

《深度探索嵌入式操作系统》

图书基本信息

书名：《深度探索嵌入式操作系统》

13位ISBN编号：9787111514874

出版时间：2015-10

作者：彭东

页数：527

版权说明：本站所提供下载的PDF图书仅提供预览和简介以及在线试读，请支持正版图书。

更多资源请访问：www.tushu111.com

《深度探索嵌入式操作系统》

内容概要

相信很多同行和笔者以前一样，阅读过很多操作系统理论方面的书籍，感觉它们所讲述的理论大同小异且比较容易理解，但是真要我们自己去动手开发和实现操作系统时，我们可能连操作系统的第一行代码该怎么写都不知道。为什么会这样？

因为操作系统是硬件平台上的第一层软件，若不了解硬件平台的细节，是不能实现操作系统的，所以本书一开始就选择了一款常用的硬件平台，并对其进行了详细描述，重点介绍了硬件平台上的实时时钟、定时器、串口、中断控制器、内存芯片、flash芯片、CPU、MMU等，然后介绍操作系统架构设计和开发环境与开发工具，尤其开发工具，详细了解开发工具会大大方便后面的开发工作并解决一些困惑，最后介绍了操作系统的初始化、HAL、内存管理、中断管理、设备管理、进程管理、文件系统、系统调用接口、应用程序库。对这些组件的介绍都是先明确这个组件要完成什么功能，达到什么要求。接着详述这个组件必须要关注的硬件的细节，然后把我们的想法和设计归纳成数据结构。最后编写完成这些功能的代码。

这些内容将奠定本书在操作系统实践领域中里程碑式的地位。

《深度探索嵌入式操作系统》

作者简介

为什么写这本书

随着移动互联网技术的成熟，物联网也开始崭露头角，由此产生了各种小型、低功耗的智能硬件，这些智能硬件被嵌入到移动电话、手表、冰箱、空调、电视机、洗衣机等常用电子产品中，使这些常用电子产品功能更加强大、更加智能，而且它们可以连接到网络，便于用户远程操控，从而大大改善了人们的生活。

因此，嵌入式行业才变得如此火热，那些被嵌入到电子产品中的智能硬件，也需要一些小巧、特殊的操作系统软件才能正常工作，这类小巧、特殊的操作系统软件，称为嵌入式系统。2013年下半年，我开始学习嵌入式系统，并编写了一个嵌入式操作系统—LMOSEM。在互联网上也认识了不少研究嵌入式的朋友，在他们的要求和规劝下，我终于有勇气把我研究出来的东西归纳、整理成册，也算是我学习的笔记，于是就有了这本书。虽然有很多的顾虑，怕贻笑方家、怕误导同道……但是我的每行代码、每个点子，都在实机上测试过并证明了其正确性，所以也就心下一片坦然了。如果这本书能够被后来者借鉴一二，或者解决他们的一些疑惑，我自然欣慰万分。

关于LMOSEM

关于LMOSEM，这得从LMOS开始说起。2010年下半年，我开始准备要写个操作系统内核，没有其他目的，只是出于学习，出于兴趣。由于是自己独立从零开始设计、编写的，我觉得自己这种行为有点疯狂，索性用LMOS (liberty madness operating system) 命名了我的操作系统。LMOS经过这几年的独立开发，现在已经发布了6个测试版本。先后从32位单CPU架构发展到64位多CPU架构，现在的LMOS已经是多进程、多线程、多CPU、支持虚拟内存的x86_64体系下的操作系统内核。LMOS的这些特性，非常适合通用计算机领域，如PC、工作站、小型服务器。这些特性导致LMOS代码量庞大，一些组件不够小巧，削剪起来非常复杂，很难保证削剪后的组件是否健壮，因此LMOS不适合于嵌入式领域，所以笔者才重新开发了LMOSEM—适合嵌入式领域的操作系统。

LMOSEM依然删除了很多代码，因为写书要做到简单，便于理解。即便如此，LMOSEM依然包含了现代操作系统的大部分重要组件，如内存管理、进程管理、驱动模型、文件系统等。这些组件的实现过程在本书中都会有详细的介绍。LMOSEM不支持实时性功能，嵌入式操作系统也不一定要是实时性的操作系统，何况我们是出于学习的目的。为了代码的清晰、简单，我们暂不考虑安全性和性能方面的问题。等到明白了操作系统原理，我们再去不断修正、优化，使之功能变得更多，性能变得更强。笔者开发的LMOSEM操作系统项目，是在Linux操作系统下开发的，用到了Linux操作系统的很多工具。笔者不会和读者讨论为什么不用常用的Windows系统，也不会说谁好、谁不好。如果读者非常喜欢Windows系统，那么也可以尝试着把这个项目迁移到Windows系统下。但是笔者书中演示的环境还是Linux系统。关于如何搭建开发环境，本书后面的章节有详细的介绍。在那里读者会发现用Linux系统开发LMOSEM内核有很多方便之处，如会用到的MAKE、GCC、LD等，这些工具在Linux系统下都很容易得到，在Windows系统下虽然也能做到，但相对麻烦一点。何况今天的Linux系统已经很好用了。

读者对象

如果读者是一位纯粹的操作系统爱好者，对其有着浓厚的兴趣，那么本书将非常适合。

如果读者是嵌入式领域的从业者或者学生，也可以从本书中获得很多帮助。

如果读者是一位普通的应用软件开发者，业余时间也可以翻翻此书，书中的一些设计方法和编程手段，或许可以借鉴一二。

如果你只是想了解一些计算机硬件系统和软件系统的常识，那么本书同样会让你获益。

如何阅读本书

为了能更轻松地阅读这本书，笔者建议先了解C语言这门编程语言，对数据结构有所了解就更好了。

除这些外，笔者假定读者没有其他任何技能。除了需要的上述技能，读者还需要对操作系统有强大的兴趣和求知欲，要有坚强的意志、永远不放弃的精神。开发操作系统内核本身就不是件容易的事，必然会有很多问题在等着我们，但是遇到问题不要害怕，静下心来从容面对，只要我们不放弃，问题最终会解决。

本书很简单，没有拐弯抹角，没有反复修饰，但是必要的细节从不漏掉。宁可在细节上啰嗦一点，也不在不相关的地方多写一句。

《深度探索嵌入式操作系统》

本书的最终目的是构建一个用于学习的嵌入式操作系统内核，并工作在真正的物理机上。为了达到这一目的，本书大体上分为三部分：综述、硬件部分和软件部分。

综述部分（第1章）。

第1章，先说明操作系统的概念、功能和演进历史，最后得出现代操作系统的模型，使我们可以了解操作系统的轮廓。

硬件部分（第2~3章）。

第2章，从选择硬件平台开始，首先概述硬件平台的整体情况，接着了解编写操作系统内核必需的一些平台上的组件，如实时时钟、定时器、串口、中断控制器、内存芯片、Flash芯片、CPU、MMU等。让读者有初步的印象，在写代码用到某个组件时再详述其内部编程细节。

第3章，详细介绍处理器，重点介绍处理器的结构和特性、处理器的地址空间、处理器的状态和工作模式、处理器的寄存器和指令集。最后介绍处理器中的MMU和Cache，对于MMU，主要介绍MMU的作用和它对操作系统内核开发的影响、如何对MMU编程、MMU的几种地址映射方式。而对于Cache，重点介绍Cache的作用、Cache的类型、Cache的使用。

软件部分（第4~12章）。

第4章，介绍操作系统内核设计、操作系统内核的开发环境、开发操作系统内核的工具：GCC、LD、MAKE，以及它们的使用方法，最后介绍硬件平台的安装与测试。

第5章，首先介绍C语言使用寄存器的约定，以及它是如何处理参数、返回值的。接着介绍C语言基本数据类型的位宽及占用内存的大小，并用它们构建后面将要用到的一些基本的数据结构，如list_h_t、spinlock_t、kwlst_t、sem_t。然后介绍C语言的数据结构在内存中存在的形式、对齐方式。最后介绍GCC独有的嵌入汇编代码的方式。

第6章，开始介绍LMOSEM的初始化，从第一行汇编代码开始、初始化MMU和中断向量、初始化串口设备、初始化内存管理数据结构和中断相应的数据结构，最后对一些数据结构进行测试。

第7章，开始介绍LMOSEM的内存管理组件。LMOSEM的内存管理组件分为三层：块级内存管理、页级内存管理、字级内存管理，这三个层分别应对不同的内存分配请求。本章将结合实际代码介绍如何一步步实现这三大内存管理层。

第8章，介绍LMOSEM的中断管理组件，内容包括中断控制器的细节、中断管理组件的结构、如何处理中断、安装中断处理的回调函数等。

第9章，介绍LMOSEM的设备驱动模型，其中介绍LMOSEM如何管理众多的硬件设备、LMOSEM支持的设备类型、驱动模型的数据结构和基础性代码，最后用两个驱动程序实例介绍如何在LMOSEM的驱动模型下编写规范的设备驱动程序。

第10章，介绍LMOSEM的进程管理组件，包括进程的由来、进程相关的数据结构、系统空转进程的建立与运行、进程调度、新建进程、进程的睡眠与唤醒、进程测试等相关内容。

第11章，介绍LMOSEM的文件系统组件，包括文件系统的设计、文件系统的建立、文件系统的基础操作、文件本身的操作，如文件的打开、新建、读写、删除等。最后对文件系统组件进行严格的测试。

第12章，介绍LMOSEM的接口，包括许多LMOSEM的API和库函数的实现细节，主要包括时间、进程、内存、文件与设备、标准输入/输出等方面的API和库函数。

勘误和支持

由于笔者水平有限，加之编写时间仓促，书中难免会出现一些不准确的地方，恳请读者批评指正，在技术之路上共勉。我的CU博客地址是：<http://blog.chinaunix.net/uid/28032128.html>。本书源代码已制作成光盘镜像文件，并上传到华章网站（www.hzbook.com），需要的读者可自行下载。

致谢

我，自幼患病，读书不多，计算机成了唯一的兴趣爱好，没有父母的长期支持，连生活都尚且不能自理，更别说完成此书了，他们对我的帮助和关爱，纵使千万言语也难表一二。由于经常在物理机上测试内核，要拆装一些设备和器件，这多亏了我的小弟，因为他一有时间就帮我做这部分工作。当然还有帮助过我的朋友，有一些是身边的，有些是网络中的。对父母、所有的亲人、朋友，我也只有常怀感恩之心，说声谢谢，谢谢他们一直的支持、帮助，谢谢他们一直对我那满满的关爱！

让笔者和你一起带着未知，带着好奇，带着兴奋，踏上操作系统的旅程吧！

书籍目录

前言

第1章 操作系统的功能及为什么需要它1

1.1从hello world开始1

1.2 操作系统功能演进3

1.3 小结7

第2章 硬件平台8

2.1 选择平台8

2.1.1 mini24408

2.1.2 mini2440平台的信息9

2.2 必须要关注的硬件13

2.2.1 原因14

2.2.2 RTC15

2.2.3 定时器16

2.2.4 串口17

2.2.5 中断控制器18

2.2.6 SDRAM19

2.2.7 Norflash23

2.2.8 Nandflash24

2.3 小结26

第3章 处理器28

3.1 ARM公司与其处理器28

3.2 ARM920T的结构与特性29

3.2.1 ARM920T CPU结构29

3.2.2 ARM920T CPU特性32

3.3 ARM920T存储体系33

3.3.1 ARM920T地址空间33

3.3.2 ARM920T存储器格式34

3.3.3 ARM920T存储地址对齐35

3.4 ARM920T 状态35

3.4.1 ARM状态36

3.4.2 Thumb 状态36

3.5 ARM920T处理器的7种工作模式37

3.6 寄存器38

3.7 异常和中断43

3.7.1 什么是异常和中断43

3.7.2 异常中断向量46

3.8 ARM920T指令集47

3.8.1 指令及其编码格式48

3.8.2 分支跳转指令50

3.8.3 数据处理指令53

3.8.4 装载和存储指令63

3.8.5 程序状态寄存器操作指令73

3.8.6 协处理器操作指令76

3.8.7 异常中断产生指令79

3.9 MMU80

3.9.1 MMU概述80

3.9.2 为什么要有MMU82

- 3.9.3 ARM920T CP15协处理器85
- 3.9.4 MMU 页表98
- 3.9.5 MMU页面访问权限的控制113
- 3.9.6 MMU的快表TLB113
- 3.9.7 MMU的编程接口114
- 3.10 Cache115
 - 3.10.1 ARM920T的Cache115
 - 3.10.2 Cache的原理116
 - 3.10.3 Cache的类型及要注意的问题117
 - 3.10.4 ARM920T Cache的编程接口119
- 3.11 小结120
- 第4章 操作系统内核的设计与构建122
 - 4.1 操作系统内核的设计122
 - 4.1.1 内核要完成的功能123
 - 4.1.2 内核的架构124
 - 4.1.3 分离硬件的相关性126
 - 4.1.4 我们的选择127
 - 4.2 开发环境及相关工具129
 - 4.2.1 Linux环境129
 - 4.2.2 文本编辑器132
 - 4.2.3 GCC134
 - 4.2.4 LD136
 - 4.2.5 make139
 - 4.3 LMOSEM的构建系统142
 - 4.3.1 LMOSEM的makefile142
 - 4.3.2 LMOSEM的链接脚本147
 - 4.4 开发板的安装150
 - 4.5 小结154
- 第5章 语言间调用约定与基本数据结构156
 - 5.1 寄存器使用约定156
 - 5.1.1 寄存器别名157
 - 5.1.2 参数传递与返回值157
 - 5.2 基本数据结构159
 - 5.2.1 C语言的基本数据结构160
 - 5.2.2 list_h_t数据结构161
 - 5.2.3 spinlock_t数据结构164
 - 5.2.4 kwlst_t数据结构165
 - 5.2.5 sem_t数据结构166
 - 5.3 数据结构存在于内存中的形式168
 - 5.4 C与汇编的混用170
 - 5.5 小结174
- 第6章 内核初始化175
 - 6.1 开始175
 - 6.1.1 第一行汇编代码175
 - 6.1.2 第一个C函数178
 - 6.2 MMU和中断向量的初始化181
 - 6.2.1 初始化MMU181
 - 6.2.2 复制中断向量186
 - 6.3 串口初始化190

- 6.3.1 串口硬件190
- 6.3.2 内核的printf196
- 6.4 机器数据结构201
 - 6.4.1 设计数据结构201
 - 6.4.2 确定一些重要数据结构与内核的地址203
- 6.5 初级内存管理初始化205
 - 6.5.1 设计一些数据结构205
 - 6.5.2 初始化mmapdsc_t结构数组209
 - 6.5.3 建立起内存分配数据结构212
- 6.6 中断初始化215
 - 6.6.1 设计一些数据结构215
 - 6.6.2 初始中断源描述符220
- 6.7 初始化测试222
- 6.8 小结225
- 第7章 内存管理226
 - 7.1 内核功能层入口226
 - 7.2 内存管理组件的设计228
 - 7.3 块级内存管理229
 - 7.3.1 块级内存管理数据结构视图229
 - 7.3.2 块级内存管理接口230
 - 7.3.3 主分配函数232
 - 7.3.4 分配时查找alcfrelst_t 234
 - 7.3.5 分配时查找和操作mmapdsc_t236
 - 7.3.6 分配代码写得对吗239
 - 7.3.7 主释放函数240
 - 7.3.8 释放时查找alcfrelst_t 241
 - 7.3.9 释放时查找和操作mmapdsc_t242
 - 7.3.10 测试块级内存管理层246
 - 7.4 页级内存管理248
 - 7.4.1 页级内存管理接口及调用流程248
 - 7.4.2 相关的数据结构251
 - 7.4.3 页级内存管理初始化254
 - 7.4.4 分配主函数256
 - 7.4.5 分配时查找mplhead_t257
 - 7.4.6 分配时新建页级内存池258
 - 7.4.7 分配时操作mplhead_t262
 - 7.4.8 分配代码写得对吗263
 - 7.4.9 释放主函数265
 - 7.4.10 释放时查找mplhead_t266
 - 7.4.11 释放时操作mplhead_t268
 - 7.4.12 释放时删除页级内存池269
 - 7.4.13 测试页级内存管理层271
 - 7.5 字级内存管理273
 - 7.5.1 字级内存接口及调用流程274
 - 7.5.2 相关的数据结构275
 - 7.5.3 分配主函数276
 - 7.5.4 分配时查找mplhead_t278
 - 7.5.5 分配时新建字级内存池279
 - 7.5.6 分配时操作mplhead_t282

- 7.5.7 分配代码写得对吗283
- 7.5.8 释放主函数285
- 7.5.9 释放时查找mplhead_t285
- 7.5.10 释放时操作mplhead_t287
- 7.5.11 释放时删除字级内存池288
- 7.5.12 测试字级内存管理层289
- 7.6 小结292
- 第8章 中断管理293
 - 8.1 中断与中断控制器293
 - 8.1.1 什么是中断293
 - 8.1.2 S3C2440A中断控制器294
 - 8.2 中断管理的架构与相关数据结构298
 - 8.2.1 中断管理的架构298
 - 8.2.2 设计数据结构intfltdsc_t和intserdsc_t298
 - 8.3 中断处理301
 - 8.3.1 中断辅助例程301
 - 8.3.2 从中断向量开始305
 - 8.3.3 保存CPU上下文306
 - 8.3.4 中断主分派例程310
 - 8.3.5 确定中断源314
 - 8.3.6 调用中断处理例程317
 - 8.4 安装中断回调例程319
 - 8.5 小结322
- 第9章 驱动模型323
 - 9.1 操作系统内核如何管理设备323
 - 9.1.1 分权而治323
 - 9.1.2 设备类型325
 - 9.1.3 驱动程序327
 - 9.2 相关数据结构328
 - 9.2.1 驱动329
 - 9.2.2 派发例程类型329
 - 9.2.3 设备ID330
 - 9.2.4 设备331
 - 9.2.5 IO包332
 - 9.2.6 设备表333
 - 9.3 驱动模型的基础设施335
 - 9.3.1 驱动程序从哪里执行335
 - 9.3.2 新建与注册设备340
 - 9.3.3 注册回调函数344
 - 9.3.4 发送IO包345
 - 9.3.5 调用驱动程序函数346
 - 9.3.6 等待服务347
 - 9.3.7 完成服务350
 - 9.3.8 驱动模型辅助函数352
 - 9.4 systick驱动程序实例356
 - 9.4.1 systick硬件356
 - 9.4.2 systick驱动程序框架360
 - 9.4.3 systick驱动程序实现362
 - 9.4.4 测试systick驱动程序368

- 9.5 RTC驱动程序实例370
 - 9.5.1 RTC硬件370
 - 9.5.2 RTC驱动程序实现375
- 9.6 小结385
- 第10章 进程386
 - 10.1 应用程序的运行386
 - 10.1.1 程序运行需要什么资源387
 - 10.1.2 任何时刻资源都可用吗388
 - 10.1.3 提出多道程序模型389
 - 10.2 相关的数据结构390
 - 10.2.1 设计进程的数据结构391
 - 10.2.2 调度进程表392
 - 10.3 LMOSEM内核的第一个进程394
 - 10.3.1 进程管理组件的初始化395
 - 10.3.2 建立空转进程396
 - 10.3.3 空转进程运行399
 - 10.4 新建进程404
 - 10.4.1 分配进程描述符404
 - 10.4.2 分配内存空间406
 - 10.4.3 加入进程调度表408
 - 10.5 进程调度410
 - 10.5.1 调度算法410
 - 10.5.2 处理进程时间片411
 - 10.5.3 检查调度状态414
 - 10.5.4 选择进程415
 - 10.5.5 进程切换418
 - 10.5.6 进程等待与唤醒421
 - 10.5.7 进程测试425
 - 10.6 小结428
- 第11章 文件系统430
 - 11.1 文件系统设计430
 - 11.1.1 文件系统只是一个设备430
 - 11.1.2 数据格式与存储块432
 - 11.1.3 如何组织文件433
 - 11.1.4 关于我们文件系统的限制434
 - 11.2 相关的数据结构434
 - 11.2.1 超级块435
 - 11.2.2 位图435
 - 11.2.3 目录437
 - 11.2.4 文件管理头438
 - 11.3 文件系统格式化440
 - 11.3.1 建立超级块440
 - 11.3.2 建立位图445
 - 11.3.3 建立根目录448
 - 11.4 文件系统基础操作452
 - 11.4.1 获取与释放根目录文件453
 - 11.4.2 字符串操作455
 - 11.4.3 分解路径名457
 - 11.4.4 检查文件是否存在459

11.5	文件操作	460
11.5.1	新建文件	461
11.5.2	删除文件	463
11.5.3	打开文件	466
11.5.4	读写文件	469
11.5.5	关闭文件	472
11.5.6	驱动整合	473
11.6	文件系统测试	475
11.6.1	格式化测试	475
11.6.2	文件操作测试	479
11.7	小结	482
第12章	系统调用与应用程序库	483
12.1	系统调用机制	483
12.1.1	软中断指令	484
12.1.2	传递系统调用参数	486
12.1.3	系统调用分发器	488
12.2	时间管理系统调用	489
12.3	进程管理系统调用	492
12.3.1	进程的运行与退出	492
12.3.2	获取进程的ID	494
12.4	内存管理系统调用	496
12.5	设备与文件系统调用	498
12.5.1	设备与文件的打开	498
12.5.2	设备与文件的关闭	506
12.5.3	设备与文件的读写	508
12.5.4	设备与文件的控制	512
12.6	应用程序库	514
12.7	测试	520
12.8	小结	526
	后记	528

《深度探索嵌入式操作系统》

精彩短评

- 1、略度过一遍，整本书思路清晰，作者的表达能力赞。从零开始，实现了操作系统的基本功能，没有虚拟内存，对于希望了解操作系统运行原理的读者会十分有用。可结合uCos学习！
- 2、深入浅出，很好的入门教程
- 3、对嵌入式硬件结构、内存数据结构、存储方式描述比较清晰，能够给予嵌入式学习进行指导。推荐读。
- 4、作者用朴实的笔法，一步一步带领读者从无到有实现了一个嵌入式操作系统。本书涵盖了很多硬件、软件架构方面的知识点，是在是一部值得反复阅读的好书。

《深度探索嵌入式操作系统》

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:www.tushu111.com