Лабораторная работа N 4

МЕТОДЫ ЗАЩИТЫ ОТ ОШИБОК, ИСПОЛЬЗУЕМЫЕ В СЕТЯХ

Цель работы: Изучить методы защиты от ошибок, применяемые в СПД. Отработать программы, реализующие процедуры формирования помехозащищенных кадров и получения информации из них.

4.1. КРАТКИЕ ТЕОРЕТИЧЕСКИЕ СВЕДЕНИЯ

В СПД вычислительных сетей для передачи данных используются линии связи различных типов: проводные (воздушные), кабельные, радиорелейные, волоконно-оптические и радиоканалы наземной и спутниковой связи. Для большинства типов линий связи из-за воздействия на них помех и наличия шумов в аппаратуре передачи данных (АПД) достоверность передачи данных низкая и составляет, как правило, 10^{-4}– 10^{-6} ошибок на бит.

Для обеспечения более высокой достоверности передачи данных (порядка 10^{-8}– 10^{-9} ошибок бит) в СПД применяют групповые методы защиты от ошибок, избыточное (помехоустойчивое) кодирование и системы с обратной связью.

К групповым методам защиты от ошибок можно отнести способ,известный как принцип Вердана: вся информация или отдельные кодовые комбинации передаются несколько раз, обычно нечетное число раз (минимум три раза).

Принимаемая информация запоминается специальным устройством и сравнивается. Суждение о правильности приема выносится по совпадению большинства из принятой информации методами "два из трех", "три из пяти" и т.д.

Недостатком метода является резкая загрузка СПД дополнительными пакетами, что приводит к снижению реальной пропускной способности каналов связи.

Другой метод, также не требующий перекодирования информации, предполагает передачу информации блоками, состоящими из нескольких кодовых комбинаций.

В конце каждого блока посылается информация, содержащая количественные характеристики переданного блока, например, число единиц или нулей в блоке. На приемной стороне эти характеристики вновь подсчитываются, сравниваются с переданными по каналу связи и если они совпадают, то блок считается принятым правильно. При несовпадении количественных характеристик на передающую сторону посылается сигнал ошибки.

Недостаток метода в том, что сам код количественных характеристик никак не защищен от ошибок, что снижает достоверность самопроверки.

Среди методов защиты от ошибок наибольшее распространение получило помехоустойчивое кодирование. Оно предполагает разработку корректирующих (помехоустойчивых) кадров, обнаруживающих и исправляющих определенного рода ошибки или только обнаруживающих их.

В помехоустойчивых кодах, кроме информационных разрядов, всегда имеются один или несколько дополнительных разрядов, являющихся проверочными и служащих для достижения более высокого качества передачи данных. В них имеются разрешенные и запрещенные кодовые комбинации. Появление при приеме запрещенной кодовой комбинации говорит о появлении ошибки в кадре.

Основными характеристиками корректирующих кодов являются кодовое расстояние, значность, корректирующая способность, избыточность и оптимальность кода, коэффициент обнаружения и исправления ошибки, простота технической реализации метода.

Кодовое расстояние (d_{min}) - это минимальное число позиций, в которых любая кодовая информация отличается от других кодовых комбинаций. Оно численно равно числу единиц, полученных при сложении этих двух кодовых комбинаций между собой по модулю 2.

Корректирующая способность кода зависит от кодового расстояния:

- при $d_{min} = 1$ ошибка не обнаруживается;
- при $d_{min} = 2$ обнаруживаются одиночные ошибки;
- при $d_{min} = 3$ обнаруживаются двойные ошибки или исправляются одиночные ошибки.

В общем случае $d_{min} = r + s + 1$, где r - число обнаруживаемых ошибок, а s - число исправляемых ошибок. При этом обязательным является условие, что $r > s$ или $r=s$. Если код только обнаруживает ошибки, то $d_{min} = r + 1$, а если только исправляет, то $d = 2s + 1$.

Значность кода (n), или длина кодовой комбинации, определяется суммой числа информационных (m) и проверочных (контрольных) разрядов (k). Как правило, значность кода равна

$$n = m + k .$$

Избыточность кода выбирается отношением числа контрольных разрядов к значности кода:

$$k_{изб.} = \frac{k}{m + k} \quad \text{или} \quad k_{изб.} = 1 - \frac{m}{m + k} .$$

На практике часто используют и такую характеристику, как коэффициент обнаружения и исправления ошибок.

$$k_{обн.} = \frac{L}{L + M} ,$$

где L – число кодовых комбинаций, ошибки в которых были обнаружены и исправлены или только обнаружены;

М – число кодовых комбинаций, ошибки в которых не были обнаружены.

Оптимальность кода указывает на полноту использования его корректирующих возможностей.

Выбор корректирующих кодов в определенной степени зависит от требований, предъявляемых к достоверности передачи. Для правильного его выбора необходимо иметь статистические данные с закономерностях возникновения ошибок, их характере, численности и распределении во времени.

Так,например, корректирующий код, исправляющий одиночные ошибки, может быть эффективным лишь при условии, что ошибки статистические независимы, а вероятность их появления не превышает некоторой величины.

Этот код оказывается совершенно непригодным, если ошибки появляются группами (пачками).

При выборе корректирующего кода следует стремиться к тому, чтобы код имел меньшую избыточность (хотя чем она больше, тем выше помехоустойчивость системы, но вместе с тем ниже пропускная способность канала связи и значительно больше время передачи данных). Следует также учитывать, что способность построения кодирующих и декодирующих устройств в огромной мере влияет на стоимость сети и надежность аппаратуры.

Разработанные различные корректирующие коды подразделяются на непрерывные и блочные.

В непрерывных или рекуррентных кодах контрольные элементы располагаются между информационными.

В блочных кодах информация кодируется, передается и декодируется отдельными группами (блоками) равной длины.

Блочные коды бывают разделимые (все информационные и контрольные элементы размещаются на строго определенных позициях) и неразделимые (элементы кодовой комбинации не имеют четкого деления на избыточные и информационные).

К неразделимым относится код с постоянным числом нулей и единиц.

Разделимые коды делятся на систематические и несистематические.

В систематических кодах проверочные символы образуются с помощью различных линейных комбинаций над информационными символами. К систематическим кодам относятся коды Хемминга, циклические коды, коды Боузы-Чоудхури-Хоквинхема(коды БЧК) и др.

Одним из самых простых и чаще всего используемых на практике методов является контроль на четность.Его сущность заключается в том, что к каждой кодовой комбинации добавляется один разряд, в который записывается единица, если число единиц в кодовой комбинации нечетное, или нуль, если четное.

При декодировании подсчитывается количество единиц в кодовой комбинации по модулю 2. Если оно оказывается четным, то поступившая информация считается правильной, если нет, то ошибочной.

Преимущества контроля на четность заключаются в минимальном значении коэффициента избыточности и в простоте его технической реализации, а недостаток - в том, что обнаруживаются ошибки, имеющие только нечетную кратность, аналогично осуществляется контроль на нечетность. В этом случае число единиц в правильной посылке должно быть нечетным.

К систематическим кодам относится и код Хемминга. Он позволяет не только обнаруживать, но и исправлять ошибки.

Свойство этого кода таково, что контрольное число указывает номер позиции, где произошла ошибка. Если ошибка отсутствует, то в контрольном коде будет последовательность нулей. Эти коды позволяют исправлять все одиночные ошибки (при $d_{min} = 3$), а также исправлять все одиночные ошибки и обнаруживать все двойные ошибки (при $d_{min} = 4$), но не исправлять их.

В качестве исходного для построения кода Хемминга берут двоичный код на все сочетания с числом информационных символов, код на все сочетания с числом информационных символов и к нему добавляют контрольные символы К. Таким образом, общая длина закодированной комбинации будет $n = m + k$.

При кодировании необходимо определять число контрольных символов. Оно определяется из соотношения

$$m = E'' \log_2(n + 1) = E'' \log_2(m + k + 1),$$

где m – число информационных символов;

k – число контрольных символов;

n – длина кода Хэмминга;

E'' – знак округления в сторону большего значения.

Далее устанавливают место, где эти контрольные символы должны быть расставлены в коде. В принципе место расположения контрольных символов не имеет значения: их можно приписывать и перед информационными символами, и после них, и чередуя информационные символы с контрольными. Однако произвольное расположение контрольных символов затрудняет проверку принятого кода.

В коде Хэмминга n – разрядное число имеет "m" информационных и "к" контрольных разрядов. Каждый из контрольных разрядов – результат проверки на четность определенной группы разрядов кода Хэмминга, т.е. контроль такого кода состоит из К проверок на четность.

Чтобы контрольный код указывал номер разряда числа, в котором произошла ошибка, нужно связать контрольный код с номером разряда, в котором может быть ошибка.

Для обеспечения этого группа разрядов для каждой проверки выбирается по следующему правилу.

Первая проверка, в результате которой заполняется разряд контрольного слова, должна охватывать те разряды числа, номера которых, представленные в двоичном коде, имеют в первом (младшем) разряде единицу. К ним относятся 1,3,5,7,9-й и т.д. разряды.

Вторая проверка, в результате которой заполняется второй разряд контрольного слова, должна охватывать те разряды числа, в двоичных номерах которых во втором разряде единица. К ним относятся 2,3,5,6,7,10-й и т.д. разряды.

Третья проверка, в результате которой заполняется третий разряд контрольного слова, должна охватывать разряды числа, в двоичных номерах которых в третьем разряде единица. К ним относятся 4,5,6,7,12-й и т.д. разряды.

Четвертая проверка должна охватывать разряды, в двоичных номерах которых в четвертом разряде единица и т.д.

Для 7-разрядного кода Хэмминга (n=7, m=4, k=3) номера проверок и проверяемые при этом разряды сведены в табл. 4.1. Нумерацию разрядов можно вести как справа налево, так и обратно.

Теперь нужно решить, какие из n позиций кода Хемминга использовать для размещения информационных разрядов числа, а какие для разрядов контрольного слова.

Таблица 4.1

Номера проверок (заполняемые разряды контрольного слова)	Проверяемые разряды (позиции)
1	1, 3, 5, 7
2	2, 3, 6, 7
3	4, 5, 6, 7

При выборе позиций размещения разрядов для контрольного слова необходимо обеспечить, чтобы каждый контрольный разряд входил в проверку только один раз.

Из табл.4.1 видно, что такими позициями являются 1,2,4 ..., в общем случае 2^i. Остальные позиции (3, 5, 6, 7 и т.д.) встречаются в табл. 4.1 два и более раз.

Следовательно, для размещений разрядов контрольного слова нужно выбрать 1,2,4... позиции в зависимости от разрядности кода Хемминга.

Структура 7-разрядного кода Хемминга имеет вид

$$A = a_7a_6a_5a_4a_3a_2a_1 = m_4m_3m_2k_3m_1k_2k_1,$$

где m_i – разряды информационного слова;,

k_i – разряды контрольного слова.

Так как проверка каждой группы производится на четность, то значение соответствующего разряда контрольного слова определяется суммой по модулю 2 значений остальных разрядов этой группы, т.е.

$$k_1 = a_1 = a_3 \oplus a_5 \oplus a_7 = m_1 \oplus m_2 \oplus m_4;$$

$$k_2 = a_2 = a_3 \oplus a_6 \oplus a_7 = m_1 \oplus m_3 \oplus m_4;$$

$$k_3 = a_4 = a_5 \oplus a_6 \oplus a_7 = m_2 \oplus m_3 \oplus m_4.$$

Например, для двоичного кода 1011 код Хемминга будет иметь вид 1010101 (т.к. $k_1 = 1$; $k_2 = 0$; $k_3 = 0$).

Для проверки правильности принятого слова (отсутствие ошибки в нем) тоже используется контроль на четность значений разрядов соответствующих групп:

$$k_1' = a_1 \oplus a_3 \oplus a_5 \oplus a_7 = k_1 \oplus m_1 \oplus m_2 \oplus m_4;$$

$$k_2' = a_2 \oplus a_3 \oplus a_6 \oplus a_7 = k_2 \oplus m_1 \oplus m_3 \oplus m_4;$$

$$k_3' = a_4 \oplus a_5 \oplus a_6 \oplus a_7 = k_3 \oplus m_2 \oplus m_3 \oplus m_4.$$

При искажении какого-либо символа в коде Хемминга, в том числе и контрольного, полученный код проверки $k_3'k_2'k_1'$ будет указывать номер разряда, в котором произошло искажение символа (появилась ошибка). Путем инвертирования значения этого разряда получим истинный код. Если окажется при проверке, что $k'k'k'$=000, то ошибок в принятом коде нет.

Например, в результате передачи предыдущего кода Хемминга (1010101) был получен код 1010001 - ошибка в третьем разряде, считая справа налево).

При проверке получим

$$k_1' = 1 \oplus 0 \oplus 1 \oplus 1 = 1$$

$$k_2' = 0 \oplus 0 \oplus 0 \oplus 1 = 1$$

$$k_3' = 0 \oplus 1 \oplus 0 \oplus 1 = 0$$

Следовательно, получим контрольный код $k_3'k_2'k_1'$ = 011, что указывает на ошибку в третьем разряде принятого кода. Инвертируя значение третьего разряда, получим код 1010101, который соответствует переданному.

Для получения информационного кода необходимо исключить из кода Хемминга значение контрольных разрядов (позиции 1,2,4). Получим код 1011, который был закодирован в предыдущем примере.

Код Хемминга имеет существенный недостаток: при обнаружении любого числа ошибок он исправляет лишь одиночные ошибки. При увеличении разрядности кода увеличивается число проверок, но уменьшается избыточность кода. К тому же код Хемминга не позволяет обнаруживать групповые ошибки, сконцентрированные в пакетах.

Наиболее эффективно обнаружение и исправление такого рода ошибок осуществляется с помощью циклических кодов.

Циклические коды относятся к числу блоковых систематических кодов, в которых каждая комбинация кодируется самостоятельно (в виде блока) таким образом, что информационные (m) и контрольные (k) символы всегда находятся на определенных местах.

Возможность обнаружения и исправления практически любых ошибок при относительно малой избыточности по сравнению с другими кодами, а также простота схемной реализации аппаратуры кодирования и декодирования сделали эти коды широко распространенными.

В основу циклического кодирования положено использование неприводимого многочлена $P(x)$, который применительно к циклическим кодам называется образующим, генераторным или порождающим многочленом (полиномом).

Неприводимые многочлены нельзя представить в виде произведения многочленов низших степеней. Они играют роль, сходную с простыми числами в теории чисел. Неприводимые полиномы $P(x)$ можно записать в виде десятичных или двоичных чисел либо в виде алгебраического многочлена (полинома).

В табл.4.2 приведены все неприводимые полиномы до пятой степени включительно, используемые для построения циклических кодов. Полные таблицы приведены в [5].

Таблица 4.2

Степень полинома	Алгебраическая форма P(x)	Двоичн.код полинома	Десятичн.код полинома
1	$x + 1$	11	3
2	$x^2 + x + 1$	111	7
3	$x^3 + x + 1$	1011	11
3	$x^3 + x^2 + 1$	1101	13
4	$x^4 + x + 1$	10011	19
4	$x^4 + x^3 + 1$	11001	25
4	$x^4 + x^3 + x^2 + x + 1$	11111	31
5	$x^5 + x^2 + 1$	100101	37
5	$x^5 + x^3 + 1$	101001	41
5	$x^5 + x^3 + x^2 + x + 1$	101111	47
5	$x^5 + x^4 + x^2 + x + 1$	110111	55
5	$x^5 + x^4 + x^3 + x + 1$	111011	59
5	$x^5 + x^4 + x^3 + x^2 + 1$	111101	61

Циклический код определяется с помощью порождающего полинома P(x) степени K. Посредством операции над полиномом, выполняемой с участием P(x) определяемым m битами сообщения, подлежащего передаче, образуется так называемый кодовый полином, который делится без остатка на P(x). Этот кодовый полином передается вместо исходного сообщения. Если при передаче в нем произошла ошибка, то он делиться без остатка не будет. Следовательно, по значению остатка можно судить о наличии ошибки в принятом коде.

Кодовый полином можно получить, если заданную кодовую комбинацию G(x) умножить на образующий полином и P(x). Однако в этом коде контрольные символы k будут располагаться в самых разнообразных местах кодовой комбинации. Такой код не является систематическим, что затрудняет его схемную реализацию.

Ситуацию можно значительно упростить, если контрольные символы приписывать в конце кода, т.е. после информационных символов. Действительно, умножим кодовую комбинацию G(x), которую мы хотим закодировать, на одночлен X^k, имеющий ту же степень, что и образующий многочлен P (x). Поделим произведение X^k* G(x) на образующий многочлен P(x):

$$\frac{X^k * G(x)}{P(x)} = Q(x) + \frac{R(x)}{P(x)}, \qquad (4.1)$$

где Q(x) - частное от деления;
R(x) - остаток;
X - основание системы счисления.

Умножая выражение 4.1. на P(x) и перенося R(x) в другую часть равенства согласно правилам алгебры двоичного поля, т.е. без перемены знака на обратный, получаем

$$F(x) = Q(x) * P(x) = G(x) * X^k + R(x). \qquad (4.2)$$

Умножение G(x) на X^k равнозначно сдвигу кода G (x) на (к) разрядов влево. Таким образом, согласно равенству 4.2 циклический код, т.е. закодированное сообщение F(x), можно получить умножением заданной комбинации G(x) на одночлен X^k, имеющий ту же степень, что и образующий полином P(x), с добавлением к этому произведению остатка R(x), полученного после деления произведения G(x) * X^k на образующий многочлен P(x). Деление выполняется согласно правилам алгебры двоичного поля, т.е. вместо вычитания делителя на каждом шаге он прибавляется поразрядно к остаткам по модулю 2.

Вид и наивысшая степень порождающего полинома выбираются заранее исходя из длины (разрядности) передаваемой кодовой комбинации и заданной кратности ошибок.

Степень порождающего полинома не должна быть меньше числа контрольных символов К. Это значит, что если К = 3, то из табл. 4.2 можно выбрать любой образующий многочлен Р(х), начиная с третьей степени и выше. Для упрощения технической реализации кодирования степень Р (х) следует выбирать равной числу контрольных разрядов К. Если в таблице имеется ряд многочленов с данной степенью, то из них следует выбирать самый короткий. Однако число ненулевых членов полинома Р (х) не должно быть меньше кодового расстояния d_{min}.

Выбор числа контрольных разрядов осуществляется так же, как и в коде Хемминга, или по эмпирической формуле

$$K = E''\log_2[(m+1) + E''\log_2(m+1)], \qquad (4.3)$$

где E" - знак округления в сторону большего значения;

m - число информационных разрядов в коде.

Пусть, например, требуется закодировать число 1101 циклическим кодом, позволяющим обнаруживать двукратные ошибки или обнаруживать и исправлять одиночную ошибку.

Заданную комбинацию 1101 можно представить в виде полинома

$$G(x) = 1 * x^3 + 1 * x^2 + 0 * x^1 + 1 = x^3 + x^2 + 1,$$

где X - основание системы счисления.

Определяем значение кодового расстояния (d_{min}).

Исходя из первого условия

$$d_{min} = r + 1 = 2 + 1 = 3.$$

из второго условия

$$d_{min} = r + s + 1 = 1 + 1 + 1 = 3,$$

где r - число обнаруживаемых ошибок;

s - число исправленных ошибок.

Выбираем для кодирования циклический код с $d_{min} = 3$.

Определяем число контрольных разрядов. Так как заданное число содержит 4 разряда (m=4), то

$$k = E''\log_2[(4+1) + E''\log_2(4+1)] = E''\log_2(5+3) = 3.$$

Из табл. 4.2 выбираем полином третьей степени, например

$$P(x) = x^3 + x + 1,$$

которому соответствует двоичный код 1011.

Умножая G(x) на одночлен X^k, который имеет третью степень, получим

$$G(x) * X^3 = (x^3 + x^2 + 1) * x^3 = x^6 + x^5 + x^3 \longrightarrow 1101000$$

Делим произведение $G(x) * X^3$ на образующий полином $P(x)$

$$1 * x^6 + 1 * x^5 + 0 * x^4 + 1 * x^3 + 0 * x^2 + 0 * x^1 + 0$$
$$\oplus\, 1 * x^6 + 0 * x^5 + 1 * x^4 + 1 * x^3$$
$$1 * x^5 + 1 * x^4 + 0 * x^3 + 0 * x^2$$
$$\oplus\, 1 * x^5 + 0 * x^4 + 1 * x^3 + 1 * x^2$$
$$1 * x^4 + 1 * x^3 + 1 * x^2 + 0 * x^1$$
$$\oplus\, 1 * x^4 + 0 * x^3 + 1 * x^2 + 1 * x^1$$
$$1 * x^3 + 0 * x^2 + 1 * x^1 + 0$$
$$\oplus\, 1 * x^3 + 0 * x^2 + 1 * x^1 + 1$$

остаток – $0 * x + 0 * x + 1 \longrightarrow 001$

или $R(x) = 001$

$$1 * x + 0 * x + 1 * x + 1$$
$$1 * x + 1 * x + 1 * x + 1 \quad \text{– частное}$$

1111

Тот же результат получим и при делении коэффициентов:

```
⊕1101000    |1011
 1011       |----
 -------    |1111 - частное
 ⊕1100
  1011
 -------
  ⊕1110
   1011
 -------
   ⊕1010
    1011
 -------
     001 - остаток
```

Разрядность остатка должна быть на единицу меньше, чем разрядность делителя.

В итоге комбинация двоичного кода, закодированная циклическим кодом, согласно выражению (4.2) примет вид

$$F(x) = 1101000 + 001 = 1101001$$

Идея обнаружения ошибок в принятом циклическом коде заключается в том, что при отсутствии ошибок закодированная комбинация $F(x)$ делится на порождающий многочлен $P(x)$ без остатка. После проверки этого условия для получения полезной информации из посылки достаточно отбросить контрольные разряды (остаток).

Пусть была передана информация в виде циклического кода $F(x)=1101001$, закодированная образующим полиномом

$$P(x)= x^3+ x + 1 \longrightarrow 1011$$

На приемной стороне получим этот же код. Путем деления его на код образующего полинома получим:

```
⊕1101001    |1011
 1011       |----
 -------    |1111
 ⊕1100
  1011
 -------
  ⊕1110
   1011
 -------
   ⊕1011
    1011
 -------
     000 - остаток
```

Остаток R(x) = 000, следовательно, ошибки нет. Теперь контрольные "k" младших разрядов (в примере k = 3) принятого кода отбрасываем и получим результат

$$G(x) = 1101$$

Этот код нами прежде и был закодирован. Если в результате передачи код будет искажен, то вместо циклического кода F(x) будет принят код H(x), который можно представить так :

$$H(x) = F(x) + E(x),$$

где E(x) - многочлен, содержащий столько единиц, сколько элементов в принятой комбинации не совпадает с элементами переданной комбинации.

Например, вместо F(x)= 1101001 принят код H(x)=1101011 (ошибка во втором справа разряде).

При делении принятого кода H(x) на код образующего полинома получим:

```
⊕ 1101011       | 1011
  1011          |------
  -------       | 1111
 ⊕ 1100
   1011
  -------
  ⊕ 1111
    1011
  -------
   ⊕ 1001
     1011
  -------
     010 - остаток
```

Получился остаток R(x) = 010, отличный от нуля. Следовательно, при передаче возникла ошибка. Дальнейшая процедура исправления ошибок протекает таким образом:

1. Подсчитывается вес W (число единиц) остатка R(x). В рассматриваемом примере W = 1, т.к. R(x) = 010.

Если вес остатка (W) равен или меньше числа исправляемых выбранным кодом ошибок (S), т.е. W < S, то принятый код складывают по модулю 2 с кодом остатка R(x) и получают исправленную комбинацию.

Действительно, для рассматриваемого примера

```
   1101011
⊕      010
----------
   1101001
```

получим истинный циклический код. Остается для получения информационного кода только отбросить три младших разряда этого кода.

2. Если $W > S$, то производят циклический сдвиг на один разряд влево от принятой посылки и полученный код снова делят на образующий полином. Если теперь вес полученного остатка $W < S$, то циклически сдвинутую комбинацию складывают с остатком и полученную сумму циклически сдвигают в обратную сторону (вправо) на один разряд (возвращают на прежнее место). В результате получают исправленный код.

3. Если же после циклического сдвига на один символ по-прежнему $W > S$, то производят дополнительные циклические сдвиги влево. При этом после каждого сдвига полученную комбинацию делят на $P(x)$ и проверяют остаток. При $W < S$ выполняют действия, указанные в пункте 2, с той лишь разницей, что обратных циклических сдвигов вправо делают столько, сколько их было сделано влево.

<u>Пример</u>. При передаче кода $F(x)$= 1101001, закодированного с использованием полинома $P(x)$= 1011 для кода с исправлением одиночной ошибки (S=1), получено сообщение $H(x)$ = 1001001 (ошибка во втором слева разряде). Исправить ошибку.

Проверяем полученный код

```
  1001001   | 1011
⊕ 1011      |------
  ----      | 1010
   0100
 ⊕ 0000
   ----
    1000
  ⊕ 1011
    ----
     111- остаток R (x)
```

Так как $R(x)$=111, то вес его W = 3, но $W > S$. Сдвигаем циклически код $\overleftarrow{H}_1(x)$ на один разряд влево. Получим $\overleftarrow{H}_2(x)$ = 0010011. В результате деления этого кода на $P(x)$ имеем $R(x)$ = 101. Для него W=

= 2 > S. Еще выполняем циклический сдвиг влево на 1 разряд (второй). Получим = 0100110. Делим этот код на P(x).

Получим остаток R(x) =001, для которого W=1=S. Прибавим этот остаток к коду H (x). Получим

$$
\begin{array}{r}
0100110 \\
\oplus \quad 001 \\
\hline
0100111
\end{array}
$$

Теперь циклически сдвинем полученную сумму на 2 разряда вправо, получим код 1101001, который соответствует истинному переданному коду, из которого можно выделить информационный код.

Циклические коды широко применяются в зарубежной и отечественной аппаратуре передачи данных, в системах телеобработки ЕС ЭВМ и в целом ряде других устройств, поскольку они имеют сравнительно небольшую избыточность и достаточно простые кодирующие и декодирующие устройства, которые реализуются на основе обычных сдвигающих регистров. МККТТ рекомендует применять в вычислительных сетях циклические коды с полиномом

$$P(x) = x^{16} + x^{12} + x^{5} + 1$$

Обнаруживать и исправлять пакеты ошибок могут рекуррентные коды. В этом случае в канал связи последовательно передаются контрольные и информационные элементы

$$k_i m_j k_{i+1} m_{j+1} k_{i+2} m_{j+2} \dots$$

При этом контрольные элементы определяются следующим образом:

$$k_i = m_{j+d} + m_{j+2d} ,$$

где 2d – шаг, выбираемый произвольно и указывающий на число исправленных ошибок, следующих одна за одной.

Значение i в контрольном элементе находится как j+d, т.е. контрольный элемент располагается перед информационным, имеющим номер на d единиц меньше.

При декодировании каждого информационного элемента осуществляются две проверки:

$$S_j' = m_{j+d}' \oplus m_j' \oplus k_{j-d}' \ ;$$

$$T_j' = m_{j-d}' \oplus m_j' \oplus k_{j-2d}'$$

где m_j' и k_j' принятые информационные и контрольные символы.

Информационный символ считается ошибочным, если обе проверки будут равны 1. Действительное значение m_j' определяется в этом случае по соотношению

$$m_{j\ \text{действ.}} = m_j' + S_j * T_j \ .$$

Избыточность рекуррентных кодов составляет 0,5.

Распространенным кодом, но не относящимся к группе неразделимых, является код с постоянным числом единиц и нулей в комбинациях или код M из N (код с постоянным весом).

В этом коде также частично кодовые комбинации являются разрешенными, а частично запрещенными.

Общее число разрешенных кодовых комбинаций определяется формулой

$$N = C_n^l = \frac{n!}{l!(n-l)!} \ ,$$

где l - число единиц в слове длиной n.

Наиболее употребимыми являются пятиразрядный код с двумя единицами ($N = C_5^2 = 10$) и семиразрядный код с тремя единицами ($N = C_7^3 = 35$). Примеры этих кодов приведены в табл. 4.3.

Таблица 4.3

Код C_5^2		Код C_7^3
11000	10010	1010100
01010	00011	0101010
01100	01001	1110000
00101	10001	0000111
00110	10100	1001001

Правильность принятых кодовых комбинаций в кодах определяется путем подсчета количества единиц и, если, например, в коде C_6^2 приняты не две единицы, а в коде C_7^3 – не три единицы, то в передаче произошла ошибка.

Очевидно, код C_7^3 может обнаруживать все единичные ошибки, так как при этом в комбинации будут либо две единицы, либо четыре. Кроме того, он позволяет обнаруживать часть многократных ошибок (двойные, тройные и т.д.), за исключением случаев, когда одна из единиц переходит в ноль, а один из нулей – в единицу.

Фирма IBM использует восьмиэлементный код, содержащий четыре единицы и четыре нуля ($N = C_8^4 = 70$).

Как показали исследования этой фирмы, в вычислительных сетях с помощью кода M из N можно обнаруживать в блоке, насчитывающем около 32 000 символов, все ошибки, кратные трем или меньше, или все пачки ошибок до 16-ти символов.

4.2. ЗАДАНИЕ НА ЛАБОРАТОРНУЮ РАБОТУ

1. Разработать программу кодирования информации помехозащищенным кодом в соответствии с вариантом задания (табл. 4.3).

2. Разработать программу определения правильности приема информации, устранения ошибки, если она возникла, и декодирования в соответствии с вариантом задания (табл.4.4).

Таблица 4.4

Номер варианта	Разрядность инф. кода	Используемый код
1	4	Хемминга
2	5	То же
3	7	" – "
4	8	" – "
5	4	Циклический
6	8	То же
7	12	" – "
8	16	" – "

Примечание. В качестве дополнительного задания можно предложить разработать программу контроля на четность, передачи информации блоками с указанием его характеристик и т.д.