

Документ подписан простой электронной подписью

Информация о владельце:

ФИО: Локтионова Оксана Геннадьевна

Должность: проректор по учебной работе

Дата подписания: 19.01.2022 18:25:44

Уникальный программный ключ:

0b817ca911e6668ab017a5d426d79e5f1c11eabbf73e947df4a4951fda56d089

**МИНОБРНАУКИ РОССИИ**

Федеральное государственное бюджетное образовательное  
учреждение высшего образования  
«Юго-Западный государственный университет»  
(ЮЗГУ)

Кафедра механики, мехатроники и робототехники



О.Г. Локтионова

2016 г.

**УПРАВЛЕНИЕ МЕХАТРОННЫМИ СИСТЕМАМИ  
И РОБОТАМИ**

**методические указания по выполнению лабораторных работ  
для студентов направления  
Мехатроника и робототехника**

Курск 2016

УДК 681.5.01

Составитель: П.А. Безмен

Рецензент

Кандидат технических наук, доцент кафедры механики,  
мехатроники и робототехники

Е.Н. Политов

**Управление мехатронными системами и роботами:**  
методические указания по выполнению лабораторных работ по  
дисциплине «Управление мехатронными системами и роботами»  
по направлению 15.04.06 - «Мехатроника и робототехника» / Юго-  
Зап. гос. ун-т; сост.: П.А. Безмен; Курск, 2016. 32 с.: ил. 19, табл. 3.

Содержат сведения по вопросам синтеза систем автоматического управления для мехатронных систем и роботов, сведения по методике компьютерного моделирования сетей Петри. Приведены краткие сведения из теории, методика выполнения работ, варианты заданий, примеры.

Методические указания соответствуют требованиям программы, утверждённой учебно-методическим объединением (УМО).

Предназначены для студентов направления 15.04.06 - «Мехатроника и робототехника» всех форм обучения.

Текст печатается в авторской редакции

Подписано в печать . Формат 60x84 1/16.

Усл. печ. л. . Уч.-изд. л. . Тираж 100 экз. Заказ. Бесплатно.

Юго-Западный государственный университет.

305040, Курск, ул. 50 лет Октября, 94.

## Содержание

Лабораторная работа №1. Анализ системы управления.....	4
Лабораторная работа №2. Сети Петри.....	17

## **Лабораторная работа №1. Анализ системы управления**

### **Цель работы**

Целью работы является исследование с помощью пакета MATLAB устойчивости, управляемости, наблюдаемости и минимальности системы управления заданной структурной схемой, и построить ее модель пониженного порядка.

В результате работы у обучающегося формируются компетенции:

ОПК-2 – владение в полной мере основным физико-математическим аппаратом, необходимым для описания и исследования разрабатываемых систем и устройств;

ПК-1 – способность составлять математические модели мехатронных и робототехнических систем, их подсистем, включая исполнительные, информационно-сенсорные и управляющие модули, с применением методов формальной логики, методов сетей Петри.

### **Краткие сведения из теории**

#### *Анализ устойчивости*

Понятие устойчивости является одним из основных в теории управления. Устойчивой называют систему, которая, будучи выведена из состояния равновесия, стремится вновь вернуться в это состояние. Для неустойчивых систем характерна обратная тенденция. Примером неустойчивой системы может быть карандаш, стоящий на острие. Обычно исследование устойчивости систем сводится к анализу устойчивости соответствующих дифференциальных уравнений. У неустойчивых уравнений решение неограниченно возрастает со временем. В теории автоматического управления существуют различные методы анализа устойчивости. Для линейных систем разработаны критерии устойчивости, которые можно разделить на корневые, алгебраические и частотные.

### *Корневой критерий устойчивости*

Для того, чтобы линейная динамическая система была устойчивой, необходимо и достаточно, чтобы все корни ее характеристического уравнения лежали в левой комплексной полуплоскости.

Например, математическая модель системы, описываемой дифференциальным уравнением

$$\ddot{x} + 2\dot{x} + 5x = 0,$$

устойчива, так как корни характеристического уравнения  $p^2 + 2p + 5 = 0$  имеют отрицательные вещественные части  $p_{1,2} = -1 \pm 2i$ , т.е. лежат слева от оси ординат.

### *Алгебраический критерий устойчивости*

Алгебраические критерии устойчивости позволяют судить об устойчивости на основе анализа коэффициентов характеристического полинома. Наибольшую известность получил критерий устойчивости Гурвица. Из него, в частности, следует, что все коэффициенты устойчивого дифференциального уравнения должны быть положительны.

Для уравнений второго порядка это необходимое и достаточное условие.

Для уравнений третьего порядка

$$a_3\ddot{x} + a_2\dot{x} + a_1\dot{x} + a_0x = 0,$$

помимо положительности коэффициентов, должно выполняться дополнительное условие, а именно, произведение средних коэффициентов должно быть больше произведения крайних:

$$a_2a_1 > a_0a_3.$$

Например, уравнение

$$\ddot{x} + 2\dot{x} + 0,5\dot{x} + 3x = 0$$

неустойчиво, так как  $2 \cdot 0,5 < 1 \cdot 3$ .

### *Частотный критерий устойчивости*

Частотные критерии носят графический характер. Они опираются на анализ графиков частотных характеристик - АЧХ, ФЧХ, АФХ (последняя известна также как диаграмма Найквиста). Часть из них позволяет делать заключение об устойчивости замкнутой системы управления по частотным характеристикам разомкнутой системы.

Приведем в качестве примера критерий Найквиста. Обозначим передаточную функцию разомкнутой системы  $Q(p)$  и охватим ее единичной отрицательной обратной связью (рис. 1).

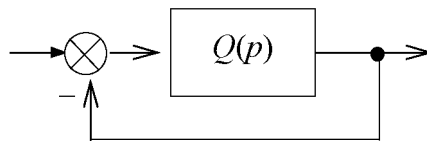


Рис. 1. Система с обратной связью

Пусть известно, что разомкнутая система устойчива. Тогда для устойчивости замкнутой системы необходимо и достаточно, чтобы годограф Найквиста разомкнутой системы не охватывал точку с координатами  $(-1, 0j)$  на комплексной плоскости. В тулбоксе CONTROL пакета MATLAB диаграмма Найквиста системы `sys` строится командой `nyquist(sys)`, указанная точка помечена на ней красным крестиком.

В работе исследуется система управления, заданная структурной схемой, поэтому для анализа устойчивости надо сначала найти ее передаточную функцию. Структурная схема исследуемой системы приведена на рис. 2. В ее состав входят три апериодических звена, а также суммирующие и вычитающие звенья.

В схеме можно указать три пути от входа до выхода с передаточными функциями

$$Q_1(p) = \frac{k_1}{T_1 p + 1}, \quad Q_2(p) = \frac{-k_2}{T_2 p + 1} \cdot \frac{k_3}{T_3 p + 1}, \quad Q_3(p) = \frac{-k_1}{T_1 p + 1} \cdot \frac{ak_3}{T_3 p + 1}.$$

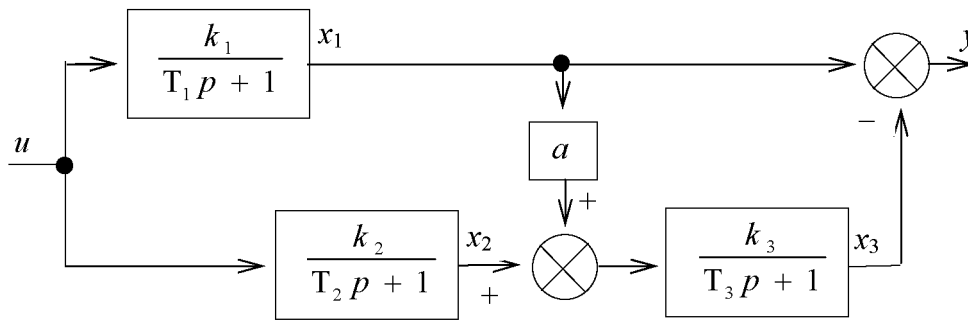


Рис. 2. Структура исследуемой системы

Общая передаточная функция получается, как их сумма

$$Q(p) = Q_1(p) + Q_2(p) + Q_3(p) = \frac{B(p)}{A(p)}. \quad (1)$$

Здесь  $A(p)$  и  $B(p)$  – некоторые полиномы от  $p$  (свои для каждого варианта). Характеристический полином системы равен знаменателю этой передаточной функции

$$A(p) = (T_1p + 1)(T_2p + 1)(T_3p + 1) \quad (2)$$

На основе приведенных выше критериев устойчивости можно заключить, что если коэффициенты  $T_1$ ,  $T_2$ ,  $T_3$  положительны, то система будет устойчивой. Анализ устойчивости замкнутой системы можно осуществить с помощью критерия Найквиста.

### *Анализ управляемости и наблюдаемости*

Понятия управляемости и наблюдаемости широко используются в современной теории автоматического управления. Линейная система называется управляемой, если с помощью входного сигнала ее можно перевести из начала координат в любую точку пространства состояний. Система называется наблюдаемой, если по измерениям входного и выходного сигналов можно однозначно определить её начальное состояние.

Анализ управляемости и наблюдаемости выполняется с помощью матриц управляемости и наблюдаемости или с помощью

грамианов управляемости и наблюдаемости. Те и другие строятся по описанию в пространстве состояний

$$\dot{X} = AX + bu, \quad y = cX. \quad (3)$$

Чтобы получить такое описание, воспользуемся структурной схемой, приведенной на рис. 2. Выпишем операторные уравнения для каждого блока схемы

$$x_1 = \frac{k_1}{T_1 p + 1} u, \quad x_2 = \frac{k_2}{T_2 p + 1} u, \quad x_3 = \frac{k_3}{T_3 p + 1} (ax_1 + x_2)$$

и преобразуем их к виду:

$$\begin{aligned} T_1 p x_1 &= -x_1 + k_1 u, \\ T_2 p x_2 &= -x_2 + k_2 u, \\ T_3 p x_3 &= ak_3 x_1 + k_3 x_2 - x_3. \end{aligned}$$

Выполняя обратное преобразование Лапласа, получаем систему линейных дифференциальных уравнений первого порядка:

$$\begin{aligned} T_1 \dot{x}_1 &= -x_1 + k_1 u, \\ T_2 \dot{x}_2 &= -x_2 + k_2 u, \\ T_3 \dot{x}_3 &= ak_3 x_1 + k_3 x_2 - x_3. \end{aligned}$$

Кроме того, имеем алгебраическое уравнение для выходного сигнала  $y = x_1 - x_3$ .

Переписывая эти уравнения в матричной форме (3), получаем следующие выражения для матриц A, b, c:

$$A = \begin{bmatrix} a_{11} & 0 & 0 \\ 0 & a_{22} & 0 \\ a_{31} & a_{32} & a_{33} \end{bmatrix},$$

$$b = \begin{bmatrix} b_1 \\ b_2 \\ 0 \end{bmatrix},$$



$$c = [c_1 \ 0 \ c_3],$$

где ненулевые элементы матриц имеют вид:

$$a_{11} = -1/T_1, \ a_{22} = -1/T_2, \ a_{31} = ak_3/T_3, \ a_{32} = k_3/T_3, \ a_{33} = -1/T_3, \ b_1 = k_1/T_1, \\ b_2 = k_2/T_2, \ c_1 = 1, \ c_3 = -1.$$

Теперь можно перейти к анализу управляемости и наблюдаемости. Сформируем на основе матриц  $A$ ,  $b$ ,  $c$  две вспомогательные матрицы

$$R = [b, Ab, \dots, A^{n-1}b], \quad D = \begin{bmatrix} c \\ cA \\ \dots \\ cA^{n-1} \end{bmatrix}.$$

Матрицы  $R$  и  $D$  называются соответственно матрицей управляемости и матрицей наблюдаемости системы. В пакете MATLAB их можно построить с помощью команд **ctrb** и **obsv**.

Критерий управляемости. Для того чтобы система (3) была управляемой, необходимо и достаточно, чтобы матрица управляемости имела полный ранг  $\text{rank}R = n$ .

Критерий наблюдаемости. Для того чтобы система (3) была наблюдаемой, необходимо и достаточно, чтобы матрица наблюдаемости имела полный ранг  $\text{rank}D = n$ .

В случае систем с одним входом и одним выходом матрицы  $R$  и  $D$  квадратные, поэтому для проверки управляемости и наблюдаемости достаточно вычислить определители матриц  $R$  и  $D$ . Если они не равны нулю, то матрицы имеют полный ранг.

Другой способ проверки управляемости и наблюдаемости опирается на вычисление грамианов управляемости и наблюдаемости. Так называются симметричные квадратные матрицы  $W_c$  и  $W_o$ , определяемые равенствами

$$W_c = \int_0^{\infty} e^{At} b b^T e^{A^T t} dt, \quad W_o = \int_0^{\infty} e^{A^T t} c^T c e^{At} dt$$

В пакете MATLAB их можно найти с помощью команд типа  $\text{gram}(\text{sys}, 'c')$ ,  $\text{gram}(\text{sys}, 'o')$ . Необходимые и достаточные условия управляемости и наблюдаемости имеют вид  $\det W_c \neq 0$ ,  $\det W_o \neq 0$ .

### *Анализ минимальности моделей*

Одной и той же передаточной функции  $Q(p)$  можно сопоставить целый класс эквивалентных реализаций в пространстве состояний, характеризуемых различными тройками матриц  $(A, b, c)$  разных размеров. Реализация называется минимальной, если размер ее матрицы  $A$  наименьший среди всех эквивалентных реализаций. Поиск такой реализации имеет практический смысл, так как ее моделирование на ЭВМ требует меньших вычислительных затрат.

Для анализа минимальности реализации нужно проверить ее управляемость и наблюдаемость.

Критерий минимальности. Для того чтобы реализация (3) была минимальной, необходимо и достаточно, чтобы она была управляемой и наблюдаемой одновременно.

Таким образом, анализ минимальности конкретной реализации сводится к проверке пары критериев  $\text{rank}R = n$ ,  $\text{rank}D = n$ .

Если хотя бы один из рангов меньше  $n$ , то реализация неминимальна. Размерность эквивалентной минимальной реализации  $n_0$  определяется по формуле  $n_0 = \text{rank}(RD)$ .

Анализ минимальности с помощью грамианов управляемости и наблюдаемости проводится аналогично. Критерием минимальности служит выполнение условия  $\det(W_c W_o) \neq 0$ , эквивалентного паре условий  $\det W_c \neq 0$ ,  $\det W_o \neq 0$ .

Если исходное описание реализации оказалось неминимальным, то возникает задача перехода к минимальной реализации. Чтобы решить ее, сначала перейдем от описания в пространстве состояний (3) к передаточной функции

$$Q(p) = c(pE - A)^{-1}b = \frac{B(p)}{A(p)}. \quad (4)$$

Далее нужно выделить общий множитель в числителе и знаменателе передаточной функции  $Q(p)$  и сократить на него. Эта процедура известна, как сокращение совпадающих нулей и полюсов системы.

Отметим ряд соотношений между элементами матриц  $A$ ,  $b$ ,  $c$  и коэффициентами передаточной функции, вытекающих из формулы (4). Знаменатель передаточной функции совпадает с характеристическим полиномом матрицы  $A$

$$A(p) = \det(pE - A) = p^n + a_{n-1}p^{n-1} + \dots + a_1p + a_0$$

Его корни  $\lambda_1, \dots, \lambda_n$  равны собственным числам матрицы  $A$ , а коэффициенты  $a_0$  и  $a_{n-1}$  с точностью до знака равны ее определителю и следу

$$\begin{aligned} a_0 &= \det(-A) = (-1)^n \lambda_1 \lambda_2 \dots \lambda_n, \\ -a_{n-1} &= \text{tr}A = a_{11} + a_{22} + \dots + a_{nn} = \lambda_1 + \lambda_2 + \dots + \lambda_n. \end{aligned} \quad (5)$$

Старший и младший коэффициенты числителя передаточной функции

$$B(p) = b_{n-1}p^{n-1} + \dots + b_1p + b_0$$

удовлетворяют соотношениям

$$b_{n-1} = cb, \quad b_0/a_0 = -c A^{-1} b. \quad (6)$$

Величина  $k_0 = b_0/a_0$  называется статическим коэффициентом усиления. Она равна установившемуся значению переходной функции системы. В пакете MATLAB для вычисления статического коэффициента усиления имеется команда **dcgain** (от direct current gain - коэффициент усиления по постоянному току). В

ее основу положена формула  $k_0 = Q(0)$ . Приведенные соотношения удобно использовать для контроля вычислений.

Получение минимальной реализации в пакете MATLAB осуществляется командой **minreal**, для вычисления нулей и полюсов можно использовать функции **zero**, **pole**, **pzmap**, **zpk**. Аргументом во всех случаях служит исследуемая система **sys**, предварительно сформированная командами **ss** или **tf**.

### Методика выполнения работы

1. Нарисовать схему рис. 2 для своего варианта и найти ее передаточную функцию по формуле (1).

2. Проверить устойчивость системы, используя алгебраический и корневой критерии.

3. Получить для своего варианта схемы описание в пространстве состояний вида (3) и выписать матрицы  $A$ ,  $B$ ,  $C$ .

4. Найти матрицы управляемости и наблюдаемости системы, определить их ранги и сделать вывод об управляемости, наблюдаемости и минимальности системы.

5. Найти передаточную функцию по формуле (4), и сравнить ее с полученной в п.1. Проверить выполнение соотношений (5), (6). Тремя способами (по структурной схеме, передаточной функции и описанию в пространстве состояний) найти статический коэффициент усиления системы.

6. Определить порядок минимальной реализации и найти ее передаточную функцию, выполнив сокращение нулей и полюсов. Найти реакцию минимальной реализации на единичный скачок и построить ее график.

7. Привести программы на языке MATLAB для выполнения пунктов 2 - 6 и краткое описание назначения и синтаксиса команд **rank**, **ss**, **tf**, **zpk**, **zero**, **pole**, **pzmap**, **ctrb**, **obsv**, **minreal**, **dcgain**.

## Порядок выполнения работы

1. В диалоговом режиме пакета MATLAB ввести матрицы  $A$ ,  $b$ ,  $c$  и сформировать ss-описание системы  $sys = ss(A, b, c, 0)$ . Используя команды **pole**, **eig**, **pzmap**, найти полюсы системы и получить график их расположения на комплексной плоскости. Сделать вывод об устойчивости. Построить диаграмму Найквиста системы и сделать заключение об устойчивости системы, охваченной обратной связью (рис. 1).

2. Найти матрицы управляемости и наблюдаемости, вычислить их определители и ранги. Сделать вывод об управляемости, наблюдаемости и минимальности. Найти грамианы управляемости и наблюдаемости.

3. С помощью команд **tf** и **zpk** перейти к передаточной функции. Двумя способами получить минимальную реализацию (сокращая нули и полюса, и с помощью команды **minreal**). Сравнить переходные функции исходной и минимальной реализаций, а также их статический коэффициент усиления.

4. Набрать схемы исходной и минимальной реализаций в SIMULINK и сравнить их реакции на одинаковые входные сигналы.

## Контрольные вопросы

1. Найти передаточные функции систем, заданных в пространстве состояний тройкой матриц:

$$a) A = \begin{bmatrix} -1 & 0 \\ 1 & -2 \end{bmatrix}, \quad b = \begin{bmatrix} 1 \\ 0 \end{bmatrix}, \quad c = [1 \ 1], \quad б) A = \begin{bmatrix} -3 & 1 \\ 1 & -2 \end{bmatrix}, \quad b = \begin{bmatrix} 1 \\ 1 \end{bmatrix}, \quad c = [0 \ 1],$$

$$в) A = \begin{bmatrix} -4 & 1 \\ 2 & -3 \end{bmatrix}, \quad b = \begin{bmatrix} 0 \\ 1 \end{bmatrix}, \quad c = [1 \ 2], \quad г) A = \begin{bmatrix} -4 & 2 \\ 1 & -3 \end{bmatrix}, \quad b = \begin{bmatrix} 1 \\ 2 \end{bmatrix}, \quad c = [0 \ 1].$$

2. Найти ранги матриц управляемости и наблюдаемости для систем из п.1.

3. Нарисовать структурные схемы систем, матрицы которых приведены ниже.

№	1	2	3	4	5	6	7
<b>A</b>	0 1 0 0	0 1 1 0	0 1 -1 0	0 1 1 1	1 0 0 2	1 0 0 1	1 1 1 1
<b>b<sup>T</sup></b>	0 1	0 1	0 1	0 1	1 1	0 1	0 1
<b>c</b>	0 1	0 1	0 1	0 1	1 1	0 1	1 1

Определить, какие из них являются:

- а) устойчивыми; б) управляемыми; в) наблюдаемыми; г) минимальными.

4. Система управления задана структурной схемой, показанной на рис. 3.

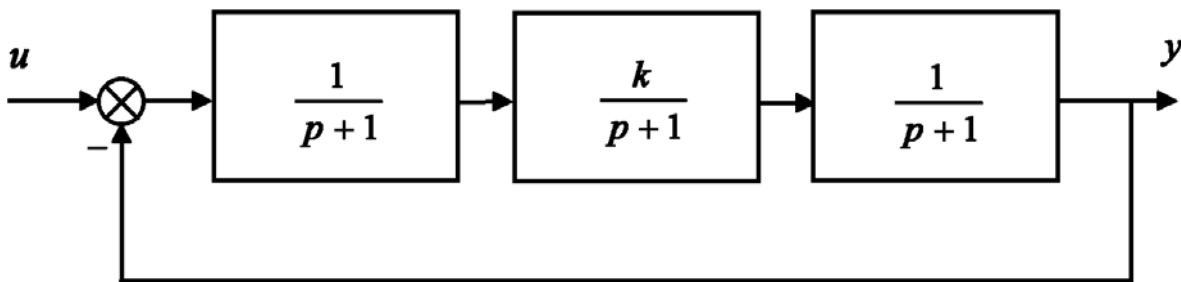


Рис. 3. Система управления третьего порядка

Требуется:

- найти статический коэффициент усиления схемы;
- найти передаточную функцию схемы и проанализировать ее устойчивость;
- найти описание схемы в пространстве состояний; построить матрицы управляемости и наблюдаемости, сделать вывод о минимальности;
- выяснить, при каких значениях коэффициента  $k$  схема будет устойчивой, управляемой, наблюдаемой.

**Варианты заданий**

Таблица 1

<b>№</b>	<b>1</b>	<b>2</b>	<b>3</b>	<b>4</b>	<b>5</b>	<b>6</b>	<b>7</b>	<b>8</b>	<b>9</b>	<b>10</b>	<b>11</b>	<b>12</b>	<b>13</b>	<b>14</b>	<b>15</b>	<b>16</b>	<b>17</b>	<b>18</b>	<b>19</b>	<b>20</b>	<b>21</b>
<b><math>k_1</math></b>	1	3	2	1	4	3	3	3	2	3	2	2	4	2	2	2	3	4	3	3	1
<b><math>T_1</math></b>	1	4	2	3	1	3	3	2	2	4	1	1	2	3	5	2	2	4	1	1	4
<b><math>k_2</math></b>	5	3	6	1	6	4	1	2	6	4	3	7	5	4	6	5	1	3	2	3	4
<b><math>T_2</math></b>	1	4	2	3	1	3	3	2	2	4	2	2	2	4	5	2	2	4	1	1	4
<b><math>k_3</math></b>	5	2	2	4	3	3	6	6	2	5	6	4	7	2	2	2	4	4	5	5	4
<b><math>T_3</math></b>	4	2	1	2	4	5	3	3	4	3	5	5	2	2	5	2	4	2	2	4	3
<b><math>a</math></b>	-2	-1	-1	-2	-1	-1	-1	-2	-1	-2	-2	-1	-2	-1	-1	-1	-2	-2	-2	-1	-1

## Литература

1. Мироновский, Л.А. Моделирование линейных систем: Методические указания к выполнению лабораторных работ / С.-Пб.: ГУАП, 2007.
2. Эльсгольц Л.Э. Обыкновенные дифференциальные уравнения: Учебник для вузов / С.-Пб.: Изд-во «Лань», 2002.
3. Есипов А.А., Сазонов Л.И., Юдович В.И. Практикум по обыкновенным дифференциальным уравнениям. - М.: Вузовская книга, 2001.
4. Романко В. К. Разностные уравнения: Учебное пособие / -М.: БИНОМ. Лаборатория знаний, 2006.
5. Мироновский Л.А. Моделирование разностных уравнений: Учебное пособие / С.-Пб.: ГУАП, 2004.
6. Мироновский Л.А., Петрова К.Ю. Введение в MATLAB: Учебное пособие / С.-Пб.: ГУАП, 2006.



## Лабораторная работа №2. Сети Петри

### Цель работы

Целью работы является изучение работы сетей Петри, овладение навыками моделирования сетей Петри в среде Mathworks MATLAB/SIMULINK.

В результате работы у обучающегося формируются компетенции:

ОПК-2 – владение в полной мере основным физико-математическим аппаратом, необходимым для описания и исследования разрабатываемых систем и устройств;

ПК-1 – способность составлять математические модели мехатронных и робототехнических систем, их подсистем, включая исполнительные, информационно-сенсорные и управляющие модули, с применением методов формальной логики, методов сетей Петри.

### Постановка задачи

Диспетчер управляет внутризаводским транспортом и имеет в своем распоряжении два грузовика. Заявки на перевозки поступают к диспетчеру каждые  $\tau_1$  мин.

Интервал времени между двумя заявками -  $\tau_1$ , распределен по равномерному закону в интервале  $\tau_1 = a_1 \pm b_1$ . Вероятность запроса диспетчером грузовика -  $P_1$ , и, если тот свободен, то принимает заявку, в противном случае запрашивается другой грузовик. Это сеанс связи, который длится ровно  $\tau$  мин. Таким образом, диспетчер продолжает сеансы связи, пока один из грузовиков не освободится. Диспетчер допускает накопление у себя до пяти заявок (максимальное количество заявок  $N_1 = 5$ , если  $N_1 > 5$ , то последующие заявки не рассматриваются).

Интервал времени выполнения заявки грузовиком  $\tau_2$  распределен по равномерному закону в интервале  $\tau_2 = a_2 \pm b_2$ . Грузовики выполняют заявки на перевозку за  $\tau_2$  мин.

### Задание

В соответствии с вариантом задания (таблица 2) смоделировать работу внутризаводского транспорта в течение  $T$  часов ( $T \neq 0$ ), считая, что первая заявка поступает в момент времени равный 0.

В результате моделирования требуется определить следующие характеристики:

1. Количество заявок, завершивших обслуживание;
2. Количество заявок, получивших отказ;
3. Коэффициенты загрузки грузовиков  $k_1$  и  $k_2$ , которые соответственно равны:  $k_1 = \frac{t_{\text{раб.1}}}{t_{\text{общ}}}$  и  $k_2 = \frac{t_{\text{раб.2}}}{t_{\text{общ}}}$ .

Таблица 2 - Варианты задания

№	$\tau$	$a_1$	$b_1$	$a_2$	$b_2$	$P_1$	$T$
1	3	5	3	60	10	0.3	10
2	3	6	3	70	20	0.4	
3	3	7	3	60	30	0.5	
4	3	8	3	70	10	0.3	
5	3	7	3	80	20	0.4	
6	3	6	3	90	30	0.5	
7	3	5	3	40	10	0.3	
8	3	6	3	50	20	0.4	
9	3	7	3	60	30	0.5	
10	3	9	3	50	10	0.3	
11	3	8	3	40	20	0.4	
12	3	7	3	60	30	0.5	
13	3	6	3	70	10	0.3	
14	3	5	3	60	20	0.4	
15	3	8	3	70	30	0.5	
16	3	9	3	80	10	0.3	
17	3	5	3	90	20	0.4	
18	3	6	3	40	30	0.5	
19	3	7	3	50	10	0.3	
20	3	8	3	60	20	0.4	
21	3	7	3	50	30	0.5	
22	3	6	3	40	10	0.3	
23	3	5	3	60	20	0.4	
24	3	6	3	70	30	0.5	
25	3	7	3	60	10	0.3	

### **Содержание отчета**

1. Название работы.
2. Цель работы.
3. Содержание работы.
4. Результаты выполнения моделирования в среде MATLAB/SIMULINK на ЭВМ.

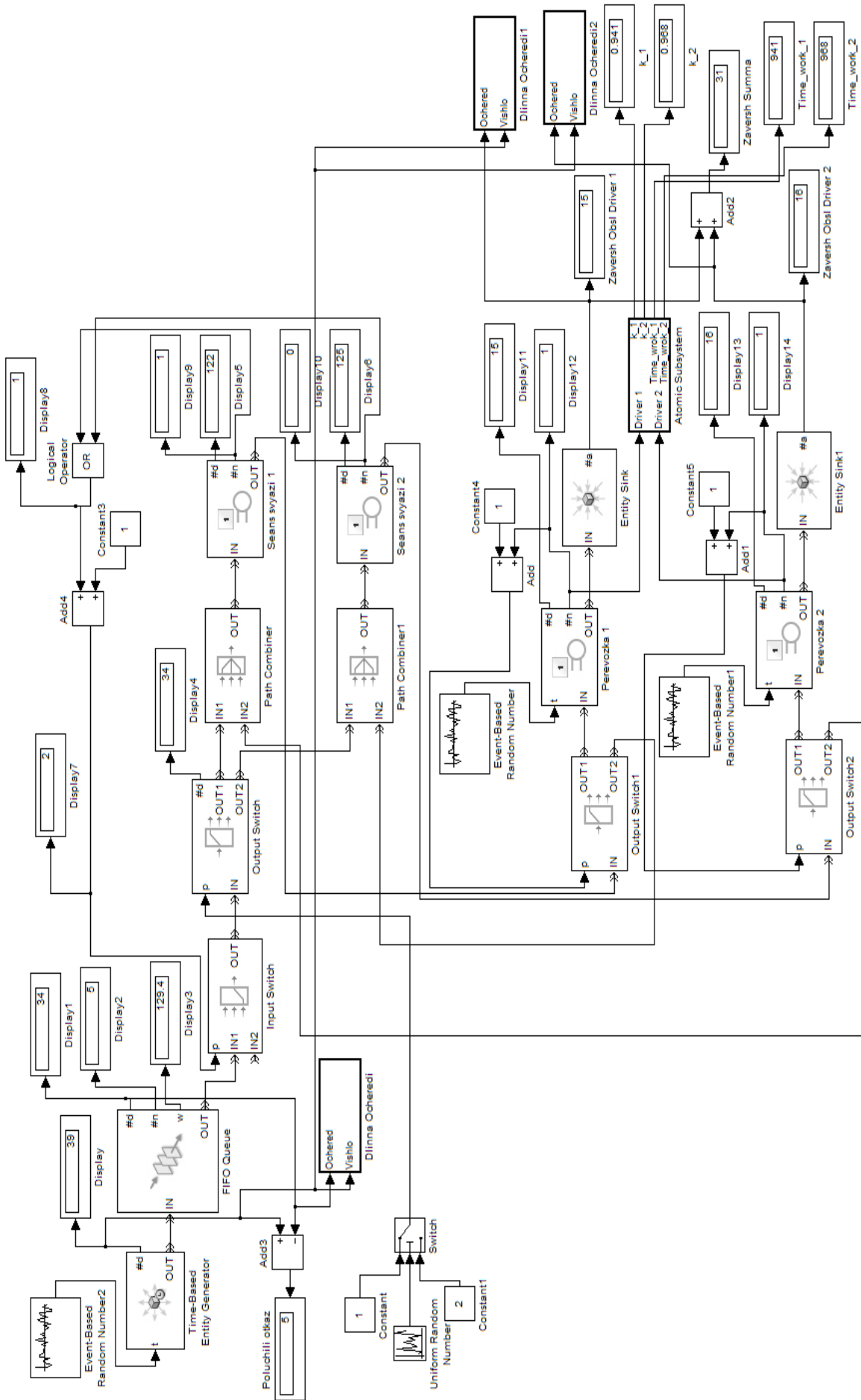


Рис.4 Схема в среде SIMULINK

### Методика выполнения работы

В процессе создания схемы (рис. 4) в среде MATLAB/SIMULINK используются следующие блоки библиотеки Sim Events:

- Time based Entity Generator – блок формирования распределенных во времени сигналов, имитирующих последовательность поступающих на вход системы запросов на обслуживание;
- Event-Based random Number – блок формирования временных интервалов, используемых в качестве времен обслуживания заявок;
- Path Combiner – блок для объединения входных сигналов;
- Input Switch – управляемый ключ;
- Output Switch – управляемый ключ;
- FIFO Queue – блок, реализующий очередь обслуживания заявок;
- Single Server – обслуживающий блок;
- Entity Sink – приемник обслуженных заявок.

Система внутризаводского транспорта состоит из следующих элементов:

- *Time-Based Entity Generator* – генератор заявок через случайные промежутки времени от 2 до 8 (определяются блоком *Event-Based Random Number*)

**Time-Based Entity Generator**

Generate entities using intergeneration times from a signal or a statistical distribution.

Use the Response when blocked and Response when unblocked parameters to determine how the block responds when the OUT port becomes blocked and later unblocked.

Entity type:

Response when blocked:

Response when unblocked:

Entity Generation

Generate entities with:

Generation event priority:

Generate entity at simulation start

Рис. 5 Окно настроек Time-Based Entity Generator

**Event-Based Random Number (mask) (link)**

Generate random numbers from the specified distribution, parameters, and initial seed.

**Parameters**

Distribution:

Minimum:

Maximum:

Initial seed:

Рис. 6 Окно настроек Event-Based Random Number

- *FIFO Queue* – очередь, определяющая максимальное количество заявок  $N_1 = 5$ . Если  $N_1 > 5$ , то последующие заявки не рассматриваются.

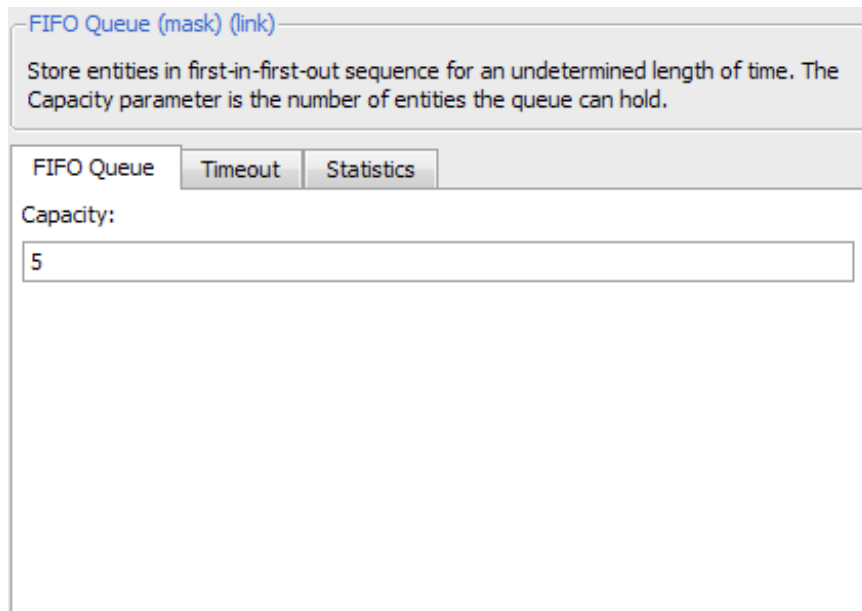


Рис. 7 Окно настроек FIFO Queue

- *Input Switch* – управляемый ключ, пропускающий заявки на выполнение только в том случае, если нет в данный момент сеанса связи ни с одним грузовиком. Управляется схемой из блока *OR* и математической суммы, для переключения входов, если «1 + 1» - второй ход, то заявки не поступают. Если «0 + 1» - заявка поступает на обработку.

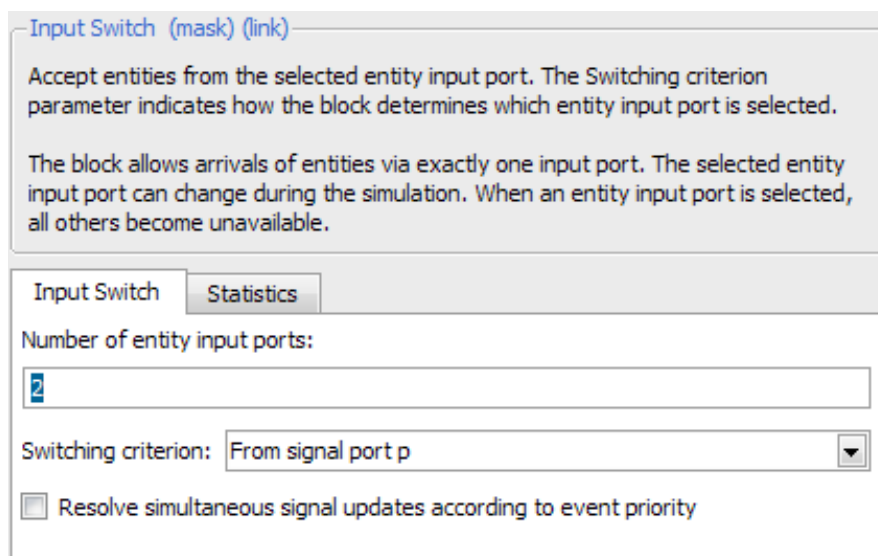


Рис. 8 Окно настроек Input Switch

- *Output Switch* – управляемый ключ, определяет, к какому из грузовиков заявка попадет на обработку. Ключ управляется

схемой на основе блока *Uniform Random Number* и блока *Switch*, который подключает первый выход *Output Switch*, если входная величина больше 0,7.

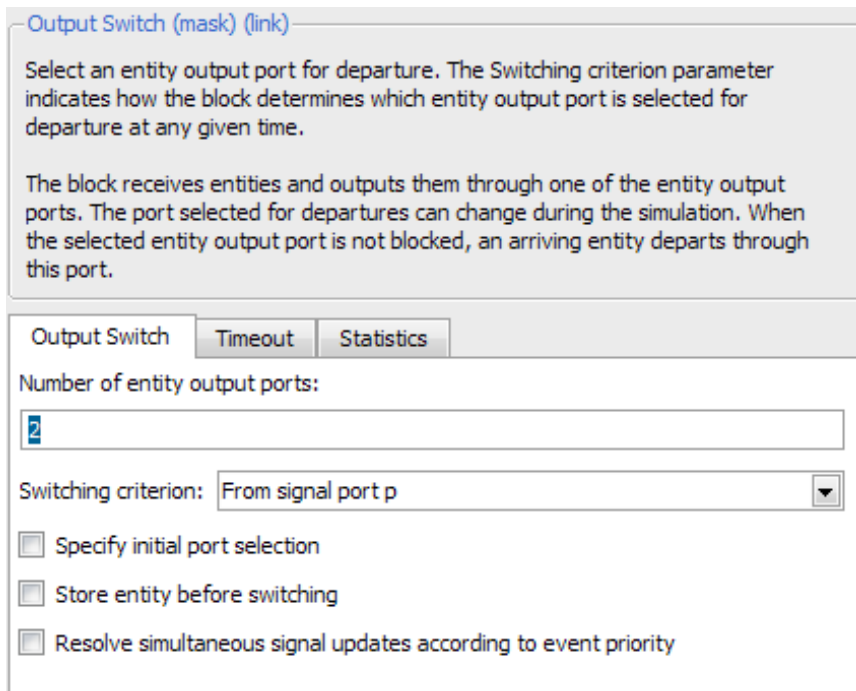


Рис. 9 Окно настроек Output Switch

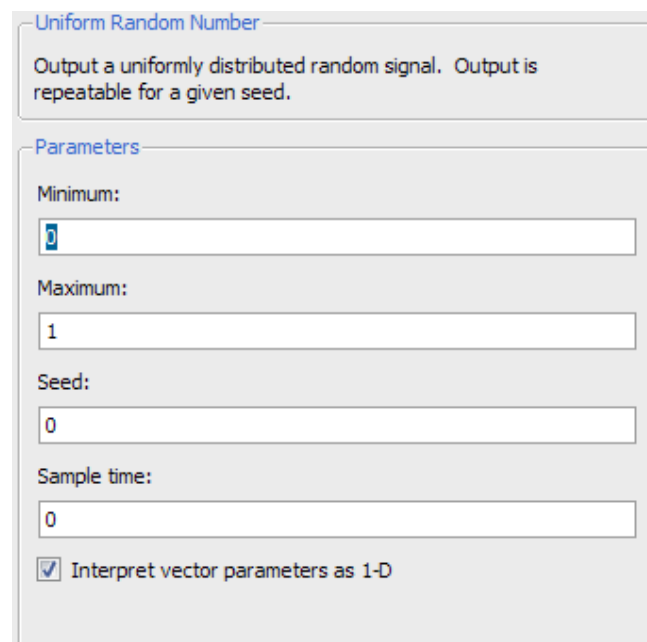


Рис. 10 Окно настроек Uniform Random Number



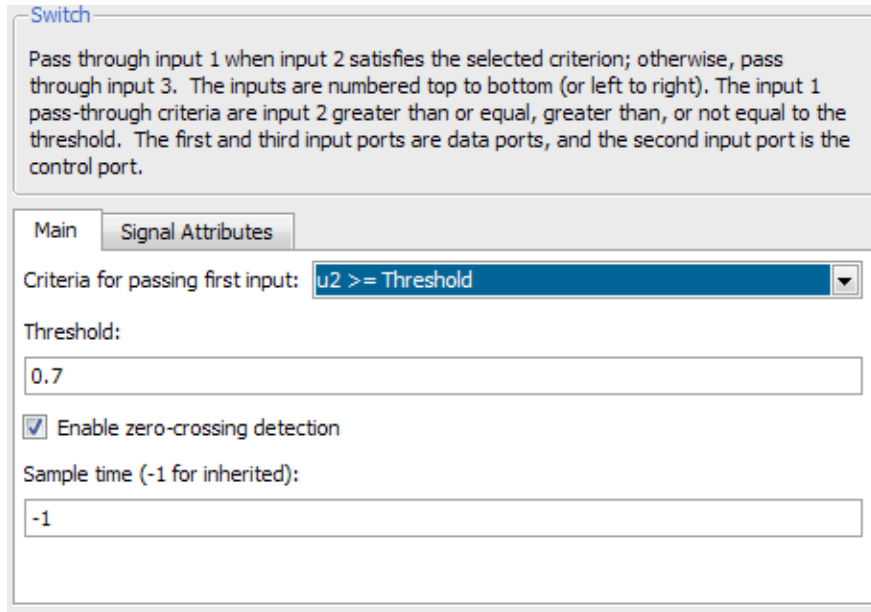


Рис. 11 Окно настроек Switch

- *Path Combiner*, *Path Combiner1* – объединяют потоки заявок к грузовикам.

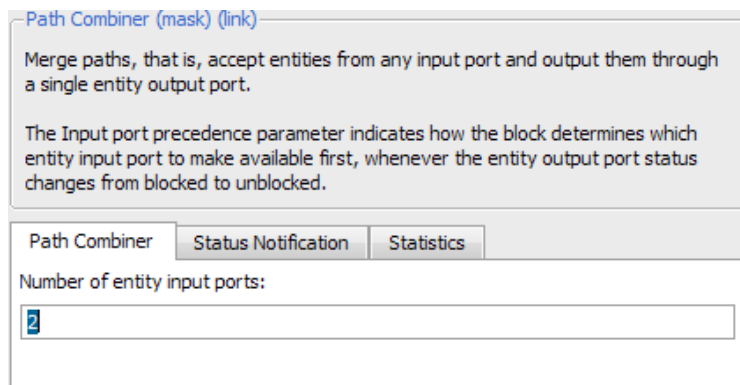


Рис. 12 Окно настроек Path Combiner

- *Seans svyazi 1*, *Seans svyazi 2* – имитация сеансов связи с грузовиками в течении фиксированного времени в τ с (блок *Single Server*).

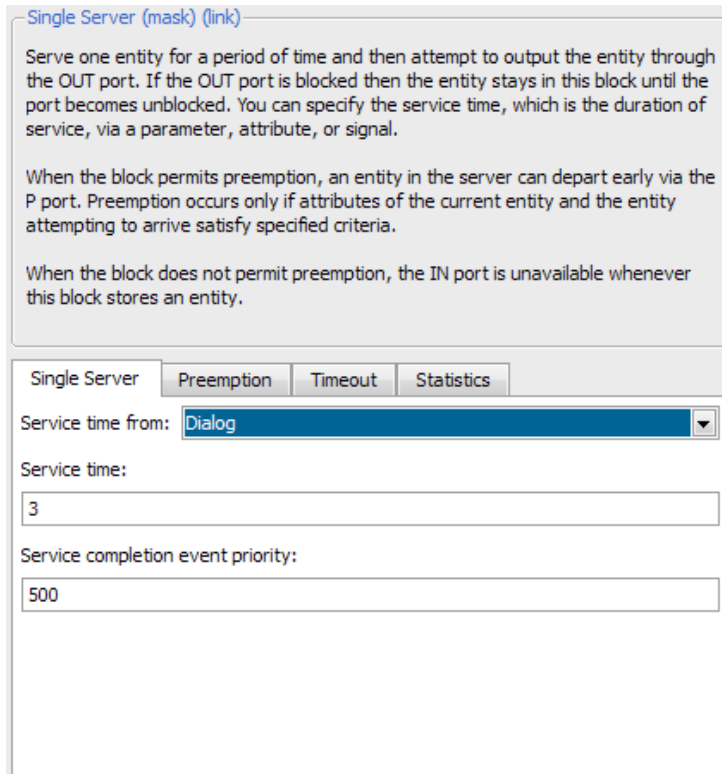


Рис. 13 Окно настроек Single Server

- *Output Switch1*, *Output Switch2* – управляемые ключи, распределяют заявки между грузовиками в зависимости от их занятости. Так, если заявка предназначалась грузовику №1, а этот грузовик «занят», то заявка направляется грузовику №2. Для управления ключами считывается параметр статистики #n блоков *Perevozka1* и *Perevozka2* и к нему прибавляется единица (для адекватного управления выходами ключей).

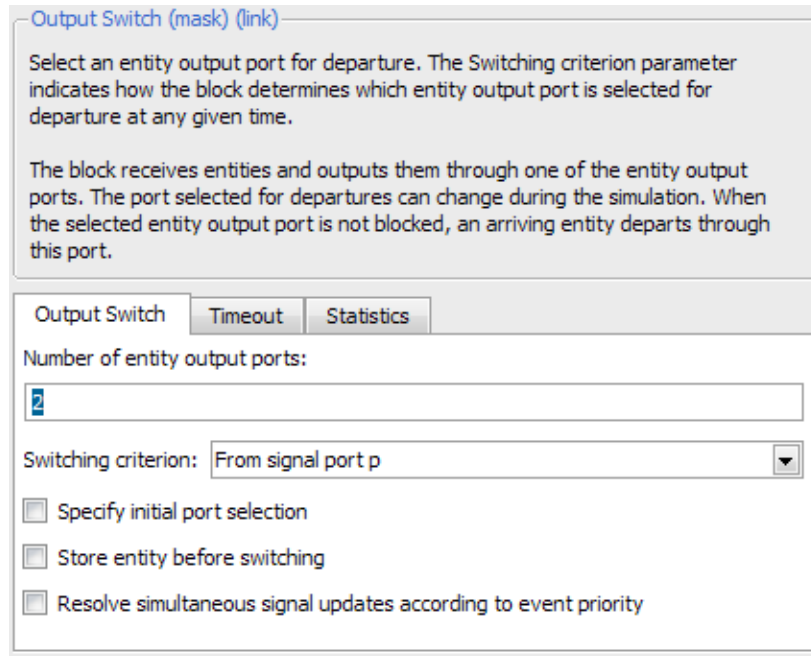


Рис. 14 Окно настроек Output Switch

- *Perevozka1*, *Perevozka2* – имитируют процесс перевозки груза первым и вторым грузовиком соответственно в течении случайного промежутка времени от 50 до 70, определяемого блоками Event-Based Random Number и Event-Based Random Number1.

[Single Server \(mask\) \(link\)](#)

Serve one entity for a period of time and then attempt to output the entity through the OUT port. If the OUT port is blocked then the entity stays in this block until the port becomes unblocked. You can specify the service time, which is the duration of service, via a parameter, attribute, or signal.

When the block permits preemption, an entity in the server can depart early via the P port. Preemption occurs only if attributes of the current entity and the entity attempting to arrive satisfy specified criteria.

When the block does not permit preemption, the IN port is unavailable whenever this block stores an entity.

Single Server   Preemption   Timeout   Statistics

Service time from:

Service completion event priority:

Рис. 15 Окно настроек Single Server

[Event-Based Random Number \(mask\) \(link\)](#)

Generate random numbers from the specified distribution, parameters, and initial seed.

Parameters

Distribution:

Minimum:

Maximum:

Initial seed:

Рис. 16 Окно настроек Event-Based Random Number

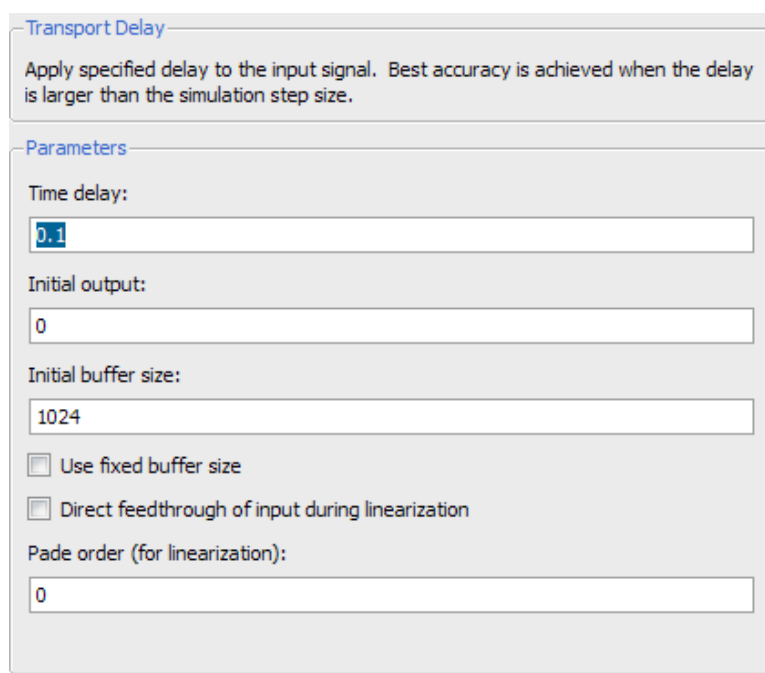
## Подсистема «Длина очереди»

Подсистемы осуществляют запись в рабочую область номеров заявок, завершивших обслуживание и получивших отказ. Для записи номеров заявок, получивших отказ, на вход 1 подается параметр статистики #d блока FIFO Queue (очередь заявок), а на вход 2 подается параметр статистики #d блока Time-Based Entity Generator (количество сгенерированных заявок).

Для записи номеров заявок, завершивших обслуживание, на вход 1 подается параметр статистики #a блока Entity Sink (количество выполненных заявок, отдельно для каждого грузовика), а на вход 2 подается параметр статистики #d блока Time-Based Entity Generator (количество сгенерированных заявок).

Триггерная подсистема осуществляет запись в рабочую область.

Параметры блока Transport Delay (необходим для получения прямоугольных импульсов для включения триггерной подсистемы) приведены на рис. 17.



Transport Delay

Apply specified delay to the input signal. Best accuracy is achieved when the delay is larger than the simulation step size.

Parameters

Time delay:  
0.1

Initial output:  
0

Initial buffer size:  
1024

Use fixed buffer size

Direct feedthrough of input during linearization

Pade order (for linearization):  
0

Рис. 17 Окно настроек Transport Delay

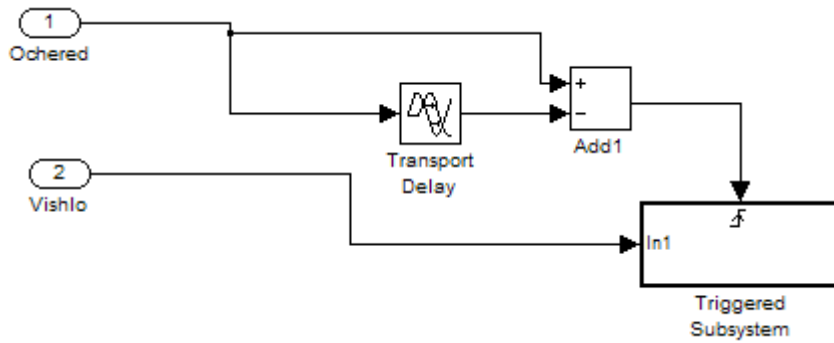


Рис. 18 Структура подсистемы «Длина очереди»

### Подсистема «Работа»

Подсистема «Работа» осуществляет вычисление коэффициентов загрузки грузовиков  $k_1$  и  $k_2$ . На входы 1 и 2 подсистемы подаются соответственно значения параметров статистики  $\#n$  блоков *Perevozka1* и *Perevozka2* (текущее состояние блока – 1 или 0). На выходах 1 и 2 получаем время работы первого и второго грузовиков, а на выходах 3 и 4 получаем значения коэффициентов  $k_1$  и  $k_2$ .

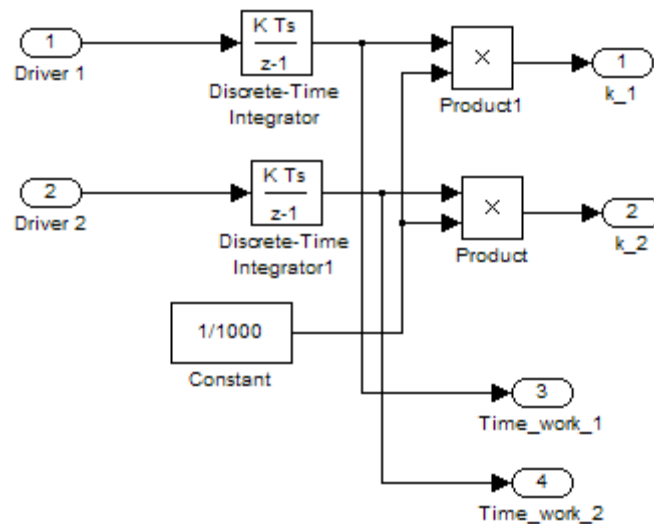


Рис. 19 Структура подсистемы «Работа»

## Задания для самостоятельной работы

Варианты заданий для самостоятельной работы представлены в таблице 3.

Таблица 3 - Варианты заданий

№	$\tau$	$a_1$	$b_1$	$a_2$	$b_2$	$P_1$	T
1	3	6	3	80	30	0.5	10
2	3	9	3	50	10	0.3	20
3	3	7	3	60	30	0.5	20
4	3	6	3	50	20	0.4	40
5	3	5	3	40	10	0.3	10
6	3	6	3	90	30	0.5	20
7	3	7	3	80	20	0.4	10
8	3	7	3	50	10	0.3	20
9	3	6	3	40	30	0.5	30
10	3	5	3	90	20	0.4	40
11	3	8	3	70	10	0.3	50
12	3	7	3	60	30	0.5	20
13	3	6	3	70	20	0.4	10
14	3	5	3	60	10	0.3	40
15	3	9	3	80	10	0.3	10
16	3	8	3	70	30	0.5	30
17	3	7	3	60	10	0.3	50
18	3	8	3	40	20	0.4	30
19	3	5	3	60	20	0.4	20
20	3	6	3	40	10	0.3	10
21	3	7	3	50	30	0.5	10
22	3	8	3	60	20	0.4	50
23	3	5	3	60	20	0.4	40
24	3	6	3	70	10	0.3	10
25	3	7	3	60	30	0.5	20

### **Контрольные вопросы**

1. Функционирование сети Петри.
2. Способы задания сети Петри.
3. Кратность позиций, расширенные входные и выходные функции.
4. Маркированная сеть Петри.
5. Условие разрешенности перехода, правило расчета новой маркировки.

### **Литература**

1. Котов, В. Е. Сети Петри / В. Е. Котов. - М.: Наука, 1984. - 160 с.
2. Питерсон Дж. Теория сетей Петри и моделирование систем / Дж. Питерсон. - М.: Мир, 1984. - 264 с.
3. Бенькович Е.С., Колесов Ю.Б., Сениченков Ю.Б. Практическое моделирование динамических систем – СПб.: БХВ-Петербург, 2002. – 464 с.