

## <<从实践中学嵌入式Linux C编程>>

### 图书基本信息

书名：<<从实践中学嵌入式Linux C编程>>

13位ISBN编号：9787121158834

10位ISBN编号：7121158833

出版时间：2012-3

出版时间：电子工业出版社

作者：华清远见嵌入式学院 编著

页数：302

版权说明：本站所提供下载的PDF图书仅提供预览和简介，请支持正版图书。

更多资源请访问：<http://www.tushu007.com>

## <<从实践中学嵌入式Linux C编程>>

### 内容概要

《从实践中学嵌入式linux c编程》详细介绍了开发工具和linux c语言基础、嵌入式linux c语言高级编程、内核常见数据结构的解析与应用、嵌入式linux编程基础、文件i/o操作相关的c语言应用及网络通信相关的c语言应用等，并设置了嵌入式linux c函数参考附录。

本书是大学院校嵌入式技术专业、电子信息类其他专业的专业课程教材，也可供高等及中等职业技术院校使用。

# <<从实践中学嵌入式Linux C编程>>

## 书籍目录

### 第1章 嵌入式Linux C语言开发工具

#### 1.1 嵌入式Linux C语言概述

##### 1.1.1 C语言简史

##### 1.1.2 C语言特点

##### 1.1.3 嵌入式Linux C语言编程环境

#### 1.2 嵌入式Linux编辑器vi的使用

##### 1.2.1 vi的基本模式

##### 1.2.2 vi的基本操作

##### 1.2.3 vi的使用实例分析

#### 1.3 嵌入式Linux编译器GCC的使用

##### 1.3.1 GCC概述

##### 1.3.2 GCC编译流程分析

##### 1.3.3 GCC警告提示

##### 1.3.4 GCC使用库函数

##### 1.3.5 GCC代码优化

#### 1.4 嵌入式Linux调试器GDB的使用

##### 1.4.1 GDB使用实例

##### 1.4.2 设置/删除断点

##### 1.4.3 数据相关命令

##### 1.4.4 调试运行环境相关命令

##### 1.4.5 堆栈相关命令

#### 1.5 make工程管理器

##### 1.5.1 Makefile基本结构

##### 1.5.2 Makefile变量

##### 1.5.3 Makefile规则

##### 1.5.4 make使用

#### 1.6 Eclipse集成开发环境

##### 1.6.1 Eclipse 简介

##### 1.6.2 Eclipse相关术语

##### 1.6.3 安装Eclipse集成开发环境（假设宿主机环境为ubuntu10.10）

##### 1.6.4 Eclipse的使用

#### 1.7 本章小结

#### 1.8 本章习题

### 第2章 嵌入式Linux C语言基础

#### 2.1 ANSI C与GNU C

##### 2.1.1 ANSI C简介

##### 2.1.2 GNU C简介

#### 2.2 基本数据类型

##### 2.2.1 整型家族

##### 2.2.2 实型家族

##### 2.2.3 字符型家族

##### 2.2.4 枚举家族

##### 2.2.5 指针家族

#### 2.3 变量与常量

##### 2.3.1 变量的定义

## <<从实践中学嵌入式Linux C编程>>

2.3.2 typedef

2.3.3 常量定义

2.4 运算符与表达式

2.4.1 算术运算符和表达式

2.4.2 赋值运算符和表达式

2.4.3 逗号运算符和表达式

2.4.4 位运算符和表达式

2.4.5 关系运算符和表达式

2.4.6 逻辑运算符和表达式

2.4.7 sizeof操作符

2.4.8 条件运算符(?)

2.4.9 运算符优先级总结

2.5 程序结构和控制语句

2.5.1 C语言程序结构

2.5.2 C语言控制语句

2.6 数组、结构体和指针

2.7 函数

2.7.1 概述

2.7.2 函数定义和声明

2.7.3 函数的参数、返回值和调用方法

2.8 \_attribute\_ 机制介绍

2.9 系统调用和应用程序编程接口

2.9.1 系统调用

2.9.2 应用程序编程接口

2.9.3 系统命令

2.10 本章小结

2.11 本章习题

第3章 嵌入式Linux C语言高级用法

3.1 预处理

3.1.1 预定义

3.1.2 文件包含

3.1.3 条件编译

3.2 C语言中的内存分配

3.2.1 C语言程序所占内存分类

3.2.2 堆和栈的区别

3.3 程序的可移植性考虑

3.3.1 字长和数据类型

3.3.2 数据对齐

3.3.3 字节顺序

3.4 C和汇编的接口

3.4.1 内嵌汇编的语法

3.4.2 编译器优化介绍

3.4.3 C语言关键字volatile

3.4.4 memory描述符

3.5 本章小结

3.6 本章习题

第4章 嵌入式Linux C内核常用数据结构

## <<从实践中学嵌入式Linux C编程>>

### 4.1 链表

#### 4.1.1 单向链表

#### 4.1.2 双向链表

#### 4.1.3 循环链表

#### 4.1.4 ARM Linux中链表使用实例

### 4.2 树、二叉树、平衡树

#### 4.2.1 树的定义

#### 4.2.2 二叉树

#### 4.2.3 平衡树

#### 4.2.4 ARM Linux中红黑树使用实例

### 4.3 哈希表

#### 4.3.1 哈希表的概念及作用

#### 4.3.2 哈希表的构造方法

#### 4.3.3 哈希表的处理冲突方法

#### 4.3.4 ARM Linux中哈希表使用实例

### 4.4 本章小结

### 4.5 本章习题

## 第5章 嵌入式Linux文件操作

### 5.1 Linux文件系统概述

#### 5.1.1 虚拟文件系统

#### 5.1.2 通用文件模型

#### 5.1.3 Linux下的设备文件

### 5.2 Linux下的I/O操作

#### 5.2.1 不带缓存的文件I/O操作

#### 5.2.2 标准I/O

### 5.3 Linux下对文件和目录的操作

#### 5.3.1 文件类型

#### 5.3.2 文件访问权限

#### 5.3.3 获取文件属性

#### 5.3.4 修改文件访问权限

#### 5.3.5 创建目录

#### 5.3.6 创建链接文件

#### 5.3.7 删除文件

#### 5.3.8 重命名文件

### 5.4 嵌入式Linux串口应用开发

#### 5.4.1 串口概述

#### 5.4.2 串口设置详解

#### 5.4.3 串口使用详解

### 5.5 本章小结

### 5.6 本章习题

## 第6章 嵌入式Linux进程和线程编程

### 6.1 Linux进程概述

#### 6.1.1 进程描述符及任务结构

#### 6.1.2 进程的调度

#### 6.1.3 Linux中的线程

### 6.2 Linux进程控制相关API

## <<从实践中学嵌入式Linux C编程>>

### 6.3 ARM Linux进程间通信

#### 6.3.1 管道通信

#### 6.3.2 信号通信

#### 6.3.3 共享内存

#### 6.3.4 消息队列

### 6.4 ARM Linux线程相关API

### 6.5 Linux守护进程

#### 6.5.1 守护进程概述

#### 6.5.2 编写规则

#### 6.5.3 守护进程实例

### 6.6 本章小结

### 6.7 本章习题

## 第7章 网络通信相关的C语言应用

### 7.1 TCP/IP协议简介

#### 7.1.1 TCP/IP的分层模型

#### 7.1.2 TCP/IP分层模型的特点

#### 7.1.3 TCP/IP核心协议

### 7.2 套接字的基本知识

#### 7.2.1 套接字概述

#### 7.2.2 地址结构和字节序

### 7.3 套接字相关的API及应用

#### 7.3.1 socket函数

#### 7.3.2 bind函数

#### 7.3.3 connect函数

#### 7.3.4 listen函数

#### 7.3.5 accept函数

#### 7.3.6 send、recv函数

#### 7.3.7 sendto和recvfrom函数

#### 7.3.8 close和shutdown函数

#### 7.3.9 setsockopt和getsockopt函数

#### 7.3.10 getpeername函数

#### 7.3.11 gethostname函数

#### 7.3.12 编程实例

### 7.4 套接字高级编程

### 7.5 本章小结

### 7.6 本章习题

## 附录A 嵌入式Linux C函数快速参考

章节摘录

版权页：第1章 嵌入式Linux C语言开发工具从实践中学嵌入式Linux C编程任何应用程序的开发都离不开编辑器、编译器及调试器，嵌入式Linux的C语言开发也一样，需要拥有一套优秀的编辑、编译及调试工具。

掌握这些工具的使用是至关重要的，它直接影响到程序开发的效率。

希望读者通过自己的实践，熟练掌握这些工具的使用。

本章主要内容 C语言产生的历史背景 嵌入式Linux下C语言的开发环境 嵌入式Linux下的编辑器vi 嵌入式Linux下的编译器GCC 嵌入式Linux下的调试器GDB 嵌入式Linux下的工程管理器make 如何使用autotools来生成Makefile 嵌入式Linux下的综合编辑器Emacs1.1 嵌入式LinuxC语言概述在嵌入式系统中，应用程序的主体是在宿主机中开发完成的。

就嵌入式Linux而言，此过程通常在安装有Linux的宿主机中完成。

本章中介绍的是嵌入式Linux下C语言的开发工具，用户在开发时先在Linux宿主机中对程序进行调试，然后再进行交叉编译。

1.1.1 C语言简史C语言于20世纪70年代诞生于美国的贝尔实验室。

在此之前，人们编写系统软件时主要使用汇编语言。

汇编语言编写的程序依赖于计算机硬件，其可读性和可移植性都比较差。

而高级语言的可读性和可移植性虽然较汇编语言好，但一般高级语言不具备低级语言能够直观地对硬件实现控制和操作而且执行速度快等特点。

在这种情况下，人们迫切需要一种既具有一般高级语言特性，又具有低级语言特性的语言，于是C语言就应运而生了。

由于C语言既具有高级语言的特点，又具有低级语言的特点，因此迅速普及，成为当今最有发展前途的计算机高级语言之一。

C语言既可以用来编写系统软件，也可以用来编写应用软件。

现在，C语言已经被广泛地应用于计算机、机械、建筑、电子等各个行业。

## <<从实践中学嵌入式Linux C编程>>

### 编辑推荐

《高等院校嵌入式人才培养规划教材:从实践中学嵌入式Linux C编程》作为嵌入式Linux开发的主要编程语言，C语言是嵌入式开发工程师的必备基础。

《高等院校嵌入式人才培养规划教材:从实践中学嵌入式Linux C编程》从嵌入式Linux环境下C语言的开发工具入手，通过大量的代码和实例分析，引领读者逐步掌握嵌入式Linux平台上C语言编程的核心知识和技能。

版权说明

本站所提供下载的PDF图书仅提供预览和简介，请支持正版图书。

更多资源请访问:<http://www.tushu007.com>