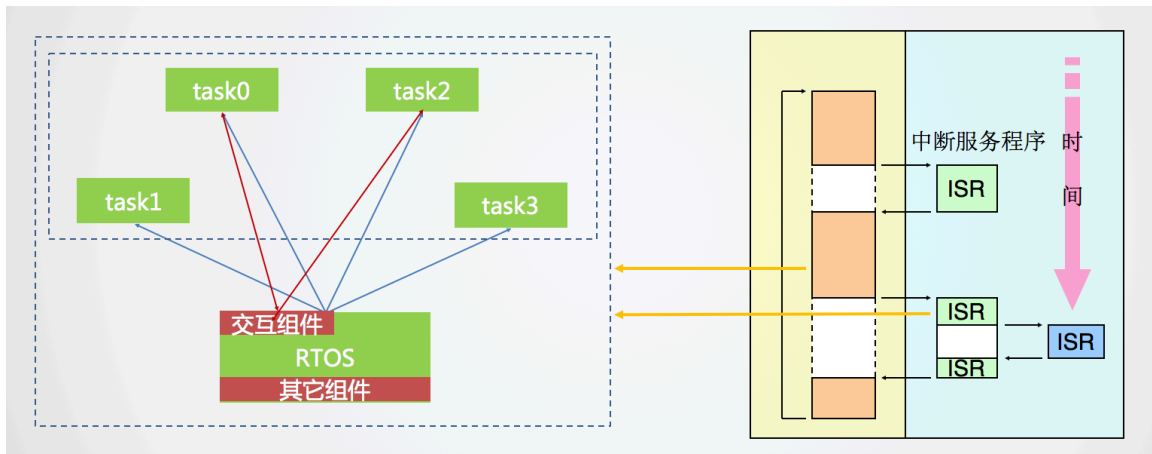


内容回顾

在这节课中，我们回顾了嵌入式操作系统(RTOS)的基本原理，进而对其基本工作流程有一个基础的认识。

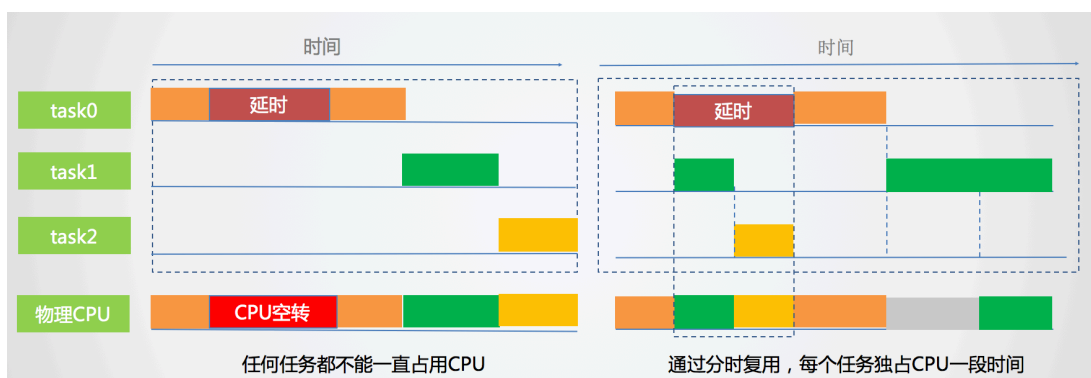
首先，**RTOS 一种系统软件**，提供了帮助我们进行任务管理的框架，以及任务之间进行交互的组件。



相比较使用于前后台方式，**引入 RTOS 之后我们能够通过任务的划分更方便的编写应用代码**。因为划分出的每个任务都可以独立地去执行它的任务函数，且各个任务函数在运行时就好像其独占 CPU 一样。

```
void task1Entry (void * param)
{
    for (;;)
    {
        // some code here
    }
}
```

在实际运行过程时，**RTOS 通过将 CPU 分时交给各个任务运行实现从表面上看“各个任务独占 CPU 运行”现象**。



此外，RTOS 还提供了其它的模块用于实现事件控制、资源共享、消息传递、存储管理等相关功能。

其它 OS 示例

除课程介绍的 tinyOS 之外，还有其它很多 RTOS。这里列出几种比较知名的，在其它课时的学习指引中，针对该课时所讨论的问题，我们会尽可能提出相应的解决方案，方便参考对照。

KEIL RTX(安装 MDK-ARM 后附带)

Keil RTX 是免版税的确定性实时操作系统，适用于 ARM 和 Cortex-M 设备。使用该系统可以创建同时执行多个功能的程序，并有助于创建结构更好且维护更加轻松的应用程序。

UC/OS II/ III([HTTP://WWW.MICRIUM.COM](http://www.micrium.com))

μC/OS II(Micro-Controller Operating System Two)是一个可以基于 ROM 运行的、可裁剪的、抢占式、实时多任务内核，具有高度可移植性，特别适合于微处理器和控制器，适合很多商业操作系统性能相当的实时操作系统(RTOS)。从 1992 年开始，由于高度可靠性、鲁棒性和安全性，μC/OS II 已经广泛使用在从照相机到航空电子产品的各种应用中。

FREERTOS([HTTP://WWW.FREERTOS.ORG](http://www.freertos.org))

FreeRTOS 是一个迷你的实时操作系统内核。作为一个轻量级的操作系统，功能包括：任务管理、时间管理、信号量、消息队列、内存管理、记录功能、软件定时器、协程等，可基本满足较小系统的需要。

RT-THREAD([HTTP://WWW.RT-THREAD.ORG](http://www.rt-thread.org))

RT-Thread 是一个集实时操作系统 (RTOS) 内核、中间件组件和开发者社区于一体的技术平台，由熊谱翔先生带领并集合开源社区力量开发而成，RT-Thread 也是一个组件完整丰富、高度可伸缩、简易开发、超低功耗、高安全性的物联网操作系统。RT-Thread 具备一个 IoT OS 平台所需的所有关键组件，例如 GUI、网络协议栈、安全传输、低功耗组件等等。经过 11 年的累积发展，RT-Thread 已经拥有一个国内最大的嵌入式开源社区，同时被广泛应用于能源、车载、医疗、消费电子等多个行业，累积装机量超过两千万台，成为国人自主开发、国内最成熟稳定和装机量最大的开源 RTOS。

思考练习

- 1、请结合自己的经验，谈下使用 RTOS 前后，你的编程方式和思考问题的方式，有哪些变化？为什么会产生这些变化？（请参考 C0.00 学习指南，分享你的心得体会）

视频勘误

暂无

重点答疑

暂无