

嵌入式操作系统应用

Embedded Operating System Application

一、基本信息

课程代码：【1050031】
课程学分：【3】
面向专业：【计算机科学与技术】
课程性质：【专业限选课】
开课院系：【信息技术学院计算机科学与技术系】
使用教材：
教材：自编电子版讲义
参考书目：
【温子祺等编著，ARM Cortex-M4 微控制器深度实战，北京：北京航空航天大学出版社，2017.】
【黄克亚. ARM Cortex-M4 嵌入式系统原理及应用——基于 STM32F407 微控制器的 HAL 库开发[M]. 北京：清华大学出版社，2024.】
【任哲、房红征编著，嵌入式实时操作系统 μ C/OS-II 原理及应用（第 5 版），北京：北京航空航天大学出版社，2021.】
【刘军等编著，例说 STM32（第 3 版），北京：北京航空航天大学出版社，2018.】
先修课程：【单片机与接口技术】

二、课程简介

本课程主要讲述基于 STM32 的 ARM Cortex-M4 微处理器的嵌入式操作系统设计，其中包含嵌入式系统的体系结构、嵌入式系统的开发方法、嵌入式系统的外设模块、嵌入式操作系统的原理及其在 ARM 平台上的移植应用。通过课程理论教学及配套相关实验的学习，使学生掌握嵌入式系统软硬件设计方法，为学习后续课程和从事实际工作打下坚实的理论和实践基础。

本课程的特点是内容广，实践性强。为了让学生真正掌握嵌入式系统的设计，必须强调对学生实际动手能力的培养。实验教学在本课程中具有举足轻重的地位，要求每个学生必须亲自编程，动手实现课程中的多个实验，了解嵌入式系统的应用，了解嵌入式设备上各个硬件模块的功能及控制方法，培养学生动手操作的兴趣。

三、选课建议

本课程是适用于计算机类专业的专业限选课，本课程的基础是 C 语言程序设计、单片机与接口技术等课程，要求具有一定的微处理器基础知识和 C 程序设计的能力。

四、课程与专业毕业要求的关联性

专业毕业要求	关联
LO1：工程知识：能够将数学、自然科学、工程基础和专业知	
识用于解决复杂工程问题	
LO2：问题分析：能够应用数学、自然科学和工程科学的基本原理，识别、表达、并通过文献研究分析复杂工程问题，以获得有效结论	●

LO3: 设计/开发解决方案: 能够设计针对复杂工程问题的解决方案, 设计满足特定需求的系统、单元(部件)或工艺流程, 并能够在设计环节中体现创新意识	●
LO4: 研究: 能够基于科学原理并采用科学方法对复杂工程问题进行研究, 包括设计实验、分析与解释数据、并通过信息综合得到合理有效的结论	
LO5: 使用现代工具: 能够针对复杂工程问题, 开发、选择与使用恰当的技术、资源、现代工程工具和信息技术工具, 包括对复杂工程问题的预测与模拟, 并能够理解其局限性	●
LO6: 工程与社会: 能够基于工程相关背景知识进行合理分析, 评价专业工程实践和复杂工程问题解决方案对社会、健康、安全、法律以及文化的影响, 并理解应承担的责任	
LO7: 环境和可持续发展: 能够理解和评价针对复杂工程问题的专业工程实践对环境、社会可持续发展的影响	
LO8: 职业规范: 具有人文社会科学素养、社会责任感, 能够在工程实践中理解并遵守工程职业道德和规范, 履行责任	●
LO9: 个人和团队: 能够在多学科背景下的团队中承担个体、团队成员以及负责人的角色	
LO10: 沟通: 能够就复杂工程问题与业界同行及社会公众进行有效沟通和交流, 包括撰写报告和设计文稿、陈述发言、清晰表达或回应指令。并具备一定的国际视野, 能够在跨文化背景下进行沟通和交流	
LO11: 项目管理: 理解并掌握工程管理原理与经济决策方法, 并能在多学科环境中应用	
LO12: 终身学习: 具有自主学习和终身学习的意识, 有不断学习和适应发展的能力	

五、课程目标/课程预期学习成果

序号	课程预期学习成果	课程目标 (细化的预期学习成果)	教与学方式	评价方式
1	LO211 具备对系统设计、软件开发等涉及到的复杂工程问题进行识别与判断, 并结合专业知识进行有效分解的能力	1. 了解嵌入式操作系统UCOS-II 源码的结构	讲课	作业
		2. 掌握 UCOS-II 中任务的创建过程与任务的启动	讲课、实验	作业、小测验、实验报告
2	LO311 对嵌入式系统设计遇到的问题能进行调研并明确相关约束条件, 针对系统设计完成需求分析	1. 熟悉嵌入式系统硬件设计的流程	讲课	作业
		2. 掌握嵌入式系统软件的架构设计	讲课、实验	作业、小测验、实验报告
3	LO511 能熟练	能熟练运用绘图工具画嵌入式	讲课	作业、小测验

	运用绘图工具，表达和解决计算机系统工程的设计问题	系统硬件结构图		
4	LO512 能根据具体项目的特点和需求，选择合适的技术工具进行设计开发	能熟练运用 KEIL 软件对嵌入式系统进行软件开发	讲课、实验	作业、小测验、实验报告
5	LO812 具备责任心和社会责任感，懂法守法；注重职业道德修养	遵守嵌入式操作系统移植的规范	讲课、实验	作业、小测验、实验报告

六、课程内容

第 1 讲 嵌入式系统概述

教学要求：

了解：基本的嵌入式硬件知识，嵌入式操作系统的概念及嵌入式系统的现状和发展趋势；

掌握：嵌入式系统的基本概念、组成、特点及应用领域；

教学要点：

- (1) 嵌入式系统概念；
- (2) 嵌入式处理器；
- (3) 嵌入式系统的组成；
- (4) 嵌入式技术发展现状及趋势。

理论课时数 4，实践课时数 0。

第 2 讲 嵌入式操作系统 UCOS-II

教学要求：

了解：实时操作系统 UCOS-II 的相关概念，如任务管理、多任务调度、进程上下文切换、任务间的同步和通信等；

熟悉：将实时操作系统 UCOS-II 移植的基本方法；

教学要点：

- (1) 嵌入式操作系统原理；
- (2) UCOS-II 源码的结构；
- (3) UCOS-II 任务的基本概念；
- (4) UCOS-II 中任务的创建过程与任务的启动；
- (5) UCOS-II 中任务管理的基本方法（创建、启动、挂起、解挂任务）。

理论课时数 4，实践课时数 0。

第 3 讲 ARM Cortex-M4 技术

教学要求:

了解: ARM 技术的发展流程及 Cortex-M4 产生的背景;

掌握: Cortex-M4 核心技术组成。

教学要点:

(1) Cortex-M4 内部架构;

(2) Cortex-M4 内核比较;

(3) 流水线技术;

理论课时数 4, 实践课时数 0。

第 4 讲 STM32F407 体系结构

教学要求:

了解: STM32 系列微处理器发展历程;

掌握: STM32F407 微处理器的架构;

应用: 学会搭建 STM32F407 微处理器的最小系统;

教学要点:

(1) STM32F407 的片内结构;

(2) STM32F407 的功能单元描述;

(3) STM32F407 的功能模块设计 (电源模块、复位模块、时钟模块、看门狗模块、存储器模块)。

理论课时数 4, 实践课时数 0。

第 5 讲 基于 STM32F407 的嵌入式系统硬件设计及软件编程

教学要求:

了解: 基于 STM32F407 的嵌入式系统硬件设计规则;

掌握: STM32F407 外设硬件模块结构性能及相关驱动编程;

应用: 利用 STM32F407 平台的硬软件资源构建完整的嵌入式系统;

教学要点:

(1) GPIO;

(2) 中断;

(3) 定时器;

(4) 键盘及 LED 控制;

(5) 串口;

(6) PWM 输出;

(7) A/D 转换;

(8) 触摸屏控制;

理论课时数 4, 实践课时数 4。

第 6 讲 UCOS-II 在 STM32 平台上的移植

教学要求:

了解: UCOS-II 在 STM32 移植的流程;

掌握: 在 Keil 中建立 UCOS-II 开发的模板的方法;

应用: UCOS-II 在 STM32 平台上移植;

教学要点:

(1) Keil 的安装和建立开发模板;

(2) 建立带有 UCOS-II 的 STM32 的工程模板的方法;

(3) UCOS-II 移植文件的编写;

理论课时数 4, 实践课时数 4。

第 7 讲 任务定义与任务切换的实现

教学要求:

了解: UCOS 操作系统的任务概念, 基于 STM32F4xx 芯片实现 UCOS 的任务管理;

掌握: UCOS 操作系统的任务定义与任务切换;

教学要点:

(1) STM32F4xx 中 Cortex-M4 内核的基础知识点;

(2) 创建任务 (任务栈、任务函数、任务控制块 TCB 等);

(3) 操作系统系统初始化代码实现;

(4) 操作系统启动代码;

(5) 任务切换的实现;

教学难点:

(1) STM32F4xx 中 Cortex-M4 内核的基础知识点;

(2) 系统栈和任务栈的概念;

(3) 任务切换的实现;

理论课时数 4, 实践课时数 2。

第 8 讲 任务时间片运行

教学要求:

了解: 操作系统时钟节拍的概念, 基于 STM32F4xx 芯片实现 UCOS 的系统时钟;

掌握: UCOS 操作系统时钟节拍的实现;

教学要点:

(1) STM32F4xx 中 Cortex-M4 内核中的 SysTick 外设;

(2) 初始化 SysTick;

(3) 编写 SysTick 中断服务程序;

(4) 主函数 main 的实现;

(5) 观察实验现象;

教学难点: Cortex-M4 处理器内核中的 SysTick 外设;

理论课时数 4, 实践课时数 0。

第 9 讲 阻塞延时与空闲任务

教学要求:

了解: 操作系统阻塞延时和空闲任务的概念, 基于 STM32F4xx 芯片实现空闲任务和阻塞延时;

掌握: UCOS 操作系统空闲任务和阻塞延时的实现;

教学要点:

(1) 空闲任务的原理和实现;

(2) 阻塞延时的原理和实现;

(3) 任务阻塞和任务调度;

(4) 主函数 main 的实现;

(5) 观察实验现象;

教学难点: Sleep 函数内部的实现;

理论课时数 4，实践课时数 2。

第 10 讲 临界段

教学要求：

了解：操作系统临界段的概念，基于 STM32F4xx 芯片实现临界段；

掌握：UCOS 操作系统中临界段的实现；

教学要点：

- (1) 临界段的概念；
- (2) Cortex-M 内核快速关中断指令；
- (3) 开关中断指令；
- (4) 临界段代码的应用；
- (5) 测量临界段的时间；

教学难点：处理器的开关中断指令的理解；

理论课时数 4，实践课时数 2。

第 11 讲 就绪列表

教学要求：

了解：操作系统就绪任务列表的概念，基于 STM32F4xx 芯片实现就绪任务列表；

掌握：UCOS 操作系统就绪任务列表的实现；

教学要点：

- (1) 优先级表的概念；
- (2) 就绪列表的概念；
- (3) 优先级操作函数的实现；
- (4) 就绪列表操作函数的实现；

教学难点：

- (1) 优先级表的操作函数；
- (2) 双向链表的操作函数；

理论课时数 4，实践课时数 0。

第 12 讲 支持多优先级

教学要求：

了解：操作系统优先级的支持，基于 STM32F4xx 芯片实现任务优先级；

掌握：UCOS 操作系统中优先级的实现；

教学要点：

- (1) 多优先级的概念；
- (2) 操作系统初始化函数 OSInit；
- (3) 操作系统启动函数 OSStart；
- (4) 任务创建函数 OSTaskCreate；
- (5) 系统时钟处理函数 OSTimeTick；

教学难点：操作系统时钟节拍处理函数 OSTimeTick；

理论课时数 4，实践课时数 2。

七、课内实验名称及基本要求

实验序号	实验名称	主要内容	实验时数	实验类型	备注
1	STM32F407 开发环境的建立	(1) RealView MDK 软件的安装与调试; (2) 在 RealView MDK 中建立完整的工程项目; (3) 对工程项目进行编译和运行。	4	验证型	
2	UCOS-II 操作系统移植实验	实现在 STM32 实验平台上移植 UCOS-II 实时操作系统	4	设计型	
3	UCOS 任务定义与切换的实现	(1) 掌握 UCOS 任务定义的实现; (2) 在 UCOSII 里面创建 3 个任务测试。	4	设计型	
4	UCOS 任务优先级的实现	(1) 掌握 UCOS 任务多优先级的实现; (2) 编写 main 函数测试。	4	设计型	

八、评价方式与成绩

总评构成 (X)	评价方式	占比
X1	课程大作业	40%
X2	实验报告	20%
X3	平时作业	20%
X4	上机测试	20%

撰写： 孙锦中

系主任审核：戴智明

审核时间：2025.9.1