

# 项目移植到freerots上

## 1) 配置

- 版本选择：项目使用RT-Thread Nano版本，适合资源受限的小型电子产品开发
- 生态优势：具有RTOS领域最大的开源软件包生态体系
- 开发工具：RT-Thread Studio是官方推荐的一站式开发IDE，可显著简化开发流程
- 硬件支持：官网已适配大量芯片（特别是国产芯片）和开发板，降低移植难度
- 内核移植关键点:labCPU目录包含CPU架构相关代码（如context\_switch.s）

## 2) 项目移植难点

- 业务逻辑适配：RT-Thread与FreeRTOS的API差异（如任务创建、信号量等）
- 组件兼容性：(RT-Thread的Shell/Device组件需替换或移植)
- 编译问题
  - 常见错误类型：
    - 未定义符号错误：缺少对应源文件
    - 函数未实现：API未正确替换
    - 函数隐式声明警告表明编译器无法找到函数原型
    - 替换RT-Thread的rt\_kprintf为标准printf
  - 解决方法：
    - 添加缺失的源文件到工程
    - 检查所有的rtthread特有的API是否完成替换
    - 需要显式包含声明这些函数的头文件
    - 复制代码时可能出现问号字符，需检查源文件编码格式
    - 栈大小需保持相同配置；优先级需转换为FreeRTOS对应的优先级值；任务函数签名需保持一致
    - 包含顺序：
      - 基础头文件在前，专用头文件在后，避免循环依赖
    - 常见问题：

- 忘记添加属性结构体声明
- 优先级数值设置反序
  - MONITOR TASK: 95
  - PROTECT TASK: 104
  - ANALYSIS TASK: 113
  - ENERGY TASK: 122
  - INFO TASK: 131
  - COMM TASK: 133
- 栈大小单位混淆（字节 vs 字）
- 硬件中断注意事项：
  - 使用硬件中断通知时，烧写程序后必须重新上下电或复位芯片
  - 两个特定中断可能造成系统故障：
    - 第一个报警表示芯片故障
    - 第二个可能是电磁信号干扰导致，曾通过更换与官方相同阻值的电阻解决
- 错误修复过程：
  - 移除了未使用的level变量和相关保存机制
  - 统一使用FreeRTOS的中断控制API
  - 解决了因RT-Thread和FreeRTOSAPI混用导致的编译错误

### 3) 项目移植思路

- 用CubeMX添加freertos进入工程，生成代码在keil中去掉rtthread相关部分和业务相关部分
- 单独编译freertos部分并测试通过，逐步添加业务代码，并移植相关代码，编译运行
- 移除rtthread特有组件，如shell和device

### 4) 项目展望

- 添加CAN通信体系
- 添加业务场景下的算法优化
- 分布式BMS方案
- 快充和充电桩协议

