

位屏蔽多叉树搜索射频识别防碰撞算法

莫 磊, 陈 伟, 任 菊

(1. 成都航空职业技术学院信息工程学院, 四川成都 610100; 2. 四川省高校校企联合“航空电子技术”应用技术创新基地, 四川成都 610100)

摘 要: 针对 RFID 树型防碰撞算法中时隙数多、数据通行量大等问题, 提出了一种改进的多叉树防碰撞算法, 阅读器准确检测碰撞位并向标签反馈碰撞位信息, 标签对阅读器已知的 ID 位进行屏蔽, 把 ID 号转换成连续碰撞的序列号. 阅读器利用屏蔽位信息和标签返回的碰撞位编码信息, 对标签进行分层分类搜索. 通过对标签 ID 进行屏蔽, 阅读器和标签间仅发送对方不知道的碰撞位信息. 该算法减少了碰撞时隙和识别时隙, 避免了空闲时隙, 减少了阅读器和标签间的数据通信量. 理论分析和仿真结果表明, 该算法减少了系统的时隙总数和数据通信量, 提高了阅读器的识别效率.

关键词: 位屏蔽; 射频识别; 防碰撞; 多叉树; 预查询

中图分类号: TP301.6 **文献标识码:** A **文章编号:** 0372-2112 (2018)05-1200-07

电子学报 URL: <http://www.ejournal.org.cn> **DOI:** 10.3969/j.issn.0372-2112.2018.05.027

A RFID Anti-collision Algorithm Based on Bit-shield and Multi-tree Search

MO Lei, CHEN Wei, REN Jü

(1. Chengdu Aeronautic Polytechnic, Chengdu, Sichuan 610100, China;

2. The Innovation Base of School - Enterprise Cooperation Aviation Electronic Technology in Sichuan, Chengdu, Sichuan 610100, China)

Abstract: Focusing on the problem of multi slots and large amount of data traffic in RFID tree anti-collision algorithm, an improved anti-collision algorithm based on multi-tree is proposed. The reader can detect the collision accurately and send feedback to the tags regarding the collision bits information. The tag shields the known ID bits of the reader and converts the ID number into the serial number of consecutive collisions. The reader makes use of shield bits information and the collision bits coding information returned by the tags in order to search the tags in a hierarchical classified manner. By shielding the tag ID, the reader and the tag only sends the collision bits information which the other party unknown. The algorithm reduces the collision time slots and the recognition time slots, avoids the idle time slots, and reduces the communication data between the reader and the tags. Theoretical analysis and simulation results show that the algorithm reduces the total slots and data traffic, and improves the recognition efficiency of reader.

Key words: bit-shield; RFID; anti-collision; multi-tree; pre query

1 引言

射频识别 (Radio frequency identification, RFID) 技术是物联网的核心技术之一, 是一种非接触的自动识别技术, 典型的射频识别系统一般由电子标签和阅读器组成^[1], 射频识别技术中最关键的问题是多标签识别问题, 由于多个标签共用一个信道, 当多个标签同时向阅读器发送数据时, 就会引起数据冲突或碰撞, 导致 RFID 系统不能正常工作, 解决冲突的方法就是防碰撞

方法^[2]. 大多数标签为无源标签, 能量来自阅读器, 处理能力弱, 不具备载波侦测能力, 很多传统的防碰撞方法不能用于 RFID 系统, 现阶段的防碰撞方法主要是基于时分多路的方法, 主要有: 基于 ALOHA 的不确定性算法和基于搜索树的确定性算法^[3]. 基于 ALOHA 的算法相对较简单, 易于实现, 但由于其时隙是随机分配的, 存在某一标签在长期内不能被识别而导致的“饥饿”问题, 而且随着标签数量的增加, 其性能急剧恶化^[4]. 基于搜索树的算法具有 100% 的标签识别率, 但存在搜索

次数多,识别时间长等问题^[5]。

常用的搜索树算法有:基本二进制搜索树(Binary Search, BS)算法^[6],动态二进制搜索树(Dynamic Binary Search, DBS)算法^[7],跳跃式动态搜索树(Jumping and Dynamic searching, JDS)算法^[8]等,其中 DBS 算法为 ISO/IEC1443A 标准推荐的算法,但在实际应用中,这些算法难以完全满足要求,为了提高搜索树算法的识别效率,很多学者对此展开了研究,文献[9]中提出一种四叉树搜索算法,根据碰撞位信息进行四叉树搜索,相对二叉树搜索,减少了搜索时隙,但增加了空闲时隙,文献[10]中提出了一种基于多叉树搜索的防碰撞算法,根据碰撞位的不同来动态选择二叉树搜索或四叉树搜索,但仍有很多空闲时隙,文献[11]提出 IAMS 算法,通过碰撞因子来动态选择二叉树或四叉树搜索,并对四叉树的查询前缀进行优化,以避免空闲时隙,文献[12]指出 IAMS 算法存在碰撞因子减少了利用四叉树减少碰撞时隙机会等问题,并提出了 CCBBD 算法,利用连续碰撞位进行探索查询和多叉树搜索,但是当遇到非连续碰撞位时,这种方法无能为力。

针对这些方法存在的缺陷,本文提出一种新的算法:位屏蔽多叉树搜索射频识别防碰撞算法(A RFID anti-collision algorithm based on bit-shield and multi-tree search, BMS),该算法利用屏蔽寄存器,把阅读器已知的标签信息进行屏蔽,把阅读器未知的标签位组成一个从最低位到最高位都连续碰撞的新序列号,并从最高两个碰撞位开始进行预查询和四叉树搜索,避免了空闲时隙,提高了搜索效率。

2 位屏蔽多叉树防碰撞算法

BMS 算法的指导思想是:尽量减少碰撞时隙,完全避免空闲时隙,尽量减少阅读器和标签间的数据通信量。

标签采用曼彻斯特编码,以便阅读器能够准确的检测碰撞位。

为了减少碰撞时隙,在标签中设置屏蔽寄存器,把阅读器已知的非碰撞位屏蔽起来,把剩下的阅读器未知的 ID 位组成一个完全连续碰撞的新序列号,并采用后退式四叉树进行搜索;为了避免空闲时隙,对最高两个碰撞位进行预查询,以获知标签最高两个碰撞位的准确信息;为了减少数据通信量,通过屏蔽寄存器,标签只发送阅读器不知道的碰撞位信息,阅读器在屏蔽某层标签后,只发送该层标签的非屏蔽位个数信息。

2.1 算法命令

2.1.1 屏蔽寄存器

为了实现本算法,在标签中设置屏蔽寄存器 S,其长度同标签 ID 的长度,屏蔽寄存器 S 的数据和标签 ID

一一对应,与屏蔽寄存器 S 的‘0’对应的标签 ID 位为屏蔽位,与屏蔽寄存器 S 的‘1’对应的标签 ID 位为非屏蔽位,由标签 ID 的非屏蔽位组成的序列号称为非屏蔽序列号。

增加屏蔽寄存器,在提高系统性能的同时,也对标签性能提出了更高的要求,但屏蔽寄存器占用存储空间小,计算量不大,以较小的代价获得了较高的性能。

2.1.2 编码和译码

为了准确的检测标签最高两个碰撞位的数据信息,需要对标签最高两个碰撞位进行编码和译码。

在 BMS 算法中,非屏蔽位就是相应子集标签的碰撞位,因此,最高两个碰撞位就是最高两个非屏蔽位。

设标签最高两个碰撞位为 A_1A_0 ,对应编码为 $B_3B_2B_1B_0$,则编码方法如表 1 所示。

表 1 编码对应表

A_1A_0	00	01	10	11
$B_3B_2B_1B_0$	0001	0010	0100	1000

阅读器接收编码数据,由碰撞位信息就可知道最高两个碰撞位信息,这一过程就是译码,例如:假设接收数据为 0XXX,则译码为:“00”、“01”、“10”。

2.1.3 阅读器命令

为了描述算法方便,定义了以下几个阅读器请求命令。

(1) REQUEST1(NULL):首次搜索命令,阅读器作用范围内所有标签响应命令。

(2) REQUEST2(P):屏蔽预查询命令,屏蔽寄存器 S 中‘1’的个数和 P 的长度相等的标签响应命令,分三步进行:

(a)更新屏蔽寄存器 S:用 P 的各位取代 S 的对应‘1’位。

(b)编码:对最高两个非屏蔽位进行编码。

(c)标签返回编码给阅读器。

(3) REQUEST3(Q,L):标签搜索命令,Q 为两位二进制,非屏蔽位最高两位为 Q 且非屏蔽位长度为 L 的标签响应命令,分两步进行:

(a)更新屏蔽寄存器 S:非屏蔽位最高两位转为屏蔽位(即 S 最高两个‘1’清零)。

(b)发送非屏蔽序列号给阅读器。

2.2 算法步骤

阅读器设置堆栈,按先进后出的规则存取数据。

阅读器设深度寄存器,以区分不同搜索深度的标签。

Step1:阅读器初始化堆栈为空,阅读器发送请求命令 REQUEST1(NULL)。

Step2:阅读器作用范围内所有标签返回 ID 数据,

同时初始化屏蔽寄存器 S 为全“1”。

Step3: 阅读器接收数据, 如果没有碰撞位, 则识别一个标签, 转至 Step6; 如果有一个碰撞位, 则识别两个标签, 转至 Step6; 如果有两个或两个以上的碰撞位, 设碰撞位个数为 L , 则更新深度寄存器的值也为 L ; 令碰撞位为‘1’, 其余位为‘0’, 可得到一个序列号, 设为 P , 阅读器发送请求命令 REQUEST2(P)。

Step4: S 中‘1’的个数和 P 的长度相等的标签更新屏蔽寄存器 S , 对最高两个非屏蔽位编码, 返回编码数据给阅读器。

Step5: 阅读器接收数据并译码, 得到标签最高两个碰撞前缀信息, 即: “00”、“01”、“10”、“11”中的两个、三个或四个, 取出深度寄存器值 L , 把“前缀 + L ”分别写入堆栈。

Step6: 阅读器读取堆栈数据, 如果堆栈为空, 搜索结束; 如果堆栈不为空, 读取栈首前缀和 L 值, 如果 L 值为 2, 则识别标签, 重新开始 Step6; 如果 L 值大于 2, 设前缀为 Q , 阅读器发送请求命令 REQUEST3(Q, L)。

Step7: 非屏蔽位最高两位为 Q 且非屏蔽位长度为 L 的标签响应命令, 先更新屏蔽寄存器 S , 再发送非屏蔽序列号给阅读器, 转至 Step3。

由以上步骤可以看出, 通过屏蔽寄存器, 能够确保标签每次处理分析的序列号都是连续碰撞的序列号, 标签每次发送的数据都是阅读器不知道的碰撞位信息, 避免重复发送阅读器已知的数据信息; 阅读器也仅需发送碰撞位位置信息或碰撞位个数信息, 避免重复发送标签已知的数据信息; 标签往下逐级分层搜索, 通

过深度寄存器, 能够确保标签在完成本层标签识别后, 能够后退到上一层进行搜索, 以便控制标签进行后退式搜索并减少标签搜索次数。

在 Step3 中, 如果只有一个碰撞位, 则可判定总共有两个标签, 一个标签碰撞位为‘1’, 一个标签碰撞位为‘0’, 故可直接识别两个标签。

在 Step6 中, 如果 L 为 2 (L 不可能小于 2), 则只有两个碰撞位, 通过 Step5, 这两个碰撞位已被识别, 故可直接识别多个标签 (2 个、3 个或 4 个)。

2.3 算法举例

假设有 8 个标签, 标签 ID 号如表 2 所示。

表 2 标签及其序列号

标签	序列号	标签	序列号
A	1100110011	E	1101110110
B	1000110101	F	1001011000
C	1001011010	G	1100010011
D	1100010111	H	1001010010

BMS 算法识别过程如表 3 所示。由表 3 可以看出, 8 个标签, 总共用了 8 个时隙。在时隙 5, 阅读器接收数据为 XXX0, 译码得到“11”、“10”、“01”, 并与深度寄存器值 2 一起分别写入堆栈, 接着, 取出栈首数据: “01”和 2, 由于深度寄存器值为 2, 说明标签只有两个碰撞位, 且这两个碰撞位为“01”, 则识别标签 H; 同理, 堆栈弹出数据“10”和 2 可识别标签 F; 堆栈弹出数据“11”和 2 可识别标签 C。标签 A、D、G 的识别过程与标签 C、F、H 类似; 标签 B、E 无碰撞, 直接识别。

表 3 BMS 算法识别过程

搜索次数	请求命令	屏蔽寄存器 R	标签返回数据	阅读器接收数据	堆栈	深度寄存器	识别标签
1	REQUEST1 (NULL)	A:1111111111 B:1111111111 C:1111111111 D:1111111111 E:1111111111 F:1111111111 G:1111111111 H:1111111111	A:1100110011 B:1000110101 C:1001011010 D:1100010111 E:1101110110 F:1001011000 G:1100010011 H:1001010010	1X0XX1XXXX		7	
2	REQUEST2 (0101101111)	A:0101101111 B:0101101111 C:0101101111 D:0101101111 E:0101101111 F:0101101111 G:0101101111 H:0101101111	A:0100 B:0001 C:0010 D:0100 E:1000 F:0010 G:0100 H:0010	XXXX	00,7 01,7 10,7 11,7	7	

续表

搜索次数	请求命令	屏蔽寄存器 R	标签返回数据	阅读器接收数据	堆栈	深度寄存器	识别标签
3	REQUEST3(00,7)	B:0000101111	B:10101	B:10101	01,7 10,7 11,7	7	B
4	REQUEST3(01,7)	C:0000101111 F:0000101111 H:0000101111	C:01010 F:01000 H:00010	0X0X0	10,7 11,7	2	
5	REQUEST2(01010)	C:0000001010 F:0000001010 H:0000001010	C:1000 F:0100 H:0010	XXX0	10,7 11,7	2	CFH
6	REQUEST3(10,7)	A:0000101111 D:0000101111 G:0000101111	A:10011 D:00111 G:00011	X0X11	11,7	2	
7	REQUEST2(10100)	A:0000100100 D:0000100100 G:0000100100	A:0100 D:0010 G:0001	0XXX	11,7	2	ADG
8	REQUEST3(11,7)	E:0000101111	E:10110	10110		2	E

3 性能分析

下面对 BMS 算法进行性能分析。

对于满四叉树,在 X 层有 4^X 个节点,在 4^X 个节点中选择一个节点的概率为 $P = 4^{-X}$, K 个标签中有 Y 个标签在 X 层选择同一节点的概率服从二项分布^[13]:

$$P(Y/K) = C_K^Y P^Y (1-P)^{K-Y} = C_K^Y 4^{-XY} (1-4^{-X})^{K-Y} \quad (1)$$

若 $Y=0$,表示空闲时隙概率 $P(0/K)$;若 $Y=1$,表示识别时隙概率 $P(1/K)$,则碰撞时隙概率为:

$$P(Y>1/K) = 1 - P(0/K) - P(1/K) \quad (2)$$

假设同一层每个节点发生碰撞的概率相同,四叉树的平均碰撞时隙总数为:

$$\begin{aligned} C(K) &= \sum_{X=0}^{\infty} \sum_{i=1}^{4^X} P(Y>1/K) \\ &= \sum_{X=0}^{\infty} \sum_{i=1}^{4^X} (1 - P(0/K) - P(1/K)) \\ &= \sum_{X=0}^{\infty} \sum_{i=1}^{4^X} (1 - (1 - 4^{-X})^K - C_K^1 4^{-X} (1 - 4^{-X})^{K-1}) = \sum_{X=0}^{\infty} 4^X [1 - (1 - 4^{-X})^K - K 4^{-X} (1 - 4^{-X})^{K-1}] \end{aligned} \quad (3)$$

BMS 算法的时隙总数由碰撞时隙、预查询时隙和识别时隙组成,如果不考虑只有一个碰撞位时直接识别两个标签和只有两个碰撞位时,通过预查询直接识别多个标签的情况,则预查询时隙和碰撞时隙相等,识

别时隙为 K ,总时隙数为:

$$T'(K) = 2C(K) + K \quad (4)$$

设只有一个碰撞位时直接识别的标签数为 T_1 ,下面计算 T_1 .

只有一个碰撞位时,一定是两个标签发生碰撞,只有两个标签发生碰撞的时隙总数为:

$$C_2(K) = \sum_{X=0}^{\infty} \sum_{i=1}^{4^X} P(2/K) \quad (5)$$

设标签 ID 各位是相互独立的随机值,则两个标签之中任意一位不会发生碰撞的概率服从二项分布:

$$\begin{aligned} P'_2 &= 2C_2^2 (2^{-1})^2 (1-2^{-1})^{2-2} \\ &= 2^{-1} \end{aligned} \quad (6)$$

则两个标签之中任意一位发生碰撞的概率为:

$$\begin{aligned} P_2 &= 1 - P'_2 \\ &= 2^{-1} \end{aligned} \quad (7)$$

设标签 ID 的长度为 L ,则两个标签之中,只有一位发生碰撞的概率服从二项分布:

$$\begin{aligned} P_{12} &= C_L^1 (P_2)^1 (1-P_2)^{L-1} \\ &= L 2^{-L} \end{aligned} \quad (8)$$

由于发生一次碰撞可以识别两个标签,所以:

$$\begin{aligned} T_1 &= 2C_2(K) P_{12} \\ &= 2L 2^{-L} \sum_{X=0}^{\infty} \sum_{i=1}^{4^X} C_K^2 4^{-2X} (1-4^{-X})^{K-2} \\ &= 2L 2^{-L} \sum_{X=0}^{\infty} C_K^2 4^{-X} (1-4^{-X})^{K-2} \end{aligned} \quad (9)$$

设只有两个碰撞位时,通过预查询直接识别的标签数为 T_2 ,下面计算 T_2 .

当有两个碰撞位时,发生碰撞的标签数不小于 2 个,不大于 4 个.

只有两个标签发生碰撞的时隙总数见式(5).

只有 3 个标签发生碰撞的时隙总数为:

$$C_3(K) = \sum_{X=0}^{\infty} \sum_{i=1}^{4^X} P(3/K) \quad (10)$$

只有 4 个标签发生碰撞的时隙总数为:

$$C_4(K) = \sum_{X=0}^{\infty} \sum_{i=1}^{4^X} P(4/K) \quad (11)$$

先分析两个标签碰撞的情况:两个标签之中,只有 2 位发生碰撞的概率为:

$$\begin{aligned} P_{22} &= C_L^2(P_2)^2(1-P_2)^{L-2} \\ &= L(L-1)2^{-(L+1)} \end{aligned} \quad (12)$$

再分析三个标签碰撞的情况:三个标签之中任意一位不会发生碰撞的概率为:

$$\begin{aligned} P_3' &= 2C_3^3(2^{-1})^3(1-2^{-1})^{3-3} \\ &= 4^{-1} \end{aligned} \quad (13)$$

三个标签之中任意一位发生碰撞的概率为:

$$P_3 = 1 - P_3' = \frac{3}{4} \quad (14)$$

三个标签之中,只有两位发生碰撞的概率为:

$$\begin{aligned} P_{23} &= C_L^2(P_3)^2(1-P_3)^{L-2} \\ &= 9L(L-1)2^{-(2L+1)} \end{aligned} \quad (15)$$

最后分析四个标签碰撞的情况:四个标签之中任意一位不会发生碰撞的概率为:

$$\begin{aligned} P_4' &= 2C_4^4(2^{-1})^4(1-2^{-1})^{4-4} \\ &= 8^{-1} \end{aligned} \quad (16)$$

四个标签之中任意一位发生碰撞的概率为:

$$P_4 = 1 - P_4' = \frac{7}{8} \quad (17)$$

四个标签之中,只有两位发生碰撞的概率为:

$$\begin{aligned} P_{24} &= C_L^2(P_4)^2(1-P_4)^{L-2} \\ &= 49L(L-1)2^{-(3L+1)} \end{aligned} \quad (18)$$

则只有两个碰撞位时,通过预查询直接识别的标签数为:

$$\begin{aligned} T_2 &= 2C_2(K)P_{22} + 3C_3(K)P_{23} + 4C_4(K)P_{24} \\ &= 2L(L-1)2^{-(L+1)} \sum_{X=0}^{\infty} \sum_{i=1}^{4^X} P(2/K) \\ &\quad + 27L(L-1)2^{-(2L+1)} \sum_{X=0}^{\infty} \sum_{i=1}^{4^X} P(3/K) \\ &\quad + 196L(L-1)2^{-(3L+1)} \sum_{X=0}^{\infty} \sum_{i=1}^{4^X} P(4/K) \\ &= 2L(L-1)2^{-(L+1)} \sum_{X=0}^{\infty} C_K^2 4^{-X} (1-4^{-X})^{K-2} \end{aligned}$$

$$\begin{aligned} &+ 27L(L-1)2^{-(2L+1)} \sum_{X=0}^{\infty} C_K^3 4^{-2X} (1-4^{-X})^{K-3} \\ &+ 196L(L-1)2^{-(3L+1)} \sum_{X=0}^{\infty} C_K^4 4^{-3X} (1-4^{-X})^{K-4} \end{aligned} \quad (19)$$

可得到 BMS 算法总时隙数为:

$$\begin{aligned} T(K) &= T'(K) - T_1 - T_2 \\ &= 2 \sum_{X=0}^{\infty} 4^X [1 - (1-4^{-X})^K - K4^{-X}(1-4^{-X})^{K-1}] \\ &\quad + K - L(L+1)2^{-L} \sum_{X=0}^{\infty} C_K^2 4^{-X} (1-4^{-X})^{K-2} \\ &\quad - 27L(L-1)2^{-(2L+1)} \sum_{X=0}^{\infty} C_K^3 4^{-2X} (1-4^{-X})^{K-3} \\ &\quad - 196L(L-1)2^{-(3L+1)} \sum_{X=0}^{\infty} C_K^4 4^{-3X} (1-4^{-X})^{K-4} \end{aligned} \quad (20)$$

吞吐率为:

$$S_{RMS} = \frac{K}{T(K)} \quad (21)$$

4 算法仿真

利用 MATLAB 对 DBS 算法、JDS 算法、IAMS 算法、CCBD 算法和 BMS 算法进行仿真和比较,假设标签 ID 为 96bit,标签数量为 10-100 个,取 10 次仿真实验的平均值作为仿真结果.

图 1 是几种算法的总时隙比较图,由图可见,BMS 算法总时隙明显小于 IAMS 算法和 CCBD 算法,更远小于 DBS 算法和 JDS 算法,这是由于 BMS 算法利用位屏蔽形成连续的碰撞位,利用四叉树减少碰撞时隙,利用预查询避免空闲时隙和减少识别时隙;当标签数量为 100 时,BMS 算法总时隙数为 131,IAMS 算法和 CCBD 算法分别为 191 和 153,BMS 算法的总时隙数相比 IAMS 算法和 CCBD 算法分别减少了 31% 和 14%.

图 2 是几种算法吞吐率比较图,由图可见,在标签数量 10-100 之间,DBS 算法、JDS 算法、IAMS 算法和 CCBD 算法吞吐率变化比较平稳,BMS 算法随着标签数量的增大有略微增大的趋势,BMS 算法吞吐率明显大于其他几种算法,这是由于 BMS 算法具有更少的总时隙数;当标签数量为 100 时,BMS 算法吞吐率为 0.77,IAMS 算法和 CCBD 算法吞吐率分别为 0.52 和 0.65,BMS 算法吞吐率相比 IAMS 算法和 CCBD 算法分别提高了 48% 和 18%.

图 3 是几种算法阅读器和标签间的数据通信量比较,由图可见,BMS 算法相比其他几种算法在减少数据通信量方面有较大优势,这是由于 BMS 算法减少了总时隙数,同时,通过位屏蔽,也减少了阅读器和标签间

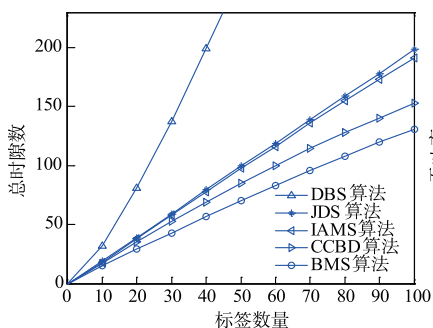


图1 总时隙数比较

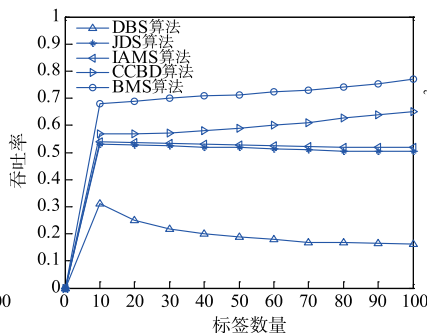


图2 吞吐率比较

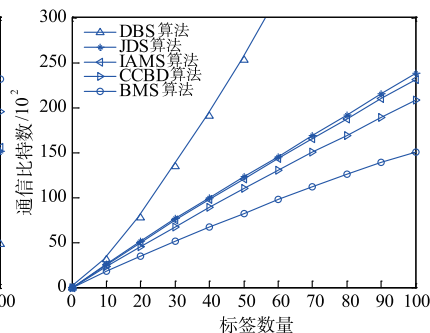


图3 通信数据量比较

的通信比特数;CCBD 算法由于探测命令较长以及总时隙减少不明显,仍有较大的通信比特数,IAMS 算法通信比特数略小于 JDS 算法,DBS 算法由于搜索次数多,其通信比特数最高;当标签数量为 100 时,BMS 算法通信量为 15003bit,IAMS 算法和 CCB 算法分别为 23131bit 和 20878bit,BMS 算法的总时隙数相比 IAMS 算法和 CCB 算法分别减少了 35% 和 28%。

5 结束语

本文提出了一种新的基于位屏蔽和多叉树搜索的 RFID 防碰撞算法,通过标签中的位屏蔽寄存器,把需要处理的标签 ID 号转换成连续碰撞的序列号,再利用多叉树搜索和高碰撞位预查询来减少碰撞时隙、识别时隙和避免空闲时隙;通过位屏蔽寄存器,把阅读器已知的信息进行屏蔽,标签仅发送阅读器不知道的信息,阅读器也仅发送标签不知道的信息,使阅读器和标签间的数据通信比特数得到减少.这种方法克服了 IAMS 算法和 CCB 算法的不足,有效的减少了总时隙数,提高了吞吐率、减少了通信复杂度,提高了 RFID 系统的防碰撞性能。

参考文献

[1] 王心妍,杨博. 基于多进制查询树的多标签识别方法[J]. 计算机工程,2015,41(8):95-99.
Wang Xin-yan, Yang Bo. Multi-tag identification method based on multi-ary query tree[J]. Computer Engineering, 2015,41(8):95-99. (in Chinese)

[2] 韦冬雪,郑嘉利,黄庆欢,等. 基于搜索树的增强型 RFID 防碰撞算法[J]. 计算机应用与软件,2015,32(11):226-231.
Wei Dong-xue, Zheng Jia-li, Huang Qing-huan, et al. An enhanced RFID anti-collision algorithm based on search tree[J]. Computer Applications and Software, 2015, 32(11):226-231. (in Chinese)

[3] 邓红卫,梁小满,许航,等. 一种新型的混合查询树防碰撞算法[J]. 衡阳师范学院学报,2015,26(3):40-44.

Deng Hong-wei, Liang Xiao-man, Xu Hang, et al. A new hybrid query tree anti-collision algorithm [J]. Journal of Hengyang Normal University, 2015, 26(3):40-44. (in Chinese)

[4] 宋建华,郭亚军,韩兰胜,等. 自调整混合树 RFID 多标签防碰撞算法[J]. 电子学报,2014,42(4):685-689.
Song Jian-hua, Guo Ya-jun, Han Lan-sheng, et al. An Adaptive hybrid tree anti-collision algorithm for RFID multi-tag identification [J]. Acta Electronica Sinica, 2014, 42(4):685-689. (in Chinese)

[5] 李志坚,肖熠琳. 一种基于二进制码调制的射频识别防碰撞算法[J]. 计算机工程,2015,41(2):308-312.
Li Zhi-jian, Xiao Yi-lin. A radio frequency identification anti-collision algorithm based on binary code modulation [J]. Computer Engineering, 2015, 41(2):308-312. (in Chinese)

[6] FINKENZELLER K. RFID Handbook; Radio-Frequency Identification Fundamentals and Applications (2nd ed) [M]. John Wiley and sons Ltd,2003.

[7] LAW C. LEE K. SIU K Y. Efficient memoryless protocol for tag identification[A]. Proc 4th International Workshop on DIALM[C]. Boston,2000. 75-84.

[8] Yu S, Zhan Y, Wang Z, et al. Anti-collision algorithm based on jumping and dynamic searching and its analysis [J]. Computer engineering,2005,31(9):19-20.

[9] 李宝山,乔聪. 改进的二进制搜索防冲突算法[J]. 微电子学与计算机,2014,31(5):94-97.
Li Bao-shan, Qiao Cong, An improved binary search anti-collision algorithm [J]. Microelectronics & Computer, 2014,31(5):94-97. (in Chinese)

[10] 林伟,李景霞,叶林锋. 基于多叉树搜索算法改进的 RFID 防碰撞算法[J]. 电子技术应用,2013,39(2):130-133.
Lin Wei, Li Jing-xia, Ye Lin-feng. An improved anti-collision algorithm based on multi tree search in RFID [J]. Computer Technology and its Applications, 2013, 39(2):130-133. (in Chinese)

[11] 张学军,蔡文琦,王锁萍. 改进型自适应多叉树防碰撞

算法研究[J]. 电子学报, 2012, 40(1): 193 - 198.

Zhang Xue-jun, Cai Wen-qi, Wang Suo-ping. One anti-collision algorithm based on improved adaptive multi-tree search[J]. Acta Electronica Sinica, 2012, 40(1): 193 - 198. (in Chinese)

- [12] 刘子龙, 纪金水, 刘彩虹, 等. 基于连续碰撞位探测的防碰撞算法研究[J]. 电子学报, 2013, 41(11): 2156 - 2160.

Liu Zhi-long, Ji Jing-shui, Liu Chai-hong, et al. An anti-collision algorithm based on continuous collision bit detection[J]. Acta Electronica Sinica, 2013, 41(11): 2156 - 2160. (in Chinese)

- [13] D R Hush, C Wood. Analysis of tree algorithms for RFID arbitration[A]. Proceedings of IEEE Symposium on Information Theory (ISIT) [C]. Cambridge, MA, USA, 1998. 107 - 116.

作者简介



莫 磊 男, 1969 年出生于四川省遂宁市. 1991 年在重庆大学获得工学学士学位, 现为成都航空职业技术学院副教授, 主要研究方向为物联网技术、RFID 技术.
E-mail: nqnt@163.com



陈 伟(通信作者) 男, 1978 年出生于山东泗水, 2008 年在成都理工大学获得硕士学位, 现为成都航空职业技术学院讲师, 主要研究方向为信号获取预处理、核测控技术.
E-mail: ndew@qq.com