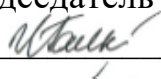


ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ ОБРАЗОВАТЕЛЬНОЕ
УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ
«ФИНАНСОВЫЙ УНИВЕРСИТЕТ ПРИ ПРАВИТЕЛЬСТВЕ
РОССИЙСКОЙ ФЕДЕРАЦИИ»
ПЕРМСКИЙ ФИНАНСОВО-ЭКОНОМИЧЕСКИЙ КОЛЛЕДЖ –
ФИЛИАЛ ФИНУНИВЕРСИТЕТА

РАССМОТРЕНО
на заседании ПЦК
информационных систем и
программирования
Протокол № 6 от «24» января 2024 г.
Председатель ПЦК
 /И.В. Галкин

УТВЕРЖДАЮ
Зам. директора по УМР
Пермского филиала
Финуниверситета
 /Ю.А. Хакимова/
«25» января 2024 г.

ФОНД
оценочных средств учебной дисциплины

«ОП.02 Архитектура аппаратных средств»

программы подготовки специалистов среднего звена по специальности
09.02.07 Информационные системы и программирование

Составитель:

Галкин И.В., преподаватель Пермского филиала ФГОБУ ВО «Финансовый университет при Правительстве Российской Федерации»

Рецензенты (эксперты):

Одобрено Научно-методическим советом филиала

Протокол № 3 от «12» февраля 2024 г.

1. Общие положения

Контрольно-оценочные средства (далее КОС) предназначены для контроля и оценки образовательных достижений обучающихся, освоивших программу учебной дисциплины «ОП.02 Архитектура аппаратных средств».

КОС включают контрольные материалы для проведения текущего контроля и итоговой аттестации в форме экзамена.

КОС разработаны на основании положений:

- ФГОС по специальности 09.02.07 Информационные системы и программирование;
- Положения о ФОС Пермского филиала Финуниверситета;
- программы профессионального модуля «ОП.02 Архитектура аппаратных средств».

2. Результаты освоения дисциплины, подлежащие проверке

Объекты оценивания	Показатели оценки	Критерии оценки
3.1 виды информации и способы ее представления в ЭВМ;	Представление числовой информации используются 3 вида чисел: с фиксированной точкой, с плавающей точкой, двоично-десятичное представление.	Верное представление числовой информации используются 3 вида чисел: с фиксированной точкой, с плавающей точкой, двоично-десятичное представление.
3.2 классификацию и типовые узлы вычислительной техники (ВТ);	выявляет общие принципы построения и классификацию современных вычислительных систем; приводит примеры вычислительных платформ различных типов; выбирает тип вычислительной системы в соответствии с решаемой задачей.	Правильно выявляет общие принципы построения и классификацию современных вычислительных систем; приводит несколько примеры вычислительных платформ различных типов; грамотно выбирает тип вычислительной системы в соответствии с решаемой задачей.
3.3 архитектуру электронно-вычислительных машин и вычислительных систем;	Знание блоков ВТ: устройство ввода, центральный процессор, запоминающие устройство, устройство вывода.	Верно называет и характеризует все блоки ВТ: устройство ввода, центральный процессор, запоминающие устройство, устройство вывода.
3.4 назначение и принципы действия отдельных архитектурных конфигураций.	классифицирует и определяет типовые узлы средств вычислительной техники; представляет архитектуру электронно-вычислительных машин и вычислительных систем; различает назначение и принципы действия отдельных архитектурных конфигураций.	Правильно классифицирует и определяет типовые узлы средств вычислительной техники; представляет в полном объеме архитектуру электронно-вычислительных машин и вычислительных систем; грамотно различает назначение и принципы действия отдельных архитектурных конфигураций.
У.1 выбирать рациональную	Успешное конфигурирование оборудования ВТ и ее работа	Правильно определяет устройства ЭВМ для успешной

конфигурацию оборудования в соответствии с решаемой задачей;		организации работы вычислительных систем.
У.2 обеспечивать совместимость аппаратных и программных средств ВТ.	Подбор аппаратных и программных средств.	Верно различает аппаратные и программные средства вычислительной техники; грамотно контролирует совместимость аппаратных и программных средств вычислительной техники.
ОК 1- ОК 11	Использование в заданиях профильной информации по специальности подготовки.	

3. Распределение оценивания результатов обучения по видам контроля

Наименование элемента умений или знаний	Виды аттестации	
	Текущий контроль	Промежуточная аттестация
3.1 виды информации и способы ее представления в ЭВМ;	+	+
3.2 классификацию и типовые узлы вычислительной техники (ВТ);	+	+
3.3 архитектуру электронно-вычислительных машин и вычислительных систем;	+	+
3.4 назначение и принципы действия отдельных архитектурных конфигураций.	+	+
У.1 выбирать рациональную конфигурацию оборудования в соответствии с решаемой задачей;	+	+
У.2 обеспечивать совместимость аппаратных и программных средств ВТ.	+	+

4. Распределение типов контрольных заданий по элементам знаний и умений

Содержание учебного материала по программе УД	Тип контрольного задания					
	З1	З2	З3	З4	У1	У2
Раздел 1. Вычислительные приборы и устройства						
Тема 1.1. Классы вычислительных машин	Пр					
Раздел 2. Архитектура и принципы работы основных логических блоков системы						
Тема 2.1 Логические основы ЭВМ, элементы и узлы		Пр				
Тема 2.2. Принципы организации ЭВМ	Пр					
Тема 2.3 Классификация и типовая структура микропроцессоров		Пр				
Тема 2.4. Технологии повышения производительности процессоров						
Тема 2.5 Компоненты системного блока		Пр	Пр			
Тема 2.6 Запоминающие устройства ЭВМ						
Раздел 3. Периферийные устройства						
Тема 3.1 Периферийные устройства вычислительной техники				Пр	Пр	Пр
Тема 3.2 Нестандартные периферийные устройства						Пр

5. Распределение типов и количества контрольных заданий по элементам знаний и умений, контролируемых на промежуточной аттестации

Содержание учебного материала по программе УД	Тип контрольного задания					
	З1	З2	З3	З4	У1	У2
Раздел 1. Вычислительные приборы и устройства						
Тема 1.1. Классы вычислительных машин	Т					
Раздел 2. Архитектура и принципы работы основных логических блоков системы						
Тема 2.1 Логические основы ЭВМ, элементы и узлы		Т				
Тема 2.2. Принципы организации ЭВМ	Т					
Тема 2.3 Классификация и типовая структура микропроцессоров		Т				
Тема 2.4. Технологии повышения производительности процессоров						
Тема 2.5 Компоненты системного блока		Т	Т			
Тема 2.6 Запоминающие устройства ЭВМ						
Раздел 3. Периферийные устройства						
Тема 3.1 Периферийные устройства вычислительной техники				Т	Т	
Тема 3.2 Нестандартные периферийные устройства						Т

6. Структура контрольных заданий

Тестовые задания

Задание #1

Вопрос:

Классическая архитектура называется
Выберите один из 4 вариантов ответа:

- 1) архитектурой Джона фон Неймана
- 2) архитектурой Била Гейтса
- 3) архитектурой Блеза Паскаля
- 4) архитектурой Чарльза Беббиджа

Задание #2

Вопрос:

К устройствам ввода-вывода относятся:
Выберите один из 4 вариантов ответа:

- 1) процессор, материнская плата, системный блок
- 2) клавиатура, принтер, сканер, монитор, манипуляторы, акустическая система
- 3) жесткие диски, гибкие диски, оперативная память
- 4) контроллеры, драйвера, порты, модемы

Задание #3

Вопрос:

Определите какое высказывание является верным:

Выберите один из 4 вариантов ответа:

- 1) постоянная память внешнее устройство компьютера
- 2) центральный процессор является внешним устройством компьютера
- 3) оперативная память-внешнее устройство компьютера
- 4) принтер-внешнее устройство компьютера

Задание #4

Вопрос:

Оперативная память имеет следующую структуру:

Выберите один из 3 вариантов ответа:

- 1) разбита на сектора и дорожки, информация записана в виде намагниченных и не намагниченных областей
- 2) разбита на кластеры, информация записана в виде намагниченных и не намагниченных областей
- 3) состоит из ячеек, каждая ячейка имеет адрес и содержание

Задание #5

Вопрос:

это устройство, осуществляющее арифметические, логические операции и руководящее работой ПК с помощью электрических импульсов.

Запишите ответ:

Задание #6

Вопрос:

К основным характеристикам монитора относятся:
Выберите несколько из 6 вариантов ответа:

- 1) цветность
- 2) дизайн
- 3) размер по диагонали
- 4) способ формирования изображения
- 5) тип видеокарты
- 6) разрешающая способность экрана

Задание #7

Вопрос:

это конструкционный элемент компьютера, на котором размещено большое число деталей: процессор, оперативная память, ПЗУ, слоты для подключения дополнительных карт.

Запишите ответ:

Задание #8

Вопрос:

Типы процессоров:

Выберите несколько из 7 вариантов ответа:

- 1) RISC-процессоры
- 2) NISC-процессоры
- 3) CISC-процессоры
- 4) MISC-процессоры
- 5) Многоядерные процессоры
- 6) JISC-процессоры
- 7) DISC-процессоры

Задание #9

Вопрос:

После отключения компьютера все информация стирается...

Выберите один из 4 вариантов ответа:

- 1) с CD - ROM
- 2) из оперативной памяти
- 3) с жесткого диска
- 4) с гибкого диска

Задание #10

Вопрос:

Манипулятор мышь - это устройство _____ информации.

Запишите ответ:

Задание #11

Вопрос:

Какой из вентиляторов будет создавать больший воздушный поток, если они работают на одинаковом количестве оборотов?

Выберите один из 3 вариантов ответа:

- 1) 80 мм
- 2) 120 мм
- 3) 60 мм

Задание #12

Вопрос:

Программы сопряжения устройств компьютера называются:

Выберите один из 5 вариантов ответа:

- 1) трансляторами
- 2) драйверами
- 3) компиляторами
- 4) интерпретаторами
- 5) загрузчиками

Задание #13

Вопрос:

Процедура разметки нового диска называется

Запишите ответ:

Задание #14

Вопрос:

Расставьте по порядку этапы выполнения цикла команд процессором

Укажите порядок следования всех 5 вариантов ответа:

___ выставленное число является для памяти адресом; память, получив адрес и команду чтения, выставляет содержимое, хранящееся по этому адресу, на шину данных, и сообщает о готовности
___ если последняя команда не является командой перехода, процессор увеличивает на единицу (в предположении, что длина каждой команды равна единице) число, хранящееся в счётчике команд; в результате там образуется адрес следующей команды

___ процессор выставляет число, хранящееся в регистре счётчика команд, на шину адреса, и отдаёт памяти команду чтения

___ процессор получает число с шины данных, интерпретирует его как команду (машинную инструкцию) из своей системы команд и исполняет её

___ снова выполняется с первого пункта

Задание #15

Вопрос:

Арифметически-логическое устройство - это ...

Выберите один из 4 вариантов ответа:

- 1) регистр
- 2) устройство увеличения оперативной памяти
- 3) блок, выполняющий команды программы
- 4) ячейка

Задание #16

Вопрос:

Программное управление работой компьютера предполагает:

Выберите один из 4 вариантов ответа:

- 1) использование специальных формул для реализации команд в компьютере
- 2) необходимость использования операционной системы для синхронной работы аппаратных средств
- 3) выполнение компьютером серии команд без участия пользователя
- 4) двоичное кодирование данных в компьютере

Задание #17

Вопрос:

Сопоставьте:

Укажите соответствие для всех 4 вариантов ответа:

- 1) RAID 0
- 2) RAID 1
- 3) RAID 10
- 4) RAID 5

___ все данные разбиваются на блоки и для каждого блока формируется блок 'четности', по которому можно восстановить утерянные данные. Блоки с данными и блоки 'четности' записываются вперемешку на все диски.

___ в этом режиме из нескольких дисков формируется один массив. При доступе к этому массиву обращение к дискам

происходит параллельно, благодаря чему скорость работы повышается. Но если на любом из жестких дисков происходит сбой, то данные теряются.

___ на двух жестких дисках хранятся идентичные данные. При неисправности одного жесткого диска все данные остаются доступными на другом диске без ущерба для целостности данных.

___ представляет собой комбинацию RAID 0 для повышения производительности и RAID 1 для защиты данных. Для такого массива необходимо четыре диска.

Задание #18

Вопрос:

К внутренней памяти не относится:

Выберите один из 4 вариантов ответа:

- 1) Кэш-память
- 2) ПЗУ
- 3) Жесткий диск
- 4) ОЗУ

Задание #19

Вопрос:

Для того, чтобы информация хранилась долгое время ее, надо записать

Выберите один из 4 вариантов ответа:

- 1) в ПЗУ
- 2) на жесткий диск
- 3) в оперативную память
- 4) в регистры процессора

Задание #20

Вопрос:

Адресуемость оперативной памяти означает:

Выберите один из 4 вариантов ответа:

- 1) дискретность структурных единиц памяти
- 2) возможность произвольного доступа к каждой единице памяти
- 3) энергозависимость оперативной памяти
- 4) наличие номера у каждой ячейки оперативной памяти

Задание #21

Вопрос:

Информация, записанная на магнитный диск, называется:

Выберите один из 3 вариантов ответа:

- 1) файл
- 2) регистр

3) ячейка

Задание #22

Вопрос:

1 короткий сигнал BIOS AMI

Выберите один из 5 вариантов ответа:

- 1) Ошибок не найдено. Нормальная загрузка ПК.
- 2) Серьезная ошибка оперативной памяти (первых 64 Кбайт). Перезагрузитесь через Reset, проверьте установку модулей памяти. При частом появлении ошибки меняйте память.
- 3) Неисправен системный таймер. Перезагрузитесь через Reset, при повторном появлении придется заменить материнскую плату.
- 4) Ошибка четности оперативной памяти. Перезагрузитесь через Reset, проверьте установку модулей памяти. При частом появлении ошибки меняйте память.
- 5) Неисправен центральный процессор. Перезагрузитесь через Reset, не помогло замените процессор.

Задание #23

Вопрос:

К устройствам внешней памяти относятся...?

Выберите несколько из 4 вариантов ответа:

- 1) накопители на жестком и гибком магнитных дисках (HDD и FDD).
- 2) стример.
- 3) плоттер.
- 4) CD- ROM.

Задание #24

Вопрос:

Позволяют объединить две видеокарты, установленные на одной материнской плате.

Выберите несколько из 5 вариантов ответа:

- 1) Visual Interface
- 2) SLI
- 3) CrossFire
- 4) High Definition Multimedia Interface
- 5) Video Input Video Output

Задание #25

Вопрос:

Дисковод - это устройство для:

Выберите один из 4 вариантов ответа:

- 1) хранения информации

- 2) чтения/записи данных с внешнего носителя
- 3) вывода информации на бумагу
- 4) обработки команд исполняемой программы

Задание #26

Вопрос:

К основным характеристикам принтера относятся:
Выберите несколько из 5 вариантов ответа:

- 1) скорость печати
- 2) настройка печатаемого шрифта
- 3) цветность
- 4) качество печати
- 5) число печатаемых копий документа

Задание #27

Вопрос:

От каких факторов зависит уровень шума в системе охлаждения?

Выберите несколько из 6 вариантов ответа:

- 1) Конструкция крыльчатки
- 2) Скорость вращения
- 3) Тип разъема питания
- 4) Диаметр вентиляторов
- 5) Тип подшипников
- 6) Материал радиатора

Задание #28

Вопрос:

Для подключения к какому интерфейсу предназначен данный кабель

Изображение:



Выберите один из 5 вариантов ответа:

- 1) Интерфейс GAME/MIDI
- 2) FireWire (IEEE1394a)
- 3) Интерфейс LPT
- 4) Компонентный видеовыход
- 5) COM-порт

Задание #29

Вопрос:

Какое устройство обладает наибольшей скоростью обмена информацией?

Выберите один из 5 вариантов ответа:

- 1) CD-ROM дисковод
- 2) дисковод для гибких магнитных дисков
- 3) оперативная память
- 4) регистры процессора
- 5) жесткий диск

Промежуточная аттестация

Задание #30

Вопрос:

Открытая архитектура - это... ?

Выберите один из 3 вариантов ответа:

- 1) архитектура, предназначенная для выявления и устранения грубых погрешностей
- 2) архитектура компьютера или периферийного устройства, на которую опубликованы спецификации, что позволяет другим производителям разрабатывать дополнительные устройства к системам с такой архитектурой
- 3) архитектура, при которой происходит уменьшение погрешностей по сравнению с обычными цифровыми приборами при прочих равных условиях достигается за счет исключения систематических погрешностей в процессе самокалибровки.

Задание #31

Вопрос:

Электронный блок, управляющий работой внешнего устройства, называется:

Выберите один из 5 вариантов ответа:

- 1) драйвер
- 2) адаптер (контроллер)
- 3) регистр процессора
- 4) интерфейс
- 5) общая шина

Задание #32

Вопрос:

Постоянное запоминающее устройство служит для:

Выберите один из 4 вариантов ответа:

- 1) записи особо ценных прикладных программ
- 2) хранения программы пользователя во время его работы
- 3) хранения программ начальной загрузки компьютера и тестирования его узлов

4) постоянного хранения особо ценных документов

Задание #33

Вопрос:

Укажите на какой скорости может работать контроллер Ethernet

Выберите несколько из 6 вариантов ответа:

- 1) 10 Гбит/с
- 2) 102400 Кбит/с
- 3) 1000 Мбит/с
- 4) 102400 Мбит/с
- 5) 100 Мбит/с
- 6) 100 Гбит/с

Задание #34

Вопрос:

Каждый байт ОЗУ имеет

Выберите один из 4 вариантов ответа:

- 1) имя
- 2) индекс
- 3) название
- 4) адрес

Задание #35

Вопрос:

Какое количество основных информационных шин входит в системную магистраль микропроцессорной системы ?

Выберите один из 3 вариантов ответа:

- 1) Две шины.
- 2) Четыре шины.
- 3) Три шины.

Задание #36

Вопрос:

Хранение информации на внешних носителях отличается от хранения информации в оперативной памяти:

Выберите один из 4 вариантов ответа:

- 1) тем, что на внешних носителях информация может храниться после отключения питания компьютера
- 2) способами доступа к хранимой информации
- 3) объемом хранения информации
- 4) возможность защиты информации

Задание #37

Вопрос:

ОЗУ размещается

Выберите один из 4 вариантов ответа:

- 1) на жестком диске
- 2) на материнской плате

3) в процессоре

4) на магистрали

Задание #38

Вопрос:

Это главная микросхема компьютера, его 'мозг'. Он выполняет программный код, находящийся в памяти и руководит работой всех устройств компьютера.

Выберите один из 4 вариантов ответа:

- 1) Оперативная память
- 2) Чипсет
- 3) Процессор
- 4) Материнская плата

Задание #39

Вопрос:

Набор микросхем, обеспечивающих взаимодействие всех узлов компьютера.

Выберите один из 5 вариантов ответа:

- 1) Магистраль
- 2) ПЗУ
- 3) Сокет
- 4) Контроллер
- 5) Чипсет

Задание #40

Вопрос:

Стандартный форм фактор жесткого диска для настольного персонального компьютера

Выберите один из 6 вариантов ответа:

- 1) 1.8"
- 2) 1"
- 3) 3.5"
- 4) 1.3"
- 5) 4.5"
- 6) 2.5"

Задание #41

Вопрос:

_____ - это устройство, позволяющее получить электронную копию изображения с бумажного носителя.

Запишите ответ:

Задание #42

Вопрос:

Процессор IntelCore i5-3330 (3.0G) Sокет LGA1155 (OEM), какая система охлаждения подойдет к этому процессору?

Выберите несколько из 4 вариантов ответа:

- 1) Вентилятор Cooler Master Hyper 412P (RR-H412-20PK-R1) s.1155, 1156, 2011, 775, AM2, AM3, FM1 T
- 2) Intel Core i3 3240 (3.40GHz/3MB) Sокет LGA1155 (OEM)
- 3) Вентилятор Enermax ETS-T40-TB S775, S1155/1156, S1366, AM2, AM2+, AM3/AM3+/FM1
- 4) Вентилятор CPU Cooler Floston for AMD FCAM-23SQ, AL, тихий

Задание #43

Вопрос:

Материнская плата ASRock 970 Extreme3 R2.0 AM3 AMD970 4*DDR3 2 x PCI Express 2.0 x16 2 x PCI, какой в ней сокет?

Выберите один из 4 вариантов ответа:

- 1) R2.0
- 2) AM3
- 3) ASRock
- 4) 970

Задание #44

Вопрос:

Тактовая частота процессора - это

Выберите один из 5 вариантов ответа:

- 1) число возможных обращений процессора к оперативной памяти в единицу времени
- 2) количество тактов, выполняемых процессором в единицу времени
- 3) скорость обмена информацией между процессором и ПЗУ
- 4) число двоичных операций, совершаемых процессором в единицу времени
- 5) скорость обмена информацией между процессором и устройством ввода/вывода

Задание #45

Вопрос:

Какая кэш-память считается самой быстрой?

Выберите один из 3 вариантов ответа:

- 1) L3
- 2) L2
- 3) L1

Задание #46

Вопрос:

Принцип программного управления работой компьютера предполагает:

Выберите один из 3 вариантов ответа:

- 1) двоичное кодирование данных в компьютере
- 2) необходимость использования операционной системы для синхронной работы аппаратных средств
- 3) возможность выполнения без внешнего вмешательства целой серии команд

Задание #47

Вопрос:

В чем заключается принцип модернизации компьютера ?

Выберите несколько из 3 вариантов ответа:

- 1) В взаимозаменяемости деталей и узлов.
- 2) В функциональной избыточности деталей и узлов.
- 3) В совместимости деталей и узлов.

Задание #48

Вопрос:

Расположите пронумерованные команды так, чтобы был получен алгоритм, с помощью которого на пустой дискете создается файл с полным именем A:\TOWN \ STREET \ home.txt

Укажите порядок следования всех 5 вариантов ответа:

- ___ создать файл home.txt;
- ___ создать каталог TOWN;
- ___ сделать диск A: текущим.
- ___ войти в созданный каталог
- ___ создать каталог STREET;

Задание #49

Вопрос:

С какими видами данных работает компьютер...

Выберите один из 4 вариантов ответа:

- 1) истинными и ложными
- 2) символьными, числовыми, графическими, звуковыми
- 3) объективными и субъективными
- 4) аналоговыми и числовыми

Задание #50

Вопрос:

Основная шина, ради которой и создается вся система. Количество ее разрядов определяет скорость и эффективность информационного обмена, а также максимально возможное количество команд.

Выберите один из 4 вариантов ответа:

- 1) Шина данных

- 2) Шина адреса
- 3) Сервисная шина
- 4) Шина управления

Задание #51

Вопрос:

Наименьшая адресуемая часть оперативной памяти

Выберите один из 4 вариантов ответа:

- 1) бит
- 2) байт
- 3) файл
- 4) килобайт

Задание #52

Вопрос:

Основная характеристика кулеров обозначающая производительность вентилятора

Выберите один из 4 вариантов ответа:

- 1) dB
- 2) RPM
- 3) % об
- 4) CFM

Задание #53

Вопрос:

Разъем для подключения принтера

Выберите несколько из 4 вариантов ответа:

- 1) USB
- 2) PS/2
- 3) D-Sub
- 4) LPT

Задание #54

Вопрос:

Соответствие между поколениями ЭВМ и элементной базой.

Укажите соответствие для всех 5 вариантов ответа:

- 1) ламповые машины
 - 2) Элементная база ЭВМ были полупроводниковые приборы.
 - 3) ЭВМ применяются электронные микросхемы.
 - 4) Элементной базой ЭВМ были большие интегральные схемы.
 - 5) ЭВМ способны к самообучению, логической обработке информации, диалогу с пользователем в форме вопросов и ответов.
- ___ второе поколение
- ___ третье поколение
- ___ пятое поколение
- ___ четвертое поколение
- ___ первое поколение

Задание #55

Вопрос:

Перечислите основные характеристики компьютера ?

Выберите несколько из 4 вариантов ответа:

- 1) тактовая частота.
- 2) объем оперативной памяти.
- 3) разрядность.
- 4) производительность.

Задание #56

Вопрос:

Постоянное запоминающее устройство служит для:

Выберите один из 5 вариантов ответа:

- 1) записи особо ценных прикладных программ
- 2) хранения постоянно используемых программ
- 3) постоянно хранения особо ценных документов
- 4) хранение программ начальной загрузки компьютера и тестирование его узлов
- 5) хранения программы пользователя во время работы

Задание #57

Вопрос:

Объем ОЗУ измеряется:

Выберите один из 4 вариантов ответа:

- 1) в пикселях
- 2) в ГГц
- 3) в байтах
- 4) в ячейках

Задание #58

Вопрос:

Расположите носители информации по увеличению их возможной емкости.

Укажите порядок следования всех 4 вариантов ответа:

- ___ CD-RW
- ___ DVD-RW
- ___ Жесткий диск
- ___ Флоппи-диск (дискета)

Задание #59

Вопрос:

Основной разъем питания на материнской плате

Выберите несколько из 5 вариантов ответа:

- 1) 20-pin
- 2) 18-pin
- 3) 22-pin

4) 34-pin

5) 24-pin

Задание #60

Вопрос:

Какие из сокетов от фирмы Интел?

Выберите несколько из 5 вариантов ответа:

1) AM3+

2) LGA 2011

3) FM1

4) FM2

5) LGA 775

Практические задания

Лабораторная работа № 1 «Изучение открытой архитектуры»

Цель: Изучение устройства персонального компьютера.

Оборудование: ПК, клавиатура, мышь, набор отверток, программы SPEEDSYS и Ndiags.

Краткие теоретические сведения

Материнская плата является основной, на нее устанавливаются все остальные устройства ПК.

Рассмотрим основные устройства, расположенные на материнской плате.

Элементы питания. Устройства, расположенные на материнской плате, потребляют достаточно большую мощность (процессор до 110 Вт, видеокарты до 125 Вт и т. д.). Кроме того устройства, оснащенные электродвигателями (CD-ROM, жесткий диск и т. д.), в момент включения потребляют достаточно большой ток. Поэтому для нормального функционирования ПК необходимо стабилизировать питание, подаваемое на материнскую плату. При снижении напряжения ниже определенного уровня материнская плата вырабатывает сигнал сбоя питания и перезагружает ПК. Вторая функция системы питания – подавать и поддерживать на процессоре и некоторых других устройствах необходимое напряжение питания, которое может зависеть от производителя и модели этого устройства.

SMB (System Management Bus) – устройства. Это специальная система управления состоянием ПК. Она отвечает за работоспособность материнской платы (контроль напряжений и температур, скорость вращения вентиляторов и т. д.) и функционирование различных систем энергосбережения (переход в режимы с пониженным энергопотреблением и возврат в рабочее состояние).

BIOS (Basic Input/Output System) – базовая система ввода-вывода. Обеспечивает взаимодействие программных и аппаратных средств, настройку и диагностику устройств ПК, выполняет низкоуровневые операции ввода-вывода.

Чипсет (Chipset). Чипсет – набор микросхем, который обеспечивает функционирование материнской платы и устройств, установленных и подключенных к ней.

Кроме того, в состав чипсета могут входить дополнительные устройства, расширяющие функции материнской платы (встроенные видео, звуковые, сетевые и т. д. контроллеры).

Интегрированные контроллеры (не входящие в состав чипсета). Обычно это устройства, которые не встраивают в чипсет, или устройства, которые по своим свойствам превосходят встроенные. Например, это может быть достаточно мощная встроенная видеокарта, многоканальная звуковая карта, контроллер внешнего устройства (RAID, iLink и т. д.). Встроенные устройства имеют свои достоинства и недостатки.

Достоинства:

- Отсутствие необходимости в дополнительном оборудовании, если есть встроенное;
- Расширение функциональности (перепрошив BIOS устройства можно получить новые функции, которые ранее не использовались).

Недостатки:

- Невозможно заменить встроенное устройство;
- Некоторые устройства невозможно отключить (недоработка производителя);
- Обычно встроенные устройства уступают по своим характеристикам и набору функций дискретным (самостоятельным) устройствам.

Устройства настройки предназначены для настройки материнской платы и отдельных устройств.

Джампер – это перемычка, играющая роль выключателя, может быть в двух состояниях: включено, выключено. Иногда их собирают в блоки для выбора из нескольких вариантов, например, напряжение питания процессора.

Переключатели – аналогичны джамперам, так же собираются в блоки.

В последнее время все функции настройки переводят в BIOS, а джамперы с платы убирают, чтобы производить настройки, не вскрывая корпус.

Слоты и разъемы предназначены для установки различных устройств на материнскую плату. Их разновидности:

- Sокет – для установки процессора;
- Slot – для установки плат, процессора, памяти;
- Колодка – для установки BIOS и некоторых других микросхем;
- Штырьковые разъемы – для подключения CD-ROM, HDD, USB-порта;
- Прочие – могут быть как стандартными, так и зависеть от производителя, предназначены для облегчения подключения какого-либо устройства.

Ход работы

1. Перечислите разъемы для подключения внешних устройств на задней стенке корпуса.
2. Снимите кожух с ПК. Отсоедините провода питания и кабели для передачи данных. Снимите платы расширения.
3. Зарисуйте платы расширения и материнскую плату. Выясните назначение компонентов, расположенных на них, производителя и модель данного устройства. Результат занесите в таблицу 1.1.
4. Соберите ПК.
5. С помощью программ SPEEDSYS и Ndiags выясните компоненты ПК и результаты занесите в таблицу 1.2.
6. Ответьте на вопросы к лабораторной работе.
7. Напишите вывод по работе. Сравните способы состава устройств ПК и их характеристик, укажите достоинства и недостатки обоих способов.

Таблица 1.1

Компоненты ПК

Устройство	Производитель	Модель	Основные характеристики

Таблица 1.2

Компоненты ПК

Устройство	Производитель	Модель	Основные характеристики

Вопросы к лабораторной работе

- 1) Перечислите основные элементы системного блока.
- 2) Укажите основные элементы материнской платы.
- 3) Укажите основные элементы плат расширения (видеоплаты, сетевой платы, звуковой платы).
- 4) Изучите устройство клавиатуры и мыши.
- 5) Изучите устройство дисководов (жесткого диска, CD-ROMа, дисковода).

Лабораторная работа № 2 «Изучение логических и арифметических основ ЭВМ»

Цель: Изучение принципа действия логических элементов, триггеров, сумматоров.
Оборудование: ПК, программа Electronics Workbench.

Краткие теоретические сведения

Основой современных цифровых устройств, например ЦП, являются логические элементы.

Логический элемент – это устройство, преобразующее входные сигналы в сигналы на выходе в соответствии с реализуемой им логической функцией. Соответствие между входными и выходными сигналами задается в виде таблицы истинности.

Чаще всего используются логические функции (и элементы, их реализующие) «И», «ИЛИ», «НЕ» и «ИСКЛЮЧАЮЩЕЕ ИЛИ».

Более сложные узлы, такие как сумматор, триггер, регистр и т. д. построены из нескольких логических элементов.

Все узлы можно условно разделить на две категории. Первый тип узлов предназначен для преобразования (обработки) сигналов. Второй тип – для хранения информации в процессе ее обработки.

Элемент «И». Условно графическое обозначение (отечественное и зарубежное) представлено на рис. 2.1.

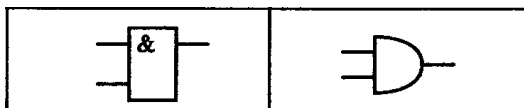


Рис. 2.1. Условно графическое обозначение логического элемента «И».

Таблица 2.1

Таблица истинности элемента «И»

X1	X2	Y
0	0	0
0	1	0
1	0	0
1	1	1

Элемент «ИЛИ». Условно графическое обозначение (отечественное и зарубежное) представлено на рис. 2.2.

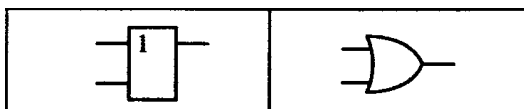


Рис. 2.2. Условно графическое обозначение логического элемента «ИЛИ».

Таблица 2.2

Таблица истинности элемента «ИЛИ»

X1	X2	Y
0	0	0
0	1	1
1	0	1
1	1	1

Элемент «НЕ». Условно графическое обозначение (отечественное и зарубежное) представлено на рис. 2.3.

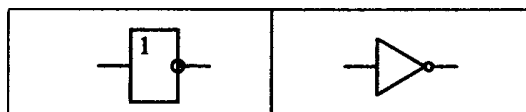


Рис. 2.3. Условно графическое обозначение логического элемента «НЕ».

Таблица 2.3

Таблица истинности элемента «НЕ»

X	Y
0	1
1	0

Элемент «ИСКЛЮЧАЮЩЕЕ ИЛИ». Условно графическое обозначение (отечественное и зарубежное) представлено на рис. 2.4

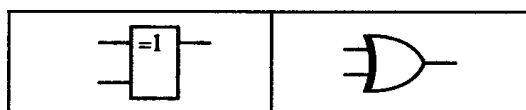


Рис. 2.4. Условно графическое обозначение логического элемента «Исключающее ИЛИ».

Таблица 2.4

Таблица истинности элемента «ИСКЛЮЧАЮЩЕЕ ИЛИ»

X1	X2	Y
0	0	0
0	1	1
1	0	1
1	1	0

Элемент «2И-НЕ». Этот элемент называют универсальным потому, что на его основе можно реализовать любую логическую функцию. Кроме того, технологически выгоднее изготовить десять однотипных элементов, чем два разного типа. Поэтому во всех реальных цифровых приборах используют этот элемент.

Условно графическое обозначение (отечественное и зарубежное) представлено на рис. 2.5.

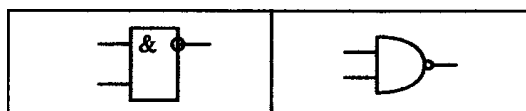


Рис. 2.5. Условно графическое обозначение логического элемента «2И-НЕ».

Таблица 2.5

Таблица истинности элемента «2И-НЕ»

X1	X2	Y
0	0	1

0	1	1
1	0	1
1	1	0

Триггер – это устройство с двумя устойчивыми состояниями предназначенное для хранения информации. Под действием сигналов на входах триггер может переходить из одного состояния в другое. Для удобства в использовании триггеры имеют два выхода: прямой Q и инверсный $\bar{Q}$. Существуют триггеры нескольких типов RS, JK, D, T. Рассмотрим принцип действия RS-триггера. Вход S (set) предназначен для установки 1 на прямом выходе триггера, вход R (reset) предназначен для установки 0 на прямом выходе триггера. Условно графическое обозначение триггера представлено на рис. 2.6.

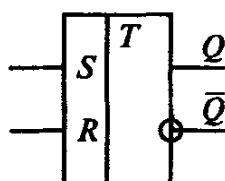


Рис. 2.6. Условно графическое обозначение триггера.

Таблица 2.6

Таблица переходов RS-триггера

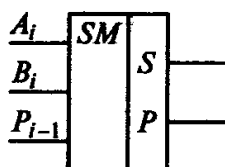
R	S	Q	$\bar{Q}$	Примечание
0	0	Q	$\bar{Q}$	Режим хранения
0	1	1	0	Установка 1
1	0	0	1	Установка 0
1	1	-	-	Запрещенное состояние

JK-триггер является универсальным, то есть на его основе можно построить любой другой триггер.

Для сложения двух одноразрядных двоичных чисел без учета переноса можно использовать полусумматор. Если необходимо учитывать перенос из предыдущих разрядов, используют сумматор. Условно графическое обозначение сумматора представлено на рис. 2.7.

Рис. 2.7. Условно графическое обозначение сумматора.

Слагаемые подаются на входы A и B. На вход P_{i-1} подается перенос из предыдущего



разряда (если он есть). На выходе S формируется сумма поданных на входы чисел, на выходе P – перенос в следующий разряд.

Из нескольких параллельно соединенных сумматоров можно составить сумматор для сложения двоичных чисел заданной разрядности.

Ход работы

1. Соберите цепь, состоящую из логического элемента, генератора цифровых сигналов и индикаторной лампы. Включите и убедитесь в ее работоспособности.
2. С помощью этой цепи составьте таблицу истинности для логических элементов «И», «ИЛИ» и «НЕ». Результаты запишите.
3. С помощью этой цепи составьте таблицу истинности для универсального логического элемента «2И-НЕ». Результаты запишите.
4. Составьте элементы «И», «ИЛИ», «НЕ» и «ИСКЛЮЧАЮЩЕЕ ИЛИ» используя универсальный логический элемент «2И-НЕ» и нарисуйте полученные схемы в тетрадь. Проверьте правильность с помощью генератора цифровых сигналов. Результаты запишите.
5. Соберите цепь, состоящую из триггера, генератора цифровых сигналов и индикаторных ламп. Включите и убедитесь в ее работоспособности.
6. С помощью этой цепи составьте таблицу состояний для RS-триггера. Результаты запишите. Нарисуйте временную диаграмму его работы.
7. Аналогично исследовать JK и D-триггеры.
8. Соберите цепь, состоящую из полусумматора, генератора цифровых сигналов и индикаторных ламп. Включите и убедитесь в ее работоспособности.
9. С помощью этой цепи составьте таблицу состояний для полусумматора. Результаты запишите.
10. Соберите цепь, состоящую из полного сумматора, генератора цифровых сигналов и индикаторных ламп. Включите и убедитесь в ее работоспособности.
11. С помощью этой цепи составьте таблицу состояний для полного сумматора. Результаты запишите.
12. Из 8 полных сумматоров составьте АЛУ для сложения 8-разрядных двоичных чисел. Проверьте его работоспособность. Результаты запишите.
13. Ответьте на вопросы к лабораторной работе.
14. Сделайте выводы по проделанной работе.

Вопросы к лабораторной работе

- 1) Перечислите основные элементы, предназначенные для преобразования информации.
- 2) Перечислите основные элементы, предназначенные для запоминания информации.
- 3) В чем заключается универсальность элемента «2И-НЕ»?
- 4) Какие значения будут получены на выходах RS-триггера, если на его входы R и S подать сигнал одновременно?
- 5) Для чего предназначены выходы сумматора?

Лабораторная работа № 3 «Сравнение скорости работы двух ПК»

Цель: Изучение методов сравнения скорости работы различных ЭВМ.

Оборудование: два ПК, Qbasic, SPEEDSYS, секундомер (часы).

Краткие теоретические сведения

Для сравнения различных типов компьютеров между собой применяют стандартные методики измерения производительности. Они позволяют разработчикам и пользователям осуществлять выбор между альтернативами на основе количественных показателей.

Производительность ПК во многом определяется производительностью ЦП. Производительность ЦП зависит от двух параметров: частоты синхронизации (тактовой частоты) и среднего количества тактов на команду. Невозможно изменить ни один из указанных параметров изолированно от другого, поскольку базовые технологии, используемые для изменения каждого из этих параметров, взаимосвязаны: частота синхронизации определяется технологией изготовления аппаратных средств и функциональной организацией процессора; среднее количество тактов на команду зависит от функциональной организации и архитектуры системы команд. Когда сравниваются две машины, необходимо рассматривать все компоненты, чтобы понять относительную производительность.

Кроме этого на производительность ПК влияет разрядность ЦП. Разрядность ЦП определяет, сколько разрядов отводится для формирования адреса при обращении к памяти. От этого зависит количество данных, обрабатываемых одной командой, и объем памяти, которую может использовать ЦП.

MIPS. Одной из альтернативных единиц измерения производительности процессора (по отношению ко времени выполнения) является MIPS - (миллион команд в секунду). В общем случае MIPS есть скорость операций в единицу времени, т.е. для любой данной программы MIPS есть просто отношение количества команд в программе к времени ее выполнения. Таким образом, производительность может быть определена как обратная ко времени выполнения величина, причем более быстрые машины при этом будут иметь более высокий рейтинг MIPS.

Однако использование MIPS в качестве метрики для сравнения наталкивается на две проблемы. Во-первых, MIPS зависит от набора команд процессора, что затрудняет сравнение по MIPS компьютеров, имеющих разные системы команд. Во-вторых, MIPS даже на одном и том же компьютере меняется от программы к программе.

MFLOPS. Обычно для научно-технических задач производительность процессора оценивается в MFLOPS (миллионах чисел-результатов вычислений с плавающей точкой в

секунду, или миллионах элементарных арифметических операций над числами с плавающей точкой, выполненных в секунду). Как единица измерения, MFLOPS, предназначена для оценки производительности только операций с плавающей точкой, и поэтому не применима вне этой ограниченной области. Ясно, что рейтинг MFLOPS зависит от машины, от программы. Этот термин менее безобидный, чем MIPS. Он базируется на количестве выполняемых операций, а не на количестве выполняемых команд. Именно поэтому рейтинг MFLOPS предназначался для справедливого сравнения различных машин между собой.

LINPACK – это пакет программ на Фортране для решения систем линейных алгебраических уравнений. Целью создания LINPACK отнюдь не было измерение производительности, но они могут быть использованы для этой цели. В основе действующего варианта LINPACK лежит решение системы 100 уравнений, содержащих 100 неизвестных (в последнем варианте размером 1000x1000). Все операции выполняются над числами с плавающей точкой, представленными с двойной точностью. Результат измеряется в MFLOPS.

Для многопроцессорных систем также имеются параллельные версии LINPACK, и такие системы часто показывают линейное увеличение производительности с ростом числа процессоров.

Однако, как и любая другая единица измерения, рейтинг MFLOPS для отдельной программы не может быть обобщен на все случаи жизни, чтобы представлять единственную единицу измерения производительности компьютера, хотя очень соблазнительно характеризовать машину единственным рейтингом MIPS или MFLOPS без указания программы.

Ход работы

1. Запустите Qbasic и наберите программу:

```
Cls: n=1000: dim a(n)

For I=1 to n

A(i)=int(rnd(1)*100)

Print a(i):print

Next I

For I=1 to n-1

For j=I+1 to n

If a(i)>a(j) then swap a(i), a(j)
```

```

Next j

Next I

For I=1 to n

Print a(i): print

Next I

Print "Конец работы"

```

2. Запустите программу и измерьте время ее выполнения. Для повышения достоверности результатов произведите измерения не менее 3 раз.
3. С помощью программы SPEEDSYS выясните характеристики ПК: тактовую частоту процессора, его разрядность, индекс производительности.
4. Повторите пункты 1-4 на другом ПК.
5. Результаты всех измерений занесите в таблицу 3.1.
6. Ответьте на вопросы к лабораторной работе.
7. Напишите вывод по проделанной работе.

Таблица 3.1

Сравнение характеристик двух ПК		
	ПК № 1	ПК № 2
Время выполнения программы, с		
Время выполнения программы, с		
Время выполнения программы, с		
Среднее время выполнения программы, с		
Частота процессора, МГц		
Разрядность процессора, бит		
Индекс производительности		

Вопросы к лабораторной работе

- 1) Каким образом влияет на производительность ПК разрядность процессора?
- 2) Каким образом влияет на производительность ПК тактовая частота процессора?
- 3) Скорость работы каких устройств ПК проверяется с помощью программы из этой лабораторной работы?
- 4) В чем разница между производительностью ЭВМ измеренной в MIPS и в MFLOPS?
- 5) Какой из тестов наиболее объективно позволяет сравнить производительность ЭВМ с различной архитектурой?

Лабораторная работа № 4 «Изучение системы программирования TASM»

Цель: Изучение системы программирования TASM.

Оборудование: ПК, MS-DOS, Pascal, ассемблер.

Краткие теоретические сведения

Процесс подготовки и отладки программы на языке ассемблера включает этапы подготовки исходного текста, трансляции, компоновки и отладки.

Подготовка исходного текста программы выполняется с помощью любого текстового редактора MS-DOS, или редактора, который может сохранять файл в формате «текст MS-DOS», однако в этом случае возникнут неприятности с русскими буквами. Файл с исходным текстом должен иметь расширение .ASM.

Следующая операция состоит в трансляции исходного текста программы, т.е. в преобразовании строк исходного языка в коды машинных команд. Эта операция выполняется с помощью транслятора с языка ассемблера (TASM.EXE).

После трансляции образуются два файла — листинг трансляции и объектный файл с расширением OBJ. Объектный файл является основным результатом работы транслятора и представляет собой текст программы, преобразованный в машинные коды. Хотя в этом файле уже присутствуют коды команд, он не может быть выполнен. Для того чтобы получить выполнимую программу, объектный файл необходимо скомпоновать.

Компоновка объектного файла выполняется с помощью программы-компоновщика (редактора связей – TLINK.EXE). Эта программа получила такое название потому, что ее основное назначение — подсоединение к файлу с основной программой файлов с подпрограммами и настройка связей между ними. Однако компоновать необходимо даже простейшие программы, не содержащие подпрограмм. Дело в том, что у компоновщика имеется и вторая функция — изменение формата объектного файла и преобразование его в выполнимый файл, который может быть загружен в оперативную память и выполнен. В зависимости от режима компоновки может получиться программа типа EXE или типа COM.

Отладку и изучение работы готовой программы удобнее всего осуществлять с помощью интерактивного отладчика (TD.EXE), который позволяет выполнять отлаживаемую программу по шагам или с точками останова, выводить на экран содержимое регистров и областей памяти, модифицировать (в известных пределах) загруженную в память программу, принудительно изменять содержимое регистров и выполнять другие действия, позволяющие в наглядной и удобной форме контролировать выполнение программы.

Ход работы

1. Создайте рабочий каталог, например, ASM. Скопируйте в него из каталога BP\BIN файлы: TASM.EXE, TLINK.EXE, TD.EXE, DPMIMEM.DLL, DPMINST.EXE, DPMILOAD.EXE, DPMI16BI.OVL.
2. Наберите в любом текстовом редакторе тексты программ 4.1. и 4.2. и сохраните их с расширением .ASM.
3. Откомпилируйте программы. Для этого наберите команды:


```
TASM ИМЯ_ФАЙЛА.ASM
TLINK ИМЯ_ФАЙЛА.OBJ
```

 (программа типа .EXE)


```
TASM ИМЯ_ФАЙЛА.ASM
TLINK /T ИМЯ_ФАЙЛА.OBJ
```

 (программа типа .COM)
4. Запустите программы. Убедитесь в их работоспособности.
5. Загрузите откомпилированные программы в отладчик. Просмотрите структуру программы, выполните ее в пошаговом режиме. При выполнении программ просмотрите содержимое регистров.
6. Ответьте на вопросы к лабораторной работе.
7. Сделайте вывод по проделанной работе.

Пример программы типа EXE 4.1.

```
.model small
.stack 100h
.data
hm db 'H',13,10,'e',13,10,'l',13,10,'l',13,10,'o','$'
.code
    mov ax,@data
    mov ds,ax
    mov ah,9
    mov dx,offset hm
    int 21h
    mov ah,4ch
    int 21h
end
```

Пример программы типа COM 4.2.

```
.model tiny
.code
```

```

org 100h
start:
    mov bl,10h
    mov dh,1
    mov dl,1
    mov ah,09
    mov al,'t'
    int 10h
    mov ah,4ch
    int 21h
end start

```

Вопросы к лабораторной работе

- 1) Почему для редактирования исходных текстов программ необходимо использовать текстовые редакторы MS-DOS?
- 2) Чем отличаются программы типа EXE от программ типа COM?
- 3) Сколько сегментов в памяти занимают приведенные в данной работе программы?
- 4) Какие типы файлов получаются после компиляции программы, приведите их назначение.
- 5) Для чего предназначен отладчик, и какие действия с программой он позволяет производить?

Лабораторная работа № 5 «Изучение системы ввода-вывода»

Цель: Изучение принципов ввода и вывода информации.

Оборудование: ПК, ассемблер.

Краткие теоретические сведения

Вывод одиночных символов. Для вывода символа на экран можно использовать непосредственное обращение к функции ввода-вывода операционной системы или BIOS, например:

```

MOV AH,6
MOV DL,'C'
INT 21H

```

Здесь в регистр AH загружается код функции, указывающий операционной системе, что должен быть выведен символ, содержащийся в регистре DL. Выводимый символ заносится в регистр DL, по команде INT 21H происходит прерывание и управление передается DOS. Одновременно с появлением символа на экране курсор перемещается на

одну позицию вправо. Затем управление возвращается прикладной программе и начинается выполнение команды, следующей за командой INT 21H. В регистр DL можно помещать также ASCII код символа. Например, заглавной букве А соответствует код 41H.

В рассмотренном выше примере на экран выводится один символ, однако существует способ, при котором за одно обращение на экран выводится сразу несколько символов: эта возможность реализуется путем вызова функции вывода символьной строки. Соответствующая последовательность команд имеет вид

```
MOV AH, 9H
MOV DX, OFFSET MSG
INT 21H
```

Здесь в регистр AH помещается код вызываемой функции 9H, а в регистр DX загружается адрес последовательности байтов, которая должна быть выведена на экран. При обработке прерывания управление передается операционной системе, и строка символов выдается на экран. Выводимая строка может состоять из любых символов, но должна заканчиваться знаком конца строки – \$

Например, необходимо вывести на экран строку: THIS IS A MESSAGE Для описания строки, выводимой на экран, используется инструкция:

```
MSG DB 'THIS IS A MESSAGES', '$'
```

Здесь MSG – имя переменной, содержащей строку, DB – тип переменной (последовательность байтов), '\$' – конец строки.

Управление дисплеем. При других обращениях к дисплею, используется прерывание INT 10H, по которой управление передается соответствующей подпрограмме BIOS.

Выбор режима. Дисплей IBM PC допускает вывод изображения в различных форматах, как текстовых, так и графических. Задание режима производится с помощью следующей программы:

```
MOV AH, 0
MOV AL, 13H
INT 10H
```

В регистр AH загружается 0, что указывает на изменение режима вывода информации на экран. В регистр AL также заносится число, определяющее режим, в котором на экране будет выводиться информация.

Вывод точки на экран. Для вывода на экран точки в графическом режиме можно использовать непосредственное обращение к функции BIOS или запись в видеопамять, например:

```
MOV AL, 4  
  
MOV AH, 0CH  
  
INT 10H
```

В регистр AL загружается номер цвета выводимой точки (в различных видеорежимах число цветов может быть разным – 2, 16 или 256). В регистр AH загружается номер функции BIOS (0C – вывод точки). По команде INT 10H управление передается BIOS и производится вывод точки на экран.

Считывание с клавиатуры. Ввод данных с клавиатуры можно производить двумя способами: считывание отдельных символов и ввод целой строки символов. Для ввода одного символа можно использовать функции BIOS, например:

```
MOV AH, 00H  
  
INT 16H
```

В регистр AH загружается 0 – код функции BIOS для ввода 1 символа. По команде INT 16H происходит прерывание, и управление передается BIOS. Код введенного символа помещается в регистр AL.

Считывание строки производится иначе. Пока нажаты не все клавиши, управление программе не возвращается, а операционная система организует считывание данных с клавиатуры и запоминание кодов, соответствующих нажатым клавишам. Когда нажимается клавиша ENTER, управление передается основной программе вместе с последовательностью кодов, соответствующих нажатым клавишам. Такой процесс называется буферизованным вводом с клавиатуры. Буферизованный ввод сопровождается эхо-отображением, т. е. при считывании кода нажатой клавиши символ, соответствующий этому коду, выводится на экран дисплея. Допустим, что была нажата клавиша А; в этом случае буква А появится на экране, а в буфер прикладной программы будет передан один байт данных — код, соответствующий букве А.

При использовании функции буферизованного ввода необходимо указывать максимальное число символов, подлежащих считыванию, а при возврате из функции программе должно сообщаться фактическое число считанных с клавиатуры символов. Для обеспечения возможности обработки цепочки введенных символов пользователь, обращаясь к функции считывания строки, в качестве параметра передает адрес

переменной, в которую будет помещена считанная строка. Строка может быть описана следующим образом:

```
BUF DB 10,0,0,0,0,0,0,0,0,0,0
```

В первом байте (число 10) указывается максимальное число клавиш, коды которых предполагается считывать в программе. Сюда же включается и код клавиши ENTER, которая должна нажиматься последней. Если пользователь попытается ввести больше символов, чем указано в буфере, раздастся сигнал звукового предупреждения, а очередной «лишний» код будет проигнорирован. Когда после завершения операции считывания управление будет вновь возвращено прикладной программе, второй байт строки будет содержать число считанных кодов. Таким образом, первые два байта не содержат самих кодов символов, соответствующих нажимавшимся клавишам. Эти представления находятся в следующих десяти байтах строки. Последним занесенным в буфер кодом будет код, соответствующий клавише ENTER.

Последовательность команд обращения к функции буферизованного ввода выглядит следующим образом:

```
MOV AH, .0AH  
MOV DX.OFFSET BUF  
INT 21H
```

В регистр AH загружается код (0AH) вызова функции ввода строки. Метка BUF указывает на описанную выше строку. По команде INT 21H управление передается операционной системе. По мере того как будут нажиматься клавиши на клавиатуре, символы, соответствующие этим клавишам, будут появляться на экране, а соответствующие им восьмибайтовые коды будут заноситься в строку BUF. После введения необходимого числа символов необходимо нажать клавишу ENTER. Код, соответствующий этой клавише, помещается в строку BUF, и управление передается прикладной программе.

Ход работы

1. Наберите в любом текстовом редакторе тексты программ и сохраните их с расширением .ASM.
2. Откомпилируйте программы.
3. Запустите программы.
4. Загрузите откомпилированные программы в отладчик. Просмотрите структуру программы, выполните ее в пошаговом режиме. Для программы из таблицы 5.3 запишите код введенного с клавиатуры символа. Для программы из таблицы 5.4 найдите в памяти введенную строку символов.

5. Ответьте на вопросы к лабораторной работе.
6. Сделайте вывод по проделанной работе.

Пример программы 5.1. Вывод текста на экран, используя функции операционной системы.

.model small	Выбираем модель памяти
.stack 100h	Задаем размер стека (256 байт)
.data	
st1 db 'Ok!',13,10,'\$'	На экран будет выведена строка – "Ok!"
.code	
start:	
mov ax,@data	Настроим сегментный регистр DS на
mov ds,ax	сегмент данных
mov ah,09h	Функция DOS для вывода на экран
	строки, адрес которой помещен в
	регистр DX
mov dx,offset st1	Загрузим в DX адрес строки, которую
	надо вывести на экран
int 21h	Вызов DOS
mov ah,4ch	Функция DOS завершения программы
int 21h	Вызов DOS
end start	

Пример программы 5.2. Рисование горизонтальной линии, используя функции BIOS.

.model small	
.stack 100h	
.code	
mov ah,00h	
mov al,13h	Задаем номер графического экрана (13h –
int 10h	EGA 320x240 256 цветов)
mov cx,319	Задаем число повторений цикла
mov dx,100	Задаем координаты точки
lin:	
mov al,4	Задаем цвет точки (4 – красный)
mov ah,0ch	Функция BIOS для вывода точки на экран
int 10h	
loop lin	Переход на метку lin
mov ah,4ch	Функция DOS завершения программы
int 21h	
end	

Пример программы 5.3. Ввод символа с клавиатуры, используя функции BIOS.

.model small	
.stack 100h	
.code	
mov ah,00h	Функция BIOS ввода с клавиатуры (ASCII код
	введенного символа в регистре AL)
int 16h	Вызов BIOS
mov ah,4ch	Функция DOS завершения программы
int 21h	
end	

Пример программы 5.4. Ввод строки символов с клавиатуры, используя функцию DOS.

<pre>.model small .stack 100h .data string DB 50 dup("\$") .code mov ax,@data mov ds,ax mov dx,offset string mov ah,0ah int 21h mov ah,4ch int 21h end</pre>	Подготовим место в памяти для введенной строки Настроим регистр DS на начало сегмента данных Настроим регистр DX на начало отведенной под строку области памяти Функция DOS ввода строки символов, ENTER – конец ввода Функция DOS завершения программы
---	--

Вопросы к лабораторной работе

- 1) Можно ли изменить позицию, в которую будет выведена строка в программе 5.1?
 - 2) Как изменить разрешение экрана и количество цветов в программе 5.2?
 - 3) Как изменить цвет и позицию линии на экране в программе 5.2?
 - 4) Как изменяется содержимое флагового регистра в процессе работы программ 5.3. и 5.4?
- Для чего производится сохранение в стек содержимого регистров на время обработки прерывания?
- 5) Как изменить программу 5.2, чтобы линия состояла из точек разного цвета?

Лабораторная работа № 6 «Изучение работы обработчика аппаратных прерываний»

Цель: Изучение работы обработчика аппаратных прерываний.
Оборудование: ПК, ассемблер.

Краткие теоретические сведения

Обработчики прерываний являются важнейшей составной частью многих программных продуктов. В прикладных программах приходится обрабатывать, главным образом, внешние и программные прерывания. Общие принципы обслуживания тех и других прерываний одинаковы, однако условия функционирования обработчиков аппаратных прерываний имеют значительную специфику, связанную главным образом с тем, что прерывания от аппаратуры приходят в произвольные моменты времени и могут прервать текущую программу в любой ее точке. Обработчик прерывания должен быть написан таким образом, чтобы его выполнение ни в какой степени не отразилось на правильном функционировании текущей (прерываемой) программы.

Обработчик прерываний может входить в состав программы в виде процедуры, или просто являться частью программы, начинающейся с некоторой метки (входной точки обработчика) и завершающейся командой выхода из прерывания `iret`.

Программа, начиная свою работу, прежде всего должна выполнить инициализирующие действия по установке обработчика прерываний. В простейшем случае эти действия заключаются в занесении в соответствующий вектор полного адреса (сегмента и смещения) обработчика. Часто инициализация обработчика, помимо установки вектора, предполагает и другие действия: сохранение исходного содержимого вектора прерывания, размаскирование соответствующего уровня прерываний в контроллере прерываний, посылка в устройство команды разрешения прерываний и т. д.

Установив обработчик, программа может приступить к дальнейшей работе. В случае прихода прерывания, процессор сохраняет в стеке флаги и текущий адрес программы, извлекает из вектора адрес обработчика и передает управление на его входную точку. Все время, пока выполняется программа обработчика, основная программа, естественно, стоит. Завершающая обработчик команда `iret` извлекает из стека сохраненные там данные и возвращает управление в прерванную программу, которая может продолжить свою работу. Последующие прерывания обслуживаются точно так же.

Рассмотрим структуру программы с обработкой аппаратных прерываний. Наиболее удобным аппаратным прерыванием, которое можно использовать в исследовательских целях, является прерывание от системного таймера, которое генерируется 18.2 раза в секунду и служит источником сигналов для хода системных часов, отсчитывающих время, истекшее после включения машины. Замена системного обработчика на прикладной не приводит к каким-либо неприятностям, кроме, возможно, остановки на некоторое время системных часов.

В приведенном примере обработчик прерываний расположен в конце программы, после главной процедуры `main`, хотя взаимное расположение процедур не имеет ни малейшего значения. Не имеет также значения, выделен ли обработчик в отдельную процедуру или просто начинается с метки.

Для того, чтобы прикладной обработчик получал управление в результате прерываний, его адрес следует поместить в соответствующий вектор прерывания. При этом исходное содержимое вектора будет затерто, и если прерывания будут поступать и после завершения программы, возникнет весьма неприятная ситуация, когда управление будет передаваться по адресу, по которому в памяти может располагаться что угодно. Поэтому стандартной методикой является сохранение в памяти исходного содержимого вектора и восстановление этого содержимого перед завершением программы.

Для чтения и заполнения вектора прерываний можно использовать специально предусмотренные для этого функции DOS. Для чтения вектора используется функция с номером 35h. В регистр AL помещается номер вектора. Функция возвращает исходное содержимое вектора в паре регистров ES:BX (в ES сегментный адрес, а в BX смещение). Для хранения исходного содержимого вектора в сегменте данных предусмотрена переменная `old_08`.

Сохранив исходный вектор, можно установить в нем адрес нашего обработчика. Для установки вектора в DOS предусмотрена функция 25h. Она требует указания номера устанавливаемого вектора в регистре AL, а его полного адреса — в паре регистров DS:DX. Занести в регистр DX смещение нашего обработчика `new_08` не составляет труда, это делается командой

```
mov DX,offset new_08.
```

Однако регистр DS у нас занят — в нем хранится сегментный адрес сегмента данных. Придется на какое-то время сохранить этот адрес, для чего удобнее всего воспользоваться стеком. Содержимое CS отправляется в стек и тут же извлекается оттуда в регистр DS командами:

```
push CS
```

```
pop DS
```

После возврата из DOS надо не забыть восстановить исходное содержимое DS, сохраненное в стеке. Инициализация обработчика прерываний закончена. Начиная с этого момента, каждый сигнал таймера будет приводить к прерыванию продолжающейся основной программы и передаче управления на процедуру `nevv_08`.

Перед завершением программы необходимо поместить в вектор 8 адрес исходного, системного обработчика, который был сохранен в `old_08`. Перед вызовом функции 25h установки вектора в регистры DS:DX надо занести содержимое этой переменной. Эту операцию можно выполнить одной командой `lds`, если указать в качестве ее первого операнда регистр DX, а в качестве второго — адрес `old_08`.

Последнее, что нам осталось рассмотреть — это стандартные действия по завершению самого обработчика прерываний. Выше уже говорилось, что последней командой обработчика должна быть команда `iret`, возвращающая управление в прерванную программу. Однако перед ней необходимо выполнить еще одно обязательное действие — послать в контроллер прерываний команду конца прерываний. Дело в том, что контроллер прерываний, передав в процессор сигнал прерывания INT, блокирует внутри себя линии прерываний, начиная с той, которая вызвала данное прерывание, и до последней в порядке возрастания номеров IRQ. Таким образом, прерывание, пришедшее, например, по линии

IRQ 6 (гибкий диск) заблокирует дальнейшую обработку прерываний по линиям 6 и 7, а прерывание от таймера (IRQ 0) блокирует вообще все прерывания. Любой обработчик аппаратного прерывания обязан перед своим завершением снять блокировку в контроллере прерываний, иначе вся система прерываний выйдет из строя. Снятие блокировки осуществляется посылкой команды с кодом 20h в один из двух портов, закрепленных за контроллером прерываний. Для ведущего контроллера эта команда посылается в порт 20h, для ведомого — порт A0h.

Ход работы

1. Наберите в любом текстовом редакторе текст программы из таблицы 6.1. и сохраните ее с расширением .ASM.
2. Откомпилируйте программу.
3. Запустите программу. Пронаблюдайте ее работу.
4. Ответьте на вопросы к лабораторной работе.
5. Сделайте вывод по проделанной работе.

Вопросы к лабораторной работе

- 1) Для чего предназначены обработчики аппаратных прерываний?
- 2) Обработку какого прерывания производит приведенная в этой работе программа?
- 3) Для чего сохраняется исходный вектор прерывания?
- 4) Для чего производится сохранение в стек содержимого сегментных регистров на время обработки прерывания?
- 5) В каких случаях происходит выход из программы?

Пример программы 6.1. Обработчик аппаратных прерываний

code segment	
assume CS:code,DS:data	
main proc	Начало главной процедуры
mov AX,data	
mov DS,AX	
	Сохраним исходный вектор
mov AH,35h	Функция получения вектора
mov AL,08h	Номер вектора
int 21h	
mov word ptr old_08,BX	Смещение исходного обработчика
mov word ptr old_08+2,ES	
	Установим новый обработчик
mov AH,25h	Функция заполнения вектора
mov AL,08h	Номер вектора
mov DX,offset new_08;	Смещение нашего обработчика
push DS	Сохраним DS=data
push CS	Перешлем CS в DS
pop DS	через стек. DS:DX >new_08
int 21h	
pop DS	Восстановим DS=data

mov ah,01h	Ждем нажатия любой клавиши
int 21h	Перед завершением программы восстановим исходный вектор
lds DX,old_08	Заполним DS:DX из old_08
mov AH,25h	Функция заполнения вектора
mov AL,08h	Номер вектора
int 21h	
mov Ah,4Ch	Функция завершения программы
int 21h	
main endp	Конец главной процедуры
new_08 proc	Процедура – обработчик прерывания
push AX	Сохраним исходное значение AX
mov AH,0Eh	Функция вывода символа
mov AL,'R'	Выводимый символ
int 10h	
mov AL,20h	Разблокировка прерываний в контроллере прерываний
out 20h,AL	восстановим AX
pop AX	
iret	Возврат в прерванную программу
new_08 endp	
code ends	
data segment	
old_08 dd 0	Ячейка для хранения исходного вектора
data ends	
stk segment stack	
db 256 dup (0)	
stk ends	
end main	

Лабораторная работа № 7 «Изучение способов обмена данными с внешними устройствами»

Цели: Изучение работы системы ввода-вывода.

Оборудование: ПК, MS-DOS, Pascal, ассемблер.

Краткие теоретические сведения

Система ввода-вывода является одной из важнейших частей архитектуры процессора и машины в целом. К системе ввода-вывода можно отнести и способы подключения к системной шине различного оборудования и команды процессора, предназначенные для обмена данными с внешними устройствами. Системная шина представляет собой набор линий — проводов, к которым подключаются все устройства компьютера. Сигналы, распространяющиеся по шине, доступны всем подключенным к ней устройствам. В задачу каждого устройства входит выбор предназначенных ему сигналов. Процессор связан с системной шиной тремя типами линий: линии адресов, линии данных и линии управления.

Для передачи данных между программой и устройством применяют разные методы:

- программно-управляемый обмен;
- программированный ввод-вывод;
- обмен по каналу DMA;
- обмен в режиме прямого управления шиной.

Первые два метода позволяют программе непосредственно обмениваться данными с устройством — все данные во время обмена с устройством проходят через процессор.

Программно-управляемый обмен сильно загружает процессор, особенно если программа формирует и управляющие сигналы обмена. Так, например, работает драйвер параллельного порта в стандартном режиме. В результате предел пропускной способности такого порта, в зависимости от производительности процессора, может быть порядка 150 Кбайт/с.

Режим программного обмена PIO (Programmed Input/Output) построен на инструкциях блочной пересылки данных. Для этих инструкций задается начальный адрес памяти, длина блока, адрес порта и направление изменения адреса памяти (инкремент или декремент). Пересылки выполняются быстро, скорость передачи определяется производительностью (частотой) процессора и шины. Обмен в режиме PIO успешно применяется и для обмена с устройствами ATA (IDE). Поскольку полную скорость PIO периферийные устройства воспринять обычно не могут, контроллер интерфейса, аппаратно вводя такты ожидания, «притормаживает» обмен до разумных скоростей, определяемых режимом обмена.

Обмен по прямому доступу к памяти (DMA) почти не загружает процессор — выполняемые им инструкции ввода-вывода относятся лишь к анализу состояния и инициализации канала DMA, но не к самой передаче данных. Время отклика на одиночный запрос, когда контроллер «заряжен» на обмен, не превышает сотен наносекунд. Однако скорость стандартной передачи данных ниже, чем в режиме PIO.

Обмен в режиме прямого управления шиной (bus mastering) выполняется по инициативе и под управлением относительно интеллектуального контроллера, здесь центральный процессор загружается в наименьшей степени (занимается только шиной). Производительность в режимах прямого управления шиной обычно выше, чем у DMA. В таком режиме могут работать контроллеры жестких дисков, видеокарты для шины AGP и некоторые другие устройства.

В данной работе для большей наглядности во всех программах был использован графический видеорежим.

Ход работы

1. Включите ПК и убедитесь в его работоспособности. Запустите Turbo Pascal.

2. Установите видеорежим. Запустите Pascal, для установки видеорежима наберите и запустите программу 7.1. Будет установлен видеорежим EGA 13h (320x240, 256 цветов). Опробуйте работу в других видеорежимах (10h - 12h).
3. Вывод точки на экран. Для вывода точки на экран прямой записью в видеопамять наберите и запустите программу 7.2.
4. Вывод палитры для данного видеорежима. Добавьте в программу строки.


```
for i:=1 to 255 do
  for j:=0 to 10 do mem[$A000:(320*j+i)]:=i;
```
5. Нарисуйте прямую (вертикальную и горизонтальную), используя прямой доступ к видеопамяти.
6. Сравнение способов работы с видеоизображениями. Наберите программы 7.3. 7.4. и сравните скорость их выполнения.
7. Ответьте на вопросы к лабораторной работе.
8. Сделайте вывод по проделанной работе.

Пример программы 7.1. Установка видеорежима.

```
begin
  asm
    mov ax,13h
    int 10h
  end;
  readln;
end.
```

Пример программы 7.2. Рисование точки.

```
begin
  asm
    mov ax,13h
    int 10h
  end;
  mem[$A000:0]:=2;
  readln;
end.
```

Пример программы 7.3. Запись в видеопамять.

```
var i,j:integer;
begin
```

```

asm
mov ax,13h
int 10h
end;
for i:=0 to 239 do
for j:=0 to 319 do
mem[$a000:320*i+j]:=1;
readln;
end.

```

Пример программы 7.4. Ввод-вывод по прерываниям (программно-управляемый обмен данными).

```

label l1;
begin
asm
mov ah,00h
mov al,13h
int 10h
mov cx,0h
mov dx,0h
l1:
inc cx
mov al,4
mov ah,0ch
int 10h
cmp cx,319
jne l1
inc dx
mov cx,0
cmp dx,239
jne l1
end;
readln;

end.

```

Вопросы к лабораторной работе

- 1) Назовите три составные части шины.

- 2) Перечислите типы видеорежимов, в которых может работать ПК.
- 3) Назовите способы ввода-вывода данных.
- 4) Какой способ ввода-вывода позволяет передавать данные с максимальной скоростью?
- 5) Почему запись данных в видеобuffer производится быстрее, чем ввод-вывод по прерываниям?

Лабораторная работа № 8 «Изучение BIOS SETUP»

Цель: Изучение возможности настройки ПК с помощью BIOS SETUP.

Оборудование: ПК, справочные материалы.

Краткие теоретические сведения

BIOS — базовая система ввода-вывода, предназначена для того, чтобы операционная система и прикладные программы не зависели от специфических особенностей конкретной аппаратуры. BIOS находится в микросхемах энергонезависимой памяти, расположенных на системной плате, на картах расширения могут находиться дополнительные модули BIOS, поддерживающие функционирование этих карт. *ROM BIOS* хранится в микросхемах ПЗУ, которые могут быть и перепрограммируемыми. Флэш-BIOS (*Flash-BIOS*) хранится в микросхемах флэш-памяти, допускающей перепрограммирование прямо на месте установки. В нормальном режиме работы компьютера информация в микросхемах как ROM, так и флэш-BIOS является постоянной.

Системная ROM BIOS обеспечивает программную поддержку стандартных устройств PC, конфигурирование аппаратных средств, их диагностику и вызов загрузчика операционной системы. Системная ROM BIOS в значительной степени привязана к конкретной реализации системной платы, поскольку именно ей приходится программировать все микросхемы чипсета системной платы. Функции BIOS разделяются на следующие группы:

- инициализация и начальное тестирование аппаратных средств — POST (Power On Self Test);
- настройка и конфигурирование аппаратных средств и системных ресурсов — CMOS Setup;
- автоматическое распределение системных ресурсов — PnP BIOS;
- идентификация и конфигурирование устройств PCI — PCI BIOS;
- начальная загрузка (первый шаг загрузки операционной системы) — Bootstrap Loader;

- обслуживание аппаратных прерываний от системных устройств (таймера, клавиатуры, дисков) — BIOS Hardware Interrupts;
- отработка базовых функций программных обращений (сервисов) к системным устройствам — ROM BIOS Services;
- поддержка управляемости конфигурированием — DMI BIOS;
- поддержка управления энергопотреблением и автоматического конфигурирования - APM и ACPI BIOS.

Все эти функции исполняет системный модуль System BIOS, хранящейся в микросхеме ПЗУ или флэш-памяти на системной плате. Системная BIOS должна обслуживать по вышеуказанным функциям все компоненты, установленные на системной плате: процессор, контроллер памяти (ОЗУ и кэш), стандартные архитектурные компоненты (контроллеры прерываний и DMA, системный таймер, системный порт, CMOS RTC), контроллер клавиатуры, а также набор стандартных периферийных контроллеров и адаптеров, даже если они и не установлены на системной плате.

PnP BIOS описывает следующие расширения возможностей традиционной BIOS:

- распределение ресурсов и разрешение конфликтов на этапе выполнения POST;
- введение контролируемого механизма удаленной загрузки (RPL);
- поддержка конфигурирования в рабочем режиме;
- обеспечение уведомления о динамическом изменении конфигураций (подключения и отключения устройств).

ACPI Advanced Configuration and Power Interface — расширенный интерфейс конфигурирования и питания. ACPI представляет собой довольно сложную комбинацию функций, часть из которых раньше возлагались на относительно независимые системы PnP и APM (расширенное управление питанием). Спецификация ACPI позволяет операционной системе управлять конфигурированием и энергопотреблением устройств и компьютера в целом. ACPI определяет аппаратные и программные интерфейсы.

Интерфейс ACPI предоставляет операционной системе возможность прямого (и эксклюзивного) управления потреблением и конфигурированием устройств системной платы. При запуске ACPI забирает эти функции от старых интерфейсов BIOS (APM BIOS, PnP BIOS) и берет на себя ответственность за обработку событий конфигурирования устройств системной платы, управление питанием, производительностью и температурным состоянием системы в соответствии с предпочтениями пользователя и требованиями приложений. Ниже перечислены области, охватываемые спецификацией ACPI:

- Управление питанием системы.

- Управление питанием устройств.
- Управление производительностью процессора и устройств.
- Plug and Play.
- Системные события.
- Управление батареями.
- Термоконтроль.
- Встроенные контроллеры.
- Контроллер SMBus (вспомогательной последовательной шины системного управления).

Ход работы

1. Включите ПК и убедитесь в его работоспособности.
2. Перезагрузите ПК и войдите в BIOS SETUP (в начале загрузки ПК нажать DEL или F2).
3. Пользуясь справочными материалами, изучите и запишите назначение основных пунктов меню в таблицу 8.1.
4. Пользуясь справочными материалами, найдите и запишите пункты меню, отвечающие за настройку ПК.
5. Измените пароль на BIOS SETUP.
6. Перезагрузите ПК и убедитесь в наличии пароля.
7. Запустите Qbasic и сбросьте настройки BIOS с помощью программы:


```
For I=1 to 255
  Out &h70,I
  Out &h71,0
Next I
```
8. Восстановите настройки (используйте автонастройку, определите жесткие диски, настройте порядок загрузки и дату/время).
9. Ответьте на вопросы к лабораторной работе.
10. Сделайте вывод по проделанной работе.

Таблица 8.1.

Назначение основных пунктов меню

	Пункт меню.	Назначение.
1.		
2.		
3.		
4.		
5.		
...	...	...

Вопросы к лабораторной работе

- 1) Укажите назначение и основные функции BIOS.
- 2) Перечислите способы изменения пароля на BIOS SETUP.
- 3) Приведите и сравните 2 способа выхода из BIOS SETUP.
- 4) Перечислите возможные варианты порядка загрузки ПК, обоснуйте их необходимость.
- 5) Приведите и сравните способы автоматической настройки BIOS SETUP, обоснуйте их необходимость.

Вопросы для экзамена

1. Шинная архитектура. Типы процессоров.
2. Математические основы, способы организации.
3. Канальная архитектура.
4. Особенности ассоциативных процессоров.
5. Конвейерные и матричные процессоры.
6. Принцип адресности.
7. Гарвардская архитектура.
8. Повышения производительности процессоров.
9. Кэш-память.
10. Критерии классификации компьютеров.
11. Физическое разделение линий передачи команд и данных.
12. Номенклатура комплектующих компьютеров.
13. Архитектура процессоров - CISC.
14. Архитектура процессоров - RISC.
15. Архитектура закрытого типа и устройств, входящих в данную архитектуру.
16. Микропроцессоры.
17. Архитектура фон Неймана.
18. Сопроцессоры, микропроцессорные системы, системам на кристалле.
19. Виртуальная машина.
20. Платформы и архитектуры CPU
21. Принцип однородности памяти.
22. Платформы и архитектуры NetBSD.
23. Платформы-анклавы.
24. Общее представление архитектуры компьютера.
25. Типы, виды, классы архитектур.
26. Векторно-конвейерные суперкомпьютеры.
27. Микропроцессоры.
28. Симметричные мультимикропроцессорные системы (SMP).
29. Сопроцессоры, микропроцессорные системы, системам на кристалле;
30. Системы с массовым параллелизмом (MPP).
31. Дешифратор, шифратор, триггерные схемы различных типов.
32. Платформы и архитектуры CPU.
33. Архитектуры с фиксированным набором устройств.
34. Типы данных, форматы файлов.
35. Счетчик, регистры хранения и сдвига.
36. Принципы вычислений в многоядерных системах.
37. Несовместимые аппаратные платформы.
38. Многопроцессорные вычислительные системы.
39. Кодирование символьной информации, код ASCII.
40. Таблицы истинности RS- триггера.
41. Принципы работы основных логических блоков системы, параллелизм и конвейеризация вычислений.
42. Таблицы истинности JK -триггера.
43. Классификация многомашинных вычислительных систем.
44. Конвейер команд.
45. Таблицы истинности T-триггера.
46. Чипсет - назначение и схема работы.
47. Структура процессора - регистры процессора,
48. Классы CISC, RISC, MiSC-процессора.
49. Шины PCI, AGP, PCIe-xpress и их характеристики.
50. Последовательные и параллельные порты.

51. Основные характеристики процессора, типы сокетов.
52. Классификация вычислительных систем в зависимости от числа потоков и данных.
53. Назначение и характеристика вычислительных систем.
54. Конвейер данных.
55. Архитектура ПК с периферийными устройствами
56. Системная плата - архитектура и основные разъемы.
57. Режимы работы процессора
58. Основные понятия реального и защищенного режимов работы процессора.
59. Конвейеризация вычислений.
60. Структура, основные характеристики КЭШ - памяти

Пакет экзаменатора

ПАКЕТ ЭКЗАМЕНАТОРА		
Задания теоретические проводятся в форме тестового контроля в оболочке Veraltest в центре тестирования. Задания практические проводятся в форме решения задач профессиональной направленности.		
Объекты оценки	Критерии оценки результата	Отметка о выполнении
Знания:		
виды информации и способы ее представления в ЭВМ;	Верное представление числовой информации используются 3 вида чисел: с фиксированной точкой, с плавающей точкой, двоично-десятичное представление.	
классификацию и типовые узлы вычислительной техники (ВТ);	Правильно выявляет общие принципы построения и классификацию современных вычислительных систем; приводит несколько примеры вычислительных платформ различных типов; грамотно выбирает тип вычислительной системы в соответствии с решаемой задачей.	
архитектуру электронно-вычислительных машин и вычислительных систем;	Верно называет и характеризует все блоки ВТ: устройство ввода, центральный процессор, запоминающие устройство, устройство вывода.	
назначение и принципы действия отдельных архитектурных конфигураций.	Правильно классифицирует и определяет типовые узлы средств вычислительной техники; представляет в полном объеме архитектуру электронно-вычислительных машин и вычислительных систем; грамотно различает назначение и принципы действия отдельных архитектурных конфигураций.	
Умения:		
выбирать рациональную конфигурацию оборудования в соответствии с решаемой задачей;	Правильно определяет устройства ЭВМ для успешной организации работы вычислительных систем.	

обеспечивать совместимость аппаратных и программных средств ВТ.	Верно различает аппаратные и программные средства вычислительной техники; грамотно контролирует совместимость аппаратных и программных средств вычислительной техники.	
ОК		
ОК 1 – ОК 10.	Правильно выполняет задания с профильной информацией по специальности подготовки.	

Список использованной литературы

Для реализации программы библиотечный фонд имеет печатные и/или электронные образовательные и информационные ресурсы, рекомендуемые для использования в образовательном процессе.

Основная литература:

1. Архитектура компьютера: учебное пособие. —: учебное пособие / Н.Б. Догадин. — 3-е издание. — Москва: Лаборатория знаний, 2015. — 274 с. — ISBN 978-5-9963-2638-9.
2. Архитектура ЭВМ: учеб. пособие / В.Д. Колдаев, С.А. Лупин. — М.: ИД «ФОРУМ» : ИНФРА-М, 2019. — 383 с. — (Среднее профессиональное образование). - Режим доступа: <http://znanium.com/catalog/product/1010475>;
3. Архитектура ЭВМ и вычислительных систем: учебник / Н.В. Максимов, Т.Л. Партыка, И.И. Попов. — 5-е изд., перераб. и доп. — М.: ФОРУМ: ИНФРА-М, 2017. — 511 с. — (Среднее профессиональное образование). - Режим доступа: <http://znanium.com/catalog/product/814513>.

Дополнительная литература:

4. Панфилов, И.В. Архитектура ЭВМ и информационных систем. Структурная организация [Электронный ресурс]: учебное пособие / И.В. Панфилов, А.М. Заяц. — Электрон. дан. — Санкт-Петербург: СПбГЛТУ, 2013. — 96 с. — Режим доступа: <https://ez.el.fa.ru:2055/book/58860>.