

学科门类: 工 学 分类号: \_\_\_\_\_单位代码: 10293 密 级: \_\_\_\_\_

南京邮电大学

## 硕 士 学 位 论 文

论文题目: 高速率近距离通信(NFC)  
关键技术研究学 生 姓 名 曹 平学 号 Y040532指 导 教 师 包志华 教 授学 科 专 业 通信与信息系统研 究 方 向 移动通信与无线技术论文提交日期 二〇〇七年六月

## 摘要

近距离通信 (NFC) 技术是最近几年兴起的、从射频识别 (RFID) 技术演化而来的一种新技术。它具有数据量大、保密性高、抗干扰能力强、识别时间短、费用较低等优点, 具有良好的应用前景。特别是它与手机相结合应用的趋势, 使其成为未来一项令人瞩目的新技术。

本文首先介绍近距离通信技术的基本原理、结构、特点及其相关的基础理论知识, 包括信号编码原理、调制解调技术以及数据完整性原理。在分析了近距离通信技术主要标准 EMCA-340 之后, 重点研究了以下几个方面的内容:

(1) 防碰撞原理。分析在近距离通信系统中碰撞产生的原因以及防碰撞算法的分类, 总结多种可引入 NFC 系统使用的防碰撞算法, 提出一种新的防碰撞算法, 经过计算机模拟证明其性能优于 EMCA-340 标准中现有算法。

(2) 高速率 NFC 系统实现关键技术。高速率 NFC 系统关键技术包括编码方式、调制原理、工作方式、防碰撞算法等内容。本文提出了可用在高速率 NFC 系统上的系统模型, 并进行了性能仿真。根据系统速率的不同, 高速率系统模型主要有两种: 一种是采用 EMCA-340 标准调制解调方案的通信速率为 847kbps 的 NFC 系统; 另一个是采用本文提出的调制编码方案的通信速率大于 1Mbps 的 NFC 系统。通过对两种速率的 NFC 系统的仿真, 证明在本文采用的高速率 NFC 系统的关键技术和系统模型是可行的。

本文最后总结了整个课题, 提出值得进一步研究和优化的部分, 并对课题进行了展望。

关键词: 近距离通信、高速率、防碰撞技术、调制与编码、系统性能分析

## ABSTRACT

Near Field Communication is a kind of new technique in recent year. It is evolved from RFID and very hot in theory and application research now. It has many advantages, such as abundance of data information, high security of data, short recognizing time, low cost and has a promising future, especially combine with mobile phone.

This paper first discussed the basic theory that used in NFC system, including the principle of single coding, modulation and data integrity. Then it focused on the following areas base on analysing the key technologies of NFC system and EMCA-340 protocol:

(1) Anti-collision algorithms. The paper discussed how to bring collision in system and sorts of collision, summed up several algorithms that can use in NFC system, proposed a new anti-collision algorithm and proved its validation by simulation.

(2) The key technologies that would make the achievement of the high bit rate NFC system. Base on discussing the key technologies, the paper introduced a high bit rate communication system model that is suitable for high bit rate NFC system. For the system, Two main bit rates system model were introduced: One is used in EMCA-340 when bit rate is 847kbps, another is introduced in the paper when bit rate is greater than 1Mbps. The simulation results proved the validity of model and key technologies that introduce in the paper.

**Key Words:** Near Field Communication, high bit rate, anti-collision algorithms, modulation and coding, system performance analysis

## 南京邮电大学学位论文独创性声明

本文声明所呈交的学位论文是我个人在导师指导下进行的研究工作及取得的研究成果。尽我所知，除了文中特别加以标注和致谢的地方外，论文中不包含其他人已发表或撰写过的研究成果，也不包括为获得南京邮电大学或其它教育机构的学位或证书而使用过的材料。与我一同工作的同志对本研究所做的任何贡献均已在论文中作了明确的说明并表示了谢意。

研究生签名： 曹平 日期： 2006.6.5

## 南京邮电大学学位论文使用授权声明

南京邮电大学、中国科学技术信息研究所、国家图书馆有权保留本人所送交学位论文的复印件和电子文档，可以采用影印、缩印或其他复制手段保存论文。本人电子文档的内容和纸质论文的内容相一致。除在保密期内的保密论文外，允许论文被查阅和借阅，可以公布（包括刊登）论文的全部或部分内 容。论文的公布（包括刊登）授权南京邮电大学研究生部办理。

研究生签名： 曹平 导师签名： 包志华 日期： 2007.6.13

## 第一章 绪论

### 1.1 研究背景

随着网络及通信技术的飞速发展，人们对无线通信的要求越来越高，近距离无线技术正在逐步成为关注的焦点。目前使用较广泛的近距无线通信技术有蓝牙（Bluetooth）、无线局域网802.11（Wi-Fi）和红外数据传输（IrDA），同时还有一些具有发展潜力的近距无线技术标准，它们分别是：ZigBee、超宽带（UWB）、近距离通信（NFC）、WiMedia、DECT、无线1394和专用无线系统等。它们都有其各自的特点。或基于传输速度、距离、耗电量等特殊要求；或着眼于功能的扩充性；或符合某些单一应用的特别要求；或建立竞争技术的差异化等。而近距离通信技术在众多近距离通信技术中，以其简单性、安全性、高速率、低功耗、低成本的优点成为近距离无线通信领域未来最有竞争力的技术之一。

NFC技术由无线设备间的非接触式射频识别及互连技术发展而来，它可以满足任何两个无线设备间的信息交换、内容访问、服务交换，并且使之更为简约——只要任意两个设备靠近而不需要线缆接插，就可以实现相互间的通信。这将任意两个无线设备间的通信距离大大缩短。除了以上功能之外，NFC技术还可以用于快速建立各种设备之间其他类型的无线通信，可作为一种虚拟连接器。通过将两个设备，NFC可以设置和初始化其他无线协议，例如蓝牙和802.11（Wi-Fi），使设备能在更远的距离或以更高的速率传输数据。NFC比红外更快、更可靠而且简单得多。与蓝牙相比，NFC通信距离更短，安全性更好，适用于重要数据的传输。蓝牙能够弥补NFC通信距离不足的缺点，适用于较长距离数据通信。因此，NFC和蓝牙互为补充，共同存在。

虽然NFC技术是近期才出现的（由飞利浦和索尼于2002年联合研发），但是在近年来却取得了长足的发展。2004年由诺基亚、飞利浦和索尼共同推出了旨在推动近距离通信技术发展的非赢利性的行业协会——NFC论坛，成立2年以来，相继有20多家公司和组织加入该论坛。

NFC技术之所以成为无线通信领域关注的热点之一，是由其技术性的特点和应用前景共同决定的。NFC正以其在无线支付中的影响引起无线世界的关注，并持续在许多市场势头不减。NFC从RFID与互联技术的结合演变而来，是一种在小范围内为移动电话、PC机以及电子设备设计的直接、简单而且安全的通信无线连接技术标准。由于NFC技术的这些特点和诺基亚、NXP的共同推进及来自移动运营商和交易公司不断增长的兴趣，在一些移动支付领域，NFC的商业部署已经取得了一些成绩，除了多款包含NFC功能的手机面世以外，

全球NFC技术商用的步伐也在加快之中。2006年4月19日，飞利浦、诺基亚、沃达丰以及德国法兰克福、美因茨地区的公交网络运营商美因茨交通公司宣布：在成功地进行为期10个月的现场试验后，NFC技术即将投入商用。这是全球首个NFC商用案例。NFC在中国的应用也在迅速展开之中。2006年6月27日，100名厦门移动招募的志愿者参与现场试验，率先使用具备NFC功能的手机，他们在厦门市易通卡覆盖的公交汽车、渡轮、餐厅、电影院、便利店等营业网点，只要轻轻刷一下手中的手机，就可以实现消费交易。调研公司ABI预测，到2009年，50%的新增手机将具备NFC功能，2010年NFC手机发货量将达到5亿台。在新兴移动应用中，手机支付是最被用户看好的项目之一。

1.2 NFC 的基本概念

近距离通信（近场通信）技术是一种新的、短距离的无线连接技术。它是由非接触识别（RFID）与互连技术演化发展而来。在单一芯片上结合了感应式读卡器，感应式卡片和点对点的功能。

NFC 技术的工作频率为 13.56MHz，工作距离在 10cm 之内。NFC 技术符合国际标准化组织的 ISO 18092 和 ISO 21481 标准，欧洲计算机制造业协会的 EMCA-340、352 和 356 标准，欧洲电信标准协会的 ETSI TS 102、190 标准。同时还与已经制定的关于无线智能卡的 ISO 14443A 标准兼容。表 1-1 列出了与 NFC 有关的 EMCA 和 ISO 标准<sup>[1]</sup>。

表1-1 NFC的EMCA和ISO标准说明

EMCA	ISO	Content
EMCA-340	ISO18092	NFC interface and protocol
EMCA-352	ISO21481	Select mode
EMCA-356	ISO22536	Test for RF interface
EMCA-362	ISO23917	Test for protocol

在 13.56MHz 的工作频率下，NFC 的传输速率可分为 106kbps/212kbps/424kbps 三种，将来最高可以到达 1Mbps 左右，一般调制采用 ASK 调制，视不同速率有不同的调制系数。编码方式主要有两种：改进型 Miller 编码、Manchester 编码。通信模式可分为主动模式与被动模式，主动模式是指主动通信方（Initiator）与目标通信方（Target）皆可用自身电源供应产生射频信号，而被动模式下则是主动通信方自身供应电源产生射频信号，而目标通信方则利用全波整流线路将主动通信方的射频信号的能量转换为电压来供应自己的电源。值得一提的是，在被动模式下为了满足省电的要求采用了负载调制（Load modulation）

的方式，此调制方式可以达到省电的效果<sup>[2]</sup>。

使用上，因为 NFC 的使用通常会遇到使用尖峰时期，为了避免不同的主动通信方或目标通信方同时通信造成数据链路错误，所以 NFC 采用了一种 Listen before talk 的机制。此机制会在主动通信方要发出询问信号前，先侦测外界磁场强度来判断是否有其它的设备正在通信中，这种机制的实现称为 RF Collision Avoidance (RFCA)，其动作行为是在每次主动发出询问信号时侦测外界磁场，当磁场强度超过门限强度时则会停止询问，直至外界磁场强度低于门限值<sup>[3]</sup>。

在开启自身的射频信号后，对进入射频信号覆盖范围内目标通信方进行多目标识别，选出唯一目标进行点对点通信。被动通信模式下主要采用多目标识别算法有二进制搜索树和时隙算法。前者主要针对通信速率为106kbps，后者主要应用于通信速率为212、424kbps。主动通信模式与被动通信模式采用不同的识别算法。数据的传输在多目标识别之后进行，其传输协议结构与TCP/IP协议类似，但相比TCP/IP协议要简单很多，只包含了数据链路层和传输控制层的协议标准<sup>[4][5]</sup>。

### 1.3 NFC技术的应用

NFC技术就目前情况看主要应用于手机支付上。同时NFC技术还能实现智能门控、电子票务等功能，符合现代消费者的需求，也能为移动化服务创造更多商机。因为它基于国际标准，所以可在世界范围提供各种服务，应用前景广泛。

NFC 技术的应用可以分为以下四种类型<sup>[6][7]</sup>：

#### 1) 用于付款和购票

第一，NFC手机可作为乘车凭证，通过接触进行支付。目前中国一些大城市推行的一卡通，通过使用载有信息的智能卡片与特定设备接触从而实现乘车票的支付。NFC手机可以将车票内容保存在芯片，从而实现瞬间购票除了电子车票的应用之外，NFC手机的使用范围还可以扩大到观光地点的入场券、住宿券等

第二，NFC手机可当作电子钱包，通过接触和密码确认进行支付。现在一些便利店都只有现金收款机设备。允许使用内置NFC功能的手机来付款将提高便利性，并且手机购买积分或者手机实物抽奖的形式还能刺激用户消费。

#### 2) 用于电子票证

作为电子入场券和钥匙的NFC手机可通过接触完成认证使用者只要通过手机上网下载电子票券，带着手机就可以入场。NFC手机当作公寓钥匙时，只要将手机贴近门，就可以

开锁,还可在大楼内设置一台多媒体终端,方便用户直接利用手机交付房租及水电费。

### 3) 用于智能媒体

智能媒体将成为手机下载的上端。例如,使用内嵌NFC卡的手机,只需要在智能媒体表面晃动即可下载票务信息、铃声和壁纸等;用户在智能海报旁闪动一下手机,就能从海报的智能芯片中下载关于该活动的信息

### 4) 用于交换、传输数据

将两台支持NFC的设备,即可进行点对点网络数据传输,例如下载音乐、交换图像或同步处理通信录等。现今大多数手机都配备了蓝牙等相关功能,所以NFC可以充当启动设备,使电话之间的数据交换传输更加便捷。NFC还支持多台手机间的多人游戏,允许用户与环境进行交互式通信,而无需浏览复杂的菜单或执行复杂的设置程序。

NFC技术不仅会淘汰传统信用卡,而且还将取代传统的钥匙、员工身份识别卡、音乐会门票等。银行的发卡方式也将改变,目前,万事达卡已经将信用卡数据通过移动网络传输装载于NFC手机中,从而省下传统的发卡成本。

## 1.4 NFC国内外发展情况

### 1) 国外<sup>[8][9]</sup>

NFC技术最早在2002年由荷兰飞利浦公司和日本索尼公司联合推出,所以国外在NFC技术的标准制定和产品的开发一直处于领先状态。由于巨大的市场潜力和良好的应用前景,越来越多的公司加入到NFC标准制定和产品的开发中。2004年,由诺基亚、飞利浦和索尼共同推出了旨在推动近距离通信(NFC)技术发展的非赢利性的行业协会——NFC论坛。成立2年以来,相继已有20多家公司和组织加入该论坛,该论坛致力于开发先进的NFC技术,向公众宣传其优势,并进一步深化其在全球范围内的部署。

作为NFC技术的积极推动者,2004年11月,诺基亚在摩纳哥举行的诺基亚移动会议上推出移动通信创新产品开发中的最新进展——诺基亚NFC机壳。在电话上装配诺基亚NFC机壳后,消费者将能够轻松获取各种各样的服务,并且只需简单地轻动一下手指便可利用NFC技术交流信息。其后,在2006年4月诺基亚推出了世界上第一款NFC手机—3220,并且在德国的Hanau与沃达丰推出第一个NFC支持的商用移动无线支付服务。

2005年10月,在法国的Caen,飞利浦、法国电信、Orange、三星、零售集团LaSer、Vinci Park共同启动了一项多应用NFC试验,试验期为6个月,通过该试验,200个居民可使用三星D500手机作为安全支付工具在特约零售店和停车场进行支付,并可下载旅游与公交服务信息。2005年11月,德国哈瑙市公共交通运营商Rhein-Main Verkehrsverbund (RMV)



正式宣布推出全球第一个NFC实验运行计划。这是一个为期6个月的区域性公车网络票务系统应用,合作伙伴包括飞利浦、诺基亚以及RMV厂商代表。2005年11月,北美最大的NFC项目在亚特兰大的飞利浦体育馆进行,观众可使用手机在场馆内购物,还可访问并下载铃声、墙纸、屏保、视频片段等移动内容。2005年底,美国最大的移动运营商Cingular无线开始了NFC手机付费功能的测试。2006年6月,飞利浦和韩国SKT公司宣布将在NFC技术的开发和部署上展开合作,在韩国首都汉城进行大规模试验。根据协议条款,两家公司将与当地合作伙伴从2006年6月份起展开一场为期6个月的大规模NFC实地测试。该试验将在SKT汉城办公大楼内进行,400名SKT员工和访问者可通过简单地晃动嵌有NFC的移动电话来进入各种各样的NFC应用设备。两家公司旨在简化NFC应用在全球范围内的进程与部署,包括携手合作作为网络运营商制定完整的NFC解决方案,以推动NFC在全球的商业化进程并推出更多基于非接触式科技的应用。其解决方案包含技术、完整的硬件和软件产品、手持设备和读卡系统及支持NFC的应用程序。

2006年4月19日,飞利浦、诺基亚、沃达丰及德国法兰克福美因茨地区的公交网络运营商美因茨交通公司宣布,在成功地进行为期10个月的现场试验后,NFC技术即将投入商用。这是全球首个NFC商用案例,它揭开了NFC手机支付商用的序幕。

## 2) 国内<sup>[10]</sup>

相比与国际上,NFC技术及其应用在我国的发展相对滞后。国内各种文献开始出现NFC技术的介绍以及对其标准的讨论最早是在2004年前后,而对于加入NFC论坛,中国公司似乎也不是积极性不高。就目前情况来看,在国内NFC的应用中,国外公司的产品与技术处于支配地位。

2006年6月,诺基亚、厦门移动、厦门易通卡公司、飞利浦公司共同在厦门启动中国首个NFC手机支付现场试验。现场使用具备NFC功能的诺基亚3220,实现了厦门市任何一个厦门易通卡覆盖的公交汽车、轮渡、餐厅、电影院、便利店等营业网点的手机支付,中国联通也正在积极研究包括NFC在内的各种非接触支付技术,并计划出台移动支付标准。2006年8月,诺基亚与银联商务联袂宣布在上海第一八佰伴百货公司启动NFC手机积分卡支付现场试验,中国银联的集利卡Union Bonus(具有标志的卡片,又称银联积分通卡)将集成至NFC手机中。

目前,国内手机用户数接近4亿,同时银行卡发行总量超过7.6亿张,而预计到2008年中国的手机用户将达到5亿。数目巨大的手机消费群体和银行卡持有者数量,为将手机和银行卡绑定来进行各种消费的移动商务、移动支付提供了庞大的用户基础。由此可以预计,

NFC技术在我国的应用前景十分光明。

## 1.5 本文所做的工作

本文从 NFC 技术的主要标准 EMCA-340、352、356、362 出发, 首先讨论 NFC 技术的基本原理, 包括数据传输过程中的编码与调制技术以及数据完整性原理。在此基础上, 重点放在 NFC 系统在高速率通信中关键技术的研究上, 提出了适用于高速率 NFC 系统的模型, 最后对系统模型的性能进行仿真分析。本文的具体安排如下:

1. 第一章介绍本文的研究背景, 简要介绍 NFC 技术及其应用场合, 国内外发展现状, 并指出了本文研究的重点。
2. 第二章主要内容是 NFC 技术基础, 讨论 EMCA-340 标准中近距离通信系统信号的编码原理与调制解调技术。
3. 第三章讨论近距离通信系统数据的安全性和完整性原理。重点研究了通信过程中防碰撞算法, 引入了若干种可用在 NFC 系统的防碰撞改进算法。最后, 提出一种二进制防碰撞算法, 并用软件进行性能模拟分析。
4. 第四章是本文的重点, 主要研究实现高速率 NFC 系统所涉及的关键技术, 主要包括高速率通信时的工作模式、调制和编码方式。在研究各项关键技术的基础上提出了高速率 NFC 系统的系统模型。最后简要讨论了可在高速率通信时采用的防碰撞算法。
5. 第五章在前一章讨论的基础上, 主要是对本文提出的高速率 NFC 系统模型性能进行仿真和验证, 系统仿真模型主要有两种, 它们的通信速率分别为 847kbps 和大于 1Mbps。
6. 最后给出本文研究的结论与下一步研究工作的展望。

第二章 近距离通信系统中的编码与调制技术

本章介绍近距离通信系统中使用的编码与调制技术，主要包括 EMCA-340 标准中信号基带编码和调制解调技术。

2.1 基带编码

近距离通信系统中，基带数据编码可以采用不同形式的编码来表示1或0。常用的近距离通信系统中的编码方式<sup>[11]</sup>有NRZ编码、Manchester编码、单极归零编码、差动双相编码、Miller编码、改进型Miller编码等，码元波形如图2-1所示。

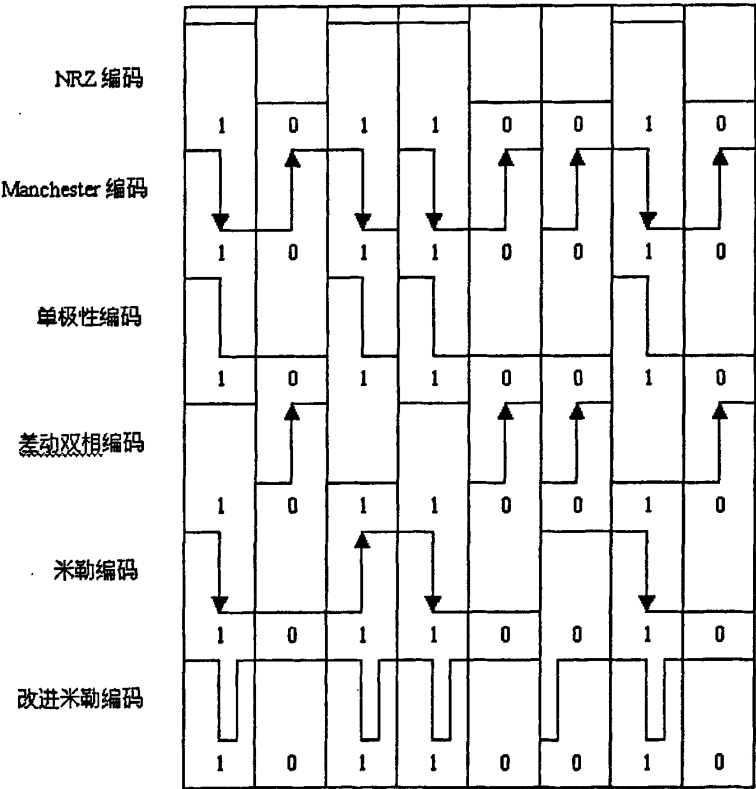


图2-1 常用信号编码

NRZ编码：在这种编码中“高”信号表示二进制1，“低”信号表示二进制0，在整个码元区间电平保持不变。

曼彻斯特编码：在半个比特周期中的负边沿便是二进制1，半个比特周期中的正边沿表示二进制0。由于曼彻斯特编码在每个码元间隔的中心部分都存在电平跳变，因此在频谱中存在很强的定时分量。此外，由于方波周期内正、负电平各占一半，因此也不存在直流分量。

单极归零编码：在前半个比特周期中为“高”，后半半个比特周期中为“低”的信号表示1，而持续整个比特周期的“低”信号表示二进制的0。

差动双向编码：在半比特周期中任意的边沿跳变表示二进制0，而没有边沿持续整个比特周期表示二进制1，在每一个比特周期开始时，电平要反相。

米勒编码：在半比特周期内的任意边沿表示二进制1，而经过下一个比特周期中的1电平表示二进制0。一连串的0在比特周期开始时产生电平交换。

变形米勒编码：也就是在米勒编码的基础上有一些细微的变化，在每个边沿都有一个“负”脉冲所取代。

在选择信号编码时，应当考虑每种编码的边界条件，其中最重要的是调制后的信号频谱，以及对传输故障的敏感度。在近距离通信系统的被动通信模式下，不允许因信号编码与调制方法的不适当的组合而导致能量转化的中断；在主动通信模式下，由于不是一个无源系统，可以考虑采用多种编码方式。近距离通信系统协议标准EMCA-340中，前者所用的编码方式是曼切斯特编码和改进型Miller码；后者采用改进型Miller码。

曼切斯特编码的特点是每个码元中间都有跳变，使得这种码低频能量较少，而且这种码型的特点也便于接收端提取时钟信息。

改进型米勒码的特点是一连串的零在比特周期开始时产生电平交变，因此对接收端来说，位节拍比较容易重建。很短的脉冲持续时间，就可以在数据传输过程中保证从通信发起方的高频场中连续供给通信目标方能量，而且短的负脉冲宽度使传送信号能量集中在窄的带宽内，更利于通信目标方对信号的接收。另外，该编码方式因为没有相位反转问题，所以非常适合调制。

## 2.2 数字调制技术

调制就是用基带信号对载波波形的某些参量进行控制，这些参量随基带信号的变化而变化。调制的载波可以分为两大类：用正弦信号作为载波和用脉冲作为载波。通过调制不仅可以进行频谱搬移，把调制信号的频谱搬移到所希望的位置上，将调制信号转换成适合信道传输或便于信道多路复用的已调信号，而且调制对系统的传输有效性和传输的可靠性有很大的影响，调制的方式往往决定了一个通信系统的性能。

数字信号一般调制方法有三种：对载波振幅调制称为振幅键控（ASK-Amplitude Shift Keying），对载波频率调制称为频移键控（FSK-Frequency Shift Keying），对载波相位调制称为相移键控（PSK-Phase Shift Keying）。

在近距离通信系统中采用ASK调制方式<sup>[12]</sup>，视速率不同而有不同的调制系数。在通信

速率为106kbps时调制系数为100%，速率为212kbps、424kbps时，调制系数为8%-30%。在振幅键控时，载波振荡的振幅按二进制编码信号在两种状态 $u_0$ 和 $u_1$ 之间切换（键控）。 $u_1$ 可以取 $u_0$ 和0之间的值。为了描述不同程度的键控，定义键控度 $m$ ：

$$m = 1 - \frac{u_1}{u_0} \quad (2.1)$$

当 $u_1$ 等于 $u_0$ 时， $m=0$ ，所以没有调制；当 $u_1$ 等于零时， $m=1$ ，所以称为100%振幅调制（或是开关键控）。对于ASK调制的信号，可以描述为：

$$v(t) = u_0 [1 + m(s(t) - 1)] \cos(\omega_c t) \quad (2.2)$$

其中 $s(t)$ 是二进制信号1或0， $\omega_c$ 是载波频率。基于编码信号 $s(t)$ 的调制信号 $g(t)$ 可表述为：

$$g(t) = u_0 [1 + m(s(t) - 1)] = mu_0 s(t) + u_0(1 - m) \quad (2.3)$$

$g(t)$ 的频谱为：

$$G(f) = mu_0 S(f) + u_0(1 - m)\delta(f) \quad (2.4)$$

最后可得到，调制信号的能谱密度为：

$$PSD_v = \frac{1}{4} [(mu_0)^2 PSD_s(f - f_c) + [u_0(1 - m)]^2 \delta(f - f_c)] \quad f \gg 0 \quad (2.5)$$

可以注意到，调制信号的带宽完全由基带信号的能谱密度所决定，但它的幅度由载波幅度和键控度共同决定。当键控度为1时，调制信号的能谱密度就是基带信号的能谱密度乘于载波最大幅度的平方的四分之一倍。降低键控度就会降低边带的幅度，且会引入和增加载波频率中delta函数的权重，这样降低了高低电平之间的差异，并且使误码率增加。

在NFC系统中，还采用一种特殊的调制方式：负载调制<sup>[11]</sup>。

近距离通信系统的射频接口是一个电感耦合系统，对于这个系统来说，它是一种变压器耦合型。这就是说作为初级线圈的主动通信方和作为次级线圈的目标通信方之间的耦合。只要线圈的距离不大于 $0.16\lambda$ （波长），在NFC系统中，波长为22米，也就是说线圈距离不大于3.52米目标通信方处于天线的近场之内，变压器耦合就是有效的。

如果把谐振的目标通信方（就是说，目标通信方的固有谐振频率与主动通信方的发送频率相符合）放入主动通信方天线的交变磁场中，那么该目标通信方就能从磁场取得能量。从供应主动通信方天线的电流在主动通信方内阻上的降压可以测得此附加功耗。目标通信方天线上的负载电阻的接同和断开促使主动通信方天线上的电压发生变化，实现远距离目标通信方对天线电压进行振幅调制。如果通过数据控制负载电压的接通和断开，那么数据

就能实现在NFC设备之间的传输。这种传输方式称为负载调制。在近距离通信系统中采用了使用副载波的负载调制，由于NFC设备之间的耦合很弱，因此检测很小电压变化的开销巨大，所以在实际系统中利用天线电压振幅调制所产生的调制波边带。

如果目标通信方的附加负载电阻以很高的始终频率  $f_H$  接通或断开，那么在主动通信方发送频率  $\pm f_H$  的距离上产生两条谱线，它们容易检测到的（然而，必须是  $f_H < f_{Initiator}$ ）。这种新的基本频率称为副载波。数据传输是即使在数据流中通过振幅键控（ASK）、频移键控（FSK）或相移键控（PSK）调制来完成的，这就是副载波调制。

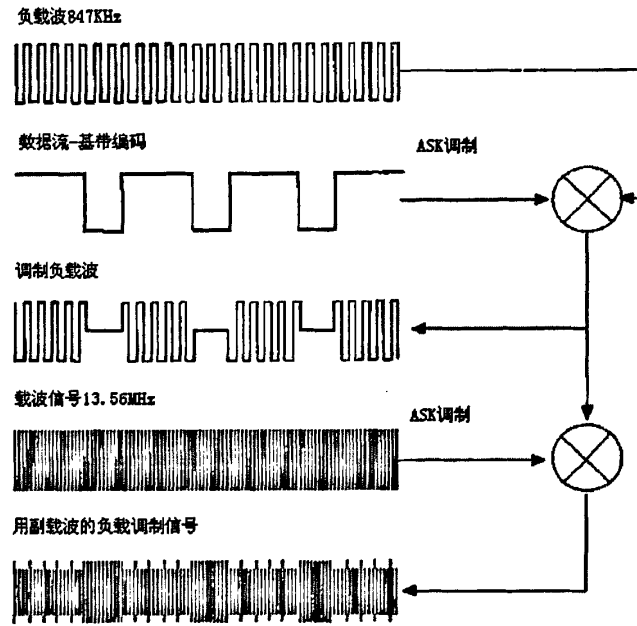


图2-2 用负载调制的振幅键控的副载波调制

在近距离通信系统中采用的副载波频率为847KHz，副载波的调制方式为ASK，其调制过程如图2-2所示。

在观察产生的频谱时，才能理解使用副载波带来的好处，副载波进行负载调制时，首先在围绕操作频率  $\pm$  副载波频率  $f_H$  的距离上产生两条谱线。真实的信息随着基带编码的数据流对副载波的调制被传输到两条副载波谱线的边带中。另一方面，如果采用的是在基带中进行的负载调制时，数据流的边带将直接围绕着工作频率的载波信号。

对于很松散耦合的近距离通信系统来说，在主动通信方的载波信号  $f_T$  与接收的负载调制的调制边带之间的差别在80-90dB的范围内波动。通过数据流的调制边带的频移，可以

将两个副载调制产物中之一滤出并解调。至于是使用  $f_T + f_H$  还是  $f_T - f_H$  都是无所谓的，因为在所有的边带中都包含了信息。调制结果如图2-3所示。

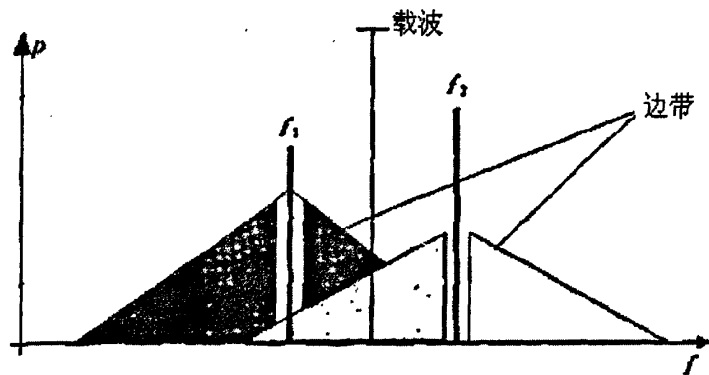


图2-3 用副载波的负载调制时的调制结果

2.3 近距离通信系统的调制编码方式分析

在近距离通信技术标准EMCA-340中，主要的通信模式为主动通信模式和被动通信模式。被动通信模式类似于射频识别系统的数据传输过程，主动通信模式下，通信双方各自产生射频信号，处于同等地位，不存在主从关系。下面列出了在EMCA-340中各种模式和通信速率下的编码与调制方式。

2.3.1 主动通信方（Initiator）到目标通信方（Target）调制编码方式

表2-1 Initiator到Target调制编码

通信模式		主动通信模式	被动通信模式
传输速率			
106kbps	调制	ASK 调制系数100%	ASK 调制系数100%
	编码	改进型Miller编码	改进型Miller编码
212kbps	调制	ASK 调制系数8%—30%	ASK 调制系数8%—30%
424kbps	编码	Manchester编码	Manchester编码

2.3.2 目标通信方（Target）到主动通信方（Initiator）调制编码方式

表2-2 Target到Initiator调制编码

通信模式		主动通信模式	被动通信模式
传输速率	106kbps	调制	ASK 调制系数100% 采用振幅键控调制847kHz 的负载调制的负载波
		编码	改进型Miller编码 Manchester编码
	212kbps 424kbps	调制	ASK 调制系数8%—30% 采用振幅键控调制847kHz 的负载调制的负载波
		编码	Manchester编码 Manchester编码

2.3.3 调制编码技术标准分析与比较

NFC脱胎于RFID技术中载波为13.56MHz的ISO14443技术标准。在NFC技术标准EMCA-340中，将载波频率为13.56MHz的RFID的ISO14443技术标准中的调制编码方式进行了改进和优化组合。

载波为13.56MHz的RFID的技术标准ISO14443中，主要的调制编码技术分为两种类型<sup>[13]-[15]</sup>：TYPE A、TYPE B。NFC技术同时采用了两种类型的技术，并对其中的某些技术细节进行了改进，在速率为106kbps时完全采用了ISO14443的TYPE A标准；速率为212、424kbps时采用了ISO14443的TYPE B的改进标准。下面对NFC技术中的这两种相差较大的调制编码技术进行分析比较。

1) 能量传输

依据EMCA-340定义的射频信号接口标准，对于速率为106kbps，天线接收到的是100% ASK调制的改进Miller编码信号，采用此种信号的优点是100%ASK调制能以100%的能量进行数据传输，保证了信号的较高抗干扰性，在一定程度上提高了通信的可靠性。理论分析表明它在调制间隙(Pause)处信号电压不足1V，不能保证卡上数字部分的正常工作，即在数据传输期间，数字处理部分不能正常工作，所以在数字处理部分工作时停止数据传输。因此，尽管100%ASK调制以100%的能量进行数据传输，但它是 以数据传输与数据处理分步工作，即以通信时间的延长为代价的。同时，它也不适用于常规的数字信号处理器，除非在外加时钟的情况下可以采用常规的DSP。

对于212、424kbps的通信速率，由于它采用8-30%的ASK调制，仅用部分能量传输数据，同时NFC设备上的数字信号处理器并未用到所供给的全部能量。当受到噪声干扰时，



对信号的可靠性会有一些影响, 缩短其有效读写距离, 但它能保证能量的无中断供给, 可以实现数据传输与处理的同步进行, 保障芯片在工作时永远不会失去电压供应以及时钟, 使内部逻辑与软件连续正常地工作, 而不会在数据接收过程中因为电源消失而必须暂停工作, 应用软件在工作时不必担心和处理电源消失或数据中断等问题。这对于高安全性芯片来说是一个显著的优点, 因为其安全算法需要不间断地运行。调制系数8-30%的ASK调制的另一优点是在完成NFC设备的选择之后, 两个NFC设备之间可以有更高的通信速率。因此, 这种通信方式更适合于未来的各种应用开发。

以上能量传输问题主要针对被动通信模式下。在NFC技术中除了兼容ISO14443标准, 还采用了ISO14443没有的主动通信模式, 在此种通信模式下并无能量中断的问题, 其通信过程中, 数据交换双方始终处于有源状态, 射频信号交替关闭发射。

## 2) 编码方式与负载调制方式

目前在被动通信模式下, NFC设备之间的通信在三种不同的速率下都采用了负载调制方式。106kbps速率的通信方式使用847.5kHz副载波Manchester编码。而212、424kbps速率的通信方式的负载调制也使用847.5kHz的副载波, 两台NFC设备之间的数据传输是连续进行的。副载波采用振幅键控的调制方式(ASK)。106kbps采用改进密勒(Modified Miller)编码和曼彻斯特(Manchester)编码, 因而速度很快, 必须采用专门的硬件解码。

## 3) NFC系统射频前端

NFC系统一般会在RF前端电路中引入了分频电路来产生副载波(副载波频率为847.5KHz, 为载波频率13.56MHz的1/16, 数字部分的时钟由此副载波再经分频产生), 数字部分产生的数据信号先以ASK方式对此副载波进行调制, 调制后的信号仍然以ASK的方式叠加在高频载波上经天线发出, 这样就以相对简单的电路形式实现了以较低误码率进行传输的目的。在NFC设备中, 无论是载波还是副载波均由主动通信方产生, 因此在主动通信方进行解调时不用考虑相干信号的产生问题(若不考虑两个NFC设备之间距离引起的相位变化, 则主动通信方本身的发射载波信号与被动通信方的发射信号同频同相, 当然, 精确的解调仍需引入同步系统)。对于主动通信方信号的调制方式, 主要应考虑解调的难易, ASK解调最为简单, 具体操作时, 可适当调整NFC设备电路, 保证输出已调信号的幅度和质量, 降低NFC设备接收信号解调后的误码率。

## 2.4 本章小结

本章讨论了EMCA-340标准中的调制与编码原理, 并对标准中各种方案的优劣和特点进行分析比较, 为后面各章内容的深入研究打下了理论基础。

## 第三章 数据校验与防碰撞算法

近距离通信系统中数据传输的完整性,存在两个方面的问题,外界的各种干扰和多个目标同时占用信道时导致发送数据产生碰撞。运用数据检验也就是差错检测和纠正、防碰撞算法分别解决这两个问题。在简要介绍近距离通信系统数据校验原理之后,本章主要研究防碰撞算法。在总结了几种可用在NFC技术中的防碰撞算法之后,结合每个NFC设备具有唯一序列号的特点,提出了一种二进制防碰撞改进算法。

### 3.1 差错检验和纠正

数字信号在NFC系统中传输时,由于系统特性不理想和信道中有噪声干扰,从而引起数字信号波形的失真,在接收判断时可能误判而造成误码,导致传输错误。针对这种情况有两种基本的策略<sup>[11]</sup>来处理:一种方法是在每一个要发送的数据块上附加足够的冗余位信息,使接收方能够推导出已发出的字符应该是什么,这种策略是使用纠错码(error-correcting code);另外一种只是假设有足够的冗余位,使接收方知道有差错发生,但不知道是什么样的差错,然后让接收方请求重传,这种策略使用的是检错码(error-detecting code)。

#### 3.1.1 纠错码

这里我们先分析一下错码,通常一数据块包括, $m$ 个数据位和 $r$ 个冗余位或检验位,整个长度为 $n$ 位码字(即 $n=m+r$ )。两个码字中不同的位的个数,称作海明距离(Hamming Distance),假如两个码字具有海明距离 $d$ ,则需要 $d$ 个位差错才能将其中一个码字转换成为另一个。一个编码的校验和纠错能力取决于它的海明距离。为检测出 $d$ 比特错,则需要使用距离为 $d+1$ 的编码;因为这种编码 $d$ 个单比特错决不可能将一个有效的码字改变成为另一个有效的码字,当接收方看到无效的码字时,它就能明白发生了传输差错。

检错码最简单的是奇偶检验,在数据后面加上一个奇偶位(parity bit)的编码,奇偶位的选取原则是使码字内的1的数目是奇数(或偶数)。奇偶校验位的值是这样设定的:奇校验时,若字节的数据位中的个数为奇数时,则奇偶校验位的值为0,反之则为1;偶校验时,则当每个字节的数据位中1的个数为奇数时,则奇偶校验位的值为1,反之则为0。例如,当10110101通过末尾加一位,以偶数位传送时,就变成为了101101011;以奇数位传送时,就变成为了101101010。这种代码的海明距离为2,缺点是识别错误的能力低,只能检测单比特错。

#### 3.1.2 检错码

检错码有时也用于数据传输。例如,当信道为单工方式,无法要求重传的情况下,大

多数采用检错码加重传的方式,因为这样效率才更高,现在广泛应用的是多项式编码(也叫做循环冗余码,或CRC码)。多项式编码是基于将位串看成系数为0或1的多项式,一个 $k$ 位帧可以看成是从 $x^{k-1}$ 到 $x^0$ 的 $k$ 次多项式的系数系列,这个多项式的阶数为 $k-1$ 。从数学角度来看,多项式校验是用所谓的生成多项式去除一个多项式(数据字节)即可求得,CRC值为相除后所得余项,整个相除的过程采用异或计算。

如果采用CRC编码校验,发送方和接收方必须事先约定一个生成多项式 $G(x)$ ,生成多项式的高位和低位必须是1。要计算 $m$ 位的帧 $M(x)$ 的校验和,生成多项式必须比该多项式短。基本思想是:将校验和加在帧的末尾,使这个带校验和的帧的多项式能被 $G(x)$ 除尽;当接收方收到带有校验和的帧时,用 $G(x)$ 去除它,如果有余数,则传输出错。

当数据帧被传输时,在发送方计算数据的CRC值,并将它附在数据帧后一起传输,在接收方计算接收数据帧(包括附加的CRC字节)的CRC值,结果为0表示传输正确,否则说明接收的数据帧错误。用零校验能够很容易地分析CRC校验和,这里必须注意双方的CRC计算都应从相同的起始值开始。

CRC的优点是识别错误的可靠性较好,即使在很多重错误时,也只需要少量的操作就可以识别。

## 3.2 防碰撞算法

### 3.2.1 碰撞产生原理

在近距离通信系统工作时,可能会有一个以上的目标通信方同时处在主动通信方的作用范围内。这样,如果有两个或两个以上的目标通信方同时发送数据时就会出现通信冲突,数据相互的干扰(碰撞);同样有时也有可能多个目标通信方处在多个主动通信方的工作范围之内,他们之间的数据通信也会引起数据干扰,不过在本文中我们不考虑后面的这种情况。为了防止这些冲突的产生,NFC系统中需要设置一定的相关命令,解决冲突问题,这些命令被称为防冲突命令或算法。

### 3.2.2 防碰撞方法分类

在无线通信技术中,通信冲突的问题是长久以来存在的问题,但同时也研究出许多相应的解决方法。基本上有四种不同的方法<sup>[11]</sup>:空分多路法(SDMA-Space Division Multiple Access),频分多路法(FDMA-Frequency Division Multiple Access)、码分多路法(CDMA-Code Division Multiple Access)和时分多路法(TDMA-Time Division Multiple

Access)。

3.2.3 常用的防碰撞算法

主动通信方为了能够顺利地在其射频信号作用范围内的目标通信方的识别、数据信息的读写操作。主要是采用时分多路的方法，使每个目标通信方在单独的某个时隙内占用信道与阅读器进行通信，防止碰撞产生，也就是数据能够准确地在通信双方之间进行传输。在EMCA-340标准中的采用常用算法有ALOHA算法和二进制搜索算法，下面分别进行讨论。

3.2.3.1 ALOHA算法<sup>[16]</sup>

ALOHA算法在EMCA-340中，主要应用在通信速率为212、424kbps的情况下。

1) 纯ALOHA算法：

该算法为最初的ALOHA算法，如图3-1所示，算法原理描述如下：

- (1) 各个标签随机的在某时间点上发送信息；
- (2) 阅读器检测收到的信息，判断是成功接收或是碰撞；
- (3) 标签在发送完信息后等待随机长时间再重新发送信息；
- (4) 若假设一帧信息的长度为 $F$ ，起始时间为 $t_0$ ，那么当另一帧的起始时间 $t_1$ 满足关系式： $t_0-F \leq t_1 \leq t_0+F$ 时，碰撞发生。

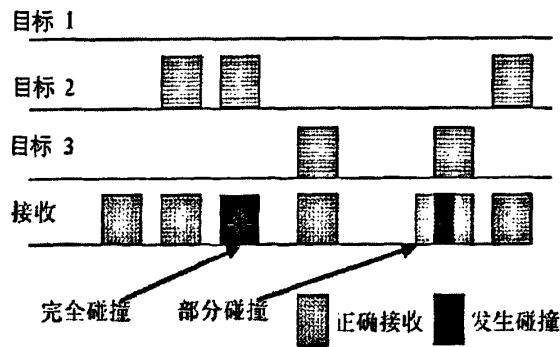


图3-1 纯ALOHA算法示意图

该方法的缺陷较为明显：冲突发生的概率很大，并且由于发送时间的随机性，可能导致完全冲突或部分冲突。

2) 时隙ALOHA：

该算法是纯ALOHA算法的改进，将时间分成多个时隙，如图3-2所示，算法原理描述如下：

- (1) 时间域被分为离散的时间间隔，即时隙(slots)；
- (2) 目标通信方发送信息的起始点不能任意，只能在一个时隙的起始处；

- (3) 目标通信方传送的信息要么不碰撞，要么完全碰撞；
- (4) 目标通信方信息的发送需要主动通信方对其进行同步；
- (5) 与纯ALOHA算法相比，碰撞的区间缩小了一半，信道利用率提高了一倍。

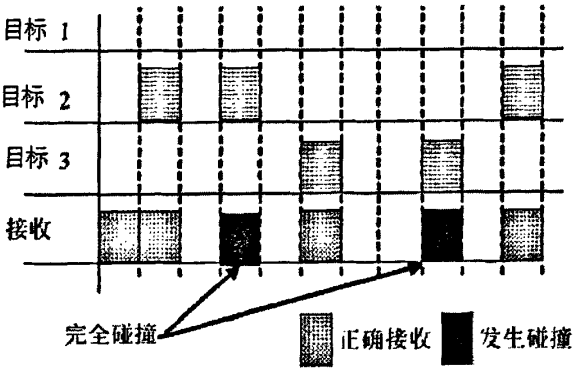


图3-2 时隙ALOHA算法示意图

较高速率（212、424kbps）下采用的就是这个算法，在EMCA-340标准中，最多将时间域分成16个时隙。

在时隙ALOHA算法的基础上还可以进一步进行改进，可以在时间域上进一步离散化。下面两种算法是参考了有关文献<sup>[17]</sup>，可引入NFC系统中使用的改进时隙算法。

3) 帧时隙ALOHA算法

如图3-3所示，算法描述如下：

- (1) 将N个时隙打包成一个帧；
- (2) 标签在每N个时隙中只随机发送一次信息；
- (3) 该方法需要通信双方之间的同步操作包括：每个时隙需要主动通信方进行同步；每一帧的最大时隙数N为某默认值，需要预先设定。

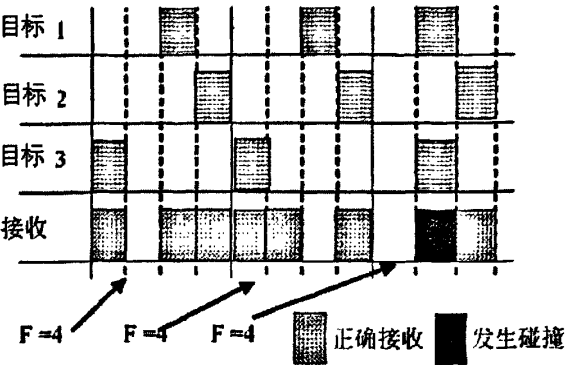


图3-3 帧时隙ALOHA算法示意图

4) 动态帧时隙ALOHA算法

在这个算法中，主动通信方可以为了适应下个周期的目标信息清点临时扩大或者减小一帧中的最大时隙数。如图3-4所示：

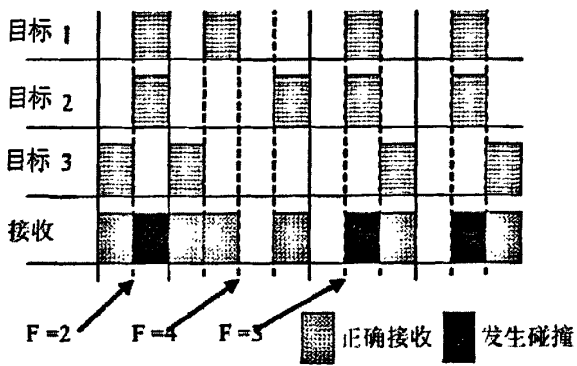


图3-4 动态帧时隙ALOHA算法

5) ALOHA算法比较

上面我们讨论的4种ALOHA算法是在纯ALOHA算法的基础上改进形成的，主要改进手段是把时间域逐步离散化，其中后三种可用在NFC系统中。下面对这几种ALOHA算法进行比较：

- (1) ALOHA算法是最原始的算法，它的碰撞发生几率很大。根据有关文献<sup>[18]</sup>，其信道最大利用率仅为18.4%，所以在实际应用中并不采用。
- (2) EMCA-340标准使用的时隙ALOHA算法是对纯ALOHA算法时间域的初步离散化，信道利用率可提高一倍。
- (3) 后面两种算法，对时间域进行进一步的离散化，将时隙组成帧，并动态调整时隙的数目。这两种方法适用于数据传输量比较大的场合，并且信道利用率和识别率都比较高，软件实现也比较容易，所以可以应用在NFC系统中。

3.2.3.2 二进制搜索算法

该算法主要应用在106kbps速率通信的情况。采用了位冲突监测协议实现防碰撞过程，即主动通信方对目标方返回的唯一标识号（即NFCID）中的每一位进行冲突监测。当多个目标方在同一时刻向发起方发送标识号数据帧时，会在同时返回的某一位上有不同的位值。由这个特性，在二进制防碰撞算法中采用Manchester编码，根据Manchester编码规则，这一位正负边沿抵消了，故发起方无法识别该数据位即为冲突位。冲突位监测到后马上启动防碰撞过程。发起方主动地发出一系列命令（即下传），主要是SDD\_REQ和SEL\_REQ命令。目标方被动地响应每一条指令（即上传）完成交互的会话过程。从而选定唯一目标进行通信，完成防碰撞的过程<sup>[19]</sup>。

图3-5说明了Manchester编码在冲突位检测过程中的应用原理：

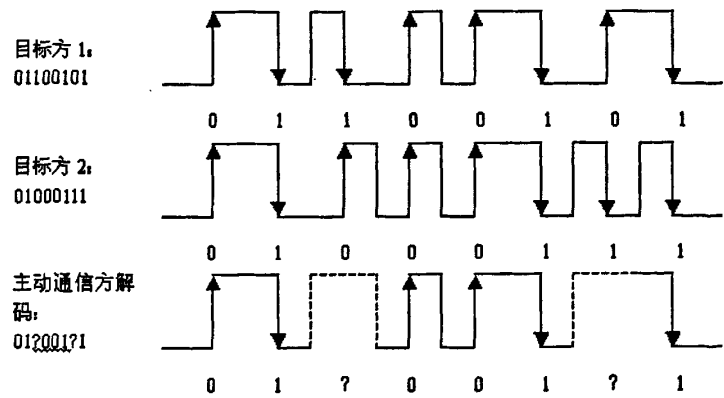


图 3-5 碰撞产生过程示意

当两个目标方同时响应时，可以看到，在第三、七两个比特位，由于数据位不同，信号混叠在一起后解码器无法辨识这两位，所以可以判断出在射频域范围内存在多个目标方。为了方便比较，在这里假定目标通信方的标识号为 8 位。

在 EMCA-340 标准中,采用的是一种改进的二进制搜索算法——动态二进制搜索算法。这种改进算法与原始的二进制搜索算法相比，不再要求发送全部标识号进行整体比较，而是只要求随着主动通信方对冲突比特判别位置的不同而动态的发送目标通信方的一部分标识号，最终选定一个目标进行通信。经实验证明该算法所需传送的数据量和所需时间减少可达 50%<sup>[19]</sup>。

3.2.3.3 两种算法性能比较

EMCA-340 标准中这两种不同类型算法，主要有以下几个优缺点：

- 1) 时隙算法不能很好防止碰撞，当冲突的目标方较多时，防碰撞的时间将过长，但实现较为简单。
- 2) 隙算法存在错误判决问题，当某个目标方长时间没有被识别，则有可能被认为不在射频域范围内。
- 3) 进制搜索算法，算法识别率较高，不存在误判问题，但时延长，泄露信息多，安全性较差。
- 4) 时隙算法信道利用率较低，不适合大量目标方存在的场合，二进制搜索算法信道利用率较高。

3.2.4 动态跳跃防碰撞算法

该算法是在 EMCA-340 中采用的动态二进制搜索算法的基础上，提出的一种新的二进

制防碰撞算法，下面将做重点分析。

该算法相比于动态二进制算法采用了一种回跳策略。在 EMCA-340 标准的二进制搜索算法中，完成与一个目标通信方的通信任务后，主动通信方对剩余目标必须从标识号的第一位开始进行冲突比特位识别。而动态回跳的反碰撞算法对前一次碰撞比特的位置进行保存，当与一个目标通信方通信完毕后，主动通信方回到所有目标第一次的冲突比特位置上，从这里开始对剩余目标进行冲突比特位置的识别，这样就避免了从头开始的工作。从而进一步减少了问询的次数和数据传输量。

在对以上说明的算法进行归纳前，先说明一些约定，请求 ( $NFCID_{m-x}, x$ )，主动通信方发送一个  $NFCID_{m-x}$  参数 ( $NFCID$  值的第  $m-x$  位) 给区域内标签，区域内  $NFCID$  值第  $m-x$  位与之相符的目标通信方应答，返回剩余的  $x$  位信息 (第  $x-1 \sim 0$  位)，这样可缩小预选的目标通信方范围。说明：假设标签的  $NFCID$  为 8 位，则  $m=7$ 。

算法要点归纳如下：

(1) 主动通信方发送 Request ( $NULL, 8$ ) (这个表示方法中，Request 代表由主动通信方发出的命令，NULL 表示第一个命令是不带任何标识号的，8 是  $NFCID$  长度，要求区域内所有目标通信方应答。

(2) 检测有无碰撞发生。若有，确定发生碰撞的第一位。

(3) 碰撞发生时，将碰撞的最高位置 0，高于该位的数值位不变，

(4) 若无碰撞时，可识别出一个单独的目标通信方。 $NFCID$  值为原  $NFCID_{m-x}$  和应答时返回的  $NFCID_{x-1-0}$  组合。处理完后，再屏蔽掉它。同时下一次 Request 命令所需的两个参数，采用回跳策略，从其父节点获得。

(5) 请求与检测的过程重复进行，直到执行 Request ( $NULL, 8$ ) 命令无碰撞发生时结束。

由以上算法过程可知，识别  $N$  个目标通信方所需的总问询次数为  $S(N)=1+(N-1)*2=2N-1$ ，单个目标通信方平均问询次数  $L(N)=S(N)/N \approx 2$ 。

对于上面的算法归纳，我们以下面的实例具体说明：在这个实例中，假设  $NFCID$  为 8 位，在主动通信方范围内有 5 个被动通信方，执行过程如图 3-6 所示。



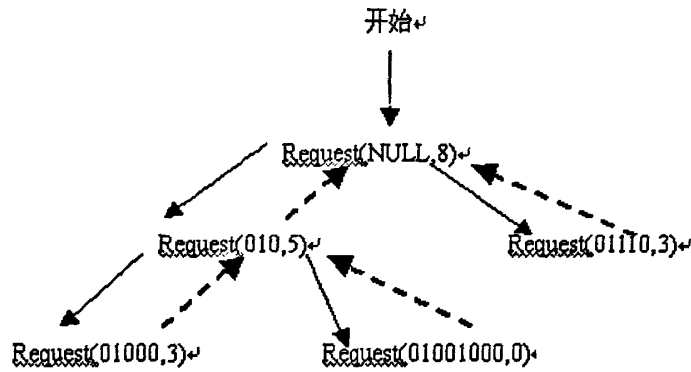


图3-6 防碰撞算法示意图

为了定量分析该算法动态传输NFCID的效率，可采用软件来模拟。根据线索树的遍历思想，该软件利用堆栈结构描述如下：

(1) 随机产生若干目标Target[N];

(2) NFCID=11111111; 置初值

(3) Push-stack(NFCID); 压栈

while (栈不空时)

{

1) pop-stack(NFCID); 取栈顶元素

2) collision=Request(Target, NFCID, x); 找到碰撞位, x为碰撞最高位的下标

3) calculate(Target, NFCID, x); 统计各自总位数

4) if (collision==1); 发生碰撞?

{

① Push-stack(NFCID); 当前NFCID压栈

② Setup\_next\_nfcid(Target, NFCID, x); 形成下一次的NFCID值

③ Push-stack(NFCID); NFCID值入栈

}

else

{

④ data\_nfcid=get\_nfcid(Target, NFCID); 得到一个目标的NFCID值

⑤ Select(NFCID); 选中该目标

⑥ Read\_Data; 读数据

⑦ Unselect; 屏蔽

}

具体模拟代码见附录。

在软件模拟过程中，可一方面变化区域内的目标数，另一方面分别统计动态传输NFCID所需的总位数和传输定长的NFCID所需的总位数。假设NFCID为8位，统计结果如表3-1所示。

表3-1 软件模拟统计结果

目标数	10	20	30	40	50
新算法传输总位数	696	1585	2575	3554	4615
旧算法传输总位数	1020	2422	3927	5522	7098
信道节省率	32.6%	33.9%	34.8%	35.8%	34.7%

结果表明，采用动态方式传输NFCID参数的新算法，比EMCA-340中用定长方式传输NFCID参数的方法节省信息量接近34.4%。

3.3 本章小结

本章在简要讨论数据检验问题之后，重点放在对现有防碰撞算法的研究和改进上。在总结了EMCA-340标准的防碰撞标准和多种可以应用于NFC系统的防碰撞改进算法后，详细讨论了一种新的二进制位搜索算法的改进算法。通过软件模拟进行了性能分析，结果表明，采用动态方式传输NFCID参数的新算法，比EMCA-340中用定长方式传输NFCID参数的方法节省信息量接近34.4%。

## 第四章 高速率近距离通信关键技术

### 4.1 引言

在现有的近距离通信协议标准EMCA-340中,提出的标准主要是围绕通信速率在106kbps、212kbps和424kbps的几种情况,对更高通信速率的协议标准并未有相应的讨论。但是针对未来近距离通信技术应用的场合,希望有更高的数据传输速率以满足日益增长的数据传输量的需要<sup>[20][21]</sup>。

实现高速率的通信,最简单的方法就是通过提高数据流的速率或是改变调制方式来实现。如果采用这样的方法,就要从调制方式和编码方式入手进行考虑。从编码方式来说,原来EMCA-340协议中,用了两种编码方法,改进型Miller码和Manchester码。调制方式只采用了一种调制方式,ASK调制。另外,在低速率情况下,近距离通信系统可在主动模式或被动模式下工作。

本章将从EMCA-340标准现有的调制编码方式和工作模式着手,研究高速率NFC系统实现的关键技术。

### 4.2 高速率通信下的工作模式

通过研究近距离通信协议标准EMCA-340可以知道,在低速率情况下,NFC设备可以工作两种模式下,即主动通信模式和被动通信模式。在高速率(848kbps以上,包括848kbps)情况下,标准中规定工作在主动模式(值得注意的是:这也是在EMCA-340标准中唯一提到高速率通信标准的地方)。也就是说,在高速率通信中,NFC设备不再使用负载调制。由EMCA-340可以知道,在主动通信模式下,通信双方均采用改进型Miller编码进行通信,通信方式为:在两个NFC设备进行通信时,要发送命令或数据的一方打开射频域,而另一方则不打开射频域,被动的参与接收。由于这两个设备之间的作用方式是近场下的电感耦合,故可以实现这种通信模式。

### 4.3 调制技术

NFC的工作频段是12.33-14.99MHz。工作模式为主动通信模式时,为了保证NFC信号的频谱范围在13.56MHz频段内,NFC信号的波特率必须小于1Mbps<sup>[22]</sup>。所以当数据传输速率大于424kbps而小于1Mbps时,调制技术仍然可以采用EMCA-340协议中原有标准,即100%ASK调制;数据传输速率大于1Mbps时,为实现高通信速率,只有采用多进制调制才能满足高速传输要求。采用多进制数字调制具有以下两个特点:

- 1) 在相同的码元速率下，多进制系统的信息传输速率显然比二进制系统高。比如，四进制系统的信息传输速率是二进制系统的2倍。
- 2) 在相同的信息速率下，由于多进制码元传输速率比二进制的低，因而多进制信号码元的持续时间要比二进制的长。显然，增大码元宽度，就会增加码元的能量，并能减小由于信道特性引起的码间干扰等影响。

4.3.1 可用的多进制调制技术

现有比较成熟的多进制调制技术，主要有MASK、MFSK、MPSK、MDPSK、MQAM、MSK、GMSK等等<sup>[23]</sup>。高速率通信情况下，对于调制技术的选择有其自身的要求，其需要满足以下几个条件：

- 1) 调制与解调技术的实现要简单，便于将其集成在手机上。
- 2) 干扰性能好，系统误码率低。

以上条件的提出，一个是对系统的普遍要求：抗噪性能要好，频带利用率要高；另一个是由NFC技术自身特点决定。高速率通信时，采用主动通信模式，要求调制和解调实现简单。由于NFC设备本身产生所用的载波信号，在解调时不用考虑相干或非相干的问题，只要做到解调的复杂度较低即可，以达到降低成本的目的。

下图给出了各种调制方式实现的复杂性的比较<sup>[24]</sup>：

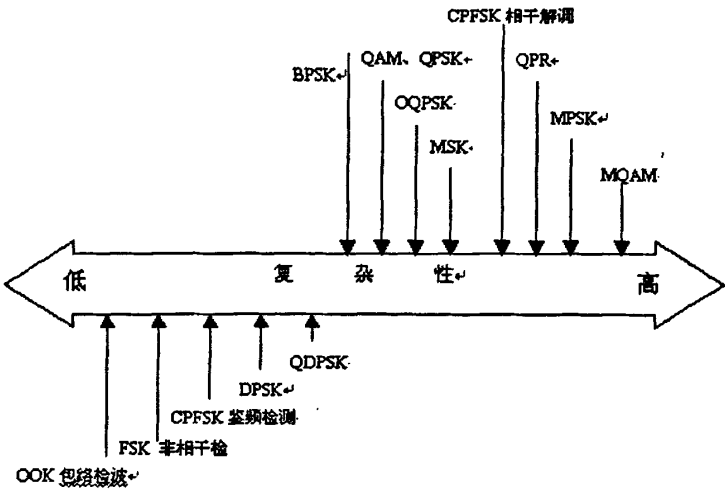


图4-1 数字调制的相对复杂性比较

从图4-1我们可以看到，实现复杂度最低的是OOK包络检波，也最容易硬件实现。这是在EMCA-340标准中选用ASK调制的主要原因。参考图4-1，本章选取以下二种多进制调制方式讨论。

## (1) MASK

在EMCA-340标准低速率情况下, 3种主要速率均采用ASK调制, 所以在本文中首先讨论MASK。

$M$ 进制振幅键控信号中, 载波振幅有 $M$ 种取值, 每个符号间隔 $T_s$ 内发送一种幅度的载波信号,  $M$ 进制振幅键控信号的时域表达式是:

$$S_{MASK}(t) = [\sum_n a_n g(t - nT_s)] \cos \omega_c t \quad (4.1)$$

其中,  $g(t)$ 为基带信号波形,  $\omega_c$ 为载波角频率,  $T_s$ 为信号间隔,  $a_n$ 为幅度值。  $a_n$ 可以有 $M$ 种取值,  $a_n \in \{A_i\}$ ,  $i=0,1,\dots,M-1$ , 这种 $M$ 种取值的出现的概率分别为  $P_0, P_1, \dots, P_{M-1}$ , 可用下列概率场表示:

$$\begin{pmatrix} A_0, & A_1, A_2, \dots, A_{M-1} \\ P_0, & P_1, P_2, \dots, P_{M-1} \end{pmatrix} \quad (4.2)$$

$$\sum_{i=0}^{M-1} P_i = 1$$

下图是4ASK基带波形及调制波形:

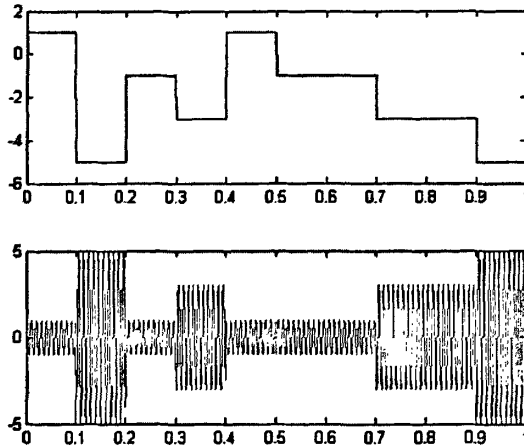


图4-2 4ASK基带波形及调制波形

由上面的理论分析可以知道, MASK信号的功率谱与2ASK时完全相同。它相当于 $M$ 电平基带信号对载波进行双边带调幅, 因此带宽是 $M$ 电平基带信号的两倍, 由于 $M$ 电平信号每个码元间隔内可以传送  $\log_2 M$  比特信息, 码元速率降为信息速率的  $1/\log_2 M$  倍, 因此信号的带宽在相同信息速率时是2ASK的  $1/\log_2 M$  倍。

显然, MASK的调制方法与2ASK相同, 不同的只是基带信号由二电平变为多电平。为

此, 可以将二进制信息序列分为 $n$ 个一组,  $n = \log_2 M$ , 然后变换为 $M$ 电平基带信号, 再送入调制器。由于采用多电平, 因而要求调制器为线形调制器, 即已调信号幅度应与输入基带信号幅度成正比。

假定信道不使信号发生任何畸变, 而且接收端输入带通滤波器有理想特性, 信道内噪声假定为窄带高斯过程, 采用同步检测法, 则多电平调制系统的总误码率为:

$$P_e = \left(1 - \frac{1}{L}\right) P(|n_c| > d) \quad (4.3)$$

因为 $n_c$ 是均值为零、方差为 $\sigma_n^2$ 的正态随机变量, 则可得

$$P_e = \left(1 - \frac{1}{L}\right) \operatorname{erfc}\left(\frac{d}{\sqrt{2}\sigma_n}\right) \quad (4.4)$$

若各信号出现的可能性相同, 则信号功率即为均方值的统计平均:

$$P_s = d^2 \frac{L^2 - 1}{6} \quad (4.5)$$

在这里 $d$ 是多电平调制时相邻电平的一半。有上面两式, 我们可以得到:

$$P_e = \left(1 - \frac{1}{L}\right) \operatorname{erfc}\left(\frac{3}{L^2 - 1} \frac{P_s}{\sigma_n^2}\right)^{1/2} \quad (4.6)$$

因为 $\sigma_n^2$ 即是输入噪声的平均功率, 若令信噪比 $P_s/\sigma_n^2$ 等于 $r$ , 则可得:

$$P_e = \left(1 - \frac{1}{L}\right) \operatorname{erfc}\left(\frac{3}{L^2 - 1} r\right)^{1/2} \quad (4.7)$$

根据上面的分析, 图4-3给出了系统误码率 $P_e$ 与信噪比 $r$ 的关系曲线。

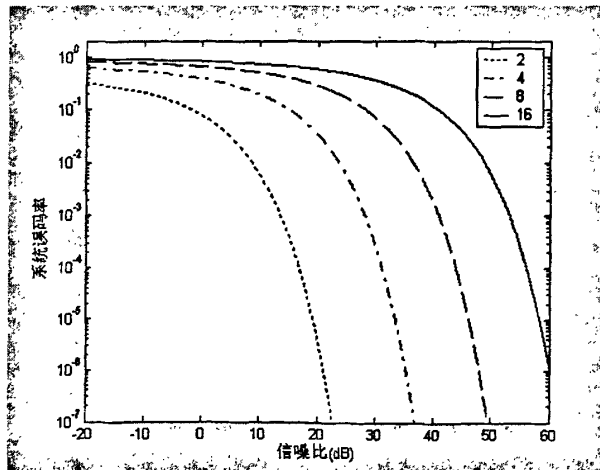


图4-3 多电平调幅时的系统性能曲线

如果  $M$  进制振幅调制的输入码元与二进制振幅调制的输入码元具有相同速率, 那么  $M$  进制振幅调制与二进制振幅调制具有相同的带宽, 而  $M$  进制振幅调制具有更高的带宽利用率, 因为在单位时间内  $M$  进制振幅调制能传输更多的比特。虽然  $M$  进制振幅调制具有更高的带宽利用率, 但是在相同信噪比下, 随着  $M$  增大, 误码率也增大, 也就是因为功率受限, 因而它一般只适宜在恒参信道使用。

## (2) MDPSK

MDPSK 调制方式具有较强的抗噪能力和较高的频谱利用率。四进制时, 最大频谱利用率为  $2\text{bit/s/Hz}$ ; 八进制时为  $3\text{bit/s/Hz}$ 。

$M$  进制的 DPSK 调制可以表示为:

$$e_0(t) = \sum_{k=-\infty}^{\infty} g(t - kT_s) \cos(\omega_c t + \varphi_k) = \sum_{k=-\infty}^{\infty} a_k g(t - kT_s) \cos \omega_c t - \sum_{k=-\infty}^{\infty} b_k g(t - kT_s) \sin \omega_c t \quad (4.8)$$

式中:  $\varphi_k$ ——受调相位, 可以有  $M$  种不同的取值;

$$a_k = \cos \varphi_k ;$$

$$b_k = \sin \varphi_k ;$$

多相调制波形可以看作是对两个正交载波进行多电平双边带调制所得之和。这就说明, 多相调制信号的带宽和多电平双边带调制时相同。

$M$  进制的 DPSK 调制, 虽然  $M$  越大频谱利用率越高, 但是信号之间的相位差也变的越来越小, 可靠性降低, 因此在通常情况下,  $M$  值不能选取的太大, 一般来说, 四进制和八进制是最常用的。下面我们将以四进制为例讨论 MDPSK, 即 4DPSK。4DPSK 调制框图和调制信号及其频谱如图 4-4 及 4-5 所示。

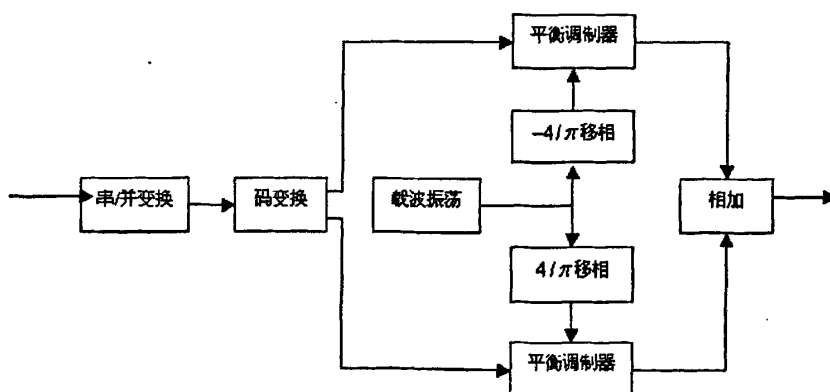


图 4-4 4DPSK 调制框图

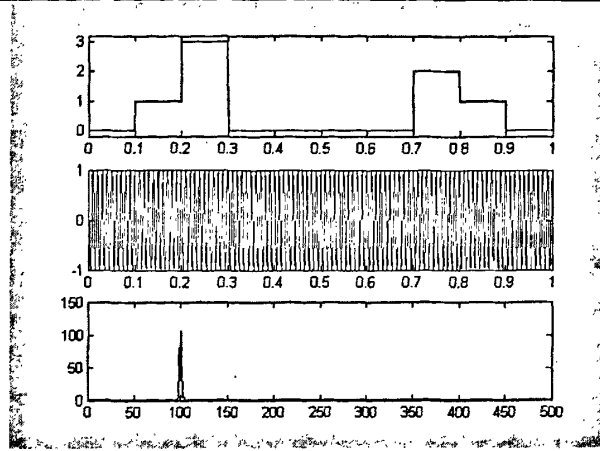


图 4-5 未调制波形及调制后波形频谱

MDPSK的解调方法有相位比较法和极性比较法，我们在本文中以4DPSK为例讨论相位比较法解调问题。比较法解调框图如图4-6所示：

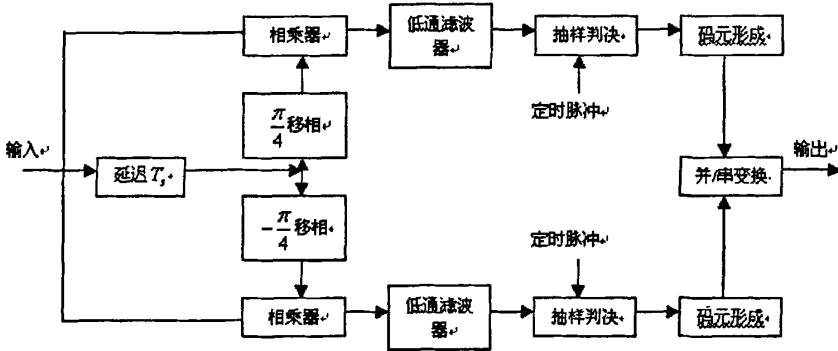


图4-6 4DPSK解调原理框图

若不考虑信道引起的失真及噪声的影响，加到接收端的信号在一个码元持续时间内可表示为：

$$s(t) = g(t) \cos(\omega_c t + \varphi_k) \quad (4.9)$$

上支路相乘器输出为：

$$g(t) \cos(\omega_c t + \varphi_k) \cdot g(t - T_s) \cos(\omega_c t - \omega_c T_s + \varphi_k + \frac{\pi}{4}) = \quad (4.10)$$

$$\begin{aligned} & \frac{1}{2} g(t) g(t - T_s) \cdot \cos(2\omega_c t + 2\varphi_k + \frac{\pi}{4} - \omega_c T_s) \\ & + \frac{1}{2} g(t) g(t - T_s) \cos(\omega_c T_s - \frac{\pi}{4}) \end{aligned} \quad (4.11)$$

低通滤波器输出为：

$$\frac{1}{2} g(t) g(t - T_s) \cos(\omega_c T_s - \frac{\pi}{4}) \quad (4.12)$$

同理，下支路经低通滤波器输出为：



$$\frac{1}{2}g(t)g(t-T_s)\cos(\omega_c T_s + \frac{\pi}{4}) \quad (4.13)$$

上下支路在  $t = T_s$  时刻抽样，可分别表示为：

$$U_A \propto \cos(\omega_c T_s - \frac{\pi}{4}) \quad (4.14)$$

$$U_B \propto \cos(\omega_c T_s + \frac{\pi}{4})$$

由上面的分析，我们可以得到表4-1的判决规则：

表4-1 4DPSK判决规则

载波相位变化 $\varphi_k$	上支路抽样 $U_A$ 极性	下支路抽样 $U_B$ 极性	判决器输出	
			第一路码元	第二路码元
$0^\circ$	+	+	0	0
$90^\circ$	-	+	1	0
$180^\circ$	-	-	1	1
$270^\circ$	+	-	0	1

在差分检测时，由于前一码元的相位是受干扰的，故合成相位波形相位  $\theta$  在

$$\varphi_0 - \frac{\pi}{M} < \theta < \varphi_0 + \frac{\pi}{M} \quad (4.15)$$

范围内才不发生错判，其中  $\varphi_0$  为参考信号（即前一码元信号）相位，这时错判概率为：

$$P_e(\varphi_0) = 1 - \int_{\varphi_0 - \frac{\pi}{M}}^{\varphi_0 + \frac{\pi}{M}} f(\theta) d\theta \quad (4.16)$$

考虑到  $\varphi_0$  也是随机的，故其概率密度为  $q(\varphi_0)$ ，则系统的总误码率为：

$$P_e = \int_{-\pi}^{\pi} q(\varphi_0) P_e(\varphi_0) d\varphi_0 \quad (4.17)$$

由上面的分析，仿真结果如图4-7所示：

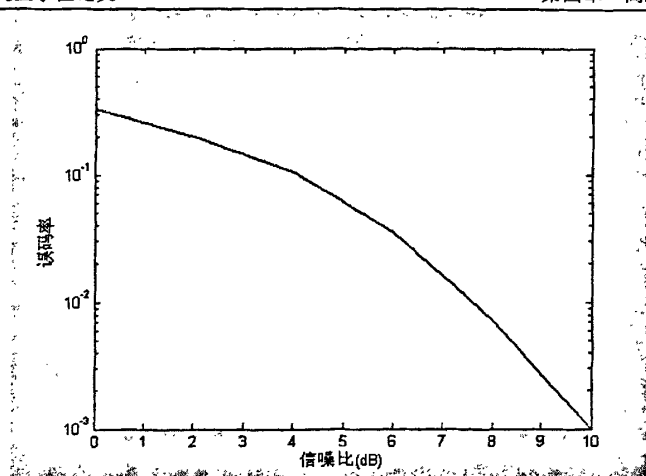


图4-7 4DPSK解调误码率仿真曲线

### 4.3.2 高速率NFC系统调制方式

经过上面的内容讨论,通过比较调制解调方式硬件实现难易、信道利用率等多个性能,在高速率近距离通信系统中,可以选用MDPSK作为高速率通信时的调制方式。

## 4.4 编码方式

在高速率通信情况下,编码的脉冲宽度越来越窄,对电路脉冲响应的要求也越来越高,除此之外编码在信道传输过程中的抗噪声性能也是十分关键的技术指标。在这节的讨论中,主要是为高速率通信选取一种或几种适合的码型,码型的来源已经在第二章做了简要讨论。

### 4.4.1 码型选取方法

本节首先分析码型在信道传输中的抗噪性能,判断不同码型在传输中的误码情况,然后以此为依据选取码元。

为了能够仿真各信道编码的抗噪声性能,必须引入一个能够产生随机错误的信道,由于所有的数据都是以二进制的方式传送的,所以引入的信道是最常见的二进制对称信道。这样的信道是一个抽象程度非常高的信道模型。任何的信道,不管是采用基带传输还是频带传输,也不管采取什么调制编码方式,只要是用来传输数字信号,都可以抽象成这样一个数学模型。

信号源随机的产生一组二进制数据,代表NFC系统发出的原始数据或命令,将这个数据采用前面的提到的编码方式进行编码(下面会根据需要给出部分编码方式的仿真),编码后的数据送入二进制对称信道信道进行传输。在接收端解码以前,先根据码型判断是不是有错误产生,再将数据解码,将解码后的数据与原始数据进行比较,看是不是有错误发生,以此来检验该编码方式的检错能力,作为选择编码码型的判断依据。

### 4.4.2 编码方式仿真

常用码型有单极性归零编码、Manchester编码、Miller编码和改进型Miller编码。

#### 1) 单极性归零编码

单极性归零码的编码规则是：在第一个半比特周期中的“高”信号表示1，而持续整个比特周期的“低”信号表示0；该编码的simulink模块内部实现如图4-8所示。

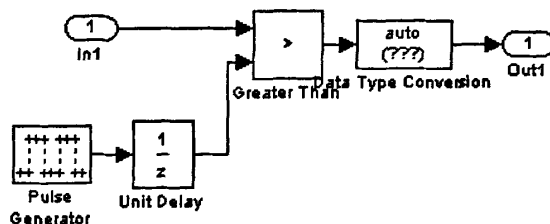


图4-8 单极性归零码编码子系统仿真模型

即用一个频率为原始数据发送频率两倍的起始电平为低电平的矩形波与原始数据做异或运算，即可实现单极性归零码的编码。

单极性归零码的解码方法是用一个频率为原始信号频率两倍的起始电平为高电平的矩形波与原始信号作与运算后，再将原始信号经过下采样就可以得到解码后的信号。解码的simulink实现如图4-9所示：

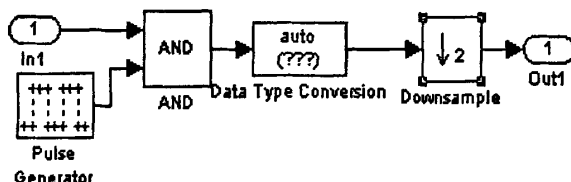


图4-9 单极性归零码解码子系统仿真模型

编解码仿真模型如图4-10所示：

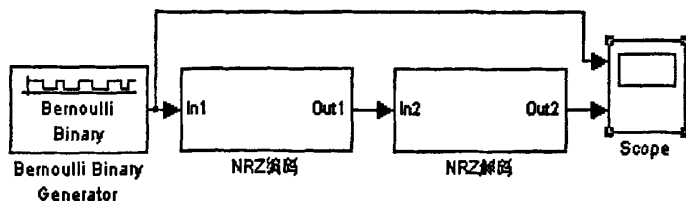


图4-10 单极性归零码编解码仿真模型

仿真结果：

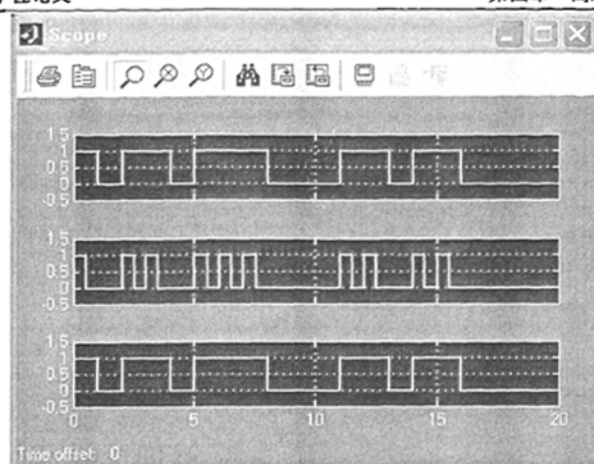


图4-11 单极性归零码编码仿真结果

图4-11中第一行显示的是原始信号的波形，第二行显示的经单极性归零码编码后的波形，第三行显示的是解码后的波形，可见仿真的结果符合要求。

## 2) Manchester编码

Manchester编码的编码规则是：如果原始数据为“1”，将其编码成“10”；如果原始数据为“0”，将其编码成“01”。这种编码的特点是每个码元中间都有跳变，使得这种码低频能量较少，而且这种码的特点也便于接收端提取时钟信号。

仿真实现时，采用一个频率为原始数据发送频率两倍的矩形波与原始数据做异或或者同或运算即可实现Manchester编码。其实现如图4-12所示：

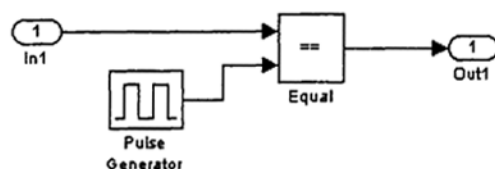


图4-12 Manchester编码仿真子系统模型

Manchester解码的目的是从收到的Manchester码流中恢复出原始信号，仿真实现时可以使用和编码相反的方法，即用一个频率为原始码流频率一半的矩形波与原始数据做异或或者同或运算即可实现Manchester解码。模块和形式上，编码与解码模块相同，不同的地方在于脉冲发生器的频率，接下来对这两个子系统进行封装，可以利用simulink的子系统模块封装并仿真。

编解码仿真模型如图4-13所示：

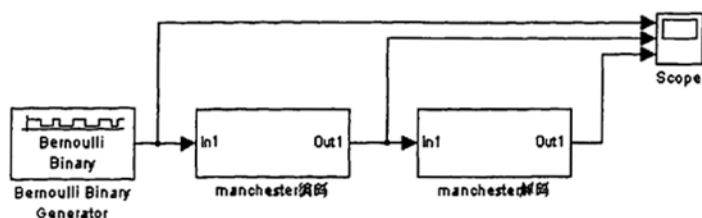


图4-13 Manchester编码解码仿真模型

仿真结果:

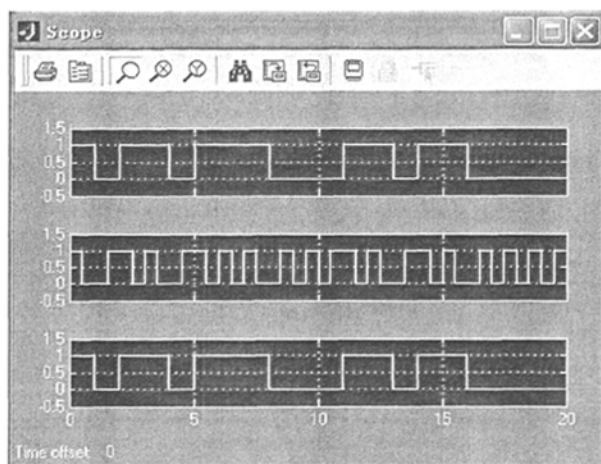


图4-14 Manchester编码解码仿真波形

图4-14中，第一行显示的是原始信号波形，第二行显示的是经Manchester编码后的波形，第三行显示的是解码后的波形，可见仿真的结果符合我们的要求。

### 3) Miller编码、改进型Miller编码

在本文的工作过程中，发现Miller编码和改进型Miller编码用普通的Matlab/simulink模块并不容易搭建，因此利用S函数<sup>[25]</sup>来自己编程实现所需要的模块，S函数可以极大的扩展simulink的功能，使得simulink的仿真变的更灵活，更强大。

要使用S函数编程，必须了解simulink的工作机制，simulink仿真时必须经历三个阶段：

(1) 模型的初始化阶段：在这一阶段，simulink将确定模块的执行顺序，初始化模块的参数，确定模块的输入输出个数、离散或连续状态的个数，采样时间等。

(2) 状态的动态更新阶段：这一阶段在每一次系统要更新模块是离散还是连续分别更新模块的状态，计算下一个采样点的时刻。

(3) 输出信号的计算阶段，根据模块现在状态和当前模块的输入来计算模块的输出。

S函数可以用matlab语言编写，也可以用c、c++等多种语言编写，本文采用matlab语言编写S函数。

S函数的引导语句为：

```
function[sys,x0,str,ts]=f(t,x,u,flag,p1,p2,...)
```

其中 $f$ 为 $s$ 函数的函数名， $t$ 、 $x$ 、 $u$ 分别为时间、状态和输入信号， $flag$ 为标志位，其意义和有关信息在表4-2中给出，一般应用中很少使用 $flag$ 为4和9。在该表中还解释了在不同的 $flag$ 下返回的参数类型。在该函数中还允许使用任意数量的附加参数 $p1, p2, \dots$ 等。编程实现时，在各个状态对应的函数下编写相应的功能模块即可。

表4-2 flag参数表

Flag	功能	返回参数
0	初始化	初始化参数
1	计算连续状态	连续状态
2	计算离散状态	离散状态
3	计算输出	系统输出
4	下一步时间	下一步仿真时间
9	终止仿真	无返回值

先讨论Miller编码的实现，改进型Miller编码是在前者的基础上，在每个脉冲跳变的地方用四分之一负脉冲替代。

Miller编码的规则是：二进制“1”用“01”或“10”表示，二进制“0”以“00”或“11”表示，单个二进制“0”延续前一位比特位状态，多个连续“0”，每出现一个“0”发生一次电平状态跳变。

根据以上内容，结合Miller编码的编码格式，设计的 $s$ 函数如下：

```
function[sys,x0,str,ts]=f(t,x,u,flag,p)
switch flag,
case 0, [sys,x0,str,ts]=mdlInitializeSizes(p);
case 1, sys=mdlDerivatives(t,x,u);
case 2, sys=mdlUpdate(t,x,u);
case 3, sys=mdlOutputs(t,x,u);
case 4, sys=mdlGetTimeOfNextVarHit(t,x,u);
case 9, sys=mdlTerminate(t,x,u);
otherwise
    error(['Unhandled flag=',num2str(flag)]);
end
```

%模型初始化

```
function[sys,x0,str,ts]=mdlInitializeSizes(p)
```

```
sizes=simsizes;
```

```
sizes.NumContStates=0;
```

```
sizes.NumDiscStates=3;
```

```
sizes.NumOutputs=1;
```

```
sizes.NumInputs=1;
```

```
sizes.DirFeedthrough=1
```

```
sizes.NumSampletimes=1;
```

```
sys=simsizes(sizes);
```

```
x0=[0;0;0];
```

```
str=[];
```

```
ts=[p 0];
```

%计算连续状态

```
function sys=mdlDerivatives(t,x,u)
```

```
sys=[];
```

%计算离散状态

```
function sys=mdlUpdate(t,x,u)
```

```
x(1)=x(1)+1;
```

```
if rem(x(1),2)==0 %如果是1bit周期的开始
```

```
if u==1
```

```
if x(2)==0 x(2)=0;
```

```
else x(2)=1;
```

```
end
```

```
else
```

```
if x(3)==0
```

```
if x(2)==1 x(2)=0
```

```
else x(2)=1
```

```
end
```

```
else
```

```
if x(2)==0 x(2)=0;
```

```

        else x(2)=1;
    end
end
end
else %如果是1bit周期的中间
    if u==1
        if x(2)==0 x(2)=1;
        else x(2)=0
        end
    end
end
x(3)=u;
sys=x;
%将x(2)输出，即得所需编码
function sys=mdlOutputs(t,x,u)
sys=x(2);
%计算下一次仿真时间
function sys=mdlGetTimeOfNextVarHit(t,x,u)
sampleTime=1;
sys=t+sampleTime
%结束仿真
function sys=mdlTerminate(t,x,u)
sys=[];

```

mdlInitializeSizes(p)函数完成模块的初始化，仿真时在模型的初始化阶段运行该函数，由于仿真的是离散系统，所以设定连续状态数为0；设定离散状态数为3，它们是x(1)、x(2)和x(3)。x(1)用来记录状态数，每仿真一次该值加1，x(2)用来记录下一次的输出，后面将要用到这些状态，还使用x(3)来记录数据流中前一时刻数据流的输入状态，用来和下一时刻输入比较同时设定系统的输入输出数、采样时间等参数。

mdlUpdate(t,x,u)函数用来更新模型的状态，它在一次仿真结束时执行，目的是为下一次仿真做准备，Miller编码的实现的主要工作就是在这一函数中完成。

MdlOutputs(t,x,u)函数用来输出本次仿真的输出结果，在Miller编码中只需将在上一次



仿真结束后更新得到的结果输出即可。

在仿真得到Miller编码的基础上，以上面的模块为输入，根据改进型Miller码的编码原则，即可得到改进型Miller编码。设计的s函数如下：

```
function[sys,x0,str,ts]=f(t,x,u,flag,p)

switch flag,

case 0, [sys,x0,str,ts]=mdlInitializeSizes(p);

case 1, sys=mdlDerivatives(t,x,u);

case 2, sys=mdlUpdate(t,x,u);

case 3, sys=mdlOutputs(t,x,u);

case 4, sys=mdlGetTimeOfNextVarHit(t,x,u);

case 9, sys=mdlTerminate(t,x,u);

otherwise

    error(['Unhandled flag=',num2str(flag)]);

end

%模型初始化

function[sys,x0,str,ts]=mdlInitializeSizes(p1)

sizes=simsizes;

sizes.NumContStates=0;

sizes.NumDiscStates=3;

sizes.NumOutputs=1;

sizes.NumInputs=1;

sizes.DirFeedthrough=1

sizes.NumSampletimes=1;

sys=simsizes(sizes);

x0=[0;0;0];

str=[];

ts=[p 0];

%计算连续状态

function sys=mdlDerivatives(t,x,u)

sys=[];

%计算离散状态
```

```
function sys=mdlUpdate(t,x,u)

x(1)=x(1)+1;

if rem(x(1),4)==0 %如果是1bit周期的开始

    if x(1)==1

        x(2)=1;

    else

        if u==x(3)

            x(2)=1;

        else x(2)=0;

        end

    end

end

if rem(x(1),4)==1 %如果是1bit周期的四分之一处

    if u==x(3)

        x(2)=1;

    else x(2)=0;

    end

end

if rem(x(1),4)==2 %如果是1bit周期的中间

    if u==x(3)

        x(2)=1;

    else x(2)=0;

    end

end

if rem(x(1),4)==3 %如果是1bit周期的四分之三处

    if u==x(3)

        x(2)=1;

    else x(2)=0;

    end

end

x(3)=u;
```

```
end
%将x(2)输出，即得所需编码
function sys=mdlOutputs(t,x,u)
sys=x(2);
%计算下一次仿真时间
function sys=mdlGetTimeOfNextVarHit(t,x,u)
sampleTime=1;
sys=t+sampleTime
%结束仿真
function sys=mdlTerminate(t,x,u)
sys=[];
```

mdlInitializeSizes(p)函数完成模块的初始化，仿真时在模型的初始化阶段运行该函数，由于仿真的是离散系统，所以设定连续状态数为0；设定离散状态数为3，它们是 $x(1)$ 、 $x(2)$ 和 $x(3)$ 。 $x(1)$ 用来记录状态数，每仿真一次该值加1， $x(2)$ 用来记录下一次的输出，后面将要用到这些状态，还使用 $x(3)$ ，它的作用是在每次仿真结束后记录下本次仿真中数据流的信号值，以便在下一次仿真时与数据流的输入值进行比较。同时设定系统的输入输出数、采样时间等参数。

mdlUpdate(t,x,u)函数用来更新模型的状态，它在一次仿真结束时执行，目的是为下一次仿真做准备，改进型Miller编码的实现的主要工作就是在这一个函数中完成。

MdlOutputs(t,x,u)函数用来输出本次仿真的输出结果，在改进型Miller编码中只需将在上一次仿真结束后更新得到的结果输出即可。

改进型Miller码到Miller码的解码过程为：先将输入的数据流解码为Miller编码，在此基础上，再将Miller编码解码为原始数据流。

将改进型Miller码转换成Miller编码的规则为：输入数据流的前半个比特译为高电平，半比特时间后，若输入数据流为高电平，则第一比特译为“0”，整个比特时间内电平为低，没有变化；若输入数据流电平发生跳变，则第一比特译为“1”整个比特时间内，前半比特电平为高，后半比特电平为低。此后的译码原则为，如输入为高电平，则延续前一比特电平值不变，若出现四分之一负脉冲跳变则改变输出电平状态，如此循环直至结束，可将数据流译码为Miller码。

在得到Miller编码以后，以此作为输入数据流，在执行到1比特的中间时，若此时的输入与上一时刻相同，则译为“0”，否则译为“1”。

根据以上原则, 还可以分别设计出改进型Miller编码解码和Miller编码的解码S函数。由上面的讨论, 仿真模型与Manchester编码相似, 仿真结果如图4-15及4-16所示:

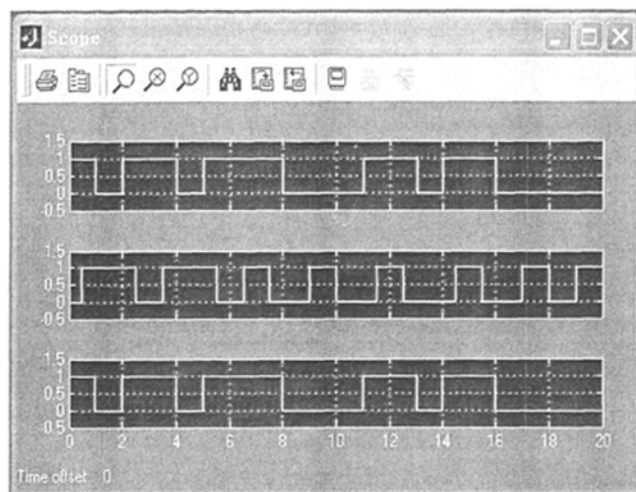


图4-15 Miller编码仿真结果

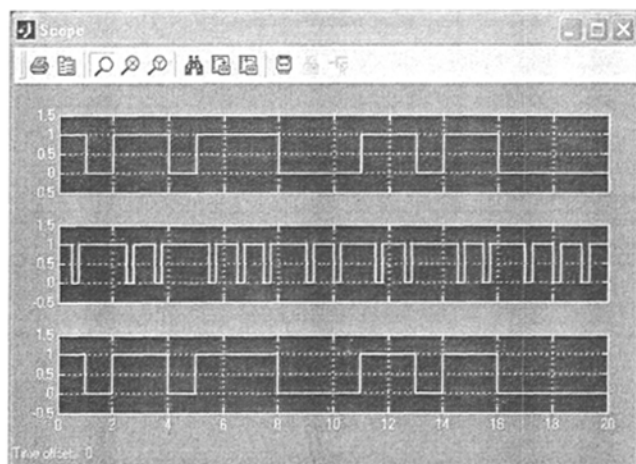


图4-16 改进Miller编码仿真结果

从以上两图可以看出, 仿真结果满足要求。依据上面的讨论, 还可以分别编写其他几种常见编码方式的编解码的S函数, 在本文就不再详细讨论了。

#### 4.4.3 NFC系统数据编码方式检错能力仿真

为了获得相应的结果, 本文采用模型加程序的仿真方式, 即以图4-17模型为基础, 由程序提供相关模型参数并控制模型运行。

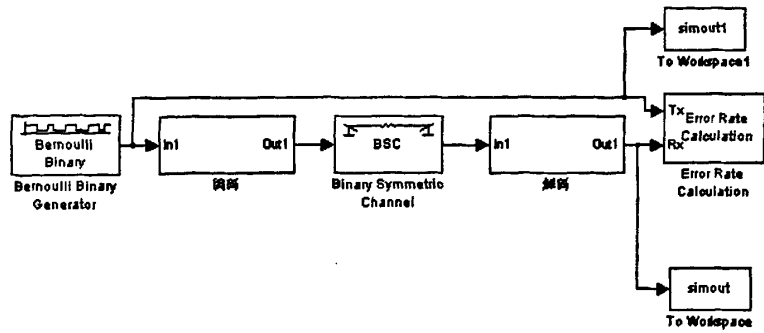


图4-17 编码方式仿真模型

根据近距离通信的实际应用情况，数据通信时每帧数据大体在几个字节到十几个字节之间，因此仿真时设定一帧的长度为100，并让二进制对称信道的错误率在1%-50%之间变化，仿真中，设定仿真次数为500次。具体过程为：由信源发出数据帧，经过Manchester编码或改进型Miller编码二进制对称信道传输；接收端将接收到的数据与原编码进行比较，并累计检出错误的个数，并由下式计算差错的检出率：

$$\alpha = \frac{err1}{err} \%$$

(4.18)

式中 $\alpha$ 为差错的检出率， $err1$ 为累计检出错误次数， $err$ 为总的实际错误个数。我们可以将仿真模型的Manchester编码模块和解码模块更换成其他的编码和解码模块并用同样的方式进行仿真，从而比较各种编码方式的检测错误的能力。仿真的结果见表4-3：

表4-3 编码检错能力仿真结果

编码方式	仿真次数	总的错误次数	检出错误次数	检出率	漏检率
Manchester	500	424	301	71%	29%
单极性归零码	500	423	361	85%	15%
改进Miller码	500	426	299	70%	30%
Miller码	500	403	0	0%	100%

从仿真结果可以看出，由于Miller编码后的波形在接收端几乎没有办法把握该码型的特点，也就没有办法在解码以前判断有没有错误发生，因此Miller编码不能用来在NFC系统数据通信做冲突检测和错误检测的工作。Manchester编码、单极性归零码以及改进miller编码都具有一定的错误检测的能力，但是检测错误的能力都不是很高，大体上都有15%到30%的漏检率。也可以看出编码方式所具有的检测错误的能力随编码方式的不同而不同，因此如果想提高错误检测的能力，选择检测错误能力较高的码是一种不错的方案。

由上面的分析，在高速近距离通信系统中，比较适宜采用Manchester编码、单极性归

零码、改进型Miller码。

4.5 高速率通信防碰撞技术

在EMCA-340中，主动通信模式下，采用了一种与常用防碰撞方法不同的算法。在高速率情况下，我们将继续沿用这种方法。

主动防碰撞算法的执行时序如图4-18所示：

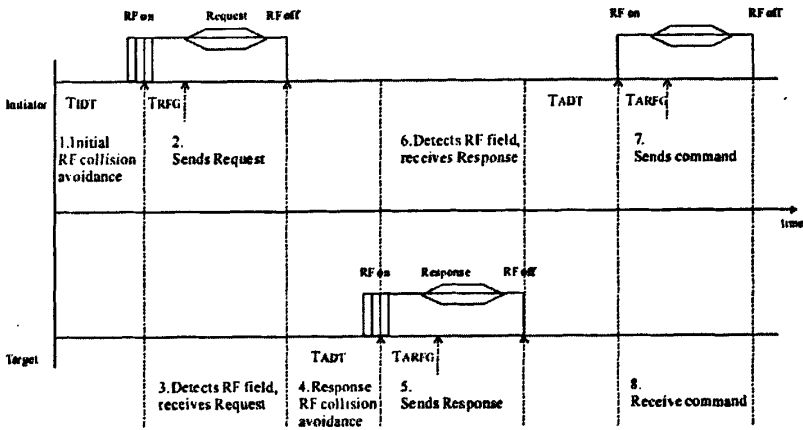


图4-18 主动通信模式防碰撞过程时序

主动模式下使用的防碰撞算法充分的利用了NFC 技术双向识别和连接的特点。它把通信过程分为不同的时间片段，在不同的时间段内，或主动通信方打开射频域，或目标通信方打开射频域。当某一方打开自己射频域时，另一方对该射频域进行侦听被动的响应接收对方发出的命令。

在目标通信方的识别上，该算法使得每一个目标通信方都配上一个跳跃系数，在某一特定的通信时段内，多个通信目标方做出响应，主动通信方从中选出系数最小的一个进行通信。通信结束后，跳跃系数重新置零，进行下一次通信。这个方法较为简单，在有源条件下大大的缩短了防碰撞的时间和过程。

4.6 本章小结

本章是本文重点章节。结合EMCA-340标准，主要讨论在高速率近距离通信时的关键技术，主要包括：工作模式，编码技术，调制方式以及防碰撞技术，并进行了仿真分析。

## 第五章 系统性能仿真与分析

### 5.1 引言

本章将在前一章的基础上,以高速率NFC系统的各项关键技术为基础,对系统性能进行仿真分析。根据前面的结论,本章中的仿真主要分成两个部分,一个是采用EMCA-340调制编码方案的通信速率等于847kbps的高速率NFC系统;另一个是采用第四章调制编码方案的通信速率大于1Mbps的高速率NFC系统。

下一小节开始,先从理论上分析NFC系统的信道模型,然后以此信道为基础,对两个主要系统的系统性能进行仿真分析。

### 5.2 近距离通信系统信道模型

通信信道通常可以分为恒参信道和随参信道,对于NFC系统使用的无线信道来说,传输路径往往是多径的,由于多径传输条件随环境的多变,合成的传输参数也是经常变化的,因此NFC系统的通信信道也应该作为随参信道来处理。

无线信道有一个明显的特点,就是传输过程中的不稳定性。NFC系统的通信信道具有无线电信道的基本特性——衰落特性,因而我们将它作为无线衰落信道来进行分析。对NFC系统的通信信道来说,衰落可以看成是信道的参数在不断的变化,它所引起的对通信的干扰实际上是乘性干扰。

从本质上讲,NFC系统的衰落是由于多径所引起的。在具体的应用环境中,多径性主要来自以下几个方面:

- 地面的反射
- 人体的反射
- 周围其他介质的反射

由于多径性的影响,在发射天线的读写场区的场强有时候会不稳定,时强时弱,导致读写也出现一定的不稳定性。下面分析一下这种信道的特点。

要传输的信号为 $u_0(t)$ ,经过信道传输接收到的信号为 $u(t)$ 。传输的无线电信号是把 $u_0(t)$ 幅度调制在频率为 $f_0$ 的无线电载波上。上一章讨论的几种多进制调制方式,其数学表达式与振幅调制基本相同。因此可设发送的数字信号的已调载波的形式为

$$S_0(t) = \text{Re}[u_0 \exp(j2\pi f_0 t)] \quad (5.1)$$

设有 $N$ 条路径传输,收到的信号为经过各条路径传输而来的信号之和

$$S(t) = \sum_{k=1}^N a_k S_0(t - t_0 - \tau_k) \quad (5.2)$$

式中,  $a_k$  为第K条路径的传输衰减,  $t_0$  为传输参考时间, 可选为主传输路径的传输时间,  $\tau_k$  为第K条路径的传输时间与参考时间差。

对于NFC系统来说, 被传输信号  $u_0(t)$  的频率要远低于载波频率, 用  $U_0(f)$  来表示  $u_0(t)$  的频谱, 则有

$$u_0(t) = \int_{-\infty}^{\infty} U_0(f) \exp(j2\pi ft) df \quad (5.3)$$

将式5.1代入式5.2就得到接收到信号的表达式

$$S(t) = \text{Re}\{u(t - t_0) \exp[j2\pi f_0(t - t_0)]\} \quad (5.4)$$

式中幅度信号的表达式为

$$u(t) = \sum \alpha_k u_0(t - \tau_k) \exp(-j2\pi f_0 \tau_k) \quad (5.5)$$

将传输信号用频域表示, 得到下式

$$u(t) = \int_{-\infty}^{\infty} U_0(f) \exp(j2\pi ft) \left\{ \sum_K \alpha_k \exp[-j2\pi(f + f_0)\tau_k] \right\} df \quad (5.6)$$

由上式可以看出信道的等效传输函数为

$$H(f) = \sum_K \alpha_k \exp[-j2\pi(f + f_0)\tau_k] \quad (5.7)$$

如果  $H(f)$  在频带范围内是常数, 则由上式可以看出, 经过信道收到的信号只不过受到一个衰减, 并没有时延, 在波形上和原信号一样没有失真。为分析  $H(f)$  的特性, 现将它展开, 只取前三项:

$$\begin{aligned} H(f) = & \left[ \sum_K \exp(-j2\pi f_0 \tau_k) \right] - j2\pi f \left[ \sum_K \alpha_k \tau_k \exp(-j2\pi f_0 \tau_k) \right] \\ & + \frac{(j2\pi f)^2}{2} \left[ \sum_K \alpha_k \tau_k^2 \exp(-j2\pi f_0 \tau_k) \right] \end{aligned} \quad (5.8)$$

将式中各项分别作如下定义:

$$\begin{cases} H^{(0)} = \sum_K \alpha_k \exp[-j2\pi f_0 \tau_k] \end{cases} \quad (5.9)$$

$$\begin{cases} H^{(1)} = \sum_K \alpha_k \tau_k \exp[-j2\pi f_0 \tau_k] \end{cases} \quad (5.10)$$

$$\begin{cases} H^{(2)} = 2\pi^2 f^2 \sum_K \alpha_k \tau_k^2 \exp[-j2\pi f_0 \tau_k] \end{cases} \quad (5.11)$$

$H^{(0)}$  是和频率无关的复数,  $H^{(1)}$  和频率呈线性关系,  $H^{(2)}$  和频率的平方有关。

对于NFC系统的信道来说, 这三项的作用与地位也是随系统的体制的不同而不同的。NFC



系统的载频为13.56M，所以可以不考虑 $H^{(1)}(f)$ 和 $H^{(2)}(f)$ ，既不考虑多径的影响，又由于信号的作用距离近，所以可将NFC系统的信道看作平坦衰落高斯白噪声信道。

5.3 系统仿真流程

系统仿真流程图如图5-1所示：

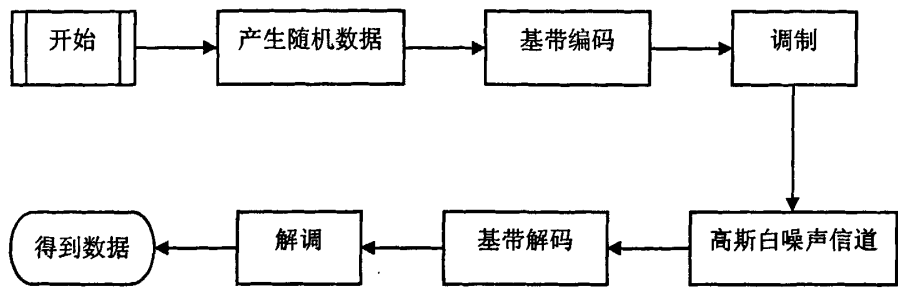


图5-1 系统仿真流程图

在系统仿真流程图中，数据的传输流程可描述为：在数据发射端，NFC设备将所要传送的命令和数据组合起来，形成需要传送的数据部分；对这部分数据按照适合NFC系统的格式进行基带编码，并加上帧头，形成NFC数据帧，数据帧的长度有几到十几个字节，然后将这一帧数据经调制后发送出去。发送后的数据通过无线信道传输。在接收端，另一个NFC设备对接收到的信号进行解调并整形后形成二进制信号；再将该信号经过基带解码，形成最后的数据。

5.4 高速率NFC系统性能仿真

高速率系统性能仿真按照前面的讨论分成两个部分：一个是采用EMCA-340标准的通信速率为847kbps的系统；另一个是采用第四章讨论方案的通信速率大于1Mbps的系统。这两种不同速率系统的性能仿真均采用上一小节的仿真流程。

5.4.1 通信速率等于847kbps

5.4.1.1 仿真模型

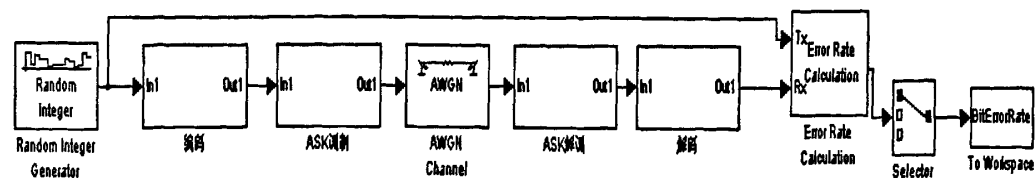


图5-2 通信速率为847kbps时系统性能仿真模型

从图5-2可以看出，原始数据与解码结果在Error Rate Calculation中进行比较，Error Rate Calculation产生的数据是一个三维向量，分别表示误码率、误码个数以及信号总数。可通

过一个Selector选择第一个变量，将其输入工作空间，并保存为变量BitErrorRate。最后通过程序得到信噪比误码率仿真结果图。

5.4.1.2 仿真过程

当高速率NFC系统的通信速率为847kbps时，在本文中仍然采用EMCA-340标准，即调制方式采用100%ASK调制，编码方式采用改进Miller码。为了对系统性能做出判断，还将对通信速率为106kbps的低速率NFC系统的系统性能进行仿真，最后对两个系统进行比较。

在本仿真中，编码，调制模块自行设计，100%ASK调制解调模块如图5-3所示：

1) 100%ASK调制

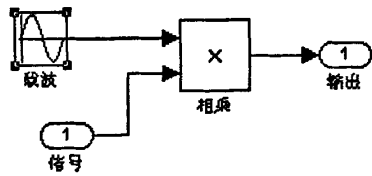


图5-3 100%调制子模块

这一模块完成信号的调制功能，方法是让原始信号与载波信号相乘，所得到的输出即为调制后的信号。

2) 100%ASK解调

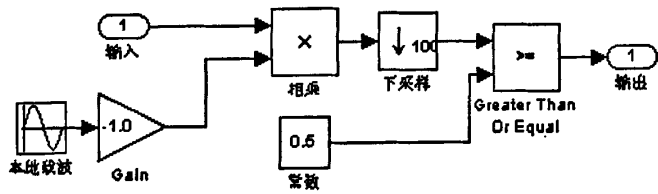


图5-4 100%ASK解调子模块

解调的方法是用一个本地载波信号的倒数与原始信号相乘再经下采样得到输出信号，由于输出的信号经过信道后产生了畸变，所以在解调后加入了整形功能，将所得到的信号编程二进制信号。

5.4.1.3 仿真结果与分析

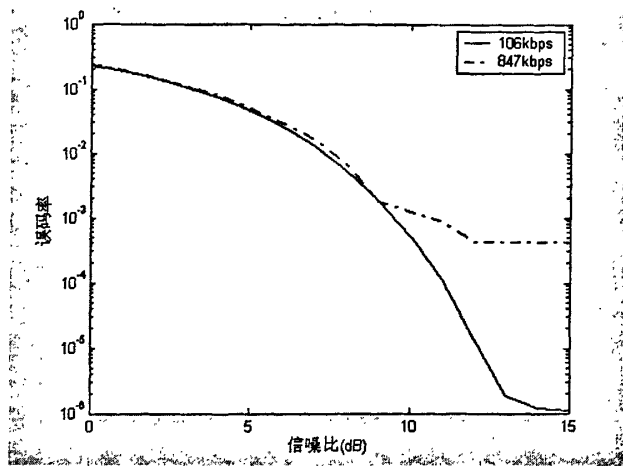


图5-5 847kbps与106kbps系统误码率比较

由图5-5的仿真结果看出，采用EMCA-340标准方案的通信速率为847kbps和106kbps的NFC系统，在信噪比较低时系统性能相差不多，在信噪比较高时，847kbps系统的误码率性能提升较慢，性能较106kbps要差不少。

## 5.4.2 通信速率大于1Mbps

### 5.4.2.1 仿真模型

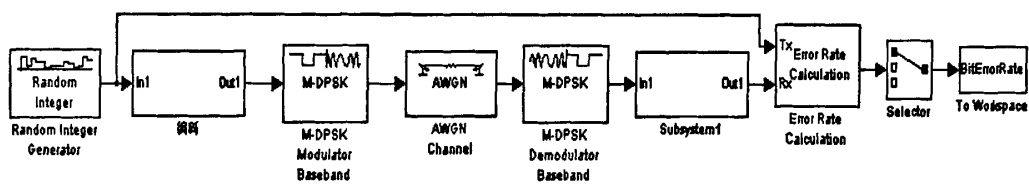


图5-6 通信速率大于1Mbps时NFC系统性能仿真模型

### 5.4.2.2 仿真过程

通信速率大于1Mbps时，根据EMCA-340标准，在本文中选取通信速率为1695kbps的情况进行仿真。速率大于1Mbps时，调制和编码采用第四章中讨论的方案，即调制采用MDPSK方式，在本章中采用4DPSK；同时对3种编码方式（改进Miller编码、NRZ编码、Manchester编码）进行仿真，以进一步比较它们在系统中的性能。

仿真过程与847kbps时的情况类似，采用相同的仿真流程。

### 5.4.2.3 仿真结果与分析

#### 1) NRZ编码

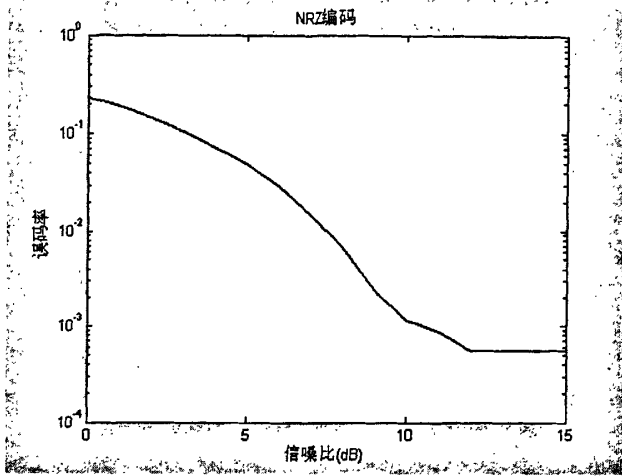


图5-7 采用NRZ编码时系统性能

2) Manchester编码

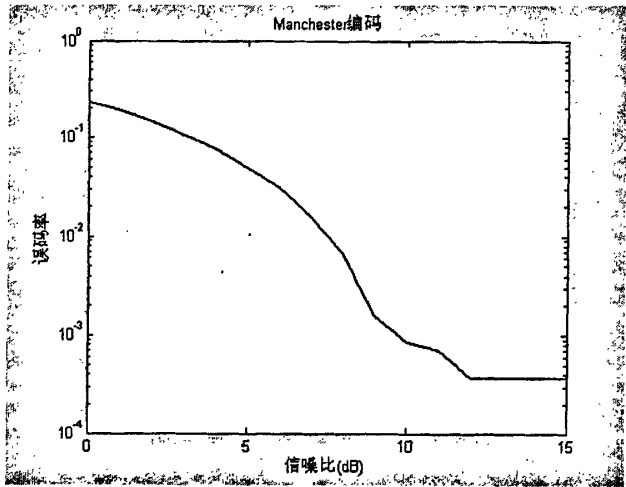


图5-8 采用Manchester编码时系统性能

3) 改进Miller编码

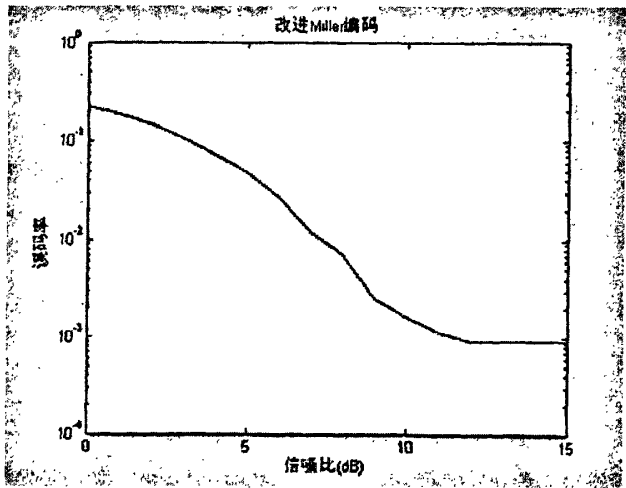


图5-9 采用改进Miller编码时系统性能

4) 三种编码方式比较

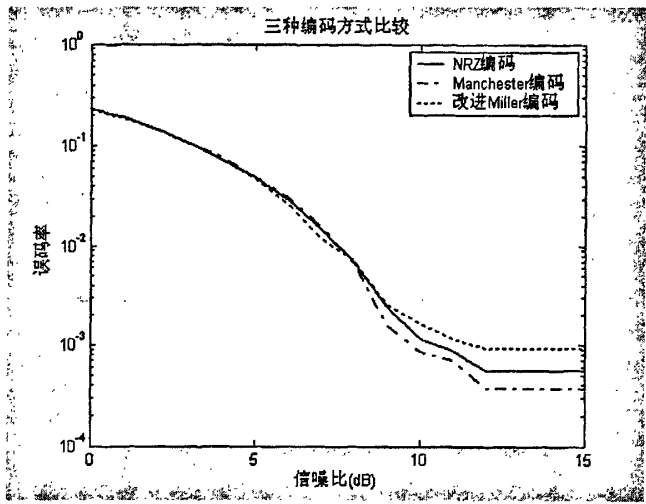


图5-10 三种编码方式系统性能比较

由图5-10的仿真结果可以看出，在通信速率大于1Mbps的情况下，采用第四章讨论的方案，分别采用三种编码方式进行系统仿真时，它们在信噪比较小情况下，系统误码率基本上差别不大；信噪比变大时，三种编码方式的系统误码性能出现差别，在较大信噪比的情况下，使用Manchester编码的系统性能最好，NRZ编码次之，改进Miller编码性能最差。

5.5 本章小结

本章主要工作是对第四章提出的高速率NFC系统模型的性能进行仿真，根据第四章的讨论，主要有两种速率进行了仿真，一种是通信速率大于424而小于1Mbps，另一种是通信速率大于1Mbps。通过仿真，能得到这样的结论：在大于424kbps的两类高速率NFC系统模型中，采用EMCA-340标准调制编码方案的通信速率为847kbps的系统的系统性能和采用第四章调制编码方案的通信速率大于1Mbps的NFC系统的系统性能大致相同，但在高信噪比的情况下，高速率NFC系统的系统性能提升减缓，比106kbps系统要差不少。由此证明第四章讨论的高速率NFC系统关键技术和本章提出的高速率NFC系统的系统模型是可行和有效的，只是在高信噪比情况下系统性能稍差。

## 第六章 总结

本文研究的课题是一个较新的课题，国内刚起步不久，具有极其广阔的发展前景，所以本文的工作是相当有意义的。

本文主要工作是在理解EMCA-340协议中低速率NFC系统标准的基础上，对高速率NFC系统实现的可能性进行理论研究。主要的工作放在高速率NFC系统实现的关键技术的理论研究上。在分析可用的调制和编码方式之后，提出了不包含射频前端和数据传输控制的高速率NFC系统模型，并从理论上对这种系统模型的进行了分析。

论文完成了高速率通信情况下，调制编码方式和工作模式的选定，新系统模型的建立与性能仿真。

本文的另一个重点是防碰撞算法的研究，提出了一种新的防碰撞算法，并进行了软件仿真，仿真结果表明防碰撞性能较原算法有了一定的提高。

由于时间所限，本文只是对NFC系统一部分情况进行理论性研究，没有考虑NFC系统的射频以及数据传输控制部分的情况，在实际应用中肯定是要把这些方面结合起来综合考虑，情况将更为复杂。

通过这次毕业论文过程，我熟悉了NFC技术相关的各个方面，为以后继续在相关领域进行研究开发打下了一个较好的基础。同时在这一过程中，培养了我进行科学研究的一些基本素质，为以后的科学研究工作建立了一个较好的开端。

当然论文研究的深度还有待于进一步的加深，整个仿真模型与实际NFC应用还有一定的距离，还没有做到针对特定的NFC高速率通信应用来进行仿真分析。今后应该在熟悉NFC系统各个方面的基础上，针对特定的高速率应用展开分析研究，这些都有待于在以后的工作中进一步完善。

## 参考文献

- [1] Vic Shen. “NFC规范与测试”. 电子与电脑, 2006,10: 96-99.
- [2] Standard ECMA-340 2<sup>nd</sup> Edition December / 2004.
- [3] Standard ECMA-356 1<sup>st</sup> Edition June / 2004.
- [4] Standard ECMA-352 1<sup>st</sup> Edition December / 2003.
- [5] Standard ECMA-362 2<sup>nd</sup> Edition December / 2004.
- [6] 魏金榕、李天明. “NFC移动支付未来的 ”杀手级“ 增值业务”. 通信世界, 2006,7 : 19-20.
- [7] 蒋华、孙强. “近距离无线通信技术标准解析”. 信息技术与标准化, 2006,5: 26-30.
- [8] 建山. “近距离无线通信(NFC)技术浅述”. 数码世界, 2005,11: 16.
- [9] 周慧芳. “NFC走向现实”. 通信世界, 2006,11: 22.
- [10] 崔瑞琳. “NFC带动移动电子支付发展”. 现代通信, 2006,9: 7-9.
- [11] Klaus Finkenzeller. 射频识别(RFID)技术. 电子工业出版社, 2001
- [12] 张利等、张振华、李永明、张春、王志华、陈弘毅. “Clock generator and OOK modulator for RFID application”. Zhang et al./J Zhejiang Univ. SCI 2005 6A(10):1051-1054.
- [13] ISO/ IEC 14443-3:2000(E) Identification cards-Contactless integrated circuit ( s) cards-Proximity cards-Part1 : Physical characteristics.
- [14] ISO/ IEC 14443-3:2000(E) Identification cards-Contactless integrated circuit ( s) cards-Proximity cards-Part2 : Radio frequency power and signal interface.
- [15] ISO/ IEC 14443-3:2000(E) Identification cards-Contactless integrated circuit ( s) cards-Proximity cards-Part3 : Initialization and anticollision.
- [16] 陈香、张思东、薛小平. “RFID防碰撞技术的研究”. 金卡工程, 2005,9: 34-37.
- [17] 胡建云、李强、闵昊. “时隙ALOHA法在RFID系统防碰撞问题中的应用”. 应用科学学报. 2005,9,23(5): 489-492.
- [18] 吴春华、陈军. “动态ALOHA法在解决RFID反碰撞问题中的应用”. 电子器件, 2003,6,26(2): 173-176.
- [19] 陈志云、王豪才、龚关珍、姜义初. “RFID电子标签A型卡的防碰撞模块设计, 微控制与嵌入式系统”. 2005,8: 5-8.
- [20] Kenichi Okada, Tomohiro Yamada, Takumi Uezono, Kazuya Masti, Akio Oki, Yasuhiro

Horiike. "Near Field Communication Chip using PIM for Bio MEMS Sensors". 2004 IEEE Asia-Pacific Conference on Advanced System Integrated Circuits(AP-ASIC2004)I/Aug. 4-5,2004:16-19.

[21] ZMJ EWSK A. Evaluating wireless technologies in mobile payments a customer centric approach C . Sydney : IEEE Proceedings of ICMB'05. 2005.

[22] 张士兵、包志华、徐晨. "近距离无线通信及其关键技术". 电视技术, 2006,6: 62-64.

[23] 樊昌信, 通信原理. 国防工业出版社, 2001

[24] 曹志刚, 现代通信原理. 清华大学出版社, 2003.7

[25] 邓华, MATLAB通信仿真及应用实例详解, 人民邮电出版社, 2003



## 致谢

在写致谢时，我想感谢很多很多给我帮助的人，这里首先要感谢的是我的恩师包志华教授，在研究生期间，师从包老师，我觉的非常幸运，包老师渊博的知识，严谨的治学态度，使我受益匪浅。在三年的学习中，包老师对我的指导和教诲，教给了我知识和技能的同时，也培养了我独立思考和科研的能力，教给我为人处世和做人的道理。

同时还要感谢张士兵、徐晨老师以及三年中一起学习的我的同学游玉俊，他们在我的设计过程中，给予我很多帮助和鼓励，使得我的毕业论文得以顺利进行。

## 研究生期间的研究成果

1. 《近距离无线通信防碰撞技术研究》 信息技术 2007年第1期 第一作者
2. 《ECMA-340标准介绍》 电信工程技术与标准化 2006年第5期 第三作者

## 附录

防碰撞算法模拟程序

```
#include <stdio.h>
#include <math.h>
#include <time.h>
#include <stdlib.h>
#define N 15                                //随机产生目标的个数:
struct data_tag
{
    int nfcidt;
    int serial;
};
unsigned int collision times;                //定义总的解决碰撞所需要的时序;
void main()
{
    int x, nfcid;
    int i, j, m;
    int stk_num, collision, data_nfcid;
    int logic, new_data;
    float a;
    struct data_stack
    {
        int nfcids;
        int xs;
    };
    struct data_stack stack[N];
    struct data_tag tag[N];
    int request(data_tag tag, long nfcid, int x);    //函数说明;
```

```

void makeup_next_nfcid(struct data_tag tag[N], int &nfcid, int &x);
int get_nfcid(data_tag tag[N], int nfcid);
void unselect(data_tag tag[N], int data_nfcid);
collision times=0;
randomize();                                //随机产生10组nfcid;
for(j=0; j<10; j)
{
    tag[0].nfcidt = random(1000)&0xff;        //保证产生10个nfcid值不同的应答器:
    tag[0].serial=0;
    for(i=1; i<N; i++)
    {
        do
        {
            new_data=random(1000)&0xff;
            logic=1;
            for(m=0; m<i; m++)
                if (newdata==tag[m].nfcidt) logic=0;
        }
        while (logic==0);
        tag[i].nfcidt=new_data;
        tag[i].serial=0;
    }
for(i=0; i<N; i++)
    printf("%0x", tag[i].nfcidt);            //依次输出产生的nfcid;
printf("\n");
epc=0x7fff; x=SIZE;                        //置初值;
stk_num=1;                                //压栈:
stack[stk_num].nfcids=nfcid;
stack[stk_num].xs=x;
collision_times++;
while (stk_num!=0)                          //栈不空时;

```

```

{
    nfcid=stack[stk_num].nfcids;           //取栈顶元素;
    x=stack[stk_num].xs;
    stk_num--;
    collision=request(tag, nfcid, x);
    if (collision==1)                       //发生碰撞?;
    {
        stk_num++;                         //当前nfcid压栈;
        stack[stk_num].nfcids=nfcid;
        stack[stk_num].xs=x;
        collision_times++;
        makeup_next_nfcid(tag, nfcid, x);   //形成下一次的nfcid值;
        stk_num++;                         //当前nfcid压栈;
        stack[stk_num].nfcids=nfcid;
        stack[stk_num].xs=x;
        collision_times++;
    }
    else
    {
        data_nfcid=get_nfcid(tag, nfcid);   //得到一个NFC设备的nfcid值;
        printf(" %x", data_nfcid);         //输出识别的nfcid;
        unselect(tag, data_nfcid);         //屏蔽;
    }
    printf("\n");
    collision_times/=10;
    printf("collision times=%d\n", collision_times); //输出结果
    a=10.0000/collision_times;
    printf("a=1.4f", a);
}

int request(data_tag tag[N], long nfcid, int x)
    //模拟阅读器询问，检测有无碰撞、有碰撞返回1;无碰撞返回0;

```

```

{
    int i, num;
    num=0;
    for(i=0; i<N; i++)
        if ((tag[i].nfcid<=epc)&&(tag[i].serial=0))
            num++;
    if (num<=0) printf("error1 ");
    if (num==1)
        return(0);
    else
        return(1);
} //函数makeup_next_nfcid()产生下一次询问所需的nfcid, 位置x;
//结果由nfcid, x带回;

void makeup_next_nfcid(struct data_tag tag[N], int &epc, int &x)
{
    int temp, templ, temp2, templ_cpy, temp_tag[N];
    int i, num, count;
    num=0;
    for(i=0; i<N+1; i++)
        if ((tag[i].nfcid<=epc)&&(tag[i].serials))
        {
            temp_tag[num]=tag[i].nfcid;
            num++;
        }
    tempt=temp_tag[0];
    for(i=1; i<num; i++)
        templ&=temp_tag[i];
    for(i=0; i<num; i++) //取反;
        temptag[i]=~temptag[i];
    temp2=temp_tag[0];
    for(i=1; i<num; i++)

```

```

temp2&=temp_tag[i];
temp=temp1|temp2;           // temp为temp1,temp2的或;
temp<<=SIZE-x;//左移SIZE-x位;
temp1<<=SIZE-x;
while ((temp<0)&&(x>0))
{
    temp1_cpy=temp1;
    temp1_cpy|=0x7FFF;
    count=SIZE-x;
    while (count>0)
    {
        temp1_cpy=1;
        temp1_cpy|=0x8000;
        count--;
    }
    nfcid&=temp1_cpy;        //形成nfcid的一位;
    temp<<=1;
    temp1 <<= 1;
    x--;
}
temp1_cpy=temp1;           //再将nfcid当前位0;
temp1_cpy|=0x7FFF;
count=SIZE-x;
while (count>0)
{
    temp1_cpy>>=1;
    temp1_cpy|=0x8000;
    count--;
}
nfcid&=temp1_cpy;
x--;

```

```
}

int get_nfcid(data_tag tag[N], int nfcid)           //识别出一个NFC设备;
{
    int i, sum=0;
    int temp;
    for (i=0; i<N; i++)
        if ((tag[i].nfcidt<=nfcid)&&(tag[i].serial==0))
        {
            temp=tag[i].nfcidt;
            sum++;
        }
    if (sum> 1)                                     //如果同时读出NFC设备个数大于1出错;
        printf("error\n");
    return(temp);
}

void unselect(data_tag tag[N], int data_nfcid)      //屏蔽掉一个目标;
{
    int i;
    for (i=0; i<N; i++)
        if ((tag[i].nfcidt==data_nfcid)&&(tag[i].serial==0))
            tag[i].serial=1;
}
```



# 高速率近距离通信（NFC）关键技术研究

作者：[曹平](#)  
学位授予单位：[南京邮电大学](#)

## 相似文献(0条)

本文链接：[http://d.g.wanfangdata.com.cn/Thesis\\_Y1412265.aspx](http://d.g.wanfangdata.com.cn/Thesis_Y1412265.aspx)

下载时间：2010年3月19日