ЛАБОРАТОРНАЯ РАБОТА №5

ПОСТРОЕНИЕ И РЕАЛИЗАЦИЯ ЦИКЛИЧЕСКИХ КОДОВ

Целью работы является усвоение методов построения и технической реализации кодирующих и декодирующих устройств циклических кодов.

Указания к построению кодов.

Математической основой циклических кодов является теория колец. В качестве основных операций в циклическом коде используются операции сложения по модулю два и символического умножения, в котором для сохранения степени многочлена не выше $(n\text{-}1)$ используется искусственный прием. Если степень многочлена после умножения не выше $n\text{-}1$, то он и принимается за результат умножения, а если выше, то он делится на другой многочлен (x^n+1) и в качестве результата умножения принимается остаток от деления.

Выбор образующего многочлена

Любой групповой (n,k) код может быть записан в виде матрицы, включающей k линейно-независимых строк по n символов. Среди всего многообразия таких кодов можно выделить коды, у которых строки образующих матриц связаны дополнительным условием цикличности.

Все строки образующей матрицы такого кода могут быть получены циклическим сдвигом одной комбинации: называемой образующей для данного кода. Коды, удовлетворяющие этому условию, получили название циклических кодов.

Сдвиг осуществляется справа налево, причем крайний левый символ каждый раз переносится в конец комбинации. Запишем, например, образующую матрицу кода, получающуюся циклическим сдвигом комбинации 001011:

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

При описании циклических кодов n-разрядные кодовые комбинации представляются в виде многочленов фиктивной переменной x. Показатели степени у x соответствуют номерам разрядов (начиная с нулевого), а коэффициентами при x являются цифры 0 и 1 (поскольку мы рассматриваем двоичные коды).

Запишем, например, в виде многочлена образующую кодовую комбинацию 001011

$$g_0(x)=0\cdot x^5+0\cdot x^4+1\cdot x^3+0\cdot x^2+1\cdot x^1+1\cdot x^0=x^3+x+1$$

Действия над кодовыми комбинациями теперь сводятся к действию над многочленами.

Вышеупомянутый циклический сдвиг образующего многочлена без переноса единицы в конец кодовой комбинации соответствует простому умножению на x. Например, вторая строка матрицы есть

$$g_0(x)x=x^4+x^2+x \ (010110).$$

Циклический сдвиг строки матрицы с единицей в старшем разряде (слева) равносилен умножению соответствующего многочлена на x с одновременным вычитанием из результата многочлена x^n+1

Поскольку операции вычитания и сложения по модулю два тождественны, многочлен, соответствующий любой строке матрицы, может быть записан в виде

$$g_i(x)=g_0(x)x^i+C(x^n+1),$$

где C равно 1, если степень $g_0(x)x^i$ превышает $n-1$, и нулю, если не превышает.

Если выбрать за образующий такой многочлен $g_0(x)$, который является делителем двучлена x^n+1, то все многочлены матрицы, а поэтому и все многочлены кода (разрешенные кодовые комбинации), будут делиться на $g_0(x)$ без остатка. Ни один многочлен, соответствующий запрещенной кодовой комбинации, на образующий многочлен нацело не делится. Это свойство позволяет обнаруживать ошибки. По виду остатка можно определить и вектор ошибки, а, следовательно, и устранить ее.

Любая принятая кодовая комбинация $h(x)$, содержащая ошибку, может быть представлена в виде суммы по модулю два неискаженной комбинации кода $f(x)$, делящейся на $g_0(x)$ без остатка, и вектора ошибки $e(x)$. Для однозначности декодирования каждому вектору ошибки должен соответствовать отличный от других остаток - опознаватель. Чем больше различных остатков может быть образовано при делении $h(x)$ на $g_0(x)$, тем больше разновидностей ошибок способен исправить данный циклический код.

Наибольшее число остатков, равное 2^m-1 (исключая нулевой), может обеспечить только неприводимый (простой) многочлен $g_0(x)$ степени m , принадлежащий показателю степени n, если m и n связаны соотношением $n=2^m-1$.

В литературе по кодированию доказано, что для любого m существует по крайней мере один неприводимый многочлен степени m, принадлежащий показателю степени n, если $n=2^m-1$.

При известном числе информационных символов $(к)$ задача, следовательно, сводится к тому, чтобы определить наименьшую степень m образующего многочлена, обеспечивающего обнаружение или исправление ошибок заданной кратности. Зная n и m по имеющимся в ряде книг таблицам многочленов, неприводимых при двоичных коэффициентах, можно выбрать и конкретный образующий многочлен.

Для исправления одиночных ошибок требуемая минимальная степень образующего многочлена *(m)* находится из соотношения

$$2^m\text{-}1=2^{n\text{-}k}\text{-}1\geq C^1{}_n$$

Выберем, например образующий многочлен для случая $к$=4. Тогда n=7 и m=3.

В таблице неприводимых многочленов, принадлежащих степени n, находим два многочлена третей степени, так как $x^7+1=(x+1)(x^3+x+1)(x^3+x^2+1)$.

Примем за образующий многочлен $g(x)=x^3+x^2+1$ (1101). Чтобы убедится, что каждому вектору ошибки соответствует отличный от других остаток, поделим каждый из этих векторов на $g_0(x)$.

Векторы ошибок m младших разрядов имеют вид:

0	0	1
0	1	0
1	0	0

Степени соответствующих им многочленов меньше степени образующего многочлена $g_0(x)$. Поэтому они сами являются остатками при нулевой целой части. Остаток, соответствующий ошибке в следующем разряде, получается при делении 1000 на 1101, т.е.

```
1000 |1101
1101
 101
```

Аналогично могут быть найдены и остальные остатки. Однако их можно получить проще, деля на $g_0(x)$ комбинацию в виде единицы с рядом нулей и выписывая все промежуточные остатки

```
1000000000 |1101          Остатки
1101
 1010  ----------------> 101
 1101                    111
  1110                   011
  1101                   110
   01100                 001
    1101                 010
     001000              100
```

При последующем делении остатки повторяются.

Если выбрать в качестве образующего многочлена $g_0(x)=x^3+x+1$, то тоже получим требуемое число различных остатков – 7.

Если k=5 и требуется исправлять тоже одиночную ошибку, то $2^{n-5}-1\geq n$. Откуда получаем n_{min}=9 и m= n – k =9–5=4. Примем $g_0(x)=x^4+x+1$. Этот образующий многочлен сохранится до k=11, так как неравенство $2^{n-11}-1\geq n$ будет выполняться при n_{min}=15 и m=4.

Метод и средства кодирования

Применительно к циклическим кодам принято отводить под информационные символы k старших разрядов многочлена кода, а под проверочные символы n-k низших разрядов.

Применяется следующая процедура кодирования: многочлен $a(x)$, соответствующий k-разрядной комбинации информационных разрядов кода, умножается на x^m, где m - степень образующего многочлена. Это соответствует добавлению к комбинации $a(x)$ m нулей со стороны младших разрядов. Произведение $a(x)x^m$ делится на образующий многочлен $g_0(x)$. В общем случае при этом получаем некоторое частное $q(x)$ и остаток $r(x)$: $a(x)x^m=q(x)g_0(x) \oplus r(x)$. Остаток прибавляется к $a(x)x^m$. Поскольку степень остатка $r(x)$ не превышает m-1, а в комбинации, соответствующей многочлену $a(x)x^m$, m младших разрядов-нулевые, то указанная операция сложения равносильна приписыванию $r(x)$ к $a(x)$ со стороны младших разрядов. Полученный многочлен $f(x)=a(x)x^m \oplus r(x)=q(x)g_0(x)$ делится на $g(x)$ без остатка и, следовательно, соответствует разрешенной комбинации кода.

Техническая реализация описанного процесса кодирования в случае двоичных кодов осуществляется посредством регистра сдвига с обратными связями, состоящего из ячеек памяти и сумматоров по модулю два. Сдвиг информации в регистре осуществляется импульсами, поступающими с генератора продвигающих импульсов, который на схеме, как правило, не указывается. На вход регистра поступают только коэффициенты многочленов, причем, начиная с коэффициента при переменной в старшей степени. На рис. 1 представлена схема, выполняющая деление произвольного многочлена (например, $a(x)=a_{n-1}x^{n-1}+,...,+a_1(x)+a_0)$ на некоторый фиксированный (например, образующий) многочлен $g(x)=g^{n-k}x^{n-к}+...+g_1(x)+g_0$.

Обратные связи регистра соответствуют виду многочлена $g_0(x)$. Количество включаемых в него сумматоров равно числу отличных от нуля коэффициентов $g_0(x)$, уменьшенному на единицу. Это объясняется тем, что сумматор сложения коэффициентов старших разрядов многочленов делимого и делителя в регистр не включается, так как результат сложения заранее известен (он равен нулю).

За первые m тактов коэффициенты многочлена-делимого заполняют регистр, причем коэффициент при x в старшей степени достигает крайней правой ячейки. На следующем такте единица делимого, выходящая из крайней ячейки регистра по цепи обратной связи, подается к сумматорам по модулю два, что равносильно вычитанию многочлена-делителя из многочлена-делимого. Если в результате предыдущей операции коэффициент при старшей степени x у остатка оказался равным нулю, то на следующем такте вычитания делителя не происходит. Коэффициенты делимого только сдвигаются вперед по регистру на один разряд, что находится в полном соответствии с тем, как это делается при делении многочленов столбиком.

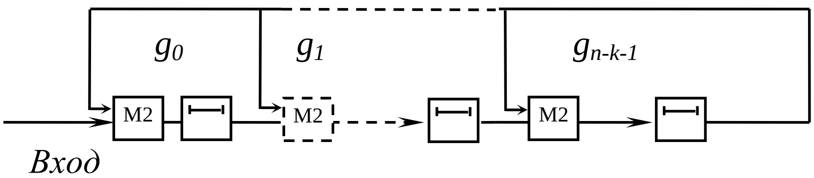

Рис. 1 Схема деления на произвольный многочлен.

Деление заканчивается с приходом последнего символа многочлена-делимого. При этом разность будет иметь более низкую степень чем делитель. Эта разность и есть остаток.

Рассмотренная схема деления многочленов может использоваться при декодировании. При кодировании она не применяется в силу того, что между информационными и проверочными символами образуется разрыв в *m* тактов.

Для кодирования используется схема, позволяющая разделить многочлен типа $a(x)x^m$ за $к$ тактов. Она отличается от рассмотренной тем, что коэффициенты кодируемого многочлена участвуют в обратной связи не через *m* сдвигов, а сразу с первого такта.

Для случая $g_0(x)=x^3+x^2+1$ и $a(x)=a^3+1$ схема кодирующего устройства приведена на рис. 2.

В исходном состоянии ключ K_1 находится в положении 1, а ключ K_2 замкнут. Информационные символы одновременно поступают как в линию связи, так и в регистр сдвига, где за $к$ тактов образуется остаток. Затем ключ K_2 размыкается, ключ K_1 переходит в положение 2 и остаток выводится в канал связи.

Процесс формирования кодовой комбинации шаг за шагом представлен в табл. 1, где черточками отмечены освобождающиеся ячейки, занимаемые новыми информационными символами.

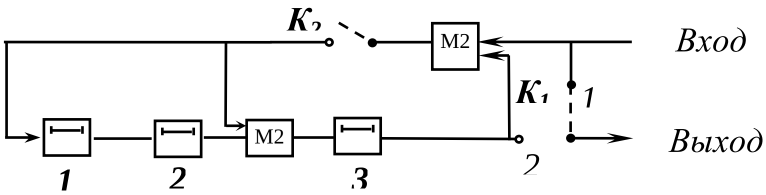

Рис. 2 Схема кодирующего устройства.

Таблица 1

N такта	Вход	Состояние ячеек регистров			Выход
		1	2	3	
1	1	1	0	1	1
2	0	1	1	1	01
3	0	1	1	0	001
4	1	1	1	0	1001
5	0	-	1	1	01001
6	0	-	-	1	101001
7	0	-	-	-	1101001

Описание лабораторной работы

Представленные схемы деления позволяют закодировать информационные разряды кодом (7,4) с образующим многочленом $g(x)= x^3+x+1$, рассчитанным на исправление одиночных ошибок, и кодом (7,3) с образующим многочленом $g(x)= (x+1)(x^3+x^2+1) = x^4+x^2+x+1$, рассчитанным на исправление одиночных и двойных смежных ошибок. Заданные значения информационных разрядов записываются в регистр числа. Влияние помехи в канале связи имитируется сложением кодовой комбинации с вектором ошибки, формируемым в регистре помехи.

Задание

1. Изучить теоретическую часть работы, а также описание и порядок проведения лабораторной работы.

2. Составить таблицу, отражающую процесс кодирования четырехзначной комбинации информационных символов кодом (7,4).

Требования к отчету

Отчет должен включать:

1. Исходные последовательности для кодирования, вектора ошибок.
2. Таблицы, отражающие процесс кодирования для кодов (7,4).