

Ц8481
Ш - 859

СООБЩЕНИЯ
ОБЪЕДИНЕННОГО
ИНСТИТУТА
ЯДЕРНЫХ
ИССЛЕДОВАНИЙ

ДУБНА



5007/2-78

P11 - 11758

Э.Шрейбер, Г.Бергхолц

ПРИМЕНЕНИЕ ИМИТАЦИОННОГО МОДЕЛИРОВАНИЯ
ДЛЯ АНАЛИЗА ЭФФЕКТИВНОСТИ ЭВМ
В КОМПЛЕКСЕ
С ЭКСПЕРИМЕНТАЛЬНОЙ УСТАНОВКОЙ
(Часть 1)

1978

P11 - 11758

Э.Шрейбер, Г.Бергхолц*

ПРИМЕНЕНИЕ ИМИТАЦИОННОГО МОДЕЛИРОВАНИЯ
ДЛЯ АНАЛИЗА ЭФФЕКТИВНОСТИ ЭВМ
В КОМПЛЕКСЕ
С ЭКСПЕРИМЕНТАЛЬНОЙ УСТАНОВКОЙ
(Часть 1)

* Технический университет, Дрезден, ГДР

Шрейбер Э., Бергхолц Т.

PII - 11758

Применение имитационного моделирования для анализа эффективности ЭВМ в комплексе с экспериментальной установкой (Часть 1).

Рассмотрено применение имитационного моделирования для анализа эффективности ЭВМ в комплексе с экспериментальной установкой. Моделирование производится статистическим методом на основе системы программ SIMDIS, предназначенной для дискретного моделирования на ЭВМ ЕС-1040.

Разработана программа, осуществляющая прием измерительных данных с установки и их обработку на языке SIMDIS. В результате статистического анализа получены законы распределения времени реакции, времени пребывания и времени ожидания поступающих заявок, а также математическое ожидание времени занятий ресурсов модели.

Проведенное исследование важно для анализа алгоритмов с целью повышения их эффективности.

Работа выполнена в Лаборатории вычислительной техники и автоматизации ОИЯИ.

Сообщение Объединенного института ядерных исследований. Дубна 1978

Schreiber E., Bergholz G.

PII - 11758

The Application of Simulation for the Analysis of Computer Effectiveness On-Line with Data Processing System

Simulation is performed by statistical methods on the basis of operating systems SIMDIS intended for discrete simulation on EC-1040 computer. A specific program representing a system program for data processing in SIMDIS is developed. As a result of statistical analysis, the laws of reaction time distribution and time of staying and waiting for entering requests and also a middle value of execution time for processing programs and devices have been obtained. The investigation is important for the analysis of algorithms for the purpose of increasing their efficiency.

The investigation has been performed at the Laboratory of Computing Techniques and Automation, JINR.

Communication of the Joint Institute for Nuclear Research. Dubna 1978

I. Введение

Исследование эффективности ЭВМ можно разделить на две части: анализ независимых факторов и собственно анализ эффективности /1/. Анализ факторов заключается прежде всего в установке графической схемы заявок и измерении распределения функции входных потоков и времен обслуживания. Анализ эффективности состоит в установлении связи между этими факторами и значениями оценок. Под значениями оценок мы понимаем характеристики времен реакции и пропускной способности ЭВМ. При этом независимым фактором является сама графическая схема заявок, представляющая собой структуру модели. Другими независимыми факторами являются характеристики входных потоков и характеристики обслуживания ресурсов.

В основе анализа эффективности лежат так называемые модели поведения. Анализ базируется на следующих трех методах:

1. Аналитическая теория.
2. Имитационное моделирование.
3. Системные измерения.

В нашем частном случае ограничимся имитационным моделированием. Применение этого метода полезно в том случае, когда исследуемая система не поддается изучению аналитическими методами, а прямое экспериментирование с самой системой нецелесообразно. Наш объект - это установка "ФОТОН", работающая на линии с ЭВМ НР-2И16В /2/.

2. Принцип имитационного моделирования

Моделирование – это метод исследования, в котором проводятся эксперименты на основе данной модели. Основу имитационного моделирования составляет отношение между моделируемым объектом и рассматриваемой моделью, которое охватывает наиболее важные свойства оригинала.

Моделируемый объект характеризуется изменением его состояния, что изображено в модели. Под главным элементом исследуемой ЭВМ мы понимаем заявки и ресурсы обслуживания, которые являются характеристиками состояния объекта [3]. Ресурсы обслуживания – это такие компоненты вычислительной машины, которые помогают обслуживать заявку. Связь между элементами модели реализуется с помощью операционных блоков.

Ресурсы обслуживания состоят из процессоров (P_1, P_2, \dots, P_p), данных программ (AP_1, AP_2, \dots, AP_m) информационных каналов ($IAK_1, IAK_2, \dots, IAK_n$), областей основной памяти (SP_1, SP_2, \dots, SP_p). Ресурсы обслуживания можно пронумеровать, в результате чего мы получаем q ресурсов обслуживания R_1, R_2, \dots, R_q , где

$$q = m + l + n + p. \quad (I)$$

Каждому ресурсу можно поставить в соответствие характеристику занятости $R_i (i=1, 2, \dots, q)$. R_i может принимать значение, равное числу поступивших заявок:

$$R_i = \begin{cases} 0 & - \text{заявок не поступило} \\ 1 & - \text{поступила одна заявка} \\ 2 & - \text{поступило две заявки} \\ \vdots & \\ N_i & - \text{поступило } N_i \text{ заявок} \end{cases}$$

Здесь N_i – число каналов ресурса $R_i (i=1, 2, \dots, q)$. При изменении состояния ресурса его занятость $R_i (i=1, 2, \dots, q)$ изменяется

$$R_i = R_i + \Delta R_i. \quad (2)$$

При этом $\Delta R_i = \pm 1$, в зависимости от того, поступила или освободилась данная заявка. Многоканальному ресурсу можно поставить в соответствие граф состояния.

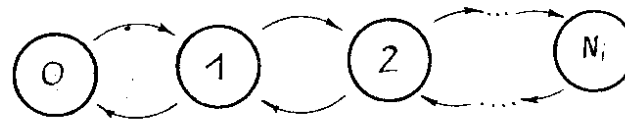


Рис.1. Граф состояния N -канального ресурса

Каждая заявка $F_j (j=1, 2, \dots)$ характеризуется состоянием существования EZ_j , принимающей следующие значения:

- 0 – неосуществимая
- pl – планируемая
- be – обслуживаемая
- au – задерживаемая
- ve – остановленная
- wa – ожидаемая

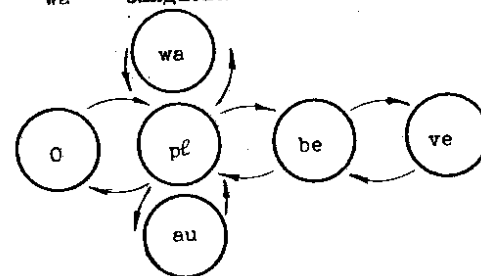


Рис.2. Состояние существования одной заявки

Изменение состояния производится специальными операционными блоками. Их совокупность образует графическую схему заявок исследуемого комплекса. В нашем случае моделирование заключается в рассмотрении последовательности определенных событий, каждое из которых характеризуется случайной временной точкой. Между этими временными точками состояние процесса остается постоянным. Соответствие значений x_1, x_2, \dots, x_n случайной величине X и временным точкам показано на рис.3.

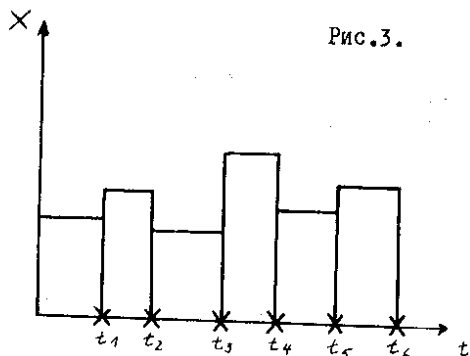


Рис.3.

3. SIMDIS - графическая схема заявок

Облегчить работу позволяет система программирования SIMDIS /4/, предназначенная для дискретного моделирования. Моделирование с помощью этой системы осуществляется несколькими операционными блоками в реальном масштабе времени. Внутри SIMDIS-программы заявки движутся от блока к блоку и заставляют их срабатывать. Заявки могут по необходимости генерироваться или уничтожаться.

Каждая заявка имеет определенное состояние, которое вследствие движения ее изменяется в разных местах графической схемы заявок. В SIMDIS мы рассматриваем только внешние состояния, которые можно вызывать с помощью стандартных символов. В SIMDIS имеются разные виды ресурсов:

- устройства,
- очереди,
- логические переключатели.

Для состояния разных ресурсов введены стандартные символы:

- F_j - состояние устройства j ,
- Q_j - мгновенная длина очереди j ,
- L_j - состояние логического переключателя.

Характеристики главных блоков:

При генерировании заявок мы различаем внутренние и внешние их источники. За внешним источником заявок расположен блок GENERATE, в котором состояние существования EZ_j изменяется от 0 до p^e . Одновременно генерируется ссылка, позволяющая обращаться к объекту. Уничтожение заявок осуществляется блоком TERMINATE.

При срабатывании этого блока заявка покидает систему, а ее место в памяти остается свободным и пригодным для другого использования. Для определения нового временного события и моделирования операций обслуживания существует блок ADVANCE.

Для внутренних источников, которые копируют заявку (оригинал), мы используем блок SPLIT. Оригинал и множество копий составляют группу. Этот блок помещается в графической схеме заявок внутри вызывающей программы или в местах информационных каналов. Условная задержка заявок осуществляется операциями при выполнении определенного арифметического условия или условия состояния логического переключателя блока GATE. Для разветвления используются модифицированные блоки TEST, GATE и TRANSFER.

Для изменения состояния ресурсов существуют следующие блоки: SEIZE и RELEASE, которые могут отправлять заявки без приоритета. В SIMDIS осуществляется контроль очереди заявок, емкость их практически бесконечна. Для статистики очереди используются блоки QUEUE и DEPART.

Требование заявки на обслуживание, которое не может быть выполнено каким-либо устройством, ставит его в очередь (QUEUE) по отношению к данному устройству. Продвижение в очереди происходит по правилу "первый вошел - первый вышел". Блок DEPART регистрирует выход из очереди. Напомним, что в нашем случае моделируемой системой являются установка "ФОТОН" и ЭВМ. Часть графической схемы заявок приведена на рис.4. Использование моделью системы блоков SIMDIS показано на рис.5.

4. Статистический анализ

Хорошо известно, что имитационное моделирование является статистическим исследовательским методом. Мы видели выше, что моделирование случайных процессов происходит путем присвоения случайным величинам временных точек. Статистический анализ состоит из двух частей:

- измерение соответствующих величин,
- статистическая оценка выполненных измерений.

В результате статистического анализа мы получаем оценки параметров распределения случайных величин, а именно:

- математическое ожидание,
- стандартное отклонение,
- вид закона распределения.

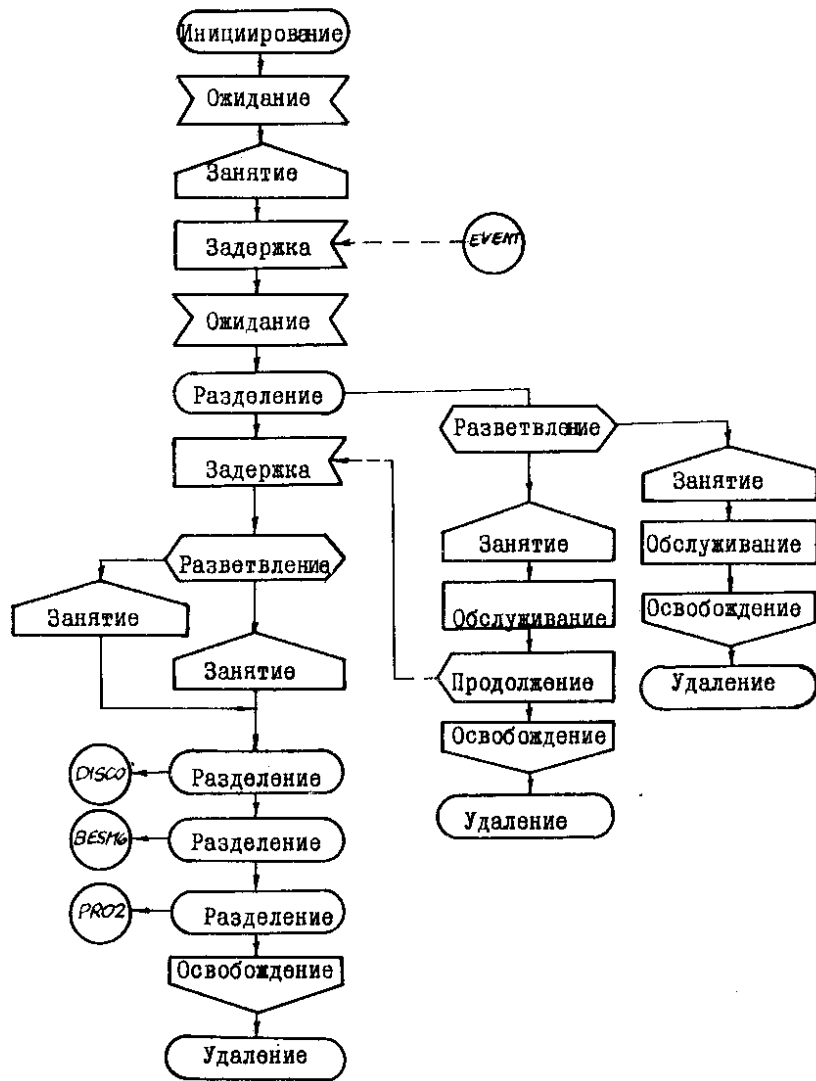
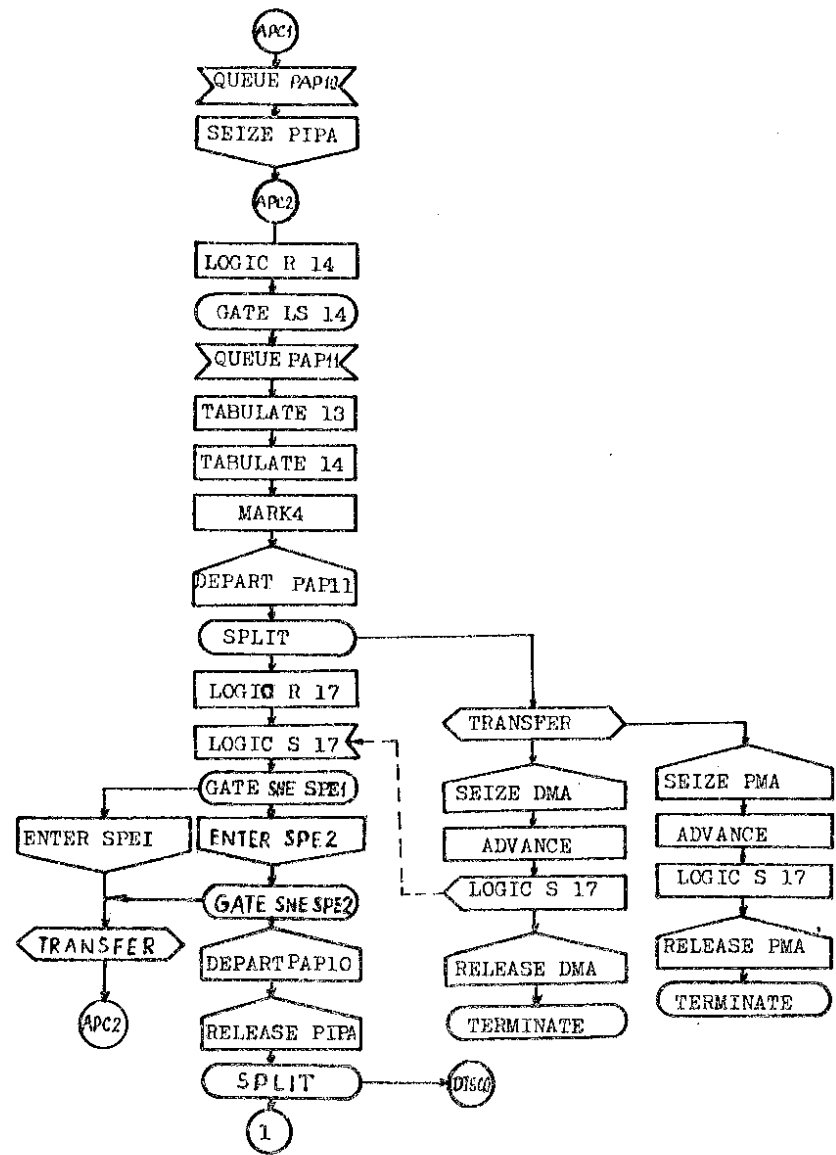


Рис.4. Часть графической схемы заявок



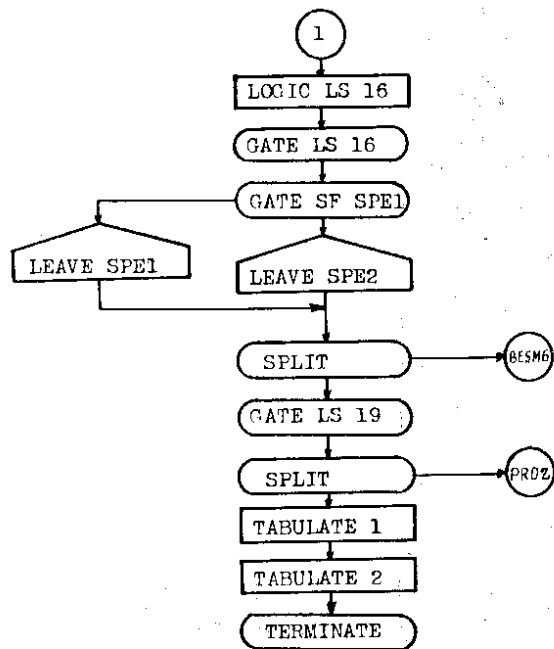


Рис.5. Графическая схема заявок при использовании блоков системы SIMDIS.

Для оценки математического ожидания и стандартного отклонения случайной величины T можно использовать хорошо известные формулы:

$$E(T) = \bar{T} = \frac{1}{n} \sum_{i=1}^n t_i \quad (3)$$

$$\sigma(T) = \sqrt{\frac{1}{n-1} \sum_{i=1}^n (t_i - \bar{T})^2} \quad (4)$$

Оценки закона распределения следуют из моделирования таблицы частоты. При этом мы рассматриваем случайную величину T в области между t_{\min} и t_{\max} , которую разделим на k одинаковых классов. Для каждого класса получаем в результате моделирования абсолютную частоту n_i ($i=1,2,\dots,k$) и относительную частоту:

$$N_i = (n_i / \sum_{j=1}^k n_j) \quad (5)$$

В SIMDIS есть разные возможности статистического анализа.

Операторы анализа частоты QTABLE и TABLE

Для статистического анализа очереди можно использовать блоки QUEUE и DEPART, связанные с оператором QTABLE, определяющим частоту присутствия заявки в очереди. С помощью QTABLE можно получить информацию о частоте вызова данной программы и о месте пребывания заявки.

Оператор TABLE устанавливает распределение частот по классам событий. При этом TABLE имеет следующие параметры: объект регистрации в каждой таблице, число интервалов и их границы. Цифровые стандартные символы, задающие объект регистрации (MP, IA, RT, FT), определяют тип таблицы. Следующие параметры определяют нижнюю границу первого класса частоты и его ширину. Эта ширина совпадает с временем, которое необходимо заявке для прохождения через программу.

Оператор QTABLE для нашей программы имеет следующую структуру:

```

1 QTABLE   PAP 10, 1,2110, 5
  TABLE   1
  
```

Табличная оценка времени пребывания

При использовании SIMDIS с помощью блоков MARK и TABULATE можно определить время пребывания заявки между прохождением этих блоков. Таким путем можно установить время с момента генерации

заявки до конца обслуживания, и занести его в таблицу. С помощью оператора определения в таблице частоты размещаются: значения границы t_{\min} , широта класса T и числа класса K . При этом будет распечатана результирующая таблица, содержащая время пребывания T_v , оценку математического ожидания $E(T_v)$, стандартное отклонение и закон распределения функции. Время пребывания заявки в нашей программе можно представить в виде:

```
12 TABLE FT11, 0, 240, 10
   TABULATE 12
```

Начало программы содержит блоки MARK4 и MARK5. В каждом из них складываются времена пребывания событий для разных ступеней обработки.

Время пребывания событий до запоминания на диске есть:

```
15 TABLE MP4, 0, 164, 15
   MARK 4
   TABULATE 15
```

а время пребывания событий до обработки через всю программы равно:

```
16 TABLE MP5, 0, 11847, 10
   MARK 5
   TABULATE 16
```

Для оценки количества событий в каждом цикле используется оператор следующего типа:

```
13 TABLE RT, 0, 1, 25, 30000
   TABULATE 13
```

Последний параметр определяет предусмотренный временной интервал. Другой важный оператор служит для сбора временных расстояний между двумя последовательными событиями:

```
14 TABLE IA, 0, 30000, 10
   TABULATE 14
```

С обработкой измерительных величин, участвующих в процессе, связана транспортировка данных по каналам информации и запоминания данных в памяти. Для этой цели существует хранилище - устройство с разделенным объемом и заданной емкостью. Оператор ENTER

определяет величину требуемого объема, а оператор LEAVE освобождает объем.

Для использования данного устройства необходимо задать табличную оценку всех обращений и выходов в течение определенного интервала времени.

Моделируемая единица времени

Моделируемая единица времени (SZE) указывает интервал времени моделирования. Все временные показания должны преобразоваться в SZE. В данной программе SZE имеет следующее значение:

$$1 \text{ SZE} = 10 \mu \text{с.}$$

Для транспортировки данных по информационному каналу эта величина соответствует значению параметра времени обслуживания $T_1 = 11250 \mu \text{с}$. Данная информация содержится в скалярной переменной блока ADVANCE $X19 = 1125 \text{ SZE}$.

В нашей модели полный процесс представляет собой единство процесса реакции и процесса обработки информации. Оба процесса необходимо синхронизовать с помощью определенных блоков. При представлении процесса в графической схеме заявок учитываются как семантический, так и синтаксический аспекты процесса.

Описание процесса реакции

Процесс реакции начинается с блока GENERATE, в котором генератор случайных чисел генерирует заявки с периодом $T = 10 \text{ с}$. Эти заявки занимают устройства, и прохождение соответствующих логических блоков констатирует допустимый интервал. Внутри этого интервала выступают события, которые удовлетворяют распределению Пуассона. При этом каждое событие требует своей обработки в данном процессе. Переключатели управляются посредством определенных цепочек задержки, которые либо задерживают заявки, либо их ускоряют.

5. Вопросы программирования в SIMDIS

SIMDIS - программа, состоящая из определенных утверждений. Отметим следующие важные типы операторов:

- операторы блоков,
- операторы определения,
- операторы управления.

Операторы блоков описывают модель в соответствии с графической схемой заявок (SIMDIS). Операторы определений служат прежде всего для проверки начальных данных и согласования параметров таблицы частоты, необходимых для статистического анализа.

Управление процессом моделирования осуществляется в нашей модели с помощью следующей конструкции

```
GENERATE 12 00 000 000 00
TERMINATE 1
START 1
END
```

помещаемой в конце моделирующей программы. Здесь в блоке GENERATE стоит время моделирования. Количество процессов моделирования указывается в блоках TERMINATE и START. START представляет собой счетчик пуска TG1=1. Блок TERMINATE вычисляет TG1-TG1-1 и производит проверку счетчика пуска. При положительном значении счетчика пуска моделирование заканчивается.

Моделирование реализовано на ЭВМ ЕС-1040. Моделирующая программа записана на магнитную ленту. Необходимо записать ее в библиотеку загрузочных модулей (CL) с помощью задания.

```
// JOB CATAL
// ASSGN SYSIPT, 380
// OPTION CATAL
// EXEC MAINT
// MTC PSF, SYSIPT, 2
INCLUDE
// EXEC LNKEDT
// MTC RUN, SYSIPT
// RESET SYSIPT
// EXEC DSERV
DSPCY RO,CD
/*
// EXEC CSERV
DSPLY ALL
/*
/&
```

Работа с данной программой возможна только в операционной системе DOS. Отметим, что программно-логические устройства (SYS000, SYS001, SYS002, SYS003) указывают зоны памяти на системных дисках. Для каждого файла должны существовать соответствующие информационные метки (указание DLBL и EXTENT). Оператор задания моделируемой системы определяется этими информационными метками. Затем необходимо обработать следующие задания:

```
// JOB NAME
// OPTION LINK
INCLUDE VSIMDIS
// EXEC LNKEDT
// ASSGN SYS000, SYS001
// DLBL IJSYS00, 'SYSTEM-ARBEITSDATEI 0', 0.
// EXTENT SYS000, , 1, 0, 1750, 250
// DLBL ARBDT1E, 'SYSTEM-ARBEITSDATEI 0', 0
// EXTENT SYS000, , 1, 0, 1750, 250
// DLBL ARBDT2E, 'ARBEITSDATEISYS001', 0
// EXTENT SYS001, 111111, 8, 1, 10, 600, 4
// DLBL SYMBOLE, 'ARBEITSDATEISYS002', 0.
// EXTENT SYS002, 111111, 8, 1, 15, 600, 9
// DLBL REPORTE, 'ARBEITSDATEISYS003', 0
// EXTENT SYS003, 111111, 0, 1210, 540
// EXEC
```

6. Представление статистических результатов

Известно как имитационное моделирование на основе системы SIMDIS. Такое моделирование проводится с целью определения оценки характеристик теории вероятности. При этом важно знать зависимость величины оценки от независимых факторов влияния. В нашей конкретной модели мы рассматриваем следующие величины:

- закон распределения времени реакции, времени пребывания и времени ожидания, являющийся функцией от функции распределения времени прибытия и времени обслуживания;
- математические ожидания времени реакции, времени пребывания и времени ожидания как функции нормы прибытия и математического ожидания времени занятия ресурсов;

- пропускная способность разных ресурсов, также являющихся функцией нормы прибытия и математического ожидания времени занятий ресурсов.

SIMDIS допускает индивидуальное обслуживание. Наряду со стандартной печатью имеется возможность распечатки результатов в виде, наиболее удобном для пользователя. При этом выводятся на печать: гистограммы, средние значения случайных величин, их средние - квадратичные отклонения.

Авторы считают своим долгом выразить благодарность М.Кунике и А.Е.Сеннеру за ценные советы и помощь при подготовке рукописи к публикации.

Литература

1. Шрейбер Э., Бергхолц Г. ОИЯИ, Р11-11759, Дубна, 1978.
2. Говорун Н.Н. и др. ОИЯИ, Р10-7460, Дубна, 1973.
3. Bergholz G. ZfR INFORMATION NR - 01. 77
Akademie der Wissenschaften DDR Berlin 1977.
4. Anwender - Handbuch, Programm System für diskrete Simulation
Vops Simdis, veb Robotron, Dresden (1974).

Рукопись поступила в издательский отдел
14 июля 1978 года.