

1.3 进程的状态管理_物联网 / 嵌入式工程师 - 慕课网

“ 慕课网慕课教程 1.3 进程的状态管理涵盖海量编程基础技术教程，以图文图表的形式，把晦涩难懂的编程专业用语，以通俗易懂的方式呈现给用户。

全部开发者教程

物联网 / 嵌入式工程师

第 24 周 stm32 芯片 – 智能硬件项目实战与企业笔试

未命名节

第 25 周 大厂必备 – linux 内核与文件系统移植

未命名节

第 26 周 嵌入式开发 – 系统移植 – bootloader、yocto

未命名节

第 27 周 嵌入式底层核心技能 – Linux 设备驱动初级

未命名节

第 28 周 嵌入式底层核心技能 – Linux 设备驱动中级

未命名节

第 29 周 嵌入式底层核心技能 – Linux 设备驱动高级 1

未命名节

第 30 周 嵌入式底层核心技能 – Linux 设备驱动高级 2

未命名节

第 31 周 嵌入式人工智能必备 – Python

未命名节

第 32 周 智能家居项目实战之客户端功能开发

未命名节

第 33 周 智能家居项目实战之网关端功能开发

未命名节

第 34 周 智能家居项目实战之设备端功能开发

未命名节

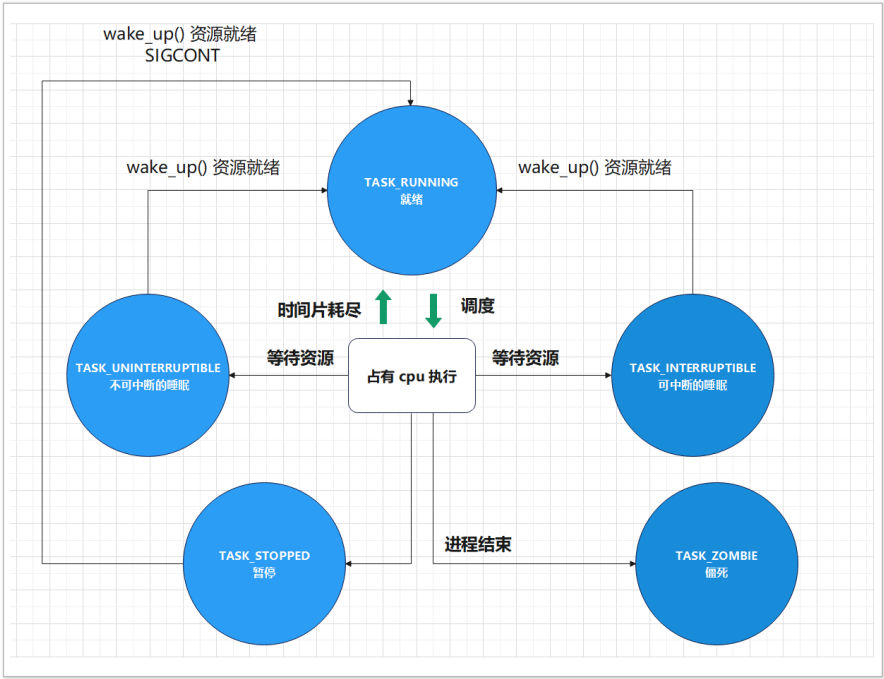
第 35 周 物联网 / 嵌入式项目答辩和就业指导

未命名节

未命名节

3. 进程的状态管理

- 进程是动态过程，操作系统内核在管理整个动态过程时会使用了状态机，
 - 给不同时间节点设计一个状态，通过状态来确定当前的过程进度
 - 在管理动态过程时，使用状态机是一种非常好的方式
- 进程的状态一般分为如下：
 - 运行态 (TASK_RUNNING)：此时进程或者正在运行，或者准备运行，就绪或者正在进行都属于运行态
 - 睡眠态 ()：此时进程在等待一个事件的发生或某种系统资源
 - 可中断的睡眠 (TASK_INTERRUPT)：可以被信号唤醒或者等待事件或者资源就绪
 - 不可中断的睡眠 (TASK_UNTERRUPT)：只能等待特定的事件或者资源就绪
 - 停止态 (TASK_STOPPED)：进程暂停接受某种处理。例如：gdb 调试断点信息处理。
 - 僵尸态 (TASK_ZOMBIE)：进程已经结束但是还没有释放进程资源



示例：通过 用户输入来理解进程状态的变化

```
#include <stdio.h>

int main(void)
{
    int num;

    printf("Please input : ");
    scanf("%d",&num);
}
```

```
    return 0;  
}
```

全文完

本文由 简悦 SimpRead 优化，用以提升阅读体验

使用了 全新的简悦词法分析引擎 beta，[点击查看详细说明](#)

