

基于 EPC Class-1 Gen-2 标准的防冲突算法与改进

张瑞子^{1,2}, 南 琳¹, 胡琨元¹, 田景贺^{1,2}

(1. 中科院沈阳自动化研究所, 沈阳 110016; 2. 中国科学院研究生院, 北京 100039)

摘 要: 针对 RFID 读写器识别多标签过程中出现的冲突问题, 研究并实现了 EPC Class-1 Gen-2 标准中的防冲突算法, 即时隙随机算法(SR 算法), 同时针对 SR 算法的不足提出改进算法。改进算法采用不避让冲突时隙的处理方式, 降低了由时隙的随机选取所导致的标签间冲突的概率。实验结果证明, 改进后的算法在通信次数和吞吐率方面均优于原算法, 有效提高标签识别效率。

关键词: 防冲突; EPC Class-1 Gen-2 标准; ALOHA 算法; 标签识别

Anti-collision Algorithm Based on EPC Class-1 Gen-2 Standard and Its Improvement

ZHANG Rui-zi^{1,2}, NAN Lin¹, HU Kun-yuan¹, TIAN Jing-he^{1,2}

(1. Shenyang Institute of Automation, Chinese Academy of Sciences, Shenyang 110016;

2. Graduate University of Chinese Academy of Sciences, Beijing 100039)

【Abstract】 This paper analyzes and implements the anti-collision algorithm of EPC Class-1 Gen-2 protocol for the problem of signal collision. It presents an improved algorithm of it. The improved algorithm does not parry the collided slots, which can reduce the probability of collision caused by random slot choice. The results of the experiment show that the improved algorithm, compared with the original algorithm, has less communications and better throughput. So the improved algorithm is a feasible and efficient solution to identify EPC Class-1 Gen-2 tags.

【Key words】 anti-collision; EPC Class-1 Gen-2 protocol; ALOHA algorithm; tag identification

1 概述

RFID读写器读取大量标签的过程中不可避免地会存在标签间的冲突问题, 造成了数据读取的不可靠与不正确, 因此制定合理有效的防冲突机制已成为RFID读写器设计中的关键问题。目前国际上流行的ISO/IEC 18000-6 TypeA标准和EPC Class-1 Gen-2 标准在解决标签间的冲突问题时均利用了ALOHA算法。ALOHA算法可分为时隙ALOHA算法、帧时隙ALOHA算法和动态帧时隙ALOHA算法^[1-2]。时隙ALOHA算法将时间域划分成几个时隙, 标签可随机选择一个时隙作为其应答时隙, 若有多个标签选择了同一时隙, 便产生了标签间的冲突。帧时隙ALOHA算法将时间域划分成若干个由时隙组成的帧, 且每个帧中所包含的时隙数量均固定不变。读写器可通过多次帧循环操作识别出所有的标签。这种算法虽然能够有效地减小冲突发生的概率, 但由于帧时隙数量固定不变, 因此影响了识别效率。针对这一问题, 又提出了动态帧时隙ALOHA算法, 使得读写器可以动态调整每个帧中的时隙数量。由于调整帧时隙数量的方法不同, 动态帧时隙ALOHA算法也各有差异。

EPC Class-1 Gen-2 标准所采用的是时隙随机防冲突算法(SR算法), 该算法也是一种动态帧时隙ALOHA算法。SR算法与其他动态帧时隙ALOHA算法一样, 读写器可以动态调整每一帧中的时隙数量, 但不同的是, SR算法可以根据标签时隙的分布情况随时令标签进入到下一轮的帧循环操作中, 而无需将这一帧中的时隙全部处理完毕。SR算法能够更加灵活地调整帧的大小, 在通信次数与吞吐率方面均优于其他ALOHA算法^[3]。本文研究并实现了EPC Class-1 Gen-2 标准中的防冲

突算法(SR算法), 并对该算法进行了改进。

2 SR 算法

在 SR 算法中, 每个标签都有一个随机数产生器和一个 15 bit 的时隙计数器。当接收到读写器发来的时隙信息时, 标签便在读写器所规定的时隙范围内随机选择一个时隙载入到时隙计数器中。若发生标签间的冲突或在当前时隙内无标签应答时, 读写器便会对标签选择的时隙进行调整。所涉及的命令有: Query, QueryAdjust, QueryRep。

Query 命令: 为标签提供初始的 Q 值, 标签可在 $[0, 2^Q - 1]$ 范围内随机选择一个时隙作为应答时隙。

QueryAdjust 命令: 使参数 Q 加 1、减 1 或保持不变, 同时使所有未被识别的标签进入到下一轮帧循环操作中, 并重新选择应答时隙。

QueryRep 命令: 使标签时隙计数器中的时隙减 1。

当读写器要对标签进行读写操作时, 读写器首先向所有的标签发送 Query 命令, 在此命令中包含一个参数 Q (Q 为 0~15 之间的整数), 处于读写器射频场中的标签收到此命令后会产生 0~ $2^Q - 1$ 之间的随机数作为其应答时隙, 并将此随机数载入时隙计数器中。只有时隙计数器中的时隙为 0 的标签才会向读写器发送 RN16(16 位的随机数)作为应答信息。

基金项目: RFID 系统集成应用关键技术研究基金资助项目(06F2060201); 智能 RFID 中间件平台开发与关键技术攻关基金资助项目(07L2020201)

作者简介: 张瑞子(1979 -), 女, 硕士, 主研方向: UHF RFID 读写器; 南 琳, 研究员; 胡琨元, 副研究员; 田景贺, 博士

收稿日期: 2008-08-14 **E-mail:** zhangruizi@sia.cn

因此,读写器发送 Query 命令后会出现 3 个结果:(1)有且只有一个标签向读写器发送了 RN16,此时读写器成功地识别出了该标签;(2)所有标签均没有向读写器发送 RN16,此时读写器既可以使所有标签的时隙减 1(发送 QueryRep 命令),也可以通过减小 Q 值(发送 QueryAdjust 命令)的方法令所有标签重新选择应答时隙,这样便可使标签的时隙为 0 的概率增大;(3)有多于一个的标签向读写器发送了 RN16,这时便产生了标签间的冲突,读写器既可以使所有标签的时隙减 1(发送 QueryRep 命令),也可以增大 Q 值(发送 QueryAdjust 命令)使标签在更大的范围内随机选择时隙,从而减小发生冲突的概率。读写器通过这种机制不断地调整各个标签的时隙,使得标签分别占用不同的时隙与读写器进行数据传输。

从 SR 算法中可以看出, Q 值越大,随机选择时隙的范围也就越大,发生冲突的可能性就会越小,但将导致空闲时隙增加;反之, Q 值越小,射频场中的标签发生冲突的可能性就会越大,导致冲突时隙增加。过多的空闲时隙和冲突时隙都会降低识别效率,因此读写器需要不断地调整 Q 值,使时隙数量接近于标签的数量,从而使得标签在最为合理的取值范围内随机选择时隙。

在 EPC Class-1 Gen-2 标准中提出了一个选取 Q 值的方法(如图 1 所示),其中 Q_{fp} 为 Q 的浮点表示,读写器将 Q_{fp} 四舍五入为一个整数值,并用这个整数值代替 Query 命令中的 Q 值。 C 的典型值为 $0.1 < C < 0.5$ 。当 Q 值较大时,读写器一般用 C 的较小值,如果 Q 值较小,则用 C 的较大值^[4]。

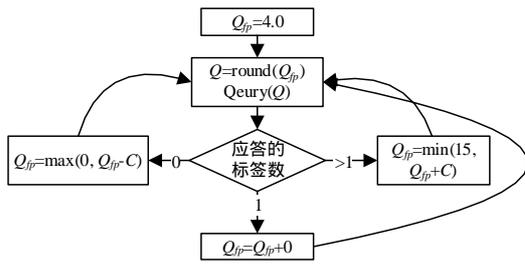


图 1 Q 值的选取算法

综上所述, SR 算法的实现流程如图 2 所示。

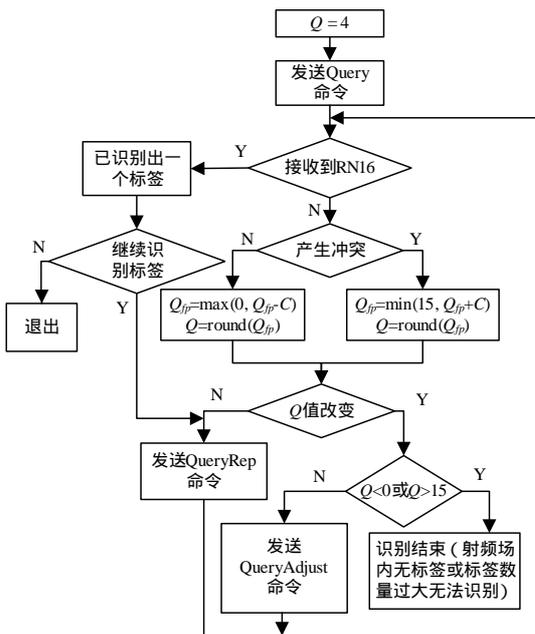


图 2 SR 算法的实现流程

由于标签的时隙是随机选择的,因此 SR 算法很容易出现多个标签同时占用同一个时隙及空闲时隙未被占用的现象。这样在识别标签的过程中,读写器会不断地发送命令来调整所有标签随机选择的时隙,使得标签能够占用不同的时隙与读写器进行数据传输。

针对这一问题,本文提出了一种改进算法,使标签能够更为均匀地分布在各个时隙中,以此来提高识别效率。

3 改进算法

在 SR 算法的基础上,本文提出了一种改进算法(如图 3 所示)。

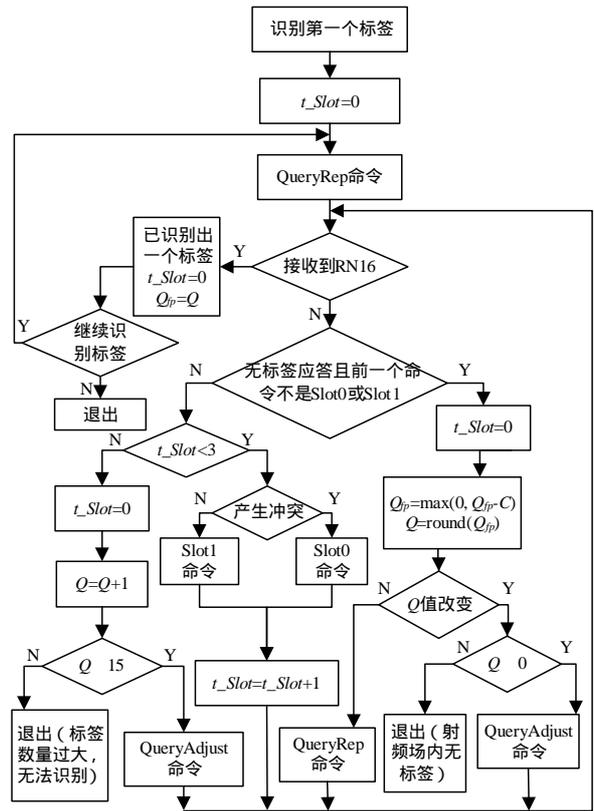


图 3 改进算法

该改进算法在 SR 算法的基础上增加了 2 条命令: Slot0 命令使时隙为 0 的标签在 $[0, 1]$ 范围内随机选择一个新的时隙并载入时隙计数器中,时隙不为 0 的标签将其时隙加 1。Slot1 命令使时隙为 1 的标签在 $[0, 1]$ 范围内随机选择一个新的时隙并载入时隙计数器中,时隙不为 1 的标签将其时隙加 1。

改进后的算法的实现过程如下所述:

步骤 1 采用 SR 算法识别出第一个标签,以获得合理的 Q 值。

当读写器与标签进行数据传输时需要获得与标签数量相匹配的时隙数量。若标签数量远小于帧时隙数量则会造成空闲时隙的浪费,若标签数量远高于帧时隙数量则会产生过多的冲突时隙,这 2 种情况都会降低识别效率。时隙数量与标签数量一致为最佳选择,此时系统吞吐率最大^[3,5],因此读写器需要在识别标签的过程中估算标签的数量。SR 算法规定了时隙数量只能为 2^Q ,标签可在 $[0, 2^Q - 1]$ 范围内随机选择一个时隙, Q 的初始值为 4。若经常发生标签间的冲突,读写器会增大 Q 值,以此来增加时隙数量;若经常出现无标签应答的情况,读写器则会减小 Q 值,时隙数量也就会随之减小。读写器识别第一个标签时,可以利用时隙数量的这种调整过程

获得相对于初始值更为合理的 Q 值,使 2^Q 接近标签总数。

步骤 2 在识别出第一个标签的基础上识别标签群中的其他标签,其具体步骤如下所述:

(1)读写器向标签发送 QueryRep 命令。

(2)读写器接收标签的应答,并对应答结果作出判断,会出现(3)、(4)、(5) 3种可能的情况。

(3)若有且仅有一个标签应答读写器发送的命令,则读写器已识别出了该应答标签。执行步骤(8)。

(4)若没有标签应答读写器发来的命令,其处理方法与 SR 算法相同,即通过参数 C 来调整 Q 值的大小,使 Q 值在参数 C 的控制下减小。读写器根据 Q 值的改变情况来判断下一步将发送的是 QueryAdjust 命令还是 QueryRep 命令。若 $Q = 0$,则返回到步骤(2);否则,说明射频场内无标签,退出。

(5)若产生标签间的冲突时,读写器便向标签发送 Slot0 命令,使得时隙为 0 的标签在[0, 1]之间重新随机选择一个新的时隙,而其他标签则将其时隙加 1,这样可以降低多个标签的时隙同时为 1 的概率,进而也就减小了下一时隙发生冲突的可能性。执行步骤(6)。

(6)读写器发送 Slot0 命令或 Slot1 命令后接收标签的应答,并对应答结果作出判断,会出现 3 种可能的情况:1)时隙为 0 的标签重新选择新时隙后均没有向读写器发送 RN16,也就是说所有标签的时隙均不为 0。此时,若已达到读写器预先设定的循环次数(本文的试验中将其设置为 3),则执行步骤(7);否则读写器将发送 Slot1 命令,重新执行步骤(6);2)时隙为 0 的标签重新选择新的时隙后仍然出现标签间的冲突,也就是说仍然有多于一个的标签选择了时隙 0。此时,若达到读写器预先设定的循环次数,则执行步骤(7);否则读写器将发送 Slot0 命令,重新执行步骤(6);3)读写器接收到标签发送的 RN16,即已识别出一个标签,执行步骤(8)。

(7)达到预先设定的循环次数后仍然没有标签被识别出来,说明选择的 Q 值过小,使得有过多的标签选择了同一时隙,导致了只依靠冲突标签在时隙 0 和时隙 1 之间的重新分配已经无法解决冲突问题。此时,读写器便会发出 QueryAdjust 命令使 Q 值加 1 以增加时隙数量。标签收到 QueryAdjust 命令后会重新随机选择时隙并将新的时隙载入时隙计数器中。返回到步骤(2)。

(8)若需要继续识别下一个标签,则返回步骤(1),否则退出。

4 仿真与分析

读写器与标签的通信次数 t 随标签数量 n 的变化曲线如图 4 所示。

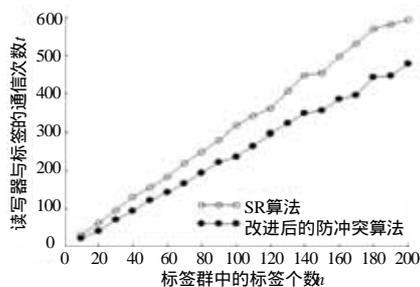


图 4 通信次数随标签数量的变化曲线

图 4 中 t 为标签群中所有标签均被识别完毕时的通信次数。由于标签的时隙是随机选取的,因此在仿真实验中将标签群重复操作 10 次,将每次得出的 t 累加后再取平均值。从

图中可以看出,改进后的算法可以用更少的通信次数识别出所有的标签。图 5 为系统的吞吐率随标签数量的变化曲线,这 2 种算法的吞吐率均比较稳定,但改进后的算法的吞吐率要高于 SR 算法。当标签数量小于 30 时,这一优势更加明显。这是因为当标签数量较少时,用 2^Q 来估算的标签数量会更接近于实际的标签数量。

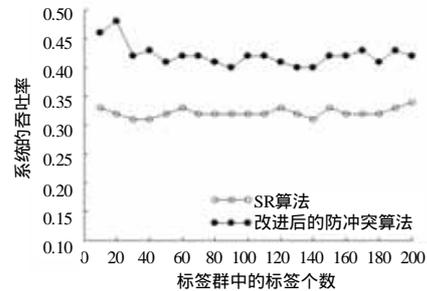


图 5 吞吐率随标签数量的变化曲线

从仿真结果中可以看出,改进后的防冲突算法无论在通信次数还是在吞吐率方面均优于 SR 算法,其原因就在于这 2 种算法对冲突标签的处理不同。当发生标签间的冲突时,SR 算法采用完全避让的原则,通过发送 QueryRep 命令使所有标签的时隙减 1,这样读写器就放弃了对冲突标签的处理而去处理下一时隙中的标签。冲突标签只能在以后的帧循环操作中重新随机选择时隙,而重新选择后仍会存在冲突时隙与空闲时隙的数量无法调整的现象。改进后的算法将发生冲突的标签在其临近时隙内重新分布,并且增大其他标签的时隙,这样便可将聚集在同一时隙中的标签分散开,同时减小空闲时隙和冲突时隙的数量,使得标签能够更为均匀地分布在各个时隙中。

5 结束语

本文针对 RFID 读写器识别多标签过程中出现的冲突问题,结合 EPC Class-1 Gen-2 标准中的读写器与标签的通信指令,研究并实现了防冲突算法,同时针对其不足提出了改进算法。从仿真结果可以看出,改进后的算法在性能上要优于原算法。

参考文献

- [1] Tao Cheng, Li Jin. Analysis and Simulation of RFID Anti-collision Algorithms[C]//Proc. of the 9th International Conference on Advanced Communication Technology. Seoul, Korea: IEEE Press, 2007: 697-701.
- [2] Peng Qingsong, Zhang Ming, Wu Weimin. Variant Enhanced Dynamic Frame Slotted ALOHA Algorithm for Fast Object Identification in RFID System[C]//Proc. of 2007 IEEE International Workshop on Anti-counterfeiting, Security, Identification. Xiamen, China: IEEE Press, 2007: 88-91.
- [3] 刘 佳, 张有光. 基于时隙的 RFID 防碰撞算法分析[J]. 电子技术应用, 2007, 33(5): 94-96.
- [4] EPCglobal Inc.. EPCTM Radio-frequency Identity Protocols Class-1 Generation-2 UHF RFID Protocol for Communications at 860 MHz-960 MHz[EB/OL]. (2005-01-02). http://www.epcglobalinc.org/standards/uhfclg2/uhfclg2_1_0_9-standard-20050126.pdf.
- [5] Wang Jianwei, Wang Dong, Zhao Yuping. A Novel Anti-collision Algorithm with Dynamic Tag Number Estimation for RFID Systems[C]//Proc. of International Conference on Communication Technology. Guilin, China: IEEE Press, 2006: 1-4.