Vilniaus universitetas

Matematikos ir informatikos fakultetas

Interaktyvios operacinės sistemos projektas

Rengė: Jokūbas Beivydas

Justas Janušis

**Anotacija**

Šiame darbe pateikiamas operacinių sistemų kurso 1-as laboratorinis darbas – “Operacinės sistemos projektas”. Šiame dokumente aprašyta reali mašina, kuriai yra kuriama operacinė sistema, nusakytos jos komponentės.

**Darbą atliko**

Jokūbas Beivydas

*El. Paštas:* [*jokubas.beivydas @mif.vu.lt*](mailto:vaidas.adomauskas@mif.vu.lt)

Justas Janušis

*El. Paštas:* [*justas.janusis @mif.vu.lt*](mailto:asta.murauskaitė@mif.vu.lt)

Turinys

[**Anotacija** 2](#_Toc34383370)

[1 Realios mašinos aprašymas 4](#_Toc34383371)

[1.1 Realios mašinos komponentės 4](#_Toc34383372)

[1.1.1 Centrinis procesorius 4](#_Toc34383373)

[1.1.2 Atmintis 5](#_Toc34383374)

[1.1.3 Įvedimo išvedimo įrenginiai 5](#_Toc34383375)

[1.1.4 Išorinė atmintis 5](#_Toc34383376)

[1.1.5 Kanalų įrenginys 5](#_Toc34383377)

[1.1.6 Puslapiavimo mechanizmas 5](#_Toc34383378)

[1.1.7 Pertraukimų mechanizmas 7](#_Toc34383379)

[1.1.8 Taimeris 8](#_Toc34383380)

[2. Virtualio mašinos modelis 8](#_Toc34383381)

[VM atmintis 9](#_Toc34383382)

[VM CPU 9](#_Toc34383383)

# Realios mašinos aprašymas

Reali mašina – tai kompiuteris, kuriam yra kuriama operacinė sistema, aprašoma šiame projekte. Šiame darbe pateikta realios mašinos architektūra iš esmės remiasi Šou modeliu.

## Realios mašinos komponentės

### Centrinis procesorius

Centrinio procesoriaus paskirtis skaityti komandą iš atminties ir ją vykdyti. Vienu metu yra skaitoma ir vykdoma tik viena komanda. Procesorius gali dirbti dviem režimais: supervizoriaus ir vartotojo. Supervizoriaus režime komandos, iš supervizorinės atminties, yra apdorojamos betarpiškai aukšto lygio kalbos procesoriaus HLP (HLP – bet kuris aukšto lygio kalbos procesorius). Tiksliau supervizoriaus režime procesorius vykdo tas komandas ir aptarnauja tuos procesus, kurie yra skirti pačios operacinės sistemos reikmėms, o ne vartotojo programoms. Vartotojo režime HLP vykdo užduočių programas. Vykdydamas užduoties programą HLP imituoja virtualios mašinos procesorių.

Centrinio procesoriaus registrai:

* PTR – 4 baitų puslapių lentelės registras. Kiekviena virtuali mašina prieš pradėdama darbą nustato šį registrą reikalinga reikšme. Pirmasis ir antrasis registro baitai yra nenaudojami
* , o trečias ir ketvirtas baitai nurodo puslapių lentelės adresą virtualioje atmintyje.
* A ir B – 4 baitų bendro naudojimo registrai
* PC – 2 baitų virtualios mašinos programos skaitiklis
* C – 1 baito loginis trigeris
* MODE – registras nusakantis procesoriaus darbo režimą (vartotojo arba supervizoriaus)
* PI – programinių pertraukimų registras. Šiame registre įvykus programiniam pertraukimui yra išsaugoma reikšmė, nusakanti koks įvyko pertraukimas ir kas jį sukėlė. Įvykus programiniam pertraukimui PI reikšmė lygi:
  + 1 kai bandoma naudoti neteisinga adresą,
  + 2 reiškia neteisingą operacijos kodą,
  + 3 – neteisingas priskyrimas,
  + 4 – perpildymas (overflow).
* SI – supervizorinių pertraukimų registras. Šiame registre įvykus supervizoriniam pertraukimui yra išsaugoma reikšmė nusakanti koks įvyko pertraukimas ir kas jį sukėlė. Supervizorinį pertraukimą gali iššaukti virtuali mašina bandydama įvykdyti veiksmą, kuris gali būti įvykdytas tik supervizoriniame režime. Supervizorinį pertraukimą iššaukti gali operacijos GD, PD ir HALT. Įvykus supervizoriniam pertraukimui registre SI bus išsaugoma reikšmė:
  + 1 vykdant operaciją GD;
  + 2 vykdant komandą PD,
  + 3 vykdant komandą HALT,

Įvykus pertraukimui pertraukimų registruose išsaugoma reikšmė, o įvykęs pertraukimas pastebimas įvykdžius komandą test(), kuri patikrina pertraukimų registrus.

* TI – taimerio registras. Šis registras yra proceso naudojimosi centriniu procesoriumi laiko apribojimam, t.y. kad vienas procesas nebūtų per ilgai užėmęs procesoriaus. Esant TI registro reikšmei 0 bus iššaukiamas taimerio pertraukimas.

### Atmintis

Skirstoma į supervizorinę ir vartotojo. Supervizorinė atmintis yra skirta pačiai operacinei sistemai. Ten laikomi sisteminiai kintamieji, sisteminiai procesai, resursai ir mikroprogramos. Vartotojo atmintis yra skirta vartotojo programų poreikiams, t.y. atmintyje laikomos virtualių mašinų puslapių lentelės ir jų atmintis. Realios mašinos vartotojo atmintis yra 64 blokų. Blokas tai 16 žodžių po 4 baitus.

### Įvedimo išvedimo įrenginiai

Realios mašinos įvedimo įrenginys klaviatūra, iš kurios skaitoma po simbolį, o paspaudus klavišą „enter“ įvesta seka pasiunčiama į įvedimo srautą. Išvedimo įrenginys yra monitorius, jame atvaizduojama į išvedimo srautą patekusi eilute.

### Išorinė atmintis

Išorinė atmintis bus realizuojama failu kietame diske. Iš ten bus nuskaitomos vartotojo programos. Laikysime, kad realios mašinos išorinės atmintis 1024 blokų, t.y. 16384 žodžių po 4 baitus.

### Kanalų įrenginys

Kanalų įrenginys yra skirtas darbui su atmintimis ir išvedimo, įvedimo įrenginiais. Priklausomai nuo jo registrų reikšmių jis gali vykdyti apsikeitimą duomenimis visomis kryptimis, pavaizduotomis realios mašinos modelio schemoje. Procesorius su kanalų įrenginiu gali dirbti tik būdamas supervizoriniame režime. Kanalų įrenginys turi komandą XCHG, tačiau neturi procesoriaus, kuris ją galėtu vykdyti. Todėl šis įrenginys nėra lygiagrečiai su procesoriumi veikianti aparatūra.

Kanalų įrenginio registrai:

* SB – takelio, iš kurio kopijuosime, numeris
* DB – takelio, į kurį kopijuosime, numeris
* ST – Objekto, iš kurio kopijuosime, numeris
  + Vartotojo atmintis
  + Supervizorinė atmintis
  + Išorinė atmintis
  + Įvedimo srautas
* DT – Objekto, į kurį kopijuosime, numeris
  + Vartotojo atmintis
  + Supervizorinė atmintis
  + Išorinė atmintis
  + Išvedimo srautas

### Puslapiavimo mechanizmas

Virtualiai mašinai yra išskiriama 16 blokų (arba puslapių, bet toliau blokų). Tuos blokus kiekviena VM susinumeruoja nuo 0 iki 15, tačiau jai taip pat reikia žinoti bei realius šių blokų adresus. Tam yra naudojamas puslapiavimo mechanizmas.  
  
Taigi realius bloko numerius talpinsime į puslapių lentelę. VM puslapių lentelei bus išskiriamas 1 blokas vartotojo atminty. Lentelėje kiekvieno žodžio eilės numeris atitiks VM bloko numerį ir jame bus laikomas realus to bloko numeris.

Reikia nepamiršti, kad puslapių lentelė taip pat saugoma atmintyje, ir kad VM žinoti kur yra būtent jos puslapių lentelėje naudosime registrą PTR.  
  
PTR yra 4 baitų ir simboliškai žymėsime taip a0a1a2a3. O x1x2 virtualius adresas.

* a0 – nenaudojamas.
* a1 – nenaudojamas.
* 16\*a2+a3 – puslapių lentelės bloko numeris vartotojo atmintyje.
* 16\*(16\*a2+a3) – puslapių lentelės bloko adresas.
* 16\*(16\*a2+a3) + x1 – bloko x1 adresas puslapių lentelėje. Jame saugomas bloko numeris į kurį atvaizduotas yra x1 blokas VM.
* 16\*[16\*(16\*a2+a3) + x1] – VM bloko x1 realus bloko adresas.
* 16\*[16\*(16\*a2+a3) + x1] + x2 – realus adresas atitinkantis virtualų adresą x1x2.

A close up of text on a white background

Description automatically generated

**Realios mašinos modelis**

Taimerio mechanizmas

Skirtas užduotims suderinti. Yra sakoma, kad užduotis negali trukti ilgiau nei tam tikrą T laiko momentų. Susitarsime, jog išvedimo / įvedimo instrukcijos reikalauja 3 laiko momentų, o kitos 1 ar 2. Taigi kai VM pradeda darbą, speciali supervizorinės atminties ląstelė TI yra nustatoma tam tikrai reikšmei. Tarkime 50, tuomet kai yra įvykdoma instrukcija, TI yra mažinamas priklausomai nuo to kiek laiko momentų reikia instrukcijai. Kai TI tampa lygus nuliui, mikrokomanda Test() aptinka taimerio pertraukimą. TI reikšmę galima nustatyti ar pakeisti supervizoriaus rėžime

Pertraukimų mechanizmas

### Pertraukimų mechanizmas

Tai tam tikri signalai apie specialius įvykius. Gali būti aptikti tik vartotojo rėžime. Jam įvykus VM registrų reikšmės išsaugomos ir procesorius perjungiamas į supervisoriaus rėžimą, kuriame nustatomas pertaukimo pobūdis bei kviečiama pertraukimą apdorojanti programa. Vėliau valdymas grįžta atgal į VM, vartotojo rėžimą ir atstatomi visi registrai. Pertaukimus aptinka komanda Test(). Išskirsime trijų rūšių pertraukimus: o Programiniai, kurių registras yra PI. Galimi atvejai:

 PI = 1 – neteisingas adresas.

 PI = 2 – blogas operacijos kodas.

 PI = 3 – neteisingas priskyrimas.

 PI = 4 – perpildymas (overflow)

o Supervizoriniai, kurių registras SI. Galimi atvejai:

 SI = 1 – komanda PRTS

 SI = 2 – komanda PRTN

 SI = 3 – komanda P

 SI = 4 – komanda READ

 SI = 5 – komanda STOPF

o Taimerio, kurio registras TI. Galimi atvejai:

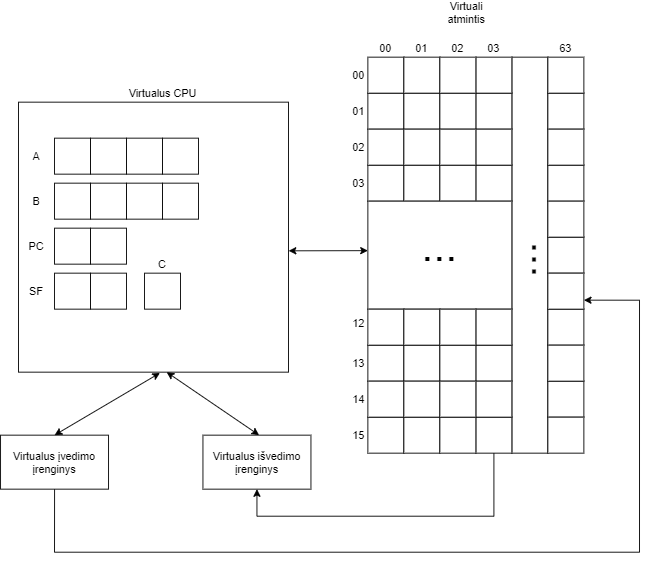
 TI = 0 – taimerio skaitliukas lygus 0

### Taimeris

Skirtas užduotims suderinti. Yra sakoma, kad užduotis negali trukti ilgiau nei tam tikrą T laiko momentų. Susitarsime, jog išvedimo / įvedimo operacijos reikalauja 3 laiko momentų, o kitos 1 ar 2.

Taigi kai VM pradeda darbą, speciali supervizorinės atminties ląstelė TI yra nustatoma tam tikrai reikšmei. Tarkime 50, tuomet kai yra įvykdoma instrukcija, TI yra mažinamas priklausomai nuo to kiek laiko momentų reikia instrukcijai. Kai TI tampa lygus nuliui, mikrokomanda Test() aptinka taimerio pertraukimą.  
  
TI reikšmę galima nustatyti ar pakeisti supervizoriaus rėžime.

# Virtualio mašinos modelis



## VM atmintis

Kiekvienai virtualiai mašinai yra skiriama 16 vartotojo atminties blokų. Tuose šešiolikoje blokų (256 žodžių po 4 baitus) turi tilpti programos kodas, duomenys ir stekas. 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. Tai bus aptarta skyrelyje „puslapiavimo mechanizmas“.

## VM CPU

Kaip galima pastebėti iš virtualios mašinos modelio schemos, centrinis virtualus procesorius yra gerokai paprastesnis realios mašinos atvejis. Virtualios mašinos procesoriaus paskirtis - vykdyti programą, kuri yra virtualioje atmintyje. Kiekvienas procesas turi savo virtualų centrinį procesorių, tačiau modelyje sisteminių procesų programas vykdys aukšto lygio kalbos procesorius. Taigi realiai mūsų projekte virtualius procesorius turės tik procesai – virtualios mašinos. Virtualus procesorius turi du bendros paskirties registrus - A ir B (po 4 baitus), komandų skaitiklio registrą - PC (2 baitų), požymių registrą - SF (2 baitų) ir loginio trigerio registrą – C (1 baito).