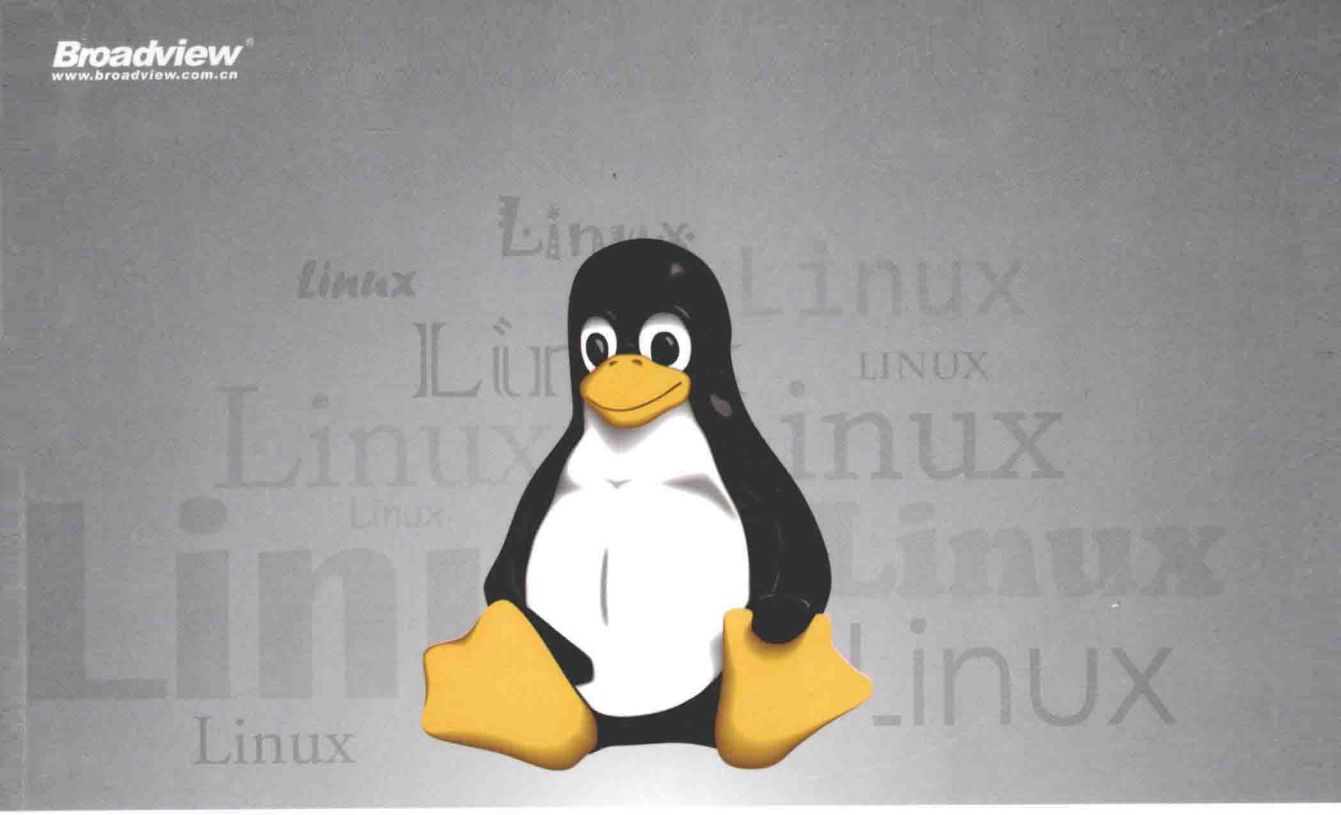
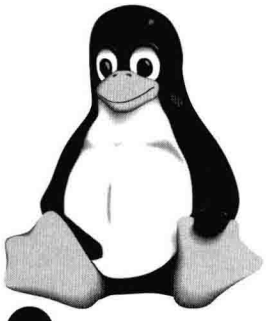
Broadview
www.broadview.com.cn



嵌入式 Linux 系统开发全程解析

韩超 等著

電子工業出版社
PUBLISHING HOUSE OF ELECTRONICS INDUSTRY
http://www.phei.com.cn



嵌入式 Linux 系统开发全程解析

韩超 等著

電子工業出版社
Publishing House of Electronics Industry
北京•BEIJING

内 容 简 介

本书是一本全面介绍嵌入式 Linux 开发的专著，书中涵盖了程序生成工具、调试工具、引导加载器、Linux 系统结构、Linux 内核、驱动程序、用户空间编程、用户空间中间件等方面的内容。本书内容前后照应、贴近实践，且有较强的延伸型，有利于读者建立嵌入式 Linux 开发系统化的知识结构和技术理念。

本书不仅适用于嵌入式 Linux 的工程师增强能力，也适用于其他领域的技术人员了解嵌入式 Linux。

图书在版编目（CIP）数据

嵌入式 Linux 系统开发全程解析 / 韩超等著. —北京：电子工业出版社，2014.5
ISBN 978-7-121-22888-9

Ⅰ. ①嵌… Ⅱ. ①韩… Ⅲ. ①Linux 操作系统－程序设计 Ⅳ. ①TP316.89

中国版本图书馆 CIP 数据核字（2014）第 066337 号

策划编辑：李　冰
责任编辑：李云静
印　　刷：中国电影出版社印刷厂
装　　订：三河市鹏成印业有限公司
出版发行：电子工业出版社
　　北京市海淀区万寿路 173 信箱　　邮编：100036
开　　本：787×1092　1/16　印张：27.25　字数：646 千字
印　　次：2014 年 5 月第 1 次印刷
定　　价：59.00 元

前　言

本书写作目的

嵌入式 Linux 开发已经不算很新的技术。本书作者从 2003 年开始从事 Linux 方面的研究、开发和科普工作。在 10 年多的时间里，Linux 内核已经从 2.4 版本发展到了 3.x 版本，广泛使用的嵌入式处理器也从 ARMv4 的 ARM7 核心发展成为 ARMv6 的 Cortex 核心。这 10 年多嵌入式 Linux 技术逐步成熟，产品已经数不胜数。

作者在几年前出版过一些有关嵌入式系统、Linux 和 C 语言编程的书籍。目前面对技术的发展，感觉有必要出版一本全面介绍嵌入式 Linux 的书籍，以帮助相关行业的学习者和开发者更高效地了解嵌入式 Linux，更好地从事相关工作。

希望将本书打造成简明、高效的工具书，成为快速开发 Linux 软件的指导书、硬件移植的工具书，以及计算机专业从业者理论联系实际的桥梁。

本书特点

本书结合了作者多年的开发经验和知识技术的传播经验，主要有下面一些特点。

- 内容来自工程实践，实用性强。
- 覆盖面更全面、知识系统完整。
- 使用框架图+代码路径+关键代码的方式，一目了然。
- 内容紧凑，读者可以结合手头代码对照学习。
- 将工程技巧蕴含于理论知识的网络之中。
- 包含 Linux 软件编程开发的常用技巧：查找代码、运行时看信息等。
- 结合硬件和操作系统的知识。
- 帮助读者深入理解 Linux 系统的关键结构，具有完备的开发调试能力。
- 重点关注目前的主要应用场景：用户空间开发和驱动开发。

本书以 Linux 尤其是嵌入式 Linux 中最常用的内容为主，这些内容大部分是 Linux 开发不同方面的工程师均需要掌握的。通过对本书的学习，可以让有 C 语言基础的工程师在 Linux 环境中开发用户空间软件；让有硬件基础的工程师可以在嵌入式 Linux 平台中具有适

配硬件的能力；让具有一定经验的嵌入式 Linux 工程师具有更广泛的视野、更强的开发能力。

本书主要内容

本书包含了嵌入式 Linux 系统的主要内容，按照知识结构分成四个方面。

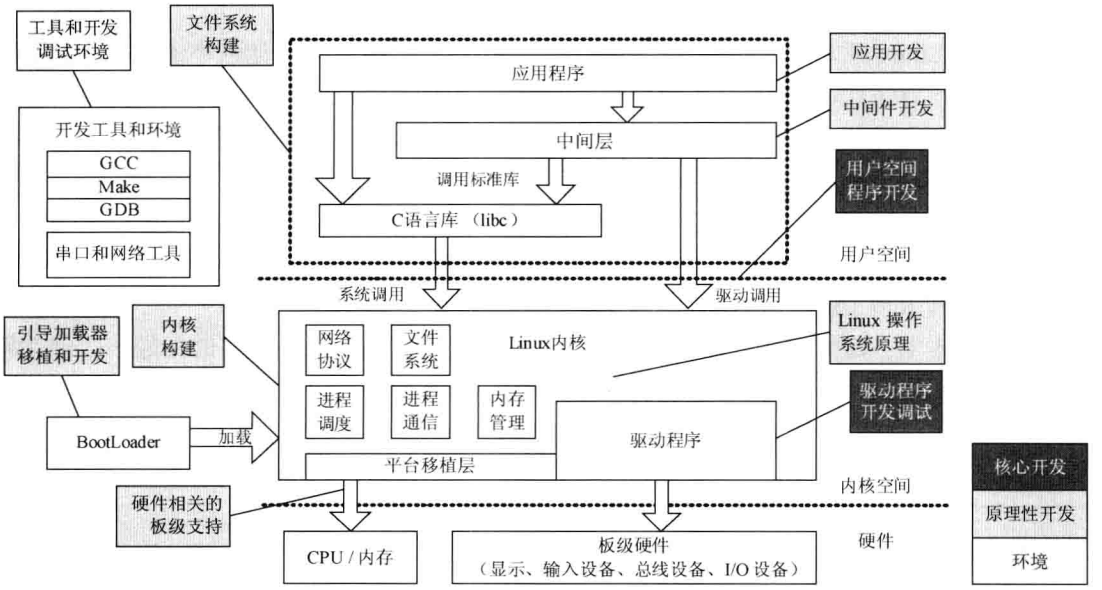
第一个方面：开发环境和编程基础（第 1 章到第 4 章）。

第二个方面：Linux 系统结构（第 5 章到第 8 章）。

第三个方面：嵌入式 Linux 的用户空间（第 9 章到第 13 章）。

第四个方面：嵌入式 Linux 的驱动开发（第 14 章到第 24 章）。

按照内容的侧重点，本书具有环境、原理性开发和核心开发几个方面。嵌入式 Linux 系统和本书知识结构如下图所示。



对读者的话

目前的 IT 技术领域有很多热点，除了嵌入式设备的开发外，还有移动开发、互联网开发等。嵌入式 Linux 是很多产品的技术基础。在实践过程中，很多问题都是嵌入式 Linux 最基础的问题，只是由于工程师可能来自其他领域，不熟悉嵌入式 Linux，从而小问题就成了大问题。因此，对于嵌入式 Linux，是目前从事 IT 技术领域的各类工程师都至少需要了解的。

本书不仅适用于嵌入式 Linux 的工程师增强能力，也适用于其他领域的技术人员了解嵌入式 Linux。本书尤其注重和高校计算机专业的互补关系，基于高校计算机专业知识基础，书中大量补充了在实践中的应用，帮助读者将知识“落地扎根”，引领读者进一步在工作中

让知识和技能“生根发芽”，直到在工作中“结出丰硕的果实”。

本书的几个基础方面是C语言编程、嵌入式处理器、操作系统，这也是计算机行业的基础。通过对本书的学习，读者得到的不仅仅是各个方面的知识和技巧，更有它们之间的有机结合。

本书作者

韩超是中国大陆长期工作于一线的知名工程师、架构师，也是嵌入式 Linux 相关技术在大陆发展 10 年的技术领航人之一，同时也是畅销书作者。其主要从事相关技术研发方向，包括嵌入式 Linux 板级平台、GUI 系统和应用、移动多媒体。韩超对嵌入式 Linux 的技术把握以实用技术为主，以操作系统本身为辅，重视在系统使用特定硬件的技术，重视内核与用户空间的交互的要点，适用于嵌入式 Linux 的软件工程等方面。

韩超完成了本书主要部分的编写工作，此外，众多不同规模的企业开发成果也为本书的编写提供了重要的素材。参与本书编写的还有康硕、于仕林、张超等人，以及清华大学计算机系操作系统研究兴趣小组的肖奇学、徐永健、王欢、何嘉权、范文良、茅俊杰等人。

目　录

第1章 Linux的开发环境

1.1 开发环境概述

在 Linux 和嵌入式 Linux 的开发中，一般将生成代码的机器称为“主机”，而将运行代码的机器称为“目标机”。主机和目标机不是同一个机器的开发，则被称为交叉开发。

主机和目标机代码生成的过程如图 1-1 所示。

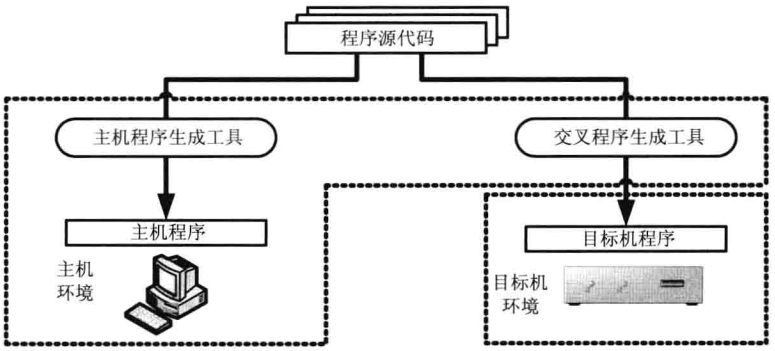


图 1-1 主机和目标机的代码生成

如果是单纯桌面（x86）上的程序开发，主机和目标机的结构类似，生成的程序一般可以在开发机和目标机同时运行。如果在嵌入式系统中开发，通常在主机环境中开发，通过交叉编译工具生成目标机的程序，然后通过某种手段放入目标机系统中运行。

对于嵌入式的开发，一般分成主机和目标机通常是两个不同的设备。在嵌入式 Linux 的开发中，主要是指目标机使用 Linux 作为操作系统，并在操作系统上构造不同应用。嵌入式 Linux 开发的主机环境，主要是指在主机上生成运行于目标机的代码。主机一般是运行 Linux 操作系统的 x86 主机，也可以在 Windows 上开发嵌入式 Linux。

Linux 系统最基本的程序开发是 C 语言的程序开发。C 语言程序的生成分成编译、汇

编、连接等几个步骤。最终的目标文件的主要部分是处理器可执行的机器代码组合。根据系统的不同，生成的目标包括了可执行的二进制机器代码，也包括一定的头信息。

Linux 系统的开发主要涉及了如下内容：

- GCC 和交叉 GCC 均运行于开发机，但程序生成的部分存在区别。
- Make 和 Makefile 运行于开发机。
- GDB 可分为运行于主机的部分和运行于目标机的部分，也适用于本机开发。
- 主机-目标机的连接方式有多种，也适用于本机开发。

在嵌入式系统的开发中，除了程序生成工具之外，还需要使用主机到目标机的通信工具。主要的工具包括目标机的终端模拟工具和文件传输工具，前者一般使用从主机到目标机的串口连接完成，后者一般使用网络协议来实现。

嵌入式系统主机和目标机的连接如图 1-2 所示。

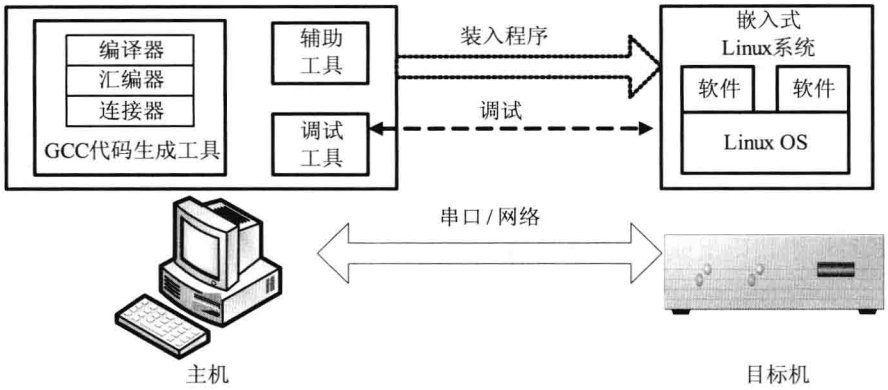


图 1-2　主机-目标机的开发结构

在嵌入式系统的开发中，需要使用主机到目标机的通信工具。主要的工具包括目标机的终端模拟工具和文件传输工具，前者一般使用从主机到目标机的串口连接完成，后者一般使用网络协议来实现。

主机到目标机通常有两条通道：一条是目标机 UART 到主机串口的连接，可以提供打印信息和控制台支持；另一条是网络连接，可以建立于网络协议 TCP/IP 之上，这样可以屏蔽数据链路层以下的细节，并且有多种基于上层协议的软件可以使用。基于网络协议的通信不仅限于在网络芯片硬件上，也可以通过 USB 或其他通信端口作为网络协议的硬件基础。

1.2　串口终端工具

串口是嵌入式开发过程中，主机和目标机最基本的连接方式，可以用于字符形式的输出和输入。例如：在 C 语言的程序开发中，使用 printf()作为调试语句，主机的终端常使用显示器作为屏幕输出。在嵌入式系统的目标机调试过程中，由于一般不具有屏幕等输出设

备，因此通常使用目标机处理器的串口作为输出，在主机端使用串口工具接收这些信息。同样，在目标机需要交互调试的时候，串口的连接方式也可以提供目标机的输入，例如：getchar()、scanf()等功能。在开发过程中，终端虽然硬件基础不一样，但是其表现形式是类似的。

使用 UART 连接的几个主要参数。

- 波特率（Speed，Baud rate），串口数据的速率，一般可能有 1200 b/s、38 400 b/s、115 200 b/s 等。
- 数据位（Data）：通常有 5、6、7、8 几种。
- 奇偶校验方式（Parity）：一般有无校验（None）、偶校验（Even）、奇校验（Odd）、标记（Mark）、空格（Space）等。
- 停止位（Stopbits）：一般有 1 位、2 位两种方式。
- 流控制（Flow Control）：使用硬件流控制或者软件流控制。

提示：如果主机的串口终端的输出为乱码，则通常是波特率设置的问题；如果主机串口的控制台不能输入，通常是流控制的设置问题。

对于主机上使用的串口终端工具，在 Windows 系统中可使用超级终端，在 Linux 系统中可以使用 minicom 工具。

1. Windows 的超级终端

超级终端是 Windows 自带的工具，在使用的过程中很简单。在超级终端中，选择“文件”→“新建连接”，在建立的过程中，需要设置连接方式为 com（串口），根据实际连接的情况选择使用 com1 或 com2，设置的对话框如图 1-3 所示。

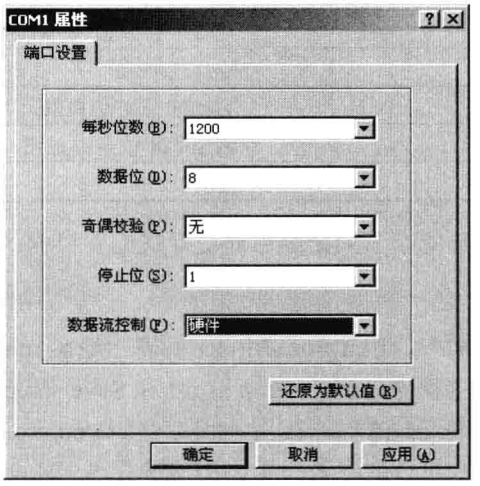


图 1-3　主机-目标机的连接

以上设置的参数，需要与目标机处理器的 UART 端口的设置一致，目标机的设置一般在 UART 的初始化过程中通过配置寄存器完成。

2. Linux 的 minicom

minicom 是 Linux 下的串口终端工具，使用 minicom 可以在 Linux 下实现目标机和主机的连接。minicom 可以建立于 Linux 的串口设备（ttyS0）或者调制解调器设备（modem）上，其使用起来没有 Windows 下的工具友好，需要从命令行启动。

minicom 命令行启动的参数如下所示：

```
minicom [OPTION]... [configuration]
```

minicom 命令主要参数的含义如表 1-1 所示。

表 1-1　minicom 的参数

参数	含义
-s, --setup	输入建立模式，仅在 root 上
-o, --noinit	不要初始化调制解调器
-m, --metakey	使用 Alt 或 meta 键作为命令键
-M, --metakey8	使用 8 位的 meta 键作为命令键
-l, --ansi	逐字翻译，假设屏幕使用 IBM-PC 字符集
-L, --iso	同 L，但是假设屏幕使用 SO8859
-w, --wrap	换行
-z, --statline	使用终端状态行
-8, --8bit	8 位干净模式，例如为日本字符所使用的
-c, --color=on/off	ANSI 样式的颜色使用开关
-a, --attrib=on/off	用户反转高亮的开关
-t, --term=TERM	重载 TERM 环境变量
-S, --script=SCRIPT	使用启动的脚本文件
-d, --dial=ENTRY	从拨号目录拨入 ENTRY
-p, --ptty=TTYP	连接伪终端
-C, --capturefile=FILE	启动捕获文件 FILE
-v, --version	输出版本信息并退出
configuration	配置文件

一种较为方便的方式是使用下列命令启动交互式配置界面：

```
# minicom -s
```

在 minicom 主配置界面中，Exit from Minicom 表示退出 minicom 应用，Exit 表示退出配置程序，进入 minicom 主程序。Save setup as dfl 和 Save setup as 可以选择保存当前的配置。minicom 选择 Save setup as dfl 时，配置将保存在/etc/minirc.dfl 文件中。

配置的主要工作是针对串口的参数设置，因此选择 Serial port setup 选项。

minicom 主配置界面和串口配置界面如图 1-4 所示。

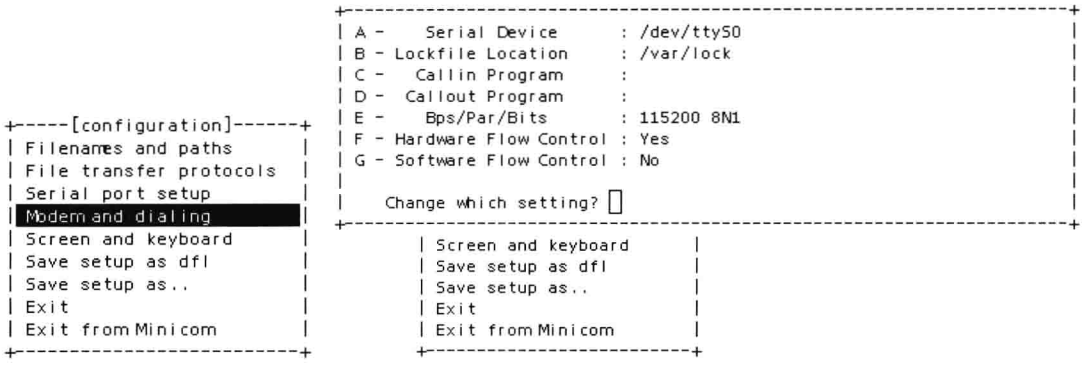


图 1-4　minicom 主配置界面（左）和串口配置界面（右）

在 minicom 串口界面中，可以使用 A~G 等字母选择相应的配置内容，A 用于选择串行的设备，使用 Linux 下的设备名称，如/dev/modem、/dev/ttyS0 等。

在串口终端的配置中，主要需要设置的内容是波特率、数据位等内容，使用键 E 启动串口的配置界面，如图 1-5 所示。

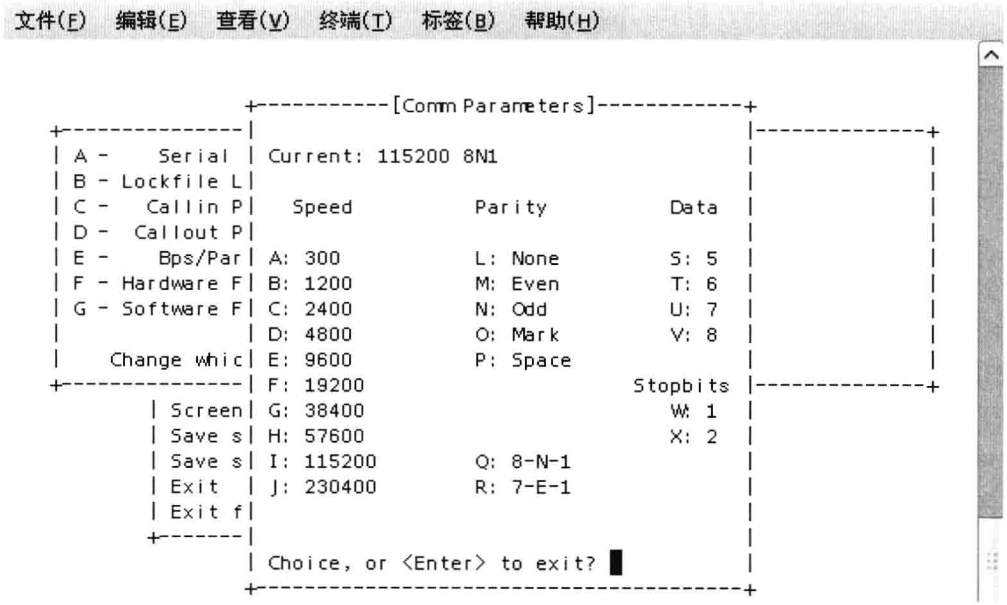


图 1-5　串口波特率、数据位、奇偶校验、停止位界面

本界面用于配置，串口波特率、数据位、奇偶校验、停止位使用按键选择实现。在配置完成后，可以使用没有命令行参数的方式启动 minicom。

在 minicom 启动主界面中如果有来自串口的数据，将在屏幕上显示。在主界面下可以继续使用 Ctrl+A 组合键，然后再按 Z 键实现进行配置。minicom 启动主界面的帮助界面，如图 1-6 所示。

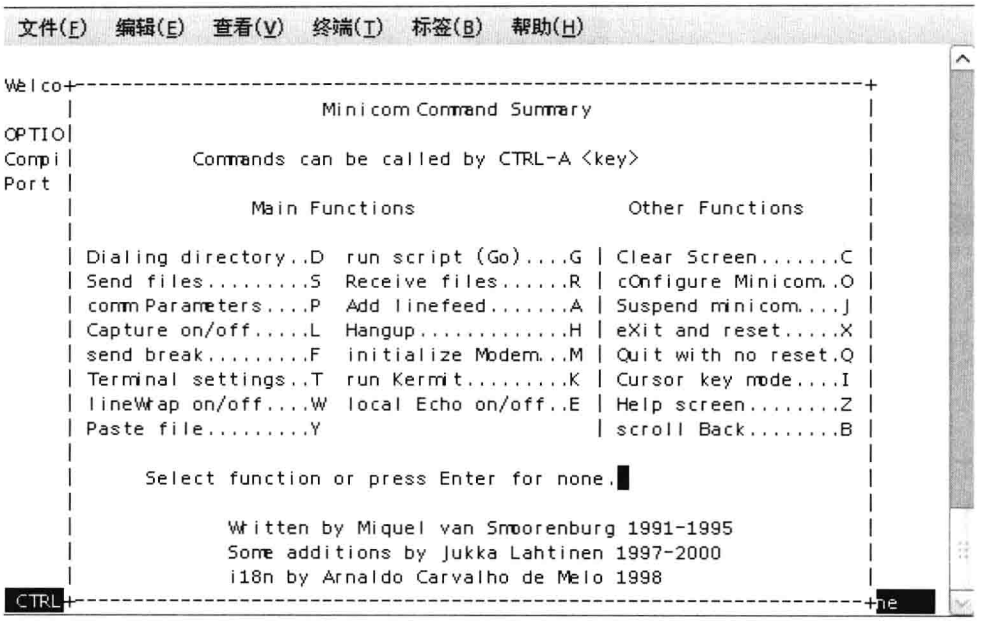


图 1-6 minicom 启动主界面的帮助界面

在主界面的帮助界面中，可以完成一些使用过程中的配置和辅助功能。

1.3 TFTP

TFTP 的全称为 Trivial File Transfer Protocol（简单文件传输协议）。此协议设计时的目标是进行小文件传输的。与基于 TCP 的 FTP（File Transfer Protocol，文件传输协议）不同，TFTP 使用 UDP 的端口 69。TFTP 只能从文件服务器上获得或写入文件，不能列出目录，不进行认证，按照 8 位来传输数据。

TFTP 使用以下 3 种传输模式。

- Netascii：这是 8 位的 ASCII 码形式。
- octet：这是 8 位源数据类型。
- mail：它将返回的数据直接返回给用户而不是保存为文件。

在嵌入式系统的开发过程中，TFTP 常常被使用作为主机-目标机文件传输的协议。一个典型的用法是在主机端开启 TFTP 服务器，在目标机端由是支持 TFTP 的 Bootloader 或者运行于嵌入式 Linux 操作系统的 TFTP 客户端。

例如：在 Ubuntu 环境中设置 TFTP 服务器，需要下载以下的包：

```
$ sudo apt-get install tftpd-hpa
```

进一步，修改 etc 目录中的配置文件/etc/default/tftp-hpa：

```
#Defaults for tftpd-hpa
RUN_DAEMON="no"
OPTIONS="-l -s /var/lib/tftpboot"
```