

新的自调整多叉树 RFID 防碰撞算法的 FPGA 实现

任少杰¹, 郝永生¹, 许博浩²

(1 军械工程学院, 河北 石家庄 050000;

2 电子信息系统复杂电磁环境效应国家重点实验室, 河南 洛阳 471003)

摘 要: 标签碰撞是射频识别(Radio Frequency Identification, RFID)系统中的关键问题之一, 直接影响系统的效率. 针对标签碰撞问题, 在研究现阶段防碰撞算法的基础上提出了一种新的自调整多叉树 RFID 防碰撞算法, 简要介绍了新算法的基本思想和流程. 设计了基于 FPGA 的新算法实现方案, 并对关键模块功能进行了仿真验证. 仿真结果表明了新算法的优越性和可行性.

关键词: 射频识别; 防碰撞算法; 现场可编程门阵列; 堆栈

中图分类号: TP301.6; TN4

文献标识码: A

文章编号: 1000-7180(2016)06-0049-04

Implementation of the New Anti-collision Algorithm Based on Adjustive Multi-tree for RFID on FPGA

REN Shao-jie¹, HAO Yong-sheng¹, XU Bo-hao²

(1 The Missile Engineering, Ordnance Engineering College, Shijiazhuang 050000, China;

2 State Key Laboratory of Complex Electromagnetic Environment Effects on Electronics and Information System, Luoyang 471003, China)

Abstract: Tag collision is one of the key problem of RFID system, which directly affects the efficiency of the system. For tag collision problem, a new anti-collision algorithm based on adjustive multi-tree was proposed on the basic of studying on present anti-collision algorithm. In this paper, the basic ideas and processes of new anti-collision algorithm were introduced. Then, a new implementation scheme of new anti-collision algorithm based on FPGA was designed and the simulation to the function of the key modules were carried on. The simulation results show the superiority and feasibility of the new algorithm.

Key words: RFID; anti-collision algorithm; FPGA; LIFO

1 引言

射频识别技术是一种可用于信息采集的非接触式自动识别技术, 已广泛应用于门禁、运输、商场、图书馆等多个领域. RFID 系统的工作原理是, 阅读器通过向周围环境广播射频信号来获取其作用范围内标签的响应, 继而完成对标签的识别和数据传输. 但在 RFID 系统的实际应用中, 可能会因为多个标签同时响应阅读器而出现信号混叠, 导致阅读器不能准确地对标签进行识别处理, 这就是标签碰撞问题. 作为解决标签碰撞问题的有效策略, 防碰撞算法是

现阶段该领域的研究热点^[1-3].

2 自调整多叉树防碰撞算法介绍

目前, 最常见的防碰撞算法是基于树型搜索的防碰撞算法, 此类算法能够有效解决“标签饿死”问题, 但算法本身比较复杂, 且存在通信复杂度大、识别时间长等问题^[4-5]. 为此, 本文在对现阶段防碰撞算法进行研究分析的基础上, 提出了一种新的自调整多叉树防碰撞算法(New anti-collision algorithm based on adjustive multi-tree, NAM).

2.1 基本思想

NAM算法的基本思想是阅读器发送查询请求,其作用范围内的所有标签响应,并向阅读器发送自身 EPC 码,阅读器根据碰撞位类型的不同自动选择二叉树或四叉树搜索,具体为:若标签碰撞位只有一位或非连续时,阅读器采用动态二叉树搜索方式,并依据最高碰撞位信息确定两个查询压入堆栈,作为后续查询的请求依次发送给标签;若碰撞位为连续两位,则采用动态四叉树搜索,但此时阅读器将先发送一个异或运算命令给碰撞标签,并依据标签返回消息来确定合适的请求压入堆栈,接着再将其依次发送给标签。

2.2 编码方式

NAM算法实现的基础是准确确定标签碰撞的位置,所以编码方式采用曼彻斯特编码。曼彻斯特编码中逻辑“1”和“0”分别代表电平的下降沿和上升沿跳变,两个以上标签同时向阅读器发送数据时,若数位下降沿和上升沿抵消,就会出现无状态跳变或明显被削弱的状况,此时阅读器即可判断发生碰撞并准确找出碰撞位。不仅如此,相较于差动双向编码、米勒编码和极性归零或不归零编码等其他常见数据编码方式,曼彻斯特编码还具有编码方式简单、稳定性强等优点^[6]。

2.3 命令定义

为便于算法描述,定义表 1 所示命令。

表 1 NAM 算法的命令定义

名称	形式	含义
请求命令	REQUEST	用于查询,符合事先约定要求的标签响应
读写命令	READ-WRITE	阅读器成功识别标签后完成对标签数据的读写
屏蔽命令	SLEEP	用于屏蔽已成功识别的标签,使其不再参与后续识别过程
异或运算命令	XOR	用于辅助判断,以便更精确地确定下一步请求命令

需要说明的是,请求命令 REQUEST 包括 REQUEST(11111111)和 REQUEST(A,B)两种形式。REQUEST(11111111)表示满足 EPC 码小于 11111111 的标签响应;REQUEST(A,B)则代表最高碰撞位第 B 位(B 为二进制数)为 A 的标签响应,当碰撞位连续两位时,A 也为两位,此时最高碰撞位为第 B 位且最高两碰撞位为 A 的标签响应。

2.4 算法流程

NAM算法的整个识别流程如图 1 所示。由 NAM 算法的识别流程图可以看出,算法的一大特

点是引入了 XOR 碰撞位异或运算命令来减少四叉树搜索时不必要空闲时隙的产生。以标签 Tag1: 01010101 和 Tag2:01100101 为例,阅读器接收到碰撞译码 01XX0101 后,确定碰撞位为连续两位并且最高碰撞位为第 5 位,即 REQUEST(A,B)里的 B 用二进制数表示为 101,但此时不直接将 REQUEST(00,101)、REQUEST(01,101)、REQUEST(10,101)和 REQUEST(11,101)压入堆栈,而是先发送一个 XOR 命令使碰撞标签对各自最高两碰撞位进行异或运算,Tag1:01010101 和 Tag2:01100101 两碰撞标签第 5 位和第 4 位的异或运算结果只有“1”,说明最高两碰撞位“XX”只可能为“01”和“10”,从而确定只需 REQUEST(01,101)和 REQUEST(10,101)两个请求入栈,这样的话就可以避免两个空闲时隙的产生。同理,标签碰撞位异或运算结果只有“0”时只需将 REQUEST(00,101)和 REQUEST(11,101)两个请求压入堆栈;当运算结果既有“0”又有“1”时则 REQUEST(00,101)、REQUEST(01,101)、REQUEST(10,101)和 REQUEST(11,101)四个请求均需入栈并依次发送给标签。不难看出,新算法在标签识别过程中,会更精确地确定查询请求,继而选择不同的叉树搜索方式,故新算法称为自调整多叉树防碰撞算法。关于算法性能的数学推导和仿真分析将另文发表。

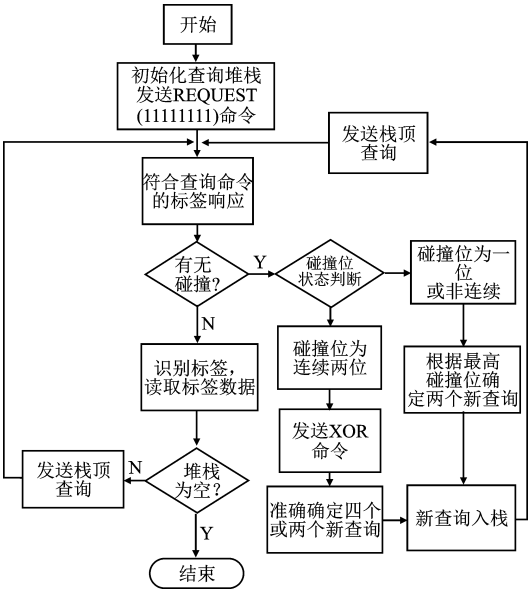


图 1 NAM 算法的识别流程

3 方案总体设计

现场可编程门阵列(Field Program Gate Array, FPGA)实际上是一种半定制的集成电路,它的

出现一方面解决了原有可编程器件门电路数有限的问题,另一方面也克服了定制电路的众多不足. 与传统的通用处理器平台相比,FPGA 不但体积小、造价低、处理速度快,更重要的是 FPGA 逻辑功能的实现全部通过硬件电路实现,用户可根据需要自由地对电路进行设计和定制,更适用于小型终端^[7]. 鉴于此,本文亦选用 FPGA 作为新算法的实现方式,以获取与实际吻合的效果.

RFID 系统的组成包括标签、阅读器和后台数据管理系统三部分,但是 NAM 算法的识别过程主要是通过阅读器和标签之间的多次信息交互来实现的,所以在硬件设计时只需完成对阅读器和标签的设计. 这里,我们将标签看做是 Manchester 编码器模块,将阅读器抽象为 Manchester 解码器模块、算法控制状态机模块和 LIFO 模块的结合体,则 NAM 算法的 FPGA 实现时各模块间的连接关系如图 2 所示.

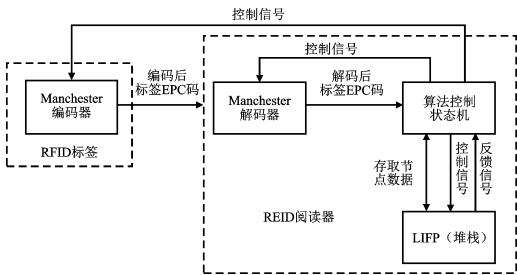


图 2 自调整多叉树防碰撞算法基本模块连接示意图

模块间的工作流程如下:

(1)算法控制状态机每隔一定的时间发送一次 REQUEST 请求命令;

(2)所有阅读器作用范围内的标签接收请求命令并判断自身的 EPC 码是否满足 REQUEST 请求命令,满足的话对阅读器作出响应并向阅读器发送自身 EPC 码,若不满足则继续等待被识别;

(3)阅读器对标签返回的数据进行曼彻斯特编

码,若无碰撞发生则直接将标签 EPC 码压入 LIFO,并利用屏蔽命令 SLEEP 将其屏蔽,不再参与后续识别,若发生碰撞,则根据解码出的数据及碰撞位标志计算下一步的 REQUEST 请求命令参数;

(4)循环上述过程,直到阅读器作用范围内的所有标签被识别.

4 关键模块设计

NAM 算法与一般的基于树型搜索的防碰撞算法最大的不同是引入了存储节点信息的堆栈,使得算法无论是进行二叉树搜索还是四叉树搜索,其搜索路径都将被记录,并在识别出一个标签后返回到离该节点最近的也就是最后被记录下的节点,再沿其他分支继续执行算法. 这样的话,就同时减少了总时隙和不必要空闲时隙的产生,大大提高算法的效率. 与 Manchester 编码器模块、Manchester 解码器模块及其他模块相比,LIFO 模块在整个 NAM 算法中的作用更为凸显,加之 Manchester 编码器、解码器等模块已有较多研究和证明,故本节将着重对 LIFO 模块进行设计,以便后续对其功能进行模拟和验证.

首先,该设计使用 Verilog HDL 语言实现,在 Quartus II 9.1 上综合仿真. 器件型号为 Cyclone II 系列 EP2C35F484C8 器件,它包含 322 引脚和 33 216个逻辑单元,工作电压为 1.2 V,完全能够满足本仿真的需求.

接着,建立 NAM 算法的 LIFO 模块,并对各端口进行定义,如表 2 所示.

另外,LIFO 模块最重要的部分在于要建立良好的读写指针,确保数据读写位置的准确,所以需要一个小型的内部状态机来控制 LIFO 模块的工作状态,按照表 2 的端口设计及定义,LIFO 模块内部状态机的工作状态转换流程如图 3 所示.

表 2 NAM 算法 LIFO 模块建立及端口定义

名称	定义	名称	定义
module lifo	新建堆栈模块	data_push	压栈信号,即高电平有效地数据写入控制信号
clk	系统时钟	data_pop	出栈信号,即高电平有效地数据读取控制信号
clr	清零端	cnt	堆栈指针
din	数据输入端	empty	堆栈为空信号
dout	数据输出端	full	堆栈为满信号

5 功能仿真

5.1 数据读写功能验证

LIFO 模块数据读写功能的验证的主要通过数

据的穿插写入和读取来进行,以验证数据是否满足“后入现出”的过程. 本实验设定的仿真的流程为:首先,连续写入两个数据 01010101 和 01110111;然后,进行一次数据读取;接着,再连续写入两个数据

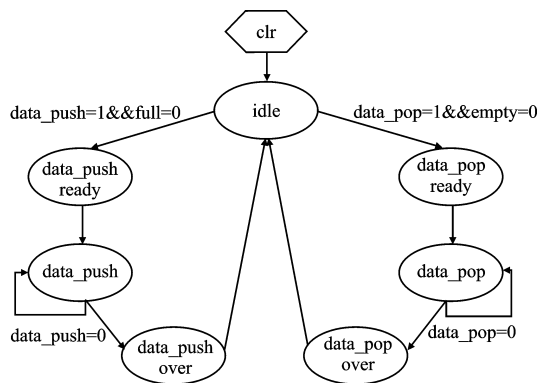


图3 LIFO 内部状态机工作状态转换流程

10101010 和 11101110;最后进行连续的数据读取操作。

得到的仿真结果如图 4 所示,由仿真结果可以看出:第一次写入数据完毕后,栈内只有 01010101 和 01110111,此时进行一次数据读取操作,读出的栈顶数据为 01110111;连续写入数据 10101010 和 11101110 后,栈顶的数据变为 11101110;最后进行连续数据读取时,栈内数据按 11101110、10101010、01010101 的顺序依次被读取,并且在 01010101 数据读取操作结束后,堆栈为空信号 empty 变为高电平,表明堆栈内已没有数据.这说明,LIFO 能够满足数据读写“后入先出”的过程,能够确保 NAM 算法的顺利执行。

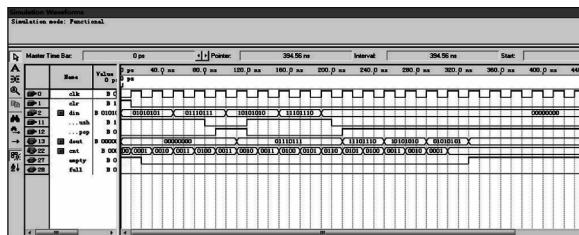


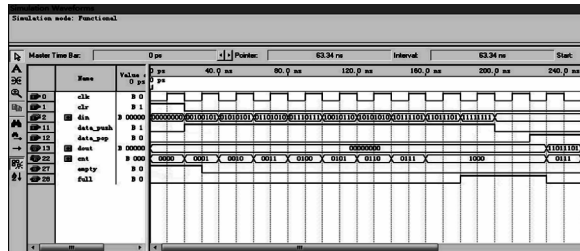
图4 LIFO 模块数据读写功能功能仿真

5.2 溢出实验测试

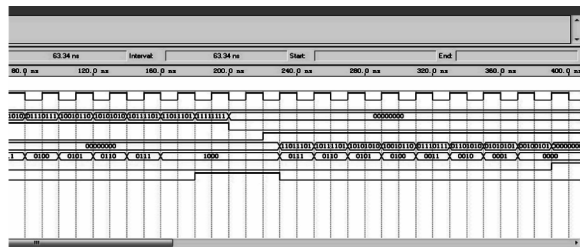
溢出实验是指对 LIFO 进行数据写入,当写入数据量大于 LIFO 的大小时进行数据读取,通过读取栈顶信息来判断溢出情况。

本仿真中,设定 LIFO 的大小为 8,连续向 LIFO 进行 9 次数据写入,得到溢出实验的功能仿真结果如图 5(a)、(b)所示.从仿真结果可以看出:向堆栈连续写入数据,当第八个数据写入后,栈满信号 full 变为高电平,继续写入第九个数据,接着使用高电平有效地 data_pop 出栈信号进行数据读取.读出的第一个数据为第八次写入的数据 11011101,紧接着为第七次写入的数据 10111101,后面依次为第

六次、第五次、第四次,直到最后一个读出第一次写入的数据 00100101,而第九次写入的数据 11111111 并未读出.这说明,在第八次数据写入后,堆栈将不再保存后续写入的数据。



(a)堆栈数据写入仿真



(b)堆栈数据溢出仿真

图5 LIFO 模块溢出实验功能仿真

溢出实验表明,LIFO 的大小直接决定存储节点信息的个数,而节点信息的个数又与算法一次最多处理的标签数量相关,故 LIFO 的大小也决定着算法一次最多能够处理的标签数.目前,低成本 Cyclone II 系列产品最大可提供 3.9 Mbits 的存储容量^[8],由于用于存储节点信息、碰撞位标识、标签数据和系统保留的空间各占存储容量的 1/4,所以可以估算标签长度为 96 bit 时 Cyclone II 系列 FPGA 器件一次性最多可读取的标签数为 $N_{\max} = 3.9 \times 10^6 / (4 \times 96) \approx 10\ 200$,这个数量足以满足绝大多数情况下的应用需求,性能比较优越。

从数据读写功能和溢出试验的功能仿真来看,LIFO 模块功能的正确性得到了验证,达到了预期的要求,能够满足 NAM 算法执行的条件,为算法的可行性提供了有力证明。

6 结束语

对 RFID 标签碰撞问题做了概述,在现有基于树型搜索的防碰撞算法的基础上提出了一种新的自调整多叉树 RFID 防碰撞算法,对算法原理和流程进行了简要说.设计了基于 FPGA 的 NAM 算法的实现方案,并着重对关键模块 LIFO 的功能进行了

(下转第 58 页)

成本以及系统负载等情况

实验结果表明,TCBSA-ACO 算法性能明显优于 BACO 以及 DSFAO 算法,有效避免了系统资源负载过重,资源浪费的情况,降低任务执行的时间和成本,同时能让系统负载始终保持一个均衡的状态.

参考文献:

- [1] 张春艳,刘清林,孟珂. 基于蚁群算法的云计算任务分配[J]. 计算机应用,2012,32(5):1418-1420.
- [2] 查英华,杨静丽. 改进蚁群算法在云计算任务调度中的应用[J]. 计算机工程与设计,2013,34(5):1716-1719.
- [3] 张焕青,张学平. 基于负载均衡蚁群优化算法的云计算任务调度[J]. 微电子学与计算机,2015,32(5):31-40.
- [4] Kaur S, Verma A. An efficient approach to genetic algorithm for task scheduling in cloud computing environment[J]. Information Technology and Computer Science,2012,49(10):74-79.
- [5] Zheng Z N, Wang R. An approach for cloud resource scheduling based on parallel genetic algorithm [C]// Proc. of 3rd International Conference on Computer Re-

search and Development. China, Shanghai, 2011: 444-447.

- [6] Brotoji Mondal, Kousik Dasgupta, Paramartha Dutta. Load balancing in cloud computing using stochastic hill climbing-A soft computing approach[J]. Procedia Technology,2012,28(4):783-789.
- [7] Zhan S B. Huo H Y. Improved PSO-based task scheduling algorithm in cloud computing[J]. Information & Computational Science,2013,13(9):3821-3829.
- [8] Cristian Mateos, Elina Pacini, Carlos Garcia garion. An ACO-inspired algorithm for minimizing weighted flow time in cloud-based parameter sweep experiments [J]. Advances in Engineering Software,2013,56(2):38-50.

作者简介:

聂清彬 男,(1982-),硕士,讲师.研究方向为计算机软件与理论、云计算与物联网. E-mail: 270104318@qq.com.

霍敏霞 女,(1983-),硕士,讲师.研究方向为软件测试.

曹耀钦 男,(1962-),博士,教授,博士生导师.研究方向为计算机网络.

(上接第 52 页)

模拟和验证,仿真结果证明了模块功能的正确性,进而说明了新算法的可行性和性能优越性.

参考文献:

- [1] 任少杰,郝永生,许博浩. 射频识别技术综述[J]. 飞航导弹,2015,44(1):70-74.
- [2] 谢磊,殷亚凤,陈曦,等. RFID 数据管理:算法、协议与性能评测[J]. 计算机学报,2013,36(3):457-470.
- [3] 颜晓莲,陈庆奎,郝聚涛. 基于动态二进制的新防碰撞算法研究[J]. 小型微型计算机系统,2013,34(9):2148-2151.
- [4] 高金辉,郑晓彦. RFID 系统中二进制搜索防碰撞改进算法[J]. 计算机测量与控制,2013,39(2):2754-2756.
- [5] 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):94-102.

[6] 魏峰. 远距离射频识别系统关键技术的研究[D]. 西安:西安电子科技大学. 2009.

[7] 周润景,苏良碧. 基于 Quartus II 的数字系统 Verilog HDL 设计实例详解 [M]. 北京:电子工业出版社,2010.

[8] 王蒙蒙,段本强. 基于按位排序的射频识别防碰撞算法[J]. 计算机工程与应用,2009,45(11):126-128.

作者简介:

任少杰 男,(1991-),硕士研究生.研究方向为信号采集与处理、导弹装备保障及运用. E-mail:RSJ5217@163.com.

郝永生 男,(1966-),副教授,硕士生导师.研究方向为信号采集与处理、导弹装备保障及运用.

许博浩 男,(1989-),硕士研究生.研究方向为信息采集与处理技术.