

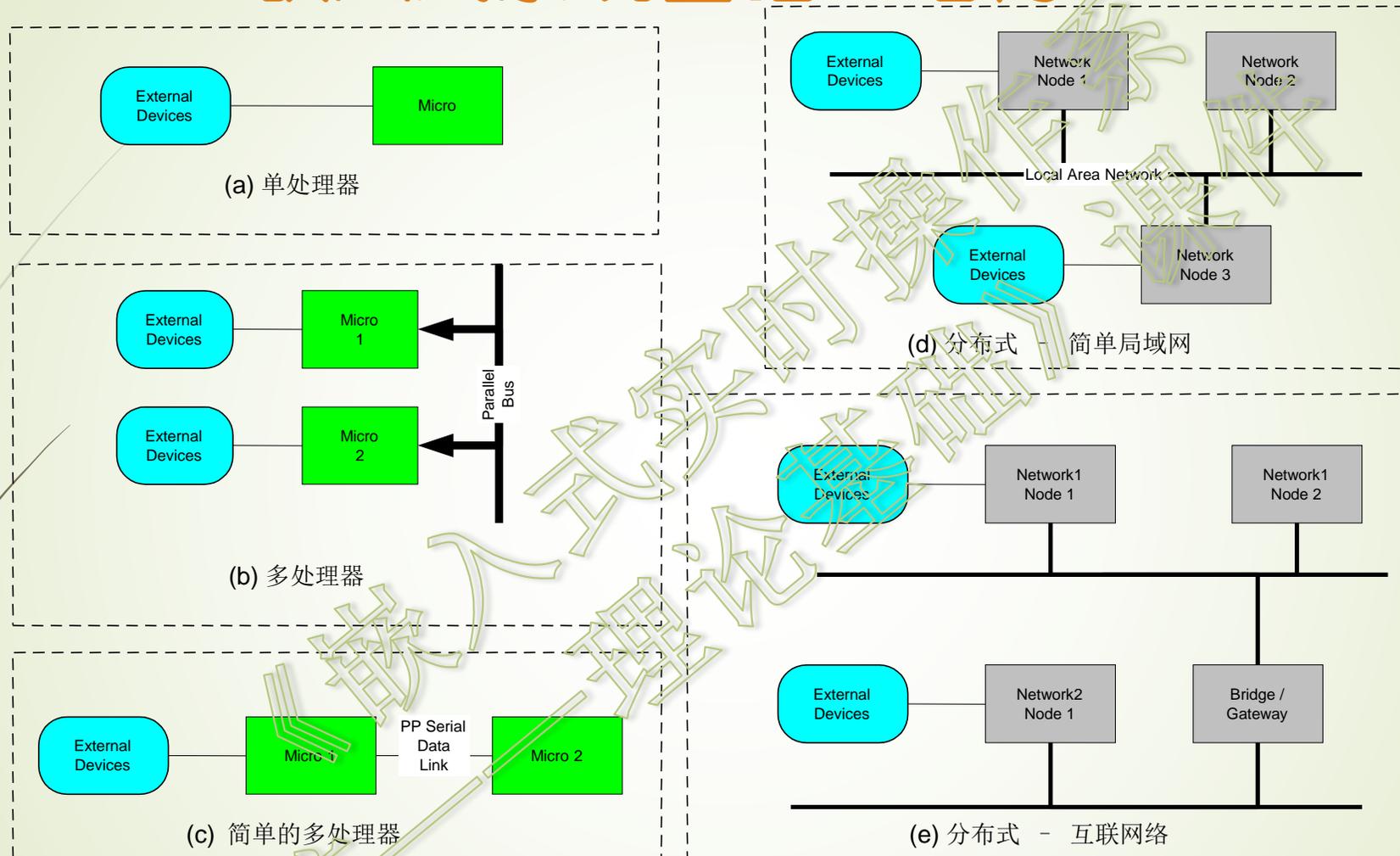
# 第10章 操作系统基本结构和功能

# 本章内容

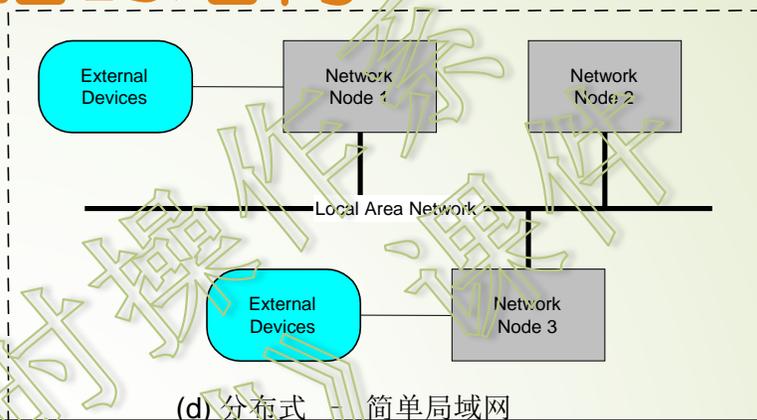
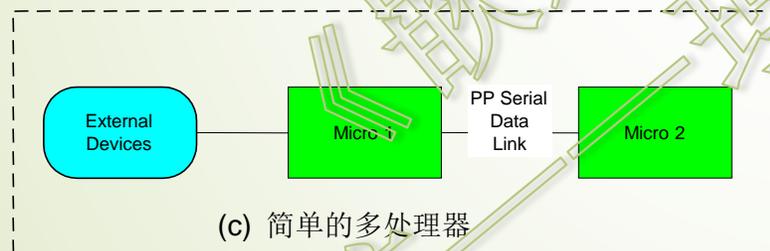
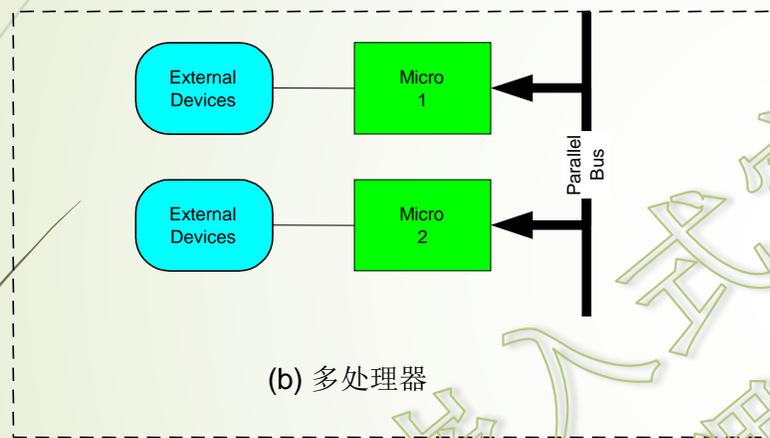
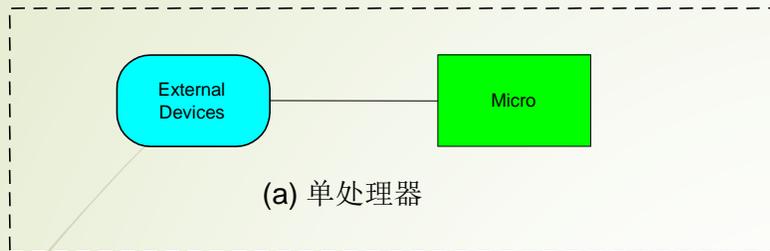
- 回顾常见的实时嵌入式系统硬件架构。
- 展示了在单处理器非实时嵌入式系统环境下的代码结构。
- 在单处理器系统中开发代码结构的概念模型。
- 阐述了超微内核和微内核实时操作系统的概念模型。
- 讨论一个典型大型通用实时系统RTOS的概念模型。

《嵌入式实时操作系统——理论基础》

# 嵌入式系统基础的结构



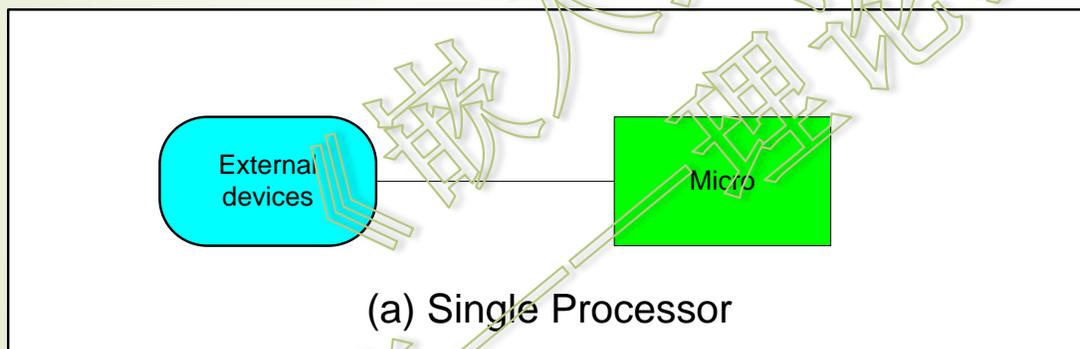
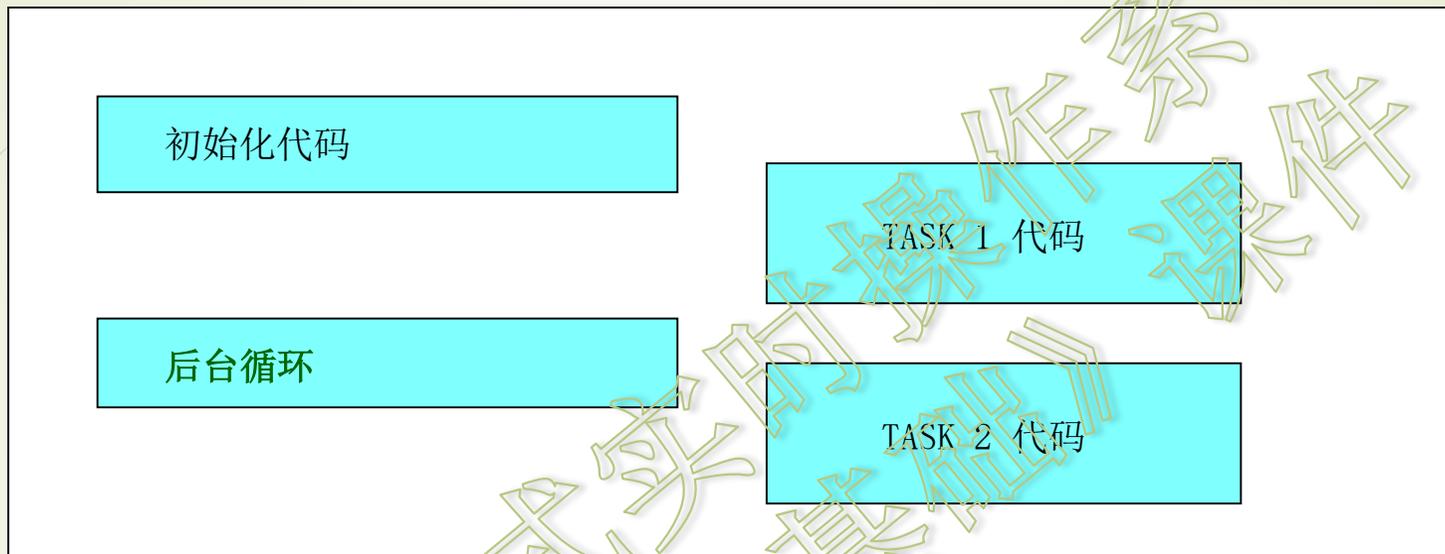
# 嵌入式系统基础的结构



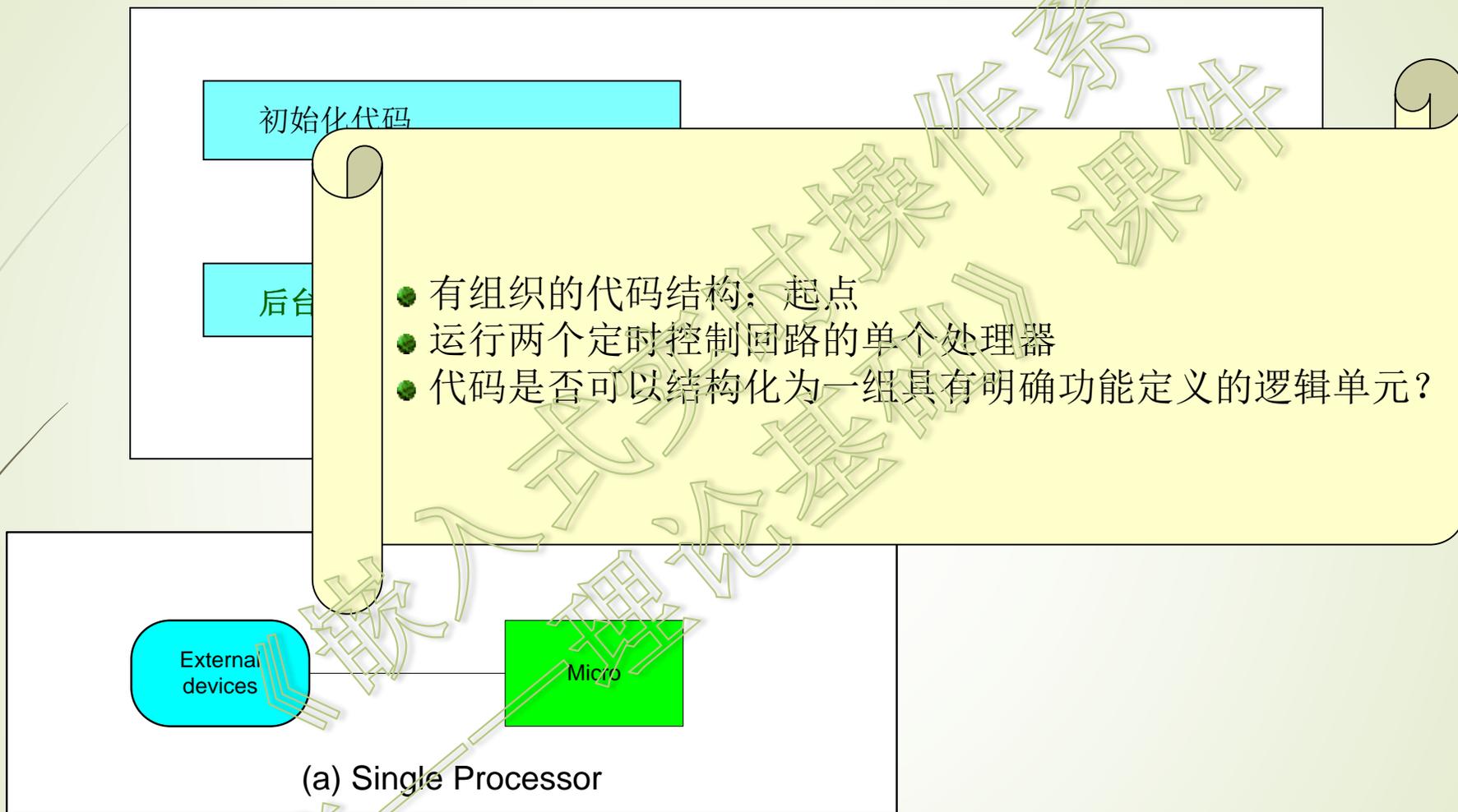
主要的问题:

1. 需要哪些RTOS功能来支持所有这些架构?
2. 哪些 RTOS 功能适用于特定架构?
3. “一刀切”的方法有意义吗?
4. RTOS 如何能做到可配置?

# 代码组织：一个简单单处理器应用 (1)



# 代码组织：一个简单单处理器应用 (1)



# 代码组织：一个简单的单处理器应用 (2)

## 初始化代码

### Declaration of Program Items

- Variables
- Constants
- Absolute Address (Devices, Peripherals, etc)

### Set-up Operations

- CPU (registers)
- Devices
- Peripherals
- Interrupt vectors
- Interrupt timers

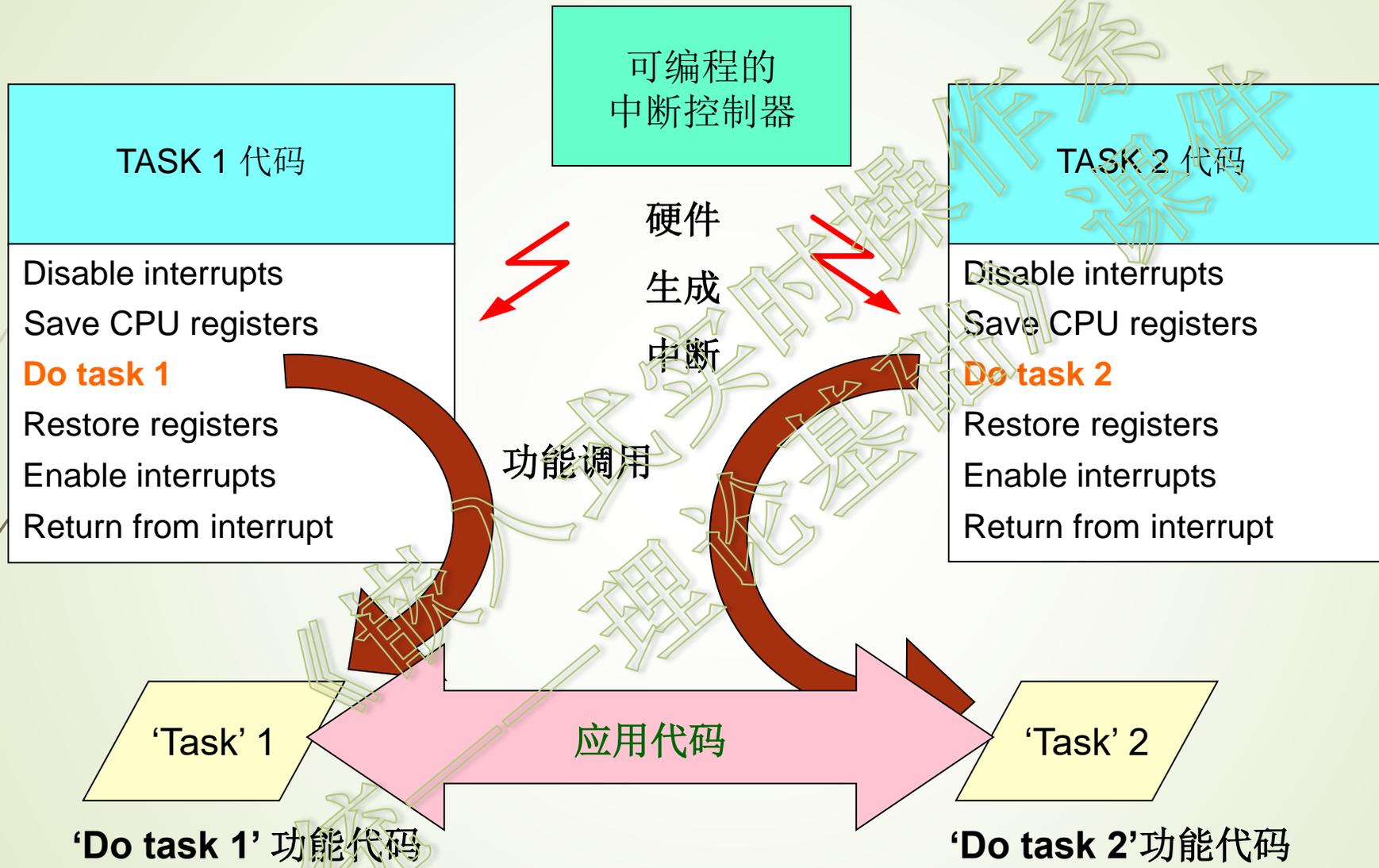
## 后台循环 (2)

```
Begin  
Halt;  
End.
```

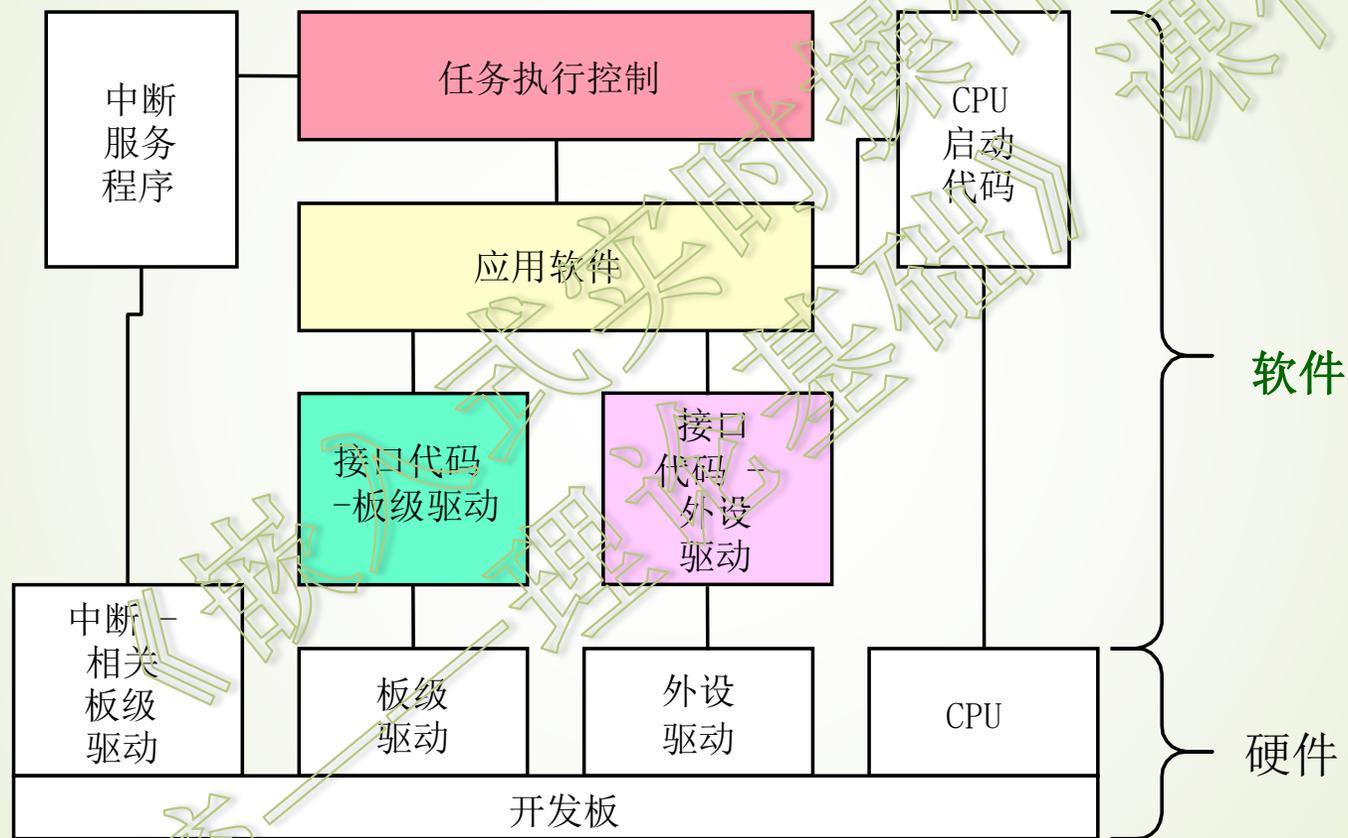
## 后台循环

```
Begin  
    loop:  
        Jump to loop:  
End.
```

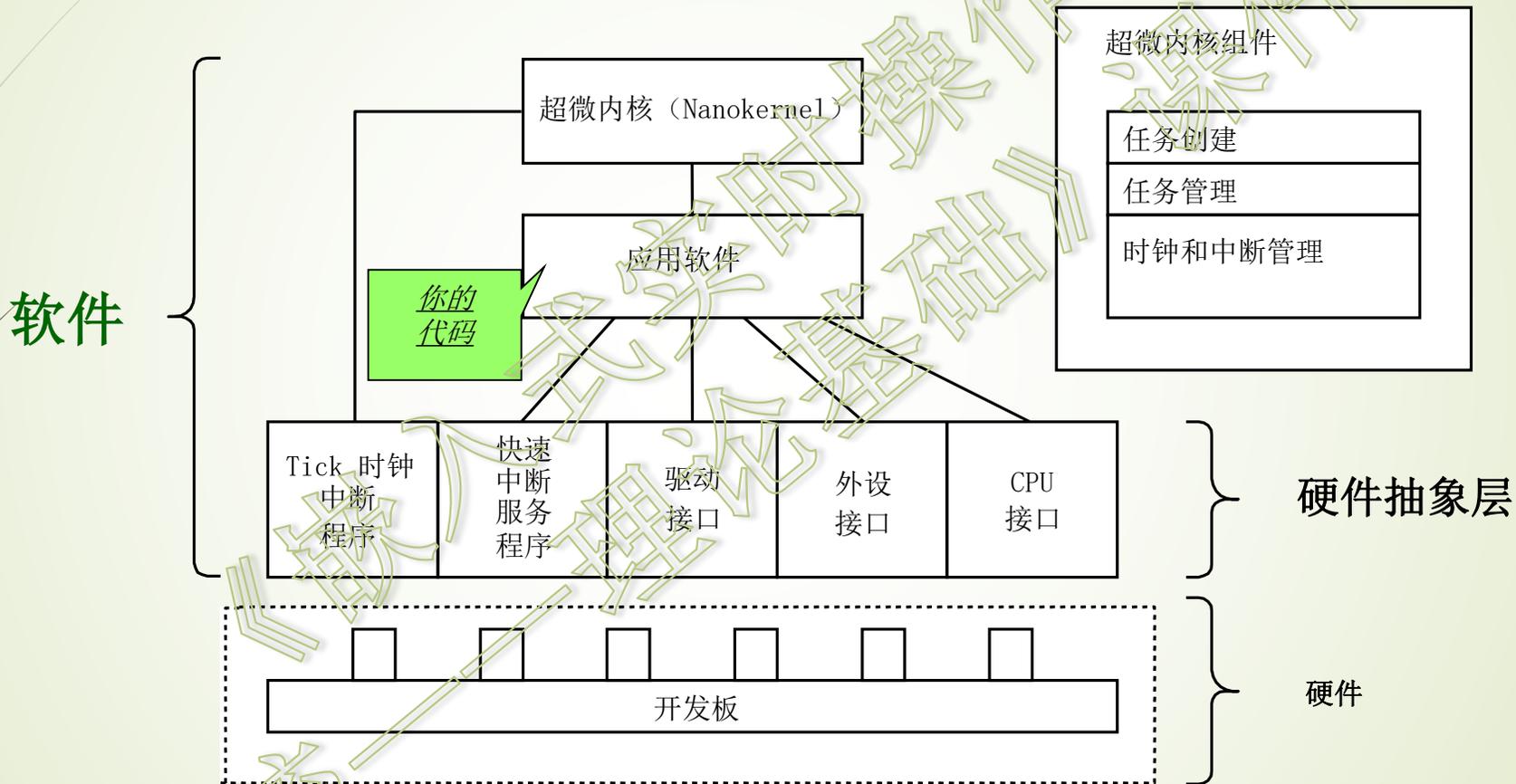
# 代码组织：一个简单的单处理器应用 (3)



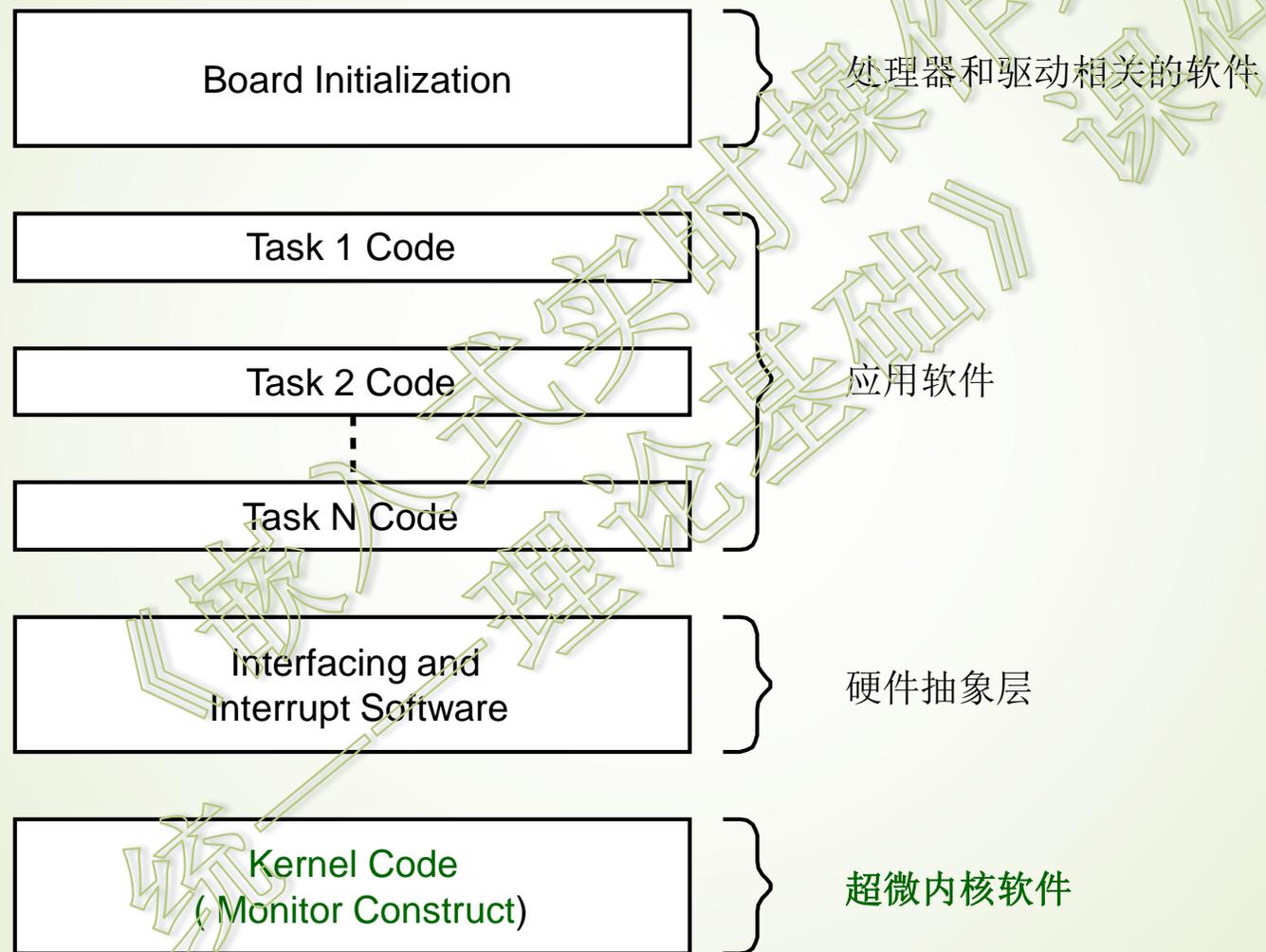
# 概念模型——简单的单处理器硬件和软件架构



# 概念模型——基于超微内核的RTOS



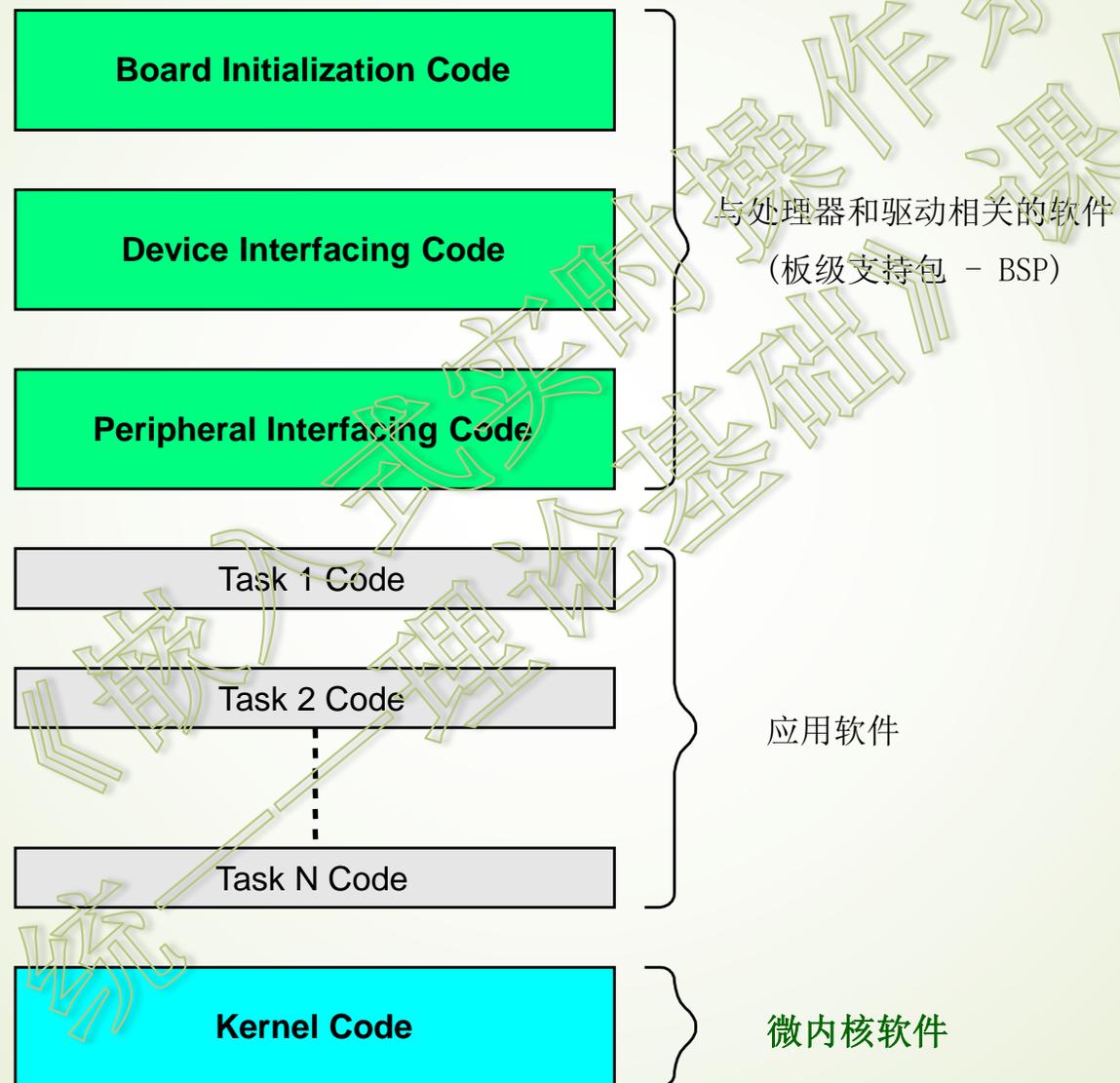
# 代码组织——基于超微内核的RTOS



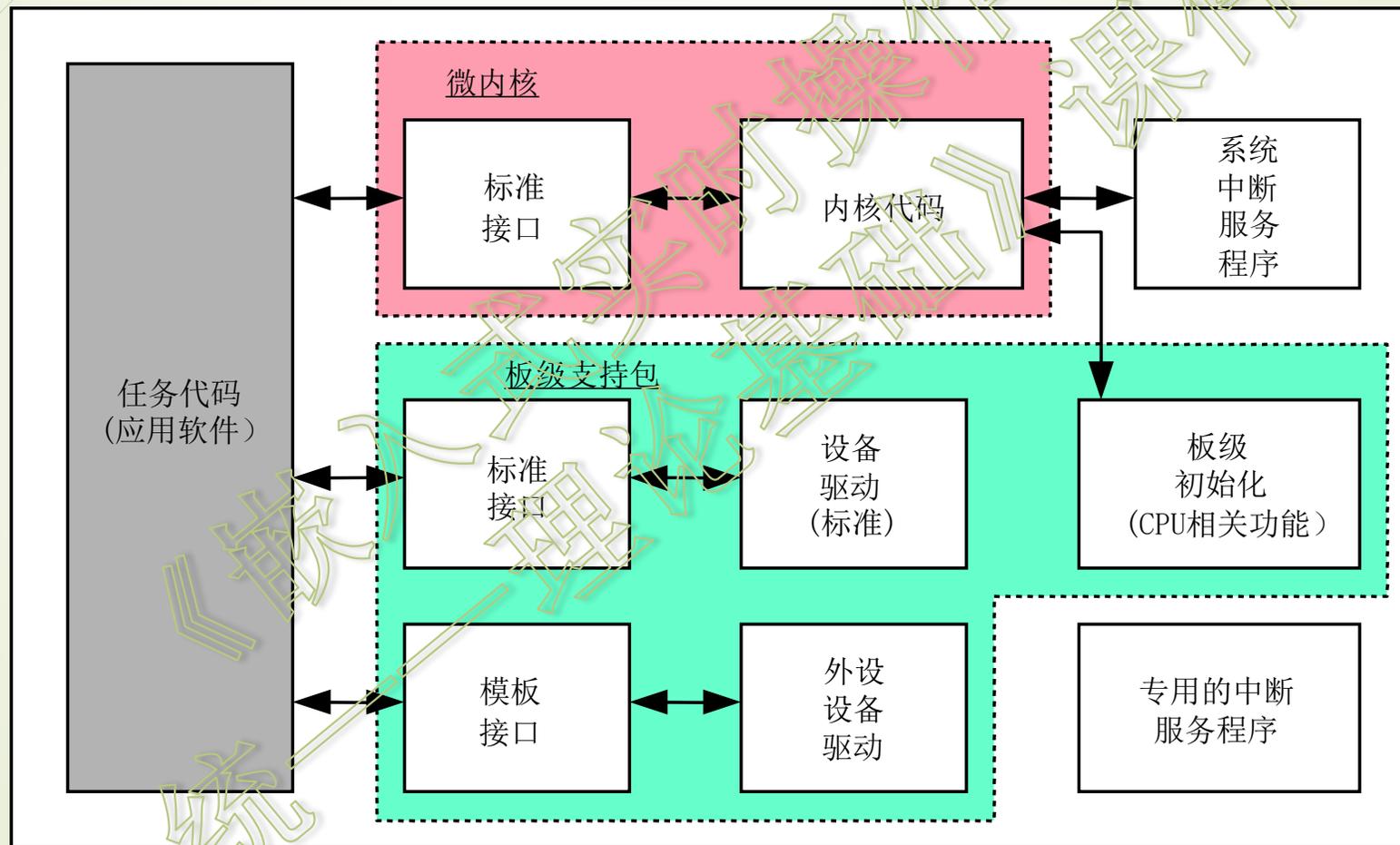
# 超微内核软件——典型的子程序

- 初始化OS。
- 创建一个任务。
- 启动一个任务。
- 删除一个任务。
- 控制滴答。
- 设置时间片和延迟。
- 设置和获取时钟时间。

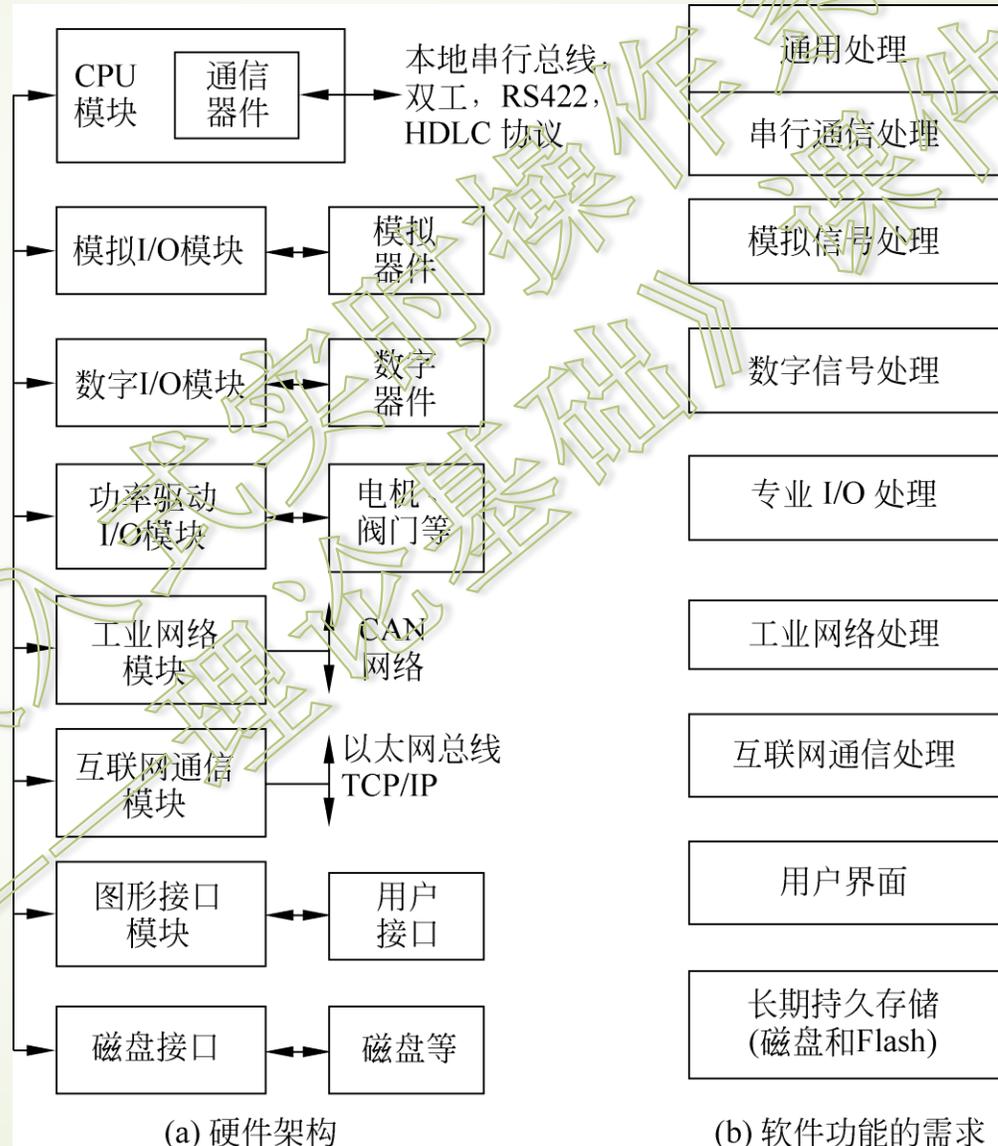
# 代码组织——基于微内核的RTOS



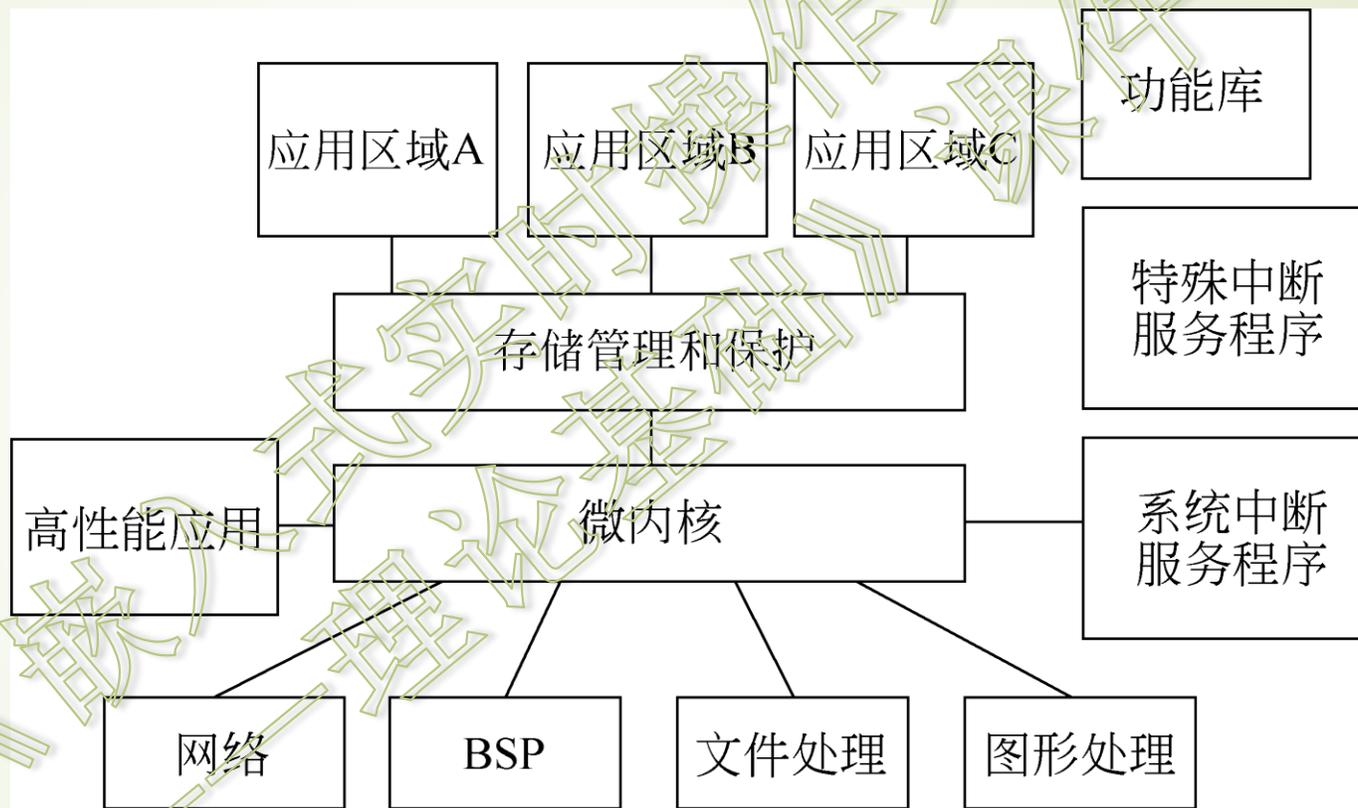
# 软件概念模型——小型的基于微内核的RTOS



# 大型通用嵌入式系统的硬件架构和软件功能



# 大型通用嵌入式系统RTOS概念模型



本章结束

# 版权说明和联系方式

- 课件由《嵌入式实时操作系统——理论基础》一书翻译团队成员何灵渊、何小庆、张爱华和付元斌编写，图书原作者提供原始素材，清华大学出版社提供部分图片。课件可用于非商业场合，比如教学和研究课题，商业使用需联系作者。
- 如果你是高校老师，希望以本书内容开设课程，需要全套的课件（PPT格式）可以联系：
  - ① 本书译者：何小庆老师 [xiaoqinghe@live.com](mailto:xiaoqinghe@live.com) 或者添加何老师微信 allanhexq（注明你的姓名+学校+专业）
  - ② 本书责编：刘星 [liux@tup.tsinghua.edu.cn](mailto:liux@tup.tsinghua.edu.cn) 电话：83470219 QQ：1468682976

注意：请提供教材订购证明，获取全书的PPT课件。