

分布式网络操作系统的设计实现

夏 晓 东

(上海交通大学)

摘要 随着微机局部网络(LAN)产品的不断增加, LAN的体系结构, 尤其是其高层结构的标准化已提到的议事日程上。本文的前半部分, 提出了一个个人计算机局部网络体系结构的五层模型, 并定义了各层的功能及相邻层之间的接口关系。后半部分介绍了一个基于该模型设计实现的个人计算机分布式网络操作系统。

THE DESIGN AND IMPLEMENTATION OF DISTRIBUTED NETWORK OPERATING SYSTEM

Abstract Along with the increase of microcomputer Local Area Network (LAN) products, the standardization in the architecture of LAN, especially in high layers, is under investigation. A five-layer-model of personal computer LAN architecture is proposed in the first part. The functions within each layer and interfaces between adjacent layers are well defined. The second half describes the design and implementation of a distributed network operating system (PC-DNOS) based on the model.

一、概 述

国际标准化组织(ISO)为了协调世界各地计算机网络的发展, 定义了开放式系统互连的七层参考模型(OSI/RM), 明确地定义了计算机网络的七层体系结构及各层之间的相互关系, 其中包括物理层、数据链路层、网络层、传送层、会话层、表达层和应用层。它作为公共数据网的一个世界性标准, 已被越来越多的生产厂家、公司以及研究机构所接受。然而, 对于微机局部网络, 尤其是个人计算机或微型计算机局部网络的体系结构应当如何分层, 如何定义各层的功能及层间的接口关系, 除了著名的 IEEE 802 标准草案对物理层和数据链路层的描述外, 其余部分尚在研究之中。

对于微机局部网络, 是否也要严格地按照 OSI/RM 来设计, 存在着许多争论。一种意见认为, OSI/RM 是普遍适用的, 微机局部网络也应当具有 OSI/RM 七层体系结构。这种意见的主要依据是按 OSI/RM 七层结构实现的局部网络可以比较容易地实现与远程公共数据网的互联。另一种意见则认为, 微机局部网络由于不具有公共数据网那样的复杂性, 因而, 其结构也应有自己的

独立性。至于与远程公共数据网的互联,可以通过设计闸门机(Gateway),进行规程转换来实现。本文要介绍的正是作者在微机局部网络体系结构方面的设想以及一个以此设想为基础设计实现的一个分布式微机局部网络操作系统 PC-DNOS。

二、微机局部网络体系结构的描述

微机局部网络相对于远程公共数据网有四个明显的特点:

第一,由于网内工作站分布在一个不太大的物理范围内,因而,数据传送是以高速、可靠的方式进行的;

第二,如果不考虑网间互联,那么在 IEEE 802 标准草案所定义的介质存取方法中,信息的传输一般都是点到点的(Point to point),它与远程公共数据网中端到端(End to End)的意义是一致的。这样,在微机局部网中,便无所谓交换(switching)问题;

第三,由于信息在网内都是点到点传送的,任意两个工作站之间一般只存在一条通路。因而,几乎就没有路径选择(Routing)问题。我们注意到某些大公司已经推出或正在酝酿双环或三环系统。这样,在一对工作站之间就有可能存在二条或二条以上的路径,也就有通过哪一条路径进行传送的问题。但这与公共数据网中的路径选择有很大的不同。前者如果也可以称为是路径选择的话,那么,它在本质上是点到点的。这可以在信息发送的源点按某种策略简单地实现。但是,公共数据网内信息的传输可能要经过多个交换结点,在其上都必须有路径选择设施;

最后一个特点是,支持局部网的微型机一般要充当两种角色。它既是一个网络的资源站,又是一个网络的工作站。作为一个资源站,它支持网内的全体用户存取本地资源,如外存贮器,外围设备等。有时,还包括工作站本身的 CPU 处理资源。而作为一个工作站,它又应支持本地用户访问网内的其它资源。这样,对于处理能力有限的工作站来说,要实现比较复杂的功能,就受到一定的限制。

根据这些特点,如果将 OSI/RM 用于微机局部网络,那么其结构可以简化。在 OSI/RM 中,网络层的主要功能是在两个传送实体之间建立虚电路,以实现网络层数据单元的透明传送。由传送层下达的传送层数据单元在网络层被分组发送;由数据链路层上交的发往本地的数据分组在网络层进行组合。可见,在网络层报头中,除了应当供组合用的数据分组的标识信息外,还应当有目标实体的标识信息,供交换结点网络层对该分组进行路径选择。当数据传送跨越不同数据网时,网络层报头还应包括网际控制信息,供闸门机进行网际规程转换。而传送层的主要作用是提供端到端的传送控制,避免数据流的拥挤和阻塞,优化传送服务性能,以实现会话层实体间数据可靠而有效的传送。显然,OSI/RM 这两层的作用,表现在微机局部网上,主要就是数据分组和流量控制。可以这样设想:相应于 OSI/RM 网络层和传送层的功能可以在一层中实现,不妨称为网络传送层。同理,OSI/RM 会话层和表达层的功能,体现在微机局部网中,主要是资源管理和资源存取控制,包括建立联接、格式化数据、避免冲突等。这些功能要依赖于微机操作系统的支持。因此,也可将这两层简化为一层。由于当今流行的微机操作系统如 CP/M、MS-DOS 等,大多是以 DOS 为内核的,故称该层为网络 DOS 层或 NDOS 层。OSI/RM 对其它三层的描述与微机局部网应当是一致的。因此,微机局部网络的体系结构可大致分为五个层次,即物理层、数据链路层、网络传送层、网络 DOS 层和应用层。它与 OSI/RM 各层的对应关系如图 1 所示。

IEEE 802 标准草案对局部网络的物理层、数据链路层及其相互关系给出了详尽的描述。由于

| OSI/RM | 微机局部网络体系结构 |
|--------|------------|
| 应用层 | 应用层 |
| 表达层 | 网络 DOS 层 |
| 会话层 | |
| 传送层 | 网络传送层 |
| 网络层 | |
| 数据链路层 | 数据链路层 |
| 物理层 | 物理层 |

图 1 微机局部网络体系结构与 OSI-RM 的对照。

这些标准草案大多都已公布，故这里不再对物理层和数据链路层作更进一步的说明。现在，我们依次给出微机局部网络体系结构中的网络传送层、NDOS 层以及应用层的功能描述。

1. 网络传送层

网络传送层的作用主要是在一对已经建立了联接关系的 NDOS 系统进程之间提供数据的透明传输服务。一方面，NDOS 层经由接口界面的访问点下达的 NDOS 数据单元在这里被分组、编号，并加上网络传送层报头，成为网络传送层数据单元，逐一向数据链路层递交发送。在这过程中，为了避免数据发送节奏过快而与接收端失去同步，还应当有流量控制设施。另一方面，它通过与数据链路层接口界面的访问点，接收来自数据链路层的网络传送层数据单元，并且解释和移去该层报头，按原编号，重新组织 NDOS 层数据单元，送到与 NDOS 接口界面的访问点，供 NDOS 层接收。其过程如图 2 所示。可见，网络传送层的功能主要有两个，即 NDOS 层数据单元的分组、装配和网络传送层数据单元的透明传送。

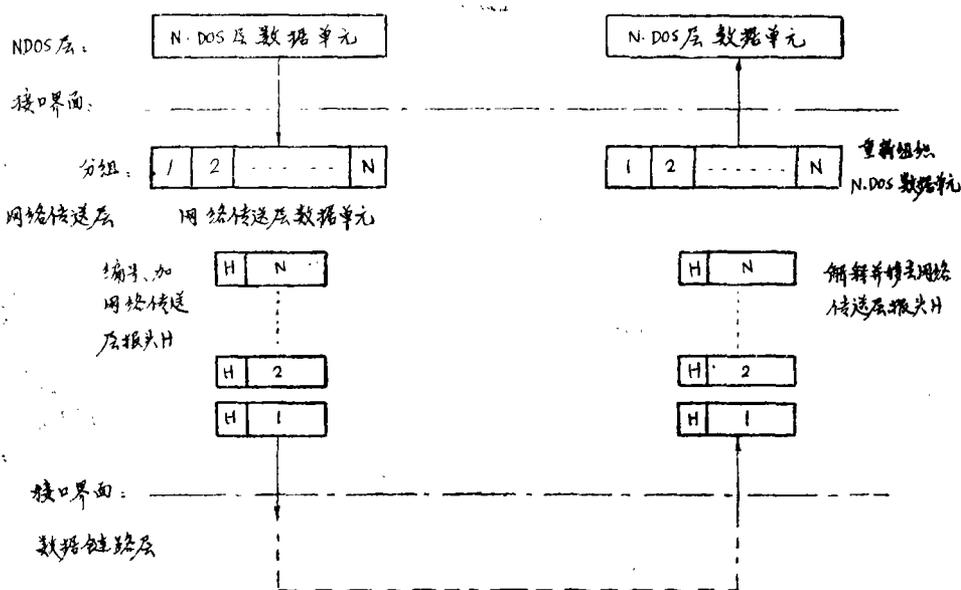


图 2 网络传送层功能示意图

2. 网络 DOS 层(即 NDOS 层)。

NDOS 层不仅为本地用户访问网络资源提供一个界面, 并且也为其它工作站的用户进程访问本地资源服务。NDOS 接收来自用户进程的网络服务请求, 并以网络传送层提供的数据透明传输为基础, 同用户进程所请求的目的工作站 NDOS 层的有关系统进程建立联接, 然后, 根据用户进程请求服务的类型, 组织相应的 NDOS 层数据单元, 经接口界面上的访问点, 交网络传送层发送。同时, NDOS 层要对来自远程的 NDOS 层数据单元进行解释, 激活相应的服务进程, 为远程用户访问本地资源服务。在一次联接结束时, 应及时地拆除。不难看出, 一次网络资源的访问过程, 实际上是由两个位于不同工作站上的 NDOS 层之间的若干次数据传送组成的。图 3 给出了从工作站 A 上读工作站 B 的 File 1 文件的一个记录的过程。显然, NDOS 是微机局部网络高层的基础。并且, NDOS 层规程的实现应当是与通信子系统无关的。也就是说, 无论通信子系统的设计实现采用何种方式, 只要在与 NDOS 层接口的界面上能提供标准的服务, 就可以支持 NDOS 规程。然而, 它与工作站所采用的微型机及其微型机的操作系统是密切相关的, 尤其是在异种操作系统支持的工作站上, NDOS 的实现方式是不同的。归纳起来, NDOS 层的功能也主要有两个, 一是同远程的 NDOS 层建立联接, 并在通信结束时拆除这个联接; 另一是管理作为网络资源一部分的本地资源, 向用户进程提供网络服务源语。

3. 应用层

这是微机局部网络体系结构的最高层。应用层是用户进程活动的场所。它通过标准的 I/O 设备接收操作员给出的命令, 并经由与 NDOS 层接口界面的访问点激活有关系统进程, 以提供网络服务。由于有了 NDOS 层的强有力的支持, 用户进程便可以将任何远程的网络资源都当作是本地资源的扩充, 这就使得网络环境中的应用软件具有极大的灵活性。同时, 应用层还应包括提供各

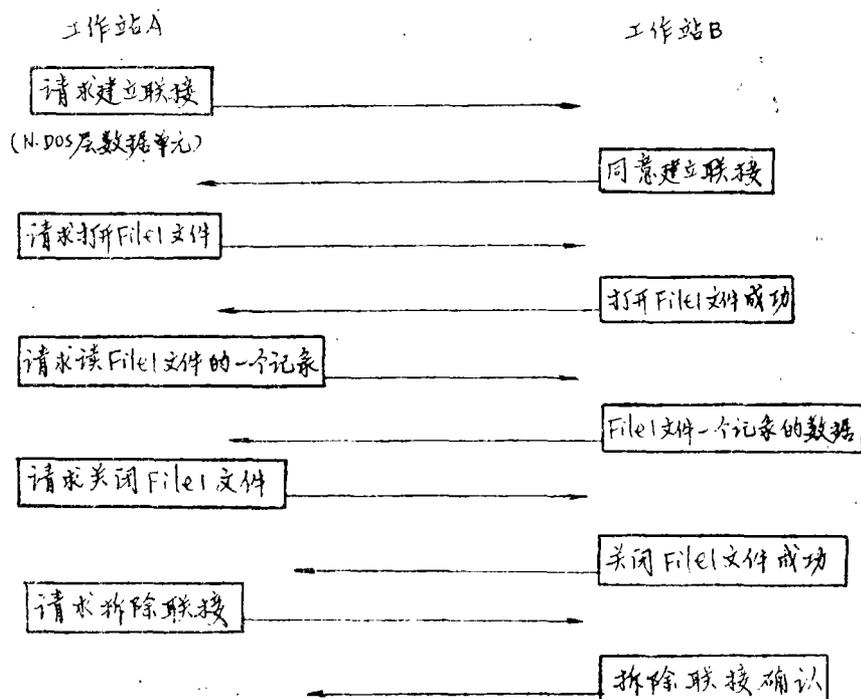


图 3 从工作站 A 上读工作站 B File 1 文件的过程

种专门服务的内部和外部命令、带有网络访问语句的高级语言编译或解释程序以及各种面向功能的应用软件包，如分布式数据库系统和电子邮件系统等。

必须强调指出的是，尽管微机局部网络不一定要严格地按照 OSI/RM 来设计，但 OSI 的基本原则却是必须遵循的：各层所定义的功能应当相互独立，相邻两层仅在接口界面的访问点上发生关系。其中，上一层规程要以下一层规程所实现的功能为基础。我们这里所提出的关于微机局部网络的体系结构正是按照这一原则进行定义和描述的。关于层间接口关系，我们在介绍 PC-DNOS 分布式网络操作系统的设计实现时，再给出其全部细节。

三、分布式网络操作系统 PC-DNOS 的结构及其所实现的功能

目前，微机局部网络操作系统的开发一般有三种形式。一是舍弃原来封闭的单机操作系统，从头开始设计网络操作系统。这样的设计必然会有较强的针对性，故效率是很高的，并且可以获得很灵活的功能。其弊病是除了设计上比较复杂外，有可能引起新的网络操作系统与原操作系统在应用软件上不能兼容。这对于那些已在单机操作系统下开发了许多应用软件的用户来说，往往是难以接受的。二是以现有的单机操作系统为基础，扩展其内核，增加网络驱动器并实现与原操作系统的衔接。经这种方式生成的网络操作系统，虽然在效率上要低于第一种方式，比如内存开销和系统服务的时间开销都较前者要大，但它不仅可以向用户提供与前者相同的系统服务，并且能够保证与原操作系统的兼容。三是各种面向应用的软件包的开发。它直接面向操作员(而不是面向应用程序)提供各种专门的网络服务，如文件传送、图象传送、存贮器共享、外围设备共享等。这种形式的开发比较简单，往往是一个专用系统，不够灵活。主要是网络服务的各个命令不能编程，因而不能向程序员提供服务。

PC-DNOS 是按照第二种方式设计的。它以 PC-DOS 为基础，将网络驱动器作为一个设备文件嵌入其 BIO(基本输入输出)部分，并将网络服务软件扩充到 DOS 中，以形成对网络用户程序的强有力支持。这样，由于 PC-DNOS 尽可能地利用了 PC-DOS 的支持，因而既避免了重新设计网络操作系统的复杂性，又使之较一般只有面向应用服务的软件包具有更多的灵活性。

PC-DOS 是将 Microsoft 公司的 MS-DOS 移植到 IBM PC 上所实现的一个个人计算机磁盘操作系统。它由引导记录、BIO 模块、IBMBIO.COM、IBMDOS.COM 以及 COMMAND.COM 五个部分组成。PC-DNOS 对 PC-DOS 的 IBMBIO.COM 和 IBMDOS.COM 进行了扩展。IBMBIO.COM 主要是由称为设备驱动器的设备驱动程序组成，这一部分是可以扩充的。用户可以通过增加设备驱动器来实现对所扩展的外设的管理。IBMDOS.COM 是用户访问系统资源的一个界面，也是 PC-DOS 的核心。其主要功能是系统资源的管理。它从应用程序接收系统服务的请求并获得所需参数，然后将其转化为能被 BIO 接受的形式，交有关设备驱动器处理。

PC-DNOS 是通过在 PC-DOS 的 BIO 上增加设备驱动器 CEDCS.SYS、在 DOS 上增加网络服务模块 NServer 实现的。它除了与 PC-DOS 完全兼容外，主要增加了分布式通信服务和分布式资源管理等功能。在硬件上，它利用了美国 3COM 公司的 EtherLink 以太网控制器的支持，故亦可称 PC-DNOS 为以太网操作系统。然而，必须强调的是，作为 PC-DNOS 主体的 NServer 模块，是与硬件无关的。不论采用何种拓扑结构的微机局部网络，也不论采用何种形式的硬件控制器，NServer 都毫无例外地可以兼容。当然，作为设备驱动器的 CEDCS.SYS 应随网络类型和硬件结构的不同有所变化。图 4 给出了 PC-DNOS 操作系统的结构。设备驱动器 CEDCS.SYS 与 Eth-

PC-DNOS

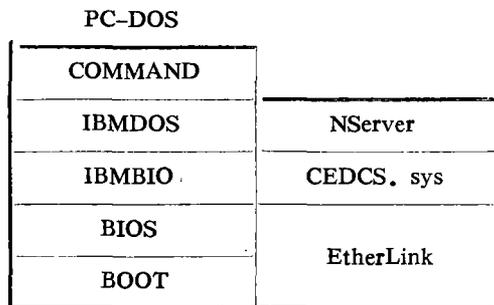


图 4 PC-DNOS 操作系统的结构

erLink 上固化的软件一起，实现了低级通信管理的全部功能。主要包括：数据帧的发送管理、接收管理和差错恢复。这也是微机局部网络体系结构中物理层和数据链路层(包括 介质访问控制子层 MAC 和逻辑链路控制子层 LLC)的功能。它给网络传送层接口界面的访问点上提供的服务可用三个函数来表示：

```
SEND-FRAME(FRAME-BUFFER, LENGTH, RESEND-TIME);
RECE-FRAME(FRAME-BUFFER);
GET-STATUS( );
```

函数 SEND-FRAME 提供了数据帧的发送服务。参数 FRAME-BUFFER、LENGTH、RESEND-TIME 分别为帧缓存地址、帧长度和帧重发次数。函数值为 0 时，表明发送成功。函数 RECE-FRAME 提供了数据帧的接收服务，参数 FRAME-BUFFER 为帧缓存地址。函数值为 0 时，表明收到了一个正确的数据帧。非 0 时，表明尚未收到数据帧。函数 GET-STATUS 起检测通信子系统的作用。函数值即为检测结果代码。网络传送层正是通过调用这些函数来得到数据链路层所提供的服务的。因此，函数 SEND-FRAME、RECE-FRAME、GET-STATUS 完整地给出了网络传送层和数据链路层的接口描述。

网络服务模块 NServer 实现了网络传送层和 NDOS 层的功能。网络传送层的功能相当简单，仅是对欲发送的数据进行分组和装配。它在 NDOS 层接口界面的访问点上提供了两个服务源语，即 SEND-PACKET(PACKET-BUFFER, LENGTH)和 RECE-PACKET(PACKET-BUFFER)。NDOS 层利用这两个服务源语分别进行信包的发送和接收，以实现同活跃在其它工作站的系统进程或用户进程之间的通信。NServer 在 NDOS 层与用户程序的界面上，提供了二十多条网络服务源语或网络系统调用，并在 NDOS 层上形成了各网络服务源语的服务程序。这使得在 PC-DNOS 支持下的用户程序，可以通过调用各网络服务源语，灵活方便地访问网络资源或与其它工作站进行实时通信。

NServer 模块向用户程序提供的服务源语主要分为站名服务、显示服务和文件系统服务三类。站名服务就是允许用户进程实时地得到当前网上的站名信息。有关站名的意义及管理方式，我们在下一节中介绍。这里，我们仅列出它所包括的下列五条服务源语：

```
GET-LOCAL-NAME——得到本地站名；
GET-NAME-TAB——得到名表首址；
SEEK-NAME——搜索某站是否存在；
```

SET-PACKET-BUFFER——设置信包缓存地址；
GET-PACKET-BUFFER——得到信包缓存地址；
显示服务主要包括下列三条服务源语：
REMOTE-DISPLAY(mode, buffer, dest-name)——远程屏幕显示；
MASK——屏蔽一般显示；
UNMASK——开放一般显示。

这里，参数 **Mode** 为显示方式。当 **mode** 等于 0、1、2、3 时分别对应广播窗口显示、单站窗口显示、广播一般显示、单站一般显示。**buffer**、**dest-name** 分别表示欲显示的内容的内存地址及目标站名。

PC-DNOS 的文件系统服务支持用户程序在网中其它工作站的外存空间上建立、查询、修改文件，并设置信号灯以维护全网文件的一致性。它主要包括下列九条服务源语：

REMOTE-OPEN——打开一个远程文件；
REMOTE-CLOSE——关闭一个远程文件；
REMOTE-READ——读某个已打开的远程文件；
REMOTE-WRITE——写某个已打开的远程文件；
REMOTE-SEEK——搜索某个文件是否存在；
REMOTE-CREATE——建立一个远程文件；
REMOTE-ERASE——删除一个远程文件；
REMOTE-TEST——测试某个文件是否正在被访问；
REMOTE-POINTER——移动某个已打开的远程文件读写指针。

关于文件系统一致性的维护原理，我们放到本文第七节中介绍。

四、通信管理

由于 **PC-DNOS** 是对 **PC-DOS** 的扩展，因而它与 **PC-DOS** 相同的部分如对本本地资源的管理等已为人们所熟悉，这里就不再赘述。我们仅对 **PC-DNOS** 中的通信管理、通信规程和网络资源管理部分进行介绍。

通信管理是分布式网络操作系统的一个重要部分。在 **PC-DNOS** 中，通信管理分为帧缓存管理和站名管理两个部分。

为了保证在对接收帧处理的同时，允许接收新的数据帧，我们以队列的形式来实施帧缓存管理。所有的接收帧按到达的先后顺序排列在帧队列中，系统进程按先来先服务的策略，依次处理每一个列队项。帧队列的结构如图 5 所示。队首指针 **Head-Pointer** 指向当前正在处理的队列项，队尾指针 **Tail-Pointer** 指向最末一个帧的结束处。当当前队列项处理完毕时，首指针移向下一个队列项，顺序进行处理，直至最后一项。

帧队列的管理采用通常队列管理的方式。当有数据帧到达时，首先通过函数 **GET-BUFFER(LENGTH)** 得到与帧相应长度的缓存。这里，参数 **LENGTH** 是所请求缓存的字节长度。然后，将所接收的数据帧在所获得的缓存区建立一个新的队列项。现在，我们给出函数 **GET-BUFFER** 的 C 语言源程序，供读者参考。

```

GET-BUFFER(LENGTH)
{ IF(Head-Pointer == Tail-Pointer)
  { Head-Pointer = Buffer-top,
    Tail-Pointer = Buffer-top,
  }
  WHILE(NOT FALSE)
  { IF(LENGTH <= Buffer-bottom - Tail-Pointer)
    { VI = Tail-Pointer,
      Tail-Pointer = <(Tail-Pointer + LENGTH),
      return(VI);
    }
    IF(Head-Pointer == Buffer-top)
    { return(0);
    }
    V 2 = Tail-Pointer - Head-Pointer,
    V 3 = Head-Pointer - Buffer-top,
    V 4 = Head-Pointer
    V 5 = Buffer-top
    WHILE(V 2 > 0)do
    { V 5[ ] = V 4[ ],
      V 5 = V 5 + 1,
      V 4 = V 4 + 1;
    }
    Head-Pointer = Head-Pointer - V 3,
    Tail-Pointer = Tail-Pointer - V 4;
  }
}

```

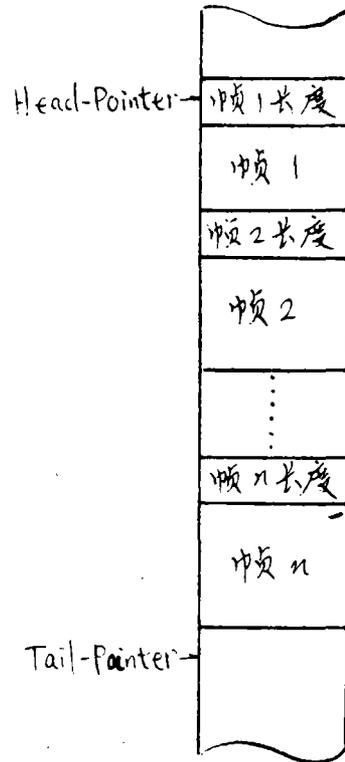


图 5 帧队列结构

这里，变量 **Buffer-top** 和 **Buffer-bottom** 分别为帧队列空间的上、下界，参数 **LENGTH** 给出了所要求的帧缓存长度。当函数值为 0 时，表明缓存空间已不足；非 0 时，函数值即为所得到的缓存区首址。

通信管理的另一部分是站名管理。在每个工作站的 **EtherLink** 以太网控制器上，都有一个称为以太网地址的物理地址，共 6 字节 48 位长。它唯一地标识了该工作站。欲向一个工作站发送数据帧，必须在帧的目的地址域上填入该站的物理地址。显然，对于用户程序，采用物理地址来标识工作站是很不直观的。

在 **PC-DNOS** 中，每一个工作站被赋予一个站名，它与工作站的物理地址是相对应的。用户可以用站名来标识工作站。在进行网络系统调用时，只要给出目标站名而不是其物理地址即可实现对其它工作站的访问。至于站名到物理地址的转换，则在系统程序中实现。**PC-DNOS** 为了支持用户方便地访问网络，实时地维持了一个站名地址表，称为 **ADDR-NAME**，其结构如图 6 所示。图中给出了 n 个表项，每一项分别表示当前网内可能开机的工作站的物理地址和站名信息。

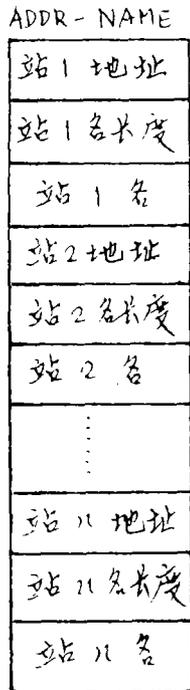


图 6 站名地址表 ADDR-NAME 格式

这样，通过查询 ADDR-NAME，用户程序即可知道当前可能有多少工作站开机及开机工作站的物理地址和站名。

ADDR-NAME 是在工作站开机入网时建立的。一个工作站入网时，要把本站的物理地址和站名通过广播帧向网中进行广播。每一个收到该广播帧的工作站首先检查该帧所广播的物理地址在本站的 ADDR-NAME 中是否存在。若存在，则表明该广播帧的作用是修改站名。此时，要将所对应的表项的站名信息进行修改。若不存在，还要检查 ADDR-NAME 中是否有与该帧所广播的站名相同的表项。如果发现相同站名，要向原发送站名拒绝帧。否则，建立一个新的表项并填入该广播帧所含的物理地址和站名。随后，经过一个随机时延，再向原发送站发送带有本站物理地址和站名的地址站名响应帧。而新入网的工作站在收到各站发回的地址站名响应帧后，按所给地址和站名，建立 ADDR-NAME 表。图 7、图 8 分别给出了这一过程的示意图和一个工作站对地址站名广播帧进行处理的流程图。最后，还有两点要作进一步的说明。一是 PC-DNOS 支持用户在不重新启动的情况下，动态地修改本站站名。这时，它与新入网时一样，发送一个地址站名广播帧，并且清除原 ADDR-NAME，待各

站送回地址站名响应帧后，重新建立新的 ADDR-NAME。这样，用户便可以比较确切地知道当前网上工作站的开机情况。虽然 PC-DNOS 要求工作站在关机前执行一个退网程序，使其它工作站在自己的 ADDR-NAME 中去除该站所对应的表项。但要让每一个操作员都记住并且在任何情况下都照办是有困难的。若工作站没有执行退网程序就关机，则它在其它工作站的 ADDR-NAME 中就一直存在。这就是我们前面称 ADDR-NAME 包含的是“可能”开机的工作站站名地址信息的原因。因而，如果再执行一次入网程序，则所得的 ADDR-NAME 就有较强的实时性。另一点

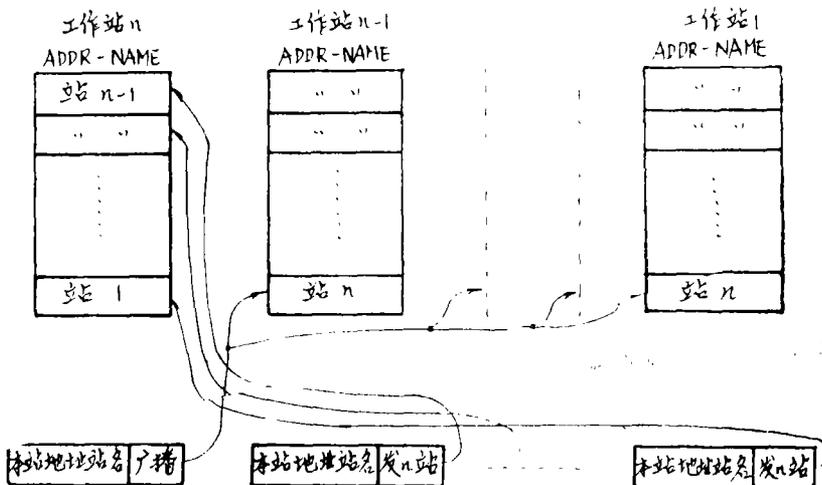


图 7 新入网的工作站 n 建立 ADDR-NAME 的过程

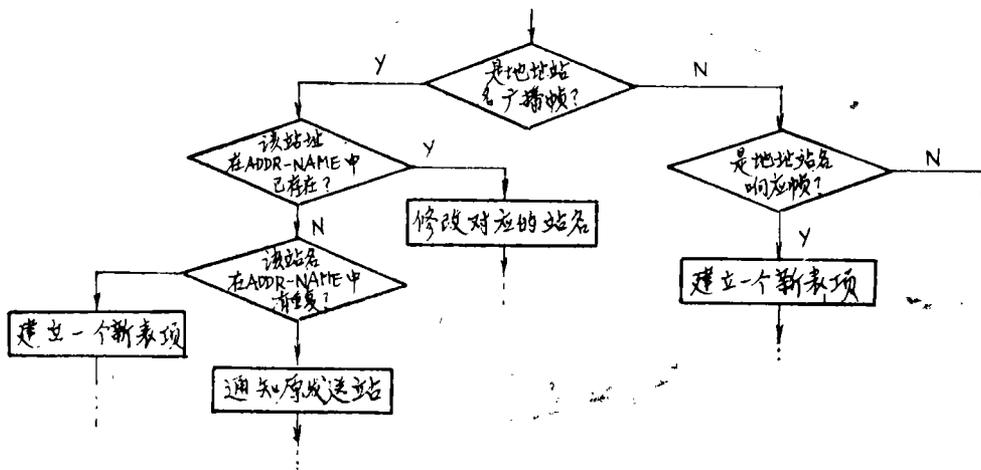


图 8 PC-DNOS 站名管理的流程图

要说明的是，每个工作站在收到了一个新入网或新改名的地址站名广播帧后，并不立即发本站的地址站名响应帧，而是要经过一个随机时延。这是由于已入网的所有工作站都要给出自己的地址站名响应帧。若不加时延，则响应帧会几乎同时地涌向原广播帧发送站。这容易造成工作站缓存溢出。加了随机延时后，各工作站发送地址站名响应帧的时间被错开，经反复试验表明是十分成功的。PC-DNOS 这种独特而有效的分布式站名管理方法，使用户感到十分灵活和方便。

五、通信规程

通信规程实际上是活动在不同工作站对等层内的系统进程在进行信息交换时所应遵循的规则，这里介绍一个 PC-DNOS 支持下实现的 NDOS 通信规程。NDOS 层数据单元是 NDOS 层之间信息交换的基本单位，其格式如图 9 所示。服务标识字反映了该数据单元的意义。目前，我们将其分成显示类、文件类和其它类。显示类和文件类数据单元分别起显示服务、文件服务的作用。其它类数据单元则在分布式处理和分布式外设资源共享时用到。当 NDOS 数据单元与某一类服务有关时，其第一个字即为与该类有关的服务标识字。

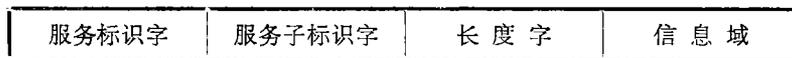


图 9 NDOS 层数据单元格式

我们定义：

显示服务标识字 = $\times 00000000010000(\text{B})$;

文件服务标识字 = $\times 000000000110000(\text{B})$;

其它服务标识字 = $\times 000000001110000(\text{B})$;

当服务标识字最高位为 0 时，表明是请求方给出的 NDOS 数据单元，否则是服务方给出的 NDOS 数据单元。服务子标识字定义了服务的方式。长度字定义了信息域的字节长度。信息的格式随服务类型和方式的不同而不同。我们在图 2 中给出了一个 NDOS 层数据单元的实例。其意义是请求某站将一个以字节 0 结尾的字符串写入文件 EXAMPLE，字符串的内容是 'HOW ARE YOU!'。图中采用的是十六进制数表示，对于文件名和字符串，使用了 ASCII 字符表示。

图 3 给出的是在工作站 A 上活跃的某进程要从工作站 B 的 File1 文件上读一个记录的过程。我们看到,通信过程分为三个阶段。第一阶段是联接阶段。在工作站 A 发出了联接请求后,可能发生三种情况。第一种情况是工作站 B 用户屏蔽了网络服务命令或工作站 B 的网络服务已饱和,如缓存队列已满等。这时,工作站 B 将发回一个拒绝联接的数据单元。另一种情况是工作站 B 在入网后,没有执行退网程序就关机了。这样,工作站 A 将因长时间得不到响应而超时。此时,PC-DNOS 通过网络服务源语的返回值向用户进程报告。用户可以根据情况决定是再次尝试还是放弃。最后一种情况就是象图 3 给出的那样,工作站 B 同意接受来自其它工作站的网络服务请求,给出联接确认的数据单元,并建立已同工作站 A 联接的标志。工作站 A 只有在这种情况下,才能向工作站 B 的 NDOS 层请求进一步的网络服务。

第二阶段是服务阶段。工作站 A 在收到来自工作站 B 的联接确认后,进一步向工作站 B 给出打开 File1 文件的请求。工作站 B 在收到该请求后,要通过检查本地的文件系统来确认该文件是否已被上锁。若已上锁,表明这是一个不可共享的文件或该文件正在被访问。这时,工作站 B 应当通知原请求站取消该项服务;若未上锁,则执行打开 File1 文件的操作,并把执行结果通过有关 NDOS 数据单元向工作站 A 报告。File1 文件被打开后,工作站 A 即可向工作站 B 发送请求顺序读一个文件记录的 NDOS 数据单元。工作站 B 的系统服务进程通过系统调用顺序读 File1 文件,并将所读的一个记录回送工作站 A。在第二阶段的最后,工作站 A 要请求工作站 B 关闭 File1 文件并解锁。当工作站 A 收到了来自工作站 B 关于文件关闭成功的 NDOS 数据单元后,第二阶段便告结束。

第三阶段是拆除联接阶段。由工作站 A 提出请求,工作站 B 予以响应。

一次联接建立后,可以进行多次服务,直至联接被拆除。不难设想,其它的远程服务在第一阶段和第三阶段都是一致的。所不同的是,在第二阶段要使用不同的网络服务源语以实现相应的网络服务功能。

六、显示服务

PC-DNOS 向程序员提供的网络显示服务分为四类,即单站窗口显示、广播窗口显示、单站一般显示以及广播一般显示。

窗口显示是不可屏蔽的远程显示方式。显示窗口位于右上角,共四行,每行可显示 40 个字节或 20 个汉字。其中,前三行供用户程序显示用,第四行由系统程序显示请求显示的站名。当一个工作站向另一个工作站进行窗口显示时,目标工作站当前进程被挂起,显示窗口内显示所收到的消息并附有请求站的站名。窗口显示的内容持续 15 秒钟。在这之后,原窗口内显示的内容被恢复,被挂起的进程继续运行。

为了避免与原用户进程在显示屏幕上发生冲突,PC-DNOS 在显示服务中采用了直接屏幕缓存区写入的方法,即将显示的代码转换成点阵直接送入屏幕缓存区。这样,要保存和恢复原窗口内显示的内容就很容易。在显示之前,先将窗口内的全部点阵码保护在系统内部专门设置的临时缓存区,并将显示窗口所对应的显示缓存全部清 0,然后再将远程用户所要求显示的内容转换成点阵送入显示窗口缓存进行显示。显示结束时,把保存在临时缓存区的原显示点阵重新送回,即可恢复原显示内容。

单站显示指的是仅向某一个工作站进行显示,而广播显示则是向当前开机的所有工作站进行

显示。有了窗口显示服务，PC-DNOS 支持下的程序员就可以设计各种灵活的程序，支持工作站操作员进行实时通信。例如，可以随时发送各种临时性的通知、消息等。由于窗口显示是不可屏蔽的，因而在各工作站正常运行的情况下，窗口显示总是可以完成的。

由于窗口显示的有效时间不长，若显示时，操作员离开了工作站控制台，那便不能产生应有的作用。为此，PC-DNOS 还支持另外一种显示方式，即一般显示。一般显示是从当前光标的下一行开始显示，无长度限制。显示结束后，光标位于所显示内容末尾的下一行行首。一般显示不对原屏幕内容进行恢复，但它可以被屏蔽。当操作员不希望当前屏幕所显示的内容被破坏时，可以通过命令将一般显示屏蔽。而当操作员要离开时，可将一般显示开放。这样，即使工作站上暂时无操作员，其它站发来的消息也可进行显示并保留在屏幕上。一般显示既有单站方式，也有广播方式。

PC-DNOS 所具有的灵活的开放式远程显示功能为用户在其上作进一步的开发提供了极大的方便。有了 PC-DNOS 的显示服务，电子邮件、电子会议等各种分布式系统的设计就可以简化。为了方便用户开发，我们给出各显示服务的调用入口和返回值。调用以软中断方式进行，总入口为 2DH，即 INT 2DH，各种服务的入口参数和返回值如下：

1. AH = 0，广播窗口显示；

入口：DS : DX 指向欲显示的信息串。其中，第一个字为信息长度，最大为 54。

出口：AL = 0，显示成功；AL = FF，发送失败。

2. AH = 1，单站窗口显示；

入口：DS : DX 同广播窗口显示相同，DS : SI 指向目的站名串。其中，第一个字为目的站名长度。

出口：AL = 0，显示成功；AL = FF，目标站无响应；AL = 80 H，目标站不存在；AL = F0，发送失败。

3. AH = 2，广播一般显示；

入口：出口与广播窗口显示相同。

4. AH = 3，单站一般显示；

入口：与单站窗口显示相同。

出口：AL = F1，目标站已将一般显示屏蔽，其它部分与单站窗口显示相同。

5. AH = 4，屏蔽本站一般显示；

无入口参数和返回值。

6. AH = 5，开放本站一般显示；

无入口参数。

出口：AL = 0，在这之前，工作站一般显示是开放的；AL = 1，在这之前，工作站一般显示是屏蔽的。

七、文件系统服务

PC-DNOS 支持用户在一个工作站上访问其它工作站上的文件系统。在第二节中，我们已给出了与文件系统服务有关的九条服务源语。用户程序员可以根据自己的应用要求，在应用程序中，有序地使用这些服务源语，来实现许多有用的功能。关于这些服务源语的调用方式本文不再一一

介绍。我们只想着重就 PC-DNOS 中文件系统服务设计实现的两个关键问题作一点说明。一是文件系统数据的一致性问题，另一则是系统调用的再入问题。

文件系统数据一致性的维护，要求一个文件不能同时被两个或两个以上的用户访问。由于 PC-DNOS 是在 PC-DOS 上扩充、修改而成的，而 PC-DOS 是单用户操作系统，设计时，并没有考虑到会有多用户时访问一个文件的情形。因此，PC-DOS 作为一个支持多用户访问一个工作站的文件系统的网络操作系统，必须要附加文件系统数据一致性的维护设施。

基本的方法是增加文件属性。在 PC-DOS 中，每一个文件的目录项内含有一个属性字节，它定义了该文件的属性。属性字节的低六位 b_0 、 b_1 、 b_2 、 b_3 、 b_4 、 b_5 为有效位，其意义如下所示：

- $b_0 = 1$ ，该文件为只读文件；
- $b_1 = 1$ ，该文件为隐文件；
- $b_2 = 1$ ，该文件为系统文件；
- $b_3 = 1$ ，表明该目录项为该磁盘的标号；
- $b_4 = 1$ ，表明该目录项定义了一个子目录；
- $b_5 = 1$ ，该文件为可读写文件。

文件属性字节的高二位是保留的。PC-DNOS 将文件属性进行了扩展，利用了属性字节的高二位，并作了如下定义：

- $b_7, b_6 = 00$ ，该文件不可被本工作站以外的其它用户访问；
- $b_7, b_6 = 01$ ，该文件正在被某个用户访问，且已被上锁；
- $b_7, b_6 = 1x$ ，该文件可以被全网共享。

这样，一个远程用户要访问本地文件，只有在文件属性字节的最高位 b_7 为 1 时，才可能成功。

PC-DNOS 的文件系统服务进程在接到某一远程用户的文件服务请求之后，立即被激活，它要检查欲被远程用户访问的文件是否已上锁，即其属性字节的最高位是否为 1。若是一个可共享的文件且未上锁，那么才允许进一步的服务。原则，通知原请求站，该文件至少暂时不能被访问。至于原请求站是否再次调用服务源语进行尝试，用户可在程序中设定。

系统调用的再入是指两次和两次以上进行系统调用的嵌套，这在个人计算机操作系统中通常是不允许的。在 PC-DNOS 中，为简化设计节省内存开销，仍采用了原 PC-DOS 的系统调用来为远程用户进行文件系统服务。这就使得系统调用的再入不可避免地要发生。

PC-DOS 在系统调用时所使用的系统堆栈和系统工作区是绝对定位的。这就决定了在 PC-DOS 中不能进行系统调用的嵌套。PC-DNOS 中，网络服务进程是靠中断激活的，而中断在任何时候都可能发生。因此，当某用户进程正在用系统调用访问系统资源时，一旦有网络中断发生，它就要被挂起且网络服务进程要因此被激活。这样，如果网络服务进程要再次进行系统调用，就会出现系统调用的嵌套即在系统调用中再次进行系统调用的情形。这无疑会导致前一次系统调用时的系统堆栈和系统工作区的破坏。为了更清楚起见，我们用图 10 来进一步说明两次进行系统调用对系统堆栈的影响。

设某一用户进程正在进行系统调用。在时刻①，发生了网络中断。于是，原用户进程要被挂起，网络服务进程被激活投入运行。假定网络服务进程运行到时刻②时，又要进行系统调用，如图 10(a)所示。这样，系统堆栈又要被绝对定位到原栈顶。尽管网络服务进程可以在再次进行系统调用前对系统堆栈的当前栈指针进行保护，但这只能保证将堆栈指针在退出第二次系统调用后恢复到原来位置，系统堆栈的刷新仍然是不可避免的。因为再次进行系统调用时，系统堆栈又要

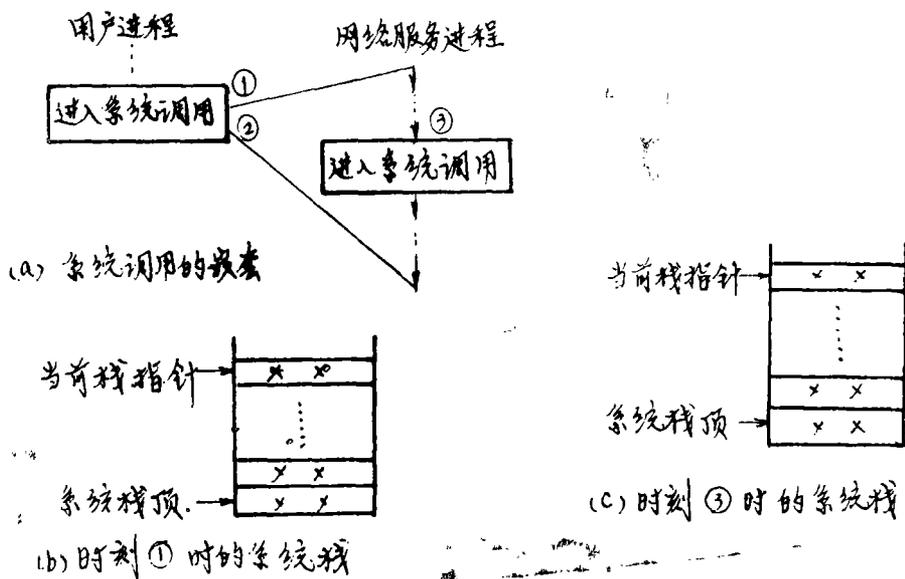


图 10 系统调用再入时对系统堆栈区的破坏

从原栈顶开始用起。前一次系统调用时写入系统栈的内容将被后一次系统调用覆盖，如图 10(b)、(c)所示。在图 10(b)、(c)中系统栈指针是处于相同位置的，但内容不同，我们分别用‘×’和‘*’来表示二者的区别。对于位于绝对地址上的工作区，也有类似的问题。

为了解决系统调用的再入问题，我们详细地分析了 PC-DOS 对系统堆栈的管理以及各系统调用时对系统堆栈的使用情况。我们发现，在进行系统调用时，尽管系统堆栈的空间有 160 即 AOH 字节长，但实际最多只用了不到 60 个字节。于是，我们专门编制了一段二次系统调用的处理子程序，在二次系统调用之前，先保护第一次系统调用时，对用户堆栈指针的保持单元（这也是绝对定位的）以及一些可能破坏的系统工作区。然后，修改系统代码，将系统堆栈的栈顶位置上移 80 个字节，即 50H，如图 11(a)、(b)所示。这样，二次进行系统调用时，系统栈顶被定位到了图 11(b)所指示的位置。两次操作系统调用所使用的系统栈区就被分开了。在二次系统调用返回后，使系统栈顶复原。采用这样的措施，系统调用的再入就可能了。

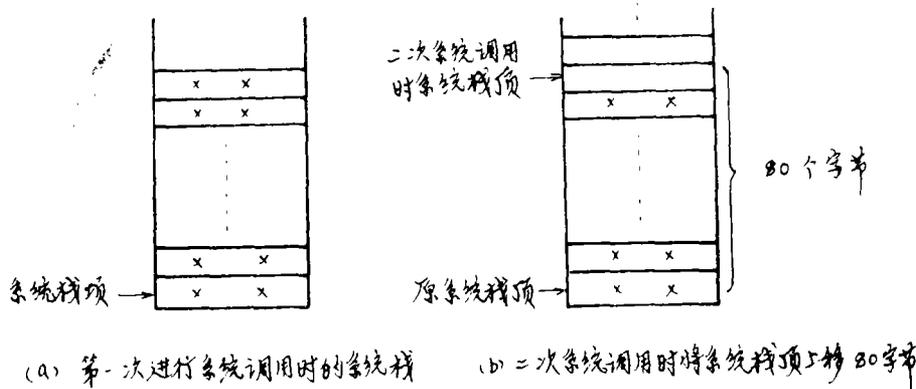


图 11 二次系调用时，对系统栈的处理

(下转第 63 页)

解决这个问题的。图中 CPU 设置为定时中断方式，时间长度定为接收一帧所需的最大时间间隔。在 CPU 处于等待状态，CPU 的 ZC/T 引脚有效(高电平)时，不管 SIO 的 W/RDY 引脚的状态如何，都可通过触发器 S 端将触发器 Q 端拉回到高电平，即 CPU 的 WAIT 脚回复到高电平来把 CPU 从等待状态中解脱。

图 5 中的驱动接收电路由差分长线驱动接收器 SN 75108、SN 75109 芯片组成。这一对芯片是把 TTL 标准电平转换成极性相反的差分信号和把极性相反的差分信号转换成标准 TTL 电平。使用双绞线时，传输距离可达公里级。该芯片还有以下特点：

1) 信号传输过程中，外界干扰在双绞线上只表现为共模信号，而差分放大器具有较高的

共模抑制力，抗干扰力很强。

2) 驱动接收器的输入输出均和 TTL 相容，使用很方便。

3) 驱动接收器都具有“选通”、“禁止”功能，能方便地控制通道开闭，很适于多点网络。

五、结 束 语

用于集散系统中的数传网经实践证明，具有成本低、可靠性高、扩充容易、纠错功能较强、操作简便等优点。该成套设备系统 1986 年已通过部级鉴定，受到有关方面专家的好评。现在我们正在 IBM-PC 系列机上开发同类型的数传网络，以满足生产实际的更高要求。

(上接第 41 页)

八、PC-DNOS 对用户操作员和用户程序员的支持

PC-DOS 的所有外部命令在 PC-DNOS 中全部有效。除此之外，我们专门为 PC-DNOS 操作员设计了若干实用程序，主要有下列六个：

List. com——列出本地和其它当前已入网的工作站站名及已入网的工作站数；

Mask. com——屏蔽本地工作站的一般显示服务；

Unmask. com——开放本地工作站的一般显示服务；

Note. com——通知发送；

Fsend. exe——文件发送；

Msend. com——消息发送。

操作员利用这些实用程序，可以实现很多有用的功能。

PC-DNOS 的特点，更多地是表现在它对用户程序的支持。除了本文在前面已给出的网络服务源语之外，PC-DNOS 还支持不同工作站上的用户进程之间的通信。进程间通信通过管道进行。用户进程可随时建立一个通往其它一个或几个工作站的管道，然后，将信息通过该管道，送往目标进程或通过管道提取来自其它工作站用户进程的信息。显然，在 PC-DNOS 的支持下，可以实现分布式处理的功能。

在目前阶段，PC-DNOS 仅支持汇编语言一级的用户程序，待进一步对系统编译和解释程序进行有关的开发后，便可直接支持高级语言程序。

参 考 资 料(略)