

《嵌入式系统开发》课程标准

实验学时数：54 学分：3.0

适用的专业范围：物联网应用技术、电子信息工程技术等

授课教师：郑博伟

一、课程定位

《嵌入式系统开发》是面向电子信息大类（包括电子信息类、计算机类、通信类）和自动化类专业开设的专业核心课程，该课程面向嵌入式系统设计师工作岗位。本课程的先修课程是《电子技术基础》和《C 语言程序设计》。本课程的主要任务是：使学生掌握嵌入式系统相关的通用知识、嵌入式硬件构件设计和嵌入式软件构件设计，并在此基础上根据系统功能需求进行嵌入式应用层程序设计，为《物联网技术》、《毕业设计论文》、《岗位实习》等后续课程奠定坚实的嵌入式技术基础，为社会培养嵌入式智能产品设计、分析、调试与创新能力强的高素质技术技能人才。

二、实训课内容和学时分配

课程内容和学时分配表

| 序号 | 章节 | 课时 |
|----|--|----|
| 一 | 课程介绍及 C 语言复习 | 2 |
| 二 | 初识 STM32Cube 生态系统 | 4 |
| | 使用 GPIO 点亮一个 LED | |
| 三 | 使用 GPIO 进行按键检测 | 2 |
| 四 | 使用 EXTI 中断检测按键 | 4 |
| 五 | 使用 USART 发送和接收数据(查询模式) | 2 |
| 六 | 使用 USART 发送和接收数据(中断模式) 使用 USART 发送和接收数据(DMA 模式) | 4 |
| 七 | 使用通用定时器闪烁 LED | 2 |
| 八 | 使用通用定时器产生 PWM 驱动蜂鸣器 | 4 |
| 九 | 使用 ADC 读取气体传感器数据 (MQ-2) 重定向 printf 函数到串口输出的多种方法 | 2 |
| 十 | 使用硬件 I2C 读取环境光强度传感器数据(BH1750) | 4 |
| 十一 | 使用硬件 I2C 读取温湿度传感器数据 (SHT30) | 2 |
| 十二 | 读取超声波传感器 (HC-SR04) 数据到串口 | 4 |
| 十三 | 嵌入式系统开发技能练兵及检验 01 (混合知识点) | 2 |
| 十四 | 嵌入式系统开发技能练兵及检验 02 (混合知识点) | 4 |
| 十五 | 嵌入式系统开发技能练兵及检验 03 (混合知识点) | 2 |

| | | |
|----|--------------------------|----|
| 十六 | 嵌入式系统开发技能练兵及检验 04（混合知识点） | 4 |
| 十七 | 嵌入式系统开发技能练兵及检验 05（混合知识点） | 2 |
| 十八 | 练兵题目及验收结果作回顾 | 4 |
| | 总计 | 54 |

三、教学建议

原则上教师应该遵照教学大纲的要求，以及大纲所确定的基本内容完成教学任务，但对教学内容的顺序安排，教学时数的分配等方面，任课教师可根据实际情况对集中授课和上机实验学时作相应调整。教师在讲授时应注重培养学生的自学能力，理论与实践并重，以适应计算机知识不断更新的需求。

四、教学目标

1. 知识目标

- (1) 熟悉嵌入式系统的概念、组成以及嵌入式技术的学习方法
- (2) 熟悉 MCU 的资源
- (3) 掌握嵌入式硬件最小系统设计
- (4) 掌握 GPIO 的通用知识
- (5) 熟悉 MCU 的 GPIO 底层驱动构件的设计方法
- (6) 掌握 MCU 的 GPIO 底层驱动构件头文件的使用方法
- (7) 掌握小灯的硬件构件和软件构件的设计及使用方法
- (8) 掌握嵌入式软件最小系统（闪灯）、流水灯的设计与实现方法
- (9) 掌握开关硬件构件和开关软件构件的设计及使用方法
- (10) 掌握开关检测与控制功能的应用层程序设计方法
- (11) 理解中断的概念及中断管理过程
- (12) 熟悉 MCU 的定时器模块及其底层驱动构件设计方法
- (13) 掌握 MCU 的定时器底层驱动构件头文件的使用方法
- (14) 掌握定时中断的应用层程序设计方法
- (15) 掌握数码管的通用知识、数码管的硬件构件设计方法
- (16) 掌握数码管软件构件设计及使用方法
- (17) 掌握数码管显示的应用层程序设计方法
- (18) 掌握键盘的通用知识、键盘的硬件构件设计方法
- (19) 掌握键盘软件构件设计及使用方法
- (20) 掌握键盘检测与控制功能的应用层程序设计方法

- (21) 掌握 UART 通信的通用知识
- (22) 熟悉 MCU 的 UART 模块及其底层驱动构件设计方法
- (23) 掌握 MCU 的 UART 底层驱动构件头文件的使用方法
- (24) 掌握 UART 通信的应用层程序设计方法
- (25) 掌握上位机和下位机的串口通信与调试方法
- (26) 掌握通过 UART 接口实现利用格式化输出函数 printf 向 PC 输出数据的方法
- (27) 掌握 PWM 的通用知识（基本概念、技术指标及应用场合）
- (28) 熟悉 MCU 的 PWM 模块及其底层驱动构件设计方法
- (29) 掌握 MCU 的 PWM 底层驱动构件头文件的使用方法
- (30) 掌握 PWM 控制功能的应用层程序设计方法
- (31) 掌握输入捕捉的通用知识（输入捕捉的过程和原理）
- (32) 熟悉 MCU 的输入捕捉模块及其底层驱动构件设计方法
- (33) 掌握 MCU 的输入捕捉底层驱动构件头文件的使用方法
- (34) 掌握输入捕捉功能的应用层程序设计方法
- (35) 掌握 ADC 的通用知识
- (36) 熟悉 MCU 的 ADC 模块及其底层驱动构件设计方法
- (37) 掌握 MCU 的 ADC 底层驱动构件头文件的使用方法
- (38) 掌握 ADC 功能的应用层程序设计方法
- (39) 掌握 CAN 通信的通用知识
- (40) 熟悉 MCU 的 CAN 模块及其底层驱动构件设计方法
- (41) 掌握 MCU 的 CAN 底层驱动构件头文件的使用方法
- (42) 掌握 CAN 通信功能的应用层程序设计方法
- (43) 掌握多机之间的 CAN 通信与调试方法
- (44) 了解 MCU 的系统时钟组成及分配

2. 能力目标

- (1) 能利用 Keil MDK 集成开发环境下的工程模板，进行工程文件的物理存储和逻辑组织与管理。

(2) 能利用 ST-Link 进行目标程序的下载

(2) 能利用 GPIO、TIM、SysTcik. UART、PWM、INCAP, ADC、CAN 底层驱动构件头文件及 LIGHT、SW、KB、LED 等应用外设构件进行综合应用系统设计。

(3) 能借助 MCU 参考手册分析 GPIO、FTM、SysTcik. UART、PWM、INCAP、ADC、CAN 等底层驱动构件源文件代码

3. 素质目标

(1) 基本职业素养：遵守工作时间，在教学活动中渗透企业的“8S”制度（教学实施：使用实践设备时注重用电安全，实践设备使用完毕后要断电并放于指定位置），培养良好的工作习惯与职业意识。

(2) 团结协作素养：根据工作任务进行合理的分工与协作，按照要求完成任务，培养责任意识、团队意识与协作精神。

(3) 自主学习素养：对所进行的工作任务收集、整理，并进行工作总结（教学实施：每个项目的实施，需要自行查找资料并进整理，项目完成后，需要进行总结并填写学习任务手册），培养自主学习意识与一丝不苟工作作风。

(4) 思政和劳动素养：具有一定的辩证唯物主义运用能力、产品成本意识、劳动意识、创新意识和创新能力。

五、教学设计

牢固树立以学习者为中心的教学理念，按照“以学生为中心、学习成果为导向、促进自主学习”思路进行课程教学设计，充分体现“做中学、学中做”、“教、学、做一体化”等职业教育教学特色，使学校教学过程与企业的生产过程相对接。以实际、实用、必需、够用为原则，采用项目化教学方式，以“项目、任务、活动”等理实一体教学模式呈现教学内容。

实验一

| | |
|------------------------------|---|
| 实验名称：初识 STM32Cube 生态系统（学时：2） | |
| 知识目标 | (1) 掌握 ARM 的基本概念 (2) 熟悉常见的 ARM 处理器生产商及产品 (3) 掌握 STM32 及命名规则？ (4) 了解 STM32 优势 |
| 能力目标 | (1) 能读懂各类 STM32 种类命令的含义 |

| | |
|------|---------------------------|
| | (2) 能输出 STM32 与 51 单片机的区别 |
| 教学重点 | 无 (了解为主) |
| 教学难点 | 无 |

实验二

| | |
|------------------------------|--|
| 实验名称：使用 GPIO 点亮一个 LED (学时：4) | |
| 知识目标 | <ul style="list-style-type: none"> (1) 熟悉嵌入式系统的概念、组成及嵌入式技术的学习方法。 (2) 熟悉 MCU 的资源。 (3) 掌握嵌入式硬件最小系统设计。 (4) 掌握 GPIO 的通用知识。 (5) 熟悉 MCU 的 GPIO 底层驱动构件的设计方法。 (6) 掌握 MCU 的 GPIO 底层驱动构件头文件的使用方法。 (7) 掌握小灯的硬件构件和软件构件的设计及使用方法。 (8) 掌握闪灯、流水灯的设计与实现方法 |
| 能力目标 | <ul style="list-style-type: none"> (1) 能利用 Keil MDK 集成开发环境下的工程模板进行工程文件的组织和管 理。 (2) 能利用 st-link 进行目标程序的下载和运行。 (3) 能进行 MCU 硬件最小系统和小灯的硬件构件设计。 (4) 能利用给定的 GPIO 底层驱动构件头文件进行小灯软件构件设计。 (5) 能利用小灯构件进行闪灯和流水灯的应用层程序设计。 (6) 能借助于 MCU 参考手册分析 GPIO 底层驱动构件源文件的程序代码。 |
| 教学重点 | <ul style="list-style-type: none"> (1) 嵌入式系统的概念、组成以及嵌入式技术的学习方法。 (2) 嵌入式硬件最小系统设计。 (3) GPIO 的通用知识 (4) MCU 的 GPIO 底层驱动构件头文件的使用方法 (5) 小灯的硬件构件和软件构件的设计及使用方法。 (6) 掌握小灯的硬件构件和软件构件的设计及使用方法。 (7) 闪灯、流水灯的设计与实现方法。 (8) Keil MDK 和 st-link 软件的使用方法。 |

| | |
|------|---|
| 教学难点 | <p>(1) MCU 的 GPIO 底层驱动构件的设计方法。</p> <p>(2) 嵌入式软件最小系统的程序流程分析。</p> |
|------|---|

实验三

| | |
|---------------------------|--|
| 实验名称：使用 GPIO 进行按键检测（学时：2） | |
| 知识目标 | <p>(1) 理解 GPIO 的基本概念：掌握通用输入输出（GPIO）引脚的功能与用途，理解其在嵌入式系统中的作用。</p> <p>(2) 了解按键检测原理：掌握通过 GPIO 引脚检测按键状态的基本原理，包括上拉电阻、下拉电阻的作用。</p> |
| 能力目标 | <p>(1) 编程能力：能够编写嵌入式 C 程序，利用 GPIO 进行按键检测，并实现简单的按键功能。</p> <p>(2) 调试能力：具备使用调试工具（如 MDK-ARM、STM32CubeIDE 等）进行程序调试的能力。</p> <p>(3) 问题解决能力：能够分析并解决按键检测过程中遇到的问题，如误触发、按键响应延迟等。</p> |
| 教学重点 | <p>(1) GPIO 引脚配置：讲解如何在 STM32 等嵌入式平台上配置 GPIO 引脚为输出或输入模式。</p> <p>(2) 按键检测程序编写：演示如何编写程序来检测按键的按下和释放状态，并处理这些状态变化。</p> |
| 教学难点 | <p>(1) 按键抖动问题：掌握如何通过软件或硬件方式消除按键抖动对检测结果的影响。</p> <p>(2) 去抖动算法实现：理解并实践不同的去抖动算法，以确保按键检测的准确性和稳定性。</p> |

实验四

| | |
|---------------------------|---|
| 实验名称：使用 EXTI 中断检测按键（学时：4） | |
| 知识目标 | <p>(1) 理解 EXTI 中断的基本概念：掌握 EXTI（外部中断/事件控制器）在 STM32 中的作用及其配置方法。</p> |

| | |
|------|---|
| | <p>(2) 掌握 NVIC 中断优先级配置：了解 NVIC（嵌套向量中断控制器）的优先级配置机制，并能在 STM32 上配置中断优先级。</p> <p>(3) 理解 GPIO 引脚的中断触发方式：了解如何通过配置 GPIO 引脚来触发 EXTI 中断，包括上升沿、下降沿或双边沿触发。</p> <p>(4) 掌握 HAL 库函数的使用：学会使用 STM32 HAL 库中的函数来配置 GPIO、EXTI 和 NVIC，以实现按键中断检测。</p> |
| 能力目标 | <p>(1) 硬件连接与配置能力：能够正确连接按键到 STM32 开发板上的 GPIO 引脚，并配置相应的硬件资源。</p> <p>(2) 软件编程能力：能够编写基于 STM32 HAL 库的代码，配置 EXTI 中断，并编写中断服务例程（ISR）或回调函数来处理按键事件。</p> <p>(3) 调试与排错能力：能够使用 Keil MDK-ARM 开发环境进行代码调试，解决中断配置和按键检测过程中出现的问题。</p> <p>(4) 文档编写能力：能够编写实验报告，记录实验过程、实验结果和遇到的问题及解决方案。</p> |
| 教学重点 | <p>(1) EXTI 中断的配置流程：包括 GPIO 引脚的选择、EXTI 触发方式的设置、NVIC 中断优先级的配置等。</p> <p>(2) 中断服务例程或回调函数的编写：如何在中断发生时执行特定的操作，如读取按键状态、翻转 LED 状态等。</p> <p>(3) HAL 库函数的使用：掌握 HAL 库中与 GPIO、EXTI 和 NVIC 相关的函数，理解其参数和返回值的意义。</p> |
| 教学难点 | <p>(1) 中断优先级和抢占优先级的理解：理解 NVIC 中断优先级配置中的抢占优先级和子优先级的区别和作用，以及如何合理设置中断优先级以避免优先级反转等问题。</p> <p>(2) 中断服务例程或回调函数的正确编写：确保在中断服务例程或回调函数中执行的代码不会过长或过于复杂，以避免影响中断的响应时间和系统的稳定性。</p> <p>(3) 调试与排错：在中断配置和按键检测过程中，可能会遇到中断不触发、按键状态读取错误等问题，需要学生能够快速定位问题并给出解决方案。</p> |

实验五

| | |
|-----------------------------------|---|
| 实验名称：使用 USART 发送和接收数据(查询模式)（学时：2） | |
| 知识目标 | <p>(1) 理解 USART 的基本概念：使学生掌握 USART（通用同步/异步收发传输器）的工作原理，包括其作为串行通信接口的角色，以及波特率、数据位、停止位、校验位等参数的设置。</p> <p>(2) 掌握 USART 的硬件结构：了解微控制器中 USART 模块的硬件组成，包括发送器（TX）、接收器（RX）、控制寄存器、状态寄存器等关键部分。</p> <p>(3) 熟悉查询模式的工作原理：理解在嵌入式系统中，如何通过软件轮询（查询模式）来检测 USART 的状态，从而控制数据的发送和接收过程。</p> |
| 能力目标 | <p>(1) 配置 USART 参数：能够根据实验要求，正确配置微控制器的 USART 模块参数，如波特率、数据格式等。</p> <p>(2) 编写 USART 通信程序：掌握在查询模式下，编写代码实现数据的发送和接收功能，包括初始化 USART、发送数据、接收数据及处理接收到的数据。</p> <p>(3) 调试与故障排查：能够使用开发环境和调试工具，对 USART 通信程序进行调试，识别并解决通信中的问题，如数据丢失、乱码等。</p> |
| 教学重点 | <p>(1) USART 初始化配置：详细讲解如何通过配置相关寄存器来初始化 USART，包括波特率设置、数据格式配置等。</p> <p>(2) 查询模式的实现：演示如何通过查询 USART 的状态寄存器来判断数据是否发送完毕或接收完成，并据此控制数据流的读写。</p> <p>(3) 实验案例分析与实践：提供一个或多个基于具体微控制器的 USART 通信实验案例，包括代码讲解和实验步骤，让学生在实践中加深理解。</p> |
| 教学难点 | <p>(1) 状态监测与错误处理：在查询模式下，如何有效地监测 USART 的状态，及时响应数据接收完成或发送错误等事件，并设计合理的错误处理机制。</p> |

实验六

| | |
|--|--|
| 实验名称：使用 USART 发送和接收数据(中断模式)和读取超声波传感器（HC-SR04）数据到串口（学时：4） | |
| 知识目标 | <p>(1) 理解 USART 中断机制：使学生深入理解 USART（通用同步/异步收发传输器）的中断工作原理，包括发送中断、接收中断及其触发条件。</p> |

| | |
|------|--|
| | <p>理解 DMA（直接存储器访问）工作原理：使学生深入理解 DMA 控制器的工作机制，包括数据传输流程、通道配置、传输优先级等。</p> <p>（2）掌握 USART 中断配置：了解如何在微控制器中配置 USART 中断，包括中断优先级、中断向量表、中断服务程序（ISR）的编写等。</p> <p>（3）熟悉中断模式下的数据处理：理解在中断模式下，如何高效地处理 USART 发送和接收的数据，确保数据的完整性和实时性。</p> |
| 能力目标 | <p>（1）配置 USART 中断：能够根据实验要求，正确配置微控制器的 USART 中断参数，实现数据的异步发送和接收。</p> <p>（2）编写中断服务程序：掌握编写 USART 中断服务程序的方法，包括数据发送、接收及处理逻辑，确保在中断发生时能够正确响应。</p> <p>（3）调试与优化中断程序：能够使用开发环境和调试工具，对 USART 中断程序进行调试，识别并解决中断处理中的延迟、数据丢失等问题。</p> |
| 教学重点 | <p>（1）USART 中断配置与初始化：详细讲解 USART 中断的配置步骤，包括中断优先级设置、中断使能、中断服务程序的关联等。</p> <p>（2）中断服务程序的编写：演示如何编写 USART 发送和接收的中断服务程序，包括数据缓冲区的管理、数据发送和接收的逻辑实现。</p> <p>（3）实验案例分析与实践：提供一个或多个基于具体微控制器的 USART 中断通信实验案例，包括代码讲解和实验步骤，让学生在实践中加深理解。</p> |
| 教学难点 | <p>（1）中断优先级与嵌套处理：理解中断优先级的概念，以及如何在多个中断源共存的情况下，合理设置优先级，避免中断嵌套处理中的冲突和死锁。</p> <p>（2）DMA 传输参数的设置：理解 DMA 传输参数（如传输长度、传输方向、传输模式等）的配置方法，以及如何根据实际需求进行优化。</p> |

实验七

| | |
|--------------------------|--|
| 实验名称：使用通用定时器闪烁 LED（学时：2） | |
| 知识目标 | <p>（1）理解通用定时器的工作原理：使学生深入理解通用定时器的基本结构、工作原理和配置方法，包括计数模式、时钟源选择、预分频器设置等。</p> <p>（2）掌握 LED 闪烁的实现原理：了解 LED 的工作原理，以及如何通过控制 GPIO（通用输入输出）引脚的高低电平来实现 LED 的点亮和熄灭。</p> |

| | |
|------|--|
| | <p>(3) 熟悉定时器中断的使用：理解定时器中断的概念、中断优先级和中断服务函数的编写，以及如何通过定时器中断来控制 LED 的闪烁频率。</p> |
| 能力目标 | <p>(1) 配置通用定时器：能够根据实验要求，正确配置微控制器的通用定时器参数，如计数模式、时钟源、预分频值等。</p> <p>(2) 编写中断服务函数：掌握编写定时器中断服务函数的方法，实现定时器的中断处理，从而控制 LED 的闪烁。</p> |
| 教学重点 | <p>(1) 通用定时器的配置：详细讲解通用定时器的配置步骤，包括计数模式的选择、时钟源的设置、预分频器的调整等。</p> <p>(2) LED 闪烁的实现：介绍如何通过控制 GPIO 引脚的高低电平来实现 LED 的点亮和熄灭，以及如何通过定时器中断来控制 LED 的闪烁频率。</p> <p>(3) 中断服务函数的编写：演示如何编写定时器中断服务函数，包括中断服务函数的入口、中断标志位的清除、LED 状态的控制等。</p> |
| 教学难点 | <p>(1) 定时器参数的优化：如何根据 LED 闪烁的频率要求，合理设置定时器的参数（如预分频值、计数值等），以实现精确的定时控制。</p> <p>(2) 中断优先级与嵌套：理解中断优先级的概念，以及如何在复杂系统中处理多个中断的嵌套和优先级冲突。</p> <p>(3) 硬件连接与调试：如何正确连接 LED 与微控制器的 GPIO 引脚，并进行硬件调试，以确保 LED 能够正常闪烁。</p> |

实验八

| | |
|--------------------------------|---|
| 实验名称：使用通用定时器产生 PWM 驱动蜂鸣器（学时：4） | |
| 知识目标 | <p>(1) 理解 PWM（脉冲宽度调制）的基本原理：使学生深入理解 PWM 的概念、工作原理及其在驱动负载（如蜂鸣器）中的应用。</p> <p>(2) 掌握通用定时器的 PWM 功能：了解通用定时器如何配置为 PWM 模式，包括输出比较模式、占空比调整、频率设置等。</p> <p>(3) 熟悉蜂鸣器的工作原理：了解蜂鸣器的种类、驱动方式及在音频信号产生中的应用。</p> |
| 能力目标 | <p>(1) 配置通用定时器为 PWM 模式：能够根据实验要求，正确配置微控制器的通用定时器为 PWM 模式，并调整占空比和频率。</p> |

| | |
|------|---|
| | <p>(2) 编写 PWM 控制程序：掌握编写 PWM 控制程序的方法，实现 PWM 信号的输出，并驱动蜂鸣器产生不同频率的声音。</p> <p>(3) 调试与验证：能够使用开发环境和调试工具，对程序进行调试，验证 PWM 信号的正确性和蜂鸣器的发声效果。</p> |
| 教学重点 | <p>(1) PWM 的基本原理与应用：详细讲解 PWM 的概念、工作原理及其在驱动蜂鸣器等负载中的应用。</p> <p>(2) 通用定时器的 PWM 配置：介绍如何将通用定时器配置为 PWM 模式，包括输出比较模式的选择、占空比的调整、频率的设置等。</p> <p>(3) PWM 控制程序的编写：演示如何编写 PWM 控制程序，包括初始化定时器、设置 PWM 参数、启动 PWM 输出等。</p> <p>(4) 蜂鸣器的驱动与调试：讲解如何连接蜂鸣器到微控制器的 PWM 输出引脚，并进行调试，确保蜂鸣器能够正确发声。</p> |
| 教学难点 | <p>(1) PWM 参数的精确设置：如何根据蜂鸣器的特性和实验要求，精确设置 PWM 的占空比和频率，以实现所需的发声效果。</p> <p>(2) 定时器资源的合理分配：在微控制器资源有限的情况下，如何合理分配定时器资源，以同时满足多个 PWM 输出的需求。</p> |

实验九

| | |
|---|---|
| <p>实验名称：使用 ADC 读取气体传感器数据 (MQ-2) 和重定向 printf 函数到串口输出的多种方法 (学时：2)</p> | |
| 知识目标 | <p>(1) 理解 ADC (模数转换器) 的工作原理：使学生深入理解 ADC 的基本概念和工作原理，包括采样、量化、编码等过程，以及 ADC 的主要性能指标 (如分辨率、转换速率等)。</p> <p>(2) 掌握串口通信基础：了解串口通信的基本原理，包括波特率、数据位、停止位、校验位等参数设置，以及 UART (通用异步收发传输器) 的工作原理。</p> <p>(3) 掌握 MQ-2 气体传感器的特性与应用：了解 MQ-2 气体传感器的工作原理、特性参数、灵敏度调整方法及其在气体检测中的应用，特别是对可燃气体 (如液化气、丙烷、氢气等) 的检测。</p> |

| | |
|------|---|
| | <p>(4) 熟悉嵌入式系统中 ADC 的配置与使用：了解在嵌入式系统中如何配置和使用 ADC 外设，包括选择 ADC 通道、设置采样率、读取 ADC 值等。</p> |
| 能力目标 | <p>(1) 配置 ADC 外设：能够根据实验要求，正确配置微控制器的 ADC 外设参数，如选择通道、设置分辨率、启动转换等。</p> <p>(2) 读取 MQ-2 传感器数据：掌握通过 ADC 读取 MQ-2 气体传感器数据的方法，包括数据预处理、滤波、标度转换等。</p> <p>(3) 数据处理与显示：能够对读取到的数据进行处理，如计算气体浓度、判断气体类型等，并通过串口或其他接口将结果显示出来。</p> |
| 教学重点 | <p>(1) ADC 的配置与使用：详细讲解 ADC 的配置步骤和使用方法，包括选择通道、设置参数、启动转换和读取结果等。</p> <p>(2) MQ-2 传感器的特性与应用：介绍 MQ-2 传感器的工作原理、特性参数及其在气体检测中的应用，特别是如何调整传感器的灵敏度以适应不同的检测需求。</p> <p>(3) 数据读取与处理：演示如何通过 ADC 读取 MQ-2 传感器的数据，并进行预处理、滤波和标度转换，最终得到气体浓度的实际值。</p> <p>(4) 实验案例分析与实践：提供一个或多个基于具体微控制器的实验案例，包括代码讲解和实验步骤，让学生在实践中加深理解。</p> <p>(5) 串口通信配置：详细讲解串口通信的配置步骤，包括波特率设置、数据格式配置、串口初始化等。</p> <p>(6) 重定向方法实现：演示多种重定向 printf 函数到串口输出的方法</p> |
| 教学难点 | <p>(1) ADC 配置参数的优化：如何根据 MQ-2 传感器的特性和实验需求，合理设置 ADC 的采样率、分辨率等参数，以获得最佳的测量精度和响应速度。</p> <p>(2) 数据预处理与滤波：在读取 MQ-2 传感器数据时，如何进行有效的预处理和滤波，以减少噪声干扰，提高数据的准确性和可靠性。</p> <p>(3) 气体浓度计算与判断：如何将读取到的 ADC 值转换为实际的气体浓度，并根据浓度值判断气体的类型和危险程度。</p> <p>(4) 硬件连接与调试：如何正确连接 MQ-2 传感器与微控制器的 ADC 通道，并进行硬件调试，以确保数据传输的正确性和稳定性。</p> |

实验十

| | |
|--|---|
| 实验名称：使用硬件 I2C 读取环境光强度传感器数据（BH1750）（学时：4） | |
| 知识目标 | <p>（1）理解 I2C 总线通信协议：使学生深入理解 I2C 总线的通信原理、协议规范及其在嵌入式系统中的应用。</p> <p>（2）掌握 BH1750 环境光强度传感器的工作原理：了解 BH1750 传感器的特性、测量原理、数据输出格式及通信协议。</p> <p>（3）熟悉微控制器的 I2C 接口配置：掌握微控制器 I2C 接口的初始化、配置及数据传输方法。</p> |
| 能力目标 | <p>（1）配置微控制器 I2C 接口：能够根据实验要求，正确配置微控制器的 I2C 接口参数，实现与 BH1750 传感器的通信。</p> <p>（2）编写 I2C 通信程序：掌握 I2C 通信程序的编写方法，包括初始化 I2C 接口、发送/接收数据、处理应答信号等。</p> <p>（3）读取 BH1750 传感器数据：能够使用 I2C 总线从 BH1750 传感器读取环境光强度数据，并解析数据格式。</p> <p>（4）数据处理与显示：能够对读取到的环境光强度数据进行处理，并通过合适的显示方式（如串口打印、LCD 显示等）呈现出来。</p> |
| 教学重点 | <p>（1）I2C 总线通信协议：详细讲解 I2C 总线的通信原理、协议规范、数据传输格式及错误处理机制。</p> <p>（2）BH1750 传感器工作原理：介绍 BH1750 传感器的特性、测量原理、数据输出格式及通信协议，重点讲解如何通过 I2C 总线读取传感器数据。</p> |
| 教学难点 | <p>（1）I2C 总线通信协议的深入理解：如何准确理解 I2C 总线的通信协议，包括起始条件、停止条件、应答信号等，以确保通信的可靠性。</p> <p>（2）BH1750 传感器数据读取与解析：如何正确读取 BH1750 传感器的数据，并解析数据格式，包括数据包的组成、数据位的含义及数据转换方法等。</p> <p>（3）I2C 接口配置与错误处理：如何根据实验要求和微控制器的特性，正确配置 I2C 接口参数，并处理可能出现的通信错误，如超时、应答失败等。</p> |

实验十一

| | |
|---------------------------------------|--|
| 实验名称：使用硬件 I2C 读取温湿度传感器数据（SHT30）（学时：2） | |
| 知识目标 | <p>（1）理解 I2C 总线通信协议：深入理解 I2C 总线的通信原理、协议规范及其在嵌入式系统中的应用。</p> <p>（2）掌握 SHT30 温湿度传感器的工作原理：了解 SHT30 传感器的特性、测量原理、数据输出格式及通信协议。</p> <p>（3）熟悉微控制器的 I2C 接口配置：掌握微控制器 I2C 接口的初始化、配置及数据传输方法。</p> |
| 能力目标 | <p>（1）配置微控制器 I2C 接口：能够根据实验要求，正确配置微控制器的 I2C 接口参数，实现与 SHT30 传感器的通信。</p> <p>（2）编写 I2C 通信程序：掌握 I2C 通信程序的编写方法，包括初始化 I2C 接口、发送/接收数据、处理应答信号等。</p> <p>（3）读取 SHT30 传感器数据：能够使用 I2C 总线从 SHT30 传感器读取温湿度数据，并解析数据格式。</p> <p>（4）数据处理与显示：能够对读取到的温湿度数据进行处理，并通过合适的显示方式（如串口打印、LCD 显示等）呈现出来。</p> |
| 教学重点 | <p>（1）I2C 总线通信协议：详细讲解 I2C 总线的通信原理、协议规范、数据传输格式及错误处理机制。</p> <p>（2）SHT30 传感器工作原理：介绍 SHT30 传感器的特性、测量原理、数据输出格式及通信协议，重点讲解如何通过 I2C 总线读取传感器数据。</p> |
| 教学难点 | <p>（1）I2C 总线通信协议的深入理解：如何准确理解 I2C 总线的通信协议，包括起始条件、停止条件、应答信号等，以确保通信的可靠性。</p> |

实验十二

| | |
|-----------------------------------|--|
| 实验名称：读取超声波传感器（HC-SR04）数据到串口（学时：4） | |
| 知识目标 | <p>（1）超声波传感器（HC-SR04）的工作原理及其在嵌入式系统中的应用。</p> <p>（2）掌握超声波时序图</p> |
| 能力目标 | <p>（1）配置微控制器 GPIO 接口模式。</p> <p>（2）根据时序图编写驱动程序</p> <p>（3）串口输出</p> |

| | |
|------|-------------------------------|
| 教学重点 | (1) 超声波时序图：详细读懂时序图，根据时序图编写程序。 |
| 教学难点 | (1) 超声波时序图。 |

实验十三

| | |
|-------------------------------------|---|
| 实验名称：嵌入式系统开发技能练兵及检验 01（混合知识点）（学时：2） | |
| 知识目标 | <p>题目：定时器 4 秒闪烁一次灯，并且发送串口数据，如“灯亮”，4 秒后，“灯灭”，4 秒后，“灯亮”，如此循环。</p> <p>(1) 混合知识点：定时器、LED 灯、串口。</p> <p>(2) 检验学生掌握情况。</p> |
| 能力目标 | <p>(1) 将三个实验（定时器、LED 灯、串口）融合贯通，正确配置。</p> <p>(2) 根据题目要求编写控制程序</p> <p>(3) 理解和学会。</p> |
| 教学重点 | (1) 融合知识点。 |

实验十四

| | |
|-------------------------------------|--|
| 实验名称：嵌入式系统开发技能练兵及检验 02（混合知识点）（学时：4） | |
| 知识目标 | <p>题目：使用光照强度传感器，LED 灯实现：当光照强度小于 100Lux 时，LED 灯亮；否则不亮。</p> <p>(1) 混合知识点：光照强度传感器 IIC、LED。</p> <p>(2) 软件代码编写</p> <p>(3) 检验学生掌握情况。</p> |
| 能力目标 | <p>(1) 将三个实验（光照强度传感器 IIC、LED 灯、串口）融合贯通，正确配置。</p> <p>(2) 根据题目要求编写控制程序</p> <p>(3) 理解和学会。</p> |
| 教学重点 | (1) (1) 融合知识点。 |

实验十五

| | |
|-------------------------------------|--|
| 实验名称：嵌入式系统开发技能练兵及检验 03（混合知识点）（学时：2） | |
|-------------------------------------|--|

| | |
|------|---|
| 知识目标 | <p>题目：定时 4 秒采集一次超声波数据发送到串口上面，并且当距离小于 100cm 时，LED 灯亮，当距离大于 100cm 时，LED 灯灭。</p> <p>(1) 混合知识点：超声波传感器、LED 灯。</p> <p>(2) 软件代码编写</p> <p>(3) 检验学生掌握情况。</p> |
| 能力目标 | <p>(1) 将三个实验（超声波传感器、LED 灯、串口）融合贯通，正确配置。</p> <p>(2) 根据题目要求编写控制程序</p> <p>(3) 理解和学会。</p> |
| 教学重点 | <p>(1) (1) 融合知识点。</p> |

实验十六

| | |
|-------------------------------------|---|
| 实验名称：嵌入式系统开发技能练兵及检验 04（混合知识点）（学时：4） | |
| 知识目标 | <p>题目：按按键 F1，采集距离，发送到串口；按按键 F2，采集温湿度，发送到串口。</p> <p>(1) 混合知识点：按键、超声波、温湿度、LED 灯。</p> <p>(2) 软件代码编写</p> <p>(3) 检验学生掌握情况。</p> |
| 能力目标 | <p>(1) 将四个实验（按键、超声波、温湿度、LED 灯）融合贯通，正确配置。</p> <p>(2) 根据题目要求编写控制程序</p> <p>(3) 理解和学会。</p> |
| 教学重点 | <p>(1) 融合知识点。</p> |

实验十七

| | |
|-------------------------------------|---|
| 实验名称：嵌入式系统开发技能练兵及检验 05（混合知识点）（学时：2） | |
| 知识目标 | <p>题目：选择一款传感器，将数据显示到 LCD 屏上。</p> <p>(1) 混合知识点：传感器、驱动 LCD 屏幕（自学）。</p> <p>(2) 软件代码编写</p> <p>(3) 检验学生掌握情况。</p> |
| 能力目标 | <p>(1) 学会自学驱动 LCD 屏幕程序。</p> |

| | |
|------|----------------------------------|
| | (2) 将所学知识点（传感器）融入。 (3) 理解和学会。 |
| 教学重点 | (1) 自学驱动 LCD 屏幕程序。 |

实验十八

| | |
|---------------------------|------|
| 实验名称：对练兵题目验收及本学期作回顾（学时：4） | |
| 知识目标 | 回顾总结 |

六、考核评价

重视对学生学习全过程的质量监控和考评，关注评价的多元性，建立教师评价和学生自评、互评、过程评价和结果评价、课内评价和课外评价、理论评价和实践评价相结合的 课程评价体系，采取知识、能力及素质三方面考核相结合的方式，最终对学生 进行综合评价。学生最终的成绩由几个方面确定：

综合成绩（100%） = 平时考评（30%） + 期末技能练兵考评（70%）

平时成绩包含学生作业提交情况和出勤情况。

期末技能练兵考评包含练兵题目的验收打分情况。