

ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ АВТОНОМНОЕ ОБРАЗОВАТЕЛЬНОЕ УЧЕРЕЖДЕНИЕ
ВЫСШЕГО ОБРАЗОВАНИЯ
«Санкт-Петербургский политехнический университет Петра Великого»

ИНСТИТУТ КОМПЬЮТЕРНЫХ НАУК И ТЕХНОЛОГИЙ

Курсовая работа

“Реализация Системы Массового Обслуживания”

Выполнил студент группы: 33504/2:

Лаппо С.С

Руководитель
старший преподаватель

Александрова О. В.

6 декабря 2017 г.

Санкт-Петербург
2017

Содержание

1	Постановка задачи	2
1.1	Принцип Δt	2
1.2	Принцип особых состояний	2
2	Формализованная схема и описание СМО.	3
2.1	Вариант рассматриваемой мной СМО	3
3	Временная диаграмма функционирования системы	4
4	Вывод законов распределения	5
4.1	Пуассоновский закон распределения	5
4.2	Равномерный закон распределения	5
5	Пример технической системы (ВС или части ВС), удовлетворяющей формализованному описанию	6
6	Ограничения и требуемые характеристики:	6
7	Документация на ПО	7
7.1	Блок-схема	7
7.2	Модульная структура	7
8	Результаты работы имитационной модели	10
8.1	Определение количества реализаций	10
8.2	Анализ результатов, выводы и рекомендации по выбору конфигурации системы.	10
9	Выводы	12

1 Постановка задачи

Целью курсовой работы является создание модели вычислительной системы (ВС) или ее части на некотором уровне детализации, описывающей и имитирующей ее структуру и функциональность.

Каждый реальный объект (реальная ВС) обладает бесконечной сложностью, множеством характеристик, внутренних и внешних связей. Модель есть приближенное описание объекта с целью получения требуемых результатов с определенной точностью и достоверностью.

При необходимости исследования поведенческих характеристик ВС в процессе исследования выгодно использовать не сам объект, а его модель. Степень приближения модели к описываемому объекту может быть различной и зависит от требований задачи.

Существуют различные типы моделей:

- 1) Аналитические (математические) модели
- 2) Аналоговые модели
- 3) Физические модели
- 4) Имитационные модели

Последний тип моделей является предметом нашего изучения. Одним из подходов к построению имитационной модели является построение ее в виде системы массового обслуживания (СМО), с характерной для СМО терминологией: источник, буфер, прибор, диспетчер, заявка (требование). Существуют два подхода к построению моделирующего алгоритма:

1.1 Принцип Δt

Принцип Δt Универсальный метод построения моделирующего алгоритма, когда состояние объекта проверяется через фиксированный интервал модельного времени. Суть его заключается в следующем: в каждый момент времени t получают приближенные значения характеристик исследуемого объекта. Δt можно получить детерминированным способом. Основным критерий выбора Δt — он должен быть настолько мал, чтобы не пропустить событие в моделируемой системе, которое должно быть учтено при выбранной детальности моделирования. Метод неэффективен, т.к. постоянно проверяет состояние объектов моделирования, не изменяющихся при этом, особенно при малых Δt .

1.2 Принцип особых состояний

При исследовании реальной системы интервалы, в которых состояние ее не меняется, не представляют интереса. Имеют значение только переходы системы из одного состояния в другое в некоторые моменты времени. Эти переходы определяются особыми состояниями или событиями.

Рассмотрим некоторые типы особых событий, которые изменяют состояние системы:

- 1) Поступление заявки в СМО (момент генерации заявки источником);
- 2) Освобождение прибора (готовность прибора взять заявку на обслуживание);
- 3) Окончание процесса моделирования.

Использование принципа особых событий для построения имитационной модели наиболее эффективно. В настоящей курсовой работе предлагается использовать именно этот принцип.

2 Формализованная схема и описание СМО.

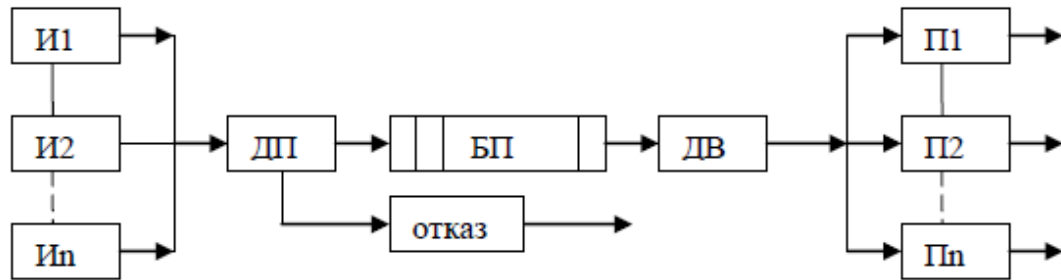


Рис. 1: Формализованная схема СМО

Здесь $И_i$ ($i = 1..n$) – источник заявок, который генерирует заявки, а все вместе n источников создают входной поток заявок в систему. Каждая заявка приходит в СМО со своими характеристиками. Это входное T – время генерации заявки (время поступления её в СМО) и номер заявки составленный из номера источника, сгенерировавшего заявку, и порядкового номера заявки от этого источника. Например, (2.3) – третья заявка от второго источника.

$П$ – приборы, которые обслуживают заявки и создают выходной поток заявок после обслуживания.

$БП$ – буферная память (место для хранения очереди заявок). В общей памяти хранятся заявки от различных источников. Порядок их записи в $БП$ определяется только дисциплиной буферизации.

$ДП$ – диспетчер постановки заявок.

$ДВ$ – диспетчер выбора заявок.

2.1 Вариант рассматриваемой мной СМО

ИБ ИЗ1 ПЗ2 Д10З1 Д10ОЗ Д2П1 Д2БЗ ОР1 ОДЗ

Источники:

ИБ – бесконечный источник;

ИЗ1 – Пуассоновский закон распределения заявок;

Приборы:

ПЗ2 – равномерный закон распределения времени обслуживания;

Описание дисциплин постановки и выбора:

Дисциплина буферизации:

Д10З1 – по кольцу;

Дисциплина отказа:

Д10ОЗ – самая старая заявка в буфере;

Д2П1 – приоритет прибора по номеру;

Д2БЗ – выбор из буфера по кольцу;

Динамическое отражение результатов:

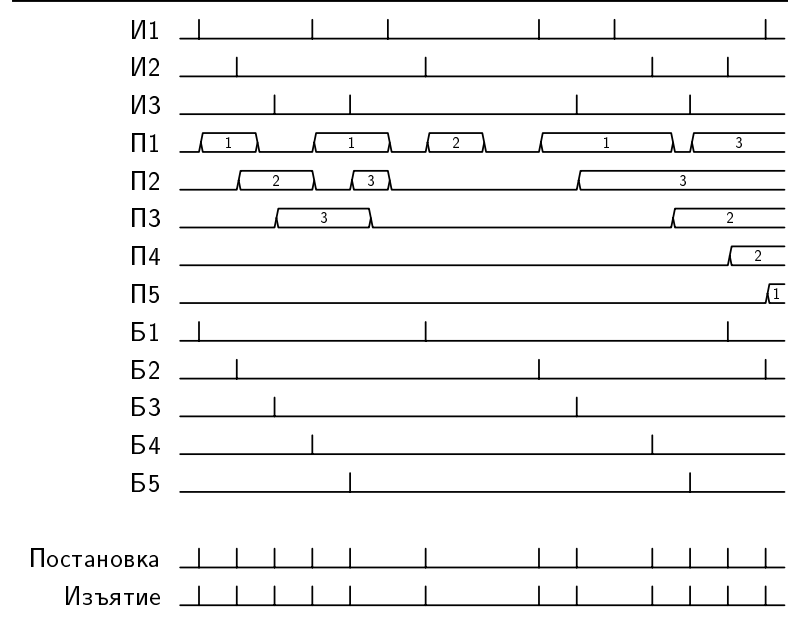
ОДЗ – сводная таблица результатов;

Отражение результатов после сбора статистики:

ОР1 – текущее состояние.

3 Временная диаграмма функционирования системы

Исходя из заданного задания на работу рассмотрим, как должна будет функционировать система массового обслуживания на временной диаграмме. Для примера возьмем 3 источника (И1, И2, И3), 4 позиции в буфере (Б1, Б2, Б3, Б4) и 5 приборов (П1, П2, П3, П4, П5). В результате получим следующую диаграмму:



4 Вывод законов распределения

4.1 Пуассоновский закон распределения

$$F_k = \frac{e^{-\gamma} * \gamma^k}{k!}$$

Где γ - заданное значение.

$$x = \frac{-1}{\gamma} * \ln(F_k)$$

В программе данное выражение записано следующим образом:

```
this -> last_time_of_gen += (-1/lambda)*log((double)qrand()/RAND_MAX);
```

4.2 Равномерный закон распределения

$$F(x) = \begin{cases} 0, & x < a, \\ \frac{x-a}{b-a}, & a \leq x \leq b, \\ 1, & x \geq b. \end{cases}$$

где a и b – заданные значения.

$$x = F(x)(b - a) + a$$

В программе данное выражение записано следующим образом:

```
double rand_time=(double)a+(double)(b-a)*(rand()%100)/100;
```

5 Пример технической системы (ВС или части ВС), удовлетворяющей формализованному описанию

Техническая система	Автомобильный парктроник
Источники	Источниками являются датчики расстояния, которые отсылают данные на обработку в виде пакета размером 64 Кбайт. Необходимо получать и обрабатывать информацию с максимального датчиков.
Приборы	Приборами являются ЭВМ, которые обрабатывают полученную информацию и отправляют результирующий сигнал на дисплей.
Буфер	Буфером является буфер коммутатора, который может быть от 64 Кбайт (1 заявка) и может быть наращен до 256 кб (4 заявки) с шагом 64 Кбайт.
Дисциплина постановки в буфер	Постановка по кольцу
Дисциплина выборки из буфера	Выборка по кольцу. Мы разом получаем всю информацию о необходимых изменениях в расположении автомобиля и в порядке приоритетности отправляем заявки на обработку.
Дисциплина отказа	Самая старая заявка в буфере
Дисциплина постановки на обслуживание	Приоритет по номеру прибора

6 Ограничения и требуемые характеристики:

Вероятность отказа должна составлять не более 10%.

Загрузка приборов более 90%.

Время пребывания заявки в системе не более 10 мс. Рассматриваемый диапазон характеристик системы и доступные типы процессоров и характеристики программного-аппаратного комплекса, построенного на данном типе процессора приведены ниже в таблицах компонентов системы.

Количество датчиков расстояния	От 2 до 6
Вес заявки	64 Кб
Объем буфера	От 64Кб до 256 Кб
Количество приборов	От 1 до 7
Скорость генерации заявок	Пуассоновский поток с $\lambda = 3$ мс.
Скорость обработки заявок	Равномерный поток с границами(мс): 1. [4;5] 2. [5;6] 3. [6;7]

Стоимость компонентов системы:

Считаем, что количество датчиков строго сконфигурировано: 2/3 спереди и сзади автомобиля, поэтому стоимость их не учитывается. Нужно подобрать минимальную конфигурацию по размеру буфера и количеству приборов, удовлетворяющих условиям, для наименьших затрат, связанных с их покупкой. Условимся, что стоимость одного прибора для обработки со временем обслуживания:

- 1) От 3 до 4 мс – 25000 рублей;
- 2) От 5 до 6 мс – 12000 рублей;
- 3) От 6 до 7 мс – 8000 рублей.

Стоимость расширения буфера на 1 слот – 3000 рублей.

7 Документация на ПО

7.1 Блок-схема

Приведем для понимания работы программы обобщенную блок-схему:

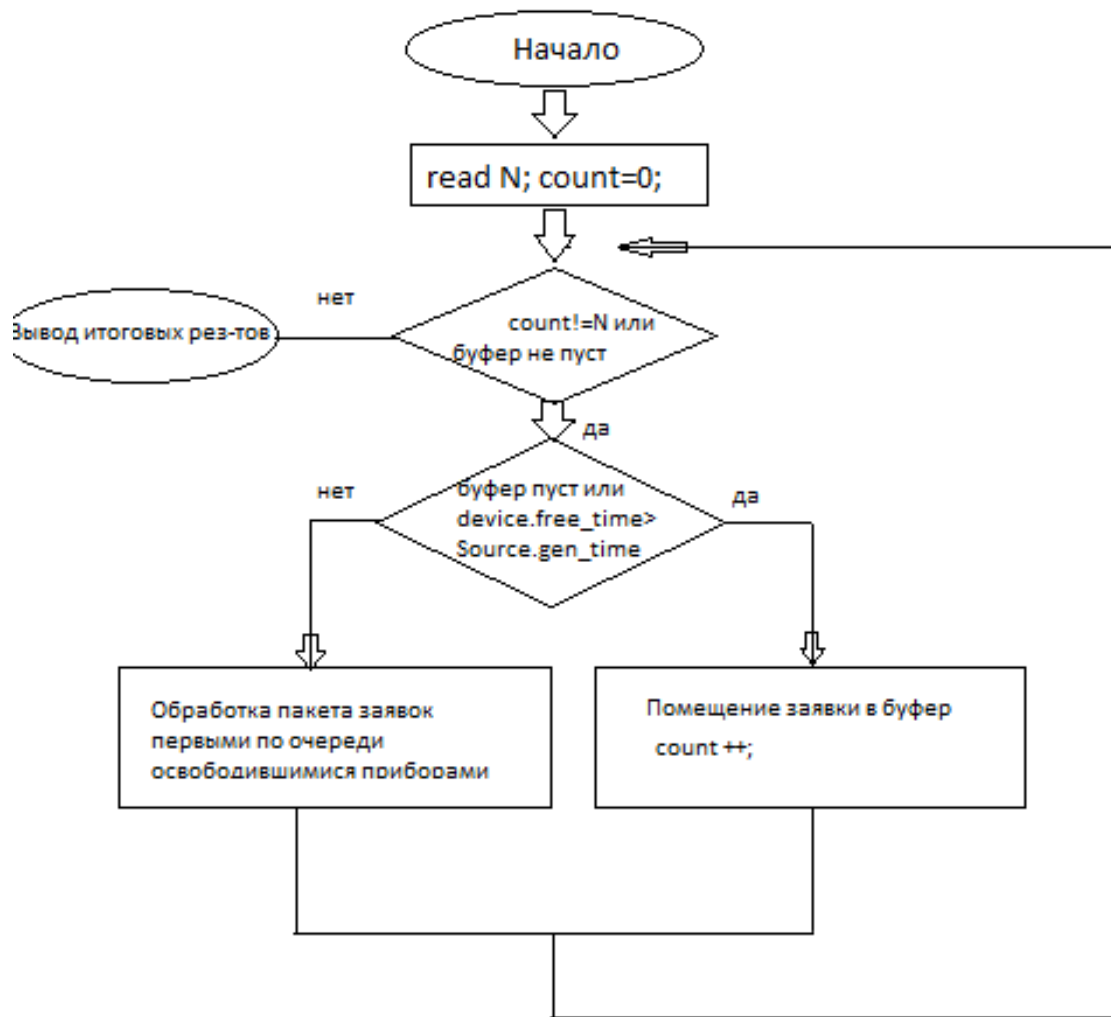


Рис. 2: Описание работы программы

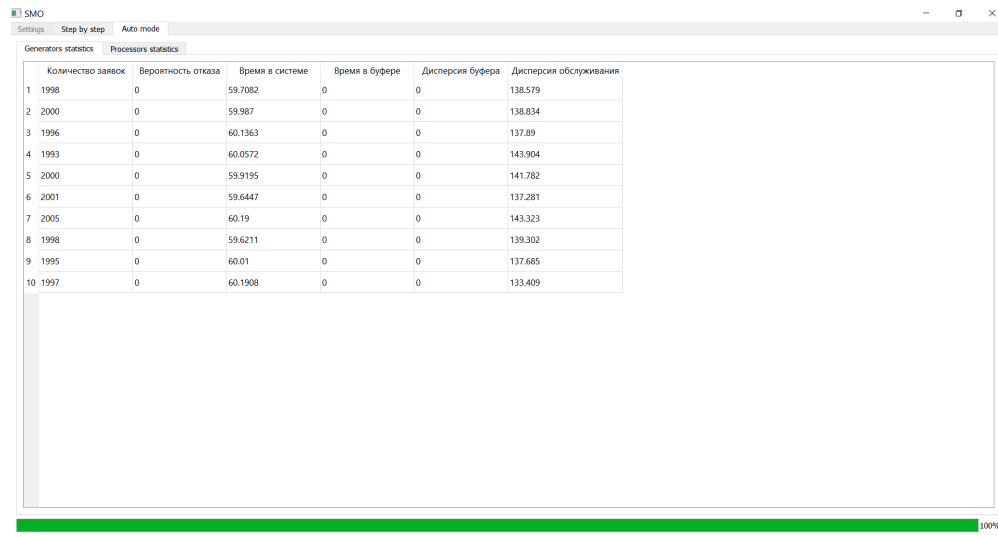
7.2 Модульная структура

Разработка производилась в среде Qt creator на языке C++ с использованием графической библиотеки Qt. Приложение использует объектно-ориентированную парадигму программирования и содержит набор классов:

- 1) BufferQueue - класс буфера;
- 2) mainwindow - класс ui, создающий весь интерфейс;
- 3) OperatingDevice - класс прибора;

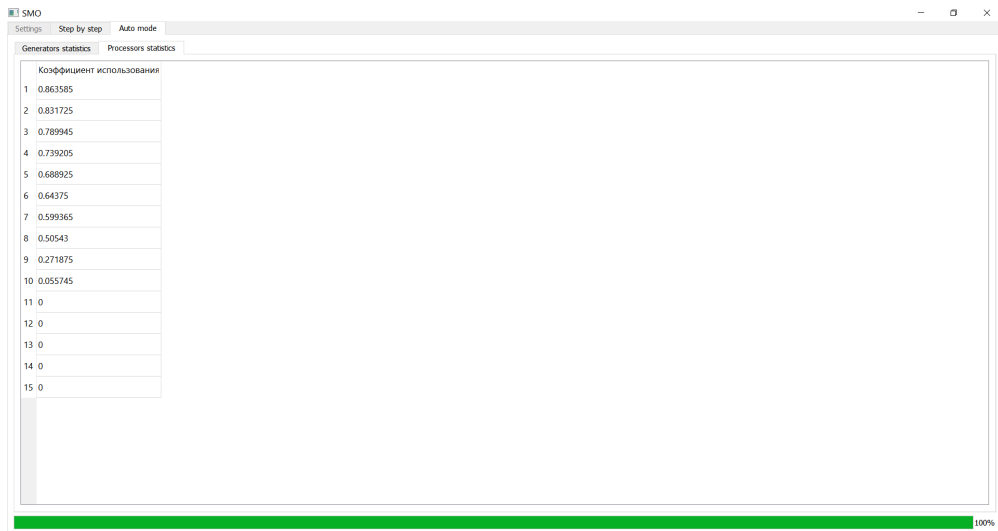
- 4) SourceDevice - класс источника;
- 5) StatisticsController - класс, отвечающий за обработку статистики;
- 6) SystemController - класс, который занимается управлением системой;
- 7) SystemEvent - класс, характеризующий системные события;
- 8) SystemTask - класс, характеризующий задачу.

Программа содержит точку входа в файле main.cpp. Основное действие процедуры main – создание объекта окна программы и его отображение. После появления окна программы, запускается цикл обработки событий (действий пользователя). Отображение результатов в автоматическом режиме:



	Количество заявок	Вероятность отказа	Время в системе	Время в буфере	Дисперсия буфера	Дисперсия обслуживания
1	1998	0	59.7082	0	0	138.579
2	2000	0	59.987	0	0	138.834
3	1996	0	60.1363	0	0	137.89
4	1993	0	60.0572	0	0	143.904
5	2000	0	59.9195	0	0	141.782
6	2001	0	59.6447	0	0	137.281
7	2005	0	60.19	0	0	143.323
8	1998	0	59.6211	0	0	139.302
9	1995	0	60.01	0	0	137.685
10	1997	0	60.1908	0	0	133.409

Рис. 3: Отображение статистики источников



	Кoeffициент использования
1	0.863585
2	0.831725
3	0.789945
4	0.739205
5	0.688925
6	0.64375
7	0.599365
8	0.50543
9	0.271875
10	0.055745
11	0
12	0
13	0
14	0
15	0

Рис. 4: Отображение статистики приборов

Отображение результатов в пошаговом режиме:

Settings

Step by step

Auto mode

Buffer state

№ в буфере	№ источника	№ заявки
0	0	0.4
1	2	2.8
2	2	2.7
3	3	3.2

System state

	Время	Действие	Назначенное устройство
1	13	Переместить задачу ...	-1
2	13	Переместить задачу ...	-1
3	13	Переместить задачу ...	-1
4	13	Создать задачу	3
5	13	Создать задачу	0
6	14	Окончить задачу	1
7	14	Окончить задачу	3
8	14	Создать задачу	1
9	14	Создать задачу	2
10	15	Окончить задачу	5
11	17	Окончить задачу	6

Processing devices state

№	Состояние	Время начала	№ заявки
0	Свободен	13	Нет заявки
2	Свободен	13	Нет заявки
4	Свободен	13	Нет заявки
1	Занят	8	1.3
3	Занят	9	0.3
5	Занят	10	2.6
6	Занят	11	1.4

Next step

0%

Рис. 5: Пошаговый режим

8 Результаты работы имитационной модели

8.1 Определение количества реализаций

Количество реализаций, необходимое для получения нужной точности при заданной доверительной вероятности, можно оценивать по формуле:

$$N = \frac{t_{\alpha}^2(1-p)}{p\delta^2}$$

где p — вероятность отказа заявкам в обслуживании,

$t_{\alpha} = 1.643$, для $\alpha = 0.9$,

$\sigma = 0.1$ - относительная точность. По результатам работы программы получено, что в большинстве случаев для достижения заданной точности необходимо от 3000 до 5000 заявок. Однако, в случаях, когда p мало (<0.05) для достижения точности в 10% может потребоваться существенно больше заявок (20000-30000).

8.2 Анализ результатов, выводы и рекомендации по выбору конфигурации системы.

Рассмотрим первый возможный вариант конфигурации, когда имеется по два источника спереди и сзади. Т.к. целью моделирования является выбор конфигурации системы, требующей наименьшее количество ресурсов и обрабатывающей максимальный поток информации, то начнем с проверки конфигурации с минимальным числом приборов и мин. размером буфера.

Количество источников во всех опытах возьмем равным 4.

Число источников	Число приборов	a;b	lambda	Размер буфера	Вероятность отказа	T преб	Коэф. польз.
4	1	6;7	3	1	0,87	1,5	1

В первом опыте прибор эффективно используется, время нахождения в системе допустимо, но 87% заявок уходят в отказ, что является недопустимым, поэтому расширяем буфер.

Число источников	Число приборов	a;b	lambda	Размер буфера	Вероятность отказа	T преб	Коэф. польз.
4	1	6;7	3	4	0,87	2,96	0,999

Вероятность отказа не снизилась, и до сих пор не является удовлетворительной. Время нахождения в системе допустимо. Возьмем для обработки 3 самых дешевых прибора.

Число источников	Число приборов	a;b	lambda	Размер буфера	Вероятность отказа	T преб	Коэф. польз.
4	3	6;7	3	4	0,66	2,87	0,999

Увеличим количество приборов до 7(максимум) самых дешевых

Число источников	Число приборов	a;b	lambda	Размер буфера	Вероятность отказа	T преб	Коэф. польз.
4	7	6;7	3	4	0,2	7,42	0,995

Время нахождения в системе допустимо. Вероятность отказа упала, но все еще достаточно велика, поэтому возьмем приборы со средними характеристиками

Число источников	Число приборов	a;b	lambda	Размер буфера	Вероятность отказа	T преб	Коэф. польз.
4	7	5;6	3	4	0,06	6,54	0,94

Время нахождения в системе допустимо. Вероятность отказа снизилась до 6%, что нас полностью устраивает. При уменьшении количества приборов (и даже при улучшении их качества) система либо не будет удовлетворять требованиям, либо обойдется нам дороже.

Стоимость системы (без учета цен на датчики): $12000*7+3000*4=96000$ рублей.

Теперь рассмотрим систему, когда количество датчиков на торцах автомобиля увеличено до 3.

Попробуем вначале использовать аналогичную предыдущему случаю конфигурацию:

Число источников	Число приборов	a;b	lambda	Размер буфера	Вероятность отказа	T преб	Коэф. польз.
6	7	5;6	3	4	0,36	5,19	0,99

Время обработки недопустимо. Возьмём самые лучшие приборы.

Число источников	Число приборов	a;b	lambda	Размер буфера	Вероятность отказа	T преб	Коэф. использ.
6	7	3;4	3	4	0,04	4.11	0,88

При данной конфигурации количество отказов минимально, также значительно сократилось время нахождения в системе. Попробуем уменьшить количество приборов.

Число источников	Число приборов	a;b	lambda	Размер буфера	Вероятность отказа	T преб	Коэф. использ.
6	6	3;4	3	4	0,14	4.37	0,98

Время нахождения в системе осталось в допустимых пределах, загрузка приборов более оптимальна, однако мы незначительно превысили порог отказов, так что этот вариант нам недоступен.

Попробуем альтернативное удешевление системы - уменьшим количество мест в буфере.

Число источников	Число приборов	a;b	lambda	Размер буфера	Вероятность отказа	T преб	Коэф. использ.
6	7	3;4	3	2	0,07	3.56	0,90

Дальнейшее уменьшение буфера приведёт к превышению допустимого времени обработки, а данная конфигурация нас полностью устраивает.

Стоимость системы (без учета цен на датчики): $25000 \cdot 7 + 3000 \cdot 2 = 181000$ рублей.

9 Выводы

В ходе выполнения курсовой работы мной была написана система массового обслуживания на языке C++ в среде разработки Qt Creator с использованием библиотеки Qt. С помощью получившейся программы была проанализирована реальная система и подобрана максимально выгодная комплектация данной системы.