

Instrukcje arytmetyczne

IA32- rejestry

EAX

EBX

ECX

EDX

31

16

8

0

AH

BH

CH

DH

AL

BL

CL

DL

rejestry
ogólnego
przeznaczenia

ESI

EDI

SI

DI

rejestry
indeksowe

EBP

ESP

BP

SP

rejestry
wskaźnikowe

31

0

EF

FLAGS

flagi

31

0

EIP

IP

wskaźnik instrukcji

CS

DS

ES

SS

FS

GS

rejestry
segmentowe

(C) IISI d.KIK PCz 2019

Programowanie niskopoziomowe

2

Rejestr flag

15

0

OF

DF

IF

OF

DF

IF

OF

DF

IF

OF

DF

IF

OF

DF

IF

OF

bit	Skrot/wartosc	Opis	typ
0	CF	flaga przeniesienia (carry)	S
2	PF	flaga parzystosci (parity)	S
4	AF	flaga wyrównania (adjust)	S
6	ZF	flaga zera (zero)	S
7	SF	flaga znaku (sign)	S
10	DF	flaga kierunku (direction)	C
11	OF	flaga przepełnienia (overflow)	S

S: Znacznik stanu
C: Znacznik kontrolny
X: Znacznik systemowy
(C) IISI d.KIK PCz 2019

Programowanie niskopoziomowe

3

Instrukcje arytmetyczne

- ADD
- ADC
- ADCX
- ADOX
- SUB
- SBB
- MUL
- MULX
- IMUL
- DIV
- IDIV
- INC
- DEC
- NEG
- CMP

dodawanie całkowitoliczbowe

dodawanie z przeniesieniem

dodawanie z przeniesieniem bez znaku

dodawanie z przeniesieniem bez znaku

odejmowanie

odejmowanie z pożyczką

mnożenie bez znaku

mnożenie bez znaku

mnożenie ze znakiem

dzielenie bez znaku

dzielenie ze znakiem

inkrementacja (zwiększenie)

dekrementacja (zmniejszenie)

zmiana znaku

porównanie

(C) IISI d.KIK PCz 2019

Programowanie niskopoziomowe

4

Wpływa na flagi: OSZAPC

Instrukcja ADD

add cel, źródło

Dodaje zawartość źródła i celu, sumę umieszcza w miejscu przeznaczenia (cel).

cel:=cel+źródło

add eax, zmienna

add edx, [ebx+esi*4]

add rcx, rbx

Uwaga:

Jeśli źródło jest adresowane w trybie prostym może mieć do 32 bitów.

(C) IISI d.KIK PCz 2019

Programowanie niskopoziomowe

5

Wpływa na flagi: OSZAPC

Instrukcja ADC

adc cel, źródło

Dodaje zawartość źródła, celu i przeniesienia, sumę umieszcza w miejscu przeznaczenia (cel).

cel:=cel+źródło+CF

adc eax, zmienna

adc edx, [ebx+esi*4]

add rcx, rbx

Uwaga:

Jeśli źródło jest adresowane w trybie prostym może mieć do 32 bitów.

(C) IISI d.KIK PCz 2019

Programowanie niskopoziomowe

6

Wpływa na flagi: ----C
Wymagane: ADX

Instrukcja ADCX

adcx cel, źródło

Dodaje bez znaku zawartość źródła, celu i przeniesienia, sumę umieszcza w miejscu przeznaczenia (cel – rejestr 32|64 bitowy).

cel:=cel+źródło+CF

adcx eax, zmienna

adcx edx, [ebx+esi*4]

adcx rcx, rbx

(C) IISI d.KIK PCz 2019

Programowanie niskopoziomowe

7

Wpływa na flagi: O-----
Wymagane: ADX

Instrukcja ADOX

adox cel, źródło

Dodaje bez znaku zawartość źródła, celu i flagi przepełnienia, sumę umieszcza w miejscu przeznaczenia (cel – rejestr 32|64 bitowy).

cel:=cel+źródło+OF

adox eax, zmienna

adox edx, [ebx+esi*4]

adox rcx, rbx

(C) IISI d.KIK PCz 2019

Programowanie niskopoziomowe

8

Wpływa na flagi: OSZAPC

Instrukcja SUB

sub cel, źródło

Odejmuje zawartość źródła od celu, różnicę umieszcza w miejscu przeznaczenia (cel).

cel:=cel-źródło

sub ecx, zmienna

sub ebx, [ebx+esi*4]

sub rax, rdx

Uwaga:

Jeśli źródło jest adresowane w trybie prostym może mieć do 32 bitów.

(C) IISI d.KIK PCz 2019

Programowanie niskopoziomowe

9

Wpływa na flagi: OSZAPC

Instrukcja SBB

sbb cel, źródło

Odejmuje zawartość źródła od celu z uwzględnieniem pożyczki, różnicę umieszcza w miejscu przeznaczenia (cel).

cel:=cel-(źródło+CF)

sbb edx, zmienna

sbb eax, [ebx+esi*4]

sbb rax, rdx

Uwaga:

Jeśli źródło jest adresowane w trybie prostym może mieć do 32 bitów.

(C) IISI d.KIK PCz 2019

Programowanie niskopoziomowe

10

Wpływa na flagi: OxxxxC

Instrukcja MUL

mul źródło

Mnoży bez znaku zawartość akumulatora (al, ax, eax, rax) i źródła, wynik umieszcza w miejscu przeznaczenia (ax, dx:ax, edx:eax, rdx:rax). Flagi CF i OF są zerem, jeśli starsza połowa bitów wyniku jest równa zero.

wynik:=acc*źródło

mul zmienna

mul word ptr[ebx+esi*4]

(C) IISI d.KIK PCz 2019

Programowanie niskopoziomowe

11

Wpływa na flagi: -----
Wymaga BMI2

Instrukcja MULX

mulx cel1, cel2, źródło

Mnoży bez znaku zawartość rejestru edx|rdx i źródła, wynik umieszcza w rejestrach celu(cel1:cel2).

cel1:cel2:=e(r)dx*źródło

mulx ecx, ebx, [tab+esi*4]

mulx rcx, rbx, rax

(C) IISI d.KIK PCz 2019

Programowanie niskopoziomowe

12

Wpływa na flagi: OxxxxC

Instrukcja IMUL -1

imul źródło

Mnoży ze znakiem zawartość akumulatora (al, ax, eax, rax) i źródła, wynik umieszcza w miejscu przeznaczenia (ax, dx:ax, edx:eax, rdx:rax). Flagi CF i OF są zerem, jeśli iloczyn mieści się w młodszej połowie bitów wyniku.

wynik:=acc*źródło

imul zmienna

imul ecx

(C) IISI d.KIK PCz 2019

Programowanie niskopoziomowe

13

Wpływa na flagi: OxxxxC

Instrukcja IMUL -2

imul cel, źródło

Mnoży ze znakiem zawartość rejestru celu (16|32|64 bity) przez źródło, wynik umieszcza w miejscu przeznaczenia (cel). Flagi CF i OF są zerem, jeśli iloczyn mieści się w rejestrze celu.

cel:=cel*źródło

imul eax, zmienna

imul rdx, rcx

(C) IISI d.KIK PCz 2019

Programowanie niskopoziomowe

14

Wpływa na flagi: OxxxxC

Instrukcja IMUL -3

imul cel, źródło1, źródło2

Mnoży ze znakiem zawartość źródła1 (16|32|64 bity) przez źródło2 (stała), wynik umieszcza w miejscu przeznaczenia (cel). Flagi CF i OF są zerem, jeśli iloczyn mieści się w rejestrze celu.

cel:=źródło1*źródło2

imul eax, zmienna, 5

imul cx, dx, 77

Uwaga:

Jeśli źródło2 jest adresowane w trybie prostym może mieć do 32 bitów.

(C) IISI d.KIK PCz 2019

Programowanie niskopoziomowe

15

Wpływa na flagi: xxxxxx

Instrukcja DIV

div źródło

Dzieli bez znaku zawartość AX, DX:AX, EDX:EAX, RDX:RAX przez źródło, iloraz umieszcza w AL, AX, EAX, RAX a resztę w AH, DX, EDX, RDX.

div byte ptr zmienna

div ebx

(C) IISI d.KIK PCz 2019

Programowanie niskopoziomowe

16

Wpływa na flagi: xxxxxx

Instrukcja IDIV

idiv źródło

Dzieli ze znakiem zawartość AX, DX:AX, EDX:EAX, RDX:RAX przez źródło, iloraz umieszcza w AL, AX, EAX, RAX a resztę w AH, DX, EDX, RDX.

idiv byte ptr zmienna

idiv ebx

(C) IISI d.KIK PCz 2019

Programowanie niskopoziomowe

17

Wpływa na flagi: OSZAP

Instrukcja INC

inc cel

Zwiększa zawartość celu o 1.

inc zmienna

inc edx

inc rcx

(C) IISI d.KIK PCz 2019

Programowanie niskopoziomowe

18

Wpływa na flagi: OSZAPC

Instrukcja DEC

```
dec cel
```

Zmniejsza zawartość celu o 1.

```
dec zmienna
dec edx
dec r8
```

(C) IISI d.KIK PCz 2019

Programowanie niskopoziomowe

19

Wpływa na flagi: OSZAPC

Instrukcja NEG

```
neg cel
```

Zmienia znak celu w kodzie U2.

```
cel:=-cel
```

```
neg ax
neg byte ptr[ebx+esi*4]
neg r11
```

Flaga CF=0, tylko dla argumentu=0.

(C) IISI d.KIK PCz 2019

Programowanie niskopoziomowe

20

Wpływa na flagi: OSZAPC

Instrukcja CMP

```
cmp źródło1, źródło2
```

Porównuje zawartość źródła1 i źródła2, wynik nie jest zapamiętywany, tylko są ustawiane flagi.

```
źródło1-źródło2
```

```
cmp ax, zmienna
cmp edx, [ebx+esi*4]
cmp rcx, 123
```

Uwaga:

Jeśli źródło2 jest adresowane w trybie prostym może mieć do 32 bitów.

(C) IISI d.KIK PCz 2019

Programowanie niskopoziomowe

21

Wpływa na flagi: OSZAPC

Przykład – oblicz sumę kwadratów liczb w tablicy

```
mov eax, 0 ;wartość początkowa sumy
mov esi, eax ;indeks
mov ebx, offset wektor ;tablica liczb całkowitych
mov ecx, 123 ;licznik
petla: mov edx, [ebx+esi*4]
imul edx, edx ;kwadrat liczby
add eax, edx ;suma
inc esi
dec ecx
jnz petla ;wynik w eax
```

(C) IISI d.KIK PCz 2019

Programowanie niskopoziomowe

22

Wpływa na flagi: OSZAPC

Instrukcje arytmetyczne BCD

- DAA korekta upakowanego kodu BCD po dodawaniu
Decimal adjust after addition
- DAS korekta upakowanego kodu BCD po odejmowaniu
Decimal adjust after subtraction
- AAA ASCII korekta po dodawaniu
ASCII adjust after addition
- AAS ASCII korekta po odejmowaniu
ASCII adjust after subtraction
- AAM ASCII korekta po mnożeniu
ASCII adjust after multiplication
- AAD ASCII korekta przed dzieleniem
ASCII adjust before division

!!! Nie działają w trybie 64 bitowym !!!

(C) IISI d.KIK PCz 2019

Programowanie niskopoziomowe

23

Wpływa na flagi: OSZAPC

Instrukcja DAA

```
daa
```

Korekta upakowanego kodu BCD (w AL) po dodawaniu. Polega na dodaniu 6 najpierw do młodszej półbajty, a potem do starszej, jeśli ich zawartości były większe od 9 lub wystąpiło przeniesienie AF (CF)

```
daa
```

(C) IISI d.KIK PCz 2019

Programowanie niskopoziomowe

24

Wplywa na flagi: OSZAPC

Instrukcja DAS

das

Korekta upakowanego kodu BCD (w AL) po odejmowaniu. Polega na odjęciu 6 najpierw od młodszego półbajta, a potem od starszego, jeśli ich zawartości były większe od 9 lub wystąpiło przeniesienie AF (CF).

das

(C) IISI d.KIK PCz 2019

Programowanie niskopoziomowe

25

Wplywa na flagi: OSZAPC

Instrukcja AAA

aaa

Korekta nieupakowanego kodu BCD (w AL) po dodawaniu. Polega na dodaniu 6 do młodszego półbajta i 1 do AH, jeśli zawartość AL była większa od 9 lub wystąpiło przeniesienie AF.

aaa

(C) IISI d.KIK PCz 2019

Programowanie niskopoziomowe

26

Wplywa na flagi: OSZAPC

Instrukcja AAS

aas

Korekta nieupakowanego kodu BCD (w AL) po odejmowaniu. Polega na odjęciu 6 od młodszego półbajta i 1 od AH, jeśli zawartość AL. była większa od 9 lub wystąpiło przeniesienie AF.

aas

(C) IISI d.KIK PCz 2019

Programowanie niskopoziomowe

27

Wplywa na flagi: OSZAPC

Instrukcja AAM

aam

Korekta nieupakowanego kodu BCD (w AX) po mnożeniu. Polega na jednoczesnym wykonaniu:
AH=AL div 10
AL:=AL Mod 10.

aam

(C) IISI d.KIK PCz 2019

Programowanie niskopoziomowe

28

Wplywa na flagi: OSZAPC

Instrukcja AAD

aad

Korekta nieupakowanego kodu BCD (w AX) przed dzieleniem. Polega na jednoczesnym wykonaniu:
AL:=AH*10+AL.
AH=0

aad

(C) IISI d.KIK PCz 2019

Programowanie niskopoziomowe

29

Przykład – dodawanie liczb BCD

```
mov    al, 0
add    al, al           ;CF=0
petla: mov  al, [esi]    ;pobierz cyfrę źródła
        adc  al, ds:[edi] ;dodaj cyfrę celu z przeniesieniem
        aaa                     ;korekta
        mov  ds:[edi], al    ;zapamiętaj cyfrę
        inc  esi             ;następna cyfra
        inc  edi
        dec  ecx
        jnz  petla          ;CF nie zmieniło się od AAA!!!
```

(C) IISI d.KIK PCz 2019

Programowanie niskopoziomowe

30