

Міністерство освіти і науки України
Національний університет «Запорізька політехніка»

Кафедра програмних засобів

ЗВІТ

Дисципліна «Операційні системи»

Робота №1

Тема «Функції операційної системи MS-DOS: визначення версії ОС»

Виконав варіант 19

Студент КНТ-122

Онищенко О. А.

Прийняли

Викладач

Качан О. І.

2024

Основи програмування мовою Асемблер	3
Робота простих функцій вводу з клавіатури та виводу тексту	4
Програма Асемблеру для визначення версії MS DOS	4
Склад картки пам'яті – файлу типу MAP	5
Робота функції 3306h INT 21h	5
Структура PSP	5
Структура та образ пам'яті програми EXE	6
Структура програми COM	7
Використання утиліти MS DOS EXE2BIN	8
Поняття операційної системи	9
Структура ОС MS DOS	9
Завантаження операційної системи	11
<i>Запуск операційної системи MS DOS</i>	11

ОСНОВИ ПРОГРАМУВАННЯ МОВОЮ АСЕМБЛЕР

Асемблер – низькорівнева мова. Чим вижче рівень мови програмування, тим більший рівень абстракції. Абстракція – наскільки багато функцій мова «ховає» або виконує за лаштунками без відома програміста. Асемблер вважається низькорівневою мовою бо його інструкції майже точно відповідають вихідному машинному коду, який розуміє комп'ютер.

Оскільки команди мовою асемблер керують процесором, корисно розуміти як комп'ютер працює зсередини.

Запис чисел:

- Шістнадцяткові: суфікс H
- Двійкові: суфікс B
- Десяткові: суфікс D або без суфіксу

При написанні шістнадцяткових констант потрібно дописати 0 на початку аби система не сприйняла число як змінну.

Арифметичні оператори за ступенем прецеденту:

- () дужки
- +, - унарний плюс та мінус
- *, / множення та ділення
- MOD модуль
- +, - додавання та віднімання

Запис символів відбувається беручи їх до лапок. Можна використовувати як одинарні ('a'), так і подвійні ("T").

Рядки записуються так само у лапках: "32", 'ІСУС ГОСПОДЬ'.

Асемблер має кілька зарезервованих слів:

- Інструкції: MOV, ADD, MUL
- Атрибути: BYTE, WORD

[Джерело Українською](#), [Джерело Англійською](#)

РОБОТА ПРОСТИХ ФУНКЦІЙ ВВОДУ З КЛАВІАТУРИ ТА ВИВОДУ ТЕКСТУ

Для введення числа у консоль потрібен наступний код:

```
MOV AH,1  
INT 21H
```

Для виведення символу до консолі потрібен наступний код:

```
MOV AH,2  
MOV DL, 'T'  
INT 21H
```

[Джерело відповіді](#)

ПРОГРАМА АСЕМБЛЕРУ ДЛЯ ВИЗНАЧЕННЯ ВЕРСІЇ MS DOS

Програмний код мовою Assembly:

```
.model small  
.stack 100h  
.data  
vns db 'DOS Version $'  
  
.code  
main proc  
  
lea dx,vns  
mov ah,9  
int 21h  
  
mov ah,30h  
int 21h  
  
main endp  
end main
```

СКЛАД КАРТКИ ПАМ'ЯТІ – ФАЙЛУ ТИПУ MAP

Структура файлу MAP складається з таких елементів:

1. Inputs: змінні для вхідних даних асемблювання мапи
2. Outputs: змінні з вихідними даними асемблювання мапи
3. Action: масив дій які треба виконати по порядку. Кожна дія має директиву set або create – визначає вихідну змінну.
4. Options: вибір опцій, як от «Включати порожні елементи XML» чи «Обробка порожніх масивів JSON»

[Джерело відповіді](#)

РОБОТА ФУНКЦІЇ 3306H INT 21H

Призначення для отримання версії DOS. Параметри:

- Вхід: ah – 30h
- Вихід: ah – номер версії, більша версія у нижньому байті; bl:cx – серійний номер користувача; bh – номер OEM, або прапорці
- Помилка: al – 00h

[Джерело відповіді](#)

СТРУКТУРА PSP

Program Segment Prefix або Префікс програмного сегмента призначений для збереження поточного вигляд програми. Має наступну структуру:

Зсув	Розмір
00h-01h	2 байти (код)
02h-03h	2 байти (слово)
04h	байт

05h-09h	5 байтів (код)
0Ah-0Dh	4 байти (дслово)
0Eh-11h	дслово
12h-15h	дслово
16h-17h	слово
18h-2Bh	20 байт
2Ch-2Dh	слово
2Eh-31h	дслово
32h-33h	слово
34h-37h	дслово
38h-3Bh	дслово
3Ch-3Fh	4 байти
40h-41h	слово
42h-4Fh	14 байт
50h-52h	3 байти (код)
53h-54h	2 байти
55h-5Bh	7 байт
5Ch-6Bh	16 байт
6Ch-7Fh	20 байт
80h	1 байт
81h-FFh	127 байт

[Джерело відповіді](#)

СТРУКТУРА ТА ОБРАЗ ПАМ'ЯТІ ПРОГРАМИ EXE

Файле EXE складається з кількох частин:

- Заголовок MS DOS: інформація про переміщення, довжиною 64 байти, належить до системи MS DOS. Призначення – показувати помилки
- Заголовок PE: завантажувач ОС бере інформацію з цього заголовку. Каже куди треба завантажити програму у пам'ять
- Секції: різні параметри файлу - .text, .code, .tests, .rdata, .data, .bss, .rsrc, .debug, .idata, .edata, .reloc

Образ пам'яті програми EXE виглядає наступним чином:

Offset (h)	00	01	02	03	04	05	06	07	08	09	0A	0B	0C	0D	0E	0F	Decoded text
00000000	4D	5A	90	00	03	00	00	00	04	00	00	00	FF	FF	00	00	MZ.....ÿÿ..
00000010	B8	00	00	00	00	00	00	00	40	00	00	00	00	00	00	00@.....
00000020	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00
00000030	00	00	00	00	00	00	00	00	00	00	00	00	00	80	00	00€...
00000040	0E	1F	BA	0E	00	B4	09	CD	21	B8	01	4C	CD	21	54	68	..°...'.í!..Lí!Th
00000050	69	73	20	70	72	6F	67	72	61	6D	20	63	61	6E	6E	6F	is program canno
00000060	74	20	62	65	20	72	75	6E	20	69	6E	20	44	4F	53	20	t be run in DOS
00000070	6D	6F	64	65	2E	0D	0D	0A	24	00	00	00	00	00	00	00	mode....\$.....
00000080	50	45	00	00	4C	01	03	00	26	7B	C8	82	00	00	00	00	PE..L...&{È,....
00000090	00	00	00	00	E0	00	22	00	0B	01	30	00	00	1E	00	00à."...0.....
000000A0	00	08	00	00	00	00	00	00	32	3C	00	00	00	20	00	002<... ..
000000B0	00	40	00	00	00	00	40	00	00	20	00	00	00	02	00	00	.@....@..

Рисунок 1.1 – Образ пам'яті файлу EXE

[Джерело](#)

СТРУКТУРА ПРОГРАМИ COM

COM файли містять машинний код: без заголовків, таблиць переадресації, тощо. Максимальний об'єм такого файлу 64 КБ.

Завантаження файлу COM відбувається так:

1. Відвести максимально можливий обсяг вільної пам'яті
2. Збудувати PSP на початку блоку пам'яті
3. Прочитати COM файл з префіксо, передати керування на перший байт програми

Структура COM файлу виглядає так:

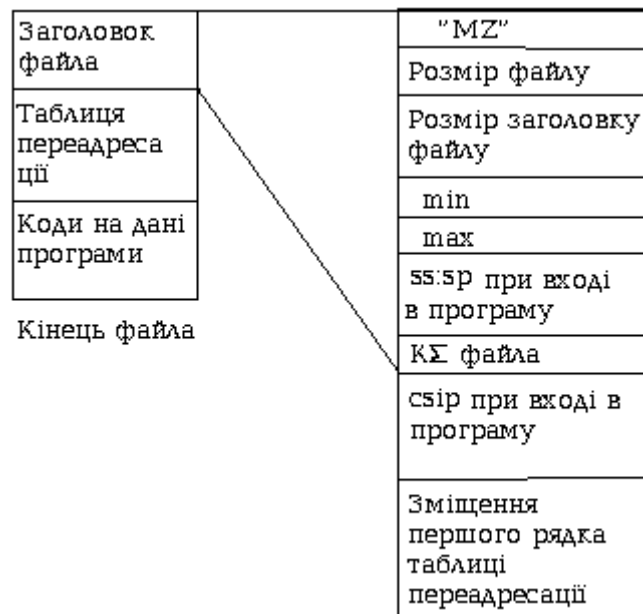


Рисунок 1.2 – Структура COM файлу

Пояснення:

- ss: сегмент стеку
- sp: покажчик стеку
- cs: сегменту коду
- ip: покажчик команд
- min: мінімальна додаткова пам'ять для файлу
- max: максимальна додаткова пам'ять для файлу

[Джерело](#)

ВИКОРИСТАННЯ УТИЛІТИ MS DOS EXE2BIN

Утиліта EXE2BIN призначена для конвертування файлів з формату EXE у бінарний. Це зменшує розміри файлів та пришвидшує їх завантаження.

Синтаксис команди такий:

EXE2BIN [шлях] назва_файлу [шлях] назва_файлу

Перша назва файлу це початковий файл. Якщо ввести без розширення, за замовчуванням вважатиме що файл EXE.

Друга назва файлу це новий файл, який створить у форматі BIN. Якщо не ввести цю назву, використає однакове ім'я з початковим.

Початковий файл має бути правильним EXE, резидентна частина (код та дані) не мають перевищувати 64 КБ за обсягом. Не має мати сегменту Stack.

Приклад використання програми EXE2BIN:

```
exe2bin test.exe new.bin
```

[Джерело](#)

ПОНЯТТЯ ОПЕРАЦІЙНОЇ СИСТЕМИ

Операційна система (ОС) – набір програм який керує роботою комп'ютера.

ОС виконує такі дії:

- керує роботою апаратного забезпечення
- організовує обмін даних між пристроями
- надає інтерфейс взаємодії користувачеві з комп'ютером
- організовує збереження даних на носіях
- дозволяє виконувати інші програми
- узгоджує та керує роботою програм

[Джерело](#)

СТРУКТУРА ОС MS DOS

ОС Microsoft Disk Operating System (MS DOS) створена для використання на персональних комп'ютерах і є прикладом системи з багаторівневою структурою.

MS DOS розділена на різні шари, кожен з яких виконує свої функції. Такий підхід полегшує взаємодію з системою бо кожен з кожним рівнем можна працювати окремо.

Загальна структура виглядає так:

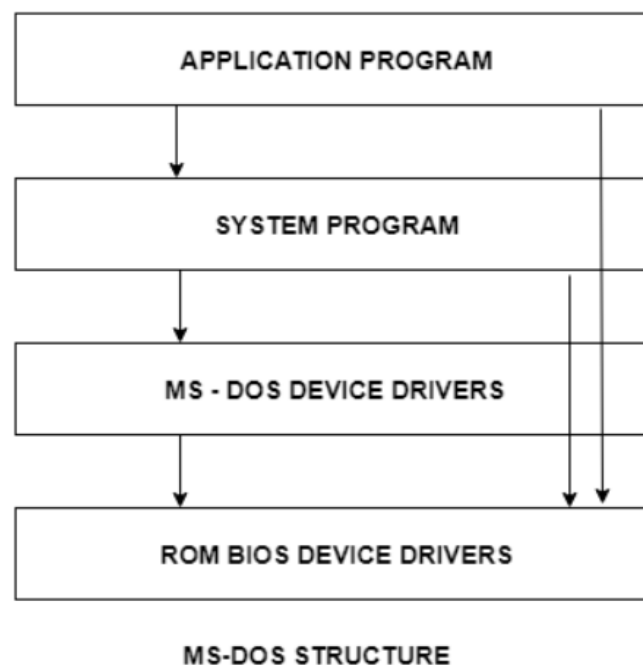


Рисунок 1.3 – Структура MS DOS

Нижче наведено пояснення кожного рівня:

Application Program або *Програмне забезпечення* містить програми корисні користувачам. Це рівень звичайних застосунків або прикладних програм.

System Program або *Системне забезпечення* надають програми для керування системою. Ці програми зазвичай не використовуються користувачем але допомагають системі взаємодіяти з іншими.

MS-DOS Device Drivers або *Драйвери пристроїв MS-DOS* є частиною операційної системи і надають різні драйвери, наприклад, клавіатурі, миші, жорстким дискам, тощо: різним зовнішнім пристроям.

ROM BIOS Device Drivers або *Драйвери пристроїв ROM BIOS* зберігають драйвери на схемах EPROM або EEPROM на материнській платі. Це драйвери основні для запуску комп'ютера. *BIOS* – Basic Input Output System або Базова система вводу виводу.

[Джерело](#)

ЗАВАНТАЖЕННЯ ОПЕРАЦІЙНОЇ СИСТЕМИ

Процес завантаження операційної системи на комп'ютері це перенесення операційної системи з вторинної пам'яті в основну. Цей процес виконується BIOSом.

Є два типи запуску системи:

- холодний: звичайний запуск коли комп'ютер не працював, а потім натиснули кнопку живлення
- теплий: виконується при перезавантаженні, коли комп'ютер вже працював і його перезавантажують

[Джерело](#)

Запуск операційної системи MS DOS

У MS DOS запуск складається з завантаження трьох файлів:

1. io.sys
2. msdos.sys

3. command.com

Процес запуску MS DOS виглядає так:

1. BIOS виконує Power-On Self Test (POST) – перевіряє конфігурації периферії, дивиться чи під'єднані усі пристрої, тощо. Робить це у Read Only Memory (ROM)
2. Шукає Master Boot Record (MBR) або Головний завантажувальний запис на фізичному диску. Якщо не знайдено на жорсткому диску, буде шукати на дисководі. Якщо не знайдено на дисководі, буде шукати на CD-ROM. Якщо не знайдено ніде, видасть повідомлення «No Boot Device Found» та зупинить роботу. Якщо завантажувальний запис знайдено, підвантажує Завантажувач операційної системи (Bootstrap Loader) до оперативної пам'яті.
3. Завантажувач підвантажує файл io.sys, потім msdos.sys – це основний файл ОС MS DOS.
4. Файл msdos.sys шукає командний інтерпретатор у файлі config.sys. Якщо знайдено, вантажить його до оперативної пам'яті. Якщо не знайдено, до пам'яті завантажує файл command.com як інтерпретатор команд за замовчуванням для MS DOS.
5. Завантажує і запускає файл autoexec.bat який містить послідовність команд DOS.
6. Показує командний рядок DOS на екрані комп'ютера, виводить літеру завантажувального диску. Якщо бачимо це, операційна система успішно завантажена з диску до оперативної пам'яті.

[Джерело](#)