

ИТМО

А.Ю. Краснов, С.А. Чепинский

ПРАКТИКУМ ПО ПРИКЛАДНОЙ ТЕОРИИ ИНФОРМАЦИИ



Санкт-Петербург
2023

МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РОССИЙСКОЙ
ФЕДЕРАЦИИ

УНИВЕРСИТЕТ ИТМО

А.Ю. Краснов, С.А. Чепинский
ПРАКТИКУМ ПО ПРИКЛАДНОЙ ТЕОРИИ
ИНФОРМАЦИИ

УЧЕБНО-МЕТОДИЧЕСКОЕ ПОСОБИЕ

РЕКОМЕНДОВАНО К ИСПОЛЬЗОВАНИЮ В УНИВЕРСИТЕТЕ ИТМО
по направлению подготовки 15.03.06, 27.03.04
в качестве Учебно-методического пособия для реализации основных
профессиональных образовательных программ высшего образования
бакалавриата

ИТМО

Санкт-Петербург
2023

Краснов А.Ю., Чепинский С.А., Практикум по прикладной теории информации– СПб: Университет ИТМО, 2023. – 72 с.

Рецензент(ы):

Власов Сергей Михайлович, кандидат технических наук, доцент (квалификационная категория "ординарный доцент") факультета систем управления и робототехники, Университета ИТМО.

Практикум по курсу "Прикладная теория информации" содержит методические указания к выполнению практических заданий, предусмотренных учебным планом специальностей 15.03.06 и 27.03.04 при изучении дисциплины "Прикладная теория информации". Задания практикума выполняются с целью закрепления теоретических разделов дисциплины и приобретения навыков построения и исследования функциональных узлов двоичных динамических систем, методов кодирования информации, а также передачи непрерывных сигналов в дискретных (двоичных) средах. В пособии представлен цикл из 8 работ, сориентированных на практическом закреплении знаний, входящих в дисциплину "Прикладная теория информации".

The logo of ITMO University, consisting of the letters 'ITMO' in a bold, black, sans-serif font. The 'I' and 'T' are connected, and the 'O' is a solid circle.

Университет ИТМО – национальный исследовательский университет, ведущий вуз России в области информационных, фотонных и биохимических технологий. Альма-матер победителей международных соревнований по программированию – ICPC (единственный в мире семикратный чемпион), Google Code Jam, Facebook Hacker Cup, Яндекс.Алгоритм, Russian Code Cup, Topcoder Open и др. Приоритетные направления: IT, фотоника, робототехника, квантовые коммуникации, трансляционная медицина, Life Sciences, Art&Science, Science Communication. Входит в ТОП-100 по направлению «Автоматизация и управление» Шанхайского предметного рейтинга (ARWU) и занимает 74 место в мире в британском предметном рейтинге QS по компьютерным наукам (Computer Science and Information Systems). С 2013 по 2020 гг. – лидер Проекта 5–100.

© Университет ИТМО, 2023

© Краснов А.Ю., Чепинский С.А., 2023

Содержание

1	Предисловие	4
2	Алгоритмы эффективного кодирования	5
3	Моделирование двоичных каналов с помощью конечных цепей Маркова	14
4	Помехозащитное кодирование	20
5	Формирование ПЗК на основе действий с модулярными многочленами	30
6	Преобразование кодов с помощью линейных двоичных динамических систем (ЛДДС)	39
7	ЛДДС в задачах рекуррентного помехозащитного кодирования и декодирования	48
8	Синтез динамических наблюдающих устройств на базе концепции подобия	57
9	Синтез ДДС в логике произвольных линейных триггеров	64

1 Предисловие

В учебном пособии изложен цикл практических работ по дисциплине "Прикладная теория информации" с приложениями к синтезу двоичных систем автоматического управления и телерегулирования. Всего представлено 8 работ, охватывающих следующие разделы дисциплины: алгоритмы эффективного кодирования, алгоритмы помехозащитного кодирования, проектирование кодирующих и декодирующих устройств и линейные двоичные динамические системы дискретной автоматики. В каждой работе приведены сжатые теоретические сведения, необходимые для успешного решения поставленной в работе задачи, наглядный пример выполнения задания, порядок выполнения работы и список вариантов.

Практикум рекомендован к использованию в качестве пособия к лабораторным или практическим работам при реализации таких дисциплин как "Прикладная теория информации", "Кодирование", "Управление двоичными динамическими системами" и других дисциплин, включающих в себя разделы, связанные с эффективным и помехозащитным кодированием информации, выработкой и преобразованием кодовых последовательностей, а также с анализом и синтезом линейных двоичных динамических систем дискретной автоматики.

Учебное пособие рассчитано на студентов направлений Робототехника, Интеллектуальная робототехника и Управление в технических системах, однако оно может быть рекомендовано также аспирантам и молодым специалистам, которым по роду своей образовательной и практической деятельности приходится иметь дело с проблемами канализации информации, обеспечения её помехозащиты при передаче и хранении, а также разработки устройств дискретной автоматики и устройств преобразования кодов.

2 Алгоритмы эффективного кодирования

Основной задачей эффективного кодирования является формирование таких кодов, которые обеспечивали бы наибольшую эффективность использования имеющихся ресурсов канальной среды. Одним из способов решения такой задачи является сокращение среднего числа разрядов кода на символ. Рассматриваемые далее методы эффективного кодирования базируются на основной теореме К. Шеннона [1].

Пусть источник дискретной информации (ИДИ) генерирует алфавит

$$X = \{x_i : p(x_i); i = \overline{1, n}\},$$

составленный из n дискретных статистически независимых символов, характеризующийся энтропией, рассчитываемой по формуле

$$H(X) = - \sum_{i=1}^n p(x_i) \log_2 p(x_i), \quad (1)$$

тогда:

- существует такой способ кодирования символов, при котором среднее на символ число двоичных разрядов кода

$$l_{\text{cp}} \{K(X)\} = \sum_{i=1}^n p(x_i) l \{K(x_i)\} \quad (2)$$

будет сколь угодно близким к энтропии $H(X)$ ИДИ;

- не существует такого способа кодирования, при котором среднее на символ число двоичных разрядов кода (2) будет меньше энтропии $H(X)$ ИДИ.

Стоит отметить, что в случае, когда вероятности $p(x_i)$ символов представляют собой отрицательные степени числа два, средняя на символ длина кода будет всегда совпадать с энтропией источника.

В качестве характеристик эффективного кода принято использовать две величины: избыточность и эффективность.

Избыточностью эффективного кода называется показатель D_k , определяемый выражением

$$D_k = \frac{l_{\text{cp}} \{K(X)\} - l_{\min} \{K(X)\}}{l_{\text{cp}} \{K(X)\}} = \frac{\sum_{i=1}^n p(x_i) l \{K(x_i)\} - H(X)}{\sum_{i=1}^n p(x_i) l \{K(x_i)\}}. \quad (3)$$

Эффективностью эффективного кода называется показатель η , определяемый выражением

$$\eta = \frac{l_{\min} \{K(X)\}}{l_{\text{cp}} \{K(X)\}} = \frac{H(X)}{\sum_{i=1}^n p(x_i) l \{K(x_i)\}}. \quad (4)$$

Приведенные выше характеристики эффективного кода связаны соотношением

$$D_k = 1 - \eta.$$

Критерием корректного формирования эффективного кода служит степень близости среднего на символ числа двоичных разрядов кода $l_{\text{cp}} \{K(X)\}$ к энтропии $H(X)$ ИДИ, выражаемая через неравенство

$$\frac{l_{\text{cp}} \{K(X)\} - H(X)}{H(X)} \leq \delta_l = 0.05 \div 0.1. \quad (5)$$

Далее рассмотрим два наиболее общепотребительных алгоритма эффективного кодирования:

- Алгоритм К.Шеннона—Р.Фэнно,
- Алгоритм Д. Хаффмэна.

Оба алгоритма опираются на ключевую идею о том, что чем больше вероятность появления символа на выходе ИДИ, тем короче должна быть соответствующая ему кодовая комбинация эффективного кода. Этот факт позволяет сказать, что эффективные коды относятся к классу неравномерных кодов.

Алгоритм эффективного кодирования Шеннона—Фэнно

Алгоритм эффективного кодирования К.Шеннона-Р.Фэнно (АШФ) является полностью табличным и основывается на группировании символов алфавита, генерируемого ИДИ, согласно вероятностям их появления на выходе ИДИ.

Проиллюстрируем работу АШФ следующим примером. Рассмотрим задачу эффективного кодирования символов ИДИ, генерирующего алфавит

$$X = \{x_1, x_2 : p(x_1) = 0.9, p(x_2) = 0.1\}$$

так, чтобы выполнялось условие (5). Вычислим в соответствии с (1) значение энтропии заданного ИДИ:

$$H(X) = - \sum_{i=1}^2 p(x_i) \log_2 p(x_i) = 0.9 \cdot 0.137 + 0.1 \cdot 0.322 = 0.47 \text{ (бит / символ)}.$$

Таблица 1: Таблица эффективного кодирования ИДИ

№ П/П	x_i	$p(x_i)$	$K(x_i)$	$l\{K(x_i)\}$
			1	
1	x_1	0.9	1	1
2	x_2	0.1	0	1

Далее составим таблицу формирования эффективного кода символов ИДИ. В таблицу необходимо вписать символы кодируемого алфавита в порядке уменьшения вероятности их появления на выходе ИДИ. После этого осуществляется первое деление символов на группы, характеризующиеся как можно более близкими суммарными вероятностями появления символов группы на выходе ИДИ. Старшим разрядам кодовых комбинаций символов из первой группы присваиваются значения 1, а символов из второй подгруппы – 0. Далее процедура повторяется для обеих групп до тех пор, пока не будет сформирован эффективный код для каждого символа кодируемого алфавита. Наконец, в крайний правый столбец таблицы вносятся значения длин сформированных эффективных кодов. Результат применения АШФ для кодирования символов заданного ИДИ приводит к таблице 1.

Далее необходимо вычислить среднее на символ число двоичных разрядов полученного эффективного кода с помощью соотношения (2):

$$l_{\text{ср}}\{K(X)\} = \sum_{i=1}^2 p(x_i) l\{K(x_i)\} = 0.9 \cdot 1 + 0.1 \cdot 1 = 1 \text{ (бит / символ)}.$$

Оценим степень близости средней длины эффективного кода на символ $l_{\text{ср}}\{K(X)\}$ к энтропии ИДИ $H(X)$, осуществив проверку выполнения неравенства (5):

$$\frac{l_{\text{ср}}\{K(X)\} - H(X)}{H(X)} = \frac{1 - 0.47}{0.47} \simeq 1.128 \gg \delta_l = 0.05 \div 0.1.$$

Так как неравенство (5) не выполнено, модифицируем исходный ИДИ путем введения мультипликативно агрегированных символов

$$\tilde{x}_k = x_i x_j, \quad (i, j = \overline{1, n}; k = \overline{1, n^a}),$$

где $a = 2$ – количество исходных символов кодируемого алфавита в каждом агрегированном символе, при этом вероятность появления агрегированного символа на выходе модифицированного ИДИ вычисляется как $p(\tilde{x}_k) = p(x_i) p(x_j)$.

Осуществим процедуру эффективного кодирования символов модифицированного ИДИ, в результате чего получим таблицу 2.

Таблица 2: Таблица эффективного кодирования модифицированного ИДИ

№ П/П	\tilde{x}_k	$p(\tilde{x}_k)$	$K(\tilde{x}_k)$			$l\{K(\tilde{x}_k)\}$
			1	2	3	
1	$\tilde{x}_1 = x_1x_1$	0.81	1			1
2	$\tilde{x}_2 = x_1x_2$	0.09	0	1		2
3	$\tilde{x}_3 = x_2x_1$	0.09	0	0	1	3
4	$\tilde{x}_4 = x_2x_2$	0.01	0	0	0	3

Вычислим среднее на блок символов число двоичных разрядов полученного эффективного кода с помощью соотношения (2):

$$l_{\text{срб}} \{K(\tilde{X})\} = \sum_{k=1}^4 p(\tilde{x}_k) l\{K(\tilde{x}_k)\} = 1.29 \text{ (бит / блок)},$$

и среднее на символ число двоичных разрядов полученного эффективного кода

$$l_{\text{ср}} \{K(X)\} = \frac{l_{\text{срб}} \{K(\tilde{X})\}}{a} = \frac{1.29}{2} = 0.645 \text{ (бит / символ)}.$$

Оценим степень близости средней длины эффективного кода на символ $l_{\text{ср}} \{K(X)\}$ к энтропии исходного ИДИ $H(X)$, осуществив проверку выполнения неравенства (5):

$$\frac{l_{\text{ср}} \{K(X)\} - H(X)}{l_{\text{ср}} \{K(X)\}} = \frac{0.645 - 0.47}{0.47} \simeq 0.37 > \delta_l = 0.05 \div 0.1.$$

Так как неравенство (5) не выполнено, продолжим модификацию исходного ИДИ, введя новые агрегированные символы $\tilde{\tilde{x}}_m = x_i x_j x_k$, ($i, j, k = \overline{1, n}; m = \overline{1, n^a}$), где $a = 3$ - количество исходных символов кодируемого алфавита в каждом агрегированном символе, при этом вероятность появления агрегированного символа на выходе модифицированного ИДИ вычисляется как $p(\tilde{x}_k) = p(x_i) p(x_j) p(x_k)$, в результате чего получим таблицу 3.

Вычислим среднее на блок символов число двоичных разрядов полученного эффективного кода с помощью соотношения (2):

$$l_{\text{срб}} \{K(\tilde{\tilde{X}})\} = \sum_{m=1}^8 p(\tilde{\tilde{x}}_m) l\{K(\tilde{\tilde{x}}_m)\} = 1.59 \text{ (бит / блок)},$$

и среднее на символ число двоичных разрядов полученного эффективного кода

$$l_{\text{ср}} \{K(X)\} = \frac{l_{\text{срб}} \{K(\tilde{\tilde{X}})\}}{a} = \frac{1.59}{3} = 0.53 \text{ (бит / символ)}.$$

Таблица 3: Таблица эффективного кодирования модифицированного ИДИ

№ П/П	\tilde{x}_m	$p(\tilde{x}_m)$	$K(\tilde{x}_m)$					$l\{K(\tilde{x}_m)\}$
			1	2	3	4	5	
1	$\tilde{x}_1 = x_1x_1x_1$	0.729	1					1
2	$\tilde{x}_2 = x_1x_1x_2$	0.081	0	1	1			3
3	$\tilde{x}_3 = x_1x_2x_1$	0.081	0	1	0			3
4	$\tilde{x}_4 = x_2x_1x_1$	0.081	0	0	1			3
5	$\tilde{x}_5 = x_1x_2x_2$	0.009	0	0	0	1	1	5
6	$\tilde{x}_6 = x_2x_1x_2$	0.009	0	0	0	1	0	5
7	$\tilde{x}_7 = x_2x_2x_1$	0.009	0	0	0	0	1	5
8	$\tilde{x}_8 = x_2x_2x_2$	0.001	0	0	0	0	0	5

Оценим степень близости средней длины эффективного кода на символ $l_{\text{cp}}\{K(X)\}$ к энтропии исходного ИДИ $H(X)$, осуществив проверку выполнения неравенства (5):

$$\frac{l_{\text{cp}}\{K(X)\} - H(X)}{l_{\text{cp}}\{K(X)\}} = \frac{0.53 - 0.47}{0.47} \simeq 0.127 \approx \delta_l = 0.05 \div 0.1.$$

Так как неравенство (5) "почти" выполнено, можно сделать вывод, что полученный эффективный код обеспечивает приемлемое решение поставленной задачи (хотя для точного выполнения условий задачи нужно было бы осуществить еще одну итерацию агрегирования символов ИДИ).

Очевидно, что рассмотренный алгоритм является вариативным, так как деление символов на группы можно осуществлять различными способами, что повлияет на итоговое решение задачи эффективного кодирования.

Алгоритм эффективного кодирования Д. Хаффмэна

Алгоритм эффективного кодирования Д.Хаффмэна (АХ) является таблично-графическим и, в отличие от АШФ, предлагает однозначное решение задачи эффективного кодирования. Реализация алгоритма делится на три этапа. На первом этапе строится таблица аддитивно агрегированных символов. На втором этапе по полученной таблице строится граф эффективного кода. На третьем этапе с ветвей графов списываются эффективные коды символов кодируемого алфавита при перемещении по ветвям графа от исходной вершины до соответствующей конечной. В случае, если построенный эффективный код не удовлетворяет неравенству (5), осуществляется его мультипликативная модификация с целью перехода к построению эффективного кода блока.

Таблица 4: Таблица эффективного кодирования АХ

№ П/П	x_i	$p(x_i)$	Вероятности аддитивно модифицированного источника						
			1	2	3	4	5	6	7
1	x_1	0.22	0.22	0.22	0.26	0.32	0.42	0.58	1
2	x_2	0.2	0.2	0.2	0.22	0.26	0.32	0.42	
3	x_3	0.16	0.16	0.16	0.2	0.22	0.26		
4	x_4	0.16	0.16	0.16	0.16	0.2			
5	x_5	0.1	0.1	0.16	0.16				
6	x_6	0.1	0.1	0.1					
7	x_7	0.04	0.06						
8	x_8	0.02							

Рассмотрим АХ на примере решения задачи эффективного кодирования символов ИДИ, генерирующего алфавит $X = \{x_i; i = \overline{1, 8}\}$ с вероятностями появления символов на выходе ИДИ, равными $p(x_1) = 0.22$, $p(x_2) = 0.2$, $p(x_3) = 0.16$, $p(x_4) = 0.16$, $p(x_5) = 0.1$, $p(x_6) = 0.1$, $p(x_7) = 0.04$ и $p(x_8) = 0.02$.

Для начала, как и в АШФ, рассчитаем энтропию ИДИ:

$$H(X) = - \sum_{i=1}^8 p(x_i) \log_2 p(x_i) = 2.7545 \text{ (бит / символ)}.$$

Далее необходимо составить таблицу формирования эффективного кода. В таблицу необходимо вписать символы кодируемого алфавита в порядке уменьшения вероятности их появления на выходе ИДИ. Далее производится процедура последовательного аддитивного агрегирования символов. Она заключается в суммировании двух наименьших вероятностей появления символов. После этого в четвертый столбец таблицы записываются значения вероятностей в порядке убывания с учетом вероятности появления аддитивно агрегированного символа. Данная операция повторяется до тех пор, пока вероятность появления аддитивно агрегированного символа не станет равной 1. Результат описанной процедуры для рассматриваемого примера приведен в таблице 4.

Следующий этап АХ состоит в построении графа кода. В верхнем узле графа располагается финальный агрегированный символ (с единичной вероятностью появления). Далее развертывание графа происходит согласно таблице 4 эффективного кодирования от последнего столбца к третьему, каждый агрегированный символ служит промежуточным узлом графа, а каждый из исходных символов - конечным узлом. Граф эффективного кода, соответствующий таблице 4, приведен на рисунке 1.

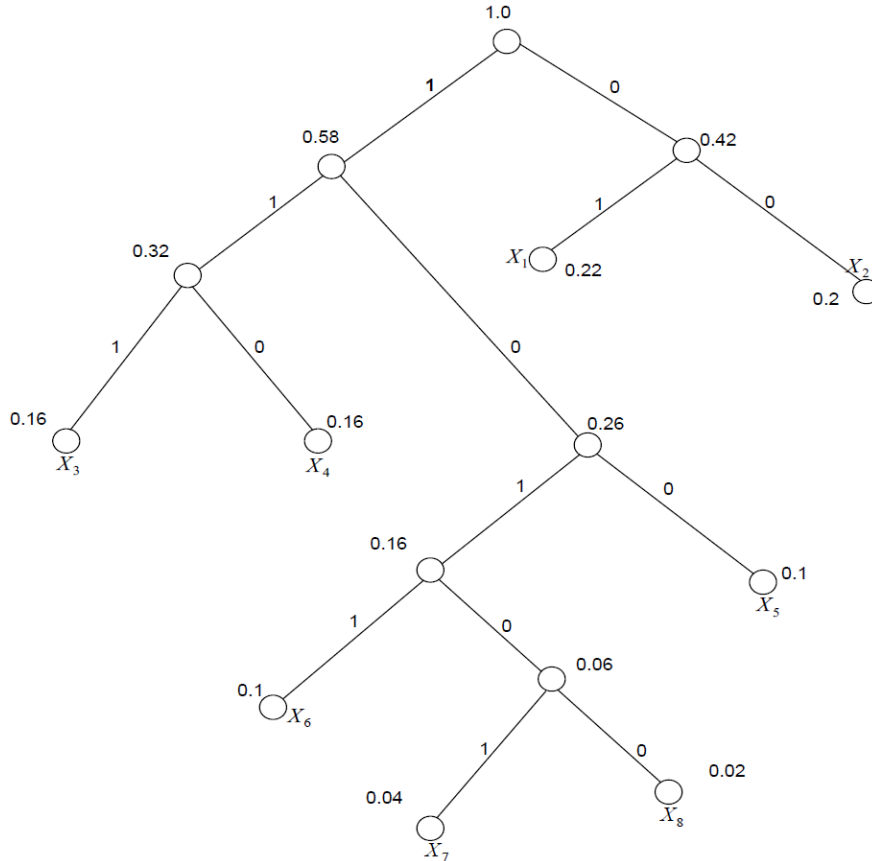


Рис. 1: Граф эффективного кода, построенный по таблице 4.

Далее остается составить эффективные коды символов, списав их с ветвей графа на рисунке 1. Полученные эффективные коды сведены в таблицу 5.

Далее необходимо вычислить среднее на символ число двоичных разрядов полученного эффективного кода с помощью соотношения (2):

$$l_{\text{ср}} \{K(X)\} = \sum_{i=1}^8 p(x_i) l \{K(x_i)\} = 2.8 \text{ (бит / символ)}.$$

Оценим степень близости средней длины эффективного кода на символ $l_{\text{ср}} \{K(X)\}$ к энтропии ИДИ $H(X)$, осуществив проверку выполнения неравенства (5):

$$\frac{l_{\text{ср}} \{K(X)\} - H(X)}{H(X)} = \frac{2.8 - 2.7545}{2.7545} \simeq 0.0165 < \delta_l = 0.05 \div 0.1.$$

Так как неравенство (5) выполнено, можно сделать вывод, что полученный эффективный код обеспечивает решение поставленной задачи эффективного кодирования.

Таблица 5: Эффективные коды, полученные согласно АХ

№ П/П	x_i	$K(X)$					$l_{\text{ср}} \{K(X)\}$
		1	2	3	4	5	
1	x_1	0	1				2
2	x_2	0	0				2
3	x_3	1	1	1			3
4	x_4	1	1	0			3
5	x_5	1	0	0			3
6	x_6	1	0	1	1		4
7	x_7	1	0	1	0	1	5
8	x_8	1	0	1	0	0	5

Порядок выполнения работы

1. Проранжировать элементы алфавита, который требуется закодировать согласно варианту задания, по вероятности их появления на выходе ИДИ;
2. Выполнить процедуру эффективного кодирования по алгоритму эффективного кодирования К. Шеннона - Р. Фэнно;
3. Выполнить процедуру эффективного кодирования по алгоритму эффективного кодирования Д. Хаффмэна;
4. Сравнить полученные в результате применения двух алгоритмов эффективного кодирования результаты;
5. Сформулировать выводы по работе.

Варианты задания

1. ИДИ генерирует алфавит $X = \{x_i; i = \overline{1,4}\}$ со следующими вероятностями появления символов: $p(x_1) = 0.9, p(x_2) = p(x_3) = p(x_4) = 1/30$;
2. ИДИ генерирует алфавит $X = \{x_i; i = \overline{1,4}\}$ со следующими вероятностями появления символов: $p(x_1) = 0.9, p(x_2) = 0.05, p(x_3) = p(x_4) = 0.025$;
3. ИДИ генерирует алфавит $X = \{x_i; i = \overline{1,4}\}$ со следующими вероятностями появления символов: $p(x_1) = 0.8, p(x_2) = p(x_3) = p(x_4) = 1/15$;
4. ИДИ генерирует алфавит $X = \{x_i; i = \overline{1,4}\}$ со следующими вероятностями появления символов: $p(x_1) = 0.8, p(x_2) = 0.1, p(x_3) = p(x_4) = 0.05$;

5. ИДИ генерирует алфавит $X = \{x_i; i = \overline{1,4}\}$ со следующими вероятностями появления символов: $p(x_1) = 0.8, p(x_2) = p(x_3) = 0.075, p(x_4) = 0.05$;
6. ИДИ генерирует алфавит $X = \{x_i; i = \overline{1,4}\}$ со следующими вероятностями появления символов: $p(x_1) = 0.7, p(x_2) = 0.25, p(x_3) = p(x_4) = 0.025$;
7. ИДИ генерирует алфавит $X = \{x_i; i = \overline{1,4}\}$ со следующими вероятностями появления символов: $p(x_1) = 0.75, p(x_2) = 0.2, p(x_3) = p(x_4) = 0.025$;
8. ИДИ генерирует алфавит $X = \{x_i; i = \overline{1,4}\}$ со следующими вероятностями появления символов: $p(x_1) = 0.65, p(x_2) = 0.3, p(x_3) = p(x_4) = 0.025$;
9. ИДИ генерирует алфавит $X = \{x_i; i = \overline{1,4}\}$ со следующими вероятностями появления символов: $p(x_1) = 0.8, p(x_2) = 0.1, p(x_3) = 0.075, p(x_4) = 0.025$;
10. ИДИ генерирует алфавит $X = \{x_i; i = \overline{1,4}\}$ со следующими вероятностями появления символов: $p(x_1) = 0.9, p(x_2) = 0.05, p(x_3) = 0.03, p(x_4) = 0.02$.
11. ИДИ генерирует алфавит $X = \{x_i; i = \overline{1,4}\}$ со следующими вероятностями появления символов: $p(x_1) = 0.1, p(x_2) = 0.3, p(x_3) = 0.09, p(x_4) = 0.51$.
12. ИДИ генерирует алфавит $X = \{x_i; i = \overline{1,4}\}$ со следующими вероятностями появления символов: $p(x_1) = 0.25, p(x_2) = 0.15, p(x_3) = 0.5, p(x_4) = 0.1$.
13. ИДИ генерирует алфавит $X = \{x_i; i = \overline{1,4}\}$ со следующими вероятностями появления символов: $p(x_1) = 0.2, p(x_2) = 0.4, p(x_3) = 0.1, p(x_4) = 0.3$.
14. ИДИ генерирует алфавит $X = \{x_i; i = \overline{1,4}\}$ со следующими вероятностями появления символов: $p(x_1) = 0.1, p(x_2) = 0.07, p(x_3) = 0.03, p(x_4) = 0.8$.
15. ИДИ генерирует алфавит $X = \{x_i; i = \overline{1,4}\}$ со следующими вероятностями появления символов: $p(x_1) = 0.05, p(x_2) = 0.6, p(x_3) = 0.15, p(x_4) = 0.2$.

3 Моделирование двоичных каналов с помощью конечных цепей Маркова

Рассматриваются симметричные, несимметричные и симметрированные двоичные каналы связи, характеризующиеся в зависимости от принятой модели вероятностями p_{01} и p_{10} превращения ноля в единицу и единицы в ноль соответственно. Если $p_{01} = p_{10}$, канал связи называют симметричным, если же это условие не выполняется, говорят, что канал связи несимметричный. В последнем случае используют симметрированную модель искажений, характеризующуюся вероятностью искажения бита $p = \max\{p_{01}, p_{10}\}$.

Возникает задача нахождения инструментальных и алгоритмических средств, с использованием которых могут быть получены приведенные выше вероятности трансформации бита в двоичном канале связи. Вопрос инструментальной части решается с помощью хорошо функционирующей системы статистического контроля состояния каналов связи в службах, задействованных в организации телекоммуникационного процесса. Что касается алгоритмических средств, то они должны быть согласованы с используемыми модельными представлениями процессов искажения информационных потоков в двоичных каналах при их передаче.

В настоящее время наиболее часто используемыми моделями двоичных каналов связи являются модели Гилберта, Эллиота–Гилберта, Смита–Боуэна–Джойса и Фричмана–Свободы. Первые три модели каналов связи используют для этих целей конечные цепи Маркова с двумя состояниями, последняя ориентирована на возможность использования конечных цепей Маркова с произвольным числом состояний.

Конечная цепь Маркова (КЦМ) есть макровектор, задаваемый в виде [1, 2]

$$\text{КЦМ} = \{S, P, T, \Pi, K\}, \quad (6)$$

где $S = \{S_i; i = \overline{1, n}\}$ - счетное множество состояний моделируемого процесса, $P = \text{row}\{P_i; i = \overline{1, n}\}$ - вектор-строка вероятностей пребывания в состояниях S_i , $T = \{\Delta t_i; i = \overline{1, n}\}$ - множество длительностей пребывания процесса в состояниях S_i , $\Pi = \text{col}\{[P_{ij}; j = \overline{1, n}]; i = \overline{1, n}\}$ - матрица вероятностей перехода моделируемого процесса из состояния S_i в состояние S_j , $K = \{k : 0 \leq k < \infty\}$ - счетное множество дискретных моментов времени, с помощью которых параметризуется моделируемый процесс так, что для вектора вероятностей $P(k+1)$ пребывания в состоянии перехода $S(k+1)$ при условии, что стохастический процесс находился с вероятностью $P(k)$ в исходном (текущем) состоянии $S(k)$, оказывается справедливым представление в виде рекуррентной процеду-

ры

$$P(k+1) = P(k)\Pi(k), P(0) = P\{S(0)\}, \quad (7)$$

где $P(0) = P\{S(0)\}$ - вектор вероятности пребывания в начальном состоянии.

Моделирование стохастического процесса с помощью поглощающей однородной дискретной конечной цепи Маркова (с не зависящей от номера шага матрицей переходных вероятностей и, как минимум, одним поглощающим состоянием) используется в задачах оценки временных затрат на стохастический переход из некоторого состояния $S_l(0)$, принимаемого за начальное, путем задания вектора распределения вероятностей пребывания стохастического процесса в состояниях $S(0)$ в форме

$$P(0) = [P_1(0) = 0 \quad P_2(0) = 0 \quad \dots \quad P_n(0) = 0]$$

и придания состоянию финального перехода $S_f(k_f)$ свойства поглощения (вероятность перехода из такого состояния в него же равна 1) с целью остановки стохастического процесса в состоянии S_f и оценки значения $k = k_f$ длительности этого перехода, выраженной в числе шагов.

Следует заметить, что при моделировании канала связи как стохастического процесса в основном используются модельные представления КС в форме дискретных однородных регулярных КЦМ. Нетрудно видеть, что дискретная однородная регулярная КЦМ может быть задана с помощью графа. На рисунке 8.1 представлен граф КЦМ = $\{S, P, T, \Pi, K\}$ для случая $n = 3$. На основании представленного графа для компонентов рекуррентной процедуры (7) нетрудно записать

$$\begin{aligned} P_1(k+1) &= P_1(k)P_{11} + P_2(k)P_{21} + P_3(k)P_{31}, \\ P_2(k+1) &= P_1(k)P_{12} + P_2(k)P_{22} + P_3(k)P_{32}, \\ P_3(k+1) &= P_1(k)P_{13} + P_2(k)P_{23} + P_3(k)P_{33}. \end{aligned} \quad (8)$$

Полученные соотношения позволяют сформировать алгоритмы вычисления вектора-строки P_f финальных вероятностей однородной дискретной регулярной КЦМ, моделирующей некоторый дискретный стохастический процесс.

Рассмотрим в качестве примера регулярную КЦМ с заданной матрицей переходных вероятностей $\Pi = \begin{bmatrix} 0.1 & 0.2 & 0.7 \\ 1/3 & 1/3 & 1/3 \\ 0.6 & 0.25 & 0.15 \end{bmatrix}$. Необходимо найти вектор-строку финальных вероятностей. Эту задачу можно решить двумя способами.

В первом случае необходимо решить векторно-матричное уравнение $P_f = P_f\Pi$ с учетом выполнения условия $\sum_{i=1}^n P_{fi} = 1$, или, в развернутом виде:

$$P_{f1} = 0.1P_{f1} + 0.33P_{f2} + 0.6P_{f3};$$

$$P_{f2} = 0.2P_{f1} + 0.33P_{f2} + 0.25P_{f3};$$

$$P_{f3} = 0.7P_{f1} + 0.33P_{f2} + 0.15P_{f3};$$

$$P_{f1} + P_{f2} + P_{f3} = 1.$$

Полученные соотношения позволяют записать векторно-матричное выражение, если просуммировать последнюю строчку с первой, в форме

$$[1 \ 0 \ 0] = [P_{f1} \ P_{f2} \ P_{f3}] \begin{bmatrix} 1.9 & 0.2 & 0.7 \\ 0.667 & -0.667 & 0.333 \\ 0.4 & 0.25 & -0.85 \end{bmatrix},$$

откуда для вектора-строки финальных вероятностей получим

$$P_f = [P_{f1} \ P_{f2} \ P_{f3}] = [0.355 \ 0.253 \ 0.392].$$

Второй способ подразумевает формирование матрицы Π^k , где

$$k = \arg \{ rcond(\Pi^k) \leq 10^{-12} \}.$$

Для рассматриваемого примера получим

$$\Pi^k = \begin{bmatrix} 0.355 & 0.253 & 0.392 \\ 0.355 & 0.253 & 0.392 \\ 0.355 & 0.253 & 0.392 \end{bmatrix};$$

$$k = \arg \{ rcond(\Pi^k) \leq 10^{-12} \} = \arg \{ 1.4e - 014 \leq 10^{-12} \} = 14.$$

Тогда вектор финальных вероятностей можно найти, положив $P_f = (\Pi^k)^i$; $i = \overline{1, n}$:

$$P_f = [P_{f1} \ P_{f2} \ P_{f3}] = (\Pi^k)^j = [0.355 \ 0.253 \ 0.392].$$

Удостовериться в правильности полученного вектора финальных вероятностей можно, проверив выполнение условия $P_f = P_f \Pi$:

$$[0.355 \ 0.253 \ 0.392] \begin{bmatrix} 0.1 & 0.2 & 0.7 \\ 1/3 & 1/3 & 1/3 \\ 0.6 & 0.25 & 0.15 \end{bmatrix} = [0.355 \ 0.253 \ 0.392].$$

Условие выполняется, а значит, вектор финальных вероятностей рассчитан корректно.

Погрузим теперь двоичный канал связи в инструментарий конечных регулярных однородных цепей Маркова. ДКС, процессы в котором моделируются однородной регулярной КЦМ, называется макровектор

$$\text{ДКС} = \{S, P, T, \Pi, K, \{(p_{01}, p_{10})_i; i = \overline{1, n}\}\}, \quad (9)$$

где $S = \{S_i; i = \overline{1, n}\}$ – счетное множество состояний;
 $P = \text{row} \{P_i; i = \overline{1, n}\}$ – вектор-строка вероятностей пребывания в состояниях S_i ;
 $T = \{\Delta t_i; i = \overline{1, n}\}$ – множество длительностей пребывания в состояниях S_i ;
 $\Pi = \{[\text{row}(P_{ij}); j = \overline{1, n}]; i = \overline{1, n}\}$ – матрица переходных вероятностей;
 $K = \{k : 0 \leq k < \infty\}$ – счетное множество дискретных моментов времени, с помощью которых параметризуется стохастический процесс так, что для вектора вероятностей $P(k+1)$ пребывания в состоянии перехода $S(k+1)$ при условии, что стохастический процесс находился с вероятностью $P(k)$ в исходном (текущем) состоянии $S(k)$, оказывается справедливым представление в виде рекуррентной процедуры

$$P(k+1) = P(k)\Pi(k), P(0) = P\{S(0)\}, \quad (10)$$

где $P(0) = P\{S(0)\}$ – вектор вероятности пребывания в начальном состоянии;
 $\{(p_{01}, p_{10})_i; i = \overline{1, n}\}$ – множество пар вероятностей искажения в состояниях S_i .

Если ДКС моделируется однородной регулярной КЦМ, характеризующейся вектором финальных вероятностей $P_f = \text{row} \{P_{fi}; i = \overline{1, n}\}$, а искажения характеризуются парой вероятностей $\{(p_{01}, p_{10})_i; i = \overline{1, n}\}$ для состояния S_i , то интегральная пара вероятностей трансформации примет вид

$$p_{01} = \sum_{i=1}^n P_{fi} (p_{01})_i; p_{10} = \sum_{i=1}^n P_{fi} (p_{10})_i. \quad (11)$$

Сформулированное утверждение позволяет построить алгоритм формирования пользовательской модели двоичного канала связи, характеризующегося интегральной парой (p_{01}, p_{10}) вероятностей трансформации соответственно 0 в 1 и наоборот бита двоичного кода при передаче его по ДКС.

Пример

В качестве примера рассмотрим случай, когда ДКС моделируется регулярной КЦМ с двумя состояниями $S_1 : (p_{01}, p_{10})_1 = (10^{-3}, 5 \cdot 10^{-4})$ и $S_2 : (p_{01}, p_{10})_2 = (10^{-4}, 3 \cdot 10^{-5})$ и матрицей переходных вероятностей $\Pi = \begin{bmatrix} 1 - P_{12} & P_{12} \\ P_{21} & 1 - P_{21} \end{bmatrix}$, в которой $P_{12} = 0.45; P_{21} = 0.55$. Необходимо вычислить интегральную пару (p_{01}, p_{10}) вероятностей трансформации соответственно 0 в 1 и наоборот бита двоичного кода при передаче его по ДКС.

Рассчитаем сначала вектор финальных вероятностей для заданной КЦМ:

$$P_f = [1 \ 0] \begin{bmatrix} 1 - P_{12} & P_{12} \\ 1 + P_{21} & -P_{21} \end{bmatrix}^{-1} = \begin{bmatrix} \frac{P_{21}}{P_{12} + P_{21}} & \frac{P_{12}}{P_{12} + P_{21}} \end{bmatrix} = [0.55 \ 0.45].$$

Далее вычислим искомую интегральную пару (p_{01}, p_{10}) вероятностей трансформации соответственно 0 в 1 и наоборот бита двоичного кода при передаче его по ДКС, воспользовавшись соотношениями (11):

$$p_{01} = \sum_{i=1}^n P_{fi} (P_{01})_i = 0.55 \cdot 10^{-3} + 0.45 \cdot 10^{-4} = 0.595 \cdot 10^{-3};$$

$$p_{10} = \sum_{i=1}^n P_{fi} (P_{10})_i = 0.55 \cdot 5 \cdot 10^{-4} + 0.45 \cdot 3 \cdot 10^{-5} = 0.1785 \cdot 10^{-4};$$

Порядок выполнения работы

1. Построить граф состояний и переходов КЦМ с матрицей вероятностей перехода вида

$$\begin{bmatrix} 1 - P_{12} - P_{13} & P_{12} & P_{13} \\ P_{21} & 1 - P_{21} - P_{23} & P_{23} \\ P_{31} & P_{32} & 1 - P_{31} - P_{32} \end{bmatrix};$$

2. рассчитать вектор вероятностей пребывания КЦМ в состояниях $S(m)$;
3. рассчитать вектор финальных вероятностей КЦМ двумя способами, сравнить полученные результаты;
4. рассчитать интегральную пару вероятностей трансформации бита двоичного кода;
5. сформулировать выводы по работе.

Варианты задания

P_{12}	P_{13}	P_{21}	P_{23}	P_{31}	P_{32}	$P(0)$	m	$(p_{01}, p_{10})_i$
0.5	0.3	0.2	0.35	0.25	0.35	$P(0) = [0.25 \ 0.5 \ 0.25]$	6	$S_1 : (6 \cdot 10^{-4}, 2 \cdot 10^{-3})$ $S_2 : (10^{-4}, 10^{-5})$ $S_3 : (2 \cdot 10^{-4}, 10^{-4})$
0.6	0.15	0.3	0.4	0.55	0.2	$P(0) = [0.3 \ 0.4 \ 0.3]$	7	$S_1 : (7 \cdot 10^{-5}, 3 \cdot 10^{-4})$ $S_2 : (6 \cdot 10^{-5}, 3 \cdot 10^{-4})$ $S_3 : (4 \cdot 10^{-5}, 2 \cdot 10^{-4})$
0.1	0.7	0.15	0.3	0.6	0.2	$P(0) = [0.25 \ 0.2 \ 0.55]$	6	$S_1 : (3 \cdot 10^{-5}, 2 \cdot 10^{-4})$ $S_2 : (10^{-5}, 2 \cdot 10^{-5})$ $S_3 : (5 \cdot 10^{-4}, 10^{-3})$
0.35	0.25	0.2	0.3	0.15	0.15	$P(0) = [0.7 \ 0.1 \ 0.2]$	5	$S_1 : (3 \cdot 10^{-5}, 2 \cdot 10^{-5})$ $S_2 : (10^{-5}, 5 \cdot 10^{-5})$ $S_3 : (8 \cdot 10^{-4}, 4 \cdot 10^{-3})$
0.3	0.1	0.25	0.2	0.6	0.1	$P(0) = [0.25 \ 0.6 \ 0.15]$	6	$S_1 : (5 \cdot 10^{-5}, 4 \cdot 10^{-3})$ $S_2 : (10^{-3}, 7 \cdot 10^{-4})$ $S_3 : (3 \cdot 10^{-4}, 6 \cdot 10^{-3})$
0.7	0.15	0.05	0.8	0.3	0.2	$P(0) = [0.1 \ 0.2 \ 0.7]$	7	$S_1 : (7 \cdot 10^{-4}, 3 \cdot 10^{-4})$ $S_2 : (10^{-4}, 6 \cdot 10^{-5})$ $S_3 : (4 \cdot 10^{-4}, 7 \cdot 10^{-3})$
0.8	0.05	0.4	0.45	0.1	0.6	$P(0) = [0.5 \ 0.2 \ 0.3]$	5	$S_1 : (6 \cdot 10^{-5}, 2 \cdot 10^{-3})$ $S_2 : (10^{-5}, 3 \cdot 10^{-5})$ $S_3 : (2 \cdot 10^{-4}, 9 \cdot 10^{-3})$
0.25	0.35	0.15	0.25	0.45	0.2	$P(0) = [0.6 \ 0.2 \ 0.2]$	6	$S_1 : (2 \cdot 10^{-4}, 8 \cdot 10^{-5})$ $S_2 : (10^{-3}, 8 \cdot 10^{-3})$ $S_3 : (6 \cdot 10^{-4}, 3 \cdot 10^{-3})$
0.3	0.5	0.1	0.6	0.3	0.3	$P(0) = [0.3 \ 0.15 \ 0.55]$	7	$S_1 : (9 \cdot 10^{-5}, 2 \cdot 10^{-5})$ $S_2 : (10^{-4}, 9 \cdot 10^{-4})$ $S_3 : (5 \cdot 10^{-4}, 5 \cdot 10^{-3})$
0.4	0.25	0.3	0.35	0.7	0.1	$P(0) = [0.4 \ 0.5 \ 0.1]$	6	$S_1 : (5 \cdot 10^{-3}, 5 \cdot 10^{-4})$ $S_2 : (10^{-5}, 4 \cdot 10^{-5})$ $S_3 : (7 \cdot 10^{-4}, 2 \cdot 10^{-3})$

4 Помехозащитное кодирование

Для того, чтобы дать определение помехозащищенного кода (ПЗК), дадим сначала определение кодовому расстоянию. Кодовое расстояние между двумя n -разрядными кодовыми комбинациями x и y есть количество соответствующих разрядов этих кодовых комбинаций, принимающих разные значения, и вычисляется по формуле

$$d(x, y) = \sum_{i=1}^n x_i \oplus y_i. \quad (12)$$

Код является помехозащищенным, если минимальное кодовое расстояние между любыми двумя разрешенными комбинациями этого кода больше 1.

Увеличение минимального кодового расстояния в ПЗК обеспечивается путем введения в структуру кода m проверочных разрядов. При этом формат ПЗК записывается в виде (n, k) , где k - число информационных разрядов кода, а $n = k + m$ - суммарное число разрядов ПЗК. Если число m проверочных разрядов подбираются эвристически, ПЗК называется несистематическим, если же проверочные разряды функционально связаны с информационными, то ПЗК называется систематическим. Если эта функциональная связь линейная, то и систематический ПЗК называют линейным, в противном случае систематический ПЗК - нелинейный. Систематика может носить блоковый характер, когда информационные и проверочные разряды занимают зафиксированные позиции в структуре кода, при этом блоковая систематика может быть полной (старшие разряды - информационные, младшие - проверочные) либо неполной.

Значения k , m , n и минимальное кодовое расстояние d_{\min} относят к базовым параметрам ПЗК. Также к ним относятся модель искажений в канале связи (КС), характеризующаяся вероятностями преобразования из "0" в "1" p_{01} и из "1" в "0" p_{10} , количество исправляемых ошибок s и допустимая вероятность ложного приема $P_{\text{доп}}$, которая задается категорией системы (см. таблицу 6).

Таблица 6:

	Категория системы передачи информации		
	<i>I</i>	<i>II</i>	<i>III</i>
Допустимая вероятность ложного приема $P_{\text{доп}}$	10^{-14}	10^{-10}	10^{-7}

Формирование базовых параметров ПЗК осуществляется следующим образом [1]. Пусть задан массив сообщений в виде их количества $V_{\text{и}}$, и необходимо сформировать формат (n, k) -ПЗК при известных вероятности искажений в КС $p = \max\{p_{01}, p_{10}\}$ и допустимой вероятности ложного приема $P_{\text{доп}}$.

Тогда необходимое число k информационных разрядов вычисляется из соотношения

$$k = \min_k \arg \{2^k \geq V_{\text{и}}\}. \quad (13)$$

Число m проверочных разрядов вычисляется на основе заданных базовых параметров ПЗК в силу следующего соотношения:

$$m = \min_m \arg \left\{ N_c = 2^m - 1 \geq N_{\xi} = \sum_{i=1}^s C_n^i \right\}, \quad (14)$$

где N_c - число ненулевых синдромов ошибок, которое могут породить m проверочных разрядов, а N_{ξ} - число возможных ошибок кратности от 1 до s . Полученное значение m должно удовлетворять условию

$$P_{\text{ош}} = \sum_{i=s+1}^n C_n^i p^i (1-p)^{n-i} \leq P_{\text{доп}}, \quad (15)$$

где $P_{\text{ош}}$ - вероятность ошибки при передаче информации.

Расчет осуществляется по рекуррентной процедуре: на первом шаге предполагается, что для обеспечения требуемой допустимой вероятности ложного приема $P_{\text{доп}}$ достаточно исправлять ошибки кратности $s = 1$. Путем подстановки значения s в (14) производится расчет значения m , затем значения s и m подставляются в (15). В случае, если итоговая вероятность ошибки $P_{\text{ош}}$ превышает $P_{\text{доп}}$, кратность исправляемых ошибок s увеличивается на 1, и процедура повторяется до тех пор, пока условие в (15) не будет выполнено.

Стоит отметить, что в случае, если выполняется соотношение $2^m - 1 = n$, ПЗК называют оптимальным. Такой ПЗК отличается тем, что использует все возможные синдромы ошибок, порождаемые проверочными разрядами.

Конструирование матриц систематических ПЗК

Полная схема, описывающая процесс кодирования, состоящий в преобразовании исходного помехозащищенного кода (ПНЗК) в помехозащищенный, его передачу по двоичному каналу связи, сопровождающуюся искажением помехозащищенного кода, и процесс декодирования принятого из КС кода с целью формирования кода синдрома (опознавателя) внесенной при передаче ошибки (искажения), приведена на рисунке 2.

На рисунке 2 КУ – кодирующее устройство, КС – канал связи, искажение в котором моделируется сумматором по модулю два помехозащищенного кода и кода ошибки, ДКУ – декодирующее устройство, формирующее синдром

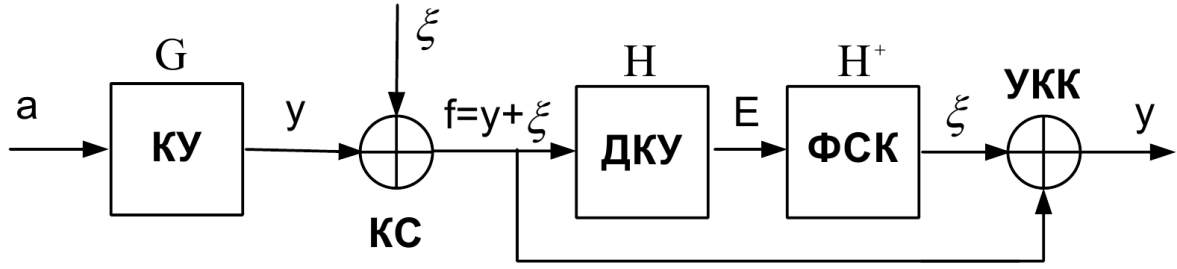


Рис. 2: Схема процесса помехозащитного кодирования-декодирования.

ошибки, a – вектор-строка исходного ПЗК размерности k , y – вектор-строка помехозащищенного (n, k) -ПЗК размерности n на выходе КУ, ξ – вектор-строка помехи размерности n , воздействующей на код y при его передаче по КС, f – вектор-строка искаженного кода, принимаемого из КС, E – вектор-строка размерности t синдрома ошибки в принятой из КС кодовой комбинации f .

Процесс формирования вектора-строки ПЗК y из вектора-строки ПЗК a , осуществляемый в КУ, может быть описан векторно-матричным соотношением

$$y = aG, \quad (16)$$

где G – образующая матрица ПЗК размерности $(k \times n)$.

Процесс искажения передаваемой кодовой комбинации y в КС под действием помехи ξ может быть представлен операцией суммирования

$$f = y + \xi. \quad (17)$$

Процесс декодирования, состоящий в формировании вектора-строки синдрома ошибки E из вектора-строки принятого из КС искаженного кода f , осуществляемый в ДКУ, может быть описан векторно-матричным соотношением

$$E = fH, \quad (18)$$

где H – проверочная матрица ПЗК размерности $(n \times t)$.

Для того, чтобы матрицы G и H порождали помехозащищенный код, необходимо выполнение условия

$$GH = O. \quad (19)$$

Рассмотрим два способа формирования матриц G и H без требований к систематике кода. Первый способ основан на использовании аналитических проверочных равенств процедур кодирования (16) и декодирования (18). В рамках данного способа необходимо составить таблицу кодировок векторов-строк однократных ошибок ξ_i соответствующими синдромами ошибок E_i начиная с

ошибки в старшем разряде и сформировать из синдромов ошибок проверочную матрицу H .

Далее, на основании выражения (16), формируются аналитические выражения для вычисления каждого разряда синдрома ошибки как произведения вектора-строки искаженной кодовой комбинации f на соответствующий столбец матрицы H . Затем необходимо сформировать аналитические выражения для помехозащитного кодирования, характеризующемся выполнением условий $\xi = 0$, $f = y$ и $E = 0$, получив в результате систему равенств, допускающих явное разрешение относительно разрядов y_i ПЗК как линейных комбинаций информационных разрядов ПЗК. Наконец, на основании равенства (16) сформировать образующую матрицу G . Стоит отметить, что данный способ допускает неоднозначность нахождения образующей матрицы.

Второй способ формирования матриц G и H опирается на утверждение о том, что столбцы транспонированной образующей матрицы G^T принадлежат ядру транспонированной проверочной матрицы H^T кода, при этом выполняется соотношение

$$G_j^T \in \ker H^T \vee H^T G_j^T = O, \quad (20)$$

где G_j^T - j -й столбец транспонированной образующей матрицы G^T . Стоит отметить, что можно добиться желаемых свойств ПЗК путем перестановки и/или суммирования строк полученной образующей матрицы G .

Наконец, рассмотрим формирование матриц ПЗК с полной блоковой систематикой. Для того, чтобы ПЗК, порождаемый образующей матрицей G , с проверочной матрицей H обладал полной блоковой систематикой, его матрицы должны быть сформированы в виде

$$G = [I_k \quad \tilde{G}], \quad H = \begin{bmatrix} \tilde{G} \\ I_m \end{bmatrix}, \quad (21)$$

где I_a - единичная матрица размерности $(a \times a)$, а \tilde{G} - матрица синдромов ошибок размерности $(k \times m)$.

Иными словами, для формирования ПЗК с полной блоковой систематикой при формировании проверочной матрицы синдромы ошибок в первых m разрядах формируются в виде единичной матрицы. Данный способ отличается вычислительной простотой и не требует большого количества вычислений.

Пример

В качестве примера рассмотрим процедуру формирования ПЗК для массива сообщений объемом $V_{\text{н}} = 60$ при вероятности искажений в КС

$$p = \max \{p_{01} = 5 \cdot 10^{-5}, p_{10} = 10^{-4}\} = 10^{-4}$$

и допустимой вероятности ложного приема $P_{\text{доп}} = 10^{-7}$.

Начнем с формирования базовых параметров ПЗК. В силу (13) число информационных разрядов будет равно

$$k = \min_k \arg \{2^k \geq 60\} = 6.$$

Число проверочных разрядов m будем искать согласно (14). Положим сначала $s = 1$, тогда

$$m = \min_m \arg \left\{ N_c = 2^m - 1 \geq N_\xi = \sum_{i=1}^1 C_n^i = n = 6 + m \right\} = 4.$$

Проверим выполнение условия (15):

$$P_{\text{ош}} = \sum_{i=2}^{10} C_1 0^i p^i (1-p)^{10-i} \simeq 4.5 \cdot 10^{-7} > P_{\text{доп}} = 10^{-7}.$$

Так как вероятность ошибки превышает допустимую, увеличим s на единицу. Тогда

$$m = \min_m \arg \left\{ N_c = 2^m - 1 \geq N_\xi = \sum_{i=1}^2 C_n^i = n + \frac{(n-1)n}{2} = \frac{(m+7)(m+6)}{2} \right\} = 7,$$

а

$$P_{\text{ош}} = \sum_{i=3}^{13} C_1 3^i p^i (1-p)^{13-i} \simeq 2.86 \cdot 10^{-10} \ll P_{\text{доп}} = 10^{-7}.$$

Так как вероятность ошибки существенно ниже допустимой, искомый формат ПЗК имеет вид (13, 6).

Теперь сформируем образующую и проверочную матрицы ПЗК. Воспользуемся первым способом. Синдромы однократных ошибок в (13, 6)-ПЗК сведены в таблицу 4.

Составленные синдромы ошибок построчно составляют проверочную матрицу H , которая в транспонированном виде имеет вид

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

	Вектор-строка ошибки	Вектор-строка синдрома ошибки						
		E_{i7}	E_{i6}	E_{i5}	E_{i4}	E_{i3}	E_{i2}	E_{i1}
$\xi_{13} =$	1 0 0 0 0 0 0 0 0 0 0 0 0 0	1	0	0	0	0	0	0
$\xi_{12} =$	0 1 0 0 0 0 0 0 0 0 0 0 0 0	0	0	0	0	0	1	0
$\xi_{11} =$	0 0 1 0 0 0 0 0 0 0 0 0 0 0	0	0	0	0	1	0	0
$\xi_{10} =$	0 0 0 1 0 0 0 0 0 0 0 0 0 0	0	0	0	1	0	0	0
$\xi_9 =$	0 0 0 0 1 0 0 0 0 0 0 0 0 0	0	0	1	0	0	0	0
$\xi_8 =$	0 0 0 0 0 1 0 0 0 0 0 0 0 0	0	1	0	0	0	0	0
$\xi_7 =$	0 0 0 0 0 0 1 0 0 0 0 0 0 0	1	0	0	0	0	1	0
$\xi_6 =$	0 0 0 0 0 0 0 1 0 0 0 0 0 0	0	0	0	0	0	0	1
$\xi_5 =$	0 0 0 0 0 0 0 0 1 0 0 0 0 0	0	0	0	1	1	0	0
$\xi_4 =$	0 0 0 0 0 0 0 0 0 1 0 0 0 0	0	0	1	1	0	0	0
$\xi_3 =$	0 0 0 0 0 0 0 0 0 0 1 0 0 0	0	1	1	0	0	0	0
$\xi_2 =$	0 0 0 0 0 0 0 0 0 0 0 1 0 0	1	1	0	0	0	0	1
$\xi_1 =$	0 0 0 0 0 0 0 0 0 0 0 0 0 1	0	0	0	0	1	1	1

Составим теперь на основании (18) аналитические выражения для вычисления каждого разряда синдрома ошибки:

$$E_7 = f_{13} + f_7 + f_2;$$

$$E_6 = f_8 + f_3 + f_2;$$

$$E_5 = f_9 + f_4 + f_3;$$

$$E_4 = f_{10} + f_5 + f_4;$$

$$E_3 = f_{11} + f_5 + f_1;$$

$$E_2 = f_{12} + f_7 + f_1;$$

$$E_1 = f_6 + f_1 + f_2.$$

Далее сформируем аналитические выражения для помехозащитного кодирования, характеризующегося выполнением условий $\xi = 0$, $f = y$ и $E = 0$, предположив, что $y_7 = a_6$, $y_5 = a_5$, $y_4 = a_4$, $y_3 = a_3$, $y_2 = a_2$, $y_1 = a_1$, получим систему уравнений

$$y_6 = a_1 + a_2;$$

$$y_8 = a_3 + a_2;$$

$$y_9 = a_4 + a_3;$$

$$y_{10} = a_5 + a_4;$$

$$y_{11} = a_5 + a_1;$$

$$y_{12} = a_6 + a_1;$$

$$y_{13} = a_6 + a_2.$$

Наконец, на основании (16) сформируем образующую матрицу G :

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

Путем проверки выполнения условия (19) можно убедиться, что образующая и проверочная матрицы сформированного (13,6)-ПЗК составлены корректно.

Для иллюстрации второго способа формирования матриц воспользуемся полученной ранее матрицей H^T . Необходимо сформировать транспонированную образующую матрицу G^T в силу условия (20), в результате получим

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

Запишем теперь образующую матрицу G :

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

Путем проверки выполнения условия (19) можно убедиться, что образующая и проверочная матрицы сформированного (13, 6)-ПЗК составлены корректно.

Сформируем теперь для полученного сформированного (13, 6)-ПЗК матрицы, придающие ему полную блоковую систематику. Действуя по аналогии с процедурой, приведенной в таблице 4, с учетом (21) получим проверочную матрицу ПЗК с полной блоковой систематикой:

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

Тогда, согласно (21), образующая матрица ПЗК с полной блоковой систематикой примет вид

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

Путем проверки выполнения условия (19) можно убедиться, что образующая и проверочная матрицы сформированного (13, 6)-ПЗК составлены корректно.

Порядок выполнения работы

1. Сформировать по исходным данным базовые параметры ПЗК;
2. Сформировать образующую и проверочную матрицы ПЗК с использованием аналитических проверочных равенств;

3. Сформировать образующую и проверочную матрицы ПЗК с использованием аппарата нуль-пространств;
4. Сформировать образующую и проверочную матрицы ПЗК с полной блоковой систематикой;
5. Сформулировать выводы по работе.

Варианты задания

1. Массив сообщений объемом $V_{\text{и}} = 60$, вероятности искажений в КС $p_{01} = 7 \cdot 10^{-4}$ и $P_{10} = 2 \cdot 10^{-3}$, допустимая вероятность ложного приема $P_{\text{доп}} = 10^{-7}$;
2. Массив сообщений объемом $V_{\text{и}} = 100$, вероятности искажений в КС $p_{01} = 6 \cdot 10^{-4}$ и $P_{10} = 2 \cdot 10^{-5}$, допустимая вероятность ложного приема $P_{\text{доп}} = 10^{-7}$;
3. Массив сообщений объемом $V_{\text{и}} = 130$, вероятности искажений в КС $p_{01} = 6 \cdot 10^{-4}$ и $P_{10} = 10^{-3}$, допустимая вероятность ложного приема $P_{\text{доп}} = 10^{-7}$;
4. Массив сообщений объемом $V_{\text{и}} = 130$, вероятности искажений в КС $p_{01} = 6 \cdot 10^{-4}$ и $P_{10} = 10^{-3}$, допустимая вероятность ложного приема $P_{\text{доп}} = 10^{-7}$;
5. Массив сообщений объемом $V_{\text{и}} = 60$, вероятности искажений в КС $p_{01} = 4 \cdot 10^{-5}$ и $P_{10} = 2 \cdot 10^{-5}$, допустимая вероятность ложного приема $P_{\text{доп}} = 10^{-10}$;
6. Массив сообщений объемом $V_{\text{и}} = 100$, вероятности искажений в КС $p_{01} = 9 \cdot 10^{-5}$ и $P_{10} = 3 \cdot 10^{-4}$, допустимая вероятность ложного приема $P_{\text{доп}} = 10^{-10}$;
7. Массив сообщений объемом $V_{\text{и}} = 130$, вероятности искажений в КС $p_{01} = 4 \cdot 10^{-4}$ и $P_{10} = 10^{-4}$, допустимая вероятность ложного приема $P_{\text{доп}} = 10^{-10}$;
8. Массив сообщений объемом $V_{\text{и}} = 60$, вероятности искажений в КС $p_{01} = 5 \cdot 10^{-6}$ и $P_{10} = 4 \cdot 10^{-5}$, допустимая вероятность ложного приема $P_{\text{доп}} = 10^{-14}$;
9. Массив сообщений объемом $V_{\text{и}} = 100$, вероятности искажений в КС $p_{01} = 10^{-6}$ и $P_{10} = 3 \cdot 10^{-6}$, допустимая вероятность ложного приема $P_{\text{доп}} = 10^{-14}$;
10. Массив сообщений объемом $V_{\text{и}} = 130$, вероятности искажений в КС $p_{01} = 8 \cdot 10^{-7}$ и $P_{10} = 2 \cdot 10^{-6}$, допустимая вероятность ложного приема $P_{\text{доп}} = 10^{-14}$;
11. Массив сообщений объемом $V_{\text{и}} = 15$, вероятности искажений в КС $p_{01} = 2 \cdot 10^{-4}$ и $P_{10} = 2 \cdot 10^{-3}$, допустимая вероятность ложного приема $P_{\text{доп}} = 10^{-7}$;

12. Массив сообщений объемом $V_{\text{и}} = 50$, вероятности искажений в КС $p_{01} = 4 \cdot 10^{-4}$ и $P_{10} = 3 \cdot 10^{-5}$, допустимая вероятность ложного приема $P_{\text{доп}} = 10^{-10}$;
13. Массив сообщений объемом $V_{\text{и}} = 150$, вероятности искажений в КС $p_{01} = 4 \cdot 10^{-4}$ и $P_{10} = 6 \cdot 10^{-4}$, допустимая вероятность ложного приема $P_{\text{доп}} = 10^{-7}$;
14. Массив сообщений объемом $V_{\text{и}} = 280$, вероятности искажений в КС $p_{01} = 3 \cdot 10^{-5}$ и $P_{10} = 10^{-5}$, допустимая вероятность ложного приема $P_{\text{доп}} = 10^{-10}$;
15. Массив сообщений объемом $V_{\text{и}} = 7$, вероятности искажений в КС $p_{01} = 3 \cdot 10^{-5}$ и $P_{10} = 7 \cdot 10^{-5}$, допустимая вероятность ложного приема $P_{\text{доп}} = 10^{-14}$;

5 Формирование ПЗК на основе действий с модулярными многочленами

Модулярный многочлен (ММ) - это такой многочлен, коэффициенты которого принадлежат простому полю Галуа $GF(p)$. В задачах формирования двоичных ПЗК применяются ММ, коэффициенты которых принадлежат простому полю Галуа с модулем $p = 2$. Так как в данной работе кодовые последовательности представляются в виде ММ, схема процесса помехозащитного кодирования-декодирования, приведенная на рисунке 2, преобразуется в схему на рисунке 3.

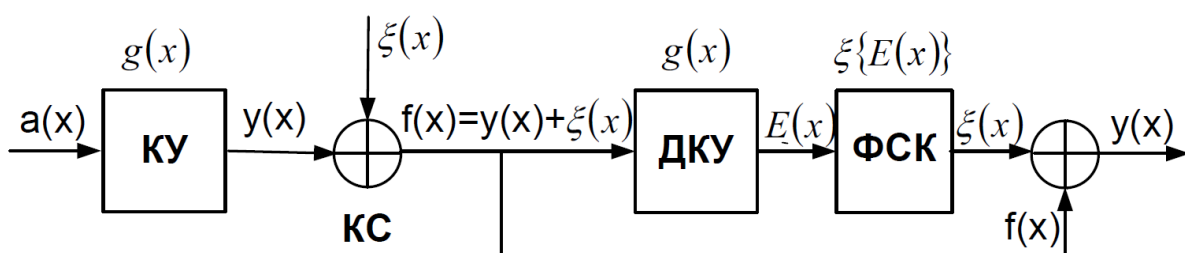


Рис. 3: Схема процесса помехозащитного кодирования-декодирования.

Функциональные блоки на схеме на рисунке 3 имеют тот же смысл, что и на схеме на рисунке 2, $a(x)$ - ММ исходного ПЗК степени $k - 1$, $y(x)$ - ММ (n, k) -ПЗК y степени $n - 1$, $\xi(x)$ - ММ помехи ξ , воздействующей на код y при его передаче по КС, степени $n - 1$, $f(x) = y(x) + \xi(x)$ - ММ искаженного кода f , принимаемого из КС, степени $n - 1$, $E(x)$ - ММ синдрома E ошибки в принятой из КС кодовой комбинации степени $m - 1$.

Помехозащитным кодированием при использовании представления кодов в виде ММ называется процесс формирования ММ $y(x)$ помехозащищенного кода y из ММ $a(x)$ ПЗК a , осуществляемый в КУ, при котором на основе ММ $a(x)$ степени $k - 1$ при помощи образующего ММ $g(x)$ степени m ММ $y(x)$ приобретает свойство делимости на образующий ММ без остатка, т.е.

$$\text{rest} \frac{y(x)}{g(x)} = 0. \quad (22)$$

ПЗК, кодовые комбинации которого удовлетворяют соотношению (22), принято называть циклическим ПЗК, так как характеристическое свойство (22) сохраняется на любой кодовой комбинации, получаемой из начальной путем ее циклического сдвига влево на любые l разрядов, осуществляемого умножением ММ начальной кодовой комбинации на x^l .

Помехозащитным декодированием при использовании представления кодов в виде ММ называется процесс формирования ММ $E(x)$ остатка от деления

ММ $f(x)$ принятого из КС кода f на образующий ММ $g(x)$, с целью проверки сохранности при передаче характеристического свойства ПЗК (22), в форме

$$E = \text{rest} \frac{f(x)}{g(x)}. \quad (23)$$

В случае, если образуется нулевой синдром ошибки, принятый код f не содержит искажений, в противном случае принятая кодовая комбинация искажена. При этом в случае использования режима обнаружения ненулевой ММ синдрома ошибки свидетельствует только о факте ошибки в принятой из КС кодовой комбинации, а в случае использования режима исправления искажений код ММ синдрома ошибки является адресом искаженных разрядов.

Образующий ММ $g(x)$ должен генерировать как можно большее количество уникальных синдромов ошибок. В связи с этим требуется, чтобы образующий ММ был неприводимым, т.е. образующий ММ $g(x)$ должен без остатка делиться только на себя и на 1. Это свойство гарантирует порождение при делении на образующий ММ множества остатков максимальной мощности. Неприводимый ММ, как правило, не имеет корней в простом поле Галуа $GF(2)$, а значит, всегда имеет нечетное число членов и свободный член, равный единице. Также стоит отметить, что минимальное кодовое расстояние d_{\min} циклического ПЗК с образующим ММ $g(x)$ не превышает числа ненулевых элементов образующего ММ. Это обстоятельство важно помнить, если требования к корректирующей способности кода включают требуемую кратность исправляемых ошибок s и обнаруживаемых ошибок r , которые связаны с d_{\min} следующими соотношениями:

$$d_{\min} \geq 2s + 1; \quad (24)$$

$$d_{\min} \geq r + 1. \quad (25)$$

Формирование циклического ПЗК в форме ММ $y(x)$, удовлетворяющего соотношению (22) осуществляется одним из трех способов:

1. способ, основанный на перемножении ММ [3];
2. способ, основанный на делении ММ [3];
3. способ, основанный на формировании образующей и проверочной матриц ПЗК [1].

Согласно первому способу, циклический (n, k) -ПЗК, задаваемый ММ $y(x)$, удовлетворяющим соотношению (22), может быть сформирован помехозащитным кодирующим устройством, осуществляющим перемножение ММ $a(x)$ исходного ПЗК и образующего ММ $g(x)$ так, что ММ $y(x)$ принимает представление

$$y(x) = a(x) \cdot g(x). \quad (26)$$

С точки зрения схемотехники устройство перемножения ММ можно реализовать в виде линейной переключательной схемы (ЛПС), состоящей из ячеек памяти и сумматоров и представляющей собой регистр сдвига с логическими обратными связями.

Рассмотрим процедуру формирования схемы, используемой для умножения произвольного двоичного многочлена $a(x) = a_{n-1}x^{n-1} + a_{n-2}x^{n-2} + \dots + a_1x + a_0$, $a_i \in \{0, 1\}$ на фиксированный многочлен без нулевых членов $h(x) = x^r + x^{r-1} + \dots + x^2 + x + 1$.

Вход и выход схемы предполагаются последовательными. На вход схемы поступают коэффициенты многочлена $a(x)$, начиная с коэффициентов старших разрядов, после чего следует r нулей. Произведение, равное $a(x)h(x) = a_{n-1}x^{n+r-1} + (a_{n-2} + a_{n-1})x^{n+r-2} + \dots + (a_0 + a_1)x + a_0$, формируется на выходе схемы за $(n + r)$ тактов. Построение схемы отвечает следующим правилам:

- число ячеек памяти равно степени многочлена $h(x)$;
- число сумматоров на 1 меньше числа ненулевых членов многочлена; если коэффициент при x^0 равен 0, то число сумматоров равно числу ненулевых членов и множимое подается на сумматоры;
- сумматоры устанавливаются перед ячейками регистра, соответствующими ненулевым членам многочлена тех же степеней; при этом сумматор на выходе схемы стоит перед отсутствующей ячейкой старшей степени;
- сумматор, соответствующий младшему члену многочлена, отбрасывается, т.е. сумматор на входе ячейки П1 отсутствует;
- множимое подается одновременно на вход П1 и на все сумматоры.

Формирование циклического ПЗК на основе перемножения ММ в форме (26) не нашло практического использования в силу отсутствия какой-либо систематики, его разряды невозможно разделить на информационные и проверочные за счет перемешивания разрядов при перемножении. Кроме того, техническое обеспечение процесса кодирования-декодирования согласно данному способу требует использования аппаратуры разного типа, так как кодирование осуществляется путем умножения ММ, а декодирование - путем деления ММ.

Второй способ формирования циклического ПЗК оперирует представлением ММ циклического ПЗК в виде

$$y(x) = a(x)x^m + r(x), \quad (27)$$

где

$$r(x) = \text{rest} \frac{a(x)x^m}{g(x)}, \quad (28)$$

где ММ $y(x)$ удовлетворяет условию (22).

В выражении (27) путем домножения на x^m осуществляется сдвиг исходного ПЗК на m разрядов, что освобождает m младших разрядов для проверочных разрядов, формируемых в силу (28). Таким образом для формируемого ПЗК обеспечивается полная блоковая систематика.

Аппаратно циклический (n, k) -ПЗК в соответствии с выражением (27) формируется следующим образом. Исходный ПЗК a в течение k тактов из устройства формирования кода сообщения (УФКС) подается через линейное устройство (ЛУ) в двоичный канал связи и на вход последовательного регистра, составленного из m элементов памяти с обратными связями, представляющего собой устройство деления на образующий ММ $g(x)$ ПЗК с целью вычисления остатка $r(x)$ в соответствии с выражением (28). На $(k + 1)$ -м такте разрываются обратные связи в устройстве деления ММ (УДММ) с тем, чтобы приостановить процесс деления путем перевода УДММ в режим последовательного регистра сдвига, при этом вход ЛУ КС переключается с выхода УФКС на выход УДММ. Далее в течение m тактов остаток от деления выводится из УДММ через ЛУ в двоичный КС, чем завершается аппаратное формирование циклического (n, k) -ПЗК. Ячейки памяти в исходном состоянии должны содержать нули. Для первых r тактов работы схемы, пока первый входной символ не достигнет конца регистра, выходные символы равны нулю. На $(r + 1)$ такте появляется первый ненулевой выходной символ, который равен первому коэффициенту частного, а в регистре будет находиться остаток от деления (остаток может быть и нулевым).

Построение регистров подобной структуры отвечает следующим правилам:

- число ячеек регистра равно степени многочлена, на который производится деление;
- число сумматоров на 1 меньше числа ненулевых членов много-члена $g(x)$;
- сумматоры ставятся перед ячейками регистра, соответствующими ненулевым членам многочлена $g(x)$ тех же степеней;
- ячейка памяти для члена со старшей степенью и соответствующий ей сумматор отбрасываются, т.к. при делении многочлена старшие члены делимого и делителя всегда равны 1 и, складываясь по модулю два, всегда дают в результате 0.

Декодирующее устройство состоит из входного регистра, регистра деления и детектора ошибок и функционирует следующим образом. На вход ВР и РД одновременно подается входная кодовая комбинация. За первые n тактов происходит деление входной кодовой последовательности на образующий ММ в

РД (в ячейках памяти на n -ом такте находится остаток от деления) и заполнение ВР. В течение следующих n тактов остаток от деления (синдром ошибки) поступает в детектор ошибок, который на основании синдрома восстанавливает вектор ошибки, и полученный вектор ошибки суммируется по модулю 2 со входной кодовой комбинацией из ВР. В результате получается исправленная кодовая комбинация (которая в случае отсутствия ошибок совпадает со входной кодовой комбинацией, при этом первыми выходят k информационных разрядов старшим разрядом вперед (декодированный исходный ПНЗК), а затем m проверочных разрядов).

Третий способ основан на конструировании проверочной матрицы H циклического ПЗК на основании (23) и образующей матрицы циклического ПЗК G , вычисление которой опирается на факт полной блоковой систематики циклических ПЗК. В отличие от второго способа формирования циклического ПЗК, рассматриваемый способ возвращает к параллельному преобразованию кодов в решении задач их помехозащиты. Построенные образующая матрица G и проверочная матрица H используются для аналитического описания процессов помехозащитного кодирования в виде (16) и помехозащитного декодирования в виде (18) циклических кодов.

Стоит отметить, что для последней (k -й) строчки матрицы \tilde{G} справедлива цепочка равенств

$$\tilde{G}^k = H^k = K \left\{ E_{m+1}(x) = \text{rest} \frac{x^m}{g(x)} \right\} = K \{x^m + g(x)\}. \quad (29)$$

Соотношение (29) обладает хорошими идентификационными свойствами, позволяя по последней строке матрицы \tilde{G} определять корректность составления образующей матрицы G ПЗК по заданному образующему ММ $g(x)$ кода, а также определять образующий ММ $g(x)$ ПЗК по заданной образующей матрице G кода.

Пример

В качестве примера рассмотрим ПЗК с параметрами, полученными в примере в задании 4 ((13, 6)-ПЗК). Выберем в качестве образующего ММ $g(x) = x^7 + x^3 + x^2 + x + 1$.

Процесс кодирования исходного ПНЗК с ММ $a(x) = x^5 + x^2 + 1$ методом умножения на образующий ММ проиллюстрирован таблицей 5. Структурная схема соответствующего КУ, реализованная в среде Matlab Simulink, представлена на рисунке 4. Произведение $a(x) \cdot g(x) = x^{12} + x^9 + x^8 + x^6 + x^4 + x + 1$ формируется на выходе за $k + m = 6$ тактов. Первоначально в ячейках памяти содержатся нули.

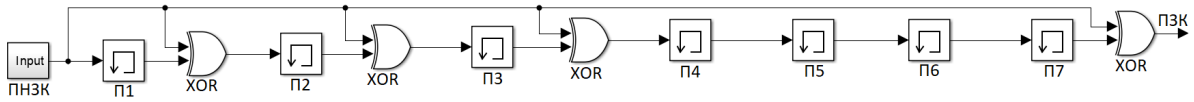


Рис. 4: Структурная схема КУ (13,6)-ПЗК, основанного на методе умножения на образующий ММ.

Номер такта	Множимое (вход)	Состояние ячеек памяти							Произведение (выход)
		П1	П2	П3	П4	П5	П6	П7	
0	—	0	0	0	0	0	0	0	—
1	1	1	1	1	1	0	0	0	1
2	0	0	1	1	1	1	0	0	0
3	0	0	0	1	1	1	1	0	0
4	1	1	1	1	0	1	1	1	1
5	0	0	1	1	1	1	0	1	1
6	1	1	1	0	0	1	0	1	0
7	0	0	1	1	0	0	1	0	1
8	0	0	0	1	1	0	0	1	0
9	0	0	0	0	1	1	0	0	1
10	0	0	0	0	0	1	1	0	0
11	0	0	0	0	0	0	1	1	0
12	0	0	0	0	0	0	0	1	1
13	0	0	0	0	0	0	0	0	1

Полученный результат демонстрирует полное отсутствие в полученной ПЗК какой-либо систематики. Процесс декодирования состоит в делении полученной кодовой комбинации на образующий ММ $g(x)$.

Функционирование устройства деления на образующий ММ многочлена $a(x) \cdot x^m$, где $a(x) = x^5 + x^2 + 1$ проиллюстрировано таблицей 5. Структурная схема соответствующего КУ, реализованная в среде Matlab Simulink, представлена на рисунке 5. Деление осуществляется за $k + 1$ тактов. Первоначальное состояние всех ячеек регистра — нулевое.

По прошествии $k + 1$ тактов в канал связи передается остаток от деления, тем самым дополняя кодовое слово m проверочными разрядами, занимающими младшие позиции. Очевидно, что данный способ кодирования обеспечивает ПЗК полную блоковую систематику.

Структурная схема ДКУ, осуществляющего декодирование ПЗК с ММ исходного ПНЗК $a(x) = x^5 + x^2 + 1$ с использованием регистра деления на образующий ММ $g(x)$, реализованная в среде Matlab Simulink, представлена на рисунке 6. Декодирование осуществляется за $n + k$ тактов в режиме исправле-

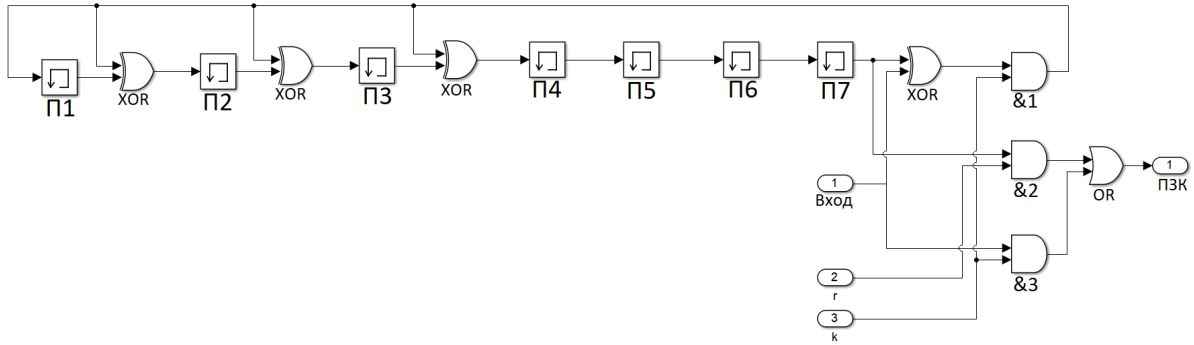


Рис. 5: Структурная схема КУ (13,6)-ПЗК, основанного на методе деления на образующий ММ.

Номер такта	Делимое (вход)	Состояние ячеек памяти							Частное (выход)
		П1	П2	П3	П4	П5	П6	П7	
0	—	0	0	0	0	0	0	0	—
1	1	1	1	1	1	0	0	0	1
2	0	0	1	1	1	1	0	0	0
3	0	1	0	1	1	1	1	0	0
4	1	1	1	1	0	1	1	1	1
5	0	0	0	0	0	0	1	1	1
6	1	0	1	0	0	0	0	1	0
7	0	0	0	1	0	0	0	0	1

ния ошибок первой кратности (исправление ошибок во всей входной кодовой комбинации займет $2n$ тактов, но, так как на выходе ДКУ нужно получить исходный k -разрядный ПЗК, достаточно $n + k$ тактов). Первоначальное состояние всех ячеек регистра - нулевое.

Структуру ДО определим по первой строчке таблицы 7 синдромов однократных ошибок в виде логической функции $y = abc\bar{d}\bar{e}\bar{f}g$.

Наконец, с целью формирования проверочной матрицы H (13,6)-ПЗК с полной блоковой систематикой составим таблицу 7 синдромов однократных ошибок, где синдромы ошибок в первых $m = 7$ разрядах образуют единичную матрицу, а оставшиеся синдромы вычисляются в соответствии с выражением (23).

Из таблицы 7 выпишем проверочную матрицу ПЗК с полной блоковой си-

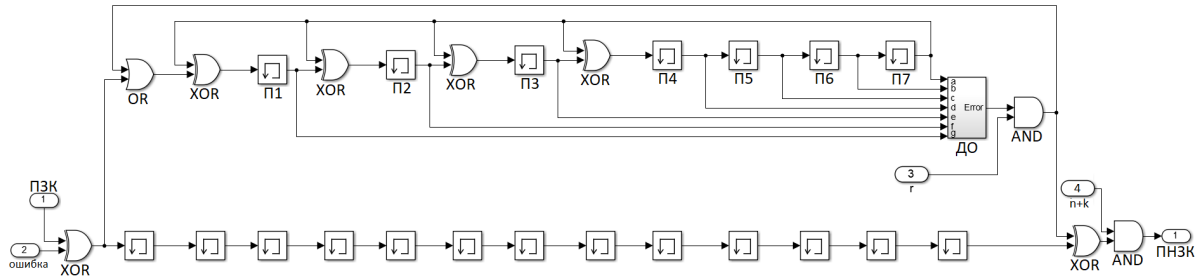


Рис. 6: Структурная схема ДКУ (13,6)-ПЗК, основанного на методе деления на образующий ММ.

Таблица 7: Таблица синдромов однократных ошибок (13,6)-ПЗК с полной блочковой систематикой

ММ ошибки	Вектор-строка синдрома ошибки						
	E_{i7}	E_{i6}	E_{i5}	E_{i4}	E_{i3}	E_{i2}	E_{i1}
x^{12}	1	1	1	0	0	0	1
x^{11}	1	1	1	1	1	1	1
x^{10}	1	1	1	1	0	0	0
x^9	0	1	1	1	1	0	0
x^8	0	0	1	1	1	1	0
x^7	0	0	0	1	1	1	1
x^6	1	0	0	0	0	0	0
x^5	0	1	0	0	0	0	0
x^4	0	0	1	0	0	0	0
x^3	0	0	0	1	0	0	0
x^2	0	0	0	0	1	0	0
x	0	0	0	0	0	1	0
1	0	0	0	0	0	0	1

стематикой:

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

Тогда, согласно (21), образующая матрица ПЗК с полной блоковой систематикой примет вид

$$G = [I_k \quad \tilde{G}] = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 1 & 1 & 1 & 0 & 0 & 0 & 1 \\ 0 & 1 & 0 & 0 & 0 & 0 & 1 & 1 & 1 & 1 & 1 & 1 & 1 \\ 0 & 0 & 1 & 0 & 0 & 0 & 1 & 1 & 1 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 1 & 1 & 1 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 1 & 1 & 1 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 1 & 1 & 1 & 1 \end{bmatrix}.$$

Нетрудно увидеть, что последняя строчка матрицы G удовлетворяет цепочке равенств (29).

Порядок выполнения работы

1. Выбрать образующий ММ для ПЗК с параметрами, полученными в работе 3;
2. Построить и смоделировать кодирующее устройство на основе умножения на выбранный образующий ММ;
3. Построить и смоделировать кодирующее устройство на основе деления на выбранный образующий ММ;
4. Построить и смоделировать декодирующее устройство, продемонстрировать эффективность его работы;
5. Сформулировать выводы по работе.

6 Преобразование кодов с помощью линейных двоичных динамических систем (ЛДДС)

Двоичные динамические системы (ДДС), интегрированные в некоторую техническую среду приема, хранения, обработки и передачи двоичной информации, при выполнении конкретных функций решают в основном задачи преобразования кодов, элементы которых принадлежат простому полю Галуа $GF(p) = \{0, 1, 2, \dots, p-1\}$, которое при $p = 2$ принимает вид $GF(2) = \{0, 1\}$. Преобразуемые коды могут быть представлены тремя основными способами: в виде вектора, не параметризованного дискретным временем; в виде кодовой последовательности (скалярной или векторной), параметризованной дискретным временем; в виде модулярных многочленов (ММ). Если процесс преобразования кода, поданного на вход ДДС, в код, наблюдаемый на ее выходе, осуществляется с помощью линейной композиции результатов линейных операций умножения и суммирования по модулю два, то такая двоичная динамическая система является линейной (ЛДДС) [1].

Описание ЛДДС, как и случае непрерывных и дискретных динамических систем, может быть задано в форме “вход - выход”, либо в форме “вход - состояние - выход”. Для формирования математических моделей в форме “вход - выход” применяется аппарат передаточных функций (матриц). В его основе лежит алгебраизация отношения “вход - выход”, которое в случае ЛДДС осуществляется с помощью D -преобразования кодовых последовательностей и модулярных многочленов.

Прямым D -преобразованием $F(d)$ двоичной последовательности $f(k)$, где k - дискретное время, выраженное в числе тактов длительности Δt , над простым полем Галуа $GF(p) = \{0, 1, 2, \dots, p-1\}$, где $p = 2$, называется бесконечная сумма

$$F(d) = D \{f(k)\} = \sum_{k=0}^{\infty} f(k)d^k, \quad (30)$$

при условии, что она сходится. Функция $F(d)$ называется D -образом двоичной последовательности $f(k)$.

Обратным D -преобразованием $D^{-1} \{F(d)\}$ D -образа $F(d)$ двоичной последовательности $f(k)$, называется преобразование, позволяющее по D -образу $F(d)$ двоичной последовательности $f(k)$ восстановить исходную последовательность $f(k)$ в силу соотношения

$$D^{-1} \{F(d)\} = f(k). \quad (31)$$

Последовательность $f(k)$ именуется оригиналом D -преобразования.

Если ЛДДС преобразует r -мерную входную ДКП $u(k)$ в m -мерную выходную ДКП $y(k)$, то передаточной матрицей $\Phi(d)$ такой ЛДДС называется матрица, связывающая D -образ $Y(d)$ выходной ДКП $y(k)$ с D -образом входной ДКП $u(k)$ при нулевом начальном состоянии ЛДДС в силу соотношения

$$\Phi(d) = \text{arg} \{Y(d) = \Phi(d)U(d), Y(d), U(d) - \text{fix}\}. \quad (32)$$

Если входная и выходная ДКП скалярные, то передаточная матрица (32) ЛДДС вырождается в передаточную функцию

$$\Phi(d) \triangleq \frac{Y(d)}{U(d)} = \frac{M(d)}{N(d)} = \frac{\sum_{i=0}^l \alpha_i d^i}{\sum_{j=0}^m \beta_j d^j}, \beta_0 = 1, \quad (33)$$

где $M(d)$ и $N(d)$ - модулярные многочлены степеней l и m соответственно.

Процесс преобразования входного кода u , задаваемого ММ $u(x) = u_l x^l + u_{l-1} x^{l-1} + \dots + u_1 x + u_0$, в выходной вектор y , задаваемый модулярным многочленом $y(x) = u_m x^m + u_{m-1} x^{m-1} + \dots + u_m x + u_0$, может быть также описан с помощью передаточной функции вида (33), если будут сконструированы D -образы $U(d)$ и $Y(d)$ модулярных многочленов $u(x)$ и $y(x)$ соответственно.

D -образ модулярного многочлена зависит от того, каким разрядом вперед организована в среде линейных ДДС передача (преобразование) модулярных многочленов. Так, D -образ ММ $F(d) = D\{f(x)\}$ при его передаче младшим разрядом вперед задается выражением

$$F(d) = D\{f(x)\} = f_0 + f_1 d + f_2 d^2 + \dots + f_{n-1} d^{n-1} + f_n d^n. \quad (34)$$

D -образ ММ $F(d) = D\{f(x)\}$ при его передаче старшим разрядом вперед задается выражением

$$F(d) = D\{f(x)\} = \tilde{f}(x^{-1})|_{x^{-1}=d} = f_n + f_{n-1} d + \dots + f_1 d^{n-1} + f_0 d^n; \tilde{f}(x^{-1}) = x^{-n} f(x). \quad (35)$$

Стоит отметить, что в современных устройствах преобразования кодов, заданных с помощью модулярных многочленов, их передача осуществляется старшим разрядом вперед.

D -образ периодической кодовой последовательности $f(k) = f(k+T)$, где T - период, конструируется в силу соотношения

$$F(d) = D\{f(k) = f(k+T)\} = \frac{D\{f(k)\}}{1+d^T} = \frac{\sum_{i=0}^{T-1} f(i) d^i}{1+d^T}. \quad (36)$$

В случае, когда ставится задача конструирования ДДС, осуществляющей умножение модулярных многочленов $a(x)$ и $b(x)$, передаточная функция $\Phi(d)$

будет определяться в силу правила

$$\Phi(d) = \arg \{(a(x) \vee b(x)) \& \deg \Phi(d) = \min \{\deg a(d), \deg b(d)\}\}. \quad (37)$$

В случае, когда ставится задача конструирования ДДС, осуществляющей деление модулярного многочлена $a(x)$ на модулярный многочлен $b(x)$, передаточная функция $\Phi(d)$ будет иметь вид

$$\Phi(d) = \frac{1}{\beta(d)}. \quad (38)$$

Для ЛДДС также существует векторно-матричное представление, имеющее следующий вид:

$$x(k+1) = Ax(k) + Bu(k), \quad x(0), \quad (39)$$

$$y(k) = Cx(k) + Nu(k), \quad (40)$$

где x - n -мерный вектор состояния ЛДДС, A - $(n \times n)$ -матрица состояния ЛДДС, B - $(n \times r)$ -матрица входов ЛДДС, u - r -мерный вектор входных воздействий, $x(0)$ - вектор начального состояния ЛДДС, y - m -мерный вектор выходов ЛДДС, C - $(m \times n)$ -матрица выходов ЛДДС, N - $(m \times r)$ -матрица вход-выход ЛДДС.

Векторно-матричное представление (39)-(40) ЛДДС называется рекуррентным.

Векторно-матричное представление (39)-(40) ЛДДС связано с передаточной функцией ЛДДС в виде (33) следующим соотношением:

$$\Phi(d) = C(d^{-1}I + A)^{-1}B + N. \quad (41)$$

Структура модельного представления ЛДДС, описываемой передаточной функцией вида (33) с единичным свободным членом знаменателя, может быть построена с использованием правила некасающихся контуров метода Мейсона, в соответствии с которым она выразится в форме касающихся (вложенных друг в друга) контуров, передаточные функции которых заданы мультипликативной структурой из постоянного коэффициента α_i и соответствующей степени i переменной d знаменателя передаточной функции так, что их число не превышает m , а число прямых ветвей от входа к выходу этой реализации определяется числом ненулевых элементов числителя передаточной функции с передаточными функциями ветвей $\beta_j d^j$, число которых не превышает $m + 1$. Так, для ЛДДС, описываемых передаточной функцией вида (33) с переменными состояниями $x_i(k)$, можно сформировать структурную схему в каноническом управляемом (рисунок 7) и каноническом наблюдаемом (рисунок 8) базисах, хорошо известных в теории систем автоматического управления.

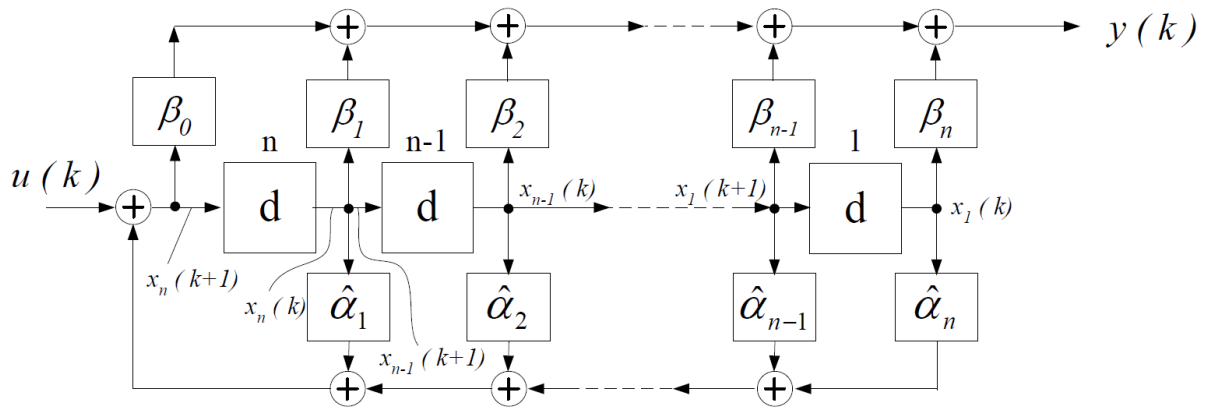


Рис. 7: Структурное представление ЛДДС в каноническом управляемом базисе.

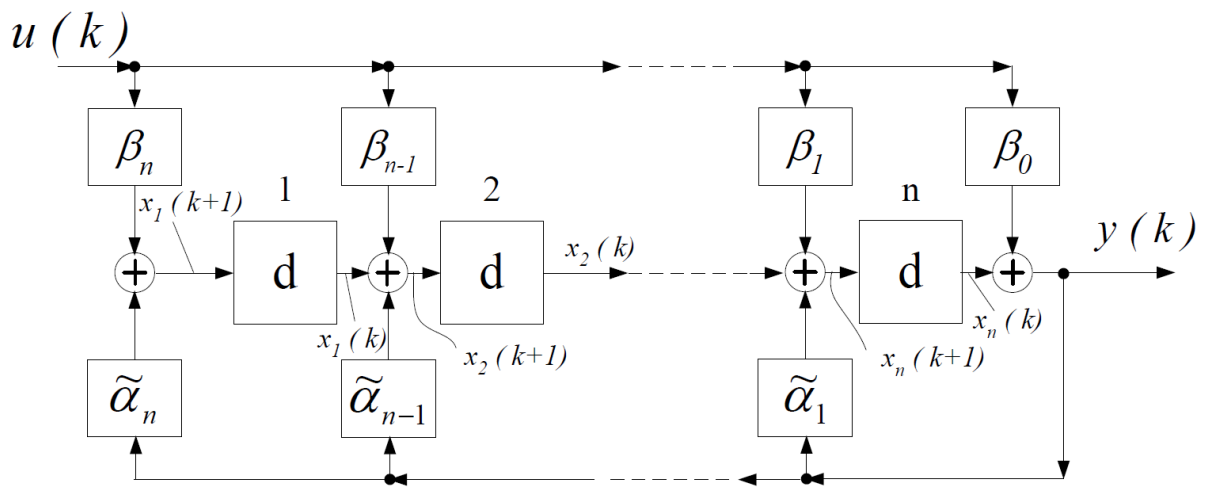


Рис. 8: Структурное представление ЛДДС в каноническом наблюдаемом базисе.

Элементы d модельных представлений, показанных на рисунках 7 и 8, имеют смысл элемента памяти, передаточная функция которого имеет представление $\Phi_{ЭП}(d) = d$ и который является D -триггером.

Пример

Рассмотрим на примере процедуру конструирования передаточной функции и построения структурного представления ЛДДС. Необходимо синтезировать ЛДДС, преобразующая входную единичную последовательность $u(k) = 1(k)$ в периодическую периода $T = 7$, обеспечивающую размещение в регистре хранения информационных разрядов кода Хэмминга (7, 4). В соответствии с расположением информационных разрядов в кодах Хэмминга (7, 4) зададим выходной двоичный код в форме ДКП $y(k) : 1110100|1110100 \dots 1110100$.

Вычислим согласно (36) D -образы входной и выходной последовательно-

стей:

$$U(d) = \frac{1}{1+d}; \quad Y(d) = \frac{1+d+d^2+d^4}{1+d^7}.$$

Далее сформируем передаточную функцию синтезируемой ЛДДС в форме (33):

$$\Phi(d) = \frac{(1+d+d^2+d^4)(1+d)}{1+d^7} = \frac{1+d^3+d^4+d^5}{1+d^7}. \quad (42)$$

Построим структурное представление синтезируемой ЛДДС в каноническом управляемом (см. рисунок 9) и в каноническом наблюдаемом (см. рисунок 10) базисах и разметим на них элементы вектора состояния ЛДДС.

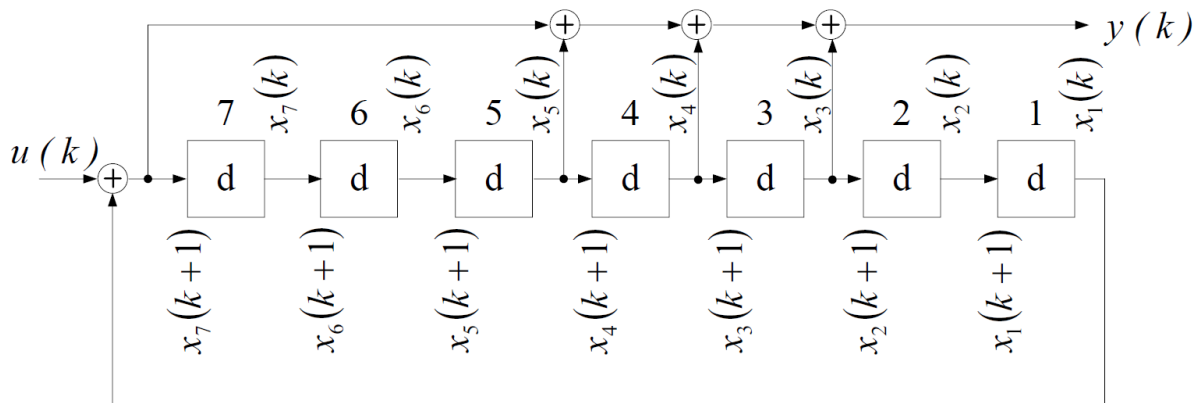


Рис. 9: Представление ЛДДС с передаточной функцией (42) в каноническом управляемом базисе.

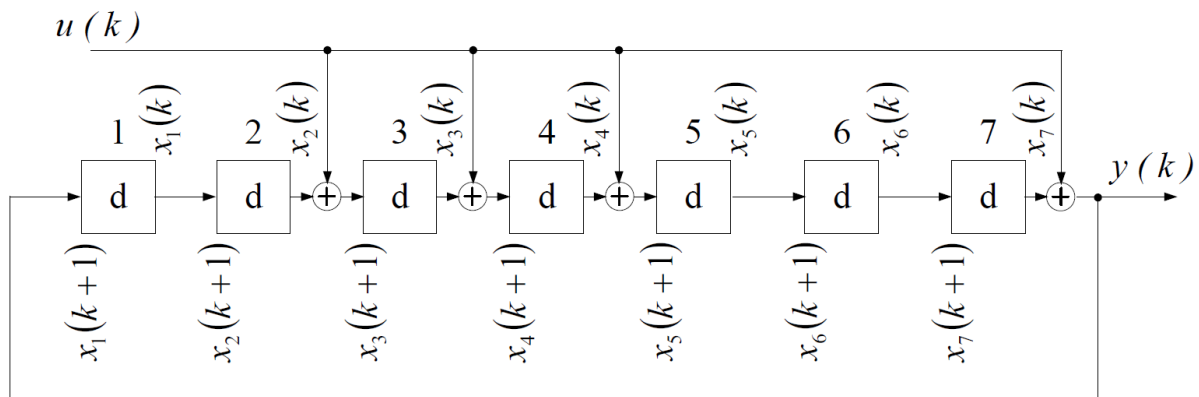


Рис. 10: Представление ЛДДС с передаточной функцией (42) в каноническом наблюдаемом базисе.

Теперь по построенным структурным представлениям можно составить матрицы A , B , C и N векторно-матричного представления (39)-(40) ЛДДС в ка-

ноническом управляемом базисе:

$$A = \begin{bmatrix} 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 \\ 1 & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}, B = \begin{bmatrix} 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 1 \end{bmatrix},$$

$$C = [1 \ 0 \ 1 \ 1 \ 1 \ 0 \ 0], N = [1],$$

и в каноническом наблюдаемом базисе:

$$A = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 & 1 \\ 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 \end{bmatrix}, B = \begin{bmatrix} 1 \\ 0 \\ 1 \\ 1 \\ 1 \\ 0 \\ 0 \end{bmatrix},$$

$$C = [0 \ 0 \ 0 \ 0 \ 0 \ 0 \ 1], N = [1].$$

Путем подстановки полученных матриц в выражение (41) можно убедиться, что векторно-матричное описание ЛДДС сформировано корректно.

В качестве второго примера рассмотрим задачу конструирования ЛДДС, осуществляющей деление произвольной входной двоичной кодовой последовательности на модулярный многочлен $\beta(x) = x^7 + x^3 + x^2 + x + 1$ с учетом передачи входной последовательности старшим разрядом вперед.

Передаточная функция синтезируемой ЛДДС в форме (38) с учетом передачи входной последовательности старшим разрядом вперед может быть найдена следующим образом:

$$\beta(d) = \tilde{\beta}(x^{-1}) \Big|_{x^{-1}=d} = x^{-7} \beta(x) \Big|_{x^{-1}=d} = 1 + d^4 + d^5 + d^6 + d^7;$$

$$\Phi(d) = \frac{1}{\beta(d)} = \frac{1}{1 + d^4 + d^5 + d^6 + d^7}. \quad (43)$$

Построим структурное представление синтезируемой ЛДДС в каноническом управляемом (см. рисунок 11) и в каноническом наблюдаемом (см. рисунок 12) базисах.

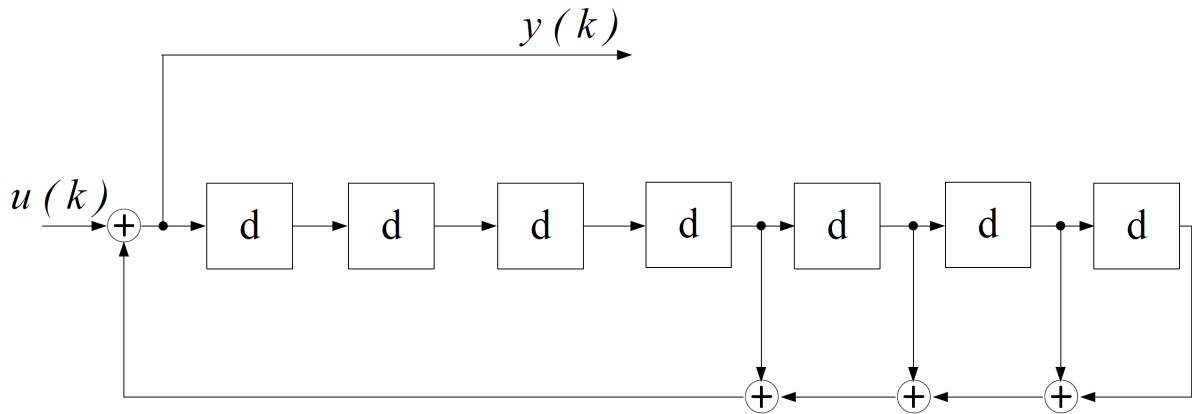


Рис. 11: Представление ЛДДС с передаточной функцией (43) в каноническом управляемом базисе.

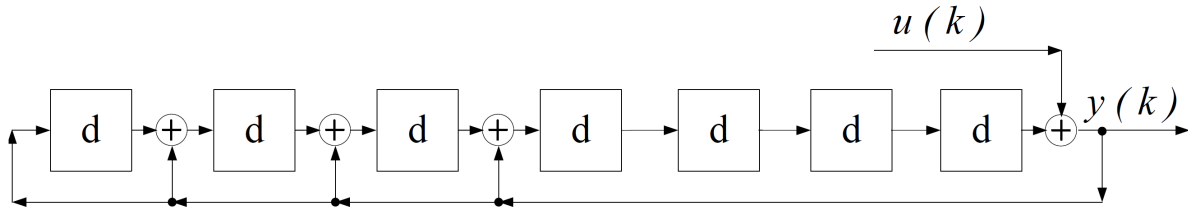


Рис. 12: Представление ЛДДС с передаточной функцией (43) в каноническом наблюдаемом базисе.

Теперь по построенным структурным представлениям можно составить матрицы A , B , C и N векторно-матричного представления (39)-(40) ЛДДС в каноническом управляемом базисе:

$$A = \begin{bmatrix} 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 \\ 1 & 1 & 1 & 1 & 0 & 0 & 0 \end{bmatrix}, B = \begin{bmatrix} 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 1 \end{bmatrix},$$

$$C = [1 \ 1 \ 1 \ 1 \ 0 \ 0 \ 0], N = [1],$$

и в каноническом наблюдаемом базисе:

$$A = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 & 1 \\ 1 & 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 1 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 \end{bmatrix}, B = \begin{bmatrix} 1 \\ 1 \\ 1 \\ 1 \\ 0 \\ 0 \\ 0 \end{bmatrix},$$
$$C = [0 \ 0 \ 0 \ 0 \ 0 \ 0 \ 1], N = [1].$$

Путем подстановки полученных матриц в выражение (41) можно убедиться, что векторно-матричное описание ЛДДС сформировано корректно.

Порядок выполнения работы

1. Составить структурное и векторно-матричное представление ЛДДС, реализующей преобразование входную ДКП $u(k)$ в периодическую ДКП $y(k)$ периода T согласно варианту (см. таблицу 8) с учетом передачи входной ДКП младшим разрядом вперед;
2. Составить структурное и векторно-матричное представление ЛДДС, реализующей деление на фиксированный ММ $\beta(x)$ согласно варианту;
3. Сформулировать выводы по работе.

Варианты задания

Таблица 8: Исходные данные для выполнения задания 5

№	$y(k)$	$u(k)$	$\beta(x)$
1	1010101	101	$x^8 + x^6 + x^4 + 1$
2	1001100	011	$x^8 + x^7 + x^4 + x + 1$
3	1111110	111	$x^9 + x^3 + x^2 + 1$
4	1011100	011	$x^8 + x^5 + x^4 + x^3 + 1$
5	1011000	101	$x^8 + x^6 + x^5 + x + 1$
6	1000111	101	$x^8 + x^5 + x^3 + x^2 + 1$
7	1101101	001	$x^9 + x^2 + x + 1$
8	1110011	011	$x^8 + x^7 + x^6 + x^5 + 1$
9	1100001	111	$x^9 + x^6 + x + 1$
10	1111001	011	$x^8 + x^6 + x^4 + x^2 + x + 1$
11	1011011	111	$x^8 + x^7 + x^3 + x + 1$
12	1001111	101	$x^9 + x^8 + x^5 + x + 1$
13	1111101	111	$x^9 + x^7 + x^6 + 1$
14	1010101	011	$x^8 + x^7 + x^3 + x^2 + 1$
15	1110111	101	$x^8 + x^5 + x^4 + x^3 + 1$

7 ЛДДС в задачах рекуррентного помехозащитного кодирования и декодирования

Рассматриваются алгоритмы синтеза ЛДДС рекуррентного помехозащитного кодирования и декодирования с учетом того факта, что УДММ в КУ и ДКУ с точностью до матриц входа синтезируются однотипно. Разнотипность функционирования КУ и ДКУ состоит в том, что ДКУ функционирует без коммутации структуры, принимая из канала связи входную последовательность в виде кодовой посылки до момента $k = n$, когда фиксируется результат $x^T(n) = E[1]$.

Рекуррентное представление процессов помехозащитного кодопреобразования в фазе кодирования и декодирования использует векторно-матричное описание, параметризованное дискретным временем k этих процессов в форме линейных двоичных динамических систем, опирающиеся на модели “вход - состояние” вида (39)

$$x(k+1) = Ax(k) + Bu(k), x(0) \equiv 0. \quad (44)$$

В зависимости от задачи помехозащитного кодопреобразования $u(k)$ принимает смысл ПЗК $u(k) = a(k)$ при формировании помехозащищенного кода $y(k)$ и смысл принятого из канала связи искаженного кода $f(k) = y(k) + \xi(k)$ так, что $u(k) = f(k)$ в задаче декодирования. Характерной особенностью модельного представления (44) является то, что матрица $A_{\text{ку}}$ состояния устройства деления модулярных многочленов (УДММ) кодирующего устройства и матрица $A_{\text{дку}}$ состояния УДММ декодирующего устройства совпадают так, что выполняется равенство $A_{\text{ку}} = A_{\text{дку}} = A$. Матрица A состояния УДММ КУ и ДКУ задается в одном базисе, при этом чаще всего в сопровождающей характеристический полином (ХП) форме, причем ХП $D(\lambda) = \det(\lambda I + A)$ совпадает с образующим ПЗК модулярным многочленом $g(x)$ так, что выполняется соотношение $D(\lambda) = g(x)|_{x=\lambda}$.

Матрицы входа УДММ устройств кодирования и декодирования не совпадают, поэтому для КУ и ДКУ модель (44) соответственно получает представление

$$x(k+1) = Ax(k) + B_{\text{ку}}u(k); \quad x(k+1) = Ax(k) + B_{\text{дку}}u(k). \quad (45)$$

Если при формировании ПЗК предполагается возможность перехода от их матричного задания, не параметризованного дискретным временем k , с помощью образующей матрицы G и проверочной матрицы H , то необходимо связать их с матрицами модели вида (45). Так, проверочная матрица H представима с помощью системной матрицы отношения «вход - состояния» индекса n пары

матриц $(A, B_{\text{дкУ}})$ УДММ ДКУ (45) в виде

$$H = [A^{n-1}B_{\text{дкУ}} \quad A^{n-2}B_{\text{дкУ}} \quad \dots \quad A^2B_{\text{дкУ}} \quad AB_{\text{дкУ}} \quad B_{\text{дкУ}}]^T. \quad (46)$$

С другой стороны, матрица $B_{\text{дкУ}}$ входа декодирующего рекуррентного устройства (45) удовлетворяет соотношению

$$B_{\text{дкУ}} = (H^n)^T.$$

Если декодируемый ПЗК является кодом с полной блоковой систематикой, то матрица входа рекуррентного декодирующего устройства будет иметь вид

$$B_{\text{дкУ}} = [0_{1 \times (m-1)} \quad 1]^T. \quad (47)$$

Матрица $B_{\text{кУ}}$ ЛДДС УДММ кодирующего устройства (45) с точностью до операции транспонирования совпадает с последней (k -ой) строкой \tilde{G}^k матрицы \tilde{G} , являющейся блоком образующей матрицы G так, что выполняется соотношение

$$B_{\text{кУ}}^T = \tilde{G}^k = K \{g(x) + x^m\} \quad (48)$$

где $K \{(\cdot)\}$ – код модулярного многочлена (\cdot) .

Матрицы $B_{\text{кУ}}$ и $B_{\text{дкУ}}$ входа УДММ кодирующего и декодирующего устройств (45) формирующих и декодирующих ПЗК с полной блоковой систематикой связаны соотношениями

$$B_{\text{кУ}} = A^m B_{\text{дкУ}}, \quad B_{\text{дкУ}} = (A^m)^{-1} B_{\text{кУ}}. \quad (49)$$

КУ функционирует с коммутацией структуры, это вызвано тем, что при формировании (n, k) -ПЗК в течение первых k тактов k -разрядная информационная часть в виде кодовой последовательности одновременно подается в канал связи и на вход устройства деления ММ для вычисления остатка в форме $x^T(k) = K \left\{ \text{rest} \left(\frac{a(x)x^m}{g(x)} \right) \right\}$. По принятии информационной части из источника дискретной информации (ИДИ) вход канала связи переключается с выхода ИДИ на выход УДММ, в котором сформировался остаток. Все обратные связи в УДММ в этот момент разрываются, чем останавливается процесс деления, а УДММ без связей становится m -разрядным регистром сдвига. Все перечисленные коммутации цепей и связей осуществляются специально вводимым в состав КУ устройством коммутации. Таким образом, помехозащитное КУ есть функциональное объединение устройства деления ММ и устройства коммутации. КУ до коммутации его цепей в силу (45) аналитически описывается векторно-матричными соотношениями

$$x(k+1) = Ax(k) + B_{\text{кУ}}u(k); \quad y(k) = Nu(k); \quad N = [1]; \quad u(k) = a(k). \quad (50)$$

После коммутации КУ описывается соотношениями

$$x(k+1) = \bar{A}x(k); \quad x^T(k) = K \left\{ rest \left(\frac{a(x)x^m}{g(x)} \right) \right\}; \quad y(k) = Cx(k), \quad (51)$$

где \bar{A} - нильпотентная матрица с индексом нильпотентности $\nu = m$, $C = [1 \ 0_{1 \times (n-1)}]$, $y(k)$ - формируемый ПЗК, $u(k)$ - входная помехонезащищенная последовательность.

Пример

Требуется синтезировать устройства помехозащитного кодирования и декодирования рекуррентного типа на основе ЛДДС для массива сообщений мощностью $V_{\text{и}} = 120$, т.е. число информационных разрядов ПЗК

$$k = \min \arg \{2^k \geq V_{\text{и}} = 120\} = 7.$$

Передача сообщений должна осуществляться средствами ПЗК с полной блоковой систематикой, обладающим способностью исправлять ошибки первой и второй кратности ($s = 2$) и обнаруживать ошибки четвертой кратности ($r = 4$).

Определим число m проверочных разрядов формируемого ПЗК

$$m = \min \arg \left\{ N_c = 2^m - 1 \geq N_{\xi} = \sum_{i=1}^{s=2} C_{n=k+m}^i = \frac{(k+m)(k+m+1)}{2} \right\} = 7.$$

Тогда формат ПЗК $(n, k) = (14, 7)$, степень образующего ММ ПЗК равна количеству проверочных разрядов, выберем в качестве образующего неприводимый ММ $g(x) = x^7 + x^3 + x^2 + x + 1$. Вычислим D -образ ММ $g(x)$ в форме

$$g(d) = \tilde{g}(x^{-1})|_{x^{-1}=d} = x^{-m}g(x)|_{x^{-1}=d} = 1 + d^4 + d^5 + d^6 + d^7.$$

Передаточная функция УДММ будет иметь вид

$$\Phi(d) = \frac{1}{g(d)} = \frac{1}{1 + d^4 + d^5 + d^6 + d^7}.$$

Пользуясь правилом Мейсона некасающихся контуров, построим структурные реализации передаточной функции реализуемого УДММ на элементах памяти (ЭП) с передаточной функцией $\Phi_{\text{ЭП}}(d) = d$ в двух канонических базисах базисах и разметим их переменными состояниями $x_i(k+1)$ на входах и $x_i(k)$ на выходах соответствующих ЭП (см. рисунки 11 и 12 в примере к заданию 6).

Спишем со схемы структурной реализации (рисунок 11) матрицу A в каноническом управляемом базисе:

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

Спишем со схемы структурной реализации (рисунок 12) матрицу A в канонических наблюдаемом базисе:

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

Сформируем матрицу входа $B_{\text{ку}}$ УДММ кодирующего устройства (45), воспользовавшись представлением (48), для которого получим

$$B_{\text{ку}} = K \{x^m + g(x)\}|_{m=7} = K \{x^3 + x^2 + x + 1\} = [0 \ 0 \ 0 \ 1 \ 1 \ 1 \ 1].$$

Объединим полученную матрицу входа УДММ КУ со структурными реализациями в канонических базисах, заметив при этом, что матрица $B_{\text{ку}}$ уже встроена в УДММ КУ в каноническом наблюдаемом базисе (рисунок 12) и при этом сохраняется реализуемость передаточной функции УДММ КУ, а введение ее в структурную схему в каноническом управляемом базисе (рисунок 11) явно приводит к нереализуемости этой передаточной функции.

В результате в качестве окончательного варианта УДММ КУ принимается версия, представленная на рисунке 12, совмещающая в себе реализацию передаточной функции $\Phi(d)$, матриц A и $B_{\text{ку}}$.

Сформируем теперь матрицу A для случая, когда все обратные связи струк-

турной реализации $\Phi(d)$ разорваны, для которой получим

$$\bar{A} = \begin{bmatrix} 0_{6 \times 1} & I_{6 \times 6} \\ 0_{1 \times 1} & 0_{1 \times 6} \end{bmatrix} = \begin{bmatrix} 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}.$$

Объединим сформированный УДММ КУ, представленный на рисунке 12, с устройством коммутации (УК), осуществляющим переключение КС с выхода ИДИ на выход УДММ с помощью ключей К1, К2 и разрыв обратных связей в УДММ ключом К3, останавливая тем самым процесс деления и превращая УДММ в регистр сдвига для вывода из него остатка от деления в КС, чем завершим формирование рекуррентного КУ, представленного на рисунке 13 (реализация аналогичного КУ в среде Matlab Simulink приведена на рисунке 5).

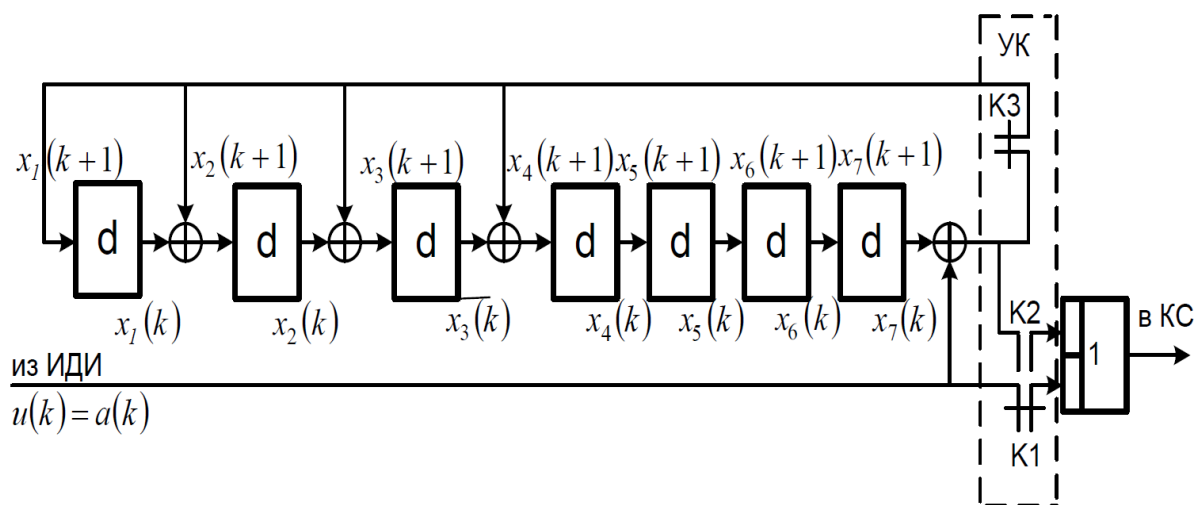


Рис. 13: Структурная реализация синтезируемого КУ в каноническом наблюдаемом базисе.

На конкретном примере проверим правильность функционирования устройства рекуррентного кодирования, задаваемого парой векторно-матричных описаний (50) и (51) со сформированными матрицами $(A, B_{\text{КУ}}, N)$ и (\bar{A}, C) . Для этой цели предположим, что передаваемые сообщения есть команды телеуправления типа «включить – отключить», а конкретная команда такова «включить 37-й насос», которой соответствует семиразрядный ПНЗК $a = [1 \ 0 \ 0 \ 1 \ 0 \ 1 \ 1]$, в котором первые 6 старших разрядов задают число «37», а последний – команду «включить». Этот код определяет входную последовательность КУ $u(k)$:

$u(0) = 1, u(1) = 0, u(2) = 0, u(3) = 1, u(4) = 0, u(5) = 1, u(6) = 1$ и в силу полной блоковой систематики формируемого ПЗК информационную часть кода так, что для него можно записать

$$y = [a|z] = [1 \ 0 \ 0 \ 1 \ 0 \ 1 \ 1 \ y_7 \ y_6 \ y_5 \ y_4 \ y_3 \ y_2 \ y_1].$$

Вычисление остатка с помощью рекуррентной процедуры (50) сведем в таблицу 9.

Таблица 9: Вычисление остатка с помощью рекуррентной процедуры (50)

k	0	1	2	3	4	5	6	7	8	9
$u(k)$	1	0	0	1	0	1	1	0	0	0
$x(k)$	0	0	0	0	1	1	1	0		
	0	0	0	1	1	1	0	0		
	0	0	1	1	1	0	0	0		
	0	1	1	1	0	0	0	0		
	0	1	1	1	1	0	0	1		
	0	1	1	0	1	0	1	0		
	0	1	0	0	1	1	0	0		

Из таблицы 9 видно, что на седьмом такте деления в УДММ КУ сформировался остаток, который через замкнутый на седьмом такте ключ К2 вслед за информационными разрядами будет передан в КС в составе ПЗК, который примет вид $y = [a|z] = [1 \ 0 \ 0 \ 1 \ 0 \ 1 \ 1 \ 0 \ 0 \ 0 \ 0 \ 1 \ 0 \ 0]$.

Далее приступим к синтезу ДКУ. Сформируем матрицу $B_{\text{дку}}$ входа УДММ декодирующего устройства с помощью соотношения (47), в результате получим $B_{\text{дку}} = [0_{1 \times (m-1)} \ 1]^T \Big|_{m=7} = [0 \ 0 \ 0 \ 0 \ 0 \ 0 \ 1]^T$.

Объединим полученную матрицу $B_{\text{дку}}$ входа рекуррентного ДКУ со структурной реализацией передаточной функции в каноническом наблюдаемом базисе (рисунок 12), в результате чего сформируем ДКУ в окончательном виде, представленном на рисунке 14 (реализация аналогичного КУ в среде Matlab Simulink приведена на рисунке 6).

С помощью правила Мейсона сформируем передаточную матрицу–столбец ДКУ $\Phi(d) = g^{-1}(d) \text{col}(\Phi_j(d); j = \overline{1, m})$, для которой получим

$$\Phi(d) = \frac{[d^7 \ d^6 \ d^5 \ d^4 \ d^3(1+d^4) \ d^2(1+d^4+d^5) \ d(1+d^4+d^5+d^6)]}{1+d^4+d^5+d^6+d^7}.$$

Чтобы удостовериться в корректности работы синтезированного ДКУ, указим кодовую комбинацию, полученную ранее на выходе КУ, помехой первой

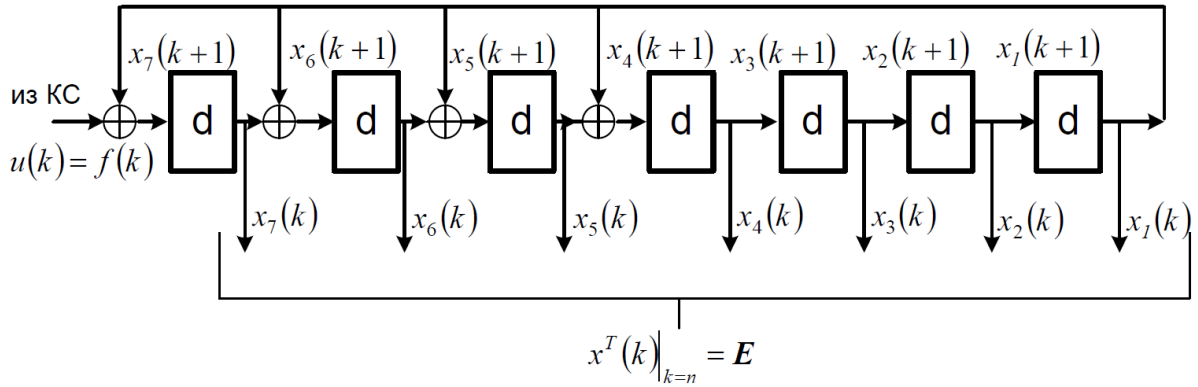


Рис. 14: Структурная реализация синтезируемого ДКУ в каноническом наблюдаемом базисе.

кратности, вносящей ошибку в пятый разряд кодовой комбинации

$$\xi(k) = [0 \ 0 \ 0 \ 0 \ 0 \ 0 \ 0 \ 0 \ 0 \ 0 \ 1 \ 0 \ 0 \ 0 \ 0].$$

Тогда искаженная кодовая последовательность, поступающая на вход ДКУ, примет вид

$$u(k) = [1 \ 0 \ 0 \ 1 \ 0 \ 1 \ 1 \ 0 \ 0 \ 1 \ 0 \ 1 \ 0 \ 0].$$

Вычисление синдрома ошибки с помощью рекуррентной процедуры сведем в таблицу 10.

Таблица 10: Вычисление синдрома ошибки с помощью рекуррентной процедуры в ДКУ

k	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14
u(k)	1	0	0	1	0	1	1	0	0	1	0	1	0	0	0
x(k)	0	0	0	0	0	0	0	1	0	0	1	1	0	0	0
	0	0	0	0	0	0	1	0	0	1	1	0	0	0	0
	0	0	0	0	0	1	0	0	1	1	0	0	0	0	1
	0	0	0	0	1	0	0	1	1	0	0	0	0	1	0
	0	0	0	1	0	0	1	0	0	0	1	1	1	0	0
	0	0	1	0	0	1	0	1	0	1	0	0	0	0	0
	0	1	0	0	1	0	1	1	1	0	1	1	0	0	0

Таблица 10 позволяет записать для синдрома ошибки

$$E = x^T(n)|_{n=14} = [0 \ 0 \ 1 \ 0 \ 0 \ 0 \ 0],$$

который является синдромом однократной ошибки в пятом разряде. Действительно, если воспользоваться представлением проверочной матрицы H в форме

(46), то для нее получим

$$H = \begin{bmatrix} (A^{n-1}B_{\text{дкУ}})^T \\ \vdots \\ (A^m B_{\text{дкУ}})^T \\ (A^{m-1}B_{\text{дкУ}})^T \\ \vdots \\ (AB_{\text{дкУ}})^T \\ B_{\text{дкУ}}^T \end{bmatrix} = \begin{bmatrix} 1 & 1 & 1 & 1 & 0 & 0 & 0 & 1 & 0 & \mathbf{0} & 0 & 0 & 0 & 0 \\ 1 & 1 & 1 & 1 & 1 & 0 & 0 & 0 & 1 & \mathbf{0} & 0 & 0 & 0 & 0 \\ 0 & 1 & 1 & 1 & 1 & 1 & 0 & 0 & 0 & \mathbf{1} & 0 & 0 & 0 & 0 \\ 1 & 0 & 1 & 1 & 1 & 1 & 1 & 0 & 0 & \mathbf{0} & 1 & 0 & 0 & 0 \\ 1 & 0 & 1 & 0 & 1 & 1 & 1 & 0 & 0 & \mathbf{0} & 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 1 & 1 & 0 & 0 & \mathbf{0} & 0 & 0 & 1 & 0 \\ 1 & 1 & 1 & 0 & 0 & 0 & 1 & 0 & 0 & \mathbf{0} & 0 & 0 & 0 & 1 \end{bmatrix}.$$

Нетрудно убедиться, что пятый справа столбец матрицы H содержит полученный синдром ошибки в пятом разряде, что позволяет сделать вывод о корректности работы синтезированного ДКУ и, с учетом корректно синтезированного КУ, констатировать успешное решение задачи рекуррентного кодирования и декодирования в рамках аппарата ЛДДС.

Порядок выполнения работы

1. Сформировать формат (n, k) -ПЗК на основании мощности массива сообщений и требований к корректирующей способности кода (см. таблицу 11);
2. Выбрать образующий ММ;
3. Сформировать передаточную функцию УДММ и ее структурные реализации;
4. Сформировать векторно-матричное описание КУ и проверить правильность его функционирования;
5. Сформировать векторно-матричное описание ДКУ и проверить правильность его функционирования;
6. Сформулировать выводы по работе.

Варианты задания

Таблица 11: Исходные данные для выполнения задания 6

№	$V_{\text{н}}$	s	r
1	100	2	4
2	120	2	5
3	15	3	6
4	10	3	7
5	15	2	4
6	10	2	5
7	2	3	6
8	2	3	7
9	1000	2	4
10	900	2	5

8 Синтез динамических наблюдающих устройств на базе концепции подобия

Концепция подобия в теории динамических систем над бесконечными полями получила в последнее время заметное распространение при решении широкого круга задач управления. В рамках векторно-матричного формализма метода пространства состояний в не параметризованной временем форме концепция подобия сводится к выполнению соотношения [1]

$$\eta = T\xi, \quad (52)$$

где χ - n -мерный вектор состояния некоторого эталонного (в рамках задачи наблюдения вектора состояния - оцениваемого) динамического процесса, η - m -мерный вектор состояния конструируемой динамической среды (в рамках задачи наблюдения вектора состояния - вектор состояния наблюдающего устройства), T - $(m \times n)$ -матрица преобразования подобия (в общем случае - особого).

В параметризованном временем виде соотношение (52) достигается в асимптотике так, что

$$\eta(\tau) = T\xi(\tau) - \theta(\tau), \quad (53)$$

где переменная τ принимает смысл непрерывного либо дискретного времени в зависимости от постановки задачи, а θ - вектор невязки выполнения векторно-матричного подобия, причем

$$\lim_{\tau \rightarrow \infty} \theta(\tau) = 0, \quad \forall \chi(0), \eta(0). \quad (54)$$

Пусть ЛДДС, состояние которой подлежит наблюдению, имеет векторно-матричное описание вида

$$\chi(k+1) = A\chi(k) + Bu(k), \quad \chi(0) = \chi_0, \quad \xi(k) = C\chi(k), \quad (55)$$

где χ - n -мерный вектор состояния ЛДДС, u - r -мерный вектор входной кодовой последовательности, ξ - l -мерный вектор выходной кодовой последовательности, A - $(n \times n)$ -матрица состояния ЛДДС, B - $(n \times r)$ -матрица входов ЛДДС, C - $(l \times n)$ -матрица выходов ЛДДС. Элементы всех перечисленных векторов и матриц принадлежат двоичному простому полю Галуа $GF(2)$.

Двоичное динамическое наблюдающее устройство (ДНУ), использующее всю доступную для непосредственного измерения информацию об ЛДДС (55) в виде входной кодовой последовательности $u(k)$ и выходной кодовой последовательности $y(k)$, строится в форме

$$z(k+1) = \Gamma z(k) + L\xi(k) + Gu(k), \quad (56)$$

где z - m -мерный вектор состояния ДНУ, Γ - $(m \times m)$ -матрица динамики процесса наблюдения, при этом пары матриц (Γ, L) и (Γ, G) должны обладать свойством полной управляемости.

Задачу наблюдения вектора $\chi(k)$ состояния системы (55) в среде ДНУ (56) сформулируем в форме (53), записываемой в виде

$$z(k) = T\chi(k) + \theta(k), \quad \forall k, \quad (57)$$

где T - матрица преобразования подобия (в общем случае - обособого). Уравнение (54) позволяет построить модель процесса наблюдения по вектору невязки наблюдения, которое принимает вид

$$\theta(k+1) = T\chi(k+1) + z(k+1). \quad (58)$$

Для того, чтобы ДНУ вида (56) корректно выполняло свою функцию, необходимо, чтобы матрицы T , L и G удовлетворяли следующим матричным соотношениям:

$$\Gamma T + TA = LC, \quad G = TB. \quad (59)$$

В этом случае процесс по вектору невязки наблюдения $\theta(k)$ описывается рекуррентным векторно-матричным уравнением

$$\theta(k+1) = \Gamma\theta(k), \quad \theta(0) = T\chi(0) + z(0). \quad (60)$$

Модель процесса двоичного динамического наблюдения в форме (60) позволяет сформулировать требования к матричным компонентам наблюдаемой ЛДДС (55) и ДНУ (56), которые позволят обеспечить решение всех возможных задач наблюдения. Если ставится задача наблюдения вектора $\chi(k)$ текущего состояния ЛДДС (55), то следует воспользоваться явным (показательным) решением уравнения (60), записываемым в форме

$$\theta(k) = \Gamma^k\theta(0); \quad \theta(0) = T\chi(0) + z(0). \quad (61)$$

Следует заметить, что при нормальном использовании ДНУ его состояние при запуске обнуляется так, что $z(0) = 0$. С учетом этого обстоятельства (61) принимает вид

$$\theta(k) = \Gamma^k T\chi(0). \quad (62)$$

В свою очередь подстановка (56) в (54) дает

$$z(k) = T\chi(k) + \Gamma^k\theta(0). \quad (63)$$

Потребуем от матрицы Γ состояния ДНУ обладания свойством нильпотентности с индексом ν , тогда при $k \geq \nu$ устанавливается равенство

$$z(k) = T\chi(k), \quad k \geq \nu. \quad (64)$$

Таким образом, вектор $z(k)$ состояния ДНУ с точностью до матрицы преобразования подобия T задает текущее состояние вектора $\chi(k)$ наблюдаемой ДДС (55). Заметим, что подобие (64) можно преобразовать в тождество, если в матричном уравнении Сильвестра (59) положить $T = I$, где I – единичная матрица, и решить уравнение относительно матрицы L .

Пример

Рассмотрим процедуру синтеза двоичного ДНУ на примере. Пусть требуется синтезировать ДНУ для наблюдения вектора состояния ЛДДС с передаточной функцией

$$\Phi(d) = \frac{d^3 + d^2 + d + 1}{d^7 + d^3 + d^2 + d + 1},$$

матрицы векторно-матричного описания которой имеют вид

$$A = \begin{bmatrix} 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 \\ 1 & 1 & 1 & 1 & 0 & 0 & 0 \end{bmatrix}, \quad B = \begin{bmatrix} 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 1 \end{bmatrix},$$

$$C = [1 \ 1 \ 1 \ 1 \ 0 \ 0 \ 0], \quad H = [0].$$

С целью решения поставленной задачи в соответствии с (62) и (64) выберем в качестве модели ДНУ регистр сдвига седьмого порядка, матрица Γ векторно-матричного описания которого будет иметь вид

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

Будем решать поставленную задачу, руководствуясь соотношением (64). С этой целью выберем в качестве матрицы подобия T единичную матрицу $T = I$ соответствующей размерности. Тогда решение уравнения Сильвестра вида (59)

относительно матрицы L и последующее вычисление матрицы G дает следующий результат:

$$L = \begin{bmatrix} 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 1 \end{bmatrix}, \quad G = \begin{bmatrix} 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 1 \end{bmatrix}.$$

С использованием полученных результатов структурно-функциональная схема процесса двоичного динамического наблюдения вектора состояния заданной ДДС примет вид, как показано на рисунке 15, а пример ее реализации в Matlab Simulink представлен на рисунке 16.

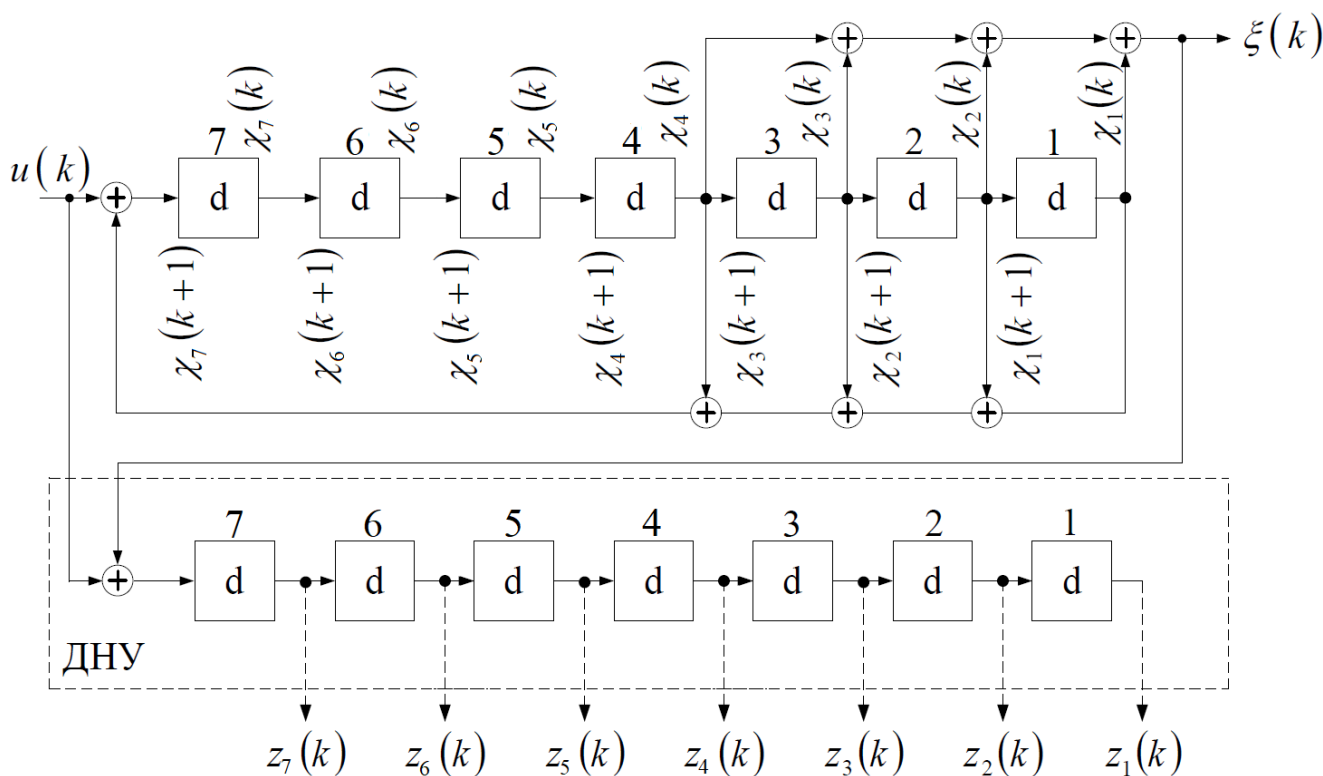


Рис. 15: Структурная реализация синтезированного ДНУ.

В силу того, что выбранная матрица Γ имеет индекс нильпотентности $\nu = 7$, вектор состояния ДНУ $z(k)$ при $k \geq 7$ будет совпадать с вектором состояния χ исходной ЛДДС. Проиллюстрируем данный факт в предположении, что входная последовательность имеет вид $u(k) = [1 \ 0 \ 0 \ 1 \ 0 \ 1 \ 0]$, а вектор начального состояния исходной ЛДДС имеет вид $\chi(0) = [1 \ 1 \ 0 \ 1 \ 0 \ 1 \ 0]^T$.

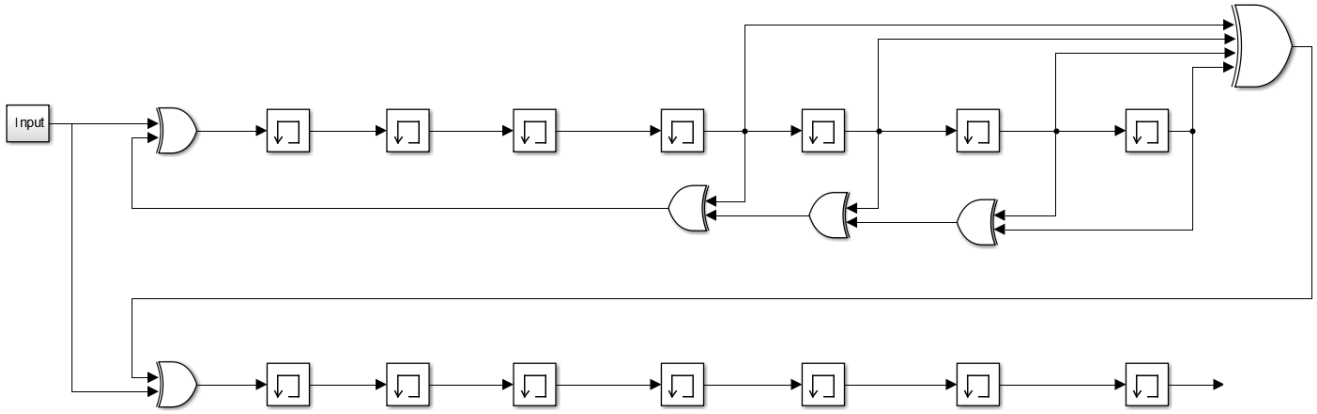


Рис. 16: Структурная реализация синтезированного ДНУ.

Тогда функционирование синтезированного ДНУ может быть описано таблицей 12.

Таблица 12: Таблица функционирования ДНУ

k	0	1	2	3	4	5	6	7	8	9	10	11
k	0	1	2	3	4	5	6	7	8	9	10	11
χ_7	0	1	0	1	0	1	1	1	0	0	0	0
χ_6	1	0	1	0	1	1	1	0	0	0	0	0
χ_5	0	1	0	1	1	1	0	0	0	0	0	0
χ_4	1	0	1	1	1	0	0	0	0	0	0	1
χ_3	0	1	1	1	0	0	0	0	0	0	1	0
χ_2	1	1	1	0	0	0	0	0	0	1	0	1
χ_1	1	1	0	0	0	0	0	0	1	0	1	1
z_7	0	0	0	0	0	0	0	1	0	0	0	0
z_6	0	0	0	0	0	0	1	0	0	0	0	0
z_5	0	0	0	0	0	1	0	0	0	0	0	0
z_4	0	0	0	0	1	0	0	0	0	0	0	1
z_3	0	0	0	1	0	0	0	0	0	0	1	0
z_2	0	0	1	0	0	0	0	0	0	1	0	1
z_1	0	1	0	0	0	0	0	0	1	0	1	1

Из таблицы 12 видно, что начиная с седьмого такта вектор состояния $z(k)$ синтезированного ДНУ повторяет вектор состояния $\chi(k)$ наблюдаемой ДДС.

Порядок выполнения работы

1. Получить векторно-матричное описание наблюдаемой ЛДДС по заданной передаточной функции.
2. Получить матрицы ДНУ, составив матрицу подобия и решив уравнение Сильвестра.
3. Осуществить моделирование синтезированного ДНУ.
4. Проанализировать работу синтезированного ДНУ и показать, что оно работает корректно.
5. Сделать выводы по работе.

Варианты задания

Таблица 13: Исходные данные для выполнения задания 2

№	$\Phi(d)$	$\chi(0)^T$
1	$\frac{d^6+d^2+1}{d^7+d^6+d^2+1}$	$[1 \ 0 \ 0 \ 1 \ 0 \ 1 \ 1]$
2	$\frac{d^6+d^5+1}{d^8+d^6+d^5+1}$	$[1 \ 0 \ 0 \ 1 \ 0 \ 1 \ 1 \ 0]$
3	$\frac{d^6+d^3+1}{d^7+d^6+d^5+d^2+1}$	$[1 \ 1 \ 1 \ 1 \ 0 \ 0 \ 0]$
4	$\frac{d^4+d^3+1}{d^7+d^5+d^4+d^2+1}$	$[0 \ 0 \ 0 \ 1 \ 1 \ 1 \ 1]$
5	$\frac{d^5+d+1}{d^8+d^6+d^5+d^2+1}$	$[1 \ 1 \ 0 \ 0 \ 0 \ 0 \ 1 \ 1]$
6	$\frac{d^5+d^4+d^3+1}{d^8+d^7+d^2+d+1}$	$[1 \ 1 \ 1 \ 1 \ 0 \ 0 \ 1 \ 0]$
7	$\frac{d^6+d^3+d+1}{d^8+d^6+d^3+d+1}$	$[0 \ 0 \ 1 \ 1 \ 0 \ 1 \ 0 \ 0]$
8	$\frac{d^2+d+1}{d^8+d^4+d^2+1}$	$[1 \ 0 \ 1 \ 1 \ 0 \ 1 \ 0 \ 1]$
9	$\frac{d^3+d+1}{d^7+d^5+d^3+d+1}$	$[1 \ 1 \ 0 \ 0 \ 0 \ 0 \ 1]$
10	$\frac{d+1}{d^8+1}$	$[1 \ 0 \ 1 \ 0 \ 0 \ 0 \ 0 \ 0]$
11	$\frac{d^5+d^4+d+1}{d^7+d^2+d+1}$	$[1 \ 0 \ 1 \ 1 \ 0 \ 0 \ 1]$
12	$\frac{d^4+d^2+1}{d^8+d^7+d+1}$	$[0 \ 1 \ 0 \ 1 \ 1 \ 0 \ 1 \ 0]$
13	$\frac{d^3+d^2+d+1}{d^8+d^3+d^2+d+1}$	$[1 \ 1 \ 0 \ 0 \ 0 \ 1 \ 1 \ 1]$
14	$\frac{d+1}{d^8+d^5+d^4+d+1}$	$[0 \ 0 \ 0 \ 0 \ 1 \ 1 \ 1 \ 1]$
15	$\frac{d^6+1}{d^7+d^6+d^5+d^4+d+1}$	$[0 \ 0 \ 0 \ 0 \ 1 \ 1 \ 0]$
16	$\frac{d^4+d+1}{d^8+d^2+d+1}$	$[1 \ 1 \ 1 \ 0 \ 0 \ 0 \ 1 \ 1]$
17	$\frac{d^5+d^3+1}{d^7+d^6+d+1}$	$[1 \ 0 \ 1 \ 1 \ 0 \ 0 \ 0]$

Таблица 13: Исходные данные для выполнения задания 2

№	$\Phi(d)$	$\chi(0)^T$
18	$\frac{d^3+d^2+d+1}{d^7+d^4+d^3+d^2+1}$	$[1 \ 0 \ 0 \ 0 \ 0 \ 1 \ 0]$
19	$\frac{d^5+d^4+d+1}{d^8+d^5+d^4+d^3+d+1}$	$[1 \ 1 \ 0 \ 0 \ 0 \ 1 \ 1 \ 1]$
20	$\frac{d^4+1}{d^7+d^6+d^2+d+1}$	$[1 \ 1 \ 0 \ 1 \ 1 \ 1 \ 1]$

9 Синтез ДДС в логике произвольных линейных триггеров

В схемотехнике существует два типа линейных триггеров [1]

- линейный синхронный D -триггер с передаточной функцией

$$\Phi_D = d;$$

- линейный асинхронный T -триггер с передаточной функцией

$$\Phi_D = \frac{d}{1+d}.$$

Векторы состояний ДДС, имеющих D - и T -триггерную реализацию, связаны отношением подобия

$$x_T(k) = Mx_D(k), \quad \forall k. \quad (65)$$

Пусть в результате синтеза ДДС, решающей задачу преобразования входной последовательности $u(k)$ в выходную последовательность $y(k)$, получена D -триггерная реализация системы, имеющая векторно-матричное представление

$$x_D(k+1) = A_D x_D(k) + B_D u(k), \quad y_D(k) = C_D x_D(k) + N u(k). \quad (66)$$

Требуется, опираясь на условие векторно-матричного подобия (65), сформировать T -триггерную реализацию синтезированной ДДС в виде

$$x_T(k+1) = A_T x_T(k) + B_T u(k), \quad y_T(k) = C_T x_T(k) + N u(k), \quad (67)$$

которая бы осуществляла точно такое же кодопреобразование, что и ДДС (67).

Матричные компоненты векторно-матричных представлений D -триггерной и T -триггерной реализаций ДДС, решающих одну и ту же задачу кодопреобразования входной последовательности $u(k)$ в выходную последовательность $y(k)$, связаны соотношениями подобия вида

$$A_T = M A_D M^{-1}, \quad (68)$$

$$B_T = M B_D, \quad C_T = C_D M^{-1}. \quad (69)$$

Матричное условие подобия (68), переписанное в виде $MA_D = A_TM$, представимо в виде неоднородного матричного уравнения Сильвестра

$$MA_D + \bar{A}_TM = \bar{B}_TL_D, \quad (70)$$

где $\dim A_D = \dim \bar{A}_T$, (A_D, L_D) - полностью наблюдаемая пара матриц, (\bar{A}_T, \bar{B}_T) - полностью управляемая пара матриц, алгебраические спектры собственных значений матриц A_D и \bar{A}_T не пересекаются, то есть $\sigma\{A_D\} \cap \sigma\{\bar{A}_T\} = \emptyset$, а размерности матриц \bar{B}_T и L_D согласованы.

Доказательство данного утверждения строится на представлении матрицы A_T в форме

$$A_T = \bar{A}_T + \bar{B}_T\bar{N}_T, \quad (71)$$

где матрица \bar{N}_T допускает представление в виде

$$\bar{N}_T = L_DM^{-1}. \quad (72)$$

Выражение (72) допускает эквивалентное представление вида

$$L_D = \tilde{N}_TM. \quad (73)$$

Подстановка выражения (73) в уравнение Сильвестра (70) с учетом выражения (71) приводит к выражению (68).

Пример

Рассмотрим в качестве примера задачу построения для декодирующего устройства циклического кода с образующим многочленом $g(x) = x^3 + x + 1$ модельное представление ДДС в логике линейных T -триггеров. Для этого сформируем модельное представление «вход-состояние-выход» декодирующего устройства с матричными компонентами

$$A_D = \begin{bmatrix} 0 & 1 & 0 \\ 1 & 0 & 1 \\ 1 & 0 & 0 \end{bmatrix}, \quad B_D = \begin{bmatrix} 0 \\ 1 \\ 1 \end{bmatrix}, \quad C_D = [1 \ 0 \ 0], \quad H = [1].$$

Далее назначим произвольные матрицы \bar{A}_T , \bar{B}_T и L_D :

$$\bar{A}_T = \begin{bmatrix} 1 & 1 & 0 \\ 0 & 1 & 1 \\ 0 & 0 & 1 \end{bmatrix}, \quad \bar{B}_T = \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix}, \quad L_D = [1 \ 0 \ 0].$$

Далее решим матричное уравнение Сильвестра (70) относительно матрицы подобия M и осуществим ее обращение:

$$M = \begin{bmatrix} 0 & 1 & 1 \\ 0 & 1 & 0 \\ 1 & 1 & 1 \end{bmatrix}, \quad M^{-1} = \begin{bmatrix} 1 & 0 & 1 \\ 0 & 1 & 0 \\ 1 & 1 & 0 \end{bmatrix}.$$

С помощью соотношений (68) и (69) вычислим матричные компоненты T -триггерной реализации ДДС:

$$A_T = \begin{bmatrix} 1 & 1 & 0 \\ 0 & 1 & 1 \\ 1 & 0 & 0 \end{bmatrix}, \quad B_T = \begin{bmatrix} 0 \\ 1 \\ 0 \end{bmatrix}, \quad C_T = [1 \ 0 \ 1].$$

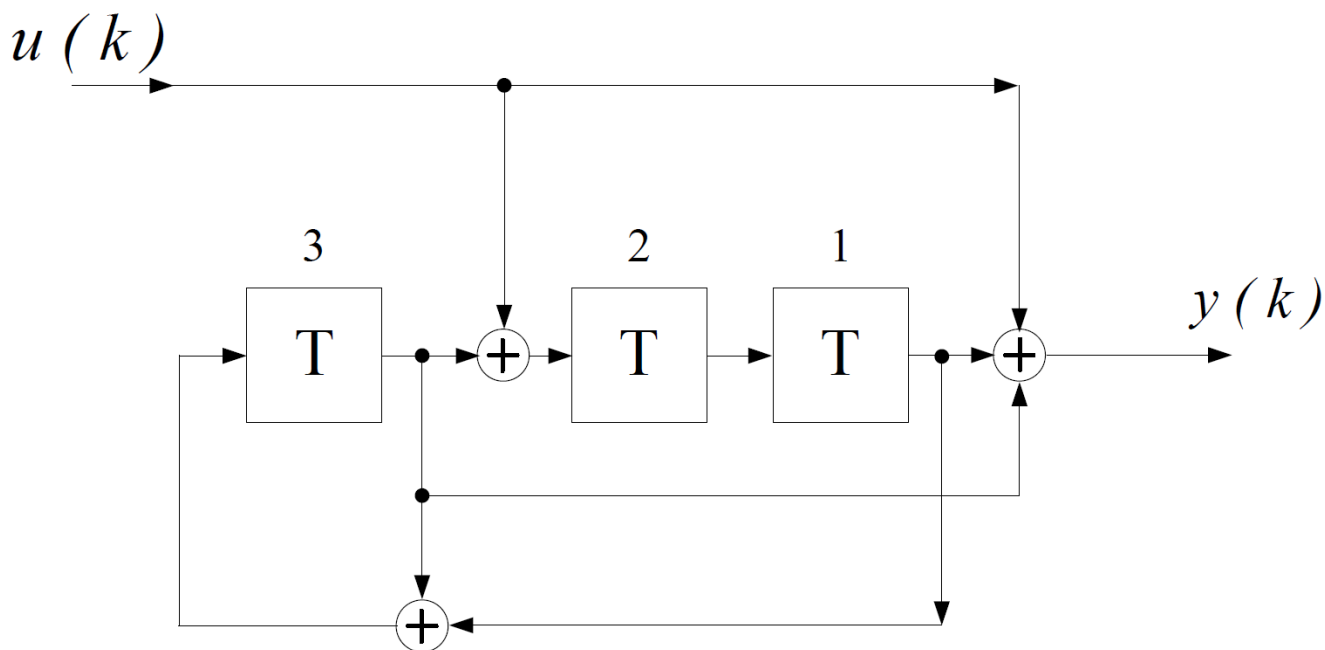


Рис. 17: Структурная реализация синтезированной T -триггерной реализации ДДС.

Структурное представление векторно-матричного описания искомой T -триггерной реализации ДДС приведено на рисунке 17.

Порядок выполнения работы

1. Сформировать векторно-матричное D -триггерное представление синтезируемой ДДС.

2. Получить векторно-матричное T -триггерного представление синтезированной ДДС.
3. Составить структурные схемы D - и T -триггерных реализаций синтезированной ДДС.
4. Осуществить моделирование D - и T -триггерных реализаций синтезированной ДДС и продемонстрировать их эквивалентность.
5. Сделать выводы по работе.

Варианты задания

Таблица 14: Исходные данные для выполнения задания 3

№	$g(x)$
1	$x^8 + x^7 + x^2 + x + 1$
2	$x^8 + x^5 + x^3 + x^2 + 1$
3	$x^8 + x^7 + x^4 + x^3 + x^2 + x + 1$
4	$x^8 + x^6 + x^3 + x^2 + 1$
5	$x^8 + x^7 + x^5 + x^4 + x^3 + x^2 + 1$
6	$x^8 + x^7 + x^6 + x + 1$
7	$x^8 + x^7 + x^6 + x^3 + x^2 + 1$
8	$x^8 + x^7 + x^6 + x^4 + x^2 + x + 1$
9	$x^8 + x^7 + x^6 + x^4 + x^3 + x^2 + 1$
10	$x^8 + x^7 + x^6 + x^5 + x^2 + x + 1$
11	$x^8 + x^7 + x^6 + x^5 + x^4 + x + 1$
12	$x^8 + x^7 + x^6 + x^5 + x^4 + x^2 + 1$
13	$x^8 + x^7 + x^6 + x^5 + x^4 + x^3 + 1$
14	$x^8 + x^4 + x^3 + x^2 + 1$
15	$x^8 + x^5 + x^3 + x + 1$
16	$x^8 + x^5 + x^3 + x^2 + 1$
17	$x^8 + x^5 + x^4 + x^3 + 1$
18	$x^8 + x^5 + x^4 + x^3 + x^2 + x + 1$
19	$x^8 + x^6 + x^3 + x^2 + 1$
20	$x^8 + x^6 + x^4 + x^3 + x^2 + x + 1$

Список литературы

- [1] А.В.Ушаков Прикладная теория информации: элементы теории и практикум - Санкт-Петербург: НИУ ИТМО, 2012. - 325 с.
- [2] Тихонов В. И., Миронов М. А. Марковские процессы. – М.: Советское радио, 1977. – 488 с.
- [3] Никифорова Л. Т., Салмыгин И. П., Ушаков А. В. Методические указания для проведения лабораторных работ по курсу "Телемеханика". – Москва, 1989. – 82 с.

Краснов Александр Юрьевич

Практикум по прикладной теории информации

Практикум

В авторской редакции

Редакционно-издательский отдел Университета ИТМО

Зав. РИО

Н.Ф. Гусарова

Подписано к печати

Заказ №

Тираж

Отпечатано на ризографе

Редакционно-издательский отдел
Университета ИТМО
197101, Санкт-Петербург, Кронверкский пр., 49, литер А