

## Разбор контрольной работы № 1.

№ 1	ЭВМ и ПУ. 4 семестр	
1. В чем преимущества двоичной системы счисления для ЭВМ? Что такое число в форме с плавающей точкой? Объяснить удобства такой формы представления чисел.		
2. Перечислить все 8-битные общие регистры. Какие из них выполняют специальные функции?		
3. Как осуществляется переход к подпрограмме обработки прерываний в реальном режиме?		
4. Задан фрагмент программы. Определить способы адресации операндов в каждой команде.		
<pre> varC    equ    dword ptr[bp+4]; адрес параметра C         push   bp    ; сохранение bp         mov    bp,sp; настройка bp на вершину стека         les   bx,varC; es:[bx]:=адрес C         mov   al,es:[bx]; al:=символ         cmp   al,'a'  ; символ меньше "a"?</pre>		
5. Какие из приведенных ниже команд являются недопустимыми? Указать ошибку в каждой недопустимой команде. Считать все имена переменными, которые определены как слова: а) MOV [BX+4*3], SP; б) MOV WORD_OP1, WORD_OP2; в) MOV AX, WORD_OP [DX]; г) MOV SP, SS: [BX+SI]; д) MOV CS, AX;		
6. Каково будет содержимое регистров стека арифметического сопроцессора после выполнения команды <div style="text-align: center;">FDIVR ST(2), ST(0),</div> если до выполнения ST(0)=0.5, ST(1)=0.3, ST(2)=-0.1, ST(3)=0.4, ST(4)=0.5.		
7. Следующий фрагмент программы записать без использования строковых команд:		
<pre>         cld         rep movsb</pre>		

Вопрос 1. Введение в архитектуру ЭВМ. Программирование на языке ассемблера.

Вопрос 2, вопрос 3. Процессоры Intel x86. Структура. Основные команды

#### Задание 4.

Задан фрагмент программы. Определить способы адресации операндов в каждой команде.

```
varC equ dword ptr[bp+4]; адрес параметра C
push bp ; сохранение bp
mov bp,sp; настройка bp на вершину стека
les bx,varC; es:[bx]:=адрес C
mov al,es:[bx]; al:=символ
cmp al,'a' ; символ меньше "a"?
```

Команда	Адресация	
	первого операнда	второго операнда
push bp	регистровая	по умолчанию
mov bp,sp	регистровая	регистровая
les bx, dword ptr[bp+4]	регистровая	косвенная (базовая)
mov al,es:[bx]	регистровая	косвенная
cmp al,'a'	регистровая	непосредственная
mov bx,fs:[task_counter]	регистровая	прямая

## Задание 5

Какие из приведенных ниже команд являются недопустимыми? Указать ошибку в каждой недопустимой команде. Считать все имена переменными, которые определены как слова:

- а) MOV [BX+4\*3], SP;
- б) MOV WORD\_OP1, WORD\_OP2;
- в) MOV AX, WORD\_OP [DX];
- г) MOV SP, SS: [BX+SI];
- д) MOV CS, AX

Команда	Ошибка
MOV [BX+4*3], SP	нет
MOV WORD_OP1, WORD_OP2	оба операнда не могут быть ячейками памяти
MOV AX, WORD_OP [DX]	регистр DX нельзя использовать для адресации
MOV SP, SS: [BX+SI];	нет
MOV CS, AX	регистр CS нельзя загружать командой MOV

Как проверить в программе на ассемблере?

data segment

WORD\_OP1 dw 5

WORD\_OP2 dw 10

WORD\_OP dw 20

data ends

code segment

assume cs: code, ds: data

mov ax, data

mov ds, ax

MOV [BX+4\*3], SP;

MOV WORD\_OP1, WORD\_OP2;

MOV AX, WORD\_OP [DX];

MOV SP, SS: [BX+SI];

MOV CS, AX

code ends

end

## Задание 6.

Каково будет содержимое регистров стека арифметического сопроцессора после выполнения команды  
FDIVR ST(2), ST(0),  
если до выполнения  $ST(0)=0.5$ ,  $ST(1)=0.3$ ,  $ST(2)=-0.1$ ,  $ST(3)=0.4$ ,  $ST(4)=0.5$ .

Команда обратного деления.

Операция приемник, источник

Обычная команда: приемник = (приемник) операция (источник)

Обратная операция: приемник = (источник) операция (приемник).

В нашем случае

$$ST(2) = ST(0)/ST(2) = 0.5/(-0.1) = -5.0$$

Изменится только регистр  $ST(2)$ :

$$ST(0)=0.5, ST(1)=0.3, ST(2)=-5.0, ST(3)=0.4, ST(4)=0.5.$$

## Задание 7.

Следующий фрагмент программы записать без использования строковых команд:

```
cld  
rep movsb  
jcxz exit  
push ax  
pushf  
m1: mov al, [si]  
mov es:[di], al  
inc si; не изменяет CF, остальные флаги по результату операции  
inc di; не изменяет CF, остальные флаги по результату операции  
loop m1  
popf  
pop ax  
exit:
```

### Задание 6 (устные комментарии)

Выполнение команд. Указать, как изменится содержимое регистров, стека, по каким адресам памяти будет обращаться процессор.

Варианты команд

MOV AX, [BX]

PUSH AX

RET n

MOV AX, [BX]

LFS EDI, [EBP+4]

LDS SI, [BP+10]

BOUND SI, dword ptr [BX]

POP AX

INTO

INT n

FAR JMP dword ptr [BX]

FAR CALL dword ptr [bx]

FADD

FSTP mem и др. команды сопроцессора

### Задание 7

Следующий фрагмент программы записать без использования строковых команд.

Программирование арифметического сопроцессора.

Вычисление целочисленных выражений.