

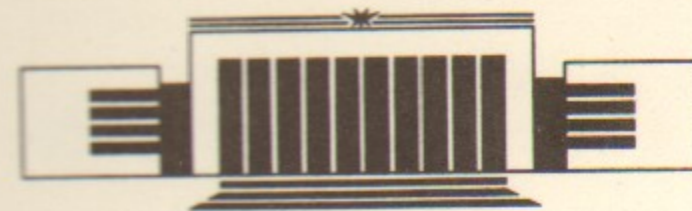


41
ИНСТИТУТ ЯДЕРНОЙ ФИЗИКИ СО АН СССР

А. Н. Алешаев

**ПРОГРАММНОЕ ОБЕСПЕЧЕНИЕ
ДЛЯ МИКРОЭВМ ОДРЕНОК.
Операционная система
ОДОС**

ПРЕПРИНТ 89-67



НОВОСИБИРСК

Программное обеспечение
для микроЭВМ ОДРЕНОК.
Операционная система ОДОС

А.Н. Алешаев

Институт ядерной физики
630090, Новосибирск 90, СССР

АННОТАЦИЯ

Описывается один из вариантов операционной системы микроЭВМ ОДРЕНОК—система ОДОС. Приводятся ее спецификации, рассматриваются алгоритмы реализации основных функций, даются представления о внутренней организации системы. Особое внимание уделено пользовательскому интерфейсу, базирующемуся на командном языке и экстракодовой технике взаимодействия программ с ОС. Описываются механизмы настройки и функционирования ОДОС. Появляется схема организации сетей рабочих станций на базе Одрят, применяемых в ИЯФ для автоматизации ряда установок и экспериментов и на рабочих местах сотрудников института. Приводятся протоколы обмена между периферийными и центральной станциями этих сетей. Описание сопровождается ссылками на программные средства сервисной оболочки операционной системы и на программы-утилиты.

1. ВВЕДЕНИЕ

Для более эффективного использования КАМАК-аппаратуры в системах управления различными установками в ИЯФ СО АН СССР в 1983 году был разработан крейт-контроллер ОДРЕНОК с системой команд универсальной ЭВМ серии ICL-1900 (польские аналоги этих машин—ОДРЫ—использовались в ИЯФ для управления УНК ВЭПП-4 и рядом других установок и экспериментов [1]). По существу Одренок представляет собой полноценную микроЭВМ, дополненную функциями крейт-контроллера. Своим 24-разрядным словом он удачно сочетается с разрядностью шин данных КАМАК-магистрали. По внутренней организации Одренок—это ЭВМ с современной регистровой архитектурой. В его командах разрешено применение 8 регистров общего назначения (РОН), называемых аккумуляторами. Еще один 48-разрядный регистр используется для операций с плавающей точкой (полное количество регистров 16). Основной набор команд ориентирован на оперативную память (ОП) размером до 32К слов, но имеются специальные команды для доступа к ОП размером до 64К. Целые числа занимают одно слово (диапазон от $-8\ 388\ 607$ до $+8\ 388\ 607$), числа с плавающей точкой—два (диапазон от $-5.6 \cdot E^{+76}$ до $+5.6 \cdot E^{+76}$ с представлением мантиссы 38 двоичными разрядами, что соответствует 11 десятичным). Отрицательные числа представляются в дополнительном коде. Специфичным является набор команд для операций с информацией символьного типа, для кодирования которой используются укороченные 6-разрядные «байты»—по 4 в слове (функции расширения алфавитов в этом внут-

реннем формате выполняют 3 выделенных символа). Система команд (примерно 120 различных типов операций) представляет обширные возможности для арифметических и логических преобразований над словами одинарной и двойной длины и включает большой набор команд сдвигов и переходов. Помимо команд условных переходов по содержимому РОН есть команды ветвления по состоянию двух флагов: V — индикатора переполнения и C — флага переноса. Реализованные в процессоре Одренка 4 способа адресации позволяют строить компактные программы обработки числовых массивов и текстов. Имеется аппаратная поддержка мультипрограммной работы, состоящая из:

- системы переключения в привилегированный режим СУПЕРВИЗОР по прерыванию с КАМАК-магистрали (LAM) или по экстракоду (команде, не имеющей аппаратной реализации) с передачей управления в 29 и 28 адреса ОП соответственно. Состояние процессора на момент прерывания сохраняется в ячейках 0—10 оперативной памяти прерванной задачи;

- двух специальных команд: команды установки регистра базы адреса (РБА) и команды восстановления состояния процессора из области памяти, адресуемой этим регистром. Эта команда переводит процессор в режим ЗАДАЧА.

В режиме ЗАДАЧА разрешены оба вида прерываний и ко всем адресам при обращении к ОП добавляется содержимое РБА. В режиме СУПЕРВИЗОР прерывания с магистрали запрещены и РБА не используется.

Прерывания по экстракодам разрешены в обоих режимах и обрабатываются одинаково. Часть экстракодов специально выделена для обращения к функциям операционной системы (ОС).

Внешние устройства (ВУ) подключаются к Одренку только через КАМАК-магистраль. Для взаимодействия с ними помимо обычной команды генерации КАМАК-цикла в микроЭВМ предусмотрен ряд эффективных команд для блочного обмена. Специальная команда чтения LAM-регистра позволяет оперативно определять наличие прерываний с магистрали. Управление этими прерываниями (разрешение/запрет) может быть реализовано через маску в 15 регистре.

Имеется также и некоторый аналог прямого доступа (ПД).

Производительность Одренка на операциях с фиксированной точкой составляет около 300 тысяч операций/с, с плавающей 20—30 тысяч.

Более детальное описание Одренка содержится в [2].

Одним из основных требований при создании Одренка было требование применимости для него систем подготовки программ с основных машин управления: ЭВМ серии ОДРА-1300. Вносимое этим упрощение программирования и отладки программ для крейт-контроллеров значительно облегчило бы процесс интеграции КАМАК-аппаратуры в существующие системы управления, повысило бы эффективность ее использования. В частности, на качественно новом уровне могли быть решены задачи визуализации информации, под которые в системах управления на ОДРАХ применялись видеоконтроллеры, выполненные также в конструктиве КАМАК (что, помимо загрузки процессоров ЭВМ, существенно загружало и их каналы).

Однако, высокая функциональная насыщенность КАМАК-аппаратуры позволяла размещение в одном крейте набора модулей, достаточного для решения нескольких, вообще говоря, разнородных подзадач управления. Возможности элементной базы микропроцессорной техники 1981—83 гг. также допускали в 2М крейт-контроллере получать необходимую для этого вычислительную мощность.

С другой стороны, опыт, приобретенный при создании сложных программных комплексов на базе ЭВМ ОДРА, показал неэффективность совмещения большого количества задач в одной программе [3].

Поэтому еще на стадии проработки архитектуры контроллера требования к нему были расширены до возможности организации в нем многозадачного режима. Последовавшая отсюда необходимость поддержки дисциплины мультипрограммного исполнения этих задач с учетом пожелания иметь единообразные среды функционирования программ в ОДРЕ и Одренке привела к целесообразности создания для Одренка достаточно полноценной ОС.

Первый вариант ОС — программа ОДОС для работы в Одренке и программа-посредник МОДО для ОДРЫ — был создан к концу 1983 года. Базовые концепции, реализованные такой распределенной ОС, были следующие:

- формирование для программ в Одренке операционной среды основных машин управления, в частности, путем предоставления им ряда ресурсов этих машин;
- преемственность командного языка, средств межпрограммного взаимодействия и синхронизации в системах на Одрях с системами для ОДР;
- предоставление системных средств обмена данными между

- ОДРОЙ и подключенными к ней станциями с Одрятами;
- управление станциями с терминалов ОДРЫ (виртуальные терминалы — ВТЛ);
 - поддержка минимальной локальной периферии общего назначения (терминалы (ТЛ), электронный диск (ЭД), каналы связи с аппаратурой управления вне крейта);
 - предоставление необходимых услуг по взаимодействию с КАМАКом;
 - динамическая настройка ОС на конфигурацию станции в фазе инициализации (без регенерации) по файлу описания станций с Одрятами (ФОС) для данной ОДРЫ;
 - подключение к одной ОДРЕ до 8 станций с Одрятами.

В таком варианте Одрята примерно 2 года использовались в управлении ряда крупных установок и экспериментов. Открытая, унифицированная магистраль подключения внешней аппаратуры, модульное построение программного обеспечения (ПО) упростили процедуры внедрения новых аппаратных средств автоматизации. Достаточно высокая степень автономности станций с Одрятами, особенно при оснащении их ЭД и ЛТЛ обусловила и эффективность использования их и как инженерных или персональных рабочих станций сотрудников института. При этом выяснилось, что при использовании в последнем качестве, да и при автоматизации ряда экспериментов, центральные узлы — машины ОДРА — стали служить в основном как файловые машины.

Появление в 1985 году компактных винчестерских дисков, сопряжение их контроллеров с КАМАК-магистралью инициировало постановку файловых машин для Одрят уже на самих Одрятах. Это привело к существенному увеличению мест их применения (сняв проблему подключения их к ОДРАМ, располагаемым на ВЦ, проблему малой емкости дисков этих машин, используемых под совместную файловую систему (ФС) самой ОДРОЙ и подключенными к ней микроЭВМ, ...). Относительная дешевизна микроЭВМ, довольно высокая ее производительность, переход к управлению, основанному преимущественно на КАМАК-аппаратуре, а также ряд других причин предопределил в последующем и перевод управления установок института, использовавших машины ОДРЫ, с сетей ОДРА — Одрята на полностью однородные сети Одрят.

Адаптация к новым условиям применения микроЭВМ и удовлетворение ряда дополнительных пожеланий к операционной среде потребовали некоторой модернизации их базового ПО, включаю-

щей частичную трансформацию функций центральной машины (ЦМ), доработку ОДОС и создание ряда новых утилит и библиотек подпрограмм.

Здесь описывается достаточно установившийся вариант программы ОДОС (версия 08.6) — специализированной операционной системы микроЭВМ Одренок, применяемой на станциях, связываемых с центральным узлом через интерфейсный модуль (ИМ) У0605. Блок-схема этой сети с типичными конфигурациями ЦМ и ПМ приведена на рис. 1.

Существенную помощь в разработке и создании данной системы оказали сотрудники ИЯФ С.Д. Белов и Б.В. Левичев. Постановку программы МОДО на Одренке, подключение винчестерского диска к ним осуществил В.Р. Козак. Он же разработал ряд необходимых утилит и подпрограмм.

Ниже, когда речь идет об ОС, подразумевается в основном ее ядро — программа ОДОС. По ходу описания делаются ссылки на подпрограммы, предоставляющие необходимый интерфейс и дополнительный сервис к тем или иным функциям ОС, или на утилиты, реализующие разнообразные услуги по использованию ресурсов ВС.

Информацию об организации ЦМ и ее ПО можно найти в [4].

2. СПЕЦИФИКАЦИИ ОС ОДОС

Основные характеристики ОС ОДОС:

- многопрограммная, многотерминальная, диалоговая система реального времени для микроЭВМ Одренок, специализированная для работ по автоматизации на базе КАМАК-аппаратуры.

Спецификации ОДОС:

- обеспечение мультипрограммного исполнения до 10 программ с гибкими средствами их загрузки и удаления;
- подключение 4 ЛТЛ и 1 ВТЛ;
- поддержка ФС на ЭД (128—512К слов) и доступа к ФС ЦМ;
- реализация командного языка на базе примерно 30 директив со встроенным исполнением;
- предоставление средств межпрограммного обмена;
- организация специальных режимов функционирования программ;
- ведение календаря;
- синхронизация программ с событиями реального времени;

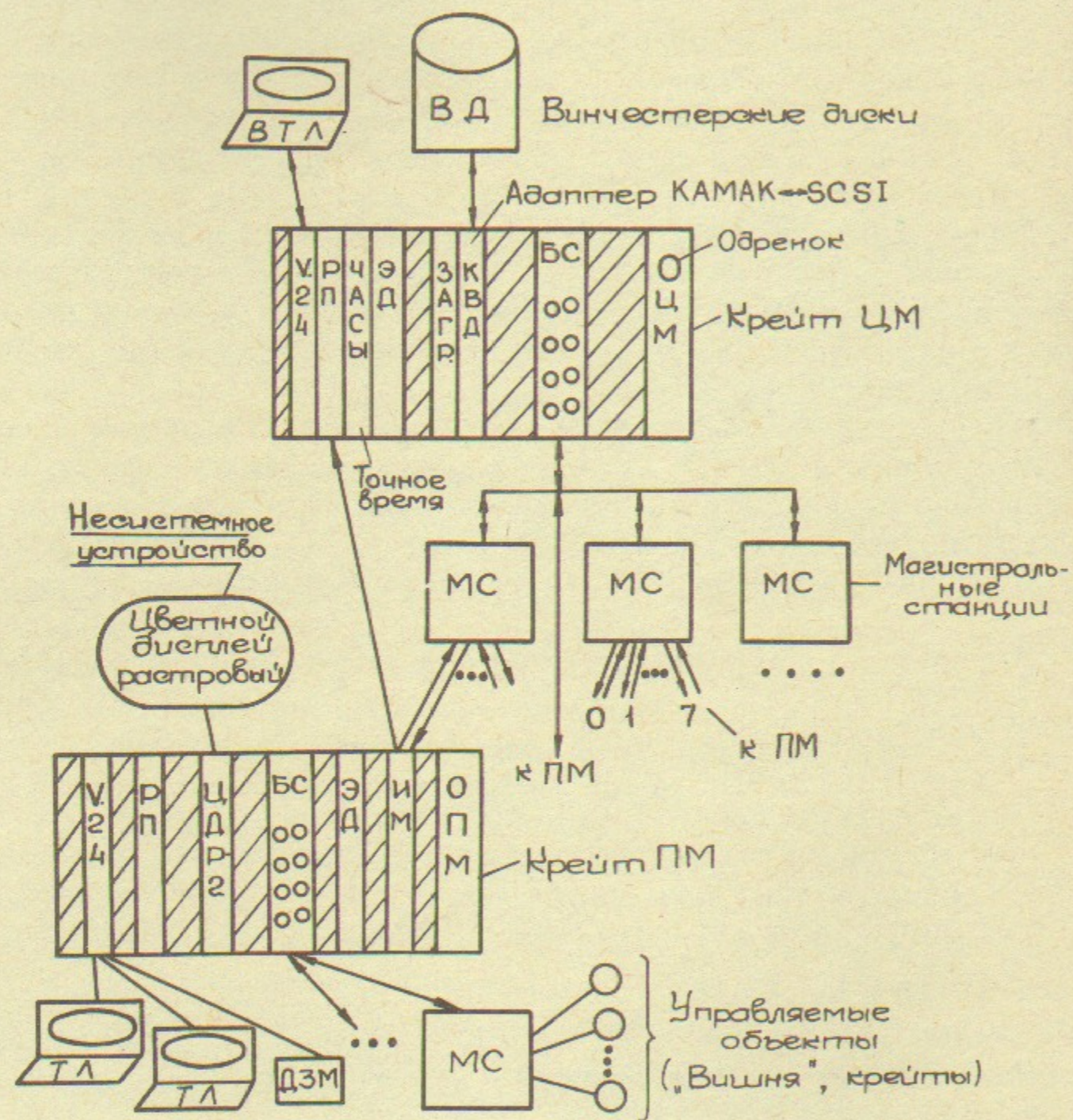


Рис. 1. Схема подключения Одрят с ОС ОДОС.

- реализация вспомогательных логических каналов обмена с ЦМ;
- поддержка каналов связи со специальной аппаратурой управления;
- дополнение аппаратных возможностей микроЭВМ (программное выполнение некоторых арифметических команд для ранних версий Одренка).

В разделах 2—4 содержится общая информация об основных функциях системы, ее структуре и процедуре загрузки. Последующие разделы посвящены описанию способов взаимодействия с ОС с терминалов и из программ. В приложениях приведена информация о механизмах взаимодействия периферийной и центральных машин, деталях настройки ОДОС на конкретную рабочую станцию и некоторые другие справки.

При ссылках на команды используется восьмеричная запись их кода, а в скобках приводится соответствующая ассемблерная мнемоника. Биты в машинном слове нумеруются слева направо и обозначаются Б0, Б1, ..., Б23.

2.1. Управление памятью и мультипрограммирование

ОДОС представляет собой резидентную программу, занимающую младшие адреса ОП (0—8К). Далее идет память программ. Каждой программе она отводится непрерывной областью с кратностью размера 256 слов. По соглашению о формате объектного кода программ (15-битовые адресные поля экстракодов) объем кодовой части программ не должен превышать 32К.

Программам разрешается по мере необходимости увеличивать свой размер с помощью одной из форм 165-го экстракода. Для программы, загруженной последней, разрешается и уменьшение.

При удалении программы из ОП операционная система сразу же предпринимает попытку реорганизации памяти (сжатие: перемещение вышележащих программ на освобождающееся место). Реорганизация может быть задержана из-за наличия незавершенных обменов с ВУ с прямым доступом в области, подлежащие перемещению. В такой фазе помеченные к удалению программы присутствуют в списке программ под именами «% % % %» и реально будут удалены по завершении этих обменов.

Программы могут быть загружены в оперативную память с ЭД или дисков ЦМ директивами LO, FI (а активированы директивой GO). В рамках ОДОС программы идентифицируются 4-символь-

ным именем и номером терминала, к которому они приписаны. Имена вне зависимости от ТЛ-номера должны быть различными, что является следствием сложившихся механизмов межпрограммного взаимодействия в ПО систем управления на ОДРАХ [5], перенесенных в системы для Одрят. Для одновременного использования нескольких копий одной программы предусмотрена директива загрузки программы с переименованием (LR).

Каждая загруженная в ОП программа описывается в ОДОС отдельной таблицей, где, помимо имени и номера ТЛ, содержится информация об адресе начала программы в ОП, ее размере и еще о двух специальных атрибутах: **приоритете** и **слове состояния** (ССП). Изначально эти атрибуты формируются системами подготовки программ. При загрузке в память приоритет и начальное значение ССП берутся из блока запросов программы, сохраняемого вместе с программой при помещении ее на внешние носители. Изменить приоритет программы, находящейся в ОП, можно директивой RE. ССП используется системой для учета причин приостановок программы.

В ОДОС отведена память под 10 таблиц описания программ, т. е. 10 программ, помимо самой ОДОС, могут одновременно находиться в ОП и конкурировать за предоставление им процессорного времени и других ресурсов ВС. На ОС процессор переключается внеконкурентным способом по возникновении любого прерывания (LAM, экстракода) во время работы программы. Значительную часть своих функций ОДОС выполняет за одно такое переключение. Но есть и задания, выполнение которых требует реализации определенных условий или же взаимодействия с медленными ВУ, причем несколько таких заданий могут пересекаться во времени. Экономичная схема их выполнения требует специальной дисциплины мультипрограммирования внутри ОС. В ОДОС эта дисциплина базируется на системных процессах, представляющих собой нечто похожее на программы, вкомпонованные в тело системы. Процессы, как и программы, организуются в приоритетный список и наделяются своими словами состояний.

Таким образом, в ОДОС имеются два типа объектов, мультипрограммное исполнение которых подлежит координации: программы и системные процессы. Далее они называются задачами.

В основе мультипрограммирования в ОДОС, как и в других ОС, лежит использование пауз, возникающих при синхронизации задач с событиями «внешнего мира». ОДОС предоставляет несколько механизмов для такой синхронизации:

- синхронизация с работой коммунальной периферии;
- ожидание LAM-сигналов от модулей крейта через ОС при прямой работе программ с КАМАК-аппаратурой;
- приостановки на определенные интервалы времени;
- ожидание сигналов от специального внешнего регистра прерываний.

Дисциплина предоставления процессорного времени реализуется одной из компонент ОС — планировщиком. Процедура эта запускается всякий раз после обработки прерывания, сменившего состояние какой-либо задачи. Атрибутом задач, определяющим для планировщика порядок предоставления им процессорного времени, является приоритет, по которому процессы/программы упорядочены в списке задач системы (для программ при равных приоритетах преимущественными правами пользуется программа, загруженная позднее). Сама же возможность постановки задачи на счет определяется по ССП. Более строго алгоритм выбора задачи для счета формулируется следующим образом: процессор всякий раз предоставляется задаче с наивысшим приоритетом, не находящейся в состоянии ожидания чего-либо. Для программ состояния ожидания могут быть по явным причинам (например, останов директивой SU, ожидание окончания обмена с ВУ, ...), а также и по неявным, связанным с занятостью тех или иных ресурсов системы. Состояние программы можно определить на терминале с помощью директивы PR. Состояние системных процессов скрыто, и о нем можно судить лишь косвенно, например, по отказам выполнения возложенных на них функций.

Во всех случаях системные процессы имеют более высокие приоритеты, чем пользовательские программы.

К дополнительным средствам мультипрограммирования в ОДОС относится возможность выделения в программах специальных фрагментов, предназначенных для асинхронной обработки прерываний от модулей крейта. Это так называемый механизм «LAM-драйвера», в соответствии с которым при возникновении указанных прерываний ОДОС переключает программу на ее часть, объявленную LAM-драйвером, и предоставляет ей возможность в течение 100-200 мс быть активной безотносительно к ССП. Алгоритм планировщика при этом сохраняется, но при анализе условий активации данной программы в первую очередь проверяются условия для «драйверной» части программы. Если эти условия выполнены, то активируется драйверная часть. По истечении указанного интервала запуск программы будет осуществляться в соответствии

ее ССП (переключения на «прерванную» часть программы здесь нет, и, если драйверная часть не успела выполнить свои действия, она может завершить их при запуске программы нормальным способом; детали см. экстракод 163).

Имеется одно существенное ограничение на программирование драйверных частей программы, обусловленное отсутствием реентерабельности в применяемых пакетах подпрограмм: нельзя обращаться в драйверной части к подпрограммам, используемым в других местах программы. Это связано с возможностью переключения программы на драйвер в фазе, когда такая подпрограмма выполнялась по запросу из другой части программы.

Реализован также и примитивный механизм разделения времени: ОС циклически с периодичностью 100 мс меняет очередность функционирования программ с нулевыми приоритетами.

В заключение подраздела следует упомянуть, что ОС функционирует целиком в режиме СУПЕРВИЗОР, т. е. при полном запрете внешних прерываний. Программы же функционируют в режиме, разрешающем все виды прерываний, возникновение которых в конечном счете и инициирует «срабатывание» планировщика.

2.2. Взаимодействие с внешними устройствами

Одренок связан с «внешним миром» только через КАМАК-магистраль, т. е. все периферийное оборудование для подключения к нему должно иметь соответствующий КАМАК-интерфейс.

В архитектуре Одренка заложено два способа взаимодействия с КАМАК-магистралью: первый — командами:

- 170 — одиночный КАМАК-цикл,
- 174 — блочный КАМАК-цикл,
- 175 — сканирование по субадресам и номерам станций крейта (в последних версиях есть еще и блочные команды для байтового обмена)

и второй — «прямой» доступ к ОП, выполняемый микропрограммно и управляемый специальным аппаратным протоколом между модулем крейта и Одренком с использованием резервных КАМАК-шин P1, P2 (вместо L и N) [2].

Оба способа явно или неявно занимают процессорное время. Первый из них инициируется и управляется программным обеспечением и дополняется механизмами управления ЛАМ-прерываниями через маску в 15-м регистре (вплоть до полного их запрета командой 173), заложенными в Одренке. Второй реализуется

аппаратно и инициируется для ПМ внешним устройством.

В описываемой системе используются оба способа.

Доступ к ВУ из программ, работающих под управлением ОДОС, может осуществляться двумя путями:

- через сформированные средствами ОС каналы обмена с ними посредством экстракодов;
- непосредственно через КАМАК-магистраль аппаратными командами.

Первый путь предназначен для доступа к ВУ, представляющим собой коммунальные ресурсы ВС (т. е. такие, которые могут использоваться совместно рядом задач). Второй — для работы с уникальным или принципиально неделимым оборудованием.

Формирование каналов доступа к ВУ в ЭВМ, не имеющей специализированного канального оборудования, является чисто техническим приемом, позволяющим эффективно решить проблему разделения коммунальных устройств между разными задачами и унифицировать большой объем процедур обслуживания этих ВУ. Каналы в ОДОС — это в основном программные структуры. Со стороны пользователей они начинаются процедурами интерфейса к ВУ, а со стороны ВУ завершаются драйверами контроллеров этих устройств. Интерфейсные процедуры определяют возможность доступа к ВУ и обусловленность запроса. Драйверы выполняют процессы декомпозиции относительно сложных запросов программ к ВУ на примитив возможностей этих устройств и их контроллеров.

Программное обеспечение каналов включено во все варианты ОДОС вне зависимости от того, присутствуют те или иные ВУ в конфигурации станции. Настройка каналов выполняется в фазе инициализации системы сразу после загрузки. Описатели используемых устройств должны присутствовать в ФОС.

В процедурах синхронизации работы ВУ и программного обеспечения каналов используется 15-й регистр, некорректные операции с которым из программ могут привести к сбоям обслуживания этих устройств.

При реализации ряда функций ОДОС сама использует доступ к ВУ через механизм каналов.

В ОДОС заложены следующие каналы:
(x помечены не используемые каналы)

- 1 — канал ВТЛ;
- x 2 — АЦПУ;
- 3 — канал обмена с спецВУ ЦМ;
- x 4 — канал обмена с спецВУ ЦМ;

- х 5 — МТ;
- 6 — канал обмена с ФС ЦМ;
- 7 — канал для управляемого из программ обмена с ЦМ;
- 8 — канал под загрузку (FI) и сброс (DU) программ из/в ФС ЦМ;
- 9 — межпрограммный обмен с ЦМ;
- 10 — спецзапросы (GIVE): дата, время, запросы о доступах к директориям и т. п.;
- 11 — канал обмена ОС с ВТЛ (прием директив);
- 12 — канал обмена ОС с ВТЛ (вывод сообщений);
- х 13 — резерв;
- 14 — канал для спецкоманд ЦМ (спецпрерывания, рестарты,...);
- 15 — канал для спецответа из ЦМ («конец обмена»);
- х 16 — резерв под локальный энергонезависимый накопитель;
- х 17 — межпроцессорный обмен помимо ЦМ;
- 18 — ЭД;
- 19 — связь с аппаратурой в стандарте «ВИШНЯ»;
- 20 — вспомогательный для 19-го;
- 21 — ЛТЛ1;
- 22 — ЛТЛ2;
- 23 — ЛТЛ3;
- 24 — ЛТЛ4

Приведенный список каналов соответствует первичному варианту ОС, ориентированному на тесное взаимодействие с периферией ОДРБ. В сетях с ЦМ на базе Одренка назначение некоторых каналов трансформировалось.

Нумерация каналов не всегда явно фигурирует в запросах на обмен с ВУ, но используется в механизмах синхронизации работы программ с ними и присутствует в сообщениях о состояниях программ типа

...SBNN, где NN — номер канала.

Обращение к ВУ из программы инициируется экстракодом 157 (PERI), которым указывается тип ВУ и операция. Как правило, такое взаимодействие программы с ОС опосредуется специальными пакетами подпрограмм, реализующими для программы более высокоуровневые, чем предоставляемые непосредственно ОС, функции обмена с ВУ (ядро ОС обычно реализует некоторые базовые наборы операций с ВУ).

Для синхронизации программы с работой ВУ предназначен экстракод 150 (SUSBY), вызывающий приостановку программы до

завершения работы соответствующего канала. В пакетах-посредниках такая синхронизация учтена.

Спецификации по типам ВУ и параметрам взаимодействия с ними в рамках данной системы приведены в описании экстракода 157.

Непосредственная работа программ с КАМАК-аппаратурой поддерживается в ОДОС двумя услугами. Первая — это услуга определения позиции и свойств нужного КАМАК-модуля с помощью экстракода 156 (ALLOT). Вторая — синхронизация работы программы с функционированием устройства с помощью механизма ожидания LAM-сигналов от модулей крейта через ОС (экстракод 141). Для возможности использования первой услуги имя модуля и его свойства должны быть внесены в описание конфигурации станции. К первой услуге можно обратиться с помощью подпрограммы-функции MPPR, ко второй — LWAIT.

ОДОС не предоставляет специальных средств для долговременного закрепления каких-либо ВУ за программами. При необходимости требуемый эффект может быть достигнут соответствующим распределением приоритетов.

Детальное описание способов взаимодействия программ с ВУ приводится в разделе «Экстракоды», в следующих же разделах даются некоторые общие пояснения по поводу работы с основными из них: терминалами, устройствами ЦМ и ЭД.

2.2.1. Терминалы

Терминальные каналы предназначены в основном для обеспечения взаимодействия пользователей с ОС и организации оперативного диалога с программами.

Как упоминалось выше, ОДОС разрешает подключение 5 ТЛ: до четырех ЛТЛ через последовательный интерфейс типа B1102, V-24 (ИЯФ) и одного ВТЛ, подключаемого к ЦМ.

Способ работы с ТЛ — посимвольный обмен в полном дуплексе (прием/вывод символа по прерываниям от интерфейса, вывод эха — через ЭВМ). Необходимые конверсии между ASCII-представлением символов и внутренним 6-битовым реализуются терминальным драйвером ОДОС.

Механизмы взаимодействия программ с ТЛ описаны в разделе «Экстракоды», экстракод 157. Для ОС терминал является источником запросов, вводимых пользователями в форме директив, и устройством для вывода диагностической информации. Обращение

к ОС с терминала осуществляется вводом с клавиатуры одного из символов прерывания:

CTRL(B) или CTRL(W),
(ASCII: #002 #027)

по приеме которых ОС прикрепляет ТЛ к себе, сбрасывая при необходимости с него заявку приема символа (вывод не сбрасывается: ОС запоминает сигнал прерывания и дожидается завершения операции) и выводит приглашение к набору директивы в форме:

*ЧЧ/МН/СС*ИМЯСИСТЕМЫ:—

(где ЧЧ—час, МН—минуты, СС—секунды текущего времени, а ИМЯСИСТЕМЫ—это имя, определенное для данной станции в ФОС). Затем ОДОС переводит терминал в режим приема директивы и осуществляет ее пословный ввод. Терминальный формат директив описан в разделе «Директивы системы». Если ОС не распознает директиву или в поле ввода оказывается символ из управляющей группы, прием сбрасывается с сообщением CANCEL, иначе начинается исполнение директивы. Ход этого исполнения, а также результат индицируются на этом же ТЛ.

Программы, пытающиеся обратиться к ТЛ во время его работы с ОС, приостанавливаются до освобождения терминала системой.

ОС имеет наивысший приоритет в использовании терминала. Следующим по приоритету является вывод из программ по экстракодам 160 и 161, и затем—работа из программ через экстракод 157.

Номер ТЛ, как уже указывалось, служит также и для защиты программ, приписанных к нему, от директивного воздействия с других терминалов (запрет не распространяется на справочные директивы. Директивное воздействие на программы из других программ разрешено безусловно). С каждым ТЛ система связывает еще и такой специальный контекст, как номер диска (1—диски ЦМ, 2—ЭД) и номер директории ФС ЦМ. Эти атрибуты определяют, с каким реально носителем ФС будут работать программы, приписанные к данному ТЛ, при обращении к ФС (при инициации загрузки программы директивой с ТЛ программе приписываются атрибуты системы, связанные с данным ТЛ; если же программа загружается по директиве из другой программы, то она наследует атрибуты программы—источника директивы).

Для программ высокоуровневые функции организации терминального диалога поддерживаются пакетами подпрограмм типа

VIDEOT, VIDA, VI...

Для более полного использования функциональных возможностей терминалов в ОДОС реализован дополнительный запрос к ним: режим «чтения с подканала», позволяющий программам переключить на себя ту часть потока информации с клавиатуры, которая не затребована обычными заявками чтения. Такое подключение в отличие от обычного не занимает канала обмена с ТЛ (оно реализовано как пристройка к каналу, отсюда термин «подканал») и позволяет принимать инструкции пользователя даже во время вывода информации на терминал. (Следует отметить, что из-за отсутствия в интерфейсе V-24 возможности передачи специальных сигналов прерывания, ОС считывает сигналы с клавиатуры при нажатии на любой символ.)

Необходимая для программ часть интерфейса чтения с подканалов терминалов содержится в функции %GETCH.

В последних версиях ОДОС разрешены также и заявки программ на прямое (без перекодировки из ASCII во внутренний код) взаимодействие с терминальными каналами, т.е. на байтовый обмен, чем обеспечивается подключение через V-24 не терминальных устройств. Соответствующий терминальный канал при этом делается «прозрачным», т.е. ОС не «отлавливает» при таких обменах символов прерывания (режим прозрачности реализован с предосторожностями: дополнительно требуется уставка 3-го ключа операционной системе. Эта дополнительная мера предупреждает от ошибочного «отключения» ТЛ от ОС, возникающего при полностью прозрачном байтовом обмене). К функции можно обратиться посредством пакета INTB (со входом OUTB).

Замечания:

1. По причине ограниченного использования ВТЛ для взаимодействия с ПМ в рамках основного способа использования Одрят с системой ОДОС для ВТЛ реализован не весь набор функций ЛТЛ: только обычный прием и вывод.

2. На один из каналов ЛТЛ можно подключать печатающее устройство, имеющее соответствующий интерфейс V.24. Во избежание недоразумений канал этот следует специальным образом определить в ФОС или настроить директивой DZ.

3. Начиная с версии ОДОС-08.6, для всех видов терминалов ввод с клавиатуры символа CTRL(S) интерпретируется ОС как сигнал приостановить вывод на ТЛ, а символа CTRL(Q)—возобновить его.

2.2.2. Каналы связи с ЦМ

ЦМ в используемых в ИЯФ сетях Одрят выполняет для всех подключенных к ней периферийных систем функции начальной загрузки, файловой машины, межмашинного обмена (по схеме «почты») и т. п.

Сведения об организации ЦМ содержатся в [4].

Информация об аппаратной среде сети и реализованных в ней протоколах взаимодействия ПМ ↔ ЦМ приводится в Приложении 1.

Здесь рассматриваются некоторые общие положения о назначении и функционировании в ОДОС логических каналов, сформированных для взаимодействия с ЦМ.

Применяемая система связи позволяет два основных способа обмена: программный и ПД. В первом случае задействуется драйвер ИМ периферийной системы (компонента программы ОДОС), активируемый ЛАМ-прерываниями ИМ. Процедура ПД с точки зрения ПО ПМ реализуется аппаратно.

Инициаторами большинства обменов является ПМ, которая совместно с программой МОДО в ЦМ образует 15 логических каналов связи. Поскольку такое распараллеливание одного физического канала носит чисто программный характер, по необходимости возникает достаточно емкий протокол обмена ПМ ↔ ЦМ.

Назначение каналов связи с ЦМ вкратце охарактеризовано в приведенном в 2.2 их списке. Первые 6 каналов предназначены для взаимодействия с периферией общего назначения, подключенной к ЦМ. Остальные служат для упрощения процедур обмена программами, управления ВТЛ, организации непредусмотренных в ОДОС сеансах взаимодействия программ в Одренке с ЦМ. Несколько каналов выделены под специальные функции протокола.

Общие аспекты функционирования каналов связи ПМ и ЦМ:

1. ОДОС допускает одновременную активность нескольких каналов связи с ЦМ, позволяя последней в достаточно произвольном порядке выполнять принятые запросы. Так, переслав запрос на обмен с диском, ОДОС сразу же может выслать запрос на обмен с ВТЛ или любой другой и т. д. и принимать ответы о результатах их исполнения по мере их удовлетворения центральной машиной. Ограничения на параллельную работу каналов связи с ЦМ приведены в Приложении 1. Здесь в связи с этим указывается, что нарушение протокола функционирования каналов может привести к завершению текущего обмена с ошибкой.

2. Обмены с ЦМ содержат не управляемые средствами ПМ фазы прямого доступа к памяти ПМ. Во время этих фаз ОС ПМ ограничивает реорганизацию ОП. При сбоях связи в таких фазах реорганизация памяти может стать невозможной до рестарта ЦМ.

3. Интерфейс программ с ОС по взаимодействию с периферией ЦМ ничем не отличается от интерфейса с аналогичной локальной периферией, за исключением возможных дополнительных кодов ошибок в ответ на запросы доступа к ним. Эти дополнительные ответы детализируют причину сбоя, например, это может быть сбой системы связи с ЦМ, или ее неоперативность, распознаваемая по длительному (более 40 с) времени ожидания периферийной машиной считывания ее запроса центральной машиной.

Сбои во взаимодействии с ЦМ по запросам, не связанным с обращением к периферии ЦМ, трансформируются в такие ответы источнику запроса, которые означают невыполнение функции.

2.2.3. Файловая система

ОДОС поддерживает взаимодействие с двумя носителями ФС: ЭД и дисками ЦМ. Условно диски ЦМ считаются диском номер 1, а ЭД — диском 2. Организация ФС одинакова как на ЭД, так и в доступной из ПМ области диска ЦМ (ДИРЕКТОРИИ). На функционировании программ, написанных с использованием стандартных средств доступа к ФС, работа с диском 1 проявляется лишь в замедлении обменов. ОДОС разрешает динамическое переключение диска директивой DR. При этом файлы, открытые на старом носителе, принудительно закрываются (см. ниже). Такие же защитные действия предпринимаются и при переключении с директории на директорию директивой DY.

Из соображений экономичности и оперативности доступа, а также по историческим причинам используется предельно простая структура ФС:

каталог (256 слов);
тела файлов неразрывными областями.

Каталог организован следующим образом:

сл0 адрес начала свободной памяти;
сл1 размер свободной памяти;
сл2 К: кол-во заложенных файлов;
сл3 имя-1;
сл4 БТ.АдрНачала-1;
имя-2;

- . БТ.АдрНачала-2;
-
- . имя-К;
- . БТ.АдрНачала-К;
- . xxxx;
- . адрес начала свободной памяти.

Здесь каждая запись—это 24-битовое слово, БТ (2 левых бита)—биты типа файла (или биты защиты), АдрНачала—22 бита.

Таким образом, ФС позволяет содержать до 125 файлов в области до 4М слов.

Назначение битов типа:

- БТ = 00 — простой файл;
- 01 — сложный файл;
- 10 — файл-программа;
- 11 — специальный файл.

Наделение файлов признаками типа имеет двойное назначение: упрощение идентификации внутреннего содержимого и защита.

На доступ к простым файлам нет никаких ограничений и нет внешних признаков, оговаривающих их содержимое. Сложные файлы имеют структуру, аналогичную структуре всей ФС, и защищаются на уровне подпрограмм, поддерживающих взаимодействие с ними. Файлы-программы содержат готовые к исполнению программы и защищаются в фазе открытия средствами ОС. На специальные файлы распространяются оба способа защиты.

Про файлы-программы следует также отметить, что они перед телом программы содержат 8-словный блок, называемый блоком запросов программы. В этом блоке определены такие атрибуты программы, как требуемый ей размер ОП, приоритет и некоторые дополнительные статусы (см. Приложение 4).

Блок запросов формируется ОС при помещении ею программы в ФС по директиве DU. Если программа помещается туда другими средствами, то формирование этого блока осуществляется ими.

ОДОС представляет программам минимальный набор возможностей доступа к ФС:

- создание файла
- открытие
- обмен

Все же прочие процедуры (закрытие файла, поддержка структуры сложных файлов, реорганизация ФС при удалении файла) выполняются на уровне пакетов подпрограмм (типа DRUM) или утилит (типа ZJAA). ОДОС не ведет детального учета открытых программой файлов, фиксируя в своих таблицах лишь факт их открытия. Описания файлов и их увязка с программными номерами файлов содержатся в пакетах подпрограмм. С этим, в частности, связаны такие «жесткие» меры, как закрытие операционной системой всех, открытых программой ранее, файлов при переключении диска или смене директории для этой программы.

Замечания по работе с ЭД:

1. После включения крейта его требуется дополнительно инициализировать. Это делается с помощью директивы CLZU программе SYST (вызывающей специальную перезагрузку ОДОС в Одренка), или программы CLZU (без перезагрузки). В обоих случаях выполняются процедуры обнуления всей памяти ЭД (требуемые для некоторых их вариантов) и формирования пустого каталога. После инициации можно загрузить в ЭД желаемый набор программ: из директорий ЦМ программой LOZO или других источников другими программами;

2. Обмен с ЭД выполняется неразрывным способом с использованием блочного КАМАК-цикла (ЭД работает в темпе КАМАК-магистрала).

2.3. Средства синхронизации

Помимо описанных выше механизмов синхронизации работы программ с функционированием ВУ в ОДОС заложено два дополнительных способа организации приостановок программ в ожидании событий «внешнего мира». Это—**таймирование** и ожидание **спецпрерываний** (СПР).

Первый способ позволяет программам приостанавливаться на определенные интервалы времени, второй—синхронизоваться с фазами работы контролируемых объектов.

Счет времени базируется на прерываниях от таймера (часов) Одренка, генерируемых им примерно каждые 100 мс. Программы с

помощью директивы TI могут «таймироваться», т. е. задавать ОС в единицах СЕК интервалы, через которые с них следует периодически снимать состояние останова (SU). Комбинированием директив TI и SU можно реализовать и разовую приостановку программы на определенное время (подпрограмма TISU). Директива TS позволяет задавать такую задержку в единицах прерываний таймера. Максимальные интервалы времени в этих процедурах 262 144 и 83 886 СЕК, соответственно.

К прерываниям таймера Одренка можно привязаться также и через механизм ожидания LAM-прерываний через систему (сигналы таймера формируются Одренком как LAM от фиктивной, нулевой, позиции крейта).

Для синхронизации работы программ с функционированием управляемых объектов в ОДОС заложена обработка сигналов от специального внешнего регистра прерываний — РП (модули P0602-РП, P0606-РП-2М, ИЯФ), в которые трансформируются тактовые импульсы этих объектов. Программы могут синхронизоваться с сигналами РП с помощью экстракода 141, которым указывают, в ожидании каких СПР они хотели бы приостановиться. ОС ведет учет СПР двояким образом: для всех программ, находящихся в ОП, логическим поразрядным добавлением 1 к 16-му слову каждой программы (нумерация СПР в слове: 24, 23, ..., 1) и, если программа запросила об этом соответствующей заявкой (одна из форм экстракода 141 — учет реперных СПР) — переносом содержимого РП, умноженного логически на заявленную программой маску, в 18-е слово.

Для правильной работы механизма первого типа программы должны сами уничтожать использованные пометки о СПР в 16-м слове.

Механизм второго типа позволяет отслеживать определенные последовательности СПР, связанных, например, с фазами работы управляемых объектов. Каждое прерывание из этой последовательности вырабатывается в начале новой фазы и носит таким образом характер реперного сигнала. Это позволяет программам, включившим данный механизм, соотносить свои действия с состоянием объекта.

Обращение к функциям синхронизации с СПР и необходимые процедуры содержатся в подпрограммах PINT, PINTA (первый способ) и %SRPM, %LRPM (второй).

ОДОС поддерживает также и механизм программной имитации СПР, который может использоваться для межпрограммной синхро-

низации. Обращение к этой функции реализуется подпрограммой IINT.

2.4. Межпрограммный обмен

Одной из часто возникающих потребностей в работе больших программных комплексов (АСУ установок или экспериментов, ...) является оперативный обмен информацией между программами. При отсутствии аппаратной поддержки доступа разных программ к некоторым общим областям ОП, такую функцию может реализовать лишь привилегированная программа типа ОС, которой доступно все адресное пространство ЭВМ.

В межпрограммном обмене, предлагаемом ОДОС, один абонент пассивен — он лишь указывает ОС область своей памяти, выделяемую для обменов. Другие программы могут выступать в качестве активных абонентов и инициировать передачу информации от себя или к себе в/из этой области. ОДОС может запомнить по одной такой области для каждой из программ, но пассивная в одном из обменов сторона может быть активной в другом. Никаких ограничений на выделяемые под обмен области ОП нет: ОС контролирует лишь принадлежность этих областей программам-абонентам. Ограничение на объем передаваемой информации: 1 — 512 слов за один обмен.

Механизм межпрограммного обмена настраивается и инициируется экстракодом 142. Соответствующий интерфейс с ОС представляется подпрограммой BERI и пакетом DAJ/NA. В ранних версиях ОДОС при отсутствии пассивного абонента в данной машине запрос передавался в ЦМ. В последних версиях этот автоматизм выключен, но к самой функции межмашинного обмена через ЦМ обратиться можно, используя одну из форм 142-го экстракода.

2.5. Спецрежимы

В ОДОС реализованы два специальных режима функционирования программ: это режим полного запрета работы всем программам, кроме одной, заявившей его, — монополизация процессора, и режим контролирования одной программой экстракодов всех других или выделенной программы.

Режим монополизации процессора инициируется экстракодом 143 и его действие распространяется на все программы, уже находящиеся в ОП или загружаемые позднее, до снятия режима пов-

торным употреблением этого же экстракода. Предназначен для отработки критических ситуаций во взаимодействии с аппаратурой управления. В состоянии программ, распечатываемом по директиве PR, соответствующая приостановка сигнализируется как SS. Необходимый интерфейс с ОС реализуется подпрограммой %STOP.

Второй режим заявляется экстракодом 155. При его исполнении система перенастраивает механизмы обработки экстракодов с тем, чтобы информировать эту, контролирующую, программу об их возникновении и передать ей управление. Последняя может сама выполнить экстракод и необходимые процедуры взаимодействия с подконтрольной программой, или перепоручить выполнение экстракода ОС с помощью другой формы экстракода 155.

Режим контролирования предназначен для исследования других программ (при отсутствии исходных текстов), коррекции их работы или отладки. Разрешен только для одной, загруженной «впритык» к ОС, программе.

3. ВНУТРЕННЯЯ ОРГАНИЗАЦИЯ И ФУНКЦИОНИРОВАНИЕ ОДОС

ОДОС — это программа объемом примерно 8К слов, состоящая из трех достаточно независимых модулей, написанных на ассемблере. По условиям создания и в той степени, в какой это оказалось возможным, основные спецификации ее совпадают со спецификациями разработанных ранее в ИЯФ систем для ЭВМ ОДРА-1300 [5]. Помимо более развитой системы услуг ОС существенные различия касаются в основном внутренней организации и принципов реализации ряда функций. Так, в отличие от систем для ОДР, система для Одрят изначально проектировалась как распределенная ОС (ОДОС + МОДО), что значительно облегчило задачу создания сетей станций на базе этой микроЭВМ. Предоставление одинакового как для локальных ВУ, так и для устройств ЦМ, интерфейса доступа к ним унифицировало процедуры взаимодействия с ВУ из программ и позволило в многих случаях обходиться ограниченной конфигурацией аппаратуры станций. Другое существенное различие касается схем организации мультипрограммного исполнения потока заданий ОС. В ОДОС для реализации заданий, требующих выполнения определенных условий или содержащих паузы в использовании процессора на синхронизацию с работой ВУ, вкомпонованы относительно автономные подсистемы — системные процессы. Единообразная схема управления системными процессами позволяет эффективно распределять процес-

сорное время между разными внутренними задачами системы со сравнительно небольшими накладными расходами. Как и пользовательские программы, процессы подлежат координации планировщиком задач. Однако, являясь встроенными в тело системы, они имеют свободный доступ к общим структурам, что снимает проблему интерфейса между ними и другими компонентами ОС.

По ряду соображений (повышенные накладные расходы, отсутствие адекватных аппаратных возможностей, ...) в большинстве процессов в ОДОС не реализовывались механизмы реентерабельности или размножения процесса для каждого нового обращения к ним. За исключением терминальных процессов (представляющих по существу один реентерабельный процесс), в ОДОС одновременно может выполняться только один запрос к процессу. Другие — или придерживаются до освобождения ресурса, или получают отказ.

Помимо терминальных в ОДОС встроены следующие процессы: FI/DU — процесс обмена программами с ЦМ; GV — спецзапросы к ЦМ; DN — межпрограммный обмен с ЦМ.

Для обработки таких ситуаций, как отложенная реорганизация ОП, или контроль оперативности работы ВУ (TIME-OUT), могут включаться еще и DE- и T-процессы.

Ту часть функций, при выполнении которых не возникает разрывов на взаимодействие с ВУ или по другим причинам, ОДОС реализует без привлечения процессов, используя «жестко» настроенные схемы их исполнения и координации. Так запрограммировано большинство директив и экстракодов.

Функции непосредственного взаимодействия с ВУ и поддержки необходимого интерфейса с источниками запросов к ним выполняют драйверы ВУ. Драйверы представляют собой достаточно автономные подсистемы, но, как и процессы, в ОДОС они имеют полный доступ к общим данным системы.

(Следует отметить, что такое решение проблемы интерфейсов между компонентами ОС, имеющее целью предельное повышение ее эффективности, несколько сказывается на ее мобильности, затрудняя, например, замену драйверов).

С точки зрения внутренней организации в ОДОС можно выделить следующие подсистемы:

- диспетчер прерываний;
- планировщик задач;
- подсистема управления процессами;

процессор экстракодов;
 директивный процессор;
 внутренние процессы;
 драйверы ВУ.

Приведенный список отражает и порядок выполнения операционной системой постоянно функционирующих и вводимых извне заданий, а сами подсистемы достаточно четко прослеживаются в архитектуре программы (рис. 2).

Задания в ОС поступают двумя путями:

- экстракодовым прерыванием через 28-й адрес ОП и
- LAM-прерыванием со стартом ОС с 29-й ячейки.

По этим событиям в ОС активируется диспетчер прерываний, который передает управление нужной процедуре обработки прерывания или выполнения экстракода. Процедура выполняет задачу сама или активирует (через подсистему управления процессами) нужный процесс. Как правило, процесс получает процессорное время лишь после решения планировщика, хотя в некоторых случаях, начальная фаза процесса рассматривается как продолжение процедуры обработки прерывания.

При обработке LAM-прерываний система исходит из возможности одновременного возникновения LAM от нескольких модулей крейта. Поэтому для ускорения их обслуживания после обработки очередного LAM управление возвращается LAM-координатору, являющемуся частью диспетчера прерываний. После выполнения же экстракода, в зависимости от сложившейся ситуации, управление передается в координатор процессов или планировщик программ, представляющие в совокупности планировщик задач системы. Начальная активация драйверов ВУ в большинстве случаев осуществляется при выполнении экстракодов через специальный (экстракодовый) вход в каждый драйвер (процессы также обращаются к драйверам через экстракод 157). Терминальный драйвер и драйвер ИМ связи с ЦМ могут быть активированы также и прерываниями от соответствующих контроллеров.

Что касается временных аспектов функционирования ОДОС, то во всех случаях процесс или драйвер ВУ полностью отработывают тот интервал времени, который им требуется в сложившейся ситуации. Размеры этих интервалов определяются спецификой работы конкретных ВУ или объемом операций, которые требуются процессу для максимального продвижения выполняемой задачи.

При оценке «накладных» расходов под работу ОС нужно так-

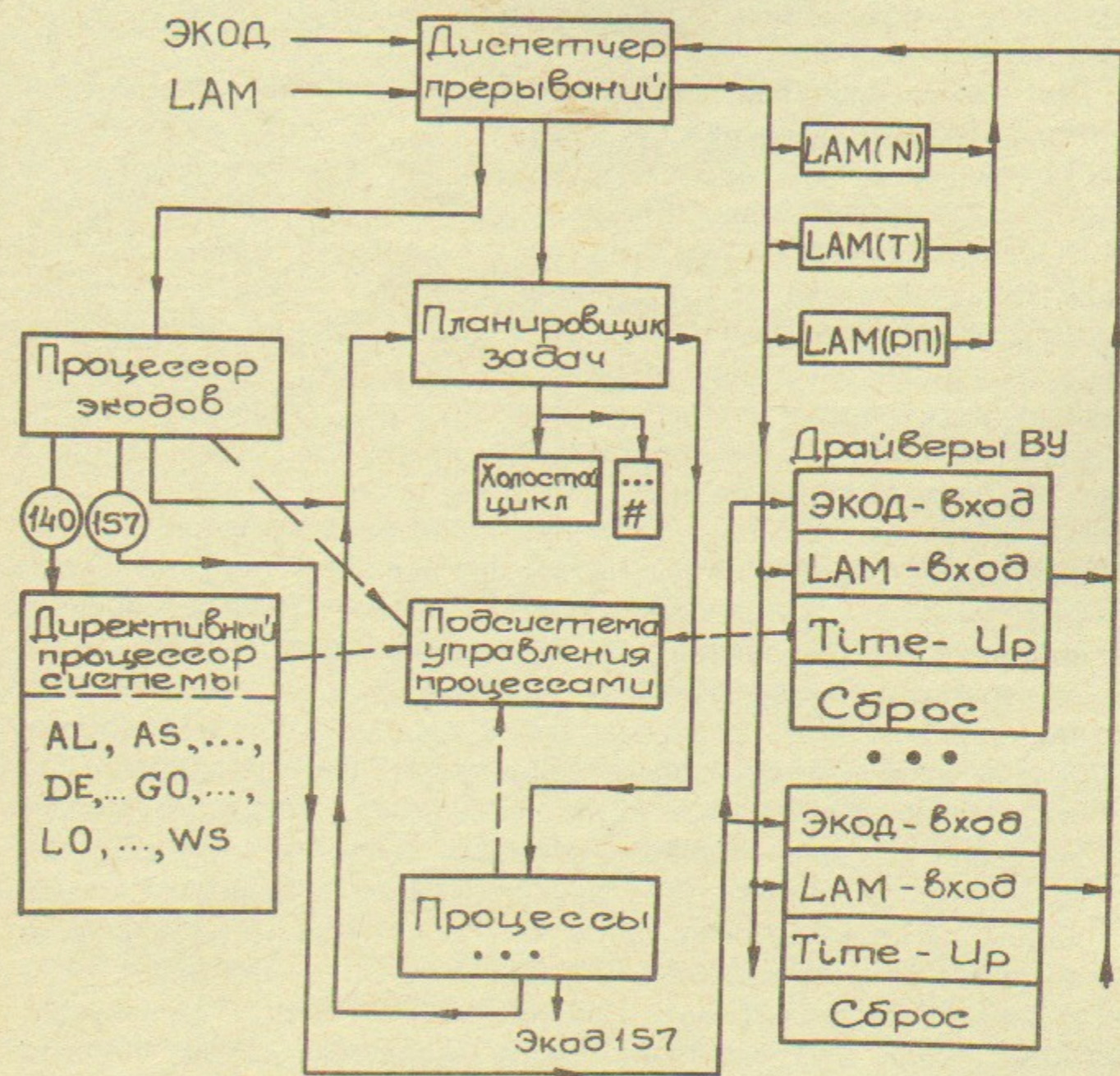


Рис. 2. Блок-схема ОДОС.

же помнить, что помимо санкционированных отвлечений процессора на выполнение запросов программ и пользователей, она поддерживает активность процесса обработки прерываний по времени, связанного с функциями таймирования программ, работы ВУ и ведением календаря.

При проектировании ОДОС большое значение придавалось оптимизации использования ею ресурсов ВС. К этому относятся, в первую очередь, сами принципы разделения функций реализации разных услуг между ядром ОС, пакетами подпрограмм и утилитами, продиктованные соображениями, связанными с обеспечением такой характеристики, как функционирование в реальном времени. В ОДОС включены в основном лишь действительно целесообразные для реализации в ядре услуги ОС. Следующим по возможности уровнем оптимизации являются архитектурные решения. Помимо упомянутых уже приемов интеграции процессов и драйверов в ядро ОС, можно привести еще некоторые из них:

- все процессы ОДОС для планировщика сведены в упорядоченный список с тем, чтобы процессы, которые могут получать задания от других процессов, имели приоритет в использовании аппаратуры. Немедленно исключаются из списков планировщика отработавшие процессы;
- применяется многоуровневая схема координации, при которой всякий вход в планировщик делается в самый последний уровень, требуемый по ситуации (например, после исполнения экстракодов, не породивших запросов к процессам, вход делается прямо в планировщик программ, а если при этом не сменилось состояние других программ, то сразу восстанавливается программа — источник экстракода);
- используются ускоренные цепочки прохождения системных запросов через экстракодовые входы драйверов, а при работе с ЭД при реализации своих функций выполняется обращение прямо к процедурам обслуживания ЭД помимо экстракодового входа в их драйвер ...

Немаловажную роль сыграли также и возможности экономичного программирования на языке ассемблера.

Некоторые цифры по времени обработки ОС различных ситуаций (из расчетов 3 мкс на команду) в мкс:

время обработки прерывания от таймера при работе одной программы с последующим ее перезапуском, минимально	220
время обработки запроса на ожидание LAM, минимально	130

время на активацию программы после «ее» LAM, минимально	150
анализ планировщиком состояния программы	20
переключение на другую программу	90
переключение на LAM-драйвер	
после «его» LAM, минимально	320
предобработка запроса на прием символа с ТЛ	220
передача символа с ТЛ в программу с ее активацией	570
отвлечение на вывод одного символа на ТЛ	400
выборка параметра директивы из ее текста при текстовом способе представления директивы	200
скорость межпрограммного обмена	180 + (15 на слово)
скорость обмена с открытым на ЭД файлом	270 + (3 на слово)
скорость загрузки программ из ФС ЭД	750 + (3 × размер)
время на уничтожение одной программы, минимально	600

Эти цифры вполне могут считаться средними при работе одной программы, так как наложение нескольких событий при этом довольно маловероятно.

4. ЗАГРУЗКА И НАСТРОЙКА ОДОС

Возможны два способа загрузки ОДОС в Одренка. В первом из них применяются аппаратно-программно реализованные модули загрузчиков, предназначенные для загрузки ОДОС с энергонезависимых носителей, например, дисков. Этот способ используется для загрузки ЦМ [4]. Другой, основной способ — это загрузка периферийной системы (станции) с помощью центральной по специальному протоколу, реализуемому программным обеспечением ЦМ, интерфейсным модулем ПМ и микропрограммой прямого доступа Одренка.

Во всех случаях применения ОДОС используется одна и та же, приготовленная в форме готовой к исполнению программы, ее копия. Настройка на конфигурацию конкретной станции (свойства ЭВМ, используемого крейта, набора и расположения различных модулей в нем и прочие особенности) выполняется при начальной инициации системы. Процедуры настройки базируются на описателе конфигурации станции (60 слов), которое должно быть вписано в ОДОС, начиная с 45-го адреса, перед ее запуском.

В загрузочной директории ЦМ программа ОДОС хранится с описателем конфигурации ЦМ. Описания периферийных систем

содержатся в ФОС CRDW, редакцию которого можно осуществлять программой LOCW.

Структура описания станции, назначение его позиций, список резервированных имен системных устройств приведены в Приложении 2.

В ОДОС определены два стартовых адреса: 20-й — с инициацией ЭД и 21-й — без таковой. Прочие стартовые действия одинаковы и заключаются в формировании необходимых NAF'ов, таблиц переходов на обработку LAM-прерываний от описанных системных модулей крейта, установке параметров ОП, ЭД, ТЛ, инициации всех системных устройств. Здесь же устанавливаются логический номер станции и символика приглашения к вводу директив с ТЛ (последнее позволяет также и идентифицировать систему при работе с терминала ЦМ).

Дальнейшие действия ОДОС зависят от того, при каких внешних условиях она используется. Если логический номер станции отличен от нуля (при этом в описателе конфигурации должен присутствовать модуль OD00, описывающий ИМ У0605), то система предпринимает попытку дополнительной настройки. Это — запросы из ЦМ текущих даты и времени, а также номера и имени директории ФС ЦМ, к которым данная станция приписана по умолчанию.

Для системы с логическим номером 0 или при отсутствии в описании модуля — интерфейса связи с ЦМ дополнительная настройка не производится и, соответственно, стартовые значения даты и времени для ведения календаря имеют нулевые значения. Их можно задать директивой коррекции времени (СТ).

Далее ОДОС, в соответствии с предписанием описателя конфигурации станции может попытаться загрузить (из ФС ЭД, а при отсутствии ЭД, из ФС ЦМ) до 5 указанных там программ.

Параллельно с этими (начальными) заданиями в ОС стартуют и терминальные процессы, которые выводят на все терминалы (кроме терминала ЦМ и описанных под типом 0) приглашения к набору директивы. Поскольку запросы времени могут быть еще не выполнены, время в первом приглашении может быть неверным.

После этого система полностью готова к взаимодействию с пользователями, и можно вводить в нее директивы.

5. ДИРЕКТИВЫ СИСТЕМЫ

Язык взаимодействия пользователя с ОДОС основывается на наборе директив со встроенным исполнением (т. е. полностью

выполняемым самой ОДОС). Управлением исполнением директив занимается одна из компонент ОДОС — Директивный Процессор Системы (ДПС). Задания ДПС передаются с помощью экстракода 140 (см. ниже) в декодированной или текстовой форме (см. ниже).

Как уже упоминалось, ОДОС реализует, в основном, лишь некоторый базовый, хотя и достаточно специфичный, спектр функций ОС. Это касается и содержания ее директив, предоставляющих пользователю лишь действительно необходимые возможности для управления ресурсами ВС и взаимодействия со своими программами. Выполнение более емких процедур возлагается на подпрограммы или утилиты, реализованные как обычные пользовательские программы.

В ОДОС встроено исполнение около 30 базовых директив, но распознает она порядка 50. Это отличие в 20 директив образуется за счет ряда составных директив и команд загрузки общеиспользуемых программ упрощенным вариантом их вызова — по двухбуквенной мнемонике. В этих случаях ОДОС сама формирует необходимые последовательности простых директив.

Директивы могут вводиться с ТЛ или выдаваться программами. Набор директив, разрешенных с ТЛ, шире набора, предназначенного для использования программами за счет ряда директив справочного характера. Попытка заказа исполнения «терминальной» директивы приведет к остановке программы с сообщением на терминал ее приписки о незаконности запроса. Что же касается диагностики, выводимой на ТЛ по ходу исполнения и в завершение директив, то она носит достаточно информативный характер и не требует детального пояснения. В случаях же, когда полноценное пояснение причины отказа повлекло бы за собой неоправданно большие накладные расходы, выводится ответ:

ОШИБКА В ДИРЕКТИВЕ

Следует помнить, что сообщение

НЕТ #NAME

в ответ на попытку воздействия на программу может проистекать не только по причине ее отсутствия в ОП, но и по причине приписанности ее к другому терминалу. При попытке выполнения директив с загрузкой программ загрузка считается выполненной, если программа с указанным именем уже находится в памяти и приписана к данному терминалу (для директив из программ последнее не проверяется).

Формат директив при вводе с ТЛ:

CD P1, P2 P3

где CD — имя (мнемоника) директивы, P_i — ее параметры.

Здесь CD и P_i — не более, чем 12-символьные слова (поля), отделяемые друг от друга разделителями — пробелами или запятыми. Ввод разделителя завершает прием слова. Шрифт (РУС, ЛАТ) не имеет значения (знаки #74 и #75 — переключения алфавитов — игнорируются). Числовые параметры, вводимые с префиксами * или #, считаются восьмеричными. Имя директивы распознается по первым 2 символам, длина других параметров — по смыслу, но не более 8. Некоторые имена директив шифруют составные директивы, т. е. такие, которые могут быть набраны за несколько приемов из простых. В этом случае система сама осуществляет нужную декомпозицию. На нераспознанную директиву или запрещенный символ система реагирует выводом сообщения CANCEL и прекращением приема.

Пустое поле трактуется как умолчание параметра и будет выполнена подстановка, соответствующая типу параметра и его возможному значению по умолчанию. Для числовых параметров при этом подставляется 0, для текстовых — пробелы, а если представляется возможным определить более содержательное значение, то подставляется оно. Там, где в качестве разделителя фигурирует запятая, умолчание первого параметра приводит к подстановке его значения из предыдущей директивы, введенной с данного ТЛ.

ОС разрешает редактирование слова (продвижение курсора, уничтожение, замена, вставка символа, выход на край слова из середины — по ETX) до ввода разделителя. Такой, упрощенный, подход к редактированию директивы продиктован соображениями экономичности и целесообразностью фильтрации ошибок оператора на как можно более ранней стадии. Поэтому ОДОС ведет прием директивы пословно, немедленно осуществляя интерпретацию каждого слова. Это позволяет после дешифрации имени директивы осуществлять контроль не только значений, но и количества полей-параметров.

Отказаться от ввода директивы можно нажатием на любой (кроме символов редактирования) символ из управляющей группы. Если же при этом требуется ввести новую директиву, можно сразу нажать на символ прерывания. Предыдущий ввод при этом полностью отменяется и выводится новое приглашение к вводу директивы.

Пример:

*13/01/54*POSITRON: — LG PROC 20

Введена директива загрузки программы из файла PROC и запуска ее с 20-го адреса.

При загрузке программ директивами с ТЛ им присваиваются номер диска и директории ЦМ из контекста системы для данного ТЛ.

Директивы из программ

Запрос на исполнение директив из программ осуществляется экстракодом 140, адресная часть которого должна указывать на не более, чем 12-словное поле описания директив в программе. Разрешаются декодированный и текстовый форматы описания директив (ДПС распознает это по знаку первого слова поля: числовые коды меньше #40, и поэтому директива считается заданной в текстовой форме, если это слово меньше 0). Если директива распознается как декодированная, то 1-е слово поля считается словом кодов:

K4K3K2K1,

а остальные — параметрами. Здесь K_i — код директивы (число в диапазоне от 0 до 31). Все параметры директивы (занимающие одно или два слова) должны иметь правильное внутреннее представление числа или текста. ДПС исполняет эти коды справа налево, начиная с K₁ и выбирая для них слева-направо соответствующие параметры. При этом после исполнения директив

DE и SU

все параметры сдвигаются влево на одну позицию. Исполнение прекращается по первому K_i=0, или невыполнению предыдущей директивы.

Пример задания директив из программы в декодированном формате:

оператор

CALL %DIR('0936ZJAATRAN000\$')

передает ДПС последовательность слов со значениями:

#00110306, ZJAA, TRAN, 20

(символ \$ имеет внутренний код 20).

Эта цепочка директив заказывает уничтожение программы ZJAA, загрузку программы из файла TRAN и запуск ее с 20-го

адреса (после выполнения директивы DE (ее код равен 06), параметром которой является ZJAA, параметры сдвигаются, так что параметром директивы LO (03) становится TRAN, а параметрами директивы GO (#11) — соответственно TRAN и 20.

Для реализации этого же с терминала нужно ввести три директивы:

```
DE ZJAA
LO TRAN
GO,,20
```

или две:

```
DE ЗJAA
ЛГ ТРАН 20
```

или одну:

```
DL ZJAA TRAN 20
```

Текстовый формат описания директив в программах напоминает их вид на терминале. Общая его форма:

```
CD1 P11,P21;CD2 P12 P22 P32; ...; CDj P1j,P2j P3j.
```

где CDj — имена (мнемоники) директив, Pij — i-й параметр j-й директивы. Разделителями являются пробелы и запятые, многократное употребление которых приводит к пропуску соответствующих параметров. Точка с запятой отделяет директивы (недобранные параметры заменяются значениями по умолчанию), точка — завершает цепочку (к этому приводит также и исчерпание текста). Символы DEL (внутренний код #77) игнорируются.

Максимальная длина текста директив 48 знаков. Реальная длина должна передаваться в счетчиковой части слова-адреса поля описания директивы (так называемая литеральная форма задания текстовой цепочки).

Например, оператором:

```
CALL %DIRA('DE ZJAA;LG TRAN,;TI,PROC,5.')
```

иницируется следующая последовательность директив системы:

```
DE ZJAA
LG TRAN,
TI,PROC,5
```

(здесь, по умолчанию, параметр запуска программы TRAN равен 0, что трактуется как запуск с текущего адреса!).

Как и в случае декодированного задания директив, все директивное поле переносится перед исполнением в систему. Выполнение прекращается как по исчерпанию цепочки (встрече точки, исчерпа-

нию счетчика литеральной конструкции), так и по ошибке в любой из директив цепочки.

Разрешено 2 формы экстракода 140 :

```
140 0 N(M)
140 6 N(M),
```

различаемых только значением поля аккумулятора экстракода. Это соответственно «форма с остановкой» и «форма с ответом» (подпрограммы %DIR и %DIRA). Для формы с остановкой при неисполнении директивы программа-источник останавливается (SU), в форме с ответом — отдается ответ о результате исполнения в 6-ю ячейку программы. Этот ответ равен номеру системного сообщения, выводимому на ТЛ по завершении исполнения директивы, если бы директива была введена с ТЛ. Нулевой ответ отдается при успешном выполнении директивы (OK=0). Список сообщений системы с их нумерацией приведен в Приложении 3.

Программа также может быть приостановлена (WD) на время исполнения своей директивы, если это исполнение не является неразрывной процедурой (например, оно включает загрузку программы из ЦМ).

Принято, что такие атрибуты загружаемой программы, как номер терминала приписки, номер диска и директории наследуются по цепочке загрузок программ.

При необходимости воздействия на систему следует употреблять имя @@@@. Употребление 0000 в качестве имени программы — объекта директивы трактуется как самовоздействие.

Например, оператор:

```
CALL %DIR('00060000')
```

приводит к самоуничтожению программы.

Группы директив

Ниже дается краткая справка о группах директив по функциональному признаку, а затем в алфавитном порядке мнемоник приводится их детальное описание. Имя программы всюду обозначается как NAME, числовые параметры — символы I, J, K, L, M, N в количестве, максимально допустимом в данном параметре (при использовании десятичного формата). Имена директорий обозначаются как DYNAMЕ, номер — ND. Всюду, за исключением директив LY и DY, вместо номера директории можно употреблять не более, чем 8-символьное имя.

1. Загрузка, сброс и уничтожение программ (DE, DL, DU, FI, FG, LO, LG, LR, LY):

DE NAME — уничтожь программу NAME в оперативной памяти;

DL NAME NAME1 NNNNN — уничтожь программу NAME, загрузи NAME1

и активируй с NNNNN-го адреса;

DU NAME — сбрось программу NAME в ФС.

Программа помещается в файл-программу NAME той ФС, к которой она приписана;

FD NAME DYNAMES — перенеси программу

из файла NAME директории DYNAMES ЦМ в такой же файл ЭД;

FI NAME DYNAMES — загрузи программу

из файла NAME директории DYNAMES ЦМ;

FG NAME DYNAMES NNNNN — то же с активацией

программы с NNNNN-го адреса;

LO NAME — загрузи программу из файла NAME ФС на ЭД.

При отсутствии файла-программы NAME запрос будет передан ЦМ;

LG NAME NNNNN — то же с активацией

программы с NNNNN-го адреса;

LR NAME NAME1 NNNNN — загрузи

с переименованием и активируй;

LY NAME ND NNNNN — загрузи, прикрепи

к директории ND и активируй;

2. Воздействие на программы (AL, AS, DR, DY, GO, ON, OF, RE, SU, TI, TS):

AL NAME NNNNN MMMMMM — измени содержимое

ячейки NNNNN программы NAME на MMMMMM;

AS NAME N — прикрепи к терминалу N;

DR NAME N — переключи программу NAME на диск N;

DY NAME ND — прикрепи программу NAME

к ND-й директории ФС ЦМ;

GO NAME NNNNN — запусти NAME с NNNNN-го адреса;

ON NAME NK — уставь программе ключ NK;

OF NAME NK — погаси ключ NK;

RE NAME PR — уставь приоритет PR;

SU NAME — останови программу;

TI NAME NNNNNN — таймируй программу

NAME интервалом NNNNNN сек;

TS NAME NNNNNNN — приостанови на NNNNNNN

прерываний часов Одренка;

3. Справочные директивы (OU, PO, PR, WS):

OU NAME NNNNN MMM — выведи на ТЛ

содержимое MMM ячеек программы NAME, начиная с NNNNN-й;

PO — выведи на ТЛ специнформацию из ЦМ;

PR — выведи на ТЛ справку о распределении

ОП и состоянии программ;

WS — запрос справки о марке системы и ее состоянии;

4. Прочие (CI, CT, CS, DZ, M1, TM):

CI III — замени дополнительный символ прерывания на III;

CT ДД/МС/ГГ ЧЧ/МН/СС — коррекция даты и времени;

CS — проверка контрольной суммы системы;

DZ N — определение терминала N как DZM;

M1 MC NAME — настройка вызова программы NAME командой MC;

TM NNNNNNN — уставка количества

прерываний часов Одренка в «секунде».

При загрузке NNNNNNN = 10.

Детальное описание директив в алфавитном порядке

Цифры в скобках и слово кодов в кавычках — соответственно десятичный код директивы и его символьное представление, которое следует использовать в декодированном формате задания цепочки директив из программ. Примечание [ТЛ] означает, что директива разрешена только с терминала. Корневое слово мнемоники приводится за строкой директивы.

1. (13 — '000 =') — AL NAME NNNNN LLLLLL
(ALTER)

замени содержимое ячейки NNNNN программы NAME на LLLLLL

Замена выполняется, если адрес NNNNN принадлежит программе. Если директива вводится с терминала, ОС дополняет ее директивой OU, в результате чего при правильном исполнении на ТЛ выводится новое содержимое указанной ячейки в формате,

описанном в директиве OU.

2. (16 — '000') — AS NAME N
(ASSIGN)

прикрепи программу NAME к ТЛ N

Программа прикрепляется к указанному терминалу.

Прием символа со старого ТЛ сбрасывается с ответом B1=1 в слово ответа контрольного поля инструкции заявки этого приема. Вывод не сбрасывается.

3. (21 —) — CI III [ТЛ]
(CHANGE INTERRUPT)

смени дополнительный символ прерывания на NNN

При загрузке для каждого ЛТЛ определено 2 таких символа: CTRL(B) с ASCII-кодом 2 и CTRL(W) с кодом #27. Директива меняет код #27 на указанный.

4. (25 —) — CS [ТЛ]
(CHECKSUM)

проверь контрольную сумму области кодов ОС.

Если она правильная, то на ТЛ выводится 'OK', иначе — рас-
согласование побитово в виде:

CS? = *IJKLMNIJ

(в единицу в IJKLMNIJ уставляются биты, несовпавшие с битами насчитанной при запуске ОДОС, контрольной суммы. Наличие редких единиц может говорить о сбоях памяти Одренка, иначе — о порче области кодов ОС).

5. (26 — '000*') — ST ДД/МС/ГГ ЧЧ/МН/СС
(CORRECT TIME)

коррекция времени. Здесь:

ДД — день месяца,	ЧЧ — часы,
МС — месяц,	МН — минуты,
ГГ — десятки и единицы года,	СС — секунды.

Например:

ST 21/07/86 09/52/34

устанавливает 21 июля 1986 г., 9 часов 52 мин 34 сек. При вводе из программы в декодированной форме примеру соответствует текст:

'000*21/07/8609/52/34'

Полная коррекция времени представляет собой достаточно емкую

процедуру, поэтому, если дата или время вводятся нулевыми, то меняются соответственно лишь время или дата. Проверка на корректность вводимой информации не осуществляется.

Замечание: отсчет даты ведется с 01/01/85 00/00/00;

6. (6 — '0006') — DE NAME
(DELETE)

уничтожь программу NAME в ОП.

Программа удаляется, а оперативная память реорганизуется. Перед этим сбрасываются все передачи программы, закрываются все открытые ею каналы обмена с ФС. Если уничтожаемая программа была источником каких-либо спецрежимов (например, SS), то режимы эти также сбрасываются. Если она была источником выполняемых в данное время директив, то это выполнение будет продолжено от имени системы.

В случае, когда реорганизация ОП по причине наличия настроенных ПД-передач в вышележащую память в момент ввода директивы не может быть выполнена, то программа переименуется в %%%% и будет реально удалена из ОП лишь по завершении обменов, препятствующих реорганизации.

Если директива была выдана программой, то при отсутствии программы NAME она считается выполненной правильно, но при работе с ТЛ будет выведен диагноз об ошибке.

7. (— '0936') — DL NAME NAME NNNNN
(DELETE-LOAD-GO)

составная директива. Соответствует цепочке:

DE NAME
LO NAME
GO NAME NNNNN

8. (23 — '000") — DR NAME N
(DRUM)

переключи программу NAME на диск N:

N=0 — переключение на диск, к которому прикреплена ОС,
1 — ФС ЦМ (диск-1),
2 — ФС ЭД (ЗУ, диск-2).

Переключение диска влияет на обмены из программ с ВУ типа 9 (барабан в системах программирования для ОДР), под которым в ОДОС поддерживается обмен с основными носителями ФС: ЭД и дисками ЦМ, и на директивы, содержащие в себе такие обмены

(LO, DU). В соответствии с номером диска для программы ОДОС выполняет обращения к ФС следующим образом: если номер диска равен 1, то все обращения направляются к ФС ЦМ, если равен 2, то к ЭД.

Файлы, открытые в старой ФС, закрываются.

Директива может быть применена к ОС. В этом случае ее действие распространяется лишь на атрибуты системы для ТЛ ввода директивы и лишь на программы, загружаемые позднее. Настройка ранее загруженных программ сохраняется. Для системы разрешенными являются значения $N=1,2$. Переключение системы на ЦМ выключает ЭД для директив LO, инициируемых с данного ТЛ.

Номер диска наследуется от источника директивы, т.е. программам присваивается номер диска программы — инициатора загрузки или терминала ввода директивы.

При начальной настройке для ОС уставляется диск-2, если есть ЭД, иначе диск-1.

9. (4 — '0004') — DU NAME
(DUMP)

помести текущее состояние программы NAME в ФС.

Текущий образ памяти программы помещается в файл-программу NAME той ФС, к которой приписана программа (!), если позволяют размеры, в противном случае файл создается. Наличие у программы открытых файлов не запоминается.

Процедура не выполняется, если программа работает с ВУ. На время исполнения директивы программа завешивается (DU), а в ФС помещается в состоянии останова (SU).

10. (19 — '000#') — DY NAME ND
(DIRECTORY)

прикрепи программу NAME к ND-й директории ЦМ.

Для программ директива автоматически дополняется директивой DR NAME 1.

Директива может быть применена к ОС. В этом случае ее действие распространяется лишь на атрибуты ТЛ ввода директивы и лишь на программы, загружаемые позднее. Настройка старых программ сохраняется. Для системы процедура прикрепления сопровождается контрольным запросом о способе доступа у ЦМ, который выводится в ответе на исполнение директивы. Для программ проверка корректности номера директории не выполняется.

При настройке ОС запрашивает в ЦМ номер основной дирек-

тории, к которой приписана данная ПМ, и запоминает его в контексте всех терминалов.

11. (15 — '000?') — DZ N
(DZM)

присвоение N-му (1, 2, 3, 4) ЛТЛ статуса DZM.

Передача с ЛТЛ(N) сбрасывается, маска приемника интерфейса с ним закрывается. Если директива приводит к смене ЛТЛ, определенного как DZM, маска приемника на старом канале DZM открывается.

$N=0$ трактуется как отказ от канала DZM. Маска приемника на канале DZM открывается.

12. (14 — '000>') — FI NAME DYNOME
(FIND)

загрузи программу из файла NAME директории DYNOME ФС ЦМ.

В качестве имени директории может фигурировать ее номер. По умолчанию будет использоваться номер директории из атрибутов источника директивы.

Локальная ФС не просматривается. Если источником запроса является программа, то она приостанавливается (WD) до выполнения запроса. При уставленном ОС ключе 1 перед загрузкой выполняется процедура обнуления всей свободной оперативной памяти.

13. (— '064>') — FD NAME DYNOME
(FIND-DUMP)

перенеси программу NAME из директории DYNOME ФС ЦМ в ФС на ЭД.

Составная директива. Эквивалентна цепочке:

FI NAME DYNOME
DU NAME
DE NAME

14. (— '009>') — FG NAME DYNOME NNNNN
(FIND-GO)

загрузи программу из файла NAME директории DYNOME ФС ЦМ и запусти ее с адреса NNNNN.

Составная директива. Эквивалентна цепочке:

FI NAME DYNOME
GO NAME NNNNN

15. (9 — '0009') — GO NAME NNNNN
(GO)

запусти программу NAME с адреса NNNNN.

В 8-е слово программы (регистр адреса) помещается NNNNN (если его значение лежит в пределах программы), с программы снимается состояние останова (SU), сбрасываются приостановка по обмену с ВУ (SBnn), ожидание спец- и LAM-прерываний. Если NNNNN=0 (опущено), состояние регистра адреса не меняется, т.е. выполняются только описанные выше сбросы. При сбросе обмена в слово ответа поля заявки обмена программы отдается ответ в виде уставленных в 1 битов Б1Б3.

16. (— '0093') — LG NAME NNNNN
(LOAD-GO)

загрузи программу из файла NAME ФС на ЭД и активируй ее с адреса NNNNN.

Составная директива. Эквивалентна цепочке:

LO NAME
GO NAME NNNNN.

17. (3 — '0003') — LO NAME
(LOAD)

загрузи программу из файла NAME ФС на ЭД.

При отсутствии файла-программы NAME выполняется директива FINAME ND, где ND берется из атрибутов источника директивы. То же при отсутствии ЭД или его отключении.

18. (18 — '000"') — LR NAME NAME NNNNN
(LOAD-RENAME-GO)

загрузи программу из файла NAME ФС на ЭД, переименуй ее в NAME и активируй с адреса NNNNN.

Правила исполнения директивы те же, что и для LO+GO, с той разницей, что в ОП не должно быть программы с именем NAME.

19. (27 — '000+') — LY NAME ND NNNNN
(LOAD-DY-GO)

загрузи программу из файла NAME ФС на ЭД, прикрепи ее к директории ND и активируй с NNNNN-го адреса.

Эквивалентна цепочке:

LO NAME
DY NAME ND

GO NAME NNNNN

20. (29 —) — M1 MC NAME [ТЛ]
(MACRO1)

создай вызов MC=LG NAME 20

Система запомнит указанную мнемонику и будет интерпретировать ее как директиву загрузки и запуска программы NAME с 20-го адреса. Разрешается по одной такой мнемонике для каждого из 4 ЛТЛ.

21. (8 — '0008') — OF NAME NK
(OFF)

Сбрось бит NK (0—23) слова «ключей» программы.

Гасится NK-й бит 30-го слова программы (см. директиву ON).

22. (7 — '0007') — ON NAME NK
(ON)

установь бит NK слова ключей программы.

Устанавливается в 1 NK-й бит 30-го слова программы, называемого словом ключей.

Оперирование со словом ключей позволяет гибко менять режим функционирования программ. Проверку состояния слова ключей из программы можно выполнить с помощью подпрограммы SSWTCH.

23. (12 —) — OU NAME MMMMM NNN [ТЛ]
(OUTRUT)

выведи на терминал содержимое NNN ячеек программы NAME, начиная с MMMMM-й.

При умолчании NNN считается равным 1.

Содержимое указанных ячеек выводится в формате:

*KKKKKK LLLLL: FFF X M/NNNN *IIIIII ABCD

где

*KKKKKK — адрес в восьмеричной форме,

LLLLL — адрес в десятичной форме,

FFF X M/NNNN — содержимое в форме команды,

*IIIIII — в восьмеричной форме,

ABCD — в текстовом виде

(в текстовом виде префикс управляющей группы, символ с кодом #76, выводится как ^).

Если после завершения вывода на терминал ввести прерывание

и пробел вместо директивы, то будет выведено NNN последующих ячеек.

24. (24 —) — PO [ТЛ]
(PRIORITY ODRA)

вывод на ТЛ, с которого введена директива, некоторой специальной информации из ЦМ (обычно: дата и время по часам ЦМ; в системах с ОДРАМИ предназначалась для справки типа PR о состоянии ЦМ).

25. (1 —) — PR [ТЛ]
(PRIORITY)

Справка о программах, находящихся в ОП машины в данный момент времени, их состоянии и атрибутах, наличии свободной памяти и атрибутах терминала, с которого введена директива.

Вид справки о программе:

#NAME PR NNNNN DRi DYjj TLk SST

где

NAME — имя программы;
PR — ее приоритет;
NNNNN — размер (слов);
i — номер диска для программы;
jj — номер директории ЦМ,
к которой она приписана;
k — номер терминала приписки;
SST — состояние программы:

DU — DUMPED (состояние записи программы в ФС ЦМ);
LO — LOADED (программа в состоянии загрузки);
SBnn — SUSBY (ожидание завершения обмена с каналом:

nn = 1 — ТЛО,
21 — 24: — ТЛ1-ТЛ4,
3 — спецВУ ЦМ,
6 — обмен с ФС ЦМ,
7 — «почта»,
19 — обмен через блок связи Б0633,
прочие каналы см. выше п.2.2).

SS — SPECIAL SUSPEND (экстракод: 'останови всех, кроме меня');
SU — SUSPEND (останов);

TO — TERMINAL OFF-LINE (при заявке обмена с выключенным терминалом);
WC — WAIT COMMAND (ждет освобождения ДПС);
WD — WAIT DIRECTIVE (ждет завершения выполнения своей директивы);
WLnn — WAIT LAM (ожидание LAM с позиции nn);
WLP*nnnnnnnn — WAIT LAM POSITIONARY,
*nnnnnnnn — маска;
WPnn — WAIT PINT — ожидание СПР с номером nn;
WPP*nnnnnnnn — WAIT PINT POSITIONARY,
*nnnnnnnn — маска;
WS — WAIT SYSTEM (ждет освобождения ТЛ системой).

Замечание. Программы могут быть приостановлены одновременно по нескольким причинам (ССП представляет собой побитовую маску причин приостановок). SST информирует лишь о наиболее «важной» (более левый бит в ССП) причине.

Последняя строка (справка о свободной на данный момент памяти и атрибутах системы, связанных с терминалом, с которого введена директива):

@@NO FREE JJJJ DRk DYll TLm

где

NO — логический номер станции;
JJJJ — размер свободной ОП;
k — номер диска ОС для данного ТЛ;
ll — номер директории ФС ЦМ;
m — номер терминала ввода директивы.

26. (11 — '000;') — RE NAME PR
(REVISE)

смени приоритет программы на PR [00—ZZ в символьном виде];

27. (19 — '0005') — SU NAME
(SUSPEND)

останови программу.

Программа переводится в состояние останова (SU);

28. (10 — '000:') — TI NAME NNNNNN

таймируй программу NAME интервалом NNNNNN сек.

Действие директивы состоит в том, что через каждые NNNNNN сек по часам Одренка с программы NAME будет сниматься состояние останова (SU). NNNNNN=0 означает отказ от таймирования;

29. (2 — '0002') — TM NNNNNNN
(TIME)

уставь масштаб времени.

NNNNNNN означает количество прерываний часов Одренка в секунде. Меняет «масштаб» секунды только для механизма таймирования программ. Нормально равно 10. Если уставляется в ноль, то часы маскируются полностью.

30. (28 — '000,') — TS NAME NNNNNNN
(TEMPORARY SUSPEND)

приостановка на NNNNNNN 100-мс интервалов.

Программа приостанавливается (SU) на NNNNNNN прерываний от часов Одренка;

31. (22 —) — WS [ТЛ]
(WHAT STATE)

Справка о версии системы в Одренке, времени ее старта, простое процессора. Может также дополнительно выводиться: Т? — было переполнение системных буферов обмена с ТЛ, или/и количество сбоев обменов с ЦМ. Переполнение терминальных буферов может произойти по причине повышенного потока диагностической информации на терминалы. При этом система сохраняет последнее сообщение, затирая предыдущее. Сбоем связи с ЦМ считается всякое нераспознанное периферийной системой обращение к ней из ЦМ. К этому могут приводить наводки на систему связи, а также применяемый ЦМ способ определения оперативности ПМ;

Директивы с кодами 17 и 20 резервированы.

Кроме описанных директив в ОДОС под упрощенный вызов часто используемых программ зарезервированы следующие мнемоники:

CA = LG	CALC 20	— вариант калькулятора с графикой;
ED —	VVVV	— загрузчик редактора текстов;
FC —	FORT	— монитор Фортрана;
HE —	HELP	— программа-советчик;
KA —	KALK	— калькулятор;

KS —	KISS	— распечатчик текстов;
LC —	LISC	— распечатчик карты глоб. символов программы;
LZ —	LOZO	— программа обмена ЭД ↔ ФС ЦМ;
MT —	DUDI	— программа обмена с магнитофонами;
PC —	XPLA	— монитор ассемблера PLAN;
TC —	TRAN	— транслятор TRAN;
SY —	SYST	— утилита для дополнительной работы с ЦМ;
ZA —	ZJAA	— основная файловая утилита;
ZW —	ZJAW/ZJAF	— утилиты для винчестерских/гибких дисков.

2 подобных вызова можно заказать в описателе системы (см. Приложение 2) и еще 4 настроить директивой M1.

6. ЭКСТРАКОДЫ

Процессор Одренка оперирует с 7-битовыми кодами команд: FFF=000 ÷ #177. Часть этих кодов не обеспечена аппаратно по техническим причинам или специально выделена под обращения к ОС. Эти команды и называются экстракодами.

В Одренке заложен следующий механизм обработки экстракодов: текущее состояние процессора (РОН'ы и необходимые индикаторы) прячутся в память программы (ячейки 0—10), включается привилегированный режим СУПЕРВИЗОР и выполняется команда из 28-й ячейки ОП. В ОДОС там лежит команда перехода на программный процессор экстракодов.

Экстракодами в зависимости от версии Одренка является часть арифметических операций из групп (2 старшие цифры кода) команд 04, 11, 12 и 13 и группы 14—16, выделенных для специального интерфейса «задача ↔ ОС».

На настоящий момент в ОДОС сложилось следующее использование последних групп:

Группа 14: передача операционной системе запросов на исполнение директив, синхронизацию со спецпрерываниями, LAM, на межпрограммный обмен. Организация режима «монополизации» процессора. Запросы атрибутов программ и ОП.

Группа 15: запросы на обмены с ВУ общего пользования, синхронизация работы программ с работой ВУ. Определение свойств модулей крейта. Организации режима контролирования одной программой других.

Группа 16: привилегированный вывод сообщений на терминал. Механизм LAM-драйвера. Запросы характеристик ЭВМ, времени и

даты, атрибутов программы или ОС. Изменения размеров программ.

В данном разделе описываются экстракоды групп 14—16. Следует обратить внимание, что различие в назначении функций экстракода проводится как по номеру поля аккумулятора команды, его содержимому, так и по значению адресной части команды. Ниже кода может приводиться известная ассемблеру мнемоника команды. Под комментариями ФЦ или ПП приводятся наименования функций или подпрограмм, реализующих соответствующий интерфейс с ОС и вспомогательный сервис. Символ / при описании структуры машинного слова делит его на 9 и 15 битов.

Группа 14

140 0 N(M) — выполни директивы.

Форма с остановкой при неисполнении.

ПП: DIRECT, %DIR

140 6 N(M) — выполни директивы.

Форма с ответом в программу.

ФЦ: %DIRA

X=0 — ответ на исполнение директив не отдается, а при неисполнении какой-либо из них программа-источник останавливается (SU).

X=6 — в X6 отдается ответ об исполнении директив в виде числа, равного номеру системного сообщения, выводимого на ТЛ по результату выполнения директив, если директивы были введены с ТЛ (OK=0). Список сообщений с их нумерацией приведен в Приложении 3.

N(M) — адрес 1-го слова 12-словного поля, описывающего директивы. Форматы описания определены выше в разделе 5.

141 0 N(M) — останови в ожидании СПР.

ПП: PINT, PINTA

N(M) задает адрес слова, содержащего номер СПР.

ОС останавливает программу в ожидании указанного сигнала от РП. Приостановки не будет, если в 16-м слове программы содержится пометка, соответствующая ожидаемому сигналу (вне зависимости от того, выполняли программы заявки на синхронизацию с СПР, или нет, ОДОС раздает пометки о возникающих СПР в 16-ые слова всех программ, находящихся в ОП).

Если номер СПР равен 63, то это запрос на приостановку по любому из 24 возможных сигналов, причем для таймированных

программ снятие состояния приостановки такого типа будет происходить и по истечении времени таймирования. В последнем случае (при активации по времени) программе устанавливается семафор 0 (0-й бит 31-го слова, с которым работают ПП SLITE и SLITET).

141 1 N(M) — имитируй СПР.

ПП: IINT

N(M) задает адрес слова, содержащего номер СПР, появление которого хотела симитировать программа.

Система трактует запрос, как сигнал с реального РП, т. е. раздает соответствующие пометки всем программам, находящимся в данный момент в ОП, в 16-ые (18-ые) слова. Со всех программ, ожидавших подобное событие, снимается состояние его ожидания (WP или WPP). Функция выполняется и при отсутствии РП.

141 2 N(M) = 141 1 N(M)

141 3 N(M) — останови в ожидании LAM.

ФЦ: LWAIT, LWAITA

N(M) — задает адрес слова, содержащего номер LAM в диапазоне 0—23 или их маску. Слово это считается побитовой маской, если в нем установлен 0-й бит, означающий заявку ожидания прерывания часов Одренка. Биты 1, 2, 3, ..., 23 маскируют соответственно позиции 1, 2, 3, ..., 23 крейта.

ОС считывает регистр LAM Одренка и, при наличии указанных сигналов, программу не останавливает. В противном случае на указанные LAM открывается системная маска, а программа останавливается.

После активации из такого состояния программа сама должна опросить соответствующие модули крейта и обработать их прерывания. Системная маска на пришедшие сигналы закрывается. С целью информирования о прерывании таймера пометки о всех возникших по маске программы LAM передаются в 17-е слово программы (B0=1 соответствует LAM 0).

Остановка по LAM с номером 0 трактуется ОС как остановка в ожидании прерывания только от часов.

141 4 N(M) — имитируй LAM.

ПП: EMITL

N(M) задает адрес слова, содержащего номер (1—23) позиции, от которой моделируется LAM-прерывание. Обращается системой как реальный LAM-сигнал с указанной позиции.

141 5 N(M) — останови в ожидании СПР по маске
ПП: PINTM

N(M) задает адрес слова, содержащего маску СПР. Биты 0, 1, 2, ..., 23 маскируют СПР 24, 23, 22, ..., 1.

ОС останавливает программу в ожидании любого из указанных сигналов РП, если пометки о таковых еще не содержались в 16-м слове программы.

141 6 N(M) — учет реперных СПР.
ПП: %SRPM, %LRPM

N(M) задает адрес слова, содержащего маску СПР, по которой ОС обновляет (с затиранием старого содержимого) 18-е слово программы (слово «реперных СПР»). Обновление осуществляется лишь при возникновении прерываний, указанных маской (хотя бы одного из них). Формат маски такой же, как и для экстракода 141 5....

142 X N(M) — межпрограммный обмен.
ПП: BERI, пакет ПП: DAJ/NA

В зависимости от номера аккумулятора X имеет следующие 4 формы:

X=3 — БЕРИ. Определение адреса N(M) обменного буфера в программе. Система запоминает этот адрес в своей таблице описания программы. В частности, вся программа может быть заявлена как обменный буфер.

X=0 — ДАЙ, НА. Собственно обмен. N(M) задает адрес 3-слового поля, содержащего:

- 1 — имя NAME программы-адресата (абонента),
- 2 — КСЛ/(смещение относительно 1-го слова обменного буфера абонента, равно 0 означает с первого слова)
- 3 — H/(адрес обмена в самой программе).

Направление обмена задается значением H: H=0 — КСЛ слов перекладывается из программы NAME в программу, выставившую экстракод, по указанному адресу (ДАЙ); H=#400 — информация передается программе NAME (НА). Если КСЛ=0, то передается 512 слов.

При выполнении обмена ОС зануляет 1-е слово этого поля (NAME), иначе не меняет. Причинами невыполнения могут быть:

- отсутствие адресата;
- неопределение им адреса обменного буфера к моменту обмена;

— попытка передачи информации за пределы любой из программ, как пассивной (выставившей БЕРИ), так и активной (ДАЙ, НА).

X=4 — межпрограммный обмен с ЦМ. Запрос формата ДАЙ передается на исполнение ЦМ. При запросе НА пересылаются и необходимые данные. Детали этого обмена приведены в Приложении 1.

Прочие значения X резервированы.

143 0 0 — Останови всех, кроме меня (монополизация процессора — SPECIAL SUSPEND).
ПП: %STOP

Все программы, находящиеся в ОП, останавливаются (SS) до повторного употребления этого экстракода. Действие экстракода распространяется и на программы, загружаемые в ОП после инициации режима.

145 0 N(M) — дай мое имя
ФЦ: MYNAME

В N(M) отдается имя, под которым программа загружена в ОП.

145 1 N(M) — дай информацию о распределении ОП
ПП: %COR

N(M) задает адрес не менее, чем 12-слового поля программы, в которое ОС положит:

- сл 1 — количество программ, находящихся в данный момент под ее контролем;
- сл 2 — размер свободной памяти на момент запроса;
- сл 3—12 — имена находящихся в памяти программ (безотносительно к порядку загрузки) в количестве, переданном в слове 1.

145 3 N(M) — дай мой номер.
ФЦ: MYNOMER

В N(M) отдается логический номер, приписанный станции ее описателем из ФОС.

Остальные коды этой группы не задействованы и запрещены для использования.

Группа 15

150 X N(M) — ожидание завершения обмена с ВУ.

SUSBY

Номер X и его содержимое несущественны.

$N(M)$ = ТИП устройства. Используемые типы ВУ описаны ниже при описании 157-го экстракода. Экстракод выполняется следующим образом:

— определяется номер NN канала, отведенного для обменов с указанным ВУ (список каналов ОС с их нумерацией см. 2.2);

— проверяется, занят ли этот канал и, если это так, программа останавливается (SB NN) до завершения процесса обмена вне зависимости от того, иницирован этот обмен данной или другой программой. Описанный механизм может «сработать» и автоматически — при попытке заявки операции с занятым ВУ, если только не используются специальные способы заявки (дополнительно к основной операции режим непосредственного ответа). Возможные режимы взаимодействия программ с ОС по поводу обмена с ВУ описаны в спецификациях 157-го экстракода.

Коды 151 — 154 не задействованы и запрещены для употребления.

155 0 $N(M)$ — организация режима контролирования экстракодов других программ.

Конкретная функция экстракода задается содержимым X0:

$(X0) = 0$ — заявка режима. ОС запоминает адрес $N(M)$, который должен определять 3-словное поле:

сл 1 — несущественно;

сл 2 — имя программы, экстракоды которой программа желает контролировать (равно нулю — контроль экстракодов всех программ);

сл 3 — несущественно.

ОС перестраивает механизм обработки экстракодов под данный режим и в ответ на принятие заявки передает в 1-е слово этого поля собственный адрес начала программы, требуемый ей для доступа в вышележащие подконтрольные программы. Далее, при возникновении в подконтрольных программах экстракодов они останавливаются операционной системой (SU), с контролирующей программы снимается состояние останова и во 2-е слово определенного выше поля передается имя подконтрольной программы, а в 3-е — адрес ее начала (8-е слово подконтрольной программы в этот момент указывает на следующую за экстракодом команду).

$(X0) = 1$ — заказ ОС на выполнение одного экстракода подконтрольной программы. $N(M)$ определяет адрес слова, содержащего имя подконтрольной программы. ОС останавливает контро-

лирующую программу (SU), возвращает подконтрольную на команду назад, запускает ее и выполняет один экстракод сама. При отсутствии программы с указанным именем во 2-е слово отдается 0 и остановка контролирующей программы не осуществляется. Во всех случаях управление далее передается планировщику программ;

$(X0) = 2$ — отказ от контроля. ОС восстанавливает обычные механизмы обработки экстракодов.

156 X $N(M)$ — дай позицию модуля в крейте и его свойства

ALLOT

ФЦ: MPPR

$N(M)$ задает адрес слова программы, содержащего наименование модуля. Если модуль с таким именем присутствует во вписанном при загрузке в систему описателе конфигурации станции, то его позиция в крейте (1—23) отдается в X, а слово свойств (в формате, в котором оно присутствует в описателе) отдается в 10-е слово. 9-е слово программы зануляется для поддержания совместности с некоторыми старыми программами.

157 X $N(M)$ — работа с ВУ и файлами.

PERI

Наиболее емкий экстракод системы, с помощью которого реализуются обмены с ВУ коммунального использования. В ОДОС таковыми являются терминалы, ЭД, диски ЦМ, блок связи Б0633 и ряд устройств и псевдоустройств ЦМ.

Всюду в этом экстракоде $N(M)$ задает адрес так называемого Контрольного Поля — КП, в котором указывается, с каким ВУ и какую операцию хочет выполнить задача и параметры этой операции. Второе слово этого поля — КП1 во всех формах запросов выделено под ответы о стадии и результатах операции и называется словом ответа.

Общие положения об обмене:

Взаимодействие с ВУ коммунального использования осуществляется через каналы доступа к ним (см. 2.2).

Каждый канал может находиться в двух основных состояниях: «свободен», и тогда он может быть предоставлен для реализации запроса на обмен, и «занят» — когда он уже задействован под обмен. Некоторые каналы могут находиться и в других состояниях, например, для терминальных каналов возможны еще и состояния «оперативен» или «неоперативен», т. е. логически включен или выключен.

Заявка на взаимодействие с ВУ принимается к исполнению с ответом КП1 (Б0) = 1 — «в исполнении», если канал связи с ним свободен, иначе — программа приостанавливается (SB NN) так, чтобы при освобождении ВУ она повторила экстракод. Приостановки не происходит и отдается ответ КП1 (Б4) = 1 — «занято», если программа предусмотрела это добавлением константы #1000 к коду основной операции (заявка так называемого непосредственного ответа).

По завершении работы канала ОДОС снимает со всех программ, ожидающих этого события, состояние приостановки (SB NN), а программе — источнику операции сигнализирует также и занулением бита Б0 слова ответа КП1 и передачей в 1—5 биты этого слова ответа о результатах операции:

КП1 (Б1) = 1 — отброшена до исполнения (например, прикреплением ТЛ к системе);

КП1 (Б1, Б3) = 1 — отброшена (по директиве GO, специальным сбросом передачи, по истечении времени ожидания ответа ЦМ на запрос ПМ);

КП1 (Б3) = 1 — с фатальной ошибкой,

а также

КП1 (Б9-23) — остаток счетчика при неполном обмене.

Несколько по-другому биты слова ответа используются в операциях создания и открытия файлов. Это пояснено в спецификациях 157-го экстракода для дисков.

Наиболее часто употребляемая структура КП:

КП0 — ТВУ/ОПЕРАЦИЯ,

КП1 — слово ответа,

КП2 — счетчик передачи (слов или знаков в пределах до 32К),

КП3 — адрес обменного буфера в программе,

(КП4 — адрес в файле),

ТВУ — это номер, идентифицирующий конкретный тип ВУ (диск, терминал, ...) в рамках данной системы. Во многих случаях он совпадает с номером канала доступа к устройству (см. 2.2), но, вообще говоря, это не так, поскольку ОДОС поддерживает взаимодействие с рядом однотипных устройств. Реальное соответствие между типом ВУ и каналом доступа к нему устанавливается либо по атрибутам источника запроса, либо явным выбором канала в запросе (например, при работе с терминальными подканалами номер канала передается в аккумуляторе экстракода). Добавление к номеру ТВУ в запросе константы #400 служит для информиро-

вания ОС о том, что аккумулятор в поле экстракода содержит дополнительную информацию об устройстве. В ОДОС это используется лишь в операциях открытия/создания файлов.

ОПЕРАЦИЯ — один из возможных кодов операций с конкретным ВУ. Может содержать аддитивно добавляемые константы — расширения кода операции, чем заявляются дополнительные возможности в рамках основной операции или изменяется интерфейс с ОС.

Ниже в порядке нумерации типов ВУ и кодов операций приводятся полные спецификации 157-го экстракода в системе ОДОС. Номера каналов взаимодействия с ВУ приводятся там, где не совпадают с их типом.

1. ТВУ = 1 — терминалы (ТЛ)

каналы: 1 — ВТЛО

21 ÷ 24 — ЛТЛ1 ÷ ЛТЛ4

пакеты ПП: VIDEOT (VIDEO), VIDA, VI, ...

КП0 — 1/ОПЕРАЦИЯ,

КП1 — слово ответа,

КП2 — счетчик передачи (знаков в пределах до 32К),

КП3 — адрес 1-го знака в обменном буфере программы.

Запросом к этому типу ВУ программа в основном может обратиться только к тому терминалу, к которому она приписана (из программы номер терминала приписки можно определить с помощью одной из форм экстракода 165).

Общие положения:

Обмен с ТЛ ведется посимвольно, но программы могут заявлять обмены любым, в пределах 32768 (32К), количеством знаков (здесь существенно отличие термина «символ» от термина «знак», так как, вообще говоря, символы в Одренке имеют 2-знаковое внутреннее представление; знаком является и переключение регистра алфавита). Принятые к исполнению заявки на чтение с ТЛ могут отвергаться до исполнения с ответом КП1 (Б1) = 1 (например, при отборе ТЛ системой). Инициированный вывод не сбрасывается, за исключением случаев применения к программе, заявившей этот вывод, директивы GO или уничтожения этой программы.

Начальное состояние вывода: ЛАТ, т.е. если в выводимой информации нет явного определения вида алфавита (знаки с внутренними кодами: #74 — ЛАТ, #75 — РУС), то ОС конвертирует текст в латынь до явного переопределения алфавита (смены

регистра). Переопределение алфавита действует только на информацию, выводимую по данной инструкции. Действие регистра управляющей группы знака с кодом #76 распространяется лишь на следующий за ним знак и не отменяет регистра алфавита. Реальная длина выводимого на ТЛ сообщения меньше счетчика операции на количество знаков с кодами #74, #75 и #76.

1.1. ОПЕРАЦИЯ=0 — вывод на терминал КП2 знаков. КП3 задает адрес 1-го знака (т. е. адрес слова и позицию 0, 1, 2, 3 знака в нем). КП2 знаков выводится на терминал приписки программы.

1.2. ОПЕРАЦИЯ=2 — сброс приема с ТЛ.
ФЦ: %REJT

Заказанный на ТЛ прием символа отвергается с ответом КП1 (Б1Б3)=1. Вывод не сбивается. Прием с подканала отбрасывается с тем же ответом. После выполнения процедуры управление передается планировщику задач.

1.3. ОПЕРАЦИЯ=3 — прием с подканала ТЛ.
ФЦ: %GETCH

Содержимое X указывает номер ТЛ, к которому направлена заявка (равно 0 — свой). Символы с клавиатуры, не затребованные обычными заявками приема, передаются в указанный буфер программы до исчерпания счетчика в КП2. По мере приема ОС, сохраняя уставленным нулевой бит слова ответа, помещает на его 15 правых битах количество принятых знаков. По исчерпании счетчика Б0 слова ответа зануляется.

1.4. ОПЕРАЦИЯ=4 — повторение символа.
пакет ПП: OUNC (со входом OUNB)

КП3 задает адрес слова программы, содержащего требуемый символ в знаках 2, 3 слова: X.X.P.C (P—код регистра, C—код знака), либо его ASCII-код на 8 правых битах, если в операции содержится указание на байтовый обмен (см. ниже). Заданный символ выводится на терминал КП2 раз.

1.5. ОПЕРАЦИЯ=5 — чтение слова до спецсимвола. Размер слова не более 12 символов.

КП0 — 1/ОПЕРАЦИЯ.

КП1 — слово ответа,

КП2 — знаков в слове максимально (1 — 12),

КП3 — адрес 1-го знака слова в программе,

КП4 — КСС/(адрес 1-го спецсимвола).

Спецсимволы без регистров, кол-во их (КСС) не более 12. Управляющие символы, за исключением символов редактирования (CR, CL, IC, DC и ETX), трактуются как спецсимволы. Переключения алфавитов игнорируются. Помимо принятого слова отдаются ответы:

КП1 = остаток счетчика,
буфер + 3 = НСС.КЗН.Р.С (4-е слово буфера),

где

НСС — номер принятого спецсимвола,
НСС = КСС + 1, если спецсимвол не принят, или принят управляющий символ,
КЗН — кол-во принятых знаков,
Р.С — Регистр.Символ (спецсимвол).

1.6. ОПЕРАЦИЯ=16 — прием символов.

КП2/2 символов принимаются с ТЛ и последовательно упаковываются в буфер программы (каждый символ трансформируется перед передачей в программу в два 6-битовых знака внутреннего представления символов: Р.С).

1.7. ОПЕРАЦИЯ=17 — прием символов с выводом эха на ТЛ.

То же самое, что и режим 16, но реализуется программный полудуплекс, т. е. перед передачей символа в программу ОС выводит его эхо на экран.

Для всех операций, кроме 2 и 5, добавление к коду операции константы #100 рассматривается как заявка на байтовый обмен. В этом случае счетчик в КП2 задает реальное количество байтов (без умножения на 2 как для режима обычного приема с ТЛ), а каждый байт берется/помещается из/в 8 правых битов очередного слова без какой-либо фильтрации.

2. ТВУ=2 — Устройство ЦМ. Все заявки на взаимодействие с ним передаются в ЦМ. ОДОС трактует его как ВУ символьного обмена, т. е. КП совпадает с КП для ТЛ.

ОПЕРАЦИЯ = 0 — прием символов;
= 4 — вывод символов.

3. ТВУ=3 — обмен со специальными устройствами ЦМ (например, DZM). Устройство рассматривается как ВУ символьного обмена. КП как для ТВУ=19.

ОПЕРАЦИЯ = 0 — прием символов;
= 16 — вывод символов.

4. ТВУ=4 — аналогично ТВУ=3.

5. ТВУ=5 — вся информация по взаимодействию с ВУ этого

типа передается в ЦМ. Устройство рассматривается как накопитель с последовательным доступом. Протокол взаимодействия с ЦМ для него аналогичен протоколу для ВУ типа 9.

6. ТВУ=6 — аналогично ТВУ=5.

7. ТВУ=7 — это устройство выделено для реализации программами дополнительного подмножества сеансов связи с ЦМ, т. е. взаимодействий с ЦМ, не предусмотренных ОДОС. Правила организации сеансов приведены в Приложении 1.

КП0 — 7/ОПЕРАЦИЯ

КП1 — слово ответа

КП2 — счетчик передачи (до 20 слов)

КП3 — адрес обменного буфера в программе

(Ограничение счетчика для программного способа обмена с ЦМ связано с использованием в этих обменах системных буферов. Счетчик обмена способом ПД от 1 до 64К слов.)

ОПЕРАЦИЯ = 1 — чтение программно КП2 слов;

= 16 — пиши программно КП2 слов;

= 17 — пиши-читай программно КП2 слов;

= 19 — пиши-ПД-читай.

Счетчики программного обмена для обоих направлений — КП2. Счетчик передачи в прямом доступе и направление обмена определяются в рамках диалога в программной моде. При выполнении операции «пиши» для операции 19 ОДОС добавляет базовый адрес программы к 5-му слову высылаемого программой блока (дополнительно см. Приложение 1).

8. ТВУ=8 — системное устройство. С его помощью процессы FI и DU взаимодействуют с драйвером интерфейсного модуля связи с ЦМ.

9. ТВУ=9 — обмен с ЭД или ФС на дисках ЦМ.

каналы: 6 — диски ЦМ,

18 — ЭД

пакеты ПП: DRUM, %DRD, ...

Номер канала определяется по номеру диска, связанного с программой: 1—канал 6, 2—канал 18. При работе с 6-м каналом обращения к ФС адресуются к директории, к которой приписана программа.

Доступ к дискам в ОДОС ориентирован на абсолютную адресацию информации на них из программ. Поэтому взаимодействие с оформленными структурами ФС—файлами—требует посреднической работы специальных пакетов (типа DRUM), берущих на

себя, в частности, и функции по формированию этих абсолютных адресов. Существует 3 разных структуры КП для операций с этими накопителями.

9.1. КП для операций чтения/записи:

КП0 — ТВУ/ОПЕРАЦИЯ (ТВУ = 9 или #400 + 9 — несущественно),

КП1 — слово ответа,

КП2 — счетчик в словах,

КП3 — адрес буфера в программе,

КП4 — адрес на накопителе (адрес 1-го слова = 0);

ОПЕРАЦИЯ = 0 — чтение,

= 1 — запись.

Для этих операций, помимо обычных ответов, в КП1 всегда отдается адрес следующего за последним обменным словом ОП программы.

9.2. Открытие существующего файла:

КП0 — ТВУ/64,

КП1 — слово ответа,

КП2 = имя файла,

КП3 — XXXX,

КП4 — XXXX,

КП5 — XXXX,

КП6 = БТ./АСМ.

Перед обращением: если ТВУ=9, то номер X=ПНФ (программный номер файла: 0—7), если ТВУ=#400+9, то ПНФ задается содержимым X.

БТ — БОБ1 — 2-битовое поле типа файла:

00 — простой файл

01 — сложный

10 — файл-программа

11 — специальный

АСМ — адрес спецмассива, куда ОС отдает дополнительные ответы об атрибутах файла: ИМЯ и АДРЕС НАЧАЛА файла с БИТАМИ ТИПА в позиции массива, соответствующие программному номеру файла (так, для ПНФ=0 ИМЯ будет передано в 1-е слово спецмассива, БТ.АДРЕС НАЧАЛА — во 2-е; ПНФ=1: ИМЯ — в 3-е слово, БТ.АДРЕС НАЧАЛА — в 4-е, ...).

XXXX — несущественные поля.

При выполнении данной операции ОС предусматривает синхро-

низацию программы с работой своих процедур взаимодействия с ФС, приостанавливая программу таким же способом, как если бы сама программа выполнила экстракод ожидания завершения обмена с данным устройством. (Синхронизация по Б0 слова ответа здесь невозможна из-за специфики использования этого бита в операциях открытия/создания файла; см. ниже).

Ответы на операцию открытия:

КП1=0 — ОК

КП5=размер файла (слов)

дополнительные ответы:

АСМ+2*ПНФ = ИМЯ файла

АСМ+2*ПНФ+1 = БТ.(22-битовый АДРЕС НАЧАЛА файла относительно 0 каталога)

КП1(Б0)=1 — ОТКАЗ и тогда

КП1(Б09-Б23)=1 — НЕТ ФАЙЛА,

=2 — попытка открытия файла-программы,

=3 — сбой носителя.

Дополнительный ответ при отказе:

АСМ+2*ПНФ = 0.

Система предупреждает ошибочное открытие файлов с типами 2 и 3, требуя для этого совпадения типа файла с типом в запросе. Если тип в запросе БТ=3, то открывается любой файл. Всегда открывается первый файл с указанным именем.

9.3. Создание файла:

КП0 — ТВУ/256,

КП1 — слово ответа,

КП2 = имя файла,

КП3 — ХХХХ,

КП4 — ХХХХ

КП5 = его размер (слов)

КП6 = БТ./АСМ

ТВУ, определение ПНФ, назначение КП6, синхронизация такие же, как и для открытия. Файл типа БТ создается безотносительно к тому, содержится или нет в ФС файл с таким именем.

Ответы на операцию создания:

КП1=0 — ОК

дополнительные ответы:

АСМ+2*ПНФ = ИМЯ файла

АСМ+2*ПНФ+1=БТ.(22-битовый АДРЕС НАЧАЛА файла

относительно 0 каталога)

КП1(Б0)=1 — ОТКАЗ и тогда

КП1(Б06-Б23)=1 — на носителе макс.файлов

= остаток памяти, если ее не хватает под файл

= 3 — сбой носителя.

Дополнительный ответ в этом случае:

АСМ+2*ПНФ = 0.

Замечание. Отсутствие операции закрытия файла объясняется тем, что ОДОС не ведет детального учета открываемых программами файлов. Это реализуется пакетами ПП для работы с ФС, где и локализуется процедура закрытия.

10. ТВУ=10÷18 : системные и резервные устройства. Системные устройства служат для поддержки протокола обменов с ЦМ и интерфейса системных процессов с драйвером связи с ЦМ. Способы их использования описаны в Приложении 1. Запрещены для обращения из программ.

11. ТВУ=19 — обмен через блок связи Б0633 (связь с ВИШНЕЙ).

Пакет ПП: INDS (со входом OUDS)

КП0 — 19/ОПЕРАЦИЯ

КП1 — слово ответа

КП2 — счетчик знаков

КП3 — адрес обменного буфера (адрес 1-го знака),

КП4 — слово адресов в системе связи (S7S6S5S5, где S7 = 0,1,2,3 — номер выхода блока),

КП5 = УБ.0.0.0, где УБ — управляющий байт.

Детально работа устройства описана в руководстве по блоку Б0633.

ОПЕРАЦИЯ= 0 — пиши в устройство;

=16 — читай из устройства.

Для обоих направлений обмена реализован режим с управляемым из программы переключением младшего адреса (S5). Пределы счетчика: 1 — 32К знаков.

Группа 16

160 X N(M) — привилегированный вывод на ТЛ

SUSTY, DISTY

оператор PAUSE

ПП: DISP

Привилегированный вывод длинного (до 48 знаков) сообщения из

программы на терминал ее приписки с остановкой (SU), если номер $X=0$ — SUSTY, или без таковой, если номер $X=1$ — DISTY.

$N(M)$ задает адрес сообщения в форме NCH/ACH1, где NCH — кол-во знаков в сообщении (≤ 48), ACH1 — адрес 1-го знака сообщения.

При необходимости программа приостанавливается, а на терминал ее приписки выводится:

#NAME DISPLAY: — «сообщение»,
или (в случае с остановкой)

#NAME HALTED: — «сообщение».

Подобный вывод сообщений на ТЛ пользуется привилегиями по сравнению с выводом 157-м экстракодом: система переносит сообщение в свой буфер и выводит его на терминал от своего имени (сбрасывая при необходимости с соответствующего канала заявку на прием символов, если этот прием выполнялся по заказу из программы). В системе предусмотрена возможность буферирования до 5 выводимых таким образом сообщений. При заполнении системных терминальных буферов программы при таких запросах приостанавливаются (безотносительно к номеру X) до разгрузки этих буферов.

161 X $N(M)$ — привилегированный вывод на ТЛ
DISP, SUSWT

Действие экстракода такое же, что и 160-го, с той разницей, что $N(M)$ представляет собой само сообщение, состоящее из 2-х знаков.

163 X $N(M)$ — заявка/отказ LAM-драйвера.
AUTO

пакет ПП: AULS (со входом AULE)

Номер $X=0$ — первая заявка, равен 1 — повторные или отказ. $N(M)$ задает адрес 1-го слова 3-словного поля, в которых перед обращением указываются:

сл 1 — маска ожидаемых LAM,
сл 2 — стартовый адрес LAM-драйвера,
сл 3 — адрес не менее, чем 11-словного массива программы для сохранения РОН'ов остальной части программы перед запуском LAM-драйвера.

Указанные адреса и маска запоминаются и, при возникновении LAM по маске программы, ОС выполняет следующие действия:

- зануляет у себя переданную маску во избежание перезапуска LAM-драйвера в фазе его активности;
- перепрятывает 11 РОН прерванной части программы в указанное поле программы;
- подменяет текущий адрес программы на адрес старта LAM-драйвера;
- помечает для программы 100—200 мс активности безотносительно к текущему ее состоянию;
- передает управление планировщику задач.

По истечении времени повышенного приоритета ОС возвращается к обычному алгоритму планирования, т.е. если программа была остановлена, то и останется остановленной, но в состоянии драйвера.

Нормальный выход из LAM-драйвера — это тот же экстракод, только с $X1$. Будут восстановлены регистры прерванной части программы. Одновременно можно сменить все параметры запроса, в частности, отказаться от него, использовав нулевую маску LAM.

165 X $N(M)$ — запросы времени, атрибутов программы, изменение ее памяти, переключения ДУ, ...

ПП: DATE, TIME, MCOR, CCOR, DATEC, TIMEC ...

Тип запроса задается значением адресной части $N(M)$, а ответы в большинстве случаев отдаются в аккумуляторы X , $X+1$, причем для $X7$ смежным является $X0$.

$N(M)=0$ — запрос даты в числовом виде. Дата, отсчитанная от 00/00/00 1-го января 1900 г. отдается в форме целого числа в X , например, дата в числовом виде в полночь на 01/01/85 имеет значение 31046.

$N(M)=1$ — запрос даты в символьном виде. Дата текущих суток в форме ДД/МЦ/ГГ отдается в X , $X+1$, например, дата в полночь на 01/01/85 передается как $X=01/0$, $X+1=1/85$.

$N(M)=2$ — запрос времени в символьном виде. Время текущих суток в форме ЧЧ/МН/СС отдается в X , $X+1$, например, время полдня 12/00/00 передается как $X=12/0$, $X+1=0/00$.

$N(M)=3$ — запрос размеров ОП, отведенной программе. Размер памяти в словах отдается в X . В ОДОС он всегда кратен 256.

$N(M)=4$ — увеличение размеров программы в памяти. Запрос памяти в словах передается программой в X . Если по ресурсам изменение возможно, то ОС выполняет запрос, отдавая в X новый размер программы. Выполнение запроса не связано с расположением программ в ОП. Отказом от изменения размеров

может также явиться наличие настроенных передач в режиме прямого доступа в область памяти, подлежащей перемещению при увеличении размеров данной программы.

Для программы, загруженной последней разрешается и уменьшение размеров.

Во всех случаях в X возвращается истинный размер программы.

$N(M) = 5$ — запрос спецсвойств процессора. Этим запросом пользуются некоторые тестовые программы ЭВМ ОДРА. Для использования их на Одренке в данной версии отдается такой же ответ, что и для модели ОДРА-1325: $X = \# 205/0$.

$N(M) = 6$ — запрос имени и терминала приписки. Имя, под которым программа загружена в ОП, отдается в X, номер терминала приписки в $X+1$.

$N(M) = 7$ — запрос номера диска (НДК) и номера директории (НД). Номер диска ОС для терминала приписки запрашивающей программы и номер директории ФС ЦМ, к которому приписана ОС для этого ТЛ отдается в X в формате: НДК./НД директории (НДК помещен в 0-й знак слова левые 6 бит). В $X+1$ отдаются аналогичные атрибуты программы.

$N(M) = 8$ — запрещен.

$N(M) = 9$ — запрещен.

$N(M) = 10$ — запрос даты в числовом виде из ЦМ. Ответ такой же, как и для $N(M) = 0$. При сбое обмена с ЦМ: здесь и в двух следующих запросах требуемая величина отдается нулевой.

$N(M) = 11$ — запрос даты в символьном виде из ЦМ. Ответ такой же, как и для $N(M) = 1$.

$N(M) = 12$ — запрос времени в символьном виде из ЦМ. Ответ такой же, как и для $N(M) = 2$.

$N(M) = 13$ и далее запрещены.

7. ЗАПРЕЩЕННЫЕ КОМАНДЫ

Запрещенными являются команды, для кода которых не предусмотрено ни аппаратной, ни программой реализации, либо, если код разрешен, другие параметры команды выходят из дозволенных пределов.

Основными причинами возникновения запрещенных команд являются потеря управления в программе или неверное использование экстракодов. Потеря управления возникает, как правило, из-за ошибок в параметрах подпрограмм или функций, поскольку

полный контроль на обусловленность запроса нецелесообразен из-за больших накладных расходов или невозможен из-за отсутствия адекватных средств. Следует также учитывать возможность аппаратных сбоев или порчи программ из-за неправильного функционирования других программ, загруженных ранее: в Одренке не заложена защита от несанкционированного доступа за старшие адреса программы (более того, этот недостаток используется в некоторых системах подготовки программ...).

В ОДОС предусмотрены следующие действия по запрещенной команде:

— программа — источник команды переводится в состояние останова (SU), причем с нее снимается таймирование, если было заявлено;

— на терминал приписки программы выводится сообщение:

```
#NAME ILLEGAL *IIII JJJJ: FFF X M/NNNN *LLLLLLLL
где
```

NAME — имя программы,

IIII, JJJJ — восьмеричный и десятичный адреса команды,

FFF — код команды (восьмеричный),

X — номер аккумулятора в команде,

M — номер модифицирующего аккумулятора,

NNNN — адресная часть команды (восьмеричная).

*LLLLLLLL — восьмеричное значение ячейки;

— в регистр адреса программы помещается адрес запрещенной команды.

Если в качестве имени фигурирует @@@@, то запрещенная команда локализована в ОДОС и это говорит о нарушении логики ее работы. Причины: порча системы или сбой процессора. Исполняемый запрос, как правило, теряется, но операционной системой предусмотрена локализация подобных явлений для предоставления возможности пользователю предпринять аварийные действия.

Приложение 1

ПРОТОКОЛЫ ОБМЕНА С ЦМ

Аппаратная среда обмена ПМ ↔ ЦМ состоит из КАМАК-блока связи (БС) Б0633 (Б0634) в ЦМ, соединяемого тремя РК-кабелями с интерфейсными модулями (ИМ) У0605 в периферийных машинах непосредственно или через магистральные станции S6 (распараллеливающие один канал БС на восемь). Все модули разработаны в ИЯФ.

Обмен информацией осуществляется по двум кабелям («от ЭВМ» и «к ЭВМ»). Способ передачи информации — последовательный, порциями по 6, 12, 18 и 24 бита. В описываемой системе применяются 24-битовые (словные) посылки. Скорость обмена порядка 200 мкс на слово.

Данная система связи реализует два вида обменов: прямой доступ к ОП ПМ и программный обмен. Обмен первого вида выполняется с точки зрения ПО ПМ аппаратно взаимодействием блока связи ЦМ с интерфейсным модулем ПМ, обслуживаемым в этом случае специальной микропрограммой Одренка, с использованием резервных КАМАК-шин Р1 и Р2. ПД применяется для начальной загрузки ПМ и для обменов большими порциями данных. Обмен второго вида основан для ПМ на обычных КАМАК-функциях и ЛАМ-прерываниях ИМ и требует участия ее программного обеспечения. Используется для организации сеансов связи и для обмена небольшими порциями данных.

Функции диспетчера обмена на аппаратном уровне в обоих случаях выполняет БС ЦМ. Блок этот иницируется в режим передачи только программным образом записью в него нужного управляющего слова (определяющего форматы адресных полей и полей данных) и запуском его в режим обмена. Обмен информацией далее осуществляется выполнением ЦМ КАМАК-функций чтения/записи по ЛАМ-прерываниям БС, т.е. со стороны ЦМ обмен всегда носит программный характер. Таким образом, ЦМ является ведущей стороной во всяком обмене. Однако, по спецификациям взаимодействия ПМ↔ЦМ инициатором обмена в основном должна выступать ПМ. Для обеспечения этого интерфейсный модуль ПМ дополнительно соединяется с регистром спецпрерываний (РП) в ЦМ. Принято, что каждая подключаемая к ЦМ периферийная машина подсоединяется к РП к каналу с номером, равным логическому номеру ПМ. При необходимости сеанса связи с ЦМ периферийная система командой интерфейсного модулю высылает в РП ЦМ сигнал. Программное обеспечение ЦМ, обнаружив его, организует с проголосовавшей ПМ сеанс связи в соответствии с определенными правилами — протоколом обмена.

Применение в этом протоколе обоих описанных выше способов обмена (в принципе, достаточно обмена способом ПД) обосновывается двумя соображениями. Первое — это надежность: обмен ПД не управляем программным обеспечением ПМ, а в условиях работы больших электрофизических установок имеется вероятность

импульсных наводок на систему связи, которые могут приводить к ложным ее запускам. Программный же диалог, контролируемый на обусловленность обеими сторонами, в значительной степени «фильтрует» подобные сбои. Второе соображение связано со сложностью алгоритмов выделения периферийной системой областей памяти под ПД-обмены и определения моментов их освобождения.

Большинство обменов ПМ↔ЦМ в описываемой системе ориентировано на поддержку взаимодействия ПМ с внешними устройствами ЦМ, представляющими собой коммунальные ресурсы сети. Это и определило структуру и построение протокола: в его основу положены принципы формирования множества логических каналов доступа к ВУ ЦМ и, как расширение этого приема, каналы сформированы и под реализацию всех других взаимодействий с ЦМ. В отличие от локальных реализация этих каналов распределена между ПМ и ЦМ. Все виды сеансов связи сводятся программным обеспечением этих машин к обменам по соответствующим каналам.

Основные схемы сеансов в этом протоколе:

запрос в ЦМ, ответ из ЦМ,

(оба обмена выполняются программным способом) и

запрос в ЦМ, ПД-обмен, ответ из ЦМ.

Основной логической единицей обмена является 20-словный Блок (Блок-Команда: Б-К, Блок-Ответ: Б-О); в зависимости от ситуации один и тот же блок может выступать как в качестве Б-К, так и в качестве Б-О и именуется ниже так, или иначе, в соответствии с выполняемой нагрузкой. Там, где этой нагрузки не ощущается, используется «Блок»), передаваемый/принимаемый периферийной системой пословно программным способом.

Посылка запроса:

ПМ NAF'ом ИМ высылает в РП ЦМ сигнал;

ЦМ иницирует программу чтения из БС двадцати слов;

ПМ по ЛАМ от ИМ передает пословно 20-словную Б-К.

Замечание: если в течение приблизительно 20 секунд после посылки сигнала в РП ЦМ не иницирует процедуру считывания запроса, то ПМ повторит эту посылку, а еще через 20 секунд прекратит попытки организации сеанса связи с ЦМ с фатальной ошибкой источнику запроса.

В Б-К ОС ПМ информирует ЦМ о назначении сеанса связи и его атрибутах:

- тип запроса (загрузка-сброс программы, обмен с ФС или ТЛ, запросы даты, времени, номера директории, способов доступа к ней, ...);
- параметры запроса (счетчики, адреса, ...);
- данные (при необходимости и возможности).

Передача ответа:

Выполнив запрос, ЦМ завершает сеанс посылкой 20-словного Б-О, где содержатся все ответы о реализации функции и, при необходимости, требуемые данные. ПМ ожидает ответ и по LAM от ИМ принимает его.

Принято, что, если объем данных превышает 11 слов, то они передаются способом ПД. Информация об адресе в ОП ПМ, отведенном ОДОС под такой обмен, передается в этом случае в предваряющей Б-К. Инициировав процедуру обмена с ЦМ, содержащую фазу ПД, ОС ПМ с момента передачи в ЦМ информации об области ОП, выделенной под обмен, блокирует реорганизацию вышележащей памяти. Режим блокирования снимается по получении из ЦМ сигнала о завершении ПД (таковым считается любой последующий программный запрос с направлением «читай» для ПМ по данному каналу, либо специальная посылка ЦМ: «Конец ПД»). При этом, при необходимости, автоматически запускается процедура реорганизации ОП.

По списку, приведенному в п.2.2.2, видно, что под взаимодействие с ЦМ в ОДОС выделено 15 логических каналов связи. ОДОС допускает параллельное функционирование этих каналов: запросы на услуги, выполняемые разными каналами, передаются в ЦМ по мере их возникновения в ПМ, порядок же их исполнения центральной машиной и пересылки ответа достаточно произволен. Ограничение существует лишь в фазе пересылки текущего запроса: эта передача не может быть «разорвана» и всякая попытка передать что-нибудь для другого канала (за исключением спецкоманд) трактуется ОДОС как нарушение протокола и приводит к завершению текущего обмена с ошибкой.

Замечание: драйвер ИМ выполняет процедуры обмена с ЦМ и в случае нераспознанных запросов из ЦМ, высылая дополнительно на каждый LAM от ИМ аппаратный сигнал «конец передачи». При этом, информация принимаемая ПМ игнорируется, а при передаче в ЦМ высылается случайным образом.

Сеансы связи с ЦМ условно разделены на мониторные, специальные и вспомогательные.

Мониторные—это обращения к реальным ВУ ЦМ. Для них в ЦМ передается все контрольное поле заявки на взаимодействие с ВУ, а ЦМ, выполнив указанную процедуру, формирует в этом поле все необходимые ответы. В существующей версии таким способом реализуется обмен с носителем ФС ЦМ, ВТЛ и коммунальным печатающим устройством.

Для специальных сеансов ВУ в ЦМ не существует, ими реализуется ряд специальных функций. Это загрузка и сброс программ из/в директорий ФС ЦМ, запросы времени, даты, способов доступа к директориям, переключения директорий, ..., работа ОДОС с ВТЛ.

Вспомогательные служат для поддержки описанных выше видов сеансов, для координации действий ПМ- и ЦМ-систем при перезагрузках и перезапусках и прочих, дополнительных, взаимодействий этих систем.

Общий формат Б-К/Б-О:

- сл 01 — заголовок;
- сл 02 ÷ 19 — поле запроса/ответа, данные;
- сл 20 — КС блока.

Заголовок Блока состоит из 4 шестибитовых полей—знаков:

У.%.%.Д

У—номер Б-К—основной параметр ветвления внутри протокола. Совпадает с номером канала доступа к ВУ в соответствии с приведенным в 2.2 их списком. Если к номеру канала добавлено #40 (0-й бит слова), то передача данных будет выполняться способом ПД.

Д—детализация запроса для обменов более емких, чем однократное взаимодействие с ВУ или вообще не связанных с реальными устройствами.

Центральное поле в виде чередующихся нулей и единиц предназначается для дополнительного контроля и в рабочем протоколе не используется.

В Б-О ЦМ поле «У» ОС ПМ обычно не используется: оно должно быть перенесено системой ЦМ в регистр S5 интерфейсного модуля ПМ. В данном протоколе разряды этого 6-битового регистра имеют следующее назначение:

0	1	2 — 5
0	п ш / ч т	у

Уставленный в единицу разряд-1 означает для ПМ «пиши», т. е. передачу информации из нее.

20-е слово всегда содержит КС (контрольную сумму) Блока, представляющую собой взятую с обратным знаком сумму (с потерей переноса влево) первых 19 слов Блока.

Ниже в порядке возрастания номера «У» Б-К описываются все реализованные в ОДОС сеансы связи с ЦМ. Содержание слов в Б-К приводится под номерами 01—20. Символами «х» помечаются несущественные поля. Несущественные слова Блока опускаются.

У=1, 2, 3, 4, 5, 6. Мониторные сеансы.

Обращения к ВУ ЦМ из пользовательских программ ПМ. Контрольное поле запроса помещается в словах 02—08 Б-К. Данные (если не уставлен бит ПД) в словах 09—19. При обращении к ВУ с номером 6 (диски) слово 03 (поле слова ответа) указывает номер директории ФСЦМ, к которой направлен запрос.

ЦМ, выполнив заказанную операцию с ВУ, и, переслав/приняв при необходимости данные по прямому доступу, посылает Б-О, в котором, в соответствии с вышеприведенными правилами, формирует все требуемые ответы. Ответ для отсутствующих в конфигурации ЦМ внешних устройств должен также соответствовать заявляемой в запросе операции с ним.

Замечание: по внутренним причинам обращение к ФС ОС ПМ выполняет через 6-й канал, не корректируя при этом тип устройства в контрольном поле заявки обмена с ним в Б-К, где эти обращения адресованы к ВУ типа 9.

Пример. Заявка приема двух символов с ВТЛ: ПМ посылает Б-К:

сл 01	: 1.х.х.х
сл 02	: 1/16
сл 03	: хххх
сл 04	: 0002
сл 5—19	: хххх
сл 20	: КС

ЦМ принимает со своего терминала латинское «А» и высылает следующий ответ:

сл 01	: 1.х.х.х
сл 02	: 1/16
сл 03	: 0
сл 04	: 0002

сл 5—8	: хххх
сл 09	: #74410000
сл 10—19	: хххх
сл 20	: КС

У=7. Специальный канал для пользовательских программ.

Предназначен для расширения возможных видов сеансов связи ПМ и ЦМ, не предусмотренных ОС ПМ и реализуемых в ПМ обычной пользовательской программой (программное обеспечение ЦМ более мобильно, нежели ПМ). В системе управления ВЭПП-4 применялся для обмена директивами, прямого (без посредничества программы-монитора в ЦМ) межпрограммного обмена и других целей.

ОДОС проверяет для него лишь общие правила оформления, не вмешиваясь без необходимости во внутреннее содержимое Б-К. Кроме этого, она позволяет программам неограниченно долго держать заявку на чтение по данному каналу (без предварительного информирования об этом ЦМ), что позволяет при необходимости передавать в ПМ информацию по инициативе ЦМ. Как указывалось при описании экстракода 157 для ВУ типа 7, ОДОС добавляет адрес начала программы к адресу в слове 05 Б-К при наличии в операции заявки на обмен в прямом доступе, считая это слово адресом 1-го слова буфера, отводимого программой под ПД.

У=8, 9, 10, 11, 12. Специальные сеансы систем ПМ и ЦМ.

А. У=8. Сеансы FI/DU.

А1. Протокол FI—загрузка программы из ФС ЦМ. Внутри него Д=0, 1, 2, 3, 63

Запрос на загрузку программы :

сл 01	: 8.%.%.2 (#10252502),
сл 02	: имя программы,
сл 03—04	: имя директории (8 знаков),
сл 11	: номер директории.

Номер директории предназначен для использования при умолчании в директиве—источнике запроса имени директории (сл 03=0000 или 4 пробела).

Может быть 2 ответа на этот запрос:

1. Б-О: определитель программы:

сл 01	: 8.%.%.0,
-------	------------

сл 02 : имя программы,
сл 04 : обменный размер программы (слов),
сл 05 — 12 : блок запросов программы.

Несовпадение имени программы в слове 02 Б-О с запрошенным рассматривается ОС ПМ как отсутствие такой программы в ФС ЦМ. По другим причинам отказа предусмотрен:

2. Блок: отказ

сл 01 : 0.%.%. — (Д = #77),
сл 02 : 0.0.0.N — код причины отказа.

Код причины отказа должен представлять собой некоторое число N, которое подставляется ОС ПМ в диагностическое сообщение:

'ОТКАЗ N ЦМ'

(в существующих версиях ОДОС в сеансе FI не используется)

Получив в качестве ответа определитель программы, ПМ может далее развивать диалог также двумя путями:

1. Если атрибуты программы позволяют разместить ее в ОП ПМ, то посылается Б-К: «адрес ПД»:

сл 01 : 8.%.%.3,
сл 05 : абсолютный адрес начала программы.

2. Если же по текущему состоянию размещение программы невозможно, то ПМ высылает Блок «отказ» с несущественными полями 02 — 19:

сл 01 : 8.%.%. —

Переслав программу в ОП ПМ по прямому доступу, ЦМ высылает Б-О: «контрольная сумма программы»:

сл 01 : 0.%.%.1,
сл 02 : КС обменной части программы.

КС обменной части программы представляет собой взятую с обратным знаком сумму всех передаваемых слов программы (с потерей переносов влево).

A2. DU — сброс программы в ФС ЦМ. Д = 1, 4, 63

Сеанс начинается с высылки системой ПМ Б-К: «запрос на сброс программы»:

сл 01 : 8.%.%.4,
сл 02 — 09 : Блок запросов программы,
сл 11 : номер директории, куда следует поместить программу,

сл 19 : КС программы.

Слово 05 Б-К при этом содержит абсолютный адрес начала программы в ОП ПМ (адрес ПД для ЦМ). Для ЦМ в этом блоке содержится полная информация о назначении и атрибутах сеанса. Если она выполняет запрос, то должна выслать Б-О: «контрольная сумма программы», оформленный в соответствии с приведенным в А1 описанием. При невозможности выполнения процедуры ЦМ должна послать Блок «отказ», со следующими значениями кода отказа N:

1 — запись запрещена,
2 — нет места в директории,
3 — несовпадение КС (сбой обмена),
4 — сбой дисков ЦМ.

Для N вне этого диапазона диагнозом ОДОС на невыполнение операции будет сообщение:

ОТКАЗ N ЦМ

Б. У = 9. ДАЙ/НА — межпрограммный обмен с ЦМ. ПМ высылает в ЦМ Б-К:

сл 01 : 9.%.%.0,
сл 02 : 0/АДРС,
сл 03 : имя программы-абонента,
сл 04 : счетчик слов для обмена,
сл 05 : Б0.АДРО,
сл 09 — 19 : данные,

где АДРС — смещение относительно 1-го слова обменного буфера программы-абонента, имя которой указано в слове 03;

Б0 = 0 — передача информации от абонента;

Б0 = 1 — передача информации абоненту;

АДРО — 16-битовый адрес обменного буфера в программе — источнике запроса (для ПД).

Данные передаются в этой же Б-К при Б0 = 1 и счетчик ≤ 11 .

Выполнив при необходимости обмен по ПД, ЦМ высылает в ПМ Б-О:

сл 01 : 9.%.%.0,
сл 03 : имя программы,
сл 09 — 19 : данные,

где по правилам межпрограммного обмена в системе ОДОС нулевое значение слова 03 (имя программы) означает положительный ответ на процедуру обмена. Условия на способ передачи данных в

ПМ те же, что и для передачи данных в ЦМ.

В. У=10. GIVE — спецзапросы. ПМ высылает в ЦМ Б-К:

сл 01 : 10.%.%.0,
сл 02 : 0.0.0.Н — номер запроса.

В1. Н=0 — запрос даты в числовом виде. Ответ ЦМ:

сл 01 : 10.%.%.0,
сл 02 : дата в числовом виде.

В2. Н=1, 2 — запрос даты/времени в символьном виде. Ответ ЦМ:

сл 01 : 10.%.%.0,
сл 02—03 : дата/время в символьном виде,

В3. Н=3 — запрос номера директории, к которой прикрепена данная ПМ. Ответ ЦМ:

сл 01 : 10.%.%.0,
сл 02 : номер директории.

В4. Н=4 — запрос разрешенных способов доступа к директории. Номер директории передается в слове 03. Ответ ЦМ:

сл 01 : 10.%.%.0,
сл 02 : Б0./номер директории,
сл 03 : способ доступа,
сл 04—05 : имя директории.

При этом, если в слове 02 уставлен нулевой бит (Б0), то директория отсутствует, а слово 03=0 означает доступ только для чтения.

В5. Н=5 — запрос специнформации. Ответ ЦМ:

сл 01 : 10.%.%.0,
сл 02 : Количество знаков в «специнформации»,
сл 03—19 : специнформация в символьном виде.

Специнформация (до 44 знаков) будет выведена на терминал ввода директивы РО, инициирующей этот запрос.

Г. У=11. Прием директивы с ВТЛ для ОС ПМ.

Привилегированная операция с ВТЛ. ОС ПМ перед этим сбросила при необходимости инициированный ранее прием символов и прикрепила ВТЛ к себе. Символ с ВТЛ запрашивается с помощью Б-К:

сл 01 : 11.%.%.0,

ЦМ пересылает принятый со своего терминала символ (выведя

эхо на него) в Б-О:

сл 01 : 11.%.%.0,
сл 09 : P.C.x.x,

Д. У=12. Вывод сообщения ОС ПМ на ВТЛ.

Привилегированная операция с ВТЛ.

сл 01 : 12.%.%.0,
сл 02 : количество знаков в сообщении,
сл 03—19 : сообщение.

После вывода на свой терминал ЦМ высылает ответ:

сл 01 : 12.%.%.0

У=13, 14, 15. Вспомогательные сеансы.

Е1. У=13. Недействующий канал.

Е2. У=14. Канал спецсигналов ЦМ.

По этому каналу ЦМ может переслать в ПМ некоторые спецсигналы по своей инициативе. ОДОС различает следующие сигналы по этому каналу (Б-К состоит из одного слова!):

!!!! (#21212121) — рестарт ЦМ. По этому сигналу ОДОС сбрасывает все инициированные до этого обмена с ЦМ;

JJJJ (#52525252) — запрос ОС с ВТЛ. В ответ на него ОДОС посылает на ВТЛ приглашение к приему директивы (канал 12) и осуществляет прием символов (канал 11);

PPPN (#606060NP) — программная имитация СПР NP (1—24). ОДОС обрабатывает этот сигнал как появление сигнала с номером NP с РП.

Е3. У=15. «Конец ПД». Однословная посылка из ЦМ, имеющая вид:

О.%.%.К

где К — номер канала, для которого завершена операция ПД, О — ответ о результате операции, добавляемый ОДОС к слову ответа КП источника обмена. Может применяться там, где не предусмотрено более содержательного ответа для завершения передачи.

Приложение 2

ОПИСАНИЕ КОНФИГУРАЦИИ СТАНЦИИ

Для правильного функционирования в ОДОС перед ее запуском должны быть вписаны 60 слов (начиная с 45 адреса), кото-

рые содержат всю необходимую настроечную информацию для данной станции (конфигурацию станции). Назначение этих слов следующее:

общая информация:

- сл 1 — 2 имя системы в символьном виде,
- сл 03 логический номер станции (число: 0 — 63),
- сл 04 кол-во позиций в крейте,
- сл 05 размер ОП Одренка (в ед. К : 16 — 64);

описание крейта:

- сл 06 имя модуля, находящегося в 1-й позиции,
- сл 07 его свойства,
- ...
- сл 50 имя модуля, находящегося в 23-й позиции,
- сл 51 его свойства;

дополнительная информация:

- сл 52 две 2-буквенные мнемоники вызовов программ.
- сл 53 имя программы, вызываемой по первой мнемонике,
- сл 54 ...
- сл 55 кол-во программ, загружаемых вместе с ОС,
- сл 56 — 60 имена этих программ.

«Имя системы» употребляется, как уже упоминалось, в символическом приглашении к набору директивы на ТЛ.

Логический номер станции предназначен для начальной настройки и для сетевых целей. Нулевые номера присваиваются станциям, работающим вне сети или как центральные.

Указание кол-ва позиций в крейте предназначено для использования нестандартных, «усеченных», крейтов.

Описание мнемоник вызова позволяет упростить процедуру вызова часто используемых на данном месте программ. Например, описание:

- сл 52 AABV
- сл 53 AAAA
- сл 54 BBVV,

связывает с мнемониками AA и BV директивы «LG AAAA 20» и «LG BBVV 20».

Загрузка до 5 программ, выполняемая системой по приводимому списку сразу после старта, имеет своей целью упростить проце-

дуру перезагрузки Одрят в системах управления при использовании на них жесткого набора программ.

По описаниям «системных» модулей ОДОС формирует адреса для каналов ВУ и КАМАК-команд взаимодействия с ними. Описания прочих модулей может быть целесообразным при создании переносимых программ, использующих специальную периферию.

Зарезервированные имена модулей (системные модули) и их свойства:

1. AI00 — индикатор активности

*20000000 — его свойства

(установленный 1-й бит слова свойств служит дополнительным признаком «системности» модуля; его указание необязательно).

Планировался для формирования светового табло, индицирующего работоспособность конкретной станции.

При каждой обработке прерывания по времени ОДОС «подпитывает» этот индикатор, выполняя команду F16A0.

2. DC00 — ЭД (ЗУ) — электронный диск

*20ttkkkk

kkkk — полный размер памяти ЭД в ед. К (до 512); t. — восьмеричная цифра — тип модуля ЭД (0 — 128 К, 1 — 256 К).

ОДОС разрешает сдваивание модулей ЭД, причем «расширителем» является не описываемый явно модуль справа (вплотную) от основного. Допустимы следующие комбинации модулей (если значение 'kkkk' больше 128):

00 — первый 128 К, второй kkkk-128, при $128 < kkkk \leq 256$

10 — 256 К, kkkk-256, при $256 \leq kkkk < 512$

11 — 256 К, 256, kkkk = 512

3. ER00

20000000 и

4. S600

20000000

Модуль Б0633 с описанием обеих позиций в указанном порядке. Служит для подключения дополнительных крейтов через контроллер К0601, обеспечения спецрежимов работы ведомого Одренка через интерфейс У0605 и для подключения аппаратуры в стандарте ВИШНЯ.

5. OD00 — 1М интерфейсный модуль У0605 (ИМ).

20000000

Модуль связи с ЦМ. При настройке ОДОС тестирует указан-

ную позицию командой F26A0. Отсутствие ответов X или Q может послужить причиной отказа ее от дальнейшей работы (система зацикливается на указанной команде).

6. RP00 — 1М модули P0602 или P0606 — регистры спецпрерываний.

#20000000

ОДОС сама разбирается с типом модуля по их внутренним функциям.

7. TE00 — 1М модули B1102, B0401 — последовательный интерфейс V-24.

#20klmnp00

Предназначен для подключения ТЛ, DZM и других устройств с указанным интерфейсом. Имеет 4 канала, нумеруемых 1, 2, 3, 4 снизу-вверх;

k, l, m, n — номер типа терминала, подключаемого к каналам 1, 2, 3, 4, со значениями:

0 — отсутствие ТЛ (или канал DZM);

1 — 2 стоп-бита, 8 бит, без паритета, 9.6 Кбод;

2 — 2 стоп-бита, 8 бит, паритет чет, 9.6 Кбод.

Если тип терминала указывается в форме '4+t', то на него блокируется вывод символов CTRL(N,O), используемых для переключений алфавитов знакогенераторов дисплейных устройств.

Приложение 3

СООБЩЕНИЯ СИСТЕМЫ

Большинство сообщений ОДОС представляют собой ответы на исполнение директив. Порядковые номера сообщений используются в качестве ответов в программы о результате исполнения их цепочек директив при применении программами формы заказа директив с ответом.

0. ОК

1. #NAME В ПАМЯТИ

2. НЕТ ПАМЯТИ

3. В ПАМЯТИ МАКС. ПРОГРАММ

4. СБОЙ В ОБМЕНЕ С ЦМ

5. НЕТ В ЦМ

6. НЕТ # NAME

7. ОБМЕН С ЦМ БЛОКИРОВАН

8. КАНАЛ СВЯЗИ С ЦМ ЗАНЯТ
9. ОШИБКА В ИНФ. ИЗ ЦМ
10. ПРЕКРАЩЕНИЕ СВЯЗИ С ЦМ ПО ВРЕМЕНИ
11. ОТКАЗ N ЦМ
12. ПРОГРАММА В ЗАГРУЗКЕ
13. ПРОГРАММА НЕДЕЛЕТУЕМА
14. Ответ по директиве OU
15. Ответ по директиве PR
16. ОШИБКА В ДИРЕКТИВЕ
17. ILLEGAL...
18. HALTED:~..., DISPLAY:~...
19. СБОЙ ЗУ
20. НЕТ ЗУ
21. ПРОГРАММА РАБОТАЕТ С УСТРОЙСТВОМ
22. НЕТ МЕСТА В ЗУ
23. НЕТ В ЗУ
24. Ответ по директиве PO
25. Ответ по директиве WS
26. Ответ на директиву CS
27. НЕТ МЕСТА В ДУ
28. Сообщение на ВТЛ о старте системы в ПМ
29. СБОЙ СВЯЗИ С ЦМ
30. Ответ на директиву DY ND
31. НЕТ ТАКОЙ ДУ
32. СБОЙ В ЦМ

Замечание. Сообщение под номером 7 связано с директивой блокировки канала связи с ЦМ, отсутствующей в последних версиях ОДОС.

Приложение 4

БЛОК ЗАПРОСОВ В ФАЙЛЕ-ПРОГРАММЕ

- сл 01 : NAME — имя программы,
сл 02 : P.R.x.x — приоритет (2 левых знака),
сл 03 : L/x — размер(L) блоков по 256 слов,
сл 04 : xxxx,
сл 05 : xxxx,
сл 06 : xxxx,
сл 07 : BS.x.x.x — статусы программы,
сл 08 : PS — состояние программы

Здесь x — несущественные поля, точками разделены знаковые позиции слова, размер программы L помещен на 9 левых битах слова (счетчиковая часть слова), из статусного знака BS используется лишь самый левый бит, означающий неуничтожаемость программы директивами с $TЛ$. Слово состояния программы PS используется ОДОС как начальное состояние программы (ССП) после загрузки ($PS = \#40000000$ означает состояние останова (SU)).

ЛИТЕРАТУРА

1. Белов С.Д. и др. Структура системы автоматизированного управления и контроля накопителя ВЭПП-4. Труды пятого Всесоюзного совещания по ускорителям заряженных частиц, т. II, М.: Наука, 1975.
2. Пискунов Г.С., Тарарышкин С.В. Двадцатичетырехразрядная ЭВМ в стандарте КАМАК. Автометрия, 1986, № 4, с.32—38.
3. Алешаев А.Н. и др. Построение распределенных систем управления крупными электрофизическими установками на базе сетей специализированных микро-ЭВМ в ИЯФ СО АН СССР и их программное обеспечение. — Автометрия, 1986, № 4, с.39—45.
4. Алешаев А.Н., Козак В.Р. Программное обеспечение для микроЭВМ ОДРЕНОК. Центральная ЭВМ. — Препринт ИЯФ СО АН СССР 88-48. Новосибирск, 1988.
5. Алешаев А.Н. и др. Операционная система ЭВМ ОДРА для управления электрофизическими установками в ИЯФ СО АН СССР. — Препринт ИЯФ СО АН СССР 80-194. Новосибирск, 1980.

Список сокращений

БС — блок связи;
ВС — вычислительная система;
ВТЛ — виртуальный терминал;
ВУ — внешнее устройство;
ДПС — директивный процессор системы;
ИМ — интерфейсный модуль;
ЛТЛ — локальный терминал;
 $N(M)$ — модифицированная адресная часть команды;
ОП — оперативная память;
ОС — операционная система;
ПД — прямой доступ;
ПНФ — программный номер файла;
ПО — программное обеспечение;
ПП — подпрограмма;
РБА — регистр базы адреса;
РОН — регистры общего назначения;
РП — регистр спецпрерываний;
сл — слово;
СПР — спецпрерывание;
ССП — слово состояния программы;
ТЛ — терминал;
ЦМ — центральная машина;
ФОС — файл описания станций;
ФС — файловая система;
ФЦ — функция;
ЭД — электронный диск (ЗУ);
X — аккумулятор (один из РОН 0-7);

Содержание

1. Введение	3
2. Спецификации ОС ОДОС	7
2.1 Управление памятью и мультипрограммирование	9
2.2 Взаимодействие с внешними устройствами	12
2.2.1 Терминалы	15
2.2.2 Каналы связи с ЦМ.	18
2.2.3 Файловая система.	19
2.3 Средства синхронизации	21
2.4 Межпрограммный обмен	23
2.5 Спецрежимы.	23
3. Внутренняя организация и функционирование ОДОС.	24
4. Загрузка и настройка ОДОС.	29
5. Директивы системы.	30
6. Экстракоды	47
7. Запрещенные команды	64
Приложение 1. Протоколы обмена с ЦМ.	65
Приложение 2. Описание конфигурации станции	75
Приложение 3. Сообщения системы	78
Приложение 4. Блок запросов в файле-программе	79
8. Литература	80
9. Список сокращений	81

А.Н. Алешаев

**Программное обеспечение
для микроЭВМ ОДРЕНОК.
Операционная система ОДОС**

Ответственный за выпуск С.Г.Попов

Работа поступила 26.04. 1989 г.
Подписано в печать 18.05. 1989 г. МН 02159
Формат бумаги 60×90 1/16 Объем 2,6 печ.л., 2,1 уч.-изд.л.
Тираж 290 экз. Бесплатно. Заказ № 67

*Набрано в автоматизированной системе на базе фото-
наборного автомата ФА1000 и ЭВМ «Электроника» и
отпечатано на ротапинтере Института ядерной физики
СО АН СССР,
Новосибирск, 630090, пр. академика Лаврентьева, 11.*