

PB375 U 盘读写模块

用户手册

深圳第五元素科技有限公司

版权所有 不得复制

手册目录

1. 概述.....3

2. 功能与特点.....3

3. 接口操作.....3

 3.1 接口概述.....3

 3.2 硬件接口.....3

 3.3 数据包及返回格式.....3

4. 模块命令及参数.....4

 4.1 操作命令参数.....4

 4.2 返回的操作状态码.....4

 4.3 预定义数据结构.....4

5. 常用操作步骤.....5

 5.1 初始化.....5

 5.2 顺序读文件.....5

 5.3 读文件的指定位置.....6

 5.4 顺序改写文件.....6

 5.5 改写文件的指定位置.....6

 5.6 向已有文件追加数据.....6

 5.7 新建文件.....6

 5.8 先读文件再改写文件.....6

 5.9 先写文件再读文件复查.....6

 5.10 文件追加数据.....7

 5.11 定期采集数据.....7

6. 其他说明.....7

7. 电气参数.....8

 7.1 极限参数.....8

 7.2 电气参数.....8

 7.3. 时序参数.....8

8. 机械安装.....9

9. 技术支持.....9

| 版本号 | 修改时间 | 更改内容 |
|-----|------------|------|
| 1.0 | 2008.12.06 | 初始发布 |

1、概述

当前信息化社会，U 盘（含闪盘、USB 闪存盘、USB 移动硬盘等）已经成为很常用的移动存储设备，用来随身携带存储数据。所以在很多产品中需要读取或者保存数据，多数采用U 盘的方式。单片机系统可以直接采用U 盘作为移动存储器，并且方便与使用WINDOWS 操作系统的计算机交换数据。

2、功能与特点

用于嵌入式系统/单片机读写U 盘、闪盘、闪存盘、USB 移动硬盘、USB 读卡器等。
支持文件系统FAT12 和FAT16 及FAT32，如果需要支持FAT32 请看本文后面的说明。
文件操作功能：新建文件、删除文件、打开文件、关闭文件、读数据、写数据等。
读写模式：方便的字节模式。
采用方便、高速、简单的SPI接口

3、接口操作

3.1 接口概述

基本操作步骤是，单片机系统将命令码、后续参数长度（因为各命令码所需要的参数不等长）和参数写给模块，并通知其启动操作，模块执行完成后以中断方式通知单片机，并返回操作状态和操作结果。注意，数据流模式的命令执行完成后不返回状态。
因为接口操作看起来比较复杂，所以实际过程可以参考随模块一起提供的几个示例程序，直接用

3.2 硬件接口

模块电接口，采用SPI通信方式，定义如下

CS-----SPI协议使能信号

SCK-----SPI时钟信号

SIN-----数据输入

SOUT----数据输出

3.3V----外接3.3V电源（若使用板上1117将内部5V转3.3V, 则可以不接）

5V-----V电源

GND-----地

3.3 数据包及返回格式

数据包结构

| | | |
|-----|------|-----|
| 操作码 | 数据长度 | 数据包 |
|-----|------|-----|

返回数据包

| | | |
|-----|--------|------|
| 状态码 | 返回数据长度 | 返回数据 |
|-----|--------|------|

4、模块命令

4.1 操作命令参数

文件操作命令总表

| 操作名 | 操作码 | 参数 | 功能 | 返回 |
|----------------|------|------------|--------------------------|-----------|
| CMD_GetVer | 0x0A | 无 | 获取当前模块的版本号 | 模块版本号 |
| CMD_DiskReady | 0x71 | 无 | 查询 U 盘是否准备就绪, 通常在就绪后才能读写 | 返回成功则表示就绪 |
| CMD_FileOpen | 0x64 | 文件路径 | 打开指定名称的文件 | 操作成功否 |
| CMD_FileCreate | 0x65 | 文件名称 | 新建文件并打开, 如果文件已存在则先删除再新建 | 操作成功否 |
| CMD_FileErase | 0x66 | 无 | 删除文件并关闭 | 操作成功否 |
| CMD_FileClose | 0x67 | 无 | 关闭当前文件 | 操作成功否 |
| CMD_ByteLocate | 0x7A | 字节偏移位置 | 以字节为单位移动当前文件指针 | 成功+移动字节数 |
| CMD_ByteRead | 0x7B | 字节长度 | 以字节为单位从当前文件读取数据块 | 成功+读取字节包 |
| CMD_ByteWrite | 0x7C | 字节长度+字节缓冲区 | 以字节为单位向当前文件写入数据块 | 成功+写入字节包 |

4.2 返回的操作状态码

```
#define ERR_USB_CONNECT      0x15    /* 检测到 USB 设备已经连接
#define ERR_DISK_DISCON     0x82    /* 检测到 USB 设备已断开, 或尚未连接
#define ERR_SUCCESS         0x00    /* 操作成功 */
#define ERR_STATUS_ERR      0x83    /* 磁盘状态错误, 可能正连接或断开磁盘 */
#define ERR_FAT_ERROR       0xA3    /* 磁盘的文件系统不支持
#define ERR_DISK_FULL       0xB1    /* 磁盘文件太满, 剩余空间太少
#define ERR_MISS_DIR        0xB3    /* 指定路径的某个子目录没找到#define
#define ERR_FILE_CLOSE      0xB4    /* 文件已关闭, 如需要使用, 重新打开文件
#define ERR_OPEN_DIR        0x41    /* 指定路径的目录被打开 */
#define ERR_MISS_FILE       0x42    /* 指定路径文件没找到, 可能是文件名错误
#define ERR_USB_DISK_ERR    0x1F    /* USB 存储器操作失败
```

4.3 预定义数据结构

获取模块版本

```
struct {
    unsigned char FileLibVer; /* 返回: 子程序库的版本号 */
    unsigned char ModuleVer; /* 返回: 模块的版本号 */
} GetVer; /* CMD_GetVer, 获取当前模块的版本号 */
```

打开文件

```
struct {
    unsigned char mPathName[ MAX_PATH_LEN ]; /* 输入参数: 路径/目录名/文件名 */
```

```
} Open; /* CMD_FileOpen, 打开文件 */
```

创建文件

```
struct {  
    unsigned char mPathName[ MAX_PATH_LEN ]; /* 输入参数：路径/目录名/文件名 */  
} Create; /* CMD_FileCreate, 新建文件并打开 文件长度为8+3的格式，如  
NEWFILE1.TXT*/
```

关闭文件

```
struct {  
    unsigned char mUpdateLen; /* 输入参数：是否允许更新长度：0 禁止, 1 允许 */  
} Close; /* CMD_FileClose, 关闭当前文件 无论是否输入，都是更新长度*/
```

字节定位

```
struct {  
    unsigned long mByteOffset; /* 输入参数：以字节为单位的偏移量，以字节为单位的文件指针，返回：当前文件指针字节数，0FFFFFFFH 则已到文件尾 */  
} ByteLocate; /* CMD_ByteLocate, 以字节为单位移动当前文件指针 */
```

读取字节

```
struct {  
    unsigned char mByteCount; /* 输入参数：准备读取的字节数, 不得大于MAX_BYTE_IO,  
    返回：实际读出的字节数 */  
    unsigned char mByteBuffer[ MAX_BYTE_IO ]; /* 返回：读出的数据块 */  
} ByteRead; /* CMD_ByteRead, 以字节为单位从文件读取数据块 */
```

写入字节

```
struct {  
    unsigned char mByteCount; /* 输入参数：准备写入的字节数, 不得大于MAX_BYTE_IO,  
    返回：实际写入的字节数 */  
    unsigned char mByteBuffer[ MAX_BYTE_IO ]; /* 输入参数：准备写入的数据块 */  
} ByteWrite; /* CMD_ByteWrite, 以字节为单位向文件写入数据块 */
```

5、常用操作步骤

以下是常用步骤简述，可以根据实际情况进行调整

5.1 初始化

- (1) 初始化单片机与模块之间的接口，设置必要的参数等
- (2) 等待U 盘连接
- (3) CMD_DiskReady，可选步骤，强烈建议执行此步骤

5.2 顺序读文件

- (1) CMD_FileOpen，打开文件

- (2) 多次CMD_ByteRead, 读取数据
- (3) CMD_FileClose, 关闭文件

5.3 读文件的指定位置

- (1) CMD_FileOpen, 打开文件
- (2) CMD_ByteLocate, 移动文件指针到指定位置
- (3) CMD_ByteRead, 读取数据, 期间还可以移动文件指针
- (4) CMD_FileClose, 关闭文件

5.4 顺序改写文件 (覆盖原数据, 超过原文件长度后转变为追加数据)

- (1) CMD_FileOpen, 打开文件
- (2) CMD_ByteWrite, 写入数据
- (3) CMD_FileClose, 关闭文件

5.5 改写文件的指定位置 (覆盖原数据, 超过原文件长度后转变为追加数据)

- (1) CMD_FileOpen, 打开文件
- (2) CMD_ByteLocate, 移动文件指针到指定位置
- (3) CMD_ByteWrite, 写入数据, 期间还可以移动文件指针
- (4) CMD_FileClose, 关闭文件

5.6 向已有文件追加数据

- (1) CMD_FileOpen, 打开文件
- (2) CMD_ByteLocate, 移动文件指针到末尾, 0xFFFFFFFF
- (3) CMD_ByteWrite, 写入数据
- (4) CMD_FileClose, 关闭文件

5.7 新建文件

- (1) CMD_FileCreate, 新建文件, 文件名称为8+3的格式
- (2) CMD_ByteWrite, 写入数据
- (3) CMD_FileClose, 关闭文件, 如果是字节模式, 将自动更新文件长度

5.8 先读文件再改写文件

- (1) CMD_FileOpen, 打开文件
- (2) CMD_ByteRead, 读取数据
- (3) CMD_ByteLocate, 移动文件指针到头部, 00000000
- (4) CMD_ByteWrite, 写入数据
- (5) CMD_FileClose, 关闭文件, 如果是字节模式, 将自动更新文件长度

5.9 先写文件再读文件复查

- (1) CMD_FileOpen, 打开文件
- (2) CMD_ByteWrite, 写入数据
- (3) CMD_ByteLocate, 移动文件指针到头部, 00000000
- (4) CMD_ByteRead, 读取数据, 复查
- (5) CMD_FileClose, 关闭文件, 如果是字节模式, 将自动更新文件长度

5.10 文件已经存在则追加数据，如果文件不存在则新建文件再写入数据

- (1) CMD_FileOpen, 打开文件, 如果返回ERR_MISS_FILE 文件不存在, 则转到步骤(4)
- (2) CMD_ByteLocate, 移动文件指针到末尾, 0xFFFFFFFF
- (3) 转到步骤(5), 准备追加数据
- (4) CMD_FileCreate, 新建文件, 准备写入数据
- (5) CMD_ByteWrite, 写入数据
- (6) CMD_FileClose, 关闭文件

5.11 定期采集数据 (适用于数据量较小的情况)

- (1) 采集之前, CMD_FileCreate, 新建文件
- (2) 采集数据, 转换为相应的格式, 例如二进制数据、字符串等
- (3) CMD_ByteWrite, 写入数据, 一次写不完, 可以分多次写入
- (4) 如果要等很长时间才有下一组数据, 为了避免在此期间发生断电、U 盘拔出等事件, 导致文件长度不正确, 可以用CMD_ByteWrite 写入空数据, 强制更新文件长度
- (5) 如果整个采集过程结束, 或者文件已经太大, 那么转到步骤(6), 否则转到步骤(2)
- (6) CMD_FileClose, 关闭文件, 自动更新文件长度
- (7) 如果是因为文件已经太大的原因, 那么转到步骤(1), 新建另一个文件继续

6、其它说明

(1) 为了提高处理效率和速度, 建议对USB 闪存盘使用FAT12 或者FAT16 文件系统。对于容量较大的U 盘, 为了节约U 盘空间, 减少浪费, 可以使用FAT32 文件系统。

(2) 如何格式化U盘: 在WINDOWS系统下, 点击U盘磁盘的右键菜单的格式化, 可以将U 盘格式化成为指定的FAT12、FAT16或者FAT32 文件系统, 当总容量除以分配单元大小后的结果小于4085时是FAT12, 大于65525 时是FAT32, 否则是FAT16。分配单元较大时, 通常读写效率稍高, 分配单元较小时, 通常会节约磁盘容量。

(3) 如果操作USB 外置硬盘或者耗电较大的USB 闪存盘, 需要考虑其电源供应, 确保提供足够的工作电流, 否则在其插入过程以及读写过程中会导致电源电压波动, 甚至导致CH375 模块以及单片机复位, 建议在电源与地之间并联较大的电解电容, 或者为U 盘或者USB 外置硬盘单独提供足够的电源。

(4) 以字节为单位的文件读写命令, 优点是对单片机的RAM 没有要求, 最少十几个字节即可, 缺点是速度比以扇区为单位的文件读写慢, 并且频繁地向U 盘中的文件写数据, 会缩短U 盘中闪存的使用寿命 (因为闪存只能进行有限次擦写)。

(5) 虽然U 盘模块最大支持1GB 的文件, 但是为了提高效率, 建议单个文件的长度不要超过100MB, U 盘文件读写模块的说明 2.1 通常在几KB 到几MB 范围是比较正常的。如果文件较多, 建议使用多级目录结构分类管理, 创建多个子目录, 将具有部分相似性质的多个文件放在同一个子目录下, 采用目录分类管理可以缩短打开、枚举搜索或创建文件的时间。

7、电气参数

7.1 极限参数

| 参 数 | 符 号 | 参 数 范 围 |
|------|-------|-----------------|
| 工作电压 | VDD33 | 2.9~3.6 |
| 输入电压 | VI | -0.3~VDD33 +0.3 |
| 输出电压 | VO | -0.3~VDD33 +0.3 |
| 工作温度 | Tamb | -20~75 |
| 存储温度 | Tstg | -65~150 |

7.2 电气参数 (Tamb=25° C, VDD33=3.3V, VDD18=1.8V, OSC=24MHz)

| 参 数 | 符 号 | 测 试 条 件 | 最小值 | 典型值 | 最大值 |
|---------|--------------|---------------|------------|-----|------------|
| 电源电压 | VDD33 | 工作正常 | 2.9 | 3.3 | 3.6 |
| 静态电流 | ICCS | 系统处于待机状态 | 25 | 35 | 70 |
| | IDDS | 系统处于待机状态 | 50 | 110 | 350 |
| 工作电流 | ICC 正常工作, | MCU C1k=24MHz | 9 | 16 | |
| 高电平输入电压 | VIH | -- | 0.7* VDD33 | -- | VDD33+0.6 |
| 低电平输入电压 | VIL | -- | -0.3 | -- | 0.4* VDD33 |
| 高电平输出电压 | VOH | IOH=2mA | 2.4 | -- | -- |
| 低电平输出电压 | VOL | IOL=2mA | -- | -- | 0.4 |
| 输入漏电流 | ILI | VDD33=3.6V | -10 | -- | 10 |
| 输出漏电流 | ILO | VDD33=3.6V | -5 | -- | 5 |

7.3. 时序参数 (测试条件: TA=25°C, VCC=5V)

| 名称 | 参数说明 | 最小值 | 典型值 | 最大值 | 单位 |
|-----|-----------|-----|-----|-----|----|
| TPR | 电源上电的复位时间 | 20 | 100 | 200 | mS |
| | | | | | |

8、机械安装

模块尺寸为: 38mm*38mm

深圳第五元素科技

<http://www.chinamcu.com>

安装孔位置：安装孔1（16mm, 2mm） 安装孔2（36mm, 36mm） 安装孔半径为1.6mm

9、技术支持

公司名称：深圳第五元素科技有限责任公司；
地址：深圳市南山科技园龙泰利科技大厦306室；
电话：0755-86017852
传真：0755-29739852
邮箱：xiaowuyeah@163.com