

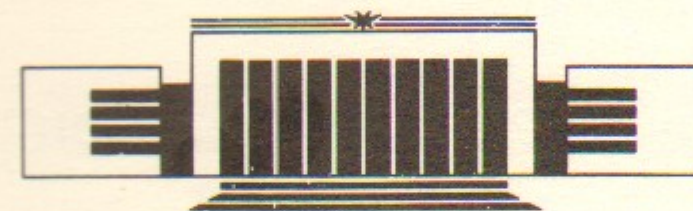


12  
ИНСТИТУТ ЯДЕРНОЙ ФИЗИКИ СО АН СССР

Р.Х. Гутче

НЕОДНОРОДНАЯ ЛОКАЛЬНАЯ СЕТЬ  
МИКРО-ЭВМ CMNet.  
ОПИСАНИЕ ДЛЯ ПОЛЬЗОВАТЕЛЕЙ  
СИСТЕМЫ RT-11

ПРЕПРИНТ 88-117



НОВОСИБИРСК



Неоднородная локальная сеть  
микро-ЭВМ CMNet.  
Описание для пользователей системы RT-11

*Р.Х. Гутче*

Институт ядерной физики  
630090, Новосибирск 90, СССР

**АННОТАЦИЯ**

В работе описана реализация неоднородной локальной сети CMNet для машин типа PDP-11 под управлением операционной системы RT-11. Главное достоинство сети — реализация виртуальных дисков в общем виде, включая системное устройство. Сеть может использоваться во всех местах, где стоит задача связи машин с помощью простых связных интерфейсов.

© Институт ядерной физики СО АН СССР

**ВВЕДЕНИЕ**

Вычислительные сети играют большую роль в развитии вычислительной техники и способствуют широкому внедрению ЭВМ во все сферы народного хозяйства и научных исследований. Однако выбор разработок по сетевому программному обеспечению очень невелик, особенно при использовании разнородной аппаратуры связи, разных операционных систем или ЭВМ нескольких типов. Поэтому в Институте ядерной физики СО АН СССР была разработана неоднородная локальная сеть микро-ЭВМ CMNet (Си-Эм-Нэт), которая не накладывает ограничений на тип машин или программного обеспечения и открыта для дальнейших расширений.

Локальная сеть CMNet предназначена для решения целого ряда проблем, возникающих при эксплуатации микро-ЭВМ. Перечислим наиболее острые из них, ориентируясь на машины типа PDP-11 (Электроника-60, ДВК, МС1212) и операционную систему RT-11 (РАФОС).

Первая, и на сегодняшний день, наверно, главная проблема заключается в отсутствии устройств внешней памяти нужного объема и в нужном количестве. Даже традиционные диски не решают все вопросы организации внешней памяти. Дело в том, что фактически любой доступный тип устройств внешней памяти содержит движущиеся детали и магнитные поверхности. В рабочих помещениях, тем более на физических установках, всегда имеются вибрации, перепады температуры, загрязненность воздуха. Все



эти параметры отрицательно влияют на работоспособность дисководов. В какой-то мере проблему решают «винчестерские» диски, однако они пока весьма дефицитны. Поэтому удаленные от ЭВМ диски, установленные в специально оборудованных помещениях, имеют право на существование до тех пор, пока не будут решены все вышеуказанные вопросы. Следует отметить, что диски с общим доступом решают еще ряд важных проблем. Коллективы, работающие над одной общей темой, часто обмениваются информацией между собой. Оперативность этого обмена открывает возможность только что полученные результаты одного члена коллектива сразу же использовать всем остальным. Удобно решается проблема общих баз данных и библиотек программ. Нет необходимости в постоянном слежении за всеми изменениями в программном обеспечении, если реализован простой доступ к единственному экземпляру общего ресурса в пределах всей сети. В сети CMNet доступ к удаленным дискам реализован в виде виртуальных дисков. Для этого машины, имеющие собственные диски, предоставляют их в качестве удаленных дисков для других машин. Драйвер виртуального диска предоставляет периферийной машине весь сервис, который нужен операционной системе. При этом считается необходимым реализовать загрузку операционной системы с виртуального диска.

Вторая проблема тесно связана с решением первой и касается быстрого действия дискового обмена. Хотя операционная система RT-11 использует системный диск не очень часто, замедление из-за удаленности диска чувствительно. Поэтому целесообразно использовать полупроводниковую память либо в виде электронного диска, либо в качестве кэш-памяти для удаленного диска. Оба варианта реализованы в сети CMNet. Следует подчеркнуть, что разница между физическими, виртуальными (удаленными) и электронными дисками для операционной системы заключается лишь в их быстродействии. Соответствующие драйверы обеспечивают полную совместимость этих устройств между собой. На более быстродействующем электронном диске хранится рабочая копия операционной системы обычно лишь потому, что объем обмена с рабочим (пользовательским) диском почти всегда намного меньше по сравнению с системным диском, к тому же существует опасность потери информации при сбоях питания электронного диска.

Кроме использования удаленных дисковых ресурсов сеть CMNet позволяет подключать фактически любой тип устройств. Разрабатывается доступ к удаленным устройствам печати, обеспе-

чивается режим виртуального терминала для входа в другие операционные системы.

Третья проблема касается взаимодействия процессов в сети. Можно различать использование вычислительной мощности удаленных машин (удаленный вызов процедур) и взаимодействие параллельных процессов. В первом случае одна задача обращается к другой машине для выполнения какой-то функции или подпрограммы. Управление напоминает схему ведущего-ведомого, хотя вызывающий может продолжать свою работу параллельно с выполнением удаленной процедуры. Можно себе представить такую организацию взаимодействия между машинами не только с целью использования мощности другого процессора, а также для иерархического управления какой-то установкой. Второй случай более сложный. Здесь работает параллельно несколько процессов, обменивающиеся информацией между собой. Каждый процесс сам себе хозяин, он работает на базе локальной информации, когда отсутствует другая информация, учитывает информацию от других машин, а также выдает свои результаты в сеть. Реализация такой схемы управления очень сложна, но обеспечивает высокую живучесть системы. Транспортная служба сети CMNet предоставляет свой сервис для обоих вариантов взаимодействия процессов. Конкретное использование этих возможностей зависит от разработчика распределенной системы управления.

Так как сеть CMNet может включать довольно большое количество машин, большое значение приобретает информационная служба сети. Сюда относится оперативная информация, такая, как сообщения на терминал пользователя, электронная почта или объявления. Можно получить информацию о текущем состоянии сети, состоянии отдельных узлов, предоставляемых услугах. При этом обеспечивается защита информации пользователей от несанкционированного доступа с помощью паролей и проверки номеров машин.

Сеть CMNet не ориентирована на конкретный тип связанного интерфейса. Можно одновременно использовать разные интерфейсы, создавая неоднородную конфигурацию. В настоящий момент поддерживаются стандартный интерфейс DL-11, работающий по протоколу V.24, а также словный DL-11, известный под названием DL-KC. Кроме этого имеется обеспечение для разработок Института ядерной физики: система ДОЗА — байтовый обмен между одиночными интерфейсами и мультиплексором на 48 линий; и SB-11 — связной интерфейс со встроенным буфером размером



1 Кбайт. Готовится разработка интерфейса с общей передающей средой (моноканалом). Топология связи между узлами сети произвольная, что обеспечивает большую гибкость при подключении новых машин и высокую живучесть сети CMNet в целом.

После знакомства с решаемыми в сети CMNet задачами можно перейти к рассмотрению принципов реализации тех или иных функций.

### ОСНОВНЫЕ ПРИНЦИПЫ

Любая машина, включенная в сеть, является узлом сети. Совокупность двух связанных интерфейсов с соответствующим кабелем между ними принято считать линией связи или просто связью. У каждого узла есть по меньшей мере один сосед, с которым он связан непосредственно через линию связи. Линии связи логически разделяются на основные и дополнительные. Основные связи образуют математическое дерево узлов. Они являются базовыми во всех отношениях, особенно при нумерации узлов и маршрутизации пакетов. Дополнительные связи используются для повышения эффективности обмена между любыми узлами сети, где это необходимо. Часто они отсутствуют совсем.

Каждый узел имеет свой уникальный номер в сети. Кроме этого, он может иметь имя, состоящее не более чем из восьми букв и цифр. Обращаться к узлу можно и по номеру, и по имени. Существует три зарезервированных слова HOST, ALL и DOWN, которые обозначают специальные формы доставки информации. HOST—это обращение к узлу, расположенному ближе к центру. Если этот узел не может удовлетворить запрос, то он обращается к еще более близкому к центру узлу и т. д. Групповая доставка информации может охватывать все узлы (ALL) или только те, которые дальше от центра (DOWN). Специальные формы доставки учитывают только основные связи между узлами. Нумерация узлов связана с топологией основных связей между ними, при чем номер тем меньше, чем ближе узел к центру сети. Поэтому абсолютный центр сети CMNet имеет обычно номер 1.

Передача информации осуществляется небольшими порциями, называемыми пакетами. Пакет сети CMNet состоит не более чем из 600 байтов, которые включают в себя заголовок пакета, собственно подлежащую передаче информацию и контрольную сумму.

Пакеты передаются от узла к узлу по линиям связи. Скрытый от пользователя алгоритм маршрутизации выбирает в каждом узле линию, по которой пакет передается дальше, пока не достигнута цель или пока не обнаружено отсутствие нужного узла.

Для связи пользователя сети с имеющимся ресурсом или объектом используется тройка номеров: номер узла пользователя, номер узла местонахождения объекта и некоторый номер объекта. Совокупность этих трех номеров образует канал пользователя. Канал считается открытым, когда установлено соответствие между тремя указанными числами и реальным объектом, и считается закрытым, когда такого соответствия нет. Объектом может быть любой ресурс, который может использоваться из другого узла сети. Самыми распространенными объектами являются обычно диски и виртуальные терминалы. Для идентификации объекта его номер разделяется на две части. Первая часть задает класс или природу объекта, вторая часть содержит произвольный номер внутри класса. Операционная система использует для связи с устройствами драйверы. Под именем устройства понимается при этом логическое или физическое имя соответствующего драйвера. Так как внутри драйвера сети установлено соответствие между восемью устройствами и соответствующими дисковыми каналами, то эти понятия можно рассматривать как синонимы.

Типичный сеанс работы начинается с открытия канала, т. е. с установления права на доступ к некоторому объекту в сети. Объем этих прав определяется при открытии канала. Это может быть только чтение информации, или чтение и запись, право на закрытие канала, уничтожение объекта и т. д. Защита от несанкционированного доступа обеспечивается с помощью паролей. Канал открывается только тогда, когда пароль совпадает с эталоном пароля, который хранится в узле местонахождения объекта. Паролем может быть любая цепочка печатных знаков длиной не более восьми. Отсутствие пароля, т. е. невозможность совпадения пароля с эталоном, достигается с помощью непечатного символа в эталоне. Пустой эталон (нулевая длина) считается всегда совпадающим с любым паролем. Разные формы доступа к объекту могут иметь разные пароли. Для дисков, например, существует кроме общего пароля, защищающего информацию от несанкционированной модификации, еще один пароль, действующий только на неразрушающий доступ (чтение). Вторым паролем обычно пустой, если только информация не секретна.

Последнее замечание касается особой формы доступа времен-



ной записи, которая используется в основном для системного диска. При открытии канала создается специальный объект (при закрытии он уничтожается), который изначально совпадает с указанным в запросе базовым объектом или диском. На этот созданный объект разрешена запись до 127 различных блоков. Базовый диск при этом не модифицируется, поэтому можно при открытии канала использовать пароль для чтения. Эта специальная форма доступа открывает возможность использования многими одновременно работающими машинами общего системного диска, что существенно сокращает абсолютный объем требуемого дискового пространства.

Многие детали нового программного обеспечения осваиваются обычно по ходу реальной работы, когда на базе практически освоенного материала возрастает понимание тех принципов, которые лежат в их основе. Поэтому целесообразно перейти к описанию запуска сети CMNet на типичной структуре машин и линий связи, а после этого изучить подробное описание отдельных компонент сети.

#### ЗАПУСК СЕТИ

Для запуска локальной сети CMNet нужна одна (центральная) машина, которая работает под управлением операционной системы RT-11 версии 5 и имеет собственные диски. Она должна быть связана подходящими интерфейсами с другими (периферийными) машинами, которые могут не иметь дисков. Адреса регистров и векторов связанных интерфейсов должны быть одинаковы для всех периферийных машин, в центральной машине они могут быть произвольными. В центральной машине наличие сетевого таймера обязательно, в периферийных машинах весьма желательно.

Для большей определенности допустим, что в качестве связанных интерфейсов используются DL-KC с адресом 177000 и вектором 270 в периферийных машинах. В центральной машине они имеют адреса и векторы прерывания дополнительных терминалов, а именно 176500/300, 176510/310 и т. д. Пусть центральная машина располагает дисками RK:, причем RK0: содержит операционную систему, а RK1: может выделяться в качестве рабочего диска для периферийных машин.

Запуск сети CMNet осуществляется в три этапа. Первый явля-

ется подготовительным и выполняется только один раз. Вторым этапом выполняется каждый раз после запуска операционной системы центральной машины и поэтому может быть заключен в командный файл. Третий этап состоит в загрузке периферийных машин.

#### Подготовка

Прежде всего нужно скопировать файлы сети CMNet на системный диск центральной машины. Это файлы CMNET.BOT, CMNET.DAT, CMN.REL, CMINI.REL, CNC.SAV, CMNDAT.SAV, CMNPAS.SAV. Кроме этого, нужны драйверы CN.SYS, CK.SYS и BK.SYS. Последние два работают на DL-KC, другие интерфейсы требуют другую пару драйверов. Если имеющиеся драйверы не подходят (другой адрес или другая система), то их нужно заново оттранслировать и собрать, например

```
MAC SYSGEN.CND + SRC:CK/OBJ  
LIN CK/EXE:CK.SYG/MAP
```

Эта процедура не отличается от генерации любого другого драйвера. Адреса регистров и векторов прерывания можно определить в файле SYSGEN.CND стандартным образом, в нашем примере CK\$CSR и CK\$VEC. Инсталлируются в центральной машине драйверы CN: и BK:.

Далее задается конфигурация узла с помощью программы CMNDAT, которая модифицирует файл CMNET.DAT. В нашем примере можно ввести команды

```
CMNDAT CMNET/N:1  
CMNDAT CMNET/U:RK0/R/W/T/P  
CMNDAT CMNET/U:RK1/R/W/P
```

Первая устанавливает номер узла, вторая и третья команды разрешают доступ к дискам RK0: и RK1: с проверкой паролей. На нулевой (системный) диск кроме этого разрешена временная запись. Для осуществления временной записи нужен файл CMNET.TMP, который нужно создать мониторной командой CREATE. Размер файла зависит от потребностей пользователей, обычно используются не более 50 блоков на каждую периферийную машину. Файл может создаваться на любом из включенных в конфигурацию дисков для записи, обычно для этой цели используется системный диск. Диск для пользователей нужно разделить на



логические диски, т. е. на файлы с расширением .DSK. Для их создания также используется мониторная команда CREATE. При желании эти логические диски можно также проинициализировать, используя драйвер логических дисков LD:.

Последние действия подготовительного этапа заключаются в определении паролей. Необходимо задать пустой пароль на чтение системного диска

```
CMNPAS RK0:/W/C  
<пустая строка>
```

а также пароли для всех логических дисков. Третий пароль (Change) пока не используется. Не следует забывать, что отсутствие пароля запрещает доступ, пустой пароль открывает его полностью. Задача CMNPAS создает и расширяет при необходимости файл CMNET.SYS, который содержит все пароли соответствующего диска.

На этом подготовительный этап заканчивается и можно перейти к запуску сети в центральном узле.

### Запуск центра

Работающий центр сети CMNet базируется на центральном драйвере узла CN:, драйвере линий BK: и задаче обслуживания запросов CMN, находящихся в оперативной памяти. Задача CMN использует дисковые файлы CMNET.DAT, CMNET.BOT и CMNET.TMP, кроме этого она нуждается в наличии в памяти USR и включенных в конфигурацию драйверов. Примерный командный файл может содержать следующие команды:

SET USR NOSWAP	фиксация USR
LOAD CN:	загрузка драйверов CN: и BK:
LOAD BK:	
R CN.SYS	запуск программы настройки драйвера CN:
/N:1	задание номера узла
/X	выход из программы настройки
R BK.SYS	запуск программы настройки драйвера BK:
/M	присоединение драйвера к CN:
/I:0/C:176500/V:300	включение интерфейсов в сеть
/I:1/C:176510/V:310	
/I:2/C:176520/V:320	
/X	выход из программы настройки
ASSIGN RK0: SYS:	системное устройство периферийных машин

ASSIGN RK0: BOT:	устройство для загрузчика CMNET.BOT
ASSIGN RK0: DAT:	устройство для файла CMNET.DAT
ASSIGN RK0: TMP:	устройство для файла CMNET.TMP
FRUN CMN/BUFF:2000	запуск программы обслуживания запросов

Последняя команда запускает задачу обслуживания запросов CMN с указанием размера буфера. Буфер можно в широком диапазоне увеличивать или уменьшать, его размер влияет на время реакции сети. При недостаточном размере буфера задача CMN не запускается.

### Запуск периферийных машин

Когда центр сети CMNet запущен, можно приступить к загрузке периферийных машин. Для этого запускается аппаратный загрузчик пультового режима процессора 177000L или команда LX загрузчика HDBOOT. Можно также вручную ввести в память известный «DEC bootstrap loader» и запустить его. В любом случае осуществляется выгрузка сетевого загрузчика по перфоленточному формату через интерфейс с указанным адресом. Так как большинство интерфейсов, в том числе DL-KC, не поддерживают сигнал запуска считывателя, то нужно дождаться тайм-аута не более 16 секунд. Потом печатается на терминале

```
CMNet bootstrap  
—приветствие системы—  
CMNet>
```

Здесь можно ввести команду HELLO с названием своего логического диска, который будет открываться для чтения и записи на нулевом канале. Системный диск открывается на седьмом канале. После ввода указанной строки или по истечению тайм-аута запускается загрузчик операционной системы RT11-FB. Во время его работы выводятся правые угловые скобки на экран дисплея.

После появления идентификатора системы начинается исполнение командного файла STARTF.COM, находящегося на системном диске. Так как он общий для центральной и периферийных машин, то он должен содержать средство, позволяющее организовать работу в этих машинах по-разному. Проще всего использовать команду CONFIG SK:, которая прекращает выполнение командного файла в центральной машине, так как там драйвер SK: не инсталлирован. Для периферийных машин целесообразно разместить команды



```
SET USR NOSWAP
CNC SEARCH
FRUN CMINI
ASSIGN CK0: DK:
$@STARTF
```

Первая команда фиксирует USR в памяти и уменьшает, таким образом, число обращений к диску. CNC SEARCH устанавливает соответствие между таблицей устройств и реально открытыми каналами, что ускоряет последующие команды CNC DEVICES. Если периферийная машина имеет собственные диски, к которым нужно обеспечить доступ из другого узла, нужно запустить задачу обслуживания запросов CMN, так же как, в центре. Обычно таких дисков нет. В этом случае для того, чтобы на терминале печатались сообщения, следует запускать минимальную задачу обслуживания запросов при помощи команды FRUN CMINI. Команда ASSIGN переназначает логическое имя рабочего диска DK: на пользовательский диск, и последняя команда запускает оттуда индивидуальный стартовый командный файл. В нем пользователь может разместить свои команды, такие, как загрузку драйверов, назначение логических имен и т. д.

С момента исполнения этих команд запуск сети CMNet можно считать законченным и работа за терминалом периферийной машины фактически не отличается от работы на машине с дисками. Добавляются лишь команды открытия и закрытия устройств, которые аналогичны установке дисков на дисководы или снятию их. Останов и выключение всех дисков имеет свой аналог в команде CNC BYE.

#### ПРОГРАММА УПРАВЛЕНИЯ СЕТЬЮ

Программа управления сетью CNC является посредником между пользователем и сетью CMNet. С ее помощью пользователь может открывать и закрывать свои каналы на объекты, находящиеся в любых узлах сети. Программа CNC выводит информацию о топологии сети, открытых каналах и сетевых устройствах. Она выполняет установку имени узла, отправляет сообщения на терминалы других узлов и обеспечивает работу в режиме виртуального терминала. Короче говоря, программа управления сетью CNC выполняет все функции взаимодействия пользователя с сетью CMNet.

Существует две разновидности программы управления сетью, находящихся в файлах CNC.SAV и CNCR.SAV. Они отличаются друг от друга сообщениями, которые выводятся, соответственно, на английском или русском языке. Во всех остальных отношениях обе задачи идентичны и рассматриваются поэтому как одна.

Программа управления сетью CNC имеет свой собственный интерпретатор команд. Синтаксис этих команд похож на язык DCL операционной системы. Знаком готовности ввода следующей команды является двоеточие. Сама команда состоит из названия команды, возможных ключей выбора режима и аргументов. Аргументы отделяются от команды и один от другого пробелами. В случае отсутствия необходимого аргумента выводится дополнительный вопрос. На него можно отвечать требуемым аргументом или вопросительным знаком. В последнем случае выводится краткая справка об ожидаемом ответе, после чего вопрос повторяется.

При наборе команд можно воспользоваться редактором командной строки SL, ввод символа Ctrl/C сбрасывает интерпретатор команд в исходное состояние.

Во время исполнения команд могут встречаться критические ситуации. Перед выполнением соответствующих действий требуется подтверждение операции. В ответе анализируется лишь первая буква. Если это Y, то команда выполняется, во всех остальных случаях выполнение команды прекращается.

Некоторые команды требуют для выполнения соответствующих действий специальный пароль. Для обеспечения секретности подавляется эхо введенных символов на терминал.

Все команды, а также дополнительные аргументы, можно включать в командные файлы. Подтверждение операции и ввод пароля осуществляется только с терминала.

При запуске программы управления сетью с одновременным указанием команды и аргументов перед названием команды следует ввести апостроф, до и после которого нужны пробелы. Это предотвращает изменение порядка следования аргументов системным монитором.

Команды программы управления сетью CNC можно разделить на группы. Первая группа включает вспомогательные команды HELP, IDENT и EXIT. Они применяются для вывода справочной информации и идентификатора программы, и для выхода из программы, соответственно. Так как описанные три команды не работают с сетью, то они могут использоваться без сетевого драйвера. Все остальные команды требуют наличия работоспособного драй-



вера сети в памяти. Вторая группа объединяет команды опроса сети и вывода полученной информации. Это команды CHANNELS, DEVICES и TOPOLOGY, которые выводят, соответственно, информацию о каналах, устройствах и топологии сети. Третья группа включает такие команды работы с каналами и объектами, как открытие или закрытие канала. Соответственно использованы названия OPEN, PURGE и BYE. Четвертая группа команд состоит из команды поиска устройства сети SEARCH и команды назначения устройства на канал ASSIGN. Пятая группа объединяет три команды специального назначения. Это команды NAME — установка имени узла, MESSAGE — отправление сообщения и TERMINAL — вход в режим виртуального терминала.

Все команды программы управления сетью CNC могут иметь ключи выбора режима. Ключ состоит из косой черты и названия ключа. Имеются следующие ключи: выводимая информация направляется с помощью ключей /TERMINAL на терминал, /PRINT на устройство печати и /OUTPUT:файл в указанный файл. Если не указан ни один из них, подразумевается терминал. Ключ /FULL управляет форматом выводимой информации. Он обычно вызывает вывод дополнительной информации, как, например, echo на вводимый пароль или полный доступ в командах CHANNELS и DEVICES. Оба ключа /NOQUERY и /YES вызывают подавление запросов подтверждения операций и ввода пароля. При этом считается, что ответ всегда положителен, пароль пустой. Любое сообщение об ошибке выполнения команд подавляется ключем /QUIET. Ошибки разбора команды выводятся всегда. Отладочный режим, печатающий на терминал содержимое всех отправленных и принятых пакетов, включается с помощью ключа /DEBUG.

Допускается сокращение имен команд и ключей до недвусмысленных частей, при этом обеспечивается распознавание команды по первым трем буквам, ключа по одной букве.

Число аргументов зависит от конкретной команды и находится в пределах от нуля до трех. Аргументы могут быть специальными, такими, как аргумент команды HELP, название объекта, текст сообщения, или же общими для нескольких команд. Последние включают устройство, канал, узел и имя.

Под *устройством* понимается название логического или физического устройства операционной системы, как, например, DK: или SK4:. Устройство должно быть сетевым, т. е. работа допускается только с центральным драйвером узла. Для обеспечения независимости от конкретного интерфейса введены внутренние

логические имена с 0: по 7: для соответствующих устройств драйвера сети.

*Канал* в узком смысле обозначает номер канала. Номер может быть задан в виде одного восьмеричного числа, или в виде двух чисел, разделенных точкой. Вторая форма указывает класс объекта и конкретный номер отдельно. Класс объекта можно также обозначать соответствующими ключевыми словами DISK, GATEWAY, PRINTER, PROGRAM, SPECIAL, TAPE, TASK, TERMINAL или их сокращениями. Каналы специальных объектов могут обозначаться любым недвусмысленным сокращением из списка BOOTSTRAP, CONFIGURATION, ECHO, MAILBOX, MESSAGES, NAME, NEWS, NODE, TIME, TOPOLOGY. Весь класс выбирается с помощью ключевого слова ALL вместо номера канала. Примерами могут служить каналы 4, 0.4, DISK.4 или SPECIAL.NAME и NA, обозначающие дисковый канал номер 4 и специальный канал имени узла, соответственно.

Многие команды допускают использовать вместо номера канала также имя сетевого устройства. Кроме этого можно указывать ключи доступа, меняющие форму доступа к объекту относительно установленной по умолчанию. Ключ доступа состоит из косой черты и одного слова. Определены ключи /GETSTATUS, /PURGE, /TEMPORARY, /LOOKUP, /BUILD, /CHANGE, /DATA, /READ, /WRITE, /MODIFY, /SPFUN, /EXECUTE, /ABORT, а также эти же ключи с приставкой NO. Ключевые слова можно сокращать до одной буквы, приставку тоже. Указанные ключи устанавливают право получения статуса канала (форму доступа, размер и название объекта), закрытия канала, уничтожения временного объекта, открытия канала, создания объекта, изменения параметров (формы доступа, пароля), обмена данными в прозрачном режиме, чтения, записи, модификации информации, выполнения специальных функций, запуска и останова программ. Например, DK:/NOGET обозначает рабочий диск с запретом на получение статуса соответствующего канала, или 4:/W/T устанавливает для четвертого сетевого устройства право на запись и на уничтожение объекта, что является формой доступа временной записи.

*Узел* обозначается восьмеричным числом, ключевыми словами HOST, ALL и DOWN или именем узла. Сокращения недопустимы. Ключевые слова ALL и DOWN обозначают не один узел, а соответствующие группы узлов, что не во всех командах допустимо.

*Именем* является строка не более чем из восьми букв или цифр. Первым символом должна быть буква. Буквы могут вводиться



в разных регистрах, хотя в программе регистр не учитывается. Употребление имени узла вместо номера — более удобный способ, но при этом увеличивается время выполнения команды.

Ниже приводится описание отдельных команд программы управления сетью CNC.

**ASSIGN устройство [канал]** — команда устанавливает соответствие между устройством и каналом. По умолчанию номера устройств и сетевых каналов совпадают. Можно назначить любой дисковый канал на каждое из устройств. Если указано только устройство в качестве аргумента, то восстанавливается первоначальное значение номера канала. Команда ASSIGN нужна только в исключительных ситуациях, обычно она не используется.

Примеры:

ASSIGN 0: DISK.10 назначает десятый дисковый канал на нулевое устройство.

ASSIGN 0: и ASSIGN 0: DISK.0 восстанавливают первоначальное соответствие между номером устройства и дисковым каналом.

**BYE** — команда предназначена для закрытия всех сетевых устройств при окончании работы в данном узле.

Примеры:

BYE/NOQUERY или B/Y закрывают все устройства, включая системное.

**CHANNELS канал [узел]** — команда выводит информацию об указанном канале. Если указан узел, то опрашивается только он. Иначе опрашиваются все узлы сети, пока не будет найден канал. Возможен вывод всех каналов одного класса. Каждая строка содержит слева направо обозначение канала, ключи доступа, номер и имя узла, имя объекта и размер его. Полный перечень ключей доступа выводится командой CHANNELS/FULL.

Примеры:

CHANNEL/FULL TERM.0 выводит информацию о нулевом канале класса виртуальных терминалов.

CHANNEL CONFIG HOST выводит информацию о специальном канале конфигурации узла, расположенного ближе к центру.

**DEVICES [устройство]** — похожа на команду CHANNELS. Выводится информация об указанном сетевом устройстве или же о

всех восьми устройствах. Перед обозначением канала указывается имя устройства, вся остальная строка просто сдвинута вправо.

Примеры:

DEVICES выводит информацию о всех восьми устройствах.

DEVICE/FULL DK: выводит полную информацию о рабочем диске DK:.

**EXIT** — команда осуществляет возврат из программы управления сетью в монитор.

**HELP [аргумент]** — команда выводит справочную информацию о командах программы управления сетью. Необязательный аргумент выбирает соответствующую часть информации. Можно указать название команды, слово OPTIONS для ключей и звездочку для списка всех команд. Команда HELP/FULL выводит весь справочный текст.

Примеры:

HELP выводит общие сведения о командах программы CNC.

HELP \* выводит список всех команд.

HELP OPEN выводит справку о команде OPEN.

**IDENT** — команда выводит идентификатор программы. Он необходим для определения номера версии программы CNC.

**MESSAGE узел текст** — команда отправляет текст на терминал указанного узла. Она используется для обеспечения оперативной связи с пользователями сети. Ключ /FULL вставляет перед указанным текстом сообщения строку, содержащую имя узла отправителя и время отправления сообщения.

Примеры:

MESSAGE/FULL ALL ЗАКАНЧИВАЙТЕ, СКОРО ПРОФИЛАКТИКА ЦЕНТРА. передает сообщение всем узлам.

MESSAGE/FULL 101 ЗВОНИ ПО ТЕЛ. 1-23 передает сообщение только узлу с номером 101.

**NAME имя** — команда устанавливает имя собственного узла. Рекомендуется включать команду NAME в индивидуальный стартовый файл.

Примеры:

NAME BORIS устанавливает имя узла BORIS. Можно также использовать буквы другого регистра, т. е. Boris или БОРИС.



**OPEN канал [узел] объект**—команда позволяет открывать канал на существующий объект. Вместо канала можно указывать устройство, с помощью ключей можно выбирать нужную форму доступа. По умолчанию канал открывается для чтения, разрешены опрос канала и закрытие его. Когда объект находится в одном из центральных узлов, то имя или номер узла можно не указывать.

Примеры:

OPEN DK:/WR 101 RK1:WORKS открывает устройство DK: на файл RK1:WORKS.DSK, находящегося в узле с номером 101. Канал будет открыт для чтения и записи.

OPEN/NOQUERY 4: DX: открывает четвертое устройство сетевого драйвера для чтения на устройство DX: узла, находящегося ближе к центру или в самом центре. Пароль с терминала не спрашивается и считается пустым.

OPEN TERM.0/NR/D EL79 TI: открывает нулевой канал виртуальных терминалов в узле EL79 на один из терминалов.

OPEN/NOQUERY 7:/TEMP/WRITE SYS:—этой командой загрузчик открывает системный диск.

**PURGE канал [узел]**—с помощью этой команды закрывается канал. Можно также использовать имя устройства. Указание узла ограничивает действие команды на этот узел, иначе канал закрывается во всех узлах.

Примеры:

PURGE DK: закрывает рабочий диск DK:.

PURGE TERM.0 закрывает нулевой канал виртуальных терминалов.

**SEARCH [устройство [узел]]**—команда ищет канал, соответствующий указанному устройству, и модифицирует таблицу каналов, находящуюся в драйвере сети. Если указан узел, то проверяется только он. Можно не указывать аргументы, тогда возобновляется таблица каналов для всех устройств. Команда необходима после загрузки драйвера в память, особенно тогда, когда устройства открыты не в центре.

Примеры:

SEARCH ищет все восемь устройств.

SEARCH DK: HOST ищет только канал рабочего диска в узлах, ближе к центру сети.

**TERMINAL [узел [канал]]**—с помощью этой команды осуществляется вход в режим виртуального терминала. Связь осуществляется с указанным узлом через указанный канал. В случае отсутствия аргументов используются аргументы предыдущей команды TERMINAL. Первоначально установлено HOST и TERM.0. Возврат в программу управления сетью CNC осуществляется нажатием клавишей Ctrl/P, а также после длительного отсутствия обмена. Ключ /FULL подавляет запуск тайм-аута, так что возврат в программу CNC по отсутствию обмена в этом случае не происходит.

Примеры:

TERMINAL EL79 устанавливает связь терминала пользователя с виртуальным терминалом, открытым на нулевом канале виртуальных терминалов в узле EL79.

**TOPOLOGY [узел]**—по этой команде осуществляется вывод топологии связей между узлами сети. Топология выводится в виде дерева основных связей. Корнем является собственный узел, если не указан другой. На один шаг табуляции отстоят все непосредственные соседи этого узла. Соседи в свою очередь могут иметь своих соседей и т. д. Каждая строка включает номер и имя узла, а также число линий. Когда вместо имени выводится знак вопроса, то это означает, что узел не возвращает свое имя. С помощью ключа /FULL можно вывести полную таблицу линий; недоступные линии отмечены звездочкой перед номером. Когда сеть состоит из нескольких несвязанных основными связями деревьев, то выводится несколько деревьев. После дерева топологии выводится таблица всех дополнительных связей, если такие имеются.

Примеры:

TOPOLOGY выводит топологию относительно собственного узла.

TOPOLOGY/FULL HOST выводит полную информацию о топологии относительно центра сети.

Замечание:

Так как CNC является оверлейной программой, работа с оверлейным устройством считается критичной ситуацией. После закрытия оверлейного устройства будут работать только команды ASSIGN, BYE, EXIT, NAME, OPEN, PURGE, SEARCH, находящиеся в корневом сегменте программы. Любая другая команда вызовет фатальную ошибку монитора.



## ПРОГРАММЫ ОБСЛУЖИВАНИЯ ЗАПРОСОВ

Программы обслуживания запросов связывают транспортную службу сети CMNet с реальными объектами узла. Под объектами понимаются диски или дисковые файлы, а также имя узла, таблица линий, средство вывода сообщений, эхо и т. п.

Существуют две программы обслуживания запросов. Одна (CMN) выполняет полный перечень функций и нужна всегда, когда узел хочет предоставить свои диски для удаленного доступа к ним из других узлов.

Вторая (CMINI) является минимальной задачей обслуживания запросов, поддерживающей только каналы имени узла, таблицы линий, сообщений и эхо. Она запускается с помощью команд FRUN CMINI или SRUN CMINI и занимает не больше одного килобита в оперативной памяти. Минимальная задача обслуживания запросов используется с целью экономии места в оперативной памяти во всех узлах, не предоставляющих свои диски для доступа к ним через сеть. В случае отсутствия места в оперативной памяти машины для задачи обслуживания запросов можно ее не запускать совсем. Однако тогда не печатаются порой очень важные сообщения на терминал пользователя, кроме этого, нельзя обращаться к данному узлу по имени. Поэтому рекомендуется запускать задачу обслуживания запросов.

Программа обслуживания запросов CMN работает с реальными дисками, включенных в конфигурацию узла. Запускается она с помощью команды FRUN CMN или SRUN CMN с указанием размера буфера, например, FRUN CMN/BUFF:2000. До запуска должны быть загружены командой LOAD все драйверы дисков, включенных в конфигурацию, а также файловый процессор USR командой SET USR NOSWAP. Кроме этого нужны логические имена (также включенных в конфигурацию) устройств DAT:, TMP:, BOT: и SYS: для поиска файлов DAT:CMNET.DAT, TMP:CMNET.TMP, BOT:CMNET.BOT, и для системного устройства периферийных машин. Любые из указанных логических имен устройств или файлов могут отсутствовать, что влияет лишь на объем предоставляемых узлом услуг.

Файл описания конфигурации CMNET.DAT содержит полное описание устройств, распределяемых через сеть. Он также включает таблицу открытых в данном узле каналов. Менять содержимое этого файла можно при неактивной задаче обслуживания запросов с помощью программы модификации конфигурации узла CMNDAT.

Задача CMN постоянно модифицирует этот файл для отражения текущего состояния узла, которое восстанавливается при перезапуске программы. Поэтому файл должен находиться на устройстве, на которое разрешена запись. Когда файл конфигурации отсутствует, то действует правило умолчания, которое включает устройства с DK0: по DK6: для чтения и DK7: для чтения и записи, но с проверкой паролей.

Файл временной записи CMNET.TMP содержит таблицы переназначений блоков и сами записанные блоки для временной записи. Он так же, как и файл конфигурации узла, постоянно модифицируется и используется при перезапуске задачи. Отсутствие его означает отсутствие формы доступа временной записи в данном узле.

Файл загрузчика CMNET.BOT только читается и нужен для загрузки периферийных машин. Если его нет, то узел загрузку не выполняет.

Сам загрузчик открывает для чтения устройство SYS:, на котором должна находиться работоспособная система с монитором RT11-FB и соответствующим интерфейсу связи драйвером сети. Когда отсутствует логическое имя SYS:, то данный узел не предоставляет периферийным узлам системное устройство для запуска операционной системы.

Программа обслуживания запросов CMN имеет сложный алгоритм распределения буфера, состоящего из указанного в команде запуска и пространства, которое было занято подпрограммой инициализации. Используется он для хранения описания конфигурации узла (содержимое файла CMNET.DAT), для кэширования таблиц переназначений блоков временной записи и для буферов под пакеты. Количество последних можно узнать в статистике центрального драйвера узла. Чем больше буферов под пакеты, тем больше размер кэша таблиц переназначений и тем эффективнее работает программа, естественно. При недостаточном размере буфера программа CMN не запускается.

Во время работы программы CMN можно выгружать USR из оперативной памяти. При этом сохраняется полная работоспособность, за исключением открытия каналов, которое возвращает ошибку удаленной системы. Эта же ошибка возникает при переполнении таблицы каналов. Не следует забывать, что загрузчик открывает системное устройство, без которого он не может работать. Фиксация USR в памяти восстанавливает возможность открытия каналов в данном узле.



С целью предотвращения разрушения важной информации, особенно каталога, реализован специальный алгоритм останова программы обслуживания запросов CMN. В соответствии с этим алгоритмом после команды ABORT CMN начинается постепенное выключение отдельных функций, и лишь по истечении 90 секунд программа заканчивает свою работу.

#### ПРОГРАММА МОДИФИКАЦИИ КОНФИГУРАЦИИ УЗЛА

Программа обслуживания запросов CMN использует файл CMNET.DAT, в котором содержится описание конфигурации узла. Для изменения содержимого этого файла существует программа модификации конфигурации узла CMNDAT.

Так как файл CMNET.DAT используется задачей CMN, то модификации имеют смысл только тогда, когда она не запущена. Управляется программа CMNDAT стандартной строкой CSI. В качестве входного файла нужно указывать CMNET (расширение .DAT можно опускать). Необходимое действие выбирается с помощью привычных ключей CSI.

Если не указан ни один ключ, то выводится содержимое файла CMNET.DAT на терминал или в выходной файл. Модификации файла в этом случае не происходит. Информация включает время последней модификации файла программой CMN, номер узла, таблицу устройств, включенных в конфигурацию, и таблицу каналов, открытых на эти устройства.

Ключ /N:n меняет номер узла, для которого готовится файл конфигурации. Этот номер проверяется при запуске программы обслуживания запросов CMN.

Ключ /U:device устанавливает атрибуты доступа к указанному устройству. Атрибуты задаются с помощью ключей /R для чтения, /W для записи, /T для временной записи и /P для проверки паролей доступа. Если не указан ни один из этих ключей, то указанное устройство вычеркивается из таблицы устройств. При использовании ключа /U закрываются все каналы, имеющие отношение к данному устройству.

Ключ /A:n задает размер таблицы каналов. Если файл недостаточно длинный, то его можно расширить командой монитора CREATE/EXTEND и еще раз задать ключ /A. Размер таблицы каналов определяет максимальное число одновременно открываемых в данном узле каналов.

Количество элементов таблицы устройств менять невозможно. Оно задано равным десяти, и этого должно быть достаточно для подавляющего большинства сетей. В случае необходимости может быть поставлен файл CMNET.DAT с большей таблицей устройств.

Для получения краткой справки о ключах программы CMNDAT используется ключ /H. Примеры использования программы содержатся в главе «Запуск сети».

#### ПРОГРАММА ЗАДАНИЯ ПАРОЛЕЙ ДОСТУПА

Для защиты информации от несанкционированного доступа введены пароли на открытие каналов. Канал открывается только тогда, когда пароль совпадает с эталоном пароля, который хранится на соответствующем диске в файле CMNET.SYS. Несовпадение паролей обрабатывается как отсутствие самого объекта. Файл CMNET.SYS содержит три пароля: на чтение, запись и изменение характеристик. Последний пока еще нигде не используется и может поэтому быть любым. Указание пароля на запись позволяет осуществлять любой доступ, пароль для чтения только неразрушающий, т. е. чтение или временную запись. Обычно паролем защищаются только от несанкционированной записи, пароли для чтения, как правило, пустые.

Программа задания паролей CMNPAS управляется стандартной строкой CSI. В качестве файла указывается название объекта, для которого должны устанавливаться эталоны паролей. Кроме этого можно указывать следующие ключи:

Ключ /R задает отсутствие пароля для чтения, ключ /W для записи и ключ /C для изменения. Отсутствие эталона пароля обозначает, с одной стороны, что нельзя подобрать пароль, совпадающим с этим эталоном, с другой стороны программа CMNPAS не будет его вводить дополнительной строкой.

Ключ /D удаляет запись о паролях из файла CMNET.SYS для указанного объекта, для которого больше не нужны ранее введенные пароли.

Ключ /N запрещает запись на объект. Пока не задействован третий пароль, он действует аналогично /W. Ключ /T запрещает временную запись на объект и применяется только в исключительных ситуациях. Справку о ключах выводит, также как в программе CMNDAT, ключ /H.



После строки управления, содержащей название объекта и, возможно, ключи, вводятся эталоны паролей, каждый отдельной строкой с соответствующей подсказкой. Можно ввести до восьми любых печатных знаков; пустая строка соответствует пустому паролю. После ввода всех паролей программа CMNPAS модифицирует содержимое файла CMNET.SYS на том устройстве, где находится объект, для которого вводились эталоны паролей. Если такого файла не оказалось, то он создается автоматически. Таким же образом программа CMNPAS его расширяет, когда не хватает места в нем. Старые пароли объекта уничтожаются.

Примеры:

```
CMNPAS RK1:WORKS/C
```

```
<пустая строка>
```

```
2X+
```

устанавливает пароль 2X+ для файла RK1:WORKS.DSK. Доступ на чтение этого файла свободный, так как соответствующий пароль пустой.

```
CMNPAS DX:/W/C
```

```
A
```

запрещает запись на диск DX: и разрешает чтение только тем, кто знает пароль, в данном случае A.

```
CMNPAS RK3:OLD/D
```

удаляет все пароли для указанного файла.

## ДРАЙВЕРЫ

Драйверы сети CMNet выполняют передачу пакетов от узла к узлу и обеспечивают связь между операционной системой и линиями связи. Можно различать два вида драйверов. Первый включает драйверы узла, центральные драйверы, или главные драйверы (CMNet node handler). Имена их начинаются на букву С, вторая буква зависит от конкретного интерфейса. Эти драйверы обеспечивают выполнение функций виртуального диска, содержат подпрограммы транспортного и сетевого уровней и работают на одну линию, ведущую к центру. Все драйверы узла исключают друг друга, инсталлирован может быть только один из них.

Вторая группа драйверов, имена которых начинаются с буквы В, состоит из драйверов линий. Они сами не обслуживают запросы операционной системы и являются, по существу, загружаемыми подпрограммами центрального драйвера. Каждый драйвер линий отвечает за один тип интерфейса, количество линий может быть произвольным и задается при генерации драйвера. Допускается одновременное использование нескольких драйверов линий.

Вторая буква в названии драйверов использована для обозначения типа соответствующего интерфейса. В настоящий момент определены:

N — отсутствие интерфейса;

M — байтовый интерфейс DL-11 по протоколу V.24;

J — быстрый байтовый интерфейс DL-11 с готовностью;

K — быстрый словный интерфейс DL-11 типа DL-KC;

P — байтовый интерфейс системы ДОЗА;

Q — мультиплексор системы ДОЗА;

S — интерфейс с буфером SB-11.

Все драйверы сети CMNet перед работой должны быть загружены в память командой LOAD, а также настроены. Для этого файлы драйверов содержат свои программы настройки, которые запускаются командами R или RUN. Программы настройки управляются ключами обычной строки CSI. Общими для всех драйверов являются ключи:

/X — выход из программы настройки;

/U — разгрузка драйвера и выход из программы настройки;

/H — вывод справочной информации о ключах;

/L — вывод статистики работы драйвера.

Вывод осуществляется в выходной файл, если он указан, или на консольный терминал.

Центральные драйверы кроме указанных ключей имеют еще следующие:

/N:n — установка номера текущего узла;

/S — включение режима асинхронного чтения;

а драйверы линий ключ

/M — подключение драйвера к главному.



Замечания:

Только что загруженный центральный драйвер узла всегда нуждается в настройке. Для этого достаточно выполнить любую команду программы настройки, например /X.

Драйвер линий без команды /M работать не будет.

Использовать команду монитора UNLOAD для драйверов сети разрешается только в одном случае, а именно тогда, когда после команды /U программы настройки драйвер не разгружен и любая последующая команда программы настройки отвергается с сообщением «Bad handler».

Режим асинхронного чтения позволяет отправлять один пакет запроса на чтение с виртуального диска независимо от количества ответных пакетов, что уменьшает объем передаваемой информации. Однако существуют конфигурации сети, в которых синхронный режим, где на каждый блок посылается отдельный запрос, работает более эффективно.

Существует еще ряд ключей, имеющих отношение к линиям связи:

- /I:n — включение линии в топологию сети. Номер линии распространяется только на соответствующий драйвер;
- /C:n — адрес регистра состояния приемника;
- /V:n — адрес вектора прерывания приемника;
- /A — указание, что это дополнительная связь вне основного дерева топологии;
- /E:n — задание ожидаемого номера узла на другом конце линии. Младшие 15 разрядов содержат номер узла, старший разряд указывает на необходимость ответа именно указанного узла. Когда этот разряд в нуле, допускается также ответ от любого узла, входящего в поддерево указанного. Дополнительные связи всегда требуют указания единственного ожидаемого номера узла. Для основных связей можно его не указывать, тогда он вычисляется из номера текущего узла, сдвинутого на необходимое количество разрядов, плюс номер линии.
- /R:n — выключение линии из топологии сети;
- /F — вспомогательный ключ для вывода полной статистики (шире 80 символов).

Драйверы узла обычно обслуживают обращения к виртуальному диску. Кроме этого они выполняют целый ряд специальных функций, которыми пользуются программа настройки драйвера,

программа управления сетью CNC и программы обслуживания запросов CMN и CMINI. Программы пользователя могут использовать пять функций:

lin\$fn=204 считывает номер текущего узла в буфер. Счетчик слов должен быть равным единице, номер блока может быть произвольным (рекомендуется использовать нуль). Ошибка исполнения функции говорит о принципиальной неработоспособности драйвера. Когда номер узла равен нулю, то драйвер следующие запросы не может выполнять.

Для получения уникального ненулевого номера запроса используется функция rnb\$fn=206. Номер считывается в первое слово буфера. Рекомендуется использовать счетчик слов равным единице и номер блока равным нулю. Ошибок не может быть, если lin\$fn возвратила ненулевой номер узла.

Отправка пакета осуществляется с помощью функции get\$fn=277. Адрес буфера и счетчик слов описывают пакет, номер блока должен быть равным нулю. Ошибки возвращаются на плохой формат пакета, если только не имеет место одна из вышеуказанных причин ошибок.

Для отправления пакета и получения ответа на него существует функция rpl\$fn=300. Аргументы такие же, как у get\$fn. Ответный пакет может иметь любой размер, при приеме его не проверяется заданный счетчик слов. Поэтому рекомендуется выделить буфер для максимального размера пакета r.leng=600 байтам. Причиной ошибки кроме вышеуказанных может быть отсутствие ответа на запрос. При успешном завершении функции в буфере всегда находится ответный пакет, который в свою очередь может содержать сведения об ошибках.

Получение пакетов запросов по определенному номеру объекта реализовано с помощью функции geq\$fn=301. Аргументы и ошибки такие же, как у get\$fn. В пакете указывается номер объекта, а также пустой узел-получатель go.nop=100000. По завершению функции в указанном буфере (размером r.leng=600 байт) находится пакет с запросом к определенному объекту. Функция geq\$fn может использоваться для отправления ответа на запрос перед ожиданием следующего запроса, если поле узла-получателя в пакете содержит допустимый номер узла.

Все драйверы сети CMNet выводят на ключ /L (возможны аргументы) статистику своей работы в выходной файл или на терминал. Формат вывода зависит от конкретного драйвера. Здесь описана лишь общая часть.



Заголовок содержит имя и версию драйвера, номер узла, дату и время сбора информации.

Центральный драйвер узла печатает несколько строк, свидетельствующих о состоянии узла.

1. Наличие или отсутствие задачи обслуживания запросов (CMNet service job locked).
2. Количество свободных буферов (free packet buffer(s) available).
3. Количество неиспользованных буферов, принадлежащих задаче обслуживания запросов (unused service job packet buffer(s)).
4. Число отказов выделения буфера задачи обслуживания запросов (service job packet buffer allocation(s) failed).
5. Число отказов выделения любого буфера (packet buffer allocation(s) failed).
6. Наличие запросов на выделение буфера (allocation request(s) pending).
7. Состояние внутренней очереди обычных запросов чтения или записи виртуального диска (internal I/O requests queue not empty).
8. Состояние внутренней очереди запросов, ожидающих ответ из сети (answer queue not empty).
9. Состояние внутренней очереди запросов, ждущих повторного отправления (retry queue not empty).
10. Режим асинхронного чтения (asynchronous read enabled).
11. Отправитель и причина последней ошибки для обычных запросов виртуального диска (last error from node):

- диск находится в состоянии временной неготовности (object not ready).
- аппаратная ошибка чтения или записи (execution error).
- синтаксическая ошибка в пакете запроса (syntax error).
- ошибка маршрутизации (routing error).
- выключение узла (node offline).
- отсутствие свободного буфера (no free buffer).
- ошибка данных на линии (data transfer error).

Строки, не имеющие смысла, в распечатку не включаются. В частности, сведения о последней ошибке выводятся только один раз, что позволяет проконтролировать возникновение новой ошибки после последнего просмотра статистики.

Все драйверы, обслуживающие какие-то интерфейсы, включают

в распечатку статистику работы аппаратуры и логических линий. После некоторой общей информации для групповых интерфейсов следует таблица линий, в которой каждая линия занимает одну строку.

Так как ширина таблицы может превышать 80 символов, выводится лишь часть ее. Полную таблицу можно вывести с помощью ключа /F. Аргументы ключа /L выбирают линии, для которых заполняется таблица. По умолчанию выводятся все установленные линии.

Таблица включает слева направо следующие столбцы: (почти любой из них может отсутствовать в зависимости от интерфейса и ключа /F)

1. (line) Номер линии в драйвере.
2. (status) Состояние линии (одно слово).
3. (stbit) Бит состояния линии.
4. (node) Номер узла на другом конце линии. Звездочка перед номером узла показывает, что номер ошибочный, логической связи по этой линии нет.
5. (aux) Пометка дополнительной связи.
6. (tim) Состояние таймера на другом конце линии.
7. (queue) Состояние очереди на передачу в линию.
8. (csr/vector) Адреса регистра состояния и вектора прерывания интерфейса.
9. (xmtd) Количество успешно переданных пакетов.
10. (rcvd) Количество успешно принятых пакетов.
11. (poff) Количество отвергнутых пакетов по причине отключения узла.
12. (pdat) Количество отвергнутых пакетов по ошибке данных на линии.
13. (xdat) Количество попыток передачи, законченных по ошибке данных на линии.
14. (rdat) Количество попыток приема, законченных по ошибке данных в линии.
15. (pbuf) Количество отвергнутых пакетов из-за отсутствия свободного буфера.
16. (xbuf) Количество попыток передачи, законченных из-за отсутствия свободного буфера.
17. (rbuf) Количество попыток приема, законченных из-за отсутствия свободного буфера.
18. (hder) Количество аппаратных ошибок.
19. (sfer) Количество нарушений протокола.



20. (last\_error) Причина, приводящая к последнему нарушению протокола, а также бит состояния линии в этот момент.

Для большей наглядности нормальное состояние (основная связь, включенный таймер, пустая очередь) отмечено пустыми полями соответствующих столбцов, нулевые счетчики событий знаком минус.

Большой процент информации, выводимой по ключу /L, особенно вместе с ключем /F, нужен для выявления исключительных ситуаций в сети. Пользователя интересует обычно лишь малая часть всей информации:

1. Заголовок содержит номер собственного узла.

2. Состояние узла включает сведения о задаче обслуживания запросов и количестве свободных буферов. Указана причина последней ошибки обычных запросов чтения или записи.

3. Состояние линии свидетельствует о работоспособности связи.

Исправная, логически доступная линия характеризуется тем, что

- состояние idle или active,
- номер узла отличен от нуля и не помечен звездочкой,
- количество ошибок не растет.

Работоспособность линии снижена, когда

- состояние waiting, error или restart,
- медленно возрастает любой тип ошибок или попыток приема/передачи.

Линия физически не работает, когда

- состояние passive, turnoff или offline,
- быстро возрастает любой тип ошибок или попыток приема/передачи.

Линия логически недоступна, когда

- номер узла равен нулю или он помечен звездочкой, указывающей на расхождение между ожидаемым номером (задается при инсталлировании) и фактическим номером узла на другом конце линии.

### ЗАГРУЗЧИК

Сетевой загрузчик выполняет несколько функций. Обычный загрузчик считывает нулевой блок с диска в память и запускает

его. В сети нужно перед этим открыть системное устройство. Для разных интерфейсов нулевые блоки системных устройств отличаются, так что для загрузки из сети нужно искать соответствующий файл драйвера и грузить его содержимое в память. Для выполнения этих действий существует специальный сетевой загрузчик, который разделяется на две части:

Первичный сетевой загрузчик находится в драйвере линий и выгружается по перфоленточному протоколу в периферийную машину. Он содержит всю интерфейсно-зависимую часть сетевого загрузчика и печатает после запуска на терминале слово «CMNet».

Первичный сетевой загрузчик считывает вторичный сетевой загрузчик, находящийся в файле CMNET.BOT и доступный через специальный канал загрузчика SPEC.Boot. Сообщение «error» появляется, когда невозможно отправить запрос на чтение, сообщение «read error» на ошибку чтения загрузчика или отсутствие его.

Сообщение «bootstrap» выводит уже вторичный загрузчик. Далее он печатает на терминале приветствие системы и переходит в командный режим, о чем свидетельствует приглашение ввода команды «CMNet».

Возможны команды:

HELLO — общий вход в систему с загрузкой RT11-FB,

REENTER — перезапуск операционной системы,

OPEN — открытие пользовательского диска,

BOOT — загрузка операционной системы,

GET — загрузка файла образа памяти,

LOAD — загрузка файла формата абсолютного загрузчика,

IDENT — вывод идентификатора сетевого загрузчика,

EXIT — выход в аппаратный загрузчик HDBOOT,

? — вывод краткой справки о командах сетевого загрузчика.

Команды распознаются по первой букве. Работают все функции редактирования, Ctrl/C отменяет введенную команду. Самый первый вводимый символ после вывода приветствия системы обрабатывается особым способом. После истечения некоторого времени считается, что была введена команда HELLO без аргумента. В других местах время ввода не ограничено.

Далее следует описание отдельных команд сетевого загрузчика. Для простоты понимания приведены действия загрузчика в термини-



нах мониторинговых команд, включая команды программы управления сетью CNC.

**HELLO [диск]** — является главной командой сетевого загрузчика. Она открывает седьмой дисковый канал на общий системный диск, открывает пользовательский диск, если он указан, на нулевом дисковом канале и запускает систему RT11-FB с седьмого устройства. В терминах мониторинговых команд это можно себе представить следующим образом:

```
CNC ' PURGE DISK.7
CNC ' OPEN/NOQUERY DISK.7/TEMP/WRITE HOST SYS:
[CNC ' PURGE DISK.0]
[CNC ' OPEN DISK.0/WRITE HOST диск]
BOOT 7:RT11FB
```

**REENTER [система]** — выполняет загрузку операционной системы с седьмого сетевого устройства. Если название системы не указано, то запускается система RT11-FB.

```
BOOT 7:система
```

**OPEN диск** — открывает любой диск для чтения на нулевом дисковом канале. С этого диска можно запускать задачи, работающие без операционной системы. Команда также позволяет открывать пользовательский диск для чтения, после чего можно ввести команду HELLO без аргумента.

```
CNC ' PURGE DISK.0
CNC ' OPEN DISK.0 HOST диск
```

**BOOT система** — запускает указанную систему с нулевого канала. Эта команда нужна только в редких случаях, когда операционная система запускается не с общего системного диска.

```
BOOT 0:система
```

**GET файл** — загружает файл образа памяти и запускает задачу, если в файле имеется стартовый адрес. По умолчанию ищется файл с расширением .SAV .

```
GET файл
START
```

**LOAD файл** — загружает файл в формате абсолютного загрузчика и запускает задачу, если указан стартовый адрес. По умолчанию файл имеет расширение .LDA .

— Аналога команды нет. —

**IDENT** — выводит на терминал идентификатор сетевого загрузчика.

**EXIT** — запускает аппаратный загрузчик HDBOOT, с помощью которого можно осуществить загрузку операционной системы с любого из поддерживаемых этим загрузчиком устройств. Так как программа загрузчика находится в оперативной памяти, тест памяти работать не будет.

? — знак вопроса вместо команды выводит на терминал краткий текст с перечислением всех команд сетевого загрузчика.

После любой команды загрузки операционной системы должны выводиться на терминал правые угловые скобки. Их выводит подпрограмма чтения системного диска, которая уже не имеет отношения к сетевому загрузчику.

#### ОСОБЕННОСТИ РАЗНЫХ МОНИТОРОВ

Последняя глава описания сети CMNet для пользователей системы RT-11 посвящена особенностям отдельных мониторов этой операционной системы, а также особенностям системы TSX-Plus. Все изложенное в предыдущих главах справедливо для двух- или восьмизадачного монитора RT11-FB, которого можно считать основным монитором системы RT-11.

Использование однозадачного монитора RT11-SJ проблематично по нескольким причинам. Во-первых, он не позволяет запускать задачи обслуживания запросов CMN или CMINI. Во-вторых, алгоритмы работы монитора с очередями настолько упрощены, что они неправильно работают с внутренними очередями, которыми пользуется сетевой драйвер. Разрушение структур данных может произойти в момент ненормального выхода из любой программы (2 Ctrl/C или фатальные ошибки), а также при исполнении запроса .hreset, если в этот момент драйвер узла имел хотя бы один активный элемент очереди. В-третьих, использование монитором команды reset процессора запрещает прерывания от связанного интерфейса, которые разрешаются только по следующему тайм-ауту. Это приводит к постоянному изменению работоспособности линии связи и, соответственно, топологии сети, а отсутствие таймера способно вызвать зависание драйвера совсем.



Работа с расширенной памятью с помощью монитора RT11-XM отличается очень мало от работы под управлением монитора RT11-FB. Для экономного использования нижней части памяти существует вариант программы CMN, построенный на резидентном оверлее. Задача использует виртуальную память для программы и для буфера и находится в файле CMN.SAV. Драйверы сети имеют суффикс X и генерируются с файлом XM.MAC, как все драйверы для монитора RT11-XM. Однако по ошибке, содержащейся в самом мониторе, не удастся запустить программы настройки драйверов. Поэтому нужно скопировать файл соответствующего драйвера в другой (с расширением .SAV, например) и обнулить в нем ячейки 64 и 66 (восьм.).

Система TSX-Plus является, по существу, многопользовательской RT-11. В ней работают те же драйверы и те же программы, как и в RT-11. Сеть CMNet поддерживает TSX-Plus версии 6.1, которая базируется на функциях RT-11 версии 5.2. Программа управления сетью CNC использует посимвольный ввод с терминала и должна, поэтому, запускаться с ключем \*/SINGLE или инсталлироваться с ключами /SINGLE и /NOWAIT. Для обеспечения вывода сообщений на все терминалы системы существует специальный вариант программы обслуживания запросов CMN, находящийся в файле CMN.TSX. Эта задача не использует обычный терминальный ввод/вывод и может, таким образом, запускаться на открепленной линии системы (Detached job). Драйверы генерируются обычным для TSX-Plus способом, используя файл TSXCND.MAC, который содержит определения

MMG\$T=1

TSX\$P=1

и другие, если нужно. Для инсталлирования драйверы сети должны быть включены в список системы при генерации или программой TSXMOD; кроме этого они должны быть инсталлированы в RT-11, из которого запускается TSX-Plus. При инсталлировании следует указать параметр NOMAPH, который фиксирует драйверы в нижней части памяти системы.

Систему TSX-Plus рекомендуется запускать на всех машинах, имеющих расширенную память, так как она использует эту память гораздо эффективнее чем монитор RT11-XM. Главное преимущество TSX-Plus состоит, скорее всего, даже не в возможности много-терминальной работы. Каждая задача может без каких-либо изменений относительно RT-11 использовать память размера 28 Кслов.

С каждого терминала таких задач можно запускать несколько одновременно. Терминал переключается между задачами по вводу и выводу и может при необходимости восстанавливать изображение, относящееся к конкретной задаче. Сохранена полная совместимость с RT-11, кроме этого имеется богатый дополнительный сервис, как, например, встроенный отладчик, который вызывается без перекомпоновки отлаживаемой программы.

В периферийных машинах сети CMNet самая эффективная работа с системой TSX-Plus достигается, наверно, при соблюдении следующих рекомендаций:

Исходим из того, что память имеет размер 0,5 Мбайт и на машине работает только один человек. Систему следует сгенерировать без SWAP и с кэшем размером около 400 блоков. Тогда остается место для трех задач по 28 Кслов каждая. Одна задача — это CMN, две — в распоряжении пользователя. Эту систему можно запускать с удаленного диска. Кэширование его приводит к очень высокой скорости обращения к часто используемым задачам системного диска. Если имеется электронный диск, то он остается полностью свободным для нужд пользователя.

При наличии памяти больше 0,5 Мбайт можно работать с нескольких терминалов, запускать больше задач, увеличивать размер кэша и/или выделять часть памяти под электронный диск VM. Работу нескольких терминалов на 0,5 Мбайт или одного терминала при меньшей памяти целесообразно организовать в системе, загруженной с электронного диска. В этом случае можно использовать кэш и SWAP, но не надо забывать исключить электронный диск из кэширования с помощью параметра инсталляции NOCACHE.



## ЛИТЕРАТУРА

1. Гутче Р.Х. Неоднородная локальная сеть микро-ЭВМ для научно-исследовательских лабораторий. — Тезисы докладов 4 всесоюзного семинара по обработке физической информации, Ереван, М.: ЦНИИАтоминформ, 1988, с.9—11.
2. Кочеев А.А., Путьмаков А.Н., Зеленцов А.Г. Аппаратура связи для терминального класса. — Использование ЭВМ в учебной и научно-исследовательской работе студентов: Тезисы докладов, Новосибирск: НГУ, 1988, с.191—192.
3. Неханевич Э.Л., Романов А.В. Применение последовательной передачи данных для повышения помехоустойчивости измерительно-вычислительных систем. — Тезисы докладов всесоюзной научно-технической конференции «Аналитическое приборостроение», Тбилиси, 1986, с.33—34.
4. Селиванов А.Н. Интерфейс SB-11 с буферизацией данных для локальной сети. — Тезисы докладов 4 всесоюзного семинара по обработке физической информации, Ереван, М.: ЦНИИАтоминформ, 1988, с.12—13.
5. RT-11 Software Support Manual. — Digital Equipment Corporation, Maynard, Mass., 1983.
6. TSX-Plus Reference Manual. — S&H Computer Systems, Inc., Nashville, Tenn., 1985.

## Содержание

Введение . . . . .	3
Основные принципы . . . . .	6
Запуск сети . . . . .	8
Подготовка . . . . .	9
Запуск центра . . . . .	10
Запуск периферийных машин . . . . .	11
Программа управления сетью . . . . .	12
Программы обслуживания запросов . . . . .	20
Программа модификации конфигурации узла . . . . .	22
Программа задания паролей доступа . . . . .	23
Драйверы . . . . .	24
Загрузчик . . . . .	30
Особенности разных мониторов . . . . .	33



*Р.Х. Гутче*

**Неоднородная локальная сеть  
микро-ЭВМ CMNet.  
Описание для пользователей системы RT-11**

Ответственный за выпуск С.Г.Попов

---

Работа поступила 12 августа 1988 г.  
Подписано в печать 29.08.88 г. МН 08452  
Формат бумаги 60×90 1/16 Объем 2,2 печ.л., 1,8 уч.-изд.л.  
Тираж 290 экз. Бесплатно. Заказ № 117

---

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