Start\_Stop – procesas atsakingas už korektišką darbo pradžią ir pabaigą.

Read\_In\_Cards – procesas atsakingas už betarpišką užduoties įvedimą.

JCL – analizuoja užduoties struktūrą: sintaksę, duomenis.

Job\_To\_Drum – atpažintas užduotis kaupia išorinėje atmintyje.

Loader – paruoštas vykdymui užduotis krauna į operatyviąją atmintį.

Lines\_from\_Drum – procesas kaupiantis įvykdytų užduočių rezultatus paruošdamas juos išvedimui.

Get\_Put\_Data – procesas atsakingas už užduoties metu reikalingus įvedimo/išvedimo veiksmus.

Chan3\_Device – procesas atsakingas už darbą su išoriniais įrenginiais.

Interrupt – procesas atsakingas už sisteminius pertraukimus, jų transformavimą į resursus.

Main\_Process – užduočių atlikimo valdymo sisteminis procesas.

Job\_Governer – procesas atsakingas už kiekvienos užduoties vykdymo prižiūrėjimą.

Virtual\_Machine – virtualios mašinos vykdymo procesas.

Start\_Stop procesas

Tai procesas inicializuojantis sisteminius statinius resursus bei sisteminius statinius procesus. Tuomet jis užsiblokuoja laukdamas resurso reiškiančio operacinės sistemos darbo pabaigą. Gavęs šį resursą procesas sunaikina sukurtus sisteminius procesus bei sisteminius resursus.

Read\_In\_Cards

Procesas laukia įvedimo srauto resurso. Gavęs šį resursą procesas laukia supervisorinės atminties resurso. Gavęs supervisorinės atminties resursą procesas nukopijuoja užduotį iš įvedimo srauto į supervisorinę atmintį ir iškviečia užduoties analizavimo procesą. Jam baigus darbą procesas grąžina supervisorinėje atmintyje esančios užduoties resursą ir grįžta į pradinę laukimo būseną.

JCL

Procesas laukia užduoties supervisorinėje atmintyje resurso. Gavęs šį resursą inicializuoja programos duomenų ir rezultatų sąrašus. Procesas analizuoja programos struktūrą ir grąžina arba užduoties resursą arba pranešimo resursą, reiškiantį, kad programoje yra klaidų.

Job\_to\_Drum

Procesas laukia užduoties programos, užduoties duomenų, užduoties parametrų ir užduoties rezultatų nesupervizorinėje atmintyje resursų. Taip pat išorinės atminties, trečiojo kanalo resursų. Gavus šiuos resursus yra sukuriamas Chan3\_Device pranešimas, kuriame saugoma einamoji užduotis. Toliau laukiama, kol Chan3\_Device pateiks pranešimą, jog baigė darbą. Sukuriamas užduotis išorinėje atmintyje resursas. Atlaisvinama supervizorinė atmintis ir grįžtama į pradinę laukimo būseną.

Main\_Process

Procesas laukia užduoties išorinėje atmintyje resurso. Gavęs šį resursą patikrina ar jis tikras ar fiktyvus. Jei resursas fiktyvus – sunaikinamas atitinkamas Job\_Governer. Jei resursas tikras – sukuriamas naujas Job\_Governer. Atlikus darbą grįžtama į laukimo būseną.

Job\_Governer

Procesas laukia vartotojo atminties resurso. Gavęs šį resursą sukuria pakrovimo paketo resursą, kurį nusiunčia Loader procesui ir laukia jo darbo pabaigos pranešimo. Gavęs šį pranešimą sukuria procesą Virtual\_Machine. Sukūręs procesą blokuojasi ir laukia pertraukimo resurso. Gavęs šį resursą procesas tikrina ar tai yra įvedimo/išvedimo pertraukimas. Jei taip, sukuriamas pranešimas procesui Get\_Put\_Data ir laukia jo darbo pabaigos pranešimo. Gavęs šį pranešimą aktyvuoja procesą Virtual\_Machine ir grįžta prie pertraukimo pranešimo resurso laukimo. Jei pertraukimas buvo ne įvedimo/išvedimo tuomet naikinamas Virtual\_Machine procesas ir atlaisvianama vartotojo atmintis. Tuomet laukiama trečiojo kanalo resurso, kurį gavus suformuojamas pranešimas procesui Chan3\_Device ir laukiama jo darbo pabaigos. Tuomet suformuojamas užduoties darbo pabaigos resursas bei sukuriamas fiktyvus užduoties išorinėje atmintyje resursas.

Loader

Procesas laukia pakrovimo paketo resurso. Gavus laukiama trečiojo kanalo resurso. Toliau pasiunčiamas pranešimas Chan3\_device ir laukiama jo darbo pabaigos. Jo sulaukus, pasiunčiamas Job\_Governer pranešimas ir grįžtama į pradinę laukimo būseną.

Virtual\_Machine

Procesas perjungia procesoriaus režimą į vartotojo. Tai padarius, vykdoma Virtual\_Machine progama. Tikrinama ar neįvyko pertraukimas, jei taip, sukuriamas pertraukimo įvykio resursas.

Interrupt

Procesas laukia pertraukimo įvykio resurso. Sulaukus, identifikuojamas pertraukimo tipas. Tada identifikuojamas Job\_Governer ir jam nusiunčiamas pranešimas.

Get\_Put\_Data

Procesas laukia pranešimo iš Job\_Governer. Tada laukiama išorinės atminties ir priklausomai nuo to ar pranešimas yra įvedimui ar išvedimui, sukuriamas atitinkamas resursas. Laukiama trečio kanalo resurso. Sukuriamas Chan3\_Device procesas ir jam paduodamas pranešimas. Laukiama Chan3\_Device darbo pabaigos. Sukuriamas pranešimas procesui Job\_Governer ir grįžtama į pradinę laukimo būseną.

Chan3\_Device

Procesas laukia pranešimo. Tada nukopijuoja nurodytą informacijos kiekį. Pasiunčia darbo pabaigos pranešimą ir grįžta į pradinę laukimo būseną.

Lines\_From\_Drum

Procesas laukia užduoties darbo pabaigos resurso. Paimamas nurodytas informacijos kiekis išorinėje atmintyje. Laukiama supervizorinės atminties resurso. Sulaukus, laukiama trečiojo kanalo resurso. Tuomet pasiunčiamas pranešimas Chan3\_Device procesui ir laukia jo darbo pabaigos. Tada sukuriamas eilutė supervizorinėje atmintyje resursas ir grįžtama į pradinę laukimo būseną.

Proceso deskriptorius

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Id** | **CPU** | **OA** | **R** | **SR** | **ST** | **SD** | **T** | **S** | **PR** |

**Id** – Proceso išorinis vardas.  
**CPU** – Centrinio procesoriaus būsena vykdant procesą. Nutraukus proceso vykdymą einamoji būsena išsaugoma.  
**OA** – Procesui priklausanti operatyvioji atmintis.  
**R** – Proceso turimų resursų sąrašas.  
**SR** – Proceso sukurtų resursų sąrašas.  
**ST** – Proceso būsena.  
**SD** – Nuoroda į sąrašą, kuriame yra procesas (LPS arba PPS)  
**T** – Proceso kūrėjo (tėvinio proceso) vidinis vardas.  
**S** – Proceso sukurtų procesų (sūnų) sąrašas.  
**PR** – Proceso prioritetas.

Resurso deskriptorius

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Rid** | **K** | **PNR** | **PA** | **LPS** | **PASK** |

**Rid –** Resurso išorinis vardas. **K –**Kūrėjo (proceso) vidinis vardas. **PNR -** Nurodo ar tai pakartotinio naudojimo, ar vienkartinio naudojimo resursas. **PA –** Resurso prieinamumo aprašymas. **LPS –** Nuoroda į laukiančių procesų sąrašą. **PASK –** Nuroda į resurso paskirstytoją.

Planuotojas

Planuotojo pagrindinė užduotis – užtikrinti, kad visi procesai būtų vykdomi maksimaliu prioritetu (jei prioritetai vienodi, tai vykdomas tas, kuris sąraše pirmesnis). Planuotojas iškviečiamas operacinės sistemos branduolio primityvais. Jo darbas vyksta trim etapais:

* Rasti procesą su aukščiausiu prioritetu
* Rasti laisvą procesorių ir jį atiduoti rastam procesui.
* Jei tokio laisvo procesoriaus nėra – peržiūrėti vykdomus procesus, ir jei tarp jų yra procesų su mažesniu prioritetu nei norimo vykdyti proceso – atiduoti mažiausio prioriteto proceso procesorių norimam procesui.

Darbo su procesais primityvai:

KURTIP – informaciją apie kuriamą procesą suregistruota į deskriptorių.

STABDYTIP – stabdomas vienas procesas, o kiti (vaikiniai) tęsia darbą.  
AKTYVUOTIP - nuima proceso pristabdymą ir iškviečia planuotoją, jei būsena – READY.  
NAIKINTIP – gali sunaikinti bet kurį savo palikuonį, bet negali sunaikinti savęs.  
KEISTIPP – keičia proceso prioritetą.

Darbo su resursais primityvai:

PRAŠYTIR - procesas, kuriam reikia resurso, iškviečia šį primityvą. Toks procesas pereina į blokavimosi būseną.  
ATLAISVINTIR – kai procesui resurso nebereikia, resursas įjungiamas į laisvų resursų sąrašą.  
KURTIR - suregistruoja informaciją apie kuriamą resursą į jo deskriptorių.  
NAIKINTIR – resurso sunaikinimas (tai gali padaryti tik tėvai arba pirmtakai). Panaikinamas resurso deskriptorius, prieš tai suaktyvinami jo laukiantys procesai.