Лекция 1

ПРИНЦИПЫ ПОСТРОЕНИЯ СИСТЕМ ПЕРЕДАЧИ ДАННЫХ

Роль методов и средств передачи информации в системах обработки данных

В любом живом существе происходит передача информации. Так, в организме человека сбором информации о внешнем мире занимаются органы чувств. Затем нервная система передает эту информацию в головной мозг, который перерабатывает ее и рассылает приказы (выработанные на основе поступившей информации и сами по себе являющиеся информацией) по нервным волокнам в мышцы, и т.д.

Аналогичным образом передается информация на любом предприятии или в организации, где трудится совместно множество людей. Эта информация передается в виде докладных, распоряжений, запросов и т.д. - всего того, без чего невозможна коллективная деятельность. В любом выдающемся достижении современной техники главную роль играют передача, хранение и переработка информации.

Разнообразие источников и потребителей информации привело к существованию различных форм ее представления: символьной, текстовой и графической (рис. 1).

Символьная форма, основанная на использовании символов - букв, цифр, знаков и т.д., является наиболее простой, но она практически применяется только для передачи несложных сигналов о различных событиях.

Более сложной является *текстовая* форма представления информации. Здесь так же, как и в предыдущей форме, используются символы: буквы, цифры, математические знаки. Однако информация заложена не только в этих символах, но и в их сочетании, порядке следования.

Однако самой емкой и сложной является *графическая* форма представления информации. К этой форме относятся виды природы, фотографии, чертежи, схемы, рисунки, играющие большое значение в нашей жизни и содержащие большое количество информации.

Таким образом, обмен информацией предполагает использование некоторой системы знаков, например, естественного или искусственного (формального) языка.

Изучение знаковых систем наукой о знаках, словах и языках (симеотикой) проводится по крайней мере на трех уровнях:

на *синтактическом* уровне рассматриваются внутренние свойства текстов, т.е. отношения между знаками, отражающие структуру данной знаковой системы. Внешние свойства текстов изучают на семантическом и прагматическом уровнях;

на *семантическом* уровне анализируют отношения между знаками и обозначаемыми ими предметами, действиями, качествами, т.е. смысловое содержание текста, его отношение к источнику информации;

на *прагматическом* уровне рассматривают отношения между текстом и теми, кто его использует, т.е. потребительское содержание текста, его отношение к получателю.

Учитывая взаимосвязь проблем передачи информации с уровнями изучения знаковых систем, их разделяют на проблемы синтактического, семантического и прагматического уровней.

Проблемы синтактического уровня касаются создания теоретических основ построения систем связи. Это чисто технические проблемы совершенствования методов передачи сообщений и их материального воплощения - сигналов. Иначе говоря, на этом уровне рассматриваются только проблемы доставки получателю сообщений как совокупности знаков, при этом полностью абстрагируются от их смыслового и прагматического содержания.

Основу интересующей нас теории информации составляют результаты решения ряда проблем именно этого уровня. Она опирается на понятие “количество информации”, являющееся мерой частоты употребления знаков, которая никак не отражает ни смысла, ни важности передаваемых сообщений.

Возникновение математической теории информации стало возможным после того, как было осознано, что количество информации можно задать числом так же, как можно выразить числом расстояние, время, массу и т.д.

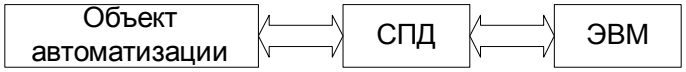
Информация - это новые сведения, подлежащие передаче, хранению и обработке.

Данные - это цифровая информация с которой оперирует ЭВМ.

Системы передачи данных необходимы для построения различных информационных систем (ИС), например:

1.ИС, работающие в режиме распределенных вычислительных сетей (РВС). Сети ЭВМ включают в себя различное число машин, которые могут использоваться коллективно. ВС могут быть в рамках кафедры, института, города, страны и всего мира. В данном случае существует система передачи данных как между ЭВМ в пределах локальных ВС, так и между ВС различных уровней.

2.ИС, работающие в режиме измерительно-вычислительных комплексов (ИВК) (Рис. 2).



В данном случае существует система передачи данных между объектом автоматизации (ОА) и ЭВМ.

Основные способы и каналы передачи данных

Пусть имеются два абонента (источник информации и получатель) A_1 и A_2, которые обмениваются между собой данными.

A_1	A_2
ЭВМ	ЭВМ
Польз.	ЭВМ
ЭВМ	Польз.
ОА	ЭВМ

ЭВМ ОА

Различают три способа передачи данных между абонентами.

1.Данные передаются только в одном направлении ($A_1 \rightarrow A_2$). Такой режим передачи называется симплексным и, соответственно, необходим симплексный канал передачи данных (КПД).

2. Данные передаются в обоих направлениях, но не одновременно. Такой режим передачи называется полудуплексным и, соответственно, необходим полудуплексный КПД.

3. Данные передаются в обоих направлениях одновременно. Такой режим передачи называется дуплексным и, соответственно, необходим дуплексный КПД.

Под *каналом передачи данных* понимается совокупность технических средств и физических сред, обеспечивающих требуемый режим передачи данных.

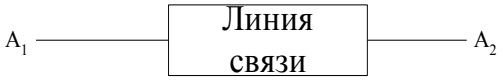
С точки зрения объема передаваемой информации наиболее важной характеристикой является полоса пропускания канала - та полоса частот, в пределах которой данные могут передаваться с допустимым уровнем искажений.

Различают следующие типы каналов:

1. Низкочастотные (телеграфные) - ПП ~ 100 Гц.
2. Каналы тональной частоты (телефонные) - ПП ~ 3 кГц.
3. Высокочастотные (радиоканалы и оптические каналы)
 - ПП ~ 100 ТЧ.
4. Каналы спутниковой связи - ПП ~ 15000 ТЧ.

В зависимости от числа абонентов различают одноканальную и многоканальную связь.

При одноканальной связи в сеансе участвуют только два абонента.



Линия связи - это физическая среда по которой передаются данные (витая пара, коаксиальный кабель, световод, атмосфера и т.д.).

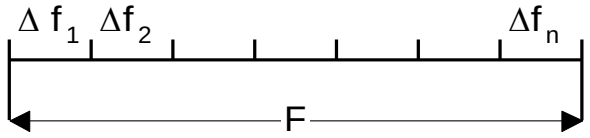
При передачи данных на большие расстояния самым дорогостоящим компонентом СПД является линия связи (ЛС). Поэтому, для наиболее рационального использования ресурсов ЛС организуют многоканальный режим передачи данных, при котором ЛС используется несколькими парами абонентов.

Меры, направленные на рациональное использование ресурсов ЛС несколькими парами абонентов называются разделением или уплотнением каналов (при многоканальной связи).

Различают временное и частотное разделение.

При временном разделении каждой паре абонентов отводится определенный интервал времени, в который она монопольно владеет всей ЛС.

Суть частотного разделения заключается в следующем. Пусть полоса пропускания ЛС = F Гц и требуется передать n - сообщений. При этом сообщения занимают полосы частот $\Delta f_1+\Delta f_2+...+\Delta f_n$. Передача сообщений возможна только при $\sum_{i=1}^{n} \Delta f_i \le F$, т.е. по частотам сообщения должны не перекрываться (Рис. 3).

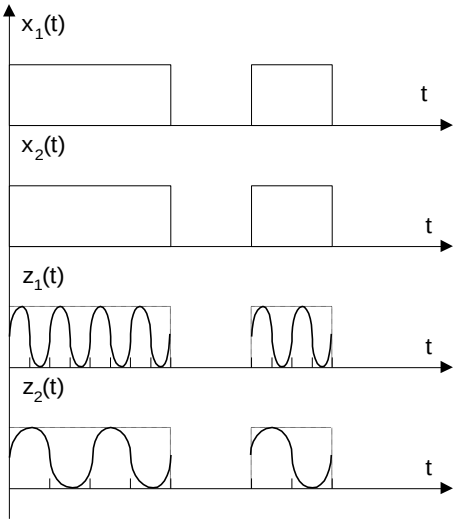


Но на практике частоты перекрываются и следовательно надо их разнести. При частотном разделении каналов для разнесения частот по каждой паре абонентов используется модуляция.

Суть модуляции в том, что по каналу передается некоторый сигнал, называемый переносчиком, и один или несколько его параметров изменяются в соответствии с законом изменения сообщения. В качестве переносчика чаще всего используется гармонический сигнал $x(t)=A \sin(\omega t + \varphi)$.

Если меняется амплитуда A, то имеет место амплитудная модуляция (АМ), если - ω, то - частотная модуляция (ЧМ), а если - φ, то - фазовая модуляция (ФМ).

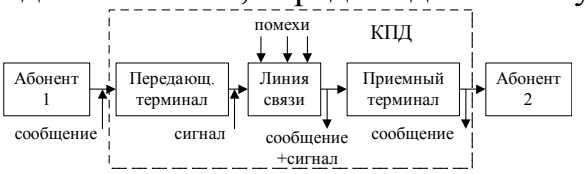
Например, если два различных абонента передают одинаковые сообщения, то благодаря АМ их можно передавать одновременно на разнонесущих частотах (Рис. 4).



Лекця 2

Система передачи данных

Рассмотрим структуру простейшей системы передачи данных (СПД). Система одноканальная, передача данных осуществляется в симплексном режиме.

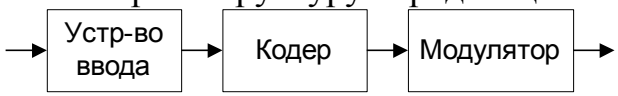


Здесь КПД - канал передачи данных. СПД=КПД + А + ПО.

Под *сообщением* понимается форма представления информации. Одна и та же информация может быть представлена различными сообщениями, т.е. в различной форме.

Под *сигналом* понимается материальный переносчик сообщений.

Рассмотрим структуру передающего терминала.



У.ВВ. - устройство чтения информации с различного рада носителей (клавиатура, НМД, сканер, мышь. CD-ROM и т.д.).

КОДЕР в общем случае выполняет две основные функции:

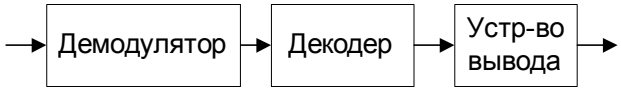
устранение вредной избыточности из сообщений, т.е. данных не несущих полезной информации (оптимальное или эффективное кодирование);

введение полезной избыточности, т.е. таких дополнительных данных, которые позволяют обнаруживать и исправлять возможные ошибки в процессе передачи сообщений по ЛС (помехоустойчивое кодирование).

МОДУЛЯТОР служит для уплотнения ЛС, а так же для согласования свойств сигнала с параметрами ЛС.

Рассмотрим структуру приемного терминала.

В нем выполняются операции обратные тем, которые выполняются в передающем терминале.



ДЕМОДУЛЯТОР выполняет функции детектирования принимаемого сообщения (функция обратная функции модуляции).

ДЕКОДЕР в общем случае выполняет так же две функции:

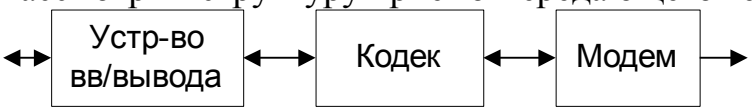
восстановление исходного сообщения в вид, пригодный для получателя;

решается задача оптимального приема, т.е. принимаемое сообщение обрабатывается с учетом априорных сведений о сообщении и о помехе (это позволяет обеспечить наиболее высокую степень достоверности принятого сообщения).

У.ВЫВ. - это любое устройство для регистрации сообщения на одном из накопителей.

При реализации ПД и Д режимов работы абоненты снабжаются приемо-передающими терминалами, т.е. такие устройства могут выполнять функции как приемных, так и передающих терминалов.

Рассмотрим структуру приемо-передающего терминала.



В ППТ модулятор и демодулятор выполняются в виде одного конструктивного устройства - модема. Кодирующее и декодирующее устройства оформляются также в одном конструктиве - кодеке.

КОДЕК = КОДЕР + ДЕКОДЕР

МОДЕМ = МОДУЛЯТОР + ДЕМОДУЛЯТОР

Исходя из основных задач теории передачи данных можно выделить следующие проблемы:

Эффективное или оптимальное кодирование.

Помехоустойчивое кодирование.

Теория модуляции и демодуляции.

Теория оптимального приема сообщений.

ОСНОВЫ ТЕОРИИ ОПТИМАЛЬНОГО КОДИРОВАНИЯ СООБЩЕНИЙ

Каждое сообщение состоит из элементов, которыми могут быть символы, группы символов, буквы, слова, биты и т.д.

Сущность оптимального кодирования состоит в том, чтобы найти такой способ построения сообщения, при котором каждый элемент этого сообщения несет наибольшее количество информации. Если такой способ найден, то для передачи заданного количества информации потребуется минимальное число элементов.

Количественная мера информации

Количественная мера информации была введена Клодом Шенноном.

Пусть имеется некоторое сообщение X, состоящее из элементов x_1, x_2,...., x_n. Каждому из элементов соответствует вероятность его появления $p(x_1)$, $p(x_2)$,...., $p(x_n)$.

При построении количественной меры Шенноном были использованы предпосылки о том, что количество информации в элементе обратно пропорционально вероятности появления этого элемента, т.е.

$I(x_i) = 1/p(x_i)$, (1)

где $I(x_i)$ - количество информации в элементе x_i, а $p(x_i)$ - вероятность появления элемента x_i.

Однако такому подходу свойственны некоторые недостатки:

При $p(x_i) = 1$, $I(x_i) = 1$, а должно быть $I(x_i) = 0$.

При двух элементах x_i и x_j , $I(x_i, x_j) = I(x_i)\, I(x_j)$, а должно быть $I(x_i,x_j) = I(x_i) + I(x_j)$, т.е. нарушается закон аддитивности количества информации.

От указанных двух недостатков свободна логарифмическая мера. Поэтому

$I(x_i) = \log 1/p(x_i)$. (2)

При таком подходе

$p(x_i) = 1$, $I(x_i) = 0$.

x_i, x_j , $I(x_i,x_j)=\log 1/p(x_i)\, p(x_j)=\log 1/p(x_i)+\log 1/p(x_j)=I(x_i)+I(x_j)$.

Единицы измерения информации в соответствии с (2) зависят от основания логарифма: lg - [дит], ln - [нит], lb - [бит].

Формула (2) определяет количество информации в одном элементе. Для определения среднего количества информации источника нужно определить среднее по всем элементам с учетом вероятностей их появления, т.е.

$$I(X) = \sum_{i=1}^{n} p(x_i)\, I(x_i) =$$

$$= \sum_{i=1}^{n} p(x_i) \log \frac{1}{p(x_i)} = \quad . (3)$$

$$= -\sum_{i=1}^{n} p(x_i) \log p(x_i)$$

Формула (3) выражает среднее количество информации, содержащееся в сообщении.

Энтропия как мера неопределенности сообщений

Обязательным условием получения информации в результате передачи сообщения является неопределенность относительно того, какое сообщение будет передано.

При этом, количество информации, получаемое в результате передачи сообщений, будет тем больше, чем больше неопределенность до передачи.

Рассмотрим механизм получения информации:

ДО ПЕРЕДАЧИ НЕОПРЕДЕЛЕННОСТЬ

ПОСЛЕ ПЕРЕДАЧИ ИНФОРМАЦИЯ

И так, неопределенность по Шеннону - это энтропия. Обозначается H.

$H(x_i) = \log 1/p(x_i)$. (4)

$$H(X) = -\sum_{i=1}^{n} p(x_i) \log p(x_i) . \quad (5)$$

Формулы энтропии и информации идентичны, но смысл разный. Энтропия - априорная характеристика (до передачи), информация - апостериорная характеристика (после передачи).

Если рассмотреть передачу данных без потерь, то имеем

	ЭНТРОПИЯ	ИНФОРМАЦИЯ
ДО ПЕРЕДАЧИ	H(X)	0
ПОСЛЕ ПЕРЕДАЧИ	0	I(X)=H(X)

Качество кодирования информации определяется энтропией:

$H(X)_{max}$ - наилучший способ кодирования;

$H(X)_{min}$ - наихудший способ кодирования.

Допустим, информация кодируется двумя способами:

$H_1(X) = 5$ бит/символ; $H_2(X) = 2$ бит/символ.

Первый способ кодирования лучше.

Оптимальное кодирование должно быть направлено на увеличении энтропии каждого элемента.

Основные свойства энтропии

Рассмотрим, при каких условиях энтропия принимает наименьшее и наибольшее значения. Предположим, что передаче подлежит двоичное сообщение X из двух элементов x_1, x_2 с вероятностями появления $p(x_1)$ и $p(x_2)$.

Рассмотрим случай, когда $p(x_1) = 1$, $p(x_2) = 0$.

$$H(X) = -p(x_1)\log p(x_1) - p(x_2)\log p(x_2) =$$

$$= -\lim_{p(x_2)\to 0} p(x_2)\log p(x_2) =$$

$$= \lim_{p(x_2)\to 0} \frac{\log \frac{1}{p(x_2)}}{\frac{1}{p(x_2)}} = \lim_{\beta\to\infty} \frac{\log \beta}{\beta} =$$

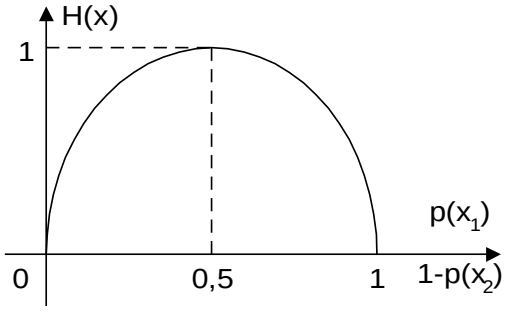
$$= \lim_{\beta\to\infty} \frac{\frac{1}{\beta}\log e}{1} = 0.$$

Здесь введено обозначение $\beta=1/p(x_2)$, а неопределенность раскрыта по правилу Лапиталя. Энтропия минимальна и равна 0, если один из элементов имеет вероятность появление равную 1.

Рассмотрим случай, когда $p(x_1) = p(x_2) = 0{,}5$.

$$H(X) = -\sum_{i=1}^{2} 0{,}5\, lb\, 0{,}5 = 1\, бит$$

Энтропия максимальна и равна 1.



Для двоичных символов H_{max} = 1бит/дв.символ.

Полученные результаты обобщаются на сообщение X из n - элементов $x_1, x_2, ..., x_{n-1}, x_n$.

$H_{min}(X) = 0$ при $p(x_i) = 1$, $p(x_j) = 0$, $i \neq j$.

$$H_{\max}(X) = -\sum_{i=1}^{n} p(x_i)\log p(x_i) = \log n,$$

при $p(x_1) = p(x_2) = ... = p(x_{n-1}) = p(x_n)$.

Следовательно, для оптимального кодирования необходимо выравнивать вероятности появления элементов.

Лекция 3

Энтропия дискретных сообщений при наличии статистических связей между элементами

Статистические связи между элементами имеют место в том случае, если вероятность появления элемента x_i зависит от того, какой элемент x_{i-1} ему предшествовал.

Статистические связи могут охватывать пары соседних элементов (в этом случае они представляют собой односвязную цепь Маркова), тройки соседних элементов (двухсвязная цепь Маркова), ..., n+1 - соседних элементов (n - связная цепь Маркова, $n \neq \infty$).

Все реальные сообщения являются n - связными цепями Маркова.

Рассмотрим сообщение X из элементов $x_1, x_2, ..., x_n$, представляющее собой односвязную цепь Маркова, т.е. пары соседних элементов находятся в статистической связи.

Обозначим $p(x_i/x_j)$ - вероятность появления элемента x_i при условии, что ему предшествовал элемент x_j (j=i-1). Пусть x_1 =1 или 0, x_2 = 0 или 1, тогда возможны следующие случаи

p(0/0) p(0/1)

p(1/0) p(1/1) .

Энтропия $H^*(x_i) = \log 1/(p(x_i/x_j))$. Усредняя по всем x_i с учетом $p(x_i/x_j)$ и x_j с учетом $p(x_j)$ получим среднее значение энтропии односвязной цепи Маркова:

$$H^*(X) = \sum_{i=1}^{n}\sum_{j=1}^{n} p(x_i / x_j)p(x_j)H^*(x_i) =$$

$$= -\sum_{i=1}^{n}\sum_{j=1}^{n} p(x_i / x_j)p(x_j)\log p(x_i / x_j) \quad \textbf{. (*)}$$

Рассмотрим выражение (*) для двух предельных случаев.

x_i и x_j не зависят друг от друга, т.е. $p(x_i/x_j) = p(x_i)$ и

$$H^*(X) = -\sum_{i=1}^{n} p(x_i)\log p(x_i)\sum_{j=1}^{n} p(x_j) =$$

$$= H(X),\ т.к. \sum_{j=1}^{n} p(x_j) = 1.$$

Имеется полная функциональная зависимость между элементами x_i и x_j, т.е. $p(x_i/x_j)$ = 0 или 1 и тогда $H^*(X) = 0$.

Таким образом $0 \le H^*(X) \le H(X)$. (**)

Следовательно, наличие статистических связей между элементами сообщения уменьшает энтропию, причем тем в большей степени, чем большее число соседних элементов охвачено статистическими связями.

Рассмотрим пример сообщения на русском языке. Если не различать *ё* и *е* , *ь* и *ъ* и учитывая, что необходим пробел, имеем 32 символа. Какова максимальная энтропия ? При условии равновероятности и независимости символов средняя энтропия на символ будет максимаоьна и равна

H_{max} = lb 32 = 5 бит/символ.

Появление символов в словах русского языка не равновероятно. Если учесть различную вероятность символов, то

H_1 = 4,39 бит/символ.

С учетом статистической связи между двумя символами энтропия уменьшается до значения

H_2 = 3,52 бит/символ (односвязная цепь Маркова),

между тремя символами - до значения

H_3 = 3,05 бит/символ (двухсвязная цепь Маркова),

.....

между восемью символами - до значения

H_8 = 2 бит/символ (семисвязная цепь Маркова) и

дальше остается неизменной.

Вывод:

Для эффективного кодирования сообщений необходимо увеличивать энтропию за счет:

Выравнивания вероятностей появления символов.

Устранения статистических связей между элементами.

Избыточность сообщений

Если в сообщении элементы равновероятны и друг от друга не зависят, то такое сообщение может быть закодировано оптимальным образом. Для передачи такого сообщения потребуется передать n_{opt} элементов. Если сообщение закодировано не оптимально, то для его передачи необходимо $n>n_{opt}$ символов. В этом случае появляется избыточность, численной характеристикой которой является коэффициент избыточности:

$k_и = (H_{max}(X) - H(X))/ H_{max}(X)$, где

$H_{max}(X)$ - энтропия при оптимальном кодировании;

H (X) - энтропия при неоптимальном кодировании.

Таким образом, $0 \le k_и \le 1$. Для русского языка $k_и = (5 - 2)/5 =0,6$.

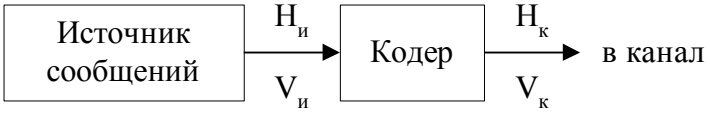
Лекция 4

СКОРОСТЬ ПЕРЕДАЧИ ИНФОРМАЦИИ И ПРОПУСКНАЯ СПОСОБНОСТЬ ДИСКРЕТНОГО КАНАЛА БЕЗ ПОМЕХ

Под *дискретным каналом передачи* информации принято понимать совокупность средств, предназначенных для передачи дискретных сигналов.

Пусть источник сообщения выдает в канал сообщение X из элементов $x_1, x_2, ..., x_n$, а получатель сообщения принимает некоторое сообщение Y из элементов $y_1, y_2, ..., y_n$. Если рассматривать канал без помех, то переданное и принятое сообщения совпадают.

Рассмотрим следующую структуру



На рисунке

$H_и$ - средняя энтропия одного символа источника;

$V_и$ - средняя скорость выдачи одного символа источником;

$H_к$- средняя энтропия одного символа на выходе кодера;

$V_к$- средняя скорость выдачи одного символа кодером.

Следуя Шеннону, выделим следующие характеристики:

Производительность источника сообщения $P = V_и H_и$ [бит/сек].

Скорость передачи информации по каналу $R = V_к H_к$ [бит/сек].

Пропускная способность канала - максимально возможная скорость передачи информации по данному каналу $C = \max R$ [бит/сек] - является характеристикой только канала и не зависит от свойств источника.

Теорема Шеннона о пропускной способности канала без помех

Если производительность источника сообщения P не превышает пропускной способности канала C, то всегда можно найти такой способ кодирования, при котором скорость передачи информации R будет сколь угодно близко приближаться к C, т.е. $C - R = \delta$ - малая величина.

Обратное утверждение.

Если производительность $P > C$, то не существует способа кодирования, обеспечивающего передачу сообщения по дискретному каналу.

Из теоремы следует фундаментальная роль характеристики - пропускная способность канала. Она заключается в том, что эта характеристика определяет границу между возможной и невозможной скоростью передачи информации по каналу ($R \le C$).

В теореме не приводится способ кодирования, но указывается возможность при которой R приближается к C.

Способы кодирования сообщений, которые позволяют приблизить R к C, носят название способов оптимального кодирования.

Оптимальное кодирование дискретных сообщений по Шеннону - Фано

Предполагается, что кодированию подвергается сообщение X из элементов $x_1, x_2, ..., x_n$. Эта операция осуществляется в следующем порядке:

1.Элементы исходного сообщения X упорядочиваются по мере убывания вероятности, т.е. $p(x_1^y) \ge p(x_2^y) \ge .... \ge p(x_n^y)$, в результате чего имеем последовательность $X_y < x_1^y, x_2^y, ..., x_n^y >$.

2.Элементы упорядоченной последовательности разбиваются на две группы таким образом, чтобы суммарные вероятности групп были по возможности равны. Одной группе присваивается символ 0, другой - 1.

3.Разбиение на группы согласно п.2 продолжается до тех пор, пока в каждой группе не останется по одному элементу.

Пример.

X	$p(x_i)$	X_y	$p_y(x_i)$	Разбиение			Коды
a	0,3	d	0,4	0			0
b	0,1	a	0,3	1	0		10
c	0,2	c	0,2	1	1	0	110
d	0,4	b	0,1	1	1	1	111

Коды Шеннона - Фано являются неравномерными, т.е. различным символам соответствуют кодовые комбинации различной длины.

Однозначное декодирование на приемной стороне кодов Шеннона - Фано обеспечивается благодаря их свойству неприводимости: ни одна кодовая комбинация меньшей длины не совпадает с началом комбинации большей длины.

Пример принятого сообщения 110 0 111 10 111 0 0 0 10 10

c d b a b d d d a a

Рассмотрим пример кодирования сообщения тремя способами

Пусть имеется троичный источник сообщения X с элементами x_1, x_2, x_3.

Элементы	x_1	x_2	x_3
$p(x_i)$	0,2	0,7	0,1

Сообщение передается по двоичному каналу, т.е. элементы x_i принимают значение только 1 или 0.

Пусть

$V_и = 1000$ дв.симв./сек;

$V_к = 1000$ дв.симв./сек;

$C = V_к H_{max} = 1000$ дв.симв./сек • 1 бит/ дв.симв. = 1000 бит/сек .

Напомним, что эффективность любого способа кодирования определяется на основании сравнения скорости передачи информации по каналу R и пропускной способности канала C. Чем ближе R к C, тем лучше закодирована информация.

1. Закодируем сообщение равномерным двоичным кодом.

Определим значность кода $n =] lb\ 3 [= 2$.

Пусть $x_1 = 00$, $x_2 = 01$ $x_3 = 10$;

$\tau = 10^{-3}$ с - длительность одного двоичного символа;

$\tau^* = 2\tau = 2 \bullet 10^{-3}$ с - длительность кодовой комбинации, соответствующей одному элементу. Найдем $R_{р.обд}$.

$V_1 = 1/\tau^* = 500$ элем./сек;

$$H = -\sum_{i=1}^{3} p(x_i)\, lb\, p(x_i) =$$

$$= -0{,}2 lb 0{,}3 - 0{,}7 lb 0{,}7 - 0{,}1 lb 0{,}1 =$$

$$= 1{,}16\, бит.$$

$R_{р.обд} = V_1 H = 500$ элем./с • 1,16 бит/элем. = 580 бит/с .
Такой способ кодирования не эффективен, т. к. $R_{р.обд} <$ C.

2. Закодируем сообщение по методу Шеннона - Фано без укрупнения.

Эл-ты $(x_i)_y$	$p(x_i)_y$	Разбиения			Кодовые комбин.	τ_i, с
x_2	0,7	0			0	10^{-3}
x_1	0,2	1	0		10	$2 \bullet 10^{-3}$
x_3	0,1	1	1		11	$2 \bullet 10^{-3}$

Найдем $R_{ШФБ}$.

$$\tau_2^* = \sum_{i=1}^{3} p(x_i)\, \tau_i = 0{,}7 \bullet 10^{-3} + 0{,}2 \bullet 2 \bullet 10^{-3} + 0{,}1 \bullet 2 \bullet 10^{-3} = 1{,}3 \bullet 10^{-3}\ c.$$

$R_{ШФБ} = V_2 H = H/\tau_2^*$ = (1,16 бит/элем.)/$1{,}3 \bullet 10^{-3}$ с. = 890 бит/с,
где $V_2 = 1/\tau_2^*$.

3. Закодируем сообщение по методу Шеннона - Фано с укрупнением, т.е. кодированию подвергаются не отдельные элементы, а группы соседних элементов (группы из 2-х элементов, из 3-х элементов и т.д.)

Рассмотрим группы из двух соседних элементов.

Группы x_i,x_j	Вер-ти $p(x_i,x_j)$	Разбиение						Кодовые комбинации	τ_i, с
x_2,x_2	0,49	0						0	10^{-3}
x_1,x_2	0,14	1	0	0				100	$3 \bullet 10^{-3}$
x_2,x_1	0,14	1	0	1				101	$3 \bullet 10^{-3}$
x_2,x_3	0,07	1	1	0	0			1100	$4 \bullet 10^{-3}$
x_3,x_2	0,07	1	1	0	1			1101	$4 \bullet 10^{-3}$
x_1,x_1	0,04	1	1	1	0			1110	$4 \bullet 10^{-3}$
x_1,x_3	0,02	1	1	1	1	0		11110	$5 \bullet 10^{-3}$
x_3,x_1	0,02	1	1	1	1	1	0	111110	$6 \bullet 10^{-3}$
x_3,x_3	0,01	1	1	1	1	1	1	111111	$6 \bullet 10^{-3}$

Найдем $R_{ШФУ}$.

$$\tau_3^* = (\sum_{i=1}^{9} p(x_i)\,\tau_i) / 2 = 1{,}165 \bullet 10^{-3}\ c.$$

$R_{ШФУ} = V_3\ H = H/\tau_3^* = (1{,}16$ бит/элем.$)/1{,}165 \bullet 10^{-3}$ с. = 995 бит/с,

где $V_3 = 1/\tau_3^*$.

Оптимальное кодирование дискретных сообщений по методу Хаффмена

Метод заключается в следующем.

Элементы алфавита источника располагаются в порядке убывания их вероятностей.

Затем два нижних элемента объединяются в новый укрупненный элемент, который занимает место в алфавите согласно своей суммарной вероятности.

Выполнение п.2 продолжается до тех пор, пока суммарная вероятность двух последних элементов не станет равной единице.

При каждом объединении условимся присваивать цифру 0 элементу, занимающему верхнюю позицию в паре объединяемых элементов, и цифру 1, если эта позиция нижняя.

Число объединений, в которых участвует данный элемент, равно значности соответствующей ему кодовой комбинации. Построенный таким образом код называется кодом Хаффмена.

Рассмотрим пример построения кода Хаффмена.

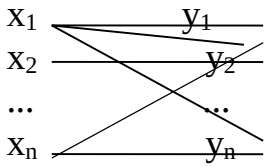
Эл-нт	Вер-ть	Вероятность объединенных символов								Код
x_i	$p(x_i)$	1	2	3	4	5	6	7	8	
									1	
								0,57		
							0,43			
x_1	0,29									00
						0,28				
x_2	0,23									10
					0,20					
				0,15						
x_3	0,13									011
x_4	0,11									110
x_5	0,09									111
x_6	0,08									0100
			0,07							
x_7	0,04									01010
		0,03								
x_8	0,02									010110
x_9	0,01									010111

Лекция 5

ПРИНЦИПЫ ПЕРЕДАЧИ ДАННЫХ ПО КАНАЛАМ С ПОМЕХАМИ

Скорость передачи информации и пропускная способность дискретного канала с помехами

Пусть источник сообщения выдает в канал сообщение X из элементов $x_1, x_2, ..., x_n$, а получатель сообщения принимает некоторое сообщение Y из элементов $y_1, y_2, ..., y_n$. Сообщение передается по каналу с помехами.



Таким образом, при передаче сообщения по каналу с помехами на приемной стороне нельзя однозначно сказать, что было передано. Можно говорить о той или иной вероятности переданного сообщения.

Если принят y_i, то можно лишь говорить, что с различными вероятностями были переданы x_i, где i=1,...,n. Такой процесс передачи описывается математически с помощью условной вероятности $p(x_i/y_j)$ - вероятности того, что был передан x_i при условии принятия элемента y_j.

Если помехи в канале отсутствуют, то $p(x_1/y_1) =1$, $p(x_1/y_2) =0$ и все остальные $p(x_1/y_n) = 0$.

До передачи элементы сообщения характеризуются энтропией $H(x_i)=\log 1/p(x_i)= -\log p(x_i)$. (*)

Величина $H(x_i)$ носит название безусловной или априорной энтропии.

В результате передачи сообщения при наличии помех энтропия уменьшается не до нуля, а до величины остаточной энтропии

$$H(x_i/y_j)=\log 1/p(x_i/y_j) = -\log p(x_i/y_j). \quad (**)$$

Эта энтропия характеризует неопределенность относительно элемента x_i исходного сообщения при условии, что принят элемент y_j .

Величина $H(x_i/y_j)$ носит название условной или апостериорной энтропии.

Определим

$$I(x_i/y_j)= H(x_i) - H(x_i/y_j) = \log (p(x_i/y_j))/p(x_i). \quad (1)$$

как количество информации в принятом элементе y_j относительно переданного элемента x_i. Выполняя усреднение $I(x_i/y_j)$ - частного количества информации - по всем элементам переданного сообщения с учетом $p(x_i)$ и по всем элементам принятого сообщения $p(y_j)$, получим количество информации, содержащееся в принятом сообщении относительно переданного

$$I(X/Y) = -\sum_{i=1}^{n}\sum_{j=1}^{n} p(x_i)\,p(x_i/y_j)\log p(x_i/y_j) = H(X) - H(X/Y). \quad (2)$$

Механизм получения информации при наличии помех

	ДО ПЕРЕДАЧИ	ПОСЛЕ ПЕРЕДАЧИ
ЭНТРОПИЯ	H(X)	H(X/Y)
ИНФОРМАЦИЯ	0	H(X)-H(X/Y)

Рассмотрим случаи:

1. Помехи отсутствуют, т.е. каждому переданному символу соответствует единственный принятый сигнал.

$p(x_i/y_j)=1$ или 0, $H(x_i/y_j) = 0$ следовательно $H(X/Y) =0$ и $I(X/Y)=H(X)$, т.е. потерь информации нет.

2. Уровень помех настолько высок, что принимаемое сообщение оказывается не связанным с передаваемым.

$p(x_i/y_j)= p(x_i)$, $H(x_i/y_j)=H(x_i)$ следовательно $H(X/Y) = H(X)$ и $I(X/Y)=0$, т.е. происходит полная потеря информации.

Таким образом $0 \le I(X/Y) \le H(X) =I(X)$.

Скорость передачи информации по дискретному каналу с помехами

$$R = V_к\, I(X/Y) \text{ бит/с.},$$

где

$V_к$ - скорость передачи символов по каналу;

$I(X/Y)$ - среднее количество информации, переносимое одним символом.

$C = \max R$ - пропускная способность канала при наличии помех.

Очевидно, что при наличии помех реальная пропускная способность канала уменьшается, причем тем в большей степени, чем больше интенсивность помех.

Теорема Шеннона о пропускной способности дискретного канала с помехами

Прямая теорема:

Если производительность источника сообщения P не превышает пропускной способности канала C, т.е. $P \le C$, то несмотря на наличие помех всегда существует такой способ кодирования, при котором передача информации будет передаваться без потерь, причем при $P \to C$, скорость передачи информации так же будет стремиться к C ($R \to C$).

Обратная теорема.

Если $P > C$, то не существует способа кодирования, при котором передача информации осуществлялась бы без потерь.

В этой теореме не указан конкретный способ кодирования, при котором потери информации равны нулю, но доказано его существование.

Пример.

Канал без помех	Канал с помехами
$V_к = 1000$ дв.симв./с	$V_к = 1000$ дв.симв./с
$C_{бп} = 1000$ бит/с	$C_п = 600$ бит/с
$P_{бп} = 1000$ бит/с	$P_п = 600$ бит/с
$V_и^{opt} = 1000$ дв.симв./с	$V_и^{opt} = 600$ дв.симв./с

При наличии в канале помех по каналу передается 1000 дв.симв./с. Из них 400 дв.симв./с является избыточной информацией, т.е. $C_{бп} - C_{п} = \Delta C$.

В соответствии с теоремой Шеннона, единственным методом безошибочной передачи информации по каналу с помехами является введение избыточности.

Лекция 6

СПОСОБЫ ВВЕДЕНИЯ ИЗБЫТОЧНОСТИ ДЛЯ БОРЬБЫ С ПОМЕХАМИ

1. Помехоустойчивое кодирование.
2. Групповые методы защиты от ошибок.
3. Организация систем передачи данных с обратной связью.

Принципы помехоустойчивого кодирования

Различают три типа помехоустойчивых кодов:

- с обнаружением ошибок;
- с исправлением ошибок;
- с обнаружением и исправлением ошибок.

Принцип построения помехоустойчивых кодов заключается в том, что все возможные кодовые комбинации N делятся на две группы: разрешенные $N_и$ (предназначенные для передачи полезной информации) и запрещенные $N_к$ (предназначенные для передачи информации, используемой для целей контроля).

Разрешенные и запрещенные кодовые комбинации подбираются таким образом, чтобы в результате действия любой помехи разрешенные комбинации перешли в запрещенные. Тогда на приемной стороне факт наличия ошибок всегда будет обнаружен.

Рассмотрим основные характеристики помехоустойчивых кодов.

Значность кода n. Под значностью кода понимается длина кодовой комбинации.

Число информационных символов $n_и$. Информационными символами считаются те, которые непосредственно представляют соответствующую букву алфавита в кодовой комбинации.

Число контрольных символов $n_к$. Дополнительные символы, служащие для целей контроля (исправления) информации

$$n_к = n - n_и.$$

Избыточность кода l. Под избыточностью кода понимается относительное увеличение длины кодового слова за счет введения в него контрольных символов

$$l = (n - n_и) / n = 1 - n_и/n.$$

Кодовое расстояние d. Кодовое расстояние - это минимальное расстояние между двумя любыми разрешенными кодовыми комбинациями. Для двоичных сообщений

определяется числом двоичных единиц, получаемых в результате суммирования по модулю два двух разрешенных кодовых комбинаций.

Для выбора кодового расстояния обычно пользуются неравенством

$$d \geq r + s + 1, \qquad (*)$$

где r - число ошибок, которые данный код позволяет обнаружить;

s - число ошибок, которые данный код позволяет исправить, причем

$r \geq s$.

Синтез помехоустойчивых кодов

Задача синтеза помехоустойчивого кода может быть сформулирована следующим образом: дано множество передаваемых сообщений N_u, указаны требования по помехоустойчивости кода. Необходимо синтезировать код, обеспечивающий наиболее экономное (в смысле минимальной избыточности) выполнение указанных требований.

Для этого надо определить:

- n - длину кодового слова (разрядность кода);
- кодовое расстояние d;
- минимально необходимое число контрольных символов $n_к$ и их значения;
- необходимое для передачи заданного множества сообщений число информационных символов n_u;
- число и порядок проверок принятого сообщения.

Кроме того, необходимо установить позиции, на которых должны размещаться контрольные и информационные символы, вычислить избыточность синтезированного кода и сравнить ее с теоретической границей, оценить трудности практической реализации кодирующих устройств.

Сначала определяется число информационных символов n_u, затем - $n_к$.

Число информационных символов определяется на основании множества передаваемых сообщений

$$N_u = 2^{n_u}. \qquad (1)$$

Для определения числа контрольных символов $n_к$ рассуждают следующим образом. Из $n_к$ контрольных символов можно образовать 2 в степени $n_к$ двоичных комбинаций, которые должны дать ответы типа "да" или "нет" на вопросы:

1. Принято ли данное кодовое слово правильно ?

2. Если в нем имеется ошибка, то на какой из n позиций, включая и контрольные ? (Для этого надо опросить n позиций, т.е. задать n вопросов.)

Таким образом, $2^{n_к}$ двоичных комбинаций должны дать ответы не менее, чем на $n + 1$ вопрос, т.е.

$$2^{n_к} \geq n+1. \qquad (2)$$

Поскольку $n = n_u + n_к$, то $2^n = 2^{n_u} 2^{n_к}$, откуда, с учетом (2), следует

$$2^n \geq 2^{n_u}(n+1) \qquad (3)$$

или

$$2^n \geq N_u(n+1). \quad (4)$$

Код с проверкой на четность

Непомехоустойчивый код не имеет избыточности ($n = n_u$ и l =0) и все символы его являются информационными. Кодовое расстояние d у такого кода равно 1 и он не в состоянии не только корректировать, но и обнаруживать ошибки.

Для повышения помехоустойчивости исходного кода может быть предложен следующий способ (См. табл.)

Исходная комбин-я	Контр-й символ	Помех-й код
1	2	3
000	0	0000
001	1	0011
010	1	0101
011	0	0110
100	1	1001
101	0	1010
110	0	1100
111	1	1111

Расположим в таблице все кодовые комбинации исходного безызбыточного кода по порядку (графа 1), а колонку контрольных символов (графа 2) составим по следующему принципу: если в кодовом слове исходного кода число единиц нечетное, контрольный символ должен дополнять его до четного.

Нетрудно видеть, что в результате (графа 3) получился помехоустойчивый код с кодовым расстоянием d =2. Такой код может обнаруживать одну ошибку. Действительно в неравенстве (*) надо положить r = 1, а s = 0, тогда d =2.

Следовательно, непомехоустойчивый исходный код добавлением контрольных символов (одного в каждое слово) превращен в помехоустойчивый, обнаруживающий ошибку. Другими словами, помехоустойчивость кода приобретена ценой избыточности.

Результирующий код имеет избыточность

$$l = (n - n_u) / n = 1 - n_u/n = 1 - 3/4 = 0{,}25.$$

Такой код называют кодом с проверкой на четность (подобным же образом может быть организован код с проверкой на нечетность).

Смысл обнаружения ошибки состоит в том, что на приемной стороне производится контроль принятой комбинации на четное число единиц. Если в ней число единиц оказывается нечетным, то она "бракуется", так как в кодовой комбинации имела место ошибка. Это является основанием для переспроса.

Рассмотренный код является простейшим помехоустойчивым кодом, однако принцип проверки на четность используется во многих достаточно сложных помехоустойчивых кодах.

Лекция 7

КОДЫ ХЕММИНГА

В разработке и создании ряда помехоустойчивых кодов существенная роль отводится различным способам проверки на четность принимаемых кодовых комбинаций. В начале 50-х годов Хеммингом был предложен код, в котором контрольные символы размещались в кодовой комбинации не произвольно, а на строго определенных местах, что, естественно, облегчало декодирование.

Была разработана система проведения проверок правильности переданного кодированного сообщения, включающая алгоритм определения синдрома ошибки, указывающего не только на наличие ошибки, но и номер искаженной кодовой позиции.

Наибольшее распространение получили две модели кода Хемминга: код с обнаружением и исправлением одиночной ошибки (минимальное кодовое расстояние $d = 3$) и код с исправлением одиночной ошибки и обнаружением двойной ($d = 4$).

Для синтеза кода Хемминга необходимо решить следующие задачи:

1. Определить число контрольных символов, обеспечивающих заданные требования по помехозащищенности.
2. Установить, на каких позициях кодовой комбинации следует разместить контрольные символы и какие позиции займут информационные символы.
3. Собрав макет кодовой комбинации, определить значение каждого контрольного символа.
4. Составить кодовые комбинации, включающие как контрольные, так и информационные символы.
5. Дать алгоритм проверок, позволяющий установить наличие и место ошибки.

Синтез кода с d = 3

При этом исходным, как правило , является число информационных символов n_u, которое, в свою очередь, определяется множеством сообщений $N_u = 2^{n_u}$. Иногда может задаваться общее число символов в кодовой комбинации n.

Первая задача.

Определить число контрольных символов $n_к$.

Для этого, если задано n или n_u, необходимо воспользоваться соответственно формулами

$$n_к = \left]lb(n+1)\right[; \qquad (1)$$

$$n_к = \left]lb\left\{(n_u+1)+\left]lb(n_u+1)\right[\right\}\right[, \qquad (2)$$

где] b [- знак округления до ближайшего большего целого числа.

Вторая задача.

Определить места, на которых в общей кодовой комбинации должны располагаться эти контрольные разряды.

Контрольные символы должны составить двоичное число (синдром ошибки), которое бы указывало номер ошибочной позиции.

В результате первой частной проверки на четность получается символ первого (младшего) разряда синдрома, в результате второй проверки - символ второго и т. д.

Итак, если синдром ошибки представить в виде двоичного четырехзначного числа и рядом записать соответствующие десятичные эквиваленты, то получим таблицу 1.

Десят-й эквив-т	Синдром ошибки
1	0001
2	0010
3	0011
4	0100
5	0101
6	0110
7	0111
8	1000
9	1001

Выпишем последовательно в виде небольшой таблицы номера позиций, участвующих в каждой проверке на четность.

В первой проверке должны участвовать те позиции, которые содержат единицу в младшем разряде. Исходя из табл. 1, это будут 1, 3, 5, 9, ...

В второй проверке должны участвовать те позиции, которые содержат единицу во втором разряде. По табл. 1, это будут 2, 3, 6, 7, ...

В третьей проверке должны участвовать 4, 5, 6, 7, ... позиции.

В результате получим таблицу 2.

Номер проверки	Номера поз-й, охватыв-х этой проверкой				
Первая	1	3	5	7	9
Вторая	2	3	6	7	
Третья	4	5	6	7	

Анализируя таблицу 2 можно заключить, что контрольные символы $К_m$ должны размещаться на следующих позициях: $К_1$ на позиции 1, т.е. 2^0;
$К_2$ на позиции 2, т.е. 2^1; $К_3$ на позиции 4, т.е. 2^2; $К_4$ на позиции 8, т.е. 2^3;
$К_m$ на позиции 2^m.

Третья задача.

Определить значение контрольных символов. Составляется макет кода Хемминга. Пусть $n_и$ =4, $n_к$ = 3, n = 7. Тогда в общем виде он выглядит следующим образом:

Номера позиций 1 2 3 4 5 6 7

Символы $К_1$ $К_2$ $И_3$ $К_3$ $И_2$ $И_1$ $И_0$,

где $И_i$ - информационные символы.

Алгоритм определения контрольных символов

Определение значения $К_1$. Составляется сумма по модулю 2 всех символов, включая и $К_1$, размещенных на позициях 1, 3, 5, 7, 9, ... , охватываемых первой проверкой, т.е.

$$К_1 \oplus И_3 \oplus И_2 \oplus И_0. \qquad (3)$$

Значение символа $К_1$ определяется из условия обращения суммы (3) в нуль, т.е. из условия четности. Если число единиц (без $К_1$) нечетное, то $К_1$=1, если четное, то $К_1$=0.

Определение значения $К_2$. Составляется сумма по модулю 2 всех символов, включая и $К_1$, размещенных на позициях, охватываемых второй проверкой, т.е.

$$К_2 \oplus И_3 \oplus И_1 \oplus И_0. \qquad (4)$$

Значение символа $К_2$ определяется из условия обращения суммы (4) в нуль, т.е. из условия четности. Если число единиц (без $К_2$) нечетное, то $К_2$=1, если четное, то $К_2$=0.

Определение значения $К_3$. Составляется сумма по модулю 2 всех символов, включая и $К_3$, размещенных на позициях, охватываемых третьей проверкой т.е.

$$К_3 \oplus И_2 \oplus И_1 \oplus И_0. \qquad (5)$$

Символ $К_3$ одолжен обращать в нуль двоичную сумму (5).

Аналогично находятся $К_4$, $К_5$ и остальные символы, только для этого требуется проводить четвертую, пятую и остальные проверки и составлять соответствующие суммы.

Четвертая задача.

Составить кодовые комбинации. Для этого надо выписать все комбинации исходного безызбыточного двоичного кода и, пользуясь макетом кодового слова, а так же вычисленными на основании сумм (3), (4), (5) и так далее значениями контрольных символов $К_m$, записать все $n_и$ комбинаций кода Хемминга.

Что касается пятой задачи - разработки алгоритма проверок, то она будет рассмотрена ниже, так как относится к декодированию кода Хемминга.

Рассмотрим пример синтеза кода Хемминга.

Пусть имеется ансамбль из 16 сообщений и его необходимо закодировать кодом Хемминга (d = 3).

Решение:

1) определяем число контрольных символов. Поскольку задан ансамбль сообщений $N_и$ = 16, то задано число информационных символов $n_и$=lb16=4. Следовательно необходимо пользоваться формулой

$$n_к = \left\lceil lb\left\{(n_и + 1) + \left\lceil lb(n_и + 1)\right\rceil\right\}\right\rceil = \left\lceil lb\{5 + 3\}\right\rceil = lb\,8 = 3.$$

Таким образом, $n_к = 3$, $n = n_и + n_к = 4 + 3 = 7$.

2) определяем позиции, на которых должны быть размещены эти три контрольных символа. Это будут позиции 1, 2 и 4 (см. таб. 2).

3) составляем таблицу комбинаций двоичного безызбыточного кода, не заполняя первую, вторую и четвертую графы. Всего комбинаций будет 16 (табл. 3).

4) находим значения контрольных символов:

для нулевой комбинации все $К_j$=0.

Для символа $К_1$.

Значение символа $К_1$ определяется из уравнения

$$К_1 \oplus И_3 \oplus И_2 \oplus И_0 = 0.$$

а) $К_1 = 0$

б) $К_1 \oplus 0 \oplus 0 \oplus 1 = 0$; $К_1 = 1$

в) $К_1 \oplus 0 \oplus 0 \oplus 0 = 0$; $К_1 = 0$

г) $К_1 \oplus 0 \oplus 1 \oplus 0 = 0$; $К_1 = 1$

д) $К_1 \oplus 0 \oplus 1 \oplus 0 = 0$; $К_1 = 1$

е) $К_1 \oplus 0 \oplus 1 \oplus 1 = 0$; $К_1 = 0$

Таблица 3.

Комби	Номера	1	2	3	4	5	6	7	Допол
нация	п/п	$К_1$	$К_2$	$И_3$	$К_3$	$И_2$	$И_1$	$И_0$	$К_{доп}$
а	0	0	0	0	0	0	0	0	0
б	1	1	1	0	1	0	0	1	0
в	2	0	1	0	1	0	1	0	1
г	3	1	0	0	0	0	1	1	1
д	4	1	0	0	1	1	0	0	1
е	5	0	1	0	0	1	0	1	1
	6			0		1	1	0	
	7			0		1	1	1	
	8			1		0	0	0	
	9			1		0	0	1	
	10			1		0	1	0	
	11			1		0	1	1	
	12			1		1	0	0	
	13			1		1	0	1	
	14			1		1	1	0	
	15			1		1	1	1	

Для символа $К_2$.

Значение символа $К_2$ определяется из уравнения

$$К_2 \oplus И_3 \oplus И_1 \oplus И_0 = 0.$$

а) $К_2 = 0$

б) $К_2 \oplus 0 \oplus 0 \oplus 1 = 0$; $К_2 = 1$

в) $К_2 \oplus 0 \oplus 1 \oplus 0 = 0$; $К_2 = 1$

г) $К_2 \oplus 0 \oplus 1 \oplus 1 = 0$; $К_2 = 0$

д) $К_2 \oplus 0 \oplus 0 \oplus 0 = 0$; $К_2 = 0$

е) $К_2 \oplus 0 \oplus 0 \oplus 1 = 0$; $К_2 = 1$

Для символа $К_3$.

Значение символа $К_3$ определяется из уравнения

$$К_3 \oplus И_2 \oplus И_1 \oplus И_0 = 0.$$

а) $К_3 = 0$

б) $К_3 \oplus 0 \oplus 0 \oplus 1 = 0$; $К_3 = 1$

в) $К_3 \oplus 0 \oplus 1 \oplus 0 = 0$; $К_3 = 1$

г) $К_3 \oplus 0 \oplus 1 \oplus 1 = 0$; $К_3 = 0$

д) $К_3 \oplus 1 \oplus 0 \oplus 0 = 0$; $К_3 = 1$

е) $К_3 \oplus 1 \oplus 0 \oplus 1 = 0$; $К_3 = 0$

Проставляем значение контрольных символов в табл. 3. Код синтезирован. Поставленная задача решена. (Студенту предлагается в порядке упражнения самостоятельно завершить заполнение табл. 3).

На примере полученного кода Хемминга с $d = 3$ можно показать, как строится код Хемминга с $d = 4$, позволяющий обнаруживать двухкратные ошибки. Число контрольных символов в таком коде должно быть на единицу больше, т.е. для рассматриваемого примера $n_к = 3 + 1 = 4$, следовательно, код будет восьмипозиционный.

Принцип получения такого кода следующий:

- к каждой кодовой комбинации добавляется еще один дополнительный контрольный символ, позволяющий осуществить общую проверку на четность;
- значение дополнительного контрольного символа определяется исходя из наличия в каждой комбинации кода Хемминга (т.е. учитывая и контрольные символы) четного числа единиц. Таким образом, в нашей табл. 3 появляется еще одна графа $К_{доп}$;
- заполняем эту графу, т.е. определяем значение контрольных символов и размещаем их на последней (восьмой) кодовой позиции:

а) $К_{доп} = 0$

б) $К_{доп} = 0$

в) $К_{доп} = 1$

г) $К_{доп} = 1$

д) $К_{доп} = 1$

е) $К_{доп} = 1$

Лекция 8

Декодирование кода Хемминга

Вначале рассмотрим декодирование кода Хемминга при $d = 3$ (рис. 1):

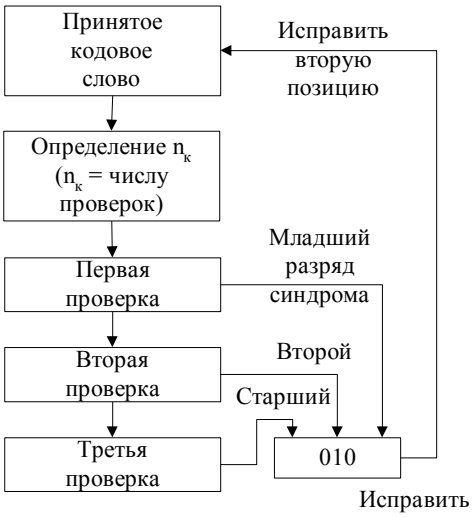


Рис. 1. Блок-схема алгоритма декодирования кода Хемминга ($d = 3$)

- полученная кодовая комбинация подвергается ряду проверок на четность. Число проверок равно числу контрольных символов в кодовой комбинации и заранее известно;
- заранее известно так же, на каких позициях эти контрольные символы размещены;
- известны номера позиций, охватываемых соответствующими проверками;
- результат каждой проверки записывается в виде соответствующего разряда двоичного числа, которое называется синдромом ошибки. В результате первой проверки получается младший разряд синдрома;
- если принятая комбинация не содержит ошибок, каждая проверка в результате дает 0;
- если на какой-либо позиции имеется однократная ошибка, то охватывающая эту позицию проверка на четность дает 1, что свидетельствует о наличии ошибки на одной из позиций;
- двоичное число, полученное в результате всех проверок (последняя проверка дает старший разряд синдрома), в десятичном эквиваленте укажет номер ошибочной позиции;
- заменив искаженный символ на обратный, мы исправим ошибку;
- контрольные символы (места расположения известны) выбрасываются, остающиеся комбинации являются рабочими.

Рассмотрим теперь декодирование кода Хемминга при $d = 4$ (рис. 2):

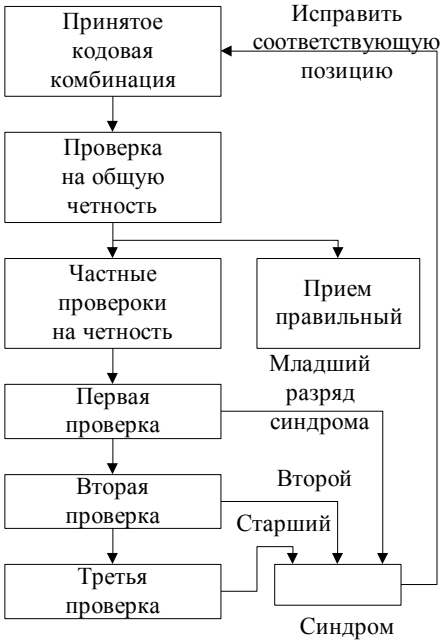


Рис. 2. Блок-схема алгоритма декодирования кода Хемминга (d =4)

•принятые кодовые комбинации проверяются на общую четность;

• нечетное число единиц свидетельствует о наличии ошибки;

• место ошибки находится последовательными частными проверками на четность (в частных проверках дополнительные контрольные символы не участвуют);

• если частные проверки показывают ошибку, а общая проверка на четность нет, (рис. 3), то это значит, что в принятой комбинации имеет место двойная ошибка, исправить которую данный код не позволяет.

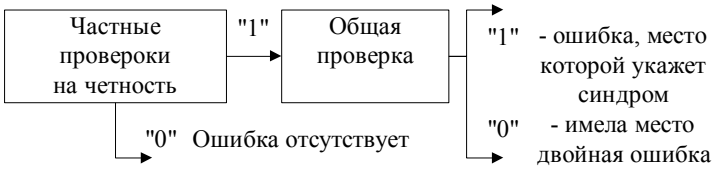


Рис. 3. Блок-схема обнарурения двухкратной ошибки

Таким образом, код Хемминга с d = 4 позволяет обнаруживать двойные ошибки (но не позволяет такие ошибки исправлять).

Примеры

Рассмотрим несколько примеров.

Пример 1. Передана кодовая комбинация "0100101", закодированная кодом Хемминга с d = 3. Принята комбинация "0100111". Показать процесс выявления ошибки и указать переданную информационную комбинацию.

Решение.

1. Определим число контрольных символов

$n_к$ = lb (n+1) = lb (7+1) = 3.

Следовательно, число проверок равно 3.

2. Первая проверка охватывает позиции 1, 3, 5, 7. Проверка дает

0 ⊕ 0 ⊕ 1 ⊕ 1 =0. Таким образом, в младший разряд синдрома ошибки записывается 0. Вторая проверка охватывает позиции 2, 3, 6, 7. Эта проверка дает 1 ⊕ 0 ⊕ 1 ⊕ 1 =1 (число единиц нечетное). В первый разряд синдрома ошибки должна быть

записана 1. Третья проверка охватывает позиции 4, 5, 6, 7. Эта проверка дает 0 ⊕ 1 ⊕ 1 ⊕ 1 =1 (число единиц нечетное). Во второй (старший) разряд синдрома ошибки должна быть записана 1.

3. Таким образом, синдром ошибки равен "110", что соответствует десятичному числу 6. Следовательно, ошибка в принятой комбинации имела место на шестой позиции, где "1" надо исправить на "0".

4. Переданная комбинация "0100101". Информационная комбинация получается путем выбрасывания контрольных символов, которые размещаются на 1, 2 и 4 позициях, т.е. ~~0~~1~~0~~0101. Окончательно получается "0101".

Пример 2. Построить код Хемминга с $d = 3$ для информационной комбинации "0101". Это обратная задача.

Решение.

1. Число информационных символов $n_и = 4$. Определим число контрольных символов. Для этого воспользуемся формулой

$n_к$ =]lb {($n_и$ +1) +]lb ($n_и$ +1)[}[=]lb {(4 +1) +]lb (4 +1)[}[=

=]lb (5 +]lb 5[)[=]lb (5 + 3)[= 3.

Таким образом, число контрольных символов $n_к = 3$, а занимают они позиции 2^{m-1} , т.е. $К_1$ на первой, $К_2$ на второй, $К_3$ на четвертой позициях.

2. Макет слова, закодированного кодом Хемминга, будет выглядеть следующим образом

$К_1$ $К_2$ 0 $К_3$ 1 0 1 .

3. Определим значение контрольных символов, составив уравнения на основе таблицы проверочных позиций

Первая проверка $К_1 ⊕ 0 ⊕ 1 ⊕ 1 = 0$, откуда $К_1 = 0$;

Вторая проверка $К_2 ⊕ 0 ⊕ 0 ⊕ 1 = 0$, откуда $К_2 = 1$;

Третья проверка $К_3 ⊕ 1 ⊕ 0 ⊕ 1 = 0$, откуда $К_3 = 0$.

4. Окончательно корректирующий код Хемминга имеет вид "0100101".

Пример 3. Переданы следующие кодовые комбинации в коде Хемминга: "1101001","0001111", "0111100". Получены "1001001", "0011111", "0110100". Показать процесс обнаружения ошибки.

Решение.

1. Принята комбинация "1001001":

а) составляем уравнения для определения элементов двоичного числа - синдрома ошибки. Для этого из таблицы выписываем номера позиций, охватываемых первой, второй и третьей проверками. Больше нам не нужно, так как число проверок равно числу контрольных символов в кодовой комбинации. В нашем случае общее число символов в комбинации $n = 7$. Число контрольных символов определяется из формулы и равно 3, следовательно трех проверок достаточно;

б) первая проверка охватывает позиции 1, 3, 5, 7

$1 ⊕ 0 ⊕ 0 ⊕ 1 = 0$.

Следовательно, в младшем разряде двоичного числа - синдрома ошибки должен быть записан 0;

в) вторая проверка охватывает позиции 2, 3, 6, 7

$$0 \oplus 0 \oplus 0 \oplus 1 = 1.$$

Следовательно, в первом разряде синдрома ошибки должна быть записана 1;

г) третья проверка охватывает позиции 4, 5, 6, 7

$$1 \oplus 0 \oplus 0 \oplus 1 = 0.$$

Следовательно, в старшем (втором) разряде синдрома ошибки должен быть записан 0.

Таким образом, синдром ошибки для данного кодового сочетания - 010, что соответствует десятичной цифре 2. Следовательно, ошибочный разряд второй.

2. Для второй кодовой комбинации "0001111":

$$0 \oplus 1 \oplus 1 \oplus 1 = 1.$$
$$0 \oplus 1 \oplus 1 \oplus 1 = 1.$$
$$1 \oplus 1 \oplus 1 \oplus 1 = 0.$$

Ошибочный разряд - третий.

3. Для кодовой комбинации ""0110100":

$$0 \oplus 1 \oplus 1 \oplus 0 = 0.$$
$$1 \oplus 1 \oplus 0 \oplus 0 = 0.$$
$$0 \oplus 1 \oplus 0 \oplus 0 = 1.$$

Ошибочный разряд - четвертый.

Пример 4. Передана кодовая комбинация "01001011", закодированная кодом Хемминга с $d = 4$. Показать процесс выявления ошибки.

Решение.

Принята комбинация "01001111":

а) проверка на общую четность указывает на наличие ошибки (число единиц четное);

б) частные проверки производятся так же, как это было в других примерах.

При составлении проверочных сумм последние единицы кодовых комбинаций (дополнительные контрольные символы) не учитываются.

2. Принята комбинация "01101111":

а) проверка на общую четность показывает, что ошибка не фиксируется;

б) частные проверки (последний символ отбрасывается)

Первая проверка $0 \oplus 1 \oplus 1 \oplus 1 = 1$

Вторая проверка $1 \oplus 1 \oplus 1 \oplus 1 = 0$

Третья проверка $0 \oplus 1 \oplus 1 \oplus 1 = 1$

Таким образом, частные проверки фиксируют наличие ошибки. Она, якобы, имела место на пятой позиции. Но так как при этом первая проверка на общую четность ошибки не зафиксировала, то значит имела место двойная ошибка. Исправить двойную ошибку такой код не может.

Пример 5. Какой вид имеют комбинации корректирующего кода Хемминга для передачи сообщений "1101", "1011".

Решение.

1. Для сообщения "1101":

а) $n_и = 4$; находим $n_к =]lb \{(n_и +1) +]lb (n_и +1)[\}[= 3$.

б) составляем макет кодового слова

$К_1$ $К_2$ 1 $К_3$ 1 0 1 .

в) определим значение контрольных символов на основании уравнений

Первая проверка $К_1 \oplus 1 \oplus 1 \oplus 1 = 0$, откуда $К_1 = 1$;

Вторая проверка $К_2 \oplus 1 \oplus 0 \oplus 1 = 0$, откуда $К_2 = 0$;

Третья проверка $К_3 \oplus 1 \oplus 0 \oplus 1 = 0$, откуда $К_3 = 0$.

Закодированная комбинация имеет вид "1010101".

2. Для сообщения "1011":

а) $n_к = 3$;

б) $К_1$ $К_2$ 1 $К_3$ 0 1 1 ;

в) Первая проверка $К_1 \oplus 1 \oplus 0 \oplus 1 = 0$, откуда $К_1 = 0$;

Вторая проверка $К_2 \oplus 1 \oplus 1 \oplus 1 = 0$, откуда $К_2 = 1$;

Третья проверка $К_3 \oplus 0 \oplus 1 \oplus 1 = 0$, откуда $К_3 = 0$.

Закодированная комбинация имеет вид "0110011".

Коды Хемминга обладают высокой помехозащищенностью и с успехом могут использоваться в каналах передачи информации, если статистика показывает, что в таких каналах возникновение ошибок с большой кратностью маловероятно, точнее, если имеются основания считать, что наиболее вероятными ошибками являются одиночные.

Правда, высокая помехозащищенность кодов Хемминга достигается ценой значительной избыточности.

Например, для кода (7,4) избыточность $l=(n-n_и)/n=1-n_и/n=1-4/7=0{,}429$

Действительно, с помощью общего числа символов, входящих в кодовую комбинацию кода Хемминга $n=-n_и+ n_к =7$, можно было бы передать $2^7 = 128$ сообщений, из которых мы используем лишь $2^4 = 16$, т.е. приблизительно 12,5 %.

Лекция 9

ЦИКЛИЧЕСКИЕ КОДЫ

Циклические коды относятся к систематическим разделимым кодам. Две характерные особенности этих кодов привлекают к себе внимание.

Прежде всего практически неограниченные корректирующие возможности, что делает их особенно привлекательными в сложной помеховой обстановке, когда статистический анализ указывает на возможность появления в канале связи

многократных и особенно пачечных ошибок, т.е. условий, при которых коды Хемминга не в состоянии обеспечить заданную помехоустойчивость.

Вторая особенность заключается в чрезвычайной простоте инженерной реализации кодирующих и декодирующих устройств циклических кодов.

Отмеченные достоинства привели к тому, что в настоящее время циклические коды получили весьма широкое распространение.

Теоретические основы циклических кодов

Теоретической основой для разработки циклических кодов, методов их синтеза, разработки алгоритмов кодирования и декодирования является область абстрактной алгебры, которая получила название алгебры групп, в силу чего циклические коды иногда называют групповыми.

Для кодирования и декодирования используется алгебра двоичных многочленов (многочленов с коэффициентами 0 и 1) или, как говорят, многочленов над двоичным полем, полем Галуа. Поле Галуа обозначается GF(2).

Полем называется множество, состоящее из элементов различной природы, для которого определены законы двух основных операций: сложения и умножения. Эти основные операции должны удовлетворять законам коммутативности и ассоциативности.

Сложение Умножение

$a + b = b + a$ - коммутативность - $ab = ba$

$a + (b + c) = (a + b) + c$ - ассоциативность - $a(bc) = (ab)сю$

Данные операции связаны законом дистрибутивности

$[(a +) c = ac + bc],$

причем сложение обладает обратной операцией - вычитанием.

Обе эти операции должны обладать условию замкнутости: в результате сложения двух элементов некоторого множества a и b может быть получен только такой элемент $c,$ который также принадлежит данному множеству,

т.е. если $a \in G, b \in G,$ и $a + b = c,$ то и $c \in G.$ То же относится и ко второй операции.

Действительно, если допустить, что в результате сложения (или умножения) может быть получен элемент, не принадлежащий данному множеству, т.е. если требования замкнутости не выполняются, то при кодировании в результате этих операций мы можем выйти за пределы принятой системы кодирования.

Может возникнуть вопрос, какое отношение имеют эти алгебраические рассуждения к рассматриваемой проблеме помехоустойчивого кодирования? Оказывается самое прямое.

Каждая кодовая комбинация может быть представлена в виде многочлена соответствующей степени некоторой абстрактной переменной x.

Например, комбинация "10101" - $1\bullet2^4 + 0\bullet2^3 + 1\bullet2^2 + 0\bullet2^1 + 1\bullet2^0$ может быть записана в виде многочлена

$$1\bullet x^4 + 0\bullet x^3 + 1\bullet x^2 + 0\bullet x^1 + 1\bullet x^0 = x^4 + x^2 + 1.$$

Мы видим, что коэффициентами при абстрактной переменной x являются либо 1, либо 0, т.е. элементы GF(2), а показатели степени соответствуют номерам разрядов. Та

же комбинация в десятичном исчислении соответствует числу 21. Мы получаем, таким образом, различные представления двоичных многочленов

$$x^4 + x^2 + 1 \rightarrow 21 \rightarrow 10101.$$

Циклические коды характерны тем, что все комбинации данного кода могут быть образованы из одной начальной комбинации путем циклического сдвига справа налево, при этом символ крайнего левого ряда перемещается на место младшего разряда - в конец комбинации. Например, исходная комбинация "10101":

1 сдвиг 01011
2 сдвиг 10110
3 сдвиг 01101
4 сдвиг 11010
5 сдвиг 10101,

т.е. повторилась исходная комбинация. Такая особенность и объясняет название этих кодов - циклические.

Циклический сдвиг кодовой комбинации аналогичен умножению соответствующего многочлена на x:

$$000101 \rightarrow 0\bullet x^5 + 0\bullet x^4 + 0\bullet x^3 + 1\bullet x^2 + 0\bullet x^1 + 1\bullet x^0 = x^2 + 1;$$
$$001010 \rightarrow 0\bullet x^5 + 0\bullet x^4 + 1\bullet x^3 + 0\bullet x^2 + 1\bullet x^1 + 0\bullet x^0 = x^3 + x;$$
$$010100 \rightarrow 0\bullet x^5 + 1\bullet x^4 + 0\bullet x^3 + 1\bullet x^2 + 0\bullet x^1 + 0\bullet x^0 = x^4 + x^2;$$
$$101000 \rightarrow 1\bullet x^5 + 0\bullet x^4 + 1\bullet x^3 + 0\bullet x^2 + 0\bullet x^1 + 0\bullet x^0 = x^5 + x^3;$$

$$(x^2 + 1)\, x \rightarrow x^3 + x \rightarrow 001010;$$
$$(x^3 + x)\, x \rightarrow x^4 + x^2 \rightarrow 010100;$$
$$(x^4 + x^2)\, x \rightarrow x^5 + x^3 \rightarrow 101000.$$

Если степень многочлена достигает разрядности кода, то происходит "перенос" в нулевую степень при x и цикл повторяется. В шифраторах циклических кодов эта операция осуществляется путем соединения выхода ячейки старшего разряда со входом ячейки нулевого разряда. Сложение по модулю 2 любых двух соседних комбинаций равносильно операции умножения многочлена, соответствующего комбинации первого слагаемого, на многочлен x + 1, если приведение подобных членов осуществляется по модулю 2.

Пример:

$$\begin{array}{l} 000101 \\ \oplus \\ \underline{001010} \\ 001111 \end{array}$$

$$\begin{array}{l} x^2 + 0 + 1 \rightarrow 000101 \\ \underline{\quad x + 1} \\ x^2 + 0 + 1 \\ \underline{x^3 + 0 + x} \\ x^3 + x^2 + x + 1 \rightarrow 001111. \end{array}$$

Таким образом , любая кодовая комбинация циклического кода может быть получена путем умножения специально подобранного многочлена на некоторый другой многочлен.

Методы построения циклических кодов

Для построения циклических кодов принципиальное значение имеют так называемые образующие (генераторные) многочлены. В качестве образующих используются многочлены, неприводимые над полем двоичных чисел. Многочлен называется неприводимым, если он делится без остатка только на себя или на единицу.

В качестве информационных символов (И) для построения циклических кодов используются комбинации двоичного безызбыточного кода.

Если некоторую комбинацию безызбыточного кода G(X) умножить на образующий многочлен P(X), в результате получится комбинация циклического кода F(X), обладающего уже некоторой помехоустойчивостью. Корректирующие свойства циклического кода определяются видом образующего многочлена P(X). Полученный код, однако, не будет систематическим: контрольные символы будут располагаться бессистемно, на произвольных местах. Это, естественно, затрудняет декодирование.

Схема дешифратора и процедура декодирования существенно упрощается, если контрольные символы размещать на строго определенных местах (мы уже использовали эту идею, рассматривая коды Хемминга).

В циклических кодах для контрольных символов отводятся места после информационных символов - в конце кодовой комбинации. Для этого достаточно исходную кодовую комбинацию G(X) умножить на одночлен, степень которого равна $n_к$. Такое умножение соответствует повышению степени исходного многочлена на величину $n_к$, что эквивалентно приписыванию справа такого же числа нулей.

Например, исходная комбинация 1010, т.е. $G(X) = x^3 + x$. Если необходимое по условиям помехоустойчивости число контрольных символов равно трем ($n_к$ =3), то одночлен равен x^3 и, следовательно,

$G(X)\, x^3 = (x^3 + x)\, x^3 = x^6 + x^4 \rightarrow 1010\underline{000}$.

Мы видим, что в новой семиразрядной кодовой комбинации в конце "приготовлены" три места для размещения трех контрольных символов. Теперь надо определить значения трех контрольных символов.

Рассмотрим этот процесс на примере.

Пусть требуется закодировать комбинацию

$G(X) = x^3 + x^2 + 1 \rightarrow 1101$. Для выбора производящих многочленов в литературе имеются подробные таблицы (воспользуемся одной из них).

$n_к$	P(X)
1	$x + 1$
2	$x^2 + x + 1$
3	$x^3 + x + 1$
4	$x^4 + x + 1$
5	$x^5 + x^2 + 1$

Не обосновывая пока свой выбор (это будет сделано ниже), возьмем многочлен P(X) = $x^3 + x + 1 \rightarrow 1011$.

Далее проделаем следующее: умножим G(X) на одночлен той же степени, что и P(X). Мы уже видели, что при этом "освобождаются места для размещения контрольных символов

G(X) $x^3 = (x^3 + x^2 + 1)\, x^3 = (x^6 + x^5 + x^3) \rightarrow 1101000$.

Разделим произведение $G(X)\,X^{n_к}$ на образующий полином P(X):

$x^6 + x^5 + x^3$ | $x^3 + x + 1$

$x^6 + x^4 + x^3$ | x^3+x^2+x+1

$x^5 + x^4$

$x^5 + x^3 + x^2$

$x^4 + x^3 + x^2$

$x^4 + x^2 + x$

$x^3 + x$

$x^3 + x + 1$

1

т.е. (G(X) x^3) / P(X) = $(x^3+x^2+x+1) + 1 / (x^3 + x + 1)$.

Таким образом, в результате деления мы получим частное Q(X) той же степени, что и кодируемая комбинация G(X):

Q(X) = $x^3+x^2+x+1 \rightarrow 1111$, G(X) = $x^3 + x + 1 \rightarrow 1101$ и остаток R(X) = 1.

В общем виде можно записать

$$\frac{G(X)\,X^{n_к}}{P(X)} = Q(X) + \frac{R(X)}{P(X)} \qquad (1)$$

или, умножив обе части на образующий многочлен P(X):

$$G(X)X^{n_к} = Q(X)P(X) + R(X)\,. \quad (2)$$

Следовательно, кодовая комбинация циклического кода

$$F(X) = Q(X)P(X) = G(X)X^{n_к} + R(X) \quad (3)$$

может быть получена двумя способами:

1) путем умножения комбинации Q(X), являющейся одной из комбинаций безызбыточного кода, подлежащего преобразованию в циклический код: Q(X) $\rightarrow$ 1111 на образующий полином P(X);
2) в результате умножения заданной комбинации безызбыточного кода G(X) на одночлен $X^{n_к}$, имеющий ту же степень, что и образующий многочлен P(X), и добавления к произведению остатка R(X), полученного от деления произведения $G(X)X^{n_к}$ на образующий многочлен P(X).

Продолжая начатый пример можем записать

F(X) = $(x^3+x^2+x+1)\,(x^3+x+1) = (x^3+x+1)x^3 + 1$ или в двоичной записи

F(X) = 1111•1011 = 1101000 + 001 = 1101 001.

$n_и$ $n_к$

Лекция 10

Циклические коды с d =2, обнаруживающие одиночную ошибку

Такой код может быть получен с помощью образующего многочлена P(X) = x + 1 $\rightarrow$ 11.

Пусть имеется заданная для кодирования комбинация

G(X) = $x^3+x^2+1 \rightarrow 1101$.

Далее умножим G(X) на $X^{n_к}$:

$$G(X)X^{n_к} = (x^3 + x^2 + 1)x = x^4 + x^3 + x \rightarrow 11010.$$

Эта операция в нашем случае ($X^{n_к} = x$, т.е. $n_к$ =1) эквивалентна приписыванию к исходной кодовой комбинации нуля справа. Разделим произведение на P(X) и находим, что остаток R(X) = 1. Таким образом, кодовый многочлен циклического кода будет иметь вид

$$F(X) = G(X)X^{n_к} + R(X) = x^4 + x^3 + x + 1 \rightarrow 11011.$$

Итак, мы разместили единственный в нашем случае контрольный символ (он оказался равным 1) на приготовленное для него первой операцией место.

$n_и$ $n_к$

В закодированнном сообщении 1101 1, n = 5, $n_и$=4, $n_к$ =1.

Полученное сообщение является одним из ансамбля N = 2^4 = 16 сообщений. Каждое из них нужно кодировать таким же образом. Чтобы избежать остальных 15 (или в общем случае $2^{n_и} - 1$) расчетов прибегают к использованию образующей матрицы.

Образующая матрица

Образующая матрица получается из отраженной единичной матрицы (ОЕМ) путем приписывания к ней справа матрицы дополнений

$$M(n_и, n_к) = \left\| E^{'} R \right\|. \quad (1)$$

Матрица дополнений получается из остатков от деления единицы с нулями на образующий многочлен P(X). Для образующего многочлена P(X)=x+1 $\rightarrow$ 11 все остатки равны единице и образующая матрица имеет вид

$$M(n_и, n_к) = \left\| \begin{matrix} a_1 0001\ 1 \\ a_2 0010\ 1 \\ a_3 0100\ 1 \\ a_4 1000\ 1 \\ n_и \quad n_к \end{matrix} \right\|$$

Четыре строки образующей матрицы представляют собой комбинации циклического кода. Нулевая комбинация состоит из нулей a_0=00000. Последняя 16-я комбинация получается в результате суммирования по модулю 2 всех четырех строк образующей матрицы. Нетрудно видеть, что a_{15}=11110. Остальные кодовые комбинации получаются

путем суммирования по модулю 2 всех возможных сочетаний строк образующей матрицы.

Коды, позволяющие образовать новую комбинацию путем сложения по модулю 2 двух или нескольких уже закодированных комбинаций, называются линейными.

Циклические коды с d =3

Для синтеза циклического кода необходимо решить следующие задачи:

1. Определить число контрольных символов $n_к$.
2. Выбрать образующий многочлен P(X).
3. Найти элементы дополнительной матрицы.
4. Составить образующую матрицу.
5. Найти все комбинации циклического кода.

Число контрольных символов. Данное число $n_к$ определяется точно так же, как при синтезе кода Хемминга - по одной из двух формул, в зависимости от того, из чего мы исходим: из полного числа символов в кодовой комбинации n или из числа информационных символов $n_и$.

Образующий многочлен. Выбирается из таблиц. При этом необходимо руководствоваться следующими требованиями:

а) степень образующего многочлена должна быть равна числу контрольных символов в кодовых комбинациях, определенному в п.1. Если, например, $n_к$ =3, то из таблицы образующих многочленов можно выбрать любой многочлен степени 3;

б) из приведенных в таблице многочленов необходимой степени рекомендуется выбирать наиболее короткий;

в) число не нулевых членов образующего многочлена P(X) не должно быть меньше кодового расстояния d.

Элементы дополнительной матрицы. Каждая строка ОЕМ с приписанными справа $n_к$ нулями делится на выбранный образующий многочлен до получения остатка, из которых формируются строки дополнительной матрицы. Необходимо руководствоваться следующими требованиями:

а) число строк дополнительной матрицы должно быть равно числу строк ОЕМ или числу строк образующей матрицы, т.е. числу информационных символов $n_и$. Следовательно, число необходимых для образования дополнительной матрицы остатков должно быть равно числу информационных символов $n_и$.

б) число столбцов дополнительной матрицы (т.е. разрядность остатков) должно быть равно числу контрольных символов $n_к$, т.е. степени образующего многочлена.

Составление образующей матрицы. Берется ОЕМ и к ней справа приписываются элементы дополнительной матрицы.

Нахождение всех комбинаций циклического кода. Все комбинации циклического кода находятся путем суммирования по модулю два всех возможных сочетаний строк образующей матрицы.

Рассмотрим методику построения циклического кода, обнаруживающего двойные ошибки с d = 3.

Пример. Образовать циклический код, позволяющий обнаруживать двухкратные ошибки или исправлять одиночную, из всех комбинаций двоичного кода на все сочетания с числом информационных символов $n_и$=4.

Решение. Определяем число контрольных символов. Так как задано $n_и$=4, то пользуемся формулой

$$n_к =]lb(n_и+1)+]lb(n_и+1)[[=]lb(5+3)[=3.$$

Из таблицы многочленов выбираем один из образующих многочленов третьей степени, причем такой, чтобы число не нулевых членов в нем было не меньше кодового расстояния (d=3), т.е. не меньше 3 $P(X)=x^3+x+1 \rightarrow 1011$.

Находим остатки от деления единицы с соответствующим числом нулей на P(X):

0001000 1011
1011
011 1-й остаток

0010000 1011
1011
110 2-й остаток

0100000 1011
1011
1100
1011
111 3-й остаток

1000000 1011
1011
1100
1011
1110
1011
101 4-й остаток

Этих остатков должно быть четыре, так как $n_и$=4.

Составляем образующую матрицу (a_i -строки; $И_j$ - информационные символы; $К_l$ - контрольные символы):

$$M(n_u, n_к) = \begin{Vmatrix} И_4 И_3 И_2 И_1 & К_1 К_2 К_3 & \\ 0\ 0\ 0\ 1 & 0\ 1\ 1 & a_1 \\ 0\ 0\ 1\ 0 & 1\ 1\ 0 & a_2 \\ 0\ 1\ 0\ 0 & 1\ 1\ 1 & a_3 \\ 1\ 0\ 0\ 0 & 1\ 0\ 1 & a_4 \end{Vmatrix}$$

Находим все остальные комбинации циклического кода:

5. $a_1 \oplus a_2$=0011101

6. $a_1 \oplus a_3 = 0101100$
7. $a_1 \oplus a_4 = 1001110$
8. $a_2 \oplus a_3 = 0110001$
9. $a_2 \oplus a_4 = 1010011$
10. $a_3 \oplus a_4 = 1100010$
11. $a_1 \oplus a_2 \oplus a_3 = 0111010$
12. $a_1 \oplus a_2 \oplus a_4 = 1011000$
13. $a_1 \oplus a_3 \oplus a_4 = 1101001$
14. $a_2 \oplus a_3 \oplus a_4 = 1110100$
15. $a_1 \oplus a_2 \oplus a_3 \oplus a_4 = 1111111$

и 16-ю нулевую комбинацию - 0000000.

Тот же результат можно получить, если каждую из строк ОЕМ умножить на образующий многочлен. Образующий многочлен $P(X)=x^3+x+1 \rightarrow 1011$. Первая строка ОЕМ 0001. В результате умножения получаем

$$\begin{array}{l} 0001 \\ \times\ \underline{1011} \\ \quad 0001 \\ \quad\ 0001 \\ \underline{0001} \\ 0001011. \end{array}$$

Вторая строка ОЕМ - 0010. В результате умножения на образующий полином получаем

$$\begin{array}{l} 0010 \\ \times\ \underline{1011} \\ \quad 0010 \\ \quad\ 0010 \\ \underline{0010} \\ 0010110. \end{array}$$

Аналогично в результате умножения на образующий многочлен третьей строки имеем 0101100, а для четвертой - 1011000. Таким образом, получаем следующие четыре комбинации циклического кода:

$$\begin{array}{l} 000\underline{1011} \\ 00\underline{1011}0 \\ 0\underline{1011}00 \\ \underline{1011}000. \end{array}$$

Видно, что эти комбинации образованы циклической перестановкой образующего многочлена $P(X) \rightarrow 1011$. Остальные комбинации циклического кода могут быть, как и ранее, получены в результате суммирования по модулю 2 приведенных четырех комбинаций. Можно убедиться, что код оказывается полностью совпадающим с предыдущим, полученным на основе образующей матрицы.

Циклический код с d=4

Этот код позволяет обнаруживать три ошибки.

Последовательность синтеза циклического кода в этом случае такая.

1. Как и ранее, вначале определяется число контрольных символов $n_к$, необходимое для обеспечения заданной помехоустойчивости. Начинают с нахождения числа контрольных символов для кода с d=3. При этом, по соответствующим формулам находят число контрольных символов $n_к$ для кода d=3. Число контрольных символов для кода с d=4 будет на единицу больше, т.е.

$$n_{к(d=4)} = n_{к(d=3)} + 1 \text{ . (*)}$$

2. По таблице выбирается образующий многочлен, степень которого должна быть равна $n_к$. Причем, выбирают наиболее короткий многочлен с числом ненулевых членов не менее кодового расстояния , т.е. в нашем случае не менее трех ($n_{к(d=3)}$). Такой многочлен позволит создать циклический код, обнаруживающий две ошибки: $d=r+s+1$, причем $r \geq s$; $r=2$; $s=0$; d=3.

Образующий многочлен $P(X)_{(d=4)}$ получается как произведение двучлена (x+1) на многочлен $P(X)$ (d=3), т.е.

$$P(X)_{(d=4)} = P(X)_{(d=3)} (x+1). \quad (**)$$

Для обнаружения трех ошибок степень образующего многочлена должна быть на единицу больше, т.е. не менее четырех. Многочлен третьей степени, при числе ненулевых членов равным трем, позволяет обнаруживать все двойные ошибки. Многочлен первой степени x+1 обеспечивает обнаружение нечетных ошибок. Таким образом, если составить образующий многочлен четвертой степени, необходимый для синтеза циклического кода при d=4, как произведение двучлена x+1 на многочлен $P(X)_{(d=3)}$, то такой многочлен будет обладать корректирующими свойствами сомножителей, т.е. позволяет обнаруживать одну, две и три ошибки .

3. Дальнейшая процедура кодирования остается такой же, как и при синтезе кода с обнаружением двойной ошибки.

Пример. Пусть требуется закодировать циклическим кодом с d=4 сообщение

$$G(X)=x^{10} + x^8 + x^7 + x^6 + x^3 + x + 1 \rightarrow 10111001011.$$

Решение. Находим число контрольных символов. В нашей комбинации, подлежащей кодированию, одиннадцать информационных символов, т.е. $n_и$=11. Тогда для d=3:

$$n_к =]lb(n_и+1)+]lb(n_и+1)[[=]lb(12+4)[=4.$$

Таким образом:

$$n_{к(d=3)} = 4; \quad n_{к(d=4)} = n_{к(d=3)} + 1 = 4 + 1 = 5.$$

Выбираем образующий многочлен:

Для d=3.

Из таблицы многочленов выбираем многочлен четвертой степени $n_{к(d=3)}$=4 с число ненулевых членов, равным трем (d=3): $P(X^4)_{(d=3)}=x^4+x+1$.

Для d=4 .

$P(X)_{(d=4)}=P(X)_{(d=3)}(x+1)=(x^4+x+1)(x+1)=$

$=x^5+x^4+x^2+1 \rightarrow 110101.$

Выбираем одночлен $X^{n_к} = X^5$.

Составим произведение $G(X)\,X^{n_к}$:

$$G(X)\,X^{n_к} = x^5(x^{10} + x^8 + x^7 + x^6 + x^3 + x + 1) =$$

$$= x^{15} + x^{13} + x^{12} + x^{11} + x^8 + x^6 + x^5 \rightarrow 1011100101100000.$$

Определяем значение пяти контрольных символов, которые должны занять место пяти нулей в конце полученного выше произведения. Для этого разделим полученное выражение на P(X) $_{(d=4)}$ до получения остатка

```
1011100101100000  110101
110101            11111
 110110
 110101
      111011
      110101
        111000
        110101
          110100
          110101
                10
```

Таким образом, остаток равен 00010, следовательно кодовая комбинация циклического кода

F(X)→ 10111001011 00010.
информ-е контр-е
символы

Проверяем полученную кодовую комбинацию. Она должна без остатка разделиться на образующий многочлен:

```
1011100101100010  110101
110101            11111
 110110
 110101
      111011
      110101
        111000
        110101
          110101
          110101
          000000
```

Лекция 11

Декодирование циклических кодов

1. Обнаружение ошибки. Принцип обнаружения ошибок основан на том, что безошибочно принятая кодовая комбинация F(X) должна разделиться на образующий

многочлен P(X) без остатка. После деления можно отбросить контрольные символы(их число и местоположение в кодовой комбинации известно) и восстановить сообщение.

Если при делении принятой комбинации на образующий многочлен получится остаток, то это свидетельствует о наличии ошибки, т.е. вместо переданной комбинации F(X) мы принимаем некоторую другую комбинацию H(X), которую можно представить в виде суммы двух многочленов

$$H(X) = F(X) + E(X), \qquad (1)$$

где E(X) - многочлен ошибок.

Если, например, была передана комбинация F(X) $\rightarrow$ 1010001, закодированная с помощью образующего многочлена P(X) ® 1101, то при без ошибочном приеме она без остатка разделится на P(X).

Если же в результате деления появился остаток, равный, например, 100 или 001, или 010, то это свидетельствует о том, что принятая комбинация ошибочна (содержит одну ошибку). Итак, остаток от деления свидетельствует о наличии ошибки, но не указывает номер ошибочной позиции.

2. Обнаружение и исправление ошибок. Последовательность действий при исправлении ошибок следующая:

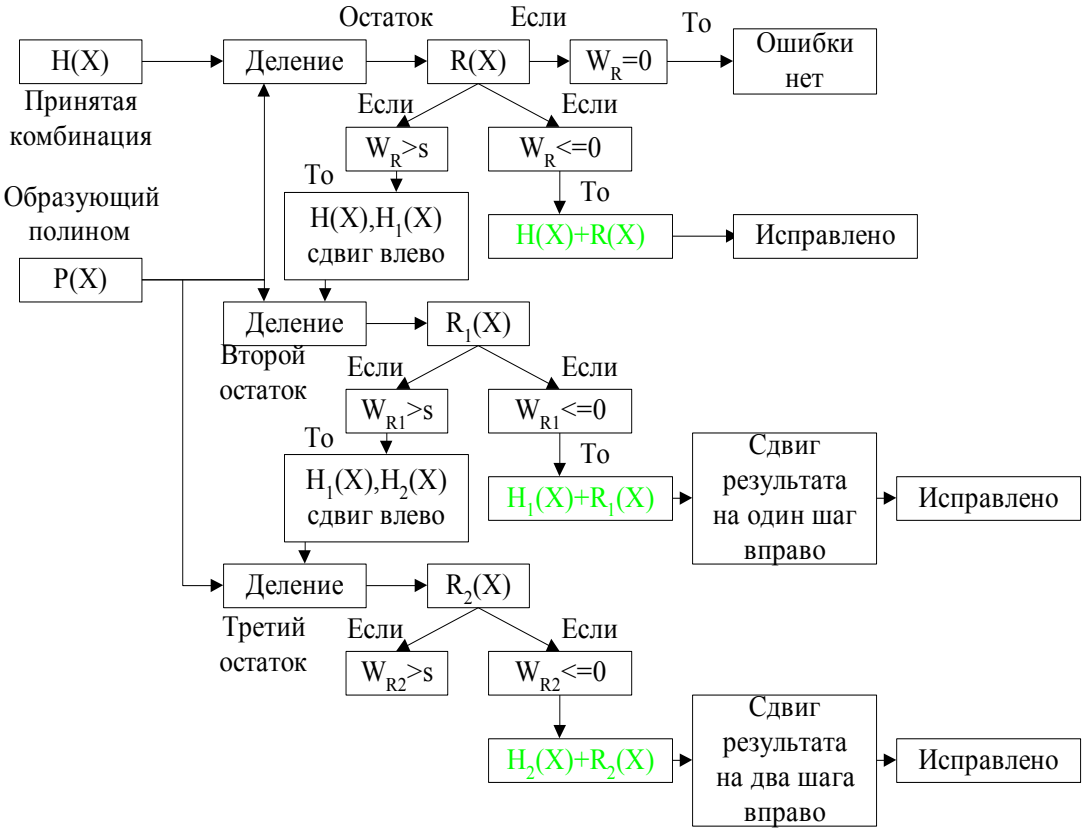
1) вычисляется остаток от деления принятой комбинации F(X) на образующий многочлен P(X). Если остаток равен 0, то комбинация не содержат ошибки. Если остаток не равен 0, то в принятой комбинации имеется ошибка;
2) при неравенстве 0 остатка подсчитывается его "вес" W, причем, если W ≤ s ("вес" остатка не более числа ошибок, исправляемых данным кодом), то принятую кодовую комбинацию надо сложить по модулю 2 с остатком. При этом получается исправленная комбинация.

Если W > s, то производится циклический сдвиг влево на один символ (на один разряд). Полученная после такого сдвига комбинация вновь делится на образующий многочлен и подсчитывается "вес" остатка. В данном случае возникают две возможности:

а) W ≤ s. При этом циклически сдвинутую комбинацию складывают с остатком и затем, после сложения, циклически сдвигают в обратную сторону (вправо) на один символ (т.е. возвращают на прежнее место). В результате получается исправленная комбинация;

б) W > s. При этом производятся дополнительные циклические сдвиги влево. После каждого циклического сдвига на один символ полученная комбинация делится на образующий многочлен и определяется "вес" остатка. Если W ≤ s, то полученную от деления комбинацию складывают с остатком, но циклическую перестановку обратно вправо осуществляют не один раз, а столько, сколько было сделано сдвигов влево. В результате получается исправленная комбинация;

Блок-схема декодирования циклического кода представлена на рисунке.



Таким образом, в отличии от кода Хемминга здесь отсутствует в явном виде синдром ошибки, указывающий номер искаженной позиции.

Примеры

Рассмотрим несколько примеров.

Пример 1. Принят код 1101110, закодированный циклическим кодом с образующим многочленом P(X) → 1011. Код позволяет исправлять одну ошибку, т.е. s=1. Проверить, имеется ли в принятой комбинации ошибка и в случае обнаружения показать процесс исправления.

Решение. 1. Делим принятую комбинацию 1101110 на образующий многочлен P(X) → 1011:

```
1101110   1011
1011
 1101
1011
 1101
 1011
  1100
  1011
   111
```

и находим, что остаток $R(X)=x^2+x+1 \rightarrow 111$. Принятая комбинация ошибочна.

2. Подсчитываем "вес" остатка W=3, что не удовлетворяет равенству W=s.

3. Сдвигаем ошибочную комбинацию 1101110 циклически на один символ влево. Получаем после сдвига 1011101.

4. Делим циклически сдвинутую комбинацию на образующий многочлен

1011101 1011

1011

101

и находим, что остаток $R(X)=x^2+1 \rightarrow 101$.

5. Подсчитываем "вес" второго остатка $W_2=2$, т.е. W_2 тоже больше s.

6. В соответствии с алгоритмом декодирования (см. рис.) осуществляем еще один циклический сдвиг сдвиг влево. Получаем после сдвига 0111011.

7. Делим сдвинутую комбинацию на образующий многочлен

0111011 1011

1011

1011

1011

1

Получаем остаток $R_2(X)=1 \rightarrow 001$, "вес" которого W=1, что удовлетворяет равенству W=s.

8. Комбинацию, полученную после второго сдвига 0111011, складываем по модулю два с остатком $R_2(X)$:

0111011

$\oplus$ 001

0111010

$\uparrow\leftarrow\leftarrow\leftarrow\downarrow$

9. Полученную в результате комбинацию сдвигаем циклически два раза вправо. После первого сдвига получаем

0011101

$\uparrow\leftarrow\leftarrow\leftarrow\downarrow$.

После второго сдвига - 1001110, что представляет исправленную кодовую комбинацию.

10. Проверим 1001110

1001110 1011

1011

1011

1011

0000 .

Остаток R(X)=0. Таким образом, ошибка исправлена.

Пример 2. Исходная комбинация 0101111000. Принятая комбинация 0001011001, т.е. произошел тройной сбой. Показать, что циклический код, образованный многочленом 101111, позволяет обнаружить трехкратную ошибку.

Решение. 1. Поскольку число ненулевых членов образующего многочлена в рассматриваемом примере равно 5, то, как было сказано выше, кодовое расстояние циклического кода $d_0 \geq 5$. Таким образом, данный код позволяет обнаружить

трехкратную ошибку (даже четырехкратную, так как $d_1 = r+s+1$. Если s=0, то $r = d_0 - 1 = 5 -1 = 4$).

2. Делим принятую комбинацию на образующий многочлен

0001011001 101111

101111

111

Имеет место остаток R (X) $\rightarrow$ 111, что свидетельствует о наличии ошибки.

Поскольку $n_к = 5$ ($n_к$ равно степени образующего многочлена), а n=10, то, следовательно, избыточность кода

$$l = (n - n_u)/n = 1 - n_u/n = 1 - 5/10 = 0{,}5.$$