

【嵌入式软件开发】

【Embedded Software Development】

一、基本信息

课程代码：【2050291】

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：

主教材【刘火良等编著，uC/OS-III 内核实现与应用开发实战指南（第一版），北京：机械工业出版社，2020.】

辅教材【何小庆等译，嵌入式实时操作系统 UCOS-III 应用开发：基于 STM32 微控制器(第一版)，北京：北京航空航天大学出版社，2020.】

参考书【程文娟，嵌入式实时操作系统 UCOS-II 教程(第二版)，西安：西安电子科技大学出版社，2019.】

课程网站网址：

<https://coursehome.zhihuishu.com/courseHome/1000012684/125429/18#teachTeam>

先修课程：【数字逻辑电路】、【C 语言程序设计】、【单片机原理与接口技术】

二、课程简介

本课程主要讲述基于 STM32F4xx 芯片与实时操作系统 UCOS-III 的嵌入式软件开发，其中包含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 Cortex 系列处理器平台上的移植应用。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，熟悉嵌入式软件开发的流程，建立嵌入式多任务程序设计的思想。

学生通过学习该课程，了解实时操作系统的相关概念，如任务管理、多任务调度、进程上下文切换、时间片轮转、任务间的同步与通信等，掌握将实时操作系统 UCOS-III 移植到 ARM 平台的方法，熟悉 UCOS-III 操作系统的部分实现及其应用程序开发，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。

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、嵌入式软硬件设计等课程，要求具有一定的嵌入式基础知识和 C 程序设计的能力。

四、课程与培养学生能力的关联性

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, 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	

备注: LO = Learning Outcomes (学习成果), 具体请参考本科生培养方案。

五、课程目标/课程预期学习成果

本课程以提高学生实践应用水平为最终目标。在学习微控制器系统工作原理的基础上, 加强实践技能和动手能力的训练, 从而使学生具备开发物联网工程、智能化电子产品、以及计算机智能控制的实践能力。

通过在 STM32F4xx 系列实验板上, 对 ARM 处理器与 UCOS 操作系统的各个模块进行实验、调试、及功能现象的观察, 增加学生对该课程直观的认识, 激发学生对专业课学习的兴趣。

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	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.了解嵌入式实时操作系统 UCOS-III 源代码结构	课堂教学	作业 课堂提问

		2.掌握 UCOS-III 操作系统中任务的创建过程与任务的上下文切换实现	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基本流程	课堂教学	作业、小测验
		2.掌握嵌入式系统软件设计的基本流程	讲课、实验	作业、小测验、实验报告
3	LO511 能熟练运用绘图工具,表达和解决计算机系统工程的设计问题	1.能熟练运用绘图工具画嵌入式系统硬件结构图 2.能熟练运用软件设计工具画嵌入式系统软件架构图	课堂教学	作业、小测验
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	1.能熟练运用 MDK KEIL 软件对嵌入式系统进行软件开发 2.能使用 FLYMCU 烧录 ROM 映像文件 3.能使用串口调试助手查看嵌入式系统的日志	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,能写出可读性高的代码,注释规范,测试规范	讲课、实验	作业、小测验、

六、课程内容

本节为课程的主要内容,共分为十五讲,每讲的教学要求包括**了解、掌握、教学要点、教学难点、理论与实践课时**等五个部分。其中:了解为本讲知识点的概述;掌握为本讲知识点的一条主线;教学要点为本讲知识点的重心;教学难点顾名思义为本讲知识点的较难理解部分;最后是理论课时与实践课时,实践课时依据各讲知识点的实践重要性来安排,老师可酌情调整。

第 1 讲 实时操作系统概述

教学要求:

了解 实时操作系统的相关概念,如实时特性、任务管理、多优先级调度、进程上下文切换、时间片轮转、任务间的同步与通信、存储器消耗需求、以及嵌入式系统设计方法等;

掌握: 实时操作系统的基本架构;

教学要点:

- (1) 嵌入式操作系统架构;
- (2) 嵌入式操作系统设计方法;
- (3) 实时操作系统特点;

- (4) 存储器消耗需求分析;
- (5) Unix 和 Linux 操作系统;

教学难点:

- (1) 嵌入式操作系统设计方法;
- (2) 存储器消耗需求分析;

理论课时数 2, 实践课时数 0。

第 2 讲 UCOS 实时操作系统概述

教学要求:

了解: UCOS 操作系统的发展历史, 应用场景, 及其软件体系结构。

掌握: UCOS 操作系统的移植和体系结构;

教学要点:

- (1) UCOS 操作系统的历史;
- (2) UCOS 操作系统的移植要点;
- (3) UCOS-II 和 UCOS-III 的体系结构;

教学难点: UCOS 操作系统的体系结构;

理论课时数 2, 实践课时数 0。

第 3 讲 任务管理

教学要求:

了解: 操作系统的任务概念;

掌握: 操作系统的任务调度和同步;

教学要点:

- (1) 任务的概念 (进程和线程区别);
- (2) 可重入性概念;
- (3) 多线程安全概念;
- (4) 处理器的编程模型;

教学难点:

- (1) Windows/Linux/UCOS 多线程 API 接口;
- (2) CPU 的编程模型;

理论课时数 2, 实践课时数 0。

第 4 讲 新建工程与调试技术

教学要求:

了解: 基于 Keil-uVison5 创建 UCOS 工程;

掌握: 在 Keil 中建立 UCOS 工程模板的方法;

教学要点:

- (1) 创建 UCOS 操作系统的目录结构
- (2) 创建 Keil-uVison5 工程;
- (3) 在 Keil 工程中添加分组;
- (4) 在 Keil 工程中添加文件;
- (5) 调试和包含路径相关配置;

教学难点: 调试相关配置及模拟器的观察;

理论课时数 2, 实践课时数 2。

第5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 6，实践课时数 2。

第6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 2，实践课时数 0。

第7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 2，实践课时数 2。

第8讲 时间戳

教学要求：

了解：操作系统中时间戳的概念，基于 STM32F4xx 芯片实现时间戳；

掌握：UCOS 操作系统中时间戳的实现；

教学要点：

- (1) STM32F4xx 芯片中的 DWT 外设；
- (2) 时间戳的概念；
- (3) 时间戳的实现；

教学难点：时间戳外设 DWT 的理解；

理论课时数 2，实践课时数 0。

第 9 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 2，实践课时数 2。

第 10 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 11 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 2，实践课时数 2。

第 12 讲 实现时基列表

教学要求：

了解：操作系统时基列表的概念，基于 STM32F4xx 芯片实现时基列表；

掌握：UCOS 操作系统中时基列表的实现；

教学要点：

- (1) 时基列表的概念；
- (2) 时基列表的实现；
- (3) 延时函数 OSTimeDly；
- (4) 系统时钟处理函数 OSTimeTick；

教学难点：时基列表与就序列表的操作；

理论课时数 2，实践课时数 2。

第 13 讲 实现时间片

教学要求：

了解：操作系统时间片的概念，基于 STM32F4xx 芯片实现时间片；

掌握：UCOS 操作系统中时间片的实现；

教学要点：

- (1) 时间片的概念；
- (2) 系统时钟处理函数 OSTimeTick；
- (3) 任务创建函数 OSTaskCreate；
- (4) 主函数 main 的实现；

教学难点：时间片的实现，任务创建支持时间片概念；

理论课时数 2，实践课时数 2。

第 14 讲 任务的挂起和恢复

教学要求：

了解：操作系统中任务挂起和恢复的概念，基于 STM32F4xx 芯片实现任务的挂起和恢复；

掌握：UCOS 操作系统中任务挂起和恢复的实现；

教学要点：

- (1) 任务挂起和恢复的概念；
- (2) 实现任务的挂起和恢复；
- (3) 主函数 main 的实现；
- (4) 观察实验现象；

教学难点：实现任务的挂起和恢复；

理论课时数 2，实践课时数 2。

第 15 讲 任务的删除

教学要求：

了解：操作系统中任务删除的概念，基于 STM32F4xx 芯片实现任务的删除；

掌握：UCOS 操作系统中任务删除的实现；

教学要点：

- (1) 实现任务删除；
- (2) 就绪列表、时基列表、等待列表的概念；
- (3) 任务处于删除态的理解；

教学难点：实现任务的删除；

理论课时数 2，实践课时数 0。

注：

1. 由于课时紧，内容多，教学进程和内容的深广度将视学生接受程度作适当的调整；

2. 教学方法与手段设计（具体请参考教案）：

- ✓ 教学方法，应该注重讲授法与讨论法；
- ✓ 教学手段，应该结合智慧树翻转课堂；
- ✓ 要点和难点，通过讨论讲解，对照代码讲解，师生互动，作业等来突出；
- ✓ 启发式教学，以提问的方式，让学生集中精力听课，并跟随老师的思路，思考问题；
- ✓ 实际教学中，应该抓住基本概念、基本原理和基本方法，教法上多举例，重应用；
- ✓ 各单元结束，应该检查学生的学习成果，便于复习，把握节奏，有的放矢。

七、课内实验名称及基本要求

实验序号	实验名称	主要内容	实验课时	实验类型	备注
1	UCOS 编译环境的建立和移植	(1) Keil 的安装和建立开发模板； (2) 建立无 UCOS-II 的 STM32 的工程模板； (3) 建立带有 UCOS-II 的工程模板；	4	验证型	
2	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	
3	UCOS 阻塞（睡眠）延时的实现	(1) 掌握 UCOS 阻塞延时的实现； (2) 编写 main 函数测试阻塞延时。	4	设计型	
4	UCOS 任务优先级和时间片的实现	(1) 掌握 UCOS 任务多优先级的实现； (2) 掌握 UCOS 任务时间片的实现； (3) 编写 main 函数测试。	4	设计型	

注 以上实验需要写实验报告，其他的实践环节均是配合课堂教学，在课堂上请根据**教学进度表**酌情安排。

八、评价方式与成绩

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

注：具体请参考课程考核方案。

撰写人：孙锦中

系主任审核签名：戴智明

审核时间：（2022年9月制订）