

Подробная программа курса "Архитектура ЭВМ и язык Ассемблера"

1. Архитектура ЭВМ

Понятие об архитектуре ЭВМ. Уровни рассмотрения архитектуры. Типовая схема ЭВМ, принципы Фон-Неймана. Оперативная память: ячейка, адрес, бит, машинное слово. Характеристики и единицы измерения памяти. Линейность и однородность памяти. Команды и данные, их неразличимость. Принцип хранимой программы. Двоичное представление данных.

Центральный процессор ЭВМ: устройство управления и арифметико-логическое устройство. Регистры. Устройства ввода/вывода.

Конкретизация машины Фон-Неймана в виде учебных ЭВМ. Архитектура и способы выполнения команд для учебных ЭВМ. Учебная трёхадресная ЭВМ, способы программирования.

Адресность ЭВМ, схема выполнения команд в трёх, двух, одно и нуль адресной ЭВМ. Понятие такта работы. Схема выполнения команд в ЭВМ с адресуемыми регистрами.

Разнообразие форматов команд и данных. Специфика применения ЭВМ в различных областях. (*) Понятие о семействе ЭВМ и моделях семейства.

Структура оперативной памяти: байт, слово (2 байта), двойное слово (4 байта), четверное слово (8 байт). Структуры данных: символы, целые и вещественные числа, строки (массивы) символов и целых чисел, битовые данные. Представление целых чисел: прямой, обратный и дополнительный код. Знаковое и беззнаковое расширение целого числа. Перевернутое представление чисел в памяти.

(*) Представление вещественных чисел, нормализованные числа, значения $\pm\infty$ и NaN.

Структура регистровой памяти, адресация регистров. Регистр флагов. Принципы адресации данных. Понятие об исполнительном и логическом адресе. Базирование и индексирование. Сегментные регистры. Deskрипторы сегментов, локальная и глобальная таблица deskрипторов. Основные форматы команд.

Понятие прерывания, процедуры-обработчики прерываний. Маскирование прерываний, таблица deskрипторов прерываний. Аппаратная реакция на сигнал прерывания. Обработка неприоритетных и приоритетных прерываний, контекст задачи, понятие о переключении задач. Программная реакция на прерывание, процедура обработки прерывания, минимальная (с закрытыми прерываниями) и полная обработка прерывания.

Понятие об архитектурных особенностях современных ЭВМ, которые позволяют им работать в мультипрограммном режиме: система прерываний, защита памяти, аппарат привилегированных команд, параллельная обработка данных, таймер.

Связь центрального процессора с внешними (периферийными) устройствами. Архитектура ЭВМ с общей шиной. Порты ввода/вывода. Архитектура ЭВМ с каналами ввода/вывода (периферийными процессорами, процессорами ввода/вывода). Параллельная работа центрального процессора и каналов. Параллелизм в выполнении команд, расслоение памяти. Принцип локальности. Понятие о памяти типа кэш, схема её работы. Понятие конвейера, спекулятивное выполнение команд, предсказание ветвлений. (*) Классификация архитектур ЭВМ по Флинну.

2. Язык Ассемблера

Назначение, достоинства и недостатки языка Ассемблера. Основные понятия языка, вид программы на Ассемблере. Классификация предложений на языке Ассемблера: машинные команды, резервирование памяти, команды транслятору (директивы), макрокоманды, комментарии. Структура предложения на Ассемблере: поля метки, кода операции, операндов, комментария.

Запись команд на Ассемблере, абсолютные и перемещаемые адресные выражения. Смещение, способы базирования и индексирования. Счётчик адреса.

Имена в языке Ассемблера, служебные и пользовательские имена, тип имени, оператор **type**. Директивы **equ** и **=**. Константные адресные выражения, счётчик адреса (счётчик размещения). Команды **mov** и **lea**, оператор **ptr**.

Арифметические действия с целыми числами, программирование вычислений по формулам.

Порядок выполнения команд в программе, условные и безусловные переходы, типы переходов. Программирование циклов. (*) Понятие о высокоуровневом программировании на Ассемблере, директивы **.if**, **.while** и **.repeat**.

Описание массивов на Ассемблере. Работа с массивами, переменные с индексами. Строковые команды. Описание структур и упакованных записей на Ассемблере, задания их начальных значений, доступ к полям. (*) Объединения на Ассемблере.

Обработка битовых данных, логические команды и сдвиги. Вычисление булевских выражений. Работа с упакованными данными и множествами.

Общая структура программы на Ассемблере, понятие модуля. Описание программных секций (сегментов). Директива **include**.

Стек, указатель вершины стека, доступ к элементам стека. Команды **push**, **pop**, **call** и **ret**. Описание и вызов процедуры, возврат из процедуры, способы передачи параметров. Стандартные соглашения о связях. Рекурсивные процедуры.

3. Макросредства языка

Макроассемблер, понятие макрогенерации. Макроопределение, макрокоманда (макровызов), макрорасширение. Библиотека макроопределений. Макрокоманды ввода/вывода.

Описание макроопределений. Способы передачи параметров, значения параметров по умолчанию, переменное число параметров, ключевые слова **Req** и **VarArg**. Сравнение подпрограмм и макроопределений. Условное ассемблирование, команды условного ассемблирования, задание логических условий. Блоки повторений **for**, **forc**, **while** и **repeat**. Макрооператоры $\langle \rangle$, **!**, **%** и **&**. Локальные объекты в макроопределениях, директива **local**.

4. Элементы системы программирования

Понятие системы программирования и её состав, многоязыковые системы программирования.

Модульное программирование, понятие программного модуля. Библиотеки модулей. Исходные, объектные, загрузочные и абсолютные модули. Характеристики выполняемых модулей: перемещаемость, повторное и параллельное выполнение.

Статические и динамические связи между модулями по управлению и по данным. Статическая связь по данным: входные и внешние имена, директивы **public** и **extrn**. Доступ к внешним именам. Вызов внешней подпрограммы. Примеры многомодульных программ.

(*) Компилятор Ассемблера, его основные функции и схема работы. Лексический и синтаксический анализ, таблицы Ассемблера.

Редактор внешних связей, его назначение и схема работы. Статический загрузчик, его назначение и схема работы. Понятие о динамическом связывании и динамической загрузке, DLL библиотеки.

Литература:

1. Кип Р.Ирвин. Язык ассемблера для процессоров Intel, 4-е изд., 2005, 912 с.
2. Магда Ю. Ассемблер для процессоров Intel Pentium, Питер, 2006, 416 с.
3. Пирогов В.Ю. Ассемблер для Windows, 4-е изд., БХВ-Петербург, 2007, 896 с.
4. Пильщиков В.В. Программирование на языке Ассемблера IBM PC. – М.: Диалог-МИФИ, 1994.
5. Ю-Чжень Лю, Гибсон Г. Микропроцессоры семейства 8086/8088. – М.: Радио и связь, 1987.
6. Королёв Л.Н. Структура ЭВМ и их математическое обеспечение. – М.: Наука, 1978.
7. Таненбаум Э. Архитектура компьютера. 5-е изд. – Питер, 2007.
8. Бек Л. Введение в системное программирование. – М.: Мир, 1988.
9. Королёв Л.Н. Микропроцессоры, микро- и мини-ЭВМ. – М.: Изд-во МГУ, 1989.
10. Королёв Л.Н. Архитектура процессоров электронных вычислительных машин. – М.: Издательский отдел факультета ВМК МГУ им. М.В. Ломоносова, 2003.
11. Баула В.Г., Томилин А.Н., Волканов Д.Ю. Архитектура ЭВМ и операционные среды. – М.: Академия, 2011,2012.

Сайт поддержки курса: <http://arch32.cs.msu.ru>

Литература к курсу: <http://arch32.cs.msu.ru/semestr2>

Ассемблер: [http://arch32.cs.msu.ru/semestr2/masm 6.14.zip](http://arch32.cs.msu.ru/semestr2/masm%206.14.zip)