УЧРЕЖДЕНИЕ ОБРАЗОВАНИЯ
«МОГИЛЕВСКИЙ ГОСУДАРСТВЕННЫЙ ПОЛИТЕХНИЧЕСКИЙ КОЛЛЕДЖ»

УТВЕРЖДАЮ
Директор колледжа
_____ С.Н.Козлов
29.08.2023

ПРОГРАММИРУЕМЫЕ КОНТРОЛЛЕРЫ И ПРОМЫШЛЕННЫЕ СЕТИ

МЕТОДИЧЕСКИЕ РЕКОМЕНДАЦИИ
ПО ИЗУЧЕНИЮ УЧЕБНОГО ПРЕДМЕТА,
ЗАДАНИЯ НА ДОМАШНЮЮ КОНТРОЛЬНУЮ РАБОТУ № 1
ДЛЯ УЧАЩИХСЯ ЗАОЧНОЙ ФОРМЫ ПОЛУЧЕНИЯ ОБРАЗОВАНИЯ
ПО СПЕЦИАЛЬНОСТИ 2-53 01 05
«АВТОМАТИЗИРОВАННЫЕ ЭЛЕКТРОПРИВОДЫ»

2023

Автор: Антипенко Т.И., преподаватель учреждения образования «Могилевский государственный политехнический колледж»

Рецензент: Комоза Т.Ф., преподаватель учреждения образования «Могилевский государственный политехнический колледж»

Разработано на основе учебной программы по учебному предмету профессионального компонента учебного плана учреждения образования по специальности 2-53 01 05 «Автоматизированные электроприводы» для реализации образовательной программы среднего специального образования, обеспечивающей получение квалификации специалиста со средним специальным образованием, утвержденной директором колледжа, 2022

Обсуждено и одобрено на заседании
цикловой комиссии электротехнических предметов
Протокол № _____ от_____
Председатель цикловой комиссии
_____ Т.Ф.Комоза

Пояснительная записка

Основными целями изучения учебного предмета «Программируемые контроллеры и промышленные сети» являются формирование профессиональной компетентности будущих специалистов в области построения электрических схем, изучение элементной базы и схемотехники технических средств программного управления технологическим оборудованием производственных систем.

Задачей изучения учебного предмета «Программируемые контроллеры и промышленные сети» является развитие у учащихся алгоритмического мышлении и необходимых знаний в области архитектурных особенностей современной вычислительной техники.

Изучение программного материала учебного предмета основывается на знаниях, полученных учащимися при изучении учебных предметов «Математика», «Физика», «Информатика».

Знания и умения, полученные при изучении учебного предмета «Программируемые контроллеры и промышленные сети», являются необходимыми для усвоения учебного программного материала учебных предметов: «Системы программного управления», «Электропривод и электроавтоматика».

При выполнении лабораторных и практических работ учащиеся должны практически изучить основы функционирования устройств ЭВМ.

В результате изучения учебного предмета учащиеся должны знать на уровне представления:

- назначение и классификацию программируемых контроллеров;
- технических характеристики программируемых контроллеров;

знать на уровне понимания:

- функциональный и аппаратный состав программируемых контроллеров;
- алгоритмы функционирования программируемых контроллеров;
- принципы программирования программируемых контроллеров;

уметь:

- составлять простейшие программы для программируемых контроллеров;
- различать различные языки программирования для программируемых контроллеров;
- запрограммировать таймеры и счётчики.

Общие методические рекомендации по выполнению домашней контрольной работы № 1

Цель методических рекомендаций – помочь учащимся в выполнении домашней контрольной работы. В процессе изучения учебного предмета учащимся необходимо получить практические навыки построения электрических схем, изучение элементной базы и схемотехники технических средств программного управления технологическим оборудованием производственных систем.

Основным методом изучения учебного предмета является самостоятельная работа, которая должна проводиться в последовательности, предусмотренной программой учебного предмета, и быть обязательно систематической.

Задания на домашнюю контрольную работу разработаны в количестве 100 вариантов в соответствии с программой курса.

Вариант работы выбирается в соответствии с двумя последними цифрами шифра учащегося по таблице вариантов. Каждый вариант содержит 5 заданий (2 теоретических вопроса и 3 практических задания).

При выполнении заданий надо помнить следующее:

- работа выполняется в отдельной тетради или на листах А4;
- следует пронумеровать страницы и оставить на них поля не менее 3 см для замечаний преподавателя;
- на обложке работы должен быть прикреплен титульный лист утвержденного образца или аккуратно записаны все данные титульного листа: шифр, фамилия, имя, отчество учащегося, название учебного предмета и номер работы, номер учебной группы;
- работа должна быть выполнена чернилами одного цвета, аккуратно, разборчиво либо должен быть представлен машинописный вариант (шрифт 12-14);
- каждую задачу надо начинать с новой страницы;
- номера задач следует указывать перед условием;
- условия задач должны быть обязательно переписаны полностью в работу;
- решения задач должны сопровождаться краткими, но достаточно обоснованными пояснениями;
- ответы на теоретические вопросы должны быть осмыслены и продуманы.

В конце работы следует указать список используемых источников, которым пользовались, проставить дату выполнения работы и подпись.

Если в работе допущены недочеты или ошибки, то учащийся должен выполнить все указания преподавателя, сделанные в рецензии.

Домашняя контрольная работа должна быть выполнена в срок (в соответствии с учебным графиком). В период лабораторно-экзаменационной сессии работы на проверку не принимаются.

Критерии оценки домашней контрольной работы № 1

Домашняя контрольная работа, признанная преподавателем удовлетворительной и содержащая 75% положенного объема, оценивается отметкой «зачтено».

Домашняя контрольная работа оценивается отметкой «не зачтено», если:

- выполнена не по варианту;
- не раскрыто основное содержание одного теоретического вопроса и есть недочеты в остальных заданиях;
- не выполнено одно практическое задание и есть незначительные недочеты в остальных заданиях;
- есть существенные недочеты в нескольких заданиях.

Учебная программа учебного предмета

Введение

Цели и задачи учебного предмета. Промышленные сети и программируемые контроллеры. Термины и определения.

Литература: [3], с.11-25

Вопросы для самоконтроля

1 Перечислите цели и задачи учебного предмета. Назовите область применения полученных знаний.

2 Объясните роль ЭВМ в развитии технического прогресса.

3 Что называют промышленными сетями?

4 Что называют микропроцессорной техникой?

Раздел 1 Общие сведения о программируемых контроллерах

Тема 1.1 Назначение и классификация программируемых контроллеров

Понятие программируемый контроллер.

Классификация и назначение программируемых контроллеров.

Литература: [2], с. 8-13

Тема 1.2 Технические характеристики программируемых контроллеров

Структура ЭВМ: определение, схема, назначение и взаимодействие основных функциональных частей ЭВМ. Понятие о микро-процессорной системе (МПС) и микроконтроллере

Типы компьютеров по условиям эксплуатации, по производительности и характеристики использования

Литература: [2], с. 8-13

Вопросы для самоконтроля

1 Что называют архитектурой программируемых микроконтроллеров?

2 Что называют машиной Фон Неймана?

3 Что называют структурными схемами программируемых микроконтроллеров?

4 Что называют структурными схемами вычислительных систем?

5 Что называют алгоритмом, программой?

6 Назовите принципы программного управления.

7 Что называют структурой команды?

Раздел 2 Способы формализации работы электроприводов технологического оборудования

Тема 2.1 Формализация технологического цикла

Понятия: технологический цикл, формализация технологического цикла, дискретный автомат.

Литература: [6]

Тема 2.2 Циклограммы работы оборудования

Понятие “циклограмма работы оборудования”.

Литература: [5]

Вопросы для самоконтроля

1 Объясните понятие «циклограмма работы оборудования».

2 Что называют формализацией технологического цикла?

3 Охарактеризуйте аналоговые и дискретные сигналы.

4 Что называют техническим циклом?

Раздел 3 Состав программируемых контроллеров

Тема 3.1 Функциональный состав программируемых контроллеров

Понятие функциональный состав программируемых контроллеров.

Функциональный состав программируемых контроллеров: блок управления контроллером, четыре канала обмена данными в режиме ПДП, четыре канала обмена данными в режиме ПДП.

Литература: [3],с. 25-27

Тема 3.2 Аппаратный состав программируемых контроллеров

Понятие аппаратного состава программируемых контроллеров.

Аппаратный состав программируемых контроллеров: базовая панель, модуль центрального вычислительного устройства, коммуникационные модули, блок питания, модули ввода/вывода.

Литература: [3], с. 25-27

Тема 3.3 Алгоритмы функционирования программируемых контроллеров

Понятие алгоритмов функционирования.

Алгоритмы функционирования программируемых контроллеров, различные циклы, тестирование, формирование слов и т.д.

Литература: [4], с. 124-129

Вопросы для самоконтроля

1 Назовите виды систем счисления.

2 Охарактеризуйте алгоритмы функционирования программируемых контроллеров.

3 Охарактеризуйте функциональный состав программируемых контроллеров.

4 Охарактеризуйте аппаратный состав программируемых контроллеров.

5 Опишите формирование слов у программируемых контроллеров.

Раздел 4 Программное обеспечение программируемых контроллеров

Тема 4.1 Система команд программируемых контроллеров

Понятие «системы команд», «командный цикл», «метка», «директивы», «регистры общего и специального назначения».

Система команд программируемых контроллеров.

Литература: [3], с. 32-43

Тема 4.2 Обзор языков программирования программируемых контроллеров

Языки программирования ПЛК, стандарт IEC(МЭК)61131-3. Графические и текстовые языки программирования.

Среды программирования ПЛК на примере CoDeSys, Simatic Step 7, Arduino.

Литература: [3], с. 114-136

Тема 4.3 Язык релейно-контактных символов

Язык программирования релейно-контактных символов.

Связь электрической цепи и языка релейно-контактных схем.

Литература: [3], с. 136-144

Тема 4.4 Операции бинарной логики

Сущность алгебры логики. Высказывание. Виды высказываний. Основные логические операции и логические элементы: И, ИЛИ, НЕ, И- НЕ, ИЛИ-НЕ, исключающее ИЛИ, исключающее ИЛИ-НЕ, их условное графическое обозначение (далее – УГО), таблицы истинности.

Литература: [1], с. 85-92

Тема 4.5 Функции для работы с памятью

Назначение оперативных ЗУ (далее – ОЗУ) и постоянных ЗУ (далее – ПЗУ).

Возможности системы команд программируемых контроллеров для разработки программы с использованием команд работающих с памятью.

Литература: [3], с.64-68

Тема 4.6 Программирование таймеров и счётчиков

Назначение таймеров и счётчиков.

Возможности системы команд программируемых контроллеров для разработки программы с использованием таймеров и счётчиков.

Литература: [2], с. 89-95

Вопросы для самоконтроля

1 Что называют «системой команд», «командным циклом», «меткой»?
2 Назовите основные языки программирования ПЛК.
3 Назовите операции бинарной логики.
4 Назовите методы минимизации логических функций.
5 Охарактеризуйте графические языки программирования ПЛК.
6 Охарактеризуйте функции для работы с памятью.

Раздел 5 Автоматизация производства на основе промышленных сетей программируемых контроллеров

Тема 5.1 Структура сетей программируемых контроллеров

Назначение промышленных сетей.
Структура сетей программируемых контроллеров.
Литература: [2], с. 177-190

Тема 5.2 Промышленные сети PROFINET, PROFIBUS

Назначение промышленных сетей PROFINET, PROFIBUS.
Литература: [2], с. 177-190

Вопросы для самоконтроля

1 Объясните назначение промышленных сетей.
2 Объясните структуру промышленных сетей.
3 Объясните последовательную схему равнозначности кодов (ПСРК).
4 Объясните назначение сети PROFINET.
5 Объясните назначение сети PROFIBUS.

Список используемых источников

1 Бойко, В.И. Цифровые устройства: В.И. Бойко. – Санкт-Петербург БХВ-Вильямс, 2004

2 Нестеров П.В., Шаньгин В.Ф. Микропроцессоры. В 3-х кн. Кн. 1. Архитектура и проектирование микро-ЭВМ: П.В. Нестеров, В.Ф. Шаньгин. – Москва: Высшая шк., 1989

3 Петров И.В. Программируемые контроллеры. Стандартные языки и приёмы прикладного проектирования:. И.В. Петров. – Москва, 2004

4 Пятибратов, Кириченко, А.А. А.П. Вычислительные машины, сети и телекоммуникационные системы:. А.П. Пятибратов, А.А. Кириченко. – Москва: Изд. Центр ЕАОИ, 2009.

5 Таненбаум, Э. Архитектура компьютера:. Э. Таненбаум. – Санкт-Петербург Питер, 2002.

1 6 Фоминых Е.И., Фоминых Т.Е., Пархоменко Ю.Л. Арифметико-логические основы вычислительной техники:. Е.И. Фоминых, Т.Е. Фоминых, Ю.Л. Пархоменко. – Минск, РИПО 2021

Задания на домашнюю контрольную работу № 1 по учебному предмету «Программируемые контроллеры и промышленные сети»

1 Перечислите и опишите устройства ЭВМ.

2 Опишите принцип программного управления.

3 Охарактеризуйте способы представление информации в ЭВМ.

4 Опишите назначение мультиплексоров и демультиплексоров, приведите схемы реализации этих узлов, условные графические обозначения.

5 Перечислите составные части микропроцессорной системы и опишите каждую из них.

6 Опишите организацию обмена информацией в микропроцессорных системах.

7 Дайте характеристики системным и локальным шинам.

8 Перечислите составные части микропроцессора, опишите принцип функционирования.

9 Опишите схему функционирования однокристального 16-разрядных микропроцессора.

10 Охарактеризуйте внутренних регистров однокристального 16-разрядных микропроцессора.

11 Назовите основные группы инструкций процессора, охарактеризуйте ассемблерные команды.

12 Раскройте понятия прерывания и его вектора, назначение таблицы векторов прерываний, опишите ее структуру, адрес размещения в оперативной памяти, типы прерываний.

13 Опишите механизм обработки прерываний процессором.

14 Раскройте особенности архитектуры 32-разрядных микропроцессоров.

15 Изложите перспективы развития микропроцессоров.

16 Опишите структурную организацию современных моделей микропроцессоров.

17 Дайте определение микроконтроллера, опишите классификацию микроконтроллеров.

18 Опишите устройство и функционирование микроконтроллера.

19 Опишите устройство и функционирование микроконтроллера с RISC-архитектурой.

20 Опишите устройство и функционирование микроконтроллера с CISC-архитектурой.

21 Опишите классификацию и параметры запоминающих устройств.

22 Объясните назначения ОЗУ и ПЗУ, поясните различия работе данных элементов памяти.

23 Приведите структурную схему ОЗУ, объясните принцип записи и чтения.

24 Приведите структурную схему ПЗУ, объясните принцип работы

25 Объясните принципы работы кэш-памяти.

26 Объясните принципы работы со стеком.

27 Раскройте принципы функционирования подсистемы ввода-вывода.

28 Объясните назначение адаптеров и портов ввода-вывода.

29 Опишите методы управления вводом-выводом.

30 Опишите программно-управляемый ввод-вывод.

31 Опишите ввод-вывод по прерываниям.

32 Опишите назначение и классификацию мультипроцессорных вычислительных систем.

33 Изложите функциональную и структурную организацию мультипроцессорных вычислительных систем.

34 Опишите программную модель 32-разрядного процессора.

35 Опишите технические характеристики программируемых микроконтроллеров.

36 Опишите структурную схему программируемых микроконтроллеров.

37 Опишите схему, назначение и взаимодействие основных функциональных частей.

38 Опишите технологический цикл и формализацию технологического цикла технологического оборудования.

39 Опишите циклограмму работы технологического оборудования.

40 Опишите функциональный состав программируемых контроллеров.

41 Опишите систему команд программируемых контроллеров.

42 Опишите командный цикл программируемых контроллеров.

43 Опишите метки и директивы программируемых контроллеров.

44 Опишите регистры общего и специального назначения программируемых контроллеров.

45 Охарактеризуйте языки программирования стандарта IEC(МЭК)61131-3.

46 Охарактеризуйте графические языки программирования программируемых контроллеров.

47 Охарактеризуйте текстовые языки программирования программируемых контроллеров.

48 Раскройте особенности работы в реальном режиме работы процессора.

49 Раскройте особенности работы в защищенном режиме работы процессора.

50 Охарактеризуйте язык релейно-контактных символов.

51 Опишите сущность алгебры логики, высказывания и виды высказываний

52 Опишите основные логические операции и логические элементы.

53 Опишите назначение промышленных сетей.

54 Опишите структуру сетей промышленных контроллеров.

55 Опишите назначение промышленной сети PROFINET.

56 Опишите назначение промышленной сети PROFIBUS.

57 Опишите связь электрической цепи и языка релейно-контактных схем.

58 Опишите систему команд программируемых контроллеров.

59 Опишите модуль центрального вычислительного устройства программируемых контроллеров.

60 Опишите коммуникационные модули и модули ввода/вывода.

Задачи 61 – 90 Запрограммируйте микроконтроллер Arduino со следующими условиями

61 Даны четыре светодиода, которые загораются по очереди друг за другом с интервалом 500 мс. После того как загорится последний, выключаются все сразу.

62 Даны четыре светодиода, которые загораются по очереди друг за другом с интервалом 500 мс. После того как загорится последний, выключаются по очереди с интервалом 500 мс, начиная с последнего.

63 Даны четыре светодиода, которые загораются по очереди друг за другом с интервалом 500 мс. После того как загорится последний, выключаются по очереди с интервалом 500 мс, начиная с первого.

64 Даны четыре светодиода, которые загораются по очереди друг за другом с интервалом 500 мс. После того как загорится последний, выключаются по одному в произвольном порядке.

65 Даны четыре светодиода, которые загораются с интервалом 500 мс в следующем порядке: первым – четвёртый, вторым – первый, третьим – второй, четвёртым третий. После того как загорится последний, выключаются все сразу.

66 Даны четыре светодиода, которые загораются с интервалом 500 мс в следующем порядке: первым – третий, вторым – первый, третьим – четвёртый, четвёртым второй. После того как загорится последний, выключаются по очереди с интервалом 500 мс, начиная с последнего.

67 Даны четыре светодиода, которые загораются с интервалом 500 мс в следующем порядке: первым – второй, вторым – четвёртый, третьим – первый, четвёртым третий. После того как загорится последний, выключаются по очереди с интервалом 500 мс, начиная с первого.

68 Даны четыре светодиода, которые загораются с интервалом 500 мс в следующем порядке: первым – второй, вторым – третий, третьим – первый, четвёртым четвёртый. После того как загорится последний, выключаются все сразу.

69 Даны четыре светодиода, которые загораются по очереди друг за другом с интервалом 750 мс. После того как загорится последний, выключаются все сразу.

70 Даны четыре светодиода, которые загораются по очереди друг за другом с интервалом 400 мс. После того как загорится последний, выключаются по очереди с интервалом 750 мс, начиная с последнего.

71 Даны четыре светодиода, которые загораются по очереди друг за другом с интервалом 150 мс. После того как загорится последний, выключаются по очереди с интервалом 200 мс, начиная с первого.

72 Даны четыре светодиода, которые загораются по очереди друг за другом с интервалом 50 мс. После того как загорится последний, выключаются по очереди с интервалом 100 мс, начиная с первого.

73 Даны четыре светодиода, которые загораются с интервалом 250 мс в следующем порядке: первым – четвёртый, вторым – первый, третьим – второй, четвёртым третий. После того как загорится последний, выключаются все сразу.

74 Даны четыре светодиода, которые загораются с интервалом 300 мс в следующем порядке: первым – третий, вторым – четвёртый, третьим – второй, четвёртым первый. После того как загорится последний, выключаются все сразу.

75 Даны четыре светодиода, которые загораются с интервалом 250 мс в следующем порядке: первым – третий, вторым – первый, третьим – четвёртый, четвёртым второй. После того как загорится последний, выключаются по очереди с интервалом 300 мс, начиная с последнего.

76 Даны четыре светодиода, которые загораются с интервалом 600 мс в следующем порядке: первым – второй, вторым – четвёртый, третьим – первый, четвёртым второй. После того как загорится последний, выключаются по очереди с интервалом 200 мс, начиная с последнего.

77 Даны четыре светодиода и датчик расстояния. Если расстояние будет от 0 до 5 см, загорается первый светодиод, если от 5 до 10 – второй светодиод, если от 10 до 15 – третий, если не соответствует ни одному условию, то включаются все.

78 Даны четыре светодиода и датчик расстояния. Если расстояние будет от 0 до 5 см, загорается первый светодиод, если от 5 до 10 – второй светодиод, если от 10 до 15 – третий, если не соответствует ни одному условию, то выключаются все.

79 Даны четыре светодиода и датчик расстояния. Если расстояние будет от 0 до 10 см, загорается первый светодиод, если от 10 до 20 – второй светодиод, если от 20 до 30 – третий, если не соответствует ни одному условию, то включаются все.

80 Даны четыре светодиода и датчик расстояния. Если расстояние будет от 0 до 10 см, загорается первый светодиод, если от 10 до 20 – второй светодиод, если от 20 до 30 – третий, если не соответствует ни одному условию, то выключаются все.

81 Даны четыре светодиода и датчик расстояния. Если расстояние будет от 0 до 5 см, загорается четвёртый светодиод, если от 5 до 10 – первый светодиод, если от 10 до 15 – второй и третий, если не соответствует ни одному условию, то включаются все.

82 Даны четыре светодиода и датчик расстояния. Если расстояние будет от 0 до 5 см, загорается четвёртый светодиод, если от 5 до 10 – первый светодиод, если от 10 до 15 – второй и третий, если не соответствует ни одному условию, то выключаются все.

83 Даны четыре светодиода и датчик расстояния. Если расстояние будет от 0 до 7 см, загорается первый светодиод, если от 8 до 14 – четвёртый светодиод, если от 14 до 30 – второй и третий, если не соответствует ни одному условию, то включаются все.

84 Даны четыре светодиода и датчик расстояния. Если расстояние будет от 0 до 7 см, загорается первый светодиод, если от 8 до 14 – четвёртый светодиод, если от 14 до 30 – второй и третий, если не соответствует ни одному условию, то выключаются все.

85 Даны четыре светодиода и датчик расстояния. Если расстояние будет от 0 до 15 см, загорается четвёртый светодиод, если от 16 до 30 – второй светодиод, если от 30 до 45 – первый и третий, если не соответствует ни одному условию, то включаются все.

86 Даны четыре светодиода и датчик расстояния. Если расстояние будет от 0 до 15 см, загорается четвёртый светодиод, если от 16 до 30 – второй светодиод, если от 30 до 45 – первый и третий, если не соответствует ни одному условию, то выключаются все.

87 Даны четыре светодиода и датчик расстояния. Если расстояние будет от 0 до 20 см, загорается первый и второй светодиод, если от 20 до 40 – третий светодиод, если от 40 до 50 – четвёртый, если не соответствует ни одному условию, то включаются все.

88 Даны четыре светодиода и датчик расстояния. Если расстояние будет от 0 до 20 см, загорается первый и второй светодиод, если от 20 до 40 – третий светодиод, если от 40 до 50 – четвёртый, если не соответствует ни одному условию, то выключаются все.

89 Даны четыре светодиода и датчик расстояния. Если расстояние будет от 10 до 20 см, загорается третий светодиод, если от 20 до 30 – первый и четвёртый светодиод, если от 30 до 40 – второй, если не соответствует ни одному условию, то включаются все.

90 Даны четыре светодиода и датчик расстояния. Если расстояние будет от 10 до 20 см, загорается третий светодиод, если от 20 до 30 – первый и четвёртый светодиод, если от 30 до 40 – второй, если не соответствует ни одному условию, то выключаются все.

Задачи 91 – 105 На основе таблицы истинности (таблица 4) выполните следующие операции:

а) запишите исходную логическую функцию в СДНФ и СКНФ;

б) минимизируйте данную логическую функцию методом Квайна-Мак-Класки.

91 Функция F1(x1,x2,x3,x4)
92 Функция F2(x1,x2,x3,x4)
93 Функция F3(x1,x2,x3,x4)
94 Функция F4(x1,x2,x3,x4)
95 Функция F5(x1,x2,x3,x4)
96 Функция F6(x1,x2,x3,x4)
97 Функция F7(x1,x2,x3,x4)
98 Функция F8(x1,x2,x3,x4)
99 Функция F9(x1,x2,x3,x4)
100 Функция F10(x1,x2,x3,x4)
101 Функция F11(x1,x2,x3,x4)
102 Функция F12(x1,x2,x3,x4)
103 Функция F13(x1,x2,x3,x4)
104 Функция F14(x1,x2,x3,x4)
105 Функция F15(x1,x2,x3,x4)

Задачи 106 – 120 На основе таблицы истинности (таблица 5) выполните следующие операции:

а) запишите исходную логическую функцию в СДНФ и СКНФ;

б) минимизируйте данную логическую функцию методом Карно-Вейча.

106 Функция F1(x1,x2,x3,x4)
107 Функция F2(x1,x2,x3,x4)
108 Функция F3(x1,x2,x3,x4)
109 Функция F4(x1,x2,x3,x4)
110 Функция F5(x1,x2,x3,x4)
111 Функция F6(x1,x2,x3,x4)
112 Функция F7(x1,x2,x3,x4)
113 Функция F8(x1,x2,x3,x4)
114 Функция F9(x1,x2,x3,x4)
115 Функция F10(x1,x2,x3,x4)
116 Функция F11(x1,x2,x3,x4)
117 Функция F12(x1,x2,x3,x4)
118 Функция F13(x1,x2,x3,x4)
119 Функция F14(x1,x2,x3,x4)
120 Функция F15(x1,x2,x3,x4)

Задачи 121 – 150 Постройте в соответствии с минимизированной функцией (предыдущее задание) логическую схему в любом из базисов.

Методические рекомендации к решению задач 61-90

Пример 1 - Подключите к плате Arduino Uno 4 светодиода к цифровым входам 9, 10, 11 и 12. Включить светодиоды в следующем порядке: первым – второй, вторым – четвёртый, третьим – первый, четвёртым – третий с интервалом в 100мс. После того как загорается четвёртый выключаются сразу все.

```
int LED1 = 9;
int LED2 = 10;
int LED3 = 11;
int LED4 = 12;

void setup()
{
  Serial.begin(9600);
  pinMode(LED1, OUTPUT);
  pinMode(LED2, OUTPUT);
  pinMode(LED3, OUTPUT);
  pinMode(LED4, OUTPUT);
  digitalWrite(LED1, LOW);
  digitalWrite(LED2, LOW);
  digitalWrite(LED3, LOW);
  digitalWrite(LED4, LOW);
}

void loop()
{
digitalWrite(LED2, HIGH);
delay(100);
digitalWrite(LED4, HIGH);
delay(100);
digitalWrite(LED1, HIGH);
delay(100);
digitalWrite(LED3, HIGH);
delay(100);
digitalWrite(LED1, HIGH);
digitalWrite(LED2, HIGH);
digitalWrite(LED3, HIGH);
digitalWrite(LED4 , HIGH);
}
```

Пример 2 - Подключите к плате Arduino Uno 4 светодиода к цифровым входам 9, 10, 11 и 12, а контакты ультразвукового датчика trigPin – 3, echoPin – 13.

Выводить в монитор порта расстояние, получаемое с помощью датчика и если оно меньше либо равно 10 см, то горит первый светодиод, если больше 10 см и меньше 20 см, то – второй, если больше 20 см и меньше 30 см, то – третий, если больше 30 см и меньше 40 см, то – четвертый, если больше или равно 40 см, то выводит сообщение Out of range.

```
#define trigPin 3
#define echoPin 13
#define led1 9
#define led2 10
#define led1 11
#define led2 12
long duration;
long distance;
void setup() {
Serial.begin (9600);
pinMode(trigPin, OUTPUT);
pinMode(echoPin, INPUT);
pinMode(led1, OUTPUT);
pinMode(led2, OUTPUT);
pinMode(led3, OUTPUT);
pinMode(led4, OUTPUT);
}
void dist() {
digitalWrite(trigPin, LOW);
delayMicroseconds(2);
digitalWrite(trigPin, HIGH);
delayMicroseconds(10);
digitalWrite(trigPin, LOW);
duration = pulseIn(echoPin, HIGH);
distance = (duration/2) / 29.1;
}
 void loop() {
dist();
if (distance<=10) { if (k1==0) {k1=1;digitalWrite(led1,HIGH);} }
    else {if (k1==1) {k1=0;digitalWrite(led1, LOW);} }
```

```
if ((distance>10)&&(distance<=20)) { if (k2==0) {k2=1;digitalWrite(led2,HIGH);} }
else {if (k2==1) {k2=0;digitalWrite(led2, LOW);} }
if ((distance>20)&&(distance<=30)) { if (k3==0) {k3=1;digitalWrite(led3,HIGH);} }
else {if (k3==1) {k3=0;digitalWrite(led3, LOW);} }
if ((distance>30)&&(distance<=40)) { if (k4==0) {k4=1;digitalWrite(led4,HIGH);} }
else {if (k4==1) {k4=0;digitalWrite(led4, LOW);} }
}
else {Serial.println("Out of rang");
}
```

Методические рекомендации к решению задач 91-150

СДНФ и СКНФ логической функции записывается по таблице истинности.

При записи СДНФ выбираем те наборы переменных, на которых значение функции равно единице. Причём если переменная на этом наборе имеет нулевое значение, она берётся с отрицанием. Потом эти наборы соединяем между собой с помощью символа дизъюнкции.

При составлении СКНФ логической функции выбирают те наборы, при которых значение функции равно нулю. С отрицанием берутся те переменные, которые на соответствующем наборе имеют единичное значение. Полученные наборы соединяем между собой при помощи символа конъюнкции.

Пример 1 - По таблице истинности (таблица 1) записать СДНФ и СКНФ логической функции.

Таблица 1 – Таблица истинности логической функции

x1	0	0	0	0	0	0	0	0	1	1	1	1	1	1	1	1
x2	0	0	0	0	1	1	1	1	0	0	0	0	1	1	1	1
x3	0	0	1	1	0	0	1	1	0	0	1	1	0	0	1	1
x4	0	1	0	1	0	1	0	1	0	1	0	1	0	1	0	1
F	0	0	1	1	0	0	0	1	1	1	0	0	1	1	0	1

Проанализируем данную таблицу. Функция принимает единичное значение на восьми наборах. Таким образом в состав СДНФ будут входить восемь элементарных конъюнкций. Запишем СДНФ данной функции:

Также функция принимает нулевое значение на восьми наборах. Значит, в состав СКНФ будут входить восемь элементарных дизъюнкций. Запишем СКНФ данной функции:

$$F=(x1+x2+x3+x4)\times(x1+x2+x3+\overline{x4})\times(x1+\overline{x2}+x3+x4)\times(x1+\overline{x2}+x3+\overline{x4})\times(x1+\overline{x2}+$$
$$+\overline{x3}+x4)\times(\overline{x1}+x2+\overline{x3}+x4)\times(\overline{x1}+x2+\overline{x3}+\overline{x4})\times(\overline{x1}+\overline{x2}+\overline{x3}+x4)$$

Минимизация, или сокращение выражения, для логической функции необходима, чтобы обеспечить минимум затрат оборудования при построении функциональной схемы в заданном базисе (наборе) логических элементов. Чаще всего используется набор И, ИЛИ, НЕ, а минимизация ведется в ДНФ.

В рамках нормальных форм минимальной будет такая разновидность функции, которая состоит из наименьшего количества дизъюнктивных (конъюнктивных) членов при наименьшем суммарном числе символов переменных. В случае ДНФ в результате минимизации получается минимальная ДНФ (МДНФ). В случае КНФ – минимальная КНФ (МКНФ).

Метод Квайна-Мак-Класки применяется к функциям, заданным в СДНФ (возможно задание и в СКНФ), и проводится в два этапа.

На первом этапе выполняется переход от СДНФ к сокращенной ДНФ путем проведения всех возможных склеиваний друг с другом сначала конъюнкций ранга n, затем ранга n-1, далее n-2 и т. д. Каждый раз в группе конъюнкций ранга r (1<=r<=n) отыскиваются пары конъюнкций вида Ax и A $\overline{x}$, где A – общая часть этих конъюнкций. Эти конъюнкции склеиваются между собой по переменной x (например, конъюнкции $\overline{x1}\times\overline{x2}\times x3\times\overline{x4}$, $\overline{x1}\times\overline{x2}\times x3\times x4$ имеют общую часть A= $\overline{x1}\times\overline{x2}\times x3$ и склеиваются по переменной x4). При этом получается конъюнкция A ранга r-1, а конъюнкции Ax и A $\overline{x}$ помечаются и сравниваются также со всеми другими конъюнкциями ранга r с целью выполнения операции склеивания. Результатом выполнения последовательности попарного сравнения и склеивания конъюнкции ранга r является группа конъюнкций ранга r-1 и непомеченные конъюнкции ранга r. Непомеченные конъюнкции ранга r не участвовали в склеивании, следовательно, являются простыми импликантами и включаются в сокращенную ДНФ. Конъюнкции ранга r-1 вновь подвергаются попарному сравнению и склеиванию, как это было описано выше для конъюнкций ранга r. В результате имеем группу конъюнкций ранга r-2 и непомеченные конъюнкции ранга r-1, которые являются простыми импликантами, включаемыми далее в сокращенную ДНФ. Первый этап заканчивается тогда, когда вновь полученная группа конъюнкций не содержит склеивающихся членов, т. е. содержит только простые импликанты. После этого записывается сокращенная ДНФ логической функции, в которую включаются все полученные простые импликанты.

Второй этап заключается в переходе от сокращенной ДНФ к тупиковым ДНФ и выборе среди них МДНФ. Тупиковой называется такая ДНФ, среди простых импликант которой нет ни одной лишней. При этом под лишней понимается такая простая импликанта, удаление которой не влияет на значение истинности этой функции. Возможны

случаи, когда в сокращенной форме нет лишних простых импликант. Тогда сокращенная ДНФ является тупиковой.

Для выявления лишних простых импликант строится импликантная матрица, которая называется также матрицей (таблицей) покрытий. Каждая строка импликантой матрицы соответствует одной простой импликанте, а столбцы – конституантам единицы, которыми они и помечаются.

Нахождение тупиковых ДНФ по импликантной матрице начинается с разметки матрицы. При этом каждая импликанта сравнивается со всеми конституентами единицы. Если импликанта является собственной частью некоторой конституенты, то на пересечении строки и столбца ставится условный знак, например X. Конституенты единицы, помеченные в строке с простой импликантой, поглощаются (покрываются) этой простой импликантой. Это значит, что на соответствующих наборах данная импликанта обеспечивает единичные значения логической функции.

Выявление лишних простых импликант выполняется следующим образом. В импликантной таблице условно вычеркивается строка с проверяемой простой импликантой вместе с соответствующими пометками в строке. Если при этом окажется, что в каждом столбце импликантной таблицы остается хотя бы по одной пометке, проверяемая импликанта является лишней и ее следует удалить. Оставшиеся простые импликанты покрывают все единичные значения логической функции. Испытание каждой последующей простой импликанты возможно лишь после удалении уже выявленных лишних простых импликант. Изменение последовательности испытаний и удаления лишних членов сокращенной ДНФ может привести к различным тупиковым формам логической функции, из которых выбирается МДНФ.

Пример 2 - Минимизировать логическую функцию, представленную в СДНФ, методом Квайна-Мак-Класки.

$$F=\overline{x1}\times\overline{x2}\times\overline{x3}+\overline{x1}x2x3+x1\overline{x2}\times\overline{x3}+x1\overline{x2}x3+x1x2x3$$

Выполним переход от СДНФ к сокращенной ДНФ. Для удобства присвоим конъюнкциям номера и запишем их над ними.

$$F=\overset{1}{\underline{\overline{x1}\times\overline{x2}\times\overline{x3}}}+\overset{2}{\underline{\overline{x1}x2x3}}+\overset{3}{\underline{x1\overline{x2}\times\overline{x3}}}+\overset{4}{\underline{x1\overline{x2}x3}}+\overset{5}{\underline{x1x2x3}}$$

Анализируя в этом выражении всевозможные пары конъюнкций 1 – 2, 1 – 3, …, 1 – 5, 2 – 3, …, 2 – 5, 3 – 4, 3 – 5, 4 – 5 находим, что операция склеивания выполняется между парами 1 – 3, 2 – 5, 3 – 4, 4 – 5. Таким образом, все конъюнкции исходной СДНФ участвуют в склеивании и помечаются подчеркиванием.

Исходная СДНФ записывается в виде:

$$F=\overset{1}{\overline{x2}\times\overline{x3}}+\overset{2}{x2x3}+\overset{3}{x1\overline{x2}}+\overset{4}{x1x3}$$

Продолжая процедуру склеивания, сравниваем попарно конъюнкции 1 – 2, 1 – 3, 1 – 4, 2 – 3, 2 – 4 и 3 – 4. Эти пары конъюнкций между собой не склеиваются. Поэтому полученная запись логической функции представляет собой сокращенную ДНФ исходной функции, содержащей только простые импликанты.

Составим импликантную матрицу для данного примера и произведём разметку (таблица 2).

Простая импликанта $\overline{x2}\times\overline{x3}$, входящая в сокращенную ДНФ, является собственной частью конституент единиц $\overline{x1}\times\overline{x2}\times\overline{x3}$ и $x1\times\overline{x2}\times\overline{x3}$ и поэтому в строке с данной простой импликантой поставлены две пометки X в соответствующих колонках. Аналогично размечены другие строки таблицы.

Из таблицы 2 видно, что только простая импликанта 1 обеспечивает единичное значение логической функции на наборе 000, а импликанта 2 – на наборе 011 (соответствующие этим наборам столбцы обозначены конституентами единицы $\overline{x1}\times\overline{x2}\times\overline{x3}$ и $\overline{x1}\times x2\times x3$), поэтому эти простые импликанты обязательно войдут во все тупиковые ДНФ. Простую импликанту 3 можно считать лишней. Однако если ее сохранить, лишней можно считать простую импликанту 4.

Таблица 2 – Импликантная матрица

Простые импликанты	Конституенты единицы				
	$\overline{x1}\times\overline{x2}\times\overline{x3}$	$\overline{x1}\times x2\times x3$	$x1\times\overline{x2}\times\overline{x3}$	$x1\times\overline{x2}\times x3$	x1x2x3
1 $\overline{x2}\times\overline{x3}$	X		X		

2 $x2x3$		X			X

Продолжение таблицы 2

Простые импликанты	Конституенты единицы				
	$\overline{x1}\times\overline{x2}\times\overline{x3}$	$\overline{x1}\times x2\times x3$	$x1\times\overline{x2}\times\overline{x3}$	$x1\times\overline{x2}\times x3$	$x1x2x3$
3 $x1\overline{x2}$			X	X	
4 $x1x3$				X	X

Таким образом, для логической функции возможны две тупиковые формы:

$F1=\overline{x1}\times\overline{x2}+x2x3+x1x3$ в которую не включена лишняя импликанта, $x1\overline{x2}$ и $F2=\overline{x1}\times\overline{x2}+x2x3+x1\overline{x2}$ в которую не вошла лишняя импликанта $x1x3$.

Тупиковые формы F1 и F2 имеют одинаковое суммарное количество переменных, поэтому любую из них можно выбрать в качестве МДНФ.

Пример 3 - Минимизируйте логическую функцию, представленную таблицей истинности (таблица 1), методом Квайна-Мак-Класки.

Выполним переход от СДНФ к сокращенной ДНФ.

$$F=\overset{1}{\overline{x1}\times\overline{x2}\times x3\times\overline{x4}}+\overset{2}{\overline{x1}\times\overline{x2}\times x3\times x4}+\overset{3}{\overline{x1}\times x2\times x3\times x4}+\overset{4}{x1\times\overline{x2}\times\overline{x3}\times\overline{x4}}+$$
$$+\overset{5}{x1\times\overline{x2}\times\overline{x3}\times x4}+\overset{6}{x1\times x2\times\overline{x3}\times\overline{x4}}+\overset{7}{x1\times x2\times\overline{x3}\times x4}+\overset{8}{x1\times x2\times x3\times x4}.$$

Анализируя всевозможные пары конъюнкций находим, что операция склеивания выполняется между парами 1 – 2, 2 – 3, 3 – 8, 4 – 5, 5 – 7, 6 – 7, 7 – 8. Таким образом, все конъюнкции исходной СДНФ участвуют в склеивании.

Исходная СДНФ записывается в виде:

1 2 3 4 5 6 7

$$F=\overline{x1}\times\overline{x2}\times x3+\overline{x1}\times x3\times x4+x2\times x3\times x4+x1\times\overline{x2}\times\overline{x3}$$
$$+x1\times\overline{x3}\times x4+x1\times x2\times\overline{x3}+x1\times x2\times x4$$

Продолжая процедуру склеивания, сравниваем попарно конъюнкции. Между собой склеивается только пара конъюнкций 4 – 6. Остальные конъюнкции между собой не склеиваются (т.е. являются простыми импликантами).

$$F=\overset{1}{\overline{x1}\times\overline{x2}\times x3}+\overset{2}{\overline{x1}\times x3\times x4}+\overset{3}{x2\times x3\times x4}+\overset{4}{x1\times\overline{x3}}+\overset{5}{x1\times\overline{x3}\times x4}+\overset{6}{x1\times x2\times x4}$$

В данном выражении конъюнкции между собой не склеиваются. Следовательно, получена сокращенная ДНФ исходной логической функции, содержащая только простые импликанты.

Составим импликантную матрицу и произведём в ней разметку (таблица 3).

Таблица 3 – Импликантная матрица

Простые импликанты	Конституенты единицы							
	$\overline{x1}\times\overline{x2}\times x3\times\overline{x4}$	$\overline{x1}\times\overline{x2}\times x3\times x4$	$\overline{x1}\times x2\times x3\times x4$	$x1\times\overline{x2}\times\overline{x3}\times\overline{x4}$	$x1\times\overline{x2}\times\overline{x3}\times x4$	$x1\times\overline{x2}\times\overline{x3}\times\overline{x4}$	$x1\times\overline{x2}\times\overline{x3}\times x4$	$x1\times x2\times x3\times x4$
1 $\overline{x1}\times\overline{x2}\times x3$	X	X						
2 $\overline{x1}\times x3\times x4$		X	X					
3 $x2\times x3\times x4$			X					X
4 $x1\times\overline{x3}$				X	X	X	X	
5 $x1\times\overline{x3}\times x4$					X		X	
6 $x1\times x2\times x4$							X	X

Выявим лишние импликанты, условно вычёркивая строки таблицы.

При вычёркивании строк заметим, что только простая импликанта 1 обеспечивает единичное значение логической функции на наборе 0010, а импликанта 4 – на наборах 1000 и 1100, поэтому эти простые импликанты обязательно войдут во все тупиковые ДНФ.

Остальные простые импликанты можно считать лишними, если сохранить одну любую из них. Таким образом, для функции возможны несколько тупиковых форм.

Сохраним простую импликанту 3. Тогда МДНФ логической функции будет:

$$F_{min} = \bar{x}_1 \times \bar{x}_2 \times x_3 \times x_4 + x_1 \times x_3$$

Метод диаграмм Вейча (карт Карно) удобен для минимизации логических функций, содержащих обычно не более четырех переменных.

Диаграмма Вейча имеет вид прямоугольника (квадрата), разбитого на 2^n клеток, где n – число аргументов логической функции. Каждой клетке диаграммы ставится в соответствие определенная конъюнкция (дизъюнкция), причем конъюнкции (дизъюнкции) располагают таким образом, чтобы в соседних клетках (в строке или в столбце) они отличались не более чем значением одной переменной. В результате любые две соседние в строке или в столбце конъюнкции (дизъюнкции) склеиваются по соответствующей переменной. Соседними на диаграмме являются также крайние (левая и правая) конъюнкции (дизъюнкции) в одной строке и (нижняя и верхняя) конъюнкции (дизъюнкции) в одном столбце.

Диаграммы Вейча для двух, трех и четырех переменных изображены на рисунке 1 а), б), в) соответственно.

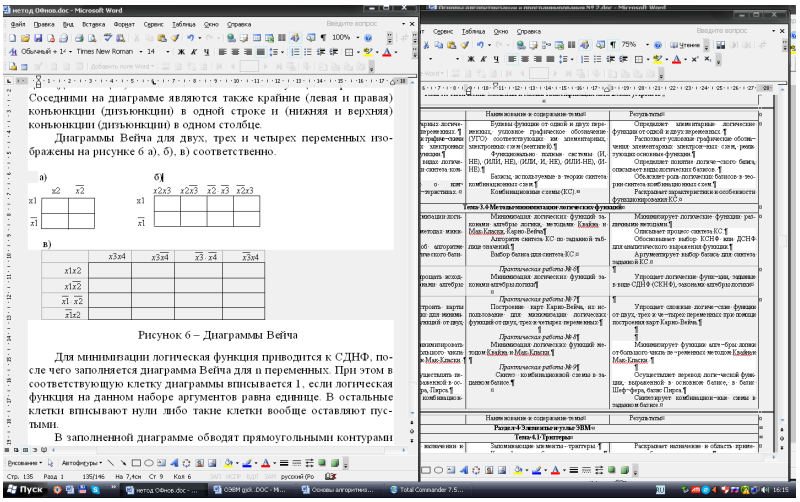

Рисунок 1 – Диаграммы Вейча

Для минимизации логическая функция приводится к СДНФ, после чего заполняется диаграмма Вейча для n переменных. При этом в соответствующую клетку диаграммы вписывается 1, если логическая функция на данном наборе аргументов равна единице. В остальные клетки вписывают нули либо такие клетки вообще оставляют пустыми.

В заполненной диаграмме обводят прямоугольными контурами клетки с единицами, после чего записывают МДНФ логической функции в виде дизъюнкции простых импликант, описывающих эти контуры. При проведении контуров придерживаются следующих правил:

− внутри контура должны быть только клетки с единицами;

− количество клеток с единицами должно выражаться величиной 2^i (где i = 0, 1,2, ...), т. е. может быть равно 1, 2, 4, 8 и т. д.;

− единицы в крайних клетках одного столбца или одной строки могут включаться в один контур;

− каждый контур должен включать как можно большее число клеток с единицами, а общее число контуров должно быть как можно меньшим.

В простую импликанту, описывающую контур, включаются те переменные, которые во всех клетках контура имеют либо только прямое, либо только инверсное значение.

Если нужно минимизировать СКНФ, в соответствующую клетку вписывается 0, если функция на данном наборе равна нулю. В остальные клетки вписывают единицы либо такие клетки вообще оставляют пустыми. В заполненной диаграмме обводят прямоугольными контурами клетки с нулями, после чего записывают МКНФ логической функции в виде конъюнкции простых импликант, описывающих эти контуры.

Пример 4 - Минимизировать логическую функцию методом диаграмм Вейча.

$$F=x_1x_2x_3+x_1x_2\overline{x_3}+x_1\overline{x_2}x_3+\overline{x_1}\,\overline{x_2}\,\overline{x_3}$$

Данная функция уже записана в СДНФ.

Заполняем диаграмму Вейча для трёх переменных и проводим 3 контура, охватывающих клетки с единицами (рисунок 2).

$x2x3 \quad x2\overline{x3} \quad \overline{x2}\,\overline{x3} \quad \overline{x2}x3$

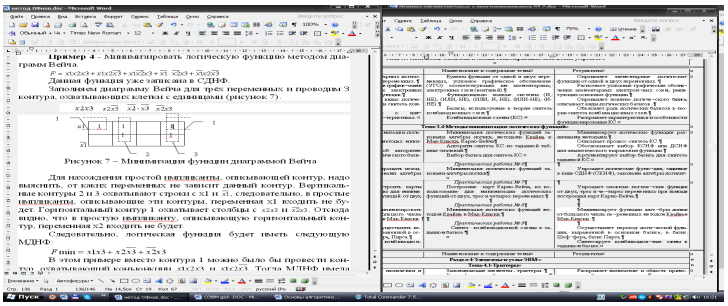

Рисунок 2 – Минимизация функции диаграммой Вейча

Для нахождения простой импликанты, описывающей контур, надо выяснить, от каких переменных не зависит данный контур. Вертикальные контуры 2 и 3 охватывают строки с x1 и $\overline{x1}$, следовательно, в простые импликанты, описывающие эти контуры, переменная x1 входить не будет. Горизонтальный контур 1 охватывает столбцы с $x2x3$ и $\overline{x2}x3$. Отсюда видно, что в простую импликанту, описывающую горизонтальный контур, переменная x2 входить не будет.

Следовательно, логическая функция будет иметь следующую МДНФ:

$$Fmin=x1x3+x2\overline{x3}+\overline{x2}x3$$

В этом примере вместо контура 1 можно было бы провести контур, охватывающий конъюнкции $x1x2x3$ и $x1x2\overline{x3}$. Тогда МДНФ имела бы вид:

$$Fmin=x1x2+x2\overline{x3}+\overline{x2}x3$$

Таким образом, для логической функции F возможны две тупиковые (минимальные) формы, равноценные по числу букв и слагаемых, и каждую из них можно выбрать для построения комбинационной схемы.

Пример 5 - Минимизируйте логическую функцию, представленную таблицей истинности (таблица 1), методом диаграмм Вейча (карт Карно).

Запишем СДНФ этой функции:

$$F = \overline{x1}\times\overline{x2}\times\overline{x3}\times x4 + \overline{x1}\times\overline{x2}\times\overline{x3}\times x4 + x1\times\overline{x2}\times x3\times x4 + x1\times\overline{x2}\times\overline{x3}\times\overline{x4} + x1\times\overline{x2}\times\overline{x3}\times x4 +$$
$$+ x1\times x3\times\overline{x4} + \overline{x1}\times x2\times x3\times\overline{x4} + x1\times x2\times x3\times x4$$

Заполним диаграмму Вейча для четырёх переменных и проведём три контура (рисунок 3).

Контур 1 охватывает строки с $x1x2$ и $x1\overline{x2}$ и столбцы с $\overline{x3}\,\overline{x4}$ и $\overline{x3}x4$. Отсюда видно, что в простую импликанту, описывающую этот контур, переменные x2 и x4 входить не будут. Контур 2 охватывает столбцы с $x3x4$ и $x3\overline{x4}$, следовательно, переменной x4 не будет в простой импликанте, описывающей данный контур. Контур 3 охватывает строки с $x1x2$ и $\overline{x1}x2$, и в простой импликанте, описывающей этот контур, не будет переменной x1.

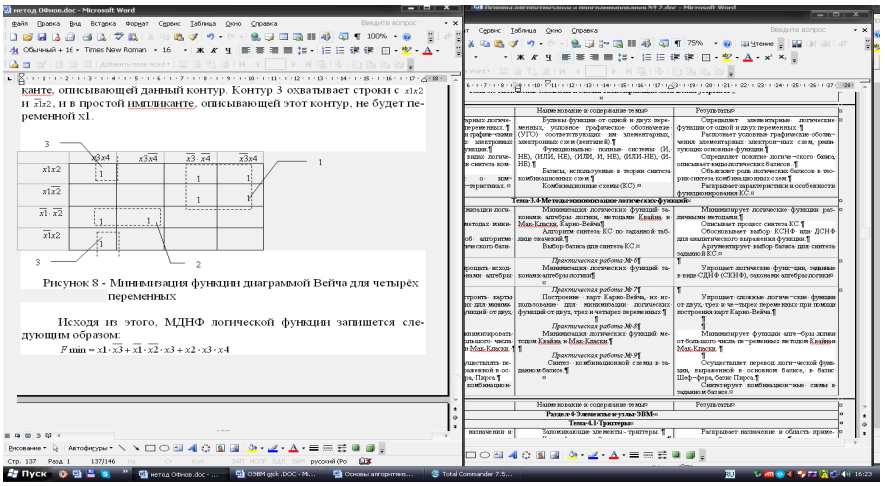

Рисунок 3 - Минимизация функции диаграммой Вейча для четырёх переменных

Исходя из этого, МДНФ логической функции запишется следующим образом:

$$F_{min} = x1\times\overline{x2}\times\overline{x3} + \overline{x2}\times x3\times x4$$

Для примера найдём МКНФ этой же логической функции, используя диаграмму Вейча.

СКНФ логической функции выглядит следующим образом:

$$F=(x1+x2+x3+x4)\times(x1+x2+x3+\overline{x4})\times(x1+x2+\overline{x3}+x4)\times$$
$$\times(x1+\overline{x2}+\overline{x3}+x4)\times(\overline{x1}+x2+\overline{x3}+x4)\times(\overline{x1}+x2+\overline{x3}+\overline{x4})\times(\overline{x1}+\overline{x2}+\overline{x3}+\overline{x4})$$

Заполним диаграмму Вейча и проведём контуры (рисунок 4).

Рассмотрим полученные контуры. Контур 1 охватывает строки с $x1+x2$ и $x1+\overline{x2}$ и столбцы с $x3+x4$ и $x3+\overline{x4}$. Т.к. только переменные x1 и x3 во всех клетках контура имеют прямое значение, они и будут составлять простую импликанту, описывающую этот контур. Контур 2 охватывает строки с $x1+\overline{x2}$ и $\overline{x1}+\overline{x2}$, следовательно, переменной x1 в простой импликанте, описывающей данный контур, не будет. Контур 3 охватывает столбцы с $\overline{x3}+\overline{x4}$ и $\overline{x3}+x4$. Отсюда видно, что в простую импликанту, описывающую горизонтальный контур 3, переменная x4 входить не будет.

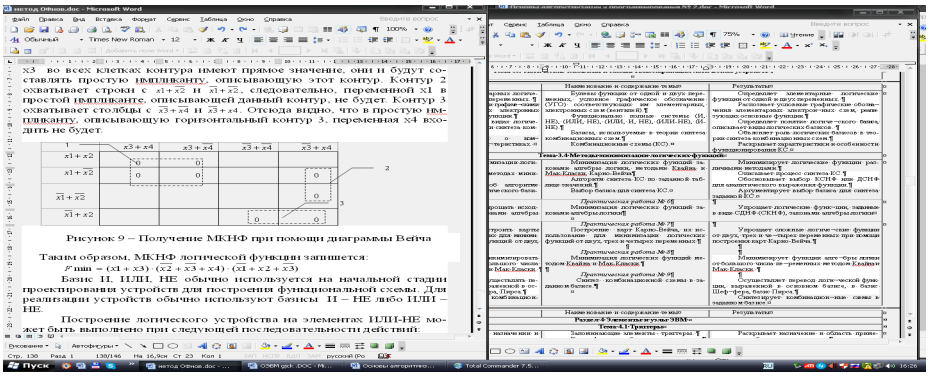

Рисунок 4 – Получение МКНФ при помощи диаграммы Вейча

Таким образом, МКНФ логической функции запишется:

$$F=(x1+x3)\times(\overline{x1}+\overline{x2}+x3)$$

Базис И, ИЛИ, НЕ обычно используется на начальной стадии проектирования устройств для построения функциональной схемы. Для реализации устройств обычно используют базисы И – НЕ либо ИЛИ – НЕ.

Построение логического устройства на элементах ИЛИ-НЕ может быть выполнено при следующей последовательности действий:

– заданная функция минимизируется с получением МКНФ;

– производится запись полученного логического выражения через операции ИЛИ-НЕ.

Рассмотрим последовательность синтеза на примере построения логического устройства, реализующего функцию, приведенную в таблице 1.

После минимизации с помощью диаграмм Вейча (смотрите пример 5) МКНФ имеет вид:

$$F=(x1+x3)\times(\overline{x2}+\overline{x3}+x4)\times(\overline{x1}+x2+\overline{x3})$$

Для перехода от базиса И, ИЛИ, НЕ, в котором представлено полученное логическое выражение, к базису ИЛИ-НЕ проводим следующие действия:

– дважды инвертируем правую часть выражения

$$F=\overline{\overline{(x1+x3)\times(\overline{x2}+\overline{x3}+x4)\times(\overline{x1}+x2+\overline{x3})}};$$

– проводим преобразование по формуле де Моргана

$$F=\overline{\overline{(x1+x3)}+\overline{(\overline{x2}+\overline{x3}+x4)}+\overline{(\overline{x1}+x2+\overline{x3})}};$$

– записываем выражение с использованием символа операции ИЛИ-НЕ

$$F=\overline{\overline{(x1+x3)}+\overline{(\overline{x2}+\overline{x3}+x4)}+\overline{(\overline{x1}+x2+\overline{x3})}}.$$

Заметим, что наличие поставленных скобок обязательно, иначе исказится функция.

Построенная в соответствии с логическим выражением схема логического устройства приведена на рисунке 5.

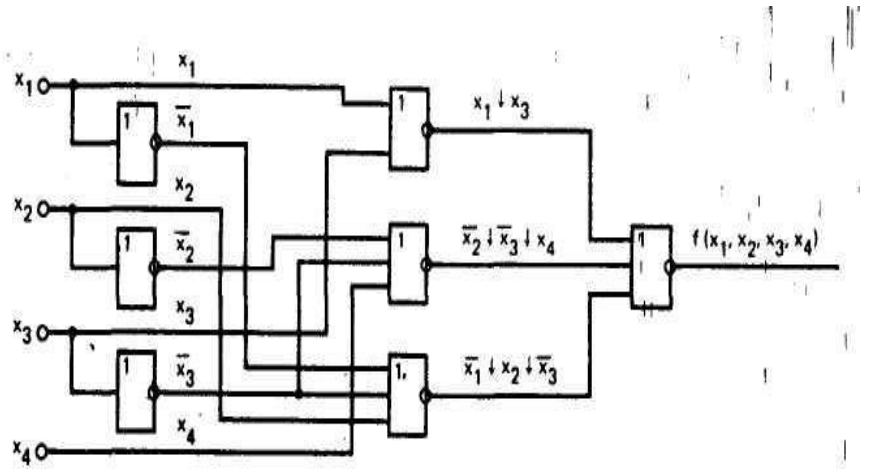

Рисунок 5 – Логическая схема в базисе ИЛИ-НЕ

Методика синтеза устройства в базисе И-НЕ сходна с рассмотренной выше методикой синтеза в базисе ИЛИ-НЕ. Рассмотрим построение с использованием элементов И-НЕ логического устройства, реализующего функцию, заданную таблицей истинности (таблица 1).

В отличие от синтеза в базисе ИЛИ-НЕ, при котором в процессе минимизации получают МКНФ функции, при синтезе в базисе И-НЕ должна быть получена МДНФ функции. Минимальная ДНФ функции (смотрите примеры 3, 5):

$$F=x1\times\overline{x3}+x2\times x3\times x4+\overline{x1}\times\overline{x2}\times x3$$

Дважды инвертируем правую часть выражения

$$F=\overline{\overline{x1\times\overline{x3}+x2\times x3\times x4+\overline{x1}\times\overline{x2}\times x3}}$$

Проводим преобразование по формуле де Моргана

$$F=\overline{\overline{(x1\times\overline{x3})}\times\overline{(x2\times x3\times x4)}\times\overline{(\overline{x1}\times\overline{x2}\times x3)}}$$

Записываем выражение с использованием символа И-НЕ

$$F=(x1|\overline{x3})|(x2|x3|x4)|(\overline{x1}|\overline{x2}|x3)$$

Данному выражению соответствует схема, приведенная на рисунке 6.

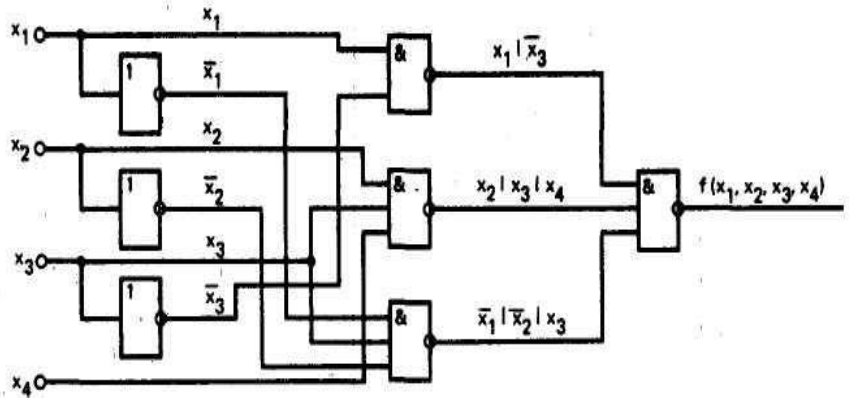

Рисунок 6 – Логическая схема в базисе И-НЕ

Таблица 4 – Таблица истинности логических функций от четырех переменных

x1	0	0	0	0	0	0	0	0	1	1	1	1
x2	0	0	0	0	1	1	1	1	0	0	0	0
x3	0	0	1	1	0	0	1	1	0	0	1	1
x4	0	1	0	1	0	1	0	1	0	1	0	1
F1(x1,x2,x3,x4)	1	1	0	0	1	0	1	0	1	1	0	1
F2(x1,x2,x3,x4)	0	1	0	1	0	1	1	0	1	1	1	0
F3(x1,x2,x3,x4)	0	0	1	1	0	0	1	1	0	0	1	1
F4(x1,x2,x3,x4)	0	1	0	1	0	1	0	1	1	1	0	0
F5(x1,x2,x3,x4)	1	1	1	0	0	1	1	1	0	0	1	0
F6(x1,x2,x3,x4)	0	1	0	1	0	0	1	0	0	1	0	1
F7(x1,x2,x3,x4)	1	1	0	1	0	1	1	0	1	0	0	1
F8(x1,x2,x3,x4)	1	1	0	0	0	1	0	1	0	1	0	0
F9(x1,x2,x3,x4)	0	1	0	0	1	0	1	0	1	1	0	0
F10(x1,x2,x3,x4)	1	0	0	1	0	0	1	1	0	1	0	1
F11(x1,x2,x3,x4)	0	0	0	1	1	1	0	0	1	1	0	1
F12(x1,x2,x3,x4)	0	1	1	0	0	1	0	1	0	0	1	1
F13(x1,x2,x3,x4)	1	0	1	1	0	1	0	0	1	1	0	1
F14(x1,x2,x3,x4)	0	1	0	1	0	1	0	1	0	0	1	1
F15(x1,x2,x3,x4)	1	1	0	1	0	0	1	0	1	0	0	1

Таблица 5 – Таблица истинности логических функций от четырёх переменных

x1	0	0	0	0	0	0	0	0	1	1	1	1	1	1	1	1
x2	0	0	0	0	1	1	1	1	0	0	0	0	1	1	1	1
x3	0	0	1	1	0	0	1	1	0	0	1	1	0	0	1	1
x4	0	1	0	1	0	1	0	1	0	1	0	1	0	1	0	1
F1(x1,x2,x3,x4)	1	0	0	1	0	1	0	1	0	0	1	1	1	1	1	0
F2(x1,x2,x3,x4)	0	0	0	1	1	0	0	1	0	0	1	1	1	0	1	1
F3(x1,x2,x3,x4)	0	0	1	0	0	1	1	1	0	1	1	0	0	1	1	0
F4(x1,x2,x3,x4)	0	1	0	1	0	1	1	0	0	1	0	1	1	1	1	1
F5(x1,x2,x3,x4)	1	1	0	1	1	1	1	0	0	0	0	1	1	0	1	1
F6(x1,x2,x3,x4)	0	1	1	0	0	0	1	1	0	1	0	0	1	1	1	1
F7(x1,x2,x3,x4)	0	1	0	1	1	0	1	0	1	1	0	0	1	1	1	1
F8(x1,x2,x3,x4)	1	0	0	1	0	1	1	1	0	1	0	1	0	1	1	0
F9(x1,x2,x3,x4)	0	0	0	1	1	1	0	0	1	1	0	1	1	1	1	0
F10(x1,x2,x3,x4)	0	1	1	0	0	1	0	1	0	1	1	1	0	0	1	1
F11(x1,x2,x3,x4)	1	1	1	0	0	1	1	0	1	1	0	1	1	0	1	0
F12(x1,x2,x3,x4)	0	0	1	1	0	1	0	1	1	0	1	1	1	0	1	1
F13(x1,x2,x3,x4)	1	0	1	1	0	0	1	0	1	1	0	1	1	0	1	1
F14(x1,x2,x3,x4)	0	1	0	1	1	0	1	1	0	1	1	1	0	1	1	0
F15(x1,x2,x3,x4)	1	1	0	1	0	0	1	0	1	0	0	1	1	0	1	1

Таблица 6 – Варианты заданий на домашнюю контрольную работу № 1 по учебному предмету «Программируемые контроллеры и промышленные сети»

Предпоследняя цифра шифра	Последняя цифра шифра									
	0	1	2	3	4	5	6	7	8	9
0	7, 55, 83, 92, 127	26, 39, 79, 102, 139	23, 36, 74, 115,150	1,43,72, 95, 125	27, 57, 82, 118,123	20,37,81, 91,128	9, 44, 86, 91,148	8,46, 65, 108, 140	15,54,67, 109, 144	10,45,78, 110, 121
1	19,41,77, 108, 141	16,32,72, 117,134	13, 58, 67, 105, 136	6, 52, 75, 99, 122	4, 47, 78, 108, 124	13,43,85, 93, 145	5, 55, 87, 102, 126	15,36, 83, 103, 149	17,48, 66, 118, 132	2, 60, 84, 109, 128
2	24, 53,88, 100, 137	30, 54, 70, 94,143	21,57,90, 114, 139	7,37,71, 120, 148	10,46,83, 111,123	14, 34, 79, 101,130	29,42,64, 107, 146	26, 56, 65, 92, 150	12,49,69, 113, 121	27,35,80, 119, 144
3	23, 38,78. 112,140	22, 59, 86, 104,125	3, 40, 89, 95, 133	28,31,62, 98, 142	11,33,61, 110, 131	9, 51. 74, 116,127	1,45,73, 97,138	25,50,81, 91, 147	20, 39, 82, 96,129	8. 44, 68, 115, 135
4	19,51,87, 91, 121	28, 55, 65, 109, 146	30, 53, 80, 102,143	23, 40, 34, 94, 131	10, 60, 75, 99,127	20, 44, 72, 95, 135	1,45,71, 107, 148	26,48,76, 110, 141	12, 34, 36, 111, 150	2, 38, 81, 112, 149
5	27, 57,65, 117,127	8, 59, 83, 97, 135	12,56, 76, 98,149	30, 58, 64, 105, 136	15,53,72, 99,142	4, 36, 71, 116,140	10, 33, 89, 119,134	1,54,80, 101,131	6,41,70, 109, 145	7, 34, 68 104, 133
6	26, 44,79, 100, 132	13,52,75, 91,86,	107, 128 96,130	5,43,77, 93, 138	21,50,66, 118, 144	2, 39. 74, 102,150	24, 48, 78, 120, 148	11,32,73, 103, 146	17, 55, 69, 92,141	28,42,67, 112, 129
7	18,49,63, 111,137	20, 35, 90, 113,139	19,40,88, 115,121	25, 47. 62 106, 123	29, 45, 87, 95,122	23,51,81, 110,124	14, 60, 85, 93,147	3, 38, 61, 114,143	16,37,82, 108, 126	9,46, 84, 94, 125
8	17,48,69, 106, 137	14,59,75, 116,147	13,40,84, 112,134	25,58,61, 96,122	29, 42, 89, 119, 142	19, 35, 80, 104,130	30,41,62, 98,149	24,51,77, 110, 138	18,33,68, 94,135	28,56,87, 105, 141
9	22, 60,70, 113,131	2, 50, 64, 101,126	11,47,73, 120, 129	16,32,85, 114,145	12,31,88, 100, 133	6, 53, 71, 103,146	5, 38, 63, 93,143	21,49,86, 117, 132	3, 34,76, 107,136	4, 52, 90, 99, 124