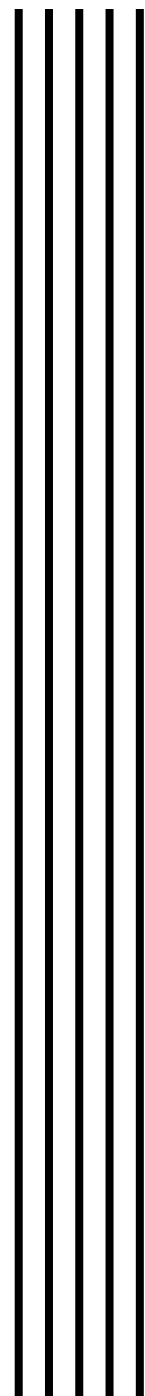


Министерство образования и науки Украины

Севастопольский национальный технический университет



АВТОМАТИЗАЦИЯ ПРОЕКТИРОВАНИЯ КОМБИНАЦИОННЫХ СЕТЕЙ

Методические указания
к практическим и лабораторным
занятиям
по дисциплине
**"ТЕХНОЛОГИЯ ПРОЕКТИРОВАНИЯ
КОМПЬЮТЕРНЫХ СИСТЕМ"**
для студентов дневной формы обучения
направления 6.050102 "Компьютерная инженерия"

Севастополь
2014

Автоматизация проектирования комбинационных сетей. Методические указания к выполнению практических и лабораторных занятий по дисциплине "Автоматизация проектирования вычислительных систем" для студентов дневной формы обучения направления 6.050102 "Компьютерная инженерия"/ составители А.И. Тертычный, В.И.Шевченко.- Севастополь: Изд-во СевНТУ, 2014.- 26 с.

В методических указаниях кратко изложены теоретические сведения по методам и алгоритмам проектирования комбинационных сетей. Рассматриваются примеры представления данных в программных системах проектирования комбинационных сетей. Приведены варианты заданий для выполнения лабораторных работ студентов.

Методические указания рассмотрены и утверждены на заседании кафедры кибернетики и вычислительной техники, протокол № 11 от 25 июня 2014 г.

Рецензент:

Мащенко Е. Н., к.т.н., доцент кафедры кибернетики и вычислительной техники.

Допущено учебно-методическим центром СевНТУ в качестве методических указаний.

СОДЕРЖАНИЕ

1. Краткие теоретические сведения.....	4
1.1 Проектирование комбинационных сетей	4
1.2 Задача синтеза ПЛМ	6
1.3 Проектирование комбинационных сетей на двухъярусных ПЛМ	7
2. Практическое занятие №1	9
2.1 Порядок проведения занятия	9
2.2 Определение способа кодирования данных	10
2.3 Разработка программной системы	10
2.4 Описание форматов представления данных	11
3. Практическое занятие №2	16
4. Практическое занятие №3	19
5. Лабораторная работа № 1. Разработка модуля автоматизации проектирования ПЛМ	22
6. Лабораторная работа № 2. Разработка модуля автоматизации проектирования КС	24
БИБЛИОГРАФИЧЕСКИЙ СПИСОК.....	26

1. КРАТКИЕ ТЕОРЕТИЧЕСКИЕ СВЕДЕНИЯ

Под комбинационной схемой понимается такая схема, у которой значения сигналов на выходе в каждый момент времени однозначно определяется только комбинацией значений сигналов на ее входе. В качестве примера комбинационной схемы можно привести разрядные шифраторы, дешифраторы, преобразователи кодов и другие схемы, не имеющие элементов памяти. Под комбинационной схемой понимается устройство, имеющее m входов и n выходов.

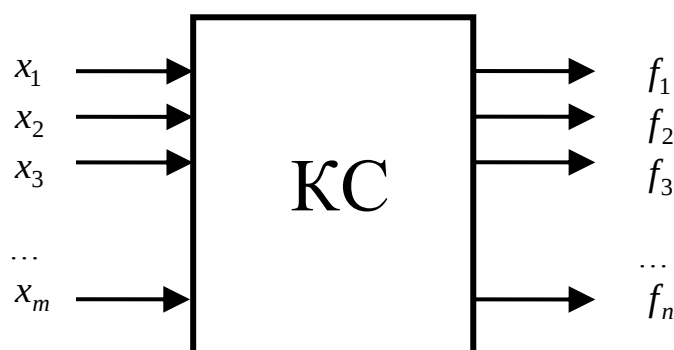


Рисунок 1 – Структурная схема комбинационной схемы

Задачей комбинационной схемы является преобразование $X \rightarrow F$, отображение множества $X = \{x_1, \dots, x_m\}$ во множество $F = \{f_1, \dots, f_n\}$. Синтез комбинационной схемы происходит в следующей последовательности:

- 1) Определяют вид каждой функции $f_1, \dots, f_n$ в виде таблицы истинности или какой-либо зависимости;
- 2) Выбирают базис логических элементов;
- 3) Представляют функции в выбранном базисе;
- 4) Минимизируют систему логических уравнений в выбранном базисе;
- 5) Строят функциональные схемы, используя заданную логику;
- 6) Строят принципиальную схему, затем монтажную.

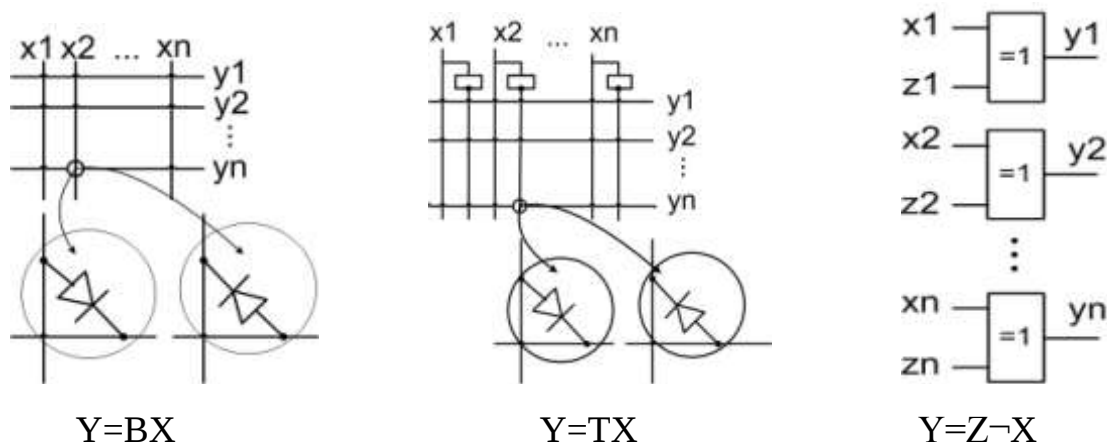
1.1 Проектирование комбинационных сетей

Комбинационные сети — это вид комбинационных схем, которые создаются на базе матричных элементов. К базовым структурам комбинационных сетей относят следующие элементы (рис.2):

- а) Двоичная матрица представлена в виде набора перпендикулярных линий, соответствующих входным сигналам x_i и выходным сигналам y_i , на пересечении которых ставится элемент связи — диод.

б) Троичная матрица представлена в виде набора перпендикулярных линий, соответствующих входным сигналам x_i с прямым и инверсным значением, и выходным сигналам y_i .

в) Векторная инверсия представлена в виде элементов «Исключающее ИЛИ».



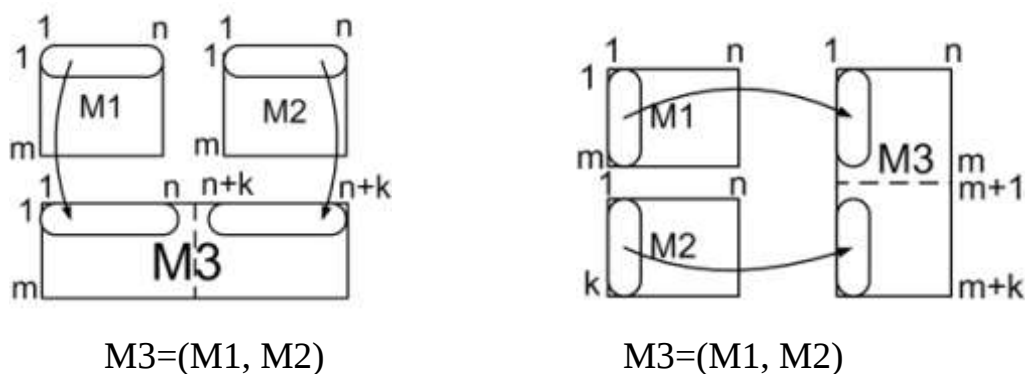
а) двоичная матрица

б) троичная матрица

в) векторная инверсия

Рисунок 2 – Базовые структуры комбинационных сетей

Комбинационная сеть — набор элементов, для каждого из которых определен набор входных сигналов $X = \{x_1, x_2, \dots, x_n\}$ и набор выходных сигналов $Y = \{y_1, y_2, \dots, y_m\}$ соединенных определенным образом для реализации функции отображения входных сигналов комбинационной сети $V = \{v_1, v_2, \dots, v_k\}$ в вектор выходных сигналов сети $U = \{u_1, u_2, \dots, u_p\}$. Для описания комбинационных сетей необходимо задать информацию об её структуре. Структура комбинационной сети задается перечнем элементов, перечнем связей между элементами и способ соединения элементов в матрице (кодировка).



А) Строчная конкатенация

Б) Столбцовая конкатенация

Рисунок 3 – Проектирование ПЛМ

Для описания структуры комбинационной сети, а также преобразования матриц используются матричные операции (рис.3):

а) Строчная конкатенация выполняется над матрицами с одинаковым числом строк. Результирующая матрица содержит конкатенацию соответствующих строк исходной матрицы.

б) Столбцевая конкатенация выполняется над матрицами с одинаковым числом столбцов. Результирующая матрица содержит столбцы, полученные конкатенацией соответствующих столбцов исходной матрицы.

1.2 Задача синтеза ПЛМ

Формулировка задачи: задана система булевых функций и параметры ПЛМ, число входов, выходов и внутренних линий (термов). Необходимо получить структуру ПЛМ с наименьшим числом внутренних линий, реализующих заданную функцию. В большинстве случаев система функций задаётся в виде совершенной ДНФ (отсутствуют интервалы, заданы только наборы) - обычно это множество единичных наборов M_1 для полностью определенных функции и M_1, M_x для частично определенной функции, где M_x – множество наборов, на которых функция не определена.

Задачу синтеза ПЛМ можно привести к математической задаче поиска минимального множества интервалов, которые покрывают единичные наборы на всех функциях, при этом не покрывают ни одного нулевого набора. Поиск минимального множества выполняется с помощью решения задачи о кратчайшем покрытии (содержит минимальное число строк, включенных в решение). Для получения кратчайшего покрытия используют метод ветвей и границ. Представленный метод позволяет получить оптимальное решение.

Точные методы гарантируют получение оптимального решения, но требуют значительных вычислительных ресурсов, поэтому в автоматизации проектирования широко применяются эвристические методы, позволяющие получить решение, приближенное к оптимальному.

Рассмотрим приближенный метод, основанный на построении дерева поиска решений. Основная идея метода соответствует методу Барти-Полянского. В методе реализуется операция расширения интервала, которая выполняется с учётом принадлежности интервала нескольким функциям. Операция может выполнять расширение интервала по переменной и расширять интервал по функции.

При расширении интервала по переменной, интервал должен расширяться по этой переменной на всех функциях интервала. Процесс формирования множества интервалов продолжается до полного покрытия всех единичных наборов. В процессе решения выделяются следующие операции:

1. Выбор первого единичного набора для определения начального интервала. Выбирается набор с наименьшим числом соседей.
2. Расширение интервала по переменной
3. Расширение интервала на функцию.
4. Запись интервала во множество.

Критерием при выполнении операций 2 и 3 является максимальное число покрываемых единичных наборов.

Для решения задачи исходная функция представляется в виде совершенной ДНФ, состоящей из множества единичных M_1 и множества неопределенных M_x наборов.

Метод требует меньшего объёма памяти для хранения данных и меньшего объёма вычислений по сравнению с точным методом, так как не решается задача построения таблицы покрытий и вычисления кратчайшего покрытия. Данный метод приближённый, поэтому не гарантируется получение оптимального решения.

1.3 Проектирование комбинационных сетей на двухъярусных ПЛМ

При проектировании комбинационных сетей (КС) наиболее широко используются двухъярусные ПЛМ, которые позволяют реализовать систему булевых функций, представленную в виде д.н.ф. или к.н.ф. ПЛМ состоит из двух матриц T , реализующая операцию конъюнкции (дизъюнкции) и B , реализующая операцию дизъюнкции (конъюнкции) (рис. 4).

Формулировка задачи. Задана система булевых функций в виде кодировки двухъярусной ПЛМ (матрица сети) и параметры элементарной матрицы: число входов, выходов и внутренних линий. Необходимо получить структуру, реализующую заданную функцию.

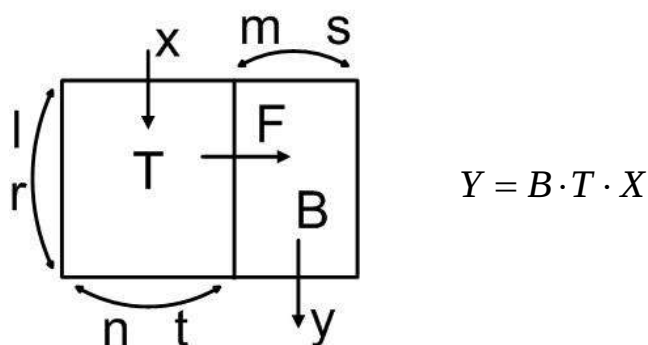


Рисунок 4 – Структура двухъярусной ПЛМ

Обозначим для матрицы сети: число входов – n , число выходов – m , число внутренних линий – l , а для элементарной матрицы: число входов – t , число выходов – s , число внутренних линий – r .

Рассмотрим процесс проектирования комбинационной сети при различных отношениях размерностей матрицы сети и элементарной матрицы (ПЛМ):

1) Случай, когда число входов, выходов и внутренних линий матрицы сети меньше или равно числу входов, выходов и внутренних линий элементарной матрицы соответственно: $n \leq t, m \leq s, l \leq r$. В этом случае МС помещается в одну элементарную матрицу, следовательно, сеть реализуется из одной ПЛМ, закодированной в соответствии с исходной матрицей. Число матриц P в структуре будет равно 1.

2) Случай, когда число входов и внутренних линий матрицы сети меньше или равно числу входов и внутренних линий элементарной матрицы, а число выходов

матрицы сети больше числа выходов элементарной матрицы: $n \leq t, m > s, l \leq r$. Для синтеза сети применяется метод простой декомпозиции (рис.5), с помощью которого матрица B разбивается на фрагменты с s выходами: $U = (U_1, U_2, \dots, U_p)$, $B = (B_1, B_2, \dots, B_p)$. Матрица T во всех ПЛМ одинакова. Число матриц в структуре будет $P = \left\lceil \frac{m}{s} \right\rceil$. Структура КС представлена на рисунке 6.

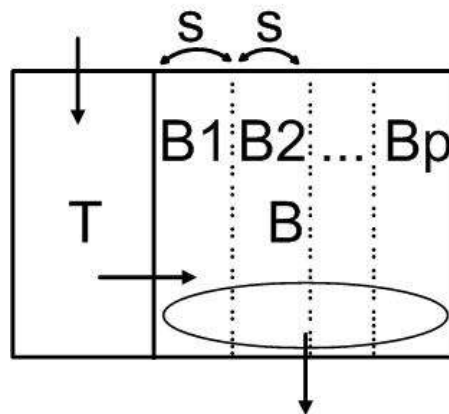


Рисунок 5 – Декомпозиция матрицы сети

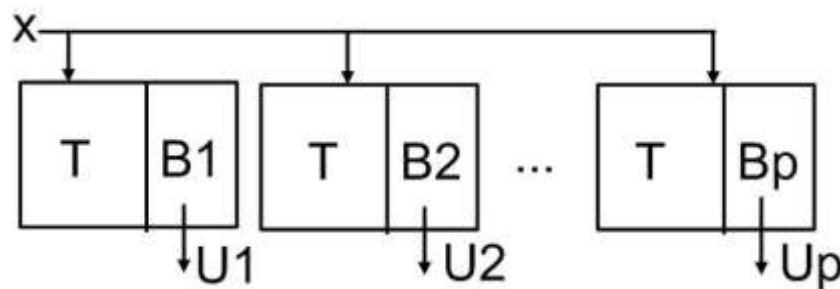


Рисунок 6 – Структура комбинационной сети

3) Случай, когда число входов и выходов матрицы сети меньше или равно числу входов и выходов элементарной матрицы, а число внутренних линий матрицы сети больше числа внутренних линий элементарной матрицы: $n \leq t, m \leq s, l > r$. В данном случае ограничения позволяют применить метод параллельной декомпозиции для матрицы B . При декомпозиции необходимо учитывать, что входам матрицы соответствуют строки, а выходам – столбцы. Следовательно, условие применения метода параллельной декомпозиции формулируется следующим образом: ранг столбца матрицы B не должен превышать числа внутренних линий r матрицы. В этом случае сеть получается применением метода параллельной декомпозиции к матрице B , получение множества матриц $(B_1, B_2, \dots, B_p)$ и формирование к каждой матрице B_i соответствующей матрицы T_i . Выходной вектор U равен: $U = (u_1, u_2, \dots, u_p)$. Число матриц в структуре будет $P = 2 \dots m$. Структура КС представлена на рисунке 7.

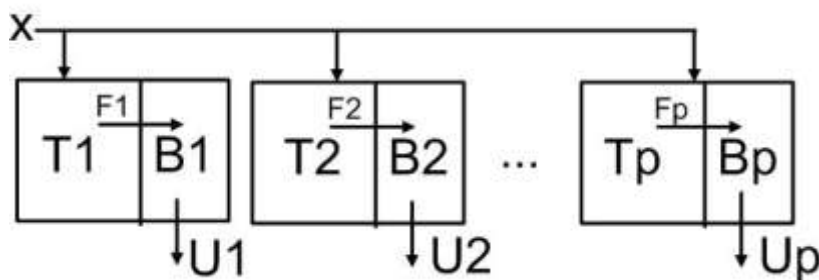


Рисунок 7 – Структура комбинационной сети

4) Случай, когда число входов матрицы сети меньше или равно числу входов элементарной матрицы, число выходов матрицы сети больше числа выходов элементарной матрицы, а число внутренних линий матрицы сети больше числа внутренних линий элементарной матрицы: $n \leq t, m > s, l > r$. Число матриц в структуре

будет равно $P = \left\lceil \frac{m}{s} \right\rceil \dots m$ Схема комбинационной сети аналогична схеме из пункта 3.

5) В случаях, когда число входов матрицы сети больше числа входов элементарной матрицы $n > t$ применяется модифицированный метод параллельной декомпозиции, основанный на одновременном формировании матриц T и B для каждой ПЛМ. Для решения задачи создается дополнительная матрица A связи входных переменных и функций, в которой строки соответствуют функциям, а столбцы - переменным. Единичное значение элемента матрицы указывает на использование переменной в соответствующей функции. Выполняется одновременная декомпозиция матриц B и A выбором функции y_i с учетом ограничений для каждой матрицы и совместных критериев. После формирования матрицы B_i выполняется формирование соответствующей матрицы T_i Структура КС представлена на рисунке 7.

2. ПРАКТИЧЕСКОЕ ЗАНЯТИЕ №1

Тема: «Проектирование структуры программной системы синтеза ПЛМ»

Цель: ознакомиться с основными элементами структуры программной системы синтеза ПЛМ и форматами представления данных для этих структур.

Постановка задачи: требуется разработать структуру программной системы синтеза ПЛМ, определить структуры и форматы представления данных.

2.1 Порядок проведения занятия

1. Проверка усвоения основных теоретических положений изученного материала.
2. Решение задачи декомпозиции процесса синтеза ПЛМ.
3. Разработка структуры системы синтеза ПЛМ.
4. Анализ полученной структуры.
5. Выбор и обоснование структур данных.
6. Составление спецификаций для программных модулей системы.

При проектировании программной системы необходимо учитывать следующие требования:

1. Программная система должна обрабатывать систему булевых функций, содержащих до шестнадцати функций и до шестнадцати переменных.
2. Исходные данные: описание списка функций, результаты работы программы, содержащие кодировку ПЛМ, представленную в виде графа, текста, программы.
3. Передача данных между модулями осуществляется посредством двоичных файлов.

2.2 Определение способа кодирования данных

В процессе преобразования систем булевых функций необходимо представить информацию о функции в виде множества наборов M_1 и M_x или интервалов. Набор представляется в виде битового вектора, переменная в наборе может принимать два значения «0» или «1», поэтому для ее представления достаточно одного бита. Для представления интервала можно использовать несколько способов.

В интервале переменная может присутствовать в трёх вариантах: входить в интервал с прямым значением – «1», входить в интервал с инверсным значением – «0» и являться внутренней переменной (не входить в интервал) – «-». Для кодирования этих вариантов используется два бита: «0» - 10; «1» - 01; «-» - 11.

Два основных способа кодирования:

1. Последовательное кодирование
2. Параллельное кодирование (регистровое представление).

В последовательном коде интервал представляется с помощью одного битового вектора и для каждой переменной выделяется по два соседних бита. При параллельном кодировании интервал задается двумя битовыми векторами, и переменные кодируются двумя битами, расположенными на одной позиции в разных векторах.

2.3 Разработка программной системы

Определим основные операции, выполняемые в процессе проектирования ПЛМ и разработаем структуру программной системы автоматизированного проектирования ПЛМ. Структура программной системы представлена на рисунке 8. Определим основные блоки данных, способы и форматы их представления.

В процессе проектирования ПЛМ можно выделить следующие операции:

- Ввод описания системы булевых функций;
- Формирование системы интервалов;
- Построение таблицы покрытий;
- Решение задачи о покрытии и формирование минимального списка интервалов;
- Формирование минимизированной системы булевых функций;
- Формирование кодировки ПЛМ.

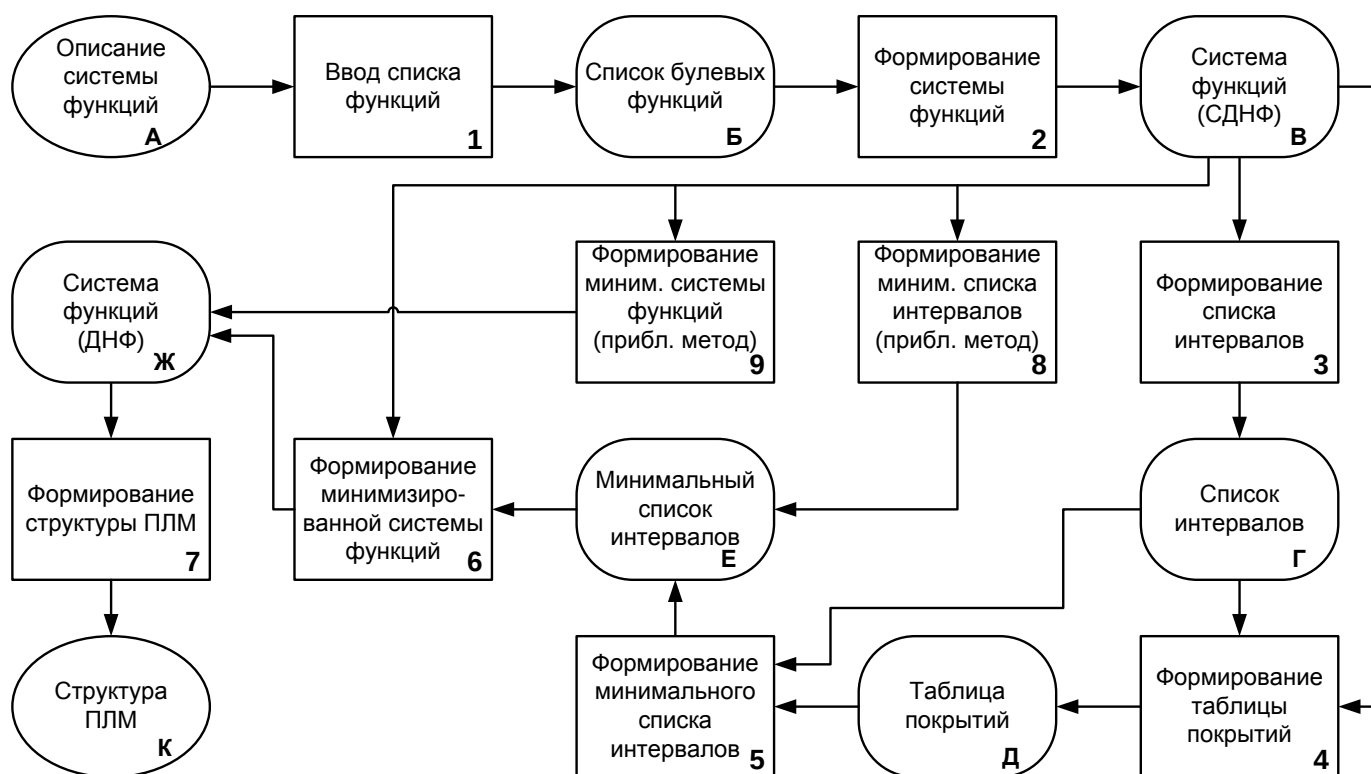


Рисунок 8 – Схема программной системы

Для реализации процесса проектирования ПЛМ необходимы следующие блоки данных:

Блок А. Содержит описание системных функций в виде списка функций: каждая функция системы представлена в виде описания отдельной функции. Данные представляются в текстовом файле.

Блок Б. Содержит список функций, имеет структуру, аналогичную текстовому файлу блока А, но все данные представлены в двоичном формате.

Блок В. Содержит описание системы булевых функций в виде совершенной ДНФ. Данные представляются в двоичном формате.

Блок Г и Е. Содержит список интервалов и минимальный список интервалов. Данные представляются в двоичном формате.

Блок Д. Содержит таблицу покрытий для формирования минимального списка интервалов. Данные представляются в двоичном формате.

Блок Ж. Содержит минимизированную систему функций в виде ДНФ. Данные представляются в двоичном формате.

Блок З. Содержит описание структуры ПЛМ. Данные представляются в текстовом формате. Кодировка ПЛМ представлена в виде рисунка, текста, программы.

2.4 Описание форматов представления данных

В программной системе передача данных между модулями осуществляется посредством файлов. Рассмотрим форматы представления данных в файлах. Для описания форматов данных введем следующие обозначения: <ЧП> - число пере-

менных в функции, $\langle \text{ЧФ} \rangle$ - число функций, $\langle \text{ЧИ} \rangle$ - число интервалов, $\langle \text{ЧН1} \rangle$ - число единичных наборов, $\langle \text{ЧНХ} \rangle$ - число неопределенных наборов. При представлении целых чисел в двоичном файле для каждого числа выделяем по два байта.

Рассмотрим форматы представления данных для функций на примере, приведенном на рисунке 9.

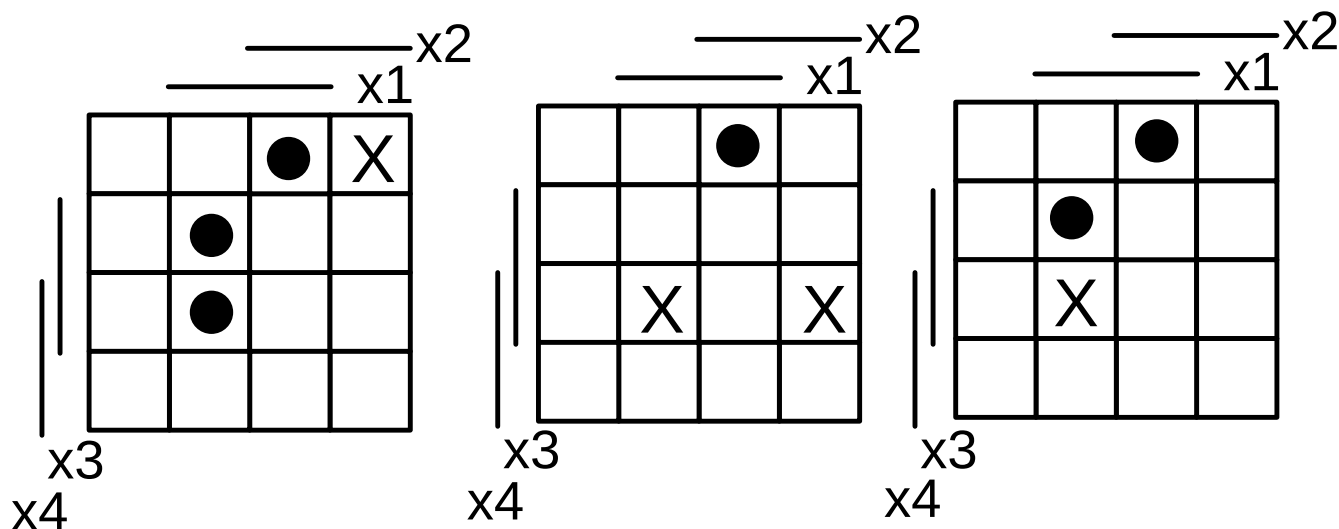


Рисунок 9 – Система булевых функций

Блок А. Содержит описание системы булевых функций в виде последовательности единичных и неопределенных наборов для каждой функции. Данные в файле имеют следующую структуру: для системы булевых функций задается число переменных и число функций, а для каждой функции задается число единичных наборов, число неопределенных наборов и коды единичных и неопределенных наборов. Формат файла и его содержимое для примера (рис.9) приведены на рис. 10.

$\langle \text{ЧП} \rangle$	$\langle \text{ЧФ} \rangle$	4	3
$\Phi_1 \left\{ \begin{array}{l} \text{ЧН1} \\ \text{Коды} \end{array} \right.$	$\left\{ \begin{array}{l} \text{ЧНХ} \\ \text{Н1} \\ \text{НХ} \end{array} \right.$	3	1
		1	0
		1	1
		1	0
		1	0
...		0	1
		0	0
$\Phi_{\text{чф}} \left\{ \begin{array}{l} \text{ЧН1} \\ \text{Коды} \end{array} \right.$	$\left\{ \begin{array}{l} \text{ЧНХ} \\ \text{Н1} \\ \text{НХ} \end{array} \right.$	1	2
		1	1
		0	0
		...	

Рисунок 10 – Описание списка функций

Блок Б. Содержит список функций. Имеет структуру, аналогичную блоку А, но все данные представляются в двоичном формате. Все числа занимают по два байта, каждый код набора так же занимает два байта, так как набор может содержать 16 переменных, а каждая переменная может принимать два значения: «0» или «1».

Данные файла в двоичном формате:

04	03
00	00
03	01
00	00
00	
00	...

Блок В. Содержит систему булевых функций в виде совершенной ДНФ. Структура файла с СДНФ представлена на рисунке 11А.

< ЧП > < ЧФ > < ЧН1 > < ЧНХ >
 [М1]
 [М2]
 [М3]
 [М4]

А) Структура файла с СДНФ

ЧН1 { $\overbrace{[M1]}^{ЧП}$ $\overbrace{[M3]}^{ЧФ}$ }
ЧНХ { $\overbrace{[M2]}^{ЧП}$ $\overbrace{[M4]}^{ЧФ}$ }

Б) Размерность матриц

Рисунок 11 – Система функций в виде совершенной ДНФ

В файле задается число переменных, число функций, общее количество единичных и неопределенных наборов. Затем следуют матрицы М1, М2, М3, М4. Матрица М1 – общий список единичных наборов, М2 – общий список неопределенных наборов, М3, М4 – список битовых векторов принадлежности наборов функциям, в которых 1 в i – ой позиции указывает что набор входит в функцию i. Элементы строк матриц и числовые параметры представляются двумя байтами. Размерность матриц можно представить по следующей схеме рисунка 11Б. Пример представления системы булевых функций в виде СДНФ приведен на рис.12

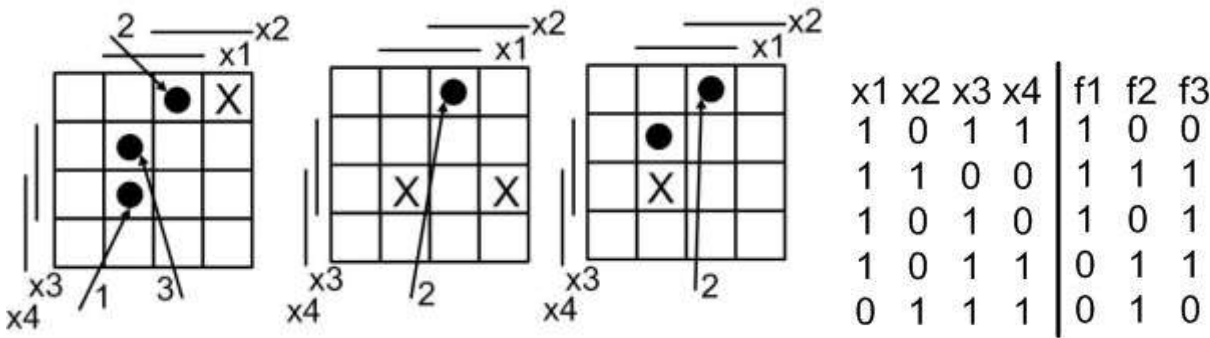


Рисунок 12 – Пример представления системы булевых функций в СДНФ

Блоки Г и Е. Содержит список интервалов и минимальный список интервалов. Структура файла и размерность списка приведены на рисунке 12.

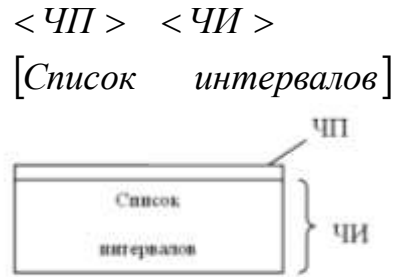
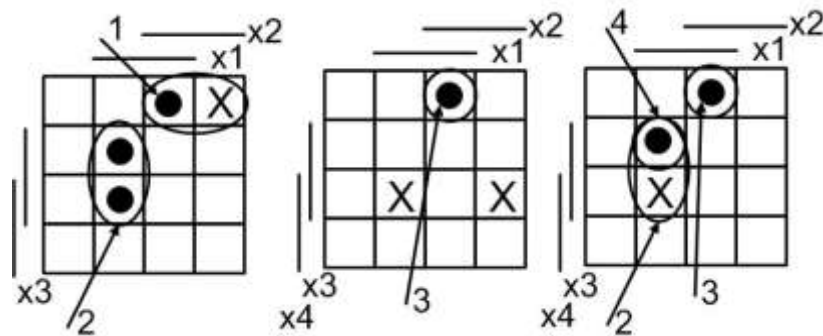


Рисунок 13 – Структура файла со списком интервалов

Каждая переменная списка интервалов кодируется тремя состояниями, поэтому для переменной должны выделяться по два бита. Каждый интервал кодируется двумя шестнадцатитривиальными векторами A и B , в которой значение переменной x_i кодируется двумя битами a_i и b_i . Пример кодирования интервалов представлен на рисунке 14.



	X1	X2	X3	X4
1	–	1	0	0
2	1	0	1	–
3	1	1	0	0
4	1	0	1	0

а) список интервалов

1A	1	0	1	1
1A	1	1	0	0
2B	0	1	0	1
2B	1	0	1	1

б) кодирование интервалов

«10» – $\bar{X}$; «01» – X ; «11» – не входит в интервал

Рисунок 14 – Представление интервалов

Блок. Д. Содержит таблицу покрытий для формирования минимального списка интервалов. Таблица покрытий представлена в виде последовательности битовых векторов каждый из которых кодирует одну строку. Вектора должны быть выровнены на границу двух байт. Структура файла:

$\langle \text{Число строк} \rangle \quad \langle \text{Число столбцов} \rangle$
 $[\text{Таблица покрытий}]$

Блок Ж. Содержит описание системы булевых функций в виде ДНФ. Структура файла:

$$\begin{array}{c}
 \langle \times \ddot{I} \rangle \quad \langle \times \hat{O} \rangle \quad \langle \times \ddot{E} \rangle \\
 \text{чП} \qquad \text{чФ} \\
 \left\{ \begin{array}{|c|c|} \hline \text{МИ} & \text{МФ} \\ \hline \end{array} \right.
 \end{array}$$

Фрагмент системы функций для примера рис. 9 приведен на рисунке 15.

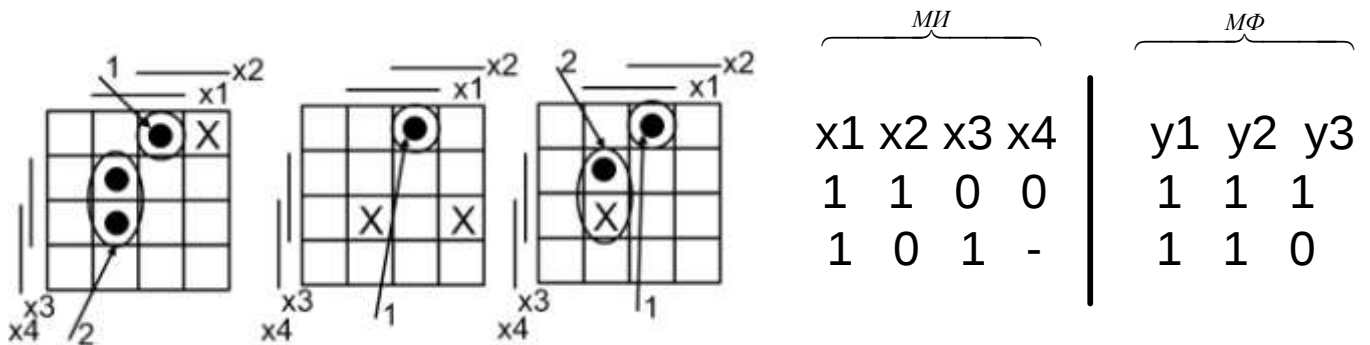


Рисунок 15 – Пример представления системы булевых функций в ДНФ

Блок 3. Содержит информацию о структуре ПЛМ заданного размера. Информация может быть представлена в трёх вариантах: представление в виде графического изображения (рисунок 16), текста, программы кодирования ПЛМ. Пример разбиения структуры ПЛМ размерностью 5 – 4 – 6 представлен на рис. 16

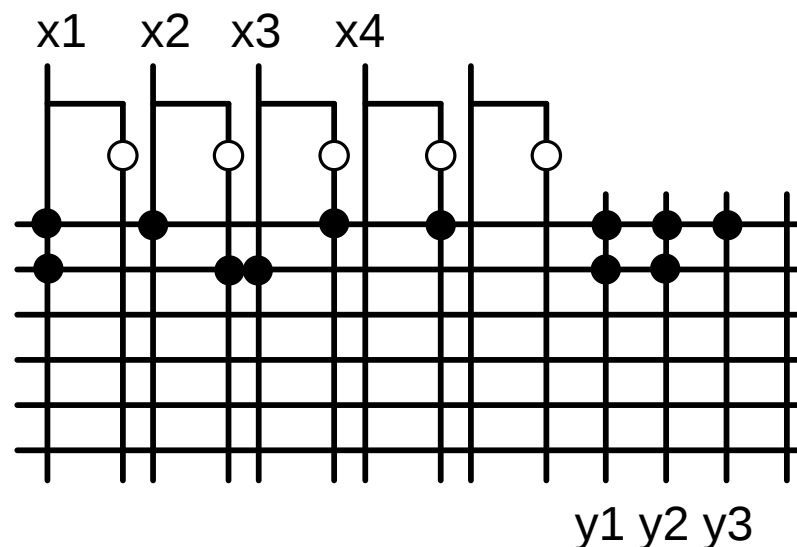


Рисунок 16 – Графическое изображение ПЛМ.

3. ПРАКТИЧЕСКОЕ ЗАНЯТИЕ №2

Тема: «Декомпозиционные методы синтеза комбинационных сетей. Метод параллельной декомпозиции»

Цель: приобрести навыки решения задач синтеза комбинационных сетей (КС) методом параллельной декомпозиции.

Постановка задачи: задана функция КС в виде кодирования ПЛМ и размеры элементарной матрицы. Необходимо получить структуру КС, реализующую заданную функцию.

Порядок проведения занятия:

1. Проверка усвоения основных теоретических положений изученного материала.
2. Решение задачи (формулировка, исходные данные, процесс решения).
3. Самостоятельное выполнение задания (варианты заданий приведены в приложении А).

Теоретические сведения:

Метод параллельной декомпозиции основан на последовательном формировании элементарных матриц, что соответствует перестановке строк и столбцов в матрице сети таким образом, чтобы образовались группы единичных элементов, размерностью не больше размерности элементарной матрицы.

Метод соответствует решению задач с помощью дерева поиска решений для получения любого результата лучевым способом при отсутствии тупиков. В этом случае дерево хранить в явном виде необязательно (так как в этом случае всегда получим результат).

Условия применения метода:

1. Максимальный ранг строки матрицы не должен превышать число входов элементарной матрицы L . Ранг строки матрицы R - число единичных элементов в строке, то есть это число переменных, входящих в функцию.

2. Метод обеспечивает построение одноуровневой сети.

В процессе решения задачи выделяют следующие операции:

1) Выбор первой строки нового множества (новой элементарной матрицы).

Критерий - максимальный ранг строки.

2) Выбор очередной строки множества.

Критерии выбора очередной строки множества:

1) Уровень максимально общих переменных между множеством и строкой;

2) Минимальное число новых переменных в строке по отношению к множеству.

При выполнении операции необходимо учитывать ограничения:

1) Число элементов множества не должно превышать число выходов элементарной матрицы;

2) Число переменных множества не должно превышать число входов элементарной матрицы.

Алгоритм метода параллельной декомпозиции:

1. Вычислить ранги строк.
2. Проверить условие применения метода, если число входов элементарной матрицы меньше максимального ранга, то метод не применим, переход к п.9
3. Если нет свободных (нераспределённых) строк, то переход к п.9
4. Выбор первой строки множества и исключение строки из множества свободных.
5. Если нет свободных строк, то переход к п.8
6. Для свободных строк вычисляется число общих и новых переменных.
7. Исключаются из рассмотрения строки с нарушением ограничения из 2.
8. Выбирается строка с наибольшим числом общих и минимальным числом новых переменных. Если такая строка есть, то добавляем её в множество, исключая из свободных и переход к п.5, иначе, переход к п.4. Если число элементов множества равняется k , то переход к п. 3, иначе к п.5.
9. Конец.

Рассмотрим синтез КС с использованием метода параллельной декомпозиции на примере. Размер элементарной матрицы – 6 входов, 4 выхода. Функция задана в виде матрицы сети размером 10×15 . Исходная матрица приведена на рис.17.

	a	b	c	d	e	f	g	h	i	j	k	l	m	n	o	R
1					1				1			1				3
2	1						1	1								3
3		1			1							1		1		4
4				1		1									1	3
5		1			1									1		3
6													1		1	2
7	1	1							1							3
8	1								1			1				3
9										1	1					2
10			1				1									2

Рисунок 17 – Матрица КС

1. Вычисляем ранг строки. Для этого требуется подсчитать количество единиц в строке.
2. Проверяем условие применение метода: число входов – «6» больше максимального ранга – «4», следовательно, метод применим.
3. Выбираем первую строку первого множества. Выбирается строка 3, поскольку она имеет максимальный ранг 4 (рис. 19а). Строка включается в множество – $M1: \{b, e, l, n\}$.
4. Выполняем включение очередной строки множества. Для этого, рассчитаем число общих (О) и новых (Н) переменных («Расчет 1» на рис. 18).

5. Исключаем из рассмотрения строки, нарушающие ограничение 2, т.е. в которых число новых переменных больше числа свободных входов (в данном случае 2). На рис. 18 эти элементы отмечены штриховкой.

6. Используя критерии максимума общих и минимума новых элементов выбираем строку 5 и включаем ее в множество (рис. 19б).

7. Следуя алгоритму в первое множество будут включены строки 1 и 7 (рис. 19в). Сформированное множество представлено на (рис. 19г).

8. Подобным образом формируются множества M2 и M3, представленные на рис.20.

№	Расчет 1		Расчет 2		Расчет 3		R	Расчет 4		Расчет 5		Расчет 6		R	Расчет 7		Расчет 8	
	О	Н	О	Н	О	Н		О	Н	О	Н	О	Н		О	Н	О	Н
1	2	1	2	1	-	-	-	-	-	-	-	-	-	-	-	-	-	-
2	0	3	0	3	0	3	3	-	-	-	-	-	-	-	-	-	-	-
3	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
4	0	3	0	3	0	3	3	0	3	0	3	0	3	3	-	-	-	-
5	3	0	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
6	0	2	0	2	0	2	2	0	2	0	2	0	2	2	1	1	-	-
7	1	2	1	2	1	2	-	-	-	-	-	-	-	-	-	-	-	-
8	1	2	1	2	1	2	3	1	2	1	2	-	-	-	-	-	-	-
9	0	2	0	2	0	2	2	0	2	0	2	0	2	2	0	2	0	2
10	0	2	0	2	0	2	2	1	1	-	-	-	-	-	-	-	-	-

Рисунок 18 – Расчеты по алгоритму параллельной декомпозиции

M1: 3(b, e, l, n)

M1: 3(b, e, l, n)

M1: 3(b, e, l, n)

M1: 3(b, e, l, n)

5(b, e, l, n)

5(b, e, l, n)

5(b, e, l, n)

1(b, e, l, n, i)

1(b, e, l, n, i)

7(b, e, l, n, i, a)

а)

б) Расчет 1

в) Расчет 2

г) Расчет 3

Рисунок 19 – Формирование матрицы элементов по алгоритму параллельной декомпозиции

9. Выполняем формирование и кодировку элементарных матриц. Набор строк элементарной матрицы выбирается из множества, а набор переменных матрицы выбирается из последней строки множества. Затем из исходной матрицы переносятся значения соответствующих элементов. Кодировка матриц и графическое изображение приведены на рис. 20 и 21.

M1: 3 (b, e, l, n)
5 (b, e, l, n)
1 (b, e, l, n, i)
7 (b, e, l, n, i, a)

M2: 2 (a, g, h)
10 (a, g, h, c)
8 (a, g, h, c, l)

M3: 4 (d, f, o)
6 (d, f, o, m)
9 (d, f, o, m, j, k)

M1:

	b	e	l	n	i	a
3	1	1	1	1		
5	1	1				
1		1	1		1	
7	1				1	1

M2:

	a	g	h	c	i	l
2	1	1	1			
10		1		1		
8	1				1	1

M3:

	d	f	o	m	j	k
4	1	1	1			
6			1	1		
9					1	1

Рисунок 20 – Кодировка элементарных матриц

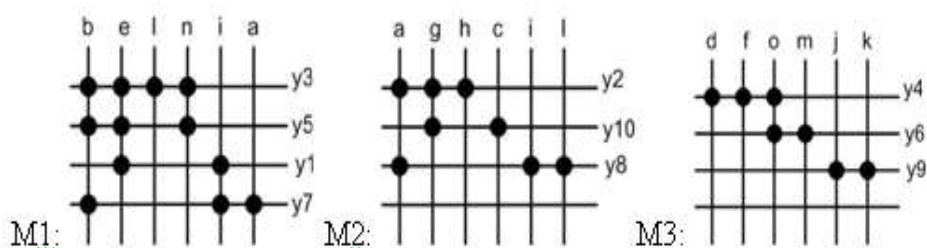


Рисунок 21 – Графическое изображение КС

Задания для самостоятельной работы

Матрица 1

	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O
1				1					1			1			
2	1						1	1							
3		1			1							1		1	
4				1		1									1
5		1			1									1	
6												1			1
7	1	1							1						
8	1								1			1			
9										1	1				
10			1				1								

Матрица 2

	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O
1		1	1									1			
2	1					1	1							1	
3				1							1	1			
4	1					1	1							1	
5		1				1		1							
6		1	1	1							1				
7													1		1
8					1						1	1			
9									1	1					
10			1		1						1				

4. ПРАКТИЧЕСКОЕ ЗАНЯТИЕ №3

Тема: «Декомпозиционные методы синтеза КС. Комбинированный метод синтеза КС»

Цель: приобрести навыки решения задач синтеза КС комбинированным методом.

Постановка задачи: задана функция КС в виде кодирования ПЛМ и размеры элементарной матрицы. Необходимо получить структуру КС, реализующую заданную функцию.

Порядок проведения занятия:

1. Проверка усвоения основных теоретических положений изученного материала.
2. Решение задачи (формулировка, исходные данные, процесс решения).

3. Самостоятельное выполнение задания (варианты заданий приведены в приложении А).

Теоретические сведения:

Комбинированный метод применяется, когда число строк с рангом, превышает число входов элементарной матрицы, составляя лишь небольшую часть. Будем данные строки называть длинными. Суть метода заключается во временном исключении длинных строк, применении метода параллельной декомпозиции к оставшейся матрице, разбиении длинных строк на фрагменты таким образом, чтобы максимально использовать уже имеющуюся часть комбинационной сети и построении матриц второго уровня для реализации операции конкатенации фрагментов длинных строк.

Рассмотрим синтез КС с использованием комбинированного метода на примере. Размер элементарной матрицы – 5 входов, 4 выхода. Функция задана в виде матрицы сети размером 10×15. Исходная матрица приведена на рисунке 22.

	a	b	c	d	e	f	g	h	i	j	k	l	m	n	o	R
1		1	1									1				3
2	1					1	1		1	1				1		6
3				1							1	1				3
4	1						1	1						1		4
5		1				1		1		1			1		1	6
6		1	1	1							1					4
7													1		1	2
8					1						1	1				3
9									1	1						2
10			1		1						1					3

Рисунок 22 – Матрица КС

1. Вычисляем ранг строки.
2. Проверяем условие применение метода: число входов – «5» меньше максимального ранга – «6», следовательно, метод параллельной декомпозиции не применим, применяем комбинированный метод.
3. Временно исключаем из рассмотрения длинные строки с рангом превышающим число входов элементарной матрицы (2 и 5).
4. Для оставшейся матрицы применяем метод параллельной декомпозиции. В результате чего получаем множество, представленное на рис. 23.
5. Формируем матрицы элементов.

M1: 4(a, g, h, n)	M2: 6(b, c, d, k)	M3: 8(e, k, l)	M4: 7(m, o)
	1(b, c, d, k, l)	10(e, k, l, c)	9(m, o, i, j)

Задания для самостоятельной работы

Матрица 1

	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O
1							1			1					
2	1		1		1	1						1			
3		1		1											1
4		1							1						1
5		1		1					1						1
6											1	1			
7	1		1		1								1		
8	1					1		1					1		
9				1					1					1	
10		1		1						1					

Матрица 2

	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O
1						1	1								1
2	1		1								1		1		
3		1		1	1										
4						1									1
5	1		1										1		
6				1								1			
7	1							1							1
8								1			1				1
9									1	1					
10			1					1			1				

5. ЛАБОРАТОРНАЯ РАБОТА № 1. РАЗРАБОТКА МОДУЛЯ АВТОМАТИЗАЦИИ ПРОЕКТИРОВАНИЯ ПЛМ

Цель работы: приобретение практических навыков разработки элементов САПР автоматизации проектирования ПЛМ.

Требования к выполнению лабораторной работы:

При выполнении лабораторной работы необходимо выполнить требования, предъявляемые к процессу разработки САПР и его составных частей. К таким требованиям относятся:

- описание подхода решения;
- спецификации на структуры данных;
- спецификации процедур и функций.

Все указанные требования должны быть учтены при разработке программной части лабораторной работы. Разработка модуля ведётся бригадой, состоящей из нескольких человек.

Требования к разработке программных модулей:

1. Данные, описывающие объект (модуль) должны быть представлены в виде класса, который должен обладать функциональной полнотой данных.
2. При разработке форматов представление данных необходимо обосновать размеры и типы переменных.
3. При определении форматов представления данных необходимо стремиться к минимизации объёма памяти, но с учётом быстродействия обработки данных. Существуют различные способы представления данных.

Задание на лабораторную работу: необходимо разработать модуль системы автоматизации проектирования ПЛМ, реализующий заданную функцию и обеспечивающий совместное функционирование с другими модулями системы. Назначение

модулей и их взаимодействие описаны в разделе 2. Структура программной системы представлена на рис. 8.

Варианты заданий

№ варианта	Число студентов в бригаде	№ модуля	№ варианта	Число студентов в бригаде	№ модуля
1	1	1т	7	2	5к
2	2	1г	8	2	2
3	1	2	9	1	7т
4	3	3	10	1	7р
5	2	4	11	1	7п
6	1	5м	12	3	8

Некоторые модули программной системы имеют несколько вариантов реализации, которые имеют буквенное обозначение:

«т» – текстовое представление данных;

«г» – графическое представление информации;

«к» – решение задачи методом ветвей и границ;

«м» – решение задачи методом максимальной строки и минимального столбца.

Содержание отчета

1. Постановка задачи.
2. Краткое описание способа решения задачи.
3. Описание структур данных.
4. Спецификация на процедуры и функции.
5. Описание алгоритмов решения задачи.
6. Описание процедур и функций.
7. Текст программы.
8. Результаты тестирования.
9. Заключение (выводы).
10. Приложение.

Контрольные вопросы

1. Сформулируйте задачи синтеза ПЛМ.
2. В чем суть точного метода синтеза ПЛМ.
3. В чем суть приближенного синтеза ПЛМ.
4. В чем достоинства и недостатки последовательного и параллельного способов кодирования интервалов.

6. ЛАБОРАТОРНАЯ РАБОТА № 2. РАЗРАБОТКА МОДУЛЯ АВТОМАТИЗАЦИИ ПРОЕКТИРОВАНИЯ КС

Цель работы: приобретение практических навыков разработки элементов САПР автоматизации проектирования комбинационных сетей.

Требования к выполнению лабораторной работы:

При выполнении лабораторной работы необходимо выполнить требования, предъявляемые к процессу разработки САПР и его составных частей. К таким требованиям относятся:

- описание подхода решения;
- спецификации на структуры данных;
- спецификации процедур и функций.

Все указанные требования должны быть учтены при разработке программной части лабораторной работы. Разработка модуля ведётся бригадой, состоящей из нескольких человек.

Требования к разработке программных модулей

1. Данные, описывающие объект (модуль) должны быть представлены в виде класса, который должен обладать функциональной полнотой данных.

2. При разработке форматов представление данных необходимо обосновать размеры и типы переменных.

3. При определении форматов представления данных необходимо стремиться к минимизации объёма памяти, но с учётом быстродействия обработки данных. Существуют различные способы.

Задание на лабораторную работу: необходимо разработать программную систему автоматизации проектирования КС двухъярусных ПЛМ. Исходными данными для программы является функция комбинационной сети, заданная в виде кодировки 2-х ярусной ПЛМ и параметры элементарной матрицы (число входов, число выходов и число внутренних переменных). Программа должна выполнять построение структуры КС заданным методом с учетом указанных ограничений. Результатом работы программы является структура КС в виде последовательности закодированных ПЛМ с указанием входных и выходных сигналов для каждой ПЛМ. Входные и выходные данные представляются в текстовых файлах.

Работа выполняется студентами в составе бригады из двух человек. Один студент разрабатывает модуль интерфейса, реализующий ввод и подготовку исходных данных, и вывод результата работы программы. Второй – разрабатывает модуль решения задачи.

Задача решается методом параллельной декомпозиции, описанном в разделе 3, или методом факторизации. Основные положения метода факторизации соответствуют методу параллельной декомпозиции. Отличие заключается только в реализации выбора первой строки множества, критерием для которой является максимальная совместимость строками с еще нераспределенными строками.

Варианты заданий представлены в таблице, где приняты обозначения:

П – метод параллельной декомпозиции;

Ф – метод факторизации;

Д – метод простой декомпозиции;

У – ограничение числа выходов;

F – ограничение числа внутренних линий;

X – ограничение входов.

Варианты заданий

№ варианта	1	2	3	4	5	6	7	8	9
№ модуля	ПГУ	ФГУ	ПХУ	ФХУ	ПXF	ФXF	ПХ	ФХ	ДУ

Содержание отчета

1. Постановка задачи.
2. Краткое описание способа решения задачи.
3. Описание структур данных.
4. Спецификация на процедуры и функции.
5. Описание алгоритмов решения задачи.
6. Описание процедур и функций.
7. Текст программы.
8. Результаты тестирования.
9. Заключение (выводы).
10. Приложение.

Контрольные вопросы

1. Когда для проектирования КС возможно применение метода простой декомпозиции.
2. Когда для проектирования КС возможно применение метода параллельной декомпозиции.
3. Приведите условия применения метода параллельной декомпозиции при ограниченном числе входов.
4. Приведите условия применения метода параллельной декомпозиции при ограниченном числе внутренних линий.

БИБЛИОГРАФИЧЕСКИЙ СПИСОК

1. Закревский А.Д. Логический синтез каскадных схем / А.Д. Закревский. - М.: Наука, 1981. – 416 с.
2. Глушков В.М. Логическое проектирование дискретных устройств / В.М. Глушков, Ю.В. Капитонова, А.Т. Мищенко. - Киев: Наукова думка, 1987. - 264с.
3. Норенков И.П. Основы автоматизированного проектирования: Учеб. для вузов / И.П. Норенков – М.: Изд-во МГУ им. Н.Э.Баумана, 2002. – 336 с.
4. Страуструп Б. Язык программирования С++/ Б. Страуструп. – М: Бином, 2008. – 1104 с.