

Міністерство освіти і науки України
Національний технічний університет України
“Київський політехнічний інститут імені Ігоря Сікорського”
Факультет інформатики та обчислювальної техніки
Кафедра автоматики та управління в технічних системах

Лабораторна робота 2

Архітектура комп'ютера

*Прямий доступ до пам'яті в архітектурі IA-32 (x86) у Real
Adress Mode*

Виконав:
Студент групи IT-01
Тимошенко Олексій

Київ 2021

Тема: прямий доступ до пам'яті в архітектурі IA-32 (x86) у Real Adress Mode

Мета: отримати основні навички роботи з відеопам'яттю та методи доступу до неї на асемблері

Файли доступні за посиланням:

<https://github.com/OlexiiT/assembler/tree/main/L2>

Хід роботи:

Відповідно до свого варіанту в таблиці, необхідно було вивести прямокутник нулів салатового кольору на жовтому фоні у місці, за заданими координатами.

Варіанти	1	2	3	4	5	6	7	8
Координата x	2	40	2	40	2	40	30	50
Координата y	2	2	10	10	15	15	30	50
Кольори прямокутника	Синій	Зелений	Бірюза	Червон	Білий	Блакитний	Жовтий	Салатов.
Кольори надпису	Зелений	Синій	Червон	Бірюза	Блакитний	Білий	Салатов.	Жовтий

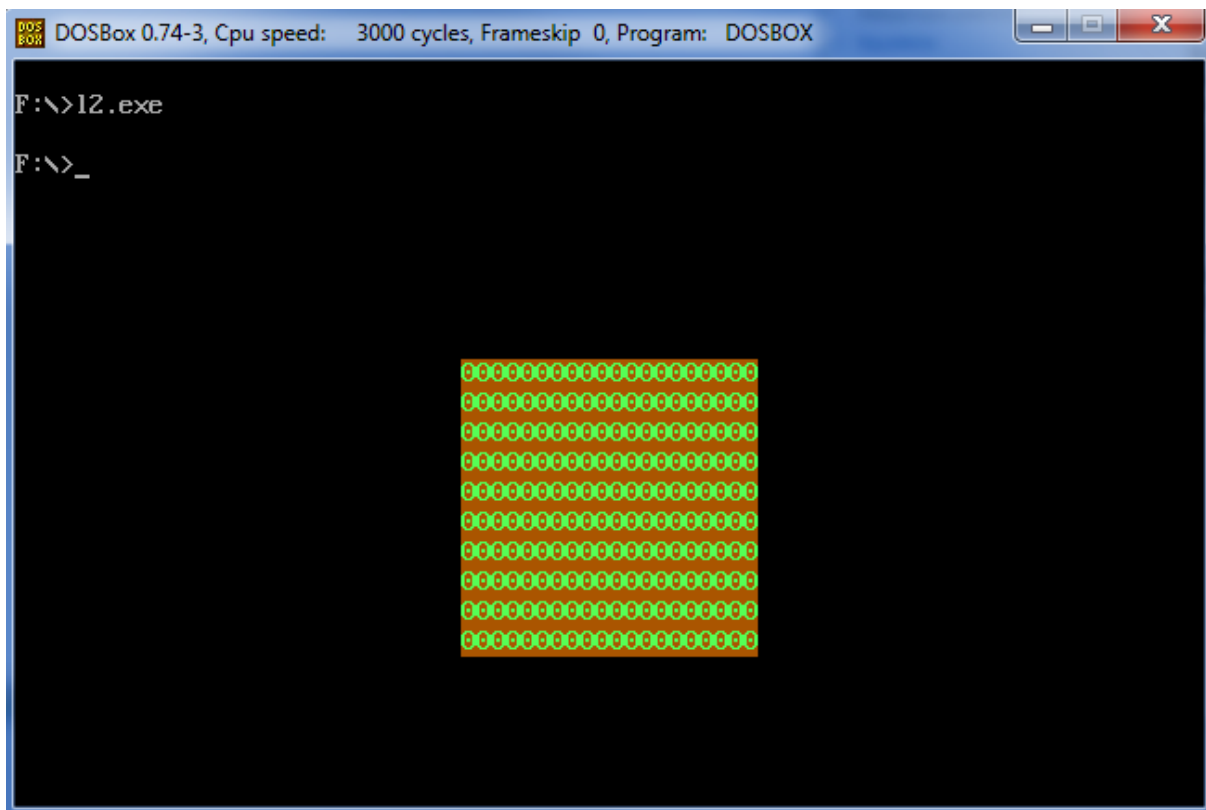
Враховуючи, що у горизонтальному рядку міститься 160 байт, порахувати початкове зміщення можна за формулою $160 * y + 2 * x$, а щоб перейти на наступний рядок треба додати 160.

```

1 TITLE Lab2
2 IDEAL ;Директива - тип Асемблера
3 MODEL small ;Директива - тип моделі пам'яті
4 Stack 512 ;Директива - розмір стеку
5 ;-----Макроси-----
6 Macro M_Init ;Початок макросу
7 mov ax, @data ;ax <- @data
8 mov ds, ax ;ds <- ax
9 mov es, ax ;es <- ax
10 ENDM M_Init ;Кінець макросу
11 ;-----Сегмент даних-----
12 DATASEG
13 exCode db 0
14 rect_line dw 0ea30h, 0ea30h, 0ea30h, 0ea30h, 0ea30h ;Рядок, що буде надруковано
15 dw 0ea30h, 0ea30h, 0ea30h, 0ea30h, 0ea30h
16 dw 0ea30h, 0ea30h, 0ea30h, 0ea30h, 0ea30h
17 dw 0ea30h, 0ea30h, 0ea30h, 0ea30h, 0ea30h
18 rect_line_length = $-rect_line ;Довжина рядка
19 ;-----Сегмент коду-----
20 CODESEG
21 Start:
22 M_Init
23
24 mov dx, 1660 ;1640 = 160 * 10 + 30 * 2 (Зміщення лівого верхнього кута у положення (30, 10))
25 mov cx, 10 ;Лічильник кількості ітерацій циклу
26
27 loopStart: ;Мітка циклу
28
29 mov bx, cx ;bx <- cx
30 mov ax, 0B800h ;Сегментна адреса відеопам'яті
31 mov es, ax ;es <- ax
32
33 mov di, dx ;Початок виведення на екран
34 mov si, offset rect_line ;si <- rect_line
35 mov cx, rect_line_length ;Число байтів
36 cld
37 rep movsb
38 mov cx, bx ;cx <- bx
39 add dx, 160 ;Зміщення на наступний рядок
40 loop loopStart
41
42 Exit:
43 mov ah, 04ch
44 mov al, [exCode] ;Отримання коду виходу
45 int 21h ;Виклик функції DOS 4ch
46 END Start

```

Результат виконання коду:



Висновки:

Ми набули знань та навичок з розробки ПЗ на Асемблері для управління відеопам'ятю в архітектурі IA-32 (x86) у Real Address Mode.