铁电模块接口说明

铁电存储模块使用芯片FM24CL64B，使用I2C接口和MUC进行通信，一页大小为8个字节，共有8192页，总大小为64KB。地址寻址范围为0000h~1FFFh。

底层接口有如下几个

（1）void I2C\_Fram\_Init(void)

主要对I2C接口进行初始化操作。改函数在Driver\_Init()函数中进行调用，完成对I2C接口的初始化工作。

（2）uint32\_t I2C\_Fram\_ByteWrite(uint8\_t \*pBuffer, uint16\_t WriteAddress);

该函数实现一个字节的写入。

返回值类型

0正确写入

1 写入错误

2 写入忙

3 写入超时（设置超时时间为100ms）

参数pBuffer为要写入数据的起始地址，WriteAddress为写入铁电的起始存储地址，取值范围为0000h~1FFFh。该函数一次写入一个字节。

（3）uint32\_t I2C\_Fram\_PageWrite(uint8\_t \*pBuffer, uint16\_t WriteAddress, uint8\_t NumByteToWrite);

该函数实现一页数据的写入。

返回值类型

0正确写入

1 写入错误

2 写入忙

3 写入超时（设置超时时间为100ms）

参数pBuffer为要写入数据的起始地址，WriteAddress为写入铁电的起始存储地址，取值范围为0000h~1FFFh。NumByteWrite为写入的字节数，范围为1~8。

（4）uint32\_t I2C\_Fram\_BufferWrite(uint8\_t\* pBuffer, uint16\_t WriteAddr, uint16\_t NumByteToWrite);

该函数实现向铁电的一段存储区域中写入数据。

返回值类型

0正确写入

1 写入错误

2 写入忙

3写入超时（设置超时时间为100ms）

参数pBuffer为要写入数据的起始地址，WriteAddress为写入铁电的起始存储地址，取值范围为0000h~1FFFh。NumByteToWrite为要写入的字节数，取值范围为1~65536。

注意这三个写入函数不进行地址存取器地址校验和写入字节数的校验工作，因此调用该函数的人员自己确保入参的正确性。

（5）uint32\_t I2C\_Fram\_BufferRead(uint8\_t\* pBuffer, uint16\_t ReadAddr, uint16\_t NumByteToRead);

该函数实现从铁电的ReadAddr的起始地址连续读取NumByteToRead字节的数据到pBuffer指向的内存空间中。

返回值类型

0正确读取

1 读取错误

2 读取忙

3 读取超时（设置超时时间为100ms）

参数pBuffer存取要读取输出的起始地址，ReadAddr从该地址开始读取数据，取值范围为0000h~1FFFh，NumByteRead需要读取的数据量。

注意该函数不进行读取地址和读取数据的校验工作，需要要求调用该函数的人员确保入参的正确性。

应用层对数据存储思想是和威卡一致，可以实现512个8位空间，512个16位空间和512个32位空间的读写操作。每种类型的存储，底层程序都会在铁电中存2份（底层自动实现，不需要应用层调用2次存储函数），一份存储在主数据区，1份存储在备份数据区。每一个区的数据存储时，底层软件都会自动计算出CRC校验值，并将校验值也存储。在上电初始化阶段，对CRC进行校验。在MUC内存中同样开辟三段内存空间，分别为512个8位空间，512个16位空间和512个32位空间，这三段内存空间和铁电中的对应空间内容保持一致，及每次应用层调用数据保存接口时，首先更新这三段内存空间中的数据，更新完成后将需要写入的数据计算一个CRC值并将CRC值和数据一同写入铁电中，调用读取函数时直接存MCU内存中获取数据，不需要从铁电中读取，使用空间换时间的方法。具体函数接口如下。

（1）S16 API\_PAR\_SaveBytes(U16 Address\_u16, const U8 \*Buffer\_u8, U16 Len\_u16)

返回值类型

0 成功写入

1 写入失败

2 参数索引错误

3 写入数据指针错误

4 数据长度错误

5 主区备区都错（我这边设计的是只要有一个数据区写入错误就返回该错误值）

~~6 写保护中（没有该错误由于存储芯片写保护引脚始终拉低）~~

该函数作用是保存512字节的数据到铁电中，保存在铁电中的具体位置的物理地址已经在底层设计中确定。

参数含义：

Address\_u16 参数地址，取值范围（0-511）

Buffer\_u8：写入的数据地址

Len\_u16: 写入数据长度，确保 Address\_u16 + Len\_u16 < 512

（2）S16 API\_PAR\_SaveWords(U16 Address\_u16, const U16 \*Buffer\_pua16, U16 Len\_u16)

返回值类型

0 成功写入

1 写入失败

2 参数索引错误

3 写入数据指针错误

4 数据长度错误

5 主区备区都错（我这边设计的是只要有一个数据区写入错误就返回该错误值）

~~6 写保护中（没有该错误由于存储芯片写保护引脚始终拉低）~~

该函数作用是保存512个16bit数据到铁电中，保存在铁电中的具体位置的物理地址已经在底层设计中确定。

参数含义：

Address\_u16：参数地址，取值范围（0-511）

Buffer\_pau16：写入数据地址

Len\_u16：写入数据长度，确保Address\_u16 + Len\_u16 < 512

（3）S16 API\_PAR\_SaveReals(U16 Address\_u16，const REAL32 \*Buff\_par32, U16 Len\_u16)

返回值类型：

0 成功写入

1 写入失败

2 参数索引错误

3 写入数据指针错误

4 数据长度错误

5 主区备区都错（我这边设计的是只要有一个数据区写入错误就返回该错误值）

~~6 写保护中（没有该错误由于存储芯片写保护引脚始终拉低）~~

该函数的作用是实现保存512个32bit的数据到铁电中，保存在铁电中的具体位置的物理地址已经在底层设计中确定。

参数含义：

Address\_u16: 参数地址（0-511）

Buff\_par32: 写入的数据地址

Len\_u16：写入数据的长度，确保Address\_u16 + Len\_u16 < 512

（4）S16 API\_PAR\_GetBytes(U16 Address\_u16, U8 \*Buffer\_pau8, U16 Len\_u16);

返回值类型；

0：成功

1：失败

2：参数索引超限

3：缓冲区故障

4：长度超限

该函数作用实现对铁电中保存的512个8bit的数据的读取。根据输入的地址和长度，读取不同的位置块到变量中。

参数含义：

Address\_u16：参数地址（0-511）

Buff\_pau8：读取的数据存放变量指针

Len\_u16：读取数据长度，确保Address\_u16 + Len\_u16 < 512

（5）S16 API\_PAR\_GetWords(U16 Address\_u16, U16 \*Buff\_pau16, U16 Len\_u16)

返回值类型：

0：成功

1：失败

2：参数索引超限

3：缓冲区故障

4：长度超限

该函数能够实现对铁电中保存的512个16bit的数据的读取。根据输入的地址和长度，读取不同的位置块到变量中。

参数含义：

Address\_u16：参数地址（0-511）

Buff\_pau16：读取的数据存放变量指针

Len\_u16：读取数据长度，确保Address\_u16 + Len\_u16 < 512

（6）S16 API\_PAR\_GetReals(U16 Address\_u16, REAL32 \*Buffer\_par32, U16 Len\_u16)

返回值类型：

0：成功

1：失败

2：参数索引超限

3：缓冲区故障

4：长度超限

该函数能够实现对在铁电中保存的512个浮点的数据的读写。根据输入的地址和长度，读取不同的位置块到变量中。

参数含义：

Address\_u16：参数地址（0-511）

Buff\_pau32：读取的数据存放变量指针

Len\_u16：读取数据长度，确保Address\_u16 + Len\_u16 < 512

（7） U16 API\_Par\_Init(void)

返回值类型：

返回值不同的bit位表示不同区域的校验结果

Bit0:读取byte主数据区数据校验结果 0：正确；1：错误；

Bit1:读取byte备份区数据校验结果 0：正确；1：错误；

Bit2:读取word主数据区数据校验结果 0：正确；1：错误；

Bit3:读取word备份区数据校验结果 0：正确；1：错误；

Bit4:读取real主数据区数据校验结果 0：正确；1：错误；

Bit5:读取real备份区数据校验结果 0：正确；1：错误；

该函数用于上电初始化阶段对铁电存储参数进行校验，如果读出的值计算出来的CRC校验值和已经存储的CRC校验值一致，则返回0，认为正确，否则返回1，认为错误。

（8）U8 API\_PAR\_Data\_Init(U8 Par\_type, const U8\* Buff\_u8, U16 Len\_u16)

返回值类型：

0：成功

1：失败

3：写入数据指针错误

0Xff：参数类型错误

该函数用于上电阶段，当检测到数据发生异常时（CRC校验失败），对空间值进行写初始值。

参数含义：

Par\_type：参数类型；1：BYTE;2:WORD;4DWORD

Buff\_u8：写入的数据指针

Len\_u16：写入数据的长度，确保 Address\_u16 + Len\_u16 < 512

恒流控制模块接口说明

恒流控制模块使用芯片tle7242实现，每块芯片可以控制4路，电路板上使用了三块该芯片，因此可以实现12路电流控制。每一路可以实现两种工作模式，恒流模式和开关模式。在tle7242.h文件中对该芯片使用的13个寄存器进行了封装，每个寄存器的中的bit位的含义都用位域进行了说明，该芯片和MCU使用SPI接口进行通信，通信的接口都是使用函数uint32 TLE\_Register\_Operation\_Data（uint32 SPI\_Data\_u32）函数实现，其中参数SPI\_Data\_u32是MCU想写入tle7242芯片的数据，返回值为tle7242反馈给MCU的数据。每次都是32bit数据进行通信。Tle7242.c文件中函数是底层寄存器的读写函数，bsp\_output\_docc.c文件是对tle7242.c中函数的封装，是为应用层程序使用。下面bsp\_output\_docc.c文件中接口功能进行说明。

（1）void API\_PO\_Mode\_Config(uint8\_t chan\_u8, uint16\_t mode\_u8)

函数是配置某一路是使用恒流模式还是开关模式。在恒流模式下，配合函数API\_ConstantCurrent\_Drive()实现恒流输出，在开关模式下，配合函数API\_Power\_Switch\_Set()实现高低电平输出。

参数含义：

Chan\_u8:PO通道，0-11分别表示PO1-PO12，也可以使用enum的成员PO1-PO12

Mode\_u8:模式配置 0：恒流模式（默认模式）， 1：开关模式

（2）void API\_ConstantCurrent\_Drive(uint8\_t chan\_u8, uint16\_t current\_u16, uint16\_t freq\_u16, float kp\_f, float kp\_f)

该函数配置PO通道在恒流输出模式下，输出的电流值，PWM频率，kp，ki值。

参数含义：

chan\_u9: PO通道，0-11分别表示PO1-PO12。

current\_u16:目标电流值，电位mA。

freq\_u16：PWM频率值，范围[50,4000]单位HZ。初始化函数中默认值为200HZ。

kp\_u16: 0.01-1，1表示写入寄存器的最大值4095。初始化函数中默认值是0.25。

ki\_u16: 0.01-1，1表示写入寄存器最大值4095。初始化函数中默认值为0.025。

（3）void API\_Dither\_Par\_Config(uint8\_t chan\_u8, uint8\_t dither\_enable, uint8\_t dither\_enable,uint16\_t dither\_freq, uint8\_t dither\_amp)

该函数配置PO通道的颤振使能和参数。

参数含义：

chan\_u8: PO通道，0-11分别表示PO1-PO12

dither\_enable:颤振使能，1-打开颤振，0-关闭颤振。

dither\_freq:颤振频率，单位Hz，输入值小于等于PWM频率的1/4。

Dither\_amp:颤振振幅，单位mA。

（4）uint16\_t API\_DynamicCurrent\_Read(uint8\_t chan\_u8)

返回值：

读取到的电流值，单位mA。

该函数获取PO通道的电流值。电流值为通过读取芯片寄存器的值转换得到的。

参数含义：

chan\_u8: PO通道，0-11分别表示PO1-PO12。

（5）void API\_Power\_Switch\_Set(uint8\_t chan\_u8, uint8\_t on\_off\_u8)

该函数实现在开关模式下，输出高低电平的控制。

参数含义：

chan\_u8: PO通道，0-11分别表示PO1-PO12

on\_off\_u8:高低电平控制，0输出低电平，1输出高电平。

（6）uint16 API\_Duty\_Feedback\_Read(uint8\_t chan\_u8)

返回值：

占空比值，0-100表示0-100%。

该函数获取PO通道的占空比。

参数含义：

chan\_u8: PO通道，0-11分别表示PO1-PO12。

（7）void bsp\_Diag\_Reset\_Fault\_PO(uint8\_t chan\_u8)(具体功能需要验证)

返回值：

故障代码

0:No fault

1:Short to Battery Fault--高电平状态时才会诊断

2:Short to Ground Fault--低电平状态时才会诊断

3:Open Load Fault--低电平状态或高电平(恒流模式)状态时才会诊断

4:Over Current Fault--通道电流大于3.5A

5:Over Voltage--UBP2大于38V或者芯片硬件上报过压

0xff:参数错误

该函数获取PO通道的故障代码。