

Министерство науки и высшего образования Российской Федерации  
Лысьвенский филиал федерального государственного автономного образовательного  
учреждения высшего образования  
«Пермский национальный исследовательский политехнический университет»

Кафедра Технических дисциплин

## **УЧЕБНО-МЕТОДИЧЕСКИЙ КОМПЛЕКС ДИСЦИПЛИНЫ**

### **«ЭВМ и периферийные устройства»**

основной профессиональной образовательной программы

высшего образования – программы бакалавриата

по направлению 09.03.01

Информатика и вычислительная техника

### **МЕТОДИЧЕСКИЕ УКАЗАНИЯ**

**по организации и проведению**

**курсовых работ**

Лысьва, 2021

Составитель ст. преподаватель кафедры ТД М.Н. Апталаев.

Методические указания рассмотрены и одобрены на заседании кафедры технических дисциплин «30» августа 2021 г, протокол №1.

## Содержание

<b>1</b>	<b>Методология разработки цифровых устройств</b> .....	<b>4</b>
<b>2</b>	<b>Этапы выполнения курсовой работы</b> .....	<b>6</b>
2.1	Исследование предметной области курсовой работы .....	6
2.2	Анализ исходных данных задания на курсовую работу .....	6
2.3	Спецификация устройства на уровне «черного ящика» .....	6
2.4	Представление устройства в виде операционной и управляющей частей .....	6
2.5	Разработка структуры устройства .....	7
2.6	Составление алгоритма работы устройства .....	7
2.7	Разработка микропрограммы работы устройства .....	7
2.8	Составление полной спецификации устройства .....	8
2.9	Составление фрагмента функциональной схемы устройства .....	8
2.10	Контрольный пример .....	9
2.11	Составление текста и оформление курсовой работы .....	9
<b>3</b>	<b>Критерии оценки курсовой работы</b> .....	<b>10</b>
	<b>Список использованной литературы</b> .....	<b>11</b>
	<b>ПРИЛОЖЕНИЕ А Правила составления и оформления курсовой работы (по ГОСТ 7.32-2001)</b> .....	<b>12</b>
	<b>ПРИЛОЖЕНИЕ Б Образец титульного листа курсовой работы</b> .....	<b>19</b>
	<b>ПРИЛОЖЕНИЕ В Бланк задания на выполнение курсовой работы</b> .....	<b>20</b>
	<b>ПРИЛОЖЕНИЕ Г Пример составления реферата к курсовой работе</b> .....	<b>22</b>
	<b>ПРИЛОЖЕНИЕ Д Варианты заданий на курсовую работу</b> .....	<b>23</b>
	<b>ПРИЛОЖЕНИЕ Е Пример выполнения курсовой работы</b> .....	<b>26</b>

# 1 Методология разработки цифровых устройств

Проектирование – разработка технической документации, позволяющей изготовить устройство с заданным законом функционирования, с заданными свойствами и в заданных условиях. Существуют две методологии проектирования: «сверху-вниз» и «снизу-вверх».

Методология «сверху-вниз» отображает процесс проектирования от технического задания до электрических схем, файлов прошивки ПЗУ и конфигурации программируемых приборов, а также конструкции устройства в целом.

Другая последовательность, соответствующая методологии «снизу-вверх», предусматривает объединение простейших модулей в более сложную структуру до тех пор, пока не будет создан конечный проект. Исходные модули – это решения, созданные проектировщиком на более ранних этапах работы или в ходе работ над другими проектами или доступные проектировщику и входящие в состав имеющихся библиотек современных систем автоматизации проектирования (САПР).

Современным условиям проектирования, когда создаются сложные проекты с привлечением большого числа разработчиков, больше соответствует применение стратегии «сверху-вниз».

Традиционно процесс проектирования цифровых устройств разбивается на следующие этапы [1]:

- системного проектирования;
- структурно-алгоритмического проектирования;
- функционально-логического проектирования;
- конструкторско-технологического проектирования.

На этапе системного проектирования определяется архитектура будущей системы, состав компонентов и основные характеристики системы при таком ее построении. При структурно-алгоритмическом проектировании определяются структура и алгоритмы функционирования аппаратных и программных компонентов системы. На этапе функционально-логического проектирования разрабатываются функциональные и принципиальные электрические схемы, программы, подготавливаются тестовые и контрольные данные. На конструкторском этапе производится привязка элементов проекта к конструктивным элементам. Этап структурно-алгоритмического проектирования некоторого блока ЭВМ является темой курсовой работы.

В основе стратегии проектирования цифровых устройств методом «сверху-вниз» лежат принцип функциональной декомпозиции и концепция «черного ящика»:

1. Принцип функциональной декомпозиции[1]. При декомпозиции (разложении) функция, выполняемая устройством, разбивается на более простые функции, между которыми должны быть установлены определенные связи, соответствующие принятому алгоритму реализации функции. Эти функции разбиваются на более мелкие до тех пор, пока не получатся типовые элементарные арифметические и логические функции.

2. Проектируемое устройство на начальном этапе рассматривается как функциональный «черный ящик», т.е. блок с неизвестным содержимым. Для него разрабатывается функциональная спецификация, включающая внешнее описание блока - входные и выходные данные и сигналы, внутреннее описание – функцию или алгоритм работы. После выделения функциональных блоков разрабатываются алгоритмы их работы. Логику алгоритмов удобно представлять графическими схемами. Общие рекомендации по разработке алгоритмов включают в себя подробное рассмотрение назначения блока, определение форматов входных и выходных данных и анализ связей данного блока с другими блоками.

3. Декомпозиция функций блоков есть многоуровневый многошаговый и итерационный процесс, с возвратами назад и пересмотром ранее принятых решений. Он выполняется сначала для «черного ящика», затем – для тех блоков, на которые разбивается «черный ящик» и т.д. Как указывалось выше, блоки разбиваются на более мелкие блоки по функциям до элементарных

функций, каждая из которых может быть реализована типовыми элементами и узлами микросхем.

4. При проектировании функционально законченных устройств, работающих по определенному алгоритму, удобно разбивать на две части: управляющую и операционную. Операционная часть обрабатывает элементарные действия (микрооперации), составляющие алгоритм функционирования устройства [2]. Управляющая – вырабатывает сигналы, определяющие моменты времени для исполнения этих действий, т.е. обрабатывает микропрограмму [3, гл.3]. Операционная и управляющая части взаимосвязаны алгоритмом микропрограммы, но их разрабатывают отдельно как блоки одного устройства.

Успех проектирования в значительной мере определяется тем, насколько хорошо разработчик понял и сформулировал решаемую проблему, трансформировал ее в рабочие характеристики устройства и далее дал техническое определение и описание, т.е. спецификацию проектируемого устройства. Практика показывает, что большинство ошибок в проектах системы появляется из-за недостаточно полных и четких спецификаций. Для них целесообразно использовать графические представления решаемых задач в виде схем алгоритмов, временных диаграмм и таблиц. Таким образом, содержанием курсовой работы является составление алгоритма работы устройства в виде микропрограммы структуры операционного или управляющего блока устройства фрагмента функциональной схемы операционного или управляющего блока устройства.

## **2 Этапы выполнения курсовой работы**

### **2.1 Исследование предметной области курсовой работы**

- а) Ознакомиться по методуказаниям с методологией (направлением, принципами и методами) разработки цифровых устройств.
- б) Проанализировать задание на работу, понять, что является объектом разработки
- в) Прочитать по литературе из рекомендуемого списка отдельные разделы, дающие информацию об объекте курсовой работы. Выбрать 2-3 источника, наиболее полно отражающих законы (принципы) функционирования объекта, и в дальнейшем использовать эти источники как основную литературу.
- г) Изучить по основной литературе параметры, характеристики и описание работы объекта, постоянно соотнося прочитанное с текстом задания на курсовую работу. Результаты анализа законов работы объекта представить в виде первоначальной схемы алгоритма.
- д) Ознакомиться с примером выполнения курсовой работы, не считая, тем не менее, ее совершенным и единственным образцом для подражания.
- е) Понять, в чем заключается результат курсовой работы по теме, и наметить план его достижения с учетом принципов структурно-алгоритмического этапа проектирования.

### **2.2 Анализ исходных данных задания на курсовую работу**

- а) Доопределить необходимые данные. Обычно заказчик на разрабатываемое устройство формулирует в задании только интересующие его требования. Но разработчику часто бывает необходимо знать и использовать и другие параметры проектируемого устройства, которые не должны быть противоречить исходным данным. На этом этапе курсовой работы надо определить подобные параметры и характеристики.
- б) Выделить условия и режимы работы, а в них, – возможно, и параметры работы устройства.

### **2.3 Спецификация устройства на уровне «черного ящика»**

- а) Исходя из функции устройства по заданию и первоначальному алгоритму
  - выявить форматы входных (операндов) и выходных (результатов обработки) данных;
  - определить количество и разрядность входных и выходных шин, присвоить шинам имена;
  - выбрать значение активного уровня для сигналов каждой шины.
- б) Уточнить первоначальный алгоритм, используя в нем имена и назначения шин.

### **2.4 Представление устройства в виде операционной и управляющей частей**

- а) Разбить устройство на две части: операционную и управляющую. Исходя из функций каждой части устройства и задания:
    - определить количество и разрядность входных и выходных шин для каждой части и присвоить всем шинам имена;
    - связать части устройства между собой шинами;
- В общем случае взаимосвязь между операционной (ОЧ) и управляющей (УЧ) частями через шины может быть проиллюстрирована рисунком 1. На нем выделены управляющие сигналы для ОЧ и осведомительные сигналы, выставляющие результаты проверки логических условий в УЧ. Как известно, схема алгоритма состоит из операторных блоков (действий по преобразованию информации, выполняемых в ОЧ в соответствии с функцией устройства) и логических блоков (действий по проверке некоторых бинарных условий, разветвляющих вычислительный процесс, т.е. изменяющих линейный тип алгоритма.).

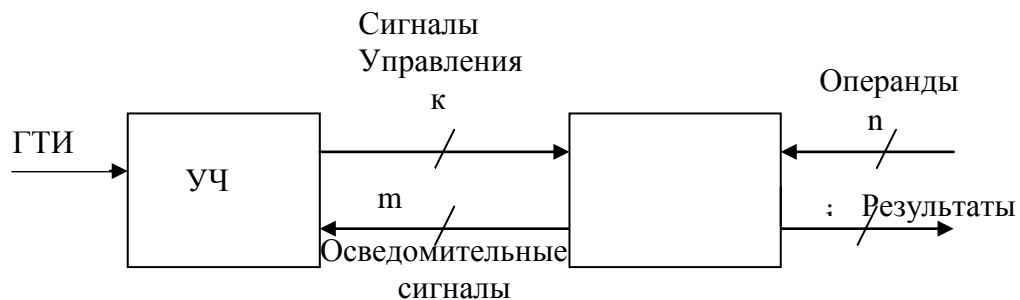


Рисунок 1. - Управляющая и операционная части устройства

## 2.5 Разработка структуры устройства

Исходя из составленного последнего алгоритма работы проектируемого устройства, нужно разработать структуру его операционной части. Ее изображают в виде операционных блоков, связанных как между собой, так и с входными и выходными шинами устройства. Названия и назначение операционных блоков должны соответствовать основным операциям в алгоритме и заданию на курсовую работу. Обычно операционными блоками являются типовые узлы (микросхемы), как комбинационные, так и на элементах памяти. Причем для каждого блока, введенного в структуру, должны быть определены и названы как входные, так и выходные шины.

## 2.6 Составление алгоритма работы устройства

а) Составить блочную схему алгоритма работы ОЧ в терминах выявленных преобразований ее операндов, формирования результатов и осведомительных сигналов.

б) Довести каждый операторный блок до уровня микроопераций, заменяя его последовательностью элементарных действий над операндами – двоичными кодами, таких как

- пересылка (установка, запись);
- поразрядный сдвиг;
- поразрядная конъюнкция двоичных кодов;
- поразрядная дизъюнкция двух двоичных кодов;
- поразрядная инверсия;
- равнозначность (или ее отрицание);
- алгебраическое сложение двоичных кодов и др.

Следует помнить, что каждая микрооперация выполняется под действием своего сигнала управления и на схеме алгоритма записывается в блоке прямоугольного типа с одним выходом.

в) Довести каждый логический блок до уровня одного или последовательности нескольких логических условий. Логическое условие формулируется в виде результата сравнения некоторой переменной (операнда, результата, их отдельных полей или даже отдельного разряда с константой или другой переменной). Следует помнить, что результат проверки логического условия есть бинарная переменная, которая может принимать одно из двух значений: истинное или ложное и записывается на схеме алгоритма в блоке ромбовидного типа с двумя выходами соответственно.

## 2.7 Разработка микропрограммы работы устройства

Выполняется на основе концепции микропрограммного управления [Зс. 131; с 300] по следующим правилам:

а) Обозначить в схеме алгоритма последовательно от блока «начало» до блока «конец» каждый операторный блок символом  $Y_i$ , где  $i$  – номер операторного блока по порядку, начиная с номера 1. Теперь  $Y_i$  – это  $i$ -тый сигнал управления, выставляемый из УЧ в ОЧ в  $i$ -тый такт ГТИ для обеспечения выполнения  $i$ -той микрооперации.

б) Обозначить в схеме алгоритма последовательно от блока «начало» до блока «конец» каждый логический блок символом  $X_j$ , где  $j$  - номер логического блока по порядку, начиная с

номера 1. Теперь  $X_j$  - это  $j$ -тый осведомительный сигнал, выставляемый из ОЧ в УЧ для обеспечения требуемых по алгоритму ветвлений.

### 2.8 Составление полной спецификации устройства

а) Составить полную таблицу сигналов устройства, включив в нее входные, выходные и промежуточные (X и Y) сигналы. В таблице указать

- имя сигнала;
- назначение сигнала;
- тип (входной или выходной);
- активный уровень сигнала.

б) Представить устройство в виде УЧ и ОЧ и соединить их всеми сигналами X и Y. Показать на схеме входные и выходные шины для передачи операндов и результатов, указать разрядность всех шин. УЧ в ней остался на уровне «черного ящика».

### 2.9 Составление фрагмента функциональной схемы устройства

Далее в курсовой работе каждый студент в соответствии с заданием из микропрограммы, полученной на предыдущем этапе курсовой работы, должен: построить функциональную схему УЧ, формирующую тактовые сигналы управления для ОЧ.

УЧ может быть построено как микропрограммный автомат с жесткой логикой [3, гл.6 с.300]. Структура УЧ в общем виде приведена на рисунке 2.

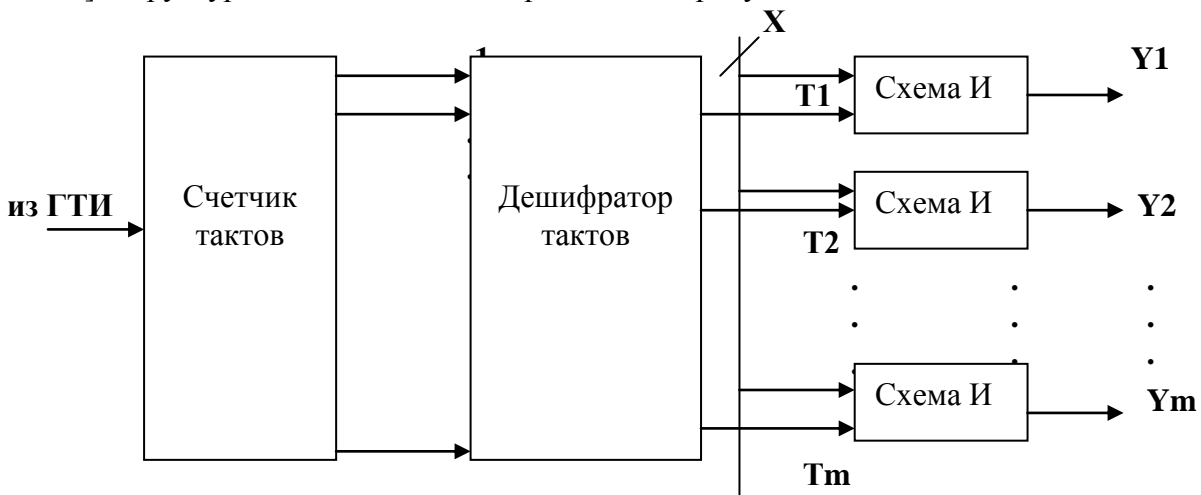


Рисунок 2. - Структура УЧ устройства

Временная диаграмма, поясняющая работу УЧ, приведена в примере Приложения Д (рисунок П10).

Очевидно, что для конкретного блока микропрограммы на ее линейном участке при выходе из операторного блока с номером  $i$  (логического блока нет, т.е. нет и сигнала  $X_i$ ), схема «И» отсутствует; т.е. сигнал  $Y_i$  снимается с линии  $T_i$ ; при выходе из логического блока с номером  $j$  по утверждению - «ДА» - схема «И» имеет вид, представленный рисунком 3а; при наличии подряд двух логических блоков и выходе из  $i$ -того по утверждению ( $X_i$ ), а из  $j$ -того – по отрицанию (инверсия  $j$ ). Будем обозначать инверсию значком #, тогда инверсия  $X_j$  имеет вид « $X_j\#$ », а схема «И» имеет вид, представленный рисунком 3б.

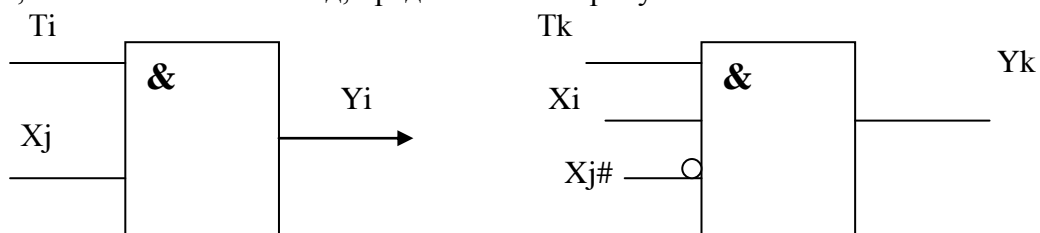


Рисунок 3. – Примеры формирователей сигналов управления



При изображении функциональных схем следует учесть, что входы схем в соответствии с ГОСТ рисуют слева, а выходы – справа от прямоугольника, изображающего схему.

### **2.10 Контрольный пример**

Составить и «прогнать» исходные данные по алгоритму микропрограммы, выполняя над ними только действия, указанные в алгоритме, и в последовательности, соответствующей алгоритму.

Для контрольного примера задаться исходными данными в двоичной системе счисления и записать весь процесс получения конечного результата преобразования из исходных данных. Проверить корректность результата, например, представив исходные данные в десятичной системе счисления и выполнив над ними требуемые по заданию преобразования. Если результат контрольного примера будет не совпадать с результатом проверки, то надо вернуться на первый этап курсовой работы, проанализировать свою работу, найти и исправить ошибки.

Для контроля работы УЧ надо построить временную диаграмму работы УЧ (см. пример в приложении Д рисунок П10).

### **2.11 Составление текста и оформление курсовой работы**

Составить полные тексты исследовательской и основной частей курсовой работы. Исследовательская часть должна содержать не только анализ объекта разработки, но и выводы, необходимые для выполнения основной части курсовой работы. Основную часть работы описывать поэтапно, аргументируя все решения и результаты, ссылаясь на выводы исследовательской части.

Оформление курсовой работы выполнять в строгом соответствии с ГОСТ 7.32-2001 (см. в Приложении А)

### **3 Критерии оценки курсовой работы**

Оценка, выставляемая за курсовую работу, определяется следующими факторами

- уровнем соответствия работы заданию;
- наличием или отсутствием ошибок разработки;
- степенью последовательности, логичности, аргументированности изложения текста;
- качеством исполнения иллюстраций;
- соответствием оформления работы требованиям существующего ГОСТ;
- разницей между фактическим и плановым сроками сдачи готовой курсовой работы преподавателю.

## Список использованной литературы

1. Микропроцессорные системы: Уч. пособие для вузов/ Под общ.ред Д.В.Пузанкова – СПб.: Политехника, 2002.-935 с.
2. Угрюмов Е.П. Цифровая схемотехника. - СПб.: БХВ-Петербург, 2002 .– 518с.
3. Цилькер Б.Я. , Орлов С.А. Организация ЭВМ и систем.-Спб.: Питер, 2004.-668 с.
4. Столлингс В. Структурная организация и архитектура компьютерных систем, 5-е изд.: Пер. с англ. – М.: Издательский дом «Вильямс», 896 с.
5. Каган Б.М. ЭВМ и системы. – М.: Энергоатомиздат, 1991 – 592 с.
6. Новиков Ю.В. Основы цифровой схемотехники. – М.: Мир, 2001 - 379с.
7. Алексенко А.Г. Основы микросхемотехники. – М.: ЮНИМЕДИАСТАЙЛ, 2002 - 448с.
8. Пятибратов А.П. Вычислительные системы, сети и телекоммуникации.- М.: Финансы и статистика, 2002-512 с.
9. Бройдо Вычислительные системы, сети и телекоммуникации.- Спб.: Питер, 2002-438 с

**Правила составления и оформления курсовой работы (по ГОСТ 7.32-2001)**

**1. Структурные элементы отчета**

Структурными элементами курсовой работы (далее – КР) являются:

- титульный лист;
- задание на КР;
- реферат;
- содержание;
- обозначения и сокращения;
- введение;
- основная часть;
- заключение;
- список использованных источников.

**2. Требования к содержанию структурных элементов отчета**

**2.1 Титульный лист**

2.1.1 Титульный лист является первой страницей отчета о КР и служит источником информации, необходимой для обработки и поиска документа.

2.1.2 Титульный лист следует оформлять в соответствии с приложением Б.

**2.2 Реферат**

2.2.1 Реферат должен содержать:

- сведения об объеме отчета, количестве иллюстраций, таблиц, приложений, количестве частей отчета, количестве использованных источников;
- перечень ключевых слов;
- текст реферата.

2.2.1.1 Перечень ключевых слов должен включать от 5 до 15 слов или словосочетаний из текста отчета, которые в наибольшей мере характеризуют его содержание и обеспечивают возможность информационного поиска. Ключевые слова приводятся в именительном падеже и печатаются строчными буквами в строку через запятые.

2.2.1.2 Текст реферата должен отражать:

- объект исследования или разработки;
- цель работы;
- методологию и методы проведения работы;
- результаты работы с параметрами и характеристиками разработанного или исследуемого устройства;
- предположения о развитии объекта исследования.

Если отчет не содержит сведений по какой-либо из перечисленных структурных частей реферата, то в тексте реферата она опускается, при этом последовательность изложения сохраняется.

2.2.2 Пример составления реферата приведен в приложении В.

**2.3 Содержание курсовой работы**

2.3.1 Содержание включает введение, наименование всех разделов, подразделов, пунктов (если они имеют наименование), заключение, список использованных источников и наименование приложений с указанием номеров страниц, с которых начинаются эти элементы отчета о КР.

2.3.2 В отчете о КР объемом не более 10 страниц содержание допускается не составлять.

## **2.4 Обозначения и сокращения**

2.4.1 Структурный элемент «Обозначения и сокращения» содержит перечень обозначений и сокращений, применяемых в данной КР.

2.4.2 Запись обозначений и сокращений проводят в порядке приведения их в тексте отчета с необходимой расшифровкой и пояснениями.

## **2.5 Введение**

Введение должно содержать данные об исследовании предметной области работы: определения, параметры и назначение устройства, классификацию, режимы работы и место устройства в ЭВМ.

## **2.6 Основная часть**

В основной части отчета приводят данные, отражающие сущность, методику и основные результаты выполненной КР:

## **2.7 Заключение**

Заключение должно содержать:

- краткие выводы по результатам выполненной КР или отдельных ее этапов;
- оценку полноты решений поставленных задач;
- проблемы и трудности, имевшие место при выполнении КР.

## **2.8 Список использованных источников**

Список должен содержать сведения об источниках, использованных при составлении отчета. Сведения об источниках приводятся в соответствии с требованиями ГОСТ. Пример оформления списка приведен в списке рекомендуемой литературы.

## **3. Правила оформления**

### **3.1 Общие требования**

3.1.1 Страницы текста отчета о КР и включенные в отчет иллюстрации и таблицы должны соответствовать формату А4 по ГОСТ 9327.

3.1.2 Отчет о КР должен быть выполнен с использованием ПЭВМ и принтера на одной стороне листа белой бумаги формата А4 через полтора интервала. Цвет шрифта должен быть черным, высота букв, цифр и других знаков — не менее 1,8 мм (кегель - не менее 12).

Текст отчета следует печатать, соблюдая следующие размеры полей: правое — 10 мм, верхнее — 20 мм, левое и нижнее — 20 мм.

Разрешается использовать компьютерные возможности акцентирования внимания на определенных терминах, формулах, теоремах, применяя шрифты разной гарнитуры.

3.1.3 Вне зависимости от способа выполнения отчета качество напечатанного текста и оформления иллюстраций, таблиц, распечаток с ПЭВМ должно удовлетворять требованию их четкого воспроизведения.

3.1.4 При выполнении отчета необходимо соблюдать равномерную плотность, контрастность и четкость изображения по всему отчету. В отчете должны быть четкие, не расплывшиеся линии, буквы, цифры и знаки.

3.1.5 Опечатки, описки и графические неточности, обнаруженные в процессе подготовки отчета, допускается исправлять подчисткой или закрашиванием белой краской и нанесением на том же месте исправленного текста (графики) машинописным способом или черными чернилами, пастой или тушью — рукописным способом.

Повреждения листов текстовых документов, помарки и следы не полностью удаленного прежнего текста (графики) не допускаются.

3.1.6 Сокращение русских слов и словосочетаний в отчете — по ГОСТ 7.12.

### **3.2 Построение отчета**

3.2.1 Наименования структурных элементов отчета, «Реферат», «Содержание», «Обозначения и сокращения», «Введение», «Заключение», «Список использованных источников» служат заголовками структурных элементов отчета.

3.2.2 Основную часть отчета следует делить на разделы, подразделы и пункты. Пункты, при необходимости, могут делиться на подпункты. При делении текста отчета на пункты и подпункты необходимо, чтобы каждый пункт содержал законченную информацию.

3.2.3 Разделы, подразделы, пункты и подпункты следует нумеровать арабскими цифрами и записывать с абзацного отступа.

Разделы должны иметь порядковую нумерацию в пределах всего текста, за исключением приложений.

*Пример — 1, 2, 3 и т. д.*

Номер подраздела или пункта включает номер раздела и порядковый номер подраздела или пункта, разделенные точкой.

*Пример — 1.1, 1.2, 1.3 и т. д.*

Номер подпункта включает номер раздела, подраздела, пункта и порядковый номер подпункта, разделенные точкой.

*Пример - 1.1.1.1, 1.1.1.2, 1.1.1.3 и т. д.*

После номера раздела, подраздела, пункта и подпункта в тексте точку не ставят.

3.2.4 Заголовки разделов, подразделов и пунктов следует печатать с абзацного отступа с прописной буквы без точки в конце, не подчеркивая.

Если заголовок состоит из двух предложений, их разделяют точкой.

### **3.3 Нумерация страниц отчета**

3.3.1 Страницы отчета следует нумеровать арабскими цифрами, соблюдая сквозную нумерацию по всему тексту отчета. Номер страницы проставляют в центре нижней части листа без точки.

3.3.2 Титульный лист включают в общую нумерацию страниц отчета. Номер страницы на титульном листе не проставляют.

3.3.3 Иллюстрации и таблицы, расположенные на отдельных листах, включают в общую нумерацию страниц отчета.

Иллюстрации и таблицы на листе формата А3 учитывают как одну страницу.

### **3.4 Нумерация разделов, подразделов, пунктов, подпунктов отчета**

3.4.1 Разделы отчета должны иметь порядковые номера в пределах всего документа, обозначенные арабскими цифрами без точки и записанные с абзацного отступа. Подразделы должны иметь нумерацию в пределах каждого раздела. Номер подраздела состоит из номеров раздела и подраздела, разделенных точкой. В конце номера подраздела точка не ставится. Разделы, как и подразделы, могут состоять из одного или нескольких пунктов.

3.4.2 Если документ не имеет подразделов, то нумерация пунктов в нем должна быть в пределах каждого раздела, и номер пункта должен состоять из номеров раздела и пункта, разделенных точкой. В конце номера пункта точка не ставится.

*Пример*

*1 XXXXXXXXXXXXXXXXXXXX*  
*1.1* }

## **1.2 Нумерация пунктов первого раздела документа**

### **1.3**

**2 XXXXXXXXXXXXXXXXXXXX**

#### **2.1**

#### **2.2**

#### **2.3**

**Нумерация пунктов второго раздела документа**

Если документ имеет подразделы, то нумерация пунктов должна быть в пределах подраздела и номер пункта должен состоять из номеров раздела, подраздела и пункта, разделенных точками, например:

### **3 Методы испытаний**

#### **3.1 XXXXXXXXXXXXX**

##### **3.1.1**

##### **3.1.2**

##### **3.1.3**

**Нумерация пунктов первого подраздела третьего раздела документа**

#### **3.2 XXXXXXXXXXXXX**

##### **3.2.1**

##### **3.2.2**

##### **3.2.3**

**Нумерация пунктов второго подраздела третьего раздела документа**

3.4.3 Если раздел состоит из одного подраздела, то подраздел не нумеруется. Если подраздел состоит из одного пункта, то пункт не нумеруется. Наличие одного подраздела в разделе эквивалентно их фактическому отсутствию.

3.4.4 Если текст отчета подразделяется только на пункты, то они нумеруются порядковыми номерами в пределах всего отчета.

3.4.5 Пункты, при необходимости, могут быть разбиты на подпункты, которые должны иметь порядковую нумерацию в пределах каждого пункта, например 4.2.1.1, 4.2.1.2, 4.2.1.3 и т. д.

4.4.6 Внутри пунктов или подпунктов могут быть приведены перечисления.

Перед каждым перечислением следует ставить дефис или, при необходимости ссылки в тексте документа на одно из перечислений, строчную букву (за исключением ё, з, о, г, ь, й, ы, ъ), после которой ставится скобка.

Для дальнейшей детализации перечислений необходимо использовать арабские цифры, после которых ставится скобка, а запись производится с абзацного отступа, как показано в примере.

### **Пример**

**а)** \_\_\_\_\_

**б)** \_\_\_\_\_

**1)** \_\_\_\_\_

**2)** \_\_\_\_\_

**в)** \_\_\_\_\_

3.4.7 Если отчет состоит из двух и более частей, каждая часть должна иметь свой порядковый номер. Номер каждой части следует проставлять арабскими цифрами на титульном листе под указанием вида отчета, например, «Часть 2».

3.4.8 Каждый структурный элемент отчета следует начинать с нового листа (страницы).

3.4.9 Нумерация страниц отчета и приложений, входящих в состав отчета, должна быть сквозная.

## **3.5 Иллюстрации**

3.5.1 Иллюстрации (чертежи, графики, схемы, компьютерные распечатки, диаграммы, фотоснимки) следует располагать в отчете непосредственно после текста, в котором они упоминаются впервые, или на следующей странице.

Иллюстрации могут быть в компьютерном исполнении, в том числе и цветные.

На все иллюстрации должны быть даны ссылки в тексте КР.

3.5.2 Чертежи, графики, диаграммы, схемы, иллюстрации, помещаемые в отчете, должны соответствовать требованиям государственных стандартов Единой системы конструкторской документации (ЕСКД).

Допускается выполнение чертежей, графиков, диаграмм, схем посредством использования компьютерной печати.

3.5.3 Фотоснимки размером меньше формата А4 должны быть наклеены на стандартные листы белой бумаги.

3.5.4 Иллюстрации, за исключением иллюстрации приложений, следует нумеровать арабскими цифрами сквозной нумерацией.

Если рисунок один, то он обозначается «Рисунок 1». Слово «рисунок» и его наименование располагают посередине строки.

3.5.5 Допускается нумеровать иллюстрации в пределах раздела. В этом случае номер иллюстрации состоит из номера раздела и порядкового номера иллюстрации, разделенных точкой. Например, Рисунок 1.1.

3.5.6 Иллюстрации, при необходимости, могут иметь наименование и пояснительные данные (подрисуночный текст). Слово «Рисунок» и наименование помещают после пояснительных данных и располагают следующим образом: **Рисунок 1 — Детали прибора.**

3.5.7 Иллюстрации каждого приложения обозначают отдельной нумерацией арабскими цифрами с добавлением перед цифрой обозначения приложения. Например, Рисунок А.3.

3.5.8 При ссылках на иллюстрации следует писать «... в соответствии с рисунком 2» при сквозной нумерации и «... в соответствии с рисунком 1.2» при нумерации в пределах раздела.

## 3.6 Таблицы

3.6.1 Таблицы применяют для лучшей наглядности и удобства сравнения показателей. Название таблицы, при его наличии, должно отражать ее содержание, быть точным, кратким. Название таблицы следует помещать над таблицей слева, без абзачного отступа в одну строку с ее номером через тире.

При переносе части таблицы название помещают только над первой частью таблицы, нижнюю горизонтальную черту, ограничивающую таблицу, не проводят.

3.6.2 Таблицу следует располагать в отчете непосредственно после текста, в котором она упоминается впервые, или на следующей странице.

3.6.3 На все таблицы должны быть ссылки в отчете. При ссылке следует писать слово «таблица» с указанием ее номера.

3.6.4 Таблицу с большим количеством строк допускается переносить на другой лист (страницу). При переносе части таблицы на другой лист (страницу) слово «Таблица» и номер ее указывают один раз справа над первой частью таблицы, над другими частями пишут слово «Продолжение» и указывают номер таблицы, например: «Продолжение таблицы 1». При переносе таблицы на другой лист (страницу) заголовок помещают только над ее первой частью.

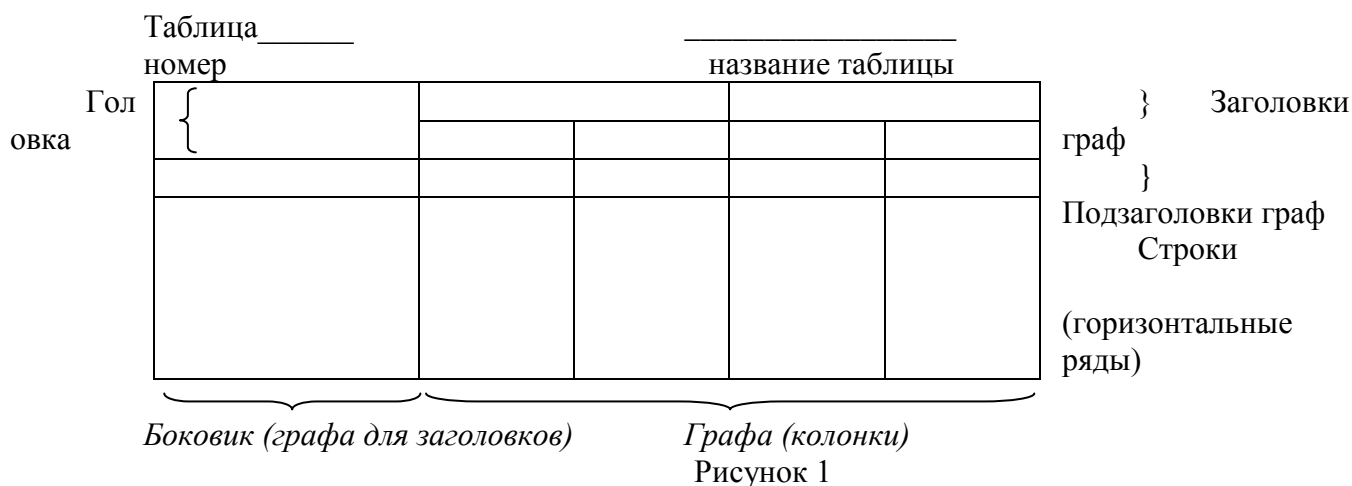
Таблицу с большим количеством граф допускается делить на части и помещать одну часть под другой в пределах одной страницы. Если строки и графы таблицы выходят за формат страницы, то в первом случае в каждой части таблицы повторяется головка, во втором случае — боковик.

Если повторяющийся в разных строках графы таблицы текст состоит из одного слова, то его после первого написания допускается заменять кавычками; если из двух и более слов, то при первом повторении его заменяют словами «То же», а далее — кавычками. Ставить кавычки вместо повторяющихся цифр, марок, знаков, математических и химических символов



не допускается. Если цифровые или иные данные в какой-либо строке таблицы не приводят, то в ней ставят прочерк.

3.6.5 Цифровой материал, как правило, оформляют в виде таблиц. Пример оформления таблицы приведен на рисунке 1.



3.6.6 Таблицы, за исключением таблиц приложений, следует нумеровать арабскими цифрами сквозной нумерацией.

Допускается нумеровать таблицы в пределах раздела. В этом случае номер таблицы состоит из номера раздела и порядкового номера таблицы, разделенных точкой.

Таблицы каждого приложения обозначают отдельной нумерацией арабскими цифрами с добавлением перед цифрой обозначения приложения.

Если в документе одна таблица, то она должна быть обозначена «Таблица 1» или «Таблица В.1», если она приведена в приложении В.

3.6.7 Заголовки граф и строк таблицы следует писать с прописной буквы в единственном числе, а подзаголовки граф — со строчной буквы, если они составляют одно предложение с заголовком, или с прописной буквы, если они имеют самостоятельное значение. В конце заголовков и подзаголовков таблиц точки не ставят.

3.6.8 Таблицы слева, справа и снизу, как правило, ограничивают линиями. Допускается применять размер шрифта в таблице меньший, чем в тексте.

Разделять заголовки и подзаголовки боковика и граф диагональными линиями не допускается.

Горизонтальные и вертикальные линии, разграничивающие строки таблицы, допускается не проводить, если их отсутствие не затрудняет пользование таблицей.

Заголовки граф, как правило, записывают параллельно строкам таблицы. При необходимости допускается перпендикулярное расположение заголовков граф.

Головка таблицы должна быть отделена линией от остальной части таблицы.

3.6.9 Оформление таблиц в отчете должно соответствовать ГОСТ 1.5 и ГОСТ 2.105.

### 3.7 Формулы и уравнения

3.7.2 Пояснение значений символов и числовых коэффициентов следует приводить непосредственно под формулой в той же последовательности, в которой они даны в формуле.

3.7.3 Формулы в отчете следует нумеровать порядковой нумерацией в пределах всего отчета арабскими цифрами в круглых скобках в крайнем правом положении на строке.

**Пример**

$$A=a:B, \tag{1}$$

$$B=c:e. \tag{2}$$

Одну формулу обозначают — (1).

3.7.4 Формулы, помещаемые в приложениях, должны нумероваться отдельной нумерацией арабскими цифрами в пределах каждого приложения с добавлением перед каждой цифрой обозначения приложения, например формула (В.1).

3.7.5 Ссылки в тексте на порядковые номера формул дают в скобках. Пример –... в формуле (1).

3.7.6 Допускается нумерация формул в пределах раздела. В этом случае номер формулы состоит из номера раздела и порядкового номера формулы, разделенных точкой, например (3.1).

3.7.7 Порядок изложения в отчете математических уравнений такой же, как и формул.

3.7.8 В отчете допускается выполнение формул и уравнений рукописным способом черными чернилами.

**3.8 Обязательны ссылки на использованные источники, в том числе на приведенные таблицы и иллюстрации, которые следует приводить в квадратных скобках.**

**3.9 Перечень обозначений и сокращений, условных обозначений, символов, единиц физических величин и терминов.**

Перечень должен располагаться столбцом. Слева в алфавитном порядке приводят сокращения, условные обозначения, символы, единицы физических величин и термины, справа — их детальную расшифровку.

### **3.10 Список использованных источников**

Сведения об источниках следует располагать в порядке появления ссылок на источники в тексте отчета и нумеровать арабскими цифрами без точки и печатать с абзачного отступа.

### **3.11 Приложения**

3.11.1 Приложение оформляют как продолжение данного документа на последующих его листах или выпускают в виде самостоятельного документа.

3.11.2 В тексте документа на все приложения должны быть даны ссылки. Приложения располагают в порядке ссылок на них в тексте документа, за исключением справочного приложения «Библиография», которое располагают последним.

3.11.3 Каждое приложение следует начинать с новой страницы с указанием наверху посередине страницы слова «Приложение», его обозначения и степени.

Приложение должно иметь заголовок, который записывают симметрично относительно текста с прописной буквы отдельной строкой.

3.11.4 Приложения обозначают заглавными буквами русского алфавита, начиная с А, за исключением букв Ё, З, Й, О, Ч, Ь, Ы, Ъ. После слова «Приложение» следует буква, обозначающая его последовательность.

Допускается обозначение приложений буквами латинского алфавита, за исключением букв I и O.

В случае полного использования букв русского и латинского алфавитов допускается обозначать приложения арабскими цифрами.

Если в документе одно приложение, оно обозначается «Приложение А».

3.11.5 Текст каждого приложения, при необходимости, может быть разделен на разделы, подразделы, пункты, подпункты, которые нумеруют в пределах каждого приложения. Перед номером ставится обозначение этого приложения.

3.11.6 Приложения должны иметь общую с остальной частью документа сквозную нумерацию страниц.

**Образец титульного листа курсовой работы**

Министерство образования и науки Российской Федерации  
Федеральное государственное бюджетное образовательное учреждение  
высшего образования  
«Пермский национальный исследовательский политехнический университет»  
Лысьвенский филиал

**Направление:** 09.03.01 Информатика и вычислительная техника

**КУРСОВАЯ РАБОТА**

**По дисциплине «ЭВМ и периферийные устройства»**

**На тему: « »**

Выполнил:

студент группы \_\_\_\_\_

И.О.Фамилия \_\_\_\_\_  
(Подпись)

Руководитель:

\_\_\_\_\_

\_\_\_\_\_ (Подпись)

Курсовая работа допущена к защите

«\_\_» \_\_\_\_\_ 20\_\_ г. \_\_\_\_\_

Курсовая работа защищена \_\_\_\_\_

«\_\_» \_\_\_\_\_ 20\_\_ г. \_\_\_\_\_

Лысьва 20\_\_ г.

**Бланк задания на выполнение курсовой работы**  
Министерство образования и науки Российской Федерации  
Федеральное государственное бюджетное образовательное учреждение  
высшего образования  
«Пермский национальный исследовательский политехнический университет»  
Лысьвенский филиал  
Кафедра Естественнонаучных дисциплин

«УТВЕРЖДАЮ»

Зав.кафедрой ЕН

\_\_\_\_\_ И.Т. Мухаметьянов

«\_\_\_\_\_» \_\_\_\_\_ 20\_\_ год

**ЗАДАНИЕ**  
**на выполнение курсовой работы**

Фамилия И.О. \_\_\_\_\_

Факультет Высшего образования      Группа \_\_\_\_\_

Начало выполнения работы: \_\_\_\_\_

Контрольные сроки просмотра работы: \_\_\_\_\_

Защита работы: \_\_\_\_\_

1. Наименование темы: «\_\_\_\_\_».

2. Исходные данные к работе (проекта):

    Объект исследования – \_\_\_\_\_

    Предмет исследования – \_\_\_\_\_

    Цель работы (проекта) – \_\_\_\_\_

3. Содержание:

\_\_\_\_\_  
\_\_\_\_\_

Руководитель курсовой работы \_\_\_\_\_ Д.С. Курушин

Задание получил \_\_\_\_\_ (\_\_\_\_\_)

КАЛЕНДАРНЫЙ ГРАФИК ВЫПОЛНЕНИЯ  
КУРСОВОЙ РАБОТЫ (ПРОЕКТА)

№ пп	Этапы работы	Объём этапа, %	Сроки выполнения		Примечание
			Начало	Конец	
1.					
2.					
3.					
4.					
5.					
6.					
7.					
8.					
9.					
10.					
11.					

Руководитель курсовой работы \_\_\_\_\_ ( \_\_\_\_\_ )  
«\_\_» \_\_\_\_\_ 20\_\_ года

**Пример составления реферата к курсовой работе**

Реферат

Отчет 16 с., 6 рис., 2 табл., 3 источника.

Формат числа с плавающей точкой; мантисса; порядок; дополнительный код; вычитание кодов двоичных чисел; сумматор; микропрограмма; микрооперация.

Цель работы – разработка алгоритма работы и структуры цифрового вычитателя двоичных чисел, представленных в дополнительных кодах в форме с плавающей десятичной точкой.

При разработке устройства использовались концепции «черного ящика», т.е. первоначальное определение общих функций устройства и системы входных и выходных сигналов. В основе дальнейшей работы с «черным ящиком» использовался принцип декомпозиции, т.е. последовательное разложение функций на подфункции до получения описания функций на элементарном уровне.

В результате работы была составлен алгоритм работы, структура цифрового вычитателя, фрагмент функциональной схемы операционной части устройства и фрагмент функциональной схемы управляющей части устройства вычитания.

Приведены контрольный пример в числовой форме и временная диаграмма работы УЧ вычитателя.

## ПРИЛОЖЕНИЕ Д

### Варианты заданий на курсовую работу

1. Разработать устройство для коррекции ошибок передачи восьмиразрядного двоичного кода методом Хемминга.
2. Разработать УУ выполнением четырёх команд (сложения, вычитания, сравнения и поразрядного логического «и» (конъюнкции)). Формат команды: КОП A1, A1; разрядность операндов и результата - 8 бит, разрядность A1 и A2 - 3. Использовать следующие способы размещения операндов: регистр-регистр, регистр - непосредственно операнд, регистр - косвенная регистровая адресация памяти. Результат операции сохранять по адресу A1
3. Разработать операционный блок процессора для выполнения четырёх команд (сложения, вычитания, сравнения и поразрядного логического «и» (конъюнкции)). Формат команды: КОП A1, A1; разрядность операндов и результата - 8 бит, разрядность A1 и A2 - 3. Использовать следующие способы размещения операндов: регистр-регистр, регистр - непосредственно операнд, регистр - косвенная регистровая адресация памяти. Результат операции сохранять по адресу A1.
4. Разработать УУ выполнением четырёх команд (сложения, вычитания, сравнения и поразрядного логического «и» (конъюнкции)). Формат команды КОП A1, A2; разрядность операндов и результата - 8 бит, разрядность A1 и A2 = 3. Использовать следующие способы размещения операндов: регистр - косвенная регистровая адресация памяти, косвенная регистровая адресация памяти - регистр. Результат операции сохранять по адресу A1.
5. Разработать операционный блок процессора для выполнения четырёх команд (сложения, вычитания, сравнения и поразрядного логического «и» (конъюнкции)). Формат команды КОП A1 A2; разрядность операндов и результата - 8 бит, разрядность A1 и A2 - 3. Использовать следующие способы размещения операндов: регистр - косвенная регистровая адресация памяти, косвенная регистровая адресация памяти - регистр. Результат операции сохранять по адресу A1.
6. Разработать УЧ устройства для перевода двухразрядных чисел в двоичную систему счисления.
7. . Разработать ОЧ устройства для перевода двухразрядных чисел в двоичную систему счисления.
8. Разработать умножитель двоичных целых шестиразрядных чисел в прямом коде.
9. Разработать УЧ сумматора двоичных вещественных чисел, представленных в прямом коде в следующем формате: мантисса- 8 разрядов, порядок- 3 разряда.
10. Разработать ОЧ сумматора двоичных вещественных чисел, представленных в прямом коде в следующем формате: мантисса- 8 разрядов, порядок- 3 разряда.
- 11 Разработать УЧ умножителя двоичных вещественных чисел, представленных в прямом коде в следующем формате: мантисса- 8 разрядов, порядок- 3 разряда.
12. Разработать УЧ умножителя двоичных вещественных чисел, представленных в прямом коде в следующем формате: мантисса- 8 разрядов, порядок- 3 разряда.

13. Разработать контроллер КЭШ-памяти с прямым способом отображения основной памяти (ОЗУ) на КЭШ и со сквозной записью.

14. Разработать контроллер КЭШ-памяти с прямым способом отображения основной памяти (ОЗУ) на КЭШ и с обратной записью.

15. Разработать контроллер КЭШ-памяти с полностью ассоциативным отображением основной памяти (ОЗУ) на КЭШ сквозной записью и алгоритмом замещения по принципу «первый вошёл, первый вышел» (или FIFO).

16. Разработать контроллер КЭШ-памяти с обратной записью и алгоритмом замещения по принципу наименее часто использовавшейся строки КЭШ (или LFU).

17. Разработать контроллер КЭШ-памяти с полностью ассоциативным отображением основной памяти (ОЗУ) на КЭШ, сквозной записью и алгоритмом замены наиболее давнего использования строки КЭШ.

18. Разработать контроллер прерываний на 6 входов по алгоритму организации прерываний на IBM PC. Предусмотреть возможность программной настройки  $2^x$  типов системы приоритетов: неизменного и изменяемого с циклической сменой приоритетов после каждого обслуживания.

19. Разработать контроллер прерываний на 5 входов по алгоритму организации прерываний на IBM PC. Предусмотреть возможность аппаратной настройки  $3^x$  типов системы приоритетов: с циклической сменой приоритетов по убыванию; циклической сменой приоритетов по возрастанию, начиная с обслуженного; с присвоением младшего приоритета обслуженному, а его приоритета – устройству с младшим уровнем приоритета.

20. Разработать УУ выполнением трёх команд обработки целых восьмиразрядных двоичных чисел (поразрядного логического «или», исключаяющего «или» и сложения). Формат команды: КОП A1, A2. Использовать следующие способы размещения операндов: регистр – память (прямая адресация), память – регистр (прямая адресация), регистр – непосредственно операнд. Разрядность адресов – 8 бит. Результат операции сохранять по адресу A1.

21. Разработать операционный блок процессора для выполнения трёх команд обработки целых восьмиразрядных двоичных чисел (поразрядного логического «или», исключаяющего «или» и сложения). Формат команды: КОП A1, A2. Использовать следующие способы размещения операндов: регистр – память (прямая адресация), память – регистр (прямая адресация), регистр – непосредственно операнд. Разрядность адресов – 8 бит. Результат операции сохранять по адресу A1

22. Разработать преобразователь пятиразрядного двоичного кода в систему кодирования ASCII.

23. Разработать преобразователь двухразрядного десятичного числа, представленного в коде ASCII, в двоичную систему счисления.

24. Разработать стек с глубиной стека 16 байт.

25. Разработать контроллер ассоциативной памяти, хранящей  $64 \cdot 8$ - разрядных двоичных кодов с выходом по «равно признаку» и «не равно признаку». Признаком поиска может быть 1,2,4,8 двоичных разрядов. Результатом поиска, выводимым на ШД, является весь байт.



26. Разработать контроллер ассоциативной памяти, хранящей 32\*8 разрядных двоичных кодов с выходом по «больше признака», «меньше признака». Признаком поиска могут быть только нечётные биты восьмиразрядного кода. Результатом поиска, выводимым на ШД, является младшая тетрада байта.

27. Разработать контроллер ассоциативной памяти, хранящей 32\*8 разрядных двоичных слов с выходом по «не больше» и «равно». Признаком поиска является байт или четыре любых бита. Результатом поиска, выводимым на ШД, является старшая тетрада байта.

28. Разработать ускоренный умножитель по алгоритму Бута. Разрядность сомножителей - восемь бит со знаком.

29. Разработать ускоренный умножитель по алгоритму Лемана. Разрядность сомножителей - восемь бит без знака.

30. Разработать контроллер арбитража запросов на управление шиной (централизованный арбитр) на 5 ведущих устройств, с ограничением времени захвата шины с переменной системой приоритетов ведущих устройств по циклическому принципу.

31. Разработать контроллер арбитража запросов на управление шиной (централизованный арбитр) на 8 ведущих устройств с неименной системой приоритетов ведущих и разным временем ограничения захвата шины, причём временем убывающим с возрастанием приоритета.

32. Разработать контроллер прямого доступа к памяти для 5 устройств с программируемыми начальным адресом и размером блока памяти для каждого устройства, со сменой приоритета между устройствами по правилу: устройство, захватившее шину, получает низший уровень приоритета, а имевшее низший – получает высший уровень. Для остальных устройств приоритеты неизменны.

33. Разработать конвертера двухразрядного восьмеричного числа в двоичное число.

34. Разработать конвертер шестиразрядного двоичного числа в восьмеричное.

35. Разработать УУ процессора, выполняющего три команды: ввода из порта (IN ASP), вывода в порт (OUT ASP), и вводом/выводом по прерываниям (INT n).

36. Разработать операционный блок процессора для выполнения трёх команд: ввода из порта (IN ASP), вывода в порт (OUT ASP), и вводом/выводом по прерываниям (INT n).

37. Разработать УУ процессора, выполняющего три команды: сложение содержимого двух регистров процессора, ввода из порта и инверсии регистра. Самостоятельно определить разрядность регистров и портов, а также достаточную разрядность адресов портов.

38. Разработать контролер динамического ОЗУ для любого типа асинхронных динамических микросхем памяти.

Пример выполнения курсовой работы

**Задание**

Разработать устройство выполнения операции вычитания двоичных чисел, представленных в дополнительном коде в форме с плавающей десятичной точкой (запятой). Мантисса имеет размер 10 разрядов после запятой, порядок - 4 разряда.

**Реферат**

(Текст реферата см. в Приложении В)

**Обозначения и сокращения**

ЧЯ	Черный ящик
УЧ	Управляющая часть устройства
ОЧ	Операционная часть устройства
АЛУ	Арифметико-логическое устройство
М	Мантисса числа
П	Порядок числа
МО	Микрооперация
МПП	Микропрограмма

**1. Исследование предметной области курсовой работы**

**1.1. Представление чисел в форме с плавающей точкой**

Вся информация в ЭВМ представляется, хранится и передается в двоичной системе счисления (цифровом коде). Наименьшей единицей информации ЭВМ является *бит*. Один двоичный разряд (0 или 1) несет информацию размером 1 бит.

В соответствии с заданием покажем на рисунке П1 формат числа в двоичном коде в форме с плавающей точкой, который используется для представления вещественных чисел

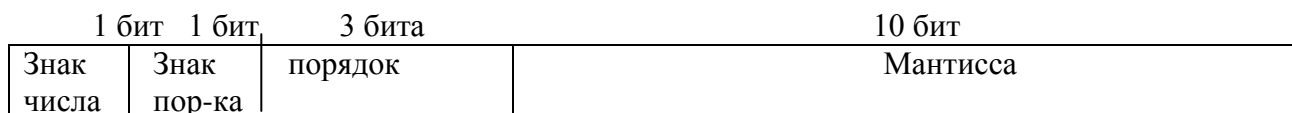


Рисунок П1. – Формат с плавающей точкой для представления вещественных чисел

В крайнем левом бите хранится знак числа, как обычно: 0 – соответствует положительному числу, 1 – отрицательному. В следующих четырех битах хранится значение порядка (П) со знаком. Последнее поле на рисунке П1 отводится для хранения значения мантиссы (М).

Любое число можно представить в М эквивалентны, если считать, что М выражена в двоичной системе счисления:

$$0.101 * 2^5 = 101 * 2^2 = 0.0101 * 2^6$$

Для упрощения алгоритмов выполнения арифметических операций обычно принято нормализовывать М. Нормализованная М числа, отличного от нуля, имеет вид

$$0.1ccc...c^*$$

где *c* представляет произвольную двоичную цифру (0 или 1). Это означает, что нормализованная М всегда меньше 1, а старший (левый) значащий разряд кода мантиссы равен 1.

## 1.2. Арифметические операции над числами с плавающей точкой

Алгоритм выполнения операций сложения над числами с плавающей точкой такой:

1) Уравнять  $P$  операндов, т.е. сделать их одинаковыми, что потребует сдвига положения разделительной точки в  $M$ , т.е. выполнения МО сдвиг  $M$  вправо на один двоичный разряд при увеличении  $P$  на единицу или МО сдвиг  $M$  влево на один двоичный разряд при уменьшении  $P$  на единицу.

2) Сложить дополнительные коды  $M$ , полученные после уравнивания  $P$ .

3) Нормализовать  $M$  разности с одновременным изменением  $P$  разности.

При выполнении операции могут возникнуть следующие особые ситуации.

а) **Переполнение порядка.** Положительный  $P$  результата превышает максимальное значение, предусмотренное форматом.

б) **Потеря значимости порядка.** Отрицательный  $P$  результата меньше минимального значения, допускаемого принятым форматом (например, получен  $P = -10$ , который меньше разрешенного  $-7$ ). Это означает, что получен очень малый результат, который можно считать равным нулю.

в) **Потеря значимости мантиссы.** В процессе уравнивания порядков  $M$  сдвигается настолько далеко вправо, что старший значащий ее разряд выходит за пределы разрядной сетки.

г) **Переполнение мантиссы.** При сложении  $M$  с одинаковыми знаками возможно появление переноса из старшего разряда.

Теперь рассмотрим вычитание в дополнительном коде.

Операция вычитания выполняется по следующему правилу: для вычитания одного числа (вычитаемого) из другого (уменьшаемого) необходимо предварительно выполнить операцию отрицания над вычитаемым, а затем сложить результат с уменьшаемым по правилам сложения в дополнительном коде.

## 1.3. Представление чисел в дополнительном коде

Дополнительный код – это формат числа, который применяется при реализации в АЛУ арифметических операций над числами.

Для перевода отрицательного числа в дополнительный код необходимо:

а) проинвертировать значение двоичного кода исходного числа в каждом разряде

б) сложить образовавшийся код с кодом единицы, т.е. с  $00\dots001b$  по правилам двоичного сложения.

в) записать в знаковый разряд 1.

Например, дополнительный код  $-5$  в формате байта имеет вид:  $11111011b$ .

## 1.4 Типовые схемы для построения ОЧ

Как следует из вышеизложенного, основная арифметическая операция в устройстве это сложение дополнительных двоичных кодов операндов. Причем такое сложение используется не только для обработки  $M$ , но и  $P$ , т.к. для уравнивания  $P$  надо получить разницу между ними путем вычитания, а затем добавить ее к меньшему порядку. Поэтому сумматор – важнейшая комбинационная схема вычитателя, осуществляющая суммирование чисел в двоичном коде. Сумматоры бывают последовательными и параллельными, комбинационными и регистровыми. Можно каскадировать сумматоры для увеличения разрядности. Если сумматор складывает меньшее число двоичных разрядов, чем нужно, берут несколько одинаковых сумматоров и соединяют между собой. Сигнал с выхода переноса сумматора, обрабатывающего младшие разряды, нужно подать на вход переноса сумматора, обрабатывающего старшие разряды. Например, при объединении трех 4-разрядных сумматоров получается 12-разрядный сумматор, имеющий дополнительный 13 разряд (выход переноса).

Другая типовая схема – сдвигатель. Может быть реализован как на сдвигающем регистре, так и как комбинационная схема. Будут использованы и другие комбинационные схемы, например, схема сравнения.

## 2. Разработка устройства

### 2.1 Анализ исходных данных задания на курсовую работу

В задании нет указаний на формат результата (разности). Для расширения возможностей устройства, например, для увеличения точности вычислений, нужно увеличивать разрядность  $M$ , поэтому, кроме 10 разрядов  $M$  разности, должен вырабатываться перенос из этой группы разрядов в следующую. Очевидно, что для этого нужно создать вычитатель с дополнительными входом и выходом по переносу, с помощью которого один вычитатель можно будет соединить с таким же вторым вычитателем.

### 2.2. Спецификация устройства на уровне «черного ящика»

В задании сказано, что устройство предназначено для выполнения операций вычитания чисел, поэтому должно быть два 15-ти разрядных входа  $A$  для уменьшаемого и один 15-ти разрядный выход  $B$  для вычитаемого, вход  $P_i$  для переноса из предыдущего вычитателя и выход  $P_{i+1}$  (рисунок П2).

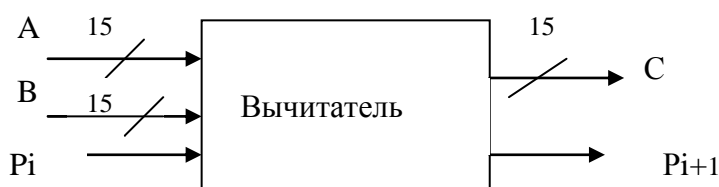


Рисунок П2. –Система выводов вычитателя

Закон вычитания на примере одной строки таблицы вычитания чисел в указанном в задании формате приведен ниже:

Пусть  $M A = +0,1010000000b$        $P A = 0 = 0000$

Пусть  $M B = -0,1100000000b$        $P B = -1 = 1001b$

Переведем исходные данные в десятичную систему счисления и выполним вычитание чисел.

$$MA = +0 \cdot 2^0 + 1 \cdot 2^{-1} + 0 \cdot 2^{-2} + 1 \cdot 2^{-3} = +0,625. \quad PA = 0. \quad \text{Число } A = +0,625 \cdot 2^0 = +0,625.$$

$$MB = -0 \cdot 2^0 + 1 \cdot 2^{-1} + 1 \cdot 2^{-2} = -0,750. \quad PB = -1. \quad \text{Число } B = -0,750 \cdot 2^{-1} = -0,375.$$

$$\text{Итак, } A = +0,625, B = -0,375. \text{ Разность } A - B = +0,625 - (-0,375) = -1.$$

### 2.3 Представление «черного ящика» в виде операционной и управляющей частей.

Упрощенно разрабатываемое устройство можно представить схемой рисунка П3. (Здесь в курсовой работе описать назначение ГТИ, определить разрядность и назначение всех шин. Величины  $k$  и  $m$  пока неизвестны. ) После этого можно разрабатывать структуру ОЧ и выбрать тип сумматора.

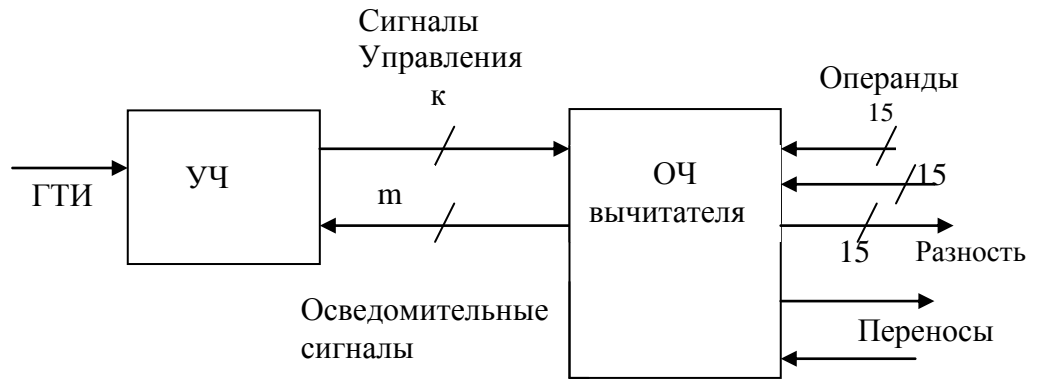


Рисунок П3.—Разбиение схемы вычитателя на ОЧ и УЧ.

#### 2.4. Разработка структуры ОЧ вычитателя

Пусть операнды размещаются в регистре А (уменьшаемое) и в регистре В (вычитаемое), как показано на рисунке П4, причем, и М, и П – в дополнительных кодах. Выделим отдельные блоки обработки М и П. Тогда, если в каждом блоке обработки использовать комбинационный сумматор, оба операнда - в разных МО это или М, или П.- должны поступать на его входы одновременно, т.е. при одном и том же сигнале с ГТИ. В том же такте на выходе сумматора будет устанавливаться алгебраическая сумма (сумма дополнительных кодов слагаемых).

МВ из регистра В, если число В отрицательное, должна поступать на инвертор, где выполняется присвоение ей знака операции вычитания. Присвоение осуществляется инверсией знака МВ, т.е. преобразованием МВ в отрицательное число, если она была положительной и наоборот. Это выполняется, естественно, через преобразования в дополнительном коде. Как следует из п.1.2. и п.2.2. курсовой работы, блоки обработки М и П включают множество схем выполнения различных действий по преобразованию М и П. Но, поскольку алгебраическое суммирование М и П происходит в разные такты, можно для сокращения аппаратных затрат поставить один комбинационный сумматор. В сумматор поступают или М, или П. Результат операции выдается из сумматора в регистр С по выходной информационной шине. Упрощенная структура ОЧ приведена на рисунке П4.

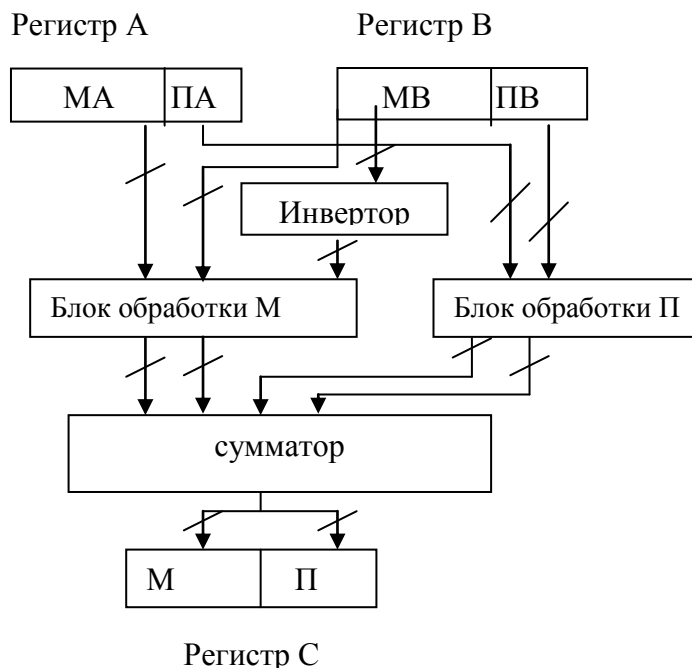


Рисунок П4 .—Упрощенная структура ОЧ вычитателя.

## 2.5. Составление алгоритма работы вычитателя

Алгоритм, изложенный п.1 и п. 2 курсовой работ, приведен на рисунке П5. Однако, ясно, что алгоритм еще не доведен до уровня МО. Действительно, список МО нам известен (см. п 2.6. Методуказаний по курсовой работе), и тогда, например, выравнивание П и изменение М нужно заменить такой последовательностью МО:

1. Сравнение П чисел путем вычитания. Если не равны, то - выравнивание порядка, т.е. определение меньшего и увеличение его на разницу между П. Если равны, то нужна запись П результата и переход на обработку М.

2. При выравнивания П необходимо выполнение сдвига М вправо для числа с меньшим П на число двоичных разрядов, равное разнице между П. Выполняется сдвиг именно М числа с меньшим П вправо, что приводит к утере ее младших разрядов. Однако, если бы уравнивание порядков делали путем уменьшения большего П, его М пришлось бы увеличивать, т.е. сдвигать М влево, что привело бы к переполнению, а значит к потере уже старших разрядов М. Т. о., одновременно со сдвигом М право на один двоичный разряд П числа увеличивается на единицу.. Сдвиги выполняются до тех пор, пока значения П обоих чисел не станут равны. Если в результате сдвигов значение корректируемой М станет равным нулю, в качестве результата берется значение М второго операнда. Т. о., если П слагаемых существенно отличаются, меньшее слагаемое в результате не учитывается.

При вычитании М чисел получается М разности. Производится по правилам вычитания чисел с фиксированной точкой, т.е. в дополнительных кодах.

3. Нормализация результата. Нормализация представляет собой серию сдвигов кода М влево (в сторону старших разрядов) с одновременным уменьшением значения П до тех пор, пока значение старшей цифры М не станет отличным от 0.

В операциях над числами в форме с плавающей точкой, в отличие от операций с фиксированной точкой, сложение и вычитание выполняются приближенно, т.к. при выравнивании П происходит потеря младших разрядов одного из слагаемых. Чтобы уменьшить погрешность, применяют округление результата, которое здесь не рассматривается.

## 2 6. Разработка схемы алгоритма работы вычитателя и его микропрограммы

Схема алгоритма вычитания чисел в форме с плавающей десятичной точкой на уровне МО приведена на рисунке П6.

Отметив операторные блоки символами  $Y_i$ , а логические блоки символами  $X_i$ , перейдем на том же рисунке П6 к МПР вычитания. Операторных блоков получилось 16, а логических – 9. Т.е. для ОЧ потребуется 16 сигналов управления из УЧ, а для УЧ – 9 осведомительных сигналов из ОЧ.

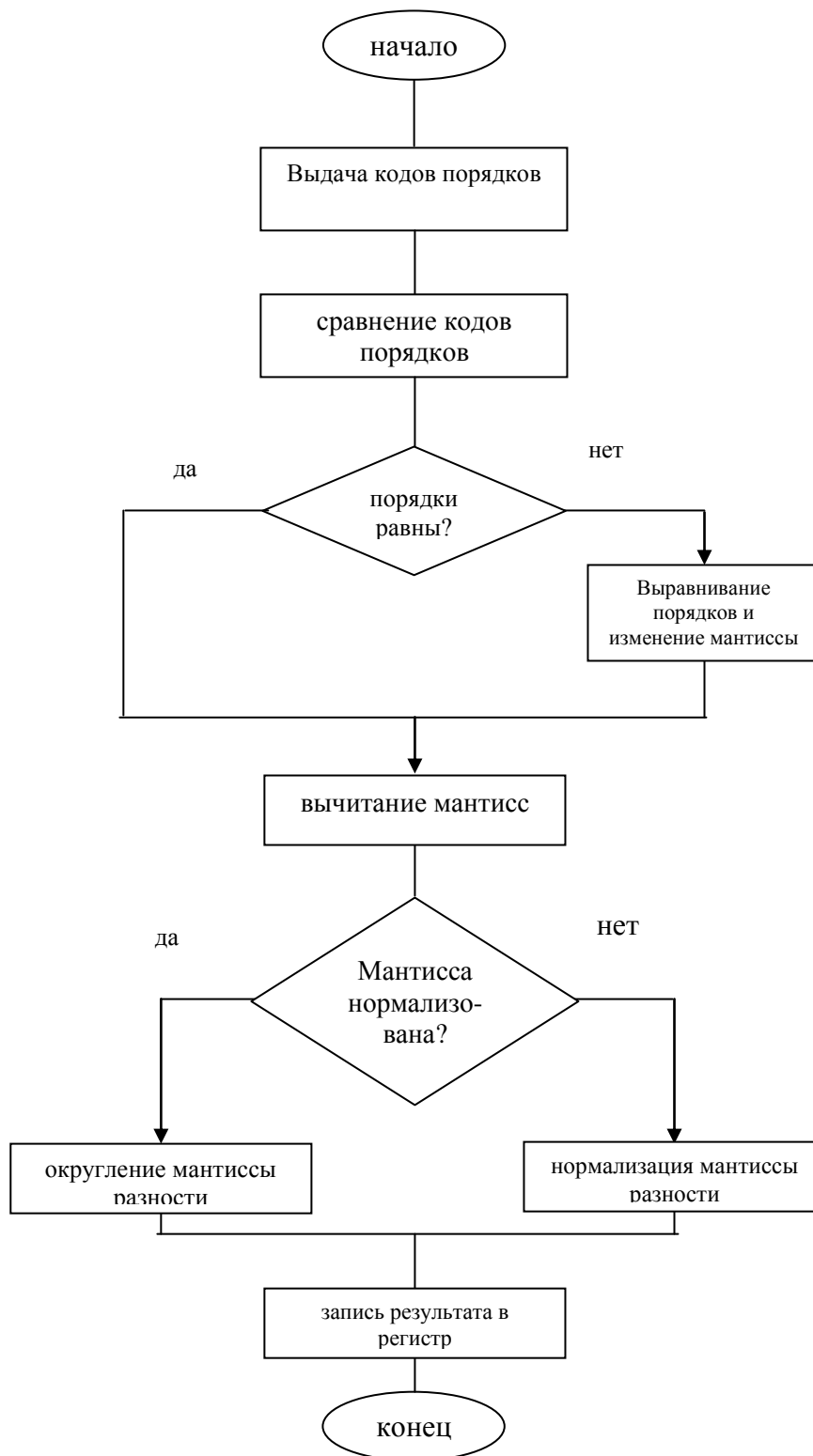


Рисунок П5 --Схема алгоритма работы вычитателя

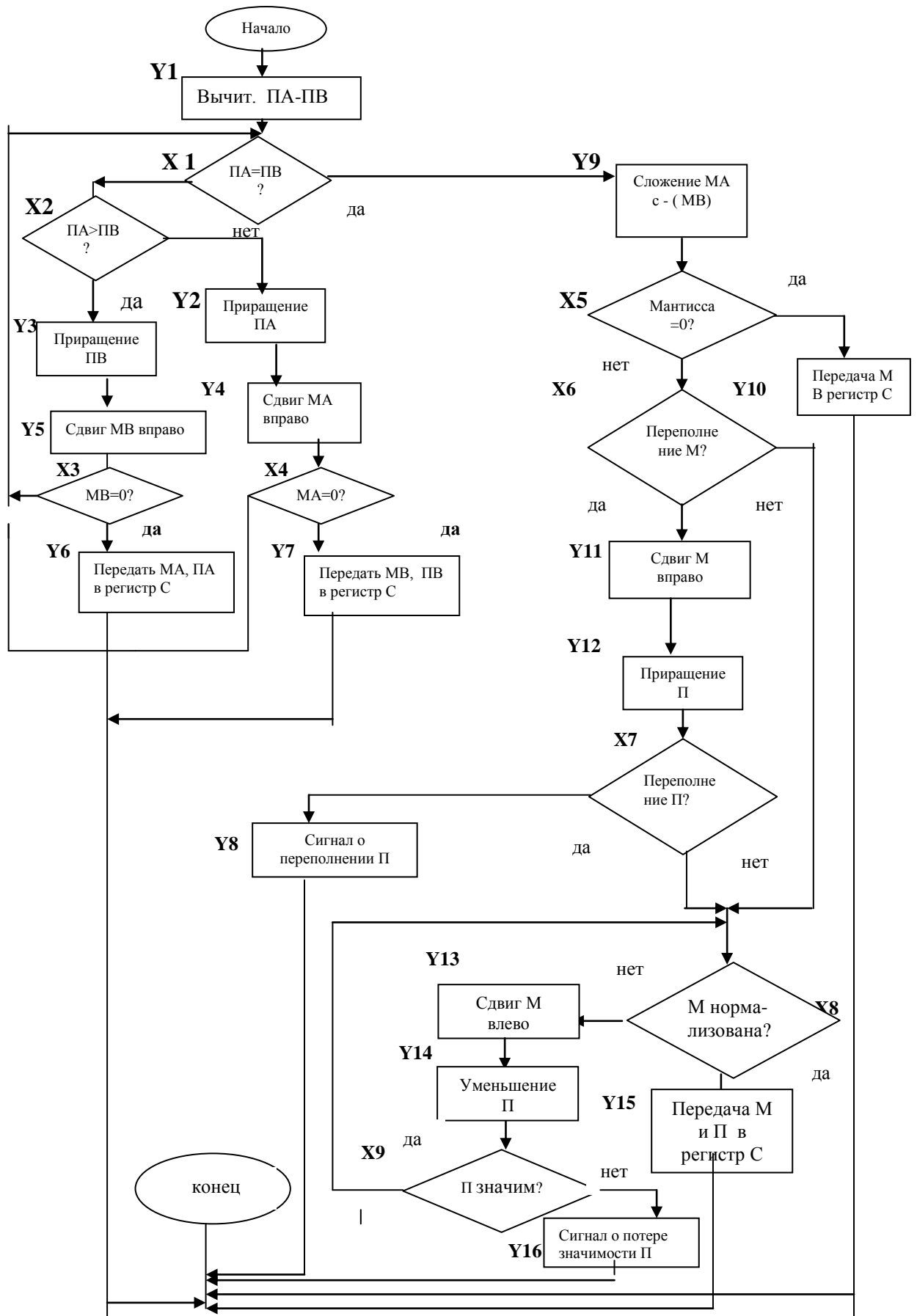


Рисунок П6. – Схема алгоритма операции вычитания на уровне микроопераций



## 2.7. Составление полной спецификации устройства

Опишем в таблице все линии и сигналы, полученные в процессе разработки вычитателя. В данном примере курсовой работы, конечно, приведены не все 31 строки этой таблицы). Пусть активным для каждого сигнала является его высокий (единичный) уровень.

Таблица		Сигналы вычитателя	
Имя сигнала/шины и разрядность	Тип (In/Out)	Назначение сигнала	
МА[0:10]	I	Мантисса уменьшаемого с разрядом знака числа	
$P_i$ МА	I	Вход переноса из предыдущего разряда мантиссы уменьшаемого	
ПА[0:3]	I	Порядок уменьшаемого	
МВ[0:10]	I	Мантисса вычитаемого с разрядом знака числа	
.....	...	.....	
М[0:10]	O	Мантисса разности чисел с разрядом знака числа	
$P_{i+1}$ М	O	П о р я д о к р а з н о с т и ч и с е л	
.....	...	.....	
Y1	I для ОЧ	Сигнал управления из УЧ на выполнение МО сложения порядка ПА с (-) порядка ПВ	
.....			
Y16	I для ОЧ	Сигнал о потере значимости порядка, если произошел его сдвиг вправо более, чем на 4 разряда.	
.....	...	..	
X1	O для ОЧ	Осведомительный сигнал из ОЧ в УЧ, активен, если порядок числа А (ПА) равен порядку числа В (ПВ).	
X2	O для ОЧ	Осведомительный сигнал из ОЧ в УЧ, активен, если порядок числа А (ПА) больше порядка числа В (ПВ).	
.....	..	.....	

Структура вычитателя на микропрограммном уровне управления приведена на рисунке П7

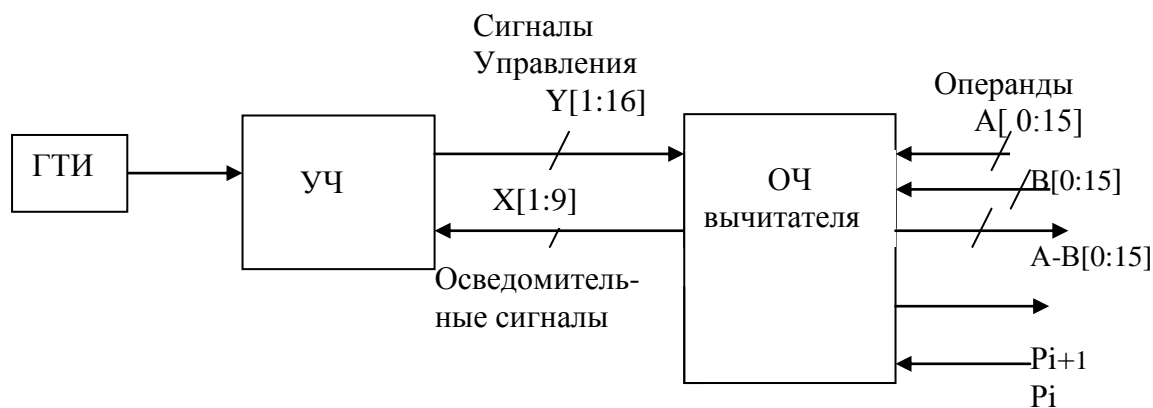


Рисунок П7. - Структура вычитателя на микропрограммном уровне управления

## 2.8. Разработка фрагмента функциональной схемы вычитателя

### 2.8.1. Разработка УЧ

Фрагмент схемы УЧ дан на рисунке П9. Схема составлен ав соответствии с правилами п.2.9.2. данных Методических указаний для фрагмента МПР с блока Y1 по блок Y4 (см. рисунок П6). Всего в МПР 16 МО, как следует из рисунка П6. Значит, суммирующий счетчик должен считать до 4-х, поэтому он должен иметь  $2^4 = 16$  выходов, а дешифратор должен выставить тактовые сигналы последовательно на 16 выходов. Из переходов по МПР следуют значение сигналов Y1. Например,  $Y2 = X1\# \& X2\#$ , а  $Y3 = X1\# \& X2$ .

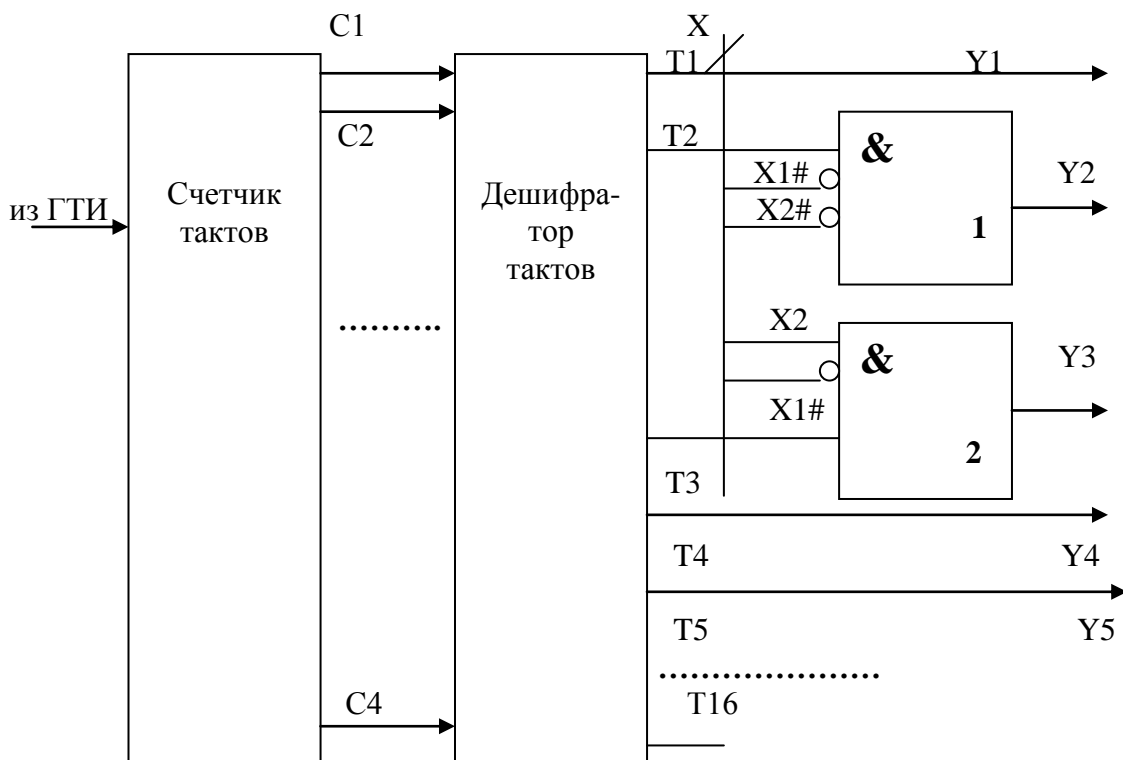


Рисунок П9. - Фрагмент схемы УЧ

### 2.9. Контрольный пример.

Для контрольного примера возьмем исходные данные из п. 2.2. :  $A = +0,625$ , а  $B = -0,375$ .

Тогда в нормализованном виде  $MA = +0,1010000000b$ , а  $PA = 0 = 0000b$ ,

$MB = -0,1100000000b$  или  $1,1100000000b$  в прямом коде, и  $MB = 0,00111111111b$   
 $+0,0000000001b = 1,0100000000b$  в дополнительном коде, а  $PB = -1 = 1001b$  в прямом коде и  
 $PB = 1111b$  в дополнительном коде.

Приведем ниже выполнение всех МО и проверок логических условий.

Y1 :  $PA - PB = 0000b - (1001)b = 0000b + 0001b = 0001b$ .

X1 : Разность  $PA - PB$  не равна 0, т.е.  $PA$  не равен  $PB$ .

X2 : Разность  $PA - PB$  положительна (знаковый разряд = 0), значит,  $PA$  больше  $PB$ ;

Y3 : Т.к.  $PA$  больше  $PB$ ,  $PB$  увеличивается на эту разность, т.е.  $PB + (PA - PB) = PA$ .

Y5 : Для того, чтобы число  $B$  не изменилось,  $MB$  уменьшается на столько порядков, какова разность между  $PA$  и  $PB$ , т.е.  $MB$  надо уменьшить на 1 порядок, т.е. сдвинуть на один двоичный порядок вправо.  $MB$  после сдвига станет  $= -0,0110000000b$  или в дополнительном коде  $= 1,1010000000b$ .

X3 :  $MB = 1,0110000000b$  в прямом коде, т.е. не равна 0.

X1 :  $PA - PB = 0001 - 0001 = 0$ .

Y9 : Т.к.  $MA - MB = MA + (-MB)$ , и  $MB$  – отрицательна, то знак  $MB$  и знак операции вычитания компенсируют друг друга и код  $MA$  суммируется с кодом  $MB$  как положительные значения:

$$\begin{array}{r} MA = 0,1010000000b \\ + \\ MB = 0,0110000000b \\ \hline \end{array}$$

$MA+MB = 1,0000000000b$ , единица переполнения появилась в знаковом разряде, т.е.  $M = 1$  – не нормализована, а  $\Pi = 0$

X5 : X5 = 0. Сумма  $MA + MB$  не равна 0, т.к. с помощью флага зафиксировано переполнение.

X6 : X6 = 1 (да. Переполнение M).

Y11 : Сдвиг M вправо на 1 разряд, т.е. в результате сдвига  $M = +0,1000000000$ .

Y12 : Приращение  $\Pi$  на 1 т.е. в результате приращения  $\Pi = 1$ .

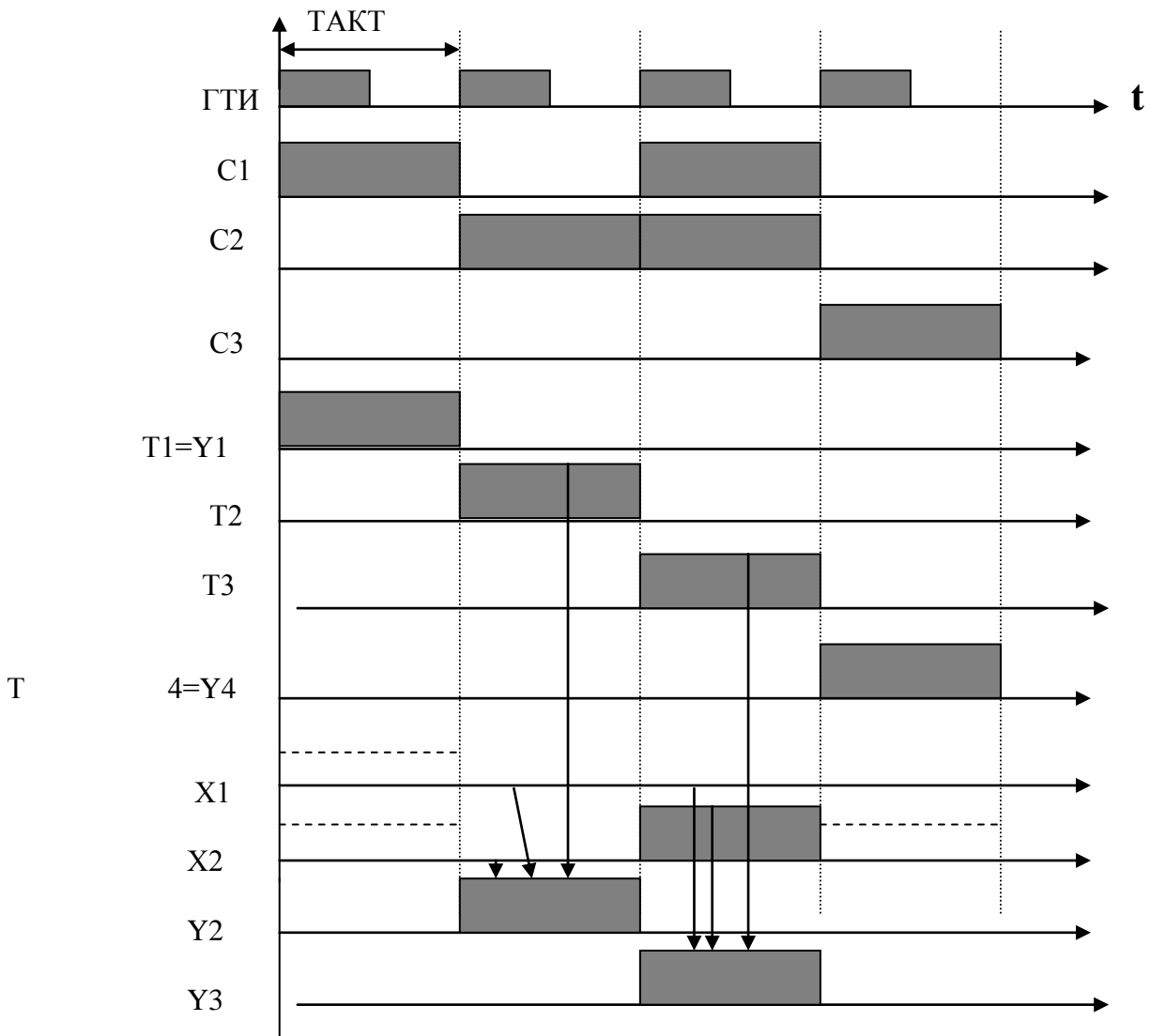
X8 :  $M = +0,1000000000$ , т.е. M нормализована.

Y15 : Передача M и  $\Pi$  как результата в регистр C.  $M = +0,1000000000$ ,  $\Pi = +1$ ,

Разность  $A - B = M * 2^{\Pi} = +0,1000000000 * 2^{+1} = 1$ , что совпадает с результатом, полученным в п. 2.2. курсовой работы.

### 2.10 Временная диаграмма работы УЧ вычитателя

На рисунке П10 приведен фрагмент временной диаграммы работы УЧ вычитателя с блока Y1 до блока Y4.



### Рисунок П10.- Фрагмент временной диаграммы работы УЧ вычитателя

Закрашенные интервалы времени соответствуют логическим 1, горизонтальные штриховые линии указывают интервалы времени, в которых значения X не имеют смысла, т.к. в эти интервалы сигналы X не проверяются в УЧ. Вертикальные штриховые линии разделяют временные такты.

#### ***Заключение***

Задача курсовой работы –разработка на структурно-алгоритми ческом уровне устройства выполнения операций вычитания чисел, представленных в дополнительном коде в форме с плавающей десятичной точкой.

Поставленная задача выполнена. В ходе курсовой работы была изучена специальная литература, разработана структура ОЧ и УЧ устройства, алгоритм их работы, спецификация сигналов, фрагменты функциональных схем ОЧ и УЧ, контрольный числовой пример и временная диаграмма работы устройства. Проблемы, возникавшие в результате выполнения работы, связаны с моей неорганизованностью.