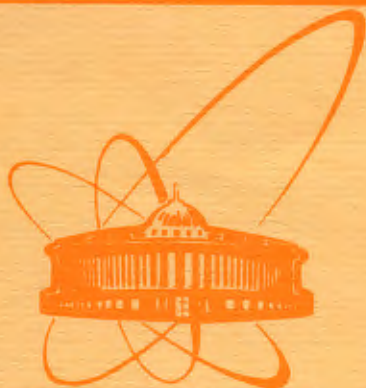


12160



объединенный  
институт  
ядерных  
исследований  
дубна

КЗ. ЧИТ. ЗАЛА

P11 - 12160

А.А.Карлов, А.С.Кирилов, Т.Ф.Смолякова

ПРОГРАММНЫЕ СРЕДСТВА  
ДЛЯ ОРГАНИЗАЦИИ ДИАЛОГА  
НА ЯЗЫКЕ ВЫСОКОГО УРОВНЯ

1979

P11 - 12160

А.А.Карлов, А.С.Кирилов, Т.Ф.Смолякова

ПРОГРАММНЫЕ СРЕДСТВА  
ДЛЯ ОРГАНИЗАЦИИ ДИАЛОГА  
НА ЯЗЫКЕ ВЫСОКОГО УРОВНЯ

*Направлено на Международный коллоквиум  
"Актуальные проблемы вычислительной техники"  
/Дрезден, 1979/*

ОИЯИ  
БИБЛИОТЕКА

Карлов А.А., Кирилов А.С., Смолякова Т.Ф.

P11 - 12160

Программные средства для организации диалога на языке высокого уровня

Рассматриваются структура и возможности комплекса программ, разработанных в ОИЯИ для удаленных дисплейных станций ЭВМ БЭСМ-6. Комплекс включает в себя диалоговый монитор, унифицированный механизм обработки сообщений, аппарат диалоговых переменных, библиотеку дисплейных подпрограмм, а также ряд специализированных подсистем для табличного и графического представления и анализа данных.

Пользователю предоставляются эффективные и удобные средства программирования диалога на языке высокого уровня.

Работа выполнена в Лаборатории вычислительной техники и автоматизации ОИЯИ.

Препринт Объединенного института ядерных исследований. Дубна 1979

Karlov A.A., Kirilov A.S., Smolyakova T.F.

P11 - 12160

The Means for Programming Dialogue in High Level Language

The structure and some possibilities of a set of programs developed at JINR for remote display stations of BESM-6 computer are considered. The set includes a dialogue monitor, a general purpose message handling package, a dialogue variable package, a display subroutine library and also a special-purpose subsystems for table and graphical data representation and analysis.

Effective and convenient facilities of programming dialogue through a high level language are given to user.

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

Preprint of the Joint Institute for Nuclear Research. Dubna 1979

## 1. ВВЕДЕНИЕ

Графические диалоговые системы представляют собой мощное средство для решения широкого класса задач в режиме активного взаимодействия пользователя с ЭВМ. Отсутствие в настоящее время удобных универсальных диалоговых языков обусловило в качестве основного подхода при разработке графических диалоговых систем распространение разного рода графических пакетов, которые ориентированы на пользователей, программирующих свои задачи на том или ином основном языке /например, ФОРТРАНе/. Однако наличие графических пакетов не решает всех проблем, связанных с организацией эффективного диалога на языке высокого уровня. Требуется разработка программных средств, которые, сохраняя все возможности работы на языке высокого уровня, обеспечивали бы в максимальной степени на системном уровне стандартные процедуры, связанные с организацией диалога.

Для удаленных дисплейных станций /УДС/ БЭСМ-6 ОИЯИ <sup>1/</sup> был создан комплекс программ, который позволяет достаточно эффективно и с минимальными затратами со стороны пользователя организовать диалог на языке высокого уровня /ФОРТРАНе/.

Этот комплекс включает в себя /рис. 1/:

- диалоговый монитор;
- унифицированный механизм обработки сообщений;
- аппарат диалоговых переменных;
- библиотеку дисплейных подпрограмм;
- ряд специализированных подсистем для табличного и графического представления и анализа данных.

Все функциональные возможности комплекса доступны пользователю, программирующему на языке высокого уровня.

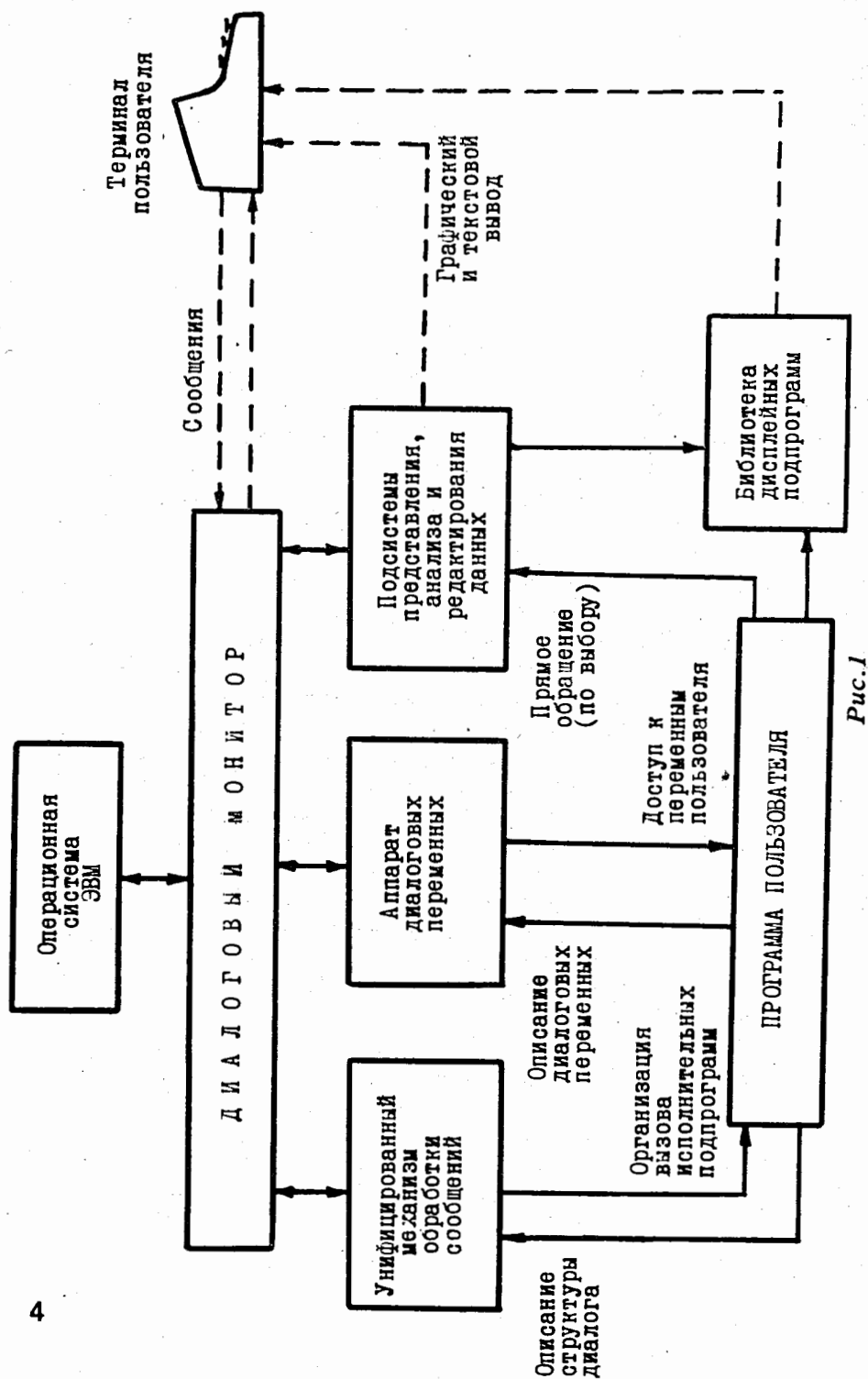


Рис.1

## 2. ДИАЛОГОВЫЙ МОНИТОР

Общее управление прохождением на ЭВМ задачи пользователя в режиме диалога, организация совместной работы отдельных блоков программного комплекса возлагаются на специальную управляющую программу - диалоговый монитор. Операционная система ЭВМ БЭСМ-6 ОИЯИ работает в режиме пакетной обработки<sup>2/</sup>, и с ее точки зрения диалоговый монитор является одной из задач пакета. Такое "подключение" диалоговой системы к существующему программному обеспечению ЭВМ БЭСМ-6 оказалось наиболее безболезненным, относительно быстрым и независимым.

Кроме того, на диалоговый монитор возлагается весьма ответственная задача по обработке особых ситуаций, в том числе вызванных ошибками пользователя. Наличие диалогового монитора, как своего рода "посредника" между операционной системой ЭВМ и программой пользователя, позволяет в случае ошибочных действий пользователя за терминалом, появления некорректных входных данных или запрещенных процедур обнаружить такую ситуацию и, не прекращая выполнения программы, информировать о ней пользователя с тем, чтобы он имел возможность исправить последствия, если это возможно, и продолжить работу.

## 3. УНИФИЦИРОВАННЫЙ МЕХАНИЗМ ОБРАБОТКИ СООБЩЕНИЙ

Унифицированный механизм обработки сообщений<sup>3/</sup> реализован в виде набора подпрограмм, доступных пользователю при программировании на языке ФОРТРАН. Его назначение - предоставить программисту гибкие и естественные в использовании программные средства для описания структуры диалога и организации взаимодействия пользователя с программой посредством сообщений, вводимых с терминала.

Основными особенностями предложенного механизма являются:

1. Возможность организации диалога с многоуровневой структурой.

2. Возможность изменения структуры диалога в процессе самого диалога.

3. Отсутствие необходимости в программировании запросов на ввод, операций обмена, синтаксического и, частично, семантического контроля сообщений.

4. Наличие аппарата стандартных диагностических сообщений, сопровождающих процесс диалога.

5. Наличие системных средств для выполнения стандартных процедур по управлению прохождением задач на центральной ЭВМ /перезапуск задачи, прекращение счета, хранение "истории" диалога с возможностью возврата на предыдущие его этапы и т.д./.

При описании структуры диалога пользователь в соответствии с требованиями конкретной задачи выбирает число уровней в структуре, число состояний на каждом уровне, для каждого состояния указывает связи между сообщениями /которые он выбирает по собственному усмотрению/ и исполнительными подпрограммами, а также описывает типы параметров этих подпрограмм.

В данном случае состояние диалога в некоторый момент времени рассматривается в ограниченном смысле: как набор допустимых в этот момент сообщений, а под числом уровней понимается максимально возможное число последовательно вложенных друг в друга циклов ожидания сообщений.

Структура диалога может быть задана программистом статически или динамически. При статическом задании она описывается в начале работы /например, с помощью отдельной подпрограммы/. При динамическом задании новый набор допустимых сообщений и соответствие между ними и исполнительными подпрограммами задаются в той подпрограмме, в которой происходит изменение состояния, благодаря чему, в частности, появляется возможность изменять структуру диалога в зависимости от промежуточных результатов работы, и требования к памяти, необходимой для описания структуры, уменьшаются.

Преимуществами статического задания является наглядность, удобство внесения изменений и дополнений, так как описание структуры может быть локализовано в одной или нескольких подпрограммах, специально предназначенных для этой цели. Естественно, полное описание структуры несколько увеличивает требования к объему памяти.

Возможность динамического описания структуры, кроме того, что она дает упомянутые выше преимущества, является еще и принципиально необходимой, так как позволяет подключать к программе пользователя специализированные диалоговые подсистемы обработки и анализа данных со своей собственной структурой диалога. Например, подсистема анализа функций двух переменных по отношению к программе пользователя может иметь свою собственную многоуровневую структуру диалога, которая по отношению к структуре диалога основной программы пользователя будет задаваться динамически при вхождении в подсистему.

При работе с унифицированным механизмом в качестве сообщения рассматривается последовательность символов, состоящая из приказа и его параметров, если таковые имеются. Приказы делятся на системные и приказы пользователя. Системные приказы предназначены для выполнения стандартных, не зависящих от решаемой задачи действий и отличаются наличием специального символа в первой позиции.

В качестве параметров сообщений могут быть заданы идентификаторы, числа и текстовые константы /как они определены в описании языка ФОРТРАН/.

По приказам пользователя с помощью рассматриваемого механизма осуществляется вызов указанных подпрограмм.

Набор допустимых в данный момент сообщений /приказов/ определяет текущее состояние диалога. Соответствие между отдельным сообщением и требуемой подпрограммой пользователь устанавливает через специальную таблицу связи подпрограмм /ТСП/ при описании структуры диалога. Таким образом, каждое состояние характеризуется своей ТСП.

Каждому приказу, допустимому в данном состоянии, в ТСП соответствует состоящий из трех слов элемент, первое слово которого содержит наименование /имя/ приказа, второй - имя исполнительской подпрограммы, а третья - описатель типа параметров исполнительской подпрограммы.

#### 4. ДИАЛОГОВЫЕ ПЕРЕМЕННЫЕ

Одним из недостатков недиалоговых языков программирования /в том числе языка ФОРТРАН/ является отсут-

ствии возможности простого доступа пользователя к переменным в своей программе с терминала. Необходимость такого доступа возникает в процессе диалога при редактировании начальных данных, анализе результатов вычислений, наблюдении за изменением переменных во время счета и т.п. В развитой диалоговой системе такой доступ должен обеспечиваться системными средствами и требовать от пользователя минимальных затрат на программирование.

Аппарат диалоговых переменных <sup>4/</sup> именно и предназначен для обеспечения доступа к переменным в программах, написанных на языке высокого уровня с использованием УДС БЭСМ-6.

Простые переменные и массивы, объявленные в качестве диалоговых в программе пользователя, становятся доступными для последующей обработки по тем именам, которые были присвоены им при объявлении их в качестве диалоговых. В результате оказывается возможным ограничивать простую и эффективную связь между прикладной программой пользователя и подпрограммами /подсистемами/ обработки, в том числе с использованием визуального представления информации для ее анализа и редактирования.

Простота заключается в том, что со стороны пользователя для реализации возможности работы с диалоговыми переменными требуются минимальные затраты на программирование. Он должен указать, какая память выделяется для учета диалоговых переменных, а также объявить, какие переменные и массивы в дальнейшем следует рассматривать как диалоговые.

Эффективность обеспечивается возможностью свободного /т.е.не программируемого заранее/ доступа к широкому набору подпрограмм и специализированных пакетов /как библиотечных, так и разработанных пользователем/, предназначенных для представления, анализа и редактирования данных. При работе с использованием унифицированного механизма обработки сообщений этот доступ осуществляется посредством системных сообщений или сообщений пользователя, параметрами которых могут быть имена диалоговых переменных.

Для учета переменных, объявленных диалоговыми, предусматривается *таблица диалоговых переменных /ТДП/*, расположение в памяти и размеры которой задаются поль-

зователем в зависимости от числа диалоговых переменных в его программе. При объявлении диалоговой переменной в программе пользователя в ТДП заносится элемент, который содержит имя указанной переменной, истинный адрес, указывающий ее расположение в памяти, а также другую вспомогательную информацию.

При задании ТДП информация о ее расположении и длине регистрируется в системных ячейках /описателе ТДП/ и впоследствии используется служебными программами для организации работы с ТДП /т.е. выполнения таких процедур, как занесение элементов, поиск диалоговых переменных по имени или адресу и т.п./.

Аппарат диалоговых переменных организован таким образом, что он предусматривает возможность динамического переопределения диалоговых переменных в процессе работы. Так, при подключении к программе пользователя специализированных диалоговых подсистем анализа и представления данных /как библиотечных, так и разработанных пользователем/ может оказаться, что в этих подсистемах определен свой собственный набор диалоговых переменных /например, в подсистеме анализа функции двух переменных в качестве диалоговых переменных могут быть заданы координаты точки наблюдения изображения поверхности, углы поворота изображения и другие параметры, изменение которых необходимо в процессе анализа функции/. Для сохранения информации о таблицах диалоговых переменных при вхождении в подсистему и ее восстановления при возврате предусмотрен системный стек /магазин/. Аппарат диалоговых переменных обеспечивает автоматическое сохранение и восстановление информации и диалоговых переменных при всех процедурах, связанных с переопределением набора диалоговых переменных.

Анализ диалоговых систем показывает, что одной из наиболее распространенных процедур является представление на экране той или иной группы начальных параметров задачи и редактирование их отдельных значений. Чтобы упростить для пользователей эту процедуру в структуре данных для диалоговых переменных предусмотрена возможность идентификации групп диалоговых переменных. В этом случае группе переменных присваивается номер,

по которому в дальнейшем можно выполнять различные операции над всей группой диалоговых переменных /представление на экране, выдача на печать и т.п./. Каждой группе диалоговых переменных соответствует элемент, в котором содержится информация о том, какие диалоговые переменные входят в данную группу.

Структура данных для диалоговых переменных показана на рис. 2.

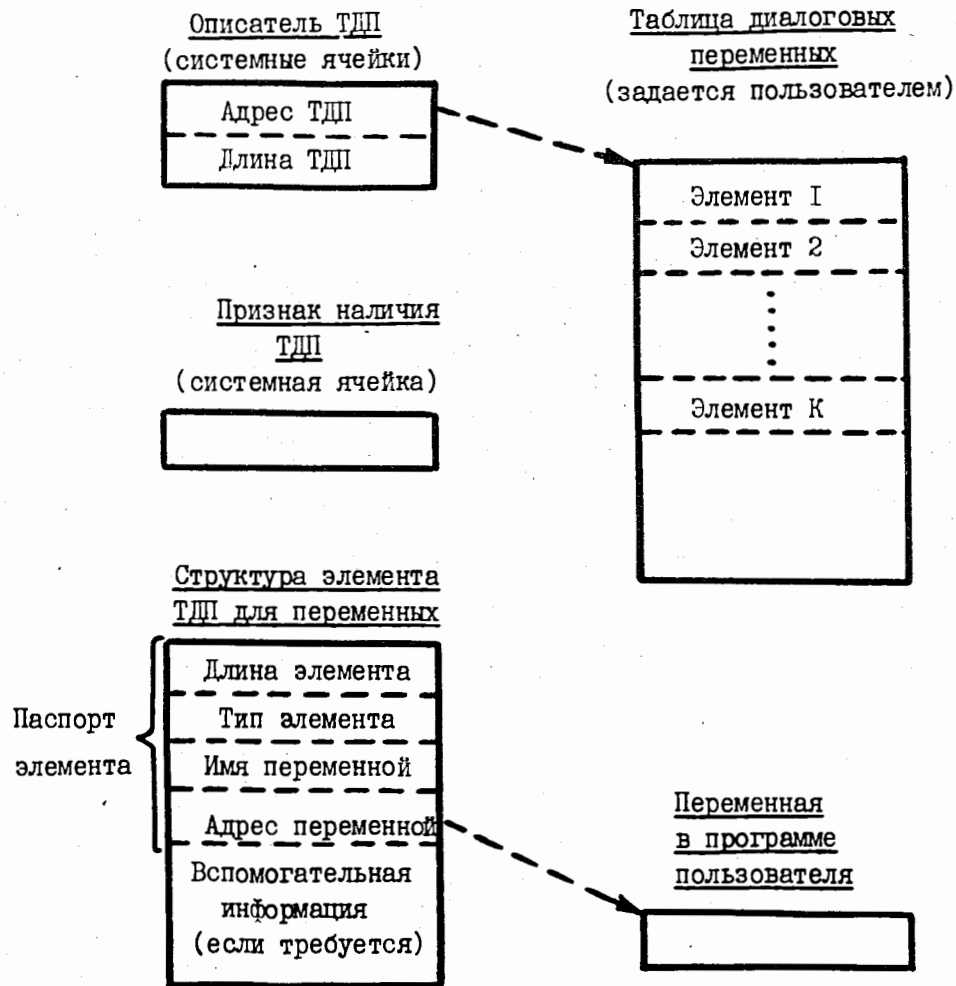


Рис.2. Структура данных для диалоговых переменных.

## 5. БИБЛИОТЕКА ДИСПЛЕЙНЫХ ПОДПРОГРАММ

Дисплейная библиотека общего назначения<sup>15/</sup> предоставляет пользователю широкий набор подпрограмм для формирования графической информации и организации управления работой УДС.

С функциональной точки зрения дисплейные подпрограммы можно разделить на три группы:

1/ подпрограммы-генераторы графической информации, с помощью которых формируется дисплейный файл. Эти подпрограммы обеспечивают возможность построения как простейших графических элементов /точки, векторы, дуги, текст/, так и более сложных графических объектов /оси, графики и т.п./;

2/ административные и вспомогательные подпрограммы, обеспечивающие выделение памяти под графическую и служебную информацию, а также выполнение всякого рода действий над графическими объектами /удаление, замещение, преобразование и т.п./;

3/ подпрограммы для организации взаимодействия с УДС, обеспечивающие обмен информацией между УДС и центральной ЭВМ, формирование запросов и специальных приказов для УДС и т.д.; эти подпрограммы позволяют пользователю рассматривать дисплейную станцию как некоторое дополнительное устройство ЭВМ БЭСМ-6, которое обладает набором достаточно сложных функциональных возможностей, доступных через подпрограммы данной группы. С их помощью пользователь организует работу с отдельными устройствами, входящими в состав УДС: световым карандашом, клавиатурой графического дисплея, накопителем на магнитной ленте, символьным дисплеем "Видеотон-340" и др.

Использование подпрограмм библиотеки полезно особенно в тех задачах пользователя, где необходимы средства для формирования специальной графической или символьной информации и требуется организация нестандартных процедур диалога, характерных для конкретной задачи.

Напротив, наиболее распространенные способы представления и анализа графической информации /в виде таблиц, графиков, гистограмм и т.п./, а также часто

используемые процедуры /например, редактирование данных/ должны обеспечиваться в развитой диалоговой системе программными средствами самой диалоговой системы и требовать минимальных затрат со стороны программиста. Это может быть достигнуто за счет наличия в составе программного обеспечения диалоговой системы специализированных подсистем представления, анализа и редактирования данных.

## 6. ПОДСИСТЕМЫ ПРЕДСТАВЛЕНИЯ, АНАЛИЗА И РЕДАКТИРОВАНИЯ ДАННЫХ

Принципиальное отличие диалоговой подсистемы как самостоятельной единицы от пакета /набора/ подпрограмм состоит в том, что от пользователя не требуется предусматривать /т.е. программировать/ в своей задаче необходимость обращения к подсистеме, заботиться о структуре диалога и всякого рода вспомогательных действиях при работе с подсистемой, которые в случае работы с набором подпрограмм неизбежны и требуют от пользователя дополнительных, иногда весьма ощутимых затрат, непосредственно не связанных с программированием его основной задачи.

Реализация унифицированного механизма обработки сообщений и аппарата диалоговых переменных обеспечивает простое подключение подсистемы на любом этапе в процессе диалога путем ввода соответствующего системного сообщения.

Например, подключение к программе пользователя подсистемы SIGN, предназначенной для графического представления и анализа одномерных числовых массивов /8/, осуществляется по системному сообщению:

/A, SING

В результате подсистеме становятся доступны все диалоговые переменные, существующие в момент ее подключения. Используя командный язык, предлагаемый подсистемой, пользователь за терминалом может проанализировать данные /которые заданы одномерными массивами и были объявлены в качестве диалоговых/, представляя

их в виде кривых, гистограмм и т.п. Например, если X и Y - одномерные массивы, то по сообщению

GRF, Y, X

на экране УДС будет построен график функции  $Y(X)$  с автоматическим масштабированием, выбором рабочей области экрана, разметкой координатных осей и т.п.

Широкое применение правила умолчания при работе с подсистемами позволяет работать с подсистемой и получать удовлетворительные результаты с помощью всего нескольких команд, что особенно привлекательно для начинающих пользователей. Опытным пользователям каждая подсистема предоставляет развитый командный язык для доступа ко всем возможностям данной подсистемы.

Иногда целесообразно иметь возможность и прямого обращения к подсистеме из программы пользователя, осуществляя связь программы с подсистемой через параметры обращения /см. рис. 1/. Однако в этом случае подсистема может рассматриваться скорее как сложная подпрограмма /а не самостоятельная единица/, допустимые моменты обращения к которой точно определены алгоритмом программы пользователя и поэтому должны быть предусмотрены заранее.

## 7. ЗАКЛЮЧЕНИЕ

Наличие развитых системных средств для организации сложных диалоговых структур, возможность простого доступа с терминала к переменным в программе пользователя и возможность подключения на различных этапах диалога специализированных диалоговых подсистем для графического представления и анализа данных позволили успешно использовать данный программный комплекс при решении ряда задач экспериментальной и теоретической физики /7,8/, а также молекулярной биологии /9/.



## ЛИТЕРАТУРА

1. Кавченко А.В. и др. ОИЯИ, 10-9325, Дубна, 1975.
2. Мазный Г.Л. Программирование на БЭСМ-6 в системе "Дубна". "Наука", М., 1978.
3. Карлов А.А., Смолякова Т.Ф. Труды Второй конференции по вычислительной науке, часть II, Будапешт, 1977, с. 568-578. См. также ОИЯИ, P11-10440, Дубна, 1977.
4. Карлов А.А., Смолякова Т.Ф. Труды Совещания по программированию и математическим методам решения физических задач /Дубна, 20-23 сент. 1977 г./ ОИЯИ, Д10,11-11264, Дубна, 1978, с. 485-493.
5. Карлов А.А. и др. ОИЯИ, Б1-10-11999, Дубна, 1978.
6. Карлов А.А., Кирилов А.С. ОИЯИ, P11-12133, Дубна, 1979.
7. Дерендяев Ю.С. и др. ОИЯИ, Б1-11-9982, Дубна, 1976.
8. Александров А.В. и др. ОИЯИ, 10-11842, Дубна, 1978.
9. Карлов А.А. и др. ОИЯИ, P11-9881, Дубна, 1976.

Рукопись поступила в издательский отдел  
5 января 1979 года.