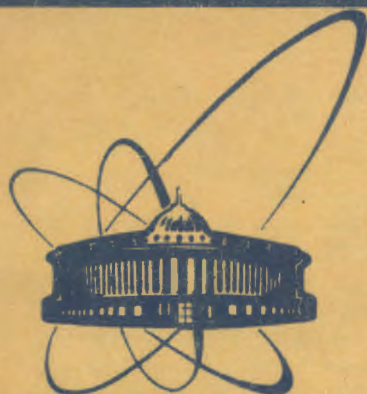


12960



сообщения
объединенного
института
ядерных
исследований
дубна

1866/2-80

21/4-80

P10 - 12960

Г.Балука, Г.П.Жуков, Ю.Намсрай, Ю.М.Останевич,
А.И.Островной, А.С.Савватеев, И.М.Саламатин,
Г.Я.Яновский

КОМПЛЕКС СРЕДСТВ

ДЛЯ ГЕНЕРАЦИИ ПРИКЛАДНЫХ СИСТЕМ

АВТОМАТИЧЕСКОГО НАКОПЛЕНИЯ

И ПРЕДВАРИТЕЛЬНОЙ ОБРАБОТКИ ДАННЫХ -

САНПО

1980

1. ВВЕДЕНИЕ

Развитие средств автоматизации научных исследований на основе применения мини-ЭВМ привело к концентрации в исследовательских центрах при уникальных^{/1,2/} и дорогостоящих базовых установках вычислительной техники, используемой в режиме работы ЭВМ на линии с экспериментальной аппаратурой. Программное обеспечение таких вычислительных машин /обычно мини-ЭВМ/ ориентировано на решение конкретных задач с учетом имеющегося экспериментального оборудования и конфигурации ЭВМ. В зависимости от конкретной методики эксперимента в результате соответствующих разработок возникает более или менее сложный комплекс программ, а зачастую создается целая специализированная система для одного эксперимента^{/3,4,5/}.

Создание и развитие систем для каждого конкретного эксперимента - трудоемкая задача, при решении которой разработчики сталкиваются с рядом проблем, известных по опыту создания систем пакетной обработки и систем реального времени^{/6,7/}. Поэтому в условиях исследовательской организации естественно возникает необходимость создать систему программ, которая в рамках интересующей проблемной области обладала бы достаточной гибкостью по отношению к составу, конфигурации оборудования и методике эксперимента^{/8,9,10/}.

Задачам реального времени присущи особенности, отличающие их от задач вычислительного характера. Основной из них является параллельное исполнение связанных процессов. Эта особенность не обеспечена необходимыми для программирования средствами в распространенных языках высокого уровня^{/11/}. Известны предложения включить в языки для программирования задач реального времени средства синхронизации параллельных процессов в виде операций с семафорами^{/12,13/}.

В данном сообщении предлагаются средства параллельного программирования, не требующие описания конкурирующих процессов на уровне операций с семафорами, описывается структура прикладной системы реального времени и комплекс средств, предназначенных для генерации систем автоматического накопления и предварительной обработки данных - САНПО.

Результаты данной работы предполагается применить при программировании экспериментов на импульсном быстром реак-



торе^{1,2/}. Комплекс программных средств находится в стадии реализации для машин с системой команд СМ-3^{14,15,16/}.

2. СОСТАВ И НАЗНАЧЕНИЕ КОМПЛЕКСА

Описываемый комплекс средств включает обобщенную систему САНПО и средства генерации.

Назначение этого комплекса - на основании составленного пользователем описания ориентировать обобщенную САНПО на конкретный эксперимент, иначе говоря, генерировать прикладную САНПО. Комплекс предназначен для экспериментаторов. Для использования средств генерации, в частности языка генерации программного обеспечения эксперимента, не требуется специальных знаний из области системного программирования.

3. ОБОБЩЕННАЯ СИСТЕМА

Под обобщенной САНПО мы понимаем проблемно-ориентированный комплекс системных модулей, подсистем и стандартных программ /СП/, не зависящих от методики конкретного эксперимента, реализующих определенную дисциплину работы с наборами данных и СП системы, а также набор правил написания и работы СП и подсистем САНПО, организации базы данных /БД/ системы и дисциплины ее работы.

Обобщенная САНПО является информационной базой для средств, генерирующих прикладные системы. Набор системных модулей и СП хранится в библиотеках системы. Организация библиотеки подробно описана в работе^{17/}.

В составе этой системы можно выделить три основные группы программ: программные модули монитора, подсистемы и стандартные программы.

Модули монитора инвариантны относительно методики конкретного эксперимента, состава экспериментального оборудования и конфигурации ЭВМ.

Подсистемы являются средством, обеспечивающим взаимодействие прикладных систем с внешней средой - оборудованием и человеком.

Стандартные программы - минимальные структурные части системы. Конкретное содержание библиотек вместе с дисциплиной работы, реализуемой модулями монитора, определяет проблемную ориентацию обобщенной САНПО.

Функции, взаимодействие и логику работы модулей, включенных в обобщенную систему, удобнее пояснить путем описания прикладной системы.

4. ПРИКЛАДНАЯ СИСТЕМА

4.1. Общее описание. Прикладная САНПО является системой для конкретного эксперимента, компонуемой из модулей обобщенной системы на основании описания на языке САНПО, функции и специализация которой определяются описанием и обеспечиваются монитором, подсистемами и прикладной библиотекой стандартных программ, а также соответствующим построением программы эксперимента.

Прикладная САНПО для конкретного эксперимента обеспечивает выполнение следующих основных функций:

- 1/ прием и накопление экспериментальных данных;
- 2/ обработку данных по заданному алгоритму и их хранение;
- 3/ управление экспериментальной установкой и процедурой проведения эксперимента;
- 4/ обмен информацией с пользователем /экспериментатором/;

Прием и накопление данных допускается по нескольким независимым асинхронным каналам, работающим одновременно.

Обработка данных может включать предварительную их обработку, использование ресурсов удаленных машин, доступных по линии связи, занесение данных в архив для долговременного хранения и другие операции.

Управление экспериментальной установкой и прикладной системой осуществляется путем выполнения операций системы в соответствии с заданными значениями параметров. Формально операции управления не отличаются от операций обработки.

Эти функции САНПО может выполнять в двух режимах:

- 1/ в режиме автоматического исполнения заданного алгоритма эксперимента без вмешательства человека.
- 2/ в интерактивном режиме, реализуемом при желании одновременно с автоматическим исполнением программы эксперимента.

В структуре прикладной системы можно выделить следующие основные части: ядро монитора, оперативную базу данных /ОБД/ и резиденты подсистемы. Помимо этого неотъемлемой частью прикладной системы является набор библиотечных СП. Ядро монитора и резиденты подсистем составляют резидентную часть САНПО. Набор модулей монитора совместно с подсистемами и СП определяет функциональные возможности системы. ОБД и программа эксперимента описывают область приложения и превращают систему в прикладную.

Элементарная операция системы, соответствующая оператору языка САНПО, реализуется стандартной программой. Несколько операторов могут быть объединены в макрооператор - операцию более высокого уровня /см. рис. 1/. Еще более мощным опера-



Рис. 1. Иерархия операционных средств.

ционным средством являются подсистемы, которые используют в работе макрооператоры и СП. Монитор /и интерпретатор, являющийся его частью/ организует работу и взаимодействие средств нижних уровней - подсистем, макросредств и СП.

На рис. 2 изображена структура прикладной системы. Рассмотрим компоненты этой системы.

4.2. Монитор является основной управляющей программой прикладной системы. Он управляет оперативной базой данных и всеми работами системы.

В состав монитора входят: ядро монитора, интерпретатор, программа динамического распределения памяти /ДРП/ и загрузчик программ, хранящихся в библиотеке в двоичном перемещаемом виде. Эти модули обеспечивают:

- 1/ управление потоком экспериментальных данных;
- 2/ синхронизацию параллельных процессов регистрации, накопления и предварительной обработки экспериментальной информации и управление ими;
- 3/ интерпретацию и исполнение программы эксперимента, а также приказов, введенных с клавиатуры телетайпа в интерактивном режиме;
- 4/ загрузку в память ЭВМ СП и подсистем и обслуживание их работы;
- 5/ доступ СП и подсистем к данным в ОБД по имени.

Монитор позволяет организовать работу подсистем в двух режимах. Один режим обеспечивает исполнение автоматически функционирующих процессов /определенных схемой ОБД/ с более высоким приоритетом, чем операции подсистем. Другой режим позволяет подсистемам монополизировать все программные ресурсы системы.

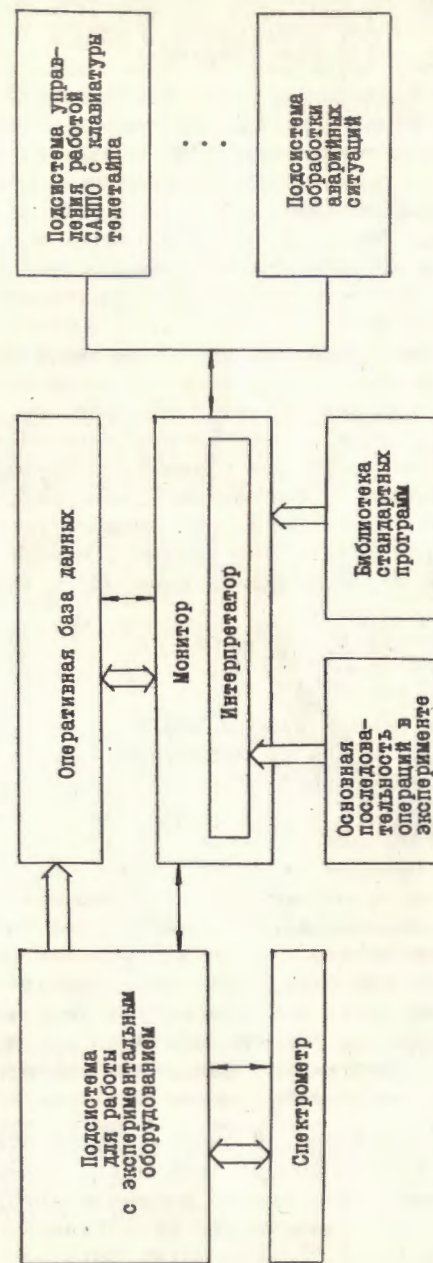


Рис. 2. Структура прикладной системы.

Помимо этого, монитор выполняет специальные действия для обеспечения надежности системы, предотвращения искажения данных и потерь информации при авариях.

4.3. Оперативная база данных. Разработанную ОБД^{/18/} отличают от традиционных БД^{/19/} следующие особенности:

- 1/ в качестве элементов данных /ЭД/ в ОБД могут выступать буферные участки памяти, предназначенные для заполнения экспериментальными данными в реальном масштабе времени;
- 2/ элементы данных ОБД могут располагаться как в оперативной памяти, так и на любых внешних носителях, подключенных к данной ЭВМ, а также в памяти, доступ к которой обеспечивают связи с другой ЭВМ;
- 3/ иерархические отношения между ЭД соответствуют в ОБД функциональным отношениям.

Функциональные отношения отражают описываемые пользователями процессы обработки данных и представляются в виде схемы процессов. Схемы процессов совместно с описанием ЭД и индикаторами состояния ЭД составляют схему ОБД.

ОБД описывается на языке САНПО для каждого конкретного эксперимента. При этом могут быть учтены особенности экспериментального оборудования, конфигурации ЭВМ и методики эксперимента.

Данные в ОБД хранятся в виде чисел типа INTEGER и REAL однократной и двойной точности. Числа содержатся в буферах либо являются значениями переменных.

ОБД заполняется в режиме работы ЭВМ на линии с экспериментальной установкой, и для ее эффективной работы предполагается использование долговременного хранилища данных - долговременной БД /ДБД/. В качестве ДБД может быть выбран архив, файловая система или БД на другой ЭВМ. Во взаимодействии ОБД и ДБД обеспечивается специальными СП. Информация из ОБД в ДБД может поступать во время эксперимента. Такая передача может быть инициирована автоматически по окончании какой-либо фазы эксперимента, после поступления заданного количества информации, по окончании эксперимента либо в интерактивном режиме по приказу с клавиатуры телетайпа. Передаваемая информация для ее последующей идентификации может сопровождаться характеристиками, заданными экспериментатором, и характеристиками, выработанными системой.

4.4. Стандартные программы. В системе стандартными называются программы со строго определенными способами входа, выхода, обмена данными, реализующие заданный алгоритм.

СП программируются и отлаживаются средствами дисковой операционной системы RT-11^{/20/}. Они могут быть написаны на

языках MACRO-11 или FORTRAN-4. После отладки СП, помещенные в двоичном перемещаемом формате REL^{/21/} в библиотеку САНПО, доступны в виде операций прикладной системы.

Разработанная структура и способ управления обеспечивают стандартным программам следующие свойства:

- 1/ СП не зависят от физического места расположения параметров /элементов данных ОБД/;
- 2/ СП не зависят от длины обрабатываемых массивов данных;
- 3/ одни и те же СП могут быть инициированы как в автоматическом, так и в интерактивном режимах работы САНПО;
- 4/ СП могут использоваться в режиме динамического распределения памяти;
- 5/ одни и те же СП без изменения могут применяться в различных прикладных системах.

4.5. Подсистемы САНПО, как правило, состоят из резидентной части, которая загружается на ДРП по требованию и фиксируется, и набора СП, реализующих отдельные операции подсистемы.

Для каждого эксперимента необходимы подсистемы:

- 1/ для работы с экспериментальным оборудованием;
- 2/ управления работой САНПО с клавиатуры телетайпа;
- 3/ обработки аварийных ситуаций.

САНПО может пополняться новыми как специализированными, так и универсальными подсистемами. Существенно расширяют возможности, например, следующие подсистемы: подсистема для обеспечения работы СП, написанных на ФОРТРАНе; подсистема для работы с дисплеем; тестирования оборудования; обеспечения справочных запросов; подготовки пакета заданий на эксперимент и сопутствующей информации; обеспечения работы фоновой задачи и другие.

4.5.1. Средства для работы с экспериментальным оборудованием в настоящее время ориентированы на оборудование в стандарте КАМАК^{/22/} и его тестирование.

Резидент этой подсистемы компонуется из стандартных программ и загрузочных модулей, настраиваемых в соответствии со способом размещения оборудования и адресами загрузки программ. Назначение резидента - диспетчеризация и обработка прерываний, накопление данных в одном или нескольких буферных участках памяти. Помимо этого, резидент содержит программы управления состоянием, в частности активностью оборудования, имеющие фиксированный способ входа и выхода. Эти программы управляют, в конечном счете, потоком данных в систему.

Операции с оборудованием, включенные в программу эксперимента или инициируемые в интерактивном режиме, реализуются стандартными программами.

4.5.2. Подсистема управления с клавиатуры телетайпа выполняет функции управления прикладной системой и ее контроля в интерактивном режиме.

Подсистема осуществляет прием приказов с клавиатуры телетайпа, их интерпретацию и исполнение. Часть приказов является инструкциями языка САНПО, а часть обеспечивает вспомогательные операции, включая средства комплексной отладки прикладных систем и отдельных СП, средства для справок и др.

Монитор позволяет работать с данной подсистемой в двух режимах. Первый реализует исполнение всех автоматически функционирующих процессов эксперимента, а оставшееся время процессора используется для обработки интерактивных приказов. Другой режим позволяет вмешаться в работу системы сразу после завершения исполняемой операции. Доступ пользователя к каждому из режимов может быть защищен паролем.

4.5.3. Подсистема обработки аварийных ситуаций предназначена для автоматизации поиска причины аварии, принятия решения и реализации алгоритма, направленного на сохранение максимума достоверной информации и работоспособности программного обеспечения эксперимента.

После аварии можно определить состояние данных и выделить формально достоверную информацию, а также определить состояние системы в момент аварии с точностью до минимальной операции системы. Информацию об авариях подсистема может накапливать на машинных носителях и визуализировать.

В благоприятных ситуациях после аварии подсистема может возобновить работу системы без потери информации, возможно, повторив прерванную операцию, либо изменив программу эксперимента путем предусмотренного отказа от второстепенного оборудования или используя резервное оборудование.

4.6. Программа эксперимента. Программа эксперимента отображается в основной последовательности операций эксперимента /ОПЭ/ и схеме процессов обработки данных /СПО/.

4.6.1. ОПЭ - это программа, представленная в виде последовательности инструкций языка высокого уровня - языка САНПО. Эта программа содержит инструкции для выполнения операций управления состоянием оборудования, состоянием САНПО, а также операции обработки данных.

В числе операций управления оборудованием имеются операции подготовки оборудования /спектрометра/ к работе, начала, продолжения, остановки эксперимента и другие. При управлении состоянием системы допускается изменение ее функций. Параметрами инструкций обработки данных должны быть элементы БД или константы.

ОПЭ хранится в библиотеке на внешнем запоминающем устройстве /ВЗУ/ и исполняется в режиме интерпретации. На ее длину не наложено ограничений. Допускается редактирование ОПЭ, при этом разрешено изменение состава операторов, набора параметров и значений констант. После редактирования ОПЭ не требуется компиляции, пока и поскольку в отредактированном тексте ее используются идентификаторы только ранее описанных элементов данных.

4.6.2. Для описания эксперимента, включающего обработку нескольких потоков информации, поступающей по одновременно работающим каналам ввода/вывода, предложены дополнительные средства представления алгоритмов. Предполагается, что потоки независимы, для каждого требуется свой процесс /последовательность операций/ обработки, процессы эти могут быть связанными и должны исполняться одновременно. В данной системе программу исполнения ряда параллельных процессов представляет конструкция, названная схемой процессов обработки данных. Эта конструкция не является традиционной, для ее пояснения введем понятия элементарного процесса и процесса.

Элементарный процесс /ЭП/ - это операция системы /одна или несколько СП/, ассоциированная с элементом данных ОБД и средствами программной индикации состояния данных.

Назначение элементарного процесса - обработать информацию, переданную в ассоциированном ЭД. Иницируется ЭП специальным сигналом о готовности данных в соответствующем ЭД к обработке. В результате обработки одного ЭД могут быть выработаны сигналы о готовности к обработке других ЭД и тем самым иницированы новые ЭП.

Процесс - совокупность функционально связанных элементарных процессов с определенными для них приоритетными отношениями, которая может быть отражена в связанном списке.

Процесс представляет собой граф автоматического преобразования в реальном масштабе времени информации, поступающей в режиме работы ЭВМ на линии с экспериментальной установкой.

Параллельность исполнения процессов реализована путем их переключения в соответствии с приоритетами операций. Программное прерывание одного процесса другим в данной реализа-

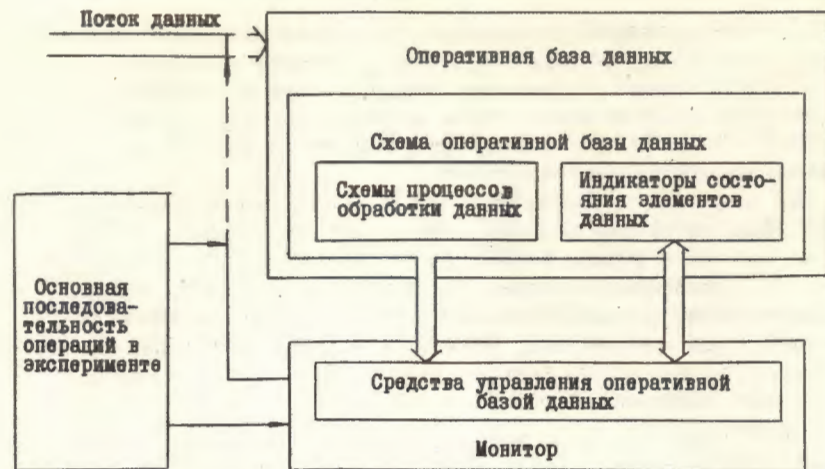


Рис. 3. Взаимодействие основной последовательности операций в эксперименте и ОБД.

ции может быть выполнено после завершения текущей операции системы.

Схема процессов обработки - это та часть таблиц ОБД, в которой заданы функциональные связи между элементарными процессами и приоритетные отношения операций над элементами данных ОБД.

На рис. 3 показано взаимодействие ОПЭ и ОБД. Схема процессов обработки данных является программой виртуальной машины, работа которой управляется потоком данных. Виртуальная машина представлена средствами управления ОБД в мониторе и базой данных. В качестве единицы программирования схемы процессов выступает элементарный процесс. Управляющее воздействие потока данных приводит к инициированию очередного элементарного процесса. Управление потоком данных осуществляет монитор и ОПЭ. В ведении монитора находятся операции управления потоком на уровне элементарных процессов, а в ведении ОПЭ - на уровне процессов и эксперимента в целом.

4.6.3. Данная организация программы эксперимента позволяет определять или менять значения параметров, управляющих процедурой эксперимента /и обработкой/, рядом сплосбов: при описании ОПЭ и СПО, по результатам обработки данных, в диалоговом режиме по запросам программ, в процессе редактирования ОПЭ, путем программируемой замены пакета используемых значений новым из числа запасенных.

5. СРЕДСТВА ГЕНЕРАЦИИ ПРИКЛАДНЫХ СИСТЕМ

К средствам генерации прикладных систем отнесены:

- 1/ язык САНПО;
- 2/ компилятор;
- 3/ компоновочные программы;
- 4/ программы для работы с библиотеками САНПО.

На рис. 4 показана схема процедуры генерации. Рассмотрим назначение используемых средств более подробно.

5.1. Язык САНПО является проблемно-ориентированным языком высокого уровня, предназначенным для описания прикладных систем, а также для управления их работой в интерактивном режиме.

В числе инструкций языка можно выделить декларативные и исполняемые инструкции.

Декларативные инструкции используются для распределения памяти, определения элементов базы данных и описания состава аппаратных и программных ресурсов. Эти инструкции обрабатываются компилятором.

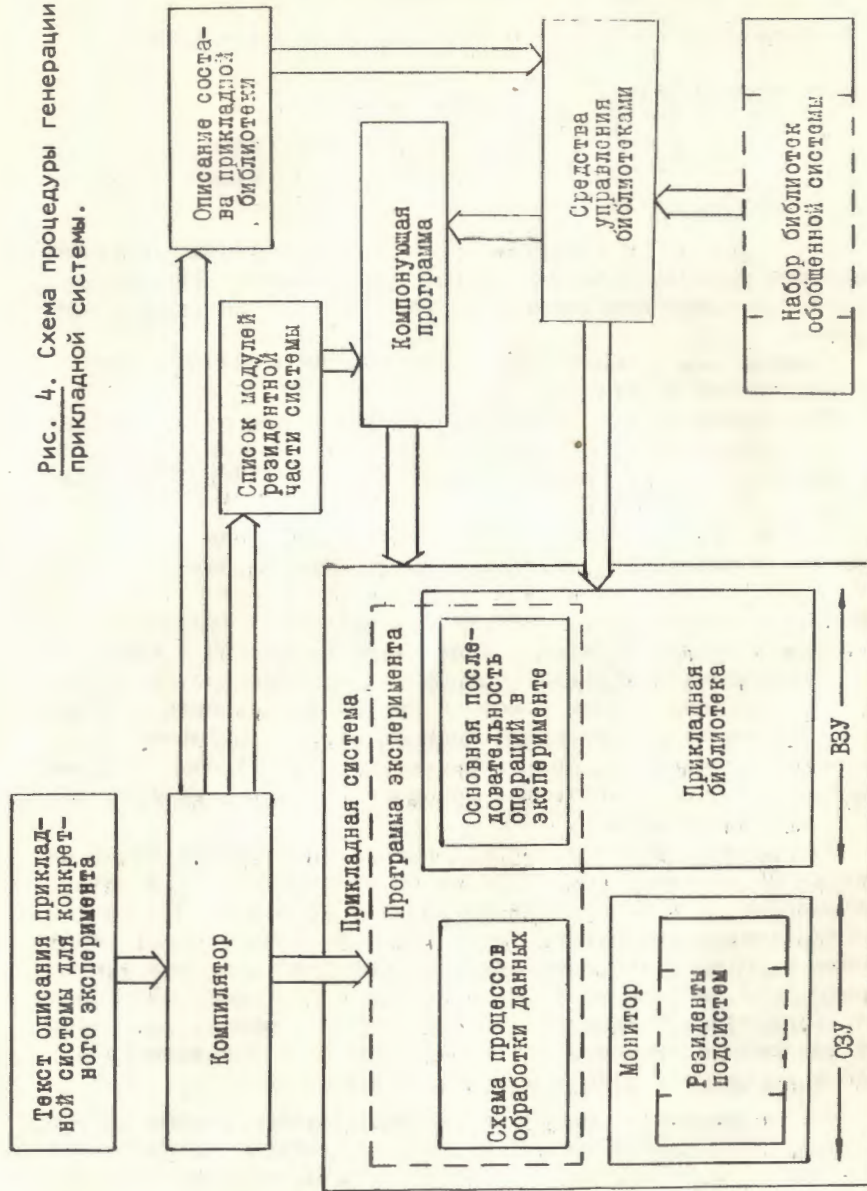
Особая группа декларативных инструкций языка позволяет описать конструкцию, названную процессом. Совокупность инструкций описания процессов обрабатывается совместно с включенными в процесс исполняемыми инструкциями. Результат этой обработки представляется в виде схемы процессов в схеме ОБД.

Исполняемые инструкции преобразования передаются в прикладную систему в виде основной последовательности операций в эксперименте для интерпретации на этапе исполнения программы эксперимента. Для программирования ОПЭ язык содержит инструкции, соответствующие основным конструкциям структурного программирования^{23,24}.

Программой реализацией исполняемых инструкция языка являются СП системы. Одни и те же СП используются и в автоматическом, и в интерактивном режимах работы САНПО. Часть декларативных инструкций также может выполняться интерпретатором в интерактивном режиме и служит для изменения функций прикладной системы на этапе исполнения программы эксперимента. Содержание библиотеки СП обеспечивает ориентацию языка на конкретную проблему или конкретный круг пользователей. Набор инструкций открыт и может пополняться.

5.2. Компилятор обрабатывает описание прикладной системы, которое представляется в виде последовательности инструкций языка. На основании этого описания компилятор создает схему ОБД, программу эксперимента /в виде ОПЭ и СПО/, а также вырабатывает информацию для программ, компоновочных системы. Эта информация включает список модулей резидента систе-

Рис. 4. Схема процедуры генерации прикладной системы.



мы и список СП и библиотек для формирования прикладной библиотеки.

5.3. Компоновку прикладной системы по заданным описаниям ее компонентов выполняет специальная программа. Эта программа предназначена для изготовления на машинных носителях двух версий САНПО, дисковой и перфоленточной, пригодных к эксплуатации.

Для дисковой версии формируются резидент системы в формате SAV^{/21/}, библиотечный файл, содержащий необходимые СП в формате REL^{/21/} и другие загрузочные модули.

Для перфоленточной версии все необходимые программы объединяются с монитором и на указанном носителе формируется система в формате абсолютной загрузки LDA^{/21/}.

Компонуемая программа использует средства для работы с библиотеками.

5.4. Средства для работы с библиотеками САНПО включают файловую систему стандартной операционной системы и программу "Библиотекарь".

В данной работе использовалась файловая система операционной системы RT-11^{/21/}.

Библиотека САНПО представляется в виде файла, имеющего определенную внутреннюю структуру^{/25/}. Элементами этой структуры являются СП и другие загрузочные модули. Содержание библиотеки и характеристики модулей определены в каталоге.

Программа "Библиотекарь" создана в двух версиях: для интерактивного^{/25/} и автоматического режимов работы. В средствах генерации используется вторая из версий. Эта версия выбирает управляющую информацию из специального файла и допускает:

- 1/ создание новых и редактирование старых библиотек;
- 2/ операции объединения библиотек;
- 3/ создание новых библиотек из старых с содержанием, заданным в виде списка модулей;
- 4/ распечатку ряда характеристик модулей, содержащихся в библиотеке.

6. ПРИМЕР ОПИСАНИЯ ПРИКЛАДНОЙ СИСТЕМЫ

6.1. Схема прикладной части системы, предназначенной для регистрации одномерных и двумерных спектров по программному каналу, приведена на рис. 5. Слева показаны блоки экспериментального оборудования, в данном примере - блоки в каркасе КАМАК.

24	КОНТРОЛЕР
23	КАНАЛА
22	ЛАМПА
21	...
20	КОДИРОВЩИК
19	АК1
18	СЧЕТЧИК
17	С1
16	...
15	...
14	КОДИРОВЩИК
13	ВК
12	КОДИРОВЩИК
11	АК2
10	СЧЕТЧИК
9	С2
8	...
7	...
6	...
5	...
4	СЧЕТЧИК
3	С3
2	...
1	...

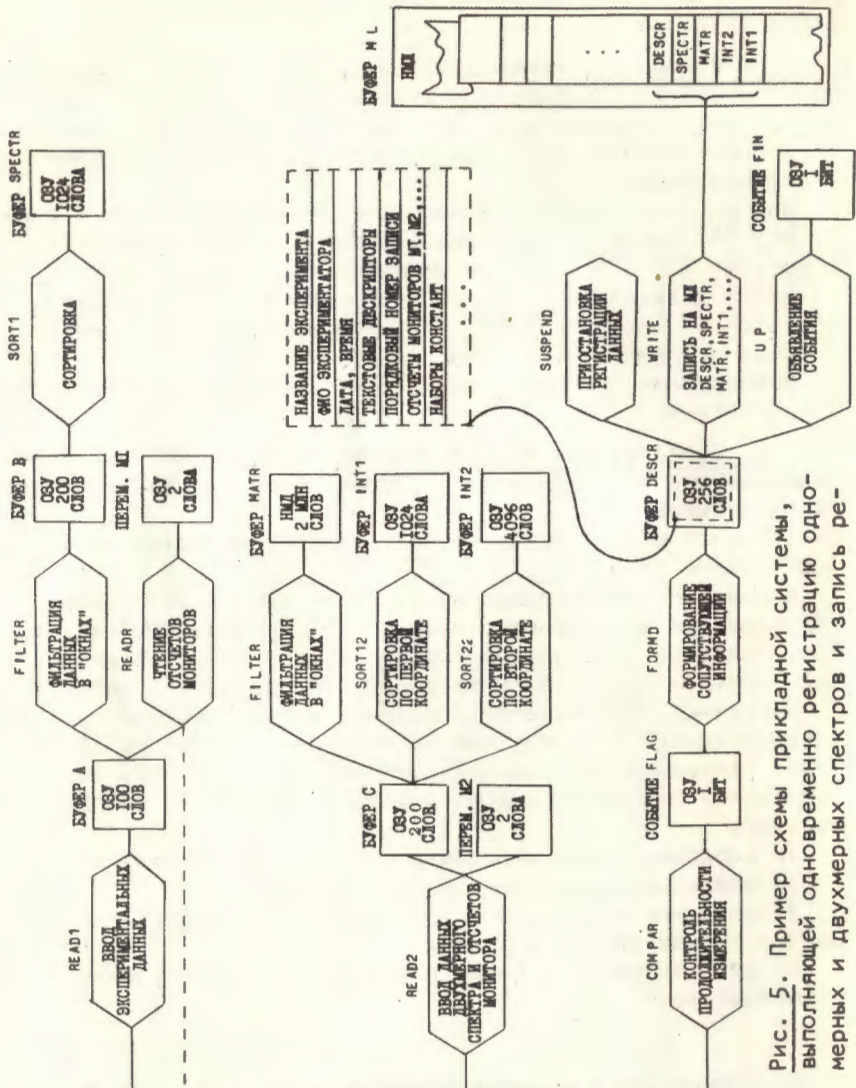


Рис. 5. Пример схемы прикладной системы, выполняющей одновременно регистрацию одномерных и двумерных спектров и запись результатов эксперимента на магнитную ленту.

Назовем каналом группу блоков /возможно, один блок/, которые предполагается обслуживать одной СП. На рисунке выделено 4 канала.

По способу взаимодействия программы и оборудования мы различаем пассивные и активные каналы. Под пассивным каналом /в данном примере - счетчик С1/ будем понимать канал, по которому ввод /вывод/ информации производится только по инициативе со стороны ЭВМ. Три канала /АК1, АК2, ВК, С2; С3/ являются активными, т.е. оборудование может генерировать сигнал запроса прерывания и соответствующие программы инициируются по этому сигналу. На рис. 5 от программ к связанным с ними каналам проведены линии /к активным - сплошные, к пассивным - пунктирные/.

Элементы данных, заполняемые программами обслуживания активных каналов, стоят в вершинах структур из связанных операций и элементов данных. Такая структура графически отображает часть программы, названную выше в тексте процессом. В данном случае используются три процесса. Первый из них связан с каналом АК1 /аналого-цифровым преобразователем/ и выполняет регистрацию данных и формирование одномерного спектра. Второй процесс предназначен для регистрации интегральной информации спектров по обоим осям. Помимо этого, в обоих процессах считываются отсчеты /значения/ М1 и М2 мониторных счетчиков С1 и С2. Третий процесс контролирует продолжительность отдельного сеанса измерения, а по истечении заданного лимита времени приостанавливает регистрацию данных и выводит их на магнитную ленту. Смысл отдельных операций, использованных для построения каждого из процессов, ясен из надписей на рисунке. Такое сочетание процессов следует рассматривать как схему системы, предназначенной для одновременного выполнения нескольких экспериментов одним экспериментатором. В данном случае таких экспериментов два: в одном измеряется одномерный спектр, в другом - двумерный. Объединение нескольких процессов в одном эксперименте ведет к повышению эффективности использования базовой установки, экономии оборудования, повышению точности результатов за счет устранения влияния неодновременности выполнения основных /двухмерный спектр/ и калибровочных /одномерный спектр/ измерений и т.п.

6.2. Текст описания на языке САНПО системы, изображенной на рис. 5, представлен на рис. 6.

В тексте пронумерованными пунктирными линиями выделены 4 части. Часть 1 ограничена операторными скобками SUBSYS-ENDH и содержит текст описания типа экспериментального

```

// ----- 1
/SUBSYS CAMAC, TYPE NE ; SUBSYSTEMS NAME AND TYPE OF HARDWARE
/DEVICE 1,1: 1-23=170 ; BRANCH, CRATE, INTERRUPT VECTORS
/LET AK1=21, BK=14, AK2=10
/LET C1=19, C2=8, C3=4 ; STATION NUMBERS
/HCASE AK1:AK1 D0 HEAD(A) ; ACTIVE CHANNEL
/ATTACH C1 TO HEADR(M1) ; PASSIVE CHANNEL
/HCASE AK2:BK,AK2,C2 D0 HEAD(C)
/HCASE C3:C3 D0 COMPAR(LIMIT,FLAG)
/END HARDWARE DESCRIPTION
// ----- 2
/FIELD#512 I RK1(EXP.DAT=2000000) ; STORAGE RESOURCES
/INTEGER BUFFER A(100), B(200), SPECTR(1K), INT1(1K), INT2(4K)
/DOUBLE INTEGER VARIABLE M1, M2
/EVENT FLAG, FIN
/INT BUF DESCR(256), MATH(1:2000000), C(400B)
/VARIABLE LIMIT=20000,N=0,N1,K1,N2,K2,F1,L1
// ----- 3
/CASE A D0
  FILTER(A,B,F1,L1)
  HEAD(M1)
/END CASE A
/CASE B D0 SORT1(B,SPECTR)
/CASE C D0
  FILTER(C,MATH,N1,K1,N2,K2)
  SORT12(C,INT1)
  SORT22(C,INT2)
/END CASE C
/CASE FLAG D0 FORMD('ALPHA','VTURIN','NEQDIM143',N,M1,M2,...,DESCR)
/CASE DESCR D0
  SUSPEND(CAMAC) ; DATA STREAM INTERRUPTION
  WRITE(DESCR,SPECTR,MATH,INT1,INT2)
  UP(FIN)
/END CASE DESCR
/HIERARCHY A => B, FLAG => DESCR
// ----- 4
F1=100 ; INITIAL VALUES OF BOUNDARIES
L1=1024 ; FOR FILTRATION (WINDOWS)
ACCEPT(N1,K1,N0,K2) ; INPUT FROM TERMINAL
SET(CAMAC)
S1: D0 S2 N=1,10
  RESUME(CAMAC) ; DATA STREAM SWITCH ON
  JALL(FIN)
  PRINT('ИЗМЕРЕНИЕ N',N)
S2: CONTINUE S1
  PRINT('END OF EXPERIMENT ALPHA')
  UNSET(CAMAC)
  END

```

Рис. 6. Пример текста описания прикладной системы на языке САНПО.

оборудования /параметр NE оператора TYPE /, его конфигурации /операторы DEVICE и LET совместно определяют адреса регистров и занятые векторы прерываний/, а также способ взаимодействия программ с обслуживаемым оборудованием /операторы HCASE и ATTACH/.

Группа операторов на участке 2 текста описывает используемые элементы данных.

Схемы процессов преобразования данных описаны операторами в разделе 3. Приоритеты операций заданы оператором HIERARCHY.

Основная последовательность операций представлена в конце текста. Смысл большинства операторов ОПЭ понятен из приведенного на рис. 6 текста. Группа подготовительных операций включает оператор RESUME, который разрешает ввод данных по активным каналам и тем самым инициирует работу всех трех процессов. Интерпретация ОПЭ будет приостановлена на операторе WAIT до объявления события FIN. Объявление этого события, как следует из представленной схемы процессов, свидетельствует о завершении отдельного измерения и позволяет продолжить цикл /в ОПЭ/ из 10 однотипных измерений с записью результатов каждого из них на МЛ.

7. ЗАКЛЮЧЕНИЕ

Использование в данном комплексе такой компоненты, как обобщенная система САНПО, обеспечивает одновременное развитие эксплуатируемых прикладных систем, доступность новых программных ресурсов без изменения всем прикладным системам, немедленное использование новых версий СП после их коррекции или модификации.

Модульная организация системы и иерархия операционных средств позволяют осуществить эволюционное развитие возможностей САНПО. Унификация системных модулей и СП гарантирует преемственность разработок, сжатые сроки создания систем, быструю модификацию программ экспериментов.

К отличительным особенностям описанного комплекса можно отнести следующие:

1/ предложен метод программирования эксперимента, отличающийся тем, что программа эксперимента подразделяется на две части: основную последовательность операций, управляющую всем экспериментом, и схему процессов обработки данных, исполнением которых управляет поток данных;

2/ введены средства высокого уровня для программирования параллельных процессов, освобождающие программиста от необходимости детально описывать процедуру их синхронизации на уровне семафоров;

3/ структура прикладной системы допускает изменение в некотором объеме функций во время ее работы и интерактивное воздействие на исполняемую программу эксперимента.

Комплекс в полном объеме представляет интерес в первую очередь для исследовательских и проектных организаций. Продукты генерации, прикладные системы, могут найти более широкое применение, например, в АСУ и АСУТП при условии соответствующей проблемной ориентации комплекса /или его библиотек/.

При проектировании прикладной системы следует учитывать, что программное переключение параллельных процессов в данной реализации выполняется после окончания текущей операции системы.

В настоящее время созданы модули монитора, некоторые подсистемы, редактор библиотек, разработаны язык и компоновочная программа.

Разработанные модули системы используются в эксперименте.

Авторы пользуются случаем поблагодарить В.П.Ширикова за полезные обсуждения, В.И.Лущикова, Л.Б.Пикельнера, А.Б.Попова за поддержку и помощь в работе и Т.Б.Журавлеву за оформление рукописи.

ЛИТЕРАТУРА

1. Голиков В.В. и др. ОИЯИ, 3-5736, Дубна, 1971.
2. Ананьев В.Д. и др. ОИЯИ, P3-10888, Дубна, 1977.
3. Блинников Н.И. и др. ОИЯИ, 9-10064, Дубна, 1976.
4. Иванченко И.М., Сеннер А.Е. Общая организация системы управляющих программ для установки "Фотон". В сб.: Лекции Международной школы по использованию ЭВМ в ядерных исследованиях. ОИЯИ, D10,11-8450, Дубна, 1974, с.296.
5. Котов В.М. ОИЯИ, 11-7941, Дубна, 1974.
6. Donelson W.S. Project Planning and Control. Datamation, 1976, vol.22, No.6, pp.73-80.
7. Ringland G. The Compt.Journ., 1975, vol.18, No.4, pp.312-317.
8. Нестерихин Ю.Е. и др. "Автометрия", 1974, №4, с.3-9.
9. Талныкин Э.А. "Автометрия", 1976, №1, с.65-72.
10. Дади К. и др. ОИЯИ, 10-9060, Дубна, 1975.
11. Perry J.R., Gamble A. J.Phys.E: Sci.Instr., 1978, vol.11, No.12, pp.1152-1156.
12. Дийкстра Э. Взаимодействие последовательных процессов. В кн.: "Языки программирования". Пер. с англ. под ред. Ф.Женюи. "Мир", М., 1972, с.406.
13. Elzer P. Facilities for a Real-Time Language. Software World, 1972, vol.3, No.4, pp.20-24.
14. Наумов Б.Н., Боярченков М.А., Кабалевский А.Н. "Приборы и системы управления", 1977, №10, с.12-15.
15. Наумов Б.Н. "Приборы и системы управления", 1977, №10, с.3-5.
16. Борисенко В.Д., Плотников В.В., Талов И.Л. "Электронная промышленность", 1978, №10/70/, с.20,21.
17. Балука Г., Саламатин И.М., Хрыкин А.С. ОИЯИ, 10-12545, Дубна, 1979.
18. Островной А.И., Саламатин И.М. ОИЯИ, P10-11349, Дубна, 1978.
19. Информационные системы общего назначения. Аналитический обзор систем управления базами данных. Пер. с англ. под ред. Е.Л.Юценко. "Статистика", М., 1975, с.500.
20. RT-11 System Reference Manual (DEC-11-ORUGA-C-D). DEC, Maynard, Massachusetts, 1975.
21. RT-11 Software Support Manual (DEC-11-ORPGA-B-D), DEC, Maynard, Massachusetts, 1975.
22. Намсрай Ю., Савватеев А.С., Саламатин И.М. ОИЯИ, P10-12206, Дубна, 1979.
23. Дийкстра Э. Заметки по структурному программированию. В кн.: У.Дал, Э.Дийкстра, К.Хоор. Структурное программирование. Пер. с англ. "Мир", М., 1975, с.7-47.
24. Nadrchal J. Compt.Phys. Commun., 1976, vol.11, No.2, pp.143-157.
25. Балука Г., Саламатин И.М., Хрыкин А.С. ОИЯИ, 10-12546, Дубна, 1979.

Рукопись поступила в издательский отдел
4 декабря 1979 года.