# Лабораторна робота №2

Тема: «Арифметичні та логічні дії над регістрами процесора»

Мета роботи: використовуючи арифметичні та логічні інструкції процесора отримати результати заданих виразів.

## Хід роботи

#### Завдання на 3 бали

1. Відкомпілювати та відлагодити приклади представлені у роботі. Пояснити їх роботу, стани регістрів та принципи роботи арифметичних команд.

# Приклад 1

Потрібно обчислити значення формули: e = a - (b + c - 1) + (-d). Всі числа  $\epsilon$  8-бітними цілими зі знаком.

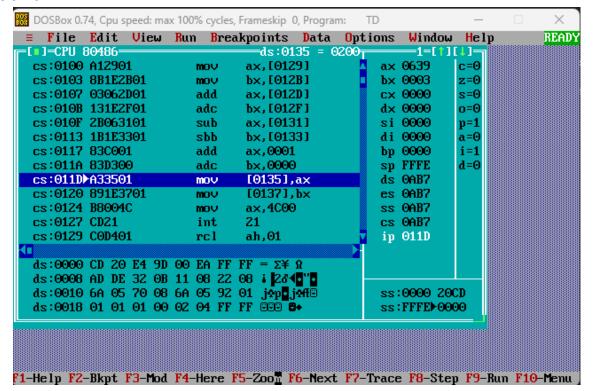
```
use16
org 100h
    mov al, [a] ;Завантажуємо значення а в AL
    mov ah, [b]; Завантажуємо значення b в АН
    add ah, [c]; AH = AH + c = b+c
    dec \ ah \ ; AH = AH - 1 = b+c-1
    sub al, ah; AL = AL - AH = a-(b+c-1)
    mov cl, [d] ; CL = d
    neg cl ; CL = -CL = -d
    add al, cl ; AL = AL + CL = a - (b+c-1) + (-d)
    mov [e], al ;Зберігаємо результат в е
    mov ax, 4C00h;\
    int 21h ;/ Завершення програми
a db 2
b db 3
c db 5
d db -8
e db?
```

```
DOSBox 0.74, Cpu speed: max 100% cycles, Frameskip 0, Program: TD
     ≡ File Edit View Run Breakpoints Data Options Window Help
        CS:0100 A01F01
                                                                                                                                                                                                             ax 0703
                                                                                                                                 al,[011F]
        cs:0103 8A262001
                                                                                                                                                                                                             b× 0000
                                                                                                                                 ah, [0120]
                                                                                                   MOV
                                                                                                                                 ah,[0121]
                                                                                                                                                                                                             cx 0008
        cs:0107 02262101
                                                                                                    add
         cs:010B FECC
                                                                                                     dec
                                                                                                                                 ah
                                                                                                                                 al,ah
                                                                                                                                                                                                             si 0000
         cs:010D 28E0
                                                                                                     sub
                                                                                                                                                                                                                                                           p=1
        cs:010F 8A0E2201
                                                                                                                                 cl,[0122]
                                                                                                                                                                                                              di 0000
                                                                                                    MOV
        cs:0113 F6D9
                                                                                                                                                                                                              bp 0000
                                                                                                    neg
                                                                                                                                 сl
        cs:0115 00C8
                                                                                                                                                                                                              sp FFFE
                                                                                                     add
        cs:0117 A22301
                                                                                                                                  [0123],al
                                                                                                                                                                                                              ds OAB7
                                                                                                     MOV
                                                                                                                                 ax,4000
        cs:011A B8004C
                                                                                                                                                                                                              es OAB7
                                                                                                     MOV
         cs:011D CD21
                                                                                                                                                                                                              ss OAB7
                                                                                                      int
        cs:011F 0203
                                                                                                     add
                                                                                                                                 al,[bp+di]
                                                                                                                                                                                                              cs OAB7
        cs:0121 05F803
                                                                                                    add
                                                                                                                                 ax,03F8
                                                                                                                                                                                                              ip 011A
        ds:011F 02 03 05 F8 <mark>03</mark> 04 00 C7 5*4°*+ ∥
ds:0127 46 FA 00 00 C7 46 F4 00 F· ∥F∫
        ds:012F 00 EB EB 83 3E BD 00 02 δ∑â> Bd ds:0137 76 29 61 P3 00 P4 P3 00 P4
                                                                                                                                                                                                              ss:0000 ZOCD
         ss:FFFE>0000
```

## Приклад 2

Обчислити значення формули k = i + j - n + 1, де змінні k, i, j і n  $\epsilon$  32-бітними цілими числами без знака.

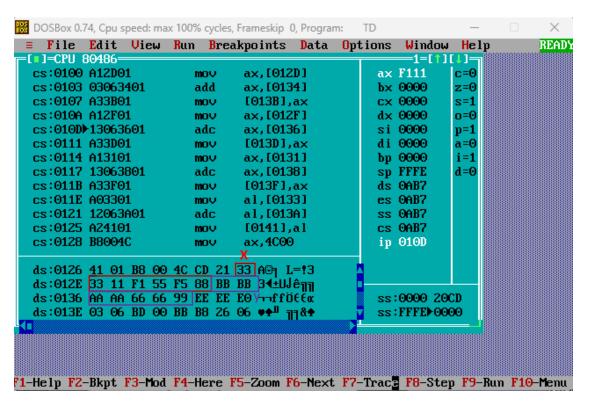
```
use16
org 100h
  mov ax, word[i] ;Завантажуємо молодшу частину і в АХ
 mov bx, word[i+2]; Завантажуємо старшу частину і в ВХ
  add ax, word[j] ;Додаємо молодші частини і та j
  adc bx, word[j+2] ;Додаємо старші частини і и j
  sub ax, word[n]
  sbb bx, word[n+2]; BX:AX = i+j-n
  add ax, 1 ; Команда INC тут не можна використовувати
  adc bx,0 ;BX:AX = i+j-n+1
 mov word[k],ax ;\
 mov word[k+2], bx ;/ Зберігаємо результат в k
 mov ax, 4C00h ; \
  int 21h ;/ Завершення програми
i dd 120000
i dd 80500
n dd 2300
k dd?
```



#### Приклад 3

Реалізувати програму яка складає два 7-байтних значення, для додавання використовувати тільки один регістр.

```
use16
org 100h
    mov ax, word[x]
    add ax, word[y]
    mov word[z],ax
    mov ax, word [x+2]
    adc ax, word [y+2]
    mov word [z+2], ax
    mov ax, word [x+4]
    adc ax, word [y+4]
    mov word[z+4], ax
    mov al, byte [x+6]
    adc al, byte[y+6]
    mov byte[z+6],al
    mov ax, 4C00h
    int 21h
x dd 0xF1113333
  dw 0xF555
  db 0x88
y dd 0xAAAABBBB
  dw 0x6666
  db 0x99
z rb 7
```



## Приклад 4

Припустимо, в програмі потрібно обчислювати координату якогось об'єкта, що рухається по формулі:  $x = x_0 + v_0 t + a t^2/2$ 

```
use16
org 100h
    mov al, [v0]
                    ; AL = v0
    mov cl,[t]; CL = t
                    ;AX = AL*CL = v0*t
    mul cl
    mov bx,ax
                    ;BX = AX = v0*t
    mov al, [a]
                     ; AL = a
    mul cl
                     ; AX = AL*CL = a*t
    mov ch, 0 ;Перетворюємо t в слово у регістрі СХ
    mul cx
                ; DX:AX = AX*CX = a*(t^2)
    mov cl,2
                ; CL = 2 = CX, \text{ так як CH } = 0
                ; AX = DX : AX/2 = a*(t^2)/2
    div cx
    add ax,bx ;AX = AX+BX = v0*t + a*(t^2)/2
    add al, [x0];
    adc ah, ch ; /AX = AX + x0 = x0 + v0*t + a*(t^2)/2
    mov [x],ax
                ;Зберігаємо результат в х
    mov ax, 4C00h;
    int 21h ;/ Завершення програми
x0 db 188
v0 db 7
a db 3
t db 25
x dw?
```

```
DOSBox 0.74, Cpu speed: max 100% cycles, Frameskip 0, Program: TD —
    File Edit View Run Breakpoints Data Options Window Help
 [ • ]=CPU 80486:
                       add
adc
mov
  cs:011A 02062801
                                  al,[0128]
                                                       ax 0514 X
  cs:011E 10EC
                                                       bx 00AF
                                   ah,ch
                                                       cx 0002
  cs:0120 A32C01
                                  [012C],ax
                         mov
int
                                  ax,4000
  cs:0123 B8004C
                                                       dx 0001
                                                       si 0000
  cs:0126 CD21
                                  21
                                  sp,0307
                        MOV
  cs:0128 BC0703
                                                       di 0000
                                                       bp 0000
  cs:012B 1946F4
                                   [bp-0Cl,ax
                                  [bx+sil,al
                                                       sp FFFE
  cs:012E 0000
                                                                   d=0
                           add
  cs:0130 EBE8
                           jmp
                                  011A
                                                       ds OAB7
                                  word ptr [00BD],
  cs:0132 833EBD0002
                                                       es OAB7
  cs:0137 7629
                                                       ss OAB7
                                   0162
                                  ax,[00D3]
                                                       cs OAB7
  cs:0139 A1D300
                           MOV
                                                        ip 0123
  cs:013C D1E0
  ds:0128 BC 07 03 19 14 05 00 00 <sup>1</sup>/<sub>2</sub> • ↓¶♠
  ds:0130 EB E8 83 3E BD 00 02 76 \delta \delta \delta \delta \rangle^{\parallel} \Theta v ds:0138 29 A1 D3 00 D1 E0 03 06 )í^{\parallel} \pi \alpha \Phi \Phi
                                                       ss:0000 ZOCD
  ds:0140 BD 00 BB B8 26 06 6A 00 <sup>□</sup> 77&+j
                                                       ss:FFFE>0000
```

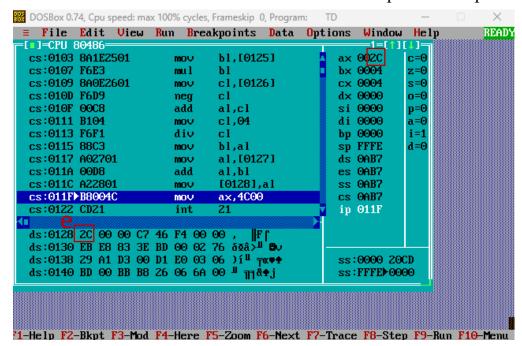
e db?

#### Завдання на 4 бали

- 1. Виконати завдання на 3 бали.
- 2. Самостійно опрацювати логічні інструкції AND, OR та XOR. Дані інструкції мають такий же синтаксис як команди ADD та SUB.
- 3. Виконати завдання згідно свого варіанту із таблиці 2.1

Варіант	Арифметичний вираз	Логічний вираз
6	$\frac{7\times3-5}{4}+40$	(19 OR (41 AND 8) XOR 7) OR 3

```
use16
org 100h
 mov al,[a] ;Завантажуємо значення а в AL
 mov bl, [b]; Завантажуємо значення b в BL
 mul bl ; AX=AL(7)*BL(3)=21
 mov cl, [c] ; CL = c(5)
 neg cl ; CL = -CL = -c(-5)
  add al,cl ;7*3-5=16
 mov cl, 4
 div cl ; (7*3-5)/4=4
 mov bl,al ;BL=4
 mov al, [d]
 add al,bl ;40+(7*3-5)/4
 mov [e],al
 mov ax, 4C00h
  int 21h
;-----
;40+(7*3-5)/4=44
a db 7
b db 3
c db 5
d db 40
```



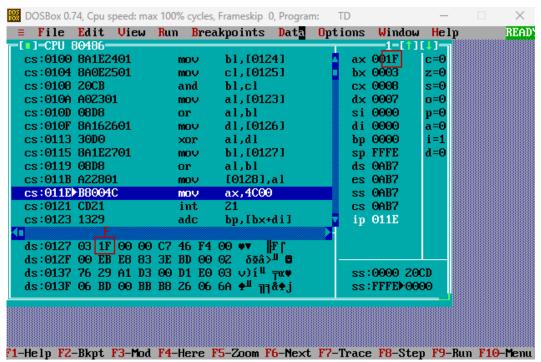
use16 org 100h

```
mov bl, [b] ; BL = 41
    mov cl,[c]
                   ; CL = 8
    and bl,cl
                    ; BL = 41 AND 8
                                                101001
                                           and 001000
                                                001000 = 8
                   ; AL = 19
    mov al, [a]
    or al, bl
                    ; AL = 19 \text{ OR } 8
                                                 10011
                                              or 01000
                                                 11011 = 27
    mov dl, [d]
                   ; DL = 7
                    ; AL = 27 XOR 7
    xor al, dl
                                                 11011
                                             xor 00111
                                                 11100 = 28
                   ; BL = 3
    mov bl, [e]
    or al, bl
                    ; AL = 28 \text{ OR } 3
                                                 11100
                                              or 00011
                                                 111111 = 31
    mov [f],al
    mov ax, 4C00h
    int 21h
a db 19
b db 41
```

```
СП.ЛР2
c db 8
```

d db 7

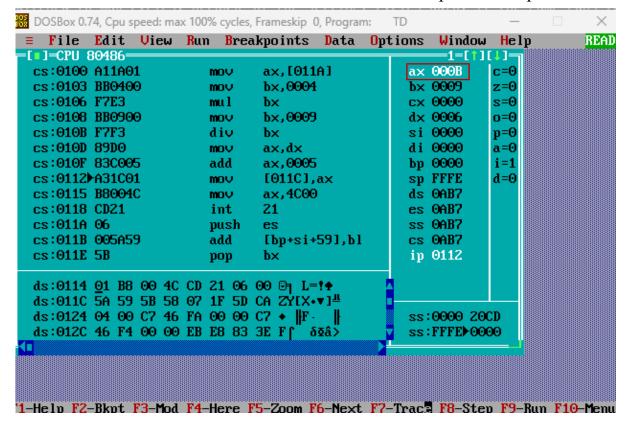
e db 3 f db?



Завдання на 5 балів

- 1. Виконати завдання на 3 та на 4 бали.
- 2. Реалізувати додавання двох M-байтних чисел. Де M обчислюється наступним чином:  $M=((C*4) \mod 9) + 5$ , де C-номер варіанту за журналом.

```
use16
org 100h
mov ax, [C]; AX = C
mov bx, 4
mul bx
                ; AX(C) * BX(4)
mov bx, 9
                ; AX = (C*4)/9, DX = (C*4) \mod 9
 div bx
                ; АХ = остача
mov ax, dx
 add ax, 5
                ; AX = M
mov [M], ax
mov ax, 4C00h
 int 21h
C dw 6
M dw ?
```



use16 org 100h

```
mov ax, word[x]
                    ;3344
add ax, word[y]
                    ;+C3D4
mov word[z],ax
                    ;=F718
mov ax, word[x+2]
                    ;1122
adc ax,word[y+2] ;+A1B2
mov word [z+2], ax
                    ; = B2D4
mov ax, word[x+4]
                    ;7788
adc ax, word[y+4]
                    ;+0718
mov word[z+4],ax
                    ;=7EA0
mov ax, word [x+6]
                    ;5566
adc ax, word[y+6]
                    ;+E5F6
mov word [z+6], ax
                    ;=3B5C
                  ;99AA
mov ax, word [x+8]
adc ax, word[y+8]
                    ;+192A
mov word[z+8],ax
                    ; = B2D5
```

```
СП.ЛР2
```

```
mov al, byte[x+10]; BB
  adc al,byte[y+10] ;+3B
 mov byte[z+10],al
                      ;=F6
 mov ax, 4C00h ;\
  int 21h ;/ Завершення програми
x dd 0x11223344
  dd 0x55667788
  dw 0x99AA
  db 0xBB
y dd 0xA1B2C3D4
  dd 0xE5F60718
  dw 0x192A
  db 0x3B
z rb 11
                      ;F6B2D53B5C7EA0B2D4F718
```

