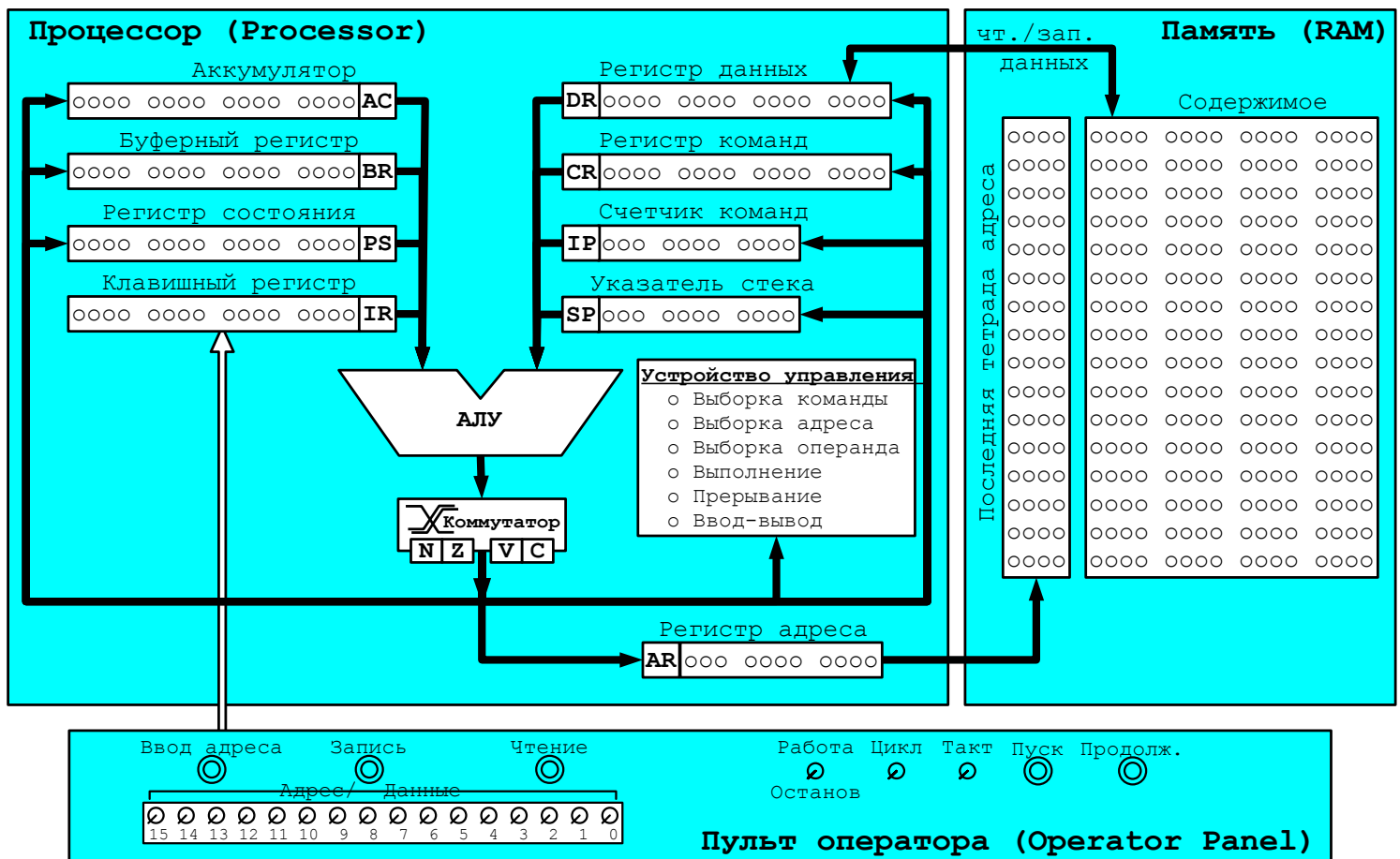


Методические указания к лабораторным работам по курсу
"Основы профессиональной деятельности"



Выходные данные, аннотация

Содержание

Раздел 1. Знакомство с кафедрой вычислительной техники.....	4
Кафедра вычислительной техники.....	4
Лабораторные работы курса ОПД.....	4
Лабораторная работа №1. <i>Основные команды ОС семейства UNIX</i>	5
Рубежный контроль №1.....	6
Раздел 2. Введение в базовую ЭВМ.....	7
Лабораторная работа №2. <i>Исследование работы БЭВМ</i>	7
Лабораторная работа №3. <i>Выполнение циклических программ</i>	9
Рубежный контроль №2.....	9
Лабораторная работа №4. <i>Выполнение комплекса программ</i>	10
Раздел 3. Организация ввода-вывода информации в БЭВМ.....	10
Лабораторная работа №5. <i>Асинхронный обмен данными с ВУ</i>	10
Лабораторная работа №6. <i>Обмен данными с ВУ по прерыванию</i>	11
Раздел 4. Организация микропрограммного устройства БЭВМ.....	13
Лабораторная работа №7. <i>Синтез команд БЭВМ</i>	13
Рубежный контроль №2.....	15
Литература.....	15
Приложение А. Краткий перечень и функциональность команд UNIX.....	16
Приложение Б. Накопление баллов БАРС в течении курса.....	17
Приложение В. Состав, структура и функционирование БЭВМ.....	18
Часть 1. Базовая ЭВМ.....	18
1.1 Назначение базовой ЭВМ.....	18
1.2 Структура базовой ЭВМ.....	18
1.3. Система команд базовой ЭВМ.....	19
1.4 Представление целых чисел в БЭВМ.....	20
1.5 Арифметические операции.....	22
1.6 Сдвиги и логические операции.....	23
1.7 Управление вычислительным процессом.....	23
1.8 Подпрограммы.....	27
1.9 Выполнение машинных команд.....	27
Часть 2. Организация ввода-вывода в базовой ЭВМ.....	31
2.1 Устройства ввода-вывода базовой ЭВМ.....	31
2.2 Команды ввода-вывода.....	32
2.3 Программно-управляемый асинхронный обмен.....	32
2.4 Управляемый по прерыванию программы ввод-вывод.....	34
Часть 3. Микропрограммное устройство управления.....	37
3.1. Микропрограммное управление вентильными схемами.....	37
3.2 Интерпретатор базовой ЭВМ.....	40
Приложение Г. Инструкция по работе с моделью БЭВМ.....	45
Приложение Д. Ассемблер БЭВМ. Краткий справочник.....	46

Настоящее методическое пособие предназначено для практического закрепления материала по дисциплине "Основы вычислительной техники" (ОВТ), преподавание которой организовано по модульному принципу и включает лекции, лабораторные работы, домашние задания и контрольные работы. Курс по основам вычислительной техники и оригинальная модель базовой ЭВМ были разработаны в начале 1980-х годов Кирилловым Владимиром Васильевичем и Приблудой Анатолием Андреевичем. В учебно-методической работе, разработке методических указаний, учебников и моделей базовой ЭВМ, лабораторных работ в разное время принимало участие большое количество сотрудников кафедры вычислительной техники, среди которых особенно хотелось отметить (перечисление в алфавитном порядке) Афанасьева Дмитрия Борисовича, Блохину Елену Николаевну, Гаврилова Антона Валерьевича, Громова Геннадия Юрьевича, Громову Ирину Владимировну, Дергачева Андрея Михайловича, Клименкова Сергея Викторовича, Лемешева Алексея Сергеевича, Максимова Андрея Николаевича, Майорова Сергея Александровича, Мартыанова Николая Васильевича, Перминова Илью Валентиновича, Приблуду Андрея Анатольевича, Приблуду Константина Анатольевича, Щелокова Ивана Викторовича, а также большое количество студентов, аспирантов и выпускников кафедры ВТ.

В 2019 году Афанасьев Д.Б. и Клименков С.В. выступили инициаторами и главными исполнителями проектирования и реализации модели БЭВМ-NG в которую были добавлены необходимые функциональные возможности для обучения студентов, к которым относились стек, режимы адресации, расширенный набор команд, организация ввода-вывода и прерываний. В разработке системы команд и эмулятора также активное участие приняли Медведева Елизавета, Доморацкий Эридан, Щербаков Виктор, Гаврилов Антон Валерьевич и Перминов Илья Валентинович.

В методическом пособии содержится информация, необходимая для успешной сдачи всех лабораторных работ, включая специально разработанную в образовательных целях учебную ЭВМ (базовая ЭВМ), обладающую типичными чертами многих современных ЭВМ. Лабораторная работа №1 предназначена для демонстрации студентам первого курса предмета "Системное программное обеспечение", а также знакомства с лабораториями, где им предстоит обучаться в дальнейшем. Остальные лабораторные работы курса посвящены базовой ЭВМ. С ее помощью студенты исследуют порядок функционирования ЭВМ при выполнении программ различных типов, подходы к организации ввода-вывода информации, принципы микропрограммного управления. В приложениях приведена справочная информация, необходимая при подготовке и защите лабораторных работ.

Раздел 1. Знакомство с кафедрой вычислительной техники

Кафедра вычислительной техники

Кафедра вычислительной техники (ВТ) ведет подготовку бакалавров по направлениям подготовки 09.03.01 - Информатика и Вычислительная техника (профиль подготовки "Вычислительные машины, комплексы, системы и сети") и 09.03.04 - Программная инженерия (профиль подготовки "Разработка программно-информационных систем"). В рамках курса ОВТ ведущие преподаватели направлений подготовки прочитают вводные лекции по своим дисциплинам.

Лабораторные работы курса ОПД

В рамках курса предусмотрено 7 лабораторных работ. Результатом выполнения работы является выполнение всех требований к работе и отчет, который должен включать ряд обязательных составляющих. К ним относятся:

- титульный лист: название университета, кафедры, дисциплины, название и номер лабораторной работы, номер группы и варианта, Ф.И.О. студента, год;
- задание к работе, включая вариант задания;

- порядок выполнения лабораторной работы, дополнительные требования и указания, которые находятся в описании к каждой работе;
- выводы, которые отвечают на вопросы "Что было изучено при выполнении лабораторной работы? Что нового вы узнали? Как можно использовать изученный материал?";
- скрепу, скобу или люверс. Листы должны быть скреплены между собой!

Лабораторная работа №1. Основные команды ОС семейства UNIX

В лабораторных аудиториях кафедры установлено разнообразное вычислительное оборудование под управлением различных операционных систем (ОС). Важное место занимают ОС семейства UNIX, включая различные версии Linux, BSD, Solaris и AIX. Основным способом взаимодействия пользователей и администраторов с такими операционными системами является командный интерфейс с использованием интерпретатора shell.

Цель работы. Знакомство с основным способом взаимодействия с ОС UNIX, командным интерфейсом, а также базовой функциональностью интерпретатора shell. Получение основных сведений о файловой системе и правах доступа к файлам.

Задание.

1. Создать приведенное в варианте дерево каталогов и файлов с содержимым. В качестве корня дерева использовать каталог lab0 своего домашнего каталога. Для создания и навигации по дереву использовать команды: `mkdir`, `echo`, `cat`, `touch`, `ls`, `pwd`, `cd`, `more`, `cp`, `rm`, `rmdir`, `mv`.

2. Установить согласно заданию права на файлы и каталоги при помощи команды `chmod`, используя различные способы указания прав.

3. Скопировать часть дерева и создать ссылки внутри дерева согласно заданию при помощи команд `cp` и `ln`, а также команды `cat` и перенаправления ввода-вывода.

4. Используя команды `cat`, `wc`, `ls`, `head`, `tail`, `echo`, `sort`, `grep` выполнить в соответствии с вариантом задания поиск и фильтрацию файлов, каталогов и содержащихся в них данных.

5. Выполнить удаление файлов и каталогов при помощи команд `rm` и `rmdir` согласно варианту задания.

Подготовка к выполнению работы. Изучить справочные страницы по указанным командам. Разобраться с основными принципами организации ввода-вывода с использованием стандартных потоков ввода-вывода (`stdin`, `stdout`, `stderr`, включая перенаправление данных потоков на команды-фильтры и в файлы), типами файлов, правами пользователей на доступ к файлу для операций чтения, записи и исполнения для владельца файла, группы владельца и остальных пользователей системы.

Порядок выполнения работы. Создать указанное в п. 1 задания дерево файлов и каталогов. Обратит внимание на точное соответствие всех атрибутов полученного дерева заданию. Последовательность команд, необходимых для создания дерева, записать в файл по одной команде на строке (командный скрипт) для возможности автоматического повтора команд этого и последующих пунктов. Изменить права на файлы, согласно п.2 задания. Выполнить п.3, включив в скрипт выполняемые команды. В случае недостатка прав для выполнения операции п.3, необходимо изменить командой `chmod` права файлов перед выполнением команды, а после выполнения вернуть их обратно. Выполнить команды вывода содержимого дерева (п.4), сохранить и включить в отчет вывод исполненных команд. Показать полученное дерево преподавателю. Выполнить команды удаления п.5. Включить в отчет последовательность команд удаления с результатом их выполнения.

Содержание отчета по работе. В дополнение к общим обязательным требованиям, отчет должен содержать:

- Иерархию файлов и каталогов, полученную при помощи команд `ls -lR` из директории `lab0`, после выполнения п.3 задания.
- Задания, команды и результаты их выполнения (если есть), включая сообщения о возникающих ошибках и, для п.3 и п.5, команды, исправляющую ошибки.
- Файл с последовательностью команд по всей лабораторной работе.

Контрольные вопросы:

1. Расскажите про команду *{имя команды}*. Какие она принимает аргументы (их количество, формат, способ задания)? Является ли данная команда фильтром?
2. Стандартные потоки ввода-вывода, назначение. Способы управления стандартными потоками в shell.
3. Стандартные права доступа к файлам и каталогам. Способы задания прав при помощи команды `chmod`. Интерпретация вывода команды `ls -l`.
4. Файлы в ОС UNIX. Типы файлов. Символические и жесткие ссылки.

Рубежный контроль №1

В рубежном контроле раздела три задания.

Задание 1. Привести последовательность команд, создающих приведенное дерево каталогов и файлов с содержимым, начиная с создания директории `lab0`.

```
/home/s113369/lab0 (каталог)
+--darumakal (каталог)
|  +--wurmple (каталог)
|  +--chinchou (каталог)
+--houndourl (файл)
```

Содержимое файлов

`houndourl:`

```
Способности  Howl Smog Roar Bite Odor Sleuth
Beat Up Fire Flamethrower Crunch
Nasty Plot Inferno
```

Различными способами установить права на файлы и директории:

```
darumakal: -wxrwx-wx
wurmple:  rwx-wx-wx
chinchou: права 307
houndourl: rw----r--
```

Решение. Создадим последовательно необходимые файлы и директории в домашнем каталоге и назначим им права. Один из множества возможных вариантов.

```
cd /home/s113369; umask 022
mkdir lab0; cd lab0
mkdir -p darumakal/wurmple darumakal/chinchou
echo "Способности  Howl Smog Roar Bite Odor Sleuth" > houndourl
echo "Beat Up Fire Flamethrower Crunch" >> houndourl
echo "Nasty Plot Inferno" >> houndourl
chmod go=wx darumakal/wurmple
chmod 307 darumakal/chinchou
chmod g-r houndourl
chmod 373 darumakal #должно быть последним, чтобы корректно
#установились вложенные права
```

Задание 2. Перевести заданные числа ($X=31681_{10}$; $Y=13825_{10}$) в 16-ричную систему счисления и представить разрядной сетке памяти БЭВМ. Вычислить значение $R=X+Y$, результат представить как число со знаком в 16-ричной системе счисления разрядной сетке БЭВМ. Укажите, произошло ли переполнение при операции? Был ли перенос?

Решение. Перевод чисел в шестнадцатеричную систему счисления и упаковка исходных чисел в разрядную сетку БЭВМ (см. Приложение В, раздел 1.4)

следующая: $X=7BC1_{16}$, $Y=3601_{16}$. Результат сложения $R=7BC1_{16}+3601_{16}=B1C2_{16}$. В ответе необходимо привести последовательность перевода, упаковки в дополнительный код (при необходимости) и сложения в столбик. Разрешается использовать промежуточное представление в виде двоичного числа.

Перенос в 17 разряд (формирование признака C) не произошел потому, что результат меньше 65535. Переполнение (признак результата V) произошло потому, что результат больше максимально допустимого положительного числа в знаковом представлении (32767). В знаковом представлении результат (-20030_{10}) является отрицательным числом, что является ошибкой, т. к. ведет к потере ожидаемого результата сложения (45506).

Задание 3. Какую операцию выполняет команда ADD AD? Перечислите все результаты этой команды.

Решение. Команда ADD AD относится к адресным командам с прямой абсолютной адресацией и складывает содержимое указанной в коде команды ячейки с аккумулятором (AC).

Результатом команды являются измененные значения регистров и ячеек памяти, а именно: счетчик команд будет содержать адрес следующей выполняемой команды; регистр AC — сумму содержимого AC до исполнения команды и ячейки ADA; AR и DR — содержать число AD; CR содержать код команды 40AD; в BR будет содержимое IP до выполнения команды; будут установлены признаки результата NZVC в PS по вычисленной сумме.

Раздел 2. Введение в базовую ЭВМ

В рамках этого раздела студент должен изучить состав, структуру и принцип функционирования БЭВМ на уровне машинных команд, систему команд БЭВМ, детальную последовательность исполнения команд с прямой и косвенной адресациями, подпрограммы, основные подходы, применяемые для низкоуровневой обработки данных.

Лабораторная работа №2. Исследование работы БЭВМ

Цель работы - изучение приемов работы на базовой ЭВМ и исследование порядка выполнения арифметических команд и команд пересылки.

Задание. По выданному преподавателем варианту определить функцию, вычисляемую программой, область представления и область допустимых значений исходных данных и результата, выполнить трассировку программы, предложить вариант с меньшим числом команд. При выполнении работы представлять результат и все операнды арифметических операций знаковыми числами, а логических операций набором из шестнадцати логических значений.

Подготовка к выполнению работы. Познакомиться с устройством и системой команд базовой ЭВМ (см. приложение В, п.п. 1.1 — 1.6), порядком выполнения машинных команд (п.1.9), инструкцией по работе с моделью базовой ЭВМ (см. приложение Г). Изучить представления в БЭВМ числовых и логических значений.

Порядок выполнения работы. Восстановить текст заданного варианта программы, отделить ячейки данных от кода программы, написать назначение программы и реализуемую функцию, которую представить в виде формулы.

Во время допуска к работе получить у преподавателя исходные данные для переменных, согласовать вариант программы для исполнения, занести в память базовой ЭВМ заданный вариант программы и, выполняя ее по командам, заполнить таблицу трассировки выполненной программы. Занесение программы с данными, а также запуск программы в пультовом режиме продемонстрировать преподавателю.

Содержание отчета по работе. В дополнение к общим обязательным требованиям, отчет должен содержать:

1. Текст исходной программы по следующей форме:

Адрес	Код команды	Мнемоника	Комментарии
021	4015	ADD 15	Добавить содержимое ячейки памяти 15 к аккумулятору

2. Описание программы:

- назначение программы и реализуемые ею функция (формула);
- область представления и область допустимых значений исходных данных и результата;
- расположение в памяти ЭВМ программы, исходных данных и результатов;
- адреса первой и последней выполняемой команд программы.

3. Таблица трассировки должна быть представлена в соответствии с форматом:

Форма таблицы трассировки выполнения команд.

Таблица 2.2

Выполняемая команда		Содержимое регистров процессора после выполнения команды.								Ячейка, содержимое которой изменилось после выполнения команды	
Адрес	Код	IP	CR	AR	DR	SP	BR	AC	NZVC	Адрес	Новый код
xxx	xxxx	xxx	xxxx	xxxx	xxxx	xxx	xxxx	xxxx	xxxx	xxx	xxxx

4. Вариант программы с меньшим числом команд.

Контрольные вопросы:

1. Форматы представления, области представления и области допустимых значений в БЭВМ для знаковых и беззнаковых чисел с фиксированной точкой и логических значений.
2. Представление чисел в разрядной сетке в прямом, обратном и дополнительном кодах.
3. Адресные и безадресные команды БЭВМ.
4. Описание команды находящейся, по указанному адресу: наименование, назначение, тип команды и вид адресации. Количество и название машинных циклов, потактовое выполнение команды.
5. Какую формулу реализует программа? Как можно упростить программу?
6. Где находятся аргументы программы? Где находится результат? Как они представлены? Какие дополнительные ячейки использует программа? Для чего?
7. Какое количество обращений к ячейкам памяти при выполнении безадресной команды? На каких циклах оно выполняется?

Рубежный контроль №2

Задание 1. Заполнить таблицу трассировки правильными значениями и восстановить формулу, вычисляемую программой. Пример варианта задания и правильного решения приведен в табл. 2.3.

Пример задания №2 и правильного ответа

Таблица 2.3

Исходные данные		Пример правильного ответа. Формула: $C = +2 * A - B + 1$								
Адрес	Команда/данные	Адрес	Команда/данные	IP	CR	AR	DR	BR	AC	NZVC
00A	3BBD	00A	3BBD	-	-	-	-	-	-	-
00B	CF34	00B	CF34	-	-	-	-	-	-	-
00C	0000	00C	0000	-	-	-	-	-	-	-
00D	+LD 00A	00D	+LD 00A	00E	A00A	00A	3BBD	000D	3BBD	----
00E	ASL	00E	ASL	00F	0500	00E	3BBD	000E	777A	----
00F	SUB 00B	00F	SUB 00B	010	600B	00B	CF34	000F	A846	N-V-
010	INC	010	INC	011	0700	010	0700	0010	A847	N---
011	ST 00C	011	ST 00C	012	E00C	00C	A847	0011	A847	N---
012	HLT	012	HLT	013	0100	012	0100	0012	A847	N---

Лабораторная работа №3. Выполнение циклических программ

Цель работы - изучение способов организации циклических программ и исследование порядка функционирования БЭВМ при выполнении циклических программ и обработки одномерных массивов.

Задание. По выданному преподавателем варианту восстановить текст заданного варианта программы, определить предназначение и составить описание программы, определить область представления и область допустимых значений исходных данных и результата, выполнить трассировку программы.

Подготовка к выполнению работы.

Получить у преподавателя номер варианта и исходные данные к лабораторной работе. Изучить способы и средства организации циклических программ с использованием системы команд базовой ЭВМ (**приложение В, п.1.7, примеры 1 и 2**). Восстановить текст заданного варианта программы. Составить описание программы.

Порядок выполнения работы. Получить допуск к лабораторной работе, предъявив преподавателю подготовленные материалы. Получить у преподавателя данные для элементов массива. Элементы массива из изначального задания используются только для определения функциональности программы! Занести в память базовой ЭВМ заданный вариант программы и заполнить таблицу трассировки, выполняя эту программу по командам. Таблицу трассировки подписать у преподавателя.

Содержание отчета по работе. Отчет по работе должен быть составлен аналогично лабораторной работе №2, за исключением п. 4 (разработка программы с сокращенным числом команд). Необходимо привести диапазон всех ячеек памяти, где может размещаться массив исходных данных.

Контрольные вопросы:

1. Организация одномерных массивов данных в памяти. Организация и обработка массивов с числом измерений, больше чем одно.
2. Сравнение значений в БЭВМ. Команды условных и безусловного переходов.
3. Организация циклических вычислений. Команда LOOP.
4. Режимы адресации БЭВМ.
5. Описание адресных команд и команд переходов с различными режимами адресации: наименование, назначение, тип команды и вид адресации. Количество и название машинных циклов, потактовое выполнение команд.
6. Количество обращений к памяти команд БЭВМ с различными режимами адресации.
7. Где находятся аргументы программы? Где находится результат? Как они представлены?
8. Какое максимальное количество элементов данных может поддерживать ваша программа?

Лабораторная работа №4. Выполнение комплекса программ

Цель работы - изучение способов связи между программными модулями, команды обращения к подпрограмме и исследование порядка функционирования БЭВМ при выполнении комплекса взаимосвязанных программ.

Задание. По выданному преподавателем варианту восстановить текст заданного варианта программы и подпрограммы (программного комплекса), определить их предназначение и составить описание, определить область представления и область допустимых значений для исходных данных и возвращаемых значений подпрограммы, выполнить трассировку программного комплекса.

Подготовка к выполнению работы.

Получить у преподавателя номер варианта и исходные данные к лабораторной работе. Изучить способы связи между программными модулями и команды обращения к подпрограмме в базовой ЭВМ (приложение В, п. 1.8). Восстановить текст заданного варианта программного комплекса, составить его описание, нарисовать график функции, которая вычисляется в подпрограмме.

Порядок выполнения работы. Получить допуск к лабораторной работе, предъявив преподавателю подготовленные материалы. Занести в память базовой ЭВМ заданный вариант программного комплекса и заполнить таблицу трассировки, выполняя этот комплекс по командам. Подписать таблицу трассировки!

Содержание отчета по работе. Отчет по работе должен быть составлен аналогично лабораторной работе №2, за исключением п. 4 (разработка программы с сокращенным числом команд). Необходимо привести график функции, реализуемый подпрограммой.

Контрольные вопросы:

1. Организация подпрограмм в БЭВМ. Команды вызова подпрограммы и возврата.
2. Аргументы и возвращаемые значения подпрограммы. Способы организации передачи аргументов и возвращаемых значений.
3. Рекурсивный вызов подпрограмм. Организация стека.
4. Описание команд CALL и RET: наименование, назначение, тип команды и вид адресации. Количество и название машинных циклов, потактовое выполнение команды, количество обращений к памяти.

Раздел 3. Организация ввода-вывода информации в БЭВМ

Основным назначением вычислительных устройства является обработка внешней по отношению к ЭВМ информации. Для того, что бы получать ее извне, обрабатывать и передавать результаты обработки используются внешние (по отношению к ЭВМ) устройства, такие как клавиатура, мышь, дисплей, принтер и т.д. Вывод-вывод информации на эти устройства должен быть специально организован. Данный раздел предназначен для изучения организации ввода-вывода БЭВМ.

Лабораторная работа №5. Асинхронный обмен данными с ВУ

Цель работы - изучение организации системы ввода-вывода базовой ЭВМ, команд ввода-вывода и исследование процесса функционирования ЭВМ при обмене данными по сигналам готовности внешних устройств (ВУ).

Задание. По выданному преподавателем варианту разработать программу асинхронного обмена данными с внешним устройством. При помощи программы осуществить ввод или вывод информации, используя в качестве подтверждения данных сигнал (кнопку) готовности ВУ.

Подготовка к выполнению работы.

Изучить организацию системы ввода-вывода и команды ввода-вывода базовой ЭВМ, организацию асинхронного программно-управляемого обмена данными (Приложение В, п.п.2.1-2.3, пример 3). Разработать заданную программу и составить ее описание. Команды программы, используемые переменные и коды символов необходимо разместить в указанных ячейках. Закодировать строку в заданной кодировке, а также в кодировках UTF-8 и UTF-16.

Порядок выполнения работы

1. Получить допуск к лабораторной работе, предъявив преподавателю подготовленные материалы.
2. Разработать и занести в БЭВМ программу, при необходимости ввести данные.
3. Предъявить преподавателю заданную работающую программу, выполняющую

- в автоматическом режиме ввод-вывод символов.
4. В режиме покомандного выполнения программы ввести (вывести) два первых символов заданного слова, заполняя таблицу трассировки.
 5. Перевести ЭВМ в режим автоматического выполнения программы и ввести (вывести) остальные символы заданного слова. Таблицу трассировки подписать у преподавателя!
 6. Опционально, по дополнительному заданию преподавателя, переделать устройство ввода-вывода (ВУ-1...ВУ-3) на ВУ-4...ВУ-10.

Содержание отчета по работе. Отчет по работе должен быть составлен аналогично лабораторной работе №2, за исключением п. 4 (разработка программы с сокращенным числом команд). Кроме того, отчет должен содержать заданное слово и коды его символов и текст исходной программы на языке Ассемблера БЭВМ. (синтаксис и особенности приведены в Приложении Д).

Контрольные вопросы:

1. Синхронный и асинхронный режимы передачи данных.
2. Программно-управляемый и управляемый прерываниями ввод-вывод, прямой доступ к памяти. Преимущества и недостатки.
3. Способы и формат представления символьных и строковых данных в БЭВМ. Кодировки ASCII, КОИ-8, Windows-1251, ISO-8859-5, UTF-8, UTF-16.
4. Порядок байтов в памяти от младшего к старшему (little-endian) и от старшего к младшему (big-endian).
5. Система команд ввода-вывода БЭВМ. Команды IN, OUT, - название, назначение и тип команды. Количество и название машинных циклов, потактовое выполнение команды, с перечислением всех шин, участвующих в обмене.
6. Какие режимы передачи данных и управления вводом-выводом реализуемы в БЭВМ? Почему не возможно реализовать другие?
7. Может ли ВУ определить в каком режиме с ним работают?
8. Назначение флага готовности ВУ, регистра данных ВУ (*DR DEV*), регистра состояния ВУ (*CR DEV*)?
9. Какие элементы БЭВМ участвуют в обмене с ВУ? Укажите направление передачи данных между элементами при операциях ввода и вывода.

Лабораторная работа №6. Обмен данными с ВУ по прерыванию

Цель работы - изучение организации процесса прерывания программы и исследования порядка функционирования ЭВМ при обмене данными в режиме прерывания программы.

Задание. По выданному преподавателем варианту разработать и исследовать работу комплекса программ обмена данными в режиме прерывания программы. Основная программа должна изменять содержимое заданной ячейки памяти (X), которое должно быть представлено как знаковое число. Область допустимых значений изменения X должна быть ограничена заданной функцией F(X) и конструктивными особенностями регистра данных ВУ (8-ми битное знаковое представление). Программа обработки прерываний должна *модифицировать* ячейку памяти для хранения X в соответствии с вариантом задания и выводить его на ВУ, а также игнорировать все необрабатываемые прерывания.

Подготовка к выполнению работы

Изучить организацию в базовой ЭВМ программно-управляемого обмена данными в режиме прерывания программы (Приложение В, п.2.4, пример 4). Разработать комплекс программ, указанный в задании. Составить методику проверки правильности выполнения разработанного комплекса программ на БЭВМ, т.е. написать последовательность действий оператора (пользователя) БЭВМ, которые

необходимо выполнить, чтобы проверить все возможные режимы работы комплекса программ (при появлении запроса прерывания от любого ВУ) и получить заданное количество результатов. Пример методики см. в разделе содержание отчета.

Порядок выполнения работы.

1. Получить допуск к лабораторной работе, предъявив преподавателю подготовленные материалы.
2. Занести разработанный комплекс программ в память БЭВМ.
3. В присутствии преподавателя провести проверку работоспособности комплекса программ в автоматическом режиме.
4. В присутствии преподавателя, используя методику проверки разработанного комплекса программ, получить 3 пары результатов, указывая для каждого выведенного значения величину X.
5. Результаты работы программного комплекса представить в виде таблицы результатов работы комплекса.

Содержание отчета по работе. В дополнение к общим обязательным требованиям, отчет должен содержать:

1. Описание программы, аналогично п.2 требований к отчету для ЛРН№2.
2. Текст исходной программы на языке Ассемблера БЭВМ (**синтаксис и особенности приведены в Приложении Д**).
3. Полностью разработанную и проверенную на БЭВМ методику проверки.

Пример. Начальный фрагмент методики проверки.

1. Загрузить комплекс программ в память базовой ЭВМ.
2. Изменить значения точки останова по адресу на HLT
3. Запустить основную программу в автоматическом режиме с адреса
4. Установить "Готовность ВУ-3".
5. Дождаться останова
6. Записать содержимое аккумулятора в момент останова программы
7. Продолжить выполнение программы
8. ...

Контрольные вопросы:

1. Особенности организации программ обмена данными с использованием прерываний. Сохранение и восстановление значений регистров.
2. Команды работы разрешения/запрещения прерываний БЭВМ, команда программного прерывания, команда возврата из прерывания. Название, назначение и тип команды. Количество и название машинных циклов, потактовое выполнение команды.
3. Вектора прерываний. Преимущества использования векторов прерываний.
4. Регистр управления ВУ (MR DEV).
5. Сигналы шины БЭВМ, назначение, временные диаграммы сигналов Input и Output.
6. Когда выполняется цикл обработки прерывания? После каких команд он не выполняется? Почему?
7. Обрабатываются ли прерывания в пошаговом режиме (режиме останов) работы программы? Почему?
8. Что происходит при одновременном поступлении сигнала готовности нескольких внешних устройств? В какой последовательности они будут обработаны?
9. За что отвечают биты 5, 6, 7 и 8 регистра состояния? Когда изменяется их значение?

Раздел 4. Организация микропрограммного устройства БЭВМ

Микропрограммное устройство предназначено для исполнения команд БЭВМ. В данном разделе изучается его состав, структура и принцип работы.

Лабораторная работа №7. Синтез команд БЭВМ

Цель работы - практическое освоение принципов микропрограммирования и разработки адресных и безадресных команд.

Задание. Синтезировать цикл исполнения для команды, соответствующей выданному преподавателем варианту задания. Разработать тестовую программу, которая проверяет синтезированную команду. Загрузить в микропрограммную память БЭВМ цикл исполнения синтезированной команды, модифицировать, при необходимости, основную микропрограмму, загрузить в основную память БЭВМ тестовую программу. Проверить и отладить разработанные тестовые программы и микропрограммы.

Подготовка к выполнению работы.

Получить у преподавателя вариант задания. Изучить организацию микропрограммного устройства базовой ЭВМ, (**Приложение В раздел 3**).

1. Синтезировать микрокоманды цикла исполнения одной из следующих команд в соответствии с вариантом задания:

- команда 9xxx — команда, предназначенная для выполнения адресных команд;
- команда Fxxx — команда, осуществляющая переход по заданному условию;
- безадресная команда с кодом 0FXX.

2. Написать тестовую программу для проверки правильности исполнения синтезированных команд базовой ЭВМ. Данная программа должна отвечать следующим требованиям:

- Тестовая программа должна состоять из отдельных тестовых блоков (частей программы или подпрограмм), которые проверяют различные результаты выполнения команды. Количество таких тестовых блоков необходимо согласовать с преподавателем.
- Каждый тестовый блок должен в случае корректной работы микропрограммы записывать 1 в выбранную ячейку памяти. Если микропрограмма работает некорректно, тест должен обнулять выбранную ячейку.
- Тестовая программа должна проверить что все тестовые блоки завершились корректно и записать 1 в выбранную ячейку памяти.
- Для синтезированных адресных и безадресных команд результат их выполнения должен быть зафиксирован в выбранной ячейке памяти БЭВМ.
- Если проверяемая арифметическая или безадресная команда устанавливает признаки результата (биты N,Z,V,C), необходимо проверить правильную установку только одного из них, используя соответствующую команду перехода.
- Для синтезированных команд переходов необходимо проверить команду как при выполнении условия перехода, так и при его невыполнении.

Таким образом, после выполнения правильно разработанной тестовой программы в автоматическом режиме в памяти базовой ЭВМ будет размещена информация, позволяющая однозначно подтвердить правильность выполнения синтезированной команды.

3. При разработке микропрограмм заданных команд следует иметь в виду:

- В разрабатываемых командах может понадобится цикл выборки операнда, а может оказаться нужен только цикл выборки адреса. Команда 9XXX выполняет оба цикла. Разумным решением будет изменение основной микропрограммы.
- После перехода на участок реализации команды в микропрограмме, возможно будет необходимым продолжить декодирование команды, чтобы убедиться, что будет исполнена только одна микрокоманда с помощью анализа CR.
- Все микропрограммы реализуемых команд должны заканчиваться микрокомандой 80C4101040 (GOTO INT @ C4), осуществляющей переход к циклу прерывания БЭВМ.

- Для реализации микропрограммы необходимо использовать режим командной строки БЭВМ `java -Dmode=cli -jar bcomp-ng.jar`. Подсказка по командам доступна по инструкции `help`.
- Для вывода листинга текущей микропрограммы необходимо использовать команду `mdecodeall` — вывод всех ненулевых ячеек микрокода.

Пример. 0F00 - инверсия содержимого аккумулятора и очистка регистра С. Дополнительного декодирования не требует.

Изменения памяти микрокоманд для команды "СОМАСЛС".

Таблица 4.1

Адрес МП	Микро-команда	Действие ; Комментарии
ВВ	81F0014002	Исправляемые ячейки интерпретатора if CR(8) = 1 then GOTO F0 ; изменение адреса перехода
F0	0010E09210	Микрокод команды ~AC → AC, N, Z, V, C; Инверсия AC и установка признаков результата
F1	80C4101040	GOTO INT @ C4 ; Переход на цикл прерывания

Таблица трассировки микрокоманд.

Таблица 4.2

MP до выборки МК	Содержимое памяти и регистров процессора после выборки и исполнения микрокоманды								
	MR	IP	CR	AR	DR	BR	AC	NZVC	MP (СчМК)
01	0000000000	00E	0000	000	0000	0000	0000	00000	02

Порядок выполнения работы

1. Получить допуск к лабораторной работе, предъявив преподавателю подготовленные материалы.

2. Занести разработанные изменения основной микропрограммы и микропрограммы циклов исполнения заданной команды в микропрограммную память базовой ЭВМ, а разработанную тестовую программу в основную память базовой ЭВМ.

3. Выполнить в пошаговом режиме тестовые программы, проверив работоспособность синтезированных команд. Заполнить таблицу трассировки цикла исполнения для разработанных микрокоманд по форме таблицы 4.2 для одного варианта выполнения каждой микрокоманды.

Содержание отчета по работе. В дополнение к общим обязательным требованиям, отчет должен содержать:

1. Текст синтезированных микропрограмм по форме таблицы 4.1
2. Текст тестовых программ на языке Ассемблера БЭВМ (см. Приложение Д).
3. Таблицу трассировки циклов исполнения разработанных микрокоманд по форме таблицы 4.2
4. Методику проверки команды с использованием тестовой программы.

Контрольные вопросы:

1. Микропрограммное устройство ЭВМ, назначение, состав, принцип работы.
2. Формат микрокоманд БЭВМ. Для чего существуют два формата команд?
3. Исполнение горизонтальных (ОМК, УМК) микрокоманд на примере заданной микрокоманды.
4. Структура и принципы функционирования АЛУ.
5. Структура и принципы функционирования коммутатора.
6. Выполнение операций суммирования и логического умножения, схема сумматора. Инверторы входов, схема инверторов входов.
7. Выполнение операций сдвигов, симметричной и несимметричной передачи.
8. Принципы построения PS, значение отдельных битов.
9. Как организована и выполняется микрокоманда безусловного перехода?
10. В какой момент происходит увеличение СчМК?
11. Что будет если на вентили SORA и PLS1 одновременно подать единицы?