

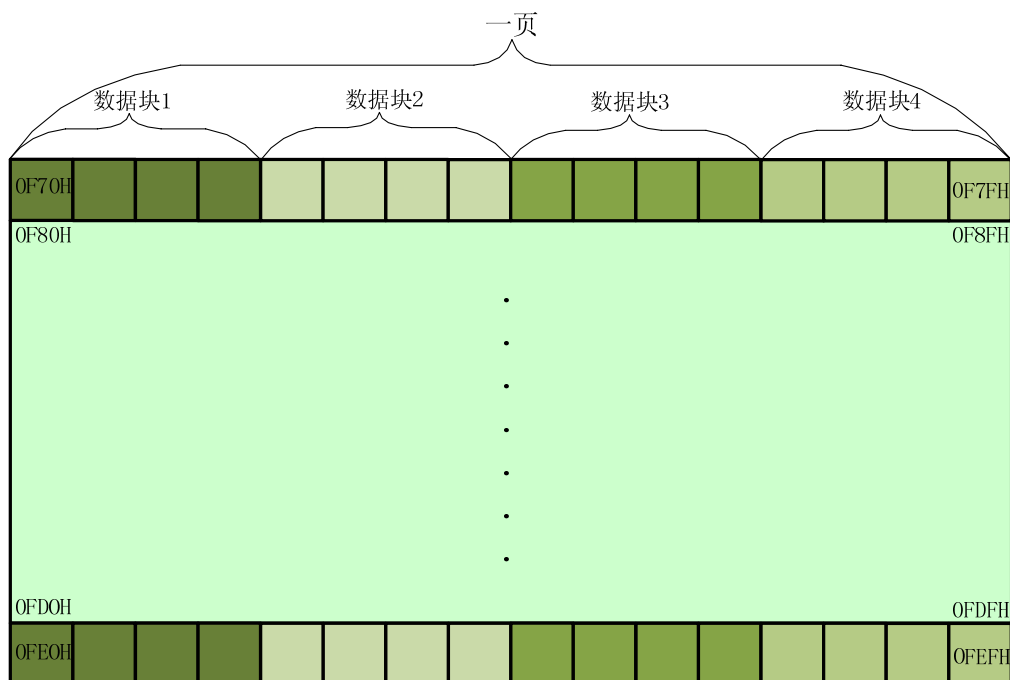
KF8FXXX 可靠读写 NVM 应用笔记

1 概述

内部带 NVM（BLEE 或者 Data EEPROM）的 MCU 芯片(如 KF8F312，本文后面以 KF8F312 为例子)，在程序执行中通常会根据条件判断去执行读写 NVM 单元的操作，特别是写 NVM 单元的操作，要注意。否则很容易造成误写 NVM 单元或者写入 NVM 单元的值错误，造成系统运行异常。

2 可靠写 NVM 的方法

KF8F312 NVM 的组织如下图所示。其中一页 16 个 WORDS 及一页 16*16 bits，一页分成四个块，每个块 4 个 WORDS。在写第一个块的时候启动擦除操作，会把整个一页的内容擦除成 0xFFFF 值，在每个块的最后一个地址（地址 0B_1111_XXXX_XX11）启动写操作，把待写的值写入对应的 NVM 单元中。



可靠写 NVM

为保证每次写操作的条件判断都正确无误，以及每次写入的数据都可靠，在程序中推荐采用如下做法：

- 1、把写入的值同时写入到三个不同的页；
- 2、写入的三个页写分三次，每写入一次都判断写入的条件是否成立；

如要写 RAM80 的值到 NVM 中，可以把 RAM80 的值同时写入 NVM 的 0X0F70 开始页的 0X0F70 地址、0X0F80 开始页的 0X0F80 地址、0X0F90 开始页的 0X0F90 地址，示例代码

如下所示:

Fun()

```
{  
    IF (Wr_NVM_Flag)          // Wr_NVM_Flag=1 时启动写 NVM  
    {  
        Call  Wr_NVM(Addr_1st_Page, Data) //把数据写入 NVM 的第一页  
    }  
  
    IF (Wr_NVM_Flag)          // Wr_NVM_Flag=1 时启动写 NVM  
    {  
        Call  Wr_NVM(Addr_2ed_Page, Data) //把数据写入 NVM 的第二页  
    }  
  
    IF (Wr_NVM_Flag)          // Wr_NVM_Flag=1 时启动写 NVM  
    {  
        Call  Wr_NVM(Addr_3rd_Page, Data) //把数据写入 NVM 的第三页  
    }  
}
```

通过三次写，三次写入条件判断，保证程序在跑飞等入口错误的情况下能可靠写 NVM。

可靠读 NVM

上面的三次写入，保证每次写入都进行了条件判断，排除了在程序错误入口时的写操作错误：即跳过条件的误写。为保证系统的可靠性，还需要配合对读操作做如下处理：对写入三个不同地址 NVM 的同一个值做比较，只有至少两个地方的值相同，才认为数据有效，否则报错。