

一种新颖的 RFID 防冲突算法

张明, 张建华, 徐国鑫, 张平

(北京邮电大学 92# 无线新技术研究室, 北京 100876)

摘要: 提出一种应用于 RFID 系统上行链路的多标签冲突检测算法, 并给出了参考实现电路。依算法, 对电子标签进行随机分群, 在群间做随机避让, 在群内进行冲突检测和标签的仲裁。与现有的随机避让算法和二进制树算法相比, 该算法不但有效降低了碰撞次数, 实现电路也更简单。

关键词: RFID 电子标签 防冲突算法 仲裁

无线射频识别 (RFID) 是一种非接触式的自动识别技术, 其基本原理是利用射频信号和空间耦合 (电感或电磁耦合) 的传输特性, 实现对特定物体的自动识别。RFID 技术可以追溯至第二次世界大战期间, 后来发展到应用到铁路、军队的货物跟踪甚至宠物识别上。在过去的半个多世纪里, RFID 的发展经历了从技术探索、试验研究、商业应用和标准化建立等几个重要阶段。从现有发展趋势看, RFID 将构建虚拟世界与物理世界的桥梁。可以预见在不久的将来, RFID 技术不仅会在各行各业被广泛采用, 最终 RFID 技术还将会与普适计算技术相融合, 对人类社会产生深远影响。

RFID 系统一般由电子标签和读写器两个部分组成, 读写器具有同时读取多个电子标签的功能。在多标签对一个读写器的 RFID 系统中, 标签经常会同时向读写器传输数据, 这就要求 RFID 系统建立一种仲裁机制来避免数据发生碰撞。考虑到电子标签本身尺寸、能耗的限制, 防碰撞机制在保障功能的同时还要求尽量简单易行, 这正是 RFID 系统设计的挑战之一。

文献[1]和[2]提出了三种 RFID 系统的防碰撞算法。算法 A 基于随机避让、冲突检测的原理, 使用 1 个 8 位寄存器和 1 个 8 位随机数产生器, 最大可以仲裁标签的数量只有 256 个。算法 B 基于二进制数的原理, 使用 1 个 8 位寄存器和 1 个 1 位随机数产生器, 理论上最大可以实现 2^{256} 个标签的仲裁。文献[3]提出了对该算法的一个实现方案, 文献[4]对该算法做了很大改进。算法 C 类似于算法 A, 使用 1 个 16 位寄存器和 16 个 1 位随机数产生器, 最大可以仲裁标签的数量是 65 536 个。本文中, 作者提出一种分群避让、群内冲突检测的算法和其改进算法, 仅需要 1 个 8 位寄存器和 1 个 1 位随机数产生器就可以实现最大 1048 576 个标签的仲裁, 而且碰撞次数相对于算法 B 要大大减少。

1 仲裁机制描述

本方法的核心思想是: 首先把电子标签随机分群,

并将群随机排序以实现群间的随机避让, 然后在群内进行冲突检测和标签的仲裁。实现时标签仅需一个寄存器: 利用其高位存储群号, 低位存储冲突检测时退避的步数, 实现极为简单。下面以 8 位寄存器为例具体说明本算法的仲裁机制。

当读写器初始化标签时, 所有标签在 0~15 之间任选一个整数存入寄存器高 4 位 (相当于随机选择一个群) 并把寄存器低 4 位设为全 0, 同时产生一个 0 或 1 的随机数加到寄存器中。如果此时寄存器中的 8 比特数为全 0 则回传该标签的 ID (ID 是指电子标签的惟一标识, 在不同的编码系统中有不同的含义)。如果多个标签同时回传数据, 则冲突发生。发生冲突后, 其他寄存器高 4 位为 0 的标签寄存器中的数加 1, 而寄存器中的 8 比特数为全 0 的标签则再产生一个 0 或 1 的随机数加到寄存器中。如果寄存器作加法后仍为零, 则继续回传该标签的 ID; 如果回传时不发生碰撞, 则其他寄存器高 4 位为 0 的标签仅把寄存器低 4 位减 1 后重复前面的回传操作。当寄存器高 4 位全为 0 的标签全部回传完 ID, 则所有其他标签把寄存器高 4 位减 1 后重复前面的操作。

此外依本算法, 由于所有标签随机选择群, 有可能出现某个群中的标签数目过大, 使该群中的标签在仲裁过程中始终发生碰撞, 标签寄存器始终加 1, 导致寄存器低 4 位向高 4 位进位。进位意味着所有进位的标签的寄存器低 4 位清零而高 4 位加 1, 这使得这些标签不再属于原有的群而归入到下一个群中, 从而优化了因随机选择而产生的分布不均匀的群标签数。

本算法中, 标签最大退让步数为 $2^4=16$ 步, 因此每个群最大能仲裁的标签数目为 $2^{16}=65\,536$, 则本算法能仲裁的标签数理论上限是 $16 \times 2^{16}=1\,048\,576$ 。

2 算法步骤

本节给出算法步骤, 假设使用一个 8 位寄存器, 则本算法包括以下步骤:

(1) 在所述 RFID 系统的被动方- 标签中设计一个 4+4

位的寄存器 (Re1) 和 1 个“0”、“1”随机数产生器 (RG1), 如图 1 所示。

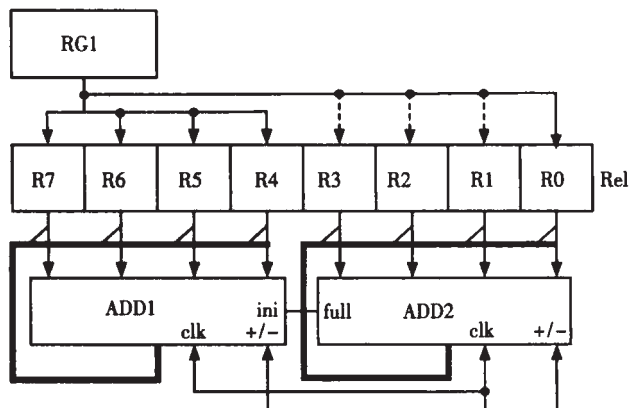


图 1 电子标签中算法实现的电路框图

(2) 在所述 RFID 系统的主动方-读写器向所有处在等待态的标签发送初始化命令。标签因此进入仲裁态, 用 RG1 产生 4 比特随机数, 加载到 Re1 高 4 位 R7~R4, 低 4 位 R3~R0 全部清零。

(3) 读写器等待一定时间后发送允许回传命令。

(4) Re1 为全零的标签向读写器回传标签 ID。

(5) 如果当前只有一个标签回传 ID, 读写器正确读取该 ID, 则发送确认命令, 附加命令参数“低位减 1”。回传了 ID 的标签接收到该命令后, 进入确认态, 其他高 4 位为全零的标签 Re1 低 4 位减 1, 回到步骤(4)重复操作。

(6) 如果当前有多个标签回传 ID, 读写器通过 CRC 校验或码长校验, 检测到错误的 ID 号, 则发送确认命令, 附加命令参数“寄存器加 1”。接收到读写器这个命令后, 所有在仲裁态且 Re1 为全零的标签由 RG1 产生 1 比特随机数和寄存器上的数相加后重新载入到寄存器中; 其他仲裁态且 Re1 高 4 位为零而低 4 位不为零的标签 Re1 加 1, 回到步骤(4)重复操作。

(7) 如果当前没有标签回传 ID, 读写器等待一定时间后发送确认命令, 附加命令参数“低位减 1”。所有在仲裁态且高 4 位为全零的标签 Re1 低 4 位减 1, 回到步骤(4)重复操作。

(8) 低 4 位减 1 操作重复 L 次(L 是一个系统参数, 由系统设定, 经验值为 4)后, 读写器认为所有在仲裁态且寄存器高 4 位为零的标签都已经被正确读取, 则发送确认命令, 附加命令参数“高 4 位减 1”, 回到步骤(4)。

(9) 标签接收到附加“高位减 1”参数的确认命令后, 所有 Re1 高 4 位不为零的标签高 4 位减 1, 回到步骤 4 重复操作; 在被要求高位减 1 前已为零的标签则回到等待态。

(10) 重复 15 次高位减 1 操作后, 读写器认为所有在仲裁态的标签都已经被读取, 则仲裁过程停止, 所有

还处于仲裁态的标签返回等待态。

算法步骤中所述等待态是指电子标签上电后的初始状态; 仲裁态是指未被读写器鉴别的电子标签开始响应读写器鉴别命令时进入的状态; 确认态是指已被读写器鉴别的电子标签进入的状态。电子标签状态转移规则如下: 上电后电子标签进入等待态; 处于等待态的电子标签可以进入仲裁态; 处于仲裁态的电子标签可以返回等待态; 处于仲裁态的电子标签可以进入确认态; 处于确认态的电子标签不能返回仲裁态; 确认态与等待态之间不能直接转移。

针对上述算法步骤, 在以下几个地方加以改进, 形成本算法的改进算法。

A、步骤(1)中, 随机数产生器产生两组随机数, 分别加载到寄存器高位和低 4 位。其中高位加载的位数 M 可以动态设为 1、2、3 或 4。

B、步骤(10)中的重复操作次数为 2^M 次。因为改进算法在寄存器低 4 位也加载了随机数, 使得标签在群间转移的概率(即低 4 位向高位进位的概率)大大增加, 特别如果最后一个群中标签的寄存器低 4 位在退让步骤中进位, 则会产生一个新群, 因此需要额外增加一次高位减 1 操作。

3 电路实现

算法实现的参考电路框图见图 1, 其中 RG1 是一个“0”“1”随机数产生器; Re1 是一个 8 位寄存器。加法器 ADD1 和 ADD2 的加减功能根据读写器命令来设定: 当执行加法操作时, 低 4 位的 ADD2 需向高 4 位的 ADD1 进位; 当执行减法操作时, 两个器件 ADD1 和 ADD2 相互独立。加法器可以工作在同步状态或异步状态, 工作在同步状态时可以使用电子标签的最大时钟。

4 仿真结果

仿真 1: 为了评估本算法的优劣, 特设计以下仿真: 标签使用 8 位寄存器, 高 4 位为高位。定义 0 个、2 个和 2 个以上的标签同时发送数据时为传输冲突; 只有一个标签发送数据时为传输成功, 平均冲突次数定义为传输冲突总次数和传输成功总次数的比; 空传率定义 0 个标签发送数据的次数与传输成功总次数的比。观察标签数为 20~10 000 时的平均冲突次数。

仿真结果如图 2 所示, 本文提出的算法与二进制算法性能接近, 平均每成功传输 1 次都要伴随 2 次传输冲突; 而改进算法则在标签数为 50~5 000 个时明显减少了碰撞次数。同时也注意到当标签个数少于 50 时, 改进算法性能下降, 这是因为此时标签数接近分群的群数导致空传率上升所致, 解决的办法是减少分群的群数。针对该问题, 特设计仿真 2 来分析。

仿真 2: 为了分析低标签密度时改进算法的性能, 特设计以下仿真: 采用改进算法, 分别使用 5~8 位寄存器, 高 1~4 位为高位, 即分群群数分别为 2、4、8 和 16。

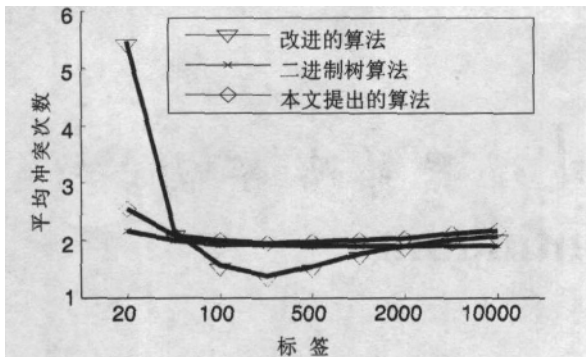


图2 采用三种算法时标签个数和碰撞次数的关系

仿真结果如图3所示,可以看到当标签总数为20时,如果把高位寄存器的位数从4降到1,则平均碰撞次数从5.5回落到1.4。而当标签总数为200和2000时,高位寄存器位数的改变对平均碰撞次数的影响不大。因此如果在某次仲裁中出现多次空传,根据这个先验知识,读写器可以在下一次仲裁时指示标签改变寄存器高位个数,以此降低空传率,进而可以降低平均碰撞次数。

本文提出的防碰撞算法仅需在电子标签中配置1个8位寄存器、1个1位“0”、“1”随机数产生器和2个4位加减1计数器以及少量选择电路就能实现最多达1048576个标签的仲裁。仿真表明本算法产生的碰撞概率明显小于二进制数算法,同时通过寄存器高位的灵活设置,还能有效解决低标签密度时空传率高的问题,从而进一步降低了碰撞概率。本算法实现简单,复杂度低,非常适合在RFID系统中应用,因而具有广阔的应用前景。

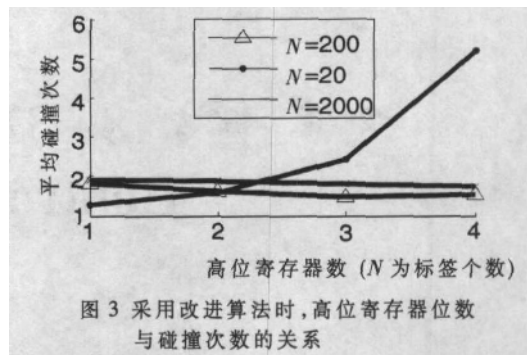


图3 采用改进算法时,高位寄存器位数与碰撞次数的关系

参考文献

- 1 ISO18000-6C 标准: Information technology- Radio- frequency identification for item management - Part 6C: Parameters for air interface communications at 860 MHz to 960MHz.
- 2 ISO18000-6A/B 标准: Information technology automatic identification and data capture techniques - Radio frequency identification for item management air interface- Part 6: Parameters for air interface communications at 860- 960MHz.
- 3 Fukumizu Y, Ohno S, Nagata M, etc. A highly collision resistive RFID system. In: proc. of APSITT 2003, Nov.2003: 223 ~228
- 4 Zhou, F, Jing, D, Huang C et al. Optimizing the power consumption of passive electronic tags for Anti-collision schemes. In: Proceedings of the 5th ASCON, Beijing, China, October 2003:1213 ~1217

(收稿日期: 2006- 01- 30)

Altera 在 Quartus II 软件 6.0 高密度设计上实现重大改进

在设计软件中为 Synopsys 设计约束 (SDC) 格式提供自然、全面支持的第一个 FPGA 供应商

2006 年 5 月 9 日, 香港- Altera 公司 (NASDAQ: ALTR) 近日宣布开始发售 6.0 版的 Quartus® II 软件。该版本包括了由 FPGA 供应商提供的第一款时序分析工具 TimeQuest 时序分析仪, 为业界标准 Synopsys 设计约束 (SDC) 时序格式提供自然、全面的支持。这一最新版本还包括扩展的团队设计功能, 能够有效管理高密度设计团队之间的协作。这些改进迎合了当今高密度 90nm 的设计要求, 同时为满足客户对更高密度 FPGA 的需求以及 Altera 发展下一代 65nm 产品系列打下了基础。

Synopsys 战略联盟总监 Lonn Fiance 评论说: “FPGA 设计人员将业界标准 SDC 时序约束格式直接读取到 TimeQuest 时序分析仪中, 能够更迅速的实现时序逼近。采用 SDC 格式可以提高 FPGA 设计人员的效率, 进一步促进标准时序验证方法在半导体业界的应用。”

Altera 在最新版 Quartus II 软件中引入了新的技术改进以满足客户对 90nm 的需求, 并为 65nm 工艺节点打下了基础。这些新特性最突出的部分包括: TimeQuest 时序分析仪; 扩展的团队设计支持; SystemVerilog 支持; 增强的 I/O 引脚规划器; 扩展的电路板级设计支持。

关于 Quartus II 软件的详细信息, 请访问 www.altera.com/quartus2。了解 TimeQuest 时序分析仪和其他的 Quartus II 特性, 请访问 Quartus II 软件在线演示网页 www.altera.com/verificationtraining。

(Altera 公司供稿)