
Kuro degalinė

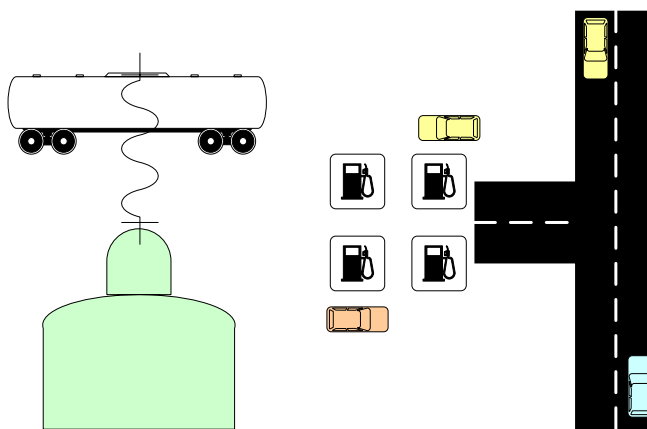
1 KONCEPTUALUSIS MODELIS

Užduotis. Sumodeliuoti kuro degalinę, kurioje yra keturios degalų užpildymo kolonėlės. Įtraukti degalų pylimo į atvykusius automobilius ir degalų rezervuaro užpildymo procesus.

Degalinėje yra 4 degalų užpildymo kolonėlės, kuriose pildomi tos pačios rūšies degalai (1 pav.). Automobiliai atvažiuoja pagal eksponentinį dėsnį vidutiniškai kas 0,8 minutės. Pilamas degalų kiekis pasiskirstęs pagal trikampio dėsnį su parametrais 16, 30 ir 60 litrų. Degalų pylimo į automobilio baką greitis yra 40 litrų per minutę. Laiko trukmė, kol vyksta mokėjimas už degalus, pasiskirsčiusi pagal trikampio dėsnį su parametrais 0,5, 1 ir 1,5 minutės.

Laiko intervalas tarp degalinės rezervuaro užpildymo momentų pasiskirstęs pagal tolydųjį dėsnį su $\min=2,75$ ir $\max=3,25$ valandos. Atvežamas toks degalų kiekis, kad jo užtektų visiškai užpildyti rezervuarą. Degalų pylimo į rezervuarą greitis yra 1200 litrų per minutę. Rezervuaro maksimali talpa 8000 litrų, o pradinis lygis 800 litrų.

Jei rezervuaras ištuštinamas dar prieš tai, kol atvyksta degalų sunkvežimis, kolonėlės yra uždaromos iki to laiko, kol pildant rezervuarą pasiekiamas minimalus 400 litrų degalų kiekis. Jei degalų pritrūko automobiliams dar bepilant norimą kiekį, automobilis nelaukia ir palieka kolonėlę.



1 pav. Kuro degalinės modelis

Tarkime, kad:

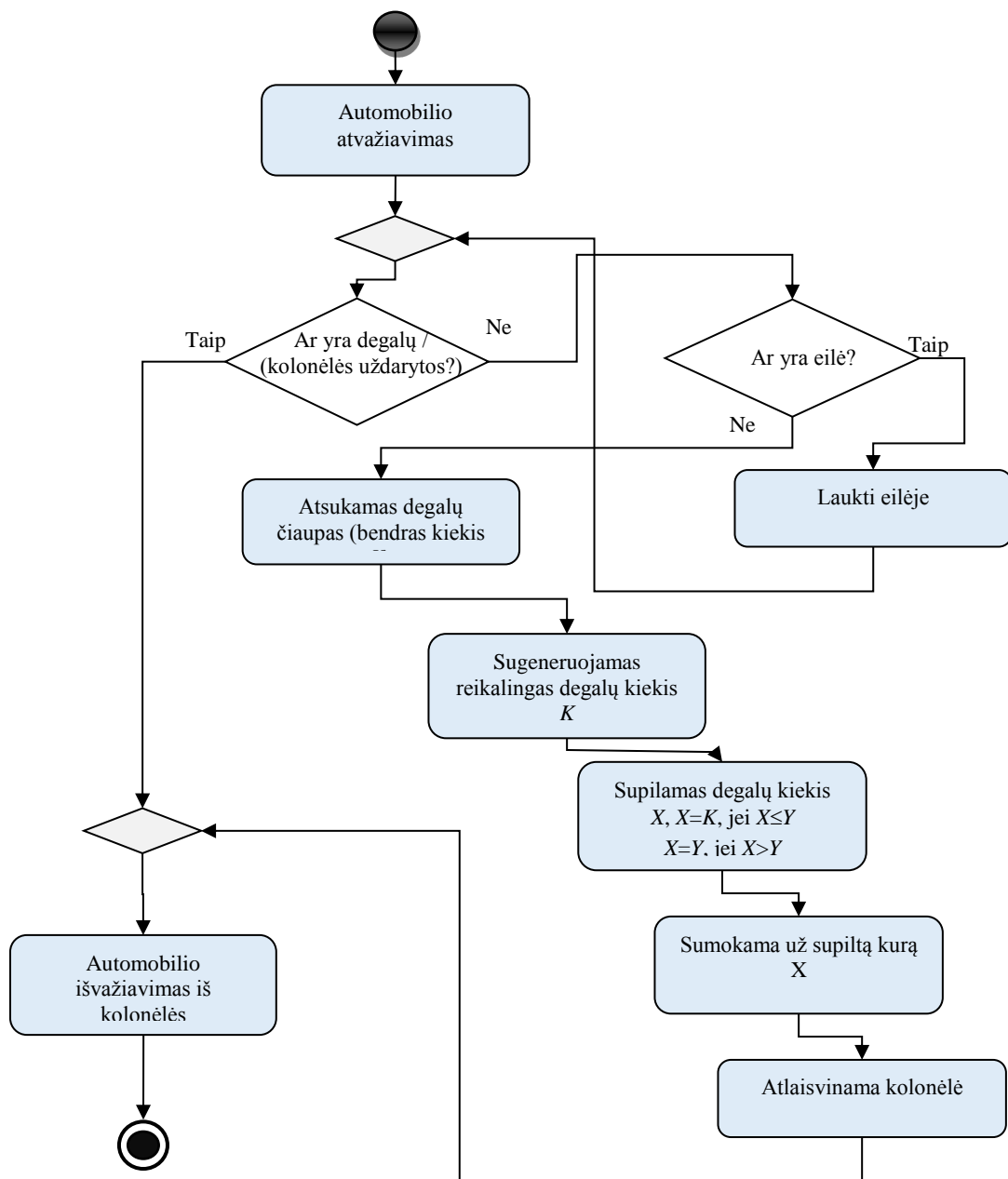
- automobiliai į degalinę nevažiuoja, jeigu rezervuaras tuščias, ir tuomet, kol rezervuarą pildant nepasiekiamas 400 litrų degalų lygis;
- jei pilant degalus į automobilį norimo kiekio neužtenka, tai šis nebelaukia, kol rezervuaras bus užpildytas, ir susimokėję už degalus išvyksta iš degalinės;
- jei automobilis stovėjo eilėje prie kolonėlių, o rezervuaras ištuštėjo, tai automobilis eilėje nebelaukia ir išvyksta iš degalinės;
- degalų cisterna į degalinę atvyksta tam tikru periodiškumu ir pripildo visą degalinės rezervuarą.

Imitacinio modelio parametrai: pradinė būsena – visi įrenginiai laisvi. Imitavimo trukmė yra 24 valandos.

Statistika renkama apie automobilius, išvykusius be degalų ir prisipildžiusius degalų.

2 DEGALŲ PYLIMO Į AUTOMOBILĮ VEIKLOS DIAGRAMA

2 pav. pateikta degalų pylimo į automobilius proceso veiklos diagrama.



2 pav. Degalų pylimo proceso veiklos diagrama

3 MODELIO REALIZAVIMAS MODELIAVIMO SISTEMOJE ARENA

Sistemai realizuoti ARENA aplinkoje į pagrindinę schema įtraukiami trys submodeliai, aprašantys:

- degalų į automobilį pylimo procesą;
- rezervuaro papildymo degalais procesą;
- automobilių pašalinimą iš eilės prie kolonėlių, kai pasibaigia degalai.

Sistemos modelyje taip pat įtraukiami trys jutikliai, nurodantys konkrečias degalų rezervuaro būsenas: kai degalų rezervuaras yra pilnas, kai rezervuaras yra tuščias, kai tuščiame rezervuare įpilama 400 litrų degalų. Iš viso sistemos modelyje yra panaudojami 32 struktūriniai moduliai viso sistemos modeliui aprašyti.

Toliau pateikta modeliui realizuoti reikalinga veiksmų seka:

1. Atidaromas naujas projektavimo langas.
2. Degalų rezervuarami aprašyti panaudojamas *Flow Process* skydelyje esantis *Tank* modulis. Nurodoma rezervuaro talpa 8000 litrų, pradinis lygis – 800 litrų. Rezervuaras turi 5 čiaupus: 4 čiaupai (*kolonele 1*, *kolonele 2*, *kolonele 3*, *kolonele 4*) yra skirti degalams į 4 kolonėles tiekti ir 1 čiaupas (*uzpildymo*

sklende) yra skirtas degalams į rezervuarą pilti. Visi penki čiaupai kuriami panaudojant *Add* mygtuką (3 pav.).


3 pav. Rezervuaro modulis

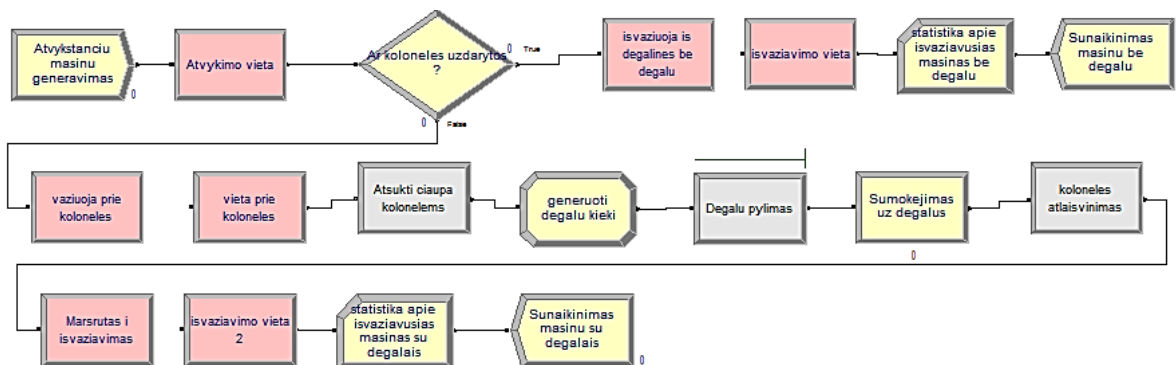
- Visos keturios kolonėlės surašomos į aibę pavadinimu **Koloneles**. Šiam tikslui *Flow Process* skydelyje panaudojamas *Regulator Set* duomenų modulis (4 pav.).

Regulator Set - Flow Process		
	Name	Members
1	Koloneles	4 rows
Double-click here to add a new row.		

Members	
	Regulator Name
1	kolonele 1
2	kolonele 2
3	kolonele 3
4	kolonele 4

4 pav. Kolonėlių aibė

- Kadangi modelis yra hierarchinės struktūros, yra įtraukiami du submodeliai: „Masinu pildymas degalais“ ir „Rezervuaro užpildymas“. Submodeliams sukurti yra panaudojamas  iš viršutinės meniu juostos. Įėjimų ir išėjimų skaičius yra lygus nuliui, nes šie submodeliai yra modelio hierarchijos viršuje. Įėjimų ir išėjimų skaičius nurodomas per *submodelio* kontekstinį meniu *Properties*.
- „Masinu pildymas degalais“ submodelis skirtas automobilių atvykimui į degalinę aprašyti. Jei rezervuaras tuščias, automobiliai pravažiuos degalinę ir nelauks, kol jame vėl bus degalų. Toliau pateikta submodelio loginė schema (5 pav.).



5 pav. Submodelio „Masinu pildymas degalais“ schema

6. Automobilių, atvykstančių į degalinę, generavimas aprašomas naudojant *Create* modulį (6 pav.).

6 pav. Automobilių generavimas

7. Panaudojant *Entity* modulį, atvykstantiems automobiliams yra priskiriamas paveikslukas.
 8. Atvykstančių automobilių padėtis nustatoma naudojant *Station* modulį iš *Advanced Transfer* skydelio (7 pav.).

7 pav. Automobilių atvykimo vietos stotelė

9. Panaudojant *Decide* modulį, tikrinama, ar kolonėlės nėra uždarytos (8 pav.).

8 pav. Kolonėlių būsenos tikrinimo modulis

Tikrinti, ar kolonėlės nėra uždarytos, yra panaudojamas loginis kintamasis „kolonelems loginis“, kurio reikšmė 1, jei rezervuare yra degalų, ir 0, jei rezervuaras tuščias. Pradinė šio kintamojo reikšmė (vienetas) nustatoma panaudojant *Variable* duomenų modulį iš *Basic Process* skydelio (9 pav.).

	Name	Rows	Columns	Data Type	Clear Option	Initial Values
1	kolonelems loginis			Real	System	1 rows

Double-click here to add a new row.

9 pav. Kolonėlių loginio kintamojo aprašymas

10. *Decide* modulio tiesos atveju automobiliai palieka degalinę, neužsipylę degalų. Automobilių išvažiavimui aprašyti panaudojamas *Route* modulis iš *Advanced Transfer* skydelio (10 pav.).

10 pav. Automobilių išvažiavimo iš degalinės modulis

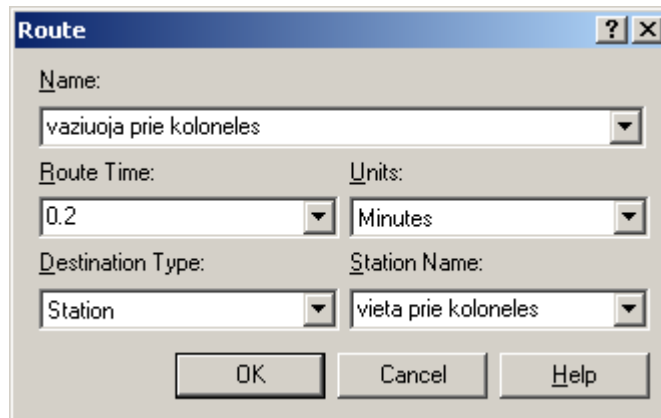
11. Automobilių, išvykstančių be degalų, padėčiai nustatyti panaudojamas *Station* modulis pavadinimu „išvažiavimo vieta“.
12. Naudojant *Record* modulį, nustatoma statistika apie išvažiavusius automobilius be degalų (11 pav.).

11 pav. Statistikos apie išvažiavusius automobilius be degalų kaupimo modulis

13. Objektai naikinami panaudojant *Dispose* modulį iš *Basic Process* skydelio (12 pav.).

12 pav. Automobilių be degalų naikinimo modulis

14. *Decide* modulio „Ar koloneles uždarytos?“ netiesos atveju automobiliai važiuoja į degalinę piltis kuro. Važiavimo procesui aprašyti panaudojamas *Route* modulis iš *Advanced Transfer* skydelio (13 pav.), nurodantis 0,2 minutės važiavimo trukmę.



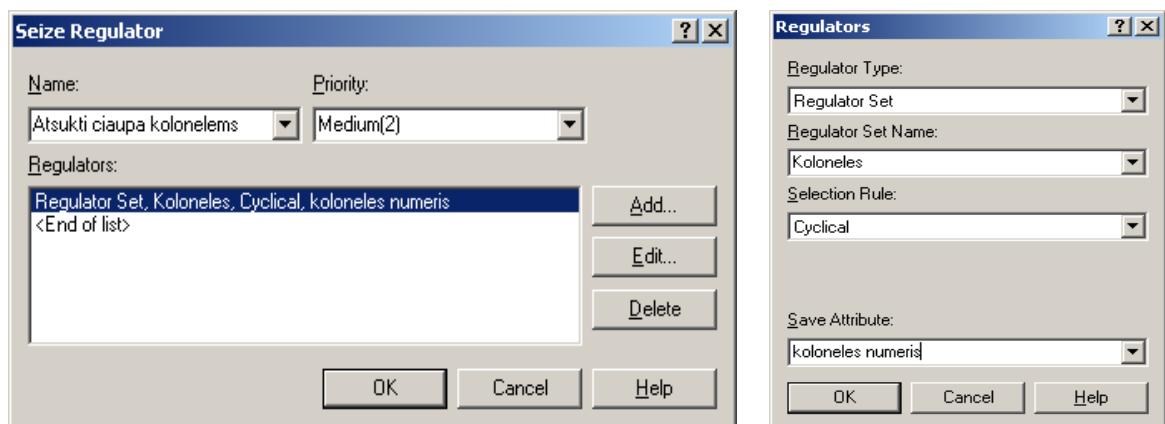
The **Route** dialog box contains the following fields:

- Name:** A dropdown menu with the value "važiuoja prie koloneles".
- Route Time:** A dropdown menu with the value "0.2".
- Units:** A dropdown menu with the value "Minutes".
- Destination Type:** A dropdown menu with the value "Station".
- Station Name:** A dropdown menu with the value "vieta prie koloneles".

Buttons at the bottom: OK, Cancel, Help.

13 pav. Važiavimo prie kolonėlės modulis

15. Automobilių padėtis prie kolonėlių yra nustatoma panaudojant *Station* modulį pavadinimu „vieta prie koloneles“.
16. Atvykę automobiliai renka kolonėlės numerį cikliškai. Pagal tai, kur atsistoja automobilis, yra atsukamas čiaupas. Tam skirtas *Seize Regulator* modulis iš *Flow Process* skydelio (14 pav.).



The **Seize Regulator** dialog box contains:

- Name:** A dropdown menu with the value "Atsukti čiaupa kolonelems".
- Priority:** A dropdown menu with the value "Medium(2)".
- Regulators:** A list box containing "Regulator Set, Koloneles, Cyclical, koloneles numeris" and "<End of list>".
- Buttons: Add..., Edit..., Delete.

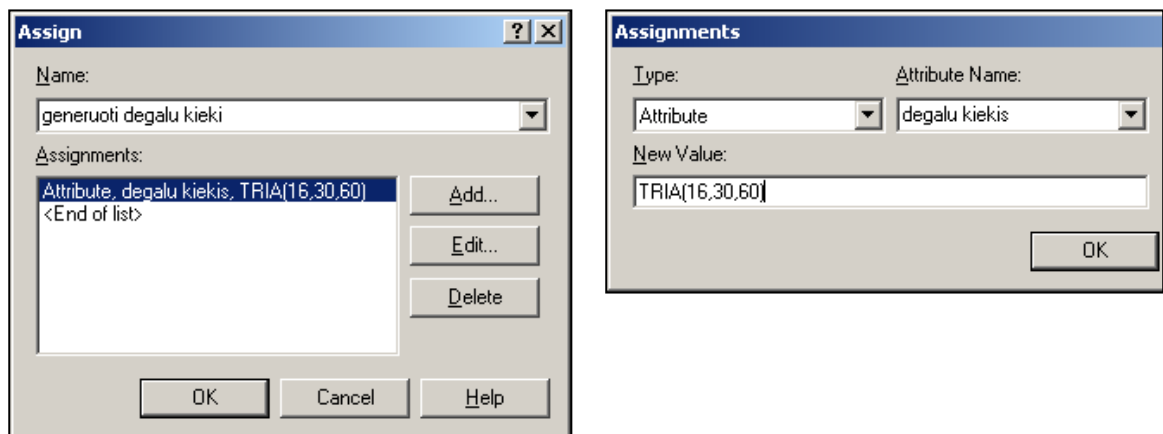
The **Regulators** dialog box contains:

- Regulator Type:** A dropdown menu with the value "Regulator Set".
- Regulator Set Name:** A dropdown menu with the value "Koloneles".
- Selection Rule:** A dropdown menu with the value "Cyclical".
- Save Attribute:** A dropdown menu with the value "koloneles numeris".

Buttons at the bottom of both: OK, Cancel, Help.

14 pav. Kolonėlės pasirinkimo modulis

17. Pilamam degalų kiekiui generuoti sukuriamas atributas **degalų kiekis**, kuris aprašomas naudojant *Assign* modulį iš *Basic Process* skydelio (15 pav.).



The **Assign** dialog box contains:

- Name:** A dropdown menu with the value "generuoti degalu kiekis".
- Assignments:** A list box containing "Attribute, degalu kiekis, TRIA(16,30,60)" and "<End of list>".
- Buttons: Add..., Edit..., Delete.

The **Assignments** dialog box contains:

- Type:** A dropdown menu with the value "Attribute".
- Attribute Name:** A dropdown menu with the value "degalu kiekis".
- New Value:** A text field containing the value "TRIA(16,30,60)".

Buttons at the bottom: OK, Cancel, Help.

15 pav. Pilamo degalų kiekio atributo modulis

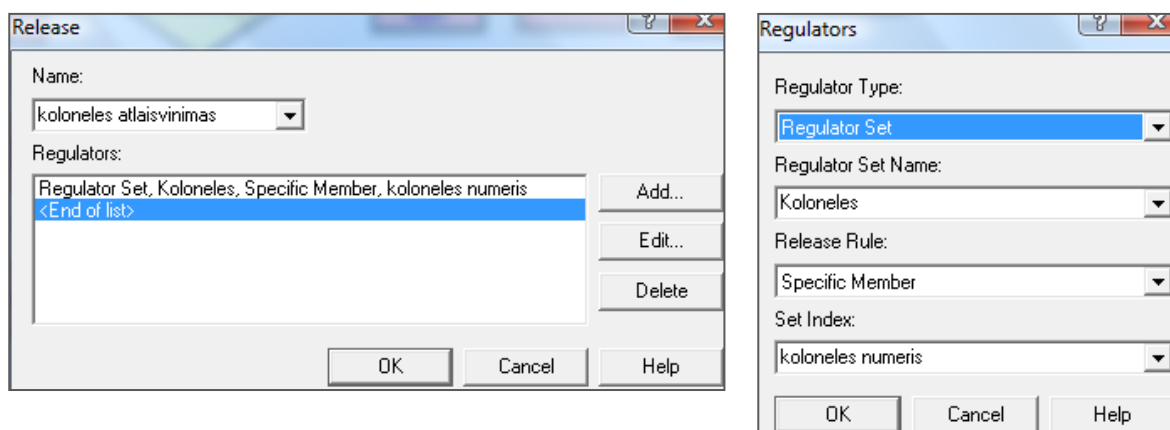
18. Kitas yra modeliuojamas pats degalų pylimo procesas panaudojant *Flow* modulį (16 pav.). Degalų pylimo greitis yra 40 l/min. Procesas yra sustabdomas, jei automobiliui supilamas visas reikalingas degalų kiekis arba gaunamas signalas **99** (99 – jutiklio signalas, parodantis, kad degalų rezervuaras yra tuščias).

16 pav. Degalų pylimo proceso modulis

19. Laiko trukmė, nurodanti, kiek laiko užtrunka klientas, mokantis už degalus, yra modeliuojama panaudojant *Process* modulį iš *Basic Process* skydelio (17 pav.).

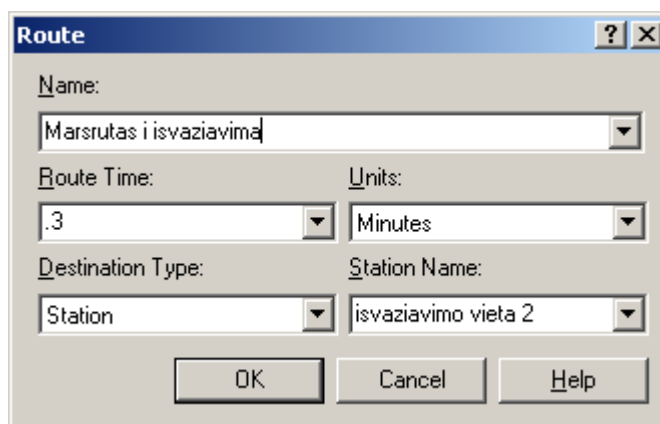
17 pav. Mokėjimo už degalus proceso modulis

20. Sumokėjus už degalus atitinkama kolonėlė yra atlaisvinama. Tam tikslui yra panaudojamas *Release Regulator* modulis iš *Flow Process* skydelio (18 pav.).



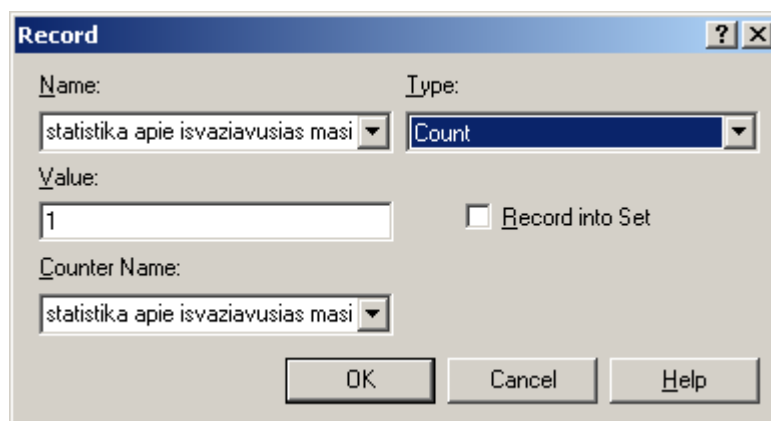
18 pav. Kolonėlės atlaisvinimo modulis

21. Automobilių išvažiavimui iš degalinės aprašyti panaudojamas *Route* modulis iš *Advanced Transfer* skydelio (19 pav.).



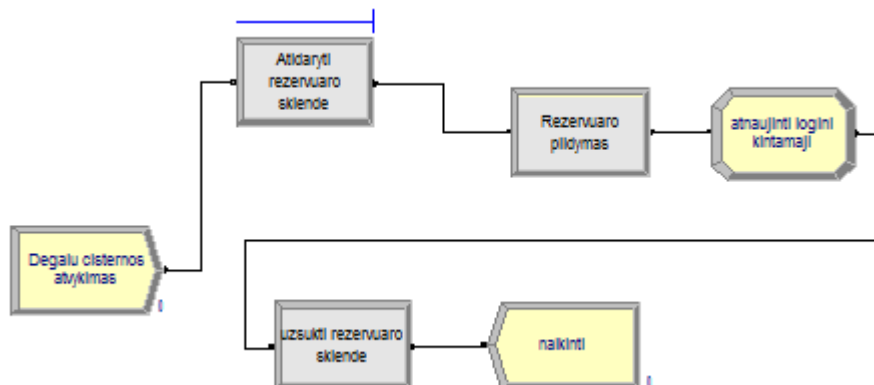
19 pav. Išvažiavimo iš kolonėlės modulis

22. Automobilių išvažiavimo padėtis nustatoma naudojant *Station* modulį „isvaziavimo vieta2“.
23. Statistika apie išvažiavusius automobilius su degalais nustatoma panaudojant *Record* modulį iš *Basic Process* skydelio (20 pav.).



20 pav. Statistikos apie išvažiavusius automobilius su degalais kaupimo modulis

24. Išeinantys objektai nustatomi panaudojant *Dispose* modulį „Sunaikinimas masinu su degalais“. Šiuo moduliu užbaigiama submodelio „Masinu pildymas degalais“ schema.
25. Kitas sudaromas submodelis „Rezervuaro užpildymas“. Toliau pateikta sudaromo modelio loginė schema (21 pav.).



21 pav. Submodelio „Rezervuaro užpildymas“ schema

26. Pirmiausia aprašomas degalų cisternos atvykimas. Tam panaudojamas *Create* modulis iš *Basic Process* skydelio (22 pav.). Cisterna sistemoje atvyksta periodiškai pagal tolygųjį pasiskirstymą kas 2,75 ir 3,25 valandos.

22 pav. Degalų cisternos atvykimo modulis

27. Rezervuaro užpildymo sklendės atsukimas aprašomas *Seize Regulator* moduliui (23 pav.).

23 pav. Rezervuaro užpildymo sklendės atsukimo modulis

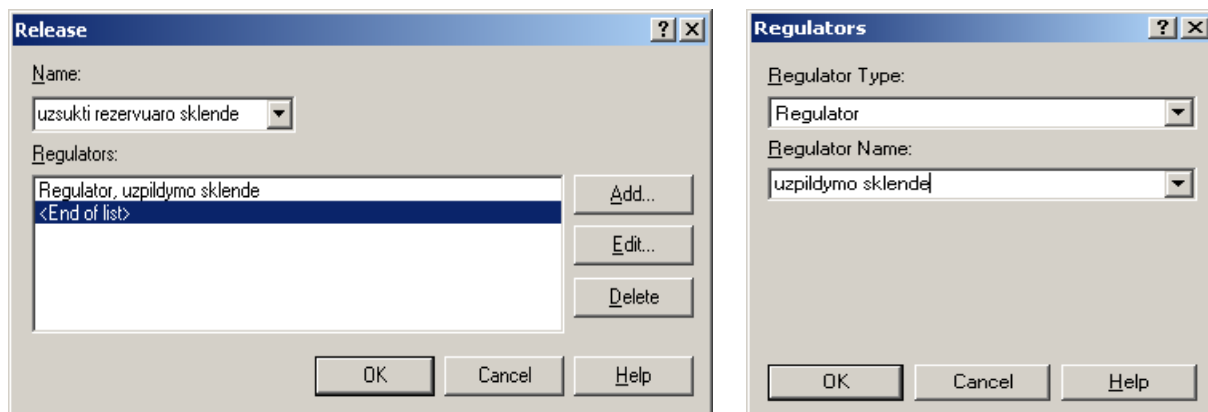
28. Panaudojant *Flow* modulį aprašomas degalų į rezervuarą pylimo procesas greičiu 1200 litrų per minutę (24 pav.). Pylimo procesas sustabdomas, jei rezervuaras pripildomas, t. y. gaunamas signalas **1** (**1** – jutiklio signalas, parodantis, kad rezervuaras pilnas).

24 pav. Degalų pylimo į rezervuarą proceso modulis

29. Be to, kintamajam *kolonelems loginis* yra priskiriama reikšmė 1, parodanti, kad rezervuaras yra netuščias ir atvykę automobiliai gali piltis kurą (25 pav.).

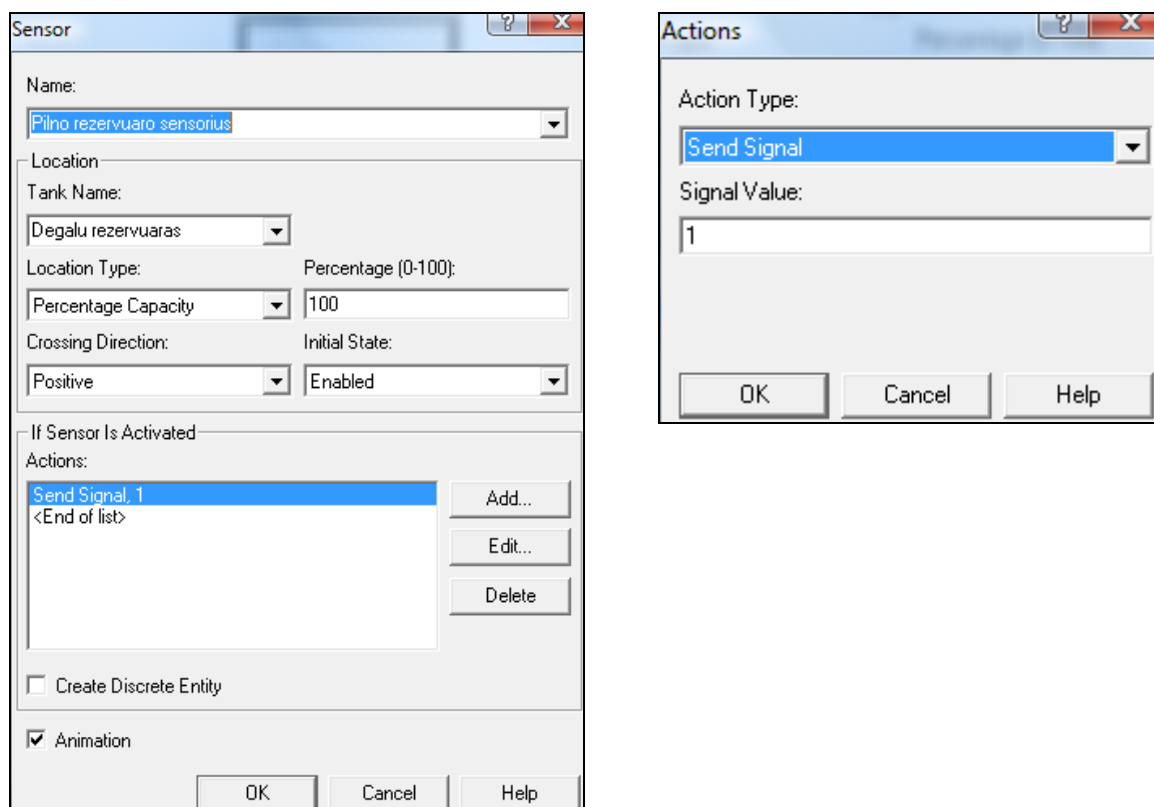
25 pav. Kolonėlių loginio kintamojo reikšmės priskyrimas

30. Degalų užpildymo sklendės užsukimas aprašomas naudojant *Release Regulator* modulį iš *Flow Process* skydelio, kuris atlaisvina prieš tai užimtą rezervuaro sklendę (26 pav.).



26 pav. Rezervuaro užpildymo sklendės užsukimo modulis

31. Submodelio objektai ištrinami naudojant *Dispose* modulį „naikinti“.
32. Aukščiausiu lygiu įtraukiami jutikliai, informuojantys apie reikšmingas rezervuaro būsenas:
 - kai rezervuaras tuščias;
 - kai rezervuaras pilnas;
 - kai pildomas rezervuaras pasiekia 400 litrų lygį.
33. Pilno rezervuaro jutiklis aprašomas panaudojant *Sensor* modulį iš *Flow Process* skydelio (27 pav.). Nurodomi tokie parametrai: 100 procentų pilnumas, pildymo kryptis *Positive* (kai degalų rezervuaro užpildymo 100 % yra pasiekiamas jį pildant, t. y. kryptis iš apačios į viršų), siunčiamas signalas 1, nurodantis, kad rezervuaras pilnas. Šis signalas nusiunčiamas *Flow* procesui „Rezervuaro pildymas“, kad būtų sustabdytas pylimo į rezervuarą procesas.



27 pav. Pilno rezervuaro sensoriaus modulis

34. 28 pav. pateiktas jutiklio aprašymas, kad rezervuare pasiektas 400 litrų lygis. Nurodomi parametrai: 400 litrų lygis, pildymo kryptis *Positive* (iš apačios į viršų), kintamajam „kolonelems loginis“ priskiriamas reikšmė 1, rodanti, kad automobiliai jau gali piltis degalus.

28 pav. 400 litrų degalų lygio jutiklio modulis

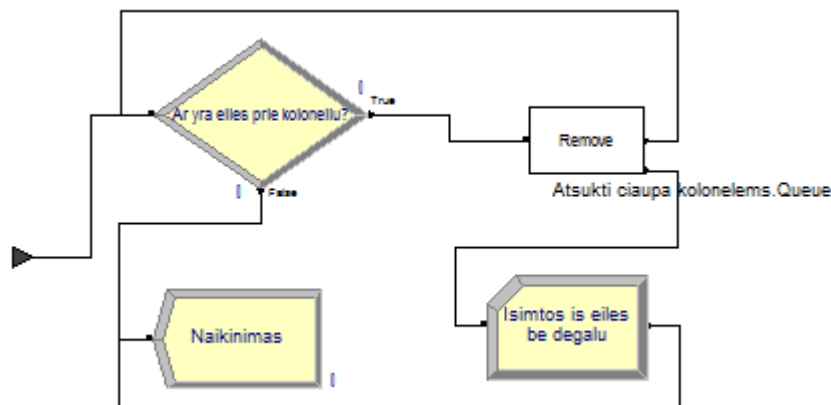
35. Tuščio rezervuaro jutiklis (29 pav.) aprašomas taip pat naudojant *Sensor* modulį, nurodomi tokie parametrai: nulinis degalų kiekis, rezervuaro tuštėjimo kryptis *negative* (degalų rezervuaras pasiekia nulį degalų lygį degalams nuolat mažėjant, t. y. kryptis iš viršaus į apačią), kintamajam *kolonelems loginis* priskiriama reikšmė **0**, nurodanti, kad automobiliai jau nebevažiuotų į degalinę. Taip pat yra siunčiamas signalas **99**, nurodantis, kad automobiliai paliktų degalinę dėl tuščio rezervuaro, nepaisant to, kad neįpildas jam reikiamas degalų kiekis. Be to, pažymima *Create Discrete Entity* modulio parinktis tam, kad būtų galima prijungti dar vieną submodelį „išimti iš eilės“ eilėje laukiantiems automobiliams (jei tokių yra) išimti iš eilės, nes rezervuaras ištuštėjo.

29 pav. Tuščio rezervuaro jutiklio modulis

36. Įtraukiant naują submodelį „išimti is eiles“ per kontekstinį meniu, jo savybėse nurodoma, kad egzistuoja vienas įvesties taškas (30 pav.).

30 pav. Submodelio „išimti is eiles“ savybės

37. Submodelis „išimti is eiles“ skirtas belaukiančių automobilių (jei tokių yra) iš eilės išėmimo procesui aprašyti, kai rezervuaras ištuštėjo. Sudaromo modelio loginė schema pateikta 31 pav.



31 pav. Submodelio „išimti iš eilės“ schema

38. Panaudojant *Decide* modulį (32 pav.), tikrinama, ar yra susidariusi automobilių eilė prie kolonėlių.

32 pav. Tikrinimo, ar yra eilė prie kolonėlių, modulis

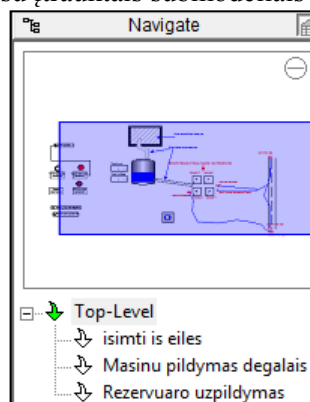
39. *Decide* modulio „Ar yra eiles prie koloneliu“ tiesos atveju (t. y. eilė egzistuoja) eilė yra mažinama dydžiu *NQ* (automobilių skaičius eilėje prie kolonėlių) naudojant *Remove* bloką iš *Blocks* skydelio (33 pav.).

33 pav. Automobilių, laukiančių eilėje prie kolonėlių, šalinimo modulis

40. Panaudojant *Record* modulį iš *Basic Process* skydelio nustatoma statistika apie išimtus automobilius iš eilės (34 pav.).

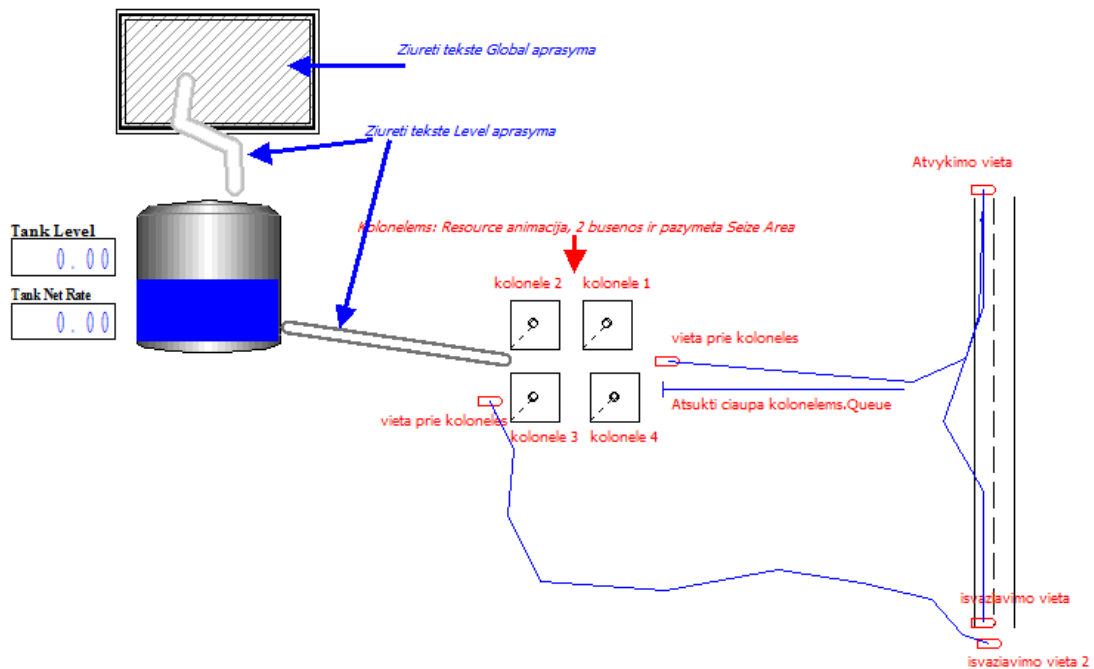
34 pav. Statistikos apie automobilius, pašalintus iš eilės, kaupimo modulis

41. Submodelio objektai naikinami naudojant *Dispose* modulį „Naikinimas“.
42. Nustatomi imitavimo parametrai: trukmė – 24 valandos, pagrindiniai vienetai – minutės (nustatomos 24 minutės). Modelio hierarchija su įtrauktais submodeliais matoma *Navigate* skydelyje (35 pav.).




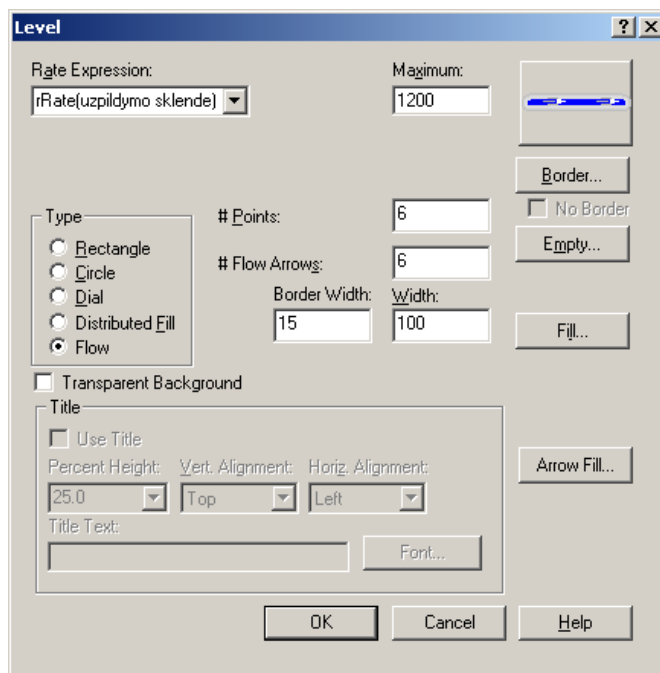
35 pav. Modelio struktūros langas

43. Modelio animacija braižoma hierarchijos viršuje (36 pav.).




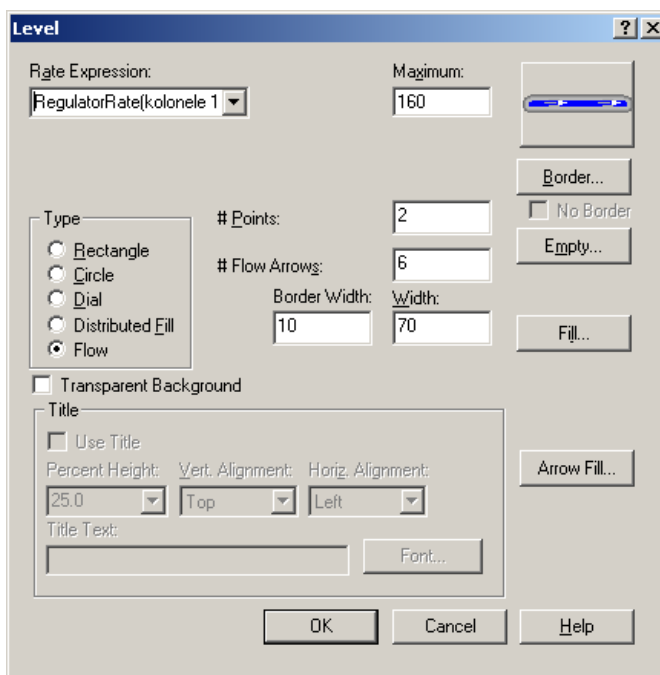
36 pav. Modelio animacija

44. Animacijoje degalų tekėjimui animuoti pildant rezervuarą iš cisternos yra panaudojamas *Level* elementas  iš viršutinės meniu juostos (37 pav.). Proceso išraiška *Rate Expression* nurodo, kokį parametą reikia animuoti. Šioje vietoje nurodoma išraiška **RegulatorRate(užpildymo sklende)**. Taip pat yra nurodomas animuoto objekto tipas, dydis ir kiti parametrai.




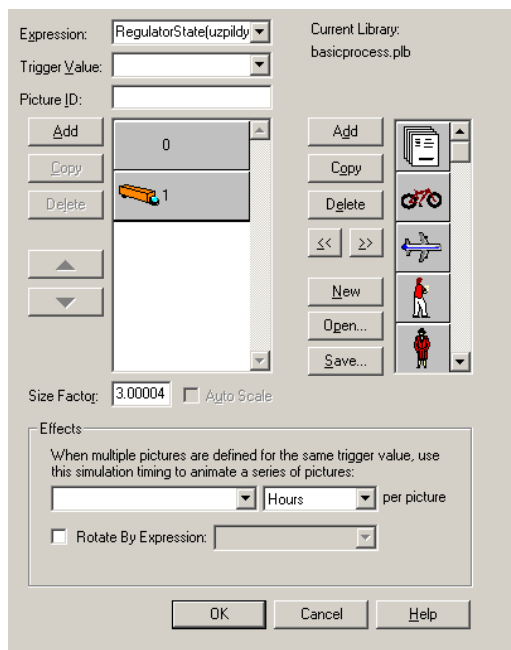
37 pav. Užpildymo sklendės animacija

45. Animacijoje degalų tekėjimui į kolonėlės animuoti yra panaudotas *Level* elementas  (38 pav.). Nurodoma proceso išraiška: **RegulatorRate(kolonele 1) + RegulatorRate(kolonele 2) + RegulatorRate(kolonele 3) + RegulatorRate(kolonele 4)**. Išraišką galima sudaryti ir panaudojant *Build Expression...*



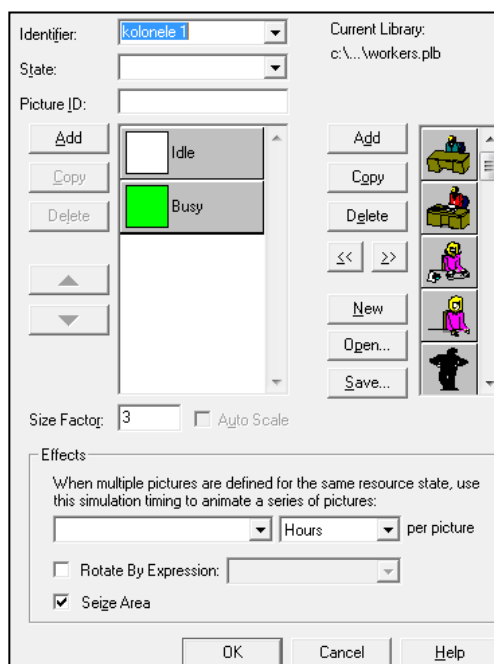
38 pav. Kuro tekėjimo į kolonėlės animacija

46. Degalų cisternos atvykimui animuoti yra panaudojamas *Global* elementas  iš meniu juostos, kuris leidžia, atsižvelgiant į tam tikro kintamojo reikšmę, rodyti arba slėpti paveiksluką (39 pav.). Išraiška, kurią reikia animuoti, yra **RegulatorState(uzpildymo sklende)**. *RegulatorState()* – įgyja reikšmę 1, jei pilama į rezervuarą; įgyja reikšmę 0, jei nieko nevyksta. Vadinasi, paveikslukas pasirodo tuomet, kai atvykusi degalų cisterna pildo rezervuarą.



39 pav. Degalų cisternos animacija

47. Automobilių važiavimo maršrutai į degalinę ir iš jos (ir maršrutas, jei nevažiuoja prie kolonėlių) braižomi naudojant *Route* iš *Antimate Transfer* meniu juostos.
48. Kolonėlės braižomos panaudojant *Resource* iš meniu juostos ir pažymint *Seize Area* savybę, kuri leidžia nurodyti, kurioje vietoje yra objektas. Iš viso braižomos keturios kolonėlės. Pirmos kolonėlės animacijos nustatymų pavyzdys pateiktas 40 pav.

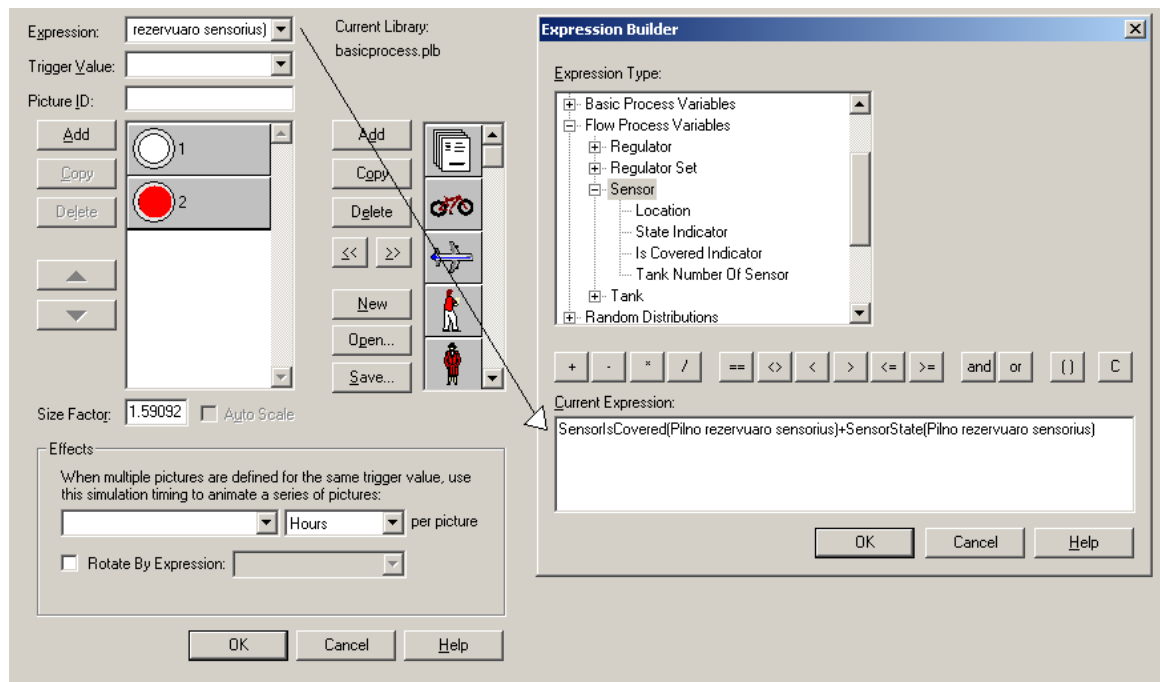


40 pav. Pirmosios kolonėlės animacija

49. Animacijoje trijų jutiklių būsenoms nustatyti yra taikomos dvi funkcijos:
 - *SensorIsCovered()* – įgyja 1, jei rezervuare lygis žemesnis, nei nurodyta; ir įgyja 0, jei rezervuare lygis aukštesnis, nei nurodyta.
 - *SensorState()* – įgyja 1, jei jutiklis aktyvus, ir įgyja 0, jei neaktyvus. Šio modelio atveju jutiklis yra visada aktyvus.

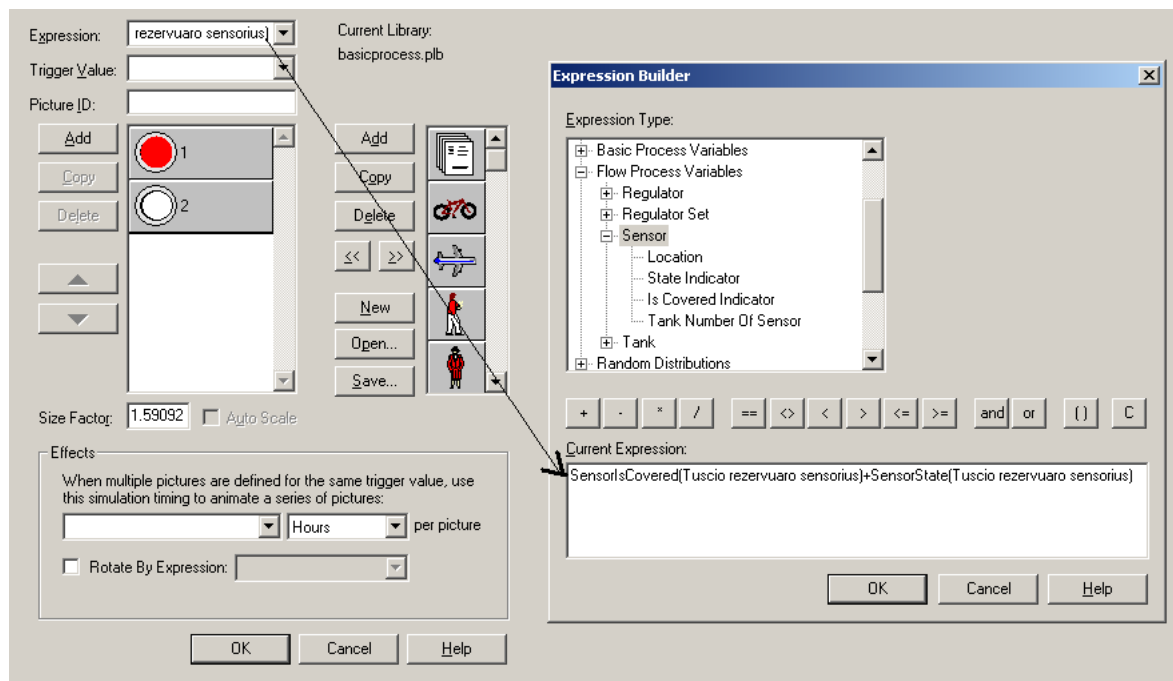
Būsenos suvedamos paspaudus du kartus ant jutiklio paveiksluko. Jei jutiklis sureaguoja, tai įjungiamo raudona spalva. Toliau pateikti visų jutiklių nustatymai:

a. Pilno rezervuaro jutiklis pateiktas 41 pav.



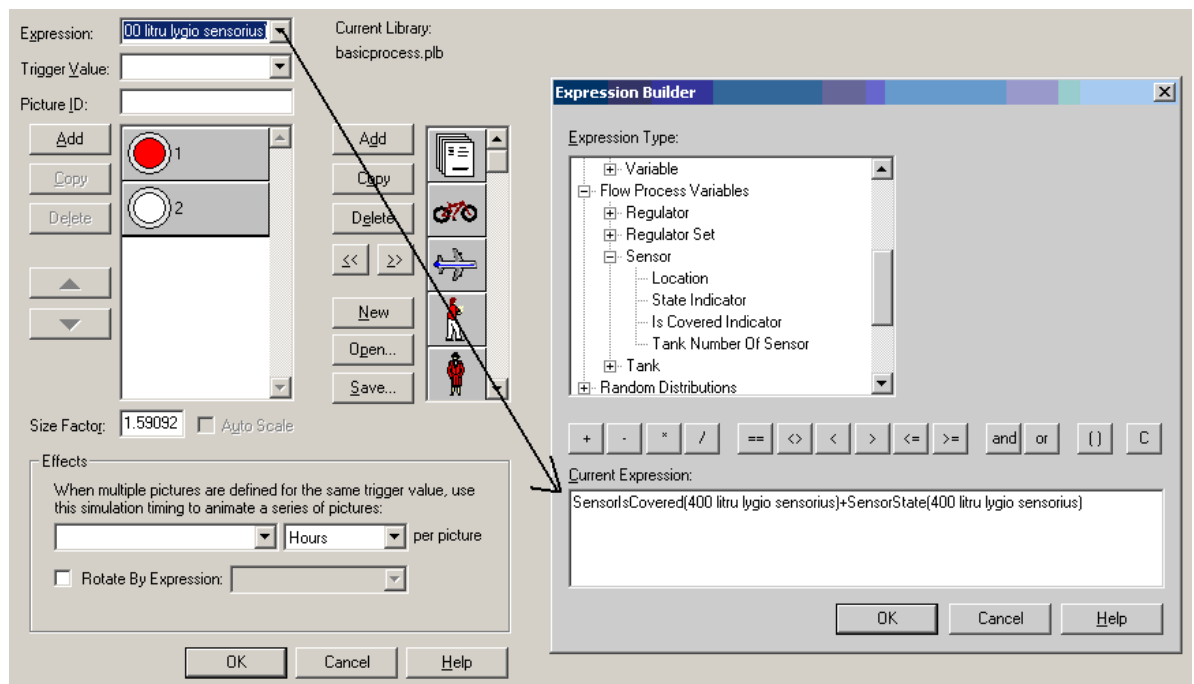
41 pav. Pilno rezervuaro jutiklio animacija

b. Tuščio rezervuaro jutiklis pateiktas 42 pav.



42 pav. Tuščio rezervuaro jutiklio animacija

c. 400 litrų lygio jutiklis pateiktas 43 pav.



43 pav. 400 litrų rezervuaro lygio jutiklio animacija

50. Papildomos užduotys

- Surinkti statistiką, kiek kartų per parą buvo atvažiavusi cisterna;
- Pakoreguoti submodelio „Masinu pildymas degalais“ aprašą įtraukiant kuro kainą. Taip pat paskaičiuoti degalinės gautą sumą per parą;
- Pakoreguoti submodelio „Masinu pildymas degalais“ aprašą įtraukiant kuro kainos sumažinimą per pusę, kai lieka 400 litrų degalų rezervuare.
- Pateikti veiklos diagramas submodeliams „Rezervuaro užpildymas“ ir „išimti iš eiles“.