

# 基于 STM32 的 Flash 存储器坏块自动检测<sup>①</sup>

张亚辉, 马胜前

(西北师范大学 物理与电子工程学院, 兰州 730070)

**摘要:** 针对 Nand Flash 存储器存在坏块的问题, 提出一种基于 STM32 的 Flash 存储器坏块自动检测方法, 通过 STM32 内部可变静态存储控制器, 发出相应的数据、地址、控制信号, 在不增加外部器件的情况下, 快速访问 Flash 存储器, 并给出了部分硬件电路和 C 语言编写的程序代码. 该设计已成功实现自动检测 Flash 坏块的功能; 操作简单、检测速度快、准确率高; 并能读取 Flash 的 ID 号检测 Flash 性能, 同时能够存储和读取 2GB 数据.

**关键词:** Nand Flash; 自动检测; STM32; 可变静态存储控制器; 串口通讯

## Auto Detection of Invalid Block of Flash Memory Based on STM32

ZHANG Ya-Hui, MA Sheng-Qian

(College of Physics and Electronic Engineering, Northwest Normal University, Lanzhou 730070, China)

**Abstract:** According to Nand Flash memory bad block problem, this paper proposed a automatic check scheme to deal with bad block based on STM32. Without increasing the external device, through its internal flexible static memory controller, sends the corresponding data, address, control signals, to rapid access Flash memory. Its part circuit schematic diagram and C language program code was introduced. The design has been successfully realized the function of automatic checking the invalid block finally, and has the advantages of simple operation, fast detection speed, high accuracy rate, and can read the ID number of Flash to get the performance of the memory. And the design also can write and read 2GB data.

**Key words:** Nand Flash; automatic detection; STM32; Flexible Static Memory Controller(FSMC); serial communication

Nand Flash 内存是 Flash 内存的一种, 其内部采用非线性宏单元模式, 为固态大容量内存的实现提供了廉价有效的解决方案. Nand Flash 存储器因具有高密度、大容量、高数据存储速率, 以及更多的擦除次数等特点, 得到了迅猛的发展. 在数码相机、手机、移动存储卡、掌上电脑、固态存储盘、MP3 播放器等方面, 得到广泛的应用. 但是由于工艺和使用环境的问题, 在 Nand Flash 的生产及使用过程中会不可避免的产生坏块. 针对 Flash 存在坏块的问题也提出了很多解决方法, 但 Nand Flash 的控制逻辑比较复杂, 命令、数据、地址采用同一总线, 对时序要求也十分严格, 这就给判断坏块、给坏块做标记和擦除等操作带来很大的难度. 于是就要求有一个简单可行的检测方法, 使用户能够方便准确地检测 Flash 存储器的好坏, 为其提供更为准

确的判断依据. 为此本文讨论了另外一种坏块处理方法, 利用 STM32 内部的 FSMC 总线实现对 Flash 存储器的读、写、擦除等操作, 很好地解决了模拟时序困难的问题, 软件代码设计简单、检测速度快, 使检测结果的可靠性得到了很大的提高.

## 1 系统设计方案

目前, 对于 Flash 存在坏块问题, 主要有以下 5 种检测方法: (1) 基于单片机控制的方法, 主要通过读取 Flash 每块的第 1、第 2 页内容, 判断该块的好坏<sup>[1]</sup>. 但单片机的运行速度低, 难以满足高速检测的需要. 对于使用过的 Flash, 其出厂坏块标记信息可能被擦除, 仅靠读取出厂坏块标记信息, 检测结果不准确(2)在 FPGA 内建立片内 RAM, 用于存储无/有效块, 将无/

<sup>①</sup> 基金项目: 国家自然科学基金(61162017); 甘肃省教育厅研究生导师项目(1101-03)

收稿时间: 2012-11-26; 收到修改稿时间: 2013-01-16

有效块地址作为数据存入 RAM 中, 从 RAM 中读出数据即为无/有效块地址<sup>[2,3]</sup>. 该方法增加了软件设计的复杂程度, 且 RAM 空间大小受到限制; 先对 Flash 进行擦除操作, 由于擦除操作时间长, 使检测速度变慢, 还有可能擦除出厂坏块标记信息; (3)在 FPGA 内部建立 RAM, RAM 地址与块地址相对应, 在与坏块地址相对应的 RAM 中存储 1, 而其他地址存 0<sup>[4]</sup>. 这样根据 RAM 中存储内容即可判断有效块地址, 该方法同样增加了对 RAM 的读写操作. (4)在 FPGA 内部建立屏蔽坏块函数, 用专门的 Flash 存储坏块信息. 这样要确保存储坏块信息的区域, 不是坏块, 否则会使坏块信息丢失, 检测不准确<sup>[5]</sup>. (5)基于 FAT 文件系统处理坏块. 该方法在文件系统层解决坏块问题, 对块、簇以及扇区的操作存在一定难度<sup>[6]</sup>.

基于上述文献, 本系统设计的特点: (1)基于 STM32 的 FSMC 访问 Flash 存储器, 软件代码设计简单, 检测精度高; (2)先读取 Flash ID, 然后读取指定位置的 Flash 出厂坏块标记信息, 快速检测出坏块数, 及其物理地址; (3)可以通过擦除, 读取寄存器状态, 再次核实坏块数; (4)对于指定块的指定位置, 可以进行读写操作, 核实写入的数据, 和读出的数据是否一致, 进一步核实是否为坏块, 提高检测的准确性. 系统总体框图, 如图 1 所示.

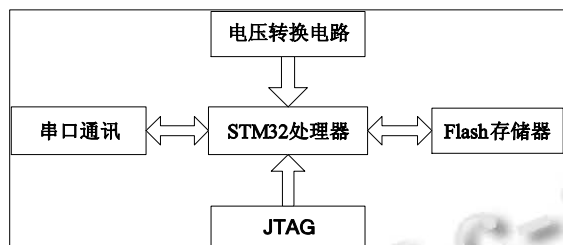


图 1 系统总体框图

## 2 硬件设计

关键硬件部分是 STM32 处理器和 Flash 存储器的连接及其外围电路设计. MCU 可以和 Flash 存储器直接相连, 不必使用电平转换芯片, 简化了硬件电路的设计.

### 2.1 核心处理器

STM32 是 ST(意法半导体)公司推出的基于 Cortex-M3 内核的 32 位微控制器系列, 采用高效的哈佛结构、三级流水线、Thumb-2 高密度指令集; Cortex-M3 内核降低了系统存储要求, 同时快速的中

断处理能够满足控制领域的高实时性要求<sup>[7,8]</sup>. 本设计选用增强型的 STM32F103ZET6, 它使用 72MHz 系统时钟频率、512KB 闪存程序存储器、64KB SRAM、11 个定时器、3 个 12 位模数转换器、2 个 12 位数模转换器、12 通道 DMA 控制器、以及 SPI、USART、I2C、I2S、USB 接口等<sup>[9,10]</sup>. 工作电压为 3.3V.

可变静态存储控制器 (Flexible Static Memory Controller, FSMC), 是 STM32 系列中内部集成 256KB 以上 Flash, 是高存储密度微控制器特有的存储控制机制. FSMC 具有的优点如下: (1)支持多种静态存储器类型; (2)支持丰富的存储器操作方法; (3)支持同时扩展多种存储器; (4)支持更为广泛的存储器型号; (5)支持代码从 FSMC 扩展的外部存储器中直接运行. 满足系统设计对存储容量、成本的综合要求.

STM32F103ZET6 的 FSMC\_D0~FSMC\_D7 连接 Flash 存储器的 I/O0~I/O7, 作为 STM32 处理器与 Flash 存储器的命令、地址、数据传输端口. FSMC\_A16、FSMC\_A17、FSMC\_NWE、FSMC\_NOE、FSMC\_NCE2、FSMC\_NWAIT/INT2 分别控制 Flash 命令锁存使能(CLE)、地址锁存使能(ALE)、写使能(WE)、读使能(RE)、片选使能(CE)、状态控制信号(R/B). STM32 与 Flash 的连接电路, 如图 2 所示.

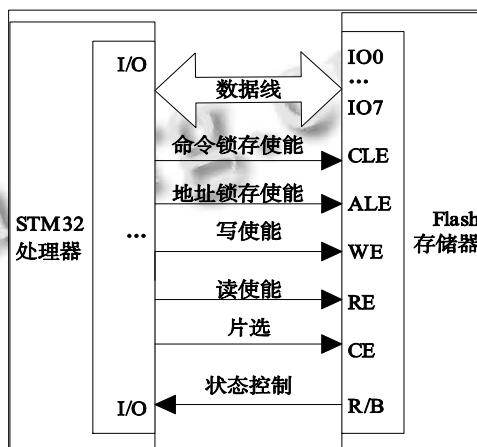


图 2 STM32 与 Flash 的连接电路图

### 2.2 Flash 存储器

K9KAG08U0M 是 SAMSUNG 公司, 2GB 容量、8 位, 高可靠、非易失 Flash 存储器, 3.3V 供电. 在 Nand Flash 中, 一个块中含有 1 个或多个位是坏的, 就称其为坏块. 坏块的稳定性是无法保证的, 不能保证你写入的数据是对的; 或者写入对了, 读出来也不一定是

对的. 而正常的块, 肯定是写入读出都是正常的. 和所有有效块一样, 它具有相同的 AC 和 DC 参数, 一个无效块不会影响有效块的运行. 因为它有相应独立的指令资源依靠选择晶体管, 该系统设计必须通过地址掩盖其无效块. 第 1 块(地址是 00h)为了保证是有效块, 在 1 KB 编程/擦除周期内都不要要求纠错.

Flash 存储器的坏块有两类: 一类是出厂时就有的坏块, 称作 invalid block, 在出厂之前, 就会标记为坏块. 另一类是在使用过程中产生的. 根据 Flash 的性能特点, 除了出厂标记的坏块信息, 其他的存储单元都被擦除, 无效块状态定义在空余区域的第 1 个字节, 在每块第 1 或者第 2 页 4096 字节的列地址中存储内容非 FFh. 在很多情况下, 出厂标记的坏块信息也是能被擦除的, 一旦擦除, 就不可能恢复其原有信息<sup>[11]</sup>. 因此, 系统必须在原来无效块的信息基础上识别无效块. 本系统设计, 就是基于读取每块的第 1 和第 2 页坏块信息标记位, 判定该块是否为无效块. 对于使用过的 Flash 存储器, 先读取出厂坏块标记信息, 再通过读写操作进一步判断该块是否为坏块.

### 3 软件设计

在软件设计方面, 使用 STM32 官方提供的完整、高效的固件库, 无需使用汇编操作外设特性, 减小了代码开发的周期、降低了开发的难度, 提高了程序的可读性和易维护性.

在使用 Nand Flash 之前, 需要对其使用的 FSMC BANK 进行参数配置, 包括: 存储器类型、占用的 FSMC BANK、存储器数据宽度、ECC 页大小, 使之与 Nand Flash 芯片的要求相符合. 还包括 FSMC 管脚初始化、配置 FSMC 存储器时间参数, 至此完成 FSMC 总线的初始化, 就可以对 Flash 存储器进行访问了. 判断坏块的流程如图 3 所示.

操作 NAND 存储器, 需要使用特别的访问协议, 所有的读写操作, 需要有如下步骤: (1)向 NAND 存储器发送一个命令. (2)发送读或写的地址. (3)读出或写入数据. 为了方便操作 Flash 存储器, FSMC 的 NAND 存储块被分为 3 个段: 数据段、命令段、地址段<sup>[12]</sup>. 如读取 Flash 数据时, 初始化用到的 FSMC 引脚, 写入命令段的读取数据的第 1 组命令 00h, 其实是向 Flash 存储器写入命令; 写入地址段地址, 其地址是向 Flash 存储器写入读操作的地址, 依据 Flash 的构造特点, 需要

5 个写入地址段才能写入一个读操作的地址; 写入命令段的读取数据的第 2 组命令 30h; 读出数据段的地址, 结果都是读出 Flash 存储器的内部单元, 该单元的地址是之前在地址段写入的那个地址. FSMC 起到“桥梁”的作用, 其他操作类似. 本系统软件设计核心部分是读取 Flash 存储器坏块函数, 此函数中包含对 Flash 的页面读取函数. 依据坏块检测流程图, 函数中检测每块的第 1 和第 2 页中, 出厂标记坏块的位置, 即列地址为 4096, 读取该地址的内容, 存入缓存中, 再判断读取数据, 若不是 FFh, 该块被记录为坏块, 继续检测, 直到最后一块. 该函数源程序代码如下:

```
for(blocknum = 0; blocknum < flash.BlockNum;
blocknum++)
{ for(check = 0; check < CHECKNUM; check++)
{ column = flash.MakesAddr;
row = blocknum * flash.PageNum + check;
Flash_ReadCycle(flash, column, row);
invalid[check]=*(__IO uint8_t*)
(Bank_NAND_ADDR | DATA_AREA);
If(invalid [0] != 0xff || invalid [1] != 0xff)
{ printf(" BlockNum = %d\n", blocknum);
invalid_count++;;
}
}
```

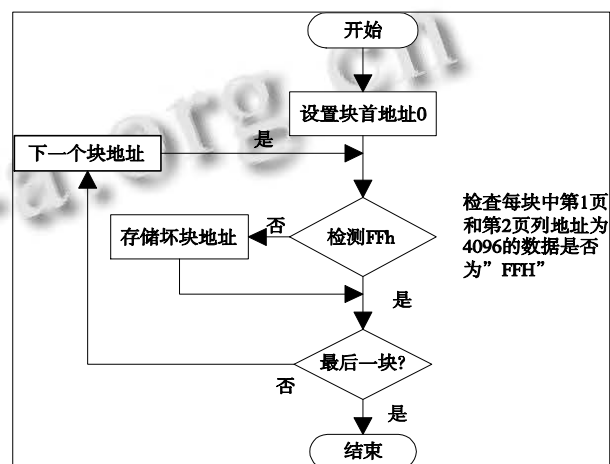


图 3 判断坏块的流程图

### 4 结语

本文介绍了一种基于 STM32 的 Flash 存储器坏块自动检测的设计思想, 重点介绍了 FSMC 的配置, 以及对 Flash 的坏块检测功能. 在 Keil uVision4 软件环境

(下转第 215 页)

并进一步通过建立消防员喷水的数学模型, 计算模拟实验所需要的数据。

与利用双目图像检测灭火水柱落点相比较, 通过数学建模计算仿真数据的方法对于更加真实的表现粒子效果具有十分重要的作用, 不仅展示了一个更加真实的画面, 同时也结合了消防器材的性能参数, 给消防演练的科学评估提供了可靠的依据。

### 参考文献

- 1 Martz P. Open scene graph quick start guide. New York: Skew Matrix Software LLC. <http://www.lulu.com/con-tent/1164927>, 2007.
- 2 Wang JW, Hu WP, Jin YF. 8-word dynamic fireworks simulation based on particle system. Computer Simulation,

2010,27(10):211-244.

- 3 Xu JS, Song WZ. Snow simulation based on particle system in OSG. Computer Engineering and Design, 2012,33(4): 1509-1513.
- 4 Hua ZX, Wang YC, Sun JS. Study of explosion simulation based on particle system. Computer Science, 2012,39(4):278-281.
- 5 Zhao JM, Zhang H, Zheng GQ. Fountain simulation based on particle system. Application Research of Computers, 2006,(1): 244-249.
- 6 Gao GZ, Hong Y, He ZY, Yu LF. The application of the image processing technology in the fire fighting simulative training. Journal of National University of Defense Technology, 2003, 25(3):49-53.

(上接第 211 页)

中, 用 C 语言编写系统程序, 通过 PC 串口打印测试信息, 该设计快速、简单地实现了对 Flash 擦除、Flash 坏块检测、Flash 读写、读取 Flash ID 等操作。此外, 该设计已成功用于对 Flash 芯片的筛选, 可以对不同厂家不同的芯片进行检测, 具有较强的通用性和可行性。

### 参考文献

- 1 王新舜,张存善,韩力英,杨振华.基于单片机的 Flash 存储器坏块自动检测.电子设计工程,2010,18(3):125-128.
- 2 柯艳,李杰,刘俊,秦丽,孔祥雷,陶瑞萍.基于 FPGA 与 USB2.0 的多功能读数盒.仪表技术与传感器,2010,11(11):32-34.
- 3 吴鹏.高速实时 Flash 阵列数据采集系统研究与实现[学位论文].南京:南京理工大学,2007.
- 4 韩茜,罗丰,吴顺君.高速大容量固态存储系统的设计.雷达科学与技术,2005,3(2):110-114.
- 5 张胜勇,高世杰,吴志勇,田丽霞.基于 FPGA 的 NAND Flash

坏块处理方法.计算机工程,2010,36(6):239-243.

- 6 罗晓,刘昊.一种基于 FAT 文件系统的 NAND Flash 坏块处理方法.电子器件,2008,31(2):716-719.
- 7 王永虹,徐炜,郝立平.STM32 系列 ARM Cortex-M3 微控制器原理与实践.北京:北京航空航天大学出版社,2008.23-62.
- 8 喻金钱,喻斌.STM32F 系列 ARM Cortex-M3 核微控制器开发与应用.北京:清华大学出版社,2011.13-39.
- 9 李宁.基于 MDK 的 STM32 处理器开发应用.北京:北京航空航天大学出版社,2008.46-68.
- 10 STM32F103xC.STM32F103xD.STM32F103xE. [2012-06-28]. <http://www.st.com>.
- 11 SAMSUNG.K9WBG08U1M.K9KAG08U0MK9NCG08 U5M. [2012-06-28].
- 12 Using the high-density STM32F10xxx FSMC peripheral to drive external memories.[2012-06-28]. <http://www.st.com>.