

АКАДЕМИЯ НАУК СССР  
СИБИРСКОЕ ОТДЕЛЕНИЕ  
АВТОМАТИЗАЦИИ НАУЧНЫХ ИССЛЕДОВАНИЙ

УДК 681.3

В. В. ЛЕОНАС

*(Переславль-Залесский Ярославской обл.)*

**ИСПОЛЬЗОВАНИЕ ОПЕРАЦИОННОЙ СИСТЕМЫ  
UNIX В РЕЖИМЕ РЕАЛЬНОГО ВРЕМЕНИ**

Операционная система (ОС) UNIX была разработана в «Bell Labs» (США) в конце 60-х годов. За прошедшие с момента ее создания почти 20 лет эта операционная система и появившиеся в большом числе совместимые с ней операционные системы получили чрезвычайно широкое распространение в самых разных областях вычислительной техники. К ним относятся разработка программного обеспечения, обучение программированию, обработка текстов, научные и экономические расчеты. Несмотря на широчайшее использование ОС UNIX и подобных ей операционных систем, сложилось устойчивое мнение, что операционные системы рассматриваемого типа непригодны для использования в режиме реального времени, который требуется, например, при решении задач автоматизации экспериментальных исследований и создания АСУ ТП. Однако имеющийся опыт использования таких операционных систем в режиме реального времени опровергает это устоявшееся ошибочное мнение. Настоящая статья представляет собой суммирование опыта использования ОС UNIX и совместимых с ней операционных систем в режиме реального времени.

Первые результаты по использованию ОС UNIX в режиме реального времени для решения задач автоматизации экспериментальных исследований, по-видимому, получены в 1977 г. в лаборатории ускорителей Саскачеванского университета (Канада) [1]. Система автоматизации экспериментальных исследований в этой лаборатории была построена на основе мини-ЭВМ PDP-11/55 (ОЗУ — 224 Кбайт, ВЗУ — три НМД каскадного типа RK05 емкостью 5 Мбайт каждый) и аппаратуры КАМАК для сопряжения ЭВМ с экспериментальной установкой, представляющей собой линейный ускоритель на 300 МэВ. Типовой эксперимент имел следующие характеристики: частота наступления событий 360 Гц; длительность одного события 1 мкс; число параметров, характеризующих одно событие, 8.

Сбор информации о каждом событии осуществлялся параллельно с подготовкой к новым экспериментам и обработкой собранной экспериментальной информации. Программное обеспечение рассматриваемой системы автоматизации экспериментальных исследований представляло собой совокупность модифицированной версии ОС UNIX и набора пользовательских программ, обеспечивающих подготовку к проведению экспериментов, проведение сбора экспериментальной информации и обработку собранной экспериментальной информации. Для обеспечения возможности использования ОС UNIX в режиме реального времени в нее были внесены следующие изменения:

введена возможность записывания процесса пользователя в ОЗУ (т. е. запрета откочки процесса пользователя), что необходимо для обеспече-

Рис. 1. Структура сети вычислительного центра штата Айова (США): I — сеть с кольцевой топологией; II — мини-ЭВМ PDP-11/34; III — кластер персональных ЭВМ Pet; IV — персональная ЭВМ Pet; V — интерфейс RS-232C; VI — большая ЭВМ Itel AS-6

ния быстрого отклика на событие со стороны процесса сбора данных;

введена возможность вы-  
данных;

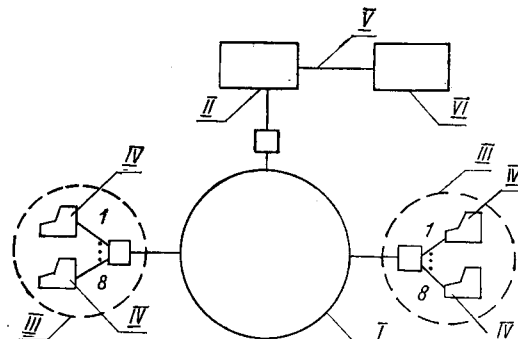
увеличен до 36 Кбайт размер пула буферов, вынесенных из ядра операционной системы (при этом на процессы пользователей остается ОЗУ размером 152 Кбайт, так как ядро операционной системы занимает столько же места в ОЗУ, сколько и пул буферов, т. е. 36 Кбайт).

С точки зрения программиста, контроллер крейта КАМАК представляет собой набор специальных файлов, что позволяет, имея драйвер, обойтись без разработки специальных подпрограмм взаимодействия с аппаратурой КАМАК. Все программное обеспечение, разработанное для рассматриваемой системы автоматизации экспериментальных исследований, было реализовано на языке программирования Си. На основе полученного авторами [1] опыта по использованию модифицированной версии ОС UNIX в режиме реального времени они делают вывод о том, что эта операционная система вполне подходит для работы в таком режиме.

Интересный опыт по использованию ОС UNIX в режиме реального времени получен в вычислительном центре штата Айова (США), где создана сеть с кольцевой топологией, предназначенная для сбора экспериментальной информации и управления ходом экспериментов [2]. Скорость обмена информацией по сети достигает величины  $3,088 \cdot 10^6$  бит/с. Структура рассматриваемой сети изображена на рис. 1. В сеть объединены мини-ЭВМ PDP-11/34 (ОЗУ — 96 Кбайт, ВЗУ — два НМД емкостью 60 Мбайт каждый) и кластеры персональных ЭВМ Pet фирмы «Commodore» (США). В состав кластера может входить до восьми персональных ЭВМ Pet. Кроме того, мини-ЭВМ PDP-11/34 связана при помощи интерфейса RS-232C с большой ЭВМ Itel AS-6, используемой для сложных расчетов. Каждая из персональных ЭВМ Pet может использоваться как автономная персональная ЭВМ, терминал мини-ЭВМ PDP-11/34 или станция сети.

Сетевое программное обеспечение занимает 2 Кбайт ОЗУ персональной ЭВМ Pet (из имеющихся 8 Кбайт ОЗУ). Мини-ЭВМ PDP-11/34 работает под управлением ОС UNIX, выполняя, в частности, функции файловой ЭВМ.

Альтернативным подходу, связанному с использованием специализированных операционных систем реального времени для создания систем автоматизации экспериментальных исследований, является примененный авторами [3] подход, основанный на том факте, что при использовании быстрого центрального процессора в случае, если темп поступления событий достаточно медленный, различия между процессами реального времени и процессами разделения времени стираются. В таких случаях оказывается возможным в результате модификации обычной операционной системы разделения времени, в частности ОС UNIX, обеспечить одновременную ее работу в режимах разделения времени и реального вре-



мени. При этом основная идея заключается в разбиении программы сбора экспериментальной информации на две части: часть реального времени, встраиваемую в ядро операционной системы и иницируемую на выполнение по прерываниям от таймера; часть разделения времени, являющуюся обычным процессом пользователя и взаимодействующую с частью для реального времени при помощи специально разработанных системных вызовов.

Описываемая методика отработана в системе автоматизации экспериментальных исследований на базе мини-ЭВМ PDP-11/40, работающей под управлением ОС UNIX. Недостаток описываемого подхода заключается в том, что в ядре ОС UNIX возможно реализовать только одну часть реального времени, обеспечивающую возможность проведения сбора экспериментальной информации.

Тщательный анализ ОС UNIX версии 7 с точки зрения возможности ее использования без каких бы-то ни было изменений в режиме реального времени для решения задач автоматизации экспериментальных исследований проведен в 1983 г. в ЦЕРНе — Европейском центре по ядерным исследованиям (Женева, Швейцария) [4]. Необходимость в таком анализе возникла в ходе работ по подключению к сети CERNNET мини-ЭВМ PDP-11/45 (ОЗУ — 256 Кбайт, ВЗУ — НМД емкостью 30 Мбайт), работающей под управлением ОС UNIX версии 7. В результате проведенного анализа выявлены следующие недостатки ОС UNIX, препятствующие использованию этой операционной системы в режиме реального времени: отсутствие возможности записывания процесса пользователя в ОЗУ (т. е. запрета откочки процесса пользователя); отсутствие механизма асинхронного ввода — вывода; отсутствие связи с памятью механизма ожидания события, что приводит к бесконечному ожиданию события в том случае, если оно происходит один-единственный раз и до того момента, как процесс пользователя начинает ожидать его наступления; слабые средства межпроцессных коммуникаций.

В то же время отметим целый ряд достоинств ОС UNIX с точки зрения возможности использования этой операционной системы в режиме реального времени: простоту написания и включения драйверов для новых и нестандартных внешних устройств; удобство, гибкость и большие возможности языка программирования Си; наличие большого числа инструментальных средств различного назначения; высокую скорость отладки программ; высокую скорость генерации ОС UNIX.

Таким образом, можно сделать вывод о пригодности ОС UNIX версии 7 к использованию ее в режиме реального времени после соответствующей модификации.

Положительный опыт по использованию ОС UNIX в режиме реального времени для решения задач автоматизации экспериментальных исследований получен в лаборатории ядерной физики университета штата Нью Джерси (США) [5]. Система автоматизации экспериментальных исследований в этой лаборатории построена на основе мини-ЭВМ PDP-11/44 (процессор арифметики с плавающей точкой FPP, ОЗУ — 1,25 Мбайт, ВЗУ — два НМД кассетного типа RL01 емкостью 5 Мбайт каждый) и аппаратуры КАМАК для сопряжения ЭВМ с экспериментальной установкой, представляющей собой ускоритель и генератор Ван-де-Граафа на 2 МВ. В экспериментах используются три типа источников отрицательных ионов, спектрографы, лазеры. Проводимые в этой лаборатории эксперименты можно условно разделить на четыре класса: эксперименты по ядерной физике низких энергий, атомной физике, физике столкновений и масс-спектрометрии.

Типовой эксперимент характеризуется скоростью поступления экспериментальной информации, не превышающей 10 Кслов/с. Сбор экспериментальной информации осуществляется параллельно с управлением ходом эксперимента (управление пучком). Один из типовых экспериментов имел следующие характеристики: частота наступления событий 300 Гц; число параметров, характеризующих одно событие, 7.

Программное обеспечение рассматриваемой системы автоматизации экспериментальных исследований представляет собой совокупность модифицированной версии ОС UNIX и набора пользовательских программ, обеспечивающих проведение сбора экспериментальной информации, управление ходом эксперимента и обработку собранной экспериментальной информации. Эта обработка частично выполняется в режиме реального времени (т. е. в процессе сбора экспериментальной информации). Для обеспечения возможности использования ОС UNIX в режиме реального времени в нее были внесены следующие изменения: введен системный вызов lock, обеспечивающий возможность записывания процесса пользователя в ОЗУ (т. е. запрета откчки процесса пользователя); модифицирован системный вызов nice с тем, чтобы обеспечить возможность выполнения процесса пользователя на более высоком, чем обычно, приоритете; увеличено до 40 число буферов (т. е. размер пула буферов возрос до 20 Кбайт); защищены от прерываний от аппаратуры КАМАК критические секции программ ядра ОС UNIX, работающих с буферами, путем увеличения их приоритета с 6 до 7; выделен отдельный НМД для организации своппинга.

Драйвер для работы с аппаратурой КАМАК получен из лаборатории ускорителей Саскачеванского университета [1]. Все программное обеспечение, разработанное для рассматриваемой системы автоматизации экспериментальных исследований, реализовано на языке программирования Си. Проведенные изменения ОС UNIX позволили достигнуть высокого темпа сбора экспериментальной информации, составляющего 20 Кслов/с. Отмечается, что основным препятствием для дальнейшего увеличения темпа сбора экспериментальной информации является необходимость периодической записи собранной экспериментальной информации (в данном случае — гистограмм) на НМД. В отсутствие такой необходимости достигается еще более высокий темп сбора экспериментальной информации, составляющий 36 Кслов/с. Поэтому в перспективе планируется переход к использованию электронных дисков. Кроме того, предполагается дальнейшая модификация ОС UNIX с целью более полной адаптации этой операционной системы к работе в режиме реального времени. В частности, намечены дальнейшее увеличение пула буферов, реализация механизма асинхронного ввода — вывода, разработка и реализация новых средств межпроцессных коммуникаций.

Рост интереса к использованию ОС UNIX и совместимых с ней операционных систем в режиме реального времени для решения задач автоматизации экспериментальных исследований нашел свое отражение на проходившей 7—8 октября 1985 г. в Женеве (Швейцария) конференции, посвященной использованию шины VME в физических экспериментах (VMEbus IN PHYSICS), организованной ЦЕРНом. Из представленных на эту конференцию 34 докладов 6 [6—11] так или иначе затрагивали проблемы, связанные с использованием ОС UNIX в режиме реального времени для решения задач автоматизации экспериментальных исследований.

Доклад [6] посвящен обсуждению архитектуры и программного обеспечения мультипроцессорной вычислительной системы MLE, предназначенной для автоматизации экспериментальных исследований в ядерной физике. Структура мультипроцессорной системы MLE изображена на рис. 2. Основой для построения мультипроцессорной вычислительной системы MLE служит шина VME, к которой подключается до 10 микропроцессоров MC68010, каждый из которых имеет локальное ОЗУ емкостью 512 Кбайт, глобальное ОЗУ, емкость которого может достигать 16 Мбайт, процессор, обеспечивающий интерфейс с аппаратурой КАМАК, и мини-ЭВМ, работающую под управлением ОС UNIX. Для мультипроцессорной вычислительной системы MLE разработана операционная система DAMOS, представляющая собой специально спроектированную мультипроцессорную версию ОС UNIX, ориентированную на использование в режиме реального времени для решения задач автоматизации

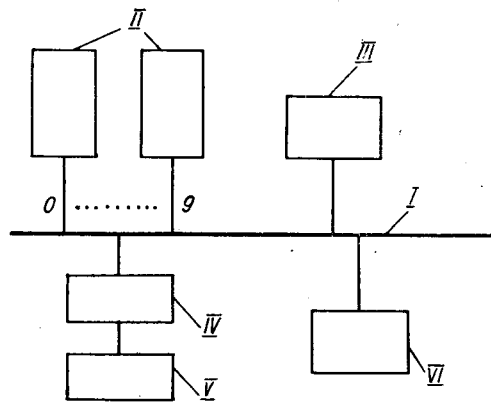


Рис. 2. Структура мультимикропроцессорной вычислительной системы MLE:

I — шина VME; II — микропроцессор MC68010 с локальным ОЗУ емкостью 512 Кбайт; III — глобальное ОЗУ емкостью 2—16 Мбайт; IV — процессор, обеспечивающий интерфейс с аппаратурой КАМАК; V — аппаратура КАМАК; VI — мини-ЭВМ, работающая под управлением ОС UNIX.

экспериментальных исследований. Операционная система DAMOS реализована на языке программирования Си. В процессе разработки ОС DAMOS в ее состав была включена ОС ULRIKE [9]. Ориентация разработчиков на ОС UNIX, язык программирования Си и командный язык shell, который активно используется на прикладном уровне, является следствием их надежды на то, что мобильность программного обеспечения, разработанного в такой операционной среде, позволит избежать необходимости разрабатывать его заново при очередной смене аппаратных средств.

Доклад [7] посвящен обсуждению работ по созданию системы DRM, начатых фирмой Филипс (Голландия) в начале 70-х годов. Они направлены на разработку как архитектуры распределенных мультимикропроцессорных вычислительных систем для работы в режиме реального времени, так и соответствующего программного обеспечения. В рамках этих изысканий создана система — прототип PHIDIAS (предназначенная для использования в области робототехники), на основе которой разработана коммерческая система DRM. Работы над системой PHIDIAS начаты в 1983 г. Файловая система DRM полностью совместима с файловой системой ОС UNIX. В ближайшее время планируется реализовать в полном объеме интерфейс с пользователем, который будет полностью совместим с соответствующим интерфейсом, реализуемым ОС UNIX, что обеспечит возможность разработки программного обеспечения в среде ОС UNIX.

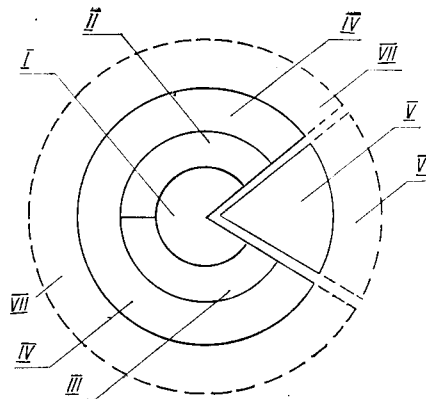
В докладе [8] обсужден предложенный Европейским отделением фирмы «Motorola» протокол обмена при помощи буферизованных каналов, представляющих собой специально разработанный механизм межпроцессорных коммуникаций в режиме реального времени для ОС UNIX. Основой для этого механизма служат метод обмена сообщениями и концепция канала (pipe), введенная в ОС UNIX.

Доклад [9] посвящен обсуждению вопроса использования ОС UNIX в качестве инструментальной системы для создания программного обеспечения, предназначенного для работы в режиме реального времени. Обсуждение основано на опыте, полученном в Мюнхенском университете (ФРГ) в результате разработки исполнительной системы реального времени ULRIKE (ULRIKE = Unix Like Runtime Kernel), совместимой по интерфейсу с пользователем с ОС UNIX и предназначенной для применения как на автономных, так и на подключенных к хост-ЭВМ (работающих под управлением ОС UNIX) рабочих станциях, построенных на базе микропроцессора MC68000. В ходе работ по созданию исполнительной системы реального времени ULRIKE были реализованы следующие компоненты: внутреннее ядро исполнительной системы реального времени ULRIKE, обеспечивающее начальную инициализацию, обработку исключительных ситуаций и включающее в себя драйверы периферийных устройств; библиотеки стандартных функций для всех уровней интерфейса с пользователем; средства коммуникаций с хост-ЭВМ и загрузки через линию связи; интерпретатор команд, очень близкий к интерпретатору команд shell в ОС UNIX.

Структура разработанного программного обеспечения изображена на рис. 3. В качестве инструментальной системы для разработки исполни-

Рис. 3. Структура программного обеспечения ULRIKE:

I — внутреннее ядро; II — загрузчик; III — интерпретатор команд; IV — прикладное программное обеспечение или операционная система DAMOS; V — системная библиотека; VI — библиотека операционной системы DAMOS; VII — программа пользователя



тельной системы реального времени ULRIKE использовалась мини-ЭВМ PDP-11/55, работающая под управлением ОС UNIX (сначала версии System III, а затем версии System V). Опыт применения ОС UNIX в качестве инструментальной системы для разработки программного обеспечения, предназначенного для работы в режиме реального времени, показал, что ОС UNIX хорошо подходит для такой цели. В частности, этому способствуют следующие обстоятельства: 1) принадлежность ОС UNIX к многопользовательским операционным системам разделения времени; 2) иерархическая древовидная файловая система ОС UNIX; 3) чрезвычайно удобный и мощный интерпретатор команд shell; 4) средства поддержки разработки программных комплексов (make и SCCS); 5) наличие исходных текстов ОС UNIX; 6) богатый набор языков программирования (Си, Фортран, Паскаль, Модула 2 и т. д.).

Удобство применения ОС UNIX в качестве инструментальной системы для разработки программного обеспечения, в частности, предназначенного для размещения в ПЗУ, отмечается также в [12].

Доклад [10] посвящен обсуждению использования совместимых с ОС UNIX операционных систем реального времени OS9 и OS9/68K, предназначенных для функционирования в комплекте с рабочими станциями, построенными на базе микропроцессора MC6809 и микропроцессоров семейства MC680xx соответственно. Первоначальная версия этой операционной системы реального времени (OS9) разработана в 1979 г. Обсуждение основано на опыте, полученном в лаборатории физики частиц Коллеж де Франс, Париж (Франция), где OS9 и OS9/68K применяются для проверки оборудования и автоматизации экспериментов по физике высоких энергий, число которых быстро растет. В частности, эксплуатация OS9 позволила в течение всего нескольких дней разработать и отладить программное обеспечение для сбора экспериментальной информации по нескольким 48-канальным АЦП, управления ходом эксперимента и представления результатов эксперимента на экране терминала в виде гистограмм и графическом виде. Кроме того, использование OS9 обеспечило создание «дружественного» пользователю интерфейса с программным обеспечением, управляющим высоковольтной системой в экспериментах по столкновениям протонов и антипротонов. Операционные системы реального времени OS9 и OS9/68K имеют ряд отличий от ОС UNIX: занимают в ОЗУ меньше места, чем ОС UNIX; могут использоваться без НМД (начальная загрузка с НГМД или из ПЗУ); имеют механизмы для монопольного захвата файла целиком и частично (записи); имеют более быстродействующую, чем в ОС UNIX, файловую систему и более быстродействующее ядро (за счет переписывания части ядра на языке Ассемблера с настройкой на архитектуру используемой ЭВМ); обеспечивают упрощение реализации драйверов внешних устройств; имеют средства для работы в режиме реального времени, отсутствующие в ОС UNIX (семафоры, разделяемая память для данных, резидентность в ОЗУ всех процессов, возможность встраивания нового драйвера без перезагрузки операционной системы). В результате удается обеспечить чрезвычайно быстрый отклик на событие, поскольку время реакции на прерывание составляет всего 100 мкс. Структура операционных систем реального времени OS9 и OS9/68K изображена на рис. 4.

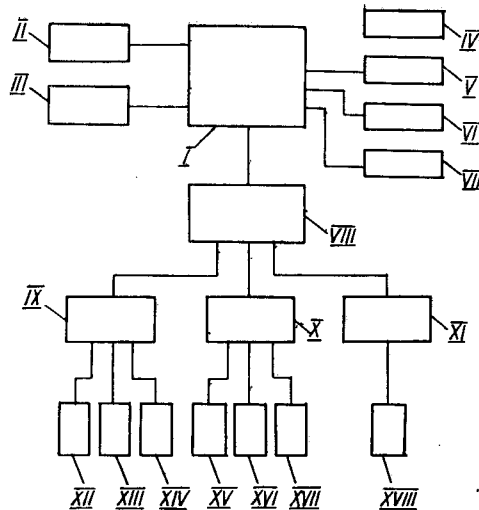


Рис. 4. Структура операционных систем реального времени:

I — ядро операционных систем; II — таблица инициализации; III — драйвер часов; IV, V — библиотеки для вычислений с плавающей точкой (IV) и трансцендентных функций (V); VI — стандартная библиотека ввода — вывода языка программирования Си; VII — обработчики ловушек, установленных пользователями; VIII — супервизоры ввода — вывода (VIII), блочных устройств (IX), байтовых устройств (X) и каналов (XI); XII — XVIII — драйверы НМД, эмулируемого с помощью ОЗУ (XII), НГМД (XIII), НЖМД (XIV), терминала (XV), АЦПУ (XVI), графических устройств (XVII), каналов (XVIII)

В докладе [14] обсуждаются проблемы и перспективы использования ОС UNIX в режиме реального времени. Побудительным мотивом работ в этом направлении, проводимых совместно Ин-

ститутутом информатики и Техническим университетом Западного Берлина, явилось отсутствие в мире операционной системы реального времени, сочетающей в себе такие достоинства ОС UNIX, как мобильность, мощность, распространенность [13]. По мнению авторов [11, 13], правильнее было бы осуществить разработку новой операционной системы реального времени, совместимой с ОС UNIX, чем адаптировать ОС UNIX к использованию в режиме реального времени путем ее модификации. На основе определения того, что собой представляет операционная система реального времени (ОС является таковой, если: 1) программы обработки данных перманентно готовы к выполнению; 2) результаты выполнения программ обработки данных доступны через фиксированный интервал времени от момента начала их выполнения, называемый временем отклика) [14], производится тщательный анализ требований, предъявляемых к операционным системам реального времени. К таким требованиям относятся малое и гарантированное время отклика, малое время переключения с контекста на контекст, наличие критических секций (которые не могут быть прерваны), резидентность процессов в ОЗУ, специальные средства синхронизации и межпроцессных коммуникаций, возможность работы с таймером и доступа к аппаратным средствам, высокая надежность. Затем обсуждаются возможные пути создания совместимой с ОС UNIX операционной среды, обеспечивающей возможность работы в режиме реального времени:

модификация ОС UNIX с целью адаптации ее к работе в режиме реального времени;

разработка и реализация новой операционной системы реального времени, полностью совместимой по интерфейсу с пользователем с ОС UNIX;

реализация в одной из существующих операционных систем реального времени ОС UNIX в виде процесса реального времени; этот подход близок к подходам, использованным при создании операционной системы CBM для ЕС ЭВМ [15] и операционной системы реального времени MERT [16];

применение мультипроцессорной вычислительной системы, в которой один из процессоров работает под управлением ОС UNIX, а на остальных процессорах выполняются процессы реального времени.

Авторы [11] выбрали для реализации последний подход. Структура исходной мультипроцессорной вычислительной системы изображена на рис. 5. Такая мультипроцессорная вычислительная система и программное обеспечение для нее были реализованы с использованием для выполнения процессов реального времени микропроцессоров MC68K.

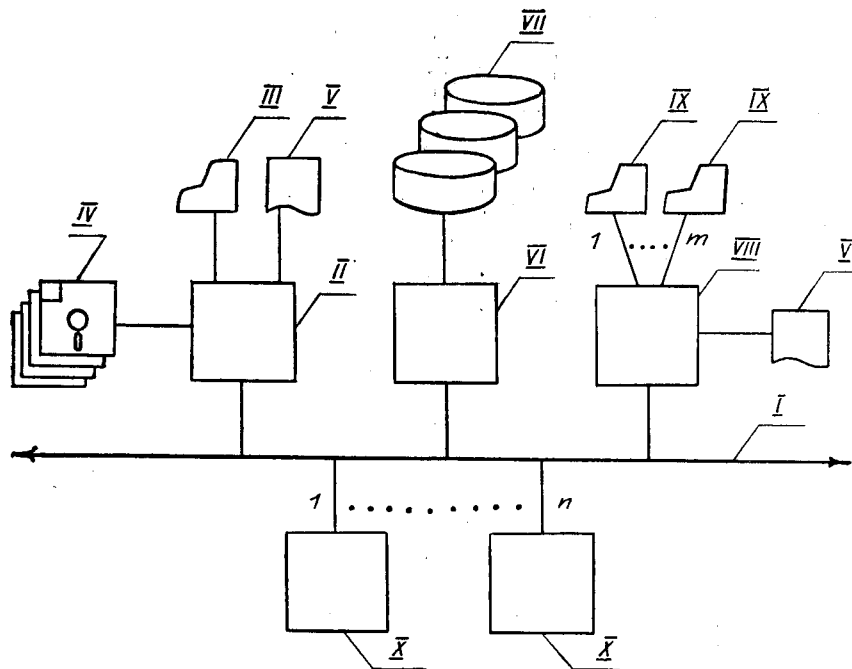


Рис. 5. Структура мультипроцессорной вычислительной системы с разделением функций между процессорами:  
 I — шина VME; II — процессор, работающий под управлением ОС UNIX; III — консольный терминал; IV — НГМД; V — АЦПУ; VI — контроллер НЖМД; VII — НЖМД; VIII — коммуникационный контроллер; IX — терминал; X — процессор, на котором выполняется процесс реального времени

Интересный опыт по эксплуатации ОС UNIX в режиме реального времени в АСУ ТП получен в результате создания АСУ ТП для управления перекачкой нефти в Саудовской Аравии [17], построенной на базе сети PNET (Pilot NETwork). Она представляет собой сеть шинной топологии типа Ethernet со скоростью передачи данных 10 Мбит/с, к которой подключены две микроЭВМ PDP-11/23, одна ЭВМ среднего класса Perkin Elmer 3230 и персональные ЭВМ Опух 8000. Все ЭВМ работают под управлением стандартных версий ОС UNIX без каких-либо модификаций. Построенная таким образом АСУ ТП обеспечивает контроль уровня нефти в баках хранилища и контроль параметров перекачки нефти по трубопроводам (давление, температура, скорость). Технические требования на АСУ ТП предусматривали отображение на экране терминала оператора каких-либо изменений не позднее чем через 15 с после их появления. Проведенные измерения показали, что при 60 %-й загрузке АСУ ТП обеспечивает отображение изменений на экране терминала оператора через 2-3 с после их появления, а при 100 %-й загрузке — через 5 с. В качестве средства межпроцессных коммуникаций использовались каналы (pipes). Все программное обеспечение рассматриваемой АСУ ТП реализовано на языке программирования Си с использованием верификатора lint и средств поддержки разработки программных комплексов make и SCCS. Авторы [17] крайне высоко отзываются об ОС UNIX в качестве как инструментальной системы для разработки программного обеспечения, так и целевой системы.

Основой для создания операционной системы реального времени FADOS, разработанной совместно Голландским национальным институтом ядерной физики и физики высоких энергий и Голландским национальным институтом информатики, для мультипроцессорной вычислительной системы FAMP, построенной на базе микропроцессоров MC68000, послужила ОС UNIX [18]. Разработанные аппаратно-программные средства предназначены для проведения сбора эксперимен-



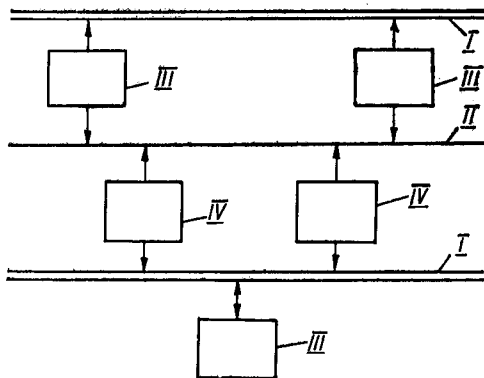


Рис. 6. Структура мультипроцессорной вычислительной системы FAMP:

I, II — внутренняя или внешняя системные шины; III, IV — модули центрального процессора и двухходового ОЗУ

тальной информации и ее анализа в режиме реального времени. Структура мультипроцессорной вычислительной системы, построенной на основе модулей двух типов: микропроцессоров MC68000 и двухходовых ОЗУ — изображена на рис. 6.

Достаточно широкое распространение получило использование ОС UNIX в режиме реального времени для автоматизации океанографических, гидрографических и биомедицинских экспериментов [19—25]. О росте популярности использования ОС UNIX в режиме реального времени свидетельствуют работы [26—31].

Широкое распространение ОС UNIX и увенчавшиеся успехом попытки ее адаптации к использованию в режиме реального времени, предпринятые академическими организациями, привели к тому, что рядом фирм разработаны версии ОС UNIX, обеспечивающие возможность их эксплуатации в режиме реального времени [32—35].

Для работы в таком же режиме предназначены системы Parasite и Concers, представляющие собой надстройки над ОС UNIX, и операционная система реального времени Rex, представляющая собой модифицированную версию ОС UNIX [32]. Операционная система Rex рассчитана на использование в мультипроцессорной вычислительной системе и предназначена для автоматизации лабораторных экспериментов. Сокращение времени реакции на внешнее прерывание обеспечивается в операционной системе Rex за счет введения нового системного вызова attach.

Для работы в режиме реального времени предназначена также операционная система реального времени Masscomp, представляющая собой модифицированную ОС UNIX версии System III, рассчитанную на использование в двухпроцессорной вычислительной системе на базе микропроцессоров MC68000 и предназначенную для сбора поступающей с высокой скоростью экспериментальной информации [33]. Модификация версии ОС UNIX System III свелась к добавлению новых функций и замене в ней некоторых модулей и коснулась в основном системы ввода — вывода и планировщика процессов. В частности, введен набор специальных, более высоких, чем обычно, приоритетов, обеспечивающих возможность гарантировать процессам пользователей выполнение в течение необходимых интервалов времени без прерываний. Кроме того, добавлен механизм непрерывных файлов, существенно сокращающий время ввода — вывода. Наконец, применен механизм асинхронных прерываний, обеспечивающий возможность проведения полностью асинхронного ввода — вывода. Модификация механизма переключения с контекста на контекст повысила его эффективность в 2 раза. В результате всех изменений удалось сократить время отклика до величины 1—4 мс, а в случае использования дополнительных периферийных процессоров с прямым доступом к памяти — до величины, не превышающей 10 мкс.

Фирма «Bull» разработала мультипроцессорную вычислительную систему SPS-7 и программное обеспечение для нее, рассчитанные на использование в режиме реального времени [34]. Мультипроцессорная вычислительная система SPS-7 построена на основе шины, к которой может подключаться до 8 модулей микропроцессоров MC68010 (с локальным ОЗУ емкостью 10 Мбайт у каждого), модуль глобального ОЗУ

емкостью до 16 Мбайт, до 16 модулей обмена (обеспечивающих управление сетью и (или) сбор экспериментальной информации). Программное обеспечение мультипроцессорной вычислительной системы SPS-7 представляет собой версию ОС UNIX с монитором реального времени. Разработанные фирмой «Bull» аппаратно-программные средства предназначены для решения задач автоматизации экспериментальных исследований и создания АСУ ТП.

Операционная система RTIX является версией ОС UNIX, предназначенной для использования в режиме реального времени на ЭВМ типа OSI 710 и OSI 720, характеристики которых приведены в [35].

В связи с появлением в СССР совместимых с ОС UNIX операционных систем ИНМОС, ДЕМОС, МНОС [36—38] возник интерес к их использованию в режиме реального времени [39, 40].

В [39] описана адаптация операционной системы ИНМОС к работе в режиме реального времени, осуществленная за счет введения понятия процесса реального времени и набора привилегированных системных вызовов. Эти системные вызовы являются привилегированными ввиду их потенциальной опасности в случае некорректного использования, поэтому процессы реального времени выступают от имени привилегированного пользователя. В частности, введены следующие привилегированные системные вызовы:

poswar: запретить откачку процесса пользователя из ОЗУ;

swar: разрешить откачку процесса пользователя из ОЗУ;

nsched: запретить перепланирование процессов;

sched: разрешить перепланирование процессов;

setpri: установить приоритет процессора в соответствии со значением аргумента системного вызова (в диапазоне 0—7).

Асинхронный ввод — вывод предложено реализовать посредством механизмов параллельных процессов и межпроцессных коммуникаций.

В [40] описана адаптация операционной системы ДЕМОС к работе в режиме реального времени, осуществленная за счет модификации планировщика процессов и планировщика свопинга. Введено 32 уровня приоритета (0—31), из которых 16 уровней соответствуют обычным процессам (0—15), а 16 уровней — процессам реального времени (16—31). Очередь на выполнение строится в соответствии с приоритетами процессов таким образом, что на выполнение всегда выбирается наиболее приоритетный процесс из числа готовых к выполнению процессов, находящихся в ОЗУ. Для обычных процессов введено ограничение на время непрерывного использования центрального процессора за счет установления обратно пропорциональной зависимости между величиной кванта и уровнем приоритета для этой категории процессов. При появлении более приоритетного процесса, готового к выполнению и находящегося в ОЗУ, возможно переключение с процесса до истечения выделенного выполняющемуся в данный момент процессу кванта. В этом случае процесс, выполнение которого было прервано, помещается в очередь на выполнение первым среди процессов с равным ему приоритетом. Планировщик подкачки-откачки модифицирован таким образом, что откачке (в случае необходимости ее проведения) подвергается наибольший по размеру процесс, имеющий уровень приоритета ниже, чем у подкачиваемого процесса, или процесс, не готовый к выполнению. Процессы реального времени и процессы, находящиеся в состоянии выполнения системного вызова ехес, откачке не подвергаются. В результате проведенных модификаций операционная система ДЕМОС, работающая на мини-ЭВМ СМ-4, обеспечивает время отклика, не превышающее 5 мс.

Таким образом, можно сделать вывод о том, что использование ОС UNIX и совместимых с ней операционных систем в режиме реального времени расширяется. Это расширение объясняется целым рядом обстоятельств. Среди них необходимо отметить следующие: широкое распространение операционных систем обсуждаемого типа; наличие опера-

онных систем этого типа для ЭВМ различных классов; удобство интерфейса с пользователем; наличие богатого инструментария; мобильность программного обеспечения, разработанного для эксплуатации под управлением рассмотренных операционных систем; мобильность пользователя; наличие перспективы превращения ОС UNIX из стандарта «де-факто» в стандарт «де-юре», что обеспечит стандартизацию всех уровней интерфейса с пользователем.

Анализ модификаций ОС UNIX и совместимых с ней операционных систем с целью их адаптации к работе в режиме реального времени показывает, что основными средствами такой адаптации являются:

- введение возможности записи процесса пользователя в ОЗУ;
- модификация планировщика процессов;
- увеличение размера пула буферов;
- реализация механизма асинхронного ввода — вывода;
- разработка новых механизмов межпроцессных коммуникаций.

дарту на интерфейс с пользователем для ОС UNIX версии System V [42], международная ассоциация производителей средств вычислительной техники X/OPEN, издавшая в 1985 г. справочное руководство, представляющее собой фактически предварительный стандарт на большую часть ОС UNIX [43], и рабочая группа IEEE P1003 (вышедшая из ассоциации пользователей ОС UNIX/usr/ group, издавшей еще в 1984 г. предварительный стандарт на ОС UNIX [44]), разрабатывающая стандарт на ОС UNIX, называемый Posix.

В свете предпринимаемых усилий по стандартизации ОС UNIX и совместимых с ней операционных систем целесообразно было бы провести предварительно стандартизацию средств, обеспечивающих возможность использования рассмотренных операционных систем в режиме реального времени.

#### ЛИТЕРАТУРА

1. Custead L. R., McAlpine J. L. Nuclear physics data acquisition with the UNIX time-sharing system // IEEE Trans. Nucl. Sci.— 1979.— NS-26.— P. 949.
2. Pohm A. V., Davies J. A., Christiansen S. e. a. A local network of mini- and micro-computers for experimental support // Comput. Networks.— 1979.— 3.— P. 381.
3. Harland D. M. High-speed data acquisition: running a realtime process and a time-shared system (UNIX) concurrently // Software — Pract. and Exper.— 1980.— 10, N 4.— P. 273.
4. Lee Q., Wiegandt D. Interfacing CAMAC to a UNIX system // Interfaces in Computing.— 1983.— 1.— P. 329.
5. Shu N. K. B. Data acquisition with a PDP-11/44 and MBD-11 branch driver with the UNIX operating system // IEEE Trans. Nucl. Sci.— 1983.— NS-30, N 5.— P. 1750.
6. Beier G., Kappen L., Lutter R. e. a. MLE — a dataflow controlled multiprocessor system // VMEbus in Physics Conf.: Proc.— Geneva, Switzerland: CERN, 1985.— P. 151.
7. Maaskant H. DRM system: A distributed real-time multiprocessor system // Ibid.— P. 238.
8. Werdehausen J. Buffered pipe protocol // Ibid.— P. 242.
9. Beier G., Kappen L., Lutter R. e. a. Cross software development in a UNIX environment // Ibid.— P. 276.
10. Fontaine G., Guglielmi L. Use of OS9/68K — A UNIX-like real-time operating system — in particle physics VMEbus applications // Ibid.— P. 318.
11. Buchheim P., Schacht J. Problems and chances of real-time data processing in UNIX-like operating system // Ibid.— P. 329.
12. Ward J. R. UNIX as an environment for non-UNIX software development: a case history // ACM SIGSOFT Software Eng. Notes.— 1985.— 10, N 3.— P. 95.
13. Schacht J. Wie bekommt man das Kamel durchs Nadelöhr? // Elektronik.— 1985.— N 26.— P. 111.
14. DIN 44 300, N 161.

- re-grass and exper.—1980.—10, N 5.—P. 425.
19. **Guberek M., Borders S., Masse S.** A digital image processing workstation for the ocean sciences // Ocean 85. Ocean Engineering and Environment: Conference Record.—San Diego, California, USA, 1985.—P. 330.
  20. **Gordon S., Borders S., Guberek M.** An interactive data processing workstation for the ocean sciences // Proc. 1986 Working Symposium on Oceanographic Data Systems.—San Diego, California, USA, 1986.—P. 52.
  21. **Delahoyde F. M.** A UNIX-based system for hydrographic data acquisition and processing // Ibid.—P. 57.
  22. **Abbot J. L., Smith S. M., Charters J. S. e. a.** Scripps seagoing computer centers: real-time data acquisition and processing // Ibid.—P. 123.
  23. **Voss A. J. R., Sherwood J., Jackson C. S. e. a.** A standard-based distributed oceanographic computing system // Ibid.—P. 145.
  24. **Webber C. L.** A C-language program for the computation of power spectra on a laboratory microcomputer // Comput. Methods and Programs for Biomedicine.—1986.—22, N 3.—P. 285.
  25. **Hanriot J.-M.** A digital measurement system under UNIX // Electron. Ind.—1986.—N 105.—P. 39.
  26. **Doyle S. J., Bunce P.** Real-time C for UNIX programmers // Electro/86 and Mini/Micro Northeastern Conference Record.—Boston, Massachusetts, USA, 1986.—P. 22/3/1.
  27. **Matlox J.** Using UNIX in a real-time environment // Ibid.—P. 25/0/1.
  28. **Romberger A.** Real-time work under a standard UNIX kernel: possible drawbacks and kernel enhancements // Ibid.—P. 25/1/1.
  29. **Beker-Harvey M.** Real-time enhancements to the UNIX kernel // Ibid.—P. 25/2/1.
  30. **Allen B.** Regulus: a real-time multiprocessor operating system // Ibid.—P. 25/3/1.
  31. **Fiddler J., Wilner D. N.** VXWORKS/UNIX real-time network and development system // Ibid.—P. 25/4/1.
  32. **Attention** focuses on UNIX's move into process control // Electron. Design.—1980.—30, N 18.—P. 88.
  33. **Brown G.** Options abound in 16-bit operating software // Comput. Design.—1983.—22, N 9.—P. 113.
  34. **Gallagher R. T.** Multiprocessor runs real-time code UNIX // Electron. Weekly.—1985, May 13.—P. 24.
  35. **Multiuser** system has software for real-time processing // Ibid.—May 27.—P. 70.
  36. **Беляков М. И., Ливеровский А. Ю., Семик В. П. и др.** Инструментальная мобильная операционная система ИНМОС.—М.: Финансы и статистика, 1985.
  37. **Всесоюз. науч.-техн. конф. «Программные средства как продукция производственно-технического назначения». Секц. Стандартизация и унификация программных средств:** Тез. докл.—Калинин: НПО ЦЕНТРОПРОГРАММСИСТЕМ, 1985.
  38. **Антонов В. Г., Аншуков С. А., Бардин В. В. и др.** Операционная система ДЕМОС 2.0 // Всесоюз. науч.-техн. семинар «Диалоговая мобильная операционная система ДЕМОС»: Тез. докл.—Калинин: НПО ЦЕНТРОПРОГРАММСИСТЕМ, 1986.
  39. **Грязнов Д. О., Леонас В. В.** Адаптация машинно-независимой операционной системы к работе в режиме реального времени // Всесоюз. науч.-техн. конф. «Проблемы создания и использования мини- и микроЭВМ»: Тез докл.—Москва; Вильнюс: ИНЭУМ, 1985.
  40. **Бурков Д. В.** Новый планировщик процессов ОС ДЕМОС // Всесоюз. науч.-техн. семинар «Диалоговая мобильная операционная система ДЕМОС»: Тез. докл.—Калинин: НПО ЦЕНТРОПРОГРАММСИСТЕМ, 1986.
  41. **Zintz W.** UNIX seeks unity // Can. Datasystems.—1985.—P. 55.
  42. **System V Interface Definition:** AT&T, 1985.
  43. **X/OPEN Portability Guide:** Elsevier Science Publishers, 1985.
  44. **1984/usr/group Standard.**

*Поступила в редакцию 26 декабря 1986 г.*