

Лабораторная работа № 2

тема «Assembler. Использование команды PUSH и POP»

1. Выполните ассемблирование и компоновку программы, используя для этого следующие команды:

```
tasm /zi stek  
tlink /v stek
```

Текст программы:

1	IDEAL
2	MODEL small
3	STACK 256
4	DATASEG
5	exCode DB 0
6	speed DB 99
7	CODESEG
8	Start:
9	mov ax, @data
10	mov ds, ax
11	push ax
12	push bx
13	mov ax, -1
14	mov bx, -2
15	mov cx, 0
16	mov dx, 0
17	push ax
18	push bx
19	pop cx
20	pop dx
21	pop bx
22	pop ax
23	Exit:
24	mov ah, 04Ch
25	mov al, [exCode]
26	int 21h
27	END Start

2. Загрузите программу в Turbo Debugger (нумерация строк соответствует тексту программы в лабораторной работе) и выполните следующие действия:

- Откройте окно CPU: <Alt+V+C>.
 - Распахните это окно на весь экран: <F5>.
 - Для пошагового выполнения программы используете клавишу <F8>.
 - Дважды нажмите <F8>, чтобы остановиться в строке 11.
 - Какое значение имеет регистр сегмента стека SS в начале программы?
 - Какое значение имеет стековый указатель SP и почему?
 - Наблюдайте за значением стека в правом нижнем углу - в окне с адресами, первый из которых - SS. Чтобы протолкнуть значение AX в стек, нажмите <F8>. Снова нажмите <F8> для проталкивания значения BX. Вершина стека, расположенная внизу окна, отмечается стрелкой.
 - Выполните команды в строках 13-16, загрузив в регистры тестовые значения.
 - Выполните команду в строке 17 для проталкивания в стек значения AX. Наблюдайте за содержимым стека и значением SP до и после проталкивания.
 - Выполните команду в строке 18.
 - Строки 19-20 выталкивают стек, выбирая значение [SS : SP] и увеличивая SP на 2, адресуя тем самым следующее слово. Выполните эти строки. Обратите внимание на то, что вы можете вытолкнуть значения из стека в регистры, отличные от тех, которые проталкивались ранее.
 - Строки 21-22 возвращают в регистры BX и AX значения, которые хранились там до выполнения строк 11-12. Выполните эти команды.
3. Выполните программу еще раз. Для восстановления исходных данных используйте <Ctrl + F2>.
4. Запишите в тетрадь, как будет выглядеть содержимое стека после выполнения команды в строке 18.
5. Запишите в тетрадь, как будет выглядеть содержимое стека после выполнения команд в строках: 20 и 22. Какие регистры изменят свое значение и какое значение будет?
6. Текст программы содержит следующие команды:
- ```
mov ax, 1
mov bx, 2
mov dx, 3
push ax
push bx
push dx
```

Как нужно записать команды извлечения из стека, чтобы в регистре AX было значение 2, в регистре BX - 3, а в регистре DX - 1 ? Запишите в тетрадь.