VILNIAUS UNIVERSITETAS

MATEMATIKOS IR INFORMATIKOS FAKULTETAS

**Modelinės multiprograminės operacinės sistemos**

**projektas**

Darbą atliko 3 kurso 1 grupės studentai:

Antanas Norkus

Edvinas Vaitkūnas

Vilnius, 2011

# Reali ir virtuali mašinos

# Realios ir virtualios mašinos modeliai

**Reali mašina:**

C MODE

IC

R

R2

PTR

Centrinis procesorius

PI

SI

TI

Aukšto lygio kalbos procesorius.

HLP

1 Kanalas

2 Kanalas

Įvedimo įrenginys

Išvedimo įrenginys

Išorinė atmintis

Atminties puslapiavimo mechanizmas

SP

Vartotojo atmintis

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 000 |  |  |  |  |
| 001 |  |  |  |  |
| 002 |  |  |  |  |
| 003 |  |  |  |  |
|  | ... | | | |
| 253 |  |  |  |  |
| 254 |  |  |  |  |
| 255 |  |  |  |  |
| Supervizorinė atmintis 256 žodžių. | | | | |

SWAP

## Realios mašinos centrinis procesorius

* **PTR** – 4 baitų puslapių lentelės registras;
* **R** – 4 baitų bendro naudojimo registras;
* **IC** – 2 baitų virtualios mašinos programos skaitiklis;
* **C** – 1 baito loginis trigeris (reikšmės – T arba F);
* **MODE** – 1 baito registras, kurio reikšmė nusako procesoriaus darbo režimą (vartotojas, supervizorius);
* **SP** – 3 baitų steko rodyklė. Šis registras nurodo poslinkį steke;
* **PI** – 1 baito programinių pertraukimų registras;
* **SI** – 1 baito supervizorinių pertraukimų registras;
* **TI** – 2 baitų taimerio pertraukimo registras;
* **SWAP** – swapnimui skirtas failas (dydis 64 blokų).
  + 1. **Virtuali mašina:**

Virtuali atmintis

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 00 |  |  |  |  |
| 01 |  |  |  |  |
| 02 |  |  |  |  |
| 03 |  |  |  |  |
|  | ... | | | |
| 29 |  |  |  |  |
| 30 |  |  |  |  |
| 31 |  |  |  |  |

R

IC

C

Virtualus procesorius

Virtualus įvedimo įrenginys

Virtualus išvedimo įrenginys

SP

R2

## 1.3 Virtualios mašinos centrinis procesorius

* R,R2 – 4 baitų bendro naudojimo registras;
* IC – 2 baitų virtualios mašinos programos skaitiklis;
* C – 1 baito loginis trigeris (reikšmės – T arba F);
* SP – 3baitų steko rodyklė.

**1.4 Procesoriaus komandos**

* **PUSHxy.** Isteką(steko viršūnė saugoma registre SP) įkeliama duomenys iš nurodytos atminties (adresas x\*256+y).
* **POPxy.** Iš steko (steko viršūnė registre SP) duomenys įkeliami į atmintį (adresas x\*256+y ).
* **ADD**. Sudedama reikšmė esanti registre R su reikšme esančia registre R2. Rezultatas įrašomas į registrą R.
* **SUB**. Iš reikšmės esančios registre R atimama reikšmė esanti registre R2. Rezultatas įrašomas į registrą R.
* **MUL.** Sudauginama reikšmė esanti registre R su reikšme esančia registre R2. Rezultatas įrašomas į registrą R2.
* **DIV.** Reikšmė esanti resistre R dalinama iš registro reikšmės R2. Rezultatas išsaugomas į registrą R2.
* **CMP.** R<>R2 -> C=1, R=R2 -> C=0**.**
* **GTxy.** Valdymas perduodamas adresu 256\*x+y**.**
* **GZxy**. Jei C=0, tai valdymas yra perduodamas adresu 256\*x+y.
* **GFxy.** Jei C=1, tai valdymas yra perduodamas adresu 256\*x+y.
* **RDxy.** Iš įvedimo srauto perskaito 256 žodžių ir įrašo juos į ląsteles [x \*256 + i ], kur i = 0.. 9. Operandas “y” reikšmės neturi.
* **PDxy** Išsiunčia išvedimui 256 žodžių srautą iš atminties ląstelių [x \*256 + i ], kur i = 0.. 9. Operandas y reikšmės neturi.
* **HALT –** Vartotojo programos vykdymo pabaiga.

**1.5 Atminties organizavimas**

Vartotojo atmintis skirta virtualių mašinų atmintims bei puslapių lentelėms laikyti. Vartotojo atmintį sudaro 256\*256 žodžių po 4 baitus. 256 žodžių laikysime bloku. Taigi vartotojo atmintis lygi 256–čiai blokų, sunumeruotų nuo 0 iki 255.

## 1.6 Virtualios mašinos atmintis

Kiekvienai virtualiai mašinai yra skiriama 32 vartotojo atminties blokų. Tuose 32-juose blokuose (8192 žodžių) turi tilpti užduoties programa. Kiekvienas virtualios atminties blokas turi virtualų ir realų adresą. Virtualiais adresais operuoja virtuali mašina, realiais – reali mašina. Ryšiai tarp virtualaus ir realaus adreso nusakomi puslapių lentelėmis.

## 1.7 Puslapiavimo mechanizmas

Realios mašinos vartotojo atminties dydis 256 blokų. Kiekvienai naujai sukurtai virtualiai mašinai reikia skirti 32 takelių iš tų 256.

Virtuali mašina gali sužinoti kokio nors jai priklausančio takelio realų adresą naudodamasi puslapiavimo mechanizmu.

Sakykime kuriama nauja virtuali mašina, kuriai parinkome takelius su numeriais: 2, 5, 6, 8, ..., 13, 20, 22, 23. Šie takeliai virtualios mašinos adresų erdvėje sunumeruoti nuo 0 iki 32, t.y. 2 takelis jai yra nulinis, 5 takelis jai yra antras, o 23 takelis – 31-as. Išlaikyti sąryšiams tarp realių ir virtualių takelių adresų naudosime puslapių lentelę. Puslapių lentelė – tai vienas blokas (t.y. 256 žodžių). Kiekvieno žodžio eilės numeris atitiks virtualios mašinos takelio numerį, ir jame (žodyje) bus laikomas realus to takelio numeris. Pavyzdžiui:

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 0 | 1 | 2 | 3 | ... | 28 | 29 | 30 | 31 |
| 2 | 5 | 6 | 8 | ... | 13 | 20 | 22 | 23 |

Virtuali mašina, norėdama sužinoti realų takelio adresą, kreipiasi į savo puslapių lentelę ir nuskaito reikšmę, esančią žodyje su takelio numeriu. Puslapių lentelė taip pat yra atminties takelis, ir šis takelis taip pat turi savo realų numerį. Todėl registro PTR reikšmė – puslapių lentelės adresas. Kiekviena virtuali mašina, prieš pradėdama darbą, nustatys šį registrą jai reikalinga reikšme.

PTR – 4 baitų registras. Pažymėsime PTR reikšmę a0a1a2a3, kur ai yra baitai. Baitai a0 ir a1 nenaudojami, a2 \* 256 + a3 žymi puslapių lentelės adresą. Formulė, kuri virtualiam adresui x1x2 gražina realų adresą:

Realus adresas = 256 \* [256 \* (256 \* a2 + a3) + x1] + x2.

Operaciniai sistemai pradedant darbą, supervizorinėje atmintyje išsaugomas SWAP failo adresas. SWAP failas sukuriamas tik 1, kuris operacinės sistemos darbo pabaigoje yra išvalomas.

Vykstant procesui swapinama jei procesui trūksta virtualios atminties, tai 64 blokus perkeliame į išorinę atmintį (SWAP failą). Jeigu SWAP faile nėra pakankamai laisvos vietos – įvyksta programinis pertraukimas.

**1.8 Pertraukimų mechanizmas**

Pertraukimas – tai signalas apie įvykusį įvykį. Kiekvienas pertraukimas turi savo identifikaciją. Pertraukimas savaime nepertraukia sistemos darbo. Pertraukimą sistema turi aptikti ir atitinkamai sureaguoti.

Modelyje bus realizuoti trijų tipų pertraukimai – programiniai, supervizoriniai ir taimerio. Programinių pertraukimų registras yra PI, supervizorinių pertraukimų registras – SI, taimerio - TI. Programiniai pertraukimai kyla vykdant virtualią mašiną, bandant įvykdyti kokį nors neleistiną veiksmą arba nuskaičius neleistiną reikšmę. Supervizoriniai pertraukimai kyla virtualiai mašinai norint įvykdyti veiksmą, kuris gali vykti tik supervizoriaus režime. Pertraukimai gali būti aptikti tik vartotojo režime, supervizoriniame režime centrinio procesoriaus darbo pertraukti negalima.

**Pertraukimai kils šiais būdais:**

PI:

1 – atminties apsauga;

1. – neegzistuoja operacijos kodas.
2. – neužtenka atminties išoriniame diske.

SI:

1 – RD;

2 – PD;

3 – HALT.

TI:

0 – kai laiko skaitliukas pasieks 0.