

**СИСТЕМЫ АВТОМАТИЗАЦИИ
ФИЗИОЛОГИЧЕСКИХ ИССЛЕДОВАНИЙ**

УДК 681.3.06

**В. П. ТРЕТЬЯКОВ, М. Б. ШТАРК, Е. И. ШУЛЬМАН,
Г. Я. ЯНОВСКИЙ**
(Новосибирск)

**ПРИНЦИПЫ ПОСТРОЕНИЯ И ФУНКЦИОНИРОВАНИЯ
ПРОБЛЕМНО-ОРИЕНТИРОВАННЫХ ПРОГРАММНЫХ СИСТЕМ
АВТОМАТИЗАЦИИ ИССЛЕДОВАНИЙ В ЭКСПЕРИМЕНТАЛЬНОЙ
БИОЛОГИИ НА ОСНОВЕ МИКРОЭВМ И КАМАК**

Принципиальным шагом на пути повышения эффективности автоматизации научных исследований, расширения сферы применения средств вычислительной техники в различных областях науки является наметившаяся в последние годы тенденция к созданию и использованию недорогих, достаточно надежных управляющих вычислительных комплексов на основе микроЭВМ и оборудования КАМАК [1, 2]. Массовый выпуск и широкое распространение микрокомпьютерных систем автоматизации научных исследований еще в большей степени подчеркивают хроническое отставание средств разработки прикладного программного обеспечения (ПО) от потребностей проблемных специалистов. Среди основных требований, которым, по нашему мнению, должны удовлетворять средства программирования экспериментов, отметим необходимость снижения затрат и сроков на создание, ввод в эксплуатацию и последующее сопровождение прикладного ПО, облегчения его освоения и использования различными категориями пользователей (прежде всего, исследователями, практически не знакомыми с программированием). Реализация этих требований позволяет существенно сократить латентное время от момента формирования требований к ПО конкретных экспериментов до получения пригодного к использованию программного продукта. Разрешение этих проблем заключается в переходе к технологическим методам разработки прикладного ПО, в обеспечении его модульной организации и за счет этого мобильности отдельных программных модулей, т. е. возможности их использования без изменений в программных системах для различных экспериментов.

В последнее десятилетие в стране успешно развивается подход к разработке прикладного ПО на основе пакетов прикладных программ (ППП). Анализ данных литературы [3—5] показывает, что подавляющее большинство работ по ППП ориентировано на нужды так называемого вычислительного эксперимента, характеризующегося сложностью расчетов, вариабельностью входных данных и применяемых численных методов [6]. Приходится констатировать, что в настоящее время практически отсутствуют программные системы (комплексы), обеспечивающие применение методики разработки ППП для экспериментальных научных исследований (в том числе для задач экспериментальной биологии) в реальном масштабе времени, проводимых с использованием микроЭВМ и КАМАК. Это обусловлено как значительными сложностями разработки специализированных средств программирования (что в условиях иссле-

довательских организаций (лабораторий) затруднено ввиду недостаточного числа профессиональных программистов), так и необходимостью разрешить ряд противоречивых проблем, вызванных особенностями программирования аппаратуры КАМАК.

Накопленный нами опыт разработки прикладного ПО систем автоматизации разнообразных физиологических и нейробиофизических экспериментов позволил сформулировать ряд специфических (в первую очередь для нашей проблемной области) требований к ПО экспериментов, среди которых выделим два: 1) потребность в обеспечении оперативной модификации функциональных возможностей программных систем непосредственно в ходе эксперимента с целью адаптации к выявленным условиям и особенностям конкретного исследования; 2) потребность в реализации типовых сервисных средств, не зависящих от методики конкретного исследования и обеспечивающих активное участие исследователя в управлении экспериментом.

С целью удовлетворения требований и разрешения проблем, обсужденных выше, в лаборатории комплексных исследований нейронных систем (ЛКИНС) ИАиЭ СО АН СССР разрабатывалась модульная инструментальная система программирования экспериментов САНПО-3 [7]. Система САНПО-3 — новая версия программного комплекса САНПО (совместная разработка ЛНФ ОИЯИ и ИАиЭ СО АН СССР [8]), работающая под управлением штатной операционной системы (ОС) РАФОС (или RT-11). Сущность новых решений, предложенных в системе САНПО-3, заключается в разработке методов и средств, обеспечивающих создание проблемно-ориентированных программных систем (ПОС) для классов экспериментальных исследований с едиными для всех систем развитыми мониторными функциями.

В работе описываются принципы построения и функционирования ПОС САНПО-3, а также обсуждается опыт создания и применения конкретных систем автоматизации различных классов исследований в экспериментальной биологии.

Инструментальная система программирования экспериментов. Общая характеристика. Система программирования САНПО-3 предназначена для создания прикладного проблемно-ориентированного программного обеспечения для экспериментальных исследований в реальном масштабе времени, проводимых с использованием управляющих вычислительных комплексов типа «МикроКАМАКлаб». Система САНПО-3 — проблемно-ориентированная надстройка штатной ОС РАФОС. Она реализует принципы модульного построения ПОС для задач в реальном времени. Эти принципы характеризуются тем, что отдельные программные модули системы программируются на традиционных алгоритмических языках в среде РАФОС, а общая логическая структура ПО конкретного эксперимента, включающая описание структур данных, перечень функциональных действий и другие спецификации, оформляется на специализированном языке высокого уровня САНПО-3 (новая версия языка САНПО [9]). Процесс компоновки ПО эксперимента из отдельных заранее созданных программных модулей на основании описания, выполненного на языке САНПО-3, будем называть генерацией ПОС, а созданные средствами САНПО-3 проблемно-ориентированные системы — исполняющими программными системами (ИПС) для конкретных исследований.

С точки зрения методики программирования САНПО-3 представляет собой двухуровневую систему разработки ПО экспериментов: нижний уровень поддерживается ОС РАФОС (разработка модулей, файловая система), а верхний — средствами собственно САНПО-3 (генерация ИПС, распределение памяти, управление сбором и обработкой экспериментальных данных).

В отношении принципов организации работы ИПС обеспечен, по существу, однопользовательский многозадачный режим под общим управлением монитора ОС РАФОС. В качестве подзадач выступают в данном

случае отдельные программные модули ИПС, чью работоспособность обеспечивают системные модули ИПС.

По характеру использования проблемно-ориентированные системы САНПО-3 являются ППП, ориентированными на пользователей-экспериментаторов.

По составу система САНПО-3 может быть разделена на инструментальную и функциональную компоненты. К первой относятся язык САНПО-3 и группа служебных программ генерации, ко второй — системные модули монитора и библиотеки функциональных программных модулей (ФПМ).

ФПМ САНПО-3 предназначены для выполнения отдельных системных и прикладных операций. Весь цикл их создания совершается в рамках ОС РАФОС с использованием ее стандартных средств подготовки программ. Программирование ФПМ может осуществляться на языках Макро-11, Фортран-IV и Паскаль — Рафос. После завершения автономной отладки очередной ФПМ преобразуется в загрузочный модуль перемещаемого формата REL и в таком виде записывается в одну из исходных САНПО-библиотек.

Прикладные алгоритмы с КАМАК-оборудованием программируются в виде унифицированных программных модулей (УПМ), в структуре которых учтены особенности работы с КАМАК. Они реализуют типичные для заданной проблемной области операции с оборудованием и обслуживают работу группы блоков КАМАК.

Все ФПМ в обязательном порядке должны содержать в тексте программы документирующее описание — паспорт модуля. Это описание включает некоторые характеристики модуля, используемые как при генерации, так и в процессе работы модулей в составе ИПС.

Особое место среди программных модулей САНПО-3 занимает монитор ИПС. Это универсальная управляющая программа, включаемая в тело резидента (ядра) любой ИПС и настраиваемая на обслуживание конкретного эксперимента. Функциональные свойства монитора во многом определяют принципы работы ИПС в эксперименте. Аналогичную роль в организации работы и управления КАМАК-оборудованием играет программный КАМАК-монитор, автоматически включаемый в тела компонуемых КАМАК-подсистем и настраиваемый на конкретную конфигурацию оборудования. КАМАК-монитор — универсальный для используемого типа крейт-контроллера программный модуль, не зависящий от функций отдельных УПМ.

Язык САНПО-3 — специализированный язык высокого уровня для описания программных систем конкретных экспериментов. Все инструкции языка декларативные. В языке нашли отражение специфические черты задач реального времени: необходимость накопления и предварительной обработки экспериментальной информации, поступающей по нескольким параллельным каналам оборудования в асинхронном режиме. Текст описания эксперимента на языке САНПО-3 включает следующие основные разделы: 1) описание состава оперативной базы данных (ОБД), включающее перечень поименованных элементов данных (ЭД); 2) перечень функциональных действий с указанием приоритетных отношений между ними; 3) перечень интерактивных приказов пользователя; 4) описание состава, способа подключения и размещения используемого КАМАК-оборудования; 5) описание состава и способа исполнения УПМ, обслуживающих данное оборудование. Функциональные действия специфицируются на языке САНПО-3 с помощью инструкций, определяющих, по существу, отношения типа «условие — действие». Условия обычно связываются с состоянием некоторого ЭД и призваны отражать готовность информации, содержащейся в ЭД, к обработке или любую другую желаемую ситуацию. Условия могут находиться в двух состояниях: активном и пассивном, где признаком состояния (готовности) является флаг события. Перевод условия в активное состояние будем называть установкой флага (объявлением) события, а обратную операцию — сбро-

сом флага события. Функциональные действия, связанные с некоторым условием, составляют определенную операцию системы, выполняемую совокупностью функциональных операторов (ФО). Программой реализации ФО будет ФПМ. Последовательность действий, поставленная в соответствие некоторому условию, совместно с признаком готовности к исполнению, образует понятие «процесс» САНПО-3. Процесс — основная алгоритмическая конструкция для программирования функциональных действий ИПС. Реализованы три типа процессов (быстрые, медленные и макропроцессы), каждый из которых специфицируется отдельной инструкцией языка [7]. Различия между типами процессов — в способе представления и механизме обслуживания в ИПС. Составляющие процессов — функциональные операторы — являются, по существу, исполняемыми операторами языка описания и могут использоваться в различных программных системах без изменений. Набор исполняемых операторов открыт и может пополняться путем создания новых программных модулей.

Для описания работы с КАМАК-оборудованием предназначена особая группа декларативных инструкций — подмножество языка САНПО-3. Она позволяет задать способ подключения оборудования к ЭВМ и его конфигурацию, состав и способ исполнения операций с КАМАК. Это описание используется для компоновки КАМАК-подсистем — программных единиц в составе ИПС, обеспечивающих взаимодействие с оборудованием. В КАМАК-подсистеме выделяются в общем случае две составные компоненты: КАМАК-резидент, работающий в режиме фиксации на рабочем поле ИПС, и прикладные программы, исполняемые в режиме динамического распределения памяти.

Модульная организация ПО КАМАК. Особенностью программирования работы с экспериментальным оборудованием является то, что программы управления КАМАК наиболее «чувствительны» к специфике конкретного экспериментального исследования. Дополнительные сложности обусловлены собственно конструктивом КАМАК, в частности «географическим» принципом адресации КАМАК-блоков. Кроме того, к программам взаимодействия с КАМАК в режиме эксперимента обычно предъявляются достаточно жесткие требования по быстродействию. Для решения этих во многом противоречивых проблем и тем самым повышения эффективности разработки КАМАК-программного обеспечения реализован специальный подход, опирающийся на принципы модульного программирования и сочетающий: 1) унифицированную структуру программных модулей; 2) методику программирования этих модулей в виде, не зависящем от варианта подключения и размещения используемых блоков оборудования; 3) метод компоновки из этих модулей ПО для работы с КАМАК в соответствии со спецификациями, оформленными на специализированном языке. Как отмечалось, структурная единица программирования КАМАК — УПМ. В структуре УПМ выделяются пять программных секций: одна основная (функциональная), иницируемая по прерыванию или программному вызову, и четыре дополнительных (управляющих) для выполнения вспомогательных по отношению к основному алгоритму действий. Важный элемент структуры УПМ — специальная таблица настройки, содержащая необходимую информацию для записи физических адресов оборудования в тело модуля согласно фактической расстановке блоков в крейте. Каждый УПМ реализует вполне определенную функционально завершенную прикладную операцию, являющуюся частью общего алгоритма взаимодействия ИПС с КАМАК. В зависимости от способа исполнения модулем этой операции УПМ делится на активные (обслуживающие прерывания от КАМАК) и пассивные (получающие управление программным путем).

Разработанная модульная организация ПО для КАМАК служит основой построения ПОС автоматизации экспериментальных исследований с применением КАМАК и предопределяет необходимость модульного анализа алгоритма управления КАМАК, отражающего основную специ-

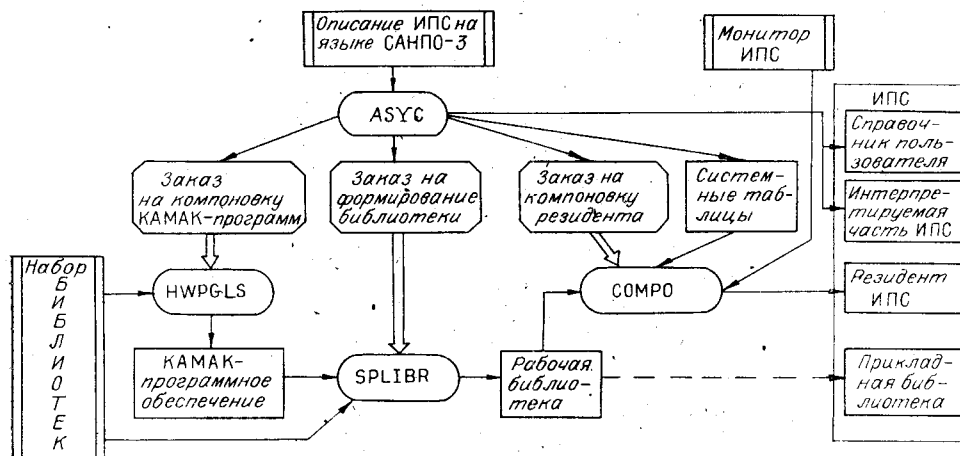


Рис. 1

фику конкретного эксперимента. Предложенная унификация структуры УПМ не накладывает ограничений на программирование требуемого алгоритма, а реализованный способ записи команд обращения к блокам КАМАК обеспечивает УПМ инвариантность по отношению к конфигурации оборудования и возможность достижения предельных (сравнимых с Ассемблером) скоростных характеристик. Сочетание отмеченных свойств позволяет строить сложные алгоритмы взаимодействия программной системы с экспериментальным оборудованием, где эксплуатационные характеристики соответствующих программ удовлетворяют достаточно жестким требованиям экспериментов в нашей проблемной области.

Генерация проблемно-ориентированных систем. К средствам генерации относятся интерактивный библиотекарь LIBRA, обеспечивающий формирование САНПО-библиотек ФПМ, а также инструментальные программы ASYC (компилятор языка САНПО-3), HWPGLS (компоновщик КАМАК-подсистем), SPLIBR (программа управления библиотеками), COMPO (компоновщик резидента ИПС), непосредственно участвующие в процедуре генерации. Их последовательное исполнение позволяет создать ИПС для конкретных исследований. Все программы генерации оформлены как файлы в формате загрузки SAV и исполняются под управлением ОС РАФОС. Схема технологического цикла генерации, а также вклад каждой из инструментальных программ генерации в создание компонентов ИПС показаны на рис. 1. Продукт генерации — ИПС для конкретного исследования — представляется четырьмя компонентами: резидентом ИПС, прикладной библиотекой, файлами интерпретируемой части и директив пользователя. Такая организация ИПС имеет следующие особенности:

1) состав ФПМ, используемых в ИПС, может расширяться путем создания новых программных модулей и включения их в прикладную библиотеку без редактирования описания системы и повторения генерации;

2) выделение описаний медленных процессов и макропроцессов в отдельную компоненту ИПС — файл интерпретируемой части — позволяет практически неограниченно расширять функциональные возможности ИПС за счет создания новых процессов этих типов, а также обеспечивает достаточную свободу при выборе типов процессов в условиях ограниченного ресурса оперативной памяти управляющей ЭВМ;

3) файл директив может рассматриваться и использоваться одновременно в двух качествах: как средство, облегчающее интерактивное взаимодействие пользователя с системой, и как документ, создаваемый автоматически и содержащий заказанные пользователями справочное текстовое описание системы и ее возможностей.

Проблемно-ориентированные программные системы. *Общая характеристика.* Проблемно-ориентированные системы САНПО-3 предназначены для обслуживания экспериментальных исследований в реальном масштабе времени и обеспечивают:

1) регистрацию и накопление экспериментальных данных одновременно по нескольким каналам оборудования, работающим в асинхронном режиме;

2) предварительную обработку данных в темпе эксперимента и сохранение промежуточных результатов на внешних носителях, а также математическую обработку для получения окончательных интерпретируемых проблемными специалистами результатов;

3) организацию обратной связи с экспериментальной установкой (объектом исследования) с целью управления ее параметрами (его состоянием) по определенным критериям;

4) интерактивное управление процедурой эксперимента при помощи приказов пользователя-исследователя.

ИПС конкретного эксперимента работает под управлением ОС РАФОС. Резидент ИПС является задачей операционной системы и исполняется в ее фоновом разделе, загружаясь в оперативную память директивой RUN. Остальные компоненты ИПС располагаются на дисковом запоминающем устройстве в виде отдельных файлов. Исполнение программы эксперимента обусловлено сочетанием интерактивного и автоматического режимов работы. Основное содержание автоматического режима состоит в исполнении процессов. В общем случае работа очередного процесса, включающего ФО разрешения регистрации, инициирует ввод экспериментальных данных в режиме обработки прерываний от блоков КАМАК. По мере готовности накопленных данных к дальнейшей обработке на соответствующих ЭД объявляются события, что приводит к запуску связанных с ними процессов согласно заданным приоритетным отношениям. Процессы функционируют в фоновом режиме ИПС, т. е. в периоды времени, свободные от обслуживания КАМАК-прерываний. Процессы исполняются в режиме интерпретации с использованием быстрого и медленного механизмов, сочетание которых, а также ряд других системных решений способствуют гибкому управлению работой ИПС с целью достижения требуемой пропускной способности. Интерактивные приказы могут быть введены в произвольный момент работы ИПС. Отдельные ФО инициируются при отсутствии готовых к исполнению процессов, т. е. имеют более низкий по отношению к ним приоритет. Процессы, вызываемые по имени, обслуживаются в соответствии с общей дисциплиной работы процессов. Для повышения реактивности системы введена особая форма интерактивного приказа, позволяющая любой из них исполнить «вне очереди», т. е. по окончании выполнения текущей операции системы.

ИПС САНПО-3 могут быть отнесены к разряду проблемно-ориентированных программных систем открытого типа, удовлетворяющих основным требованиям, предъявляемым к развитым пакетам прикладных программ [5]. Принципы функционирования ИПС определяются едиными для всех ИПС системными управляющими модулями, а проблемная ориентация — функциональным наполнением используемого пакета программ. Важная особенность проблемно-ориентированных систем САНПО-3 — единая структура резидента, которую рассмотрим, исходя из рис. 2.

Системное ядро ИПС — монитор, блок-схема алгоритма работы которого показана на рис. 3. Монитор ИПС выполняет следующие основные функции: 1) обслуживание дисциплины исполнения процессов; 2) обслуживание исполнения функциональных операторов; 3) загрузка модулей из прикладной библиотеки и обслуживание режима динамического распределения памяти; 4) поддержка интерактивного режима работы; 5) исполнение типовых системных операций; 6) поддержка режима комплексной отладки ИПС. Таблицы системной области являются сред-

ством настройки монитора на работу в составе конкретной ИПС и обеспечивают доступ и взаимодействие с другими элементами структуры резидента и составными компонентами ИПС. В таблицах ОБД содержатся описания всех ЭД системы, что, в частности, обеспечивает доступ к их характеристикам и областям фактического размещения в ОБД. Здесь же в специальном формате размещены описания процессов, которые требуют быстрого механизма обслуживания, а также схема исполнения всех процессов, являющаяся моделью функционирования ИПС.

Рабочее поле (РП) системы предназначено для размещения и исполнения ФПМ. Те ФПМ, алгоритм работы которых требует постоянного присутствия в оперативной памяти или к скоростным характеристикам которых предъявлены повышенные требования, объявляются резидентными и образуют область резидентных (фиксированных) программ. Таблица характеристик является каталогом текущего состава модулей на РП и служит для организации доступа к ним. Участок динамически распределяемой памяти (ДРП) используется для автоматической загрузки модулей из прикладной библиотеки и их исполнения.

Режимы исполнения программы эксперимента. Основные режимы работы ИПС опираются на принципы организации исполнения процессов, управления КАМАК и интерактивного взаимодействия.

Процессы функционирования ИПС образуют некоторую подсистему обработки данных. Синхронизация и запуск процессов, реализуемые монитором ИПС, подчинены некоторой дисциплине. Эта дисциплина в первую очередь определяет порядок инициирования процессов и сводится к тому, что в случае наличия в системе группы готовых к обслуживанию процессов к исполнению выбирается тот, который обладает наивысшим приоритетом. В общем случае работа процесса заключается в последовательном исполнении всех ФО, включенных в его состав. Использование же ФО управления процессами позволяет при выполнении заказанных условий изменять стандартный порядок работы процесса и производить объявление (сброс) флагов событий на других ЭД. Это способствует реализации весьма сложных алгоритмов обработки данных, повышается эффективность программирования процессов на языке описания. Кроме того, потребность в гибкой организации исполнения программы эксперимента привела к реализации ряда системных алгоритмов, дополняющих основную схему работы процессов и обеспечивающих: а) динамическое изменение приоритетов процессов, что позволяет присвоить любому процессу на время исполнения наивысший приоритет и выполнить его «вне очереди»; б) возможность переопределения приоритетов процессов; в) динамическое переключение процессов, позволяющее связать событие, объявленное на некотором ЭД, с одной из заданных последовательностей ФО.

Одни и те же ФО могут работать как в автоматическом, так и в интерактивном режиме. Обмен данными между ними осуществляется через оперативную базу данных, причем доступ к параметрам (ЭД, числовым и текстовым константам) обеспечивается монитором ИПС, формирующим соответствующие интерфейсные таблицы перед передачей управления модулям. Программные модули, не включенные в резидентную область РП, исполняются в режиме динамического распределения памяти, который обеспечивает, несмотря на ограниченный ресурс РП, работоспособ-

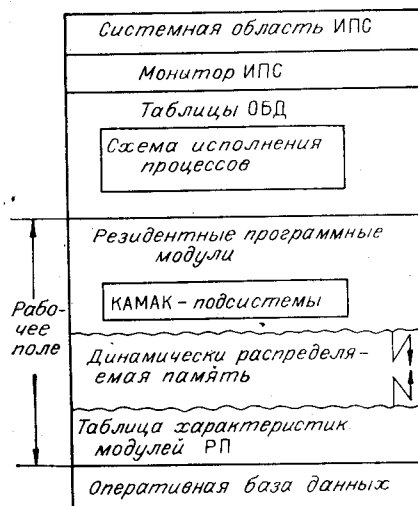


Рис. 2

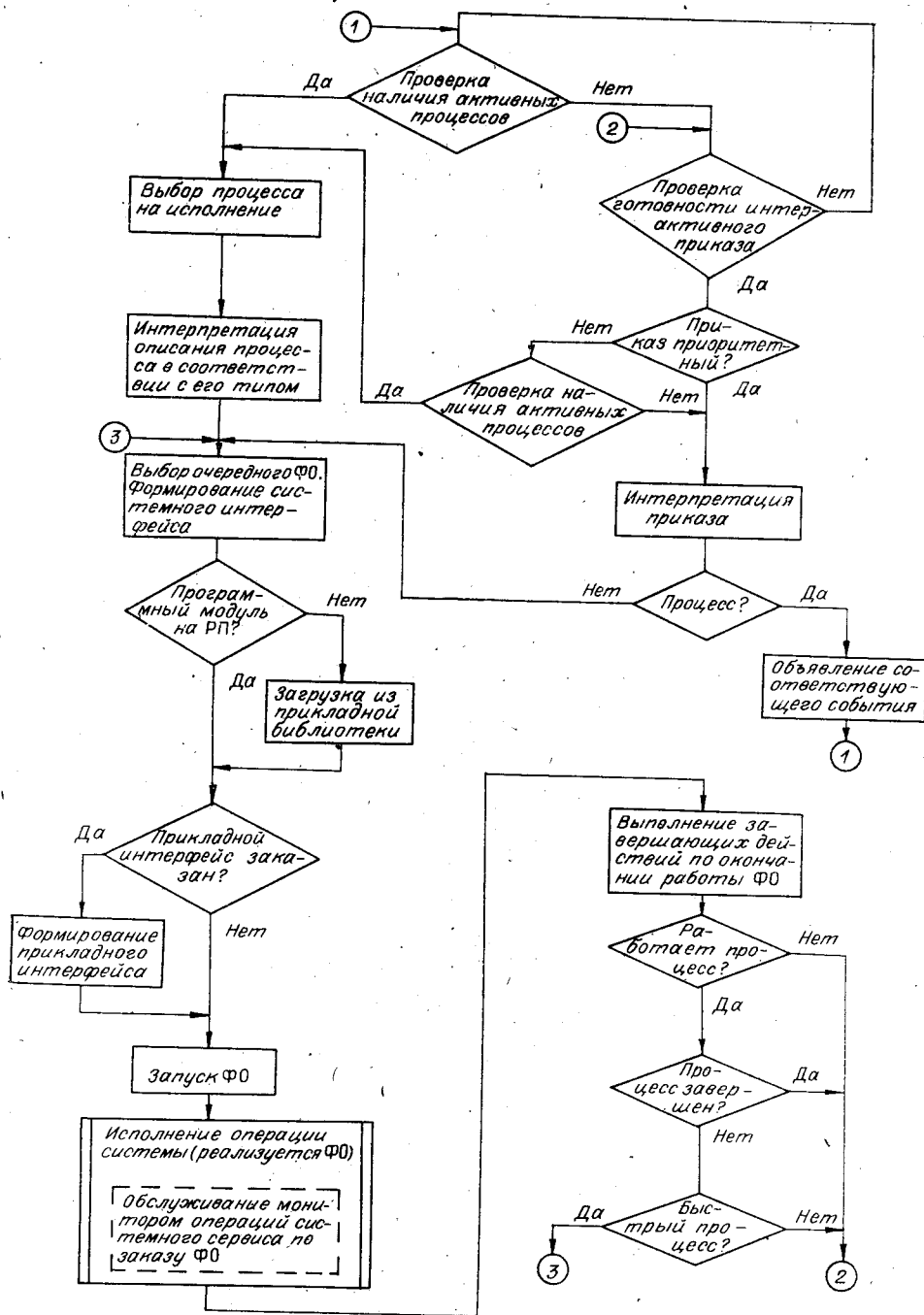


Рис. 3

ность всех модулей, включенных в состав прикладной библиотеки. Условия загрузки и исполнения определяются текущим состоянием ДРП и собственными характеристиками модулей, указанными в паспорте. Для выбора в каждой фазе работы ИПС требуемого состава фиксированных ФПМ реализован алгоритм динамического управления состоянием резидентной области РП. Это позволяет осуществлять оперативную перекомпоновку этой области с тем, чтобы уменьшить количество операций по загрузке модулей из библиотеки и тем самым повысить общее быстродействие ИПС. С другой стороны, удаление (может быть, временное) не-

которых фиксированных программных модулей приводит к расширению области ДРП и улучшению условий эксплуатации ФО, загружаемых из прикладной библиотеки.

Взаимодействие с экспериментальным оборудованием осуществляется КАМАК-подсистемами, которые позволяют организовать работу с четырьмя крейтами КАМАК. КАМАК-подсистемы — такие программные единицы, для которых возможны раздельное использование и управление. Они обеспечивают инициацию используемого оборудования, организуют начало и остановку регистрации, выполняют обмен данными и управляющими командами с КАМАК-оборудованием. Напомним, что КАМАК-подсистемы строятся из компонуемых на этапе генерации КАМАК-резидентов и прикладных программ. В составе резидента объединяются КАМАК-монитор и в общем случае несколько УПМ, причем активные модули всегда включаются в тело резидента. КАМАК-монитор управляет работой крейт-контроллера, обеспечивает диспетчеризацию прерываний, выполняет синхронизацию операций накопления и обработки экспериментальных данных, поддерживает определенную дисциплину работы управляющих секций УПМ, а также исполняет ряд других функций, типичных для обслуживания КАМАК.

Для использования КАМАК-подсистем в составе ИПС сформулированы принципы организации их исполнения. Для прикладных программ, работающих на ДРП, обеспечено последовательное исполнение всех секций УПМ в некотором фиксированном порядке. В отношении УПМ резидентов выработаны следующие правила: их функциональные секции всегда иницируются независимо (по прерыванию или программному вызову), а для управляющих секций разрешено группирование одноименных секций в единую операцию и раздельное самостоятельное использование. Это, с одной стороны, позволяет сократить число операций управления, синхронно выполнять действия с оборудованием группой УПМ, с другой — предоставляет достаточную свободу для реализации требуемых алгоритмов взаимодействия с КАМАК. Обмен данными между КАМАК-подсистемами и подсистемой обработки данных осуществляется через ОБД. В целях синхронизации их работы применяется аппарат событий. Для случаев, когда интенсивность информационного потока превышает пропускную способность системы, предусмотрен алгоритм временной приостановки регистрации, способствующий монополизации ресурсов ИПС с целью освобождения («разгрузки») буферов накопления. Реализованный алгоритм гарантирует минимально возможное время простоя оборудования в конкретных условиях.

Единый подход к разработке ПО КАМАК позволил создать универсальные сервисные средства, применимые в любой ИПС и способствующие гибкой динамической перестройке КАМАК-подсистем в соответствии с требованиями эксперимента. Они обеспечивают интерактивную настройку программ на другие адреса оборудования и ЭД, частичную предварительную проверку работоспособности используемого оборудования и корректности соответствующих КАМАК-подсистем. Особая роль принадлежит оператору конфигурирования состава КАМАК-резидентов, выполняющему исключение из резидентов и удаление с РП тех УПМ, которые не используются в конкретной фазе работы системы. Заметим, что перечисленные средства без дополнительных затрат вносят определенную гибкость в организацию работы КАМАК-подсистем и могут эффективно использоваться при тестировании и отладке программ и экспериментального оборудования.

Интерактивные приказы (в первую очередь имена процессов) образуют язык диалогового взаимодействия пользователя с системой. Он формируется при описании ИПС в терминах, отражающих функциональное назначение операций системы, адекватных понятиям данной проблемной области, и может уточняться и совершенствоваться вне связи с программированием новых ФПМ. Реализованы два режима интерактивного взаимодействия пользователя с системой. В основном режиме ин-

терактивные приказы инициируются путем непосредственного набора с терминала. В расширенном режиме, сохраняющем эту возможность, пользователям предоставляется дополнительный сервис, связанный с использованием файла директив. В этом файле строки приказов даны в таком же виде, в каком они могут вводиться с терминала. Расширенный режим обеспечивает: 1) автоматический доступ к файлу директив с возможностью запуска приказов по одному или «пакетом» последовательных строк непосредственно из файла; 2) возможность создавать новые и редактировать во время исполнения программы эксперимента имеющиеся произвольные текстовые файлы, в том числе специализированные системные файлы директив пользователя и интерпретируемой части ИПС.

Перечисленные возможности позволяют существенно повысить эффективность работы пользователя: автоматически устранить ошибки при наборе приказов, не запоминать вид строк обращения, расширить файл директив строками новых приказов, справочной информацией и т. п. В этой связи строки интерактивных приказов могут сопровождаться необходимыми комментариями и пояснениями (в том числе и формируемыми в ходе эксперимента), достаточными для использования ИПС без дополнительных инструкций.

Системное обеспечение. Для использования в составе ИПС разработана совокупность универсальных программных средств, целесообразных при подготовке и проведении практически любого экспериментального исследования под управлением ЭВМ. Эти средства выполняют определенную набор системных функций и образуют единое системное обеспечение САНПО-3, одновременно распространяемое на все создаваемые ПОС. Некоторые из них были отмечены ранее; в этом разделе дается краткая характеристика других возможностей, предоставляемых пользователям и разработчикам ИПС. К ним относятся:

1) встроенные (в монитор ИПС) средства поддержки режима комплексной отладки, обеспечивающие печать трассы работы системы, а также приостановку автоматического режима исполнения с возможностью запуска интерактивных приказов в промежуточных контрольных точках;

2) средства централизованной обработки ошибок разной природы, включающие выдачу подробного диагностического сообщения собственно об ошибке и о состоянии системных областей, а также возможность описания альтернативных действий, призванных устранить или исправить влияние ошибки и способствовать продолжению работы ИПС в эксперименте;

3) средства создания и управления САНПО-ориентированными специализированными базами экспериментальных данных, содержащими помеченные записи ЭД, которые могут сопровождаться числовыми и текстовыми ключами-идентификаторами для последующего направленного поиска, анализа и обработки;

4) средства графического отображения одно- и двумерных массивов на ряде устройств (цветной и черно-белый телевизионные дисплеи, дисплеи запоминающий графический, двухкоординатный самописец), причем автоматический вывод на дисплеи сочетается с возможностью преобразования формата изображения с помощью функциональной клавиатуры терминала;

5) средства автоматического документирования протокола диалога пользователя с системой, обеспечивающие сохранение в текстовом файле или на листинге всех сообщений, появляющихся на экране терминала.

Отдельно отметим, что в системе САНПО-3 предоставлена возможность оперативной модификации описаний процессов любой ИПС без необходимости редактирования текста на языке САНПО-3 и проведения повторной генерации. Модификация процессов выполняется в режиме специализированного редактирования, который обеспечивает полный спектр возможностей: создание новых процессов, исключение уже описанных, включение в состав процессов дополнительных ФО, удаление ФО, изменение списка параметров произвольных ФО. Реализованный

подход позволяет без дополнительных алгоритмических затрат и ухудшения скоростных характеристик программной системы осуществлять адаптацию ИПС к условиям и особенностям исследования, возникшим во время эксперимента, гибко настраиваться в достаточно широком диапазоне выполняемых экспериментальных методик. Обеспеченная таким образом возможность непосредственно продолжить эксперимент в изменившихся условиях — принципиальное свойство ИПС САНПО-3, в полной мере учитывающее особенности автоматизированных исследований в экспериментальной биологии. Реализованные средства служат, кроме того, весьма мощным инструментом комплексной отладки ИПС.

Представленный (как в этом разделе, так и ранее) набор системных функций — универсальное наполнение системы САНПО-3, инвариантное к методикам конкретных экспериментов и применимое в составе любой ИПС. Использование этих средств позволяет существенно облегчить и ускорить процесс разработки и внедрения ИПС, предоставляет пользователям разнообразный сервис во время работы, повышает мощность создаваемых программных систем, переводя их в разряд развитых проблемно-ориентированных систем для экспериментальных исследований в реальном масштабе времени.

Типовые программные системы. Требования, реализация, свойства. Специфической особенностью экспериментальных исследований является то, что они находятся в постоянном развитии: по мере расширения наших знаний об изучаемом явлении (процессе) уточняется и совершенствуется методика исследования, схема эксперимента и т. п. Организация экспериментальных исследований на линии с ЭВМ приводит к тому, что ПО эксперимента как наиболее «чувствительная» составляющая системы автоматизации также регулярно обновляется и модифицируется. В такой ситуации целесообразно еще на стадии проектирования ПО экспериментов учитывать, что эксплуатационные свойства программной системы во многом будут зависеть от того, насколько полно в системе реализованы стандартные для данного класса исследований функции и насколько просто и быстро можно перестроить систему на решение других, но близких задач. Это будет способствовать широкому распространению соответствующего прикладного ПО и как следствие принятию его в качестве типового для определенного класса исследований.

Программная система может рассматриваться как типовая для определенного класса исследований в случае, если она удовлетворяет следующим требованиям: 1) наличие общей логической структуры, адекватной большинству задач данного класса; 2) реализация базового специализированного набора функций, необходимых для постановки стандартных типовых экспериментальных методик и развития на их основе нетрадиционных методик; 3) гибкая организация, обеспечивающая простоту перестройки, модификации и расширения системы в соответствии с изменениями методики и условий проведения исследования.

Модульный принцип построения, единый способ создания и тождественность структуры ИПС, развитые мониторинговые функции и единое системное обеспечение способствуют созданию в рамках САНПО-3 типовых ПОС для автоматизации широкого круга исследований в области экспериментальной биологии. В 1983—1985 гг. в ЛКИНС ИАиЭ СО АН СССР создано около десяти ПОС. Некоторые из них внедрены и успешно эксплуатируются также в ИМБП МЗ СССР (Москва), ИКЭМ СО АМН СССР (Новосибирск) и других организациях. Разработанные ПОС используются для изучения динамики основных параметров ряда физиологических систем организма человека и высших животных в условиях функциональных нагрузок [10], исследования механизмов межнейронного взаимодействия [11], анализа ответных реакций исследуемых биосистем с целью выработки управляющих воздействий по определенным критериям [12] и т. д.

Наш практический опыт разработки, развития и применения типовых ПОС для автоматизации исследований в экспериментальной биологии показал, что:

1) существенно повышается производительность труда программистов и сокращается дублирование работ за счет модульной организации ПО КАМАК, возможности разработки функциональных модулей на трех языках программирования, обеспечения мобильности модулей;

2) развитое единое системное обеспечение ПОС облегчает их освоение проблемными специалистами, способствует повышению эффективности работы исследователей в автоматизированном эксперименте;

3) ПОС могут использоваться различными категориями пользователей, в том числе не имеющими специальной подготовки в области программирования;

4) обеспечена принципиальная возможность «отторжения» программного продукта от его авторов, сопровождение ПОС может успешно осуществляться самими пользователями-экспериментаторами.

В заключение подчеркнем, что система программирования САНПО-3 служит инструментом для создания и поддержки исполнения ПОС прежде всего в экспериментальной биологии, хотя соответствующая проблемная ориентация библиотек функциональных модулей САНПО-3 позволяет использовать инструментальную систему для автоматизации научных исследований в других областях. Отличительная черта ПОС САНПО-3 — обеспечение непрерывности процесса экспериментального исследования (включая и математическую обработку), благодаря разработке и реализации средств, обеспечивающих оперативное изменение и наращивание функциональных возможностей систем в соответствии с эволюцией хода исследования без необходимости прервать процесс исследования для выполнения требуемых модификаций.

ЛИТЕРАТУРА

1. Нестерихин Ю. Е., Золотухин Ю. Н., Лившиц З. А. Автоматизация: итоги десятилетия.— Автометрия, 1984, № 4.
2. Велихов Е. П., Выставкин А. Н. Проблемы развития работ по автоматизации научных исследований.— Упр. сист. и маш., 1984, № 4.
3. Ершов А. П., Ильин В. П. Пакеты программ как методология решения прикладных задач.— В кн.: Пакеты прикладных программ. Проблемы и перспективы. М.: Наука, 1982.
4. Бежанова М. М. Пакеты прикладных программ: Учебное пособие.— Новосибирск: НГУ, 1983.
5. Корягин Д. А., Мартынюк В. В. Развитие проблематики системного обеспечения пакетов прикладных программ СССР.— В кн.: Пакеты прикладных программ. Проблемы и перспективы. М.: Наука, 1982.
6. Легоньков В. И. О построении программного обеспечения вычислительного эксперимента.— В кн.: Пакеты прикладных программ. Вычислительный эксперимент. М.: Наука, 1983.
7. Яновский Г. Я. Модульная инструментальная система программирования экспериментов САНПО-3. (Новая очередь программного комплекса САНПО).— Новосибирск, 1985. (Препринт/АН СССР, Сиб. отд-ние, ИАиЭ; 267).
8. Балука Г. и др. Комплекс средств для генерации прикладных систем автоматического накопления и предварительной обработки данных — САНПО.— Дубна, 1980. (Сообщение/ОИЯИ; № P10-12960).
9. Островной А. И., Саламатин И. М. Язык программирования прикладных систем автоматизации экспериментов.— Дубна, 1980. (Сообщение/ОИЯИ; № P10-80-423).
10. Яновский Г. Я. и др. HUMAN — комплекс программных и аппаратных средств автоматизации исследований физиологических систем человека.— Новосибирск, 1985. (Препринт/АН СССР, Сиб. отд-ние, ИАиЭ; 269).
11. Астафьев С. В., Третьяков В. П., Штарк М. Б., Яновский Г. Я. Аппаратное и программное обеспечение унифицированного измерительного тракта для микроэлектродных исследований.— Автометрия, 1984, № 4.
12. Астафьев С. В. и др. Организация биотехнической обратной связи на основе лабораторной микрокомпьютерной системы.— Автометрия, 1986, № 3.

Поступила в редакцию 19 ноября 1985 г.