 8, R = 3$

$$p(x_1) = p(x_2) = p(x_3) = p(x_4) = 1/10, p(x_5) = p(x_6) = 1/6, p(x_7) = p(x_8) = 2/15.$$

г)  $L = 8, R = 3$

$$p(x_1) = p(x_2) = p(x_3) = p(x_4) = p(x_5) = p(x_6) = p(x_7) = p(x_8).$$

*Задача 1.8*

Построить код переменной длины, используя алгоритм Хаффмена. Рассчитать эффективность кода. Вероятности символов источника заданы в табл. 3.

*Таблица 3*

	а	б	в	г
$p(x_1)$	0,03	0,05	0,05	0,02
$p(x_2)$	0,1	0,1	0,1	0,1
$p(x_3)$	0,15	0,12	0,12	0,07
$p(x_4)$	0,15	0,15	0,18	0,15
$p(x_5)$	0,25	0,25	0,22	0,22
$p(x_6)$	0,25	0,3	0,3	0,3
$p(x_7)$	0,07	0,03	0,03	0,14

*Задача 1.9*

Построить словарь кодовых слов для кодирования последовательности символов, используя алгоритм Лемпела – Зива.

```
1100110110101011100001100010100000101010
1010000101010011001110101110001000001101
0101000100111001000100010100101000011100
1001010100100100100010101010100101000101
```

*Задача 1.10*

Принятая последовательность, закодированная с использованием алгоритма Лемпела – Зива:

```
000001 000000 000010 000111 000100 000101 001000 001100 001101
001010 010011 010000 001110 011000 011010 011101 000110 001011
010001 001111 100100 010010
```

Длина кодового слова 6 бит. Восстановить переданную последовательность.

## 2. ГЕОМЕТРИЧЕСКОЕ ПРЕДСТАВЛЕНИЕ СИГНАЛОВ

### 2.1. Ортогональное разложение сигналов

Пусть  $s(t)$  является детерминированным вещественным сигналом с ограниченной энергией:

$$e_s = \int_{-\infty}^{\infty} s^2(t) dt < \infty.$$

$\{f_n(t), n = 1, 2, \dots, N\}$  – ансамбль ортонормированных функций:

$$\int_{-\infty}^{\infty} f_n(t) f_m(t) dt = \begin{cases} 0 & (m \neq n), \\ 1 & (m = n). \end{cases}$$

Сигнал  $s(t)$  может быть аппроксимирован при помощи взвешенной линейной комбинации этих функций:

$$\hat{s}(t) = \sum_{k=1}^K s_k f_k(t),$$

где  $s_k = \int_{-\infty}^{\infty} s(t) f_k(t) dt$ .

Примером ортогонального разложения сигналов является ряд Фурье.

### 2.2. Процедура Грамма – Шмидта

Имеется ансамбль сигналов с ограниченной энергией  $\{s_i(t), i = 1, 2, \dots, M\}$ . Процедура Грамма – Шмидта позволяет конструировать ансамбль ортонормированных сигналов и может быть представлена в виде последовательности шагов.

1. Найти энергию сигнала  $s_1(t)$ :  $e_1 = \int_{-\infty}^{\infty} s_1^2(t) dt$ .

2. Первый сигнал ортонормированного ансамбля:  $f_1(t) = s_1(t) / \sqrt{e_1}$ .

3. Выполнить п. 4 – 6 для  $k = 2, 3, \dots, M$ .

4. Найти проекцию сигнала  $s_k(t)$  на все найденные сигналы ортонормированного ансамбля:  $c_{ik} = \int_{-\infty}^{\infty} s_k(t) f_i(t) dt, i = 1, 2, \dots, k-1$ . Найти

вспомогательную функцию  $f'_k(t) = s_k(t) - \sum_{i=1}^{k-1} c_{ik} f_i(t), i = 1, 2, \dots, k-1$ .

5. Найти энергию сигнала  $f_k(t)$ :  $e_k = \int_{-\infty}^{\infty} f_k^2(t) dt$ .

6. Найти  $k$ -й сигнал ортонормированного ансамбля:  $f_k(t) = f_k'(t) / \sqrt{e_k}$ .

Сигналы  $\{s_i(t), i = 1, 2, \dots, M\}$  могут быть представлены рядом  $s_i(t) = \sum_{k=1}^K s_{ki} f_k(t)$ , где коэффициенты  $s_{ki} = \int_{-\infty}^{\infty} s_i(t) f_k(t) dt$ , или вектором в пространстве сигналов, заданном ортонормированными функциями,  $\vec{s}_i = [s_{i1}, s_{i2}, \dots, s_{iK}]$ .

### Пример 2.1

Применим процедуру Грамма – Шмидта к ансамблю четырех сигналов, показанных на рис. 3.

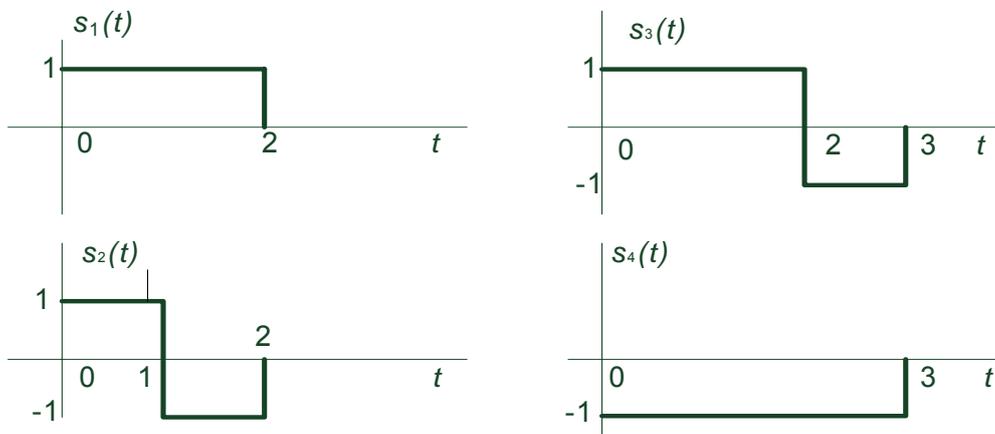


Рис. 3

Определение первой функции ортонормированного набора:

- энергия сигнала  $s_1(t)$ :  $e_1 = \int_0^2 1 dt = 2$ ;

- Ортонормированная функция:  $f_1(t) = s_1(t) / \sqrt{2}$ .

Определение второй функции:

- Проекция сигнала  $s_2(t)$  на  $f_1(t)$ :  $c_{12} = \int_0^1 1 f_1(t) dt - \int_1^2 (-1) f_1(t) dt = 0$ , следова-

тельно,  $s_2(t)$  и  $f_1(t)$  ортогональны;

- $f_2'(t) = s_2(t)$ ;

- $e_2 = \int_0^1 1^2 dt + \int_1^2 (-1)^2 dt = 2;$

- Ортонормированная функция:  $f_2(t) = s_2(t) / \sqrt{2}.$

Определение третьей функции:

- Проекция сигнала  $s_3(t)$  на  $f_1(t)$  и  $f_2(t)$ :

$$c_{13} = \int_0^2 1 f_1(t) dt + \int_2^3 (-1) f_1(t) dt = \sqrt{2}; \quad c_{23} = 0;$$

- $f_3'(t) = s_3(t) - \sqrt{2} f_1(t) = \begin{cases} -1, & 2 \leq t \leq 3 \\ 0, & t < 2 \text{ и } t > 3 \end{cases};$

- $e_3 = \int_2^3 (-1)^2 dt = 1;$

- Ортонормированная функция:  $f_3(t) = f_3'(t).$

Определение четвертой функции:

- Проекция сигнала  $s_4(t)$  на  $f_1(t), f_2(t)$  и  $f_3(t)$ :

$$c_{14} = -\sqrt{2}, \quad c_{24} = 0, \quad c_{34} = 1.$$

- $f_4'(t) = s_4(t) + \sqrt{2} f_1(t) - f_3(t) = 0.$  Сигнал  $s_4(t)$  является линейной комбинацией  $f_1(t)$  и  $f_3(t)$ . Ортонормированный набор содержит три функции.

Ортонормированные функции  $f_1(t), f_2(t)$  и  $f_3(t)$  приведены на рис. 4.

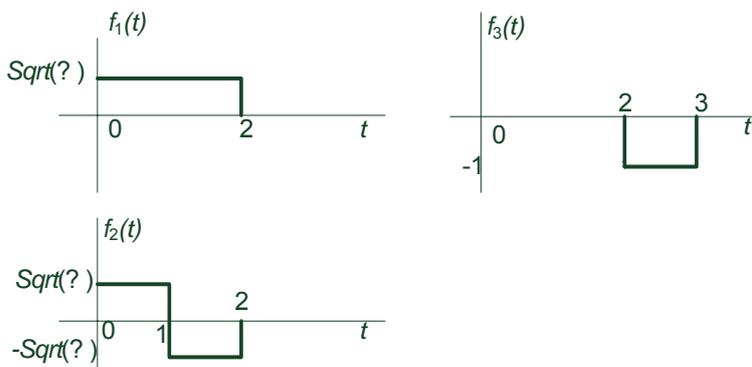


Рис. 4

### Пример 2.2

Получить векторное представление сигналов, показанных на рис. 3, используя ортонормированный ансамбль функций, полученный в примере 2.1. Найти энергию сигналов.

Размерность пространства сигналов  $K = 3$ . Используя формулу  $s_{ik} = \int_{-\infty}^{\infty} s_i(t) f_k(t) dt$ , найдем координаты векторов исходных сигналов в

пространстве сигналов:

$$\vec{s}_1 = (\sqrt{2}, 0, 0), \vec{s}_2 = (0, \sqrt{2}, 0), \vec{s}_3 = (\sqrt{2}, 0, 1), \vec{s}_4 = (-\sqrt{2}, 0, 1).$$

Энергия сигналов равна квадрату длины вектора  $\vec{s}_i$  в пространстве сигналов  $e_i = |\vec{s}_i|^2 = \int_{-\infty}^{\infty} s_i^2(t) dt$ .

$$e_1 = 2, e_2 = 2, e_3 = 3, e_4 = 3.$$

### 2.3. Задачи

#### Задача 2.1

Сформировать ансамбль ортонормированных сигналов для сигналов, изображенных на рис. 5.

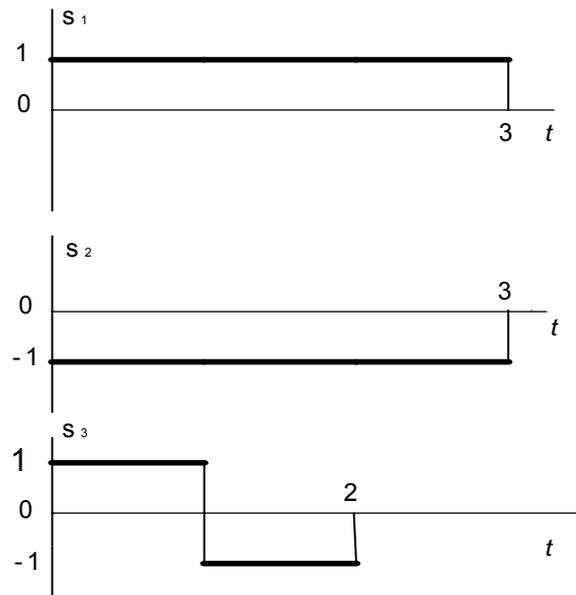


Рис. 5

#### Задача 2.2

Сформировать ансамбль ортонормированных сигналов для сигналов, изображенных на рис. 6.

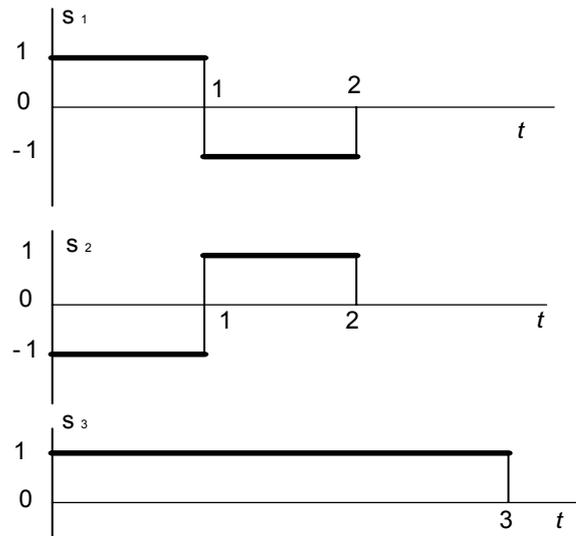


Рис. 6

### Задача 2.3

Записать сигналы  $s(t)$ , заданные в пространстве сигналов  $\{f_1, f_2\}$ :

$$f_1(t) = 10 \cos 2\pi f_c t;$$

$$f_2(t) = 3 \sin 2\pi f_c t$$

векторами:  $s_1 = (3, 0, -1)$ ,  $s_2 = (0, 1, 3)$ ,  $s_3 = (1, 3, 0)$ ,  $s_4 = (-4, 0, 1)$ .

### Задача 2.4

Определить ортонормированный базис для модулированных сигналов:

а) сигнал с амплитудной модуляцией

$$s_m(t) = A_m g(t) \cos 2\pi f_c t, \quad m = 1, 2, \dots, M, \quad 0 \leq t \leq T;$$

б) сигнал фазовой модуляции

$$s_m(t) = g(t) \cos \frac{2\pi(m-1)}{M} \cos 2\pi f_c t - g(t) \sin \frac{2\pi(m-1)}{M} \sin 2\pi f_c t,$$

$$m=1, 2, \dots, M, \quad 0 \leq t \leq T;$$

в) сигнал квадратурной амплитудной модуляции

$$s_m(t) = A_{mc} g(t) \cos 2\pi f_c t - A_{ms} g(t) \sin 2\pi f_c t, \quad m = 1, 2, \dots, M, \quad 0 \leq t \leq T;$$

г) сигнал частотной модуляции

$$s_m(t) = \sqrt{\frac{2\varepsilon}{T}} \cos[2\pi f_c t + 2\pi m \Delta f t], \quad m = 1, 2, \dots, M, \quad 0 \leq t \leq T.$$

### 3. ПОМЕХОУСТОЙЧИВОЕ КОДИРОВАНИЕ

#### 3.1. Циклические коды

Циклический код  $(n, k)$ , где  $n$  – длина кодового слова,  $k$  – количество проверочных символов генерируется с помощью порождающего полинома  $g(p)$  степени  $n - k$ , который является множителем при разложении полинома  $p^n + 1$ .

Например, рассмотрим код с длиной блока  $n = 7$ . Полином  $p^7 + 1$  можно разложить на сомножители

$$p^7 + 1 = (p + 1)(p^3 + p^2 + 1)(p^3 + p + 1).$$

Для синтеза кода  $(7, 4)$  можно использовать один из двух порождающих полиномов:

$$g_1(p) = p^3 + p^2 + 1, \quad g_2(p) = p^3 + p + 1.$$

С порождающим полиномом связана порождающая матрица  $G$  размерами  $(n \times k)$ .

##### 3.1.1. Алгоритм формирования матрицы $G$ :

1. Сформировать первую строку матрицы: первые  $k$  элементов строки соответствуют коэффициентам порождающего полинома, оставшиеся  $n - k$  элементов заполняются нулями.

2. Следующие строки порождающей матрицы получаются циклическим сдвигом предыдущей строки на одну позицию вправо, таким образом формируются  $k - 1$  строки порождающей матрицы.

Для  $k$  информационных бит  $X_i$  кодовое слово  $C_i$  формируется по правилу умножения матриц:

$$C_i = X_i G.$$

##### Пример 3.1

Для полинома  $g_1(p) = p^3 + p^2 + 1 = 1 \cdot p^3 + 1 \cdot p^2 + 0 \cdot p + 1$  кода  $(7, 4)$  первая строка: 1 0 1 1 0 0 0. Порождающая матрица:

$$G = \begin{vmatrix} 1 & 1 & 0 & 1 & 0 & 0 & 0 \\ 0 & 1 & 1 & 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 1 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 & 1 & 0 & 1 \end{vmatrix}$$

Возможные кодовые слова приведены в табл. 4.

Таблица 4

Информационные биты				Кодовые слова (4 информационных символа и 3 проверочных)						
0	0	0	0	0	0	0	0	0	0	0
0	0	0	1	0	0	0	1	1	0	1
0	0	1	0	0	0	1	1	0	1	0
0	0	1	1	0	0	1	0	1	1	1
0	1	0	0	0	1	1	0	1	0	0
0	1	0	1	0	1	1	1	0	0	1
0	1	1	0	0	1	0	1	1	1	0
0	1	1	1	0	1	0	0	0	1	1
1	0	0	0	1	1	0	1	0	0	0
1	0	0	1	1	1	0	0	1	0	1
1	0	1	0	1	1	1	0	0	1	0
1	0	1	1	1	1	1	1	1	1	1
1	1	0	0	1	0	1	1	1	0	0
1	1	0	1	1	0	1	0	0	0	1
1	1	1	0	1	0	0	0	1	1	0
1	1	1	1	1	0	0	1	0	1	1

Например, информационным символам 0110 соответствует кодовое слово 0110100 (4-я строка табл. 4), проверочные символы равны 100.

3.1.2. Формирование порождающей матрицы в систематической форме и проверочной матрицы.

Порождающая матрица в систематической форме представляется в виде  $\mathbf{G} = [\mathbf{I}_k: \mathbf{P}]$ , где  $[\mathbf{I}_k]$  – единичная матрица, подматрица  $\mathbf{P}$  имеет размеры  $k \times n$ . Для формирования  $i$ -й строки подматрицы  $\mathbf{P}$  необходимо выполнить деление полиномов  $p^{n-1}$  на порождающий полином. Коэффициенты полинома остатка от деления используются для формирования строки матрицы  $\mathbf{P}$ .

Проверочная матрица  $\mathbf{H}$  строится по порождающей в систематической форме:  $\mathbf{H} = [\mathbf{P}^T: \mathbf{I}_{n-k}]$  и имеет размеры  $(n - k) \times n$ .

*Пример 3.2.* Составим порождающую матрицу в систематической форме и проверочную матрицу для кода (7, 3) с использованием по-

линома  $g_2(p) = p^3 + p + 1$ .

Выполним деление  $p^6, p^5, p^4$  и  $p^3$  на  $g_2(p)$ . На рис. 7 приведен пример деления полинома  $p^6$  на  $p^3 + p + 1$ .

$$\begin{array}{r|l}
 p^6 & p^3+p+1 \\
 \underline{p^6+p^4+p^3} & \\
 p^4+p^3 & \\
 \underline{p^4+p^2+p} & \\
 p^3+p^2+p & \\
 \underline{p^3+p+1} & \\
 p^2+1 & 
 \end{array}$$

Рис. 7

Остаток от деления  $p^6$  на  $g_2(p)$  равен  $(p^2 + 1)$ , поэтому первая строка подматрицы  $\mathbf{P} - 101$ ; остаток от деления  $p^5 - (p^2 + p + 1)$ , вторая строка – 111; остаток от деления  $p^4 - (p^2 + p)$ , третья строка – 110; остаток от деления  $p^3 - (p + 1)$ , четвертая строка – 011. Порождающая матрица в систематической форме имеет вид:

$$G = \left[ \begin{array}{cccc|ccc}
 1 & 0 & 0 & 0 & 1 & 0 & 1 \\
 0 & 1 & 0 & 0 & 1 & 1 & 1 \\
 0 & 0 & 1 & 0 & 1 & 1 & 0 \\
 0 & 0 & 0 & 1 & 0 & 0 & 1
 \end{array} \right].$$

Построим проверочную матрицу.

$$H = \left[ \begin{array}{cccc|ccc}
 1 & 1 & 1 & 0 & 1 & 0 & 0 \\
 0 & 1 & 1 & 0 & 0 & 1 & 0 \\
 1 & 1 & 0 & 1 & 0 & 0 & 1
 \end{array} \right].$$

3.1.3. Циклический кодер может быть реализован на базе линейного сдвигающего регистра с обратной связью с использованием порождающего или проверочного полинома.

Циклический кодер с использованием порождающего полинома  $g(p) = g_{n-k}p^{n-k} + g_{n-k-1}p^{n-k-1} + \dots + g_1p + g_0$  представлен на рис. 8.

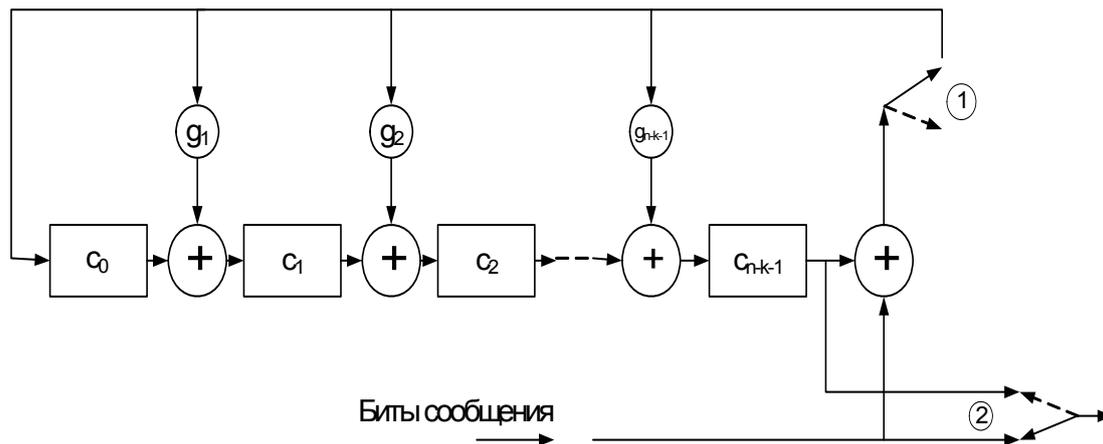


Рис. 8

Пример 3.3. Циклический кодер для кода (7, 4) с порождающим полиномом  $g_2(p) = p^3 + p + 1$  показан на рис. 9.

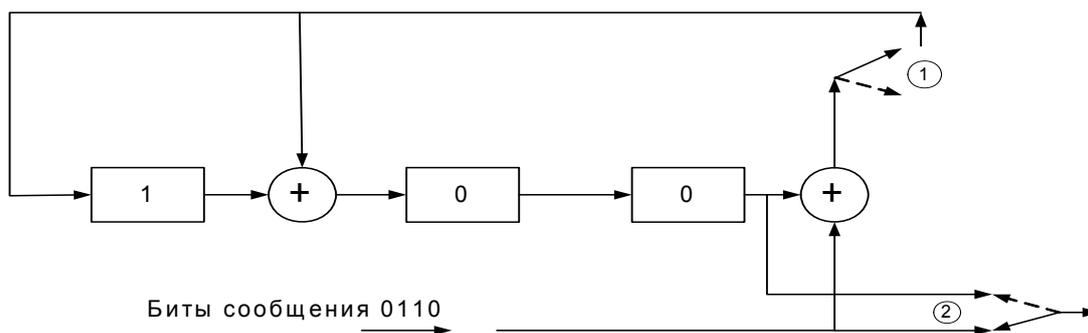


Рис. 9

Предположим, что на вход кодера поступают информационные символы 0110. Содержание сдвигового регистра показано в табл. 5.

Таблица 5

Вход	Шаг сдвига	Содержимое регистра		
	0	0	0	0
0	1	0	0	0
1	2	1	1	0
1	3	1	0	1
0	4	1	0	0

Сформированные проверочные символы 100, выход кодера 0110001.

### 3.1.4. Синдромное декодирование блочных кодов

Пусть для кода  $(n, k)$  задана порождающая матрица  $G$  в систематической форме и проверочная матрица  $H$ . Синдромом образца ошибок называется  $(n - k)$ -мерный вектор  $S = eH^T$ , т.е. имеется  $2^{n-k}$  синдромных образцов ошибок. В качестве образцов ошибок выбираются вектор с нулевой ошибкой и  $n$  векторов с одиночной ошибкой. Для выбранных образцов ошибок рассчитывается синдром. Если количество синдромов меньше  $2^{n-k}$ , выбираются образцы с большим числом ошибок (сначала двумя, потом тремя и т.д. числом ошибок), при этом необходимо проверять, чтобы синдромы для рассматриваемого образца ошибок не совпадали с уже имеющимися.

*Пример 3.4.* Определим таблицу синдромов и образцов ошибок для кода  $(5, 2)$  с порождающей матрицей

$$G = \begin{bmatrix} 1 & 0 & 1 & 0 & 1 \\ 0 & 1 & 0 & 1 & 1 \end{bmatrix}.$$

Проверочная матрица:

$$H = \begin{bmatrix} 1 & 0 & 1 & 0 & 0 \\ 0 & 1 & 0 & 1 & 0 \\ 1 & 1 & 0 & 0 & 1 \end{bmatrix}.$$

В табл. 6 приведены образцы ошибок и соответствующие им синдромы. Расчет синдромов заканчивается, когда будут получены все возможные синдромы.

Таблица 6

Образец ошибки	Синдром	Номер синдрома	Включать в таблицу
00000	000	S0	
00001	001	S1	
00010	010	S2	
00100	100	S3	
01000	011	S4	
10000	101	S5	
00011	011	S4	Нет
00101	101	oS5	Нет

Окончание табл. 6

Образец ошибки	Синдром	Номер синдрома	Включать в таблицу
00110	110	$S_6$	
01001	010	$S_2$	Нет
01010	001	$S_1$	Нет
01100	111	$S_7$	

Полученная таблица синдромов и образцов ошибок представлена в табл. 7.

Таблица 7

Синдром $S$	Образец ошибки $e$
000	00000
001	00001
010	00010
100	00100
011	01000
101	10000
110	00110
111	01100

Таким образом, код (5, 2) корректирует все одиночные и две двойные ошибки.

Алгоритм декодирования принятого кодового слова  $Y$ :

- для принятого кодового вектора вычисляется синдром  $S = YH^T$ ;
- для вычисленного синдрома по таблице определяется вектор ошибок  $e$ ;
- вектор ошибки суммируется (по модулю 2) с принятым вектором, результат – декодированное слово  $C = Y \oplus e$ .

*Пример 3.5.*

Пусть 10101 – переданное кодовое слово, 00101 – принятый кодовый вектор (одиночная ошибка 10000). Синдром равен  $S = 101$ , ему соответствует вектор ошибки 10000. Декодированный вектор  $C = 101010$ , что совпадает с переданным кодовым словом.

Рассмотрим случай двойной ошибки 01100, принятый вектор 11001. Вычисленный синдром 111. Ему соответствует вектор ошибки 01100, декодированный вектор  $11001 \oplus 01100 = 10101$  совпадает с переданным кодовым словом.

### 3.2. Сверточное кодирование

Сверточный код создается прохождением передаваемой информационной последовательности через линейный сдвиговый регистр с конечным числом состояний. Код задается генераторными (порождающими) полиномами  $g_i, i = 1, n$ . Параметрами кода являются  $k$  – размер ячеек сдвигового регистра,  $K$  – количество ячеек сдвигового регистра (или кодовое ограничение сверточного кода) и  $n$  – количество выходных бит, генерируемых для  $k$  входных бит. Скорость сверточного кода  $R = n/k$ .

*Пример 3.6.*

На рис. 10 показан кодер для сверточного кода с параметрами: скорость кода  $1/3$ , кодовое ограничение  $K = 6$ , порождающий полином  $47\ 53\ 75$ .

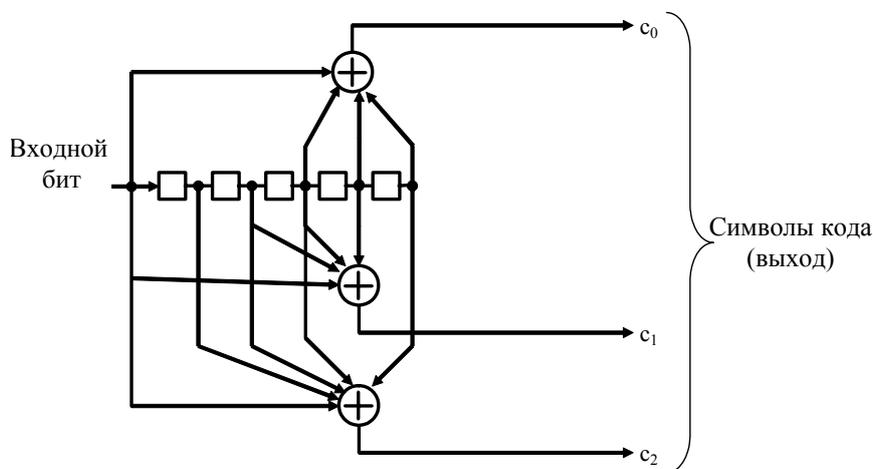


Рис. 10

Кодер состоит из 5-битного сдвигового регистра (в формировании выходных бит участвуют шесть бит – текущий бит и пять предыдущих) и трех функций  $g_0, g_1, g_2$ , генерирующих выходную тройку  $C(c_0, c_1, c_2)$ :

$$\begin{aligned} c_0 &= g_{47}(x_0 \dots x_5) ; \\ c_1 &= g_{53}(x_0 \dots x_5) ; \\ c_2 &= g_{75}(x_0 \dots x_5) . \end{aligned}$$

Индекс функции  $g$  записан в восьмеричной системе и описывает биты кодера, участвующие в свертке.

Выходная последовательность, формируемая сверточным кодом, определяется состоянием сдвигового регистра. Число состояний кодера равно  $2^{K-1} = 32$ . При поступлении нового бита происходит сдвиг и расчет новых трех выходных бит. Формирование последовательности можно представить в форме решетки Треллиса. При этом в каждое состояние приводят два пути, один из них соответствует «1», другой – «0».

Для декодирования сверточных кодов используется алгоритм Витерби. Идея алгоритма Витерби – поиск наиболее правдоподобной последовательности (пути) на решетке Треллиса, для этого входная тройка бит сравнивается с парой троек (для «0» и «1») для каждого состояния. При реализации решетка Треллиса может храниться в памяти или каждый раз вычисляться заново (по алгоритму сверточного кодирования).

В процессе вычислений формируются два массива: путей и метрик.

Для каждого состояния (узла решетки Треллиса) рассчитываются две метрики (для «0» и для «1») возможных переходов в это состояние (количество ошибок перехода с принятой тройкой). Из этих метрик выбирается наименьшая. Этот бит добавляется в массив путей (для этого состояния), а метрика – суммируется с метрикой данного пути в массиве метрик.

Алгоритм программной реализации декодера Витерби можно представить последовательностью шагов:

1. Цикл п. 1 – 10 по всем битам команды.
2. Прием  $n$  входных бит.
3. Цикл п. 3 – 9 по всем состояниям  $(1 - 2^{K-1})$ .
4. Расчет перехода (по алгоритму кодера) ( $n$  бит) в это состояние при условии, что бит – «0».
5. Расчет количества несовпадений  $n$  входных бит с битами по решетке Треллиса (метрика «0»).
6. Расчет перехода ( $n$  бит) в это состояние при условии, что бит – «1».
7. Расчет количества несовпадений  $n$  входных бит с битами по решетке Треллиса (метрика «1»).
8. Сравнение метрик «0» и «1». Для меньшей метрики добавление бита в путь для этого состояния и вычисление метрики этого пути (плюс значение вычисленной метрики).

9. Определение пути с минимальной метрикой – minMet.  
10. Конец.

### 3.3. Задачи

#### *Задача 3.1*

Составить таблицу кодовых слов для циклического кода (7, 3) с использованием полинома  $g_2(p) = p^3 + p + 1$ . Сравнить кодовые слова, полученные для полиномов  $g_1(p)$  и  $g_2(p)$ .

#### *Задача 3.2*

Получить порождающую матрицу в систематической форме и проверочную матрицу для циклического кода (7, 4) с использованием полинома  $g_1(p) = p^3 + p^2 + 1$ . Сравнить таблицы кодовых слов, получаемых при использовании порождающего полинома в систематической и несистематической формах.

#### *Задача 3.3.*

Составить схемы кодера циклического кода (7, 4) с порождающим полиномом  $g_1(p) = p^3 + p^2 + 1$ . Показать, что кодер формирует кодовые слова, аналогичные кодеру из примера 3.3.

#### *Задача 3.4.*

Для кода (7, 4) в систематической форме сформируйте таблицу синдромов и образцов ошибок. Сколько ошибок может корректировать этот код?

#### *Задача 3.5.*

Для сверточного кода из примера 3.6 сформировать последовательность выходных бит для исходной последовательности 000101011111001101010.

#### *Задача 3.6.*

Построить решетку Треллиса для кода примера 3.6.

#### *Задача 3.7.*

Используя алгоритм Витерби, декодировать последовательность, полученную в задаче 3.5, при отсутствии и при наличии ошибок. Вероятность канальной битовой ошибки 0,1.

## 4. ВОПРОСЫ К КОНТРОЛЬНЫМ РАБОТАМ

### Вопросы к работе № 1

1. Дискретный источник без памяти.
2. Дискретный источник с памятью.
3. Взаимная информация между  $(x_i$  и  $y_j)$  и  $(y_j$  и  $x_i)$ .
4. Собственная информация  $x_i$ .
5. Энтропия источника.
6. Максимальное значение энтропии источника, выдающего символы алфавита из  $n$  символов.
7. Сравнить эффективность однозначного кодирования каждого символа в кодовое слово и последовательности символов в кодовое слово.
8. Нижняя граница для среднего числа бит на символ при кодировании кодовыми словами постоянной длины.
9. Границы для среднего числа бит на символ для кодов, удовлетворяющих префиксному условию.
10. Сравнить методы кодирования при использовании кодовых слов переменной длины.
11. К чему относятся понятия «дискретизация», «квантование», «кодирование» при кодировании аналоговых источников.
12. Оценка искажений при кодировании аналоговых источников.
13. Зависимость скорости кодирования и искажений при кодировании аналоговых источников.
14. Суть скалярного и векторного квантования. Сравнить характеристики скалярного и векторного квантования.
15. Принципы импульсно-кодовой модуляции (ИКМ).
16. Принципы дифференциальной импульсно-кодовой модуляции (ДИКМ).
17. Принципы адаптивной кодовой модуляции (АИКМ и АДИКМ).
18. Принципы дельта-модуляции.
19. Принципы спектрального кодирования.
20. Принципы модельного кодирования.

## Вопросы к работе № 2

1. Понятия полосового сигнала, полосового шума.
2. Способы представления полосового сигнала.
3. Связь спектральных характеристик полосового сигнала и эквивалентного низкочастотного сигнала.
4. Связь характеристик линейных полосовых систем и эквивалентной низкочастотной системы.
5. Процедура определения отклика полосовой системы на полосовой сигнал с использованием характеристик эквивалентных низкочастотных сигналов и систем.
6. Представление белого шума при анализе прохождения сигнала через узкополосные системы.
7. Понятие ортогональности сигналов.
8. Понятие линейных видов модуляции. Какие из видов модуляции относятся к линейным?
9. Понятие нелинейных видов модуляции. Какие из рассмотренных видов модуляции относятся к нелинейным?
10. Понятие модуляции с памятью. Виды модуляции с памятью.
11. Амплитудная модуляция. Представление сигналов АМ в пространстве сигналов. Ансамбль ортонормированных сигналов для АМ-сигналов.
12. Сигналы фазовой модуляции. Представление сигналов ФМ в пространстве сигналов. Ансамбль ортонормированных сигналов для ФМ-сигналов.
13. Квадратурная амплитудная модуляция. Представление сигналов КАМ в пространстве сигналов. Ансамбль ортонормированных сигналов для КАМ-сигналов.
14. Понятие многомерных сигналов.
15. Модуляция частотным сдвигом. Представление сигналов МЧС в пространстве сигналов. Ансамбль ортонормированных сигналов для МЧС-сигналов.
16. Понятие линейной модуляции с памятью.
17. Почему сигнал ЧМНФ является сигналом с памятью?
18. Параметры сигналов модуляции с непрерывной фазой.
19. Спектральные характеристики сигналов с линейной модуляцией.
20. Характеристика спектральных характеристик сигналов с непрерывной фазой.

### Вопросы к работе № 3

1. Схема оптимального приемника.
2. Схема корреляционного демодулятора оптимального приемника.
3. Схема демодуляторов на согласованных фильтрах.
4. Оптимальный детектор по правилу максимума априорной вероятности.
5. Оптимальный детектор по правилу максимального правдоподобия.
6. Характеристика качества оптимального приемника.
7. Классификация помехоустойчивых кодов.
8. Циклические коды.
9. Сверточные коды.
10. Турбо-коды.
11. Характеристики качества блочных кодов.
12. Характеристики качества сверточных кодов.

## РЕКОМЕНДАТЕЛЬНЫЙ БИБЛИОГРАФИЧЕСКИЙ СПИСОК

1. Прокис, Дж. Цифровая связь : пер. с англ./ Дж. Прокис ; под ред. Д.Д. Кловского. – М. : Радио и связь, 2000. – 800 с. – ISBN 5-256-01434-X (рус.), ISBN 007-051726-6 (англ).

2. Зюко, А.Г. Теория электрической связи/ А.Г. Зюко [и др.]. – М. : Радио и связь, 1998. – 433 с.

## ОГЛАВЛЕНИЕ

ПРЕДИСЛОВИЕ .....	3
1. КОДИРОВАНИЕ ИСТОЧНИКОВ.....	4
1.1. Логарифмическая мера информации.....	4
1.2. Кодирование кодовыми словами фиксированной длины.....	5
1.3. Алгоритм кодирования Хаффмена .....	6
1.4. Алгоритм кодирования источников Лемпела – Зива .....	8
1.5. Задачи .....	9
2. ГЕОМЕТРИЧЕСКОЕ ПРЕДСТАВЛЕНИЕ СИГНАЛОВ .....	12
2.1. Ортогональное разложение сигналов.....	12
2.2. Процедура Грамма – Шмидта .....	12
2.3. Задачи .....	15
3. ПОМЕХОУСТОЙЧИВОЕ КОДИРОВАНИЕ .....	17
3.1. Циклические коды.....	17
3.2. Сверточное кодирование .....	23
3.3. Задачи .....	25
4. ВОПРОСЫ К КОНТРОЛЬНЫМ РАБОТАМ.....	26
РЕКОМЕНДАТЕЛЬНЫЙ БИБЛИОГРАФИЧЕСКИЙ СПИСОК ..	29

МЕТОДИЧЕСКИЕ УКАЗАНИЯ  
К ПРАКТИЧЕСКИМ ЗАНЯТИЯМ  
ПО ДИСЦИПЛИНЕ  
«ТЕОРИЯ ЭЛЕКТРИЧЕСКОЙ СВЯЗИ»

Составитель  
КАЛЫГИНА Любовь Александровна

Ответственный за выпуск – зав. кафедрой профессор В.Н. Ланцов

Подписано в печать 05.06.08.  
Формат 60x84/16. Усл. печ. л. 1,86 Тираж 70 экз.  
Заказ  
Издательство  
Владимирского государственного университета.  
600000, Владимир, ул. Горького, 87.