M9F951 Flash空间共有32KB，128字节一页，共256页，一次擦写一页。

实际应用中如外部有强干扰源，极可能使程序异常，出现诸如跑飞等问题，如果Flash未进行一些保护操作，则有可能意外更改程序区。

例程提供了一种具有高安全系数的Flash操作，可有效降低程序异常导致的Flash误操作概率。

问题根源：受干扰或程序异常跑飞后，PC指针随机指向某处，如果正好位于Flash操作函数附近，运行后将不可控的擦写未知地址数据。

解决方法：在写操作前添加一系列验证环节，此时PC指针即便指向Flash操作函数附近，也无法轻易进入擦写环节。

# 写EE相关函数

|  |  |
| --- | --- |
| 宏定义 | #define EE\_BeginAddr 0X6F00#define EE\_EndAddr 0X7000 |
| EE\_BeginAddr | 用户数据存储区首地址设置，该地址必须为128的整数倍，即低8位是0x00或0x80 |
| EE\_EndAddr | 用户数据存储区结束地址设置，该地址必须为128的整数倍，即低8位是0x00或0x80，注意存储区不包含此页 |
| 功能说明 | 用于指定用户数据存储区范围，超出此范围将无法被操作。示例中起始地址0X6F00、结束地址0X7000，则代表实际可操作范围0X6F00-0X6FFF区间 |

|  |  |
| --- | --- |
| 函数名 | void FlashMgr\_UnLockEE (void) |
| 传入参数 | 无 |
| 传出参数 | 0:失败 1:成功 |
| 功能说明 | 用户数据存储区解锁，必须先调用该函数再调用写入函数 |

|  |  |
| --- | --- |
| 函数名 | unsigned char FlashMgr\_Lock (void) |
| 传入参数 | 无 |
| 传出参数 | 无 |
| 功能说明 | Flash上锁，写入完成后必须调用该函数 |

|  |  |
| --- | --- |
| 函数名 | unsigned char FlashMgr\_EEWritePage (unsigned int pageAddr) |
| 传入参数 | 页地址，范围为设定的用户数据存储区范围 |
| 传出参数 | 写入状态 0:失败 1:成功 |
| 功能说明 | 将RAM中0X780为起始地址的128字节写入指定的页中。 |

|  |  |
| --- | --- |
| 函数名 | unsigned char FlashMgr\_WriteEE (unsigned char \*dat,unsigned int addr,unsigned char len) |
| 传入参数 | \*dat:数据 addr:写入起始地址 len:数据长度 |
| 传出参数 | 0:失败 1:成功 |
| 功能说明 | 向指定起始地址写入数据，调用前先调用FlashMgr\_UnLockEE |

# 覆写程序区

|  |  |
| --- | --- |
| 函数名 | void FlashMgr\_UnLockCode(void) |
| 传入参数 | 无 |
| 传出参数 | 0:失败 1:成功 |
| 功能说明 | 程序区解锁，必须先调用该函数再调用写入函数 |

|  |  |
| --- | --- |
| 函数名 | unsigned char FlashMgr\_OverwriteCode(unsigned char dat,unsigned int addr) |
| 传入参数 | dat:数据 addr:地址 |
| 传出参数 | 无 |
| 功能说明 | 向指定地址写入数据 |

# 使用原则

配置可修改范围，写入数据前先调用解锁函数。