

Лабораторная работа

Использование инструкций ассемблера в DEBUG. Прерывания

Оборудование: работа рассчитана на использование MS Windows XP или другой версии Windows с поддержкой 16-разрядного виртуального режима

Ход работы

Подготовка к работе

Запустить в эмуляторе командной строки Windows отладчик DEBUG: **Пуск - > Выполнить - > debug.exe**

Использование DEBUG для ввода инструкций ассемблера

DEBUG можно использовать для ввода инструкций на языке ассемблера.

Для того чтобы начать ввод инструкций нужно ввести команду «А» указав начальный адрес для инструкций в сегмента кода со смещением:

A 100

Ввод инструкций осуществляется далее подряд. После каждой инструкции нужно нажать Enter

Для окончания ввода инструкций нужно нажать Enter еще раз.

Для вывода на экран символьных инструкций служит команда «U». Синтаксис команды допускает указание адресов первой и последней инструкции.

U 100,107

Задание №1 Ввод инструкций ассемблера

Введите команды ассемблера начиная с смещения 100

MOV CL, 42

MOV DL, 2A

ADD CL, DL

JMP 100

Просмотрите введенный код и поместите скриншот в отчет.

Использование инструкции INT

Инструкция **INT** генерирует программное прерывание. Она осуществляет выход из программы, вызов подпрограммы DOS или BIOS обрабатывающей определенное прерывание и возвращение в программу.

INT <вид прерывания>

Некоторые виды операций INT требуют наличия кода функции в регистре AH, чтобы определить какое действие следует выполнить.

Для выполнения инструкции INT можно использовать команду P для выполнения всей вызываемой подпрограммы сразу.

Задание №2 Получение текущей даты.

Инструкция для получения текущей даты INT 21H с кодом функции Fn 2AH

Введите программу

Инструкция	Комментарий
MOV AH, 2A	Подготовка кода функции
INT 21	Вызов прерывания 21 для функции 2A
JMP 100	Возврат в начало программы

Таблица 1: код получения даты и времени

Выполните последовательность команд для проверки введенной программы:

Команда	Комментарий
-R IP	Обращение к регистру IP – указателю на инструкцию

Лабораторные работы. Микропроцессорные системы.

100	Установка указателя в начало программы
-R	Просмотр состояния регистров перед началом выполнения программы
-T	Выполнение первой инструкции — MOV
-P	Создание программного прерывания — INT и выполнение его обработки
-T	Возврат в начало программы — JMP

Таблица 2: проверка программы

Поместите скриншот проверки в отчет.

Регистры будут содержать в шестнадцатеричной форме следующую информацию:

Регистр ¹	Информация
AL	День недели где 0 - воскресенье
CH	Год
DH	Месяц
DL	День месяца

Таблица 3: Расшифровка регистров после выполнения INT 21 Fn 2A

Поместите скриншот содержимого регистров в отчет.

Расшифруйте дату и запишите процесс расшифровки в отчет.

Задание №3 Получение текущего времени.

Инструкция для получения текущей даты и времени INT 21H Fn 2CH

Результат обработки такого прерывания помещается в регистры следующим образом:

Регистр	Информация
CH	Часы
CL	Минуты
DH	Секунды
DL	Сотые доли секунд

Таблица 4: Расшифровка регистров после выполнения INT 21 Fn 2C

Введите последовательность инструкций ассемблера позволяющих получить текущее время.

Поместите в отчет скриншот набранной программы и результатов ее работы (регистров).

Расшифруйте время и запишите процесс расшифровки в отчет.

Задание №4 Вывод на экран

Для вывода на экран можно использовать функцию инструкции INT 21 Fn 09H. Для этого нужно внести выводимый текст в байты памяти. И указать в DX смещение в котором определены эти байты.

Инструкция	Комментарий
MOV AH, 09	Подготовка кода функции
MOV DX, 109	Смещение в котором определены выводимые байты
INT 21	Вызов прерывания 21 для функции 2A
JMP 100	Возврат в начало программы
DB 'Hello All!','\$'	Определение строки для вывода. '\$' - окончание строки

¹Помните, что AL, DH, DL и т. п. восьми битные регистры могут быть частью шестнадцати битных регистров. Например регистр AL может являться частью делимого регистра AX

Лабораторные работы. Микропроцессорные системы.

Таблица 5: код вывода на экран

Выполните программу и поместит скриншот в отчет.

Задание №5 Ввод с клавиатуры

Для ввода с клавиатуры можно использовать функцию инструкции INT 16 Fn 10H. Код введенной клавиши помещается в AL. При вызове INT 21 для этой функции программа перейдет к ожиданию нажатия клавиши.

Составьте и выполните программу и поместите скриншот в отчет.

Задание №6 Самостоятельная работа

Пользуясь указанной ниже информацией, изученным материалом, справочниками и сетью Интернет² составьте программы для отображения/изменения данных для доступа к которым используются указанные функции. Предусмотрите в программе очистку регистров в которых планируется получить данные. Выполните эти программы и поместите скриншоты в отчет.

Расшифруйте и полученные данные и поместит их в отчет

Вариант 1

INT 12H: размер используемой памяти

Вариант 2

INT 21H Fn 30H³ номер версии DOS

Вариант 3

INT 21H Fn 36H Получение информации о свободной памяти диска.

²<http://www.codenet.ru/progr/dos/>

³Например <http://www.htl-steyr.ac.at/~morg/pcinfo/hardware/interrupts/inte2zlc.htm>

Защита работы:

Для успешной защиты данной работы необходимо знать:

- Что такое бит и байт
- Структуру байта с учетом бита четности. Правило контроля четности
- Как нумеруются биты в слове. Как информация хранящаяся в битах может быть преобразована в понятный человеку формат
- Как слово передается из регистра процессора в память
- Что такое абсолютный адрес операнда и адрес в системе сегмент-смещение
- Для чего применяют сегментирование памяти, как работает сегментирования память процессора 8086
- Как выполняется адресация байт в слове
- Что такое «внутренние регистры процессора»
- Для чего служат регистры AX, BX, DX, CX?
- Для чего служат регистры CS, DS, ES, SS?
- Для чего служат регистры SP, BP, SI, DI, IP?
- Для сего служит регистр FLAGS?
- Каковы роли битов в регистре FLAGS?
- Знать команды пересылки данных и команды арифметических операций