

(12) 发明专利

(10) 授权公告号 CN 101419838 B

(45) 授权公告日 2011. 01. 19

(21) 申请号 200810165716. 1

(22) 申请日 2008. 09. 12

(73) 专利权人 中兴通讯股份有限公司

地址 518057 广东省深圳市南山区高新技术产业园科技南路中兴通讯大厦法务部

(72) 发明人 官华伯

(51) Int. Cl.

G11C 16/06 (2006. 01)

(56) 对比文件

US 5341339 A, 1994. 08. 23, 全文.

US 6292012 B1, 2001. 09. 18, 全文.

CN 1845082 A, 2006. 10. 11, 全文.

审查员 冯慧萍

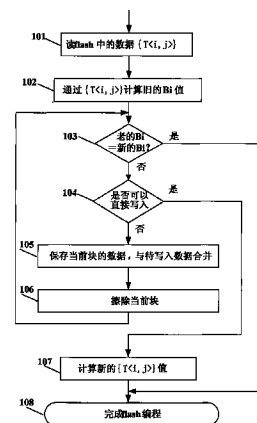
权利要求书 1 页 说明书 5 页 附图 2 页

(54) 发明名称

一种提高 flash 使用寿命的方法

(57) 摘要

本发明通过将待写入数据比特 Bi 转换为 n 比特数据 {T<i, j>}, 更新 flash 中比特串数据 {T<i, j>} 合适的值为 1 的比特为 0, 映射后得到待写入的 Bi 值, 从而无需进行擦除操作而直接写入 flash, 既避免 flash 在编程方面的限制, 提高更新 flash 中数据的速度, 又可以极大地减少擦除 flash 块的次数, 大幅度提高 flash 的寿命。



1. 一种提高 flash 使用寿命的方法,包括如下步骤:

读 flash 中的数据 $\{T\langle i, j \rangle\}$ 根据映射关系计算得到对应的老的原始数据 B_i , 其中 i 为自然数, $j = (n-1) \sim 0$, n 为偶数;

比较待写入数据 B_i 与老的原始数据 B_i 是否相同, 如果相同, 则 flash 中此字节的数据不需要更新;

如果待写入数据 B_i 值与老的原始数据 B_i 值不相同, 检查是否可以直接写入 flash;

如果可以, 则直接写入 flash; 如果不可以直接写入 flash, 执行擦除块操作。

2. 如权利要求 1 所述的方法, 其特征在于, 检查是否可以直接写入 flash 的步骤如下: 将数据 $\{T\langle i, j \rangle\}$ 中某些值为 1 的比特变为 0, 然后根据映射关系计算得到待写入数据 B_i 值, 则可以直接写入 flash, 否则, 不能直接写入 flash。

3. 如权利要求 1 或 2 所述的方法, 其特征在于, 如果 Flash 块中数据, 从上次擦除后直接写入的次数小于等于 n 次, 则可以直接写入 flash。

4. 如权利要求 3 所述的方法, 其特征在于, 直接写入 flash 的步骤具体为: 将数据 $\{T\langle i, j \rangle\}$ 中一位或多位为 1 的比特变为 0, 使得新的 $\{T\langle i, j \rangle\}$ 值映射后得到待写入数据 B_i 值, 则完成 flash 的直接写入。

5. 权利要求 1 或 4 所述的方法, 其特征在于, 映射关系指原始数据与 Flash 中的数据 $\{T\langle i, j \rangle\}$ 的对应关系, 原始数据比特取值范围为: $[0, 1]$, Flash 中的数据 $\{T\langle i, j \rangle\}$ 取值范围为: $[0, 2^n - 1]$, 映射关系用于确定比特串各比特数据经过运算后对应的原始数据比特值是 0 还是 1。

6. 如权利要求 5 所述的方法, 其特征在于, 数据 $\{T\langle i, j \rangle\}$ 中所有比特位进行相加, 确定对应的 B_i 值。

7. 如权利要求 5 所述的方法, 其特征在于, 把所述数据 $\{T\langle i, j \rangle\}$ 均分为两串或四串, 按位进行同或 / 异或运算确定对应的 B_i 值。

8. 如权利要求 1 所述的方法, 其特征在于, 在擦除块之前, 保存块中数据到缓冲区, 并与待写入数据合并。

9. 如权利要求 1 所述的方法, 其特征在于, flash 中的没有数据时, 按照映射关系将待写入数据比特直接转换为 n 比特的数据 $\{T\langle i, j \rangle\}$ 写入 flash。

一种提高 flash 使用寿命的方法

技术领域

[0001] 本发明涉及数据存储领域,具体涉及一种提高 flash 使用寿命的方法。

背景技术

[0002] Flash 存储器由于具有掉电可保存数据的特点,在数据存储领域得到了广泛的应用。Flash 在读数据方面没有什么限制,但在写数据方面有很多限制。一般 flash 都以 512bytes ~ 128Kbyte 或其他大小为单位构成一个块。写操作分为两种:编程、擦除。编程操作以字节为单位,可以把字节内的每个比特从 1 变为 0,或者维持原来值,但不能把为 0 的比特变为 1。擦除操作以块为单位,把块内的所有比特变为 1。但是 Flash 存储器有寿命方面的限制,每个块的擦除次数一般不能超过 10 万次。

[0003] 针对普通的写操作不能把为 0 的比特变为 1 的限制,目前主要采用的方法是:先读出块内原先保存的数据到 RAM 缓冲区,修改为新的数据;然后擦除原来的块;再把新的数据写入。这种方法的缺点在于:即使只有一个比特的数据需要从 0 修改为 1,也需要读写整个块的数据,并且需要擦除整个块一次。一般读 flash 的时间是纳秒级,编程 flash 的时间是微秒级,而擦除 flash 的时间则为毫秒甚至秒级。因此,这种方法既增加了操作时间,又缩短了 Flash 的寿命。

[0004] 另外一种方法是:把写 flash 的数据缓冲在 RAM 缓冲区中,一段时间后,再集中写到 flash 中。这种方法,可以避免每次修改都花很长时间更新 flash 中的内容。这种方法仍然无法克服编程时不能把为 0 的比特修改为 1 的限制,只是可以一定程度缓解此问题。同时引入了另外一个问题,即在缓冲数据的这段时间,如果系统掉电,则未写入 flash 的所有数据都会丢失。

[0005] 针对 flash 擦除次数不能超过 10 万次左右的限制,目前通用的方法为磨损均衡。如果需要修改某 flash 块的数据,假设所修改数据对应块号为 A,此时,先在整个 flash 中查找擦除次数最少的块 B;如果 B 块有数据,则在空闲块中查找擦除次数最少的块 C,将 B 块数据复制到 C 块并擦除 B 块使其成为空闲块;再将 A 块中的数据读到 RAM,修改为新的数据,写到 B 块相应扇区;最后擦除 A 块。上述处理过程中,如果 B 块没有数据,可以把新的数据直接写入 B 块。采用磨损均衡方法更新一个 flash 块的数据,需要进行两次读块、写块、擦除块操作,操作时间很长。

发明内容

[0006] 本发明要解决的技术问题是提供一种提高 flash 使用寿命的方法,可以减少擦除 flash 块的次数,大幅度提高 flash 的寿命。

[0007] 本发明提供了一种提高 flash 使用寿命的方法,包括如下步骤:

[0008] 读 flash 中的数据 $\{T<i, j>\}$ 根据映射关系计算得到对应的老的原始数据 B_i ,其中 i 为自然数, $j = (n-1) \sim 0$, n 为偶数;

[0009] 比较待写入数据 B_i 与老的原始数据 B_i 是否相同,如果相同,则不需要更新 flash

中此字节的数据；

[0010] 如果待写入数据 B_i 值与老的原始数据 B_i 值不相同,检查是否可以直接写入 flash；

[0011] 如果可以,则直接写入 flash；如果不可以直接写入 flash,执行擦除块操作。

[0012] 进一步地,如果可以将数据 $\{T\langle i, j \rangle\}$ 中某些值为 1 的比特变为 0,然后根据映射关系计算得到待写入数据 B_i 值,则可以直接写入 flash。

[0013] 进一步地,如果 Flash 块中数据,从上次擦除后直接写入的次数小于等于 n 次,则可以直接写入 flash。

[0014] 进一步地,将数据 $\{T\langle i, j \rangle\}$ 中一位或多位为 1 的比特变为 0,使得新的 $\{T\langle i, j \rangle\}$ 值映射后得到待写入数据 B_i 值,则完成 flash 的直接写入。

[0015] 进一步地,映射关系指原始数据与 Flash 中的数据 $\{T\langle i, j \rangle\}$ 的对应关系,原始数据比特取值范围为 $: [0, 1]$, Flash 中的数据 $\{T\langle i, j \rangle\}$ 取值范围为 $: [0, 2^n - 1]$, 映射关系用于确定数据 $\{T\langle i, j \rangle\}$ 各比特数据经过运算后对应的原始数据比特值是 0 还是 1。

[0016] 进一步地,flash 中的没有数据时,按照映射关系将待写入数据比特转换为 n 比特的数据 $\{T\langle i, j \rangle\}$ 写入 flash。

[0017] 本发明通过将待写入数据比特 B_i 转换为 n 比特数据 $\{T\langle i, j \rangle\}$,更新 flash 中 $\{T\langle i, j \rangle\}$ 合适的值为 1 的比特为 0,映射后得到待写入的 B_i 值,从而无需进行擦除操作而直接写入 flash,既避免 flash 在编程方面的限制,提高更新 flash 中数据的速度,又可以极大地减少擦除 flash 块的次数,大幅度提高 flash 的寿命。

附图说明

[0018] 图 1 是写入 flash 的处理流程图；

[0019] 图 2 是一种 flash 中数据变化的状态图；

[0020] 具体实施方式

[0021] 下面结合具体实施方式对本发明所述方法作进一步的描述。

[0022] 本发明提出的一种提高 flash 使用寿命的方法,具体如下：

[0023] 将待写入数据比特 B_i 转换为 n 比特数据 $\{T\langle i, j \rangle\}$ 写入 flash,其中,其中 i 为自然数, $j = (n-1) \sim 0$,选择 n 为偶数,或者记为 $\{T\langle i, j \rangle\}_{j=n-1 \sim 0}$,以下用符号 $\{T\langle i, j \rangle\}$ 表示。

[0024] 原始数据的比特数据 B_i 对应转换后的数据为：

[0025] $T\langle i, n-1 \rangle, T\langle i, n-2 \rangle \cdots T\langle i, 1 \rangle, T\langle i, 0 \rangle$,以下用符号 $\{T\langle i, j \rangle\}$ 表示。 B_i 与 $\{T\langle i, j \rangle\}$ 之间存在一对多的映射关系。

[0026] 当 flash 中已有数据,则读 flash 中的数据 $\{T\langle i, j \rangle\}$ 根据映射关系计算得到对应的老的原始数据 B_i ；比较待写入数据 B_i 与老的原始数据 B_i 是否相同,如果相同,则 flash 中此字节的数据不需要更新；如果待写入数据 B_i 值与老的原始数据 B_i 值不相同,检查是否可以直接写入 flash；如果可以则直接写入 flash,如果不可以直接写入 flash,执行擦除块操作后再写入。

[0027] 结合附图 1 所示,具体包含如下步骤：

[0028] 101. 从 flash 中读出原始数据 B_i 对应的 $\{T\langle i, j \rangle\}_{j=n-1 \sim 0}$ 。

[0029] 102. 根据映射规则, 计算 $\{T\langle i, j \rangle\}_{j=n-1 \sim 0}$ 对应的当前原始数据 B_i 。计算的目的是, 是利用统计规则, 减少写入 flash 的次数。

[0030] 103. 比较老的原始数据 B_i 与新的待写入数据 B_i 。根据统计规则, 每次写入一字节的原始数据时, 其每比特值发生改变的概率只有 50%。从 flash 中读回 $\{T\langle i, j \rangle\}$ 值, 根据映射关系, 计算得到 B_i 值。如果与当前新的 B_i 值相等, 则不需要进行写 flash 的操作。否则, 需要继续 104 处理。

[0031] 104. 检查是否可以直接写入 flash。如果可以将数据 $\{T\langle i, j \rangle\}$ 中某些值为 1 的比特变为 0, 然后根据映射关系计算得到待写入数据 B_i 值, 则可以直接写入 flash。或者简单地根据擦除后写入次数计数, 记录 Flash 块中某个范围的数据, 从上次擦除后直接写入的次数。如果写入次数小于 n 次, 则可以直接写入 flash。本方法支持在一次块擦除后, 至少可以直接写入 n 次。之后, 则可能需要再次擦除块。如果可以直接写入, 则转 107 进行写入操作。否则, 需要继续 105 处理。

[0032] 105. 在擦除块之前, 保存块中数据到缓冲区, 并与待写入数据合并。

[0033] 106. 执行块擦除操作。擦除完成后, 块中所有字节均恢复为 $8'b11111111$ 。

[0034] 107. 根据当前 $T\langle i, j \rangle$ 的值, 及待写入的 B_i 值, 通过映射规则确定新的 $T\langle i, j \rangle$ 值。要求满足 flash 编程的特性要求, 原来为 0 的比特不发生变化, 原来为 1 的比特, 可以有 1 位或多位变为 0。

[0035] 108. 完成的比特数据 B_i 的编程写入。

[0036] 在写 flash 数据时, 根据当前 $T\langle i, j \rangle$ 的值, 及 B_i 的值, 确定新的值。确定新的 $T\langle i, j \rangle$ 值时, 要求既满足确定的映射规则, 又满足 flash 编程的特性要求, 原来为 0 的比特不发生变化, 原来为 1 的比特, 根据映射关系, 可以有 1 位或多位变为 0。

[0037] 本发明中的映射关系, 指的是原始数据比特与在 Flash 中的数据 $\{T\langle i, j \rangle\}$ 的对应关系。原始数据比特取值范围为: $[0, 1]$, Flash 中的比特串取值范围为: $[0, 2^n - 1]$ 。映射关系用于确定比特串各比特数据经过运算后对应的原始数据比特值是 0 还是 1。映射关系可以有多种, 例如, 可以通过计算比特串中为所有位相加的和, 确定原始数据比特的值, 如数据 $8'b10101010$ 的所有位相加为 0, 对应原始数据比特为 0, 数据 $8'b10101000$ 的所有位相加为 1, 对应原始数据比特为 1; 或者是把比特串均分为两串 (或四串), 按位进行同或 / 异或运算, 运算结果为 $4'b0000$ 则认为对应的原始数据比特为 0, 否则为 1。

[0038] 下面举例说明一种比较简单的 B_i 与 $\{T\langle i, j \rangle\}$ 之间的映射关系。

[0039] 1) 在原始数据比特转换为 n 比特数据时, 选择 n 为偶数。在本例子中, 取 $n = 8$ 。一比特数据, 转换后, 占用 flash 中的一字节。即 B_i 转换后的数据为: $T\langle i, 7 \rangle, T\langle i, 6 \rangle \dots T\langle i, 1 \rangle, T\langle i, 0 \rangle$ 。

[0040] 2) 转换后的数据分为前半部分 $\{T\langle i, j \rangle\}_{j=7 \sim 4}$, 与后半部分 $\{T\langle i, j \rangle\}_{j=3 \sim 0}$ 。

[0041] 3) 在读 flash 数据时, 通过 $\{T\langle i, j \rangle\}_{j=7 \sim 4}$ 与 $\{T\langle i, j \rangle\}_{j=3 \sim 0}$ 进行异或运算, 得到 B_i 的值。如果每一位结果都是 0, 则对应的 B_i 值为 0, 否则对应的 B_i 值为 1, 即为 $4'b0000$, 则 $B_i = 1'b0$; 否则, $B_i = 1'b1$ 。

[0042] 当写 flash 数据时, 首先检查是否需要更新 flash 中的数据。读 flash 中的数据 $\{T\langle i, j \rangle\}_{j=7 \sim 0}$, 根据上述映射关系计算得到对应的老的 B_i 值。如果新的 B_i 值与老的 B_i 值相同, 则 flash 中此字节的数据不需要更新。否则, 继续进行处理。

[0043] 如果需要更新 flash 中的数据,则先检查是否需要执行擦除操作。如果可以找到 $\{T<i, j>\}_{j=7\sim 0}$ 中合适的值为 1 的比特,变为 0,再次映射后与新的 Bi 值相同,则无需进行擦除操作。例如 flash 中的数据为 11111111,按照上述映射关系,对应的老的 Bi 值为 0,我们只需改变 $\{T<i, j>\}_{j=7\sim 0}$ 中任意一位 1 为 0,例如把 $T<i, 0>$ 改为 0,则 flash 中的数据更新为 11111110,按照上述映射关系,新的 Bi 值为 1,则无需进行擦除操作实现了写入新的 Bi 值。或者把 $T<i, 6>$ 改为 0,则 flash 中的数据更新为 10111111,按照上述映射关系,新的 Bi 值也为 1,也可以实现无需进行擦除操作实现了写入新的 Bi 值。

[0044]

写入次数	Flash 中当前值	当前 Bi	新 Bi	Flash 中新值
1	1111-1111	0	1	1111-1110
2	1111-1110	1	0	1110-1110
3	1110-1110	0	1	1110-1100
4	1110-1100	1	0	1100-1100
5	1100-1100	0	1	1100-1000
6	1100-1000	1	0	1000-1000
7	1000-1000	0	1	1000-0000
8	1000-0000	1	0	0000-0000
9	0000-0000	0	不能直接写入,必须执行块擦除	

[0045] 表 1 原始数据与 flash 中数据的对应关系之一

[0046] 表 1 是按照比特 0、4、1、5、2、6、3、7 的顺序,查找值为 1 的比特,找到就将此比特变为 0。flash 中与一比特原始数据对应的字节数据,在多次编程操作及擦除操作时,其状态变化如图 2 所示。当然,也可以采用其它满足要求的规则,如 0、4、5、1、6、2、3、7 的顺序,查找值为 1 的比特,找到就将此比特变为 0,见表 2。

[0047]

写入次数	Flash 中当前值	当前 Bi	新 Bi	Flash 中新值
1	1111-1111	0	1	1111-1110
2	1111-1110	1	0	1110-1110
3	1110-1110	0	1	1100-1110
4	1100-1110	1	0	1100-1100
5	1100-1100	0	1	1000-1100
6	1000-1100	1	0	1000-1000
7	1000-1000	0	1	1000-0000
8	1000-0000	1	0	0000-0000
9	0000-0000	0	不能直接写入,必须执行块擦除	

[0048] 表 2 原始数据与 flash 中数据的对应关系之二

[0049] 其中,将 $\{T<i, j>\}_{j=r_0}$ 中哪些值为 1 的比特变为 0 的顺序并不重要,只要满足映射后与新的 B_i 值相同即可。Flash 不需要执行块擦除而直接写入的次数只与 n 有关,与顺序无关。综上所述,通过本发明的上述技术方案,既极大地提高了每次编程写入的速度,又减少了块擦除的次数,提高了 flash 的使用寿命。如果原始比特数据 B_i 映射为 flash 中的 8 比特数据,则在一次块擦除后,可以直接写入 8 次,中间不需要执行块擦除。直到 flash 中 8 比特数据变为 0,即 $8'b00000000$,才需要执行一次块擦除。如果原始比特数据 B_i 映射为 flash 中的 n 比特数据,则可以达到直接写入 n 次才执行一次块擦除, n 的上限可根据实际需要及芯片的实际情况选择。比特数 n 与 flash 寿命之间的关系(假设 flash 原有寿命为 10 万次),如表 3 所示。

[0050]

n 比特	1	2	4	8	16	32	64
可擦除次数:单位:10 万次	1	2	4	8	16	32	64

[0051] 表 3n 与 flash 寿命的对应关系

[0052] 当然,flash 中的没有数据时,可按照映射关系将待写入数据比特直接转换为 n 比特数据 $\{T<i, j>\}$ 写入 flash。

[0053] 本发明还可有其他多种实施例,在不背离本发明精神及其实质的情况下,熟悉本领域的技术人员当可根据本发明做出各种相应的改变和变形,这些相应的改变和变形都应属于本发明所附的权利要求的保护范围。

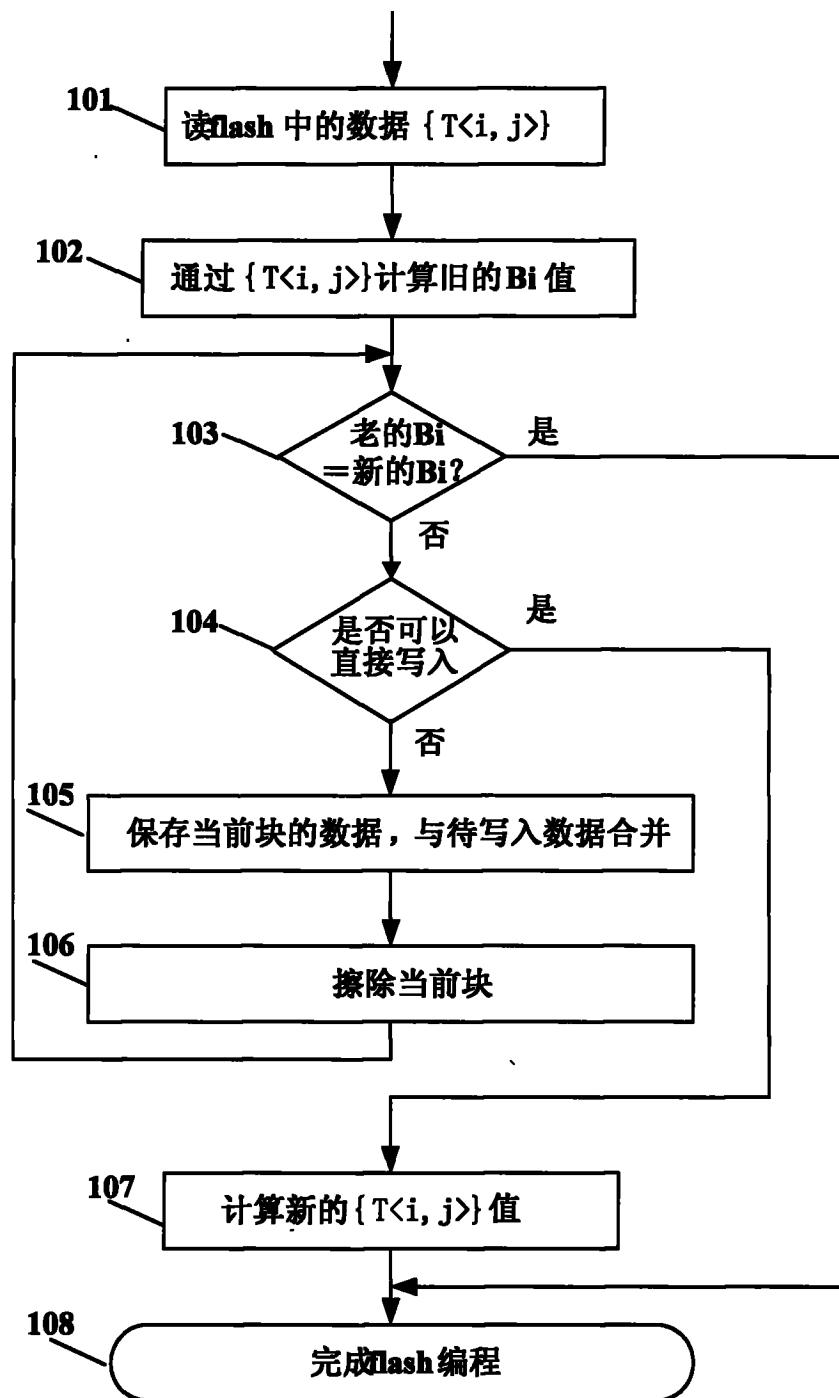


图 1

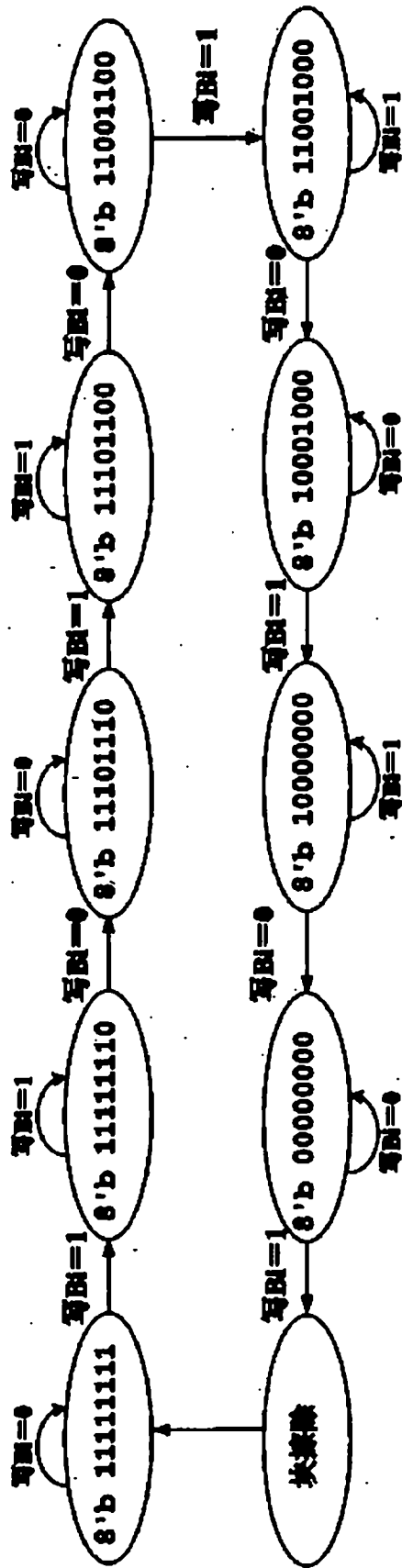


图 2