

Vienuoliktoji paskaita

Pertraukimai, komandų atpažinimas

Julius Andrikonis
julius.andrikonis@mif.vu.lt
Matematikos ir informatikos fakultetas
Vilniaus universitetas

Trumpas priminimas

- N-osios pertraukimo apdorojimo procedūros adresas saugomas:
 - Segmentas – adresu $0000:[N*4+2]$
 - Poslinkis – adresu $0000:[N*4]$
- Komanda INT į steką įdeda:
 - SF – Požymių registro reikšmę
 - CS – grįžimo adreso segmentą
 - IP – grįžimo adreso poslinkį
- Grįžimo adresas – kokia komanda bus vykdoma grįžus iš pertraukimo

Pertraukimo apdorojimo procedūra

```
PROC ApdorokPertr
```

```
    ;Įdedame naudojamų registrų reikšmes į steką
```

```
    ;Atliekame darbą
```

```
    ;Atstatome registrų reikšmes
```

```
    IRET
```

```
ApdorokPertr ENDP
```

Procedūros adreso nustatymas

```
MOV ax, 0
```

```
MOV es, ax
```

```
MOV bx, 4
```

```
PUSH es:[bx]
```

```
PUSH es:[bx+2]
```

```
MOV dx, offset ApdorokPertr
```

```
MOV es:[bx], dx
```

```
MOV es:[bx+2], cs
```

Procedūros adreso atstatymas

POP es:[bx+2]

POP es:[bx]

Kitos vykdomos komandos nustatymas

MOV bp, sp

ADD bp, 12

MOV bx, [bp]

MOV es, [bp+2]

MOV dl, [es:bx]

Komandos atpažinimas

- **MOV reg <- bet.op.: 1011 wreg**
- **Kodas:**

MOV al, dl

AND al, 0F0h

CMP al, 0B0h

JE GeraKomanda

Bito _w išvedimas

- MOV reg <- bet.op.: 1011 wreg
- Kodas:

```
MOV  cl, 3
```

```
SHR  dl, cl
```

```
AND  dl, 1
```

```
ADD  dl, 30h
```

```
MOV  ah, 2
```

```
INT  21h
```

Testavimas

- Dalyba iš nulio:
 - Kyla dalinant iš nulio arba kai rezultatas netelpa jam skirtoje vietoje;
 - Steke išsaugo klaidą sukėlusios komandos adresą
- Žingsinio režimo pertraukimas:
 - Kyla po kiekvienos komandos, jei $TF=1$.
- Kontrolinio taško pertraukimas:
 - Kyla vykdant komandą INT 3
- Perpildymo pertraukimas:
 - Kyla vykdant komandą INTO, jei $OF=1$

TF nustatymas

PUSHF

PUSHF

POP ax

OR ax, 0100h

PUSH ax

POPF

NOP

;Testavimo kodas

POPF