**Operacinės sistemos,**

**reali ir virtuali mašinos**

Turinys

Add Headings (Format > Paragraph styles) and they will appear in your table of contents.

* Virtuali Mašina 2
  + Virtualaus procesoriaus registrai 2
  + Virtualios mašinos atmintis 2
* Reali Mašina 3
  + Realios mašinos registrai 3
  + Realios mašinos atmintys 4
* Komandų sistema 4
* Procesoriaus darbo algoritmas 6
* Pertraukimai 7
* Puslapiavimo mechanizmas 8
* Kanalų įrenginys 9
* Taimerio mechanizmas 9
* Įvedimo, išvedimo įrenginiai 9

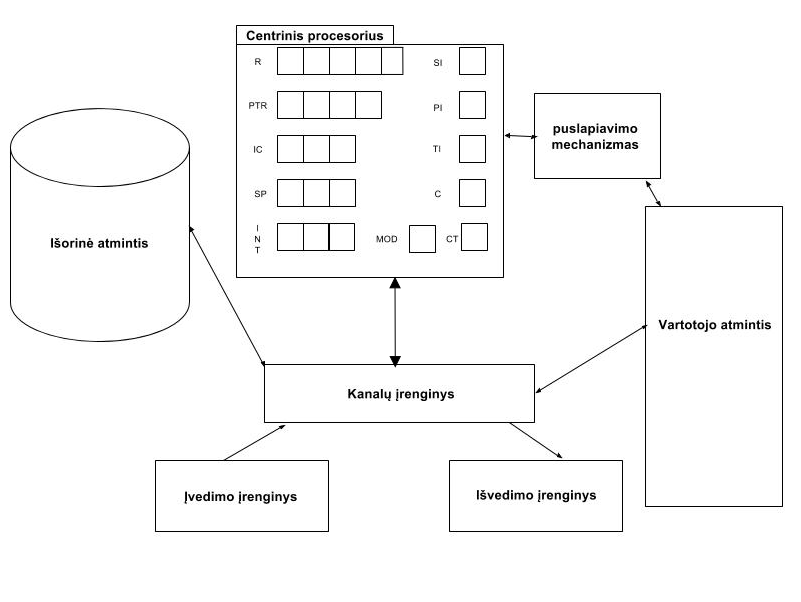
**Virtuali Mašina**



1. **Virtualaus procesoriaus registrai**
   1. R – 5 baitų bendro naudojimo registras
   2. C – 1 baito požymio registras
   3. IC – 3 baitų komandų skaitliuko registras
   4. CT – 1 baito ciklo skaitliuko registras
   5. SP – 3 baitų steko viršūnės registras
2. **Virtualios mašinos atmintis**

Kiekvienai virtualiai mašinai yra skiriama 10 vartotojo atminties blokų. Tuose dešimtyje blokų (100 žodžių) turi tilpti užduoties programa. Kiekvienas virtualios atminties blokas turi virtualų ir realų adresą.

**Reali Mašina**

****

1. **Realios mašinos registrai**
   * MOD - 1 baito procesoriaus būseną nusakantys registras
   * R - 4 baitų bendro naudojimo registras
   * PTR - 4 baitų puslapių lentelės registras
   * IC - 3 baitų komandų skaitliuko registras
   * SP - 3 baitų steko viršūnės registras
   * C - palyginimo požymių registras
   * INT - 3 baitų pertraukimo procedūros adreso registras
   * CT – 3 baito ciklo skaitliuko registras
   * PI - 3 baito programinių pertraukimų registras
   * SI - 3 baito supervisorinių pertraukimų registras
   * TI - 3 baito taimerio mechanizmo registras
2. **Realios mašinos atmintys**

Atmintis – įrenginys informacijai saugoti. Mūsų reali mašina turi dviejų rūšių atmintis: vartotojo ir išorinę.

**Vartotojo atmintis**: skirta virtualių mašinų atmintims, puslapių ir pertraukimų vektoriaus lentelėms laikyti. Mes apibrėšime vartotojo atmintį taip: lentelės dydis – 1000 žodžių po 5 baitus (1 baitas – 8 bitai). 10 žodžių laikysime bloku (takeliu). Taigi vartotojo atmintis lygi 100 blokų, sunumeruotų nuo 0 iki 99, arba 1000 žodžių, sunumeruotų nuo 0 iki 999.

**Išorinė atmintis**: šioje atmintyje saugomi programų tekstai.

Procesorius darbą su atmintimis valdo naudodamas kanalų įrenginį.

Šios atminties dydis yra pastovus – 400 blokų arba 4 000 žodžių (sunumeruotų nuo 0 - 3999) po 5 baitus (1 baitas – 8 bitai).re

**Komandų sistema**

**Komandų ilgis – 1 žodis (išskyrus komandą CHNGR, kurios komandos ilgis 2 žodziai) .**

Komandos kurias gali vykdyti tiek VM tiek RM

* CHNGR - Žodis esantis adresu [IC] yra komandos kodas, žodis esantis adresu [IC + 1] yra reikšmė kuri priskiriama registrui R. IC = IC + 1
* LRxyz – atminties žodžio, kurio adresas x\*100 + y \* 10 + z turinio kopijavimas į registrą R. R:= [x\*100 + y \* 10 + z]. x, y, z = 0..9.
* SRxyz – registro R reikšmė įrašoma į atminties žodžio, kurio adresas x\*10 + y. [x\*100 + y\*10 + z]:= R, x, y, z = 0..9.
* LOreg - Į registra R įrašoma registre reg buvusi reikšmė. R = reg,  
  reg = {MOD, PTR, IC, SP, C, INT, PI, SI, TI, CT}. Dirbant vartotojo rėžime reg = {IC, C, R, SP, CT}
* ADxyz – prie registro R reikšmės pridedama atminties ląstelės, kurios adresas x\*10 + y, reikšmė. R:= R + [x\*100 + y\*10 + z ]. x, y, z = 0..9.
* SBxyz – iš registro R reikšmės atimamas žodis, kurio adresas x\*100 + y\*10 + z, reikšmė. R:= R - [x\*100 + y\*10 + z]. x, y, z = 0..9.
* MPxyz – registro R reikšmė yra sudauginama su atminties ląstelės, kurios adresas x\*10+y, reikšme. rezultatas padedamas į registrą R. R = [x \* 100 + y \* 10 + z] \* R. x, y, z = 0..9.
* DIxyz - registro R reikšmė dalinama iš atmintimties žodždio [x\*100 + y \* 10 + z]. Į registrą R įrašoma sveikoji dalis.
* CRxyz - jei R = [x\*100 + y\*10 + z], tada C = „T“, kitu atveju C = „F“
* RLxyz - jei R < [x\*100 + y\*10 + z], tada C = „T“, kitu atveju C = „F“
* RGxyz – jei R > [x\*100 + y\*10 + z], tai registrui C priskiriama reikšme „T”, kitu atveju C := „F“
* CZreg – jei reg = 0, tada C = „T“, kitu atveju C = „F“. Reg = { MOD, PTR, IC, SP, C, INT, PI, SI, TI, CT }.
* JCxyz - jei registro C reikšmė = T tai valdymas perduodamas x\*100 + y\*10 + z adresu. IC = x\*100 + y\*10 + z. x, y, z = 0..9
* JPxyz – besąlyginis komandos perdavimas adresu [x\*100 + y\*10 + z];
* CAxyz – komanda call, išaugoma mašinos būsena (į steką išsaugomos reikšmės šia tvarka MOD, PTR C, R, IC, SP, CT) ir IC priskiriama xyz. IC =x\*100 + y\*10 + z. x, y, z = 0..9, jeigu dirbame vartotojo rėžime į steka saugome šiuos registrus C, R, IC, SP, CT.
* PUreg - į [SP] iššaugomas nurodytas registras, steko viršūnė sumažinama vienetu. Rašant reikšmę, ji praplečiama nuliais priekyje jeigu to reikia, jeigu ši komanda vykdoma vartotojo režimu galimi registrai yra: {C, IC, R, SP, CT}.
* POreg – iš [SP] reikšmė įrašoma į nurodytą registrą, steko viršūnė padidinama vienetu. Jeigu reg ={ TI, SI, PI, INT, C, SP, IC, MOD, CT}, tokiu atveju iš pradžios yra nuvaloma tiek baitu kiek reikia suvienodinti atminties ir registro dydi, jeigu ši komanda vykdoma vartotojo režimu galimi registrai yra: {C, IC, R, CT, SP}..
* RETRN – Grįžimo iš proceduros komanda, iš steko grąžinamos registrų reikšmes šia tvarka CT, SP, IC, R, C, PTR, MOD, jeigu dirbame vartotjo režime gražinamos šios reikšmės CT, SP, IC, R, C.
* SYxyz, Programa praneša operacinei sistemai apie veiksmus kurių negali atlikti pati. Supervisorinių pertraukimu registras nurodyta reikšme SI = x \* 100 + y + \*10 + z. x, y, z = 0..9.
* LPxyz – ciklo komanda, sumažina ciklo registro reikšmę vienetu, jeigu CT != 0, vykdomas šuolis į ciklo pradžią JPxyz. x, y, z = 0..9.

Komandos kurias gali vykdyti tik reali mašina

* CHNGM- keičiama mašinos būsena, IF MOD = 1 THEN MOD = 0 ELSE MOD = 1.
* PIxyz – nustatoma registro PI reikšmė nurodytu skaičiumi. PI = x,  
  x = 0..9.
* TIxyz – nustatoma registro TI reikšmė nurodytu skaičiumi. TI = x,  
  x = 0..9.
* PTRxy – nustatoma registro PTR reikšmė, kuri rodys puslapių lentelės bloko numerį. PTR = x\*10+y.  
  x, y = 0..9.
* SPxyz – nustatomas steko viršunės adresas. SP = x\*100+y\*10+z,  
  x, y, z = 0..9.
* INxyz – nustatomas pertraukimų registro adresas. INT = x \* 100+y \* 10 + z. x, y, z = 0..9.
* START - pradedama vykdyti virtuali mašina: išsaugoma realios mašinos busena, virtualios mašinos komandu skaitliukas nunulinamas IC = 0.
* CALLI - kviečiamas pertraukimas, į steką išsaugomos registrų reikšmės šia tvarka MOD, PTR C, R, IC, SP, CT, IC = INT.
* IRETN - vykdomas grįžimas po pertraukimo, grąžinamos buvusios registrų reikšmės iš steko šia tvarka: CT, SP, IC, R, C, PTR, MOD
* BSxyz - nustatomas takelio numeris iš kurio kopijuosime. Kanalų įrenginio registras SB = x \* 100 + y \*10 + z. x, y, z = 0..9
* DBxyz - nustatomas takelio numeris į kurį kopijuosime numeris. Kanalų įrenginio registras DB = x \* 100 + y \*10 + z. x, y, z = 0..9
* STxyz – nustatomas objekto numeris, iš kurio kopijuosime. Kanalų įrenginio registras ST = x \* 100 + y \* 10 + z. x, z, y = 0..9
* DTxyz - nustatomas objekto numeris į kurį kopijuosime. Kanalų įrenginio registras DT = x \* 100 + y \* 10 + z. x, z, y = 0..9
* SZxyz – nurodoma, kiek žodžių bus kopijuojama iš/į objektą(-o). SZ = x \* 100 + y \* 10 + z. x, y, z = 0..9
* XCHGN – vykdomas apsikeitimas duomenimis pagal nustatytus kanalų įrenginio registrus.

**Procesoriaus darbo algoritmas**

1. Procesorius gali dirbti dviem rėžimais: Vartotojo ir supervisoriniame. ( MOD = 1 reiškia, jog dirbama vartotojo, o MOD = 0 – supervisoriaus režime ). Priklausomai nuo rėžimo kuriame dirbame keičiasi aibė komandų kurias gali vykdyti procesorius.
2. Procesorius pasiruošęs vykdyti komandą, nurodytą IC registre. Nuskaitęs komandą, procesorius padidina IC registro reikšmę vienetu. Taip pat, TI reikšmė mažinama vienetu.
3. Vykdoma instrukcija, galimos kelios vykdymo baigtys :
   1. Komanda atpažįstama komandų aibėje – komanda įvykdoma, jeigu reikia nuskaitomi argumentai.
   2. Komanda praneša apie supervisorini pertraukima.
   3. Komanda neatpažįstama arba negalima vykdyti, nustatomas programinio pertraukimo registras.
4. Tikrinami pertraukimų registrai. IF ((SI + PI) = 0 & TI != 0) THEN GOTO1 ELSE pertraukimų apdorotojas (plačiau prie pertraukimų) GOTO1.

**Pertraukimai**

Programinių pertraukimų registras yra PI, supervizorinių registras – SI, taimerio – TI.

Supervisoriniai pertraukimai, vartotojo rėžime, sukuriami SYxyz komandos pagalba,

* Supervisoriniai pertraukimai
  + SI = 1 - virtualios mašinos prašymas gauti duomenis iš įvedimo įrenginio, registre R esanti reikšmė traktuojama: pirmi du baitai nurodo, kiek žodžių bus įrašoma, sekantys du baitai nurodo adresą virtualios mašinos atmintyje.
  + SI = 2 - virtualios mašinos prašymas siųsti duomenis į išvedimo įrenginį. Registre R esanti reikšmė traktuojama: pirmi du baitai nurodo, kiek žodžių bus įšvedama, sekantys du baitai nurodo adresą virtualios mašinos atmintyje.
  + SI = 3 - virtualios mašinos prašymas išskirti papildomos atminties, tiek blokų kokia reikšmė yra registre R, ne daugiau kaip 10 bloku.
  + SI = 4 - virtualios mašinos pranešimas apie darbo pabaigą.
* Programiniai pertraukimai:
  + PI = 1 – atminties apsaugos pažeidimas.
  + PI = 2 – neleistinas operacijos kodas.
  + PI = 3 – atminties trūkumas
  + PI = 4 – neleistinas prisk
* Esant TI = 0, bus iššaukiamas taimerio pertraukimas.

Aptikus pertraukimą išsaugoma esamos mašinos būsena steke (MOD, PTR C, R, IC, SP, CT), jeigu reikia pakeičiamas režimas, kviečiamas pertraukimų apdorotojas.

**Puslapiavimo mechanizmas**

Realios mašinos vartotojo atmintis siekia 100 takelių (arba blokų). Kiekvienai naujai sukurtai virtualiai mašinai maksimaliai galima skirti 10 takelių iš 100. Jie gali būti parinkti bet kokiu būdu. Klausimas: kaip virtuali mašina gali sužinoti kokio nors jai priklausančio takelio realų adresą? Tam naudosime puslapiavimo mechanizmą.

Puslapiavimo mechanizmo esmė: sakykime, kuriama nauja virtuali mašina. Jai reikia dešimt takelių atminties. Mes parinkome takelius su numeriais: 1, 3, 5, 7, 9,11, 13, 15, 17, 19. Šiais takeliais naudosis virtuali mašina. Pati virtuali mašina šiuos takelius mato sunumeruotus nuo 0 iki 9. t.y. 1 takelis jai yra nulinis, 5 takelis jai yra antras, o 19 takelis – devintas. Kaip išlaikyti sąryšius tarp realių ir virtualių takelių adresų? Naudosime puslapių lentelę. Puslapių lentelė – tai vienas takelis (t.y. 10 žodžių). Kiekvieno žodžio eilės numeris atitiks virtualios mašinos takelio numerį, ir jame (žodyje) bus laikomas realus to takelio numeris.

|  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Realus takelio adresas | 1 | 3 | 5 | 7 | 9 | 11 | 13 | 15 | 17 | 19 |
| Virtualaus takelio adresas | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 |

Taigi, dabar virtuali mašina, norėdama sužinoti realų takelio adresą, kreipiasi į savo puslapių lentelę ir nuskaito reikšmę, esančią žodyje su takelio numeriu. Pvz. penkto virtualios mašinos takelio realus numeris yra 11. Dabar turi išaiškėti procesoriaus registro PTR prasmė. Juk puslapių lentelė taip pat yra atminties takelis. Ir šis takelis, be abejo, 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.

Dabar galime apžvelgti patį registrą PTR. Tai 4 baitų registras. Simboliškai pažymėsime PTR reikšmę a0a1a2a3, kur ai yra baitai. Baitas a0 yra nenaudojamas, a1 puslapių lentelės ilgis atėmus 1 (arba nenaudojamas. Tai priklauso nuo puslapiavimo mechanizmo realizacijos), o štai a2 \* 10 + a3 žymi puslapių lentelės adresą. Dabar galime pateikti formulę, kuri virtualiam adresui x1x2 gražina realų adresą:

**Realus adresas = 10\*[10\*(10 \* a2 + a3) + x1] + x2**

**Kanalų įrenginys**

Kanalų įrenginys leidžia dirbti su atmintimis. Priklausomai nuo nustatytų registrų kanalų įrenginys gali vykdyti apsikeitimą duomenimis visomis galimomis kryptimis.Veiksmai su kanalų įrenginiu atliekami tik supervizoriaus režime. Dabar bus pateikta kanalų įrenginio vartotojo sąsaja:

Kanalų įrenginio registrai:

* SB: Takelio, iš kurio kopijuosime numeris.
* DB: Takelio, į kurį kopijuosime numeris
* ST: Objekto, iš kurio kopijuosime, numeris
  + 1. Vartotojo atmintis;
  + 2. Išorinė atmintis;
  + 3. Įvedimo srautas;
* DT: Objekto, į kurį kopijuosime, numeris
  + 1. Vartotojo atmintis;
  + 2. Išorinė atmintis;
  + 3. Išvedimo srautas;
* SZ - žodžių kiekis kiek kopijuosime

Kartu kanalų įrenginys turi komandą XCHG, tačiau neturi procesoriaus, kuris galėtų ją įvykdyti. Šią komandą vykdo centrinis procesorius, taigi, šis kanalų įrenginys nėra lygiagrečiai su centriniu procesoriumi veikianti aparatūra.

Procesas, norėdamas pasinaudoti kanalų įrenginiu, turi nustatyti kanalų įrenginio

registrus ir tada įvykdyti komandą XCHG.

**Taimerio mechanizmas**

Laikysim, kad įvedimo/išvedimo instrukcijos atliekamos per 3 laiko momentus, visos kitos per 1 laiko momentą. Dabar apie veikimo principą.

Pradedant virtualios mašinos užduoties vykdymą TI registro reikšmė nustatoma tam tikrai reikšmei. Tarkim TI = 10. Įvykdžius eilinę instrukciją TI reikšmė mažinama priklausomai nuo to per kiek laiko momentų ši instrukcija yra atliekama.

**Įvedimo, išvedimo įrenginiai**

Įvedimo - klaviatūra, Išvedimo - Monitorius