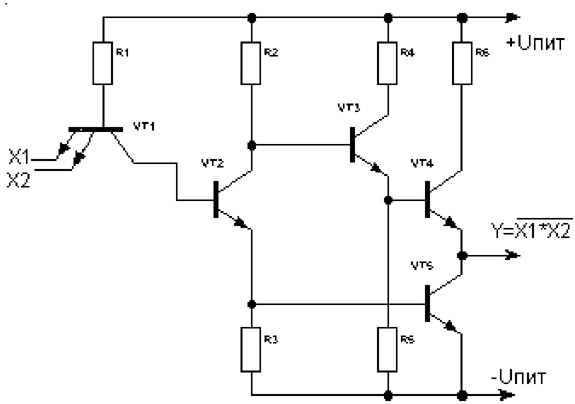
ФГБОУ ВПО “Воронежский государственный технический университет”

Кафедра полупроводниковой электроники и наноэлектроники

297-2013

МЕТОДИЧЕСКИЕ УКАЗАНИЯ

к выполнению лабораторных работ № 1 - 4
по дисциплине «Схемотехника»
для студентов направления
210100.62 «Электроника и наноэлектроника»,
профиля «Микроэлектроника и твердотельная электроника»
очной формы обучения



Воронеж 2013

Составители: канд. техн. наук А.В. Арсентьев,
ассистент Е.Ю. Плотникова

УДК 621.382

Методические указания к выполнению лабораторных работ № 1 - 4 по дисциплине «Схемотехника» для студентов направления 210100.62 «Электроника и наноэлектроника», профиля «Микроэлектроника и твердотельная электроника» очной формы обучения / ФГБОУ ВПО «Воронежский государственный технический университет»; сост. А.В. Арсентьев, Е.Ю. Плотникова. Воронеж, 2013. 36 с.

В методических указаниях описывается работа с экспериментальными учебными стендами, приводится методика работы функционально полных наборов логических элементов, описываются структуры дешифраторов, преобразователя кодов, сумматора. Приведены вопросы для самопроверки и библиографический список.

Методические указания подготовлены на магнитном носителе в текстовом редакторе Microsoft Word 2010 и содержится в файле МУ Схемотехника.rtf

Табл. 12. Ил. 18. Библиогр.: 5 назв.

Рецензент д-р техн. наук, проф. А.В. Строгонов

Ответственный за выпуск зав. кафедрой
д-р физ.-мат. наук, проф. С.И. Рембеза

Издается по решению редакционно-издательского совета Воронежского государственного технического университета

ЛАБОРАТОРНАЯ РАБОТА № 1

БАЗОВЫЕ ЛОГИЧЕСКИЕ ЭЛЕМЕНТЫ

Цель работы: изучение принципа работы базовых логических элементов и устройств, реализующих основные логические операции.

Основными операциями булевой алгебры являются отрицание (инверсия), логическое сложение (дизъюнкция) и логическое умножение (коньюнкция).

Отрицание – такая логическая операция над одной переменной, в результате которой появляется новое высказывание, которое принимает истинное значение, если входное высказывание истинно. Отрицание обозначается чертой над переменной, например, $\bar{A}$ читается «НЕ А». Таблица истинности этой операции приведена на рисунке, а.

Логическое сложение – такая логическая операция над двумя и более исходными данными (высказываниями), в результате которой появляется новое сложное высказывание, принимающее истинное значение, если истинно хотя бы одно из высказываний, и ложное, если одновременно ложны все исходные высказывания. Обозначается знаком плюс, например, А+В читается «А ИЛИ В». Таблица истинности операции логического сложения над тремя переменными приведена на рисунке, б.

Логическое умножение – такая логическая операция над двумя и более исходными данными (высказываниями), в результате которой появляется новое сложное высказывание, принимающее истинное значение, если одновременно истинны все исходные высказывания, и ложное, если ложно хотя бы одно из исходных высказываний. Обозначается в виде произведения в обычной алгебре, например, $A^{\times}B$, $A{\cdot}B$, AB, читается «А и В». Таблица истинности операции логического умножения над тремя переменными приведена на рис. 1,в.

Основные тождества булевой алгебры

Комбинируя основные логические операции, можно получать новые, более сложные высказывания или суждения. Существует 15 основных тождеств булевой алгебры (тавтология), получающихся с помощью основных логических операций:

1	$A+\bar{A}=1$	Элементарное высказывание
2	$A\cdot\bar{A}=0$	Элементарное высказывание
3	$A+1=1$	Элементарное высказывание
4	$A\cdot 1=A$	Элементарное высказывание
5	$A+A=A$	Соотношение абсорбции
6	$A\cdot A=A$	Соотношение абсорбции
7	$\bar{\bar{A}}=A$	Двойное отрицание
8	$(A+B)+C=A+(B+C)$	Сочетательное тождество для логической суммы
9	$(AB)C=A(BC)$	Сочетательное тождество для логического произведения
10	$A+B=B+A$	Переместительное тождество для логической суммы
11	$AB=BA$	Переместительное тождество для логического произведения
12	$A(B+C)=AB+AC$	Распределительное тождество
13	$(A+B)(A+C)=BC+A$	Распределительное тождество
14	$\overline{A+B}=\bar{A}\cdot\bar{B}$	Соотношение двойственности (теорема де Моргана)
15	$\overline{AB}=\bar{A}+\bar{B}$	Соотношение двойственности (теорема де Моргана)

Доказательство основных тождеств может быть проведено разными способами, например, сравнением таблиц истинности для левой и правой части каждого тождества. Особого внимания заслуживают тождества 13-15, не имеющие аналогии в обычной алгебре. Ввиду того, что в схемотехнике широко используются операции НЕ-ИЛИ (отрицание логического сложения) и НЕ-И (отрицание логического умножения), тождества 14 и 15 можно доказать с помощью таблиц истинности.

$\overline{A+B} = \bar{A}\cdot\bar{B}$						
A	B	A+B	$\overline{A+B}$	$\bar{A}$	$\bar{B}$	$\bar{A}\,\bar{B}$
0	0	0	1	1	1	1
0	1	1	0	1	0	0
1	0	1	0	0	1	0
1	1	1	0	0	0	0

$\overline{AB}=\bar{A}+\bar{B}$						
A	B	AB	$\overline{AB}$	$\bar{A}$	$\bar{B}$	$\bar{A}+\bar{B}$
0	0	0	1	1	1	1
0	1	0	1	1	0	1
1	0	0	1	0	1	1
1	1	1	0	0	0	0

а) Доказательство тождеств 14,15 с помощью таблиц истинности

На основании тождеств 1-15 могут быть доказаны полезные следствия, дополняющие основные тождества:

1. $B\bar{A}+BA=B$;
2. $A+\bar{A}B=A+B$;
3. $A\bar{B}+\bar{A}B=A\oplus B$ «исключительное ИЛИ»;
4. $\overline{A\oplus B}=\bar{A}B+AB$;
5. $A\oplus A=0$;
6. $A\oplus\bar{A}=1$;
7. $A\oplus 1=\bar{A}$;
8. $A\oplus 0=A$;
9. Если $A\oplus B=C$, то $A\oplus C=B$ и $B\oplus C=A$.

Приведенные следствия можно доказать, используя таблицы истинности.

Функционально полные наборы логических элементов

Используя основные логические операции, можно реализовать логическую схему (структуру) любой степени сложности, выполняющую заданный закон функционирования, т.е. основные логические операции образуют функционально полный набор логических элементов.

Задача синтеза сложных логических схем эквивалентна представлению сложных булевых функций простыми функциями, в качестве которых могут быть использованы операции И, ИЛИ, НЕ, образующие функционально полный набор логических элементов. Однако функционально полные наборы логических элементов могут быть образованы другими исходными элементами, например, И, НЕ.

В табл. 1 представлены пять функционально полных наборов логических элементов. Эти пять наборов не исчерпывают все функционально полные системы.

Таблица 1

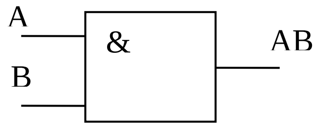
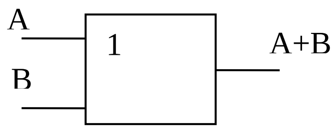
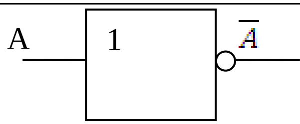
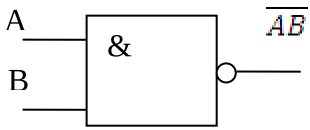
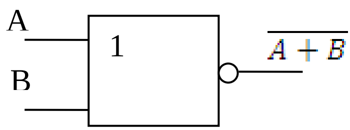
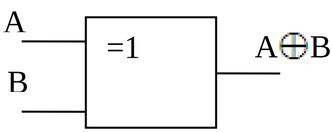
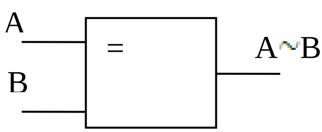
Функционально полные наборы логических элементов

Набор исходных логических элементов	Получение остальных логических операций		
	И	ИЛИ	НЕ
1	2	3	4
И, НЕ	-	$X+Y=\overline{X+Y}$	-
ИЛИ, НЕ	$X \cdot Y=\overline{\overline{X}+\overline{Y}}$	-	-
$\overline{X+Y}$ ИЛИ-НЕ (стрелка Пирса)	$\overline{\overline{X+X}+\overline{Y+Y}}$	$\overline{\overline{X+Y}+\overline{X+Y}}$	$\overline{X}=\overline{X+X}$
$\overline{XY}$ НЕ-И (штрих Шеффера)	$\overline{\overline{XY}\cdot\overline{XY}}$	$\overline{\overline{XX}+\overline{YY}}$	$\overline{X}=\overline{XX}$
Исключительное ИЛИ, И, единица	-	$X+Y=1\oplus\overline{X}\,\overline{Y}=1\oplus[1\oplus X][1\oplus Y]$	$\overline{X}=X\oplus 1$

В табл. 2 приводится условные графические обозначение булевых выражений.

Таблица 2

Условные графические обозначения булевых выражений

Булевы выражения	Операция	Обозначения
AB	И	
$A+B$	ИЛИ	
$\overline{A}$	НЕ	
$\overline{AB}$	-НЕ И НЕ-И	
$\overline{A+B}$	ИЛИ-НЕ НЕ-ИЛИ	
$A\oplus B=A\overline{B}+\overline{A}B$	Исключительное ИЛИ (неравнозначность)	
$A\sim B=\overline{A\oplus B}=AB+\overline{A}\cdot\overline{B}$	Эквивалентность (равнозначность)	

Экспериментальная часть

Целью экспериментальной части является изучение функционально полных наборов логических элементов и приобретение практических навыков реализации основных тождеств булевой алгебры на основе заданных функционально полных наборов логических элементов.

Порядок выполнения:

1. Получить от преподавателя, ведущего занятия, номер варианта задания на лабораторную работу.
2. Записать задание на лабораторную работу, пользуясь данными табл. 3.
3. Составить таблицы истинности логических операций и тождеств булевой алгебры, указанных в варианте на лабораторную работу.
4. Используя указанный в задании на лабораторную работу функционально полный набор логических элементов, нарисовать схемы, реализующие логические операции и тождества булевой алгебры.
5. Показать преподавателю составленные схемы, реализующие заданные логические функции. Получив разрешение преподавателя, собрать схемы на передней панели лабораторного макета.
6. Снять таблицу истинности каждой схемы, сравнить с ожидаемой. Выключить макет.

Таблица 3

Варианты заданий на лабораторную работу

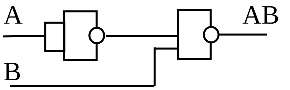
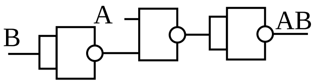
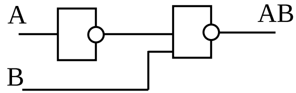
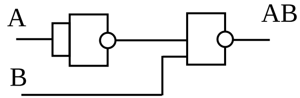
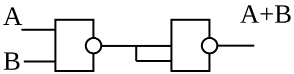
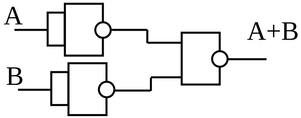
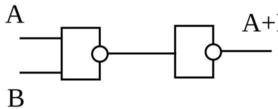
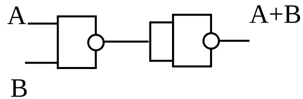
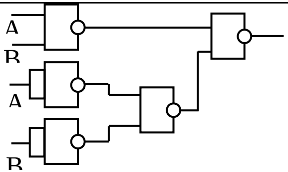
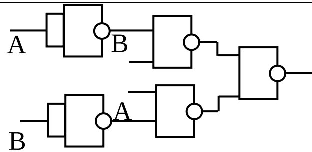
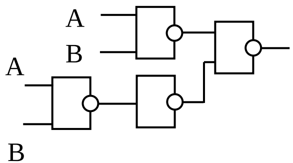
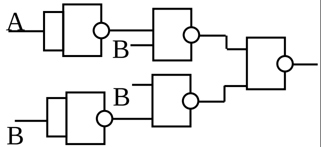
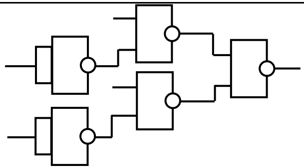
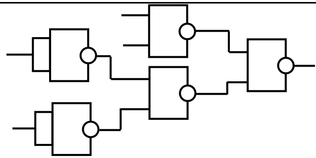
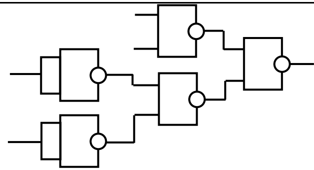
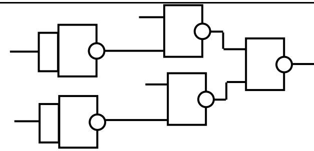
Содержание задания	Номер варианта задания			
	1	2	3	4
Проверить тождества булевой алгебры	$1 \div 7$, 12	$1 \div 7$, 13	$1 \div 7$, 14, 15	$1 \div 7$, 12,13
Реализовать булевы функции по схемам, приведенным в таб. 5	НЕ-ИЛИ	НЕ-И	НЕ-ИЛИ НЕ-И НЕ	НЕ-И НЕ-ИЛИ

Вопросы для отчета

1. Что такое логическая операция?
2. Какие операции и тождества булевой алгебры Вы знаете?
3. Что такое логические операции? Логическое отрицание.
4. Логическое сложение. Логическое умножение.
5. Что такое таблица истинности?
6. Докажите основные тождества булевой алгебры.
7. Что такое булевы функции двух аргументов? Приведите примеры булевых функций.
8. Что такое функционально полный набор логических элементов? Приведите примеры функционально полных наборов.
9. Реализуйте булевы функции двух аргументов, используя функционально полные логические наборы:

   а) И-НЕ; б) ИЛИ-НЕ; в) И, НЕ; г) ИЛИ, НЕ.
10. По заданной таблице определите логическую операцию.

Таблица 4

Варианты заданий			
1-2	3-4	5-6	7-8
 AB=A+B	 AB=AB	 AB=A+B	 AB=A+B
 A+B=A+B+A+B	 A+B=AA BB	 A+B=A+B	 A+B=A+B+A+B
 AB+AB=A+B	 AB+AB=AAB+ABB	 B+AB=A+B+AB	 AB+AB=AAB+ABB
 AB+AB=A+A+B+A+B+B	 AB+AA=AB AA BB	 AB+AB=AB A B	 AB+AB=A+B+B+A+A+B

ЛАБОРАТОРНАЯ РАБОТА №2

ДЕШИФРАТОРЫ

Цель работы: изучение структуры и характеристик дешифраторов и синтез дешифраторов на ИС.

Дешифратор, или декодер, – это комбинационная структура, предназначенная для образования минтермов n входных переменных; в полном дешифраторе образуются все минтермы n переменных, т. е. реализуется система функций:

$$
\begin{aligned}
f_0 &= \overline{A}_{n-1} \cdot \overline{A}_{n-2} \cdot \overline{A}_{n-3} \dots A_i \dots \overline{A}_2 \overline{A}_1 \overline{A}_0 \\
f_1 &= \overline{A}_{n-1} \cdot \overline{A}_{n-2} \cdot \overline{A}_{n-3} \dots A_i \dots \overline{A}_2 \overline{A}_1 A_0 \\
f_2 &= \overline{A}_{n-1} \cdot \overline{A}_{n-2} \cdot \overline{A}_{n-3} \dots A_i \dots \overline{A}_2 A_1 A_0 \\
&------------------- \\
f_{2^n-2} &= \overline{A}_{n-1} \cdot A_{n-2} \cdot A_{n-3} \dots A_i \dots A_2 A_1 A_0 \\
f_{2^n-1} &= A_{n-1} \cdot A_{n-2} \cdot A_{n-3} \dots A_i \dots A_2 A_1 A_0
\end{aligned}
\qquad (1)
$$

где A_0 , A_1 , A_2 , …, A_i , …, A_{n-3} , A_{n-2} , A_{n-1} и $\overline{A}_0$, $\overline{A}_1$, $\overline{A}_1$, …, $\overline{A}_i$, …,$\overline{A}_{n-3}$, $\overline{A}_{n-2}$, $\overline{A}_{n-1}$, т. е. n_i входных переменных и их инверсий соответственно;

f_0, f_1, f_2, …, f_{2^n-2}, f_{2^n-1}– входные сигналы дешифратора, каждая функция системы (1) представляет минтерм, индекс которого равен i.

Классификация дешифраторов

По внутренней структуре различают три типа дешифраторов:

– прямоугольная структура дешифратора;
– пирамидальная структура дешифратора;
– ступенчатая структура дешифратора.

Прямоугольная структура дешифратора на ИС

Как работает прямоугольная структура дешифратора, ясно из рис. 1, где представлена структура прямоугольного дешифратора на две переменные.

На два входа дешифратора подаются две входные переменные A_0 и A_1. Для образования инверсии входных переменных A_0и A_1 используется две схемы отрицания (НЕ). Каждому выходу схемы соответствует своя схема логического умножения (И), на входе которой подаются комбинации входных переменных и их инверсий. Таким образом, на входах схемы образуются минтермы двух переменных.

Аналогичным образом реализуются прямоугольные структуры дешифратора на большое число переменных. На рис. 2 приведена структура прямоугольного дешифратора на 3 переменных. Видно, что схема дешифратора с увеличением числа переменных до двух стала значительно сложнее.

В таблице приводятся численные значения показателей прямоугольных дешифраторов на различное число входных переменных.

Показатели прямоугольных дешифраторов

Число входных переменных	Число входов	Число выходов	Число ИС, необходимых для реализации	Требуемое число выходов схем	Число внешних выходов
2	2	4	6	2	8
3	3	8	11	3	13
4	4	16	20	4	22
5	5	32	37	5	39
6	6	64	72	6	74

Из таблицы видно, что с ростом n резко возрастает число внешних выводов дешифратора и число ИС, необходимых для реализации прямоугольной структуры, что свидетельствует о значительном увеличении сложности схемы с ростом n.

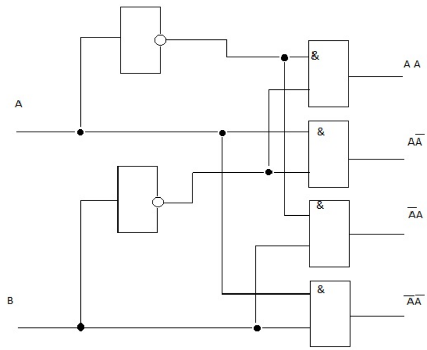


Рис. 1. Прямоугольный дешифратор на 2 переменные на элементах И

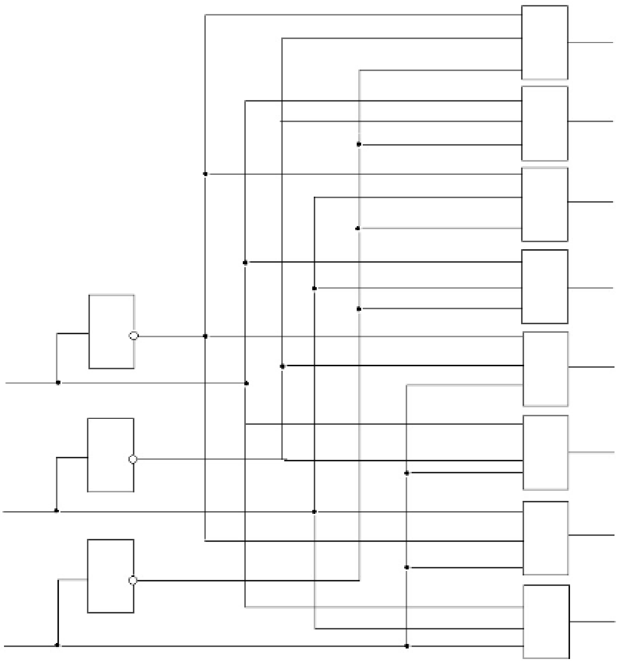


Рис. 2. Прямоугольный дешифратор на 3 переменные на элементах И

Поэтому желательная разработка структур дешифраторов, требующих меньшее число элементов. Это достигается двухступенчатым и пирамидальным дешифраторами, где используются общие элементы в цепях, формирующих сигналы для нескольких выводов.

Достоинством прямоугольного дешифратора является линейная величина времени задержки сигнала, поскольку между любым входом и выходом находится один элемент И (если не считать элементов НЕ, требуемых для образования инверсий входных переменных).

Пирамидальный дешифратор на ИС

Сокращение числа элементов в пирамидальном дешифраторе достигается за счет использования общих элементов И при формировании частных булевых произведений, входящих в минтермы.

Работу пирамидального дешифратора рассмотрим на примере дешифратора на четыре переменных, структура которого показана на рис. 3.

Выходные функции дешифратора на четыре переменных можно представить в виде четырех групп:

1 группа	2 группа	3 группа	4 группа
$f_0 = \overline{A}_3\overline{A}_2\overline{A}_1\overline{A}_0$	$f_1 = \overline{A}_3\overline{A}_2\overline{A}_1A_0$	$f_2 = \overline{A}_3\overline{A}_2A_1\overline{A}_0$	$f_3 = \overline{A}_3\overline{A}_2A_1A_0$
$f_4 = A_3A_2\overline{A}_1\overline{A}_0$	$f_5 = \overline{A}_3A_2\overline{A}_1A_0$	$f_6 = \overline{A}_3A_2A_1\overline{A}_0$	$f_7 = \overline{A}_3A_2A_1A_0$
$f_8 = A_3\overline{A}_2\overline{A}_1\overline{A}_0$	$f_9 = A_3\overline{A}_2\overline{A}_1A_0$	$f_{10} = A_3\overline{A}_2A_1\overline{A}_0$	$f_{11} = A_3\overline{A}_2A_1A_0$
$f_{12} = A_3A_2\overline{A}_1\overline{A}_0$	$f_{13} = A_3A_2\overline{A}_1A_0$	$f_{14} = A_3A_2A_1\overline{A}_0$	$f_{15} = A_3A_2A_1A_0$

В пределах каждой группы выходные функции имеют общие множители: $\overline{A}_0\overline{A}_1$, $A_0\overline{A}_1$, $\overline{A}_0A_1$, A_0A_1 соответственно. Для образования общих множителей можно использовать общие элементы.

Как видно из рис. 3, пирамидальный дешифратор на четыре переменных можно подразделить на три части.

1 часть представляет собой прямоугольный дешифратор на две переменных A_0и A_1, образующий вышеперечисленные общие множители для каждой из четырех групп выходных функций дешифратора.

2 часть образует все минтермы трех переменных A_0, A_1, A_2 путем умножения общих множителей групп на переменную A_2 или ее инверсию.

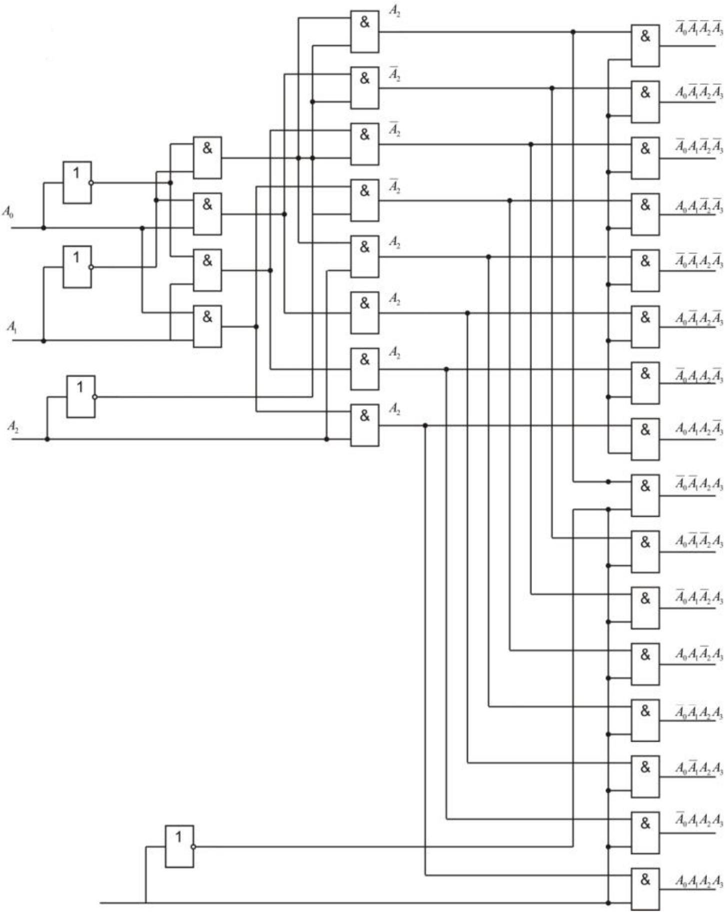


Рис. 3. Пирамидальный дешифратор на 4 переменные

3 часть образует все минтермы четырех переменных A_3, A_2, A_1, A_0, что достигается умножением минтермов трех переменных A_2, A_1, A_0 на переменную A_3 или ее инверсию. При этом как видим из рис. 3, в качестве базовых ИС применяются И, НЕ.

Пирамидальный дешифратор на ИС обеспечивает выигрыш в числе элементов по сравнению с прямоугольным дешифратором при достаточно большом числе входных переменных. Так при числе переменных не менее 16 пирамидальный дешифратор обеспечивает четырехкратный выигрыш в числе компонентов. Однако при этом время дешифрации существенно превышает задержку сигнала в прямоугольных и двухступенчатых дешифраторах. В связи с этим пирамидальный дешифратор на ИС применяется редко.

Ступенчатый дешифратор на ИС

Ступенчатую структуру дешифратора рассмотрим на примере двухступенчатого дешифратора на четыре переменных, структура которого показана на рис. 4. Дешифратор содержит две ступени.

Первая ступень включает в себя два прямоугольных дешифратора на две переменные, формирующих произведения $\overline{A}_0\overline{A}_1$, $A_0\overline{A}_1$, $\overline{A}_0A_1$, A_0A_1 и $A_2\overline{A}_3$, $\overline{A}_2A_3$, A_2A_3, $\overline{A}_2\overline{A}_3$, каждой соответственно.

На второй ступени производится перемножение произведений, полученных на первой ступени, и образование всех минтермов четырех переменных.

Двухступенчатый дешифратор содержит меньшее число компонентов, чем прямоугольный дешифратор, если число переменных превышает 4. При числе переменных, равном 10, выигрыш в числе компонентов возрастает в 4 раза.

Помимо двухступенчатых дешифраторов существуют многоступенчатые, отличающиеся тем, что при большом числе входных переменных их разбивают не на две, а большее число групп, что позволяет получить дополнительный выигрыш в числе компонентов схемы.

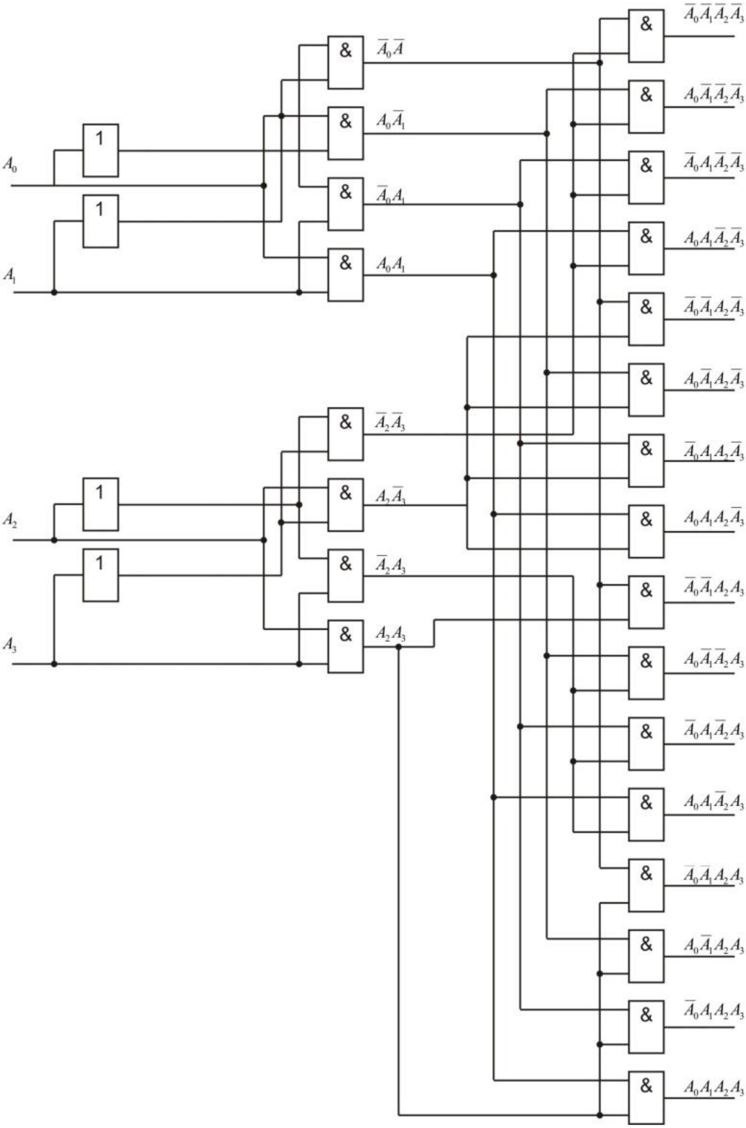


Рис. 4. Двухступенчатый дешифратор на четыре переменные

Экспериментальная часть

Целью экспериментальной части является приобретение навыков реализации различных структур дешифраторов на ИС.

Порядок выполнения:

К выполнению лабораторного задания допускаются студенты, подготовившие варианты структуры дешифратора в тетради.

1. На лицевой панели универсального лабораторного стенда собрать прямоугольную структуру дешифратора на четыре переменных.
2. Убедиться в правильном функционировании собранной логической структуры.
3. Собрать пирамидальную структуру дешифратора на четыре переменных.
4. Убедиться в правильном функционировании собранной логической структуры.
5. Собрать двухступенчатую структуру дешифратора на четыре переменные.
6. Убедиться в правильном функционировании собранной логической структуры.

Вопросы для отчета по лабораторной работе

1. Дайте классификацию дешифраторов.
2. Поясните принцип работы прямоугольного дешифратора.
3. Нарисуйте структуру прямоугольного дешифратора на четыре переменных.
4. Поясните принцип работы пирамидального дешифратора.
5. Нарисуйте структуру прямоугольного дешифратора на четыре переменных.

6. Поясните принцип работы пирамидального дешифратора.

7. Нарисуйте структуру двухступенчатого дешифратора на четыре переменных.

8. Объясните зависимость характеристик дешифратора от числа переменных.

9. Перечислите достоинства и недостатки различных структур дешифраторов.

10. Дайте сравнительную оценку быстродействия различных типов дешифраторов.

11. Перечислите область применения дешифраторов.

ЛАБОРАТОРНАЯ РАБОТА № 3

ПРЕОБРАЗОВАТЕЛИ КОДОВ

Цель работы: изучение основ проектирования и принципа работы различных типов преобразователей кодов.

В цифровых системах с целью упрощения структур и облегчения их взаимодействия с устройствами ввода и потребителя информации и для повышения надежности аппаратуры за счет автоматического обнаружения и исправления ошибок используются различные варианты арифметических кодов. При этом возникает задача преобразования одного кода в другой. Устройство, осуществляющее преобразование одного кода в другой, называется преобразователем кода, или кодопреобразователем.

Одной из характеристик кодов является дистанция между кодовыми словами, которая определяется количеством символов, которые требуется изменить при переходе от одного слова к другому. Например, дистанция между словами 0001 и 1000 равна двум, а между 1000 и 1001 равна единице.

Под минимальной дистанцией понимают наименьшее количество знаков между любыми двумя словами кода. Обычный бинарный код характеризуется минимальной дистанцией, равной 1. Максимальная дистанция для бинарного кода равна числу символов в слове (длине слова).

Карты Карно

Карты Карно являются разновидностью карт минтермов и отличаются только порядком расположения символов, перечисленных в циклическом коде. Карты Карно для четырех переменных приводятся на рис. 1.

	CD \ AB	00	01	11	01	
				A	A	
	00	$\bar{A}\bar{B}\bar{C}\bar{D}$	$\bar{A}B\bar{C}\bar{D}$	$AB\bar{C}\bar{D}$	$A\bar{B}\bar{C}\bar{D}$	
	01	$\bar{A}\bar{B}\bar{C}D$	$\bar{A}B\bar{C}D$	$AB\bar{C}D$	$A\bar{B}\bar{C}D$	D
C	11	$\bar{A}\bar{B}CD$	$\bar{A}BCD$	$ABCD$	$A\bar{B}CD$	D
C	01	$\bar{A}\bar{B}C\bar{D}$	$\bar{A}BC\bar{D}$	$ABC\bar{D}$	$A\bar{B}C\bar{D}$	

Рис. 1. Карты Карно

Циклические коды

Циклические коды имеют минимальную дистанцию, равную единице, и, кроме того, характеризуются равнодиотантностью.

Для образования циклических кодов удобно пользоваться картами Карно, соединяя стрелками смежные (соседние) клетки карты. Тем самым реализуется равнодистантность.

На рис. 2 показано образование одного из возможных циклических кодов с помощью карты Карно, так называемого кода Грея. Синтезируем устройство преобразования из бинарного кода в код Грея. Для этого воспользуемся табл. 1 соответствия бинарного кода и кода Грея, полученной с помощью рис. 2.

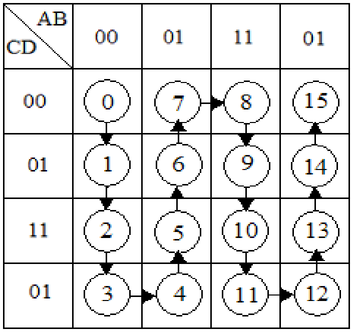


Рис. 2. Формирование кода Грея с помощью карты Карно

Для синтеза преобразователя кода нанесем на карты Карно выражения для A, B, C, D, как это показано на рис. 3, и проведем считывание минимальным образом. В результате считывания имеем

$$\left.\begin{aligned} A &= a = 0 \oplus a \\ B &= a\bar{b} + \bar{a}b = a \oplus b \\ C &= b\bar{c} + \bar{b}c = b \oplus c \\ D &= c\bar{d} + \bar{c}d = c \oplus d \end{aligned}\right\} \qquad (1)$$

Видно, что преобразователь бинарного кода в код Грея можно построить, используя логическую операцию ИСКЛЮЧИТЕЛЬНОЕ ИЛИ. Структура кодопреобразователя приведена на рис. 4.

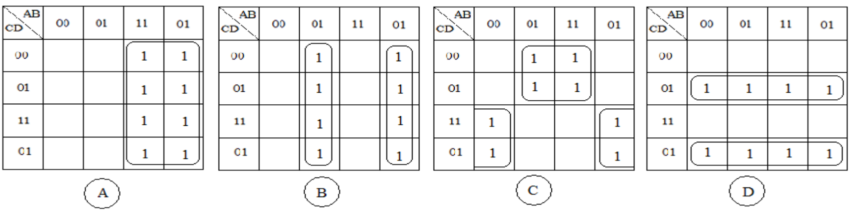


Рис. 3. Выражения для A,B,C,D преобразователя бинарного кода

Таблица 1

Таблица соответствия бинарного кода и кода Грея

№	a	b	c	d	A	B	C	D
0	0	0	0	0	0	0	0	0
1	0	0	0	1	0	0	0	1
2	0	0	1	0	0	0	1	1
3	0	0	1	1	0	0	1	0
4	0	1	0	0	0	1	1	0
5	0	1	0	1	0	1	1	1
6	0	1	1	0	0	1	0	1
7	0	1	1	1	0	1	0	0
8	1	0	0	0	1	1	0	0
9	1	0	0	1	1	1	0	1
10	1	0	1	0	1	1	1	1
11	1	0	1	1	1	1	1	0
12	1	1	0	0	1	0	1	0
13	1	1	0	1	1	0	1	1
14	1	1	1	0	1	0	0	1
15	1	1	1	1	1	0	0	0
	бинарный код				код Грея			

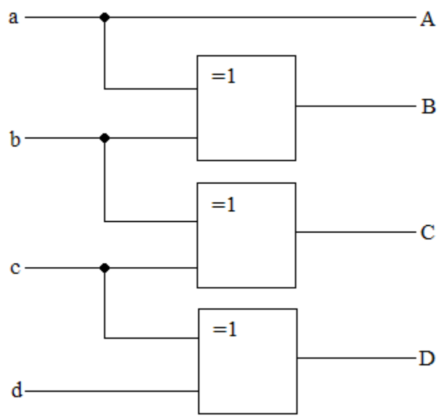


Рис. 4. Структура преобразователя бинарного кода в код Грея

Преобразователь бинарного кода в код с дополнением до 2

Преобразование числа, записанного в бинарном коде, в число, записанное кодом с дополнением до 2, производится по следующим правилам: двоичные цифры числа, записанного в бинарном коде, заменяются на противоположные; в младший разряд добавляют 1.

На основании указанных правил строим таблицу истинности, которая приведена в табл. 1. По данным табл. 2 строим карты минтермов для переменных A, B, C, D, приведенные на рис. 5.

Поскольку комбинации с 10 до 15 являются избыточными, получаем следующие выражения:

$$\left.\begin{aligned}
D &= d = d\oplus 0\\
C &= \bar{c}d + c\bar{d} = c\oplus d\\
B &= b\bar{c}\bar{d} + \bar{b}c + \bar{b}d = b\bar{c}\bar{d} + \bar{b}(c+d) = b\left(\overline{c+d}\right) + \bar{b}(c+d) = b\oplus(c+d)\\
A &= \bar{a}b + \bar{a}d + \bar{a}c + a\bar{b}\bar{c}\bar{d} = \bar{a}(b+c+d) + a\left(\overline{b+c+d}\right) = a\oplus(b+c+d)
\end{aligned}\right\} \qquad (2)$$

cd \ ab	00	01	11	10
00		1	x	1
01	1	1	x	
11	1	1	x	x
10	1	1	x	x

A

cd \ ab	00	01	11	10
00		1	x	
01	1		x	1
11	1		x	x
10	1		x	x

B

cd \ ab	00	01	11	10
00			x	
01	1	1	x	1
11			x	
10	1	1	x	x

C

cd \ ab	00	01	11	10
00			x	
01	1	1	x	1
11	1	1	x	x
10			x	x

D

Рис. 5. Выражения для A,B,C,D преобразователя бинарного кода в код с дополнением до 2

В соответствии с (2) строим структуру преобразователя, показанную на рис. 6.

Преобразователь бинарного кода в код 2421

Цифры 2421 в названии кода символизируют веса отдельных разрядов. Код 2421 называется еще кодом Айкена и относится к категории самодополняющего кода. По данным табл. 2 строим карты Карно для выражений A, B, C, D с учетом избыточных комбинаций, показанных на рис. 7.

Таблица 2

Таблица истинности для преобразователей кодов

Десятичные числа	бинарный код				код с избытком 3				код Айкена				код 2 из 5					код Джонсона				
	A	B	C	D	A	B	C	D	A	B	C	D	A	B	C	D	E	A	B	C	D	E
0	0	0	0	0	0	0	1	1	0	0	0	0	0	1	1	0	0	0	0	0	0	0
1	0	0	0	1	0	1	0	0	0	0	0	1	1	1	0	0	0	1	0	0	0	0
2	0	0	1	0	0	1	0	1	0	0	1	0	1	0	1	0	0	1	1	0	0	0
3	0	0	1	1	0	1	1	0	0	0	1	1	1	0	0	1	0	1	1	1	0	0
4	0	1	0	0	0	1	1	1	0	1	0	0	0	1	0	1	0	1	1	1	1	0
5	0	1	0	1	1	0	0	0	1	0	1	1	0	0	1	1	0	1	1	1	1	1
6	0	1	1	0	1	0	0	1	1	1	0	0	1	0	0	0	1	0	1	1	1	1
7	0	1	1	1	1	0	1	0	1	1	0	1	0	1	0	0	1	0	0	1	1	1
8	1	0	0	0	1	0	1	1	1	1	1	0	0	0	1	0	1	0	0	0	1	1
9	1	0	0	1	1	1	0	0	1	1	1	1	0	0	0	1	1	0	0	0	0	1
избыточные комбинации	1	0	1	0	0	0	0	0	0	1	0	1	Все пятиразрядные числа, где единицы занимает не два разряда					Все пятиразрядные числа, где единицы занимает не два разряда				
	1	0	1	1	0	0	0	1	0	1	1	0										
	1	1	0	0	0	0	1	0	0	1	1	1										
	1	1	0	1	1	1	0	1	1	0	0	0										
	1	1	1	0	1	1	1	0	1	0	0	1										
	1	1	1	1	1	1	1	1	1	0	1	0										

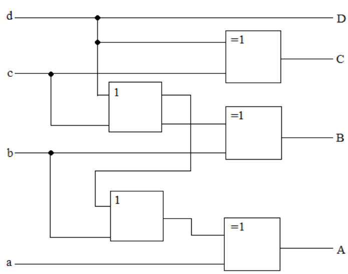


Рис. 6. Преобразователь бинарного кода в код с дополнением до 2

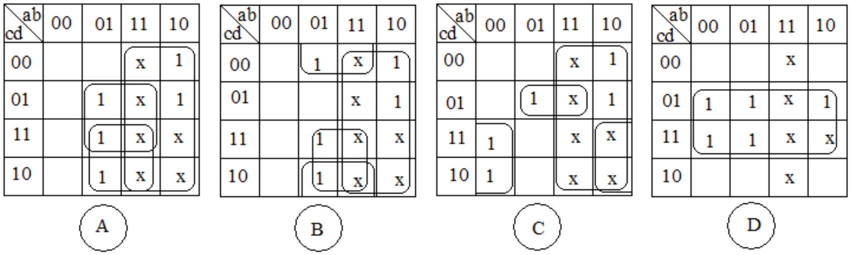


Рис. 7. Выражения для A,B,C,D преобразователя бинарного кода в код Айкена

Считывая с карт минтермов, получаем следующие выражения для преобразователя кода:

$$\left.\begin{array}{c} A = a + bd + bc \\ B = a + b\bar{d} + bc \\ C = a + \bar{b}c + b\bar{c}d \\ A = d \end{array}\right\} \qquad (3)$$

Преобразователь десятичного кода в десятичный бинарный код

Десятичный бинарный код применяется в устройствах, использующих декарную форму отображения информации.

В десятичном бинарном коде на каждую десятичную цифру требуется четыре двоичных символа A, B, C, D. При этом комбинации двоичных символов, соответствующие камерам от 10 до 15, оказываются избыточными. Сказанное иллюстрируется табл. 3.

Непосредственно из таблицы вытекают следующие выражения для преобразования десятичного кода в десятичный бинарный код:

$$\left.\begin{array}{c} A = L_8 + L_9 = \overline{\overline{L_8 + L_9}} \\ B = L_4 + L_5 + L_6 + L_7 = \overline{\overline{L_4 + L_5} * \overline{L_6 + L_7}} \\ C = L_2 + L_3 + L_6 + L_7 = \overline{\overline{L_2 + L_3} * \overline{L_6 + L_7}} \\ D = L_1 + L_3 + L_5 + L_7 + L_9 = \overline{\overline{L_1 + L_3 + L_5} * \overline{L_7 + L_9}} \end{array}\right\} \qquad (4)$$

Структура кодопреоборазователя в базисе НЕ, И-НЕ, ИЛИ-НЕ показана на рис. 8. Структура имеет десять входов и четыре выхода A, B, C, D. При подаче логической 1 на вход появляется бинарный эквивалент десятичного числа N на выходы A, B, C, D.

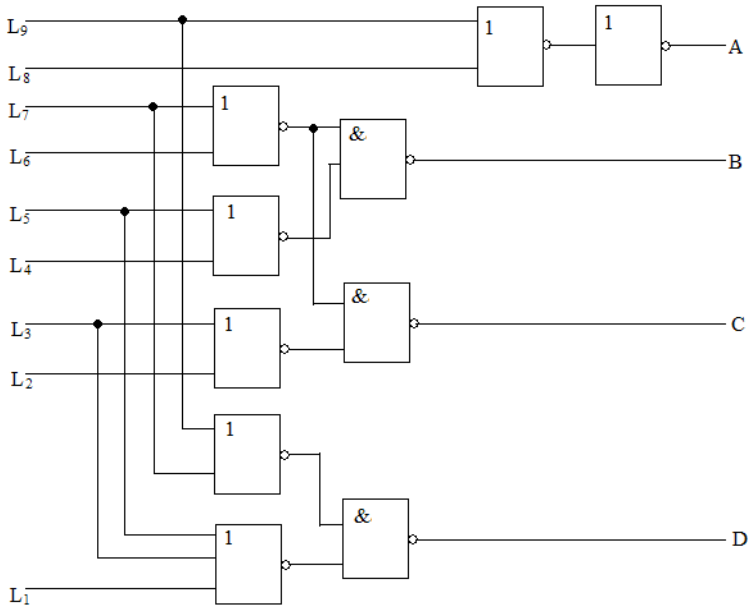


Рис. 8. Структура преобразователя десятичного кода в десятичный бинарный код

Комбинации двоичных символов на выходах преобразователя с 1010 по 1111 являются нештатными (избыточными), и их можно использовать для обнаружения ошибок в работе преобразователей.

Нетрудно показать, что функция принимает значение, равное 1, при прохождении нештатной комбинации, т. е. при наличии ошибок в работе старших разрядов цепи.

Таблица 3

Соответствие между десятичным и десятичным бинарным кодом

Входы											Выходы				
№	L_9	L_8	L_7	L_6	L_5	L_4	L_3	L_2	L_1	L_0	A	B	C	D	E
0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0
1	0	0	0	0	0	0	0	0	1	0	0	0	0	1	0
2	0	0	0	0	0	0	0	1	0	0	0	0	1	0	0
3	0	0	0	0	0	0	1	0	0	0	0	0	1	0	0
4	0	0	0	0	0	1	0	0	0	0	0	1	0	0	0
5	0	0	0	0	1	0	0	0	0	0	0	1	0	1	0
6	0	0	0	1	0	0	0	0	0	0	0	1	1	0	0
7	0	0	1	0	0	0	0	0	0	0	0	1	1	1	0
8	0	1	0	0	0	0	0	0	0	0	1	0	0	0	0
9	1	0	0	0	0	0	0	0	0	0	1	0	0	1	0
Избыточные комбинации											1	0	1	0	1
											1	0	1	1	1
											1	1	0	0	1
											1	1	0	0	1
											1	1	1	0	1
											1	1	1	1	1

Экспериментальная часть

Порядок выполнения.

1. В соответствии с номером варианта задания, выданного преподавателем, из табл. 4 взять карту Карно.

2. По заданной карте Карно составить таблицу истинности циклического кода.

3. Синтезировать структуру преобразователя бинарного кода в циклический код.

4. Реализовать полученную структуру на лабораторном стенде.

5. Снять таблицу истинности и убедиться в правильности работы преобразователя кода.

6. Синтезировать структуры преобразователя циклического кода в бинарный код.

7. Снять таблицу истинности и убедиться в правильности работы преобразователя.

8. В соответствии с номером варианта задания из табл. 4 необходимо взять заданный базис и тип преобразователя кода.

9. Составить структуру заданного преобразователя в нужном базисе, реализовать полученную структуру на стенде и снять таблицу истинности, убедившись в правильности работы.

Вопросы для отчета по работе

1. Проведите синтез преобразователя бинарного кода в циклический код в заданном базисе.
2. Синтезируйте преобразователь циклического кода в бинарный в заданном базисе.
3. Синтезируйте преобразователь бинарного кода в код Джонсона в заданном базисе.
4. Синтезируйте преобразователь кода Джонсона в бинарный в заданном базисе.
5. Синтезируйте преобразователь кода Айкена в бинарный код в заданном базисе.
6. Синтезируйте преобразователь бинарного кода в код Айкена в заданном базисе.
7. Синтезируйте преобразователь десятичного кода в бинарный код в заданном базисе.

Таблица 4

<table>
<tr><th colspan="4">Номера спаренных вариантов заданий</th></tr>
<tr><th>1</th><th>2</th><th>3</th><th>4</th></tr>
<tr>
<td>
<table>
<tr><th></th><th>00</th><th>01</th><th>11</th><th>10</th></tr>
<tr><th>00</th><td>0</td><td>1</td><td>2</td><td>3</td></tr>
<tr><th>01</th><td>11</td><td>12</td><td>13</td><td>4</td></tr>
<tr><th>11</th><td>10</td><td>15</td><td>14</td><td>5</td></tr>
<tr><th>10</th><td>9</td><td>8</td><td>7</td><td>6</td></tr>
</table>
И, ИЛИ, НЕ
Бинарный код
в код Джонсона
</td>
<td>
<table>
<tr><th></th><th>00</th><th>01</th><th>11</th><th>10</th></tr>
<tr><th>00</th><td>0</td><td>11</td><td>10</td><td>9</td></tr>
<tr><th>01</th><td>1</td><td>12</td><td>15</td><td>8</td></tr>
<tr><th>11</th><td>2</td><td>13</td><td>14</td><td>7</td></tr>
<tr><th>10</th><td>3</td><td>4</td><td>5</td><td>6</td></tr>
</table>
И-НЕ
Код Джонсона
в бинарный код
</td>
<td>
<table>
<tr><th></th><th>00</th><th>01</th><th>11</th><th>10</th></tr>
<tr><th>00</th><td>6</td><td>7</td><td>8</td><td>9</td></tr>
<tr><th>01</th><td>5</td><td>0</td><td>1</td><td>10</td></tr>
<tr><th>11</th><td>4</td><td>3</td><td>2</td><td>11</td></tr>
<tr><th>10</th><td>15</td><td>14</td><td>13</td><td>12</td></tr>
</table>
ИЛИ – НЕ
Бинарный код
в код с избытком 3
</td>
<td>
<table>
<tr><th></th><th>00</th><th>01</th><th>11</th><th>10</th></tr>
<tr><th>00</th><td>6</td><td>5</td><td>4</td><td>15</td></tr>
<tr><th>01</th><td>7</td><td>0</td><td>3</td><td>14</td></tr>
<tr><th>11</th><td>8</td><td>1</td><td>2</td><td>13</td></tr>
<tr><th>10</th><td>9</td><td>10</td><td>11</td><td>12</td></tr>
</table>
И-НЕ, НЕ
Код с избытком 3
в бинарный
</td>
</tr>
<tr>
<td>
<table>
<tr><th></th><th>00</th><th>01</th><th>11</th><th>10</th></tr>
<tr><th>00</th><td>3</td><td>2</td><td>1</td><td>0</td></tr>
<tr><th>01</th><td>4</td><td>5</td><td>6</td><td>7</td></tr>
<tr><th>11</th><td>11</td><td>10</td><td>9</td><td>8</td></tr>
<tr><th>10</th><td>12</td><td>13</td><td>14</td><td>15</td></tr>
</table>
ИЛИ-НЕ, НЕ
Бинарный код
в код Айкена
</td>
<td>
<table>
<tr><th></th><th>00</th><th>01</th><th>11</th><th>10</th></tr>
<tr><th>00</th><td>0</td><td>1</td><td>2</td><td>3</td></tr>
<tr><th>01</th><td>7</td><td>6</td><td>5</td><td>4</td></tr>
<tr><th>11</th><td>8</td><td>9</td><td>10</td><td>11</td></tr>
<tr><th>10</th><td>12</td><td>13</td><td>14</td><td>15</td></tr>
</table>
И-НЕ, ИЛИ-НЕ
Код Айкена
в бинарный код
</td>
<td>
<table>
<tr><th></th><th>00</th><th>01</th><th>11</th><th>10</th></tr>
<tr><th>00</th><td>3</td><td>4</td><td>5</td><td>6</td></tr>
<tr><th>01</th><td>0</td><td>15</td><td>14</td><td>13</td></tr>
<tr><th>11</th><td>1</td><td>10</td><td>11</td><td>12</td></tr>
<tr><th>10</th><td>2</td><td>9</td><td>8</td><td>7</td></tr>
</table>
ИЛИ, НЕ
Бинарный код
в код с дополнением до 2
</td>
<td>
<table>
<tr><th></th><th>00</th><th>01</th><th>11</th><th>10</th></tr>
<tr><th>00</th><td>10</td><td>11</td><td>6</td><td>5</td></tr>
<tr><th>01</th><td>9</td><td>8</td><td>7</td><td>4</td></tr>
<tr><th>11</th><td>0</td><td>1</td><td>2</td><td>3</td></tr>
<tr><th>10</th><td>15</td><td>12</td><td>13</td><td>14</td></tr>
</table>
И, ИЛИ, НЕ
код с дополнением до 2
в бинарный код
</td>
</tr>
</table>

ЛАБОРАТОРНАЯ РАБОТА № 4

ОДНОРАЗОВЫЕ СУММАТОРЫ

Цель работы: изучение основ проектирования и принципа работы полусумматора и одноразрядного сумматора на ИС.

Полусумматор

Полусумматором называется комбинационное логическое устройство с двумя входами и двумя выходами, функционирующее согласно табл. 1.

Таблица 1

Таблица истинности полусумматора

Входные переменные		Выходные переменные	
		Сумма	Перенос
A	B	S	P
0	0	0	0
0	1	1	0
1	0	1	0
1	1	0	1

Выходные сигналы полусумматора называются сигналами суммы переноса P соответственно. Из табл. 1 следует, что полусумматор функционирует в соответствии с

$$S = \bar{A}B + A\bar{B} = A \oplus B \quad (1)$$

$$P = AB \quad (2)$$

Структура, реализующая функции полусумматора в базисе И, ИЛИ, НЕ, и обозначения полусумматора согласно государственному стандарту, приведены на рис. 1.

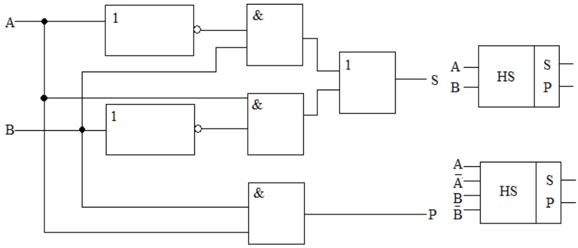


Рис. 1. Структура сумматора и его обозначения

Однозарядный комбинационный сумматор

Однообразным комбинационным сумматором называется комбинационное логическое устройство с тремя входами и двумя выходами и функционирующее согласно табл. 2.

На входах сумматора присутствуют переменные $I_{K,}$, Y_K и P_{K-1} , где P_{K-1} – сигнал переноса с предыдущего каскада. На выходах образуются сигналы суммы S_K и переноса P_K (сигнал переноса P_K подается в P_{K+1} разряд).

Входные переменные $I_{K,}$, Y_K и P_{K-1} воздействуют на сумматор в равной мере.

Таблица 2

Таблица истинности одноразрядного сумматора

Входы			Выходы	
			Сумма	Перенос
X_K	Y_K	P_{K-1}	S_K	P_K
0	0	0	0	0
0	0	1	1	0
0	1	0	1	0
0	1	1	0	1
1	0	0	1	0
1	0	1	0	1
1	1	0	0	1
1	1	1	1	1

Синтез сумматора в базисе И, ИЛИ, НЕ

Отметим выходные переменные сумматора на картах минтерма и произведем считывание (см. рис. 2.)

$$P_K = X_K Y_K + Y_K P_{K-1} + X_n P_{n-1} \quad (3)$$

$$S_K = X_K Y_K P_{K-1} + X_K \bar{Y}_K \bar{P}_{K-1} + \bar{X}_K \bar{Y}_K P_{K-1} \quad (4)$$

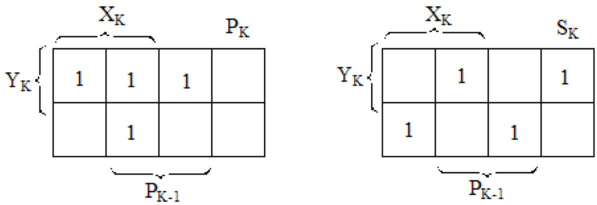


Рис. 2 - Карты минтермов для выходных функций одноразрядного комбинационного сумматора

Использование обычных методов не позволило существенным образом минимизировать выражения для входных функций сумматора. Поэтому воспользуемся приемом минимизации булевых функций.

Минимизация структуры сумматора в базисе И, ИЛИ, НЕ

Новый прием минимизации заключается в том, что для реализации булевой функции используется другая булева функция, заранее образованная.

В рассматриваемом случае выражения для P_K проще, нежели для S_K . Поэтому проведем упрощения функции S_K , используя в качестве четвертой входной переменной функцию P_K , т. е. считать, что функция $S_K = S_K(X_K, Y_K, P_{K-1}, P_K)$является функцией четырех переменных. Составим таблицу истинности для рассматриваемой функции – см. табл. 3.

При составлении таблицы будем считать, что комбинации X_K , Y_K, P_{K-1}, P_K , которые невозможны при нормальной работе сумматора, являются избыточными. Избыточные комбинации отмечаем крестиками.

Наносим выражение для $S_K= S_K(X_K, Y_K, P_{K-1}, P_K)$ на карту минтермов, отмечая крестиками избыточные комбинации (рис. 3а) и производим доопределение с целью удобства считывания (рис. 3б). Считывая, получаем структуру минимального сумматора:

$$S_k = X_k\bar{P}_k + P_{k-1}\bar{P}_k + Y_k\bar{P}_k + X_kY_kP_{k-1} = (X_k + Y_k + P_{k-1})\bar{P}_k + X_kY_kP_{k-1} \qquad (5)$$

$$P_k = X_kY_k + Y_kP_{k-1} + X_kP_{k-1} \qquad (6)$$

Таблица 3

Таблица истинности для функции $S_K= S_K(X_K, Y_K, P_{K-1}, P_K)$

X_K	0	0	0	0	0	0	0	0	1	1	1	1	1	1	1	1
Y_K	0	0	0	0	1	1	1	1	0	0	0	0	1	1	1	1
P_{K-1}	0	0	1	1	0	0	1	1	0	0	1	1	0	0	1	1
P_K	0	1	0	1	0	1	0	1	0	1	0	1	0	1	0	1
S_K	0	0	1	x	1	x	x	0	1	x	x	0	x	0	x	1

Название «минимальный сумматор» понимается в смысле канонической задачи минимизации. Однако, для техники ИС минимальная структура не является наилучшей, поскольку она не является однородной.

Минимизация структуры сумматора на ИС

Более однородная структура сумматора может быть реализована на основе полусумматоров, реализующих операцию «Исключительное ИЛИ». Выражение (6) доказывается с помощью карт минтермов на рис. 4. Нетрудно заметить, что $S_k=X_k \oplus Y_k \oplus P_{k-1}$.

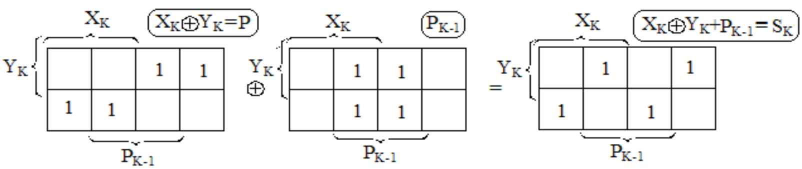


Рис. 4. Доказательство выражения (6) с помощью карт минтермов

Таким образом, сигнал суммы сумматора S_k может быть образован с помощью двух полусумматоров. Введя четвертую переменную $D= X_k \oplus Y_k$, таким образом образуем сигнал переноса $P_K= P_K(X_K, Y_K, P_{K-1}, D)$, т. е. воспользуемся приёмом, описанным ранее. Составляем таблицу истинности для P_K(табл. 4), отмечая крестиками избыточные комбинации, не встречающиеся при нормальной работе сумматора.

Наносим выражение $P_K= P_K(X_K, Y_K, P_{K-1}, D)$ на карту минтермов, отмечая крестиками избыточные комбинации (рис. 5а), и производим доопределение с целью удобства считывания (рис. 5б, в):

Таблица 4

Таблица истинности для функции $P_K= P_K(X_K, Y_K, P_{K-1}, D)$

X_K	0	0	0	0	0	0	0	0	1	1	1	1	1	1	1	1
Y_K	0	0	0	0	1	1	1	1	0	0	0	0	1	1	1	1
P_{K-1}	0	0	1	1	0	0	1	1	0	0	1	1	0	0	1	1
D	0	1	0	1	0	1	0	1	0	1	0	1	0	1	0	1
P_K	0	x	0	x	x	0	x	1	x	0	x	1	1	x	1	x

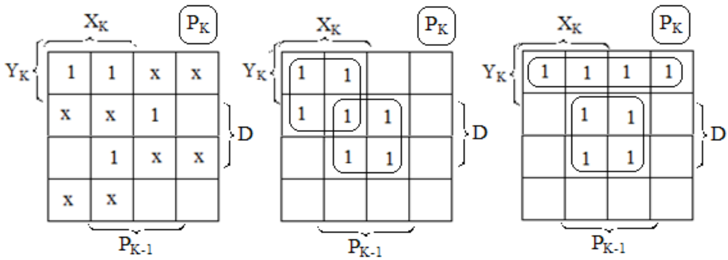


Рис. 5 – Карты минтермов для выходной функции сумматора $P_K= P_K(X_K, Y_K, P_{K-1}, D)$

После доопределения функции P_K может быть записана в различных формах:

$$P_k = X_k Y_k + P_{k-1} D = X_k Y_k + P_{k-1}(X_k \oplus Y_k) \tag{7}$$

$$P_k = Y_k \overline{D} + P_{k-1} D = Y_k \overline{X_k \oplus Y_k} + P_{k-1}(X_k \oplus Y_k) \tag{8}$$

$$P_k = \overline{D} X_k + P_{k-1} D = X_k \overline{X_k \oplus Y_k} + P_{k-1}(X_k \oplus Y_k) \tag{9}$$

Из выражений (6)-(9) следует, что однозарядный комбинационный сумматор может быть реализован на основе полусумматоров. На рис.6 показана структура сумматора, соответствующая алгоритму:

$$\left.\begin{aligned} S_k &= X_k \oplus Y_k \oplus P_{k-1} \\ P_k &= X_k Y_k + P_{k-1}(X_k \oplus Y_k) \end{aligned}\right\} \qquad (10)$$

Сумматор, построенный по (10), является минимальным для комплекта логических схем, содержащих полусумматор.

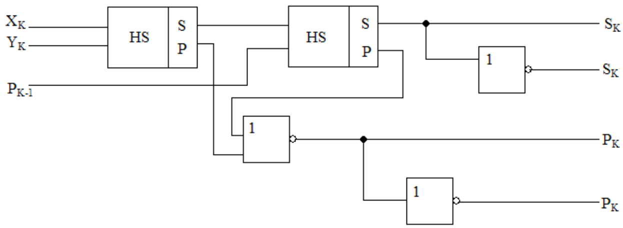


Рис. 6. Структура сумматора на основе полусумматоров

Экспериментальная часть.

Целью экспериментальной части является изучение характеристик полусумматора и сумматора, а также синтез однозарядного комбинационного сумматора.

Порядок выполнения.

1. На лицевой панели универсального лабораторного стенда собрать структуру, реализующую функции полусумматора,

2. Снять таблицу истинности, убедиться в правильном функционировании собранных устройств;

3. Снять таблицу истинности одного из сумматоров, входы и выходы которых выведены на лицевую панель универсального стенда;

4. Нарисовать структуру, реализующую функции сумматора по формулам (3) и (4);

5. На лицевой панели универсального лабораторного стенда реализовать полученную структуру;

6. Сняв таблицу истинности, убедиться в правильном функционировании собранной структуры;

7. Нарисовать структуру минимального сумматора в базисе И, ИЛИ, НЕ;

8. На лицевой панели универсального лабораторного стенда реализовать полученную структуру;

9. Сняв таблицу истинности, убедиться в правильном функционировании собранной структуры;

10. На лицевой панели универсального лабораторного стенда реализовать структуру сумматора на основе полусумматора.

Вопросы для отчета по лабораторной работе

1. Нарисуйте структуру, реализующую функцию полусумматора в указанном базисе;

2. Синтезируйте структуру сумматора в базисе И, ИЛИ, НЕ;

3. Проведите минимизацию структуры сумматора в базисе И, ИЛИ, НЕ.

4. Объясните суть метода, используемого при минимизации структур сумматора;

5. Нарисуйте структуру сумматора на основе полусумматора;

6. Синтезируйте структуру сумматора на основе полусумматора однокристального, в виде интегральной схемы 155 серии;

7. Преимущества сумматора на основе полусумматора по сравнению с минимизированным сумматором в базисе И, ИЛИ, НЕ;

8. Сравните различные структуры сумматоров, выполненных на основе базовых логических элементов различных серий;

БИБЛИОГРАФИЧЕСКИЙ СПИСОК

1. Джонсон Г.В. Конструирование высокоскоростных цифровых устройств: начальный курс чёрной магии. : пер. с англ. / Г.В. Джонсон, М. Грэхем. – М. : Издательский дом «Вильямс», 2006. – 624 с. : ил.
2. Угрюмов Е.П. Цифровая схемотехника. / Е.П. Угрюмов – СПб.: БХВ-Петербург, 2004. – 528 с. : ил..
3. Хоровиц П. Искусство схемотехники: в 3-х томах: Т. 1. пер. с англ. / П. Хоровиц, У. Хилл – 4-е изд., перераб. и доп.– М.: Мир, 1993.–413 с., ил.
4. Токхейм Р. Основы цифровой электроники: пер. с англ. / Р. Токхейм – М. : Мир, 1988. 392 с., ил.
5. Лобанов В. Азбука разработчика цифровых устройств / В. Лобанов. – М.: Горячая линия-Телеком, 2001., 192 с., ил.

СОДЕРЖАНИЕ

МЕТОДИЧЕСКИЕ УКАЗАНИЯ

к выполнению лабораторных работ № 1 – 4
по дисциплине «Схемотехника»
для студентов направления
210100.62 «Электроника и наноэлектроника»,
профиля «Микроэлектроника и твердотельная электроника»
очной формы обучения

Составители:
Арсентьев Алексей Владимирович
Плотникова Екатерина Юрьевна

В авторской редакции

Компьютерный набор А.В. Арсентьева, Е.Ю. Плотниковой

Подписано к изданию 31.10.2013.
Уч.-изд. л. 2,2.

ФГБОУ ВПО “Воронежский государственный технический университет”
394026 Воронеж, Московский просп., 14