

Informatika, 3 kursas, 4 grupė

Virtualios ir realios mašinos projektas

Andrius Bagaliūnas

Jokūbas Kondrackas

Arnoldas Čiplys

Turinys

Turinys	2
Užduoties sąlyga	3
Realioji mašina	4
Apibrėžimas	4
Centrinis procesorius	4
Atmintis	5
Taimeris	5
Pertraukimai	5
Kanalų įrenginys	5
Puslapiavimo mechanizmas	6
Realios mašinos brėžinys	6
Virtualioji mašina	8
Apibrėžimas	8
Centrinis procesorius	8
Atmintis	8
Komandų sistema	8
Bendravimas su I/O įrenginiais	9
Kompiliuojamo vykdomojo failo išeities teksto formatas ir vykdymo eiga	9
Virtualios mašinos brėžinys	11

Užduoties sąlyga

1 variantas

Projektuojama interaktyvi OS.

Virtualios mašinos procesoriaus komandos operuoja su duomenimis, esančiais registruose ir ar atmintyje. Yra komandos duomenų persiuntimui iš atminties į registrus ir atvirkščiai, aritmetinės (sudėties, atimties, palyginimo), sąlyginio ir besąlyginio valdymo perdavimo, įvedimo, išvedimo, darbo su bendra atminties sritimi (prieinama visoms vartotojo programoms; komandos leidžia į ją rašyti ir skaityti; sritis apsaugoma semaforais) ir programos pabaigos komandos. Registrai yra tokie: komandų skaitiklis, bent du bendrosios paskirties registrai, požymių registras (požymius formuoja aritmetinės, o į juos reaguoja sąlyginio valdymo perdavimo komandos). Atminties dydis yra 16 blokų po 16 žodžių (žodžio ilgį pasirinkite patys).

Realios mašinos procesorius gali dirbti dviem režimais: vartotojo ir supervizoriaus. Virtualios mašinos atmintis atvaizduojama į vartotojo atmintį naudojant puslapių transliaciją. Yra taimeris, kas tam tikrą laiko intervalą generuojantis pertraukimus. Įvedimui naudojama klaviatūra, išvedimui - ekranas. Yra išorinės atminties įrenginys - kietasis diskas.

Vartotojas, dirbantis su sistema, programas paleidžia interaktyviai, surinkdamas atitinkamą komandą. Laikoma, kad vartotojo programos yra realios mašinos kietajame diske, į kurį jos patalpinamos „išorinėmis“, modelio, o ne projektuojamos OS, priemonėmis.

Reali mašina

Apibrėžimas

Reali mašina yra kompiuteris. Ją sudaro tokie techninės įrangos komponentai:

1. Centrinis procesorius
2. Atmintis
 - 2.1. Vartotojo
 - 2.2. Supervizorinė
 - 2.3. Išorinė (kietasis diskas)
3. Įvedimo įrenginys (klaviatūra)
4. Išvedimo įrenginys (monitorius)

Centrinis procesorius

Centrinis procesorius – skaito komandą iš atminties ir ją vykdo (interpretuoja).

Gali dirbti dviem režimais: vartotojo arba supervizoriaus.

- Vartotojo režimas. HLP vykdo tam tikrą užduoties programą. Yra imituojamas virtualios mašinos procesorius ir prieinama prie vartotojo atmintyje esančių programų per puslapiavimo mechanizmą.
- Supervizoriaus režimas. Komandos iš supervizorinės atminties yra betarpiškai apdorojamos HLP.

HLP – bet kuris aukšto lygio kalbos procesorius (programavimo kalba)

Procesorius turi šiuos registrus:

- IC – komandų skaitiklis
- SF – požymių registras (flag), kuris susidaro iš
 - OF – overflow flag,
 - CF – carry flag
 - ZF – zero flag
- AX – bendrosios paskirties registras
- BX – bendrosios paskirties registras
- MODE – registras nusakantis darbo režimą (0 – vartotojas, 1 – supervizorius)
- TI – timerio registras
- PI – programinių pertraukimų registras
- SI – supervizoriaus pertraukimų registras
- PTR – puslapių lentelės registras

Atmintis

- Vartotojo – skirta virtualių mašinų atmintims bei puslapių lentelių laikymui
- Supervizorinė – skirta pačios OS poreikiams. Tai valdys HLP
- Išorinė – kietasis diskas

Taimeris

Skirtas užduotims suderinti. Laikoma, kad ta pati užduotis negali trukti negu daugiau N laiko momentų. Mūsų OS atveju: įvedimo/išvedimo operacijos atliekamos per 3 laiko momentus, visos kitos per 1 laiko momentą. Programos pradžioje, mūsų atveju TI nustatome prieš paleisdami VM.

Pradedant VM darbą, yra nustatoma registro TI reikšmė. Po kiekvienos instrukcijos TI reikšmė mažinama pagal tai, kiek operacija užima laiko momentų. Kai $TI = 0$, iškviečiama mikrokomandą `Test()`, kurio aptinka taimerio pertraukimą.

Pertraukimai

Tai tam tikri signalai apie specialius įvykius. Pertraukimas savaime nepertraukia sistemos darbo, pertraukimą sistema turi aptikti ir atitinkamai sureaguoti. Gali būti aptikti tik vartotojo režime ir jie aptinkami procesoriaus komanda `Test()`, kuri tikrina reikšmes tam tikruose registruose ir yra kviečiama kiekvieną kartą interpretatoriui atlikus programą. Nustačius įvykusį pertraukimą, virtualios mašinos procesoriaus registrų reikšmės yra išsaugomos ir procesorius perjungiamas į supervizorinį režimą, kuriame nustatomas pertraukimo pobūdis ir kviečiama pertraukimą apdorojanti programa. Vėliau valdymas grįžta atgal į VM vartotojo režimu ir visi registrai atstatomi.

Turime trijų rūšių pertraukimus:

- Programiniai pertraukimai, kurių registras yra PI
- Supervizoriniai pertraukimai, kurių registras yra SI
- Timerio pertraukimai, kurių registras yra TI

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. Supervizorinė atmintis;
3. Kietasis diskas
4. Ekranas

DT: Objekto, į kurį kopijuosime, numeris

1. Vartotojo atmintis;
2. Supervizorinė atmintis;
3. Kietasis diskas
4. Ekranas

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**.

Puslapiavimo mechanizmas

Realios mašinos atmintį sudaro 80 blokų. Sąryšis tarp realios ir virtualios atminties bus palaikomas per puslapiavimo mechanizmą. Virtualios mašinos bloką sudaro 16 blokų po 16 žodžių. Čia mes naudojame puslapių lenteles tam, kad išlaikytume sąryšius tarp realių ir virtualių blokų adresų, jos yra saugomos vartotojo atmintyje. Už mechanizmą yra atsakingas PTR registras.

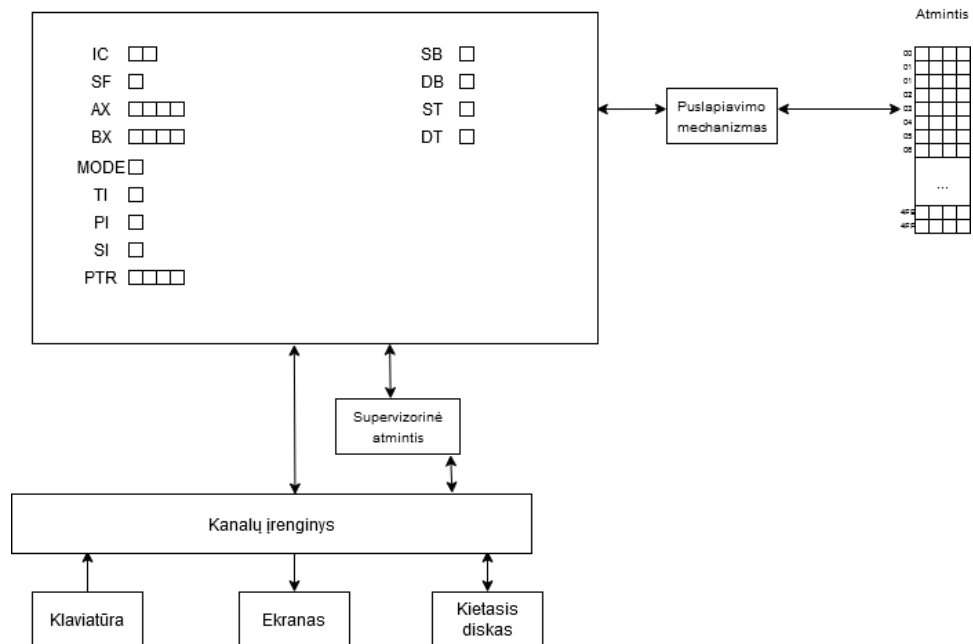
Registre PTR bus saugomas puslapių lentelės adresas: a0, a1, a2, a3, kur:

- a0 – tuščias (nenaudojamas)
- a1 – tuščias (nenaudojamas)
- a2 ir a3 yra naudojami realaus adreso formulėje

Realų adresą galima užrašyti tokia formule: $16 * [16 * (16 * a2 + a3) + x1] + x2$. Formulės narių paaiškinimai:

- x1, x2 – virtualus adresas, x1 – atminties puslapio numeris,
- x2 – žodžio, tame atminties puslapyje, numeris.
- $16 * a2 + a3$ – puslapių lentelės adresas
- $16 * (16 * a2 + a3)$ – puslapių lentelės bloko adresas
- $16 * (16 * a2 + a3) + x1$ – puslapio elemento x1 adresas puslapių lentelėje.
- $16 * [16 * (16 * a2 + a3) + x1]$ – VM bloko x1 realaus bloko adresas
- $16 * [16 * (16 * a2 + a3) + x1] + x2$ – realus adresas atitinkantis virtualų adresą x1x2

Realios mašinos brėžinys



Virtuali mašina

Apibrėžimas

Virtuali mašina (toliau VM) yra realios mašinos modelis, kuris veikia kaip tarpininkas. Ji smarkiai supaprastina tiek ir programų rašymą, tiek ir pačią realizaciją. VM pagrindinė paskirtis yra vykdyti vartotojo programą. Ją sudaro šie loginiai komponentai:

1. Centrinis procesorius
2. Atmintis
3. Komandų sistema
4. Bendravimas su I/O įrenginiais

Centrinis procesorius

Centrinio procesoriaus paskirtis – vykdyti programą, kuri yra virtualioje atmintyje.

Registrai:

- IC – komandų skaitiklis
- AX – bendro naudojimo registras
- BX – bendro naudojimo registras

Atmintis

Kiekvienai VM skiriama po 16 atminties blokų po 16 žodžių, kai žodžio ilgis – 4 baitai. Juose turi tilpti programos kodas ir duomenys. Kiekvienas virtualios atminties blokas turi virtualų ir realų adresą. Virtualiais adresais operuoja VM, o realiais – reali mašina. Ryšiai tarp virtualaus ir realaus adresos nusakomi puslapių lentelėmis.

Komandų sistema

Aritmetinės komandos:

- `_ADD` – sudeda AX ir BX, rezultatas įrašomas į AX, jei rezultatas overflow'ino, OF tampa 1
- `_SUB` – atima BX iš AX, rezultatas įrašomas į AX, jei rezultatas underflow'ino, OF tampa 1
- `_MUL` – sudaugina AX ir BX, rezultatas įrašomas į AX, jei rezultatas overflow'ino, OF tampa 1
- `_DIV` – padalina AX iš BX, rezultatas įrašomas į AX, jei BX = 0, DF tampa 1 ir dalyba neįvyksta

Palyginimo komanda:

- `_CMP` – iš AX atima BX ir pagal rezultatą (kurio niekur nesaugo) sudeda status flag'us
 - Jei $AX = BX$, $ZF = 1$, o $CF = 0$
 - Jei $AX < BX$, $ZF = 0$, o $CF = 1$
 - Jei $AX > BX$, $ZF = 0$ ir $CF = 0$

Darbo su duomenimis komandos:

- `LAX1x2` – į registrą AX užkrauna žodį iš adreso $16 * x1 + x2$
- `SAX1x2` – iš registro AX užkrauna žodį į adresą $16 * x1 + x2$
- `PRAX` – atspausdinama į ekraną AX reikšmė
- `PRx1x2` – atspausdinama į ekraną $16 * x1 + x2$ adrese esantis žodis
- `PAX1x2` – atspausdinami visi simboliai į ekraną pradedant adresu $16 * x1 + x2$ iki simbolio \$
- `RDAX` – nuskaityta iš klaviatūros reikšmė ir padedama AX
- `RDx1x2` – nuskaityta iš klaviatūros reikšmė ir padedama adrese $16 * x1 + x2$
- `RDAX1` – konstanta x1 užkraunama į registrą AX
- `SWAP` – Apkeisti registrą AX su registru BX

Pagrindiniai veiksmai tarp registrų ir atminties vyksta su registru AX, norint kažką įkelti į BX registrą, galime pirmą užkrauti į AX, tada su `SWAP` sukeisti ir įdėti į BX registrą

Valdymo komandos:

- Jump'ai:
 - `JPxy` – nesąlyginis jump į komandą adresu $16 * x + y$
 - `JMxy` – Jump More, jei $AX > BX$ (ZF ir CF yra 0), šoka į komandą adresu $16 * x + y$
 - `JLxy` – Jump Less, jei $AX < BX$ ($CF = 1$), šoka į komandą adresu $16 * x + y$
 - `JExy` – Jump Equal, jei $AX = BX$ ($ZF = 1$), šoka į komandą adresu $16 * x + y$
 - `JNxy` – Jump Not Equal, jei $AX \neq BX$ ($ZF = 0$), šoka į komandą adresu $16 * x + y$
 - `JXxy` – Jump More or Equal, jei $AX \geq BX$, šoka į komandą adresu $16 * x + y$
 - `JYxy` – Jump Less or Equal, jei $AX \leq BX$, šoka į komandą adresu $16 * x + y$
- `HALT` – programos valdymo pabaiga
- `SDx1x2 data$` - įdeda pradedant adresu $16 * x1 + x2$ visą data, kuri baigiasi simboliu \$ (store data)

Bendravimas su I/O įrenginiais

I/O operacijos atliekamos supervizoriaus režime, naudojant kanalų įrenginį.

- Įvedimo įrenginys – klaviatūra. Turi 16 žodžių buferį, jam užsipildžius, jo turinys perkeliamas į atmintį.
- Išvedimo įrenginys – spausdintuvas. Turi 16 žodžių buferį.

Kompiliuojamo vykdomojo failo išeities teksto formatas ir vykdymo eiga

Begalinis ciklas:

- SD00 JP00\$
- JP00

Suma:

- SD00 Iveskite pirma skaiciu\$
 - SD30 Iveskite antra skaiciu\$
 - SD50 Suma: \$
 - PA00
 - RDAX
 - PA30
 - RDBX
 - _ADD
 - PA50
 - PRAX
 - HALT
-

Virtualios mašinos brėžinys

