

МАШИННО-ОРИЕНТИРОВАННЫЕ ЯЗЫКИ (МОЯ)

Специальность (ЭВМбз)

“Вычислительные машины, комплексы, системы и
сети”

Назначение курса

- * Изучение архитектурных особенностей микропроцессоров.
- * Знание и понимание принципов работы микропроцессора на основе разработки и отладки программ на языке Ассемблера.
- * Без знания этого языка невозможно современное законченное компьютерное образование

Роль и значение дисциплины

1. Наиболее мощное и эффективное программное обеспечение написано на языке Ассемблер.
2. Программы, написанные на языке Ассемблер, требуют значительно меньшего объема памяти и времени выполнения.
3. Знание языка Ассемблера и результирующего машинного кода дает понимание архитектуры машины, что вряд ли обеспечивается при работе на языках высокого уровня.

ТРЕБОВАНИЯ К УРОВНЮ ОСВОЕНИЯ ДИСЦИПЛИНЫ

студент должен **Знать:**

- базовые понятия и определения;
- основные арифметические и логические команды машинно-ориентированного языка;
- команды организации ветвлений и циклов, процедур и макросов;
- прерывания, обработку прерывания от периферийных устройств.

ТРЕБОВАНИЯ К УРОВНЮ ОСВОЕНИЯ ДИСЦИПЛИНЫ:

- * студент должен:
 - знать принципы работы компьютера, его архитектуру и язык ассемблера как отражение и воплощение этих знаний;
 - уметь использовать методы структурного подхода при алгоритмизации и кодировании программы, провести тестирование программ,
 - уметь сочетать языковые средства различных уровней для разработки программного обеспечения микропроцессорных систем.

Машинные языки

- Все процессы в машине на самом низком, аппаратном уровне приводятся в действие только командами машинного языка.
- ◎ Машинная команда –это элементарная инструкция машине, выполняемая ею автоматически.
- ◎ Машинная команда состоит из двух частей:
операционной и адресной
- Q A
- ◎ Операционная часть команды это группа разрядов в команде, предназначенная для представления **кода операции** машины.

Машинные языки

⦿ Адресная часть команды это группа разрядов в команде, в которых записываются **адреса** ячеек памяти машины. Это адреса операндов, т.е. чисел, участвующих в операции.

● По количеству адресов команды делятся на:

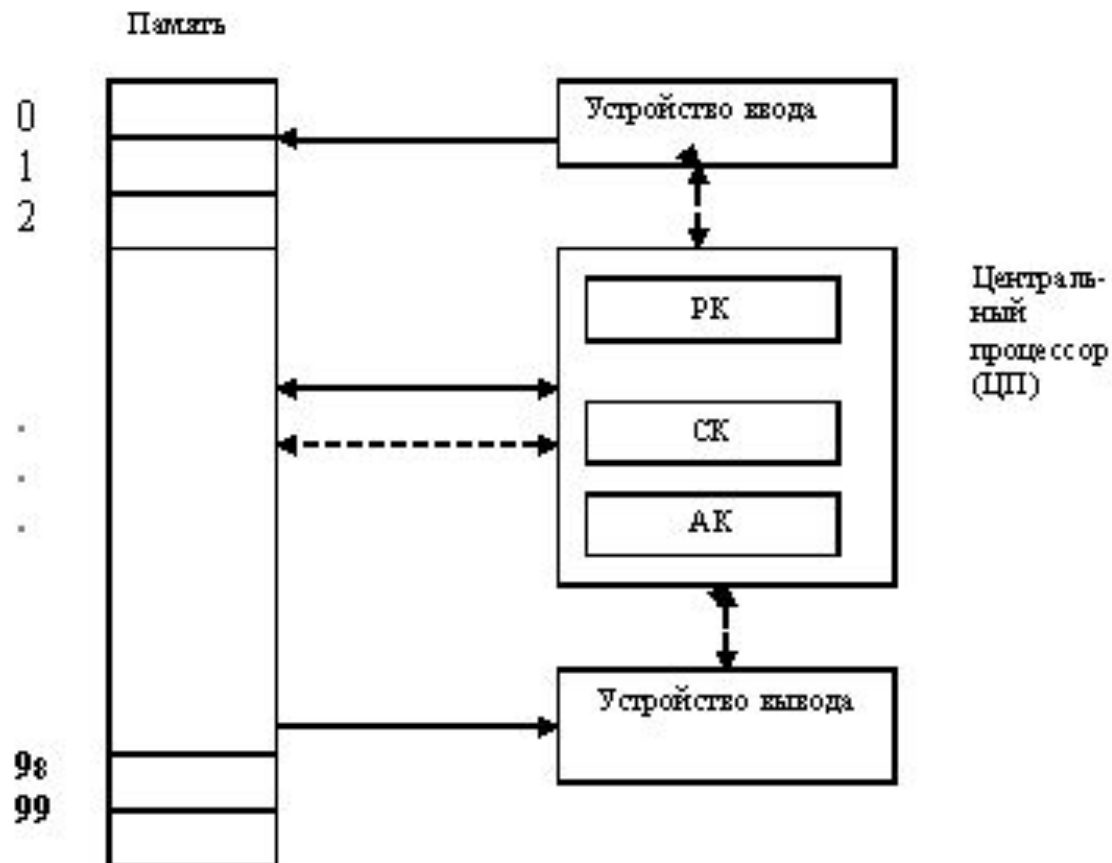
- **безадресные,**
- **одноадресные,**
- **двухадресные,**
- **трехадресные.**

Безадресная команда содержит только код операции, а информация для нее должна быть заранее помещена в определенные регистры машины.

Пример ЭВМ с одноадресными командами

- ◎ Это упрощенная машина под названием TOYCOMP (сокращенное от Toy Computer- игрушечный компьютер), фактически существует только в виде имитирующей программы, написанной на языке высокого уровня.

Архитектура ТОУСОМР



Система команд TOYCOMR

КОП	Выполняемые операции
00	Останов
01	Загрузка в аккумулятор (пересылка содержимого ячейки памяти в АК)
02	Запись в память ((пересылка содержимого АК в ячейку памяти)
03	Сложение
04	Вычитание
05	Умножение
06	Деление
07	Ввод (обеспечивает ввод числа с клавиатуры в ячейку памяти)
08	Вывод (обеспечивает вывод на дисплей числа из ячейки памяти)
09	Переход безусловный по адресу
10	Переход по адресу, если число в аккумуляторе больше нуля
11	Переход по адресу, если в аккумуляторе нуль

Программа вычисления $SUM=X+Y$

0	0799	Ввод числа X в ячейку с адресом 99
1	0798	Ввод числа Y в ячейку с адресом 98
2	0199	Загрузка X из ячейки 99 в аккумулятор
3	0398	Сложение Y-содержимого ячейки 98 с содержимым аккумулятора
4	0297	Пересылка содержимого аккумулятора в ячейку SUM
5	0899	Вывод X-содержимого ячейки с адресом 99
6	0898	Вывод Y -содержимого ячейки с адресом 98
7	0897	Вывод SUM- содержимого ячейки с адресом 97
8	0000	Останов (STOP)

Программирование в машинных командах имеет недостатки:

- * программист должен сам назначить адреса в памяти всем используемым в программе переменным и константам;
- * программистам трудно оперировать с кодами операций и адресами, потому что все они представлены в виде чисел;
- * в командах перехода используются абсолютные адреса, что сильно усложняет процесс внесения изменений в программу

Разработка языков программирования более высокого уровня, чем язык машинных команд

- ⦿ Они получили название - *машинно-ориентированные языки или языки символического кодирования (ЯСК)*.
- ⦿ Машинно-ориентированные языки – это языки, наборы операторов и изобразительные средства которых существенно зависят от особенностей ЭВМ (внутреннего языка, структуры памяти и т.д.).
- ⦿ Коды операций и адреса в командах задаются мнемоническими обозначениями.
- ⦿ Команды ЭВМ вместо истинных (физических) адресов содержат символические адреса.

Машинно-ориентированные языки

- **Ассемблером** называется системная программа, переводящая программу, написанную на машинно-ориентированном языке, в последовательность машинных команд.
- Машинно-ориентированные языки или языки символического кодирования также получили название **языков ассемблера**.
- Язык Ассемблера для каждого типа компьютера свой. Проблемы, связанные с аппаратурой, невозможно решить без знания Ассемблера.

TOYCODE- язык ассемблера TOYCOMP

КОП в числовой форме	КОП в форме символического имени	Название
00	STOP	Останов
01	LD	Загрузка в аккумулятор
02	STO	Запись в память
03	ADD	Сложение
04	SUB	Вычитание
05	MPY	Умножение
06	DIV	Деление нацело
07	IN	Ввод
08	OUT	Вывод
09	B	Переход безусловный
10	BGTR	Переход, если больше нуля
11	BZ	Переход, если равно нулю

Программа вычисления $SUM=X+Y$ на TOVCODE

0	IN x	Ввод числа X
1	IN Y	Ввод числа Y
2	LD X	Загрузка X в аккумулятор
3	ADD Y	Сложение Y с содержимым аккумулятора
4	StO SUM	Пересылка содержимого аккумулятора в ячейку SUM
5	OUT X	Вывод X
6	OUT Y	Вывод Y
7	OUT SUM	Вывод SUM
8	STOP	Останов (STOP)

Разработка программ на ассемблере состоит из следующих шагов:

- * 1. Создание исходного текста программы с использованием текстового редактора.
- * 2. Создание объектного модуля. Трансляция программы
- * 3. Создание загрузочного модуля. Компоновка программы
- * 4. Отладка программы с использованием программ-отладчиков

ОБЪЕМ ДИСЦИПЛИНЫ И ВИДЫ УЧЕБНОЙ РАБОТЫ

ЭВМбз

144 часа

● Аудиторные занятия	14
● Лекции	6/2/4
● Лабораторные занятия	8/-/8
● Самостоятельная работа	121/34/87
Подготовка к лабораторным работам	6/18
Подготовка к экзамену	-/ 8
Курсовое проектирование	-/30
Самостоятельное изучение разделов дисциплины	28/31

Основные разделы курса

- * Введение, цели и задачи дисциплины.
Программная модель и основы программирования микропроцессоров
- * Язык Ассемблера МП 8080
- * Прерывания. Команды управления МП.
- * Работа с математическим сопроцессором.
- * Работа с современными процессорами в защищенном режиме.

Лабораторные работы

- * Учебная машина TOYCOMP.
 - * №1 Составление программ на языке машинных команд TOYCOMP
 - * №2 Ассемблер TOYCOMP. Составление программ на TOYCODE.
- * МП I8086
 - * №3. Разветвляющийся процесс и простые циклы.
 - * №4 Обработка массивов..

Курсовой проект

- * Курсовой проект предназначен для закрепления знаний современной технологии разработки алгоритмов, написания и отладки программ на машинно-ориентированном языке.
- * Тема: Модульное программирование на Ассемблере

Методические материалы

- * конспект лекций;
- * методические указания по выполнению лабораторных занятий;
- * методические указания по курсовому проектированию.
- * презентации.

Формы контроля

- * Защита лабораторных работ.
- * Тестирование.
- * Виды итогового контроля:
 - * Защита курсового проекта.
 - * Экзамен.

Экзамен

- * Проводится устно по билетам, включающим теоретические вопросы и задачи.
- * Для допуска к экзамену необходимо выполнить лабораторный практикум.

ПРОГРАММНЫЕ СРЕДСТВА

- * TASM.exe (Borland)-транслятор
- * TLINK.exe(Borland)- компоновщик
- * TD.exe (Borland) -отладчик
- * EMU8086- эмулятор
- * MS Office (Microsoft Corporation)
- * Эмулятор TOYCOMP (разработка ст-та ЭВМ-07-1 Дубинина Дмитрия).
- * Транслятор TOYCODE (разработка ст-та ЭВМ-07-1 Бреусова К.)

ОСНОВНАЯ ЛИТЕРАТУРА

1. П. Абель Язык ассемблера для IBM PC и программирования. М. «Высшая школа», 2003.- 447с.
2. В. Юров ASSEMBLER УЧЕБНИК: Питер, 2004. - 624с.
3. Пильщиков В.Н. Программирование на языке ассемблера IBM PC. – М.: «Диалог-МИФИ», 1999. – 288 с.
4. Пирогов В. Ассемблер в примерах. – Спб: БХВ, 2005. – 416 с.
- 5.

ДОПОЛНИТЕЛЬНАЯ ЛИТЕРАТУРА

1. Юров В. Assembler : Спец. справ. / В. Юров. - СПб. и др.: Питер, 2000. - 489с.: ил.
2. С. Зубков –Assembler для DOS,Windows и Unix: ДМК Пресс,2000- 400с.
3. Голубь Н.Г. Искусство программирования на Ассемблере. Лекции и упражнения. :ДиаСофтЮП, 2002-460с.