

НАЦІОНАЛЬНИЙ ТЕХНІЧНИЙ УНІВЕРСИТЕТ УКРАЇНИ
«Київський політехнічний інститут імені Ігоря Сікорського»
Факультет інформатики та обчислювальної техніки
Кафедра інформатики та програмної інженерії

Звіт до комп'ютерного практикуму з дисципліни

«Системне програмне забезпечення»

Прийняв
асистент кафедри ІІІ
Пархоменко А.В.
“24” березня 2023 р.

Виконав
Студент групи ІІІ-15
Мешков А. І.

Київ – 2023

Комп'ютерний практикум № 1

Створення програм на асемблері

Загальні положення

Викладені в лекційному матеріалі.

Завдання комп'ютерного практикуму №1

1. Для програми, наведеної вище, створити файл типу .asm. Ця програма не має засобів виводу даних, тому правильність її виконання треба перевірити за допомогою додатку Турбодебагер (td.exe).
2. Скомпілювати програму, включивши потрібні опції для налагоджувача та створення файлу лістингу типу .lst.
3. Ознайомитись зі структурою файлу .lst. За вказівкою викладача, для певної команди асемблера розглянути структуру машинної команди і навести її у звіті.
4. Скомпонувати .obj-файл програми. Включити опції для налагодження та створення .map-файлу.
5. Занести до звіту адреси початку та кінця всіх сегментів з .map-файлу.
6. Завантажити до налагоджувача td.exe одержаний .exe-файл програми.
7. У вікні CPU у полі DUMP знайти початкову адресу сегмента даних та записати його до звіту. Знайти масиви SOURCE та DEST. Дані у масиві SOURCE подаються у шістнадцятковій системі.
8. У покроковому режимі за допомогою клавіші F7 виконати програму. Одержані результати у масиві DEST показати викладачеві.

Текст програми

```
STSEG SEGMENT PARA STACK "STACK"  
DB 64 DUP ("STACK")  
STSEG ENDS
```

```
DSEG SEGMENT PARA PUBLIC "DATA"  
SOURCE DB 10, 20, 30, 40  
DEST DB 4 DUP ( "?" )  
DSEG ENDS
```

```
CSEG SEGMENT PARA PUBLIC "CODE"  
MAIN PROC FAR  
ASSUME CS: CSEG, DS: DSEG, SS: STSEG
```

```
PUSH DS  
MOV AX, 0 ; або XOR AX, AX  
PUSH AX
```

```
MOV AX, DSEG  
MOV DS, AX
```

```
MOV DEST, 0  
MOV DEST+1, 0  
MOV DEST+2, 0  
MOV DEST+3, 0
```

```
MOV AL, SOURCE  
MOV DEST+3, AL  
MOV AL, SOURCE+1  
MOV DEST+2, AL  
MOV AL, SOURCE+2  
MOV DEST+1, AL  
MOV AL, SOURCE+3  
MOV DEST, AL
```

```
RET  
MAIN ENDP  
CSEG ENDS  
END MAIN
```

Введені та отримані результати

Вміст .lst файлу

Turbo Assembler Version 4.0 03/12/23 23:19:29 Page 1
kp1.asm

```
1 0000          STSEG SEGMENT PARA STACK "STAC"
2 0000 40*(53 54 41 43 4B) DB      64 DUP ("STACK")
3 0140          STSEG ENDS
4
5 0000          DSEG SEGMENT PARA PUBLIC "DATA"
6 0000 0A 14 1E 28 SOURCE DB 10, 20, 30, 40
7 0004 04*(3F)          DEST DB 4 DUP ("?")
8 0008          DSEG ENDS
9
10 0000         CSEG SEGMENT PARA PUBLIC "CODE"
11 0000         MAIN PROC FAR
12          ASSUME CS:      CSEG, DS: DSEG,      SS: STSEG
13          ; адреса повернення
14 0000 1E          PUSH DS
15 0001 B8 0000          MOV AX, 0 ; або XOR AX,      AX
16 0004 50          PUSH AX
17          ; ініціалізація DS
18 0005 B8 0000s          MOV AX, DSEG
19 0008 8E D8          MOV DS, AX
20          ; обнуляємо масив
21 000A C6 06 0004r 00      MOV DEST, 0
22 000F C6 06 0005r 00      MOV DEST+1, 0
23 0014 C6 06 0006r 00      MOV DEST+2, 0
24 0019 C6 06 0007r 00      MOV DEST+3, 0
25          ; пересилання
26 001E A0 0000r          MOV AL, SOURCE
27 0021 A2 0007r          MOV DEST+3, AL
28 0024 A0 0001r          MOV AL, SOURCE+1
29 0027 A2 0006r          MOV DEST+2, AL
30 002A A0 0002r          MOV AL, SOURCE+2
31 002D A2 0005r          MOV DEST+1, AL
32 0030 A0 0003r          MOV AL, SOURCE+3
33 0033 A2 0004r          MOV DEST, AL
34
35 0036 CB          RET
36 0037          MAIN ENDP
37 0037          CSEG ENDS
```

Turbo Assembler Version 4.0 03/12/23 23:19:29 Page 2
Symbol Table

| Symbol Name | Type | Value |
|-------------|--------|------------|
| ??DATE | Text | "03/12/23" |
| ??FILENAME | Text | "kp1 " |
| ??TIME | Text | "23:19:29" |
| ??VERSION | Number | 0400 |
| @CPU | Text | 0101H |
| @CURSEG | Text | CSEG |
| @FILENAME | Text | KP1 |
| @WORDSIZE | Text | 2 |
| DEST | Byte | DSEG:0004 |
| MAIN | Far | CSEG:0000 |
| SOURCE | Byte | DSEG:0000 |

| Groups & Segments | Bit | Size | Align | Combine | Class |
|-------------------|-----|------|-------|---------|-------------|
| CSEG | 16 | 0037 | Para | | Public CODE |
| DSEG | 16 | 0008 | Para | | Public DATA |
| STSEG | 16 | 0140 | Para | | Stack STAC |

Вміст .map файлу

| Start | Stop | Length | Name | Class |
|--------|--------|--------|-------|-------|
| 00000H | 0013FH | 00140H | STSEG | STAC |
| 00140H | 00147H | 00008H | DSEG | DATA |
| 00150H | 00186H | 00037H | CSEG | CODE |

| Address | Publics by Name |
|---------|-----------------|
|---------|-----------------|

| Address | Publics by Value |
|---------|------------------|
|---------|------------------|

Program entry point at 0015:0000

Warning: No stack

Схема функціонування програми

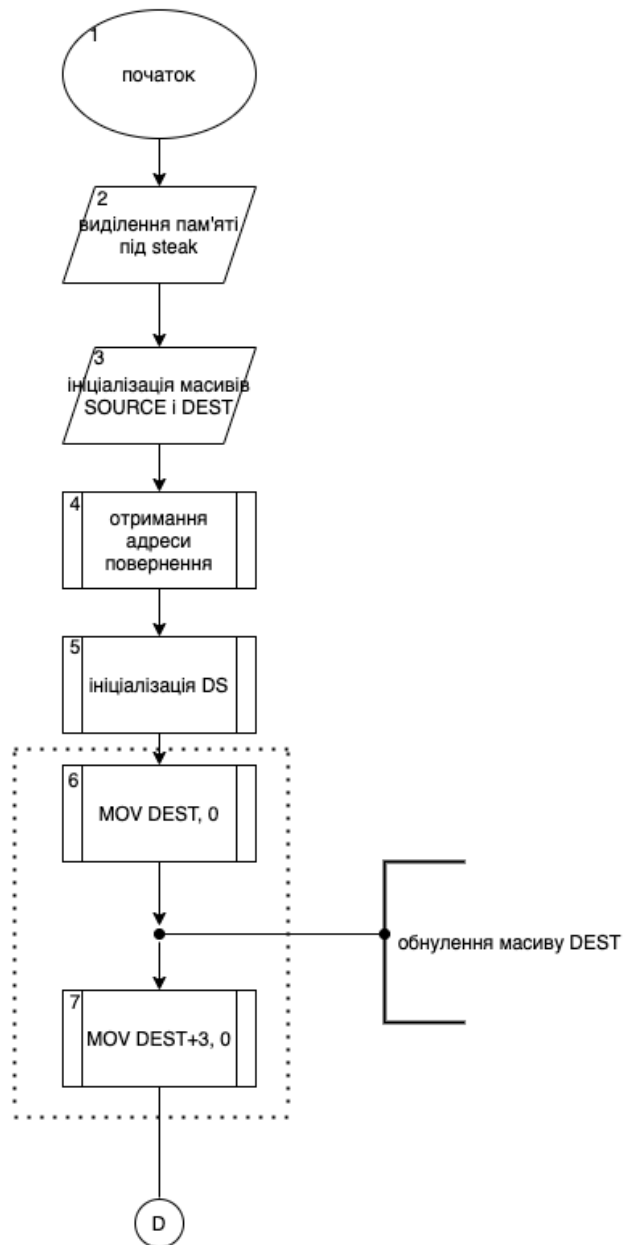
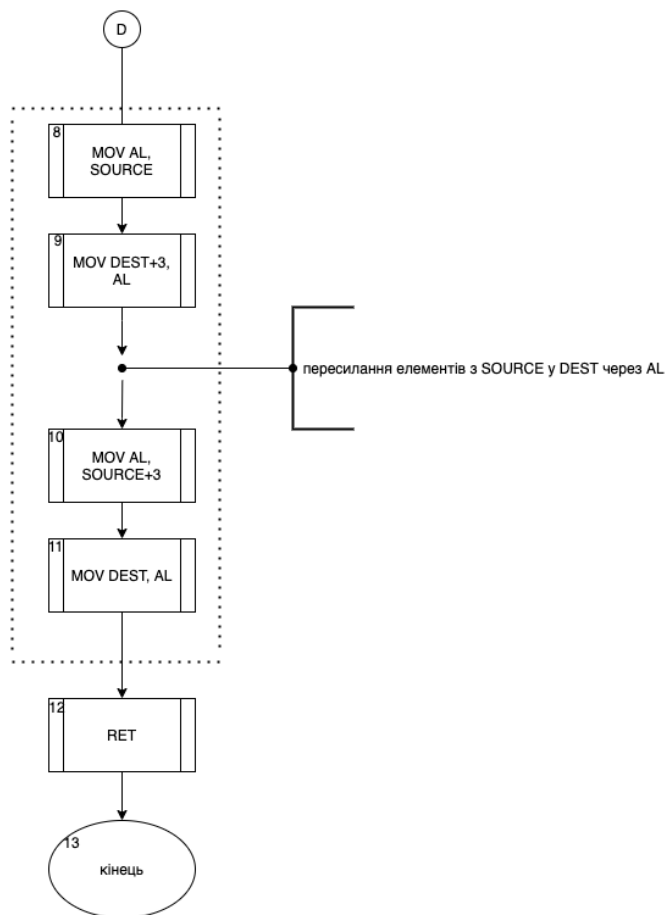


Рис. 1.1 Схема функціонування програми



Продовження рис. 1.1

Вікно DUMP

| | | | | | | | | | |
|---------|----|----|----|----|----|----|----|----|----------|
| ds:0000 | 0A | 14 | 1E | 28 | 3F | 3F | 3F | 3F | ☐9▲<???? |
| ds:0008 | 00 | 00 | 00 | 00 | 00 | 00 | 00 | 00 | |
| ds:0010 | 1E | B8 | 00 | 00 | 50 | B8 | 05 | 53 | ▲7 P7 ♣S |
| ds:0018 | 8E | D8 | C6 | 06 | 04 | 00 | 00 | C6 | 07 ♣♣ ♣ |
| ds:0020 | 06 | 05 | 00 | 00 | C6 | 06 | 06 | 00 | ♣♣ ♣♣ |

Рис. 1.2 Вікно DUMP

До виконання програми

Масив SOURCE

| | | | | | | | | | |
|---------|----|----|----|----|----|----|----|----|----------|
| ds:0000 | 0A | 14 | 1E | 28 | 3F | 3F | 3F | 3F | ☐9▲<???? |
| ds:0008 | 00 | 00 | 00 | 00 | 00 | 00 | 00 | 00 | |
| ds:0010 | 1E | B8 | 00 | 00 | 50 | B8 | 05 | 53 | ▲7 P7 ♣S |
| ds:0018 | 8E | D8 | C6 | 06 | 04 | 00 | 00 | C6 | 07 ♣♣ ♣ |
| ds:0020 | 06 | 05 | 00 | 00 | C6 | 06 | 06 | 00 | ♣♣ ♣♣ |

Рис. 1.3 Масив SOURCE

Масив DEST

| | | | | | | | | | |
|---------|----|----|----|----|----|----|----|----|----------|
| ds:0000 | 0A | 14 | 1E | 28 | 3F | 3F | 3F | 3F | ☐9▲<???? |
| ds:0008 | 00 | 00 | 00 | 00 | 00 | 00 | 00 | 00 | |
| ds:0010 | 1E | B8 | 00 | 00 | 50 | B8 | 05 | 53 | ▲7 P7♠S |
| ds:0018 | 8E | D8 | C6 | 06 | 04 | 00 | 00 | C6 | 07└┐♠♦└ |
| ds:0020 | 06 | 05 | 00 | 00 | C6 | 06 | 06 | 00 | ♠♠└┐♠♠ |

Рис. 1.4 Масив DEST перед виконанням програми

Після виконання програми
Масив DEST

| | | | | | | | | | |
|---------|----|----|----|----|----|----|----|----|----------|
| ds:0000 | 0A | 14 | 1E | 28 | 28 | 1E | 14 | 0A | ☐9▲<<▲9☐ |
| ds:0008 | 00 | 00 | 00 | 00 | 00 | 00 | 00 | 00 | |
| ds:0010 | 1E | B8 | 00 | 00 | 50 | B8 | 05 | 53 | ▲7 P7♠S |
| ds:0018 | 8E | D8 | C6 | 06 | 04 | 00 | 00 | C6 | 07└┐♠♦└ |
| ds:0020 | 06 | 05 | 00 | 00 | C6 | 06 | 06 | 00 | ♠♠└┐♠♠ |

Рисунок 1.5 Масив DEST після виконання програми

Висновок: Під час виконання комп'ютерного практикуму мною було опрацьовано теоретичний матеріал по мові програмування асемблер, створено, скомпільовано та перевірено програму: створено .asm файл з кодом програми, скомпільовано її зі створенням файлу .lst, скомпановано .obj і .map файли, завантажено програму у td.exe, у вікні CPU перевірено програму у покроковому режимі. В результаті перевірки було з'ясовано, що програма виконує своє завдання: перенесення даних з масиву SOURCE у масив DEST.