

# 《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 10:19:12

## 《嵌入式Linux开发实践教程》备课提纲

### 1. 课程概述

#### 课程性质

本课程是计算机科学与技术、电子信息工程等专业的专业核心课程  
理论与实践相结合的综合性课程

#### 地位和作用

培养嵌入式系统开发能力的必修课程  
连接底层硬件与上层应用的桥梁课程  
为物联网、智能硬件等领域开发奠定基础

### 2. 教学目标

#### 知识目标

掌握嵌入式Linux系统架构与工作原理  
理解嵌入式Python开发环境搭建与调试  
掌握TensorFlow Lite在嵌入式设备上的应用  
了解嵌入式系统性能优化方法

#### 能力目标

能够独立完成嵌入式Linux系统移植  
能够开发基于Python的嵌入式应用程序  
能够实现AI模型在嵌入式设备上的部署  
具备嵌入式系统调试与优化能力

#### 素质目标

培养严谨的系统开发思维  
提升解决复杂工程问题的能力  
培养团队协作与沟通能力  
树立工程伦理与安全意识

### 3. 教学重点难点

#### 教学重点

嵌入式Linux系统构建与移植  
交叉编译环境搭建  
嵌入式Python开发环境配置

## 教学难点

嵌入式系统硬件驱动开发  
系统性能优化与资源管理  
AI模型在资源受限设备上的优化  
系统稳定性与实时性保障

## 4. 教学内容安排

### 模块1：嵌入式Linux基础（8学时）

1.1 嵌入式系统概述  
嵌入式系统特点与应用领域  
嵌入式Linux发展历程  
1.2 嵌入式Linux系统架构  
Bootloader原理与移植  
Linux内核裁剪与配置  
根文件系统构建  
1.3 开发环境搭建  
交叉编译工具链  
嵌入式开发板连接与调试  
系统镜像烧写

### 模块2：嵌入式Python开发（12学时）

2.1 Python在嵌入式系统中的应用  
MicroPython与CPython比较  
嵌入式Python运行环境搭建  
2.2 嵌入式Python编程基础  
资源受限环境下的编程规范  
硬件接口编程（GPIO、I2C、SPI）  
传感器数据采集与处理  
2.3 性能优化技术  
内存管理与优化  
执行效率提升方法  
多线程与异步编程

### 模块3：嵌入式AI应用开发（16学时）

3.1 TensorFlow Lite基础  
模型转换与量化  
推理引擎工作原理  
性能评估指标  
3.2 模型部署实践

图像分类模型部署

目标检测模型优化

语音识别应用开发

3.3 边缘计算应用

数据预处理优化

模型更新与维护

云端协同计算

## 5. 实训练习与指导

### 实训项目1：嵌入式Linux系统移植（4学时）

任务：在开发板上完成Linux系统移植

步骤：

1. 配置编译工具链
2. 裁剪Linux内核
3. 构建根文件系统
4. 烧写系统镜像并测试

指导要点：重点讲解内核配置选项含义

### 实训项目2：传感器数据采集系统（6学时）

任务：开发Python环境下的多传感器数据采集系统

步骤：

1. 搭建Python运行环境
2. 编写GPIO控制代码
3. 实现I2C/SPI接口通信
4. 数据采集与存储

指导要点：强调异常处理与资源释放

### 实训项目3：嵌入式图像分类应用（8学时）

任务：部署TensorFlow Lite图像分类模型

步骤：

1. 模型转换与量化
2. 编写推理代码
3. 性能测试与优化
4. 实现摄像头实时分类

指导要点：讲解模型量化对精度的影响

## 6. 教学方法与策略

### 教学方法

项目驱动教学法：通过完整项目贯穿知识点

案例教学法：分析典型嵌入式应用案例

翻转课堂：课前提供资料自学，课堂重点讨论

小组协作：分组完成综合性项目

## 教学策略

循序渐进：从系统搭建到应用开发

问题导向：通过实际问题引入知识点

虚实结合：仿真环境与实际设备配合使用

分层教学：针对不同基础学生设置不同难度任务

## 7. 教学资源需求

### 硬件设备

嵌入式开发板（如Raspberry Pi、NVIDIA Jetson等）

传感器模块（温湿度、加速度计等）

摄像头模块

调试工具（JTAG、逻辑分析仪等）

### 软件工具

Linux开发环境（Ubuntu）

交叉编译工具链

Python开发环境

TensorFlow Lite框架

代码版本管理工具（Git）

### 教材与参考资料

主教材：《嵌入式Linux系统开发实战》

参考教材：《Python嵌入式编程》

在线资源：TensorFlow Lite官方文档

提供的课程资料文档

## 8. 课堂组织形式

### 理论课（40%）

教师讲授核心概念与原理

案例分析讨论

技术方案评审

### 实验课（60%）

教师演示关键操作

学生动手实践

分组项目开发

成果展示与点评

### 课外学习

线上资源学习

开发日志撰写

技术论坛交流

## 9. 评估方式与标准

### 形成性评估（40%）

课堂参与度（10%）

实验报告质量（15%）

项目中期检查（15%）

### 终结性评估（60%）

期末项目成果（30%）

功能完整性

代码质量

性能指标

项目答辩（20%）

技术讲解

问题回答

团队协作

笔试（10%）

理论知识

问题分析

## 10. 课后拓展与作业

### 基础作业

阅读Linux内核配置文档并总结

编写Python硬件控制小程序

完成TensorFlow Lite官方教程练习

### 提高作业

移植不同版本Linux内核并比较

优化Python程序内存使用

部署自定义AI模型到开发板

### 拓展学习

研究RT-Linux实时性优化

探索边缘计算框架（如EdgeX）

参加嵌入式相关开源项目

### 推荐资源

《嵌入式Linux开发详解》

TensorFlow Lite官方GitHub仓库

Raspberry Pi官方项目案例库

嵌入式系统国际会议论文

---

本备课提纲充分结合了提供的课程资料内容，特别是嵌入式Python开发和TensorFlow Lite应用部分，通过理论与实践相结合的方式，循序渐进地培养学生的嵌入式开发能力。教学安排注重项目实践，评估方式多元化，能够全面考察学生的学习效果。