

x-Eris 调研报告

小组成员：胡天羽，罗胤玻，李润时，万辰希，吴书让

2023 年 3 月 30 日

1 项目概述

2 项目背景

2.1 嵌入式操作系统

2.1.1 概述

2.1.2 小结

2.2 非虚拟文件系统

2.2.1 概述

2.2.2 小结

2.3 虚拟文件系统 - Linux

2.3.1 概述

2.3.2 小结

2.4 虚拟文件系统 - 嵌入式

2.4.1 概述

本篇报告主要介绍了 FreeRTOS-Plus-FAT 以及 RT-Thread 中的文件系统实现细节。

在早期的嵌入式系统中，需要存储的数据比较少，数据类型也比较单一，往往使用直接在存储设备中的指定地址写入数据的方法来存储数据。然而随着嵌入式设备功能的发展，需要存储的数据越来越多，也越来越复杂，这时仍使用旧方法来存储并管理数据就变得非常繁琐困难。因此需要新的数据管理方式来简化存储数据的组织形式，需要新的文件系统。

2.4.2 FreeRTOS-Plus—FAT

- 主要

- `ff_dir`: 用于访问文件夹中内容
- `ff_fat`: 用于访问 FAT 文件系统
- `ff_file`: 用于文件读写

- `ff_ioman`: 管理缓存和挂载读写对象（介质）
- `ff_format`: 格式化或分区介质
- `ff_locking`: 加锁?
- `ff_memory`: 从内存读取数据
- `ff_stdio`: 用于文件管理（统计），相对路径转换
- `ff_sys`: 用于映射文件系统到根目录
- 辅助
 - `ff_headers`: 管理所有头文件
 - `ff_time`: 获取时间
 - `ff_error`: 用于错误处理
 - `ff_crc`: 用于计算 CRC（循环检验码）
 - `ff_string`: 字符串库
- 驱动
 - 各处理器相应的文件系统驱动

FreeRTOS-Plus-FAT 文件系统的标准 API 与标准 C 库使用相同的 `errno` 值。

标准 C 库中的文件相关函数返回 0 表示通过，返回 -1 则表示失败。如果返回 -1，则失败的原因存储在名为 `errno` 的变量中，须单独检查。同样，FreeRTOS-Plus-FAT 的标准 API 返回 0 表示通过，返回 -1 则表示失败，该 API 还会针对各项 RTOS 任务维护 `errno` 变量。

2.4.3 RT-Thread

2.4.4 小结

2.5 文件系统开发

2.5.1 概述

2.5.2 小结

3 立项依据

4 前瞻性/重要性分析

4.1 嵌入式文件系统的必要性

4.2 ...

5 相关工作

6 参考资料

1.