

# МОДЕЛИРОВАНИЕ СИСТЕМ

## ИНСТРУМЕНТАЛЬНЫЕ СРЕДСТВА GPSS WORLD

.....

- *Соответствие государственному образовательному стандарту*
- *Основы построения и принципы функционирования русифицированной системы моделирования GPSS World*
- *Методы построения моделей с применением инструментальных средств GPSS World*
- *Описание и использование языка Plus для автоматизации цикла исследований*
- *Приемы разработки, отладки и эксплуатации моделей*
- *Оригинальные авторские примеры и модели из различных предметных областей*

**Василий Боев**

**МОДЕЛИРОВАНИЕ  
СИСТЕМ  
ИНСТРУМЕНТАЛЬНЫЕ СРЕДСТВА  
GPSS WORLD**

Санкт-Петербург

«БХВ-Петербург»

2004

УДК 681.3.06  
ББК 32.973.26-018.2  
Б72

**Боев В. Д.**

Б72 Моделирование систем. Инструментальные средства GPSS World:  
Учеб. пособие. — СПб.: БХВ-Петербург, 2004. — 368 с.: ил.

ISBN 5-94157-515-7

В учебном пособии рассматриваются основы построения и принципы функционирования общецелевой системы моделирования GPSS World. На многочисленных примерах раскрываются методы построения имитационных моделей с применением инструментальных средств GPSS World. Значительное внимание уделяется особенностям использования языка GPSS и языка Plus для моделирования сложных систем. Описываются методы проведения экспериментов с моделями и принятия по их результатам решений.

В приложениях в систематизированном виде приводятся дополнительные справочные материалы, необходимые для построения блок-диаграмм, написания и отладки программ моделей на современных ПЭВМ.

*Учебное пособие предназначено для студентов,  
курсантов военных учебных заведений, научных работников,  
а также всех тех, кто занимается вопросами  
моделирования систем различного назначения  
и кого интересуют современные информационные технологии*

УДК 681.3.06  
ББК 32.973.26-018.2

### Группа подготовки издания:

Главный редактор	<i>Екатерина Кондукова</i>
Зам. главного редактора	<i>Людмила Еремеевская</i>
Зав. редакцией	<i>Григорий Добин</i>
Компьютерная верстка	<i>Дмитрия Боева</i>
Корректор	<i>Зинаида Дмитриева</i>
Дизайн обложки	<i>Игоря Цырульникова</i>
Зав. производством	<i>Николай Тверских</i>

#### Рецензенты:

*К. В. Кузунжисев, заведующий кафедрой «Информационных технологий и систем» Ульяновского государственного университета, доктор технических наук, профессор; А. П. Бочков, профессор кафедры прикладной математики Санкт-Петербургского государственного политехнического университета, доктор технических наук, доцент; В. А. Кежаев, начальник кафедры управления войсками Военного артиллерийского университета, доктор технических наук, профессор*

Линейный ИД № 02429 от 24.07.00. Подписано в печать 01.06.04.

Формат 70×100<sup>1</sup>/<sub>16</sub>. Печать офсетная. Усл. печ. л. 29,67.

Тираж 3000 экз. Заказ №

"БХВ-Петербург", 190005, Санкт-Петербург, Измайловский пр., 29.

Гигиеническое заключение на продукцию, товар № 77.99.02.953.Д.001537.03.02  
от 13.03.2002 г. выдано Департаментом ГСЭН Минздрава России.

Отпечатано с готовых диапозитивов  
в ГУП «Типография «Наука»  
199034, Санкт-Петербург, 9 линия, 12.

# Содержание

Введение .....	1
1. Основы построения языка имитационного моделирования .....	4
1.1. Объекты системы моделирования GPSS World .....	4
1.2. Объекты языка имитационного моделирования .....	5
1.3. Порядок представления моделей в виде блок-диаграмм .....	11
1.4. Структура модели на языке GPSS .....	15
1.5. Объекты вычислительной категории .....	19
1.5.1. Константы .....	19
1.5.2. Системные числовые атрибуты .....	19
1.5.3. Арифметические, условные и логические операторы .....	21
1.5.4. Библиотечные математические функции .....	22
1.5.5. Библиотечные генераторы случайных чисел .....	22
1.5.6. Переменные пользователя .....	26
1.5.7. Выражения в операторах GPSS .....	27
1.5.8. Сохраняемые ячейки .....	27
1.5.9. Матрицы сохраняемых ячеек .....	29
1.5.10. Арифметические переменные и арифметические выражения .....	32
1.5.11. Булевы переменные .....	34
2. Построение моделей с одноканальными устройствами, функционирующими в режиме занятия и освобождения устройства .....	37
2.1. Организация поступления транзактов в модель и удаления транзактов из нее .....	38
2.1.1. Поступление транзактов в модель .....	38
2.1.2. Удаление транзактов из модели и завершение моделирования .....	41
2.2. Изменение значений параметров транзактов .....	43
2.2.1. Блок ASSIGN .....	43
2.2.2. Блок PLUS .....	44
2.2.3. Блок INDEX .....	44
2.2.4. Блок MARK .....	45
2.3. Занятие одноканального устройства и его освобождение .....	45
2.4. Имитация обслуживания посредством задержки во времени .....	47

2.5. Проверка состояния одноканального устройства . . . . .	49
2.5.1. Проверка состояния одноканального устройства блоком GATE . . . . .	49
2.5.2. Проверка состояния булевой переменной и блоком TEST . . . . .	50
2.6. Методы сбора статистики в имитационной модели . . . . .	53
2.6.1. Регистратор очереди . . . . .	53
2.6.2. Статистические таблицы . . . . .	56
2.7. Методы изменения маршрутов движения транзактов в модели . . . . .	57
2.7.1. Блок TRANSFER . . . . .	58
2.7.1.1. Режим безусловной передачи . . . . .	58
2.7.1.2. Режим статистической передачи . . . . .	59
2.7.1.3. Режим BOTH . . . . .	60
2.7.1.4. Режим ALL . . . . .	61
2.7.1.5. Режим PICK . . . . .	63
2.7.1.6. Функциональный режим . . . . .	65
2.7.1.7. Параметрический режим . . . . .	65
2.7.1.8. Подпрограммный режим . . . . .	66
2.7.1.9. Одновременный режим . . . . .	66
2.7.2. Блок LOOP . . . . .	67
2.7.3. Блок DISPLACE . . . . .	70
2.8. Примеры построения моделей с одноканальным устройством обслуживания . . . . .	74
3. Разработка и эксплуатация моделей в GPSS World . . . . .	80
3.1. Создание объекта «Модель» . . . . .	80
3.1.1. Запуск GPSS World . . . . .	80
3.1.2. Ввод операторов модели . . . . .	82
3.2. Создание объекта «Процесс моделирования» . . . . .	87
3.3. Команды GPSS World . . . . .	91
3.3.1. Интерактивные операторы . . . . .	91
3.3.2. Команды определения объектов GPSS . . . . .	92
3.3.3. Команды управления процессом моделирования . . . . .	92
3.3.3.1. Команда CLEAR . . . . .	93
3.3.3.2. Команда CONDUCT . . . . .	93
3.3.3.3. Команда CONTINUE . . . . .	94
3.3.3.4. Команда EXIT . . . . .	94
3.3.3.5. Команда HALT . . . . .	95
3.3.3.6. Команда INCLUDE . . . . .	95
3.3.3.7. Команда REPORT . . . . .	95
3.3.3.8. Команда RESET . . . . .	96
3.3.3.9. Команда SHOW . . . . .	97
3.3.3.10. Команда STEP . . . . .	98
3.3.3.11. Команда STOP . . . . .	98

3.4. Окна GPSS World .....	99
3.5. Отладка моделей .....	100
3.6. Отчеты .....	111
<b>4. Построение моделей систем с одноканальными устройствами, функционирующими в режимах прерывания и недоступности .....</b>	<b>115</b>
4.1. Прерывание одноканального устройства .....	115
4.1.1. Прерывание в приоритетном режиме .....	116
4.1.2. Прерывание в режиме «захвата» .....	121
4.1.3. Проверка состояния одноканального устройства, функционирующего в приоритетном режиме .....	126
4.2. Недоступность одноканального устройства .....	132
4.2.1. Перевод в недоступное состояние и восстановление доступности .....	132
4.2.2. Проверка состояний недоступности и доступности одноканального устройства .....	134
4.3. Сокращение машинного времени и изменение дисциплин обслуживания методом применения списков пользователя .....	136
4.3.1. Ввод транзактов в список пользователя в безусловном режиме .....	136
4.3.2. Вывод транзактов из списка пользователя .....	138
4.3.3. Ввод транзактов в список пользователя в условном режиме .....	144
4.4. Примеры построения моделей систем с одноканальными устройствами, функционирующими в различных режимах .....	146
<b>5. Построение моделей систем с многоканальными устройствами и переключателями .....</b>	<b>158</b>
5.1. Занятие многоканального устройства и его освобождение .....	158
5.2. Перевод многоканального устройства в недоступное состояние и восстановление доступности .....	160
5.3. Проверка состояния многоканального устройства .....	163
5.3.1. Проверка состояния блоком GATE .....	163
5.3.2. Проверка состояния булевой переменной и блоком TEST .....	165
5.4. Моделирование неисправностей многоканальных устройств .....	169
5.5. Моделирование переключателей .....	178
5.5.1. Проверка состояния логического ключа блоком GATE .....	178
5.5.2. Проверка состояния логического ключа блоком TEST и булевой переменной .....	180
5.6. Примеры построения моделей систем с многоканальным устройством обслуживания .....	183

6. Методы применения функций в имитационных моделях ..	188
6.1. Генераторы псевдослучайных чисел .....	188
6.2. Типы функций .....	189
6.3. Методы розыгрыша псевдослучайных чисел для дискретных равномерных распределений .....	190
6.4. Методы розыгрыша псевдослучайных чисел для дискретных неравномерных распределений .....	193
6.5. Процедура розыгрыша псевдослучайных чисел при непрерывном равномерном распределении .....	196
6.6. Методы розыгрыша псевдослучайных чисел для непрерывных неравномерных распределений .....	198
6.6.1. Моделирование экспоненциального распределения .....	199
6.6.2. Моделирование нормального распределения .....	201
6.7. Функции типа E, L и M .....	204
6.8. Примеры построения моделей с использованием функций ...	205
7. Особенности применения GPSS для моделирования сложных систем .....	212
7.1. Применение в моделях блоков создания копий транзактов ...	212
7.2. Организация в моделях синхронизации движения транзактов	214
7.2.1. Блок ASSEMBLE .....	214
7.2.2. Блок GATHER .....	217
7.2.3. Блок MATCH .....	219
7.3. Уменьшение числа объектов в модели методом косвенной адресации .....	222
7.4. Обработка одновременных событий .....	227
7.5. Методы построения моделей с рациональными вариантами обслуживания .....	234
7.5.1. Блок COUNT .....	234
7.5.2. Блок SELECT .....	236
7.6. Организация прогонов при изменении версий модели .....	248
7.7. Применение текстовых объектов и потоков данных .....	255
7.7.1. Блок OPEN .....	255
7.7.2. Блок CLOSE .....	256
7.7.3. Блок READ .....	257
7.7.4. Блок WRITE .....	258
7.7.5. Блок SEEK .....	259
7.7.6. Пример применения текстовых объектов и потоков данных .....	259

7.8. Непрерывное моделирование .....	266
7.8.1. Команда INTEGRATE .....	266
7.8.2. Блок INTEGRATION .....	267
7.8.3. Фазы моделирования и ошибки интегрирования .....	268
7.8.4. Пример непрерывного моделирования .....	268
8. Язык PLUS .....	272
8.1. Алфавит .....	272
8.2. Имена .....	272
8.3. Выражения .....	273
8.4. Plus-операторы .....	275
8.5. Библиотека процедур .....	277
8.6. Подготовка текстовых объектов .....	282
8.7. Реализация собственного метода интегрирования .....	286
8.8. Отладка Plus-процедур .....	288
9. Эксперименты с имитационными моделями .....	291
9.1. Основные понятия теории планирования экспериментов . . . .	291
9.1.1. Необходимость планирования экспериментов .....	291
9.1.2. Стратегическое планирование эксперимента .....	293
9.1.3. Тактическое планирование эксперимента .....	295
9.2. Дисперсионный анализ (отсеивающий эксперимент) .....	297
9.3. Эксперименты пользователя .....	310
9.4. Регрессионный анализ (оптимизирующий эксперимент) . . . .	318
Список литературы .....	322
Приложение 1 .....	324
Приложение 2 .....	332
Приложение 3 .....	336
Приложение 4 .....	342
Элементы стандартного отчета .....	342
1. Общая информация о результатах работы модели .....	342
2. Информация об именах .....	342
3. Информация о блоках .....	343
4. Информация об объектах типа «устройство» .....	344
5. Информация об объектах типа «очередь» .....	344
6. Информация об объектах типа «многоканальное устройство» .....	345



---

7. Информация о таблицах .....	345
8. Информация о списках пользователя .....	346
9. Информация о группах транзактов .....	346
10. Информация о числовых группах .....	347
11. Информация о логических переключателях .....	347
12. Информация о сохраняемых величинах (ячейках) .....	347
13. Информация о матрицах .....	347
14. Информация о списках текущих и будущих событий .....	348

## Введение

Моделирование — наиболее мощный универсальный метод исследования и оценки эффективности систем, поведение которых зависит от воздействия случайных факторов. Области применения методов имитации чрезвычайно широки и разнообразны. Однако по опыту научной работы и материалам диссертационных советов можно сделать вывод о том, что исследователи пока довольно редко используют в качестве инструментальных средств исследования системы моделирования, преимущества которых вполне очевидны [18]. Системы моделирования имеют специализированные средства, реализующие дополнительные возможности по организации модельных экспериментов на компьютере. Они также предоставляют возможность учитывать в моделях фактор времени, то есть строить динамические имитационные модели, что особенно важно для многих систем, в том числе и систем военного назначения.

Применение универсальных языков программирования при реализации имитационных моделей позволяет исследователю достигнуть гибкости при разработке, отладке и испытании модели. Однако языки моделирования, ориентированные на определенную предметную область, являются языками более высокого уровня, поэтому дают возможность с меньшими затратами создавать программы моделей для исследования сложных систем.

Специализированные языки моделирования делят на три группы, соответствующие видам имитации: для непрерывных, дискретных и комбинированных процессов. Для моделирования дискретных систем широкое распространение получил пакет моделирования дискретных систем (ПМДС) GPSS (General Purpose Simulation System — общецелевая система моделирования) [27]. Написанные с помощью языка GPSS модели получаются более адекватными исследуемой системе, чем построенные с использованием известной теории массового обслуживания. В моделях, написанных на языке GPSS, можно учесть большое количество факторов и отказаться от многих ограничений и допущений.

Еще одним языком имитационного моделирования дискретных систем является Simpas [2, 5, 19, 20], который, по мнению разработчиков [19], «по своим возможностям полностью перекрывает специализированный язык GPSS, обладая более высоким быстродействием и гибкостью». Система Simpas оформлена как модуль и потому является открытой по сравнению с GPSS/PC.

Это дает возможность исследователю разрабатывать свои процедуры и включать их в Simpas или подстраивать что-то из уже имеющегося в нем под свои нужды. При разработке моделей на GPSS/PC исследователь ограничен его возможностями, в то время как Simpas позволяет использовать возможности Object Pascal.

Однако следует отметить, что GPSS/PC и Simpas, а также GPSS/H [4, 14] предназначены для работы в операционной системе MS DOS. Поэтому имеются ограничения, которые в ряде случаев не позволяют осуществить разработку и эксплуатацию моделей сложных систем с требуемой степенью детализации.

Отмеченных недостатков практически не имеет новая общецелевая система моделирования GPSS World, разработанная компанией Minuteman (США) [22]. Эта система является развитием GPSS/PC [1, 3, 4, 21], но приобрела комбинированный характер, т. е. может моделировать как дискретные, так и непрерывные процессы. Эти возможности обеспечиваются как новыми объектами языка GPSS, так и включением в состав GPSS World языка Plus — языка программирования низкого уровня. Этот язык сделал GPSS World более открытой системой и позволяет взаимодействовать с другими приложениями, а также создавать пользователям свои библиотеки процедур. Язык Plus вместе с другими инструментальными средствами GPSS World позволил автоматизировать весь цикл исследований от разработки модели до выработки рекомендаций за счет новых функций планирования экспериментов и обработки статистики. И наконец, система моделирования GPSS World работает в операционной системе Windows и максимально ориентирована на использование современных технологий, обеспечивающих высокую интерактивность и визуальное представление информации.

Вместе с тем в настоящее время недостаточно учебного материала для изучения языков моделирования вообще и в частности GPSS World. Источники [22, 26] недоступны для широкого круга обучаемых. В [15] рассматривается Simulink, входящий в состав MatLab и предназначенный для создания моделей технических устройств. Предложение автора создавать посредством Simulink такие же модели, как и на GPSS, нельзя признать приемлемым. Известная система Agena хотя и выполняет функции, аналогичные функциям GPSS, по своим возможностям и гибкости уступает GPSS. Книги [17, 24], посвященные GPSS/PC и GPSS World, не восполняют имеющийся пробел, поскольку не раскрывают многие новые возможности GPSS World и связанные с ними методы разработки моделей. В названии [23] есть слово «технологии», однако технологии, т. е. методов построения моделей, — там также нет. Книга содержит описание средств GPSS World практически без показа на примерах, как с их помощью строить модели. В [25] дается лишь обзор инструментальных средств реализации моделей, среди которых GPSS World нет.

Настоящее учебное пособие соответствует дисциплине «Моделирование систем». В нем раскрываются методы построения моделей с применением инструментальных средств современной мощной системы моделирования GPSS World.

В первой главе излагаются основы построения и принципы функционирования языка имитационного моделирования. Во второй, четвертой, пятой, шестой и седьмой главах на многочисленных примерах, разработанных автором, раскрываются методы построения моделей процессов, которые формализуются в виде схем массового обслуживания. В третьей главе описываются приемы создания объектов GPSS World и эксплуатации моделей. В восьмой главе описан язык Plus и показано его применение, в том числе и для построения моделей непрерывных процессов, а девятая глава посвящена методам проведения различных видов экспериментов.

Автор выражает благодарность кандидату технических наук Девяткову Владимиру Васильевичу, директору ООО «Элина-Компьютер», руководителю работ по локализации GPSS World на русский язык [22, 26], встрече с которым на презентации GPSS World в Санкт-Петербурге и послужила окончательным толчком к написанию данной книги. Важным для автора является и отзыв Владимира Васильевича: «Я прочитал присланные Вами материалы по Вашей книге. Мне они понравились, чувствуется, что это не просто написано, а все проверено на множестве моделей. С терминологией я согласен. Надеюсь, что книга будет положительно воспринята».

Замыслу структуры учебного пособия, методологии изложения материала некоторых глав, а также моральной поддержке в период работы над книгой автор обязан первому постановщику дисциплин «Моделирование систем» и «Компьютерное моделирование» [16] в Военном университете связи кандидату технических наук, доценту Сыпченко Р. П. Спасибо, Руслан Павлович!

Постоянную помощь в подготовке рукописи автору оказывал Боев Д. В., который выполнил и компьютерную верстку книги, за что автор благодарен ему.

Учебную версию GPSS World бесплатно можно получить на портале [www.minutemansoftware.com/download](http://www.minutemansoftware.com/download), а русифицированную (русификация выполнена автором пособия) — у автора по адресу e-mail: [kirasir@rambler.ru](mailto:kirasir@rambler.ru).

# 1. Основы построения языка имитационного моделирования

## 1.1. Объекты системы моделирования GPSS World

В системе моделирования GPSS World различают четыре вида объектов (рис. 1.1.): модель, процесс моделирования, отчет и текстовый.

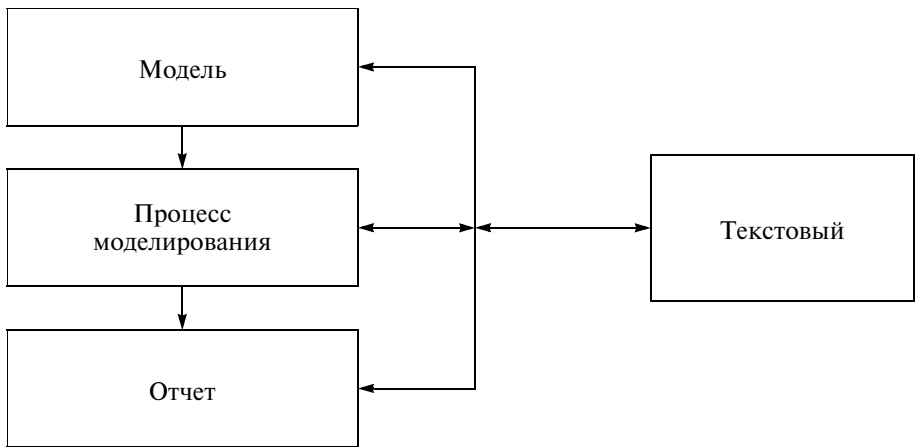


Рис. 1.1. Объекты GPSS World

Модель разрабатывается на языке GPSS и состоит из операторов, а объект «Модель» создается при помощи встроенного текстового редактора. Объект «Процесс моделирования» — это результат трансляции модели, получаемый после выполнения команды меню Create Simulation (Создать процесс моделирования). Далее процесс моделирования запускается с помощью команд GPSS. По завершении моделирования, как правило, автоматически создается объект «Отчет».

Текстовый объект (текстовый файл GPSS World) предназначен для упрощения разработки больших моделей и создания библиотеки исходных текстов, т. е. модель может быть разделена на наборы операторов, представляющие собой отдельные текстовые файлы, а затем объектом «Процесс моделирования» собрана из них. Объект «Процесс моделирования» может также создавать но-

вые текстовые файлы с фрагментами модели, результатами моделирования, а также считывать и записывать данные в текстовые файлы.

Модель, как уже отмечалось, разрабатывается на языке GPSS. Перейдем к его рассмотрению.

## 1.2. Объекты языка имитационного моделирования

Система GPSS World предназначена для имитационного моделирования систем с дискретными и непрерывными процессами. Языком моделирования в ней является язык GPSS, улучшенный встроенным языком программирования низкого уровня PLUS. Язык GPSS построен в предположении, что модель сложной системы можно представить совокупностью элементов и логических правил их взаимодействия в процессе функционирования моделируемой системы. Также предполагается, что для моделируемых систем можно выделить небольшой набор абстрактных элементов, называемых объектами. Причем набор логических правил ограничен и может быть описан небольшим числом стандартных операций. Комплекс программ, описывающих функционирование объектов и выполняющих логические операции, является основой для создания программной модели системы данного класса.

Кроме этого комплекса в составе системы GPSS World имеется специальная программа-планировщик, которая выполняет следующие функции:

- обеспечение продвижения по заданным разработчиком маршрутам динамических объектов, называемых транзактами;
- планирование событий, происходящих в модели, путем регистрации времени наступления каждого события и выполнения их в нарастающей временной последовательности;
- регистрация статистической информации о функционировании модели;
- продвижение модельного времени в процессе моделирования системы.

Чтобы обеспечить правильную последовательность обработки событий во времени, имеются системные часы, хранящие значения абсолютного модельного времени. Все отрезки времени описываются положительными числами.

Объекты в моделируемой системе предназначены для различных целей. Выбор объектов в моделируемой системе зависит от характеристик модели и, в некоторых случаях, от разработчика модели. Совершенно не обязательно, чтобы в одной модели участвовали все типы объектов. Необходимо лишь наличие блоков и транзактов, иначе модель работать не будет.

Объекты подразделяются на *7 категорий* и *15 типов*, которые представлены в табл. 1.1. Рассмотрим назначение объектов GPSS.

Динамическими объектами являются транзакты, которые создаются в определенных точках модели, продвигаются планировщиком через блоки, а затем уничтожаются. Транзакты являются аналогами единиц — потоков в реальной системе. Они могут представлять собой различные элементы даже в одной сис-

теме. Например, в модели боевых действий войск одни транзакты могут являться аналогами средств нападения противника, другие — зенитными управляемыми ракетами, третьи — вооружением и техникой войск, транспортными средствами, четвертые — донесениями. С каждым транзактом связаны параметры, которые используются для конкретных данных. В приведенных выше примерах параметрами транзактов могут быть: тип вооружения, принадлежность, техническое состояние, количество знаков в донесении и др. Каждый транзакт может иметь любое число параметров. Параметры нумеруются или им даются имена. Номера параметров и имена используются для ссылок на значения, присвоенные параметрам. Транзактам может присваиваться приоритет. Приоритет определяет предпочтение, которое получает транзакт, когда он и другие транзакты претендуют на один и тот же ресурс. Это может быть, например, категория срочности при передаче сообщений или очередность эвакуации и ремонта вооружения и техники в зависимости от их важности.

Таблица 1.1

### Объекты GPSS

Категории	Типы объектов
Динамическая	Транзакты
Операционная	Блоки
Аппаратная	Одноканальные устройства, памяти (многоканальные устройства), логические ключи
Вычислительная	Переменные, функции, генераторы случайных чисел
Статистическая	Очереди, таблицы
Запоминающая	Ячейки, матрицы ячеек
Группирующая	Числовые группы, группы транзактов, списки

*Объекты аппаратной категории* — это абстрактные элементы, на которые может быть декомпозирована реальная система. Воздействуя на эти объекты, транзакты могут изменять их состояние и влиять на движение других транзактов. К объектам этого типа относятся одноканальные устройства, памяти (многоканальные устройства) и логические ключи.

*Одноканальные устройства (ОКУ)* представляют собой оборудование, которое в любой момент времени может быть занято только одним транзактом. Например, один канал передачи данных, одноканальный ремонтный орган, один узел связи.

*Многоканальные устройства (МКУ)* предназначены для имитации оборудования, осуществляющего параллельную обработку. Они могут быть использованы одновременно несколькими транзактами. МКУ можно использовать в качестве аналога, например, многоканального ремонтного органа, нескольких каналов связи.

Ранее происходившие события в системе могут заблокировать, изменить движение транзактов и наступление последующих событий. Например, один канал связи вышел из строя и все последующие заявки на передачу сообщений должны быть направлены на исправные каналы связи. Для моделирования таких ситуаций введены *логические ключи*. Транзакт может устанавливать эти ключи в положение «включено» или «выключено». Впоследствии состояние ключей может быть проверено другими транзактами для определения пути их дальнейшего следования.

*Операционные объекты*, т. е. блоки, задают логику функционирования модели системы и определяют пути движения транзактов между объектами аппаратной категории. В блоках могут происходить события четырех основных типов:

- 1) создание или уничтожение транзактов;
- 2) изменение числового атрибута объекта;
- 3) задержка транзакта на определенный период времени;
- 4) изменение маршрута движения транзакта в модели.

Версия GPSS, реализованная в системе GPSS World, содержит 53 типа блоков. В зависимости от назначения блоки подразделяются на несколько групп.

1. Блоки, осуществляющие модификацию атрибутов транзактов:
  - а) генерирование и уничтожение транзактов GENERATE, SPLIT, TERMINATE, ASSEMBLE;
  - б) временная задержка ADVANCE;
  - в) синхронизация движения двух MATCH и нескольких GATHER транзактов;
  - г) изменение параметров транзактов ASSIGN, INDEX, MARK, PLUS;
  - д) изменение приоритета транзакта PRIORITY.
2. Блоки, изменяющие последовательность движения транзактов (блоки передачи управления): DISPLACE, TRANSFER, LOOP, TEST, GATE.
3. Блоки, связанные с группирующей категорией: ADOPT, ALTER, EXAMINE, JOIN, REMOVE, SCAN.
4. Блоки, описывающие объекты аппаратной категории:
  - а) одноканальные устройства (технические средства) SEIZE, RELEASE, PRE-EMPT, RETURN, FUNAVAIL, FAVAIL;
  - б) многоканальные устройства (памяти) ENTER, LEAVE, SAVAIL, SUNAVAIL;
  - в) ключи (логические переключатели) LOGIC.
5. Блоки, сохраняющие необходимые значения для дальнейшего использования: SAVEVALUE, MSAVEVALUE.
6. Блоки, обеспечивающие получение статистических результатов:
  - а) очереди QUEUE, DEPART;
  - б) таблицы TABULATE.
7. Блоки для организации списка пользователя: LINK, UNLINK.
8. Блоки для организации ввода-вывода:



- а) открытие/закрытие файла: OPEN/CLOSE;
  - б) считывание/запись в файл: READ/WRITE;
  - в) установка позиции текущей строки: SEEK.
9. Специальные блоки: BUFFER, COUNT, EXECUTE, INTEGRATION, SELECT, TRACE, UNTRACE.

*Вычислительная категория* служит для описания таких ситуаций в процессе моделирования, когда связи между компонентами моделируемой системы наиболее просто и компактно выражаются в виде математических (аналитических и логических) соотношений. Для этих целей в качестве объектов вычислительной категории введены арифметические и булевы переменные и функции.

*Переменные* представляют собой сложные выражения, которые включают константы, системные числовые атрибуты (СЧА), библиотечные арифметические функции, арифметические и логические операции.

Выражения могут применяться в переменных и операторах GPSS. При применении в переменных выражения определяются командами GPSS. При применении в операторах GPSS выражения определяются как часть языка PLUS.

Каждому объекту соответствуют атрибуты, описывающие его состояние в данный момент времени. Они доступны для использования в течение всего процесса моделирования и называются *системными числовыми атрибутами* (СЧА). Например, объект вычислительной категории — генератор случайных чисел имеет СЧА  $RN_n$  — число, вычисляемое генератором равномерно распределенных случайных чисел номер  $n$ ; у объекта динамической категории — транзакта СЧА: PR — приоритет обрабатываемого в данный момент транзакта;  $P_i$  — значение  $i$ -го параметра активного транзакта и др. Всего в GPSS World имеется свыше 50 СЧА.

*Булевы переменные* позволяют пользователю проверять в одном блоке GPSS одновременно несколько условий, исходя из состояния или значения этих условий и их атрибутов, т. е. в данном блоке производится обращение к булевой переменной, выражение которой содержит в себе проверку нескольких условий. Булевы переменные могут быть представлены комбинациями стандартных числовых атрибутов, связанных между собой посредством булевых операторов, включая и другие переменные. Булевы переменные определяются так же, как и арифметические, но вместо арифметических операций проверяются различные логические условия.

С помощью функций пользователь может производить вычисления непрерывных или дискретных функциональных зависимостей между аргументом функции (независимая величина) и зависимым значением функции. Все функции в GPSS задаются табличным способом с помощью команд описания функций. Как и переменные, функции не связаны с определенными блоками.

Кроме библиотечных арифметических функций GPSS World имеет 24 *встроенных генератора случайных чисел*.

*Объекты запоминаящей категории* обеспечивают обращения к сохраняемым значениям. Ячейки сохраняемых величин и матрицы ячеек сохраняемых величин используются для сохранения некоторой числовой информации. Например, значение, занесенное в ячейку, может представлять собой коэффициент использования устройства в какой-то момент времени. Любой активный транзакт может произвести запись информации в эти объекты. Впоследствии записанную в эти объекты информацию может считать любой транзакт. Матрицы могут иметь до шести измерений.

К *статистическим объектам* относятся очереди и таблицы. В любой системе движение потока транзактов может быть задержано из-за недоступности устройств. Например, требуемое ОКУ (канал связи) может быть уже занято или МКУ (каналы связи), в которые нужно войти, уже заполнены. В этом случае задержанные транзакты ставятся в *очередь* — еще один тип объектов GPSS. Учет этих очередей составляет одну из основных функций планировщика. Планировщик автоматически накапливает определенную статистику относительно устройств и очередей. Кроме этого пользователь может собирать дополнительную статистическую информацию, указав специальные точки в модели.

Для облегчения табулирования статистической информации в GPSS предусмотрен специальный объект — *таблица*. Таблицы используются для получения выборочных распределений некоторых случайных величин. Таблица состоит из частотных классов (диапазонов значений), куда заносится число попаданий конкретного числового атрибута в каждый, тот или иной, частотный класс. Для каждой таблицы вычисляется также математическое ожидание и среднеквадратическое отклонение. Эта статистика является стандартной для всех таблиц. В конце эксперимента с моделью результаты, содержащиеся в таблицах, могут выводиться либо на дисплей, либо на печать.

К *группирующей категории* относятся три типа объектов: числовая группа, группа транзактов и списки.

В некоторых моделях транзакты представляют собой объекты, характеризующиеся общими СЧА, которые обычно представляются как параметры транзакты и меняются при прохождении транзактов через блок ASSIGN. Однако это дает ограниченный доступ к атрибутам, а иногда приходится одновременно менять некоторые атрибуты всех транзактов данного множества. Объекты такого типа предоставляют возможность пользователю обращаться к атрибутам транзактов данной *группы*. Они работают в одном из режимов: в режиме транзакта или числовом. Режим определяется первым обращением к определенной группе. После того как он установлен, необходимо его поддерживать. В режиме транзакта величины, поступающие в группу, представляются номерами по очереди транзактов данной группы. В другом режиме поступающие величины представляются группой числовых значений.

При моделировании транзакты хранятся в списках. Существует пять видов списков, только в одном из которых в любой момент времени может нахо-

даться транзакт: 1) текущих событий; 2) будущих событий; 3) задержки ОКУ или МКУ; 4) отложенных прерываний ОКУ; 5) пользователя.

В *списке текущих событий* (СТС) находятся транзакты, соответствующие событиям, время наступления которых меньше (транзакты, которые должны были начать двигаться в некоторый момент в прошлом, но были заблокированы) или равно текущему времени. Транзакты из СТС готовы к входу в блоки и должны войти в них до очередного изменения модельного времени. В СТС транзакты расположены в порядке убывания приоритета. Транзакты с одинаковыми приоритетами располагаются в порядке поступления их в список.

*Список будущих событий* (СБС) содержит транзакты, соответствующие событиям, время наступления которых больше текущего времени, т. е. событиям, которые должны произойти в будущем. Такие транзакты размещаются в списке строго в порядке возрастания времени начала движения. Приоритеты не влияют на порядок движения в этом списке.

В *список повторных попыток* помещаются транзакты, для которых не выполнены условия входа в следующий блок. Эти условия проверяются при попытке входа транзакта в блоки GATE, TEST, TRANSFER ALL и TRANSFER BOTH. Транзакты из списка повторных попыток ожидают изменения СЧА. Когда СЧА изменяется, транзакт активируется, проверяются условия входа его в следующий блок. Если условия выполняются, транзакт входит в следующий блок и автоматически исключается из списка повторных попыток.

Одноканальное устройство имеет:

- *список отложенных прерываний* — список транзактов, ожидающих занятия ОКУ по приоритету;
- *список прерываний* — список транзактов, обслуживание которых данным ОКУ было прервано;
- *список задержки* — список транзактов, ожидающих занятия ОКУ в порядке приоритета;
- *список повторных попыток* — список транзактов, ожидающих изменения состояния ОКУ.

Многоканальное устройство имеет:

- *список задержки* — список транзактов в порядке приоритета, ожидающих возможность занять освободившиеся каналы МКУ;
- *список повторных попыток* — список транзактов, ожидающих изменения состояния МКУ.

*Список синхронизируемых* транзактов имеют блоки ASSEMBLE, GATHER, MATH. Он содержит транзакты, находящиеся в данный момент времени в состоянии сравнения.

*Список пользователя* содержит транзакты, удаленные пользователем из списка текущих событий и помещенные в список пользователя как временно неактивные. Списки пользователя используются для организации очередей

с дисциплинами, отличными от дисциплины «первым пришел — первым обслужен».

Более подробно использование списков транзактов будет изложено в дальнейшем при рассмотрении методов построения моделей с использованием блоков GPSS World.

#### **ЗАМЕЧАНИЕ**

Термин CHAIN в [27, 17, 23] переведен с английского как «цепь». В [1, 3, 17, 22, 26], а также в данной книге вместо «цепь» используется «список». Поясним почему.

Представьте себе, например, что у вас есть золотая цепочка, состоящая из какого-то числа звеньев. За все время пользования этой цепочкой у вас даже и мысли не возникнет о том, чтобы каждые там день или два, а то и чаще разбирать ее и менять звенья местами. Да и при покупке цепочки вы не поставите жесткое условие: цепочка должна быть обязательно разборной. Скорее наоборот.

Проанализируем теперь процесс моделирования. Предположим, что перед ОКУ находится несколько транзактов, ожидающих его освобождения. В любой момент времени поступают новые транзакты, которые в общем случае вызывают перестроение уже ожидающих транзактов. То же самое может происходить и после занятия каким-либо транзактом освободившегося ОКУ. Кроме того, один или более из ожидающих транзактов могут удаляться по каким-либо причинам, не заняв ОКУ. И это повторяется многократно. Отсюда транзакты, находящиеся перед ОКУ (да и в любом другом случае, когда возникает нехватка ресурсов), ассоциируются, по нашему мнению, со списком, а не с цепью.

После того как система, модель которой нужно разработать, формализована, ее нужно описать на языке GPSS, используя блоки, которые выполняют соответствующие операции в модели.

### **1.3. Порядок представления моделей в виде блок-диаграмм**

Модели систем на GPSS могут быть первично описаны в виде блок-диаграмм. *Блок-диаграмма* представляет собой набор с характерным очертанием блоков, соединенных между собой линиями. Вид каждого из блоков стандартен (приложение 1). Модель строится следующим образом. Из допустимого множества блоков выбирают необходимые и далее выстраивают их в диаграмму для того, чтобы в процессе функционирования модели они как бы взаимодействовали друг с другом. Использование блоков при построении моделей зависит от логических схем работы реальных систем, моделируемых на ЭВМ. При этом отдельные элементы модели и модель в целом имеют достаточно различимое подобие. Оно может быть также усилено разработчиком за счет более продуманного на этапе разработки разделения исследуемого объекта и модели на сегменты.

Типичный вид блок-диаграммы на GPSS представлен на рис. 1.2.

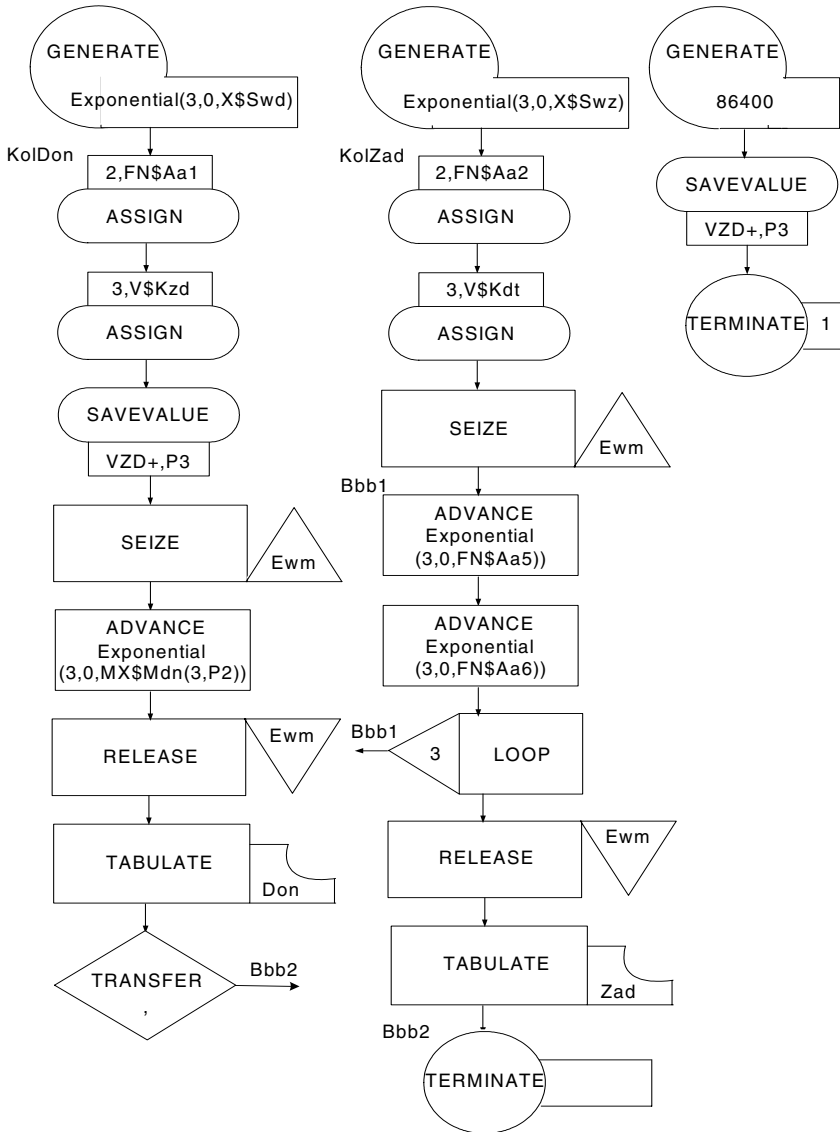


Рис. 1.2. Блок-диаграмма модели обработки информации

Конфигурация блок-диаграммы модели отражает направления, по которым происходит движение перемещающихся элементов — транзактов. Работа модели заключается в перемещении транзактов от блоков к блокам.

В самом начале моделирования в модели нет ни одного транзакта. В процессе моделирования транзакты входят в модель в определенные моменты времени в соответствии с теми логическими потребностями, которые возникают в мо-

делируемой системе. Подобным же образом транзакты покидают модель в определенные моменты времени в зависимости от специфики моделирования.

В общем случае в модели существует большое число транзактов, однако в определенный момент времени движется только один транзакт. Если транзакт начал движение, он перемещается от блока к блоку по пути, предписанному блок-диаграммой. Каждый блок можно рассматривать как некоторую точку, в которой происходит обращение к подпрограмме. В тот момент, когда транзакт входит в блок, на исполнение вызывается соответствующая подпрограмма, и далее транзакт пытается войти в следующий блок. Такое продвижение транзакта продолжается до тех пор, пока не произойдет одно из следующих событий:

- транзакт входит в блок, функцией которого является задержка транзакта на некоторое определенное модельное время;
- транзакт входит в блок, функцией которого является удаление транзакта из модели;
- транзакт пытается войти в следующий блок в соответствии с предписанной блок-диаграммой логикой, однако блок отказывается принять этот транзакт;
- транзакт остается в том блоке, в котором он в настоящее время находится, и будет повторять свои попытки войти в следующий блок.

Когда условия в модели изменятся, одна из попыток может оказаться успешной и транзакт сможет продолжить свое движение в модели.

Если возникло одно из перечисленных условий, транзакт остается на месте и начинается перемещение в модели другого транзакта. Выполнение модели заключается в последовательном обращении к подпрограммам, являющимся следствием входа в определенные блоки двигающихся транзактов. Проходя через блоки, каждый транзакт вносит свой вклад в содержимое *счетчиков блоков*. Значения этих счетчиков доступны разработчику через СЧА блоков:  $W$  — текущее содержимое блока и  $N$  — общее количество входов в блок.

Каждое продвижение транзакта в модели является *событием*, которое должно произойти в определенный момент модельного времени. Для того, чтобы поддерживать правильную временную последовательность событий, планировщик имеет *таймер* модельного времени, который автоматически корректируется в соответствии с логикой, предписанной моделью.

Таймер GPSS имеет следующие особенности:

- регистрируются только положительные вещественные значения времени;
- единица модельного времени (масштаб) определяется разработчиком модели, который задает все временные интервалы в одних и тех же единицах;
- планировщик не анализирует состояние модели в каждый следующий момент модельного времени (отстоящий от текущего на единицу модельного времени), а продвигает таймер к моменту времени, когда должно произойти ближайшее следующее событие.

Значения таймера доступны разработчику через системные СЧА С1 (*относительное модельное время*) и АС1 (*абсолютное модельное время*).

Главной задачей, выполняемой планировщиком, является определение того, какой транзакт надо выбрать следующим для продвижения в модели, когда предыдущий транзакт прекратил свое продвижение. С этой целью планировщик рассматривает каждый транзакт как элемент некоторого списка.

Планировщик GPSS помещает транзакты в зависимости от условий в модели в тот или иной список, просматривает списки, выбирает следующий транзакт для обработки, корректирует таймер модельного времени после обработки всех транзактов в списке текущих событий.

Вернемся к рис. 1.2. На нем показаны три блока GENERATE, которые являются источниками транзактов. Такие блоки являются точками, в которых транзакты появляются в модели. Два блока TERMINATE являются конечными, т. е. блок-диаграмма на этих блоках завершается и из нее нет выхода. Эти блоки являются точками вывода транзактов из модели. Блок-диаграмма состоит из трех частей — сегментов. Один сегмент, состоящий из трех блоков, является независимым, а два других взаимосвязаны друг с другом. В процессе моделирования активным является тот из сегментов, в котором находится перемещающийся в настоящее время транзакт. Когда он блокируется, начинается движение следующего транзакта, который вполне возможно, принадлежит другому сегменту модели.

Содержательное значение транзактов определяет разработчик модели. Он устанавливает аналогию между транзактами и реальными элементами моделируемой им системы. Такая аналогия никогда не указывается планировщиком GPSS. Она присутствует только в памяти разработчика модели. Например, в модели системы технического обеспечения транзакты могут символизировать средства связи, автоматизации, отдельные транспортные средства, вооружение и технику, которые применяются, выходят из строя, эвакуируются при необходимости в ремонтный орган, ремонтируются и возвращаются в строй. В модели системы технического обеспечения на GPSS для обозначения всех этих процессов можно использовать различные блоки. Перемещение транзактов в этой модели от блока к блоку, таким образом, аналогично применению вооружения и техники.

На рис. 1.2 представлена блок-диаграмма модели системы сбора и обработки информации (донесений, заявок и др.) с применением средств автоматизации и выработке на основе этого управленческих воздействий. Помимо задач сбора информации на ЭВМ решаются различные задачи по ее обработке. Аналогами транзактов являются донесения, заявки и задачи. Построению блок-диаграммы должна предшествовать достаточно детальная разработка информационной модели с указанием всех источников информации, различных документов, путей их передачи, средств и способов обработки. Степень детализации информационной модели зависит от целей моделирования.

Подробное описание системы сбора и обработки информации приведено в п. 6.8. Поэтому к блок-диаграмме на рис. 1.2 следует вернуться в ходе изучения материала п. 6.8.

## 1.4. Структура модели на языке GPSS

После того как блок-диаграмма построена, она должна быть записана в форме, удобной для создания объекта «Модель». Для этого информация об объектах GPSS записывается в виде последовательности операторов. Оператор модели в GPSS World — это либо оператор GPSS, либо PLUS-оператор. Операторы GPSS в свою очередь — это или операторы блоков, которые создают блоки, или команды, которые не создают блоков.

В отличие от PLUS-операторов, которые могут содержать несколько строк, операторы GPSS должны записываться одной текстовой строкой длиной не более 250 символов. Операторы GPSS состоят из частей, называемых полями. Поле — это набор символов, отделенный пробелами или ограничителем. В общем случае оператор состоит из следующих полей: номер строки (необязательно) | метка (необязательно) | глагол (обязательно) | операнды (зависит от глагола) | комментарий (обязательно).

Номера строк оставлены для совместимости с GPSS/PC и игнорируются GPSS World. Операторы, несмотря на установленный для каждого формат, имеют свободную форму записи в том смысле, что нет необходимости стремиться к выравниванию полей операторов по колонкам.

Для различения объектов GPSS, например, ОКУ, МКУ, очередей и т. д., которых в модели может быть несколько, им дают имена. Имена должны начинаться с буквы, могут содержать до 200 букв и цифр, а также символы подчеркивания. Имя не должно совпадать с ключевыми словами GPSS. Одно и то же имя можно использовать для различных объектов.

В процессе моделирования возможны переходы от одного объекта модели к другому. С целью осуществления этого объект можно идентифицировать меткой, для чего служит поле метки. Метки образуются так же, как и имена блоков. GPSS присваивает меткам номера, начиная с 10 000. Пользователь не может сразу записать вместо метки номер. Если он хочет это сделать, нужно перед обращением к объекту метке командой EQU присвоить номер, который не может быть больше 9999. В качестве меток также нельзя использовать ключевые слова GPSS.

В поле глагола пользователь должен записать ключевое слово GPSS, т. е. или блок, или команду. Задание исходных данных, необходимых для выполнения операций, соответствующих блоку или команде, производится в поле операндов. Операнды разделяются запятыми.

Команды используются для описания многоканальных устройств, переменных, матриц ячеек, функций, таблиц и управления процессом моделирования.



Ниже приведена модель, написанная согласно рассмотренной в п. 1.3 блок-диаграмме (см. рис. 1.2). Модель имеет один объект «Модель» и два текстовых объекта — текстовых файла DanZad.txt и DanDon.txt, подключаемых к сегменту задания исходных данных объекта «Модель» командой INCLUDE. Этот сегмент содержит команды описания функций, таблиц, матриц и арифметических переменных, а остальные сегменты — операторы блоков GPSS.

```

; Primer 6.1
; Сегмент задания исходных данных
VrMod EQU      86400      ; Время моделирования
; Определение функций для расчета данных о задачах
INCLUDE      "DanZad.txt"; Подключение файла данных о задачах
; Определение таблиц
Don TABLE    M1, 120, 300, 40      ; Таблица времени обработки донесений
Zad TABLE    M1, 3600, 600, 50     ; Таблица времени решения задач
ObVremTABLE   M1, 1600, 600, 50     ; Таблица общего времени обработки
; Инициализация матрицы – запись данных о донесениях
INCLUDE      "DanDon.txt"; Подключение файла данных о донесениях
; Определение арифметических переменных для расчета
; количества знаков в донесениях
Kzd VARIABLE  INT(Normal(5, MX$TDon(1, P2), MX$TDon(2, P2)))
; количество точек диалога
Kdt VARIABLE  INT(Normal(7, FN$TZad2, FN$TZad3))
Skzd VARIABLE X$Vzd/N$KolDon      ; Среднее число знаков в донесении
; Сегмент имитации обработки донесений
GENERATE      (Exponential(3, 0, Swd)) ; Источник донесений
KolDonASSIGN  2, FN$TDon1          ; Определение вида донесения
ASSIGN        3, V$Kzd             ; Определение количества знаков в донесении
SAVEVALUE     Vzd+, P3             ; Счет знаков в донесениях
SEIZE         Ewm                  ; Занятие ЭВМ
ADVANCE       (Exponential(3, 0, MX$TDon(3, P2))); Обработка донесения
RELEASE       Ewm                  ; Освобождение ЭВМ
TABULATE      Don                  ; Табулирование времени обработки донесений
TRANSFER      ,Bbb2
; Сегмент имитации решения задач
GENERATE      (Exponential(3, 0, Swz)) ; Источник задач
ASSIGN        2, FN$TZad1          ; Определение типа задачи
ASSIGN        3, V$Kdt             ; Определение числа точек диалога
Bbb1 SEIZE    Ewm                  ; Занятие ЭВМ
ADVANCE       (Exponential(3, 0, FN$TZad4)) ; Ведение диалога
ADVANCE       (Exponential(3, 0, FN$TZad5)) ; Обработка после диалога
RELEASE       Ewm                  ; Освобождение ЭВМ
LOOP          3, Bbb1             ; Организация цикла по числу точек диалога
TABULATE      Zad                  ; Табулирование времени решения задач
Bbb2 TABULATE ObVrem              ; Табулирование общего времени обработки
; донесений и задач
TERMINATE     ; Вывод транзактов из модели

```

```

; Сегмент задания времени моделирования
; и расчета результатов моделирования
    GENERATE   VrMod ;Задание времени моделирования 1 ед. мод. вр.=1 с
    SAVEVALUE  Skzd,V$Skzd      ;Расчет среднего количество знаков
;                                     в донесении
    TERMINATE  1      ;Число, на которое уменьшается
;                                     содержимое счетчика завершения
    
```

Текстовый файл DanZad.txt, содержащий данные о задачах.

```

; Данные о задачах
Swz EQU 1800 ;Среднее время решения задач
. 1, 1/. 23, 2/. 31, 3/. 43, 4/. 52, 5/. 69, 6/. 82, 7/1, 8
TZad1 FUNCTION RN1, D6
. 14, 1/. 31, 2/. 46, 3/. 63, 4/. 81, 5/1, 6
TZad2 FUNCTION P2, D6
1, 23/2, 13/3, 29/4, 8/5, 19/6, 31
TZad3 FUNCTION P2, D6
1, 4/2, 2/3, 4/4, 1/5, 3/6, 5
TZad4 FUNCTION P2, D6
1, 25/2, 62/3, 78/4, 115/5, 93/6, 137
TZad5 FUNCTION P2, D6
1, 9/2, 13/3, 7/4, 5/5, 11/6, 16
    
```

Текстовый файл DanDon.txt, содержащий данные о донесениях.

```

; Данные о донесениях
Swd EQU 3600 ;Среднее время поступления донесений
Mdn MATRIX ,3, 8 ;Матрица данных о донесениях
INITIAL MX$TDon(1, 1), 420
INITIAL MX$TDon(1, 2), 510
INITIAL MX$TDon(1, 3), 630
INITIAL MX$TDon(1, 4), 920
INITIAL MX$TDon(1, 5), 1200
INITIAL MX$TDon(1, 6), 730
INITIAL MX$TDon(1, 7), 340
INITIAL MX$TDon(1, 8), 480
INITIAL MX$TDon(2, 1), 60
INITIAL MX$TDon(2, 2), 90
INITIAL MX$TDon(2, 3), 120
INITIAL MX$TDon(2, 4), 170
INITIAL MX$TDon(2, 5), 230
INITIAL MX$TDon(2, 6), 140
INITIAL MX$TDon(2, 7), 60
INITIAL MX$TDon(2, 8), 80
INITIAL MX$TDon(3, 1), 8
INITIAL MX$TDon(3, 2), 11
INITIAL MX$TDon(3, 3), 6
INITIAL MX$TDon(3, 4), 16
    
```

```

INITIAL      MХ$TDon(3, 5), 10
INITIAL      MХ$TDon(3, 6), 13
INITIAL      MХ$TDon(3, 7), 3
INITIAL      MХ$TDon(3, 8), 9

```

Объект «Модель» состоит из четырех сегментов: задания исходных данных; имитации обработки донесений; имитации решения задач; задания времени моделирования и расчета результатов моделирования.

Назначение и количество сегментов определяются разработчиком модели. Объект «Модель» является обязательным, а текстовые объекты — необязательными. Их назначение, содержание и количество также определяются разработчиком. Например, если сегменты имитации обработки донесений и решения задач представить также двумя текстовыми объектами — текстовыми файлами ObrDon.txt и ObrZad.txt соответственно, объект «Модель» будет представлен так.

```

; Primer 6.1
; Сегмент задания исходных данных
; Данные о задачах
    INCLUDE      "DanZad.txt"          ;Подключение файла данных о задачах
; Определение таблиц
Don  TABLE      M1, 120, 300, 40      ;Таблица времени обработки донесений
Zad  TABLE      M1, 3600, 600, 50     ;Таблица времени решения задач
ObVremTABLE     M1, 1600, 600, 50     ;Таблица общего времени обработки
;                                         донесений и задач
; Данные о донесениях
    INCLUDE      "DanDon.txt"          ;Подключение файла данных о донесениях
; Определение арифметических переменных для расчета
; количества знаков в донесениях
Kzd  VARIABLE     INT(Normal(5, MХ$TDon(1, P2), MХ$TDon(2, P2)))
; количество точек диалога
Kdt  VARIABLE     INT(Normal(7, FN$TZad2, FN$TZad3))
Skzd VARIABLE     X$Vzd/N$KoiDon      ;Среднее число знаков в донесении
; Сегмент имитации обработки донесений
    INCLUDE      "ObrDon.txt"          ;Подключение сегмента имитации
;                                         обработки донесений
; Сегмент имитации решения задач
    INCLUDE      "ObrZad.txt"          ;Подключение сегмента имитации
;                                         решения задач
; Сегмент задания времени моделирования
; и расчета результатов моделирования
GENERATE      86400 ;Задание времени моделирования 1 ед. мод. вр.=1 с
SAVEVALUE     Skzd, V$Skzd ;Расчет среднего числа знаков в донесении
TERMINATE     1 ;Число, на которое уменьшается содержимое
;                                         счетчика завершения

```

К данной модели нужно вернуться при изучении п. 6.8.

## 1.5. Объекты вычислительной категории

Как уже отмечалось, к объектам вычислительной категории относятся арифметические и булевы переменные и функции. Переменная — это обобщенное понятие, которое в конкретном случае представляет собой выражение. Выражение может включать следующие элементы:

- константы;
- системные числовые атрибуты;
- арифметические операции;
- логические операции;
- библиотечные функции;
- переменные пользователя.

Помимо применения в переменных, выражения в скобках могут также использоваться в операторах GPSS (некоторые операторы допускают применение и без скобок, но лучше использовать скобки).

Рассмотрим элементы выражений.

### 1.5.1. Константы

Константы — числа, представляются в одном из трех типов данных — целочисленном, вещественном или строковом. Так как арифметические операторы преобразуют свои операнды в подходящую форму, то в выражениях могут использоваться все три типа данных.

Целочисленные данные хранятся в виде 32-разрядных чисел. Если происходит переполнение, целое число преобразуется в вещественное.

Вещественные данные хранятся в виде 64-разрядных целых чисел с плавающей точкой двойной точности. Мантисса может изменяться от  $-308$  до  $308$ , а точность ограничена 15 десятичными разрядами.

Строковые данные хранятся в массивах символов ASCII любой длины. Длина ограничена только значением Max Memory Request (Максимальный запрос памяти) (страница Simulation (Моделирование), настройки объекта «Процесс моделирования»). Эти настройки можно изменить с помощью меню Edit ▶ Settings (Правка ▶ Настройки).

### 1.5.2. Системные числовые атрибуты

В процессе моделирования GPSS World автоматически регистрирует и корректирует информацию, представленную системными числовыми атрибутами различных объектов, используемых в модели. Полный перечень СЧА, классифицированных по объектам GPSS, представлен в приложении 2.

Имена СЧА зарезервированы. По способу определения имен СЧА можно разбить на три группы.

*Имя СЧА в первой группе* состоит из двух частей. *Первая часть* указывает групповое имя, идентифицирующее тип объекта и тип информации об объекте. *Вторая часть* идентифицирует конкретного члена группы.

*Групповое имя* состоит из одной-двух букв, фиксированных для информации об объектах определенного типа, например, Q — ссылка на текущее значение длины очереди, FR — коэффициент использования устройства, SR — коэффициент использования памяти и т. д.

Объекты GPSS могут быть идентифицированы с помощью числовых или символьных имен. Идентификация СЧА зависит от вида адресации, которых в GPSS предусмотрено две: прямая и косвенная.

Рассмотрим *прямую адресацию*. Если объект идентифицирован с помощью номера, то ссылка на его СЧА записывается как СЧА $j$ , где  $j$  — номер объекта (положительное целое число). Например, Q3 — текущее значение длины очереди номер 3, FR2 — коэффициент использования устройства номер 2. При указании имени объекта ссылка на его СЧА записывается как СЧА\$*имя*, где *имя* — имя объекта. Например, Q\$Basa — текущее значение длины очереди с именем Basa, SR\$Rem — коэффициент использования МКУ с именем Rem.

При косвенной адресации СЧА определяются как СЧА\*параметр. Это означает следующее:

- СЧА\* $j$ ,  $j$  — номер параметра активного транзакта (положительное целое число), содержащего номер нужного блока;
- СЧА\**имя*, *имя* — имя параметра активного транзакта, содержащего номер нужного блока.

Например, Q\*3 — текущее значение длины очереди, номер которой является значением параметра 3 активного транзакта, SR\*Rem — коэффициент использования памяти, номер которой содержится в параметре с именем Rem активного транзакта.

*Вторую группу*, как частный случай первой группы, составляют матричные СЧА. Матричный СЧА MX может содержать до трех идентификаторов (косвенных адресов). Например:

MX\*Result(\*Stroka,\*Stolbez)

является ссылкой на матрицу, номер которой содержится в параметре активного транзакта с именем Result, а элемент матрицы — номер строки и номер столбца — определяется значениями параметров с именами Stroka и Stolbez соответственно того же активного транзакта.

*Третью группу* составляют СЧА A1, AC1, C1, M1, PR, TG1, XN1, Z1. В руководстве пользователя [22] они названы «атомарными». Эти СЧА, в отличие от СЧА первой и второй групп, не требуют указания номера или имени объекта.

СЧА по *признаку доступности к ним пользователя* можно разделить на две группы.

*Первую группу* составляют атрибуты, которые в части, касающейся их формирования, доступны только GPSS World, однако они могут быть использованы также и пользователем в выражениях, но не могут им изменяться в процессе моделирования. Эта группа содержит наибольшее число СЧА.

Во вторую группу входят СЧА вычислительных объектов, которые могут изменяться разработчиком модели и использоваться им в выражениях:

- FNj (FN\$Raspr) — вычисленное вещественное значение функции номер j (с именем Raspr);
- Vj (V\$Ver) — вычисленное значение вещественной или целочисленной переменной номер j (с именем Ver);
- BVj (BV\$Per) — вычисленное вещественное значение булевой переменной номер j (с именем Per).

### 1.5.3. Арифметические, условные и логические операторы

Арифметические, условные и логические операторы предназначены для указания операций, которые должны выполняться над элементами выражения. В GPSS World используются следующие операторы (табл. 1.2).

Если оператор требует определенного типа данных, данные преобразуются автоматически. Например, если над строковой величиной производится числовая операция, то будет использован числовой эквивалент этой строки символов.

Оператор «#» использован для операции умножения потому, чтобы применять оператор «\*» для указания косвенной адресации. Однако пользователь может менять значение операторов «#» и «\*». Для этого нужно воспользоваться флажком, имеющимся на странице Simulation (Моделирование) журнала настроек.

Операции над элементами выражения выполняются в следующей последовательности в порядке убывания приоритетов:  $\wedge > \# / \backslash > @ > - + > >= <= > < > != = > \& > |$ .

Таблица 1.2

Оператор	Операция	Аргумент, значение
$\wedge$	Возведение в степень	$A^B$ ; A, возведенное в степень B
#	Умножение	$A\#B$ ; произведение A на B
/	Деление	$A/B$ ; частное от деления A на B
$\backslash$	Целочисленное значение	$A\backslash B$ ; частное от целочисленного деления A на B
@	Целый остаток	$A@B$ ; целый остаток от деления A на B
-	Вычитание	
+	Сложение	
$>='GE'$	Больше или равно	1, если $A \geq B$ , иначе — 0
$<='LE'$	Меньше или равно	1, если $A \leq B$ , иначе — 0
$>'G'$	Больше	1, если $A > B$ , иначе — 0
$<'L'$	Меньше	1, если $A < B$ , иначе — 0

Таблица 1.2 (продолжение)

Оператор	Операция	Аргумент, значение
= 'E'	Равно	1, если $A=B$ , иначе — 0
!= 'NE'	Не равно	1, если $A \neq B$ , иначе — 0
& 'AND'	Логическое «И»	1, если A и B не 0, иначе — 0
'OR'	Логическое «ИЛИ»	1, если A или B или оба не 0, иначе — 0

#### 1.5.4. Библиотечные математические функции

Библиотека GPSS World имеет следующие математические функции (процедуры) (табл. 1.3).

Таблица 1.3

Оператор	Функции	Аргумент, значение
ABS(A)	Абсолютное значение	$A^B$ ; A, возведенное в степень B
ATN(A)	Арктангенс	$A \# B$ ; произведение A на B
COS(A)	Косинус	$A / B$ ; частное от деления A на B
EXP(A)	Экспонента	$A \setminus B$ ; частное от целочисленного деления A на B
INT(A)	Выделение целой части	$A @ B$ ; целый остаток от деления A на B
LOG(A)	Натуральный алгоритм	
SIN(A)	Синус	
SQR(A)	Квадратный корень	1, если $A \geq B$ , иначе — 0
TAN(A)	Тангенс	1, если $A \leq B$ , иначе — 0

Аргументы библиотечных функций во всех случаях автоматически преобразуются в числовые значения. Допустимые значения аргумента Expression (Выражение). Вычисляемые числовые значения функций — вещественного типа.

#### 1.5.5. Библиотечные генераторы случайных чисел

Генераторы случайных чисел предназначены для моделирования случайных величин и случайных событий.

При пользовании табл. 1.4 нужно иметь в виду следующее:

- все аргументы генераторов имеют допустимое значение Expression;
- после вычисления аргумент преобразуется к соответствующему типу;
- символом «\*» помечены аргументы, которые автоматически преобразуются в целое число;
- символом «\*\*» помечены аргументы, автоматически преобразующиеся в вещественное число;

- аргумент  $A$  после вычисления выражения и преобразования в целое число используется в качестве номера генератора случайных чисел;
- генерирование случайных чисел производится по выражениям [22].

Таблица 1.4

Распределение, оператор, тип	Аргументы
Бета $Beta(A, B, C, D, E)$ Вещественный	$A^*$ — номер генератора случайных чисел, $A \geq 1$ ; $[B^*, C^*]$ — интервал генерируемых значений, $B > C$ ; $D^{**}$ и $E^{**}$ — первое и второе определяющие значения, использующиеся для выбора из семейства форм. $Beta(A, B, C, 1, 1) = Uniform(A, B, C)$ , $Beta(A, B, C, 1, 2)$ — левый, $Beta(A, B, C, 2, 1)$ — правый треугольники.
Биноминальное $Binominal(A, B, C)$ Целый	$A^*$ — номер генератора случайных чисел, $A \geq 1$ ; $B^*$ — количество независимых испытаний Бернулли, $B > 0$ ; $C^{**}$ — вероятность успешных испытаний Бернулли, $C \in [0, 1]$ .
Дискретно-равномерное $DUniform(A, B, C)$ Целый	$A^*$ — номер генератора случайных чисел, $A \geq 1$ ; $[B^*, C^*]$ — интервал генерируемых значений, $B > C$ .
Экспоненциальное $Exponential(A, B, C)$ Вещественный	$A^*$ — номер генератора случайных чисел, $A \geq 1$ ; $B^{**}$ — величина сдвига, определяющего местоположение распределения; $C^{**}$ — величина, используемая для сжатия или растяжения распределения, $C > 0$ , $B + C$ — среднее значение, при $B = 0$ $C$ — среднее значение.
Экстремального значения типа А $ExtValA(A, B, C)$ Вещественный	$A^*$ — номер генератора случайных чисел, $A \geq 1$ ; $B^{**}$ — величина сдвига, определяющего местоположение распределения; $C^{**}$ — величина, используемая для сжатия или растяжения распределения, $C > 0$ .
Экстремального значения типа В $ExtValB(A, B, C)$ Вещественный	$A^*$ — номер генератора случайных чисел, $A \geq 1$ ; $B^{**}$ — величина сдвига, определяющего местоположение распределения; $C^{**}$ — величина, используемая для сжатия или растяжения распределения, $C > 0$ .
Геометрическое $Geometric(A, B)$ Целый	$A^*$ — номер генератора случайных чисел, $A \geq 1$ ; $B^{**}$ — вероятность успешных испытаний Бернулли, $B \in [0, 1]$ .



Таблица 1.4 (продолжение)

Распределение, оператор, тип	Аргументы
Гамма Gamma(A,B,C,D) Вещественный	<p>A* — номер генератора случайных чисел, <math>A \geq 1</math>;</p> <p>B** — величина сдвига, определяющего местоположение распределения;</p> <p>C** — величина, используемая для сжатия или растяжения распределения, <math>C &gt; 0</math>;</p> <p>D** — определяющее значение, используемое для выбора из семейства форм, <math>D &gt; 0</math>;</p> <p>Gamma(A,B,C,1)=Exponential(A,B,C).</p>
Обратное Гаусса InvGauss(A,B,C,D) Вещественный	<p>A* — номер генератора случайных чисел, <math>A \geq 1</math>;</p> <p>B** — величина сдвига, определяющего местоположение распределения;</p> <p>C** — величина, используемая для сжатия или растяжения распределения, <math>C &gt; 0</math>;</p> <p>D** — определяющее значение, используемое для выбора из семейства форм, <math>D &gt; 0</math>.</p>
Обратное Вейбулла InvWeibull(A,B,C,D) Вещественный	<p>A* — номер генератора случайных чисел, <math>A \geq 1</math>;</p> <p>B** — величина сдвига, определяющего местоположение распределения;</p> <p>C** — величина, используемая для сжатия или растяжения распределения, <math>C &gt; 0</math>;</p> <p>D** — определяющее значение, используемое для выбора из семейства форм, <math>D &gt; 0</math>.</p>
Лапласа Laplace(A,B,C) Вещественный	<p>A* — номер генератора случайных чисел, <math>A \geq 1</math>;</p> <p>B** — величина сдвига, определяющего местоположение распределения;</p> <p>C** — величина, используемая для сжатия или растяжения распределения, <math>C &gt; 0</math>.</p>
Логлапласово LogLaplace(A,B,C,D) Вещественный	<p>A* — номер генератора случайных чисел, <math>A \geq 1</math>;</p> <p>B** — величина сдвига, определяющего местоположение распределения;</p> <p>C** — величина, используемая для сжатия или растяжения распределения, <math>C &gt; 0</math>;</p> <p>D** — определяющее значение, используемое для выбора из семейства форм, <math>D &gt; 0</math>.</p>

Таблица 1.4 (продолжение)

Распределение, оператор, тип	Аргументы
Отрицательное биномиальное NegBinom(A, B, C) Целый	A* — номер генератора случайных чисел, $A \geq 1$ ; B* — количество успешных испытаний Бернулли, после которого возвращается число безуспешных испытаний, $B > 0$ ; C** — вероятность успешных испытаний Бернулли, $C \in [0, 1]$ .
Логистическое Logistic(A, B, C) Вещественный	A* — номер генератора случайных чисел, $A \geq 1$ ; B** — величина сдвига, определяющего местоположение распределения; C** — величина, используемая для сжатия или растяжения распределения, $C > 0$ .
Логлогистическое LogLogis(A, B, C, D) Вещественный	A* — номер генератора случайных чисел, $A \geq 1$ ; B** — величина сдвига, определяющего местоположение распределения; C** — величина, используемая для сжатия или растяжения распределения, $C > 0$ ; D** — определяющее значение, используемое для выбора из семейства форм, $D > 0$ .
Логнормальное LogNormal(A, B, C, D) Вещественный	A* — номер генератора случайных чисел, $A \geq 1$ ; B** — величина сдвига, определяющего местоположение распределения; C** — величина, используемая для сжатия или растяжения распределения, $C > 0$ ; D** — определяющее значение, используемое для выбора из семейства форм, $D > 0$ .
Парето Pareto(A, B, C) Вещественный	A* — номер генератора случайных чисел, $A \geq 1$ ; B** — величина сдвига, определяющего местоположение распределения; C** — величина, используемая для сжатия или растяжения распределения, $C > 0$ .
Пирсона типа 5 Pearson5(A, B, C, D) Вещественный	A* — номер генератора случайных чисел, $A \geq 1$ ; B** — величина сдвига, определяющего местоположение распределения; C** — величина, используемая для сжатия или растяжения распределения, $C > 0$ ; D** — определяющее значение, используемое для выбора из семейства форм, $D > 0$ .

Таблица 1.4 (продолжение)

Распределение, оператор, тип	Аргументы
Нормальное Normal(A,B,C) Вещественный	A* — номер генератора случайных чисел, $A \geq 1$ ; B** — среднее значение распределения; C** — стандартное отклонение распределения, $C > 0$ .
Пуассона Poisson(A,B) Целый	A* — номер генератора случайных чисел, $A \geq 1$ ; B** — среднее значение распределения.
Пирсона типа б Pearson6(A,B,C,D,E) Вещественный	A* — номер генератора случайных чисел, $A \geq 1$ ; [B*,C*] — интервал генерируемых значений, $B > C$ ; D** и E** — первое и второе определяющие значения, использующиеся для выбора из семейства форм, $Beta(A,B,C,1,1) = Uniform(A,B,C)$ .
Треугольное Triangular(A,B,C,D) Вещественный	A* — номер генератора случайных чисел, $A \geq 1$ ; [B*,C*] — интервал генерируемых значений, $B > C$ ; D** — мода распределения, $D \in [B,C]$ .
Равномерное Uniform(A,B,C) Вещественный	A* — номер генератора случайных чисел, $A \geq 1$ ; [B**,C**] — интервал генерируемых значений, $B > C$ , $Uniform(A,B,C) = Beta(A,B,C,1,1)$ .
Вейбулла Weibull(A,B,C,D) Вещественный	A* — номер генератора случайных чисел, $A \geq 1$ ; B** — величина сдвига, определяющего местоположение распределения; C** — величина, используемая для сжатия или растяжения распределения, $C > 0$ ; D** — определяющее значение, используемое для выбора из семейства форм, $D > 0$ ; $Weibull(A,B,C,1) = Exponential(A,B,C)$ $Weibull(A,B,C,2)$ — распределение Релея.

### 1.5.6. Переменные пользователя

GPSS предоставляет пользователю возможность иметь свои переменные для хранения нужных при моделировании как числовых, так и строковых данных. Переменные пользователя создаются с помощью команды EQU или присваивающих PLUS-процедур. Например:

```
Ver1      EQU      2.65
Stroka    EQU      11
Stolbez   EQU      8
Name      EQU      "Name"
```

В приведенных первых трех примерах переменным пользователя Ver1, Stroka, Stolbez присваиваются числовые значения, а переменной Name — строка.