



(12)发明专利

(10)授权公告号 CN 104575617 B

(45)授权公告日 2017.06.27

(21)申请号 201510005036.3

(22)申请日 2015.01.06

(65)同一申请的已公布的文献号  
申请公布号 CN 104575617 A

(43)申请公布日 2015.04.29

(73)专利权人 北京大学  
地址 100871 北京市海淀区颐和园路5号

(72)发明人 孙广宇 张宪 张炜其

(74)专利代理机构 北京万象新悦知识产权代理  
事务所(普通合伙) 11360  
代理人 苏爱华

(51)Int.Cl.  
G11C 29/42(2006.01)

(56)对比文件

CN 101228583 A,2008.07.23,  
CN 101317159 A,2008.12.03,  
US 2012/0198314 A1,2012.08.02,  
W.S.Zhao.Magnetic Domain-Wall  
Racetrack Memory for high density and  
fast data Storage.《2012 IEEE 11th  
international Conference on Solid-State  
and Integrated Circuit Technology》.2012,  
第1-4页.

审查员 徐菲

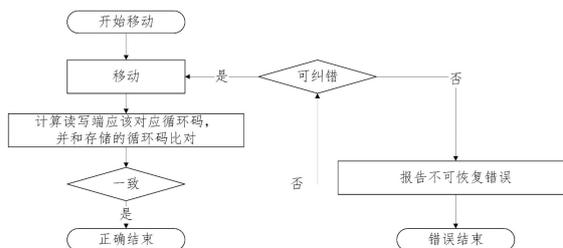
权利要求书2页 说明书6页 附图2页

(54)发明名称

赛道存储位置错误纠正码的编码方法和纠错方法

(57)摘要

本发明公布了一种赛道存储位置错误纠正码的编码方法和纠错方法,其中,纠正码的编码方法将赛道存储条(Racetrack memory stripe, RS)的一个或多个只读访问端相对RS的位置编码为一种循环码,并将其储存在RS中,使RS的位置与当前从RS中读取得到的部分循环码相互对应,通过该部分循环码得到位置码,作为赛道存储位置错误的纠正码。该循环码设为多个循环单元连续序列,循环单元的位数为一个或多个比特。纠错方法通过比较存储在寄存器中的访问端位置的位置码和存储在赛道存储条中的循环码解码出的位置码,来检测和纠正错格型位置错误,从而有效地解决错格型位置错误的问题。



1. 一种赛道存储位置错误纠正码的编码方法,所述方法将赛道存储条的一个或多个只读访问端相对于赛道存储条的位置编码为一种循环码,并将所述循环码储存在赛道存储条中,使赛道存储条的位置与当前读取得到的部分循环码相互对应,通过所述部分循环码得到位置码,作为赛道存储位置错误的纠正码;包括如下步骤:

S1. 设置一种循环码,所述循环码设为多个循环单元连续序列,所述循环单元的位数为一个或多个比特;

S2. 进行赛道存储位置错误纠正码的初始化,将所述循环码通过读写访问端和赛道存储条移动,正确写入并储存在赛道存储条的固定区域,所述循环码的每个比特分别储存在赛道存储条的一个数据域中;

S3. 赛道存储进行移动操作后,通过只读访问端读取当前存储在赛道存储条对应数据域中的部分循环码,通过计算得到位置码,作为赛道存储位置错误的纠正码。

2. 如权利要求1所述赛道存储位置错误纠正码的编码方法,其特征是,步骤S1所述循环单元的位数为 $2N$ 个比特,所述位数 $N$ 为自然数,表示纠错强度。

3. 如权利要求2所述赛道存储位置错误纠正码的编码方法,其特征是,所述循环单元包括 $N$ 个连续重复的0接 $N$ 个连续重复的1。

4. 如权利要求2所述赛道存储位置错误纠正码的编码方法,其特征是,所述循环码中连续的 $N$ 个比特表示一个位置码,对所述循环码进行编码需要 $N$ 个读写访问端和 $N$ 个只读访问端。

5. 如权利要求1所述赛道存储位置错误纠正码的编码方法,其特征是,步骤S2进行赛道存储位置错误纠正码的初始化,具体包括如下过程:

a) 从赛道存储条的最左的读写访问端 $rw1$ 依次写入所述循环码,每写入一个比特,赛道存储条往右移动一格,赛道存储条往右一次一格逐个循环;

b) 每当被写入的循环码经过 $rw1$ 右侧的每个读写访问端或只读访问端时,所有访问端逐个读出循环码并与所述写入的循环码进行比较检验;若比较结果一致则进入步骤c),如比较结果有不一致,则返回步骤a),重启初始化过程;

c) 当到达赛道存储条的最右的只读访问端后,所有访问端逐个读出循环码并与所述写入的循环码进行比较检验;若比较结果一致则进入步骤d),如比较结果有不一致,则返回步骤a),重启初始化过程;

d) 将赛道存储条进行一次一格的左移,在经过每个访问端时,所有访问端逐个读出循环码并与所述写入的循环码进行比较检验,若比较结果一致则进入步骤e),如比较结果有不一致,则返回步骤a),重启初始化过程;

e) 当到达赛道存储条的最左的只读访问端后,所有访问端逐个读出循环码并与所述写入的循环码进行比较检验;若比较结果一致则进入步骤f),如比较结果有不一致,则返回步骤a),重启初始化过程;

f) 将赛道存储条进行一次一格的右移,在经过每个访问端时,所有访问端逐个读出循环码并与所述写入的循环码进行比较检验,若比较结果一致则进入步骤g),如比较结果有不一致,则返回步骤a),重启初始化过程;

g) 到达赛道存储条的最右的只读访问端;

h) 重复步骤c)~g)多个轮数,使得写入循环码错误但是没有检测到的概率足够小,

即将所述循环码通过读写访问端和赛道存储条移动正确地写入并储存在赛道存储条的固定区域。

6. 如权利要求5所述赛道存储位置错误纠正码的编码方法,其特征是,步骤S3所述通过只读访问端读取循环码的一部分,具体是每一个只读访问端读取一个比特的循环码,得到一个比特的位置码;所有只读访问端读取得到的位置码为循环码的连续的一部分,通过所有只读访问端读取得到的位置码计算得到赛道存储条移动前后的位置,作为赛道存储位置错误的纠正码。

7. 一种使用权利要求1所述赛道存储位置错误纠正码的纠错方法,依次包括如下步骤:

1) 设定需要的检测和纠错强度;

2) 根据步骤1)所述检测和纠错强度,对所述赛道存储位置错误纠正码进行编码,通过赛道存储位置错误纠正码初始化,将所述循环码正确写入并储存在赛道存储条的固定区域;

3) 初始化读写头位置,将赛道存储条此时通过循环码标定的位置码计入访问端位置寄存器,使其对应的循环码和步骤2)存储在赛道存储条中的循环码相同;

4) 赛道存储进行正常的移动操作,存储在赛道存储条中的循环码跟随赛道存储条整体进行平移;

5) 移动操作结束后,得到两个码:

5.1) 通过访问端位置寄存器,根据正确移动的格数,计算得到若移动正确时应该得到的位置码;

5.2) 通过只读访问端读取得到当前赛道存储位置错误的纠正码;

6) 比对位置:将5.2)中当前赛道存储位置错误的纠正码与5.1)中的位置码进行比对,若两码相同,则得到移动操作正确的结果;若两码不同,若在可纠错范围则重复执行步骤4)~6);若不在纠错范围内,则报告不可恢复的错误,得到移动操作错误的结果。

## 赛道存储位置错误纠正码的编码方法和纠错方法

### 技术领域

[0001] 本发明涉及赛道存储,具体涉及一种赛道存储位置错误的纠正码的编码方法和纠错方法。

### 背景技术

[0002] 赛道存储(Racetrack Memory),又称为磁畴壁存储(Domain wall memory),是一种基于自旋存储技术的新型非易失存储。赛道存储由于其超高的存储密度和访问速度,目前正逐渐成为学界、业界的研究热点。赛道存储利用存储在条状磁材料上的大量磁畴来存储数据。然而为支持高密度的存储,赛道存储引入了新的操作:移动(shift)。一个RS(Racetrack memory Stripe,赛道存储条)上可以均匀的分布一些访问端,每个访问端负责访问一段域。为访问这一段域中的每个域,需要将域在RS上左右移动(只有域相对于RS的移动,访问端和RS物理上不运动),使得需要的域对应在访问端处。RS在上述保存数据的域两侧预留了一些不保存有效数据的冗余域,以确保含有有效数据的域在左右移动时不会超过RS的物理边界,以至于丢失。每一个RS有一个对应的访问端位置寄存器,用来保存访问端相对于RS的位置。赛道存储的结构特点决定了几乎每次数据访问都需要执行移动操作,因而这将带来对系统稳定性的额外影响。现有技术中,移动操作依靠磁畴壁的移动来完成:磁畴壁在自旋电流驱动下,在纳米线上移动并最终停止在锚定区域(pinning site)。然而计算和实验均表明,移动可能会发生错误:访问所需数据的磁畴壁停靠在错误的锚定区域,而不是需要的位置,使不应该被访问的磁畴对准访问端,从而使结果出现错误。这种错误表现为访问端位置寄存器标明的位置和实际RS移动的位置不相符。我们定义这种磁畴壁移动而产生的错位为错格型位置错误。目前,包括错误校验码(ECC)等在内的现有技术均不能有效检测和纠正这种错格型位置错误。

### 发明内容

[0003] 为了克服上述现有技术的不足,本发明提供一种赛道存储位置错误纠正码的编码方法和纠错方法,通过比较存储在寄存器中的访问端位置的位置码和存储在赛道存储条中的循环码解码出的位置码,来检测和纠正错格型位置错误,从而有效地解决错格型位置错误的问题。

[0004] 本发明中,缩略语和关键术语定义如下:

[0005] RM:Racetrack memory,赛道存储;

[0006] RS:Racetrack memory stripe,赛道存储条;

[0007] p-ECC:position error correction codes,位置错误纠正码;

[0008] Step:移动的格。

[0009] 本发明的原理是:为解决错格型位置错误,通过将访问端位置进行编码的位置码作为纠正码p-ECC,提供使用该纠正码p-ECC来对赛道存储位置错误进行纠错的位置纠错方法。p-ECC设计的基本思路是利用RS现有的结构,编码一种特殊的循环码并将其储存在RS

中。多个访问端对应循环码的一部分从而形成表示访问端的位置码，从而建立了访问端相对于RS的位置到位置码的对应。当RS整体移动时，存储在RS中的循环码也跟随RS整体地进行平移，这使得在移动后读出RS中存储的位置码相比初始时发生了变化。该位置纠错方法通过比对从RS读出的位置码和计算访问端位置得到的位置码，来判断移动是否正确。访问端位置的循环码的编码强度决定了该方法的检测和纠错强度。

[0010] 本发明提供的技术方案是：

[0011] 一种赛道存储位置错误纠正码的编码方法，将赛道存储条 (Racetrack memory stripe, RS) 的一个或多个只读访问端相对RS的位置编码为一种循环码，并将其储存在RS中，使RS的位置与当前从RS中读取得到的部分循环码相互对应，通过该部分循环码得到位置码，作为赛道存储位置错误的纠正码；包括如下步骤：

[0012] S1. 设置一种循环码，该循环码设为多个循环单元连续序列，所述循环单元的位数为

[0013] 一个或多个比特；循环单元的数量应足够多，以保证循环码能够覆盖检验范围内读取位

[0014] 置码的只读访问端可能停留的所有位置；

[0015] S2. 进行p-ECC初始化，将上述循环码通过读写访问端和RS移动，正确写入并储存在RS的固定区域，所述循环码的每个比特分别储存在RS的一个数据域中；

[0016] S3. RM进行移动操作后，通过只读访问端读取当前存储在RS对应数据域中的循环码，通过计算得到位置码，作为赛道存储位置错误的纠正码。

[0017] 赛道存储位置错误纠正码的编码方法中，循环码中连续的N个比特 (bit) 表示一个位置码，其位数N与纠错强度相关；对该循环码进行编码需要N个读写访问端和N个只读访问端。纠错强度最低为检测1格 (Step) 的位置错误SED (Single-step error detection)；增强纠错强度如检测2格的位置错误并且纠正1格的位置错误SECEDED (Single-step error correction, Double-step error detection) 到检测N+1格的位置错误并且纠正N格的位置错误 (N-step correction, N+1-step detection)，纠错强度逐渐增强。例如，检测1格的位置错误 (SED) 需要1个读写访问端，位置码为1bit；检测2格的位置错误 (SECEDED) 需要2个读写访问端，位置码为2bit。

[0018] 步骤S1中，一个循环单元共有2N个bit，包括N个连续重复的0接N个连续重复的1，N为自然数，纠错强度越高重复的数量N越大；其中，N为1表示纠错强度为检测1格 (Step) 的位置错误，N大于1表示纠错强度为检测N+1格的位置错误并且纠正N格的位置错误。

[0019] 上述赛道存储位置错误纠正码的编码方法中，步骤S2进行p-ECC初始化，将上述循环码通过读写访问端和RS移动，正确写入并储存在RS的固定区域，具体包括如下过程：

[0020] a) 从RS的最左的读写访问端rw1依次写入循环码，每写入1个比特RS往右移动1格，通过一次一格右移RS逐个循环；

[0021] b) 每当被写入的循环码经过rw1右侧其他的访问端 (包括读写访问端和只读访问端) 时，所有访问端逐个读出循环码并与所述写入的循环码进行比较检验，如果任何一次的比较发现有不一致，则返回步骤a)，重启初始化过程；

[0022] c) 当到达赛道存储条的最右的只读访问端后，所有访问端逐个读出循环码并与所述写入的循环码进行比较检验；若比较结果一致则进入步骤d)，如比较结果有不一致，则返

回步骤a), 重启初始化过程;

[0023] d) 将赛道存储条进行一次一格的左移, 在经过每个访问端时, 所有访问端逐个读出循环码并与所述写入的循环码进行比较检验, 若比较结果一致则进入步骤e), 如比较结果有不一致, 则返回步骤a), 重启初始化过程;

[0024] e) 当到达赛道存储条的最左的访问端后, 所有访问端逐个读出循环码并与所述写入的循环码进行比较检验; 若比较结果一致则进入步骤f), 如比较结果有不一致, 则返回步骤a), 重启初始化过程;

[0025] f) 将赛道存储条进行一次一格的右移, 在经过每个访问端时, 所有访问端逐个读出循环码并与所述写入的循环码进行比较检验, 若比较结果一致则进入步骤g), 如比较结果有不一致, 则返回步骤a), 重启初始化过程;

[0026] g) 到达赛道存储条的最右的只读访问端;

[0027] h) 重复步骤c) ~g) 多个轮数, 使得写入循环码若发生错误但没有检测到的概率足够小; 本发明实施例中, 此概率小于 $10E-30$ 即可, 即将所述循环码通过读写访问端和赛道存储条移动正确地写入并储存在赛道存储条的固定区域。

[0028] 步骤S3通过只读访问端读取循环码的一部分, 具体是每一个只读访问端读取一个比特的循环码, 得到一个比特的位置码; 所有只读访问端读取得到的位置码为循环码的连续的一部分, 通过所有只读访问端读取得到的位置码, 计算得到赛道存储条移动前后的位置, 作为赛道存储位置错误的纠正码。

[0029] 针对p-ECC中的循环码, 用于读取位置码的访问端为只读访问端, 只读访问端的个数为N; 每一个只读访问端读取1个bit的循环码, 得到1个bit的位置码。以强度最低的SED的编码为例: p-ECC只需一个访问端读取1bit的循环码得到1bit的位置码, 存储在RS中的编码被设计为“010101...”, 循环单元为“01”。读到0、1分别表示读写访问端在奇、偶位置处。以提供SECDED的保护强度为例, p-ECC需要两个读写访问端读取2bit的循环码, 存储在RS中的编码被设计为“00110011...”, 循环单元为“0011”。读到00、01、11、10则分别代表读写端在第 $4K+0$ 、 $4K+1$ 、 $4K+2$ 、 $4K+3$ 位, K为非零常数。更高的强度只需提供更多的读写访问端及更长的0和1重复。

[0030] 一种使用上述赛道存储位置错误的纠正码的纠错方法, 依次包括如下步骤:

[0031] 1) 设定需要的检测和纠错强度;

[0032] 2) 设置一种循环码, 将p-ECC进行初始化, 将循环码存储在RS的固定区域;

[0033] 3) 同时初始化读写头位置, 将RS此时通过循环码标定的位置码计入访问端位置寄存器, 使其对应的循环码和步骤2) 存储在RS中的循环码相同;

[0034] 4) RM进行正常的移动操作, 存储在RS中的循环码跟随RS整体进行平移;

[0035] 5) 移动操作结束后, 得到两个码:

[0036] 5.1) 通过访问端位置寄存器, 根据正确移动的格数, 计算得到若移动正确时应该得到的位置码;

[0037] 5.2) 通过只读访问端读取得到当前赛道存储位置错误的纠正码;

[0038] 6) 比对位置: 将5.2) 中的赛道存储位置错误的纠正码与5.1) 中的位置码进行比对, 若两码相同, 则得到移动操作正确的结果; 若两码不同, 若在可纠错范围则重复执行步骤4) ~6); 若不在纠错范围内, 则报告不可恢复的错误, 得到移动操作错误的结果。

[0039] 上述在可纠错范围内或不在纠错范围内的判断,可以根据以下方法进行:若发生的位置错误为能够确定错误情形的情况,为在纠错范围内的错误;若发生的位置错误不能确定错误情形,则不在纠错范围内,为不可恢复的错误。

[0040] 与现有技术相比,本发明的有益效果是:

[0041] 本发明提出了通过对访问端位置进行编码的位置码纠错机制,该机制将访问端相对RS的位置编码为一种循环码,并将其储存在各自RS中;多个访问端对应一个循环码,从而建立了位置到读写码的对应。赛道存储移动操作完成后,该机制通过比对从RS读出的位置码和计算访问端位置得到的位置码,判断赛道存储移动是否正确。编码访问端位置的循环码的强度决定了该机制的检测和纠错强度。本发明通过较小的空间和性能开销,有效地校验赛道存储位置错误,可有效减少赛道存储的移动错误引起的不可纠正错误。本发明包含多种变种,分别达到对N格位置错误的检测和纠正,从而解决错格型错误。

## 附图说明

[0042] 图1是本发明实施例中SECDED p-ECC的结构图,其中,(a)为初始化循环码的前三个比特时的状态,(b)为初始化完成后的状态。

[0043] 图2是用四格圆盘表示实施例中赛道存储移动时对应于读出的位置码的循环码。

[0044] 图3是本发明赛道存储位置错误纠正码编码的初始化方法的流程框图。

[0045] 图4是本发明赛道存储位置错误的纠错方法的流程框图。

[0046] 图5是本发明实施例中赛道存储位置错误的纠错方法的电路结构图。

## 具体实施方式

[0047] 下面结合附图,通过实施例进一步描述本发明,但不以任何方式限制本发明的范围。

[0048] 图1是本发明实施例中赛道存储位置错误的纠正码SECDED p-ECC的结构图,(a)为初始化循环码的前三个比特时的状态,(b)为初始化完成后的状态。整个长条代表1个RS,大箭头表示可读可写的读写访问端,小箭头表示读取位置码的只读访问端。RS条带包括数据域、冗余域和守卫域,图中白色的方块代表一个存储数据的数据域,其上的数字代表存入的数据;冗余域用来保存在正常移动时被移出的数据;RS条带两侧有守卫域,用来缓存由于可能的错误移动导致被额外移出的数据。访问端的位置被保存在位置寄存器中。正常工作时,每个访问端只访问自己服务的域,本例中第二个读写访问端服务的域为(b)中虚线框标明的连续的四个域。

[0049] 初始化后,读写访问端均停靠在其服务的四个域的最右端,例如第二个读写访问端停靠在虚线框标明的连续的四个域的最右端,如图1(b)所示,此时访问端相对于RS的位置被记为00,计入访问端位置寄存器。此时通过只读访问端读出的位置码为11。图2圆盘代表移动时读出的位置码,例如RS相对右移一格时,圆盘顺时针移动一格,读出的位置码为01。

[0050] 使用上述赛道存储位置错误的纠正码的位置纠错方法的流程如图3所示,赛道存储位置错误的纠正码p-ECC在初始化时将选定的循环码存入RS中,当移动结束后比对读出的位置码和通过访问端位置寄存器计算得到的位置码,来判断是否为一次正确的移动。

[0051] 本实施例中,赛道存储的存储单元从初始位置起,RS相对访问端向右移动3格,发生+1格错误,使用的纠错电路设计为:从访问端位置寄存器获得当前访问端的位置,并对编码强度所能得到的位置码个数取模,然后再对位置按照位置码的方式进行编码,得到移动正确时应该得到的位置码。然后比较p-ECC得到的位置码和正确的位置码,其结果送给控制逻辑模块进行分析,得出是否移动正确以及能否纠错的结论,如图5所示。图4为使用SECDED p-ECC纠错强度进行纠错的过程,具体包括如下步骤:

[0052] 1) SECDED p-ECC初始化:

[0053] 1.1) 将循环码“00110011...”存储在RS中对应的区域;

[0054] a) 从RS的最左访问端依次写入循环码(写入1个比特右移动1格,逐个循环);

[0055] b) 当被写入的循环码经过RS右侧的访问端时,每个访问端都逐个读出循环码并与正确的循环码进行比较,如果任何一次比较发现不一致,则重新开始步骤a),重启初始化过程;

[0056] c) 当最右访问端检验正确后,将RS进行1次1格的左移,并且在经过每个访问端时,逐个读出循环码并与正确的循环码进行比较,如果任何一次比较发现不一致,则重新开始步骤a),重启初始化过程;

[0057] d) 重复b和c共2轮(一共3个来回)后,即可保证循环码被正确的写入到RS中,此时,写入循环码若发生错误但没有检测到的概率小于 $10E^{-30}$ ;

[0058] 1.2) 初始化读写头位置,将RS此时通过循环码标定的位置码计入访问端位置寄存器;

[0059] 2) SECDED p-ECC进行校验纠错的过程如下:

[0060] 2.1) RM进行正常的移动操作,使得存储在RS中的位置码发生变化;此实施例中,RS相对访问端应该向右移动3格,实际却移动了4格;

[0061] 2.2) 比对位置:

[0062] e) 根据移动距离确定若移动正确时得到的位置码,比如若初始状态位置码为11,则RS相对读写头右移2格后应该得到的位置码为00;此实施例中,初始位置0对应的位置码为11,经过3格的移动,得到的位置码应该为10;

[0063] f) 从RS中读取当前存储在RS中的位置码并进行比对;此实施例中,因为RS左移了4格,所以读取当前存储在RS中的位置码得到的位置码为11;

[0064] 2.3) 若两码相同,则得到移动操作正确的结果;使用B、a中的例子若移动正确则得到的位置码应该为10;

[0065] 2.4) 若两码不同,若在可纠错范围则重复执行步骤2.1)和2.2);若不在纠错范围内,则报告不可恢复的错误,得到移动操作错误的结果。

[0066] 本实施例中,若读取当前存储在RS中的位置码是00说明发生了-1的位置错误,若为11则说明发生了+1格的位置错误,这两种错误为能够确定错误情形的情况,为在纠错范围内的错误;若看到的位置码是01则不能确定是发生了+2位置错误还是-2位置错误,不能确定错误情形,则不在纠错范围内,为不可恢复错误。此实施例中,比较发现RS移动发生+1位置错误,需要执行步骤2.1)和2.2)并向右移动1格;经过步骤2.1)的移动后,再次读取RS中存储的位置码,看到为10,与应该得到的位置码相同,故移动正确结束。

[0067] 需要注意的是,公布实施例的目的在于帮助进一步理解本发明,但是本领域的技

术人员可以理解：在不脱离本发明及所附权利要求的精神和范围内，各种替换和修改都是可能的。因此，本发明不应局限于实施例所公开的内容，本发明要求保护的范围以权利要求书界定的范围为准。

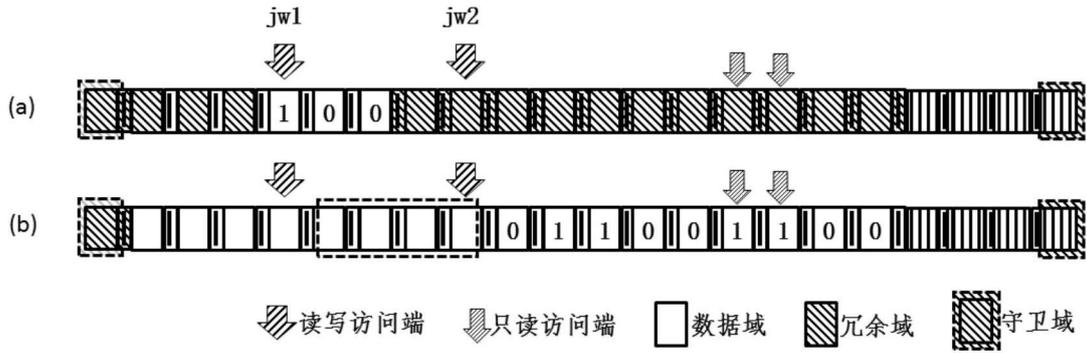


图1

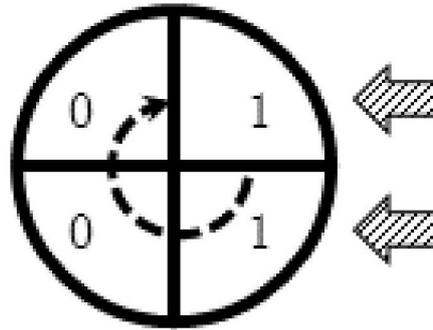


图2

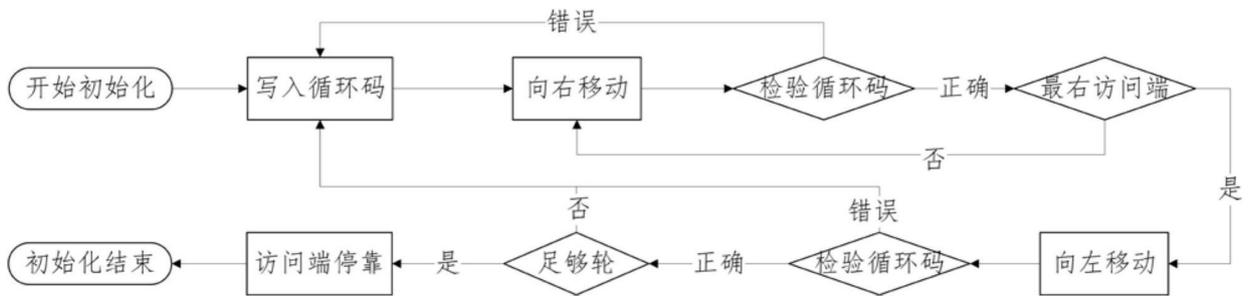


图3

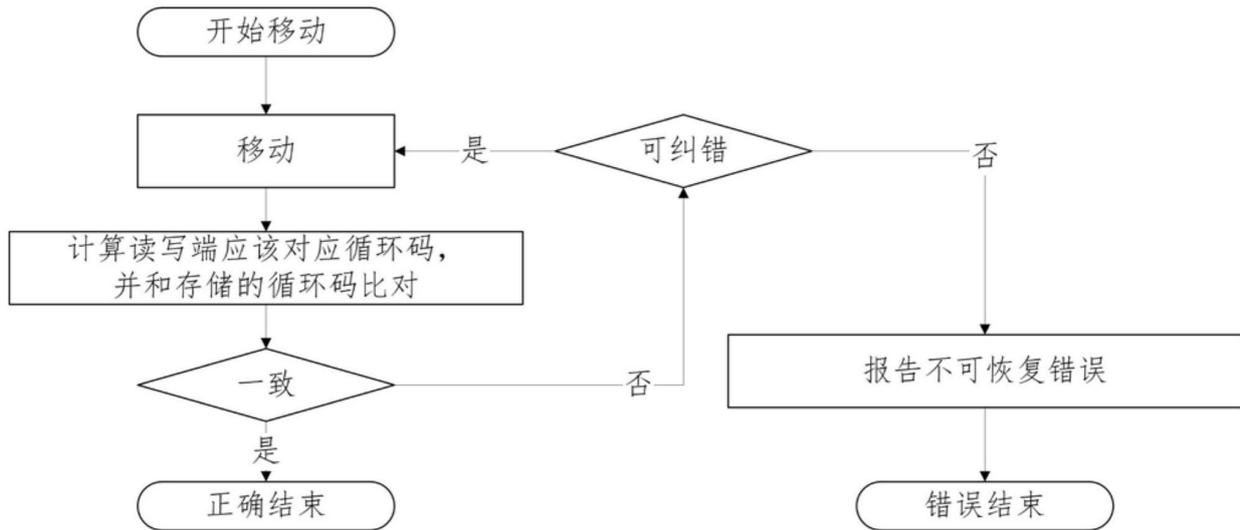


图4

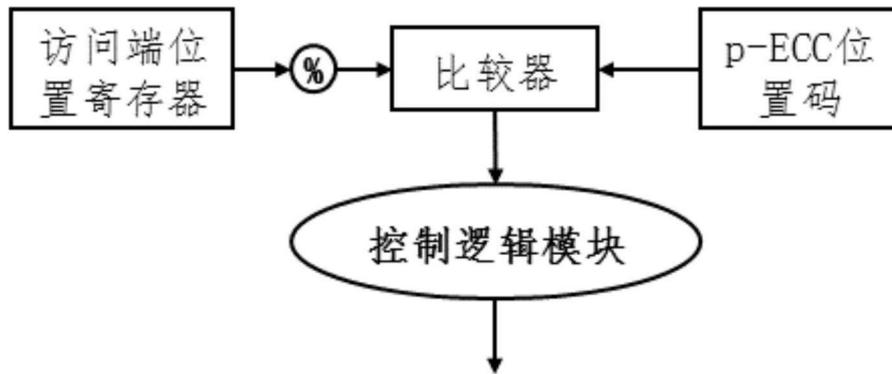


图5