**《嵌入式系统设计实践》课程教学大纲**

|  |  |  |  |
| --- | --- | --- | --- |
| **课程代码** | **SOFT0031131136** | **课程性质** | **专业必修** |
| **课程名称：** | **嵌入式系统设计实践** | | |
| **英文名称** | **Designing of Embedded Systems** | | |
| **学时/学分** | **18/1** | **其中实验/实践学时** | **18** |
| **开课单位** | **软件工程学院** | **适用专业：** | **软件工程专业嵌入式方向** |
| **先修课程** | **数字逻辑** | | |
| **大纲撰写人** | **郭建** | **大纲审核人** | **陈闻杰** |
| **课程网址** | **无** | **授课语言** | **中文（双语课程）** |

**注：课程性质选择下列类别之一：学科基础、大类平台、专业必修、专业选修、教师教育**

1. **课程说明**

本课课程与《嵌入式系统设计》课程一起，是软件工程专业嵌入式方向的专业基础课，是训练学生嵌入式开发能力的重要课程。

1. **课程目标**

1. 能够运用嵌入式系统开发工具，针对ARM Cortex M4开发板完成基础的软硬件实验，在实践中思考优化解决方案；能够善于运用基础实践知识在各种复杂工程问题上，通过实践培养团队构建、运行、协调和引领的能力。（毕业要求1）

1. **课程目标与毕业要求的对应关系**

|  |  |  |
| --- | --- | --- |
| 毕业要求 | 指标点 | 课程目标 |
| 毕业要求1：工程知识：系统掌握数学、自然科学、工程基础和专业知识，能够综合应用上述知识，解决软件工程领域中的复杂工程问题。 | 观测点1-2：能够运用数学、自然科学、工程科学的理论和方法，恰当地表述软件工程领域中的复杂工程问题；建立合适的数学模型并求解； | 课程目标1 |

1. **实验内容、学时安排和基本要求**

通过实验这一环节，使学生加深对所学的编程技术的了解，并获得上机实验技能的训练。共安排8个实验，共计16学时。上机内容和基本要求是：

**实验项目1：**嵌入式系统开发环境部署

**实验类型（验证/设计/创新）：**验证、设计

**学时：4**

**对应实验教材：**实验1,2,3

**实验内容：**

1. 软件开发系统Keil5安装、配置；
2. 导入工程例子，学习使用软件Keil5的编译、断点调试、单步调试等基本操作；
3. 新建工程并对其进行编译、调试等基本操作。
4. 在Keil5中完成汇编指令和C语言实验

**实验要求：**

1. 了解软件开发系统Keil5的安装、配置流程；
2. 熟悉实验环境Keil5的基本操作，编译、断点调试、单步调试等；
3. 掌握使用软件Keil5新建工程的基本流程。
4. 掌握汇编治理和C程序的开发流程

**重点难点：**

【本实验重点】软件开发系统Keil5的配置；新建工程并用Keil5对其进行编译、调试等。

【本实验难点】新建工程并用Keil5对其进行编译、调试等。

**实验项目2：GPIO输出（一）**

**实验类型（验证/设计/创新）：**验证、创新

**学时：**2

**对应实验教材：**实验4

**实验内容：**

1. 了解GPIO端与寄存器端之间的映射关系；
2. 学会使用寄存器点亮LED灯；
3. 学会改变寄存器的相应位来观察LED灯的亮灭情况。

**实验要求：**

1. 了解GPIO端与寄存器端之间的映射关系；
2. 学会使用寄存器点亮LED灯；

**重点难点：**

【本实验重点】GPIO端与寄存器端之间的映射关系；使用寄存器点亮LED**。**

【本实验难点】对每个控制的寄存器位手工写入特定参数进行配置。

**实验项目3：GPIO输出（二）**

**实验类型（验证/设计/创新）：**验证、设计

**学时：3**

**对应实验教材：**实验5

**实验内容：**

1. 学会使用固件库点亮LED灯。

**实验要求：**

1.了解固件库

2.运用固件库实现点亮LED灯。

**重点难点：**

【本实验重点】软件库原理；STM32固件库。

【本实验难点】使用固件库点亮LED灯。

**实验项目4：GPIO输入（一）**

**实验类型（验证/设计/创新）：**验证

**学时：**2

**对应实验教材：**实验6

**实验内容：**

1. 学习如何操作按键，并用轮询方式检测按键；
2. 完成示例实验。

**实验要求：**

1．掌握如何对按键进行检测，且控制不同按键点亮不同的LED灯。

**重点难点：**

【本实验重点】按键检测要初始化GPIO目标引脚为输入模式；实验中消抖处理。

【本实验难点】采用轮询方式检测按键；软件消抖。

**实验项目5：GPIO输入（二）**

**实验类型（验证/设计/创新）：**验证

**学时：**3

**对应实验教材：**实验7

**实验内容：**

1. 了解中断的基础知识；
2. 掌握中断编程过程，并能用中断检测按键。

**实验要求：**

1. 掌握如何对按键进行检测，且控制不同按键点亮不同的LED灯；
2. 了解中断的基本知识；
3. 掌握中断编程的具体流程，并能使用中断检测按键。

**重点难点：**

【本实验重点】中断的基本知识；中断编程的具体流程，使用中断检测按键。

【本实验难点】中断编程的具体流程。

**实验项目6：定时器实验**

**实验类型（验证/设计/创新）：**验证

**学时：**3

**对应实验教材：**实验8

**实验内容：**

1. 用基本定时器TIM6计时，每1s产生一个定时器中断；

2. 中断服务子例程完成对LED1的亮-灭-亮翻转。

**实验要求：**

1. 了解STM32系列处理器定时器工作原理；

2. 掌握STM32系列处理器定时器的使用方法和程序编写；

**重点难点：**

【本实验重点】计时器的基本原理，基本计时器的使用；

【本实验难点】计时器中断编程。

**实验项目7：呼吸灯与PWM控制实验**

**实验类型（验证/设计/创新）：**验证

**学时：**3

**对应实验教材：**实验9

**实验内容：**

1. 用定时器TIM9的通道1输出PWM信号控制LED1指示灯的亮度，由暗变亮，再由亮变暗，类似人的呼吸；

**实验要求：**

 1. 了解PWM的基本概念和应用范围；

 2. 掌握STM32系列处理器定时器的使用方法和程序编写；

**重点难点：**

【本实验重点】STM32F中GPIO复用的基本原理；

【本实验难点】配置定时器基础设置和定时器的PWM信号输出。

**实验项目8：USART**

**实验类型（验证/设计/创新）：**验证

**学时：**4

**对应实验教材：**实验10

**实验内容：**

1. 了解串口通讯和STM32的USART外设
2. 学习使用STM32的USART和电脑进行通信。

**实验要求：**

1. 简单了解串口通信；
2. 了解STM32的USART外设；
3. 熟悉使用USART进行设备间通信的方法。

**重点难点：**

【本实验重点】串口通信；USART功能；USART1接发通信。

【本实验难点】USART1接发通信。

1. **教学方法**

课堂上对新知识进行系统性、提纲性讲解，并分析难点，讲解延伸知识。在对硬件平台讲解过程中，强调通过一款硬件平台介绍掌握不同平台的开发方法，采用“范例-演示-视频-仿真-实验”的“立体化”教学模式，加深学生对硬件知识的理解。

1. **考核方法**

本课程满分100分，主要通过课堂实验检查、实验报告、综合实验成绩，具体比重及考核细则如下。

|  |  |  |
| --- | --- | --- |
| **课程成绩构成** | **比例** | **考核/评价细则** |
| 实验检查 | 50% | 每次实验的检查。 |
| 实验课堂表现 | 10% | 课堂参与度。 |
| 实验报告 | 30% | 提交的实验报告。 |
| 综合实验 | 20% | 学期末的综合实验。 |

按照在实验过程中的能力成绩的比值如下：

|  |  |  |  |
| --- | --- | --- | --- |
| 编号 | 实验评价内容 | 占比 | 要求 |
| 1 | 问题分析能力 | 20% | 能够表达问题的分析过程 |
| 2 | 方案设计能力 | 20% | 能够表达问题的解决方案 |
| 3 | 编程实现能力 | 40% | 能够编程实现，并调试运行 |
| 4 | 程序书写规范 | 10% | 标识符定义规范，程序书写风格规范 |
| 5 | 报告清晰，按时提交 | 10% | 报告清晰，提交准时 |

1. **推荐教材和参考教材**
2. 参考教材：嵌入式系统设计实验教程，北京：清华大学出版社, 2022
3. 推荐资料
   1. 零死角玩转STM32—F429挑战者
   2. STM32Fxx英文参考手册
4. **评分标准**

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| 课程目标 | 项目要求 | **90-100** | **80-89** | **70-79** | **60-69** | **0-59** |
| 课程目标1 | 问题分析能力（20分） | 能够根据问题举例，并给出分析过程，并总结规律。（18-20分） | 能够根据问题举例，并给出分析过程。（16-17分） | 能够根据问题举例，并给出分析过程，但分析不完整。（14-15分） | 能够根据问题举例，举例不准确，分析不完整。（12-13分） | 不能够正确分析问题，或者举例出入较大。（12分以下） |
| 方案设计能力（20分） | 能够表达问题的解决方案。（18-20分） | 有少许不完善。（16-17分） | 有缺失。（14-15分） | 有个别错误。（12-13分） | 错误较多。（12分以下） |
| 编程实现能力（40分） | 程序正确并清晰易读。（36-40分） | 程序正确。（32-35分） | 能得到结果，有少许缺陷。（28-31分） | 能得到结果，缺陷较多。（24-27分） | 不能得到正确结果。（24分以下） |
| 程序书写规范（10分） | 符合程序书写规范，标识符定义规范。（9-10分） | 符合程序书写较规范。  （8分） | 符合程序书写不够规范。  （7分） | 能够阅读，书写随意。（6分） | 阅读困难，书写随意。  （6分以下） |
| 报告清晰，按时提交（10分） | 报告清楚，按时提交。（9-10分） | 报告较清楚，按时提交。（8分） | 推迟一天，但报告清楚。（7分） | 推迟2天，报告不清楚。（6分） | 推迟三天以上。（6分以下） |