

Remote Access for Cisco Networks

# Cisco

## 网络

## 远程访问

(美) Bill Burton 著  
戴 锋 译



机械工业出版社  
China Machine Press



Education

Cisco专业技术丛书

# Cisco网络远程访问

(美) Bill Burton 著

戴 锋 译



机械工业出版社  
China Machine Press

本书介绍实现远程连接的基本技术。内容包括：异步终端服务、高级终端服务、点对点协议、IP拨号访问、按需拨号路由、综合业务数字网、IP路由协议、AS5200高端访问服务器、WAN服务、WAN连接备份、高级安全与TACACS+和RADIUS、协议转换、网络地址转换、故障检测技术等。本书还提供了大量配置实例和故障检测技术。

本书适于网络管理员，以及需要掌握远程连接知识的技术人员参考。

Bill Burton: Remote Access for Cisco Networks(ISBN 0-07-135200-7)

Copyright © 2000 by the McGraw-Hill Companies, Inc.

Authorized translation from the English language edition published by McGraw-Hill, Inc.

All rights reserved. For sale in the People's Republic of China.

本书中文简体字版由机械工业出版社和美国麦格劳-希尔国际公司合作出版。未经出版者书面许可，不得以任何方式复制或抄袭本书内容。

版权所有 侵权必究

本书版权登记号：图字：01-2000-1717

### 图书在版编目(CIP)数据

Cisco网络远程访问/(美)伯顿(Burton,B)著,戴锋译.—北京:机械工业出版社,2001.9  
(Cisco专业技术丛书)

书名原文 Remote Access for Cisco Networks

ISBN 7-111 09182-5

I C II ①伯 2 戴 III 远程网络 Cisco产品 IV TP393.2

中国版本图书馆CIP数据核字(2001)第056453号

机械工业出版社(北京市西城区百万庄大街22号 邮政编码 100037)

责任编辑 李冬冬 张鸿斌

北京市密云县印刷厂印刷·新华书店北京发行所发行

2001年9月第1版第1次印刷

787mm×1092mm 1/16·25印张

印数:0 001 5 000册

定价:43.00元

凡购本书,如有倒页、脱页、缺页,由本社发行部调换

# 前言

很感谢你选择了本书。在这本书中，我收录了多年的实践经验以及所教授学生的众多反馈信息，期望对你建立远程访问连接能有所帮助。

书中有许多实例，一般性地描述了建立远程网络用到的一些最好的实践经验。每个例子都完整地给出了设置文档，以及相应的帮助进行验证和故障检测的执行命令。

本书是一个实践指南，我希望它能够为你提供价值，并作为你每天携带使用的参考书。

现在市场上出现了许多书籍，它们的主要目的都是帮助读者通过CMTD或者BCRAN认证考试。本书涵盖了这些书籍的内容，同时扩展了建立远程访问网络实用性的一面。

在我开始编写本书时，400页看起来似乎是一个艰巨而难以完成的任务，哪儿来那么多的材料呢？在把材料收集起来之后，竟然发现有将近800页。从这些材料中挑选出本书要讨论的技术是一件很困难的事情，但我让我的学生们来决定。书中大部分的论题都是基于Cisco课程资料之外所需要阐述的领域来选取的。

书中前面的几章主要更深入地阐述了基础技术知识，后面的各章则集中讲解在架构各个部件中这些基本技术的使用，以及更高级的论题。

**第1章 “异步终端服务”** 该章讨论异步通信的基本知识，然后介绍一个终端拨入连接的实例，该例将带着你浏览某些基本的Modem设置。

**第2章 “高级终端服务”** 该章对第1章中的基本知识进行了扩展，讨论了用telnet建立虚拟终端话路，通过设置菜单和使用 autocommand来简化用户访问，以及设定某些基本的访问安全措施。最后一节描述了把某访问服务器设置为一个系统控制器的处理过程，并用反向远程登录技术作为连接到远程站点设备控制台端口的方式。

**第3章 “点对点协议”** 该章介绍了PPP的组成部件，从物理层的链路控制协议到网络控制协议的认证。

**第4章 “IP拨号访问”** 从第4章开始，我们进入了TCP/IP舞台，首先介绍异步拨入PPP连接。其中有许多实例，有一个实例用一台PC带两个Modem，建立一条PPP多重链路连接，从而通过一台访问服务器获取对网络的IP访问。

**第5章 “按需拨号路由”** 这一章带领读者浏览了一个报文从进入路由器开始一直到它被发送到拨号连接上的整个过程。

**第6章 “综合业务数字网”** 用第5章中学习的基本DDR技术，我们将在这一章中浏览使用静态路由的ISDN连接。其中包含了用于一对一BRI、一对多BRI、以及BRI对PRI连接的实例。在对基本知识有所了解的基础之上，进一步介绍了用Dialer Profiles对拨号处理进行的扩展。

**第7章 “DDR与IP路由协议”** 在大型的拨号网络中，用静态路由指定到目的端的路径，对于管理来说是非常可怕的事情。在考察了以静态路由到主要网络环境对称部分的传播复制实现路由器重分布之后，我们将用以下方式把路由协议扩展到远程站点：对RIP和IGRP的瞬时路由、

对OSPF的命令查询回路以及对EIGRP的按需路由（ODR）。

**第8章 “AS5200高端访问服务器”** 该章把前面所学的所有知识结合起来并用于一台AS5200访问服务器上。通过单一的ISDN PRI连接，我们可以获得异步终端连接、异步IP/PPP连接以及常规ISDN连接。要获得输入语音级呼叫，还需要其他的硬件，如集成数字Modem。该章还有一节特别用来讨论对这些内部数字Modem进行管理和故障检测。

**第9章 “WAN服务”** 该章讨论了在WAN连接中基本的同步连接项：HDLC、Cisco默认封装方式、串行链路的变种、信道化T1/E1回路的设置、X.25以及帧中继。

**第10章 “备份固定的WAN连接”** 该章考察了提供拨号备份服务的三种方法：备用接口的使用、不固定静态路由以及IOS 12.0版本中的一个新特性——拨号观测（Dialer Watch）。

**第11章 “高级安全与TACACS+和RADIUS”** 该章简单浏览了Cisco Secure ACS产品。随后介绍在一台Cisco AS5200中实现AAA的实例，在实例中，将用认证、授权和计费来验证和授权对AS5200的访问。授权处理还实现了对执行级命令多特权级的分配。

**第12章 “协议转换”** 我们如何能够远程登录到一台DEC VAX或者一台X.25服务器？那就是协议转换。该章讨论了协议转换的两步和一步方法，并介绍了一些X.25到TCP/IP以及LAT到TCP/IP交谈的实例。

**第13章 “网络地址转换”** NAT的术语很容易混淆，因此首要的事情就是对这些术语进行定义说明。该章还介绍了一个实例，其中用到了重叠的IP地址（源和目的端IP地址相同）以及一个DNS服务器。

**第14章 “故障检测技术”** 最后一章是贯穿本书的故障检测技术的一个复习总结。这一章的最后，还有一节专门讲解全部DDR的处理过程，并给出了相应的流程图。

配置Cisco路由器是一项动手技能。因此，我与Mentorlabs达成一个协议，就是建立实现本书中大部分实例的vLabs。附录H详细说明了本书中实验和vLab号的对应参考。你可以在Chesapeake网站[www.ccci.com/Sooks](http://www.ccci.com/Sooks)上找到实验室配置需求和配置文件。

在本书中有若干实例，由于篇幅的限制，对它们给出了使其正确运行所必要的某些配置详细说明，但也删除了那些对路由器操作影响不大的重复设置命令。我们鼓励读者用自己的设备尝试做这些实验，或者用vLab完成所学的实践知识。

路由器命令行的输出在实例中都给出了注解。输出序列保持了其完整性，并穿插了注解。路由器设置命令行都进行了编号，以便逐行地进行讲解说明。实例中的设备是各种各样的，和现实生活中一样，IOS的版本也不同。

最后，我要再次感谢你选择了本书，你会发现它具有很强的实用性和实践性。

——Bill Burton  
CCIE #1119

# 目 录

## 前言

|                            |    |
|----------------------------|----|
| 第1章 异步终端服务 .....           | 1  |
| 1.1 引言 .....               | 1  |
| 1.2 异步通信 .....             | 1  |
| 1.3 拨号访问设置 .....           | 3  |
| 1.4 超级终端设置 .....           | 9  |
| 1.5 Modem的设置 .....         | 14 |
| 1.5.1 自动设置 .....           | 14 |
| 1.5.2 对话脚本 .....           | 19 |
| 1.6 小结 .....               | 33 |
| 1.7 问题与解答 .....            | 33 |
| 第2章 高级终端服务 .....           | 34 |
| 2.1 虚拟终端话路 .....           | 34 |
| 2.2 菜单 .....               | 39 |
| 2.3 Autocommand过程 .....    | 44 |
| 2.4 Access-Class命令 .....   | 50 |
| 2.5 系统控制器 .....            | 53 |
| 2.6 小结 .....               | 58 |
| 第3章 点对点协议 .....            | 59 |
| 3.1 物理层 .....              | 59 |
| 3.2 链路控制协议 .....           | 59 |
| 3.3 认证 .....               | 62 |
| 3.4 网络控制协议 .....           | 63 |
| 3.4.1 IP控制协议 .....         | 63 |
| 3.4.2 AppleTalk控制协议 .....  | 64 |
| 3.4.3 IPX控制协议 .....        | 65 |
| 3.5 多链路PPP .....           | 65 |
| 3.6 小结 .....               | 65 |
| 第4章 IP拨号访问 .....           | 66 |
| 4.1 Window 95/98拨号设置 ..... | 66 |
| 4.2 Cisco路由器/访问服务器设置 ..... | 71 |
| 4.3 PPP多重链路 .....          | 81 |

|                           |     |
|---------------------------|-----|
| 4.4 故障检测技术 .....          | 95  |
| 4.5 小结 .....              | 95  |
| 第5章 按需拨号路由 .....          | 96  |
| 5.1 路由 .....              | 96  |
| 5.2 兴趣流量 .....            | 96  |
| 5.3 地址解析 .....            | 97  |
| 5.4 认证 .....              | 103 |
| 5.5 综合 .....              | 103 |
| 5.6 故障检测技术 .....          | 124 |
| 5.7 实验 .....              | 124 |
| 5.8 小结 .....              | 124 |
| 第6章 综合业务数字网 .....         | 125 |
| 6.1 用静态路由实现一对一基本连接 .....  | 126 |
| 6.2 用静态路由实现一对多基本连接 .....  | 139 |
| 6.3 基本BRI-PRI连接 .....     | 156 |
| 6.4 Dialer Profiles ..... | 163 |
| 6.5 小结 .....              | 182 |
| 6.6 问题与解答 .....           | 182 |
| 第7章 DDR与IP路由协议 .....      | 183 |
| 7.1 对多重拨入连接的路由重分配 .....   | 183 |
| 7.1.1 London .....        | 183 |
| 7.1.2 NewYork .....       | 186 |
| 7.1.3 Rome .....          | 189 |
| 7.1.4 SOHO800 .....       | 191 |
| 7.2 对距离矢量协议的瞬时路由 .....    | 195 |
| 7.2.1 London .....        | 195 |
| 7.2.2 NewYork .....       | 199 |
| 7.2.3 Rome .....          | 201 |
| 7.2.4 SOHO800 .....       | 204 |
| 7.3 OSPF拨号连接 .....        | 207 |
| 7.3.1 London .....        | 208 |
| 7.3.2 NewYork .....       | 210 |

|                    |     |                                |     |
|--------------------|-----|--------------------------------|-----|
| 7.3.3 Rome         | 214 | 10.5 实验任务                      | 294 |
| 7.3.4 Paris        | 216 | 10.6 小结                        | 294 |
| 7.4 按需路由与EIGRP     | 221 | 第11章 高级安全与TACACS+和RADIUS       | 295 |
| 7.4.1 London       | 221 | 11.1 TACACS+                   | 295 |
| 7.4.2 NewYork      | 224 | 11.2 RADIUS                    | 295 |
| 7.4.3 SOHO800      | 228 | 11.3 基于NT的CiscoSecure ACS v2.1 | 295 |
| 7.5 小结             | 229 | 11.4 AAA操作命令                   | 302 |
| 第8章 AS5200高端访问服务器  | 230 | 11.4.1 认证                      | 302 |
| 8.1 基础             | 230 | 11.4.2 授权                      | 303 |
| 8.1.1 SOHO700      | 231 | 11.4.3 计费                      | 304 |
| 8.1.2 London       | 234 | 11.4.4 London                  | 305 |
| 8.1.3 NewYork      | 237 | 11.4.5 NewYork                 | 307 |
| 8.2 小结             | 245 | 11.5 Exec级测试                   | 317 |
| 第9章 WAN服务          | 246 | 11.6 小结                        | 319 |
| 9.1 高级数据链路控制       | 246 | 第12章 协议转换                      | 320 |
| 9.1.1 NewYork      | 246 | 12.1 X.25交换机                   | 320 |
| 9.1.2 London       | 249 | 12.1.1 London                  | 321 |
| 9.2 信道化T1          | 250 | 12.1.2 NewYork                 | 323 |
| 9.2.1 London       | 250 | 12.1.3 Paris                   | 326 |
| 9.2.2 NewYork      | 252 | 12.2 小结                        | 329 |
| 9.3 X.25报文服务       | 254 | 第13章 网络地址转换                    | 330 |
| 9.3.1 X25交换机       | 255 | 13.1 网络地址转换实例                  | 330 |
| 9.3.2 London       | 258 | 13.2 小结                        | 341 |
| 9.3.3 NewYork      | 260 | 第14章 故障检测技术                    | 342 |
| 9.4 帧中继服务          | 261 | 14.1 物理层                       | 342 |
| 9.4.1 多点帧中继与静态映射   | 262 | 14.1.1 异步连接                    | 342 |
| 9.4.2 多点帧中继与反向ARP  | 267 | 14.1.2 ISDN BRI连接              | 342 |
| 9.4.3 使用子接口的点对点帧中继 | 273 | 14.1.3 同步连接                    | 342 |
| 9.5 小结             | 281 | 14.2 数据链路层                     | 342 |
| 第10章 备份固定的WAN连接    | 282 | 14.2.1 HDLC连接                  | 343 |
| 10.1 备份接口命令        | 282 | 14.2.2 PPP连接                   | 343 |
| 10.2 不固定静态路由       | 282 | 14.2.3 ISDN BRI连接              | 343 |
| 10.3 拨号器监测         | 282 | 14.2.4 ISDN PRI连接              | 343 |
| 10.4 技术应用          | 282 | 14.2.5 X.25连接                  | 343 |
| 10.4.1 London      | 283 | 14.2.6 帧中继连接                   | 344 |
| 10.4.2 Tokyo       | 288 | 14.3 网络层                       | 344 |
| 10.4.3 NewYork     | 291 |                                |     |

|                             |     |                            |     |
|-----------------------------|-----|----------------------------|-----|
| 14.3.1 X.25连接 .....         | 344 | 附录C RFC列表以及其他参考资料 .....    | 354 |
| 14.3.2 ISDN连接 .....         | 344 | 附录D ISDN SPID指南 .....      | 357 |
| 14.3.3 PPP连接 .....          | 344 | 附录E ISDN Q921调试 .....      | 360 |
| 14.4 按需拨号路由 .....           | 344 | 附录F 帧中继交换机设置 .....         | 367 |
| 14.5 小结 .....               | 349 | 附录G CiscoSecure的各种报表 ..... | 373 |
| 附录A EIA/TIA-232物理特性说明 ..... | 350 | 附录H MentorLabs vLab号 ..... | 375 |
| 附录B Modem的AT指令集 .....       | 352 | 术语表 .....                  | 376 |

# 第1章 异步终端服务

## 1.1 引言

首先，我们将简要地概述标准异步通信的概念，然后考察Cisco路由器和访问服务器提供的基本终端服务。本章的实验都很容易建立，用任何具有辅助端口、异步/同步端口或异步通信端口的Cisco路由器都可以完成，在本章最后将展示所有三种路由器的例子。

## 1.2 异步通信

异步通信是怎么工作的？在我们回答这个问题之后，将通过Cisco路由器和访问服务器上一些命令的使用加深理解。

异步数据以串行方式传送，每次一位。如果我们只发送某信息的随机位，收到这些随机位并得到确切的意思将是很困难的。因此在异步通信中，我们以字符的形式组织数据的传送。

数据中的每个字符的开始与结尾都必须用开始位和结束位标明，开始位表示什么时候是数据字节的开始，结束位表示什么时候结束。由于发送这附加的两位，将导致异步通信稍微慢于同步通信；但是，其优点在于：处理器无须应付额外的无用字符。

一根空闲的异步线将用数值1来标明（也称为标记状态）。这个数值表明没有数据在发送，这样，设备就能把空闲状态与断开状态区分开。当传送一个字符时，开始位先被发送，开始位为0（也称为间隔状态），这样当线路从1切换到0时，接收装置被警告一个数据字符将沿线路传送过来。

一旦开始位被送出，传输设备就开始发送实际的数据位，可以是5个、6个、7个或8个数据位，这根据具体情况而定。接收装置和传输设备必须有等同的数据位个数、奇偶性和波特率（参看下面）等属性。几乎所有的设备传送数据都使用7个或8个数据位。要提醒注意的是，当使用7个数据位时，你不能发送ASCII码值比127大的字符。数据被传送完以后，结束位将被发送出去，结束位为1（标记状态），占1、1.5或2位，并且即使先前的数据位也为1，它也能正确地被探测到。结束位完成字符的传送，同时产生一个标记状态（空闲）。

在同步通信中，除开始位和停止位之外，还有一个奇偶校验位和数据一起传送，该位是可选的，奇偶校验位提供了一定的差错检测功能，它可以帮助发现在传送过程中可能发生的数据失真。你可以选择·偶校验、奇校验、标志校验、间隔校验或什么都不选。当使用奇偶校验时，将计算每个数据字节里的标志位（逻辑值为“1”的位）的个数，而且，紧跟数据位将传送一个二进制位来说明：刚刚传送的数据中“1”的个数是偶数还是奇数。例如，当选择偶校验时，如果之前的数据标志位“1”的个数为偶数，则传送一个值为0的校验位。八位二进制数0110 0011的偶校验位应该为0。如果偶校验有效且传送的是二进制数1101 0110，则偶校验位将是1。奇校验与此正好相反，当前面传送的字中标志位“1”的个数是奇数，校验位的值为0。校验规则很简单：对于偶校验，标志位个数要为偶数；对于奇校验，标志位个数要为奇数。校验位是一个

标志或间隔都可使校验正确进行。校验的差错检验很初级，它将告诉你字符中是否存在差错位，但同时它无法显示哪一位出了差错。如果有偶数个位都出差错，这时校验位对任何差错都没有效果。标志校验意思是校验位始终设置为标志信号状态，间隔校验意思是校验位始终传送间隔信号状态，由于这两种校验都没有什么有效用途，也就几乎从来不被使用。图1-1展示了一些字符传送采样的例子。

**参考** 描述数据流特性的常见速记形式是：8-N-1，第一个表示数据位的位数：5、6、7或8，第二个是表示校验的类型：N (one)、E (ven)、O (dd)、S (pace) 或 M (ark)。第三个表示停止位的位数：1、1.5或2。

### 实例

采样#1展示的是位流1100 1001的采样，如果我们定义数据流为：8个数据位、无校验和一个停止位，或速记为8-N-1。在采样时间1，线路空闲或者说处于标志状态（“1”），当线路状态变为间隔状态（“0”）时，线路检测开始位并设置采样频率（按定义的每秒的位数），当下一个采样时间到来时，线路通过对数据流的“1”和“0”的采样开始按位计时。在字符流的最后，停止位强制把数据流置位为标志或空闲状态，这样来保证数据流能正确地用一个间隔位说明下一个新字符的开始。

**问题#1：**根据下列对采样#2、#3及采样#4的数据流定义，传送的字符用二进制表示是什么，它们是否是有效字符？（答案见本章末。）

采样#2的定义为：5-E-1

采样#3的定义为：8-E-2

采样#4的定义为：7-O-1.5

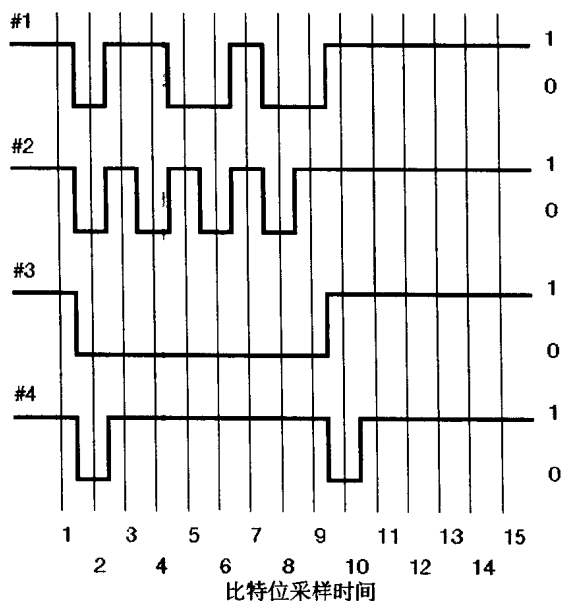


图1-1 异步Bit流

还有一个内容没有讨论，那就是传送速度，传送速度是基于时钟信号的，当我们接触到附录A中的布线和物理层信号发送，将看到在接收两端是没有时钟信号的。那么线路两端怎么知道数据流发送速度有多快呢？为了正确传送和接收数据，每端必须按每秒位数设置匹配的速度。在每端设置速度时还可有另外的选项，这一特性被称为自动波特率。如果正在接收一个已知字符，速度是正确的，那么该字符就将正确地解码，这一点仅当每个字符的位数和校验都正确定义的情况下才起作用。自动波特率以端口或线路最大有效速度开始，并检查是否接收到已知字符。如果没有把传送的数据正确解码，那么初始速度降低到下一级较低标准速度，并再检查一次数据流，这一过程将一直重复下去直到探测到正确的速度为止。

**技巧** 自动波特率看起来好像是一个有用的选项，但它并不像定义确切的所需操作速度那么可靠。

表1-1详细说明了一些选项，这些选项在传送数据到Cisco路由器和从Cisco路由器接收数据时是有效的，对于异步线也是适用的。

表1-1 异步特征选项

| 控制信息 | 参量  | 描 述                                                      |
|------|-----|----------------------------------------------------------|
| 数据位  | 位   | 描述单一字符所用的位数：5 6、7 8（缺省值）                                 |
| 校验   | 位   | 加在数据位后的校验位用于对每个已传送的字符的差错检验：None（缺省值）、Even、Odd、Space、Mark |
| 停止位  | 位   | 用来说明字符传送结束的位数 1 1.5 2（缺省值）                               |
| 速度   | 位/秒 | 用位/秒表示的传送速度：300到115200（缺省值为9600）                         |

1.3 拨号访问设置

我们已经学习了异步数据流的基础知识，现在就应用这些知识，编程控制一个 Cisco 路由器接受一个终端的拨号连接，在这个例子中我们将用到一台有内置Modem的PC、一台Cisco 3640路由器、一个 3Com 56K调制解调器和一台用来模拟电话公司的Black Box 4X16键电话系统。图1-2 描述了在这个例子中所使用设备的布局。

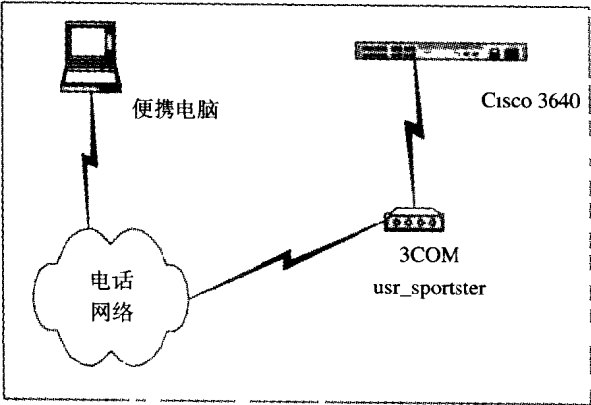


图1-2 拨号设备布局

我们将首先进行初始化设置，然后一步一步地进行随后建立异步连接的过程。

为了控制用于异步连接的Cisco 访问服务器/路由器，我们需要正确地确定与物理连接对应的线的数字编号。对于固定结构的路由器，例如Cisco 2511，有一个公式用来计算对应于每个物理连接的线的数字编号。表1-2列出了对于固定结构的Cisco 路由器线的编号分配公式。

模块化的访问服务器/路由器都有一个特定线号与设备中每一条可能配置的线相对应。表1-3定义了Cisco 3640中分配给每个标准插槽的所有线号。每个插槽最多能定义的线数为32。

表1-2 固定的配置线分配

| 连接类型 | 线的数字号            |
|------|------------------|
| 控制端口 | 线0               |
| 异步端口 | 端口号即为线号          |
| 附加端口 | 异步端口数+1          |
| 虚拟终端 | 第一条虚拟终端线为异步端口数+2 |

表1-3 Cisco 3640线的定义

| 连接类型 | 线的数字号            |
|------|------------------|
| 控制端口 | 0线               |
| 插槽0  | 1线到32线           |
| 插槽1  | 33线到64线          |
| 插槽2  | 65线到96线          |
| 插槽3  | 97线到128线         |
| 附加端口 | 129线             |
| 虚拟终端 | 130线是虚拟终端话路的起始线号 |

在这一章的例子中将用到三个不同的访问服务器/路由器：16个异步端口的Cisco 2511访问服务器；插槽3为异步/同步接口的Cisco 3640；一个Cisco 1720 标准路由器，它将用在附加端口的例子中。

在第一个例子中，我们要用一台在插槽3上有一个8 端口异步/同步插件的Cisco 3640。在表1-3中可以看到，分配给该插槽的线号是97到128，由于仅8个端口可用，因此线号只可能是97到104，如果我们为异步连接只使用插槽3的接口0，那么我们就将使用97线。Cisco有双重目的的端口，其缺省的操作模式是同步，要激活97线就必须将操作模式改变为异步。下面列出了对该路由器的所有初始设置命令，其中包括对以太网端口的连接测试ping。

```
Router>enable
Router#conf t
Enter configuration commands, one per line. End with CNTL/Z
Router(config)#no ip domain-lookup
Router(config)#enable password san-fran
Router(config)#line con 0
Router(config-line)#logging synchronous
Router(config-line)#exec-timeout 0 0
Router(config-line)#host C3640
C3640(config)#int e 0/0
```

```

C3640(config-if)#ip address 172.31.200.210 255 255.255 0
C3640(config-if)#no shut
C3640(config-if)#^Z
C3640#ping 172.31 200 210
Type escape sequence to abort
Sending 5, 100-byte ICMP Echos to 172 31 200 210, timeout is 2 seconds.
!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 1/1/4 ms

```

C3640#

现在我们可以用show line命令来看看用初始设置命令定义的所有线的情况。

```

C3640#sho line

```

| Tty | Typ | Tx/Rx     | A | Modem | Roty | AccO | AccI | Uses | Noise | Overruns | Int |
|-----|-----|-----------|---|-------|------|------|------|------|-------|----------|-----|
| * 0 | CTY |           | - | -     | -    | -    | -    | 0    | 0     | 0/0      | -   |
| 65  | TTY |           | - | inout | -    | -    | -    | 0    | 0     | 0/0      | -   |
| 66  | TTY |           | - | inout | -    | -    | -    | 0    | 0     | 0/0      | -   |
| 67  | TTY |           | - | inout | -    | -    | -    | 0    | 0     | 0/0      | -   |
| 68  | TTY |           | - | inout | -    | -    | -    | 0    | 0     | 0/0      | -   |
| 69  | TTY |           | - | inout | -    | -    | -    | 0    | 0     | 0/0      | -   |
| 70  | TTY |           | - | inout | -    | -    | -    | 0    | 0     | 0/0      | -   |
| 129 | AUX | 9600/9600 | - | -     | -    | -    | -    | 0    | 0     | 0/0      | -   |
| 130 | VTY |           | - | -     | -    | -    | -    | 0    | 0     | 0/0      | -   |
| 131 | VTY |           | - | -     | -    | -    | -    | 0    | 0     | 0/0      | -   |
| 132 | VTY |           | - | -     | -    | -    | -    | 0    | 0     | 0/0      | -   |
| 133 | VTY |           | - | -     | -    | -    | -    | 0    | 0     | 0/0      | -   |
| 134 | VTY |           | - | -     | -    | -    | -    | 0    | 0     | 0/0      | -   |

-Line(s) not in async mode -or- with no hardware support.

1-64, 71-128

C3640#

在上面的show line命令中，我们看到65到70线而不是预期的插槽3的97到104线。事实上，这些线表示插槽2上的异步设备——我们将在第7章介绍的MICA数字调制解调器。为了使97线显示出来，我们必须把缺省的同步操作模式改变异步。

```

c3640#conf t
Enter configuration commands, one per line End with CNTL/Z
C3640(config)#int s 3/0
C3640(config-if)#physical asynchronous
C3640(config-if)#^Z

```

C3640#

当你输入运行physical asynchronous命令时，路由器必将修改其内部电路；因此在执行并得到下一个提示符之前，你要注意是否确实要执行该命令。确定97线已经激活之后，让我们再做一个show line命令。

C3640#show line

| Tty | Typ | Tx/Rx     | A | Modem | Roty | AccO | AccI | Uses | Noise | Overruns | Int   |
|-----|-----|-----------|---|-------|------|------|------|------|-------|----------|-------|
| * 0 | CTY |           | - | -     | -    | -    | -    | 0    | 0     | 0/0      | -     |
| 65  | TTY |           | - | inout | -    | -    | -    | 0    | 0     | 0/0      | -     |
| 66  | TTY |           | - | inout | -    | -    | -    | 0    | 0     | 0/0      | -     |
| 67  | TTY |           | - | inout | -    | -    | -    | 0    | 0     | 0/0      | -     |
| 68  | TTY |           | - | inout | -    | -    | -    | 0    | 0     | 0/0      | -     |
| 69  | TTY |           | - | inout | -    | -    | -    | 0    | 0     | 0/0      | -     |
| 70  | TTY |           | - | inout | -    | -    | -    | 0    | 0     | 0/0      | -     |
| 97  | TTY | 9600/9600 | - | -     | -    | -    | -    | 0    | 0     | 0/0      | se3/0 |
| 129 | AUX | 9600/9600 | - | -     | -    | -    | -    | 0    | 0     | 0/0      | -     |
| 130 | VTY |           | - | -     | -    | -    | -    | 0    | 0     | 0/0      | -     |
| 131 | VTY |           | - | -     | -    | -    | -    | 0    | 0     | 0/0      | -     |
| 132 | VTY |           | - | -     | -    | -    | -    | 0    | 0     | 0/0      | -     |
| 133 | VTY |           | - | -     | -    | -    | -    | 0    | 0     | 0/0      | -     |
| 134 | VTY |           | - | -     | -    | -    | -    | 0    | 0     | 0/0      | -     |

Line(s) not in async mode -or- with no hardware support:  
1-64,71-96 98-128

C3640#

在show line命令中共有12列。作为整本书中都将使用的命令，让我们看一看命令中每列内容的含义：第一列是线号；第二列是线类型：CON(控制台)，TTY(异步)，AUX(附加)，VTY(虚拟终端)；第3列是操作速度，包括传送(TX)和接收(RX)速度；第四列指出是否自动波特率(Autobaud)，是(F)或不是(-)；第5列是调制解调器操作模式：callin(接入)、callout(拨出)、CTS-req(等待发送请求)、DTR-Act(就绪)、RIisCD、或inout(输入输出)；第6列是该线被指派的循环组；第7和第8列描述了为控制对该线的访问所指定的输出(AccO)和输入(AccI)访问列表；第9列是从系统启动该线被使用的次数；第10列是从系统启动被检测到的噪声数；第11列是硬件(UART)超限/软件缓冲区溢出数，都被定义为从系统启动指定线上发生的超限或溢出数，硬件超限就是缓冲区超限，此时UART芯片从软件接收到的数据位已经超过能够处理它们的速度；而当软件从硬件接收到数据位超过其能够处理它们的速度时，软件溢出就会发生。第12列为该线对应的接口。

在Tty字段的数字前带星号(\*)表示该线当前有效，正在执行一个面向终端的协议；在数字前带“A”表示该线在自选(autoselect)模式下当前有效；在数字前带“I”表示该线被设置为一个async mode interactive(异步交互模式)接口，当前空闲并可以用于异步模式。

现在该是进入配置命令的时候了，这些命令将控制97线实现字符的传送。

```
C3640#conf t
Enter configuration commands, one per line. End with CNTL/Z.
C3640(config)#line 97
C3640(config-line)#speed 115200
C3640(config-line)#parity none
C3640(config-line)#databits 8
C3640(config-line)#stopbits 1
C3640(config-line)#^Z
C3640#
```

我们已经定义了线将如何与外部Modem通信，现在就必须建立到该Modem的连接。我们将用到一个称为反向远程登录的技术，从控制台远程登录连接到路由器中的一个内部地址，但是我们将用端口2097而不是常规的端口23，端口数XYYY由两部分组成：X和YYY，X描述当连接到线YYY上时将进行的协议转换类型。类型2把远程登录包转换成单个的ASCII字符集，从而在该线上建立其ASCII终端话路。

```
C3640#telnet 172 31 200 210 2097
Trying 172 31 200 210, 2097
% Connection refused by remote host
```

```
C3640#
```

居然拒绝连接？哦，当远程登录到任何目标时，我们都需要提供至少一个密码用于对目标资源访问的认证。让我们给97线添加一个注册过程和一个密码，并在该线上完成设置过程。数据流控制通常都有硬件处理，但在下面的代码中列出了其它一些选项。transport input all命令用来描述哪一个有效协议能用于该线的连接，通过指定“all”，我们将可以使用任何一种协议，包括远程登录协议（telnet）。

```
C3640#conf t
Enter configuration commands, one per line End with CNTL/Z
C3640(config)#line 97
C3640(config-line)#login
C3640(config-line)#password cisco
C3640(config-line)#flowcontrol ?
    NONE          Set no flow control
    hardware      Set hardware flow control
    software      Set software flow control

C3640(config-line)#flowcontro hardware
C3640(config-line)#transport nput all
C3640(config-line)#^Z
C3640#
```

这样应该说基本完成了线的设置。现在我们应该可以用反向远程登录协议连接到我们的Modem了。发出telnet命令，将得到一个输入密码提示，正确地输入密码，我们将看到Password OK，这使我们知道已经成功地完成与Modem的连接。因为没有提示符和任何其他可以说明我们已连接上的信息，起先看起来似乎死机了。执行at得到一个应答OK，说明我们连接上了。用命令at\$将获得帮助信息（输出显示已被删除，没有写出），命令at14显示当前Modem设置。有关Modem的设置及其命令我们将在稍后花更多的时间详细讲解。

```
C3640#telnet 172 31 200 210 2097
Trying 172 31 200 210, 2097      Open

User Access Verification

Password
Password OK
```

```

at
OK
at $
HELP, Command Quick Reference (CTRL-S to Stop, CTRL-C to Cancel)

&$    HELP, Ampersand Commands                Mn    n=0    Speaker OFF

.
.
n=3    H1 Speaker Volume                        $ HELP, Command Summary

OK
at14
U.S. Robotics 56K FAX EXT Settings...

B0 E1 F1 M1 Q0 V1 X1 Y0
BAUD=115200 PARITY=N WORDLEN=8
DIAL=TONE ON HOOK CID=0

&A1 &B1 &C1 &D2 &G0 &H1 &I0 &K1
&M4 &N0 &P0 &R2 &S0 &T5 &U0 &Y1

S00=001 S01=000 S02=043 S03=013 S04=010 S05=008 S06=002
S07=060 S08=002 S09=006 S10=014 S11=070 S12=050 S13=000
S15=000 S16=000 S18=000 S19=000 S21=010 S22=017 S23=019
S25=005 S27=000 S28=008 S29=020 S30=000 S31=128 S32=002
S33=000 S34=000 S35=000 S36=014 S38=000 S39=000 S40=001
S41=000 S42=000

LAST DIALED #:
OK

```

所有的Modem使用完全一样的命令，使得我们可以用标准的AT命令控制这些Modem而没有任何问题，我们不是太运气了吗？只是玩笑而已，别当真，我们用的是一个USRobotics（3COM）Sportster Modem，附录B中列出了用来控制所有本书中用到的Modem的命令。

要得到Modem命令环境，我们必须从远程登录话路上断开，并回到路由器控制提示符。转义序列做起来很容易，小小地实践以下：第一步：按下<control>+<shift>+<6>一次；第二步：移开在键盘上所有的手指，敲x键。用这样的顺序就可以从任何远程登录话路上断开，更加复杂的情况将在第二章涉及到。断开之后，我们可以用where或show sessions命令看到我们所有有效远程登录话路在什么位置。最左边的星号表示出最后有效的连接，在命令提示符下敲〈回车〉键能重新把你连接到远程登录目标。如果你有多个有效远程登录话路，只要敲resume X或仅仅X（X是在where命令中显示的连接号）。

```

C3640#where
Conn Host                Address                Byte    Idle Conn Name
*  1 172.31.200.210      172.31.200.210        0        0
172 31 200.210
C3640#

```

要关闭当前远程登录连接，我们可以敲入命令`disconnect X`，`X`是连接号。如果要断开当前连接，我们可以只敲入`disconnect`而不带连接号。当有多个连接被终止时，输入`quit`、`exit` 或 `logout`结束通话并关闭所有的开放的远程登录话路。

```
C3640#disconnect
Closing connection to 172.31.200.210 [confirm]
C3640#
```

当你对Modem问题进行调试时，输入`telnet`命令来连接Modem可能会耗费一定的时间，但我们可以路由器上设立一个快捷方式：用一个静态的`ip host`命令，可以给某个IP地址和端口号指定一个名称。

```
c3640#conf t
Enter configuration commands, one per line. End with CNTL/Z
C3640(config)#ip host modem0 2097 172.31.200.210
C3640(config)#^Z
C3640#
```

现在我们可以敲入`modem0`完成连接，然后用Cisco路由器控制Modem适当地操作。调制解调器命令串`at&F1&C1&B1S0=1&W`将在本章后面讨论Modem初始化过程自动化时使用到。

```
C3640#modem0
Trying modem0 (172.31.200.210, 2097) . Open
```

```
User Access Verification
```

```
Password
Password OK
at
OK
at&F1&C1&B1S0=1&W
```

```
OK
```

让我们断开调制解调器的连接，保存到此为止我们执行的东西。

```
C3640#copy running startup-config
Destination filename [startup-config]?
Building configuration .
C3640#
```

## 1.4 超级终端设置

为了测试出我们的设置，需要设立一台PC作为连接终端，使我们能够用它拨号登录到Cisco路由器/访问服务器，我们将使用在Windows95和Windows98下的超级终端程序来完成，在附件中打开超级终端文件夹（如图1-3所示），双击Hypertm图标开始。

从窗口底下的一行中选择一个图标并指定一个名字给即将产生的桌面图标（如图1-4所示）。