

В. П. Семеренко, В. А. Каплун

# Системне програмування

## ***мовою Асемблера***

частина 2

Міністерство освіти і науки України  
Вінницький національний технічний університет

## **Системне програмування мовою Асемблера**

Затверджено Вченою радою Вінницького національного технічного університету як лабораторний практикум (частина 2) для студентів бакалаврських напрямків 6.0915 - “Комп’ютерна інженерія” та 1601 – “Інформаційна безпека” всіх спеціальностей.

Протокол №            від                            2004 р.

УДК 681.3.06.(075)

С31

Рецензенти:

*С.В. Юхимчук*, доктор технічних наук, професор  
*А.М. Петух*, доктор технічних наук, професор  
*А.І. Кузьмичов*, кандидат технічних наук, доцент

Рекомендовано до видання Вченою радою Вінницького національного технічного університету Міністерства освіти і науки України

**Семеренко В. П., Каплун В. А.**

С31 **Системне програмування мовою Асемблера.** Лабораторний практикум. Частина 2. - Вінниця: ВНТУ, 2004. - 93 с.

В практикумі розглянута методика складання програм мовою Асемблера для мікропроцесорів фірми Intel підвищеної складності. Вивчаються функції транслятора з мови Асемблера та компонувача, аналізується структура їх лістингів. Велика увага приділена вивченню принципів програмного керування дисплеєм, математичним співпроцесором та пристроєм "миша". Розглянуті також макрозасоби в мові Асемблера. Практикум призначений для студентів напрямів підготовки 0915 - "Комп'ютерна інженерія" та 1601 - "Інформаційна безпека" для вивчення дисциплін "Системне програмування", "Системне програмне забезпечення", "Операційні системи" а також може бути рекомендований студентам інших спеціальностей, пов'язаних з вивченням сучасного програмного забезпечення.

УДК 681.3.06.(075)

© В.П. Семеренко, В.А. Каплун 2004

## ЗМІСТ

Вступ.....	5
Лабораторна робота №5- Макрозасоби мови Асемблера.....	6
2.5.1 Призначення макрозасобів.....	6
2.5.2 Макровизначення і макрокоманди.....	6
2.5.3 Приклад програми з використанням макровизначень.....	7
2.5.4 Проблема міток у макровизначеннях.....	9
2.5.5 Циклічні макроси (директиви повторення).....	9
2.5.6 Умовні макроси (умовні директиви).....	11
2.5.7 Використання макробібліотек.....	12
2.5.8 Директиви управління лістингом програми .....	13
Порядок виконання роботи.....	13
Лабораторна робота №6 – Транслятор з мови Асемблера.....	13
2.6.1 Призначення та основні функції транслятора з мови Асемблер.....	13
2.6.2 Види трансляторів з мови Асемблера .....	14
2.6.3 Запуск на виконання транслятора TASM.....	14
2.6.4 Структура лістинга програми.....	16
2.6.5 Структура об'єктної програми.....	17
2.6.6 Приклад ручного асемблювання програми.....	20
2.6.7 Таблиця символічних імен.....	21
2.6.8 Таблиці перехресних посилань.....	23
2.6.9 Об'єктні бібліотеки.....	24
Порядок виконання роботи.....	24
Лабораторна робота №7 – Компонувачі.....	25
2.7.1 Призначення та основні функції компонентувача (редактора зв'язків)...25	
2.7.2 Типи компонентувачів .....	26
2.7.3 Запуск компонентувача на виконання.....	27
2.7.4 Карта завантаження.....	28
2.7.5 Директиви зв'язку модулів та сегментів.....	29
2.7.6 Приклад розбиття програми на зовнішні процедури.....	31
Порядок виконання роботи.....	34
Лабораторна робота №8 – Математичний співпроцесор.....	34
2.8.1 Архітектура математичного співпроцесора.....	34
2.8.2 Формати даних співпроцесора 8087.....	36
2.8.3 Система команд співпроцесора 8087.....	38
2.8.4 Приклади програм із використанням математичного співпроцесора.....	41
Порядок виконання роботи.....	43
Лабораторна робота №9 – Керування дисплеєм.....	43
2.9.1 Технічні особливості дисплеїв.....	43
2.9.2 Текстовий режим роботи дисплея.....	44
2.9.3 Приклад програми прямого доступу до відеопам'яті.....	47
2.9.4 Графічний режим роботи дисплеїв.....	49
2.9.5 Керування дисплеєм за допомогою функцій BIOS.....	52

2.9.6 Основні функції BIOS для керування дисплеєм.....	53
2.9.7 Приклади програми для виведення зображення в текстовому режимі з використанням функцій BIOS.....	57
2.9.8 Приклади програми для виведення зображення в графічному режимі з використанням функцій BIOS.....	59
Порядок виконання роботи.....	61
Лабораторна робота №10 – Робота з файлами.....	61
2.10.1 Класи функцій для роботи з файлами.....	61
2.10.2 Створення файлів.....	62
2.10.3 Відкриття файла.....	63
2.10.4 Закриття файла.....	64
2.10.5 Читання даних із файла або пристрою.....	64
2.10.6 Запис даних у файл або на пристрій.....	65
2.10.7 Вилучення файла.....	65
2.10.8 Приклад програми роботи з файлом.....	66
2.10.9 Методи доступу до файлів.....	67
2.10.10 Алгоритм запису даних у послідовний файл.....	67
2.10.11 Алгоритм читання даних із послідовного файла.....	68
2.10.12 Операції читання та запису даних у файлах прямого доступу.....	68
2.10.13 Приклад програми читання даних із послідовного файла.....	68
2.10.14 Робота з атрибутами файлів.....	70
2.10.15 Робота з дисками, каталогами та організація пошуку файлів.....	73
2.10.16 Робота з файлами в MS DOS у випадку довгих імен.....	77
Порядок виконання роботи.....	78
Лабораторна робота №11 – Керування пристроєм “миша”.....	80
2.11.1 Ініціалізація та визначення поточного стану драйвера пристрою “миші”.....	80
2.11.2 Визначення типу і форми курсора “миші” у текстовому режимі....	81
2.11.3 Визначення типу і форми курсора “миші” у графічному режимі....	83
2.11.4 Установлення чутливості курсора “миші”.....	85
2.11.5 Керування станом курсора “миші”.....	87
2.11.6 Читання позиції курсора і стану кнопок “миші”.....	87
2.11.7 Позиціонування курсора “миші” і прикладною програмою.....	89
2.11.8 Приклад програми для керування пристроєм “миша” в текстовому режимі роботи відеоадаптера.....	90
Порядок виконання роботи.....	91
Література.....	92

## Вступ

Друга частина лабораторного практикуму орієнтована на продовження практичного вивчення програмування мовою Асемблера для мікропроцесорів фірми Intel.

В першій лабораторній роботі розглядаються макроси в мові Асемблера, які є особливим різновидом підпрограм, що дозволяють скоротити обсяг асемблерних програм.

Наступні дві роботи присвячені вивченню двох основних системних програм обробки – транслятора Turbo Assembler (TASM) і компонувача TLINK. Знання особливостей їх функціонування дозволяє створювати ефективні багатомодульні програми і здійснювати їх швидке налагоджування.

Опанувавши програмування математичного співпроцесора можна використовувати арифметичні операції зі всіма типами даних, включаючи числа із плаваючою комою. В результаті значно розширюється сфера застосування Асемблера.

Дві лабораторні роботи пов'язані із програмним керуванням двох важливих периферійних пристроїв: дисплея та “миші”. Вивчивши графіку на Асемблері в текстовому та графічному режимах роботи дисплея можна легко формувати графічні зображення будь-якої складності з максимально можливою швидкодією. Це дасть змогу створювати програми з динамічними зображеннями, які часто використовуються в різноманітних ігрових та навчальних програмах. Використання в таких програмах пристрою “миша” є обов'язковим.

Велику користь програмісту надасть також вміння програмувати роботу з файлами в різних режимах.

Практично засвоївши лабораторні роботи, які розглянуті в посібнику, можна створювати достатньо складні асемблерні програми на професійному рівні.

## Література

1. Григорьев В. Л. Программирование однокристальных микропроцессоров. – М.: Энергоатомиздат, 1987.
2. Использование Turbo Assembler при разработке программ. – Киев, "Диалектика", 1994. – 288 с.
3. Абель П. Язык Ассемблера для IBM PC и программирования/ Пер. с англ. – М.: Высш. школа, 1992 – 447 с.
4. Финогенов К.Г. Основы языка ассемблера. – М.: Радио и связь, 1999. – 288 с.
5. Юров В. Assembler: Учебник. – СПб и др.: Питер, 2000. – 622 с.