

《嵌入式Linux开发实践教程》备课提纲

课程名称: 嵌入式Linux开发实践教程 | 课程代码: CS121 | 生成时间: 2025-07-20 10:07:11

《嵌入式Linux开发实践教程》备课提纲

1. 课程概述

课程性质

专业核心课程, 面向计算机科学与技术、电子信息工程等相关专业
理论与实践相结合的综合性课程

课程地位

嵌入式系统开发领域的关键技术课程
连接底层硬件与上层应用的桥梁课程

课程作用

培养学生嵌入式Linux系统开发能力
掌握从内核配置到应用开发的完整流程
为物联网、智能设备开发奠定基础

2. 教学目标

知识目标

掌握嵌入式Linux系统架构与工作原理
理解Linux内核配置与裁剪方法
掌握交叉编译工具链的使用
熟悉嵌入式Linux应用开发流程

能力目标

能够独立完成嵌入式Linux系统的移植
能够开发基于嵌入式Linux的应用程序
能够调试和优化嵌入式系统性能
能够解决开发过程中的常见问题

素质目标

培养严谨的系统开发思维
增强解决复杂工程问题的能力
培养团队协作与项目管理意识
树立持续学习和创新的精神

3. 教学重点难点

教学重点

- 嵌入式Linux系统架构
- 交叉编译环境的搭建
- 内核配置与设备驱动开发
- 嵌入式应用开发框架

教学难点

- Linux内核裁剪与优化
- 设备驱动程序的开发与调试
- 系统性能分析与优化
- 实时性要求的处理

4. 教学内容安排

模块一：嵌入式Linux基础（4学时）

- 1. 嵌入式系统概述
 - 嵌入式系统特点与应用领域
 - 嵌入式Linux的优势与特点
- 2. 嵌入式Linux开发环境
 - 开发主机与目标板
 - 交叉编译工具链
 - 开发环境搭建实践

模块二：Linux内核与系统移植（8学时）

- 1. Linux内核架构
 - 内核组成与功能模块
 - 内核配置选项解析
- 2. 内核移植实践
 - 内核裁剪与编译
 - 启动引导程序配置
 - 根文件系统构建

模块三：嵌入式应用开发（10学时）

- 1. 嵌入式开发基础
 - 系统调用与库函数
 - 多进程/多线程编程
- 2. 硬件接口编程
 - GPIO控制
 - 串口通信
 - I2C/SPI接口
- 3. 图形界面开发
 - Qt嵌入式开发

轻量级GUI框架

模块四：系统优化与调试（6学时）

1. 性能分析与优化

系统性能监测工具

内存与CPU优化

2. 调试技术

内核调试

应用调试

常见问题排查

5. 实训练习与指导

实训项目1：开发环境搭建（2学时）

任务：完成交叉编译工具链的安装与配置

指导要点：

1. 工具链版本选择

2. 环境变量配置

3. 简单测试程序编译

实训项目2：内核裁剪与移植（4学时）

任务：针对特定开发板进行内核配置与编译

指导要点：

1. 内核配置选项解析

2. 必要驱动的选择

3. 编译错误排查

实训项目3：嵌入式应用开发（6学时）

任务：开发一个基于嵌入式Linux的数据采集系统

指导要点：

1. 传感器接口编程

2. 数据存储设计

3. 用户界面实现

实训项目4：系统优化（4学时）

任务：对现有系统进行性能分析与优化

指导要点：

1. 性能监测工具使用

2. 瓶颈分析

3. 优化方案实施

6. 教学方法与策略

教学方法

项目驱动教学法

案例教学法

分组讨论法

翻转课堂

教学策略

1. 理论与实践1:1配比
2. 分层次教学（基础-提高-综合）
3. 问题导向学习
4. 线上线下混合式教学

7. 教学资源需求

教材与参考资料

主教材：《嵌入式Linux系统开发全程解析》

参考书：《Linux设备驱动程序》

在线资源：kernel.org官方文档

硬件设备

嵌入式开发板（如Raspberry Pi或BeagleBone）

传感器模块（温湿度、光照等）

调试工具（JTAG、逻辑分析仪等）

软件工具

交叉编译工具链（如arm-linux-gnueabi-hf）

开发环境（Ubuntu系统）

调试工具（gdb、strace等）

版本控制工具（git）

8. 课堂组织形式

理论课（50%）

讲授核心概念与原理

案例分析

技术讨论

实验课（50%）

演示实验

分组实践

项目开发

课外活动

技术沙龙

开源项目参与

竞赛指导

9. 评估方式与标准

形成性评估（40%）

课堂表现（10%）

实验报告（15%）

阶段项目（15%）

终结性评估（60%）

期末考试（30%）

综合项目（30%）

评估标准

1. 知识掌握程度（30%）

2. 实践能力（40%）

3. 创新能力（20%）

4. 团队协作（10%）

10. 课后拓展与作业

基础作业

1. 阅读Linux内核源码指定部分并撰写分析报告

2. 完成指定驱动程序的移植与测试

提高作业

1. 参与开源嵌入式项目贡献

2. 针对特定应用场景设计优化方案

拓展资源

1. 推荐MOOC课程：《嵌入式系统设计与实现》

2. 技术博客：Linux内核开发者博客

3. 行业动态：嵌入式Linux最新技术发展

长期项目

基于嵌入式Linux的智能家居控制系统开发

工业物联网边缘计算节点设计

本备课提纲充分结合嵌入式Linux开发的实际需求，注重理论与实践相结合，通过项目驱动的教学方式培养学生的综合能力。教学安排循序渐进，从基础环境搭建到复杂系统开发，最终目标是使学生具备独立完成嵌入式Linux系统开发的能力。