scsemicon.com

COS存储结构设计

yuxi.sun

修订历史信息

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 版本号 | 发布日期 | 发布者 | 审核人 | 批准人 | 变更内容描述（与前一版本的差别） |
| V0.1 | 2020.02.22 | 孙玉玺 |  |  | 增加文件结构和函数接口 |
| V0.2 | 2020.02.24 | 孙玉玺 |  |  | 增加写入和擦除流程 |
| V0.3 | 2020.02.25 | 孙玉玺 |  |  | 增加写入和擦除异常处理流程 |
| V0.4 | 2020.04.27 | 孙玉玺 |  |  | 完整整个文档 |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |

目录

[物理结构 4](#_Toc38898896)

[LA -> PA 4](#_Toc38898897)

[MAP 5](#_Toc38898898)

[PA 5](#_Toc38898899)

[函数说明 6](#_Toc38898900)

[写入流程 9](#_Toc38898901)

[测点，异常情况总结 10](#_Toc38898902)

[擦除流程 11](#_Toc38898903)

[擦除异常流程： 12](#_Toc38898904)

[初始化流程 13](#_Toc38898905)

# 物理结构

|  |  |
| --- | --- |
| LA | 逻辑编号，每次写入文件都会以索引为表示可用编号0 ~ 509  510 为记录编号  511为CRC |
| PA | 物理地址，实际存储的物理地址，存储文件内容 |
| BLOCK | norflash真正擦除的范围，4K bytes |
| PAGE | norflash物理地址，1 page = 1 pa ，每个page 256 bytes |
| WORD | norflash写入的最小单位，按照word计数，4 bytes |

# LA -> PA

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| INDEX | 0 | 1 | 2 | … | n-3 | n-2 | n-1 |
|  | ↓ ↓ ↓ ↓ | | | | | | |
| LA | xxxx\_bbbb | xxxx\_bbbb | xxxx\_bbbb | xxxx\_bbbb | xxxx\_bbbb | cnt | crc32 |

LA：0 ~ 3 BLOCK

xxxx：物理块号（0 ~ 383）

bbbb：文件编号 0 ~ 0x0fff，可用id

0xffff:映射，无实际意义

正常写入完成

|  |  |  |
| --- | --- | --- |
| LA 更新 | mapping | crc |
| LA 1 | 0 ~ 383 | CRC32 valid |
| LA 2 | 0 ~ 383 | CRC32 valid |
| LA 3 | 0 ~ 383 | CRC32 valid |
| … | 0 ~ 383 | CRC32 valid |
| LA n-2 | 0 ~ 383 | CRC32 valid |
| LA n-1 | 0 ~ 383 | CRC32 valid |
|  |  |  |

异常写入完成

|  |  |  |
| --- | --- | --- |
| LA 更新 | mapping | crc |
| LA 1 | 0 ~ 383 | CRC32 valid |
| LA 2 | 0 ~ 383 | CRC32 valid |
| LA 3 | 0 ~ 383 | CRC32 valid |
| … |  | CRC32 valid |
| LA n-2 | 0 ~ 383 | CRC32 valid |
| LA n-1 | 0 ~ 383 | CRC32 invalid |
|  |  |  |

# MAP

mul mapping , data more than 128 bytes

LA

PA

LA

PA

LA

PA

LA

PA

# PA

page formate

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| offset | 0 | 4 | 8 | 12 | 16 | 20~ 47 |
| 说明 | crc | level | la | next la | len | reverse |
| offset | 48 ~ 255 | | | | |  |
| 说明 | data | | | | |  |

crc : data段的crc32

level : 等级，分为MF:0x0000000 DF:0x1000xxxx EF: 0x1100xxxx

la : 当前pa指向的la

next la : 下一个有效的la，如果没有则为NULL

reverse : 保留位

# 函数说明

|  |  |
| --- | --- |
|  | |
| 函数 | void fs\_init(void) |
| 输入 | 无 |
| 输出 | 无 |
| 说明 | 初始化文件系统 |
|  | |
| 函数 | UINT32 fs\_trim() |
| 输入 | 无 |
| 输出 | 无 |
| 说明 | 垃圾回收 |
|  | |
| 函数 | UINT32 fs\_clear\_all() |
| 输入 | 无 |
| 输出 | 无 |
| 说明 | 清除文件系统 |
|  | |
| 函数 | UINT32 fs\_get\_free\_capacity（） |
| 输入 | 无 |
| 输出 | 无 |
| 说明 | 获取剩余的可用块，每个块208字节 |
|  | |
| 函数 | UINT32 fs\_check\_id(UINT16 id) |
| 输入 | id:待检测id |
| 输出 | FS\_SUCCESS: id存在  FS\_ERROR: id不存在 |
| 说明 | 检测某个ID是否存在 |
|  | |
| 函数 | UINT32 fs\_get\_id\_len(UINT32 id) |
| 输入 | id:待检测id |
| 输出 | ！= 0: 该id长度  0:该id不存在 |
| 说明 | 获取一个或多个可用的pa |
|  | |
| 函数 | UINT32 fs\_delete\_data(UINT16 id) |
| 输入 | id:待检测id |
| 输出 | FS\_ERROR : 该id不存在，删除失败  FS\_SUCCESS : 该id存在，成功删除 |
| 说明 | 判断id是否合存在 |
|  | |
| 函数 | UINT32 fs\_read\_data(UINT16 id , UINT8 \*data , UINT32 offset , UINT32 len) |
| 输入 | id : id编号  \*data : 输入数据缓存  offset: 输入数据偏移  len : 数据长度 |
| 输出 | TRUE : 写入成功  FALSE : 写入失败 |
| 说明 | 写入指定id起始从offset ~ len的数据内容 |
|  | |
| 函数 | UINT32 fs\_write\_data(UINT16 id , UINT8 \*data , UINT32 offset , UINT32 len) |
| 输入 | id : id编号  \*data : 输出数据缓存  offset: 输出数据偏移  len : 数据长度 |
| 输出 | TRUE : 读取成功  FALSE : 读取失败 |
| 说明 | 读取指定id起始从offset ~ len的数据内容 |
|  | |

# 写入流程



## 测点，异常情况总结



# 擦除流程



## 擦除异常流程：



# 初始化流程

