



# **RTOS SMHC 开发指南**

**版本号: 1.1**  
**发布日期: 2021.4.15**

## 版本历史

版本号	日期	制/修订人	内容描述
1.1	2021.4.15	Allwinner	1. 初版



# 目 录

<b>1 前言</b>	<b>1</b>
1.1 编写目的 . . . . .	1
1.2 适用范围 . . . . .	1
1.3 相关人员 . . . . .	1
1.4 相关术语 . . . . .	1
<b>2 模块介绍</b>	<b>3</b>
2.1 模块功能介绍 . . . . .	3
2.2 接口 API 定义 . . . . .	4
2.3 关键变量声明 . . . . .	5



# 1 前言

## 1.1 编写目的

介绍 melis 内核中 SD/MMC 子系统，为 SD/MMC 设备驱动的开发提供参考。

## 1.2 适用范围

本模块说明适用于 v833, F133 平台。

## 1.3 相关人员

SD/MMC 驱动的开发/维护人员。

## 1.4 相关术语

- 地址空间的定义：
- 映射的主机地址空间：MMC 设备可通过来自主机软件的读命令访问的区域
- 私有的厂商专有地址空间：MMC 设备不能通过来自主机软件的读命令访问的区域。它容纳厂商专有的内部管理数据。
- 此数据可在声场时下载或在设备工作时产生的。例如存储厂商固件和映射表。它不容纳任何主机发送至设备的数据。
- 未映射的主机地址空间：MMC 设备不能通过来自主机软件的读命令访问的区域。它不包括厂商专有地址空间。他可以容纳
- 任何主机或主机数据的副本。
- 块 (block)：一些字节，基本数据传输单元
- CID (device identification register)：设备识别寄存器
- CLK(clock signal):
- CMD(command line or EMMC 不是 command):
- CRC(cyclic Redundancy Check)
- CSD(device specific Data regis)
- DAT(Data line):

- Data Strobe(return clock signal used in HS400 mode)
- DISCARD:
- DDR (Dual data rate)
- DSR (Driver Stage Register)
- D-Vdd (positive supply voltage for a cache memory) :
- D-Vssq(positive supply voltage ground for a cache memory)
- MMC(embedded multiMediaCard)
- mpty Task Queue :
- ERASE: 快擦除操作, 不需要实际的物理 NAND 擦除操作
- Flash: 一种可多次变成非易失性存储器
- Group: 一些写块, 组成擦除和写保护单元
- HS200: 在 1.8V 或 1.2V IO 200 MHz 单倍数据率总线上高达 200MB/s 高速接口时序
- ISI: 码间干扰 (指某种噪声类型)
- LOW,HIGH: 定义电平分配的二进制接口状态
- NSAC: 定义数据访问时间的时钟频率依赖系数最坏的情况
- Non-persistent: 存储设备上电后可能丢失内容的部分
- MSB, LSB: 最高位或最低位
- OCR: 工作条件寄存器
- open-drain: 一种漏极接口工作模式。用一个外接电阻器或电流源将接口电平拉倒 HIGH, 内部电阻器将其推向 LOW.
- payload: 纯数据
- push-pull: 一种逻辑接口工作模式, 用一个互补的晶体管对将接口电平推到 HIGH 或 LOW
- RCA: 相对设备地址寄存器
- ROM: 只读存储器
- RPMB: 重发保护存储砌块
- SSO: 同步开关 (指某种草绳类型)
- stuff bit: 填入比特 0 以保证命令和应答的固定长度
- TAAC: 敌营数据访问时间以来洗漱
- TRIM: 一个从写分组一处数据的命令。当 TRIM 被执行时, 区域应读为 0
- write Protection ,permanent : 写和擦除保护方案, 一旦是能, 就不能回复
- write protection ,Power-on : 写和擦除保护方案, 一旦使能, 只能在引起设备重启的掉电时间或
- 使用复位引脚复位时回复
- write protectio, Temporary: 可以使能和回复的写和擦除保护方案

## 2 模块介绍

### 2.1 模块功能介绍

melis 提供了 MMC 子系统来实现对各种 SD/SDIO 设备访问，MMC 子系统由上到下可以分为三层，MMC/SD card 层，MMC/SD core 层以及 MMC/SD host 层，它们之间的层次关系如下所示。

MMC/SD card 层负主要是按照 melis 块设备驱动程序的框架实现一个卡的块设备驱动。负责块设备请求的处理，以及请求队列的管理。MMC/SD core 层负责通信协议的处理，包括 SD/SDIO，为上一层提供具体读写接口，同时为下一层提供 host 端接口。MMC/SD host 是实现 SD/MMC 控制器相关的操作，直接操作硬件，也是主要实现部分。



图 2-1: audio frame

Sunxi 平台 MELIS BSP SDIO 驱动设计的驱动源代码文件之间的依赖关系如下：

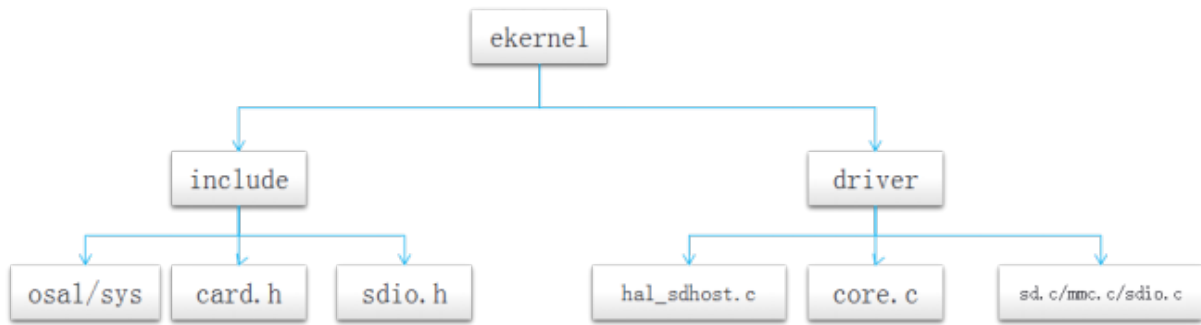


图 2-2: audio frame

Include: 对应 ekernel 下 driver/include/hal/sdmmc Driver: 对应 ekernel 下 driver/hal/-source/sdmmc 各文件主要作用: Include: osal/sys: 是对系统相关的函数的进一步封装; Card.h: struct mmc\_card 及相关结构的声明; Sdio.h: sdio 功能接口的声明; Hal\_sdhost.c: sdmmc 的驱动层实现, 主要用于底层操作寄存器实现相应的功能 core.c: 底层 host 驱动与卡设备协议的对接层 Sd.c/mmc.c/sdio.c: 相应给的 mmc 协议的封装。驱动整体框架如下图所示:



图 2-3: audio frame

注: host 对应 hal\_sdhost.c, 进行 host 控制器操作的封装 Bus 对应 core.c, 作为 bus 匹配底层 host 和相应 mmc 协议的匹配。mmc/sd/sdio 分别对应 mmc.c/sd.c/sdio.c 等文件进行了协议的封装。

## 2.2 接口 API 定义

```

uint8_t sdio_readb(struct mmc_card *card, uint32_t func_num, uint32_t addr, int32_t *err_ret);
void sdio_writeb(struct mmc_card *card, uint32_t func_num, const uint8_t b, uint32_t addr, int32_t *err_ret);
int32_t sdio_enable_func(struct mmc_card *card, uint32_t func_num);
int32_t sdio_disable_func(struct mmc_card *card, uint32_t func_num);
  
```

```
int32_t sdio_set_block_size(struct mmc_card *card, uint32_t fn_num, uint32_t blksz);
int sdio_memcpy_fromio(struct mmc_card *card, unsigned int func_num, void *dst, unsigned
    int addr, int count);
int sdio_memcpy_toio(struct mmc_card *card, unsigned int func_num, unsigned int addr, const
    void *src, int count);
void sdio_claim_host(struct mmc_card *card);
void sdio_release_host(struct mmc_card *card);
int sdio_release_irq(struct sdio_func *func);
uint16_t sdio_readw(struct sdio_func *func, unsigned int addr, int *err_ret);
uint32_t sdio_readl(struct sdio_func *func, unsigned int addr, int *err_ret);
void sdio_writew(struct sdio_func *func, uint16_t b, unsigned int addr, int *err_ret);
void sdio_writel(struct sdio_func *func, uint32_t b, unsigned int addr, int *err_ret);
struct sdio_func ** get_mmc_card_func(uint8_t card_id);
```

## 2.3 关键变量声明

/\*\* 定义了 sdc host log 打印等级, 检测方式, dma 使用使能. / SDC\_InitTypeDef  
sdc\_param, /\*\* 定义了 sdc card log 打印等级, sdio 卡类型 \*/ SDCard\_InitTypeDef  
card\_param;

/\*\* 定义了 sdc card 相关的资源, sdio 卡 cid, ocr 等信息, 及相应资源/ Struct mmc\_card  
card;

/\*\* 定义了 sdc host 相关的资源, host pinctrl 寄出去基地址等信息, 及相应资源/ Struct  
mmc\_host host;



## 著作权声明

版权所有 © 2021 珠海全志科技股份有限公司。保留一切权利。

本文档及内容受著作权法保护，其著作权由珠海全志科技股份有限公司（“全志”）拥有并保留一切权利。

本文档是全志的原创作品和版权财产，未经全志书面许可，任何单位和个人不得擅自摘抄、复制、修改、发表或传播本文档内容的部分或全部，且不得以任何形式传播。

## 商标声明

、 全志科技、（不完全列举）均为珠海全志科技股份有限公司的商标或者注册商标。在本文档描述的产品中出现的其它商标，产品名称，和服务名称，均由其各自所有人拥有。

## 免责声明

您购买的产品、服务或特性应受您与珠海全志科技股份有限公司（“全志”）之间签署的商业合同和条款的约束。本文档中描述的全部或部分产品、服务或特性可能不在您所购买或使用的范围内。使用前请认真阅读合同条款和相关说明，并严格遵循本文档的使用说明。您将自行承担任何不当使用行为（包括但不限于如超压，超频，超温使用）造成的不利后果，全志概不负责。

本文档作为使用指导仅供参考。由于产品版本升级或其他原因，本文档内容有可能修改，如有变更，恕不另行通知。全志尽全力在本文档中提供准确的信息，但并不确保内容完全没有错误，因使用本文档而发生损害（包括但不限于间接的、偶然的、特殊的损失）或发生侵犯第三方权利事件，全志概不负责。本文档中的所有陈述、信息和建议并不构成任何明示或暗示的保证或承诺。

本文档未以明示或暗示或其他方式授予全志的任何专利或知识产权。在您实施方案或使用产品的过程中，可能需要获得第三方的权利许可。请您自行向第三方权利人获取相关的许可。全志不承担也不代为支付任何关于获取第三方许可的许可费或版税（专利税）。全志不对您所使用的第三方许可技术做出任何保证、赔偿或承担其他义务。