

Analysis and Solution of Abnormal Fault of Data Storage Based on NAND-flash

Zhao Long^{*}, Shen Xiaohe, Chen Geng, Hu Xiaoxi

Beijing Institute of Aerospace Control Devices, Beijing, China

Email address:

15611796238@163.com (Zhao Long), shenxiaohe2008@163.com (Shen Xiaohe), chengeng191@163.com (Chen Geng),

HuXX615@126.com (Hu Xiaoxi)

^{*}Corresponding author

To cite this article:

Zhao Long, Shen Xiaohe, Chen Geng, Hu Xiaoxi. Analysis and Solution of Abnormal Fault of Data Storage Based on NAND-flash. *Science Discovery*. Vol. 8, No. 4, 2020, pp. 69-73. doi: 10.11648/j.sd.20200804.12

Received: June 18, 2020; **Accepted:** August 18, 2020; **Published:** August 25, 2020

Abstract: NAND-flash memory has the advantages of large capacity and fast rewriting speed. It is suitable for the storage of large amounts of data and is often used as an online storage device for embedded products. However, NAND-flash has the problems of bad blocks and other insufficient reliability. In a certain aerospace model temperature/strain measurement system, the NAND-flash memory is erased and written through DSP software, and the strain/temperature measurement data collected by the sensor is stored in real time. After the system was powered on many times and completed data collection and decoding, it was found that the data stored in the NAND-flash had abnormal faults. By analyzing the test phenomenon and failure mechanism, the failure problem is attributed to the fact that the data in the NAND-flash is not erased, and it is coupled with the newly written data, and it is finally located because the data in the original data address cannot be effectively erased when the data address is stored. Coupling also occurred, resulting in an error in the data address, and the data was coupled after power-on again. Based on the above-mentioned reasons, this paper proposed a troubleshooting method and conducted a test. The verification was successfully passed and the problem was resolved. This method has high reference significance in the large-capacity and high-reliability data storage of NAND-flash in the field of aerospace models.

Keywords: NAND-flash, Data Exception, Fault Analysis

基于NAND-flash的数据存储异常故障分析及解决方法

赵龙^{*}, 申小禾, 陈赓, 胡晓曦

北京航天控制仪器研究所, 北京, 中国

邮箱

15611796238@163.com (赵龙), shenxiaohe2008@163.com (申小禾), chengeng191@163.com (陈赓),

HuXX615@126.com (胡晓曦)

摘要: NAND-flash存储器具有容量较大、改写速度快等优点, 适用于大量数据的存储, 经常作为嵌入式产品的线上存储设备, 但NAND-flash存在坏块及其他可靠性不足的问题。某航天型号温度/应变测量系统中, 通过DSP软件对NAND-flash存储器进行擦除、写入等操作, 实时存储传感器采集得到的应变/温度测量数据。在系统多次上电并完成数据采集和解码后, 发现NAND-flash中存储的数据存在异常故障。通过分析试验现象和故障机理, 将故障问题归结为NAND-flash中数据未擦除, 与新写入数据发生耦合, 最终定位于由于储存数据地址时未能有效擦除原先数据地址中的数据, 也发生了耦合, 导致数据地址出错, 从而重新上电后数据耦合, 根据上述原因本文提出了故障解决方法并进行了试验, 顺利通过了验证, 问题得到解决。该方法在航天型号领域NAND-flash的大容量、高可靠性数据存储方面具有较高的参考意义。

关键词：NAND-flash，数据异常，故障分析

1. 引言

某航天型号温度/应变测量系统中，通过采用光纤光栅传感器，能够同时完成96路以上温度/应变的实时测量，在测量过程中产生了大量的数据，需要一种能够快速可靠的存储方案保证数据的正确性和完整性。在系统设计时，采用NAND-flash作为数据的存储器。针对NAND-flash使用的实践可靠性，经过研究发现，NAND-flash在使用过程中存在坏块、擦除失败等可靠性不足的问题，众多研究人员提出了针对此类问题的可靠性设计方法和措施，解决了部分可靠性问题，但在实际使用过程中仍然存在不够全面，问题覆盖性不足的问题[1-4]。在该航天型号系统产品设计师，集合了上述研究成果及设计思路，但在进行系统测试时，通过多次上电并完成数据采集和解码，并对数据进行分析后发现，部分数据仍出现乱码的情况，且该情况出现具有一定的概率性和偶然性，出现的时机为上电后的一段时间，往后数据即恢复正常。因数据异常导致温度/应变测量结果的不正确性，会导致总体和设计师误判系统的安全性和可靠性。因此，分析NAND-flash中存储的数据异常故障的产生机理并采取有效措施实现对系统温度/应变的正确测量在实际中有着重要意义。

2. 数据存储的可靠性设计

Flash闪存是非易失存储器，可以对称为块的存储器单元块进行擦写和再编程[5]。任何flash器件的写入操作只能在空或已擦除的单元内进行，所以大多数情况下，在进行

写入操作之前必须先执行擦除。擦除NAND器件是以8~32KB的块进行的，执行相同的操作最多只需要4ms，而擦除NOR器件时是以64~128KB的块进行的，执行一个写入/擦除操作的时间为5s，所以一般在数据存储领域采用NAND-flash[6]。NAND-flash存储器是flash存储器的一种，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存储器具有容量较大、改写速度快等优点，适用于大量数据的存储，因而在业界得到了越来越广泛的应用，如嵌入式产品中，包括数据相机、MP3随身听记忆卡、体积小巧的U盘等[7]~[12]。在某航天型号温度/应变测量系统的设计和软件编程过程中，开展了一系列数据存储的可靠性设计，主要采取以下措施和方案。

2.1. 坏块的可靠性处理

NAND-flash非常容易出现坏块，且NAND器件中的坏块是随机分布的，需要对介质进行初始化扫描以发现坏块，并将坏块标记为不可用。针对NAND-flash中的坏块，通过对全盘进行擦除过程中，进行有效性校验和检测，如存在坏块情况即将坏块地址存储于flash的某固定区域，在下次上电初始化时直接读取该固定区域获得全部的坏块地址，在进行数据存储时，判断是否为坏块，如果是则跳过，不将数据存入该坏块中，以此保证数据存储的可靠性。具体技术实现如图1所示，坏块badpage1和坏块badpage2的地址分别存储于badpageaddr中，在系统上电启动时，从badpageaddr中读取坏块的地址，从而在数据存储过程中将其跳过。



图1 坏块可靠性处理示意图。

2.2. 写入操作的可靠性设计

NAND-flash以块为单位进行擦除操作。Flash的写入操作必须在空白区域进行，如果目标区域已经有数据，必须先擦除后写入，因此擦除操作是闪存的基本操作。该系统通过擦除指令，能够保证全盘的一次性擦除，同时，对是否擦除成功进行了校验，保证擦除的完整性 [13]。

2.3. 多次上电的可靠性设计

因系统需要多次上电使用，为了保证上一次上电的数据不被本次上电存储的数据破坏，系统采用了双地址交叉

存储上一次上电已到达NAND-flash的块地址，在系统上电后，读取该地址确定目前已存储的块，同时，因存储的块地址有一定的延后性，在读取该地址后自动隔离一个存储周期的地址长度，作为本次上电的初始块地址，以保证上一次最后的块地址和本次初始块地址不存在交叉的情况。具体技术实现如图2所示，系统上电后从块地址存储区中读取交叉存储的已到达NAND-flash的块地址，比较addr1和addr2的大小，取较大值，并加上一个存储周期的地址长度，作为下一次上电的初始块地址。



图2 多次上电的可靠性设计。

3. 故障现象和原因分析

3.1. 故障现象

温度/应变测量系统多次上电并完成数据采集和解码，并对数据进行分析后发现，部分数据出现乱码的情况，且

该情况出现具有一定的概率性和偶然性，出现的时机为上电后的一段时间，往后数据即恢复正常。上电初始异常数据见图3，因上电后时间尚未同步，故输出时间应为初始化时间（2019-1-10:0:0），但输出数据时间显示为（2019-0-00:0:0），数据异常。

31388	2019-12-6 13:47:36	932	229597	30.4375	30.75	CF01F81	CH1:	1533.655	1540.7	1547.982	1554.652	1559.421	1564.631
31389	2019-12-6 13:47:36	982	229598	30.4375	30.75	11401F81	CH1:	1533.654	1540.701	1547.982	1554.652	1559.421	1564.631
31390	2019-12-6 13:47:37	32	229599	30.4375	30.75	15901F81	CH1:	1533.654	1540.7	1547.982	1554.652	1559.421	1564.632
31391	2019-12-6 13:47:37	82	229600	30.4375	30.75	19E01F81	CH1:	1533.654	1540.7	1547.982	1554.653	1559.421	1564.632
31392	2019-0-0 0:0:0	288	1	20	20	2081	CH1:	1525	1525	1525	1525	1525	1525
31393	2019-0-0 0:0:0	346	2	20	20	4502081	CH1:	1525	1525	1525	1525	1525	1525
31394	2019-0-0 0:0:0	384	3	20	20	8A02081	CH1:	1525	1525	1525	1525	1525	1525
31395	2019-0-0 0:0:0	386	4	20	20	CF02081	CH1:	1525	1525	1525	1525	1525	1525
31396	2019-0-0 0:0:0	496	5	20	20	11402081	CH1:	1525	1525	1525	1525	1525	1525
31397	2019-0-0 0:0:0	546	6	20	20	15902081	CH1:	1525	1525	1525	1525	1525	1525
31398	2019-0-0 0:0:0	592	7	20	20	19E02081	CH1:	1525	1525	1525	1525	1525	1525
31399	2019-0-0 0:0:0	642	8	20	26.25	2181	CH1:	1525	1525	1525	1525	1525	1525
31400	2019-0-0 0:0:0	696	9	26.1875	26.25	4502181	CH1:	1525	1525	1525	1525	1525	1525

图3 上电初始异常数据。

上电一段时间后，数据恢复正常，具体见图4所示，数据恢复正常过程中出现了数据的异常跳动。

36085	2019-12-6 9:36:43	142	4630	26.3125	26.25	CF0BE83	CH1:	1525	1525	1525	1525	1525	1525
36086	2019-12-6 9:36:43	248	4631	26.3125	26.25	1140BE83	CH1:	1525	1525	1525	1525	1525	1525
36087	2019-12-6 9:36:43	554	4632	26.3125	26.25	1590BE83	CH1:	1525	1525	1525	1525	1525	1525
36088	2019-12-6 9:36:43	604	4633	26.3125	26.25	19E0BE83	CH1:	1525	1525	1525	1525	1525	1525
36089	2019-12-6 9:36:43	654	4634	26.3125	26.25	0BF83	CH1:	1525	1525	1525	1525	1525	1525
36090	2019-12-6 10:04:18	432	4635	28.4375	28.75	450BF83	CH1:	1525	1525	1525	1525	1525	1525
36091	2019-12-6 10:04:18	528	4636	28.4375	28.75	8A0BF83	CH1:	1525	1525	1525	1525	1525	1525
36092	2019-12-6 10:04:19	840	4637	28.4375	28.75	CF0BF83	CH1:	1525	1525	1525	1525	1525	1525
36093	2019-12-6 14:46:40	964	4702	28.4375	28.8125	1140BF83	CH1:						
36094	2019-12-6 14:46:41	14	4703	28.4375	28.8125	1590BF83	CH1:						
36095	2019-12-6 14:46:41	64	4704	28.4375	28.8125	19E0BF83	CH1:						
36096	2019-12-6 14:46:41	114	4705	28.4375	28.8125	0C083	CH1:						
36097	2019-12-6 14:46:41	164	4706	28.4375	28.8125	450C083	CH1:						

图4 上电一段时间后从异常恢复正常数据。

3.2. 原因分析

通过观察存储的时间信息，可以看出月和日均为0的情况，但通过进行代码走查判断，该情况在逻辑中不应该出现，因软件在初始化阶段即对月和日变量进行了初始化（1月1日），且在软件运行过程中完全不存在清零的过程，除非在时间同步时输入了错误的时间信息，但由于时间输入是通过读取计算机系统时间的方式进行，不会出现日和月

为0的情况。通过数据产生的可能性分析：可能是因为两组数据耦合，导致时间信息数据异常（NAND-flash特性为数据耦合可以使得存储的数据位从1变为0，但不能从0变成1，耦合方式可以认为两组数据进行了相与（&）操作）。

表1是对帧计数进行上下两次通电的数据耦合分析，可以看出从第1帧到第4701帧均完美耦合，与存储的flash数据完全一致。

表1 帧计数上下两次通电的数据耦合分析。

序号	flash中帧计数	写入帧计数	写入帧计数（16进制）	flash中帧计数（16进制）	读取的帧计数（16进制）
	229601	1	1	380E1	1
	229602	2	2	380E2	2
	229603	3	3	380E3	3
	229604	4	4	380E4	4
	229605	5	5	380E5	5
	229606	6	6	380E6	6
	229607	7	7	380E7	7
	229608	8	8	380E8	8
	229609	9	9	380E9	9
	229610	10	A	380EA	A
	229611	11	B	380EB	B
	229612	12	C	380EC	C
	229613	13	D	380ED	D
	229614	14	E	380EE	E
	229615	15	F	380EF	F
	229616	16	10	380F0	10
	229617	17	11	380F1	11
	229618	18	12	380F2	12
	229619	19	13	380F3	13
	229620	20	14	380F4	14

序号	flash中帧计数	写入帧计数	写入帧计数（16进制）	flash中帧计数（16进制）	读取的帧计数（16进制）
	229621	21	15	380F5	15
	229622	22	16	380F6	16
	229623	23	17	380F7	17
	229624	24	18	380F8	18
	229625	25	19	380F9	19
	229626	26	1A	380FA	1A
	229627	27	1B	380FB	1B
	229628	28	1C	380FC	1C
	229629	29	1D	380FD	1D
	229630	30	1E	380FE	1E
	229631	31	1F	380FF	1F
	229632	32	20	38100	0
	229633	33	21	38101	1
...
4701	234301	4701	125D	3933D	121D

通过对时间信息和帧数据信息进行数据分析,可以得出以下结论:数据异常故障的原因是由于上下两次上电存储的部分数据进行耦合的原因导致的。该区域flash块在上一次断电前已写数,系统重新上电后,该区域又被重新写入数据,导致前后两次数据进行了耦合。

系统在进行可靠性设计时,已充分考虑了多次上电可能导致的块地址中数据重复写入导致数据的不正确性的情况,且在下次上电后自动隔离一个存储周期的地址长度,保证了NAND-flash存储器中数据存储的冗余。

通过新增代码,将存入到块地址存储区中的addr1和addr2实时读出并通过串口往外发送,保存数据进行分析后发现,addr1和addr2数据未按照设计要求的一个存储周期的地址长度递增,与上述存储的温度/应变数据出现的情况类似,出现了乱码的情况,同样可以发现该数据是由于前后连续两次存储的数据进行耦合的原因导致的。

上电后采集的温度/应变数据			上电初始地址		addr1	addr2
addr2b	addr1	addr2a				块地址存储区

图5 地址数据。

由于addr1和addr2数据的错误,可以根据图3进行以下分析,addr2a表示addr1下一个存储的地址数据,addr2b表示addr1上一个存储的地址数据,addr2a和addr2b均存于块地址的addr2区域,对addr2b擦除后,存入addr2a,同时他们之间的关系为:

$$\text{addr2a} = \text{addr1} + \text{一个存储周期的地址长度} \quad (1)$$

$$\text{addr2b} = \text{addr1} - \text{一个存储周期的地址长度} \quad (2)$$

由于addr2b与addr2a数据耦合,耦合得到的数据 $\text{addr2} < \text{addr2b} < \text{addr1}$ (flash数据耦合可以使得存储的数据位从1变为0,但不能从0变成1),故而导致原逻辑判断下一次上电的初始块地址变成addr1+一个存储周期的地址长度,与设计期望的addr2a+一个存储周期的地址长度少了一个存储周期的地址长度,导致存储的部分温度/应变数据耦合。

4. 解决措施及试验验证

为了避免出现存储的温度/应变数据的异常故障的出现,关键在于解决前后两次连续通电的数据耦合问题,也就是解决上电初始块地址的正确性问题。

针对上述问题,主要提出了两种解决措施。第一种,增加NAND-flash块地址存储区存储数据前进行擦除的等待时间,由原芯片手册中推荐使用的typicaltime改为maxtime,保证具有充分的时间将存储区中的内容擦除。第二种,保证块地址存储区写入数据的正确性校验功能,

数据地址写入后立即读出,与写入数据进行比较,如数据不一致,重新进行写入操作,以上循环进行2~3次,充分保证写入数据的正确性。

为了进一步增加NAND-flash温度/应变数据存储的可靠性,在系统上电确定初始块地址后,从初始块地址中读取数据,判断初始块地址是否为空或擦除状态(判断数据是否为0xff),如不为空和擦除状态,则初始块地址往后延一个固定长度,重新判断初始块地址的状态,以此循环,直至初始块地址为擦除状态。

将改进后的系统多次上电,并将存入到块地址存储区中的addr1和addr2实时读出并通过串口往外发送,进行数据分析后发现addr1和addr2数据按照设计要求的一个存储周期的地址长度递增,不再出现数据异常情况,同时,NAND-flash存储的温度/应变数据也未出现数据异常情况,该问题得到解决。

5. 结论

凭借着存储密度大和存储速率高的特点,基于NAND-flash的大容量存储器在航天型号领域得到了广泛的应用。但是由于NAND-flash本身存在缺陷,包括坏块、擦除和写入异常等情况,依靠NAND-flash本身无法有效保证航天型号数据存储的高可靠性要求。本文针对航天型号温度/应变测量系统中出现的数据异常情况,分析了基于NAND-flash的数据存储异常故障机理及解决措施并进行了试验验证。该方法在航天型号领域大容量、高可靠性数据存储方面具有较高的参考意义。

参考文献

- [1] 乔立岩, 李元亨, 王戈, 等. 一种NAND Flash坏块管理优化方法的实现[J], 电子测量技术, 2015.07。
- [2] 彭兵, 步凯, 徐欣, NAND Flash坏块管理研究[J], 微处理机, 2009.02。
- [3] 周军, NAND Flash的坏块管理设计[J], 单片机与嵌入式系统应用, 2010.09。
- [4] 舒文丽, 吴云峰, 赵启义, 孙长胜, NAND Flash存储的坏块管理方法[J], 电子器件, 2011.05。
- [5] Samsung, datasheet: 256MX8Bit/64MX16BitNAND Flashmemory [S], Samsung Electronics, Revision 0.2, March 2003.
- [6] Samsung. NAND Flash spare assignment recommendation [S], <http://www.Samsung.com>, 2003.
- [7] 蔡锦达, 王德福, 大容量Nand Flash在ARM嵌入式系统中的应用[J], 工业控制计算机. 2004, 06。
- [8] 谢民, 高梅国, 王超, NAND型Flash在大容量存储回放系统中的应用[J], 电子技术应用, 2006.4。
- [9] 高文青, 沈维聪, NAND存储器在嵌入式系统中的应用[J], 电测与仪表, 2008.02。
- [10] 高怡祯, 基于闪存的星载大容量存储器的研究和实现[J], 集成电路应用 2003.08。
- [11] 张伟功, 邱庆林, 荣金叶, 高可靠性大容量星载存储器的设计与研究[J], 第三次会议, 2010.08。
- [12] 李进, 金龙旭, 韩双丽, 李国宁, 王文华, 间图像存储器 NAND Flash的可靠性[J], 光学精密工程, 2012.05。
- [13] 沈浩, 付宇卓, NAND Flash存储控制器的软硬件划分设计[J], 计算机工程, 2004.24。