基于置换码的 RFID 防冲突算法

吴胜成
(合肥工业大学 管理学院,安徽 合肥 230009)

摘 要:为了解决 RFID 系统中标签识别信息碰撞问题,文中基于现有的确定性二叉树防碰撞算法,通过引入置换码机制,提出了一种基于置换码的 RFID 防冲突算法。新算法中读卡器依据碰撞位个数的不同,发送不同的查询命令。若读卡器发现碰撞位个数仅为两位或者三位,则命令标签发送其碰撞位编码的置换码,从而快速识别这几个碰撞编码。仿真结果表明,新算法可以有效地减少标签识别过程中的查询次数和通信量,提高了标签识别效率和系统吞吐率。

关键词:射频识别;防碰撞;置换码;曼彻斯特编码
中图分类号:TP301.6 **文献标识码**:A **文章编号**:1673-629X(2015)02-0060-04
doi:10.3969/j.issn.1673-629X.2015.02.014

An Anti-collision Algorithm in RFID Based on Substitution Code

WU Sheng-cheng
(School of Management, Hefei University of Technology, Hefei 230009, China)

Abstract: In order to solve the problem of information collision during the process of tags identification, based on the existing binary search tree algorithm, by introducing substitution code unit mechanism, propose an anti-collision algorithm in RFID based on substitution code. In the new algorithm, the card reader sends different request command based on the number of the collision position. In order to recognize the tags rapidly, the card reader sends the new request command if the card reader finds the number of the collision position are 2 or 3. The simulation results show that the new algorithm can effectively reduce the times of tag identification and the information transferred in the process of query, improving the efficiency of tags identification and the system throughput.

Key words: radio frequency identification; anti-collision; substitution code; Manchester encoding

0 引 言

射频识别(Radio Frequency IDentification, RFID)技术是物联网的关键技术之一,它是一种利用无线电实现的非接触式自动识别技术[1]。RFID 已经被广泛的应用于物流管理、智能制造、智能交通等等多个领域[2-3]。

在 RFID 系统中,不可避免地会出现标签碰撞问题。RFID 系统中的标签多为无源标签,主要依靠时分多址[4](TDMA)方法解决标签碰撞问题。具体方法主要分为两大类:基于 ALOHA 的不确定算法[5-7]和基于二叉树的确定性算法[8-9]。

在基于 ALOHA 的不确定算法中,文献[10]提出了一种改进的动态帧时隙算法。该算法通过精确估计读写器附近的标签数目,并调整每帧中的时隙数,从而缩短标签识别时间。文献[11]提出了一种基于差值门限判决和时隙预测二进制选择机制的 FSA 算法。该算法能够解决识别标签过程中,初始阶段帧长和标签数目不匹配造成的时隙浪费问题,对提高系统吞吐率有一定意义。

在确定性防碰撞算法中,文献[12]提出了四叉树搜索方法。该方法扩大了搜索分支,在标签识别过程中可减少碰撞次数,但是随着搜索深度的加深,会产生大量的空闲时隙,从而降低了系统的吞吐率。文献[13]提出了一种增强型防碰撞算法(EAA)。该算法通过引入计数器和堆栈来减少碰撞发送的概率,当读写器发现碰撞位大于2以后,不再接收后续的数据,直接发送新的查询命令,开始下一轮的搜索,缩短了标签识别时间。

为了提高确定性防冲突算法的效率,文中通过引入置换码机制,提出了一种新的 RFID 防冲突算法—基于置换码的 RFID 防冲突算法(Binary Search on

收稿日期:2014-03-07 **修回日期**:2014-06-11 **网络出版时间**:2014-12-27
基金项目:国家自然科学重点基金(71231004)
作者简介:吴胜成(1989-),男,硕士研究生,研究方向为 RFID 防碰撞算法。
网络出版地址:http://www.cnki.net/kcms/detail/61.1450.TP.20141227.1343.027.html

Substitution Code algorithm,BSSC)。

1　算法改进思路

文中提出的 BSSC 算法以后退动态防冲突(Retreat Dynamic binary Search,RDS)算法[14]作为基础。在采用 RDS 算法识别标签的过程中,读卡器采取后退机制识别标签,虽然比传统 BS 算法减少了一半的搜索次数,提高了系统吞吐率,但是从二叉树搜索过程可知,RDS 算法的搜索次数由响应读卡器的标签个数决定,当标签数量很大时,仍然需要较多的搜索次数。

改进算法通过引入置换码机制,从减少搜索次数入手,对 RDS 做了适当改进。读卡器对标签编码进行解码时,若发现碰撞位个数为两位或者三位,则令其重新返回碰撞位编码的置换编码,从而使读卡器可直接识别这几个标签;反之,若碰撞位个数大于三位,则仍然使用 RDS 算法查询。

BSSC 算法在 RDS 算法的基础上做了如下改进:

当碰撞位为两位或者三位时,读卡器发送 Request(XX,a,b,c)命令,要求标签返回其碰撞编码的置换码,其中置换码编码方式见表 1 和表 2,表中原始编码是指从碰撞位置提取出的编码。标签收到 Request(XX,a,b,c)命令后,依据表 1 和表 2 的编码形式,发送碰撞位编码的置换码。

表 1　二个碰撞位时置换码编码方式

原始编码	置换码
00	0001
01	0010
10	0100
11	1000

表 2　三个碰撞位时置换码编码方式

原始编码	置换码
000	00000001
001	00000010
010	00000100
011	00001000
100	00010000
101	00100000
110	01000000
111	10000000

例如:当读卡器读取范围内有 5 个标签发生碰撞时,假设标签编码分别为:000001、000010、001011、001001、001010。若使用传统的 RDS 算法所需的搜索次数为 9 次。

若使用 BSSC 算法,读卡器识别冲突位 00X0XX 以后,发送 Request(00,3,1,0)命令。标签重新发送碰撞位置编码的置换码:00000010、00000100、10000000、00100000、01000000。读卡器对上述置换码解码后,可通过对置换码碰撞位的识别,一次性识别这 5 个标签。使用 BSSC 算法所需的搜索次数为 2 次。

2　算法指令

在标签识别过程中,新算法引入如下几条命令:

Request(111…111)命令。参数 111…111 中 1 的个数与标签编码长度相等。读卡器向在其读写范围内的标签广播此命令以后,所有标签编码值小于 111…111 的标签返回其编码。

Request(XXX)命令。前缀为 XXX 的标签,返回其余后编码。

Request(XX,a,b,c)命令。参数 XX 为查询前缀,参数 a,b,c 为编码碰撞发生的位置。前缀为 XX 的标签收到此命令后,发送其第 a 位、第 b 位、第 c 位的新编码给读卡器,新编码为依据表 1 或者表 2 所述的置换码而产生的。

Select(ID)参数 ID 为标签 EPC 编码,为读卡器读写标签信息做准备。

Read(ID)读卡器读取标签信息。

Unselect(ID)读卡器读取标签信息以后,发送 Unselect 命令。标签将不再响应读卡器,除非标签移除读卡器识别范围后再次进入读写范围。

3　算法步骤和算法流程图

步骤 1:初始化堆栈,使之为空。读卡器发送 Request(111…111)命令。

步骤 2:读卡器对收到的编码进行解码,识别非碰撞位编码信息和碰撞位位置信息。若无碰撞,则识别标签,并转入步骤 5;若发生碰撞,且碰撞位个数仅为 1 位,则直接识别这两个标签,并转入步骤 5;若碰撞位个数为 2 位或者 3 位,则转入步骤 3;若碰撞位数个数大于 3 位,则转入步骤 4。

步骤 3:读卡器发送 Request(XX,a,b,c)命令。参数 XX 为查询前缀,参数 a,b,c 为编码碰撞发生的位置,若仅有两个碰撞位,则参数变为 a,b,null。前缀为 XX 的标签收到此命令后,标签发送其第 a 位、第 b 位、第 c 位的置换码给读卡器。若碰撞发送在标签编码最高位,则前缀用 NULL 表示,转入步骤 5。

步骤 4:读卡器对收到的编码信息进行解码,将产生的两个新的查询前缀依据由大到小的次序先后存入堆栈。

步骤 5:判断堆栈是否为空,若非空,则读卡器读取并发送栈首信息给标签,返回步骤 2;否则,识别过程结束。

该算法中,读卡器的工作流程图如图 1 所示。

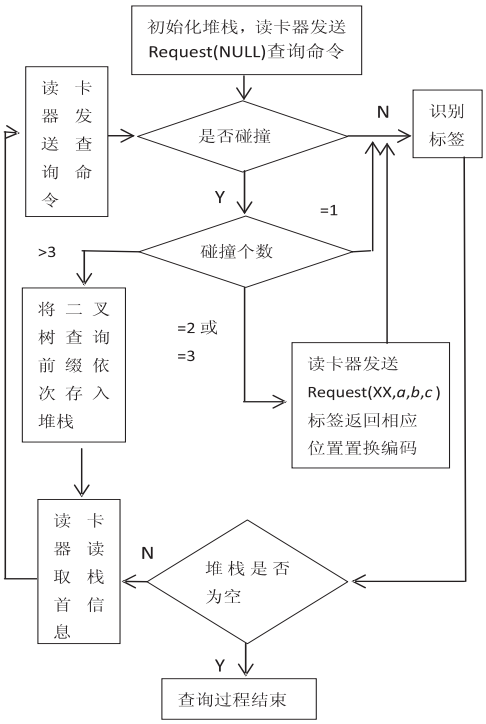

图 1　BSSC 算法工作流程图

4　算法识别实例分析

假设读卡器范围内存在 7 个位长为 8 的待识别标签，各个标签编码分别为：00100000、00000000、10100000、10000000、10001100、10001000、11111111。标签编码的识别过程如下：

步骤 1：读卡器广播 Request(11111111)命令。所有标签返回其编码。读卡器识别到的碰撞位个数大于 3，将查询前缀(1)、(0)依次存入堆栈。

步骤 2：读卡器从栈顶读取信息(0)，发送 Request(0)命令。

步骤 3：前缀为 0 的标签 A、B 响应读卡器。读卡器识别编码 0X00000，碰撞位只有一位，直接识别标签 A、B。

步骤 4：读卡器从栈顶读取信息(1)，发送 Request(1)命令。前缀为 1 的标签 C、D、E、F、G 返回其余后编码，读卡器识别编码 XXXXXXX。

步骤 5：读卡器识别到的碰撞位个数大于 3，将查询前缀(11)、(10)依次存入堆栈。

步骤 6：读卡器从栈顶读取信息(10)，发送 Request(10)命令。

步骤 7：前缀为 10 的标签 C、D、E、F 返回其余后编码，读卡器识别编码 X0XX00。

步骤 8：读卡器识别到的碰撞位个数为 3，发送 Request(10,5,3,2)命令。标签 C、D、E 和 F 分别返回置换码：00010000、00000001、00001000 和 00000010。识别标签 C、D、E 和 F。

读卡器读取栈顶碰撞信息(11)，读卡器发送 Request(11)命令。前缀为 11 的标签 G 返回其余后编码，读卡器识别编码 111111。识别标签 G。

步骤 9：读卡器从栈顶读取信息(11)，发送 Request(11)命令。

步骤 10：前缀为 11 的标签 G 返回其余后编码，读卡器识别编码 111111。识别标签 G。

步骤 11：读卡器读取栈顶信息，栈空。所有标签识别完成。

识别过程如图 2 所示。

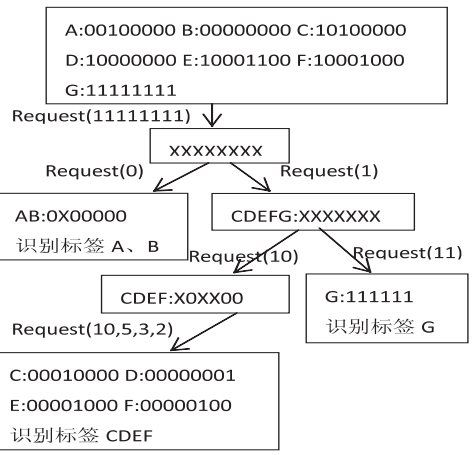

图 2　BSSC 算法标签识别过程

5　仿真实验与结果分析

下面将 BSSC 算法和 RDS 算法相比较。使用 Eclipse 软件仿真读卡器和标签之间的信息交互。在理想信道条件下进行仿真，定义标签编码长度为 8 Bits，每组数据仿真 30 次，取实验结果数据的平均值。

首先，对 BSSC 算法和 RDS 算法的搜索次数进行了比较。图 3 表示的是两种算法在搜索次数方面的区别。当标签数量较少时，BSSC 算法和 RDS 算法在搜索次数方面基本持平；但是，当标签数量增加时，BSSC 算法在搜索次数方面比 RDS 算法有很大缩减。当读卡器工作范围内标签数量达到 120 个时，BSSC 算法搜索次数仅为 90.19 次，RDS 算法为 165.60 次。BSSC 算法搜索次数比 RDS 算法缩减了 45.5%。

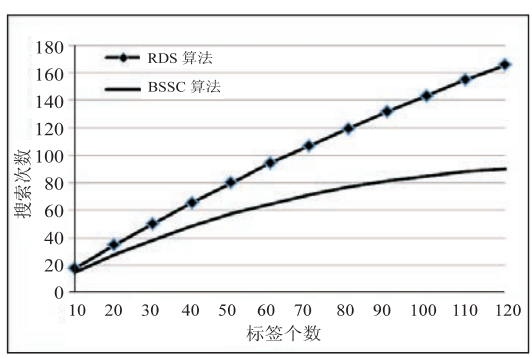

图 3　两种算法搜索次数比较图

图 4 表示的是两种算法在通信量方面的区别。随着标签数目的增长,RBS 算法的通信量较 BSSC 算法增幅越趋明显。当读卡器工作范围内标签数量达到 120 个时,BSSC 算法识别所有标签传输的通信量为 916 Bits。而 RDS 算法传输的通信量分别为 1 321 Bits。BSSC 算法通信量比 RDS 算法缩减了 30.65%。

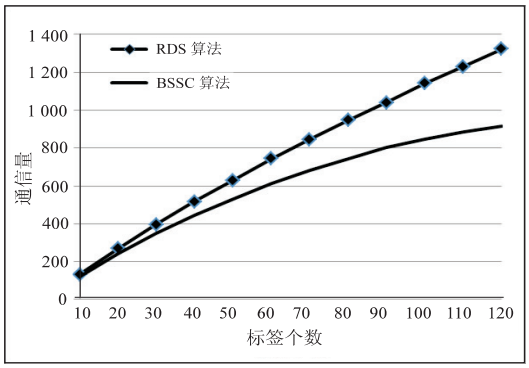

图 4　两种算法通信量比较图

图 5 表示的是两种算法在吞吐率方面的区别。由图可知,两种算法的吞吐率均维持在 50% 以上,随着标签数量的增加,BSSC 算法的吞吐率有较快上升。当读卡器工作范围内标签数量达到 120 个时,BSSC 算法吞吐率高达 133%,比 RDS 算法提高了 60.59%,吞吐率增长明显。

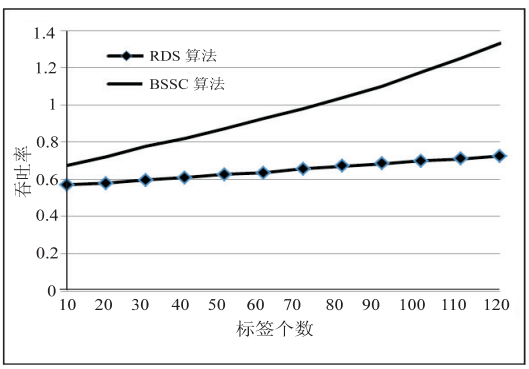

图 5　两种算法吞吐率比较图

通过对上述仿真结果的分析发现,BSSC 算法的搜索次数和吞吐率均明显优于 RDS 算法。同时,虽然 BSSC 算法在发送额外的置换码时将产生大量信息交互,但是在识别所有标签的整个过程中,BSSC 算法的总体通信量仍小于 RDS 算法。

当标签数目增多时,读卡器发送查询前缀以后,标签返回的编码出现两位或三位碰撞位的概率将大幅增加。在使用 BSSC 算法时,仅需额外的一次查询,即可识别这些标签,从而增加了 BSSC 算法和 RDS 算法在搜索次数方面的区别。

6　结束语

文中提出的 BSSC 算法在处理 RFID 读卡器内标签碰撞问题时,读卡器依据碰撞位个数的不同,发送不同的查询命令。读卡器对标签返回的编码进行解码,若发现碰撞位个数为两位或者三位,则读卡器发送新的查询命令,让标签发送其碰撞位编码的置换码,从而快速识别碰撞编码。仿真结果表明,BSSC 算法的搜索次数、通信量和吞吐率明显低于 RDS 算法。这对提高 RFID 系统的效率,具有一定的现实意义。

参考文献:

[1] Petkovic M, Jon-ker W. Lang enrich M RFID and privacy [C]//Proc of security, privacy, and trust in modern data management. [s. l.]:[s. n.],2007:433-450.

[2] 张捍东,朱　林. 物联网中的 RFID 技术及物联网的构建[J]. 计算机技术与发展,2011,21(5):56-59.

[3] 王铖岑. RFID 系统防碰撞算法[J]. 计算机技术与发展,2010,20(1):29-32.

[4] 张文解. RFID 系统防碰撞算法研究[D]. 长春:吉林大学,2012.

[5] Abrmson N. The ALOHA system-another alternative for computer communications[C]//Proc of fall joint computer conference. [s. l.]:[s. n.],1970:281-285.

[6] Namboodiri V, Gao Lixin. Energy-aware tag anti-collision protocols for RFID systems[C]//Proceedings of fifth annual IEEE international conference on pervasive computing and communications. White Plains:IEEE,2007:23-26.

[7] Ali K, Hassanein H, Taha A E M. RFID anti-collision protocol for dense passive tag environments[C]//Proc of 32nd IEEE conference on local computer networks. Dublin:IEEE,2007:819-824.

[8] Zhu L, Yum T S P. A critical survey and analysis of RFID anti-collision mechanisms[J]. IEEE Communications Magazine,2011,49(5):214-221.

[9] 张文欣,昂志敏,尹夕振. 一种改进的后退式二进制搜索 RFID 多标签防碰撞算法[J]. 合肥工业大学学报:自然科学版,2012,35(7):919-922.

[10] 程文青,赵梦欣,徐　晶. 改进的 RFID 动态帧时隙 ALOHA 算法[J]. 华中科技大学学报:自然科学版,2007,35(6):14-16.

[11] 李　萌,钱志鸿,张　旭,等. 基于时隙预测的 RFID 防碰撞 ALOHA 算法[J]. 通信学报,2011,32(12):43-50.

[12] Ryu J, Lee H, Seok Y, et al. A hybrid query tree protocol for tag collision arbitration in RFID systems[C]//Proceedings of IEEE international conference on communications. Glasgow:IEEE,2007:5981-5986.

[13] He Mingxing, Horng S J, Fan Pingzhi, et al. A fast RFID tag identification algorithm based on counter and stack[J]. Expert System with Applications,2011,38:6829-6838.

[14] 徐东升. RFID 系统防冲突算法的研究与实现[D]. 哈尔滨:哈尔滨工程大学,2009.

基于置换码的RFID防冲突算法

万方数据 WANFANG DATA 文献链接

作者： 吴胜成， WU Sheng-cheng
作者单位： 合肥工业大学 管理学院,安徽 合肥,230009
刊名： 计算机技术与发展 ISTIC
英文刊名： Computer Technology and Development
年，卷(期)： 2015(2)

引用本文格式：吴胜成.WU Sheng-cheng 基于置换码的RFID防冲突算法[期刊论文]-计算机技术与发展 2015(2)