

嵌入式Linux 应用编程

马丽洁◎主编

嵌入式 Linux 应用编程

主 编 马丽洁

副主编 李占岭 董圣华 李建华

 北京理工大学出版社

BEIJING INSTITUTE OF TECHNOLOGY PRESS

图书在版编目 (CIP) 数据

嵌入式 Linux 应用编程/马丽洁主编. —北京:北京理工大学出版社, 2016. 8

ISBN 978 - 7 - 5682 - 1745 - 3

I. ①嵌… II. ①马… III. ①Linux 操作系统 - 程序设计 - 高等学校 - 教材

IV. ①TP316. 89

中国版本图书馆 CIP 数据核字 (2015) 第 316363 号

出版发行 / 北京理工大学出版社有限责任公司

社 址 / 北京市海淀区中关村南大街 5 号

邮 编 / 100081

电 话 / (010) 68914775 (总编室)

(010) 82562903 (教材售后服务热线)

(010) 68948351 (其他图书服务热线)

网 址 / <http://www.bitpress.com.cn>

经 销 / 全国各地新华书店

印 刷 / 三河市华骏印务包装有限公司

开 本 / 787 毫米 × 1092 毫米 1/16

印 张 / 9

字 数 / 220 千字

版 次 / 2016 年 8 月第 1 版 2016 年 8 月第 1 次印刷

定 价 / 42.00 元

责任编辑 / 王玲玲

文案编辑 / 王玲玲

责任校对 / 周瑞红

责任印制 / 李志强

前 言

本书按照嵌入式系统运行流程——搭建开发平台、安装系统引导、裁剪内核、构建根文件系统、开发应用程序划分模块，每一模块将理论和原理融于实际任务中，采用任务驱动，注重培养学生逻辑思维，加强引导学生主观能动性。任务选择由易到难、由简到繁，任务具有实用性、趣味性、综合性、典型性，融入嵌入式技术与应用专业培养的典型岗位——嵌入式系统应用开发工程师岗位能力要求，结合教师系统架构高级工程师资格和北京百科融创教学仪器设备有限公司真实案例，注重理论与实践结合，通过真实任务的实现，帮助学生全方位学习掌握嵌入式 Linux 应用开发。

本书主要特色与创新：

1. 校企合作开发，由北京百科融创教学仪器设备有限公司工程师提供真实项目及实训指导。
2. 以嵌入式 Linux 应用开发工程师岗位定课程，具有针对性，实用性强，为学生将来工作打下基础。
3. 以嵌入式系统运行流程划分模块，有利于学生了解嵌入式系统的工作原理、工作流程，以便从系统的本质来了解系统，使学生知其然并知其所以然，有利于学生可持续性发展。
4. 以任务为载体，融合知识点和技能，符合学生特点，有利于学生清楚学习目标、学习角度、学习依据和学习动力，有利于激发学生的学习兴趣，培养学生分析问题、解决问题的能力。
5. 任务的选取由易到难，由简到繁，符合学生认知规律；具有实用性、趣味性、典型性。选择的嵌入式产品软硬件结合，以嵌入式软件开发为主，结合硬件，举一反三，使学生熟能生巧。

教材内容提要如下。

模块一：搭建嵌入式 Linux 开发平台

模块二：安装嵌入式 BootLoader

模块三：ARM——Linux 内核

模块四：嵌入式文件系统

模块五：开发嵌入式 Linux 应用程序

本书由马丽洁任主编，并进行了统稿工作，北京百科融创教学仪器设备有限公司提供真实案例，同时得到冯尚、鲁娅妮的悉心指导，在此表示衷心的感谢！

由于编者水平有限，书中难免有疏漏和不足之处，恳请广大读者及专家批评指正。

编 者

目 录

模块一：搭建嵌入式 Linux 开发平台	1
任务 1：嵌入式系统概述	1
任务 2：熟悉 UP - NET2410 - S 硬件开发平台	3
任务 3：搭建嵌入式应用程序开发平台	12
任务 4：熟悉嵌入式 Linux 应用程序开发环境	25
任务 5：嵌入式 Linux 系统及应用程序的烧写	26
实训一 构建和熟悉嵌入式 Linux 开发平台	28
模块二：安装嵌入式 BootLoader 技术	32
任务 1：系统启动	32
任务 2：BootLoader	33
任务 3：GRUB 引导加载程序	34
任务 4：引导加载程序 vivi	36
实训二 BootLoader	39
模块三：ARM——Linux 内核	41
任务 1：Linux 操作系统功能	41
任务 2：Linux 的内核的核心代码结构	44
任务 3：ARM——Linux 的内存管理	47
任务 4：嵌入式 Linux 进程管理与调度	53
任务 5：Linux 的模块化机制	67
实训三 内核	69
模块四：嵌入式文件系统	71
任务 1：文件系统	71
任务 2：ext2 文件系统	72
任务 3：嵌入式文件系统	77
任务 4：构建 Linux 根文件系统	78
实训四 根文件系统	82
模块五：开发嵌入式 Linux 应用程序	83
任务 1：对讲机	83
任务 2：ATM 取款机	88
任务 3：聊天室	96
任务 4：数码相框	101

任务 5：探月车	108
实训五 嵌入式 Linux 应用开发	112
附录一：中国电子学会《嵌入式系统 Linux 工程师》认证考试参考题	117
附录二：Source Insight 使用说明	124
附录三：Linux 操作系统的安装	131

模块一：搭建嵌入式 Linux 开发平台

任务 1：嵌入式系统概述

子任务 1：嵌入式系统架构

嵌入式系统本身是一个外延极广的名词，凡是与产品结合在一起的具有嵌入式特点的控制
系统都可以叫嵌入式系统，很难给它下一个准确的定义。总体上嵌入式系统可以划分成硬
件和软件两部分，硬件一般由高性能的微处理器和外围的接口电路组成，软件一般由嵌入式
操作系统和其上运行的应用软件构成，软件和硬件之间由中间层连接，中间层一般包括板级
支持包（Board Support Package，BSP）和硬件抽象层（Hardware Abstraction Layer，HAL）。
嵌入式系统架构如图 1-1 所示。

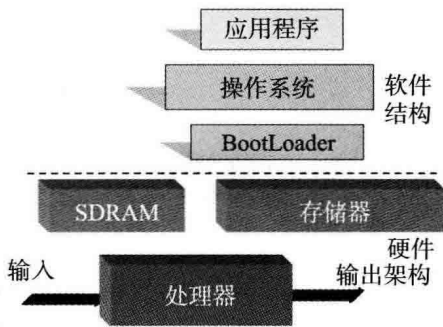


图 1-1 嵌入式系统架构

子任务 2：嵌入式系统特性

1. 系统内核小

由于嵌入式系统一般是应用于小型电子装置的，系统资源相对有限，所以嵌入式系统内
核与 PC 的操作系统相比要小得多。

2. 专用性强

嵌入式系统的个性化很强，其中的软件系统和硬件结合得非常紧密，一般要针对硬件进
行系统移植。

即使在同一品牌、同一系列的产品中，也需要根据系统硬件的变化和增减不断进行修改。

同时，针对不同的任务，往往需要对系统进行较大更改，程序的编译下载要和系统相结
合，这种修改和通用软件的“升级”是完全不同的概念。

3. 系统精简

嵌入式系统一般没有系统软件和应用软件的明显区分，不要求其功能设计及实现上过于复杂，这样一方面利于控制系统成本，同时也利于实现系统安全。

4. 高实时性

这是嵌入式软件的基本要求，而且软件要求固化存储，以提高速度。软件代码要求高质量和高可靠性、高实时性。

5. 嵌入式软件开发走向标准化，必须使用多任务操作系统

嵌入式系统的应用程序可以没有操作系统，直接在芯片上运行。

为了合理地调度多任务、利用系统资源、系统函数以及和专用库函数接口，用户必须自行选配 RTOS (Real - Time Operating System) 开发平台，这样才能保证程序执行的实时性、可靠性，并减少开发时间，保障软件质量。

子任务 3：嵌入式开发模式——交叉编译

开发时往往有主机和目标机的概念，主机用于程序的开发，目标机作为最后的执行机，开发时需要交替结合进行。嵌入式开发模式如图 1-2 所示。

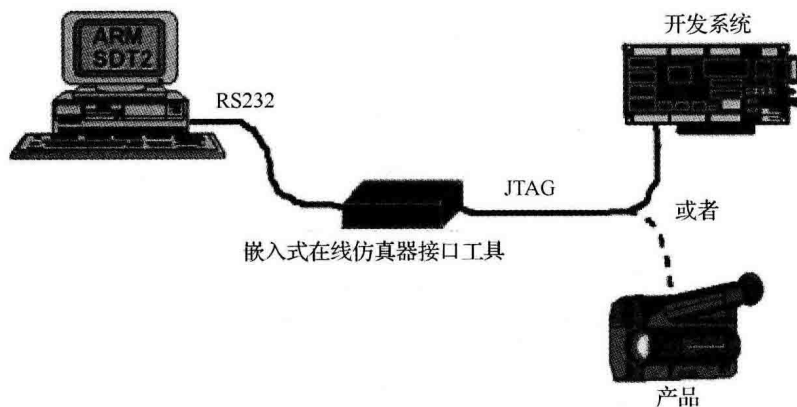


图 1-2 嵌入式开发模式

由于其本身不具备自主开发能力，即使设计完成以后，用户通常也不能对其中的程序功能进行修改，必须有一套开发工具和环境才能进行开发。

什么是交叉编译呢？简单地说，就是在一个平台上生成另一个平台上的可执行代码。

因为目的平台上不允许或不能够安装我们所需要的编译器，而我们又需要这个编译器的某些特征；有时是因为目的平台上的资源贫乏，无法运行我们所需要编译器；有时又是因为目的平台还没有建立，连操作系统都没有，根本谈不上运行什么编译器。

子任务 4：嵌入式系统≠单片机系统

目前嵌入式系统的主流是以 32 位嵌入式微处理器为核心的硬件设计和基于实时操作系统 (RTOS) 的软件设计。

单片机系统多为 4 位、8 位和 16 位机，不适合运行操作系统，难以进行复杂的运算及

处理功能。

嵌入式系统强调基于平台的设计、软硬件协同设计，单片机大多采用软硬件流水设计。

嵌入式系统设计的核心是软件设计（占 70% 左右的工作量），单片机系统软硬件设计所占比例基本相同。

任务 2：熟悉 UP - NET2410 - S 硬件开发平台

子任务 1：知道 ARM

ARM 全称为 Advanced RISC Machines（高级 RISC 微处理器设计）。

一、ARM 公司的业务模式

ARM 公司不生产芯片，而是采取出售芯片 IP（Intellectual Property）核授权的方式扩大其影响力。

世界各大半导体生产厂商从 ARM 公司购买芯片设计方案以及授权，然后根据各自不同的需要，针对不同的应用领域添加适当的外围电路，从而生产出自己的 ARM 微处理器芯片。

ARM 公司提供基于 ARM 架构的开发设计技术（如图 1-3 所示）、软件工具、评估板、调试工具、应用软件、总线架构、外围设备单元等。



图 1-3 ARM 知识产权的应用

二、ARM 处理器的应用

(1) 当前主要应用于消费类电子领域。目前为止，基于 ARM 技术的微处理器应用约占据了 32 位嵌入式微处理器 80% 以上的市场份额。

(2) 全球 90% 的 GSM/3G 手机、99% 的 CDMA 手机以及绝大多数 PDA 产品均采用 ARM 体系的嵌入式处理器。

(3) “掌上计算”相关领域皆为 ARM 所主宰。

三、ARM 处理器的特点

- (1) 体积小、低功耗、低成本、高性能；
- (2) 支持 Thumb (16 位) /ARM (32 位) 双指令集，能很好地兼容 8 位/16 位器件；
- (3) 大量使用寄存器，指令执行速度更快；
- (4) 大多数数据操作都在寄存器中完成；
- (5) 指令长度固定；
- (6) 寻址方式灵活简单，执行效率高。

四、ARM 芯片选择的一般原则

从应用的角度，对在选择 ARM 芯片时所应考虑的主要因素做详细的说明。

1. ARM 核芯

如果希望使用 WinCE 或 Linux 等操作系统以减少软件开发时间，就需要选择 ARM720T 以上带有 MMU (Memory Management Unit) 功能的 ARM 芯片。

2. 系统时钟控制器

系统时钟决定了 ARM 芯片的处理速度。ARM7 的处理速度为 0.9 MIPS/MHz，常见的 ARM7 系统主时钟为 20 ~ 133 MHz；ARM9 的处理速度为 1.1 MIPS/MHz，常见的 ARM9 系统主时钟为 100 ~ 233 MHz；ARM10 最高可以达到 700 MHz。

3. 内部存储器容量

在不需要大容量存储器时，可以考虑选用有内置存储器的 ARM 芯片。

4. USB 接口

许多 ARM 芯片内置有 USB 控制器，有些芯片甚至同时有 USB Host 和 USB Slave 控制器。

5. GPIO 数量

在某些芯片供应商提供的说明书中，往往申明的是最大可能的 GPIO 数量，但是有许多引脚是和地址线、数据线、串口线等引脚复用的。这样，在系统设计时，需要计算实际可以使用的 GPIO 数量。

6. 中断控制器

ARM 内核只提供快速中断 (FIQ) 和标准中断 (IRQ) 两个中断向量。但各个半导体厂家在设计芯片时加入了不同的中断控制器，以便支持诸如串行口、外部中断、时钟中断等硬件中断。外部中断控制是选择芯片必须考虑的重要因素，合理的外部中断设计可以很大程度上减少任务调度的工作量。

7. LCD 控制器

有些 ARM 芯片内置 LCD 控制器，有的甚至内置 64 KB 彩色 TFT LCD 控制器。在设计 PDA 和手持式显示记录设备时，选用内置 LCD 控制器的 ARM 芯片较为适宜。

8. 扩展总线

大部分 ARM 芯片具有外部 SDRAM 和 SRAM 扩展接口，不同的 ARM 芯片可以扩展的芯片数量即片选线数量不同，外部数据总线有 8 位、16 位或 32 位。某些特殊应用的 ARM 芯片如德国 Micronas 的 PUC3030A 没有外部扩展功能。

9. DSP 协处理器

DSP (Digital Signal Processor, 数字信号处理器) 是一种专门用来处理数字信号的处理器，据有独特的 DSP 数字处理指令和硬件乘法器，可以加快数据处理速度。DSP 通常被用在一些需要做大量数据处理的应用场景中，例如音乐编解码、图像编解码、视频编解码。因此，DSP 数字处理器常见于音响、相机、录像机、视频播放、视频会议等产品中。

有些芯片厂商为了满足市场需求，将 ARM 和 DSP 两种核心集成在一个芯片中，制作出双核 CPU。因此，选择芯片的时候，需要考虑自己的产品中是否需要 DSP 数字信号处理器。

10. 封装

主要的封装有 QFP、TQFP、PQFP、LQFP、BGA、LBGA 等形式。BGA 封装具有芯片面积小的特点，可以减小 PCB 板的面积，但是需要专用的焊接设备，无法手工焊接。另外，一般 BGA 封装的 ARM 芯片无法用双面板完成 PCB 布线，需要多层 PCB 板布线。

子任务 2：S3c2410 处理器概述

韩国三星公司推出的基于 ARM920T 内核和 AMBA 总线的 S3c2410 处理器，使用 0.18 μm CMOS 标准宏单元和存储器单元工艺，集成 LCD、UART、IIC、SPI、IIS、USB、SD 控制器等片内外围设备，支持 ROM 和 NAND Flash 引导。其适合面向手持设备应用，功耗较低；适合成本低廉、应用环境较好的消费类电子产品。

一、S3c2410 内部模块——ARM920T 内核

ARM920T 内核是 ARM 公司的 16/32 位 RISC 结构处理器，采用 ARMV4 指令集，体系结构为数据、地址总线分离的哈佛体系结构，16 KB 指令缓存，16 KB 数据缓存，支持 MMU，可运行 WinCE、Linux 等操作系统，外部总线采用 AMBA 总线，集成基于 JTAG 协议的片内调试 (ICE) 单元。

二、S3c2410 内部模块——AMBA 总线

AMBA (Advanced Microcontroller Bus Architecture) 总线规范是 ARM 公司设计的一种用于高性能嵌入式系统的总线标准。它是一个开放标准，可免费从 ARM 获得；在基于 ARM 处理器内核的 SoC 设计中，其已经成为事实上的工业标准；AMBA 总线是一个多总线系统。规范定义了三种可以组合使用的不同类型的总线：

- (1) AHB (Advanced High - performance Bus)；
- (2) ASB (Advanced System Bus)；
- (3) APB (Advanced Peripheral Bus)，该总线已经逐渐被废弃。

AHB 主要用以满足 CPU 和存储器之间的大带宽要求，而系统的大部分低速外部设备则连接在低带宽总线 APB 上。系统总线和外设总线之间用一个桥接器（AHB - APB - Bridge）进行连接。

三、封装与电压、工作温度

封装：272 脚 FBGA，焊盘中心间距 0.8 mm，如图 1-4 所示。

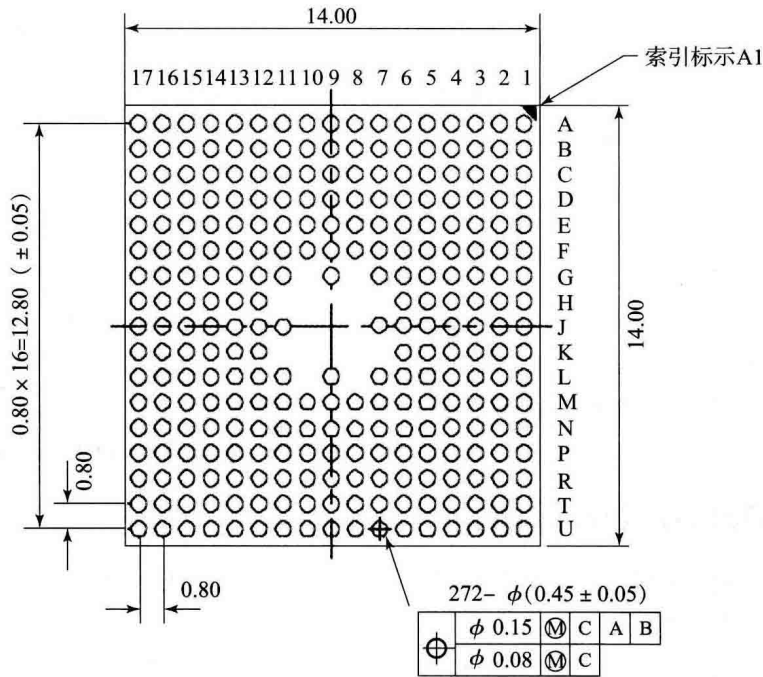


图 1-4 芯片的封装

电压：内核电压：1.8 V；存储器和 I/O 电压：3.3 V
工作温度：商业级：0℃ ~ 70℃；工业级：-40℃ ~ 85℃。

子任务 3：UP - NETARM2410 - S 硬件平台

一、UP - NETARM2410 - S 平台硬件环境架构

UP - NETARM2410 - S 平台硬件环境架构如图 1-5 所示。
UP - NETARM2410 - S 教学科研系统如图 1-6 所示。

1. 核心板（如图 1-7 所示）

- (1) CPU：ARM920T 结构芯片，工作频率 202 MHz，SAMSUNG 公司的 S3c2410X。
- (2) FLASH：64 MB NAND 型；SAMSUNG 的 K9F1208。
- (3) RAM：64 MB SDRAM 型；HY57V561620AT - H。
- (4) 200 管脚精密插座。

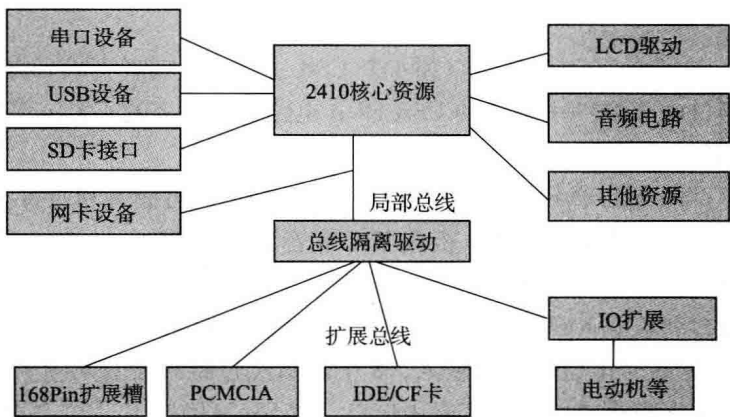


图 1-5 UP-NETARM2410-S 平台硬件环境架构

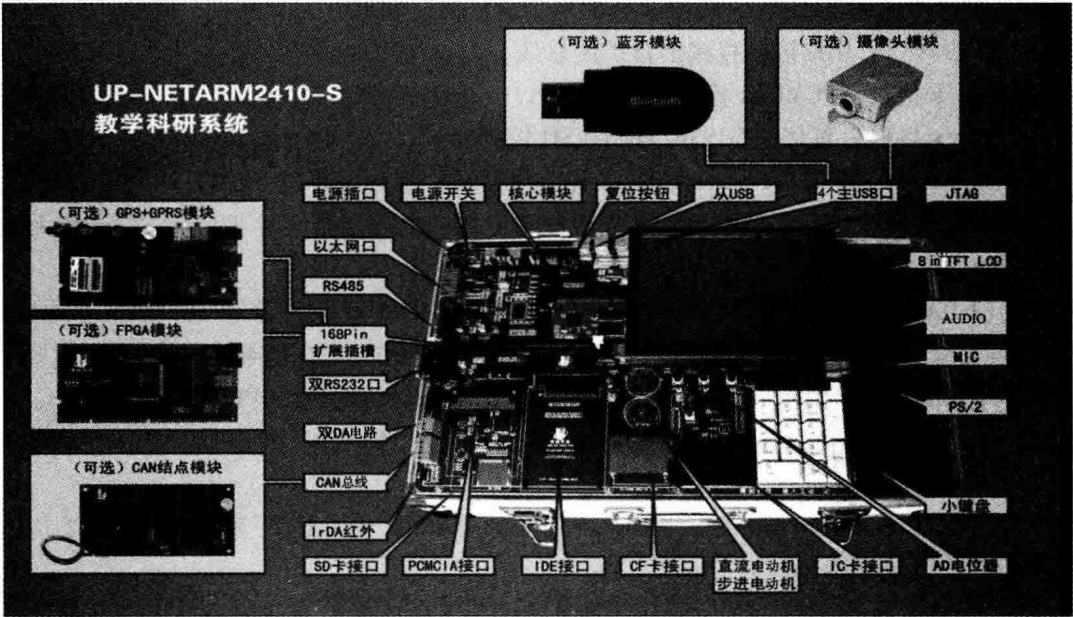


图 1-6 UP-NETARM2410-S 教学科研系统

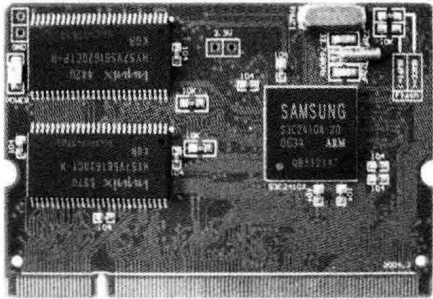


图 1-7 ARM2410 核心板

① 1 in = 2.54 cm。

2. 主板

(1) USB 口：包括 1 个 USB 从接口和 4 个 USB 主接口。前者直接来自处理器，后者是处理器的主接口经过由 AT43301 构成的 USB HUB 电路扩展出来的 4 个主接口。

(2) JTAG 调试口：20 针标准口，14 针简易口。

(3) AUDIO：44.1 kHz 音频，采用 UDA1341 芯片构成，可放音和录音。通过 IIS 总线连接到处理器，具有功放电路驱动喇叭、板载 MIC 和音频 IO 插座。

(4) ATMEGA8 单片机：2410-s 平台上通过一片 ATMEGA8 单片机作为控制器，驱动 PS2 接口、IC 卡接口和板载 KEYPAD。ATMEGA8 单片机通过 IIC 总线和 ARM 处理器连接。

(5) AD 电位器：3 个电位器控制输入，8 个通道经过跳线设置可全部作为外部模拟电压输入，同时，在板上设模拟电压输入专用接口，由 S3c2410 芯片管脚引出。

(6) 直流/步进电动机：直流电动机信号来自 PWM 输出，步进电动机信号来自 EXIO 扩展，EXIO 扩展电路由 74HC573 构成，分配专用地址段，和 IDE 接口在同一通道。

(7) IDE 接口：IDE 接口直接出自扩展总线，由软件完成驱动控制。

(8) CF 卡接口：PCMCIA 模式的 CF 卡接口。

(9) SD 卡接口：直接来自 S3c2410 的 SD 控制器。

(10) PCMCIA 接口：PCMCIA 接口挂在扩展总线上，通过 EPM3128A100 CPLD 和 HC245 芯片等实现总线隔离和控制，并需要配置专用的电源控制芯片 TPS2211。采用无线网卡的型号为 DWL-650。

(11) IrDA/RS485：用的是 UART2，分别经过 TFDU4100 收发器和 MAX485 芯片完成。要做 IrDA 实验，必须将 UART2 设置到 IrDA 电路上。不能同时做 RS485 实验。

(12) CAN 总线：由控制器 MCP2510 和驱动器 TJA1050 构成，SPI 总线接口。

(13) 双 DA 电路：设置两个 10 位 DAC 端口，采用 MAX504 接 SPI 总线，输出两路模拟电压（预留一路）。

(14) 串口：包括两个 RS232 接口，处理器的 UART0 和 UART1 经过 MAX3232 芯片电压转换后用 DB9 插座输出。

(15) 168Pin 扩展插槽：该插槽和计算机的内存条插槽规格一致，用户可以据此设计合适的扩展板。扩展槽上包含扩展总线的所有信号、可复用的系统资源以及开发平台尚未使用的空置资源。

(16) 以太网口：包括两个相同的网卡电路，芯片型号 AX88796，10M/100M 自适应（预留一个）。

3. 显示装置

(1) LCD：8 英寸 16 bit TFT，640×480 像素分辨率，型号是 LQ080V3DG01。S3c2410 的处理器内含 LCD 控制器，这使得 LCD 部分的电路非常简单。LCD 控制器的信号线经过驱动电路后即可连接 LCD 模块。

(2) 触摸屏：采用 2410 CPU AD 单元的接口方案。

4. 2410-s 电源

电压为 5 V，经 LM1085-3.3 V 和 AS1117-1.8 V 分别得到 3.3 V 和 1.8 V 的工作电压。

5. 开发板上的芯片

多数使用了 3.3 V 电压，而 1.8 V 是供给 S3c2410 内核使用的，5 V 电压供给音频功放芯片、LCD、电动机、硬盘、CAN 总线等电路使用。

二、配置目标机

1. Windows 环境

单击“开始”→“附件”→“通讯”→“超级终端或安装超级终端”。

配置：波特率 115 200，数据位 8 位，停止位 1，无奇偶校验，软硬件控制流为无。

请大家注意：在 Windows XP 操作系统下，当初次建立超级终端的时候，会出现如图 1-8 所示对话框。

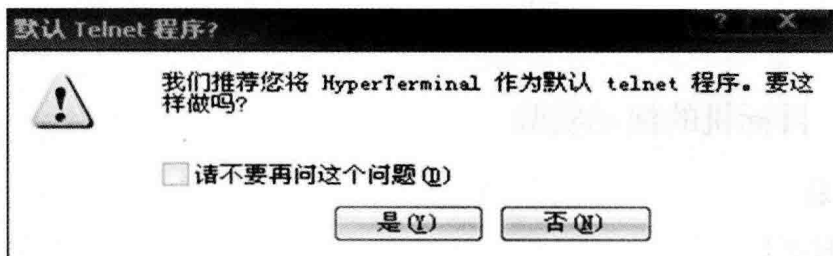


图 1-8 询问对话框

在□中打上√，并单击“否”按钮，出现如图 1-9 所示的对话框，输入自定义的终端名称，选择一个图案，终端就创建成功了。



图 1-9 新建终端对话框

2. Linux 环境

输入命令“minicom”，按 Ctrl + A + Z + O 组合键。

新建超级终端，配置 MINICOM，设置波特率 115 200 b/s，数据位 8，停止位 1，奇偶校验；无，数据流控制：无，软硬件控制流为无，如图 1-10 所示。

MINICOM 的作用是作为调试嵌入式开发板信息输出的监视器和键盘输入的工具。

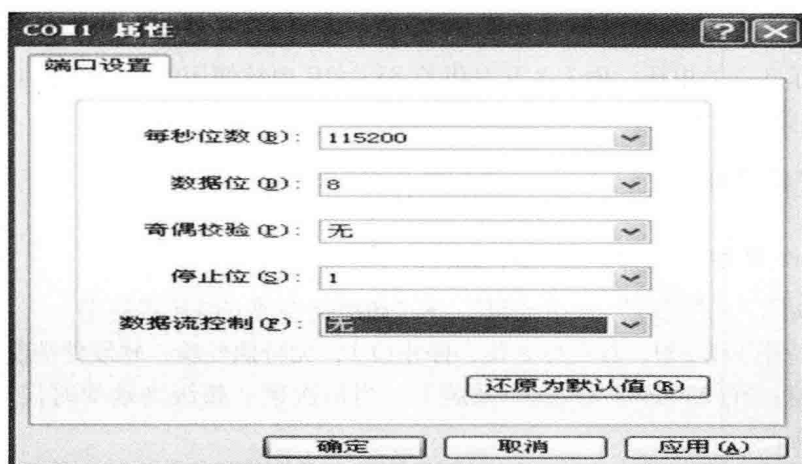


图 1-10 端口设置

子任务 2：目标机的演示实验

1. da 实验

进入 da 目录下：

```
[ /mnt /yaffs]ls
[ /mnt /yaffs]cd da/
[ /mnt /yaffs/da]ls
```

显示 da 目录下内容：

```
da_main exio.o
[ /mnt /yaffs/da]insmod exio.o
```

安装 da 模块驱动：

```
[ /mnt /yaffs/da]da_main 0 2.2
```

向 0 路输出 2.2 V 电压。

2. 演示 ad

```
[ /mnt /yaffs]cd ad/
[ /mnt /yaffs/ad]ls
ad s3c2410 -adc.o
[ /mnt /yaffs/ad]insmod s3c2410 -adc.o
[ /mnt /yaffs/ad]. /ad
```

调节 3 路 ad 即可改变大小。

```
Press Enter key exit!
a0 =3.280 7 a1 =3.296 8 a2 =0.8153
```

注意：在 ad 实验前请用 `lsmod` 查看已加载的模块，如果存在 `ts - uptech`，则需要用 `rmmod` 卸载。

3. 电动机的演示

打开电动机的红色按钮开关。

(1) 直流电动机：

```
[ /mnt /yaffs]cd motor /  
[ /mnt /yaffs /motor]ls  
DC STEP  
[ /mnt /yaffs /motor]cd DC /
```

注意大小写。

```
[ /mnt /yaffs /motor /DC]ls  
dc -motor.o dcm_main  
[ /mnt /yaffs /motor /DC]insmod dc -motor.o  
Using dc -motor.o s3c2410_dcm_init(159):s3c2410 - dc -motor device  
initialized  
[ /mnt /yaffs /motor /DC]. /dcm_main
```

启动直流电动机。

(2) 步进电动机：

```
[ /mnt /yaffs]cd motor /  
[ /mnt /yaffs /motor]ls  
DC STEP  
[ /mnt /yaffs /motor]cd STEP /  
[ /mnt /yaffs /motor /STEP]ls  
exio.o stepmotor  
[ /mnt /yaffs /motor /STEP]insmod exio.o  
Using exio.o  
[ /mnt /yaffs /motor /STEP]. /stepmotor
```

启动步进电动机。

4. 播放 mp3

```
[ /mnt /yaffs]cd sound /  
[ /mnt /yaffs /sound]ls  
1.wav madplay ttdl.mp3 wavplay wavrec  
[ /mnt /yaffs /sound]madplay ttdl.mp3  
MPEG Audio Decoder 0.14.2(beta) - Copyright(C)2000 -2001 Robert Leslie
```