

# 3.1 ls 命令设计与实现 (一)\_物联网 / 嵌入式工程师 - 慕课网

“ 慕课网慕课教程 3.1 ls 命令设计与实现 (一) 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

- 编程实现 ls -l 命令功能，具体要求如下：
  - 显示文件类型
  - 显示文件权限
  - 显示文件所属用户名
  - 显示文件所属用户组名
  - 显示文件最后一次修改时间
  - 显示文件名
  -
- 系统 ls -l 实现的效果

文件类型	文件权限	硬链接	文件所属用户	文件所属用户组	文件大小	文件最后修改时间	文件名
-	rw-rw-r--	1	ben	ben	573	Oct 6 00:00	demo_dir.c

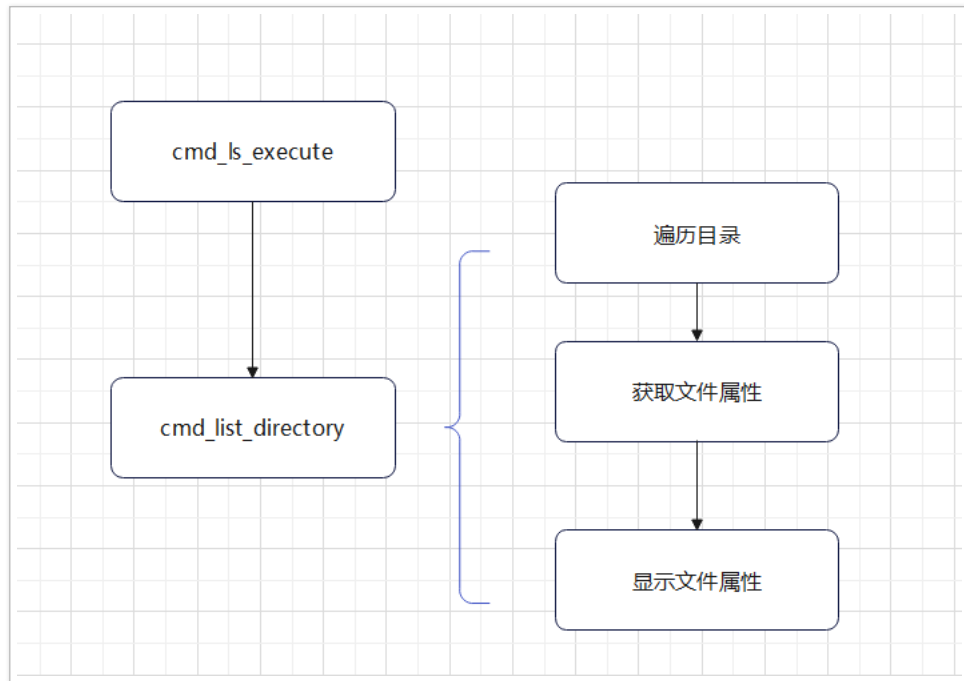
- 
- - step 1：遍历目录，并获取目录下的文件名
  - step 2：通过系统 stat 函数来获取每个文件的属性
  - step 3：解析文件属性信息，进行显示
- 目的：保存获取的文件属性信息，并进行输出
- 设计方式：在系统的 struct stat 结构体上进行扩展，产生新的结构体，具体设计如下：

```
#define SZ_LS_NAME 64
#define SZ_LS_PERMISSION 10
#define SZ_LS_TIME 32
#define SZ_LS_LINK_CONTENT 64

#define DEBUG

struct file_attribute
{
    struct stat f_attr_stat_info;

    char f_attr_type;
    char f_attr_uname[SZ_LS_NAME];
    char f_attr_gname[SZ_LS_NAME];
    char f_attr_mtime[SZ_LS_TIME];
    char f_attr_permission[SZ_LS_PERMISSION];
    char f_attr_name[SZ_LS_NAME];
    char f_attr_link_content[SZ_LS_LINK_CONTENT];
};
```



```

int cmd_ls_execute(cmd_t *pcmd)
{
    int ret;
    if (pcmd == NULL)
        return -1;

    if (pcmd->cmd_arg_count != 2){
        fprintf(stderr, "Command argument Error.\n");
        return -1;
    }

    if (pcmd->cmd_arg_list[1] != NULL)
        return cmd_list_directory(pcmd->cmd_arg_list[1]);
    else
        return -1;
}

```

- ls 命令主要业务逻辑是在 cmd\_list\_directory 函数中

- 下面实现了基本的 目录遍历

```

int cmd_list_directory(const char *dirpath)
{
    DIR *pdir = NULL;
    struct dirent *pdirent = NULL;

    char path[128] = {0};

    pdir = opendir(dirpath);
    if(pdir == NULL)
    {
        perror("open(): ");
        return -1;
    }

    while((pdirent = readdir(pdir)) != NULL)
    {
        if(strcmp(pdirent->d_name, ".") == 0 || strcmp(pdirent->d_name, "..") =
            continue;

    }

    closedir(pdir);
}

```

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

