

基于地址映射的 NAND Flash 控制器设计

徐磊^{1,2}, 王保成¹

(1. 中国科学院空天信息创新研究院, 北京 100094;

2. 中国科学院大学航空宇航学院, 北京 100049)

摘要: 针对嵌入式数据采集系统对 NAND Flash 进行读写控制时出现的坏块问题和磨损失衡问题, 对数据采集系统的工作特点进行分析, 借鉴闪存转换层的思想, 提出了一种基于地址映射的 NAND Flash 控制方法, 通过建立、维护、查询 NAND Flash 存储块逻辑地址与物理地址之间的映射关系表, 实现 NAND Flash 的坏块管理和磨损均衡功能, 同时介绍了使用地址映射方法的 NAND Flash 控制器设计过程; 仿真测试和实际应用结果表明, 基于地址映射方法设计的 NAND Flash 控制器能够识别、管理出厂坏块和突发坏块, 均衡存储块的磨损, 提高嵌入式数据采集系统的可靠性; 该方法实现过程简单, 无需移植文件系统, 硬件资源要求低, 为嵌入式数据采集系统中 NAND Flash 的读写控制提供了新的思路。

关键词: NAND Flash; 地址映射; 坏块管理; 磨损均衡; ZYNQ

Design of NAND Flash Controller Based on Address Mapping

XU Lei^{1,2}, WANG Baocheng¹

(1. Aerospace Information Research Institute, Chinese Academy of Sciences, Beijing 100094, China;

2. School of Aeronautics and Astronautics, University of Chinese Academy of Sciences, Beijing 100049, China)

Abstract: Aiming at the problems of bad blocks and uneven wear when using NAND Flash in embedded data acquisition system, a NAND Flash control method based on address mapping is proposed by analyzing the working characteristics of data acquisition system and using the idea of Flash translation layer. The functions of bad block management and wear-leveling are implemented through establishing, maintaining and querying the address mapping relation table from logical address to physical addresses of NAND Flash blocks. The design process of the NAND Flash controller using the way of address mapping is introduced. The results of simulation and application show that the NAND Flash controller based on address mapping could identify and manage factory bad blocks and burst blocks, and balance the wear of storage blocks and improve the reliability of the system. The method is simple and does not need to migrate the file system, and has low hardware requirements, it provides a new idea for NAND Flash read/write control in field of the embedded data acquisition system.

Keywords: NAND Flash; address mapping; bad block management; wear leveling; ZYNQ

0 引言

NAND Flash 是一种具有存储密度高、功耗低、读写速度快等优点的非易失性存储器, 被广泛应用于消费类电子、汽车、航空航天等领域^[1-3]。NAND Flash 具有以下特点:

1) 读、写、擦除过程中可能会因为比特位翻转出现坏块^[4]; 2) 每个存储单元块的擦除、写入次数达到上限后单元块会失效^[5], 因此需要有效的数据管理方法实现对 NAND Flash 的读写控制。

NAND Flash 的控制方法通常可以分为基于文件系统控制法和直接控制法^[6]。第一种方法通过移植文件系统实现对 NAND Flash 的读写控制, 如文献 [7] 使用 FAT 文件系统+FAST 闪存转换层算法构建了 NAND FLASH 存储系统; 文献 [8] 则通过 YAFFS 文件系统实现了星载固态

存储器的数据管理。这种方法的优点可以实现存储数据的高效统一管理, 具备完整的数据分区、坏块管理、磨损均衡功能; 缺点是需要移植嵌入式操作系统, 对硬件资源要求较高。第二种方法通过直接给出控制指令和地址对 NAND Flash 进行读写控制, 如文献 [9-10] 基于 FPGA 使用直接控制的方法实现了 NAND Flash 控制器; 文献 [1]、[11] 同样采用直接控制的方式, 将待存储数据按顺序写入 Flash 存储块, 实现了一种双 Flash 数据存储体系。直接控制法的优点是操作简单, 无需使用操作系统; 缺点是遇到坏块通常采用直接跳过的方式, 缺乏坏块管理机制, 且每次上电复位后均从原始地址块进行擦写^[6], 容易造成存储块磨损失衡。

由以上分析可知, 基于文件系统控制法和直接控制法各有优劣。考虑到传统的嵌入式数据采集系统, 如文献

收稿日期: 2022-10-12; 修回日期: 2022-11-07。

基金项目: 国家自然科学基金项目(61733017)。

作者简介: 徐磊(1998-), 男, 安徽芜湖人, 硕士研究生, 主要从事 FPGA 测试和开发方向的研究。

通讯作者: 王保成(1974-), 男, 湖南岳阳人, 副研究员, 硕士生导师, 主要从事航空航天科学与工程、机械工业、自动化技术方向的研究。

引用格式: 徐磊, 王保成. 基于地址映射的 NAND Flash 控制器设计[J]. 计算机测量与控制, 2023, 31(6): 109-116, 122.

[12] 设计的气象数据采集系统, 文献 [13] 设计的星载数据采集系统中, 对 NAND Flash 采用直接控制法, 缺乏有效的数据管理方式, 而采用基于文件系统控制法会带来硬件要求的提高和软件设计复杂度的提升。因此本文提出一种两层地址映射的方式对 NAND Flash 进行读写控制, 第一层地址映射将逻辑地址按顺序映射到中间逻辑地址, 实现存储块的均衡使用, 第二层映射通过坏块监测和好块替换坏块的方式建立坏块映射表, 将中间逻辑地址映射到物理地址, 实现坏块管理。使用地址映射方法, 设计了一款 NAND Flash 控制器。基于地址映射方式的 NAND Flash 控制器不需要移植文件系统, 操作过程简单, 解决了嵌入式数据采集系统中 NAND Flash 的坏块管理问题以及存储块均衡使用问题, 提升了系统的可靠性。

1 地址映射方法原理

1.1 方法核心思想

闪存转换层 (FTL) 是固态硬盘的核心部件, FTL 的存在让用户可以像磁盘一样使用固态硬盘^[14]。FTL 的一大功能是实现逻辑地址到物理地址的映射, 根据映射尺度, FTL 映射方式又分为块映、页映射、混合映射等。当 NAND Flash 运用于嵌入式数据采集系统中, 由于采集的数据具有数量大、连续性高的特点, 因此本文借鉴 FTL 块映射方式, 通过特定的算法将存储块逻辑地址映射到物理地址, 解决 NAND FLASH 坏块问题和磨损失衡问题; 并且通过地址映射的方式, 使得 NAND FLASH 存储块地址对于上层驱动和应用完全透明, 也方便了嵌入式软件设计。考虑到坏块管理和磨损均衡对于地址转换需求的差异性, 因此将地址映射过程分为两个层次, 第一层映射用于均衡存储块的磨损, 第二层映射用于处理坏块问题, 其过程如图 1 所示。其中中间逻辑地址为串联一、二层映射关系的中介地址, 当没有出现坏块时, 中间逻辑地址等于物理地址。Table_WL (table of wear leveling) 和 Table_BM (table of badblock management) 分别为两层映射的地址关系表, 这两张地址映射信息表的建立与维护是地址映射的方法的核心。

1.2 第一层地址映射信息表的建立与维护

Table_WL 是第一层地址映射信息表, 用于将逻辑地址映射到中间逻辑地址, 实现磨损均衡。

考虑到数据采集系统中, 采用直接控制法进行 NAND FLASH 读写控制时, 数据存储过程具有以下特点:

1) 不同类型的数据存储在初始地址不同的连续存储块中, 每次上电时均从初始地址进行数据擦写操作, 且每次写入数据的大小不确定, 因此会出现频繁被擦写的“热块”

和很少被擦写的“冷块”, 不同存储块擦写次数差异的增大可能会导致 NAND Flash 寿命的减小^[15]。

2) 数据采集系统单次上电操作不会对已写入数据的存储块进行重复的擦写操作。

基于以上特点, 让系统按照第 1 号存储块—第 N (N 表示存储块最大地址) 号存储块的顺序循环向 NAND FLASH 写入数据, 可以使每一个存储块具有相同的机会被使用, 由此建立第一层地址映射关系。

记逻辑地址的范围为 $1 \sim N$, 逻辑地址 $n (1 \leq n \leq N)$ 对应的中间逻辑地址为 Temp_Addr (n)。记一次数据采集存储过程中, 逻辑地址最后一次映射得到的中间逻辑地址为 Last_Temp_Addr, 初次上电启动时 Last_Temp_Addr 赋值为 N。记初次上电启动标志为 Flag, 当 Flag 为 1 时表示系统是第一次上电启动。Table_WL 的建立和维护过程如下:

1) 上电启动, 读取启动标志 Flag, 进行 Table_WL 初始化:

```

for n from 1 to N
    if flag = 0
        Temp_Addr(n) = Read_Table_WL(n)
    else
        Temp_Addr(n) = 1
    end
end
    
```

其中: Read_Table_WL (n) 表示通过读取前一次上电结束时保存的 Table_WL 地址映射信息, 得地址到的逻辑地址 n 对应的中间逻辑地址。

2) 控制器对逻辑地址为 n 的存储块发出擦:

除指令时更新逻辑地址和中间逻辑地址的映射关系:

```

if Last_Temp_Addr = N
    Temp_Addr(n) = 1
    Last_Temp_Addr = 1
else
    Temp_Addr(n) = Last_Temp_Addr + 1
    Last_Temp_Addr = Last_Temp_Addr + 1
end
    
```

3) 完成数据存储后, 将反映第一层地址映射信息关系的 Table_WL、Last_Temp_Addr 以及启动标志 Flag 记录到非易失性存储器中。

1.3 第二层地址映射信息表的建立与维护

Table_BM 是第二层地址映射信息表, 用于将中间逻辑地址映射到物理地址, 实现坏块管理功能。针对 NAND Flash 使用过程中出现的坏块, 常用的坏块管理方法有直接跳过法和坏块替换法, 前一种方法会使得逻辑地址和物理地址的映射关系变得混乱^[16], 因此使用好块替换坏块, 该方法可以分为坏块识别以及坏块替换两个步骤。

NAND Flash 坏块分为出厂时自带的坏块和使用过程中产生的坏块, 前者可以通过读取每一个存储块的坏



图 1 地址映射方法过程

块标记位进行区分, 后者需要使用 ECC (error checking and correction) 码对存储数据进行校验。常用的 ECC 码有汉明码、BCH (Bose-Chaudhuri-Hocquenghem) 码、LDPC (low density parity check) 码等, 不同校验码有着不同的校验和纠错能力, 当错误超出 ECC 纠错能力范围时判定出现坏块^[17]。坏块替换需要预留一定数量的好块, 当正在使用的存储块被判定为坏块时, 采用预留的好块替换掉坏块, 并记录坏块与好块的地址映射关系, 实现坏块管理功能。预留好块的数量需要根据芯片实际参数进行设置, 通常 NAND Flash 使用过程中可能出现的坏块数量不超过 2%^[16]。第二层地址映射关系本质就是使用好块地址替换坏块地址。

记中间逻辑地址为 $m(1 \leq m \leq N)$, 中间逻辑地址对应的物理地址为 $Phy_Addr(m)$, 物理地址的范围为 $1 \sim P$ 。预留一定数量的存储块用于替换使用过程中可能出现的坏块, 预留存储块的地址范围为 $(N+1) \sim P$, 占总存储块数量的 2%。记一次数据采集存储过程中, 最后用于替换坏块的保留块地址为 $Last_Res_Addr$, 初次上电启动时设置 $Last_Res_Addr$ 为 N ; $Table_BM$ 的建立以及维护过程如下:

1) 上电启动, 读取启动标志 $Flag$, 进行 $Table_BM$ 初始化, 若 $Flag$ 为 1, 代表初次启动, 则遍历前 N 个存储块的坏块标记位; 若 $Flag$ 为 0, 则直接读取上次上电结束时保存的 $Table_BM$ 信息:

```

for m from 1 to N
    if flag = 0
        Phy_Adr(m) = Read_Table_BM(m)
    elseif Bad_Block(m) = 0
        Phy_Adr(m) = m
    else
        Phy_Adr(m) = Last_Res_Adr + 1
        Last_Res_Adr = Last_Res_Adr + 1
    end
end
    
```

其中: $Read_Table_BM(m)$ 表示通过读取前一次上电结束时保存的 $Table_BM$ 地址映射信息, 得到的中间逻辑地址 m 对应的物理地址。 $Bad_Block(m)$ 表示初始化读取第 m 个存储块的坏块标记位的结果, 为 1 表示坏块, 为 0 表示好块。

2) 当控制器对中间逻辑地址为 n 的存储块进行擦除、读、写操作时, 逻辑地址经过两层地址映射得到的物理地址块被判断为坏块时, 更新中间逻辑地址和物理地址的映射关系:

```

m = Temp_Adr(n)
if Bad_Block(Phy_Adr(m)) = 1
    Phy_Adr(m) = Last_Res_Adr + 1
    Last_Res_Adr = Last_Res_Adr + 1
end
    
```

3) 一次上电完成数据存储后, 同样将反映第二层地址映射信息关系的 $Table_BM$ 以及 $Last_Res_Addr$ 记录到

非易失性存储器中。

2 NAND Flash 控制器设计

基于两层地址映射方法, 本文设计了一款 NAND Flash 控制器, 并运用于某型浮空器的传感器数据采集系统中, 用于控制片外 NAND Flash 进行实验数据存取。传感器数据采集系统主控芯片采用 ZYNQ7020, 下面介绍 NAND Flash 控制器的设计过程。

2.1 系统结构设计

NAND Flash 控制器整体结构如图 2 所示。控制器的主体功能基于 ZYNQ 的 PL (programable logic) 端实现, ZYNQ 的 PS (processing system) 端向 PL 端的控制器发送控制指令, 实现对片外 NAND Flash 的控制。片外 NAND Flash 使用 Micron 公司的 MT29F2G08ABAEAH4, 芯片共有 2 048 个存储块, 每块 64 页, 每页 2 048 + 64 字节, 总存储容量 2 Gb, I/O 接口为异步模式, 具备内部硬件 ECC 功能。

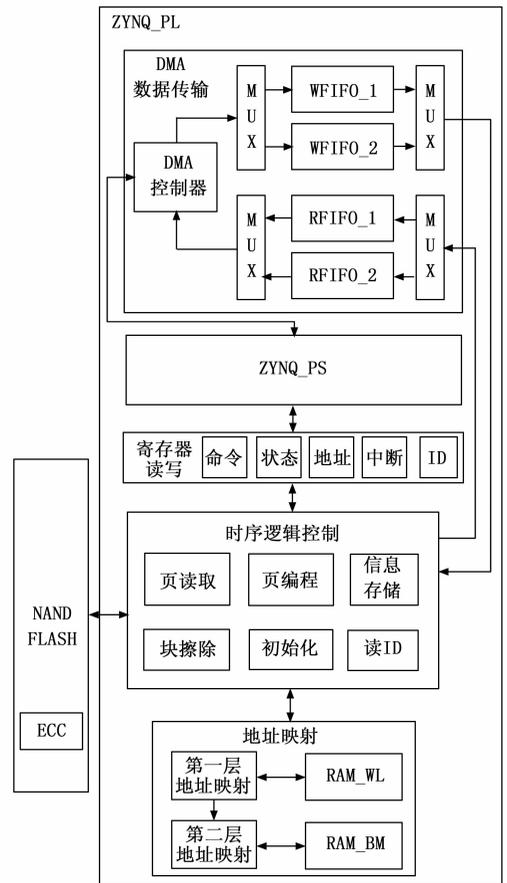


图 2 NAND Flash 控制器结构图

PL 端 NAND Flash 控制器由寄存器配置模块、时序逻辑控制模块、地址映射模块以及 DMA 数据传输模块 4 个部分构成。PS 端通过寄存器读写模块发送控制指令, 读取 PL 端 NAND Flash 控制器工作状态。时序逻辑控制模块根据 NAND Flash 接口时序要求, 驱动 NAND Flash 控制和数据信号。地址映射模块根据两层地址映射的原理, 更新地址映射信息表 $Table_WL$ 以及 $Table_BM$, 完成逻辑地址到

物理地址的映射, 实现磨损均衡和坏块管理的功能。DMA 数据传输模块采用 DMA 的方式实现 PS 端与 PL 端的数据搬运, 并通过乒乓操作的方法提高数据传输效率。

2.2 寄存器读写模块设计

寄存器读写模块用于 PS 端对 PL 端 NAND Flash 控制器的控制寄存器、状态寄存器等访问。各寄存器的定义如表 1 所示。

表 1 NAND Flash 控制器寄存器定义表

寄存器名称	访问权限	长度/bit	定义
Command_R	W/R	8	命令寄存器
Status_R	R	16	状态寄存器
Address_R	W/R	32	地址寄存器
Interrupt_R	W/R	8	中断使能寄存器
ID1_R	R	32	ID 寄存器 1
ID2_R	R	8	ID 寄存器 2

PS 端通过对命令寄存器发送相应指令码, 控制 NAND Flash 控制器执行相应操作。本文设计的 NAND Flash 控制器共有 6 种操作, 每种操作对应的指令编码如表 2 所示。

表 2 NAND Flash 操作指令码

操作	指令编码
块擦除	0x81
页编程	0x82
页读取	0x84
读 ID	0x88
初始化	0x90
信息存储	0xA0

地址寄存器用于指定 NAND Flash 读写等操作的目标地址, 包括列地址和行地址, 其中行地址又分为页地址和块地址。PS 端向命令寄存器写入指令编码前, 需要先向地址寄存器写入目标地址。状态寄存器用于指示每一种操作的执行状态, PS 端通过读状态寄存器判断每一种操作成功或失败。ID 寄存器 1 和 ID 寄存器 2 用于保存读 ID 操作读取的 NAND Flash 芯片 40 位 ID 号。

本文设计的 NAND Flash 控制器支持中断的工作方式, 每执行完一种操作就会产生中断信号, 中断信号连接至 PS 端, PS 端响应中断后可通过读状态寄存器消除中断。中断使能寄存器用于配置每一种操作是否启用中断。

PS 端通过 AIX_LITE 接口读写寄存器, 实现对 PL 端 NAND Flash 控制器的使用。这种方式便于应用层程序对 NAND Flash 的控制, 也易于 PL 端进行后续功能的扩展。

2.3 地址映射模块设计

地址映射的过程需要以映射前的地址为索引, 通过查询映射关系表, 得到映射后的地址。地址映射的过程符合 RAM (random access memory) 的工作方式, 即以地址为索引访问存储内容。BRAM (block RAM) 是 FPGA 内嵌的存储资源, 具有延迟周期确定、不占用逻辑资源的优点, 因此采用 BRAM 存储 Table_WL 以及 Table_BM 的地址

映射信息, 并通过 BRAM 的读写实现地址映射信息的更新。

考虑到使用的 NAND Flash 芯片共有 2 048 个存储块, 为了预留约 2% 的存储块用于替换坏块, 设定逻辑地址和中间逻辑地址的范围为 1~2 008, 物理地址范围为 1~2 048。分别使用两个位宽为 11 比特, 深度为 2 048 的 BRAM 存储两层地址映射信息表, 具体存储方式如图 3 所示。RAM_WL 存储第一层地址映射信息, 其中地址 0 保存单次上电结束时, 逻辑地址进行最后一次映射得到的中间逻辑地址 Last_Temp_Addr; 地址 1~2 008 保存逻辑地址对应的中间逻辑地址 Temp_Addr (1) ~ Temp_Addr (2 008)。RAM_BM 存储第二层地址信息, 地址 0 保存着单次上电结束时, 最后一次用于替换坏块的保留块地址; 地址 1~2 008 保存中间逻辑地址对应的物理 Phy_Addr (1) ~ Phy_Addr (2 008)。

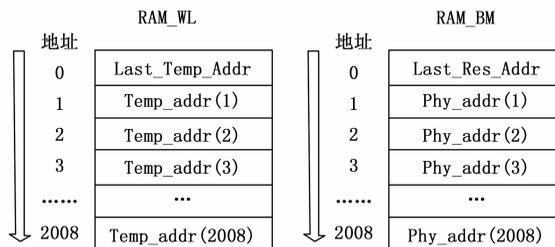


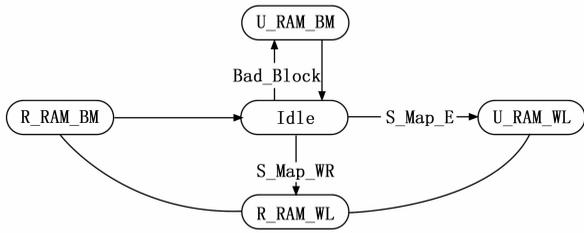
图 3 地址映射信息存储图

第二层地址映射用于实现坏块管理, 需要对芯片出厂时自带的坏块和使用过程中产生的坏块进行判断。系统使用的 NAND Flash 芯片每一个存储块首页的空闲区域 (spare area) 都有出厂坏块标志, 当读取坏块标志为 0x00 时, 表示此存储块为坏块^[18]。对于 NAND Flash 使用过程中出现的坏块, 需要使用 ECC 校验码进行判断。本文采用 NAND Flash 芯片内部自带的硬件 ECC 功能, 该 ECC 具有每 512 字节 5 比特错误检测、4 比特错误纠正的能力。在每次使用前, 需要发送相应指令使能芯片内部 ECC 功能。在使用过程中, 每完成一次读、写、擦除操作后, 通过读取芯片执行状态 (operation status) 来判断是否出现坏块。通过使用芯片内部硬件 ECC 功能, 可以减少 PL 端逻辑资源的消耗, 降低 NAND Flash 控制器设计的复杂度。

由于系统掉电后 RAM 存储的信息会丢失, 需要在系统上电结束前将 RAM_WL 和 RAM_BM 保存到非易失性存储器中。非易失性存储器可以选择在主控芯片外增加一小容量存储芯片, 如 E2PROM^[19]、非易失性随机铁电存储器^[6]等。由于系统使用的 NAND Flash 在使能内部 ECC 进行存储数据保护的前提下, 可以保证第 0 号存储块数据的正确性。因此在系统掉电前将 RAM_WL 和 RAM_BM 的信息存入 NAND Flash 芯片的第 0 块中, 在实现地址映射信息掉电不丢失的同时, 避免了增加存储芯片带来的硬件资源消耗。

RAM_WL 和 RAM_BM 的上电初始化过程以及掉电

前存入 NAND Flash 第 0 块的过程需要对 NAND Flash 进行读写, 这两个过程分别由时序逻辑控制模块的初始化功能和信息存储功能实现。地址映射模块使用状态机的方式进行设计, 状态转移如图 4 所示。根据前文所述地址映射方法原理, 地址映射模块工作过程如下: 1) 对 NAND Flash 进行擦除操作时, 时序逻辑控制模块发出 S_Map_E 信号, 地址映射模块更新 RAM_WL 信息, 然后以逻辑地址为索引访问 RAM_WL 得到对应的中间逻辑地址, 再以中间逻辑地址为索引访问 RAM_BM 得到物理地址。2) 对 NAND Flash 进行读、写操作时, 时序逻辑控制模块发出 S_Map_WR 信号, 地址映射模块通过读 RAM_BM 和 RAM_WL 得到逻辑地址对应的物理地址。3) 对 NAND Flash 进行读、写、擦除操作过程中出现坏块时, 时序逻辑控制模块发出 Bad_Block 信号, 地址映射模块更新 RAM_BM 信息。



注: Idle: 空闲状态; U_RAM_WL: 更新 RAM_WL; R_RAM_WL: 读 RAM_WL; R_RAM_BM: 读 RAM_BM; U_RAM_BM: 更新 RAM_BM。

图 4 地址映射模块状态转移图

2.4 时序逻辑控制模块设计

NAND Flash 时序逻辑模块用于实现 NAND Flash 读写等操作的复杂时序逻辑控制, 实现对 NAND Flash 的直接驱动。片外 NAND Flash 与 PL 端控制器的连接采用标准 ONFI 协议, 主要控制信号包含片选信号 CE、命令锁存信号 CLE、地址锁存信号 ALE、写使能信号 WE 以及读使能信号 RE, 另外有 8 位数据、命令、地址分时复用的 IO 信号^[20]。NAND Flash 各种操作的时序主要由异步地址输入、异步命令输入、异步数据输入、异步数据输出四种总线操作模式构成, 不同模式对应的控制信号变化如表 3 所示。

表 3 控制信号状态表

总线模式	CE	CLE	ALE	WE	RE
准备	H ¹⁾	X	X ³⁾	X	X
命令输入	L ²⁾	H	L	R ⁴⁾	H
地址输入	L	L	H	R	H
数据输入	L	L	L	R	H
数据输出	L	L	L	H	R ¹⁾

注: 1) 高电平; 2) 低电平; 3) 高电平或低电平; 4) 上升沿。

根据不同总线操作模式下控制信号的变化, NAND Flash 读写时序如图 5、图 6 所示。为了实现 NAND Flash 不同操作的时序驱动需求, 采用有限状态机的方法进行控

制逻辑设计^[21], 其中状态机状态转移如图 7 所示。基于 NAND Flash 的控制需求, 状态机实现了读页、页编程、擦除等基本控制操作, 以及初始化操作和信息存储操作的时序控制。其中初始化和信息存储操作分别用于地址映射信息的加载和存储。根据图 7 的状态转移过程, 不同操作的流程如下:

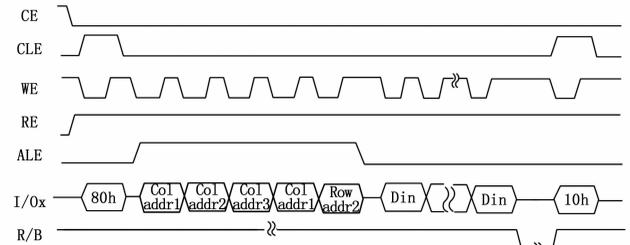


图 5 NAND Flash 页编程操作时序图

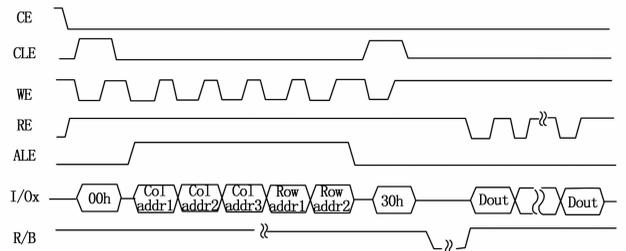


图 6 NAND Flash 读页操作时序图

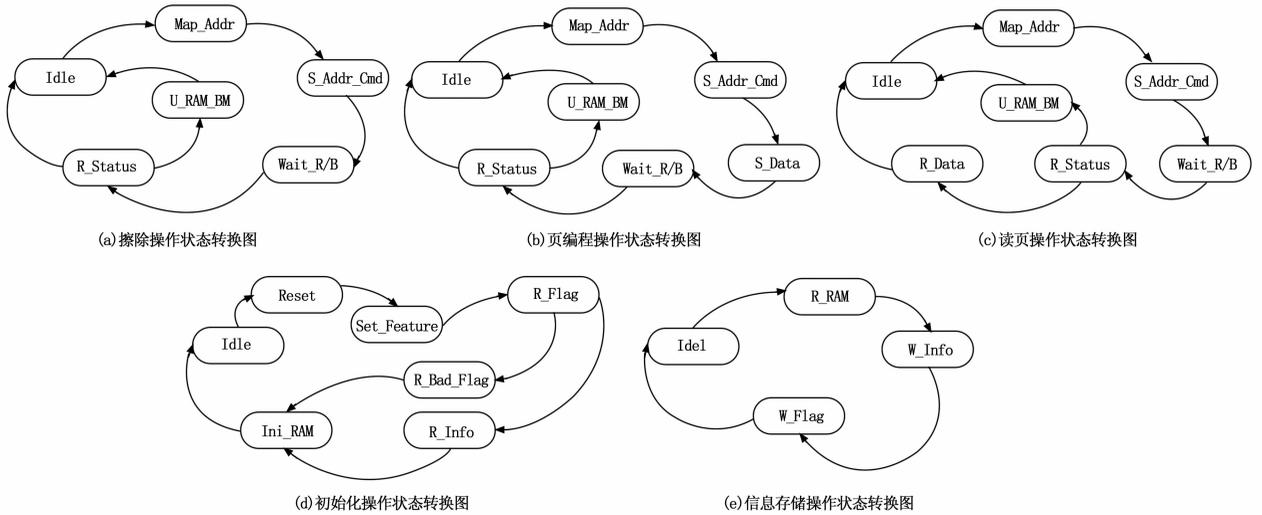
1) 读页操作: 首先通过地址映射模块得到物理地址, 接着发送读命令和读地址, 等待 R/B (忙闲信号) 信号拉高后发出读状态命令。通过状态编码判断读页操作是否成功, 若成功则读取一页数据, 若失败则通知地址映射模块更新 RAM_BM 信息。

2) 页编程操作: 通过地址映射得到待写存储块的物理地址, 接着发送写命令和写地址, 并写入一页数据。之后等待 R/B 信号拉高, 发出读状态命令来判断写页操作是否成功, 若失败则通知地址映射模块更新 RAM_BM 信息。

3) 擦除操作: 通过地址映射模块得到物理地址, 发送擦除命令和地址, 等待 R/B 信号拉高, 发出读状态命令来判断擦除操作是否成功, 若失败则通知地址映射模块更新 RAM_BM 信息。

4) 初始化操作: 发送复位命令复位 NAND Flash 芯片, 接着通过 Set_Feature 命令启动 NAND Flash 内嵌硬件 ECC 功能。读取初次启动标志 Flag 判断初始化方法, 如果是初次使用, 遍历第 1~2 008 个存储块的坏块标志位进行 RAM 表初始化; 若非初次使用, 读取 NAND Flash 第 0 块存储的地址映射信息, 将信息写入 RAM_WL 和 RAM_BM。系统每次上电使用 NAND Flash 之前, 都需要进行初始化操作。

5) 信息存储操作: 发送写命令和写地址, 读取 RAM_WL 以及 RAM_BM 的信息并写入 NAND Flash, 最后



注: Idle: 空闲状态; Map_Addr: 地址映射; S_Addr_Cmd: 发送地址、命令; Wait_R/B: 等待 R/B 信号拉高; R_Status: 读取状态; U_RAM_BM: 更新 RAM_BM; S_Data: 发送数据; R_Data: 接收数据; Reset: 闪存复位; Set_Feature: 设置工作模式; R_Flag: 读启动标志; R_Bad_Flag: 读坏块标志; R_Info: 读地址映射信息; Ini_RAM: 初始化 RAM; R_RAM: 读 RAM; W_Info: 写地址映射信息; W_Flag: 写启动标志。

图 7 时序逻辑控制模块状态转移图

写入初次启动标志 Flag, 完成地址映射信息存储。系统每次掉电之前, 都需要进行信息存储操。

2.5 DMA 数据传输模块设计

基于 NAND Flash 的结构, 通常将数据按页为单位进行写入和读出^[20]。将数据写入 NAND Flash 进行存储的过程中, 首先将待写入数据从 PS 端搬运到 PL 端的缓存中, 等待缓存满一页数据后 PS 端发出页编程操作指令, 写入数据; 读取 NAND Flash 存储数据的过程中, PS 端首先发送读页操作指令, 将数据读到 PL 端的缓存中, 完成一页数据读取后, 需要将数据搬运到 PS 端用于后续处理。对于 PS 端与 PL 端之间大容量数据的搬运, 采用 DMA (direct memory access) 的方式可以提高 PS 端 CPU 的工作效率。

采用 Xilinx 的 AXI_DMA 控制器 IP 核实现数据搬运。系统中 PS 端挂载 DDR3 芯片作为运行内存, 因此 DMA 控制器实现的是内存与 PL 端缓存之间的数据搬运。AXI_DMA 控制器共有三种数据通道, 每种通道的作用如下: 1) S_AXI_LITE 通道, 使用 AXI_LITE 接口协议, 连接到 PS 端的 M_AXI_GP 接口, 用于 DMA 控制器的功能配置以及初始化; 2) M_AXI_MM2S 通道、M_AXI_2SMM 通道, 均使用 AXI_FULL 接口协议, 连接到 PS 端的 M_AXI_GP 接口, 可通过 M_AXI_GP 接口直接访问内存, 分别用于从内存中读出数据和向内存写入数据; 3) S_AXIS_S2MM 通道、M_AXIS_MM2S 通道, 使用 AXI_STREAM 接口协议, 连接到 PL 端缓存, 分别用于从缓存中读出数据和向缓存写入数据。考虑到设计中使用的 NAND Flash 芯片一页大小为 2 048 字节, 采用位宽为 8 比特, 深度为 2048 的 FIFO 作为读写缓存, 并将 FIFO 接口设置为 AXI_STREAM 形式。

DMA 控制器写满一页缓存的速度快于将一页缓存的数据写入 NAND Flash 的速度, 将一页数据从 NAND Flash 读到缓存中的速度慢于 DMA 控制器将一页缓存数据写到内存的速度, 为了平衡缓存读写端口的速度差异, 采用乒乓操作的方式。对于页编程和读页操作, 设置两个大小相同的缓存, 分别为 WFIFO_1、WFIFO_2 以及 RFIFO_1、RFIFO_2。当 DMA 控制器向其中一个 WFIFO 写入数据时, 时序逻辑控制模块可以将数据从另外一个 WFIFO 中读出并写入 NAND Flash 中; 当时序逻辑控制模块从 NAND Flash 读出一页数据并存入其中一个 RFIFO 中, DMA 控制器可以将另一 RFIFO 的数据读出。为了实现数据读写时的乒乓操作, 需要控制读写缓存 FIFO 的切换。FIFO 接口使用 AXI_STREAM 协议, AXI_STREAM 协议的 TLAST 信号拉高表示一次数据传输完成, 因此通过检测 TLAST 信号的上升沿变化来控制缓存接口的切换, 具体电路结构如图 8 所示。

3 系统仿真与测试

3.1 仿真分析

3.1.1 仿真环境

基于 Vivado 平台, 编写 Testbench 对 PL 端 NAND Flash 控制器功能进行测试, 仿真测试模块功能划分如图 9 所示。其中 DUT (Design Under Test) 为待测模块, 即 PL 端 NAND Flash 控制器; NAND Flash MODEL 采用 Micron 公司提供的 MT29F 系列 Verilog 仿真模型, 能够模拟 MT29F2G08ABAEAH4 型 NAND Flash 的功能; CTRL (Control) 模块用于模拟 PS 端发送控制指令; BRAM 模块用于保存 NAND Flash 控制器待发送数据和读取数据。

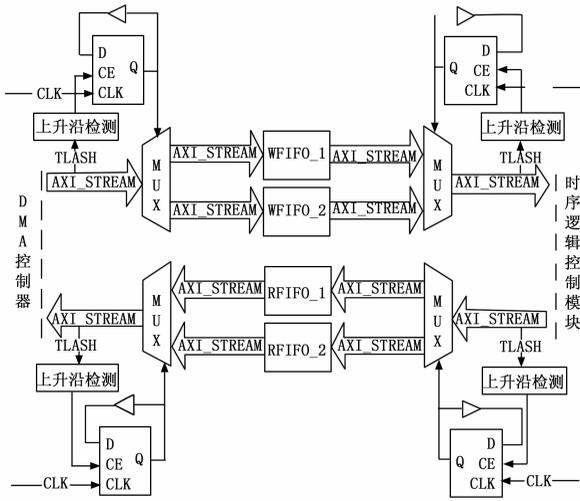


图 8 读写缓存切换电路图

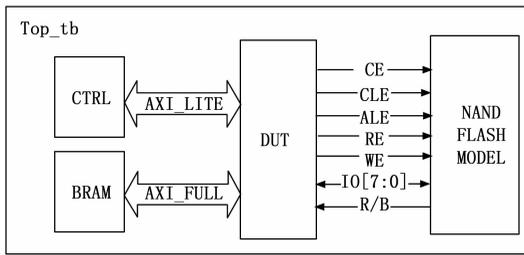


图 9 测试模块图

3.1.2 功能仿真

NAND Flash 控制器功能仿真如图 10 中各信号定义如下:

- 1) NAND Flash 控制信号: ce、ale、cle、re、we、rb、Flash_data_io。
- 2) RAM 表读写控制信号: 包括 RAM_WL_we、RAM_WL_addr、RAM_WL_din、RAM_WL_dout、RAM_BM_we、RAM_BM_addr、RAM_BM_din、RAM_BM_dout。
- 3) 地址信号: 包括 Logic_Addr (逻辑地址)、Temp_Addr (中间逻辑地址)、Phy_Addr (物理地址)。

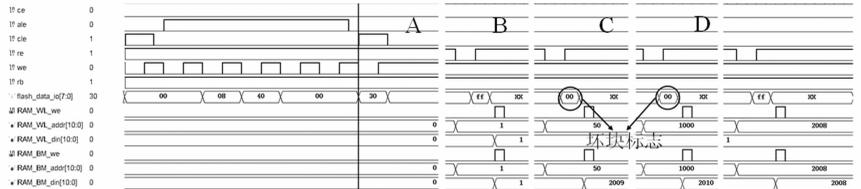
图 10 (a) 为地址映射信息初始化过程, 遍历 NAND Flash 的出厂坏块标志位, 进行地址映射信息表 RAM_WL 和 RAM_BM 的初始化。仿真过程中, 为了模拟芯片出厂时带有坏块的情况, 设置 NAND Flash 第 50 号存储块和第 1 000 号存储块的坏块标志位为 0x00。RAM_BM 初始化过程如下: 控制器发送读取每一个存储块首页的空闲区域的命令 (如区间 A、B 表示读第 1 号存储块的坏块标志), 若检测到坏块 (如区间 C、D 所示), 采用保留

区的存储块替换坏块, 并将好块一块块的映射关系写入 RAM_BM 中。RAM_WL 初始化过程中始终将逻辑地址映射到的中间逻辑地址设置为“1”。

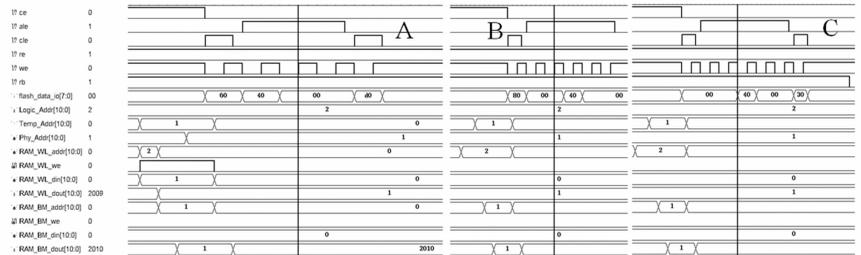
图 10 (b) 为 NAND Flash 控制器进行擦除、页编程、读页操作仿真, 三种操作的逻辑地址均设置为“2”。区间 A 为块擦除过程, 地址映射模块通过访问 RAM_WL 得到中间逻辑地址为“1”, 并更新 RAM_WL 的信息, 接着以中间逻辑地址为索引访问 RAM_BM 得到物理地址为“1”, 并向第 1 号存储块发出擦除命令。区间 B、C 分别为页编程、读页过程, 首先通过地址映射模块得到实际物理地址为“1”, 并向第 1 号存储块发出编程命令和读命令。

为了测试突发坏块的情况, 对物理地址为“1”的存储块进行编程操作时, 设置读取操作状态结果为失败, 即出现突发坏块, 仿真过程如图 10 (c) 所示。NAND Flash 控制器向物理地址为“1”的存储块发出编程指令 (如区间 A 所示)。完成存储数据发送后, 读取操作状态为 0xe1 (如区间 B 所示), 表示编程操作失败, 待编程的存储块为坏块。地址映射模块更新 RAM_BM, 使用保留区的好块地址“2010”替换坏块地址“1”, 实现坏块管理功能。

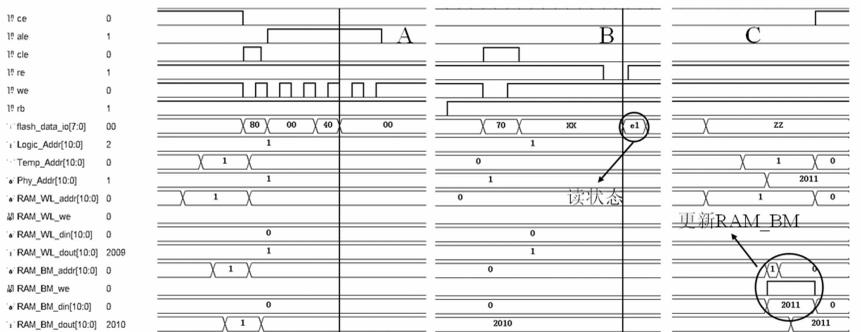
为了验证基于地址映射方法实现的磨损均衡功能的有效性, 模拟数据采集系统的工作情况, 每次从初始地址开始, 向一定数量的存储块写入数据。为了覆盖足够数量的存储块, 每次写入存储块的数量占 NAND Flash 总容量的



(a) 地址映射信息初始化仿真



(b) 块擦除、页编程、读页仿真



(c) 突发坏块仿真

图 10 功能仿真图

90%~100%，重复写入数据 1 万次。分别使用地址映射方法和直接控制法写入数据，NAND Flash 存储块磨损情况如图 11 所示。由仿真结果可见，相较于直接控制法，地址映射方法能够更为均匀地擦除存储块。这是因为每次进行数据写入时，地址映射方法将逻辑地址映射到序号递增的物理地址中，避免了序号靠后的存储块较少被写入数据的情况，达到存储块均衡使用的目的。

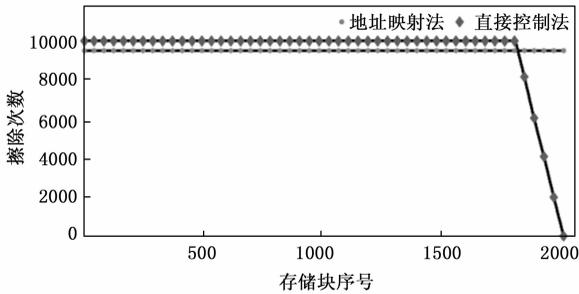


图 11 NAND Flash 存储块磨损情况

3.2 板级测试

将设计的 NAND Flash 控制器应用于某型浮空器的传感器数据采集系统，该系统通过多路串口、SPI 接口和 AD 接口实时获取压差传感器、温度传感器和电源电压数据，并存入 NAND Flash 中，系统程序基于裸机执行。为了验证 NAND Flash 控制器功能的正确性，在系统工作过程中，将以下信息通过串口发送到 PC 端存储：1) 存入 NAND Flash 的传感器数据；2) NAND Flash 擦除、编程操作的逻辑地址和物理地址；3) NAND Flash 擦除、编程操作的完成情况。PC 端接收到的部分数据如图 12 所示。数据采集完成后，将 NAND Flash 中存储的实验数据读出并与之前实时存储的数据进行对比，并统计每个存储块的擦除次数。

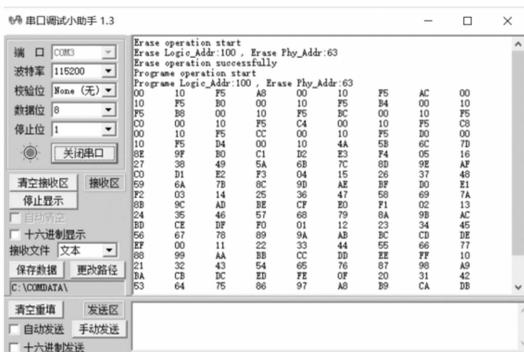


图 12 PC 端串口接收数据图

经过多次实验测试，数据采集系统写入 NAND Flash 的数据与从 NAND Flash 读出的数据均保持一致，存储块最大擦次数与最小擦除次数之差始终为 1。实验过程中未出现坏块，模拟坏块出现情况，对第 1 号存储块进行编程操作时设置编程状态为失败，经测试 NAND Flash 控制器能够使用保留区好块地址代替坏块地址。由此验证了基于地址映射方法的 NAND Flash 控制器可以在嵌入式数据采集

系统得到正确使用。

4 结束语

本文提出了一种基于地址映射的 NAND Flash 控制方法，通过建立、维护、查询逻辑地址到物理地址的映射信息表，实现嵌入式数据采集系统中 NAND Flash 的坏块管理和磨损均衡功能。相比于使用文件系统的 NAND Flash 控制方法，对硬件资源要求更低，更为简单。基于地址映射方法，本文设计了一款 NAND Flash 控制器，仿真和实际应用测试结果表明，控制器能够管理出厂坏块和突发坏块，均衡每个存储块的负载，提高系统的可靠性，在嵌入式数据采集领域具有一定的应用价值。

参考文献：

- [1] 高聪明, 石亮, 刘凯, 等. 闪存固态硬盘系统结构与技术 [J]. 计算机研究与发展, 2021, 58 (7): 1518-1532.
- [2] 孙晓磊, 王红亮, 陈航. 基于 FPGA 的双 Flash 数据记录器设计与实现 [J]. 电子测量技术, 2021, 44 (23): 36-41.
- [3] 任海, 刘伟亮, 唐振刚, 等. 一种高效的星载高速固态存储器坏块管理算法 [J]. 计算机测量与控制, 2018, 26 (1): 242-247.
- [4] 张鹏. NAND Flash 坏块管理算法研究与实现 [D]. 哈尔滨: 哈尔滨工业大学, 2015.
- [5] 姚英彪, 王发宽. 具有磨损均衡意识的混合固态硬盘 FTL 算法 [J]. 计算机学报, 2018, 41 (10): 2379-2393.
- [6] 李远哲, 贺海文, 万丽, 等. 嵌入式系统大容量 NAND Flash 存储器分区管理设计 [J]. 计算机测量与控制, 2019, 27 (2): 212-215, 220.
- [7] 龚有华, 魏德宝, 乔立岩, 等. 基于 Zynq 的 NAND Flash 存储系统研制 [J]. 电子测量技术, 2014, 37 (12): 53-57.
- [8] 李欣, 禹晔阳, 牛跃华, 等. 基于分级缓存加速的高可靠高速星载固存设计 [J]. 计算机测量与控制, 2019, 27 (10): 186-191.
- [9] 王柯, 姜一扬, 张黄鹏, 等. 基于 FPGA 的 NAND 闪存控制器的设计与验证 [J]. 计算机应用, 2018, 38 (S2): 254-257.
- [10] 张远航, 宋鹏程, 张亚朋, 等. 支持 ONFI 协议的 NAND Flash 控制器设计 [J]. 微电子学与计算机, 2017, 34 (5): 26-29.
- [11] 任勇峰, 刘晨晖, 李辉景. 基于 NANDFlash 的双备份数据存储器的设计与实现 [J]. 电子器件, 2017, 40 (4): 862-867.
- [12] 黄宏智, 周钦强, 黄飞龙. 新型自动气象站 NAND Flash 存储系统的设计与实现 [J]. 电子测量技术, 2016, 39 (12): 132-135.
- [13] 乔亚飞, 李华旺, 常亮, 等. 基于 FPGA 的星载 NAND-Flash 控制器的设计 [J]. 电子设计工程, 2018, 26 (14): 158-161, 166.
- [14] 陈玉标, 李建中, 高宏. 闪存数据管理研究综述 [J]. 智能计算机与应用, 2021, 11 (12): 201-207, 19.

(下转第 122 页)