1024
TP393.1
W57

Netware 高级编程接口与核心技术

温 锦 山 编著

本书附盘可从本馆主页 http://lib.szu.edu.cn/
上由“馆藏检索”该书详细信息后下载，
也可到视听部复制

北京航空航天大学出版社

内容简介

本书共九章，介绍了 Novell 公司 Netware 网络操作系统的系统原理、应用编程接口及网络安全技术等，其中还涉及到 Novell 公司的核心技术！书中重点讲述了 DOS、Windows 95 及 WindowsNT 不同开发平台的 Netware 编程接口技术，并配有详细的汇编、C/C++、Visual Basic、Delphi 程序设计实例，最后还列举出黑客袭击 Netware 网络的常用手段及防范措施。

本书所介绍的接口技术都是作者用 Soft-ICE 跟踪调试、运用逆向工程的方法总结而来。本书附带的光盘包括 C/C++ for DOS/Windows 接口函数库、Netware 客户动态链接库 DLL、Visual Basic 和 Delphi 的接口声明模块以及书中所有的源程序。

本书可供系统分析员、Netware 系统管理员、对底层通信感兴趣的人员以及大专院校有关专业的师生参考。

图书在版编目(CIP)数据

Netware 高级编程接口与核心技术/温锦山编著．
北京：北京航空航天大学出版社，2000.8
ISBN 7-81012-993-7

Ⅰ.N... Ⅱ.温... Ⅲ.局部网络-操作系统(软件)
,NetWare Ⅳ.TP393.1

中国版本图书馆 CIP 数据核字(2000)第 32817 号

Netware 高级编程接口与核心技术
温 锦 山 编著
责任编辑 陶金福
*
北京航空航天大学出版社出版发行
北京市学院路 37 号(100083) 发行部电话 82317024
http://www.buaapress.com.cn
E-mail:pressell@publica.bj.cninfo.net
河北省涿州市新华印刷厂印装 各地书店经销
*
开本：787mm×1092mm 1/16 印张：32.5 字数：894 千字
2000 年 9 月第 1 版 2000 年 9 月第 1 次印刷 印数：4 000 册
ISBN 7-81012-993-7/TP·407 定价：58.00 元

前言

网络技术的发展影响到社会的许多方面，在现代化的企业、各种事务机构中，技术人员总是在寻求最佳的网络解决方案。Novell 公司的 Netware 网络操作系统采用客户服务器结构，具有文件服务、目录服务、计账服务、队列服务、网络打印等强大的功能，能够与 UNIX、DOS、Macintosh、Windows 以及 OS/2 系统友好平滑地连接，并且使用方便、维护简单，曾一度主宰着微机网络的发展，已经在世界上得到广泛地安装和应用。

写作动机

笔者于 1997 年暑假着手编写一个基于 Netware 文件管理功能的无须维护的电子邮件系统。当时，我们机房的几位管理员在一起学习软件、编写程序，由于时间分配不同的缘故，为了讨论出现的问题，往往好几天都无法联系上。我们的网络是 Netware 无盘工作站，没有 Microsoft E－mail，只好在大黑板上留言。后来虽然发现 Netware 也有电子邮件功能，但是它不支持 1 000 以上的用户，对于我们这个 2 500 个用户的 Netware 来说，是个致命的打击；况且 Netware 电子邮件的界面不友好，不时冒出的两句英文让人“丈二和尚摸不着头脑”，不利于推广；另外，它不支持在线谈话，对于我们一个网络两个机房分居两地来说，也是个缺陷……总之，要找到一个适合我们网络的功能完善的电子邮件系统实在不易。因此，开发一个能够满足我们需要的电子邮件系统迫在眉睫。

可是，没有必要的技术支持，从何编起？Netware 的 SEND.EXE 是如何工作的？邮件怎样才能发出去？邮件的隐私问题怎么解决？邮件又怎么接收？邮局设置在哪里？MAP.EXE 是如何工作的？怎样才能把邮局映射为逻辑驱动器？

一系列的问题，加上毫无技术支持，迫使我不得不拿起看家本领——反汇编。

众所周知，Netware 广泛的应用已成为事实。在使用 Netware 的过程中往往需要一些高级的系统维护工具，在应用程序中需要一些底层的技术调用来完成某些功能。然而，市面上的书往往局限于网络的概念、Netware 网络操作系统的特点、安装和使用，对于 Netware 系统的内核及底层中断调用的具体方式，则避而不谈，涉而不深。

现在，绝大部分中断都已经破译，也编写了大大小小几百个实用程序。为何不把这些资料公开出来，供大家分享？这就是我写这本书的动机。

本书适合的对象

本书可供系统分析员、Netware 系统维护员、对底层中断感兴趣的人员以及大专院校有关专业的师生参考。为了更好地阅读本书，最好具备一定的汇编语言或 C 语言基础，对 Netware 亦应有一定的认识。

准备工作

为了实践这些 Netware 功能调用，必须有一台 Netware 客户机，并安装必要的工具软件：

1. 调试工具——DEBUG、SDEBUG 或 TDEBUG；
2. 高级调试工具——Soft－ICE 或 TR；
3. 编译器——MASM 或 TASM、TC 或 TC++。

作　者

2000.5

第1章 Netware系统概述

计算机网络是计算机技术和数据通信技术发展的产物。所谓计算机网络就是利用通信设备和通信线路将不同地理位置、功能独立的多个计算机系统连接起来，以实现数据通信和资源共享的系统。

计算机网络根据其覆盖范围的大小可分为广域网（wide area network）和局域网（local area network）。局域网一般分布在几百米到几公里的范围内，常见于一幢大楼、一个工厂或一个企业；而广域网的距离则大得多，可以通过电话线、公用数据网、卫星网覆盖很广的范围。

计算机网络的功能体现在三个方面：

（1）硬件资源的共享（网上任何一台微机或终端可以访问主机）；

（2）软件资源的共享（网上任何一台微机或终端可以访问主机上的各类数据库和其他应用软件）；

（3）用户之间的信息交换（E-mail）。

Netware系统包含三大部分：

（1）网络操作系统——这是Netware系统的管理核心，提供开放的网络操作服务，负责管理网络数据库访问的同步、文件和目录服务等，并支持不同软、硬件平台的文件服务器；

（2）工作站接口——一种Shell软件，在网络通信和应用软件之间提供通信接口，使它们之间的网络通信连接标准化；

（3）网桥（或路由器）——实现网络互联，又可分为本地网桥和远程网桥，每个网桥可以把4个不同的网络互连起来。

1.1 网络操作系统

Netware网络操作系统是一个面向多用户的多任务管理系统，其最重要的特点是基于模块化设计的开放系统结构，可以对其方便地进行扩充。Netware的内核结构由系统执行程序、扩展文件系统和Netware可安装模块（NLM）组成。

Netware网络操作系统具有文件服务、打印服务、管理服务、通信服务、Web服务、数据库服务等。

Netware不但具有双重目录和文件分配表、热定位和写后读验证、磁盘镜像、磁盘双工、自动归档和恢复功能、Netware事务跟踪系统TTS、不间断电源监控功能等Netware SFT三级数据安全保护措施，还具有用户口令加密、受托管理权、目录屏蔽权、文件属性等数据访问保密设置。

Netware网络操作系统有如下一些优点：

（1）Netware是一种多任务操作系统。所谓多任务是指这种操作系统能把多个程序同时装入服务器内存中，并且被装入的多个程序可以同时处于运行状态，CPU可以为处于运行

状态的不同程序分配时间。

(2) Netware 操作系统具有较高的兼容性。Netware 3.x 能与不同类型的计算机兼容，同时，也能与不同类型的操作系统兼容。

(3) Netware 操作系统具有超级容量和很好的系统容错功能。

(4) Netware 操作系统还具有完备的保密措施。

工作站上发送来的请求，经由媒体，通过网络操作系统（服务器）的网络适配器及 IPX、SPX 模块，进入应用软件模块，由服务器上的大容量硬盘进行读/写操作和在打印机上输出等相应的系统服务。服务器上响应的信息经由原路径，反向地返回到工作站应用程序中去。在 Netware 操作系统中，读/写硬盘的驱动程序没有使用 ROMBIOS，而是在 Netware 核心中配置了硬盘驱动程序，这使得在等待硬盘读/写期间可同时处理网络上其他工作站的服务请求。

1.2 工作站接口

Netware 系统网络结构是客户/服务器方式。工作站是用户应用程序的基本环境。工作站上的应用程序通过 Netware 系统提供的 Shell 软件，与 Netware 网络操作系统（服务器）进行通信。Netware 的 Shell 软件包括：

1. 网络适配器驱动程序

Netware 网络操作系统支持大多数厂家和不同协议的网络适配器，通过不同的网络适配器（网卡，下同）驱动程序，实现与不同硬件设备的平滑的透明连接，并易于实现协议转换。

2. 协议控制程序

由于从计算机网络诞生之日起，国内外已有诸多厂商致力于网络软件、硬件的开发，因而出现了不同通信协议、不同体系结构的网络类型。Netware 网络操作系统支持 IPX/SPX、TCP/IP、AppleTalk 等大部分协议，其中能够实现 Netware 系统服务的只有基于 IPX/SPX 协议的 NCP 服务，最近的 Netware 5.0 还可以选用 TCP/IP 协议作系统通信。通过协议控制，实现网络与应用程序的数据交换。

IPX/SPX 是 Netware 操作系统默认的通信协议。IPX 协议，即网间数据包交换协议（Internet Packet Exchange）是一种快捷简便的通信协议。IPX 协议以最快最简洁的方式向目的站传递数据包，而不管对方是否已经正确地接收，由于发送方不需要等待对方的确认，从而减少了发送过程因等待确认而消耗的时间，所以通信速度加快了。

SPX（Squenced Packet Exchange）协议，即有序数据包交换协议是 IPX 协议的扩展。它要求数据通信的发送者和接收者建立“连接”，保证数据包有序地准确地进行传输。

Netware 系统为了扩展其他服务功能（如 Internet），能与其他协议（如 TCP/IP）共享一块网络适配器进行通信，而不会发生冲突，提供了一个开放的数据接口，简称 ODI (Open Data - Link Interface)。ODI 使得 Netware 支持多种传输协议（包括 IPX/SPX、AppleTalk、TCP/IP），使得网络开发人员能够利用这些开放的接口开发出网络应用程序以及提供更多的软、硬件兼容产品，进而使系统不断扩充。

表 1.1 列出了网络层次的功能与相应工作站服务协议与方式。

此为试读，需要完整PDF请访问：www.ertongbook.com

表 1.1　网络层次、功能与 Netware 工作站

层　次	功　能	Netware 工作站
应用层	特定的功能，如文件传输、系统服务、电子邮件等	Netware 服务
表示层	数据表示和字符编码转换	
会话层	通信的建立和拆除	Netware CoreProtocal
传输层	端－端的可靠的连接	SPX 或 TCP
网张层	数据包在网络上的传输、路由行程选择	IPX 或 IP
数据链路层	帧的传递和差错控制	Open Data Link Interface
物理层	物理传输（Bit 或 Byte）	802．3 或 802．2

使用 ODI 有以下优点：

（1）无须在工作站上增加额外的网络适配器，系统就能够通过支持多种传输协议达到扩充网络的目的；

（2）工作站不必重新引导，就可以利用不同的协议与其他的工作站、文件服务器及主机进行通信；

（3）所有按照 ODI 规范编写的传输协议，在 Netware 网络中都进行透明通信，而且无须追加硬件投资。

3．应用程序接口

它将用户应用程序的各种网络操作请求转换成特定的数据包，并以一定的协议为载体，实现用户应用程序与网络操作系统（服务器）的平滑连接。

Novell 发展了自己独特的客户/服务器协议，即所谓的 Netware 核心协议 NCP（Netware File Service Code Protoal）。NCP 具有几百个服务功能，供用户的工作站系统调用，以完成相应的网络服务。NCP 由一系列的服务协议构成，这些服务协议由客户/服务器模式或主/从关系来确定。根据协议，由用户送到服务器的请求，在服务器中产生响应送回用户。以 NCP 为基础，形成文件和网络所有的服务，包括 Netware 核心服务和增值服务，利用这些服务，构成各种功能的客户/服务器应用程序。NCP 功能包括连接服务、文件服务、目录服务、数据库访问服务、平构库服务、打印服务、计费服务和队列服务等。

工作站上的 Shell，可以利用网络操作系统提供的 NCP 系统功能调用接口，把自身的操作系统映射到网络上，以访问网络上的共享服务和资源。用户工作站 Shell 完成这种映射功能，就等于扩展了工作站本身的操作系统功能。

因此，Netware 工作站上的 Shell 有两种连接网络操作系统（服务器）的方式：

（1）上面所介绍的 IPX/SPX 独享网络适配器的方式，一般由 IPX．COM 和 NETX．EXE（或 NETx．EXE，x 为 DOS 主版本号）驻留内存；

（2）ODI 多协议共享网络适配器是通过 LSL．COM 、NE2000．COM（视网络适配器种类而定）、RPLODI．COM（无盘工作站才需要）、IPXODI．COM 、VLM．EXE 驻留内存。

假如工作站是 DOS 工作站，请重新启动 DOS，当出现“Starting MS－DOS．．．”时，按下 F8 进入 DOS 启动菜单，单步执行启动设置，在 AUTOEXEC．BAT 里可发现上述文件。请判断工作站使用哪一种方式。

第2章 接口原理及调试环境

Netware 上 Shell 的工作原理是：用户命令由 Shell 解释器区分为 DOS 部分和网络部分：DOS 部分由 COMMAND. COM 进一步解释；网络部分由 Shell 进一步转换成对网络文件服务器请求格式（NCP），然后通过 SPX、IPX、网络适配器驱动程序发送到网络线路中去，再经网络操作系统（服务器）执行处理后，将结果送回相应的工作站。工作站将接收到的网络信息包以相反的过程进行解释，最终以特定的格式反馈给用户。

例如：当应用程序需要打开一个“C:\WS.DAT”的文件，如果 C:是本地盘，则 Netware 的 Shell 将什么都不干，DOS 内核以常规的方式去工作；如果 C:是网络盘，则 Netware 的 Shell 将以“特定的数据格式”向服务器发出请求，服务器反馈回成功与否……所有的一切都由 Shell 作通路转换，应用程序不必知道 C：盘的属性，从而实现透明的平滑的连接。

经过深入的跟踪和分析，发现：

（1）在 DOS 下，Netware 的 NCP 系统功能的实现主要是依靠扩展的 INT 21 中断调用或者 ODI 远过程（Call far）调用来实现。

（2）在 Windows 95/NT 下，Netware 的 NCP 系统功能的实现主要是通过调用 DLL 动态链接库来实现。

因此，为了便于跟踪调试，深入系统内核，特别推荐两个高级调试工具。

2.1 Soft－ICE 使用说明

1．调试工具 Soft－ICE

欲善其事，必先利其器。随着微机芯片的不断升级，在调试工具的设计上也应采用新的技术来提高跟踪能力。Soft－ICE 是 80386 芯片出现后调试工具发展上的一个重大革新。

80386 芯片是 Intel 公司 1985 年推出的产品，它与早期的 8086 和 80286 芯片在目标代码一级上完全保持了向上的兼容性。除了在总线宽度、运行速度等方面有了较大提高以外，最主要的是提供了完善的多任务自理功能。出现了三种运行程序的方式，即保护方式、虚拟 8086 方式和实方式。Soft－ICE 是将程序置入虚拟 8086 方式下运行的。Soft－ICE 使用具有 80386 保护方式的特性，实践证明，很多用 DEBUG、Turbo DEBUG 难以调试的程序在 Soft－ICE 下都被轻松地解开。

Soft－ICE 是一个非常流行的工具软件，特别适用于软件的加密和解密。Soft－ICE 目录中应当含有下列文件：

- S－ICE.EXE　为 Soft－ICE 主程序(或 S_ICE.EXE)；
- S－ICE.DAT　为 Soft－ICE 启始参数设定文件(或 S_ICE.DAT)；
- LDR.EXE　为 Soft－ICE 主程序及符号文件(symbol file)的载入程序；
- MSYM.EXE　为 Soft－ICE 符号文件的产生程序；

- EMMSETUP.EXE　供使用者自定扩展内存(expanded memory)使用方式的程序;
- UPTIME.EXE　时间校对程序(Soft－ICE 文档中没有提及),经分析,该程序可补偿因进入调试状态而造成的系统时间滞后;
- README.SI　帮助文件;
- SAMPLE.EXE　为使用者指南中所使用的一个示范(DEMO)程序;
- SAMPLE.ASM　为 SAMPLE.EXE 的组合语言原始程序;
- SAMPLE.SYM　为 SAMPLE.EXE 的符号文件。

2. 调试工具 Soft－ICE 的特性

(1) 在内存读/写方式下、读/写端口时、内存范围内及中断时设置断点。

(2) 通过 80386 保护方式隔离地调试代码,可以防止错误的程序修改或破坏 Soft－ICE,即使在 DOS 被毁坏时,仍能正常工作。

(3) 具有软件引导功能,允许对自启动软件或非 DOS 操作系统进行调试。

(4) 返回到过去跟踪的范围中。这一点有助于确定软件在达到断点之前是如何运行的,因而可以达到"由果求因"的逆向调试程序的功能。

(5) 不使用任何 DOS 中断,可以对操作系统进行调试。由于利用了硬件中断寄存器,还可以对 ROM 中的代码进行调试。

(6) 可以和其他调试工具同时调试程序。

(7) 本调试工具可在任何时刻呼叫出来,即使键盘中断被改。

2.1.1　装入 Soft－ICE 的方法

有两种方法启动 Soft－ICE 程序,一种是在 DOS 提示符下键入程序名直接运行,另一种方法是将其作为一个设备驱动程序放在 CONFIG.SYS 文件中。

S－ICE.EXE 必须在其他会配置 Extended memory 的程序之前载入(例如:VDISK.SYS、RAMDRIVE.SYS)。通常如果 Soft－ICE 在 CONFIG.SYS 中第一个载入时,能得到最好的效果。在 CONFIG.SYS 中用

```
device = drive: \ path \ S－ICE.EXE /SYM 1024 /TRA 1024
device = drive: \ path \ HIMEM.SYS
.....
```

Soft－ICE 当作第一个载入的程序是很明智的。其中 Drive 及 path 须指向 S－ICE.EXE(或 S_ICE.EXE)所在的目录。

建议在 CONFIG.SYS 中载入 Soft－ICE 后,不要再加载 EMM386 之类的程序! 也许找到的 Soft－ICE 的主文件是 S_ICE.EXE,这是低版本的 Soft－ICE,不能调试保护模式的应用程序。

在 Netware 网络中,只须要把 S－ICE.EXE 和 S－ICE.DAT 文件加入到镜像文件 NET$DOS.SYS 中去。

2.1.2　加载程序的方法

第一个方法是直接运行程序。当认为执行到关键的地方后(如输入密码),用热键(Ctrl＋D 或 Alt＋D 等)激活 Soft－ICE,再设置断点;或者事先设置好断点,然后运行程序,当触发了

相应的断点后，Soft－ICE 自动被激活。另外，还可以用 LDR programname 来加载程序。

2.1.3 进行符号调试的方法

在编译高级语言运行 link 程序时加上/MAP 选项，然后利用 Soft－ICE 中的配套程序 MSYM 将 *.MAP 文件转换为 *.SYM 文件，其格式为 MSYM programname[.exetension]；或者在像 C 这样的高级语言中，带/Debug 开关(在 C 集成环境的菜单中可以选择)编译源代码，用 LDR programname[.exetension]时将有意外的惊喜。

2.1.4 指令分析

Soft－ICE 与 DEBUG 的指令相比，有几个指令（如 U、A、G、T、P、S、F、M、C、R）是一样的，其他的指令就比 DEBUG 强得多了。Soft－ICE 的操作指令如下：

1．进入 Soft－ICE 状态

加载 Soft－ICE 后，有两种方法进入 Soft－ICE：首先是用热键（Ctrl＋D 或 Alt＋D 等）激活 Soft－ICE；其次是用 LDR programname 来加载被调试的程序。

注意：热键可以用 ALTKEY 命令更改。

2．退出 Soft－ICE 状态

使用 X、EXIT 或 Soft－ICE 的热键都可以回到原先的画面（返回 DOS）。

3．窗口的调整

进入 Soft－ICE 状态后，应该是全屏的；否则可以编辑 S－ICE.DAT 文件，来更改窗口的大小和颜色。

Soft－ICE 默认有四个窗口：寄存器窗、数据窗、代码窗和命令窗。这些窗口都可以随时切换出来或关闭，其中数据窗和代码窗还可以改变大小（高度）。但是，窗口的顺序总是固定不变。屏幕由上而下依次是寄存器窗、数据窗、代码窗和命令窗。

WR ——切换寄存器窗

WC ——切换/设定代码窗之大小

WD ——切换/设定数据窗之大小

EC ——在代码窗和命令窗之间切换

· ——在代码窗显示当前代码

4．BPMB、BPMW、BPMD 设置内存断点

格式：BPM [size] address [verb] [qualifier value] [C＝count]

当程序代码对指定的内存作存取操作时产生断点。

size 可以是 B、W、D，分别代表 Byte、Word、Dword 内存断点。

address 表示地址，由两个 16 位的字组以冒号分隔而组成。第一个字代表区段位址，第二个字代表位移。位址可以由符号或寄存器构成，也可以包括 $ 、. 、@ 等特殊符号。

verb 可以是 R、W、RW、X，分别代表读、写、读或写、执行。

qualifier 可以是 EQ 、NE 、GT 、LT 、M 等。EQ 表示相等，NE 表示不等，GT 表示大于，LT 表示小于，M 表示屏蔽。

count 是经过断点的次数。如：

BPMB 1234：0000 R 表示当对 1234：0000 字节“读”时触发断点。

BPMW DS：0000 RW　表示当对 DS：0000 字“读或写”时触发断点。

BPMW 1234：0000 X　表示当执行 1234：0000 的代码时触发断点。

BPMW 1234：SI W EQ 10　表示当对 1234：SI 字进行“写”，且写入的数据是 10H 时，触发断点。

BPMW DS：1000 W GT 5 C=3　表示当对 DS：1000 字进行“写”，且第三次写入的数据大于 5H 时，触发断点。

BPMW DS：F00 W EQ M 0XXX XXXX XXXX XXX1　表示当对 DS：F00 字进行“写”，且写入的数据是一个最高位为 0，最低位为 1 的数据时，将触发断点。

5．BPR 设置内存范围断点

格式：BPR start－adderss end－address [verb] [C=count]

start－address 、end－address 是内存范围的开始及结束地址。

verb 可以是 R、W、RW、T 或 TW，R、W、RW 分别代表读、写、读或写。使用 T 或 TW 可在指定范围内做回溯追踪（back trace）。它们并不会真正触发断点，而只是记录下代码的信息。这个信息可以用 SHOW 或 TRACE 指令显示出来。如果未指定 verb，默认值是 W。

count 是经过断点的次数。

在某些状况下，设置范围中断点会降低系统的性能。如：

BPR B000：0 B000：1000 W　表示当对 B000：0 B000：1000（单色显示区域）内存范围进行写入操作时，触发断点。

6．BPIO 读写端口时触发中断

格式：BPIO port [verb] [qualifier value] [C=count]

port 表示端口。

verb 可以是 R、W 或 RW，分别表示输入（IN）、输出（OUT）、输入或输出；如果未指定 verb，默认值是 RW。

qualifier 可以是 EQ 、NE 、GT 、LT 、M，其中 EQ 表示相等，NE 表示不等，GT 表示大于，LT 表示小于，M 表示屏蔽。

count 是经过断点的次数。如：

BPIO 21 W NE FF　表示当对端口 21H 写入 FFH 时，触发断点。

BPIO 3FE R EQ M 11XX XXXX　表示当对端口 3FEH 读，且读入的数据最高位是 1 时，将触发断点。

7．BPINT 对指定的中断触发断点

格式：BPINT int_number [< AL | AH | AX > = value] [C=count]

BPINT 命令可以在指定中断时触发，兼借指定 AX、AH 或 AL 寄存器的值可以轻易分离指定功能的 DOS 或 BIOS 中断（本书中的绝大部分接口技术的调试和破译都归功于本命令）。

int_number 指定中断号码。

value 是在产生指定中断时同时满足的条件。当中断发生时指定的 value 值将和指定的寄存器比较（AH、AL 或 AX），如果其值和指定的寄存器值相同时，将引发中断；如果没有指定 value 值，将对指定中断都触发。

count 是经过断点的次数。如：

BPINT 21 AH＝4C　表示当 21H 中断且 AH＝4CH 时（结束程序），触发断点。

再如：用热键（Ctrl＋D 或 Alt＋D）激活 Soft－ICE 后，键入“BPINT 21 AH＝3D”和“G”，表示当产生 21H 中断且 AH＝3DH 时触发断点，即设置打开文件断点；在 DOS 提示符下执行其他可执行文件，当可执行文件里有“打开文件”的代码时，Soft－ICE 自动被激活，用“D DX”就可以找出将被打开的文件名。

8．BPX 设置代码中的断点

格式：BPX [address] [C＝count]

address 表示代码的地址。

count 是经过断点的次数。如：

BPX . 1234 表示当执行原始代码的第 1234 行时，触发断点；

BPX 1234：0000 表示当执行 1234：0000 的代码时触发断点。

9．处理断点

Soft－ICE 提供许多命令来处理断点，包括显示、修改、删除、启动和禁止断点。断点是以断点号码来识别的（从 0 开始）。断点在设置后不会自动消失，必须以 BC 或 BD 指令来消除或关闭它。Soft－ICE 一般一次可以处理 16（十六进制 10）个中断点（视具体版本而定），同种形态的断点最多可以有 10（十六进制 A）个。处理断点的指令有：

BD —— 禁止断点。暂时禁止断点的活动，以后可用 BE 命令重新启用。

BE —— 启用断点。当断点第一次定义时将会自动启用，除非用 BD 把它禁止。

BL —— 显示断点，包括显示断点号码、断点条件、断点状态。

BPE —— 编辑中断点，用于修改断点的内容。

BPT —— 把中断点当样板。

BC —— 清除中断点。

语法：BD|BE|BL|BPE|BPT|BC list| *

list 为一串由逗号或空白分开的中断点号码。

* 表示所有断点。

例如：

BD 1，3 表示暂时中止 1 号和 3 号断点。

BE * 表示重新启用所有的断点。

BL 显示所有断点。断点的状态分为启用和禁止。禁止的断点号码之后会有个“*”，BPAND 命令中使用到的启用的断点其断点号码之后会有个“&”，最后一个触发的断点以高亮度显示。

BC * 清除所有的断点

10．CSIP 在断点时代码范围的判断

格式：CSIP [OFF | [NOT] start－address end－address]

本命令常用于设置断点，且略过 DOS 内核或 ROM BIOS 的代码部分。

NOT 表示只有当 CS：IP 超出范围时，才会触发断点。例如：

CSIP NOT F000：0 FFFF：0 表示只有在断点条件成立且 CS：IP 并未指向 ROM BIOS 时才会触发断点。

此为试读，需要完整PDF请访问：www.ertongbook.com

OFF 表示停止对 CS：IP 的判断。

如果没有加参数则会显示目前 CSIP 的范围。

11．BPAND 等待复合断点的发生

格式：BPAND list| * |OFF

BPAND 命令会对两或多个断点做逻辑的 AND 运算。只有当断点条件均“曾经”成立时才会真正触发断点。每次使用 BPAND 命令均会把指定的断点号码加入名单中，直到使用 BPAND OFF 指令为止。可以用 BL 指令列出断点以察看哪些断点号码被复合在一起。被复合在一起的中断点其中断点号码后会有个“&”。

list 是一串由逗号或空白分开的断点号码，表示复合所有的断点。例如：

BPAND 0，2，3 表示将复合 0 号、2 号、3 号断点。只有当三个的条件均成立时才会触发断点。如果 2 号和 3 号的条件均成立一次以上，但 0 号的条件尚未成立，则只有当 0 号的条件成立时才会触发断点。

OFF 清除复合断点。

12．显示及编辑类指令

指令：

U —— 反汇编代码

R —— 显示或更改寄存器

MAP —— 显示内存分布

D —— 显示内存

DB —— 以字节的形式显示内存

DW —— 以字的形式显示内存

DD —— 以双字的形式显示内存

E —— 编辑内存

EB —— 以字节的形式编辑内存

EW —— 以字的形式编辑内存

ED —— 以双字的形式编辑内存

INT? —— 显示最后一次的中断号码

? 或 H —— 显示帮助信息

VER —— 显示 Soft－ICE 的版本

(1) 语　法

格式：U [address] [L [=] length]

address 为开始反汇编代码的地址。如果未指定 address，这个指令会从最后一次反汇编的后一字节处开始反汇编；如果从未使用过反汇编指令，则从目前 CS：IP 开始。例如：

U $ －10　表示从目前地址的前 10H 字节处开始反汇编。

U . 349　表示从 349 行开始显示原代码。

length 为要反汇编代码的长度。

(2) 语　法

格式：R register－name [[=] value]

register－name 为 AL 、AH 、AX 、BL 、BH 、BX 、CL 、CH 、CX 、DL、DH 、

DX、DI、SI、BP、SP、IP、CS、DS、ES、SS 和 FL 中的任一个。

Value，如果 register－name 不是 FL，value 是个十六进制的值或表达式。若是 FL，value 是下列标志符号：

O —— Overflow flag	溢位标志
D —— Direcrion flag	方向标志
I —— Interrupt flag	中断标志
S —— Sign flag	正负号标志
Z —— Zero flag	零值标志
A —— Auxiliary carry flag	辅助进位标志
P —— Parity flag	极性标志
C —— Carry flag	进位标志

中的一个或多个的组合，并在标志符号前面加上“+”或“－”。

例如：

R AH 5	表示把 AH 寄存器的值改成 5。
R FL = O Z P	表示切换 O、Z、P 标志的值。
R FL	显示目前标志的值并可以修改其值。
R FL O +A －C	表示切换 O 标志，设置 A 标志并关闭 C 标志。

(3) 语　法

格式：MAP

MAP 指令显示各内存部分的名称、位置和大小。CS：IP 所属的部分会以高亮度显示。

(4) 语　法

格式：D [size] [address] [L [=] length]

size 为 B、W、D 中的任一个，分别表示字节、字、双字。

例如：

D DS：00	表示显示当前数据段。
DB DS：00 L＝8	表示以字节形式显示当前数据段的前 8 个字节。
DW 00 L＝8	表示以字形式显示当前数据段的前 8 个字。

(5) 语　法

格式：E [size] address [data－list]

SPACE 光标移至下一个元素上。

TAB 在数字区和 ASCII 区间切换。

ESC 或 Enter 离开数据窗。

size 为 B、W、D 中的任一个，分别表示字节、字、双字。

例如：

E 1234：0 表示编辑从 1234：0000 字节开始的内存。

EB 6000：0 "FIRST"，0D 表示从 6000：0000 开始的值以“FIRST”字串和一个字节代替。

(6) 语　法

格式：INT?

显示最后一次的中断号码。

(7) 语　法

格式：< ? | H > [command | expression]

? 和 H 两指令等效，都可以显示帮助信息。如果未指定参数，将会分屏幕显示所有指令的简单帮助。按任意键以继续显示或按 ESC 键离开帮助说明。

若有指定参数则会显示包括指令语法及范例的详尽说明。

若加上运算表达式，则会计算并以十六进制、十进制及 ASCII 字节显示其结果。

(8) 语　法

格式：VER

显示 Soft - ICE 的版本。

13. I/O 端口指令

指令：

I 、IB —— 由端口输入字节

IW —— 由端口输入字

O 、OB —— 由端口输出字节

OW —— 由端口输出字节

14. 转换控制命令

指令：

X —— 退出 Soft - ICE 状态

G —— 执行到某地址

T —— 追踪一条程序代码

P —— 单步执行程序

HERE —— 执行到光标当前行的代码

GENINT —— 强制执行一个中断

EXIT —— 强制离开当前的程序

BOOT —— 重新启动系统（保留 Soft - ICE）

HBOOT —— 重新启动系统（完全重启）

(1) 语　法

格式：GENINT INT1|INT3|NMI| interrupt - number

interrupt - number 表示中断号（00～FF）。

GENINT 命令会强制发生一个指定的中断。当 Soft - ICE 和另一个调试器共用时，可以用这个功能把控制权交给另一个调试器。

(2) 语　法

格式：EXIT [R] [D]

R 为恢复中断向量表，把中断向量表还原成最后一次储存的状态。

D 表示清除所有中断点。

15. 常用指令

指令：

A —— 汇编程序码

S —— 搜寻数据

F —— 将数据填入内存

M —— 搬移数据

C —— 比较两内存块

16. 特别的调试命令

指令：

SHOW —— 显示在 history buffer 中之程序码

TRACE —— 进入模拟追踪模式（trace simulation）

XT —— 在模拟追踪模式中进行单步执行

XP —— 在模拟追踪模式中进行程序单步

XG —— 在模拟追踪模式中执行到某位址

XRSET —— 重设回溯追踪缓冲区（back trace buffer）

VECS —— 储存/还原/比较中断向量

SNAP —— 拍下内存区段的快照

EMMMAP —— 显示 EMM 分配图

(1) 语　法

格式：SHOW [B|start _ line]

在调用本功能之前，必须先用 BPR start - adderss end - address [T/TW] [C = count] 设置内存范围断点，并执行该范围内的程序代码；否则将得到“Back Trace Buffer Empty”的错误信息。

B 表示从回溯追踪缓冲区中最早的程序代码开始显示。

Start _ line 表示从缓冲区中倒数第多少条程序代码开始显示。

SHOW 命令可以用上、下、PageUp 、PageDown 等键来卷动。按 Esc 键离开 SHOW 状态。

例如：

SHOW 40 表示从回溯追踪缓冲区中倒数第 40 个程序代码开始显示。

SHOW B 表示从回溯追踪缓冲区中最早的程序码开始显示。

(2) 语　法

格式：TRACE [start _ line] | [OFF]

TRACE 指令可以把回溯追踪缓冲区中之程序代码像第一次执行的情形一样再重播一次。进入模拟追踪模式后，可以使用 XT 、XP 和 XG 指令来追踪缓冲区中之程序码。还可以输入 TRACE OFF 以离开模拟追踪模式。未加参数的 TRACE 指令会显示目前模拟追踪模式是 ON 或 OFF。

在调用本功能之前，必须先用 BPR start - adderss end - address [T/TW] [C = count] 设置内存范围断点，并执行该范围内的程序代码；否则将得到“Back Trace Buffer Empty”的错误信息。

Start _ line 表示从缓冲区中倒数第几条程序代码开始追踪。

OFF 表示离开模拟追踪模式。

(3) 语　法

格式：XT [R]

R 表示反向。

注意：在模拟追踪模式中单步执行不会改变除 CS、IP 以外的寄存器的值。

(4) 语　法

格式：XP

注意：除 CS 、IP 之外的寄存器的值均不会改变。

(5) 语　法

格式：XG [R] address

R 表示反向。

address 表示回溯追踪缓冲区中欲执行到达的地址。

(6) 语　法

格式：XRSET

XRSET 指令会重设（清空）回溯追踪缓冲区。

(7) 语　法

格式：VECS [C|S|R]

C 表示比较目前的中断向量表和储存起来的中断向量表。

S 表示储存当前中断向量表。

R 表示由缓冲区中还原（恢复）中断向量表。

载入 Soft－ICE 时，中断向量表会被自动储存起来。当程序以 LDR．EXE 载入时也会自动储存向量表。每次执行 VECS S 时，上一份备份的中断向量表会被覆盖掉。

如果没有加参数，则会显示整个中断向量表。

(8) 语　法

格式：SNAP [C|S|R] address1 address2

C 表示比较缓冲区和内存范围。

S 表示把内存范围存到缓冲区中。

R 表示从缓冲区还原内存范围。

注意：要使用 SNAP 指令，最好要在 CONFIG．SYS 中 S－ICE．EXE 一行末尾加上 /TRA XXXX 参数（XXXX 为缓冲区大小，单位数为 K)。

例如：

SNAP S 1234：0 2000：0 把从 1234：0 至 2000：0 的内存存入 Soft－ICE 的回溯追踪缓冲区。

(9) 语　法

格式：EMMMAP

EMMMAP 命令会显示 EMM 内存中每一个可取得的 page 及目前映射到的 page。

17．调试器设定命令

指令：

PAUSE —— 显示满一个屏幕后暂停

ALTKEY —— 设定 Soft－ICE 的启动热键

FKEY —— 显示、修改功能键

BASE —— 设定/显示当前的进制数

CTRL + P —— 把 LOG 送到印表机

Print - Screen —— 印出目前屏幕

PRN —— 设定打印机的输出端

(1) 语 法

格式：PAUSE [ON|OFF]

如果没有指定任何参数则会显示目前 PAUSE 的状态。

PAUSE 命令会在每一页的结束时暂停屏幕。如果 PAUSE 设为 ON，Soft - ICE 会提示按任意键以继续卷动窗口。提示信息会显示在屏幕底部的状态行里。

PAUESE 的默认值是 ON。

(2) 语 法

格式：ALTKEY [ALTletter] | [CTRLletter] | [SYSREQ]

letter 可以是任何一字母 (A～Z)。

如果没有指定参数则会显示目前的热键。

ALTKEY 命令可以改变 Soft - ICE 的热键。可以把热键改成 Ctrl + 字母、Alt + 字母或是 SysRq (即 PrtScr) 键。默认热键是 Ctrl + D (或 ALT + D)。

例如：

ALTKEY ALT Z 指定 Ctrl + Z 是 Soft - ICE 的热键。

(3) 语 法

格式：FKEY [function - key - name string]

function - key - name 可以是 F1、F2…F12 中任一个。

string 包含任何 Soft - ICE 的命令和特殊字元："^" 及 ";"。"^" 是用来让命令不显示出来，";" 则代表按下 Enter。String 是空白字串时，可以取消某个功能键。

如果没有指定参数则会显示目前各功能键代表的指令。

也可以在设定档 S - ICE. DAT 中预先指定功能键的功能。

例如：

FKEY F2 ^WR；设定 F2 键代表切换寄存器窗的指令。

FKEY F1 WD 3；D DS：100；设定 F1 键代表一串指令。

(4) 语 法

格式：BASE [10 | 16]

BASE 指令是用来设定当前进制数是十进制或十六进制。

进制数的默认值是十六进制。

(5) 语 法

格式：CTRL + P

把所有显示在命令窗中的信息都送到打印机去。若要停止把 LOG 送到印表机的动作，只须再按一次 CTRL + P 即可。

当须要把许多数据输出到打印机时，可以把 PAUSE 设为 OFF，这样数据可以一直卷动下去而不须要去按键。

(6) 语 法

此为试读，需要完整PDF请访问：www.ertongbook.com