

Департамент внутренней и кадровой политики Белгородской области  
Областное государственное автономное  
профессиональное образовательное учреждение  
«Белгородский индустриальный колледж»

Рассмотрено  
цикловой комиссией  
Протокол заседания № 1  
от « 31 » августа 2020 г.  
Председатель цикловой комиссии  
\_\_\_\_\_ / Чобану Л.А./

## **МЕТОДИЧЕСКИЕ УКАЗАНИЯ**

по выполнению практических работ  
учебной дисциплине

### **ЕН.02 Компьютерное моделирование**

по специальности

#### **11.02.10 Радиосвязь, радиовещание и телевидение (углубленной подготовки)**

квалификация

**специалист по телекоммуникациям**

Разработчик:  
преподаватель  
ОГАПОУ «Белгородский  
индустриальный колледж»  
Феоктистова В.Н.

Белгород 2020 г.

## Содержание

	Стр.
1. Пояснительная записка	3
1.1. Краткая характеристика дисциплины, ее цели и задачи. Место практических работ в курсе дисциплины	3
1.2. Организация и порядок проведения практических работ	3
1.3. Общие указания по выполнению практических работ	3
1.4. Критерии оценки результатов выполнения практических работ	3
2. Тематическое планирование практических работ	5
3. Содержание практических работ	6
Практическая работа №1 Разработка модели непроизводственной системы	6
Практическая работа № 2 Разработка модели производственной системы	10
Практическая работа № 3 Разработка модели СМО	18
Практическая работа № 4 Синтез схем на основе логических элементов»	24
Практическая работа № 5 Синтез комбинационных схем	28
Практическая работа №6 Синтез последовательностных устройств»	33
4. Информационное обеспечение обучения	37

## 1. Пояснительная записка

### 1.1. Краткая характеристика дисциплины ЕН.02 Компьютерное моделирование, ее цели и задачи. Место практических работ в курсе дисциплины ЕН.02 Компьютерное моделирование

Дисциплина ЕН.02 «Компьютерное моделирование» является частью рабочей основной образовательной программы в соответствии с ФГОС по специальности СПО 11.02.10 Радиосвязь, радиовещание и телевидение (углубленной подготовки).

Дисциплина изучается в III семестре. В целом рабочей программой предусмотрено 12 часов на выполнение практических работ, что составляет 25 % от обязательной аудиторной нагрузки, которая составляет 48 часов, при этом максимальная нагрузка составляет 72 часа, из них 20 часов приходится на самостоятельную работу обучающихся.

Цель настоящих методических рекомендаций: оказание помощи обучающимся в выполнении практических работ по дисциплине ЕН.02 «Компьютерное моделирование», качественное выполнение которых поможет обучающимся освоить обязательный минимум содержания дисциплины и подготовиться к промежуточной аттестации в форме дифференцированного зачета.

### 1.2. Организация и порядок проведения практических работ

Практические работы проводятся после изучения теоретического материала. Введение практических работ в учебный процесс служит связующим звеном между теорией и практикой. Они необходимы для закрепления теоретических знаний, а также для получения практических навыков и умений. При проведении практических работ задания, выполняются студентом самостоятельно, с применением знаний и умений, усвоенных на предыдущих занятиях, а также с использованием необходимых пояснений, полученных от преподавателя. Обучающиеся должны иметь методические рекомендации по выполнению практических работ, конспекты лекций, измерительные и чертежные инструменты, средство для вычислений.

### 1.3. Общие указания по выполнению практических работ

Курс практических работ по дисциплине ЕН.02 «Компьютерное моделирование» предусматривает проведение 6 работ, посвященных изучению:

- модели непроизводственной системы;
- модели производственной системы;
- модели СМО;
- синтез схем на основе логических элементов;
- синтез комбинационных схем;
- Синтез последовательностных устройств.

При подготовке к проведению практической работы необходимо:

- ознакомиться с целями проведения практической работы;
- ознакомиться с порядком выполнения работы.

После выполнения практической работы обучающийся к следующему занятию оформляет отчет, который должен содержать:

- название практической работы, ее цель;
- краткие, теоретические сведения об изучаемой теме;
- все необходимые, предусмотренные практической работой, расчеты;
- выводы по итогам работы;
- ответы на контрольные вопросы.

### 1.4. Критерии оценки результатов выполнения практических работ

Критериями оценки результатов работы обучающихся являются:

- уровень усвоения обучающимся учебного материала;
- умение обучающегося использовать теоретические знания при выполнении практических задач;

– сформированность общих и профессиональных компетенций:

ОК 1. Понимать сущность и социальную значимость своей будущей профессии, проявлять к ней устойчивый интерес.

ОК 2. Организовывать собственную деятельность, определять методы и способы выполнения профессиональных задач, оценивать их эффективность и качество.

ОК 3. Решать проблемы, оценивать риски и принимать решения в нестандартных ситуациях.

ОК 4. Осуществлять поиск, анализ и оценку информации, необходимой для постановки и решения профессиональных задач, профессионального и личностного развития.

ОК 5. Использовать информационно-коммуникационные технологии для совершенствования профессиональной деятельности.

ОК 6. Работать в коллективе и команде, обеспечивать ее сплочение, эффективно общаться с коллегами, руководством, потребителями.

ОК 7. Ставить цели, мотивировать деятельность подчиненных, организовывать и контролировать их работу с принятием на себя ответственности за результат выполнения заданий.

ОК 8. Самостоятельно определять задачи профессионального и личностного развития, заниматься самообразованием, осознанно планировать повышение квалификации.

ОК 9. Быть готовым к смене технологий в профессиональной деятельности.

– ПК 1.1 Выполнять монтаж и первичную инсталляцию оборудования систем радиосвязи и вещания.

– ПК 1.2 Выполнять монтаж и производить настройку сетей абонентского доступа на базе систем радиосвязи и вещания.

– оборудования радиосвязи и вещания.

– ПК 2.1 Выполнять монтаж и первичную инсталляцию компьютерных сетей.

– ПК 2.2 Инсталлировать и настраивать компьютерные платформы для организации услуг связи.

– ПК 2.3 Производить администрирование сетевого оборудования.

– ПК 2.4 Выполнять монтаж и производить настройку сетей проводного и беспроводного абонентского доступа.

– ПК 2.5 Работать с сетевыми протоколами.

– ПК 2.6 Обеспечивать работоспособность оборудования мультисервисных сетей.

– ПК 3.1 Использовать программно-аппаратные средства защиты информации в системах радиосвязи и вещания.

– ПК 3.2 Применять системы анализа защищенности для обнаружения уязвимостей в сетевой инфраструктуре, давать рекомендации по их устранению.

– ПК 3.3 Обеспечивать безопасное администрирование сетей вещания.

– обоснованность и четкость изложения материала;

– уровень оформления работы.

– анализ результатов.

#### Критерии оценивания практической работы

Оценка	Критерии оценивания
5	Работа выполнена в полном объеме с соблюдением необходимой последовательности проведения, содержит результаты и выводы, все записи, таблицы, рисунки, чертежи, графики выполнены аккуратно. Обучающийся владеет теоретическим материалом, формулирует собственные, самостоятельные, обоснованные, представляет полные и развернутые ответы на дополнительные вопросы.
4	Работа выполнена в полном объеме с соблюдением необходимой последовательности проведения, содержит результаты и выводы, все записи, таблицы, рисунки, чертежи, графики выполнены аккуратно. Обучающийся владеет теоретическим материалом, допуская незначительные ошибки на дополнительные вопросы.
3	Работа выполнена в полном объеме, содержит результаты и выводы, все записи, таблицы, рисунки, чертежи, графики выполнены аккуратно. Обучающийся владеет теоретическим материалом на минимально допустимом уровне, допуская ошибки на дополнительные вопросы.
2	Работа выполнена не полностью. Студент практически не владеет теоретическим материалом, допускает ошибки при ответе на дополнительные вопросы.

## 2. Тематическое планирование практических работ

	Наименование тем	Вид и название работы студента	Количество часов на выполнение работы
<b>Раздел 2</b>	<b>Моделирование систем массового обслуживания</b>		<b>6</b>
2.2.		<b>Практическая работа №1</b> «Разработка модели непроизводственной системы»	2
		<b>Практическая работа №2</b> «Разработка модели производственной системы»	2
		<b>Практическая работа №3</b> «Разработка модели СМО»	2
<b>Раздел 3</b>	<b>Моделирование работы цифровых устройств</b>		<b>6</b>
		<b>Практическая работа №4</b> «Синтез схем на основе логических элементов»	2
		<b>Практическая работа №5</b> «Синтез комбинационных устройств»	2
		<b>Практическая работа №6</b> «Синтез последовательностных устройств»	2
		<b>Итого:</b>	<b>12</b>

### 3. Содержание практических работ

#### Практическая работа № 1

**Тема:** «Разработка модели непроизводственной системы»

**Цель работы:** «Научиться разрабатывать алгоритмы и программы для решения практических задач»

**Задача:**

Промоделировать работу небольшого магазина, который имеет один кассовый аппарат и одного продавца.

Интервал времени прибытия покупателей ( $t_1$ ) колеблется в пределах от 8,7 до 10,3 мин включительно.

Время пребывания покупателей у кассового аппарата ( $t_2$ ) составляет от 1,6 до 3 мин. После этого покупатели подходят к продавцу для получения товара.

Время, потраченное на обслуживание покупателей продавцом ( $t_3$ ), составляет от 8,6 до 11,4 мин.

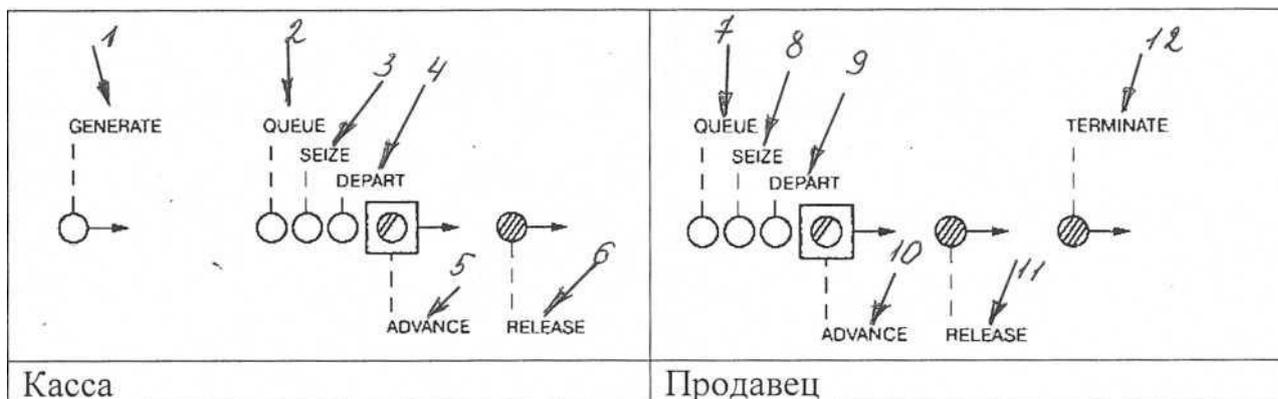
Для моделирования работы магазина необходимо сформировать входной поток покупателей (требований) и временной интервал моделирования работы магазина. В качестве единицы измерения выбрать минуту.

Для задачи Вашего варианта выполнить задание.

1. Составить структурную схему модели исследуемой системы. Пояснить назначение элементов.

2. Записать программу в системе GPSS World.

**Структурная схема имитационной модели**



1. Создание имитационной модели начнем с построения заголовка модели, который может быть представлен, например, в таком виде:

```
; GPSSW File      MAGAZIN.GPS
*****
*   Моделирование работы магазина   *
*****
```

2. В программе в самом начале добавлен оператор **QTABLE** с меткой **t\_prod** для сбора информации и построения соответствующей гистограммы функционирования очереди под именем **Ocher\_prod**.

```
t_prod QTABLE Ocher_prod, 0, 2, 32
```

3. Моделирование потока покупателей будем выполнять с помощью оператора **GENERATE** (Генерировать). В нашем примере он будет выглядеть так:

```
GENERATE 9.5, 0.8
```

В поле операнда **A** указывается средний интервал времени между прибытием в магазин двух идущих один за другим покупателей (требований,

транзактов). В нашем примере он составляет 9,5 мин.

В поле операнда В дано отклонение времени прихода покупателей от среднего. В нашем примере это отклонение составляет 0,8 мин.

4. Покупатель, пришедший в магазин, сначала встает в очередь к кассиру, если она есть. Это можно промоделировать оператором QUEUE (Очередь), который только в совокупности с соответствующим оператором DEPART (Выйти) собирает статистическую информацию о работе моделируемой очереди.

В нашем примере оператор QUEUE будет выглядеть так:

```
QUEUE Ocher_kassa
```

В поле операнда А дается символьное или числовое имя очереди. В нашей задаче дадим очереди имя Ocher\_kassa (Очередь в кассу).

5. Следуя логике, покупатель может выйти из очереди только тогда, когда освободится кассир (канал обслуживания). Для этого вводится оператор SEIZE, который определяет занятость канала обслуживания, и при его освобождении очередное требование выходит из очереди и идет в канал на обслуживание. Это может выглядеть так:

```
SEIZE Kassir
```

В поле операнда А дается символьное или числовое имя канала обслуживания. В нашей задаче каналу дано имя Kassir (Кассир).

6. Выход покупателя из очереди в кассу фиксируется оператором DEPART с соответствующим названием очереди:

```
DEPART Ocher_kassa
```

7. Далее должно быть промоделировано время пребывания покупателя, непосредственно обслуживаемого кассиром. Это время в нашем примере составляет  $2,3 \pm 0,7$  мин. Для моделирования этого процесса используется оператор ADVANCE (Задержать):

```
ADVANCE 2.3, 0.7
```

8. После обслуживания кассиром покупатель отправляется к продавцу за получением оплаченного товара. Однако перед этим системе должно быть послано сообщение об освобождении канала обслуживания. Это делается с помощью оператора RELEASE, который в нашей задаче записывается так:

```
RELEASE Kassir
```

Следует особо подчеркнуть, что парные операторы QUEUE и DEPART для каждой очереди должны иметь одно и то же, но свое уникальное имя. Это же относится и к операторам SEIZE и RELEASE.

9. После обслуживания в кассе покупатель направляется к продавцу – следующему каналу обслуживания. Процесс моделирования этой цепи аналогичен только что описанному. И в нашем примере он может быть представлен, например, в таком виде:

```
QUEUE Ocher_prod  
SEIZE Prodavec  
DEPART Ocher_prod  
ADVANCE 10, 1.4  
RELEASE Prodavec
```

10. После обслуживания продавцом (каналом обслуживания) покупатель (требование) покидает систему. Это действие может быть представлено оператором TERMINATE (Завершить). В поле операнда А стоит число 1. Это означает, что систему обслуживания (магазин) покупатели покидают по одному.

TERMINATE 1

11. Завершающим оператором в нашей задаче является управляющая команда START (Начать), позволяющая начать моделирование START 100.

В поле операнда A стоит число 100, показывающее, с каким числом покупателей будет моделироваться система работы магазина.

**Полученная программа имеет вид:**

```
t_prod QTABLE Ocher_prod,0,2,32
GENERATE 9.5,0.8
QUEUE Ocher_kassa
SEIZE Kassir
DEPART Ocher_kassa
ADVANCE 2.3,0.7
RELEASE Kassir
QUEUE Ocher_prod
SEIZE Prodavec
DEPART Ocher_prod
ADVANCE 10,1.4
RELEASE Prodavec
TERMINATE 1
START 100.
```

### *Представление имитационной модели*

Для представления имитационной модели выполните следующие действия:

- щелкните по пункту **File** главного меню системы. Появится выпадающее меню;
- щелкните по пункту **New** выпадающего меню. Появится диалоговое окно **Новый документ**;
- выделите пункт **Model** и щелкните по кнопке ОК. Появится окно модели, в котором введите данную программу. Это будет выглядеть так, как показано на рис. 2.

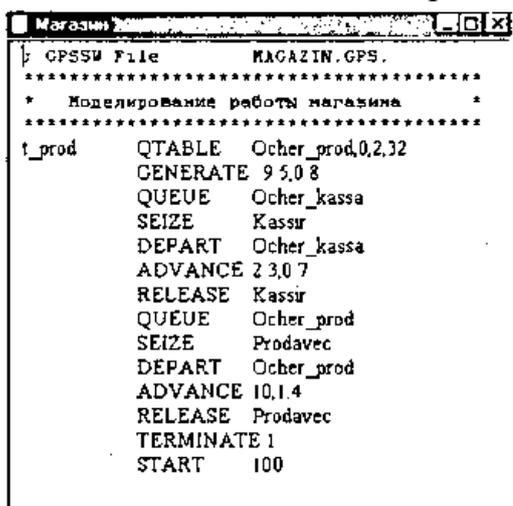


Рисунок 2. Окно имитационной модели «Магазин»

Чтобы вызвать окно для представления имитационной модели в системе GPSSW, можно также нажать комбинацию клавиш Ctrl+Alt+S.

### *Варианты заданий*

Вариант	<i>t1, мин</i>	<i>t2, мин</i>	<i>t3, мин</i>	Вариант	<i>t1, мин</i>	<i>t2, мин</i>	<i>t3, мин</i>

1	10-12	2,4-3,6	2,4-3,2	14	14-18,2	17,2-18	10,2-11
2	7-9,6	5-6,2	6,4-8	15	12,8-14	12,6-14	9,4-9,8
3	8-12,4	6,2-7	5,2-6,4	16	10,2-12	11-12,4	12,2-13
4	5-11	3,4-4,2	4,2-5	17	8,6-10,4	6,4-7,2	5,4-6,6
5	3-12,8	8,2-9,4	4,8-5,2	18	13-15,2	8,4-8,8	10,4-12
6	14-18,2	3,6-4,2	5,2-6	19	10-12	5,2-6,4	7,2-8,4
7	12,8-14	1,8-2,4	5,4-6,2	20	7-9,6	1,8-2,2	4,2-5
8	10,2-12	2,4-3,2	7,2-8	21	8-12,4	3,2-4	4,8-5,2
9	8,6-10,4	6,4-8	7,4-8,2	22	5-11	4,2-5	5,2-6
10	13-15,2	5,2-6,4	8,2-9	23	3-12,8	3,4-4,2	5,4-6,2
11	17,2-18	1,8-2,2	10,2-11	24	2,4-3,6	8,2-9,4	7,2-8
12	12,6-14	3,2-4	9,4-9,8	25	5-6,2	3,6-4,2	7,4-8,2
13	11-12,4	4,2-5	12,2-13	26	6,2-7	1,8-2,4	8,2-9

### Подготовка к моделированию системы

Перед началом моделирования можно установить вывод тех параметров моделирования, которые нужны пользователю. Для этого:

- щелкните по пункту **Edit** (Правка) главного меню системы или нажмите комбинацию клавиш Alt+E. Появится выпадающее меню;
- щелкните по пункту **Settings** (Установки) выпадающего меню. Появится диалоговое окно **SETTINGS** для данной модели, в котором можно установить нужные выходные данные, которые отмечаются флажком (галочкой). Для нашего примера это может выглядеть так, как представлено на рис. 2.

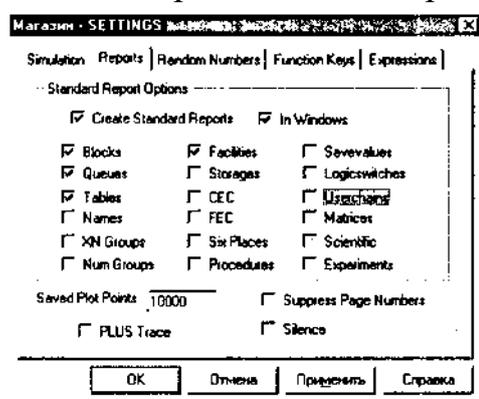


Рисунок 2. Окно SETTINGS с установками для имитационной модели магазина

Наличие галочки в окошках говорит о том, что эта информация будет выведена в окне результатов моделирования. В нашем примере будет выведена информация для следующих объектов:

- Blocks (Блоки);
- Queues (Очереди);
- Tables (Таблицы/гистограммы);
- Facilities (Каналы обслуживания).

### Контрольные вопросы

1. Понятие имитационного моделирования
2. Дайте определение имитационной модели
3. Какими наборами характеризуется имитационная модель
4. Понятие реального времени в процессе имитации
5. Понятие модельного времени в процессе имитации

6. Понятие машинного времени в процессе имитации
7. Пояснить построение моделирующего алгоритма по принципу  $\Delta t$
8. Пояснить построение моделирующего алгоритма по принципу особых состояний
9. Пояснить сущность агентного моделирования, Монте-Карло симуляции
10. Пояснить сущность дискретно-событийного моделирования, системную динамику
11. Пояснить области применения имитационного моделирования
12. Дайте определение СМО
13. Охарактеризуйте входящий поток требований
14. Охарактеризуйте выходящий поток требований
15. Дайте определение понятию требование (транзакт)
16. Дайте определение понятию канал обслуживания
17. Дайте определение понятию накопитель (буфер)
18. Дайте определение понятию имитационная модель СМО
19. Перечислите основные компоненты системы массового обслуживания
20. Охарактеризовать компоненты структурной схемы СМО – источник заявок, блок очереди.
21. Охарактеризовать компонент структурной схемы СМО – прибор обслуживания
22. Охарактеризуйте СМО с отказами, СМО с очередью
23. Охарактеризуйте СМО с многофазовым обслуживанием, открытые и замкнутые СМО, одноканальные и многоканальные СМО
24. Приведите общие характеристики СМО
25. Приведите основные характеристики потока заявок, каналов обслуживания.

Вариант	Вопрос № 1	Вопрос № 2	Вопрос № 3	Вариант	Вопрос № 1	Вопрос № 2	Вопрос № 3
<b>1</b>	1	9	18	<b>14</b>	14	22	6
<b>2</b>	2	10	19	<b>15</b>	15	23	7
<b>3</b>	3	11	20	<b>16</b>	16	24	8
<b>4</b>	4	12	21	<b>17</b>	17	25	9
<b>5</b>	5	13	22	<b>18</b>	18	1	10
<b>6</b>	6	14	23	<b>19</b>	19	2	11
<b>7</b>	7	15	24	<b>20</b>	20	3	12
<b>8</b>	8	16	25	<b>21</b>	21	4	13
<b>9</b>	9	17	1	<b>22</b>	22	5	14
<b>10</b>	10	18	2	<b>23</b>	23	6	15
<b>11</b>	11	19	3	<b>24</b>	24	7	16
<b>12</b>	12	20	4	<b>25</b>	25	8	17
<b>13</b>	13	21	5	<b>26</b>	1	10	20

## Практическая работа № 2

Тема «Разработка модели производственной системы»

**Цель работы:** «Научиться разрабатывать алгоритмы и программы для решения практических задач»

### Задача

К рабочим поступают на изготовление детали с транспортного конвейера.

Интервал между поступлениями двух идущих одна за другой деталей равен  $9 \pm 1$  мин.

Время изготовления детали первым рабочим составляет  $12 + 1$  мин, а вторым –  $13 \pm 2$  мин. Если рабочий занят, он не берет деталь с конвейера, и она перемещается к другому рабочему. Требуется смоделировать работу первого и второго рабочих в течение смены. Необходимо определить коэффициент использования первого и второго рабочих (первого и второго каналов обслуживания) и число деталей, изготовленных каждым из них.

Для задачи Вашего варианта выполнить задание.

1. Составить структурную схему модели исследуемой системы. Пояснить назначение элементов.

2. Записать программу в системе GPSS World.

*Выявление основных особенностей.*

Особенности моделирования данной системы заключаются в следующем:

– первая деталь (требование) поступает на изготовление (обслуживание) через 15 мин после начала смены;

– детали не накапливаются у первого рабочего. Если он занят, то детали поступают на обработку ко второму рабочему;

– время моделирования - рабочая смена - 480 мин.

Для реализации первой особенности используем оператор GENERATE (Генерировать):

GENERATE 5,,15

В поле операнда А указан интервал времени, через который детали поступают в систему, - 5 мин, а в поле операнда С - время поступления первой детали на изготовление - 15 мин.

Для реализации второй особенности используем оператор TRANSFER (Передать):

TRANSFER BOTH, RABO

В поле операнда А указан режим работы оператора - BOTH (Оба), а в поле операнда С - имя метки оператора, к которому переходит деталь в случае занятости первого рабочего. Детали могут образовывать очередь в связи с занятостью второго рабочего. При этом детали обрабатываются в порядке их поступления к рабочему. В поле операнда В имя оператора не указано (поле пусто). Это означает, что деталь может перейти к следующему по порядку оператору модели, если этот оператор не занят, в противном случае - к оператору с именем RABO.

Изобразим графически процесс функционирования системы «Транспортный конвейер-рабочий» (рис. 1).

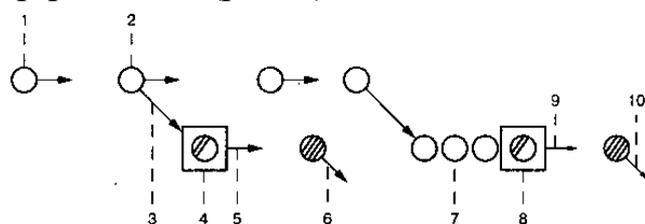


Рисунок 1. Функционирование системы «Транспортный конвейер-рабочий»

На рис. 1 представлены основные события, которые возникают в процессе работы системы «Транспортный конвейер-рабочий».

Кратко охарактеризуем каждое событие в моделируемой системе:

1. Поступление деталей (требований) на обработку (GENERATE).
2. Передача детали первому или второму рабочему (TRANSFER).
3. Передача детали первому рабочему, если он свободен (SEIZE).
4. Время выполнения операции первым рабочим (ADVANCE).
5. Освобождение первого рабочего, то есть канала обслуживания (RELEASE).
6. Завершение выполнения операции - деталь покидает систему (TERMINATE).
7. Передача детали второму рабочему, если он свободен, а первый рабочий занят (SEIZE).
8. Выполнение операции вторым рабочим (ADVANCE).
9. Освобождение второго рабочего (RELEASE).
10. Завершение выполнения операции - деталь покидает систему (TERMINATE).

#### *Создание имитационной модели процесса*

Построение имитационной модели начнем с создания заголовка модели, который может быть представлен, например, в таком виде:

```
GPSSW                               Transp_konv.GPS
*****
*   Моделирование работы системы   *
*   "Транспортный конвейер-рабочий" *
*****
```

Далее разобьем всю модель на несколько секторов.

В первом секторе введем информацию, необходимую для функционирования системы. Этот сектор может выглядеть так:

```
INITIAL  X1,9
INITIAL  X2,1
INITIAL  X3,12
INITIAL  X4,1
INITIAL  X5,13
INITIAL  X6,2
```

В этом секторе с помощью оператора INITIAL (Инициализировать) задаются значения сохраняемых величин.

Во втором секторе моделируется процесс поступления деталей на обработку, передача деталей не занятому рабочему и обработка детали первым рабочим. Эта часть модели может быть сформирована следующим образом.

С помощью оператора GENERATE генерируется поток деталей, поступающих на обработку. Среднее время поступления деталей определяется сохраняемой величиной под номером 1, значение которой должно быть определено заранее. Отклонение от среднего времени определяется сохраняемой величиной под номером 2, значение которой также должно быть определено заранее. Это можно записать так:

```
GENERATE X1, X2
```

В поле операнда A указывается средний интервал времени между поступлениями требований в систему – X1. В задаче эта величина равна 9. Поле

операнда В показывает величину отклонения от среднего интервала времени, равную  $\pm 1$  единице времени. Таким образом, интервал времени между поступлениями деталей в систему может составлять [8 – 10] единиц времени включительно. Определение конкретной величины интервала времени выполняется следующим образом. Встроенная подпрограмма случайных чисел (датчик случайных чисел) в интервале [0-1], используя равномерное распределение, определяет случайное число, например 0,25, затем система моделирования вычисляет искомую величину, используя формулу

$$X = A + (B - A) * Y,$$

где А и В – соответственно левая и правая границы интервала поступления требований в систему.

Искомое случайное число в нашей задаче будет следующим:

$$X = 8 + (10 - 8) * 0,25 = 8,5.$$

Передача деталей не занятому рабочему может быть осуществлена с использованием оператора TRANSFER в режиме BOTH (Оба). Если установлен режим передачи требований в один из двух операторов (BOTH), то требованию никогда не запрещается входить в оператор TRANSFER. Если оператор, указанный в поле В, занят, то требование переходит к оператору, указанному в поле С; если и этот оператор занят, то требование остается в операторе TRANSFER и в цепи текущих событий до освобождения занятого оператора в поле С. Это можно записать так:

TRANSFER BOTH, RABO

Если первый рабочий занят, то деталь поступает на обработку ко второму рабочему - оператору с меткой RABO - при условии, что он не занят. Если и он занят, то деталь будет ожидать освобождения любого рабочего. Если первый рабочий освободился - а это определяется с помощью пары операторов SEIZE (Занять) и RELEASE (Освободить), - то деталь поступает к первому рабочему, где и производится ее обработка. Это можно записать так:

SEIZE RAB1

ADVANCE X3, X4

RELEASE RAB1

TERMINATE

Операторы SEIZE и RELEASE с символьным именем RAB1 обеспечивают сбор статистической информации о канале обслуживания - первом рабочем. Оператор ADVANCE моделирует время выполнения операции первым рабочим (задерживает требование в канале обслуживания). Информация о времени обслуживания (задержки) требования указывается в полях операндов А и В. В поле операнда А представлено среднее время выполнения операции (обслуживания требования), а в поле операнда В - отклонение от среднего времени. Действительное время обслуживания требования определяется так же, как и время поступления деталей (требований) в операторе GENERATE. Так, если датчик случайных чисел выработал число 0,940, то время обслуживания требования будет следующим:

$$X = 11 + (13 - 11) * 0,940 = 12,88.$$

В третьем секторе моделируется процесс обработки деталей вторым рабочим. Если первый рабочий занят, то деталь поступает на обработку ко второму рабочему при условии, что он не занят. Условие незанятости второго рабочего также определяется с помощью своей пары операторов SEIZE и

RELEASE. Это можно записать так:

```
RABO SEIZE      RAB2
  ADVANCE X5, X6
  RELEASE RAB2
  TERMINATE
```

Операторы SEIZE и RELEASE определяют соответственно занятость канала обслуживания (рабочего) и его освобождение. Эти два оператора взаимно дополняют друг друга. В поле операнда A операторов указывается имя канала обслуживания (символьное или числовое). Если канал обслуживания занят, то требование не может войти в него. Если канал обслуживания освободился, то при прохождении требования через оператор RELEASE дается сигнал об освобождении канала, и таким образом, следующее требование может поступить на обслуживание. Оператор ADVANCE моделирует время выполнения операции вторым рабочим.

В четвертом секторе моделируется время функционирования системы - 480 мин. Это выполняется с помощью оператора GENERATE.

В пятом секторе открывается текстовый файл, в который записываются нужные результаты моделирования. Этот сектор может выглядеть так:

```
Fin      OPEN   ("Transp_konveer_rez.txt") , ,Met1
          WRITE  (Catenate("Facility      =",FC$RAB1)) , ,Met
          WRITE  (Catenate("Util. Facility =",FR$RAB1/1000)) , ,Met
          WRITE  (Catenate("Facility      =",FC$RAB2)) , ,Met
          WRITE  (Catenate("Util. Facility =",FR$RAB2/1000)) , ,Met
          CLOSE  Prob, ,Met2
          TERMINATE 1
Met      TERMINATE 1
Met1     TERMINATE 1
Met2     TERMINATE 1
          START   1, NP
```

Оператор OPEN (Открыть) открывает в нашем примере текстовый файл Transp\_konveer\_rez.txt для вывода конкретных результатов моделирования. Результаты моделирования записываются в открытый текстовый файл с помощью оператора WRITE (Записать). В нашем примере записывается число деталей, прошедших через рабочих, и коэффициент использования рабочих. Это выполняется с помощью соответствующих стандартных числовых атрибутов: FC\$RAB1, FR\$RAB1, FC\$RAB2, FR\$RAB2.

Затем открытый файл закрывается с помощью оператора CLOSE (Закреть).

Использование каждого оператора в третьем секторе в случае появления ошибок связано с переходом к оператору завершения - TERMINATE (Завершить).

Оператор TERMINATE удаляет из системы требования: указанное в поле A число требований покидают систему.

START представляет собой управляющий оператор, обеспечивающий выполнение процесса моделирования.

Окончательно программа моделирования этой системы будет выглядеть так, как показано на рис. 2 и 3.

Моделирование работы системы «Транспортный конвейер-рабочий»	
---	--

1	INITIAL X1, 9	Среднее время поступления деталей на обработку
2	INITIAL X2, 1	Отклонение от среднего времени поступления
3	INITIAL X3, 12	Среднее время выполнения операции 1-м рабочим
4	INITIAL X4, 1	Отклонение от среднего времени выполнения операции 1-м рабочим
5	INITIAL X5, 13	Среднее время выполнения операции 2-м рабочим
6	INITIAL X6, 2	Отклонение от среднего времени выполнения операции 2-м рабочим
	*****	
7	GENERATE X1, X2	Интервал поступления деталей на обработку
8	TRANSFER BOTH, ,RABO	Передача деталей 1-му или 2-му рабочему
9	SEIZE RAB1	Ожидание освобождения 1-го рабочего
10	ADVANCE X3, X4	Время выполнения операции 1-м рабочим
11	RELEASE RAB1	Освобождение 1-го рабочего
12	TERMINATE	Завершение выполнения операции
	*****	
13	RABO SEIZE RAB2	Ожидание освобождения 2-го рабочего
14	ADVANCE X5, X6	Время выполнения операции 2-м рабочим
15	RELEASE RAB2	Освобождение 2-го рабочего
16	TERMINATE	Завершение выполнения операции
	*****	
17	GENERATE 480	Время моделирования системы
18	TEST G TG1, 1, Fin	
	*****	
19	Fin OPEN (“ Transp konveer rez.txt”), , Met1	
20	WRITE (Catenate (“ Facility =”, FC\$RAB1)), , Met	
21	WRITE (Catenate (“ Util. Facility =”, FR\$RAB1/1000)), , Met	
22	WRITE (Catenate (“ Facility =”, FC\$RAB2)), , Met	
23	WRITE (Catenate (“ Util. Facility =”, FR\$RAB2/1000)), , Met	
24	CLOSE Prob, , Met2	
25	TERMINATE 1	
26	Met TERMINATE 1	
27	Met1 TERMINATE 1	
28	Met2 TERMINATE 1	
29	START 100	

```

Transp konv
* GPSSM      Transp konv.GPS
*****
* Моделирование работы системы *
* "Транспортный конвейер - рабочий" *
*****
INITIAL      X1, 9          ; Среднее время поступления деталей на обработку.
INITIAL      X2, 1          ; Отклонение от среднего времени поступления.
INITIAL      X3, 12         ; Среднее время выполнения операции 1-м рабочим.
INITIAL      X4, 1          ; Откл. от среднего времени выполнения операции.
INITIAL      X5, 13         ; Среднее время выполнения операции 2-м рабочим.
INITIAL      X6, 2          ; Откл. от среднего времени выполнения операции.
*****
GENERATE     X1, X2         ; Интервал поступления деталей на обработку.
TRANSFER    BOTH, , RABO   ; Передача деталей 1-му или 2-му рабочему.
SEIZE       RAB1           ; Ожидание освобождения 1-го рабочего.
ADVANCE     X3, X4         ; Время выполнения операции 1-м рабочим.
RELEASE     RAB1           ; Освобождение 1-го рабочего.
TERMINATE   ; Завершение выполнения операции.
*****
RABO SEIZE   RAB2           ; Ожидание освобождения 2-го рабочего.
ADVANCE     X5, X6         ; Время выполнения операции 2-м рабочим.
RELEASE     RAB2           ; Освобождение 2-го рабочего.
TERMINATE   ; Завершение выполнения операции.
*****

```

Рисунок 2. Начало программы моделирования системы «Транспортный конвейер-рабочий»

```

Transp konveer
*****
GENERATE     480           ; Время моделирования системы.
TEST        G            TGI, 1, Fin
*****
Fin        OPEN          ("Transp konveer rez.txt"), Met1
           WRITE         (Catenate("Facility = ", FCSRAB1)), Met
           WRITE         (Catenate("Util. Facility = ", FRSRAB1/1000)), Met
           WRITE         (Catenate("Facility = ", FCSRAB2)), Met
           WRITE         (Catenate("Util. Facility = ", FRSRAB2/1000)), Met
CLOSE      Prob, Met2
TERMINATE  1
Met        TERMINATE    1
Met1       TERMINATE    1
Met2       TERMINATE    1
START     1, NP
*****

```

Рисунок 3. Продолжение программы моделирования системы «Транспортный конвейер-рабочий»

**Варианты заданий**

№	t1, мин	t2, мин	t3, мин	№	t1, мин	t2, мин	t3, мин
1	10-12	12,4-13,6	12,4-13,2	14	11-12,4	14,2-18,1	15,2-19
2	7-9	8-9,2	10,4-11	15	9,2-10,6	12,6-14,3	13,4-15,8
3	8-12	10,2-13	12,2-14,4	16	8,2-9,4	11-12,4	12,2-13
4	8-11	12,4-14,2	9,2-11,5	17	6,4-8	8,4-10,2	8,2-12,2
5	3-7	4,2-8,4	5,8-7,2	18	10-11,8	13,4-15,2	14,4-17,6
6	10-12	13,6-18,2	14,2-18,8	19	6,4-10	10,2-13,4	11,2-14,4
7	9-10,4	11,8-14,4	11,4-15,2	20	5,2-6	7,8-12,2	9,2-12,4
8	10-12	12,4-12,6	14,2-14,8	21	6,8-7	8,2-14	11,8-12,6
9	6-8	8,4-10,2	9,4-10,4	22	9,3-9,7	10,2-13,4	10,2-14,8
10	12-12,4	15,2-15,4	13,2-16,1	23	10-11,4	11,4-13,2	12,4-15,2
11	12-13,6	17,8-19,2	17,2-20,7	24	2,2-2,8	3,2-7,4	4,2-8
12	11-13,2	13,2-14,8	13,4-	25	4-5,2	5,6-10,2	7,4-11,2

			13,8				
<b>13</b>	10-12	12,2-13,5	12,2-13,3	<b>26</b>	5,2-6,4	7,8-10,4	8,2-9

### Контрольные вопросы.

1. Пояснить структуру окна GPSSW
2. Какие файлы имеют расширение .gps.
3. Пояснить пункт меню Search и Window
4. Пояснить пункты меню Command – **Create Simulation, Retranslate**
5. Пояснить команду редактирования – Insert GPSS Blocks
6. Какие файлы имеют расширение .sim.
7. Перечислить и пояснить вкладки диалогового окна Settings ...
8. Пояснить пункты меню Command – **Repeat Last Command, Conduct**
9. Перечислить кнопки панели инструментов
10. Какие файлы имеют расширение .txt.
11. Пояснить пункты меню Command – **START, STEP1**
12. Пояснить команду редактирования – Insert Experiment
13. Какие файлы имеют расширение .gpr.
14. Пояснить пункты меню Command – **HALT, CLEAR, RESET**
15. Как определяются арифметические целые переменные и булевы переменные
16. Понятие вычислительного выражения
17. Пояснить приоритеты выполнения операций в арифметических выражениях
18. Пояснить этапы разработки программы – постановка задачи и выявление основных особенностей
19. Какие символы обозначения используются в системе GPSSW
20. Пояснить этапы разработки программы – создание исходной модели и представление имитационной модели в системе GPSSW
21. Пояснить назначение динамических объектов системы
22. Какими переменными характеризуются транзакты
23. Какие типы данных используются в GPSSW
24. Как определяются арифметические переменные с фиксированной точкой
25. Пояснить этапы разработки программы – моделирование системы и отладка модели

Вариант	Вопрос № 1	Вопрос № 2	Вопрос № 3	Вариант	Вопрос № 1	Вопрос № 2	Вопрос № 3
<b>1</b>	1	9	18	<b>14</b>	14	22	6
<b>2</b>	2	10	19	<b>15</b>	15	23	7
<b>3</b>	3	11	20	<b>16</b>	16	24	8
<b>4</b>	4	12	21	<b>17</b>	17	25	9
<b>5</b>	5	13	22	<b>18</b>	18	1	10
<b>6</b>	6	14	23	<b>19</b>	19	2	11
<b>7</b>	7	15	24	<b>20</b>	20	3	12
<b>8</b>	8	16	25	<b>21</b>	21	4	13
<b>9</b>	9	17	1	<b>22</b>	22	5	14
<b>10</b>	10	18	2	<b>23</b>	23	6	15

11	11	19	3	24	24	7	16
12	12	20	4	25	25	8	17
13	13	21	5	26	1	10	20

### Практическая работа № 3

**Тема:** «Разработка модели СМО»

**Цель работы:** «Научиться разрабатывать алгоритмы и программы для решения практических задач»

#### Задача

Промоделировать движение на пешеходном переходе, оборудованном светофором. Допустим, что в одном направлении автомобили подъезжают к переходу с равномерным распределением с интервалом времени ( $t_1$ )  $20 \pm 10$  с, а пешеходы, желающие пересечь улицу по переходу, прибывают к нему с интервалом ( $t_2$ )  $30 \pm 10$  с также с равномерным распределением. Если переход занят пешеходами, то дорожное движение останавливается, и создается очередь из ожидающих автомобилей. Если сигнал светофора зеленый и пешеходный переход не занят пешеходами, то автомобили проезжают. Если сигнал светофора красный или пешеходный переход занят пешеходами, то автомобили не могут проезжать. Время проезда через пешеходный переход составляет ( $t_3$ )  $10 \pm 2$  с и распределяется согласно равномерно распределенному закону. Время прохода пешехода через пешеходный переход составляет ( $t_4$ )  $10 \pm 2$  с

Требуется:

- определить основные параметры функционирования пешеходного перехода, накопив статистику после проезда через него в одном направлении 1000 автомобилей;
- определить среднюю загрузку перехода.

**Для задачи Вашего варианта выполнить задание.**

1. Составить структурную схему модели исследуемой системы. Пояснить назначение элементов.

2. Записать программу в системе GPSS World.

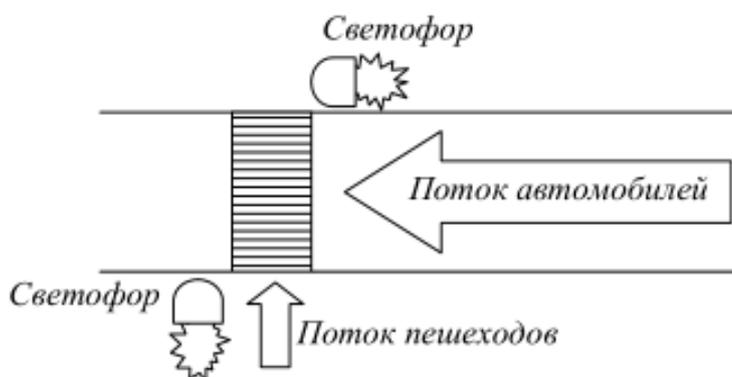


Рисунок 1. Пешеходный переход

#### **Выявление основных особенностей**

Для данного процесса можно взять в качестве единицы измерения времени секунду. Для моделирования заданного процесса необходимо сформировать два входных потока: поток автомобилей, подъезжающих к переходу, и поток пешеходов, использующих переход. В качестве ограничителя периода имитации можно использовать число автомобилей, подъезжающих к переходу с одной стороны, например 1000.

Для решения нашей задачи потребуется создать три сегмента модели для моделирования:

- поток автомобилей;
- поток пешеходов;
- режим работы светофора.

### ***Построение имитационной модели***

Построение имитационной модели следует начинать с создания заголовка модели, который может быть представлен, например, в таком виде:

GPSSW Peshehod.GPS

\*\*\*\*\*

\* Моделирование пешеходного перехода\*

\*\*\*\*\*

Сначала сформируем поток машин, пересекающих переход, – первый сегмент программы:

\*\*\*\*\*

\* Моделирование потока машин \*

\*\*\*\*\*

Это можно сделать с помощью оператора **GENERATE**, который в нашей задаче запишем в таком виде:

GENERATE 20,10

В поле операнда А указывается среднее время поступления (обязательно) требования (машины) на обслуживание (проезд через переход), в поле операнда В – отклонение от среднего времени поступления требования (не обязательно).

Далее введем пару операторов: сначала **QUEUE**, а затем через несколько операторов и **DEPART** с одним и тем же именем (числовым или символьным) в поле операнда А для сбора статистической информации по очереди машин перед переходом. Эта очередь образуется во время занятости пешеходного перехода или из-за красного сигнала на светофоре. Условно назовем эту очередь именем Ocher1. Сначала зафиксируем вход в очередь автомобиля с помощью оператора **QUEUE**. Это будет выглядеть так:

QUEUE Ocher1

Когда сигнал светофора зеленый (значение 0) для автомобилей и пешеходный переход не занят, тогда значение стандартного числового атрибута, оценивающего возможность проезда автомобиля через пешеходный переход, принимается равным 0. Это означает, что автомобиль может пересекать пешеходный переход. Эти условия должны быть протестированы блочным оператором **TEST**. Когда сигнал светофора красный (значение 100) или переход не свободен от пешеходов, тогда значению стандартного числового атрибута, оценивающего возможность проезда, возвращается значение, равное 1. Это можно записать так:

TEST E X\$\$vet\_Avtom,F\$Perehod

В операторе **TEST** использован логический оператор E (Equal – Равно). Как можно заметить, поле операнда С пусто, следовательно, требование не может войти в блок **TEST**, пока заданное условие числовых атрибутов, приведенных в полях операндов А и В, не выполнится.

В поле операнда А находится сохраняемая величина по имени Svet\_Avtom, а в поле В – стандартный числовой атрибут, характеризующий содержимое

канала обслуживания – наличие пешеходов на переходе. Когда сигнал светофора зеленый – сохраняемая величина по имени Svet\_Avtom равна 0 – и пешеходный переход не занят – стандартный числовой атрибут по имени F\$Perehod равен 0, – тогда машина (требование) может пересекать переход.

Далее введем пару операторов: сначала **SEIZE**, а затем через несколько операторов и **RELEASE** с одним и тем же именем (числовым или символьным) в поле операнда A для сбора статистической информации по каналу обслуживания – переходу:

```
SEIZE Perehod
```

Если канал обслуживания освобожден, то требование может покинуть очередь, используя оператор **DEPART**:

```
DEPART Ocher1
```

Далее автомобиль пересекает пешеходный переход, затрачивая на это время, равное  $10 \pm 2$  с. Это действие моделируется оператором **ADVANCE**:

```
ADVANCE 10,2
```

В поле операнда A указывается среднее время пересечения перехода автомобилем (обязательно), в поле операнда B – отклонение от среднего времени пересечения перехода автомобилем (не обязательно). Далее вводится оператор **RELEASE**, сообщающий об освобождении канала обслуживания:

```
RELEASE Perehod
```

Далее используется оператор **TERMINATE** для завершения моделирования в первом сегменте программы:

```
TERMINATE 1
```

После этого создаем другой сегмент программы, в котором формируется второй поток – поток пешеходов:

```
*****
```

```
*Моделирование потока пешеходов*
```

```
*****
```

Он очень похож на первый сегмент программы. Оператор **GENERATE** генерирует поток пешеходов с временем подхода пешеходов к переходу, равным  $30 \pm 10$  с:

```
GENERATE 30,10
```

Далее введем пару операторов: сначала **QUEUE**, а затем через несколько операторов и **DEPART** с одним и тем же именем (числовым или символьным) в поле операнда A для сбора статистической информации по очереди пешеходов. Эта очередь создается во время занятости пешеходного перехода машинами или из-за красного сигнала на светофоре. Условно назовем эту очередь Ocher2. Сначала зафиксируем вход в очередь автомобиля с помощью оператора **QUEUE**. Это будет выглядеть так:

```
QUEUE Ocher2
```

Когда сигнал светофора зеленый (значение 0) для пешехода, тогда значение стандартного числового атрибута, оценивающего возможность прохода пешеходом перехода, принимается равным 0. Это означает, что пешеход может пересекать улицу. Эти условия должны быть протестированы блочным оператором **TEST**. Когда сигнал светофора красный (значение 100) или переход занят автомобилем, тогда стандартному числовому атрибуту, оценивающему возможность перехода, возвращается значение, равное 1. Это можно записать так:

```
TEST E X$$vet_Pesheh, F$Perehod
```

В поле операнда А находится сохраняемая величина по имени Svet\_Pesheh, а в поле В – стандартный числовой атрибут, характеризующий содержимое канала обслуживания – наличие автомобилей на пешеходном переходе. Когда сигнал светофора зеленый – сохраняемая величина по имени Svet\_Pesheh равна 0 – и пешеходный переход не занят – стандартный числовой атрибут по имени F\$Perehod равен 0, – тогда пешеход (требование) может пойти по переходу.

Далее введем пару операторов: сначала **SEIZE**, а затем через несколько операторов и **RELEASE** с одним и тем же именем (числовым или символьным) в поле операнда А – Perehod – для сбора статистической информации по каналу обслуживания:

**SEIZE** Perehod

Если канал обслуживания – переход – разрешен и свободен от автомобилей, то требование может покинуть очередь, используя оператор **DEPART**:

**DEPART** Ocher2

Далее пешеход пересекает переход, затрачивая на это время, равное  $10 \pm 2$  с. Это действие моделируется оператором **ADVANCE**:

**ADVANCE** 10,2

В поле операнда А указывается среднее время прохода перехода пешеходом (обязательно), в поле операнда В – отклонение от среднего времени прохода перехода пешеходом (не обязательно). Далее вводится оператор **RELEASE**, сообщающий об освобождении канала обслуживания (перехода):

**RELEASE** Perehod

Затем используется оператор **TERMINATE** для завершения моделирования во втором сегменте программы:

**TERMINATE** 1

Переходим к созданию третьего сектора программы – моделирования работы светофора:

\*\*\*\*\*

\*Моделирование работы светофора\*

\*\*\*\*\*

**GENERATE** ,,1

Оператор **GENERATE** генерирует одно требование (транзакт), обеспечивающее работу светофора. Для этого в поле операнда D вводится значение 1.

Далее используется сохраняемая величина под именем Svet\_Pesheh для указания цвета сигнала светофора со стороны пешеходов – Krasn (Красный):

**Begin1 SAVEVALUE** Svet\_Pesheh, Krasn

Затем используется сохраняемая величина под именем Svet\_Avtom для указания цвета сигнала светофора со стороны автомобилей – Zelen (Зеленый):

**SAVEVALUE** Svet\_Avtom, Zelen

Далее моделируется время включенного состояния зеленого сигнала с использованием оператора **ADVANCE**. В поле А оператора **ADVANCE** вводится имя переменной Zelen\_time, в которой хранится информация о продолжительности зеленого сигнала светофора:

**ADVANCE** Zelen\_time

Затем используется сохраняемая величина под именем Svet\_Pesheh для указания в ней цвета сигнала светофора со стороны пешеходов – Zelen:

**SAVEVALUE** Svet\_Pesheh, Zelen

Потом используется сохраняемая величина под именем Svet\_Avtom для указания в ней цвета сигнала светофора со стороны автомобилей – Krasn:

```
SAVEVALUE Svet_Avtom, Krasn
```

Далее моделируется время включенного состояния зеленого сигнала с использованием оператора **ADVANCE**. В поле А оператора **ADVANCE** вводится имя переменной Krasn\_time, в которой хранится информация о продолжительности зеленого сигнала светофора:

```
ADVANCE Krasn_time
```

Затем все повторяется сначала. Для этого используется оператор **TRANSFER** в режиме безусловного перехода к оператору с символической меткой Begin1:

```
TRANSFER , Begin1
```

И наконец, используемым ранее переменным присваиваются конкретные числовые значения. Это можно записать так:

```
Zelen_time EQU 200
```

```
Zelen EQU 0
```

```
Krasn EQU 100
```

```
Krasn_time EQU 300
```

В окончательном виде модель, имитирующая движение на пешеходном переходе, будет выглядеть так, как показано на рис. 2 и 3.

### **Программа моделирования пешеходного перехода**

```
GENERATE 20,10
QUEUE Ocher1
TEST E X$Svet_Avtom,F$Perehod
SEIZE Perehod
DEPART Ocher1
ADVANCE 10,2
RELEASE Perehod
TERMINATE 1
GENERATE 30,10
QUEUE Ocher2
TEST E X$Svet_Pesheh,F$Perehod
SEIZE Perehod
DEPART Ocher2
ADVANCE 10,2
RELEASE Perehod
TERMINATE 1
GENERATE ,,,1
Begin1 SAVEVALUE Svet_Pesheh,Krasn
SAVEVALUE Svet_Avtom,Zelen
ADVANCE Zelen_time
SAVEVALUE Svet_Pesheh,Zelen
SAVEVALUE Svet_Avtom,Krasn
ADVANCE Krasn_time
TRANSFER ,Begin1
Zelen_time EQU 200
Zelen EQU 0
Krasn EQU 100
Krasn_time EQU 300
Start 1000
```

```

Peshehod_1
; GPSSW      Peshehod.GPS
*****
* Моделирование пешеходного перехода *
*****
* Моделирование потока машин *
*****
GENERATE 20,10      ; формирование потока машин у перехода.
QUEUE    Ocher1     ; Вход в очередь машин перед переходом.
TEST E   X$Svet_Avtom,F$Perehod ; Запрет движения машин до появления зеленого
SEIZE   Perehod     ; цвета светофора и освобождения перехода от пешеходов.
DEPART  Ocher1     ; Выход из очереди машин перед переходом.
ADVANCE 10,2       ; Пересечение машины перехода.
RELEASE Perehod     ; Освобождение перехода от машины.
TERMINATE 1        ; Удаление машины из системы.
*****
* Моделирование потока пешеходов *
*****
GENERATE 30,10     ; формирование потока пешеходов у перехода.
QUEUE    Ocher2     ; Вход в очередь пешеходов перед переходом.
TEST E   X$Svet_Pesheh,F$Perehod ; Запрет входа на переход до появления зеленого
SEIZE   Perehod     ; цвета светофора и освобождения перехода от машин.
DEPART  Ocher2     ; Выход из очереди пешеходов перед переходом.
ADVANCE 10,2       ; Пересечение пешеходов перехода.
RELEASE Perehod     ; Освобождение перехода от пешехода.
TERMINATE 1        ; Удаление пешехода из системы.
*****

```

Рис. 2. Первая часть имитационной модели пешеходного перехода

```

Peshehod_1
*****
* Моделирование работы светофора *
*****
GENERATE ,,1      ; Генерирование состояния светофора.
Begin1 SAVEVALUE Svet_Pesheh,Krasn ; Сохранение красного цвета светофора со стороны пешехода
SAVEVALUE Svet_Avtom,Zelen ; Сохранение зеленого цвета светофора со стороны машин.
ADVANCE Zelen_time ; Время горения зеленого цвета светофора.
SAVEVALUE Svet_Pesheh,Zelen ; Сохранение зеленого цвета светофора со стороны пешехода
SAVEVALUE Svet_Avtom,Krasn ; Сохранение красного цвета светофорасо стороны машин.
ADVANCE Krasn_time ; Время горения красного цвета светофора.
TRANSFER ,Begin1 ; Переход коператору с меткой Begin1.
Zelen_time EQU 200 ; Время горения зеленого цвета светофора.
Zelen EQU 0
Krasn EQU 100
Krasn_time EQU 300 ; Время горения красного цвета светофора.
START 1000

```

Рис. 3. Вторая часть имитационной модели пешеходного перехода

### Варианты заданий

№	t1, сек	t2, сек	t3, сек	t4, сек
1	19±10	31±10	9±2	15±2
2	18±10	32±10	8±2	14±2
3	17±10	33±10	7±2	13±2
4	16±10	34±10	6±2	12±2
5	15±10	35±10	5±2	11±2
6	21±10	29±10	11±2	16±2
7	22±10	28±10	12±2	17±2
8	23±10	277±10	13±2	18±2
9	24±10	26±10	14±2	19±2
10	25±10	25±10	15±2	20±2
11	19±5	31±10	9±3	15±3
12	18±5	32±5	8±3	14±3
13	17±5	33±5	7±3	13±3
14	16±5	34±5	6±3	12±3
15	15±5	35±5	5±3	11±3
16	21±5	29±5	11±3	16±3
17	22±5	28±5	12±3	17±3
18	23±5	277±5	13±3	18±3
19	24±5	26±5	14±3	19±3
20	25±5	25±5	15±3	20±3
21	23±15	277±15	13±4	18±4
22	24±15	26±15	14±4	19±4

23	25±15	25±15	15±4	20±4
24	19±3	31±15	9±1	15±1
25	18±3	32±3	8±1	14±1
26	17±3	33±3	7±1	13±1

### Контрольные вопросы.

Пояснить операторы GPSSW

1. GENERATE A,B,C,D,E
2. M FUNCTION A,B
3. SEIZE A
4. RELEASE A
5. ADVANCE A,B
6. ENTER A,B
7. M STORAGE A
8. QUEUE A,B
9. DEPART A,B
10. TERMINATE A
11. START A,B,C
12. TEST XX A,B,C
13. LEAVE A,B
14. SAVEVALUE A,B

Вариант					Вариант				
1	1	9	14	6	14	14	2	6	10
2	2	14	1	5	15	1	9	7	11
3	3	11	2	9	16	2	10	8	6
4	4	12	3	1	17	3	8	9	12
5	5	13	4	8	18	4	1	10	7
6	6	14	5	10	19	5	2	11	9
7	7	5	6	11	20	6	3	12	10
8	8	6	7	12	21	7	4	13	1
9	9	7	8	13	22	8	5	14	2
10	10	8	9	14	23	9	6	1	4
11	11	9	10	1	24	10	7	2	13
12	12	10	11	2	25	11	8	4	6
13	13	11	12	3	26	12	10	3	8

### Практическая работа № 4

**Тема:** «Синтез схем на основе логических элементов»

**Цель работы:** Научиться составлять схемы из логических элементов, подключать входы/выходы, выполнять компиляцию схемы.

**Порядок выполнения работы**

#### 1. Загрузить Quartus

Для этого два раза щелкнуть мышкой на пиктограмме «Quartus» на рабочем столе.

#### 2. Создать проект.

Для создания нового проекта используется утилита *New Project Wizard* (NPW). Для её вызова выполнить шаги:

- выбрать **File** на панели инструментов
- в появившемся окне выбрать строку *New Project Wizard*.

В появившемся вводном окне перечислены пять шагов, которые необходимо выполнить для создания нового проекта, нажать кнопку **Next**

1. В окне (page1) NPW выбрать рабочую папку для размещения проекта, задать имя проекта и определить модуль верхнего уровня.

2. В окне (page2) **Add Files** предлагается импортировать в проект файлы, нажать **Next**.

3. В окне (page3) пользователь выбирает ПЛИС, на которой будет реализован проект. При выборе кристалла:

- выбирается его семейство в окне **Family**,
- определяется тип корпуса интегральной схемы в окне **Package**,
- указывается количество выводов кристалла в окне **Pin Count**,
- указывается градация быстродействия кристалла в окне **Speed grade**.

Подходящий кристалл выбирается из списка в окне **Available Devices**, нажать **Next**.

4. В окне (page 4) пользователь может определить средства автоматизации проектирования сторонних производителей, которые он планирует использовать для ввода проекта, синтеза, моделирования или временного анализа, нажать **Next**.

5. В окне (page 5) приводится итоговая информация по сделанному пользователем выбору на предыдущих шагах, а также информация по условиям работы выбранного кристалла.

Нажатие на кнопку **Finish** завершит работу *New Project Wizard*.

В окне навигатора проекта на странице **Hierarchy** появится тип выбранного кристалла ПЛИС и имя модуля верхнего уровня, а на странице **Files** - добавленные к проекту файлы.

Для лабораторных работ в семействе **Cyclone** выбрать кристалл **EP1C12Q240C7**.

### 3. Синтез схемы.

#### 3.1. Создание файла верхнего уровня.

1. Выбрать пункт меню **File**, в выпадающем меню выбрать строку *New*, далее в окне возможных вариантов типов исходных файлов выбрать тип файла – **Block Diagram/Schematic File**.

2. В главном окне менеджера проекта Quartus появится изображение пустого схемного файла с названием **block1.bdf**.

3. Присвоить новому файлу собственное имя.

- выбрать пункт меню **File**,
- в выпадающем меню, выполнить команду *Save As*.
- В поле «Имя файла» напечатать назначаемое имя, установите галочку в поле *Add file to current project* и нажать **Сохранить**, что приведет к добавлению вновь созданного файла к проекту.

#### 3.2. Создание схемы.

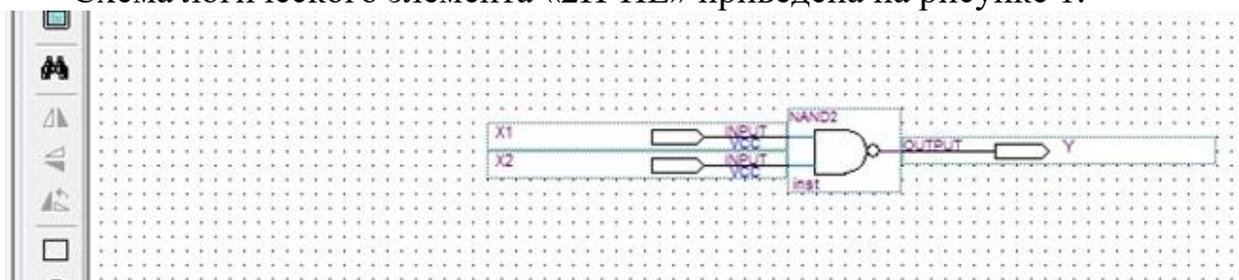
Нажатие кнопки  на панели инструментов (в левой части окна) приводит к вызову библиотеки символов, из которой могут быть извлечены необходимые элементы и помещены в схему.

Открыть библиотеку «Libraries». Из подгруппы «primitives», «logic» выбираем: двухвходовой элемент «2И-НЕ» (NAND2) – 1 шт. Из подгруппы «primitives», «pin» выбираем: входы INPUT – 2шт., выход OUTPUT – 1шт.

Для помещения элемента на рабочее поле необходимо выбрать элемент из списка или записать его имя в поле Name, нажать «OK» и разместить в нужном месте. Установка «галочки» в поле Repeat-insert позволит многократно копировать выбранный элемент.

Двойной щелчок левой кнопки на выбранном элементе открывает окно свойства элемента Properties, в котором в частности можно переименовать выходы элементов.

Схема логического элемента «2И-НЕ» приведена на рисунке 1.

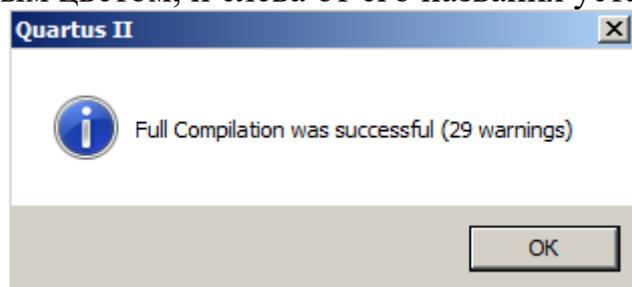


#### 4. Компиляция проекта

Для запуска процесса выполнения отдельных этапов или шагов компиляции необходимо:

- выбрать пункт меню **Processing**
- выбрать команду **Start**
- в появившемся окне указать выполняемый этап компиляции **Start Analysis & Synthesis**.

После успешного завершения этапа компиляции на экран выводится соответствующее сообщение, а в окне задач выполненный этап изображается зеленым цветом, и слева от его названия устанавливается зеленая галочка.



Отчет о результатах компиляции появляется сразу после завершения компиляции проекта.

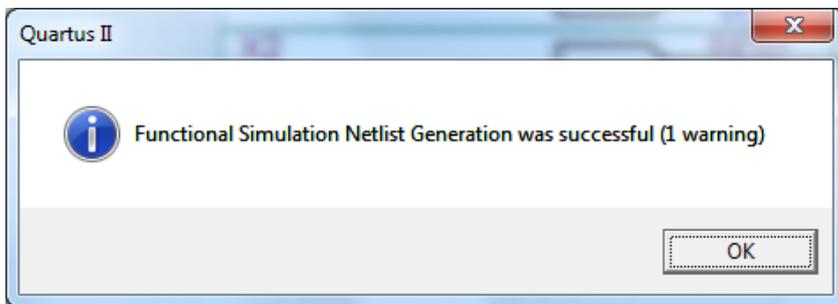
#### 5. Создание списка соединений узлов схемы

Для выполнения функционального моделирования достаточно создать модель устройства в виде списка соединений узлов схемы (**netlist for simulation**).

Для создания списка соединений необходимо:

- выбрать пункт меню **Processing**
- выполнить команду **Generate Functional Simulation Netlist**

После успешного завершения создания списка соединений на экран выводится соответствующее сообщение



**Задание.** Создать проект для анализа работы логического элемента, подключить входы и выход элемента, выполнить компиляцию схемы.

<b>№ вар</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>
Тип ИС	and2	and3	and4	nand3	nand4	or2	or3	or4	nor2	nor3
<b>№ вар</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>
Тип ИС	nor4	xor	xnor	and2	and3	and4	nand3	nand4	or2	or3
<b>№ вар</b>	<b>21</b>	<b>22</b>	<b>23</b>	<b>24</b>	<b>25</b>	<b>26</b>	<b>27</b>	<b>28</b>	<b>29</b>	<b>30</b>
Тип ИС	or4	nor2	nor3	nor4	xor	xnor	and2	and3	and4	nand2

**Таблица 1 – Соответствия российских обозначений элементов и обозначений в Quartus II.**

Название элемента	Российское обозначение	Название элемента в Quartus	Обозначение в Quartus
"И"		and	
"И-НЕ"		nand	
"ИЛИ"		or	
«ИЛИ-НЕ»		nor	
«ИСКЛЮЧАЮЩЕЕ ИЛИ»		xor	
«ИСКЛЮЧАЮЩЕЕ ИЛИ-НЕ»		xnor	
«НЕ»		not	

### Контрольные вопросы.

1. Пояснить назначение САПР «Quartus»
2. Перечислить этапы проектирования в САПР «Quartus»
3. Пояснить назначение и привести примеры логических файлов
4. Пояснить назначение и привести примеры вспомогательных файлов
5. Перечислить элементы интерфейса САПР «Quartus»
6. Пояснить назначение окна менеджера проекта «Quartus» *Project Navigator*
7. Пояснить назначение окна менеджера проекта «Quartus» *Tasks*
8. Пояснить назначение окна менеджера проекта «Quartus» *Messages*
9. Пояснить назначение утилиты *New Project Wizard*
10. Пояснить последовательность создания нового проекта

Вариант					Вариант				
1	1	9	10	6	14	4	2	6	10
2	2	9	1	5	15	1	9	7	10
3	3	10	2	9	16	2	10	8	6
4	4	2	3	1	17	3	8	9	2
5	5	3	4	8	18	4	1	10	7
6	6	4	5	10	19	5	2	10	9
7	7	5	6	10	20	6	3	2	10
8	8	6	7	10	21	7	4	3	1
9	9	7	8	3	22	8	5	4	2
10	10	8	9	4	23	9	6	1	4
11	7	9	10	1	24	10	7	2	9
12	8	10	1	2	25	10	8	4	6
13	3	10	2	6	26	2	10	3	8

## Практическая работа № 5

**Тема:** «Синтез комбинационных схем».

**Цель работы:** Научиться составлять структурные схемы комбинационных устройств, выполнять компиляцию и устранять обнаруженные ошибки.

**Задача.** Синтезировать схему двухвходового мультиплексора и выполнить компиляцию схемы.

### Порядок выполнения работы

#### 1. Загрузить Quartus

Все практические и лабораторные работы программы моделирования электронных схем будем создавать в папке Project на Рабочем столе.

**2. Создать проект.** Для этого выполнить следующие действия:

– выбрать пункт File

– выбрать New Project Wizard

– в появившемся окне нажимаем Next.

– в окне New Project Wizard: Directory, Name, Top-Level Entity (page 1 of 5) в строке What is the working directory for this project, щелкнуть кнопку с тремя точками и выбрать место расположения создаваемого проекта – Рабочий стол.

– создаем папку Project (для всех практических и лабораторных работ), а в ней папку Prakt2 и папку Prakt2 открыть и ещё раз щелкнуть открыть.

– во второй строке записываем имя проекта Prakt2.

– нажимаем Next

Окно New Project Wizard: Add, Files (page 2 of 5) предлагает импортировать в проект файлы, но т.к. их пока нет, нажимаем Next.

Окно New Project Wizard: Family & device you want to target for Compilation предлагает выбрать микросхему. Выбираем в строке Family микросхемы **Cyclone** и в строке Available device тип микросхемы **EP1C12Q240C7**. Нажимаем Next.

Окно New Project Wizard: EDA Tool Setting (page 4 of 5) предлагает выбрать встроенные программы для синтеза проекта, симуляции. Нажимаем Next.

Окно New Project Wizard: Summary (page 5 of 5) показывает все исходные настройки для нашего проекта. Нажимаем Finish. Мы создали проект, пока пустой.

В окне навигатора проекта Project Navigator на вкладке иерархия проекта Hierarchy появился проект Prakt2 (файл с расширением .qpf).

#### 3. Синтез схемы.

– Создание файла верхнего уровня. Для этого необходимо выполнить действия: «File», «New», и выбрать тип файла Block Diagram / Schematic File

Нажимаем «OK», щелкаем «Save as», записываем имя Prakt2 (файл сохраняется с расширением .bdf), убедиться в наличии галочки в поле «Add file current project» (для добавления файла к проекту) и щелкнуть сохранить.

– Создание схемы.

Нажатие кнопки  приводит к вызову библиотеки символов, из которой могут быть извлечены необходимые элементы и помещены в схему.

Открыть библиотеку «Libraries». Из подгруппы «primitives», «logic» выбираем: элементы «НЕ» (NOT) – 5шт., элементы «3И» (AND3) – 4шт., элементы «2И» (AND2) – 2 шт., элемент «4ИЛИ» (OR4) – 1шт., элемент «2ИЛИ» (OR2) – 1шт.

Из подгруппы «primitives», «pin» выбираем: входы INPUT – 3шт., выходы OUTPUT – 2шт.

Для помещения элемента на рабочее поле необходимо выбрать элемент из списка и нажать «OK» и разместить в нужном месте. Установка «галочки» в поле Repeat-insert позволит многократно копировать выбранный элемент.

Двойной щелчок левой кнопки на выбранном элементе открывает окно свойства элемента Properties, в котором можно переименовать выходы элементов.

Схема двухвходового мультиплексора приведена на рисунке 1.

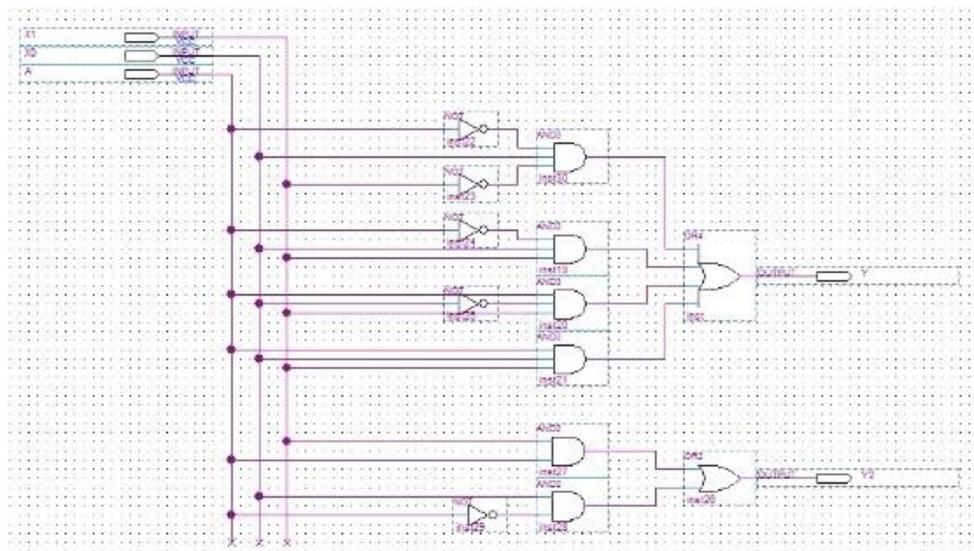


Рисунок 1. Схема двухвходового мультиплексора  
Синтезированную схему сохранить.

### 3. Компиляция схемы

Компиляция проекта состоит из трех частей: синтез, разводка и ассемблирование. Мы рассмотрим только первую часть – синтез. Синтез осуществляется кнопкой  – «Start Analysis & Synthesis». В рабочем окне происходит процесс компиляции. В появившемся окне видим, что процесс компиляции прошел успешно – «Analysis & Synthesis was successful». Щелкнем «OK». Появилось некое суммарное окно, характеризующее проект, закроем его.

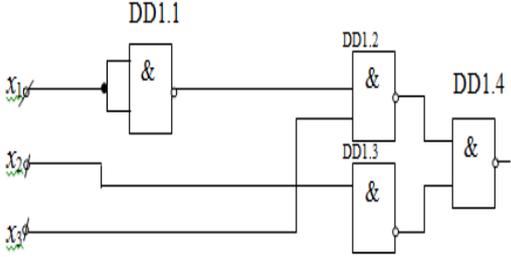
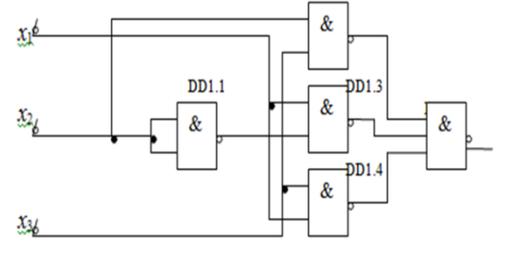
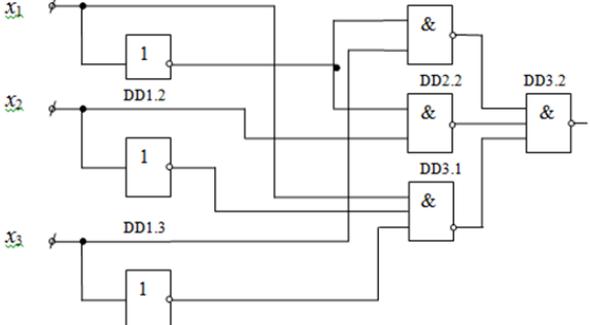
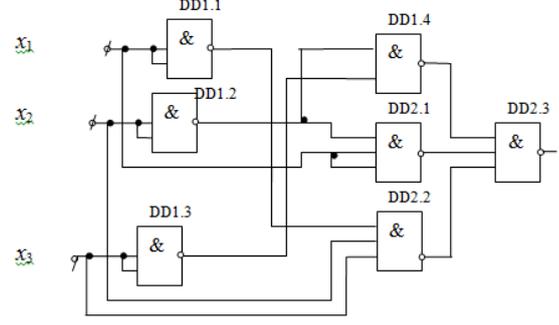
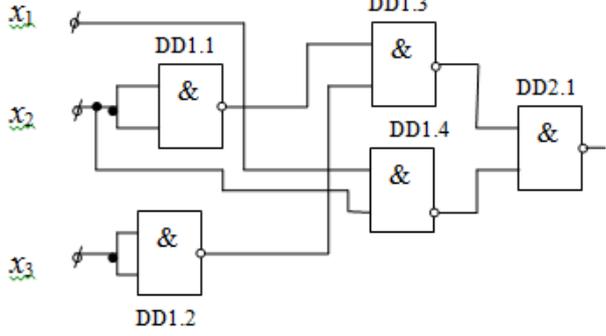
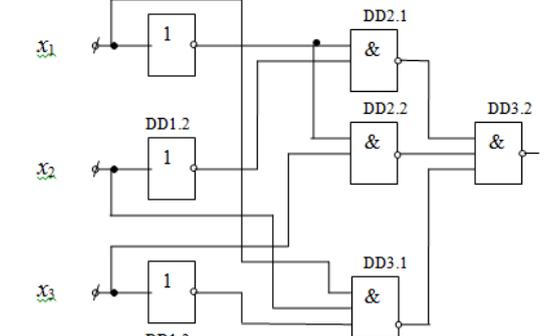
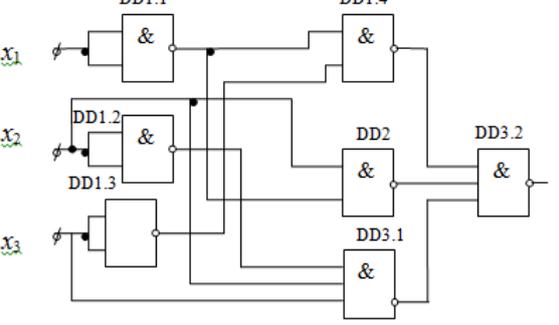
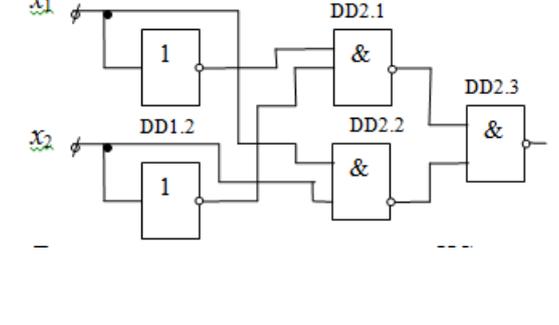
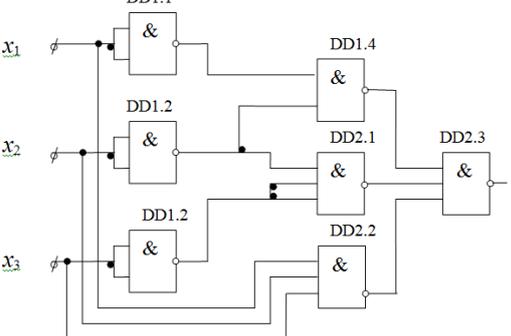
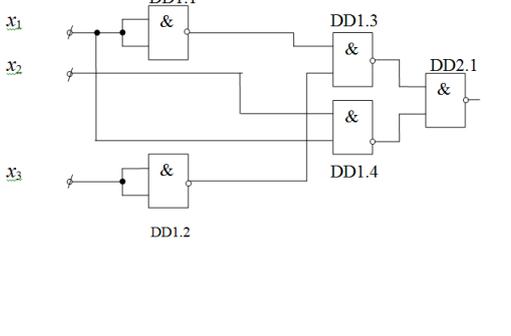
Выбираем пункт «Processing», «Simulator Tool» в результате появляется окно симуляции «Simulator Tool».

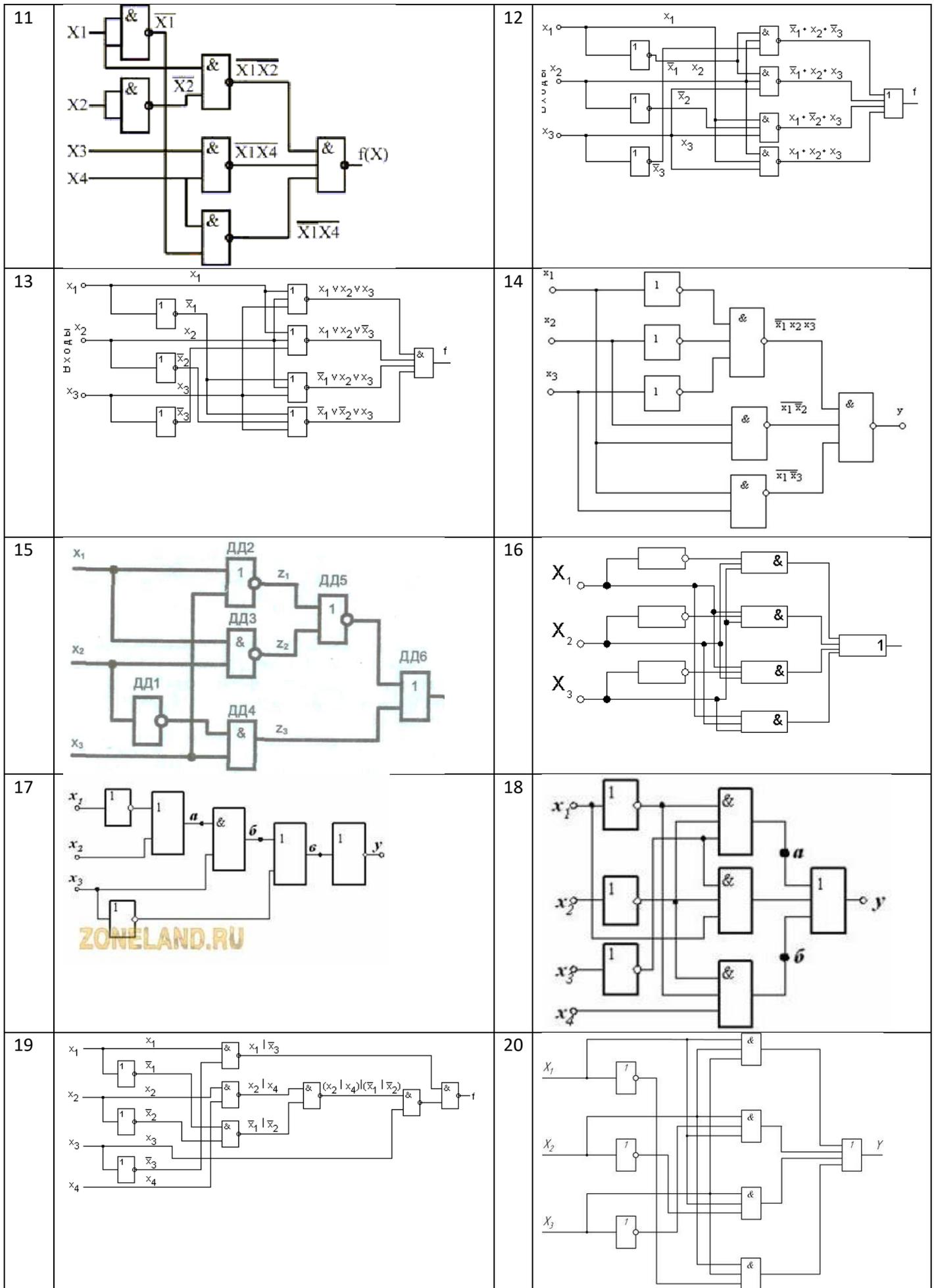
В верхней строке «Simulation mode» выбираем функциональную симуляцию – «Functional». И после каждого синтеза необходимо создавать список соединений

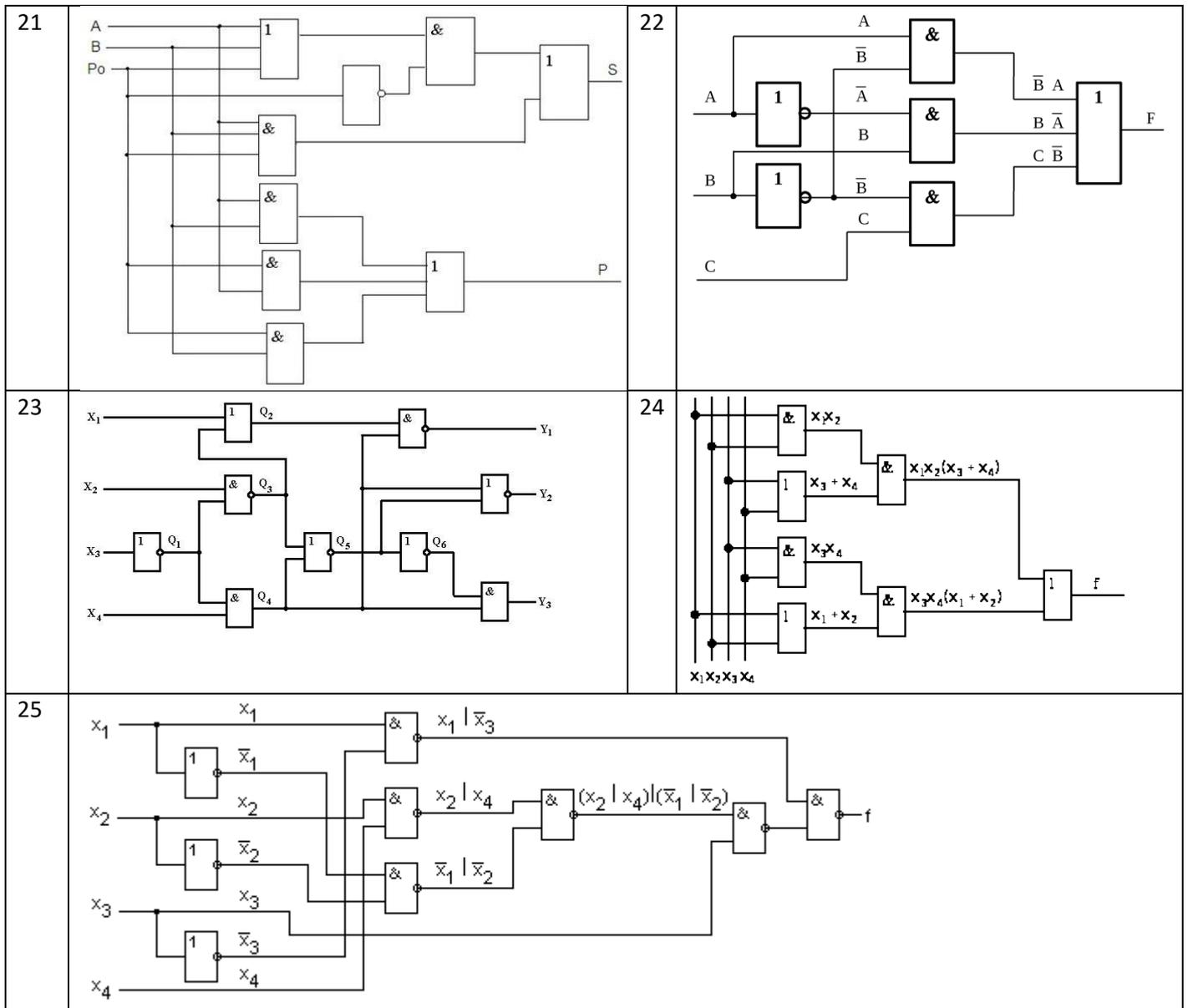
– «Netlist» нажатием кнопки «Generate Functional Simulation Netlist». Сообщение – «Functional Simulation Netlist Generate was successful» говорит о том, что генерация «Netlist» выполнена успешно. Щелкнем «ОК».

Процесс построения схемы двухвходового мультиплексора завершен. В процессе компиляции ошибок не обнаружено.

### Задание

1		2	
3		4	
5		6	
7		8	
9		10	





### Контрольные вопросы.

1. Назначение файла с расширением **.bsf**
2. Назначение файла с расширением **.rpt**
3. Назначение файла с расширением **.qsf**
4. Назначение файла с расширением **.qpf**
5. Назначение файла с расширением **.bdf**.
6. Назначение файла с расширением **vwf**
7. Пояснить создание файла верхнего уровня в «Quartus».
8. В чем заключается компиляция проекта?
9. Пояснить структуру библиотеки «Libraries».
10. Какую симуляцию требуется выбрать при создании проекта?

Вариант					Вариант				
<b>1</b>	1	9	10	6	<b>14</b>	4	2	6	10
<b>2</b>	2	9	1	5	<b>15</b>	1	9	7	10
<b>3</b>	3	10	2	9	<b>16</b>	2	10	8	6
<b>4</b>	4	2	3	1	<b>17</b>	3	8	9	2
<b>5</b>	5	3	4	8	<b>18</b>	4	1	10	7
<b>6</b>	6	4	5	10	<b>19</b>	5	2	10	9
<b>7</b>	7	5	6	10	<b>20</b>	6	3	2	10
<b>8</b>	8	6	7	10	<b>21</b>	7	4	3	1

9	9	7	8	3	22	8	5	4	2
10	10	8	9	4	23	9	6	1	4
11	7	9	10	1	24	10	7	2	9
12	8	10	1	2	25	10	8	4	6
13	3	10	2	6	26	2	10	3	8

## Практическая работа № 6

**Тема:** «Синтез последовательностных устройств»

**Цель работы:** Научиться составлять схемы последовательностных устройств, выполнять компиляцию и устранять обнаруженные ошибки

**Задача.** Проверить работу RS-триггера, получить временные диаграммы работы схемы, проанализировать результаты.

### Порядок выполнения работы

#### 1. Загрузить Quartus

Все практические и лабораторные работы программы моделирования электронных схем будем создавать в папке Project на Рабочем столе.

#### 2. Создать проект. Для этого выполнить следующие действия:

- выбрать пункт File
- выбрать New Project Wizard
- в появившемся окне нажимаем Next.
- в окне New Project Wizard: Directory, Name, Top-Level Entity (page 1 of 5) в строке What is the working directory for this project, щелкнуть кнопку с тремя точками и выбрать место расположения создаваемого проекта – Рабочий стол.
- создаем папку Project (для всех практических и лабораторных работ), а в ней папку Prakt3 и папку Prakt3 открыть и ещё раз нажать кнопку открыть.
- во второй строке записываем имя проекта Prakt3.
- нажимаем Next

Окно New Project Wizard: Add, Files (page 2 of 5) предлагает импортировать в проект файлы, но т.к. их пока нет, нажимаем Next.

Окно New Project Wizard: Family & device you want to target for Compilation предлагает выбрать микросхему. Выбираем в строке Family тип микросхемы **Cyclone** и в строке Available device тип микросхемы **EP1C12Q240C7**. Нажимаем Next.

Окно New Project Wizard: EDA Tool Setting (page 4 of 5) предлагает выбрать встроенные программы для синтеза проекта, симуляции. Нажимаем Next.

Окно New Project Wizard: Summary (page 5 of 5) показывает все исходные настройки для нашего проекта. Нажимаем Finish. Мы создали проект, пока пустой.

В окне навигатора проекта Project Navigator на вкладке иерархия проекта Hierarchy появился проект Prakt3 (с расширением .gpf).

#### 3. Синтез схемы.

– Создание файла верхнего уровня. Для этого необходимо выполнить действия: «File», «New», и в окне «New» выбрать тип файла Block Diagram / Schematic File

Нажимаем «OK», щелкаем «Save as», записываем имя Prakt3 (файл сохраняется с расширением .bdf), сохраняем, ставим внизу окна «Add file current project» галочку для добавления файла к проекту.

– Создание схемы.

Рисуем схему RS–триггера.

Из подгруппы «primitives», «logic» выбираем: двухвходовой элемент «ИЛИ-НЕ» (NOR2) – 2шт.

Из подгруппы «primitives», «buffer» выбираем буферный элемент (Icell) – 1шт.

Из подгруппы «primitives», «pin» выбираем: входы INPUT – 2шт., выходы OUTPUT – 2шт.

Для помещения элемента на рабочее поле необходимо выбрать элемент из списка Libraries в окне Symbol или записать его имя в поле Name, нажать «ОК» и разместить в нужном месте. Установка «галочки» в поле Repeat-insert позволит многократно копировать выбранный элемент.

Двойной щелчок левой кнопки на выбранном элементе открывает окно свойства элемента Properties, в котором в частности можно переименовать выходы элементов.

Далее рисуем схему RS – триггера, подсоединяя входы и выходы. Получаем схему, изображенную на рисунке 1.

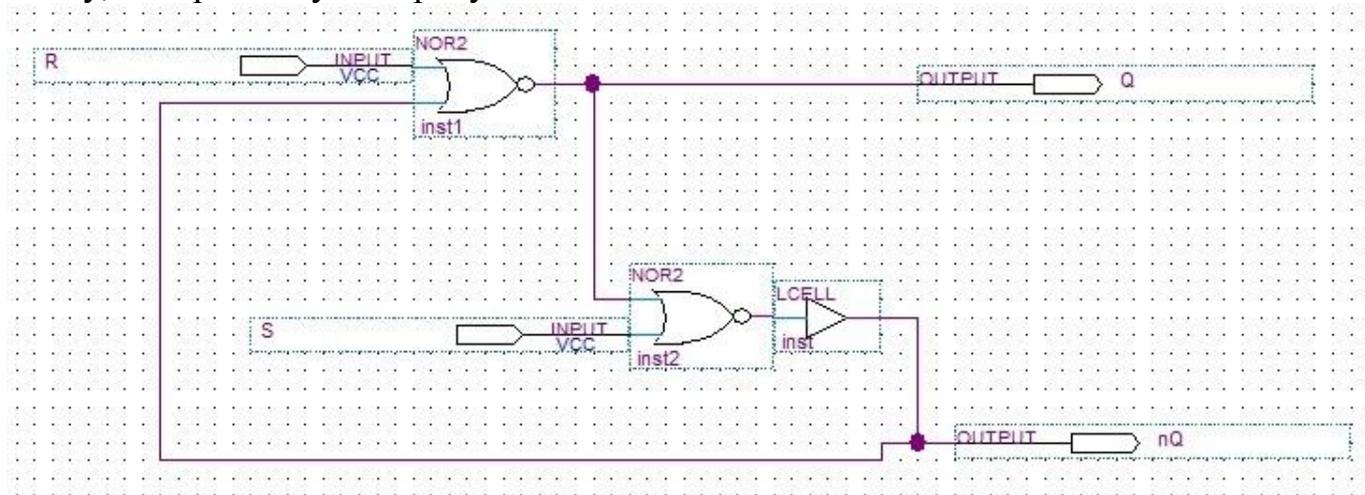


Рисунок 1. Схема RS–триггера  
Построенную схему сохранить.

### 3. Компиляция схемы

Компиляция проекта состоит из трех частей: синтез, разводка и ассемблирование. Мы рассмотрим только первую часть – синтез. Синтез осуществляется кнопкой  – «Start Analysis & Synthesis». В рабочем окне происходит процесс компиляции. В появившемся окне видим, что процесс компиляции прошел успешно – «Analysis & Synthesis was successful». Щелкнем «ОК». Появилось некое суммарное окно, характеризующее проект, закроем его.

Выбираем пункт «Processing», «Simulator Tool» в результате появляется окно симуляции «Simulator Tool».

В верхней строке «Simulation mode» выбираем функциональную симуляцию – «Functional». И после каждого синтеза необходимо создавать список соединений – «Netlist» нажатием кнопки «Generate Functional Simulation Netlist». Сообщение – «Functional Simulation Netlist Generate was successful» говорит о том, что генерация «Netlist» выполнена успешно. Щелкнем «ОК».

Процесс построения схемы двухвходового мультиплексора завершен. В процессе компиляции ошибок не обнаружено.

**Задание.** Создать проект для анализа работы триггеров, подключить входы и выход триггера, выполнить компиляцию схемы.

№ вар	1	2	3	4	5	6	7	8

Тип триггера	D (dff)	RS (srffe)	D (dff)	D (dff)	D (dff)	T (tffe)	D (dff)	D (dff)
	JK (jkff)	T (tffe)	JK (jkffe)	JK (jkffe)	RS (srff)	JK (jkffe)	RS (srff)	JK (jkff)
<b>№ вар</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>
Тип триггера	RS (srff)	D (dff)	T (tffe)	D (dff)	D (dff)	T (tff)	D (dff)	RS (srff)
	T (tff)	RS (srff)	JK (jkffe)	JK (jkff)	RS (srffe)	JK (jkff)	JK (jkffe)	T (tff)
<b>№ вар</b>	<b>17</b>	<b>18</b>	<b>19</b>	<b>20</b>	<b>21</b>	<b>22</b>	<b>23</b>	<b>24</b>
Тип триггера	D (dff)	D (dff)	RS (srffe)	T (tff)	D (dff)	D (dff)	T (tffe)	RS (srff)
	RS (srff)	JK (jkff)	T (tffe)	JK (jkff)	JK (jkffe)	RS (srff)	JK (jkffe)	T (tff)

### Контрольные вопросы.

#### 1. Пояснить назначение элементов

- and12
- and2
- and3
- and4
- and6
- and8

#### 2. Пояснить назначение элементов

- nand12
- nand2
- nand3
- nand4
- nand6
- nand8

#### 3. Пояснить назначение элементов

- nor12
- nor2
- nor3
- nor4
- nor6
- nor8

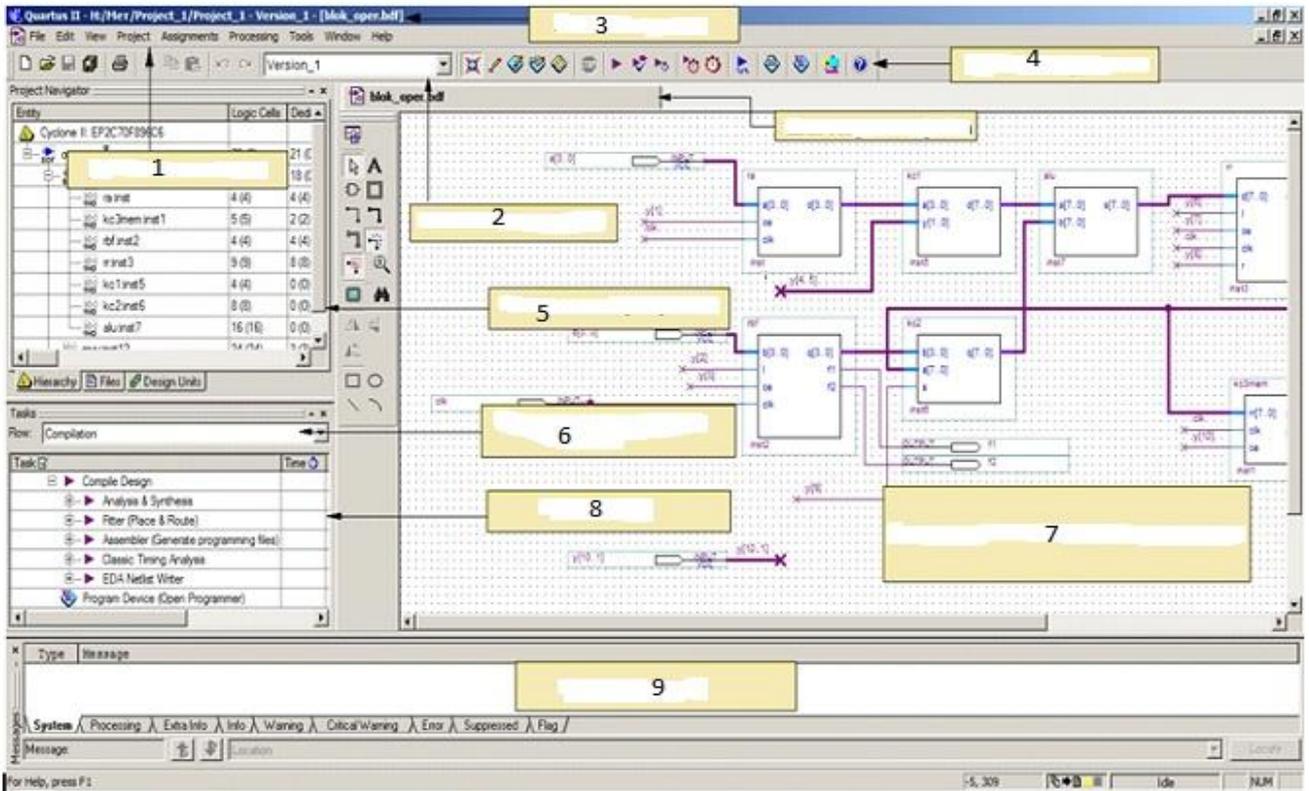
#### 4. Пояснить назначение элементов

- not
- or12
- or2
- or3
- or4

#### 5. Пояснить назначение элементов

- or6
- or8
- xnor
- xor

#### 6. Пояснить назначение элементов интерфейса Quartus



Вариант					Вариант				
1	1	2	3	6	14	1	2	3	6
2	4	5	1	6	15	4	5	1	6
3	2	3	4	6	16	2	3	4	6
4	5	1	2	6	17	5	1	2	6
5	3	4	5	6	18	3	4	5	6
6	1	2	3	6	19	1	2	3	6
7	4	5	1	6	20	4	5	1	6
8	2	3	4	6	21	2	3	4	6
9	5	1	2	6	22	5	1	2	6
10	3	4	5	6	23	3	4	5	6
11	1	2	3	6	24	1	2	3	6
12	4	5	1	6	25	4	5	1	6
13	2	3	4	6	26	2	3	4	6

#### 4. Информационное обеспечение обучения

##### Перечень рекомендуемых учебных изданий, Интернет-ресурсов, дополнительной литературы

###### Основные источники:

1. Гусева Е.Н. «Имитационное моделирование экономических процессов в среде *Arena*»: учебно-методическое пособие. – М.: издательство ФЛИНТА, 2017 г.– 132 с.
2. Дьяконов В. «MATLAB 6/6.1/6.5 + Simulink 4/5. Основы применения. Полное руководство пользователя». М.: Солон-Пресс, 2018г.– 198 с.
3. Дьяконов В., Круглов В. «MATLAB. Анализ, идентификация и моделирование систем». Специальный справочник. – СПб.: Питер. 2017.
4. Комолов Д.А., Мьялк Р.А., Зобенко А.А. «Системы автоматизированного проектирования фирмы Altera MAX+plus II и Quartus II». – М.: Издательство: РадиоСофт, 2017г. – 361с.
5. Кудрявцев Е.М. «GPSS World. Основы имитационного моделирования различных систем». – М.: DMK Press, 2016. – 320 с.
6. Овечкин Г.В. Компьютерное моделирование [Текст]: учеб. для студентов учреждений среднего проф. образования / Г. В. Овечкин, П. В. Овечкин. - 2-е изд., стер. - Москва: Академия, 2017. - 224 с. - (Профессиональное образование. Информатика и вычислительная техника).
7. Петлина, Е. М. Компьютерное моделирование [Электронный ресурс] : учебное пособие для СПО / Е. М. Петлина. — Электрон. текстовые данные. — Саратов : Профобразование, 2019. — 131 с. — 978-5-4488-0250-8. — Режим доступа: <http://www.iprbookshop.ru/83270.html>
8. Советов Б.Я., Яковлев С.А. Моделирование систем: Практикум. – М.: Высшая школа, 2017. – 224 с.
9. Советов Б.Я., Яковлев С.А. «Моделирование систем»: Учебник для вузов. – М.: Высшая школа, 2017. – 320 с.
10. Стещенко В.Б. «ПЛИС фирмы ALTERA: элементная база, система проектирования и языки описания аппаратуры». – М.: Издательство: Додэка-XXI, 2017г. – 576с.

###### Дополнительные источники:

1. Армстронг Дж. Р. Моделирование цифровых систем. – М.: Мир, 2016. – 174 с.
2. Бусленко Н.П. Моделирование сложных систем. – М.: Наука, 2016. – 400 с.
3. Киндлер Е. Языки моделирования. – М.: Энергия, 2016. – 288 с.
4. Математическое моделирование: Методы, описания и исследования сложных систем / Под ред. А.А. Самарского. – М.: Наука, 2016. – 128 с.
5. Никонов, О. И. Математическое моделирование и методы принятия решений: учебное пособие для СПО / О. И. Никонов, С. В. Кругликов, М. А. Медведева; под редакцией А. А. Астафьева. — 2-е изд. — Саратов, Екатеринбург: Профобразование, Уральский федеральный университет, 2019. — 99 с. — Текст: электронный // Электронно-библиотечная система IPR BOOKS : [сайт]. — URL: <http://www.iprbookshop.ru/87825.html>— Режим доступа: для авторизир. Пользователей.
6. Шрайбер Т.Дж. Моделирование на GPSS. – М.: Машиностроение, 2017. – 592 с.

###### Интернет- ресурсы:

1. <http://window.edu.ru/window> - Образование в области техники и технологий
2. [http://www.agtu.ru/e\\_proekt](http://www.agtu.ru/e_proekt) - Информационно-методический центр
3. <http://www.razym.ru/index>. - Электронная библиотека "Razym.ru"
4. <http://www.electronicworkbench.com> – Моделирование электронных схем
5. <http://www.gpss.ru> – Среда моделирования «GPSS Word».

Департамент внутренней и кадровой политики Белгородской области  
Областное государственное автономное  
профессиональное образовательное учреждение  
«Белгородский индустриальный колледж»

Группа \_\_\_\_\_

**ЖУРНАЛ ОТЧЕТОВ**  
по выполнению практических работ  
учебной дисциплины  
**ЕН.02 Компьютерное моделирование**

по специальности  
**11.02.10 Радиосвязь, радиовещание и телевидение**  
(углубленной подготовки)

ВЫПОЛНИЛ \_\_\_\_\_ / \_\_\_\_\_ /

ПРИНЯЛ \_\_\_\_\_ / Феоктистова В.Н. /

Белгород 2019 г.