
多频段低成本EPC标签识读者

摘要

通过与 Auto-ID 中心合作，ThingMagic LLC 已经开发了一个独特的多频段 RFID 标签识读者参考平台。这个识读者已被用于读取 Auto-ID 中心研制的符合 13.56MHz (HF) 和 902 - 928MHz (UHF) 频段规范的 EPC 标签。它的硬件体系包括一个对每个频段都通用模拟变频器，并附有无线电软件结构以便轻松的适应新的频率和协议。识读者的组态软件结构允许轻松扩展，同时通过嵌入式支持 SQL 查询的数据库引擎提供包括网络可配置性、固件动态刷新和一个 TCP/IP 识读者接口等多种特性在内的复杂网络性能的。这种设计为快速配置和内置升级路径提供了出色的可扩展性和柔性。

目 录

1. 前言.....	3
2. 设计概述.....	5
2. 1 设计理念.....	5
2. 2 基于 DSP 的结构.....	6
2. 3 设计要素.....	7
2. 3. 1 天线单元.....	7
2.3.2 UHF 和 HF 模拟信号链路	7
2.3.3. Bamboo Linux 平台	8
2.3.4. DSP 模块	8
3. 识读器接口.....	9
3. 1 UHF EPC 空中接口	9
3.2.HF EPC 空中接口	9
4. 硬件设计.....	10
4. 1 DSP 板设计	10
4. 2. 1 局部振荡器.....	13
4. 2. 2 发送链路.....	13
4. 2. 3 接收链路.....	14
4. 3 HF 波段模块	14
4. 3. 1 局部振荡器.....	15
4. 3. 2 传输放大器.....	15
4. 3. 3 接收链路.....	15
5. 软件设计.....	15
5. 1 通用软件结构.....	15
5. 1. 1 查询处理链路.....	15
5. 1. 2 多协议性能.....	18
5. 1. 3 识读器配置.....	18
5. 2 设备驱动器.....	18
5. 3 UHF 软件模块.....	20
5. 3. 1 命令结构.....	20
5. 3. 2 反碰撞算法.....	21
5. 3. 3 信号流和解调.....	23
5. 3. 4 Scroll 处理	23
5.3.5 Ping 处理	23
5. 4 HF 软件模块	23
6. 总结.....	24

1. 前言

ThingMagic LLC一直与Auto-ID中心及其成员合作来设计和规范一种基于Auto-ID中心开放标准理念的新型RFID识读器。这种识读器的结构是基于RFID识读器将来不仅仅是带有一系列端口的无线电收发器的现实基础之上的；而在Auto-ID中心看来，在RFID技术的帮助下，供应链管理要求RFID识读器在软硬件性能上有根本的改变。在Auto-ID中心看来，RFID识读器将被安装在每一个工厂、仓库货架、零售货架、入口及出口处以便在产品整个生命周期中的售前阶段中跟踪每个产品。大多数现有的识读器不能通过这些应用设备的可扩展性的严格检测。因此我们必须明确过去的RFID识读器与新一代的RFID识读器的区别以迎接这些供应链的重要挑战。

在过去的模式中，RFID识读器是一种孤立设备，它用无线电频段（RF）来识读标签并且将其ID字符串通过一个串行端口或者基本网络接口传到与公司的企业系统相连的临近一台PC上。它的上一代系统现在正以访问控制系统、简易仓库物流系统、收费以及其他应用形式来部署。这种“无声识读器”方案严重依赖于大量不可靠和不安全的，运行在民用级操作系统上的需要人工干预的PC机。从成本、可扩展性、维修费用、安装成本和能源消耗来看，这不能实现Auto-ID中心所设想的方案。

相反，将来的RFID识读器将成为大的分布式动态系统的一部分，在此系统中每个识读器对其所在地的标签管理负责。在这种真正的分布式系统中，RFID扮演了这样的角色：它成为相互“无声”的标签和非常“智能”的分布式信息系统之间的通道，这种分布式系统轮流访问使用像Savant这样分布式数据库系统的企业软件。这样，RFID识读器除了要承担数量不断增加的简单的询问器之外的责任，还要负责其所管辖的标签所有方面，通过标签的动态变化情况来反映供应链中所标签的物品的变动情况。本文所描述的识读器是被设计在一个大的网络环境中，它提供一个专门的带有RF空中接口的网关功能，这个空中接口所在的标签位于识读器的一面上，另一面上则具有TCP/IP网络接口的数据库服务器功能，以备成为分布式的数据统计和分析系统的一部分。

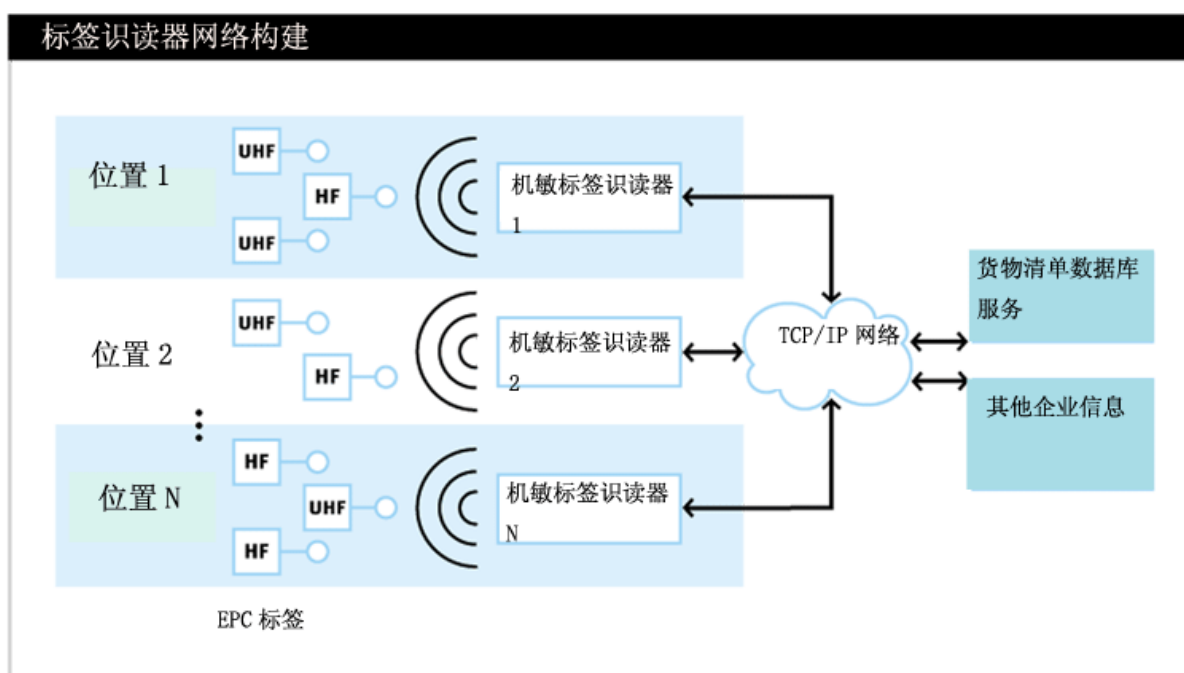


图1 标签识读器网络架构

对RFID标签的不同的应用的需要使得对标签/识读器系统的RF频道也提出了极为不同的要求。就像遵守EPC规范说明的系统，即使逻辑上的数据结构和要求一定，因为天线的基本物理特性和无线电传播是在不同的频段中，对空中接口的要求也随着应用的变化而趋于变化。我们相信不同的频段，例如13.56MHz和915MHz，以及联结器技术（近场的电磁感应或电容式联结器以及远场辐射）提供不同的利益和功能上的协调，因此我们期望在RFID市场至少有两种不同频段的使用会在可预见的将来具有现实意义。因此我们的识读器是围绕简单模拟频段模块的思想而设计的，这种模块能在同一识读器内支持不同频率的可融合相配的。

此外，在供应链中应用RFID系统需要全球性的标准化技术平台以便标签能在全世界流通并且在不同的国家中都可以独立于特定的规则要求进行识别。在这种努力下产生的RFID识读器就需要能够识读多种工作频率的标签。另外，由于在供应链管理和仓库管理中所期望的识读器的寿命是十到二十年，而标签的有效寿命可能仅仅是几天到几个周，所以识读器需要能便捷的重新配置软件以支持日后所用到的频段和协议。由于新的标签不断的涌入供应链而旧的标签不断地被淘汰，因此我们需要留有经常性的系统革新的余地来替代现有识读器基础设备以支持新的标签技术。

简言之，我们确信在供应链中所应用的“智能”RFID识读器应具备以下几点：

- 能在多种频率下工作
- 符合因特网协议
- 是客户/服务器系统的一部分
- 而且合并类似事件的行为

本文中所描述的参考实施是迈向基于开放标准的多频率、多协议识读器的第一步。它的设计目的是与AUTO-ID中心和它在产品电子码(EPC)上的伙伴所共同开发的新一代RFID标签进行通讯。这个新的标签大家庭中最初的两个成员工作在915MHZ和13.56MHZ。进一步的在世界范围内的分配给RFID技术的其他频段的标签的研制和生产正在进行。这些标签的逻辑数据结构在整个家族中被设置为可调的；与此同时，他们都被设计来存储一个64或96位的产品电子码(EPC)。识读器扮演了一个翻译者的角色，使单个标签的特定空中接口对软件基础设施是透明的。

该项新设计的主要特征包括：

- 借助于组件式模拟信号链路的频率灵活性，
- 借助于基于DSP软件无线电设计的协议灵活性，
- 借助于基于Linux的使用以太网的符合TCP/IP网络协议的标准，
- 由于系统的大部分组件在不同的频率和协议下可重复使用而使成本低廉，
- 通过Intranet或Internet对固件升级来使网络协议升级
- 被动标签、半被动反向散射标签和主动标签之间的互操作性。

将这些不同协议的硬件要求最小化并通过软件模块将不同协议之间的差异提炼出去，通过以上两种方法使这种识读器在硬件成本和软件灵活性方面超过了大多数其他多频段解决方案。使这个设计在同等开放的基础上成为 Auto-ID 中心的标签说明书，允许终端用户和卖方从中获利是 ThingMagic LLC 和 Auto-ID 中心合作的目的。

2. 设计概述

2.1 设计理念

在设计这个识读器的时候我们遵循了几个主要的原则：

1. 协议和频率的抽象

包括载体频率和通讯协议的物理标签技术应该从与识读器对话的网络设施抽象出来。识读器在隐藏具体物理传输机制的同时传达了EPC协议的逻辑特性。这要求识读器能支持大多数通用RFID频率以便于一个识读器就能识读所有的标签。

2. 可扩展性

识读者必须按照衡量空中接口和网络接口的通讯数量的方式来设计。

3. 配置方便

网络接口必须能容易安装并尽可能的使用现有的网络基础设施。因此识读者的主要接口使用的是基于TCP/IP网络的以太网方式。

4. 易于维护

识读者必须能通过企业的信息系统人员而不需要专门的维护人员就可方便的进行远程维护。

4. 对网络设施的一个比喻

识读者必须像现在广为人所知道的网络设施（例如，路由器，网络文件服务器等）一样能大规模的分布、配置并达到互操作。有这些基本原则作为指导，我们相信识读者及所识读标签的规模会成为我们当今所熟知的各公司通用的因特网和内联网的自然延伸（图1）。注意在这种结构中不需要PC作中间媒介，所有与标签识读者的交互都建立在对等网络服务器的基础上进行处理。

2. 2 基于DSP的结构

对混合模拟和数字系统（例如RFID识读者）的设计的主要观点是希望设备的数字部分能遵守摩尔定律以快速增加性能和降低成本，而设备模拟部分的价格和性能可以长期相对稳定。因此识读者是围绕着强大的数字信号处理器（DSP）而设计的，它能用软件实现所有的调制、解调和反碰撞搜索功能的数字信号处理器（DSP）。通过数字化提供信号链路的大多数要素和相关的功能，DSP提供了极大的灵活性，因为当固件可被方便的调整、改变和升级时，硬件就可在不同的协议之间保持不变。

识读者的模拟硬件由连接到DSP系统上的每个频段的独立频段模块组成。这些频段模块是简单的上行/下行转换器把信号从操作频率转换到基频，这样DSP系统中的模数转换器就能够将操作信号数字化以便于数字解调。

识读者是由采用系统通用标准的四块各自独立的印刷电路（PC）板组成的：

Bamboo-DSP

Bamboo-DSP板支持Bamboo Linux服务器、数字信号处理器（DSP）和模数转换器（ADC）。

13. 56 MHz频段模块

13. 56MHz频段模块支持13. 56MHz信号的模拟处理链。模板接收来自Bamboo-DSP板的数字控制数据，连接到13. 56MHz的天线端口并且提供下限数字信号并输出到Bamboo-DSP板的供解码的模拟信号。这个模块能以软件所适用的13. 553 - 13. 567MHz的频率传送高达7瓦的射频功率。

900 MHz 频段模块

900 MHz频段模块支持900 MHz信号的模拟处理链。模板接收来自Bamboo-DSP模板的数字控制信号，连接到900 MHz的天线端口并且提供给Bamboo-DSP模板两种模拟信号用来进行模数转换和解码。

900 MHz模块在软件中可调（902-928兆赫兹），实际上采用的是跳频模式，用软件可将电源调到+28dBm。

前控制板模块

前控制PC板连接到Bamboo-DSP板上。前控制板通过四个按钮接收用户对设置和测试的输入并且通过五个LED显示提供用户反馈。另外，前控制模块在自我检测（POST）代码处对电源设有蜂鸣器。

2. 3 设计要素

2. 3. 1 天线单元

识读器的天线单元需要支持多频段，有两到四个不同的频率范围。除了谐振频率不同，不同的频段要在不同的国家间操作，因此需要采用不同的耦合技术。

目前的天线单元使用了由UHF频段的微嵌板和HF频段的线圈组成的平面几何体。这两部分通过两根独立的射频电缆连到识读器上。最初期望用一根电缆连接着两个频段可能会比较合理，但是实验证明这两个频段需要不同的天线放置方式，因为13. 65MHz的识读范围远小于915MHz的识读范围。因此用不同的电缆连接每个频段提供了最大的灵活性。

2. 3. 2 UHF和HF模拟信号链路

我们所提出识读器的模拟链路设计旨在尽可能提高灵活性。在当前的应用中我们提供给HF和UHF信号链路两个独立的频段模块。两个板上的传输信号都是通过可编程逻辑振荡器（PLO）模块产生并且由来自DSP/CPLD单元的控制线调节。接收到的信号使用IQ解调成两个信号混合到基波。每个频道都被数字化成一个独立的12位的模数转换器信道并送交DSP进行解调。

识读器所支持频率的数量可以通过在识读器的设计中添加更多的硬件模块来扩展。

2.3.3. Bamboo Linux平台

ThingMagic先前在内部进行设计和开发的Bamboo嵌入式的Linux服务器是一个由 Motorola68000处理器、MC68EZ328、8 兆的DRAM、4 兆的闪存和完整的网络连接功能（通过SMSC LAN91C96以太网接口芯片提供）组成的价格低廉的通用的Linux服务器。Bamboo运行在Linux操作系统的一个端口上，Linux系统是一个免费的资源开放的操作系统，它提供非常完整的网络连接功能并且简单申请就可以使用免费工具。Bamboo已经被设计成为了一个硬件/软件核心，能轻松的定制成客户指定的嵌入式和手持式应用。

Bamboo的网络堆栈十分迅速并且与包括IP、TCP、UDP、HTTP和其他协议在内的所有的标准因特网协议相兼容。Bamboo的存储结构使得连到DSP的共享存储器的快速并行接口可以对DSP固件进行通讯和载入操作。这些特征使快速DSP解决底层的强大计算协议和标签处理时“分配劳动”成为可能；当Bamboo运行在较低的16 MHz时，在需要时收集数据、开始标签识读和提供网络层的标签数据库信息。

除了实时数据处理，Bamboo还支持能够进行基于HTML标签识读器的查询和配置接口的网络服务器。查询接口能使用户使用文后所述的识读器查询语言进行查询。基于网络的配置系统可以让用户配置网络和识读器的射频环境。

2.3.4. DSP模块

DSP模块负责识读器中的实时信号处理任务。这个模块接收和发送数字信号给频段模块进行发给和来自标签的数据的调制和解调。处于对DSP会遵守摩尔定律的期望，我们选择了一个十分廉价的TI TMS320VC5410 DSP芯片，据制造商说每容量十美元。这个选择主要是考虑价格理由，其他的DSP当然也适合，只是价格比5410贵得多。此种DSP在识读器应用中提供了合理的性能（160MHz时钟频率，典型值为160MIPS）。调制和解调的计算要求并不像可能用于标签系统中使用的相对简单的AM、FSK和PSK调制那么严格。

DSP本身并不含有非易失性存储器。DSP的固件储存在Bamboo Linux处理器所访问的闪存中。我们的设计利用了TI DSP的主机接口（HPI）来提供给DSP共享存储器接口。在引导时间里，DSP的固件从Bamboo的文件系统中通过共享存储器下载到DSP中。因此DSP程序出现在Bamboo已经载入了固件并且引导了之后。因为有了这种结构，DSP固件能在域中很容易的升级并且整个系统只需要一个闪存芯片，所以价格低廉。

3. 识读器接口

3.1 UHF EPC空中接口

915MHz/868MHz空中接口是遵照Auto-ID中心联合其赞助机构提议的EPC UHF协议规范设计的。

UHF EPC接口的目的是提供一个开放标准的接口使不同制造商生产相互之间可以通讯的设备。协议规范说明本身是开放的，而具体的物理实现和制造技术由制造商各自保留。因此在此中投入的公司可依靠开发专门的制造工艺并使用特别的设备方案来保持竞争优势。

当前在识读器上运行的UHF EPC协议的版本具体在文档“极低成本（VLC）射频识别（RFID）系统操作规范第一部分、Class1设备、9.1版本”中给出说明。915 MHz EPC空中接口按照第15部分的FCC规则设计。EMC的全面评估和第15部分的认证还没有在原型机上进行。

3.2. HF EPC空中接口

13.56 MHz空中接口是遵照Auto-ID 中心及其赞助机构所联合提出的EPC HF协议规范设计的。

HF标签协议的设计与UHF的设计的功能要求大致相同，但不同的物理边界条件和规则要求有不同的空中接口和逻辑层协议。例如，反碰撞算法的应用是以通用带宽的识读器到标签和标签到识读器通讯作为指导。由于两个频段的这些参数十分不同，所以为HF规范选用了十分不同的反碰撞算法。

当前在识读器上运行的HF EPC协议的版本具体在“HF EPC标签规范的修订草稿”中有说明。13.56 MHz的EPC空中接口的设计遵守13.56 MHz频段的欧洲电磁辐射标准（CE标准）。EMC全面评估和第15部分的认证还没有在原型机上进行实验。

为了使介于终端软件和识读器之间的可升级的客户-服务器基础设施成为可能，一个开放的可升级的协议（SQL）运行在标签识读器的标签数据库服务器端的TCP/IP接口之间。协议用来连接Savant分布式数据库软件，这个数据库软件可以与各种不同的EPC标签识读器通信。

识读器和网络之间的接口被标签数据库服务器所规范，服务器采用企业数据库系统中广泛应用的多种可扩展的结构化查询语言（SQL）。这种语言源于ANSI标准文件X3.135-1989和X3.168-1989。识读器SQL特别为工作和协议在多种频率的不同标签而设计，是SQL的一个扩展。SQL服务器负责标签数据库（存放在DSP/Bamboo共享存储器中）和网络上的一个主机之间的通讯。主机能进行结构化查询以获得标签的任意子集，查询可以依据许多要素，如标签ID、协议、天线等。服务器将请求标签并且仅返回那些与结

构查询相匹配的内容。服务器可以在返回的标签带有具体时间的投票模式下工作，也可以在每隔任意整数个毫秒向标签数据库报告的流模式下工作。人们也可以以可读的格式，如标准telnet（具体是在RFC 0854中）来查询标签数据库。

查询协议详见文件（3）。

4. 硬件设计

4.1 DSP板设计

图2所示的DSP模块由三个主要的电路板组成：

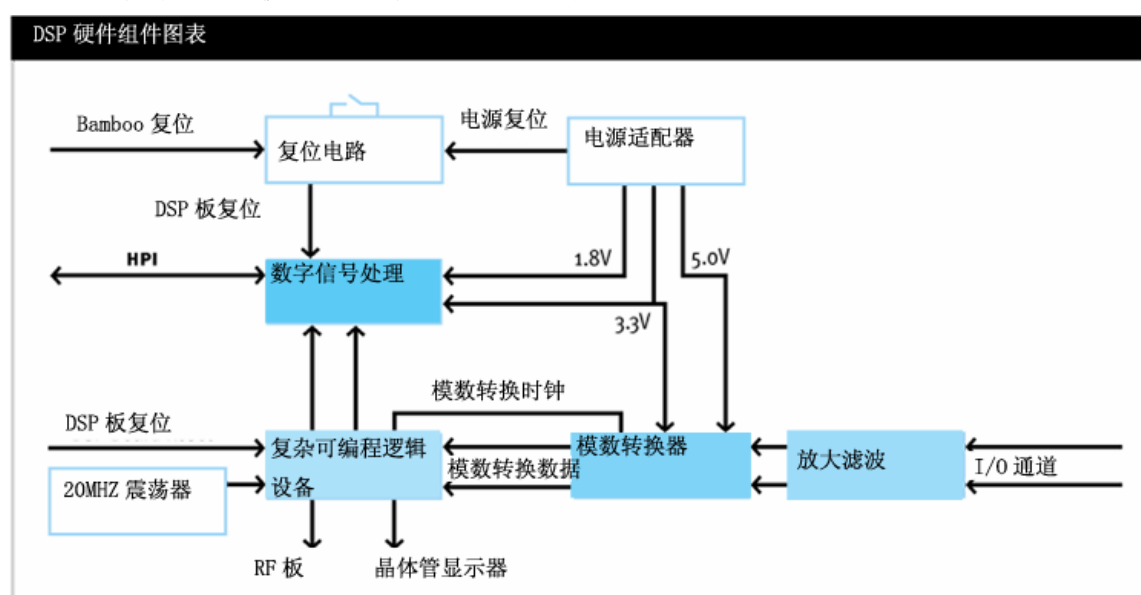


图2 DSP模块——硬件电路板示意图

1. DSP（数字信号处理）板包括由德州（德克萨斯）仪器生产的TMS320VC5410系列DSP以及诸如时钟振荡器和开电复位电路之类的相关支持电路。
2. CPLD（复杂可编程逻辑器件）使用的来自Xilinx公司 XC95144XL系列的CPLD。
3. 模拟电路板使用德州仪器的ADS2807系列12位模数转换器和采用OPA2681高速运算放大器的放大/滤波电路。

DSP系统使用来自德州仪器的160 MHz的TMS320VC5410A系列DSP。这种芯片包括64K字节的供程序和数据存储的片上随机存储器。其核心电压是1.8伏而I/O供电电压是3.3伏。DSP通过将MP/MC针接地而运行在“微处理器模式”，使其可以从Board RAM板引导。Bamboo的DSP接口经过DSP的主机接口（HPI）。处理器可以通过手动开关、电源循环或在Bamboo的软件控制下复位。复位后，DSP等待Bamboo通过HPI将DSP操作固件下载到片上存储器中。

HPI 也通过共享存储器接口在 Bamboo 和 DSP 之间传送执行时间数据。数据传输的细节在软件章节详细介绍。

DSP 到 CPLD 芯片 (Xilinx 公司出品的 XC95144XL) 的接口是经由 DSP 的两个多通道缓冲串行端口 (McBSP)。McBSP 配置清单以表格的形式列于表 1。CPLD 和 ADC 之间的接口采用的是通用并行接口。CPLD 的一个功能是锁存流向内部寄存器两个 12 位的模数转换数据流并且把这个数据连续地从 McBSP 0 移出然后由 McBSP 1 移送到 DSP。

MCBSP	方向 (相对于 DSP)	比特率	数据
0	Tx	24 兆比特 / 秒	CPLD 寄存器设置
0	Rx	24 兆比特 / 秒	输入通道模数转换数据
1	Tx	1 兆比特 / 秒	射频传送位序列
1	Rx	24 兆比特 / 秒	输出通道模数转换通道

表1 DSP—McBSP配置

CPLD 有多种内部 8 位寄存器，DSP 能用 McBSP 向这些寄存器写入数据。寄存器的地址和功能在表 2 列出。DSP 使用 McBSP Tx 线向 CPLD 寄存器写数据。McBSP 数据是按照 12 位的地址数据混合格式：前 4 位指出 CPLD 的寄存器的地址 (LSB 先)，剩下的 8 位是指出锁入 CPLD 寄存器的数据 (LSB 先)。

寄存器地址	寄存器名	注释
0	RESERVE_REG	保留
1	VERSION_REG	保存 CPLD 码的版本
2	ADC_DIV_REG	ADC 时钟的划分要素 (CPLD 时钟根据此要素被划分并且分给了 ADC)
3	RF_GPIO1_REG	射频板的数字 I/O 线 (GPIO 线 0 — 7)

4	RF_GIO2_REG	射频板的数字I/O线（GPIO线 8 — 1 5）
5	LED_REG_REG	LED状态（位 0 =LED1, 位 1= LED2, 位2= LED3）
6	SWRESET_REG	保留
7	DIAG_REG	当在这个寄存器中设置了LSB时，ADC的输出不给DSP，而是产生里一个反复测试模式给CPLD并且给了DSP
8	DDS_REG0	保留
9	DDS_REG1	保留
10	DDS_REG2	保留
11	DDS_REG3	保留
12	DDS_REG4	保留
13	MBSP_PASS_REG	当位 0 — 7 设定，相应GPIO线 8 — 1 5 分别反映了McBSP的Tx 1 线的逻辑状态

表2 寄存器地址与功能

CPLD的内部时钟是由一个频率为20 MHz的晶体振荡器驱动的。这 2 0 MHz频率被分频，供ADC进行模拟数据的采样。分频系数可以通过DSP写入CPLD的一个寄存器来规定。CPLD也包括一个寄存器，用来控制前面板上四个系统LED的状态。

DSP板包含了一个在ADC转换之前对两个输入模拟信号整形的小型模拟电路板。输入信号以2的分频系数被滤波放大。一个2.5伏的偏移量被加入到信号中以匹配ADC的动态范围。由于输入信号动态范围被设定为1.5伏—3.5伏，ADC使用5伏的模拟供电电源。信号通过截止频率为450kHz的低通滤波器，取样频率1.67 MHz。

4.2 UHF波段模块

UHF波段模块是在FCC的部分15.247规定下为跳频操作而设计的疏导902/928MHz到基波的转换模块。这些规则指定了使用至少50个频道的频率跳跃系统所使用的最大的输出功率为一瓦，在给定频率下的最大停留时间是400Ms. 因此频段模块受到了PLL锁定时间和

接收器发送接收恢复时间的限制。在频道效用和费用之间进行有意识的协调，由于其价格低廉我们选择了单一的综合设计；即使综合锁定时间会导致“死时间”效应，即在信道转换期间识读范围被封锁。

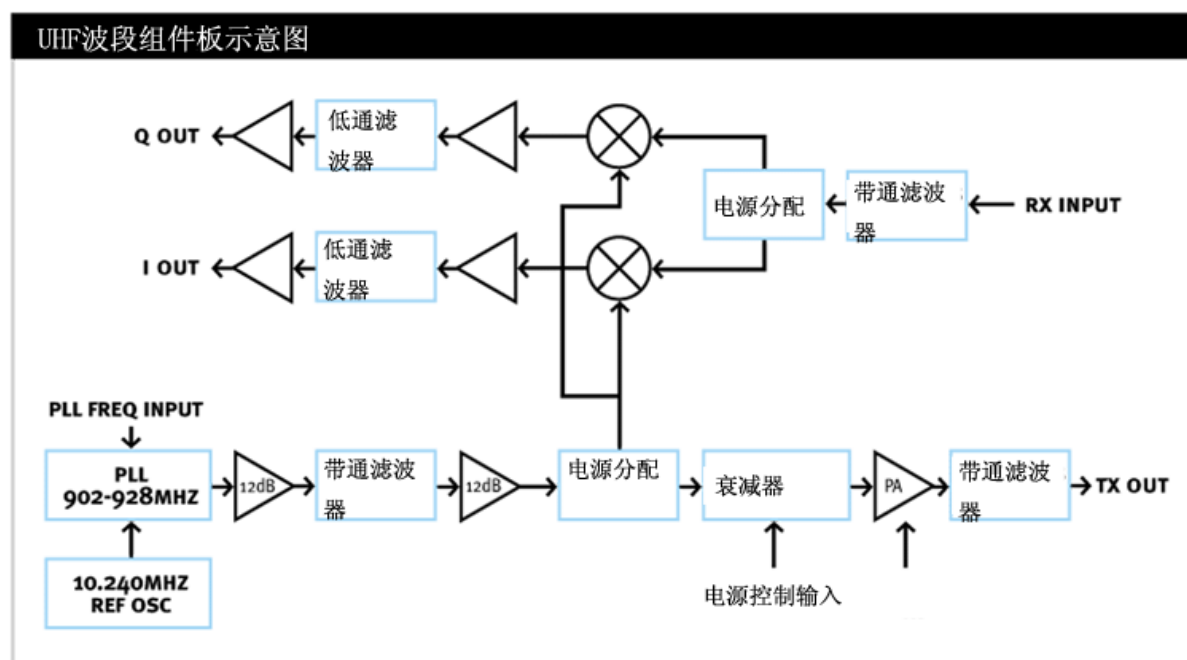


图4 UHF波段组件板示意图

4. 2. 1 局部振荡器

工作频率是由一个锁相环合成器模块（Z-Comm公司的PSN0930A）结合VCO和国家半导体所产的LMX2316 PLL IC产生的。这个廉价的模块产生带有限定在10KHz在 -100dBc/Hz 的相位噪声的+3dBm输出能量。重要的谐波能量在VCO的输出端口出现。为了增加VCO的载入隔离，在VCO和第一级MMIC（微电路ERA-3SM）放大器之间使用了一个6dB的衰减器。放大器适用由标准的L/R网络提供的+12V电源。放大器在这一点上的输出功率大约是+8dBm。放大器的输出通过一个中心频率为915MHz的两极瓷单块带通滤波器以去除二次谐波和其他乱真输出。二级MMIC放大器和3dB功率分配器把局部振荡器信号分成两路，一路发送一路接收。

4. 2. 2 发送链路

传送信号包含一个Hittite Microwave三位数字控制的阶梯衰减器（为了电源控制）和射频 Micro Devices GSM/AMPS GaAs功率放大器集成电路，并伴有抑制谐波和乱真输出的二级瓷制单块带通滤波器。此链路能够以高达+28dBm在915MHz传送信号。传送器可以通过功率放大器的功率控制输入进行振幅调节，这是在900MHz链的CPLD的数字控制

下完成的。当闭环功率控制一开始就被设计到以前的原型机中时，因为凭借合理代价获得充足的功率控制带宽相当困难与复杂，从而现在这种开环设计对应用的要求已经足够。

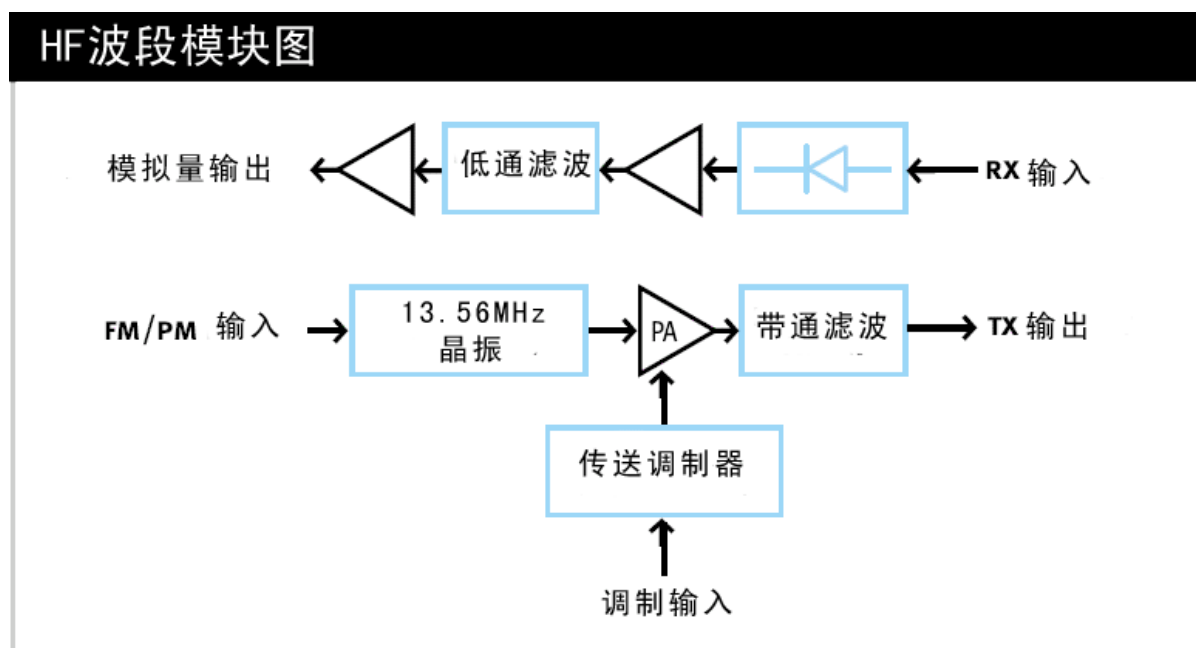
4. 2. 3 接收链路

接受链路的主要部分是在DSP中软件模块上完成的。因此模拟接受部分很简单。输入的RF信号通过谐振带通滤波器滤波之后分成两路信号进行正交（IQ）解调。这个信号供给两个微电路双平衡混频器，他们的局部振荡器信号由分配产生并且在Q路径产生相位滞后于接收的LO路90度的信号。没有使用首末射频预放大是因为大信号处理能力比接收噪声图象更重要。

I和Q频道的输出通过一个截止频率为450KHz的四极Bessel 抗失真滤波器放大滤波。这两个信号然后应用到Bamboo-DSP板上的模拟和数字转换器。更进一步的信号处理以数字形式完成，这是最通用的处理方法。

4. 3 HF波段模块

设计HF波段模块的目的是为了接收感应耦合信号、载入来自EPC标签的调制信号。它在由软件控制的13.553-13.567MHZ频率之间工作并能产生最高达7W的多种传送功率。与UHF波段模块类似，接收部分的信号处理任务主要由DSP中的软件来完成。



4. 3. 1 局部振荡器

由于13.56MHz波段仅有14KHz带宽，因此可以采用不同的晶振。在这种设计中变容二极管“带动”晶振，名义上的晶振频率是13.560MHz, 在调节范围的极限大约有14KHz的调节可在12位数模转换器的控制下完成。如果有要求的话，这种性能主要用于传送调制期间的频谱输出控制。

4. 3. 2 传输放大器

传输功率放大器被设计运行在Class E 级模式附近的开关模式下。放大器的设计的主体是一个廉价的逻辑层驱动MOSFET，IRL510门极驱动是由HCOMS逻辑集成电路提供，此集成电路驱动一对高速高增益双级晶体管。门极是以50%的工作循环驱动的，而传输功率通过来自12位DAC的其他部分驱动的线性放大器的流出电压调制。这样传输功率可以动态的和独立的以微小的和下降的调制间隔调节。输出通过正规的连续共鸣的网络滤波并且通过相同的网络对从内部12.5欧姆的目标下载阻抗到50欧姆的输出阻抗配对。

4. 3. 3 接收链路

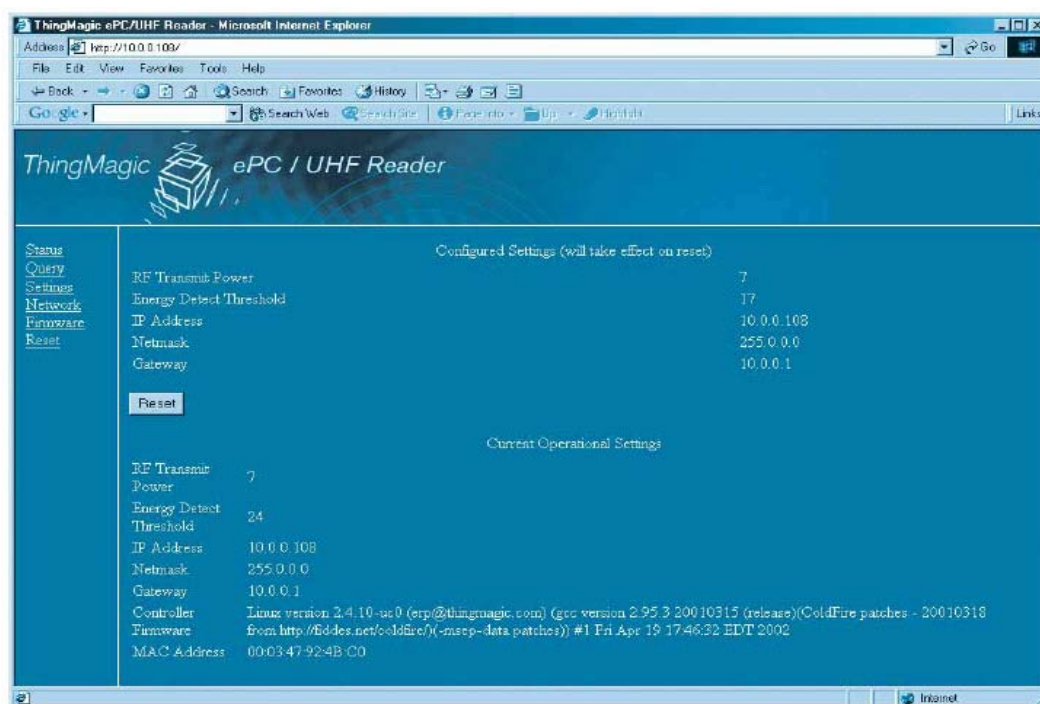
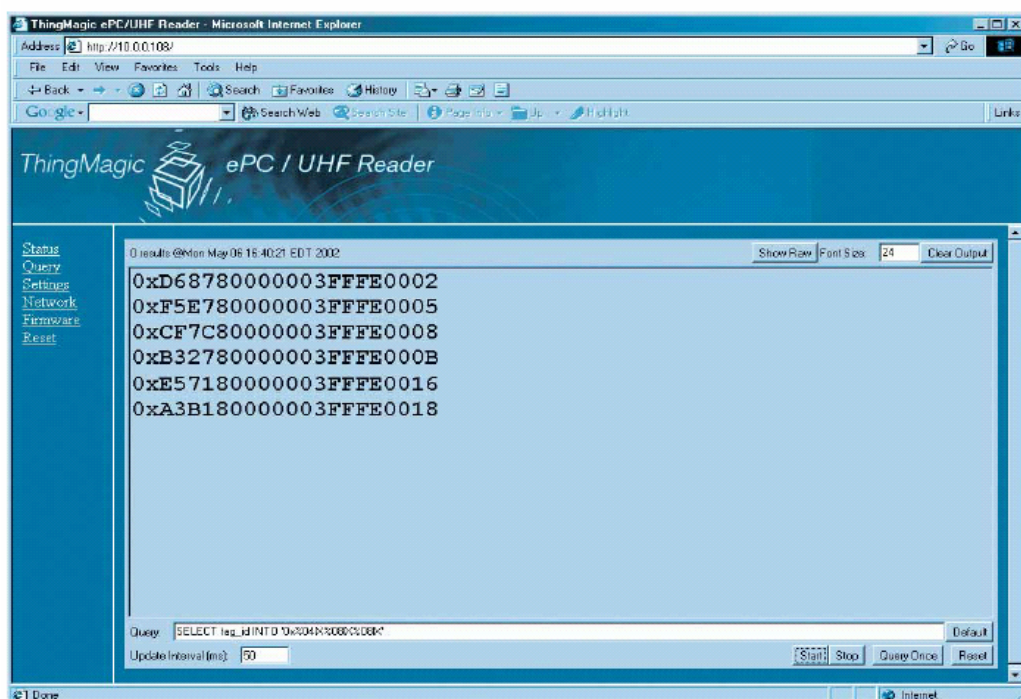
13.56MHz接收链路是基于标准的电压加倍AM检测电路，后面是13.56MHz的跳陷电路。与使用在915MHz完全相同的抗失真滤波器在Bamboo-DSP板的放大和模数转换之前使用。除了这个输出，一个未使用的opamp部分被用作精密度测量器来提供一个供CPLD检测位层解调选择的下限数字输出。

5. 软件设计

5. 1 通用软件结构

5. 1. 1 查询处理链路

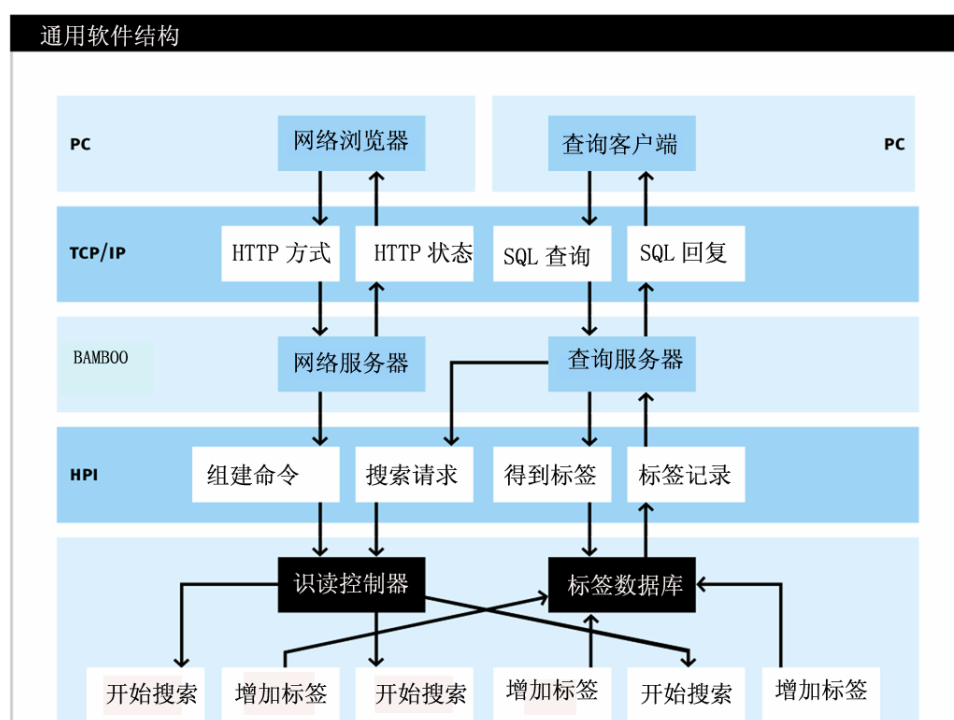
标签识读是专门由客户端软件或与已连接到TCP/IP网络的用户进行初始化。查询客户或者是嵌入在识读器支持的基于浏览器的Java查询界面（图6和图7），或者是它本身就是像Savant这样的高层数据处理结构的一部分。然而，也可以通过远程登录简单的由用户手动查询。协议是通过标准的因特网TCP流连接所携带的像SQL一样的语言（如下所示）。



查询服务器是位于Bamboo上的基于Linux的嵌入式通用处理器。它从TCP连接的另一端接收SQL请求并且把他们解释成一系列DSP可以理解的动作。一旦从DSP上接收到结果，SQL服务器就把结果送到网络客户端。Bamboo和DSP之间通过一种共享存储器机制进行通讯，这种机制经由DSP的HPI端口进行物理连接。

查询服务器命令DSP运行一个结构化的标签搜索，这种搜索基于SQL查询中所包含的参数。在DSP内部，顶层控制软件要求从压缩了标签协议细节的协议模块中搜索。协议模块和设备驱动器相连，操纵DSP硬件向标签发送和接收无线电信号。

当接收到标签响应信号时，协议模块把这些数据存储在标签识读器上运行的所有协议模块所共享的标签数据库中。完成搜索之后，SQL服务器从标签数据库中读取内容。标签识读结果会被核对并打包成SQL格式的答复信号返回到客户端。



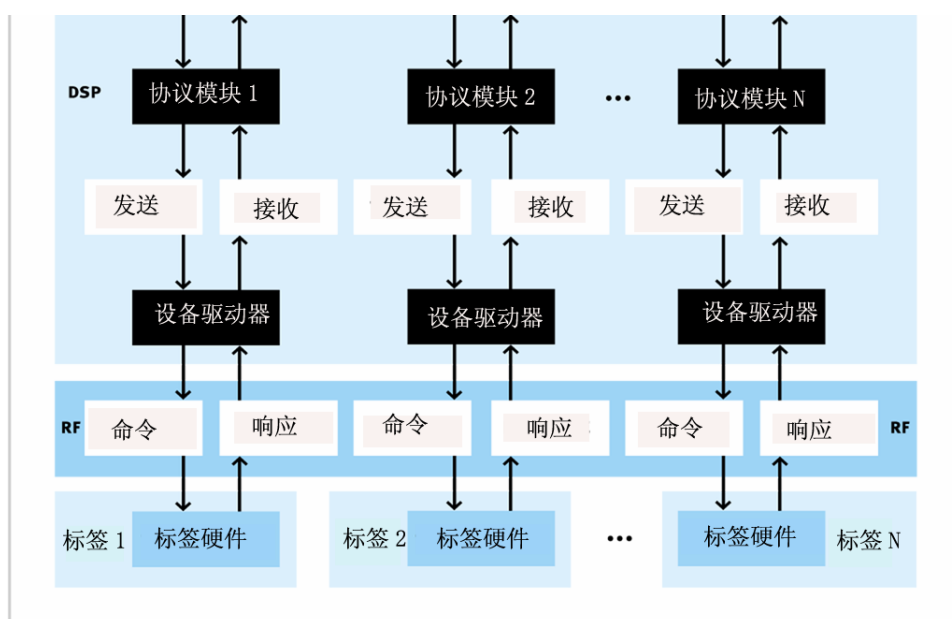


图7 识读器的软件结构概图

5. 1. 2 多协议性能

在对识读器软件系统功能进行抽象概括的基础上,为支持额外增加的协议而对系统进行的改动对于协议组件和它们的驱动设备来说是极其有限的。为更多的协议添加支持包括在对系统的高层进行较少的改动之中。在客户层,提供给用户或软件基础设施更多的以新协议为目的的选择;但网络接口仍保持不变。与之相似的,在Bamboo层,处理了新的协议,但是基本的软件结构保持不变。

标签数据库通过协议ID(一个“魔力数字”)字段特别的标签记录与特别的协议相连。查询服务器按用户要求完成具体标签的协议ID与客户之间的通讯。

5. 1. 3 识读器配置

Bamboo支持网络服务器提供给识读器配置安装的接口。使用标准的HTML页和表格,网络服务器报告状态和允许的参数配置,包括电源层和像IP地址这样的网络配置。

5. 2 设备驱动器

DSP固件到硬件之间的接口可以抽象成一套设备驱动器。如图8所示,设备驱动器把上层固件和协议模块与底层硬件接口区分开。每个RF模块的发送(TX)和接收(RX)链路以及其他的硬件功能(像LED 前面板显示的那样)都由设备驱动器提供。

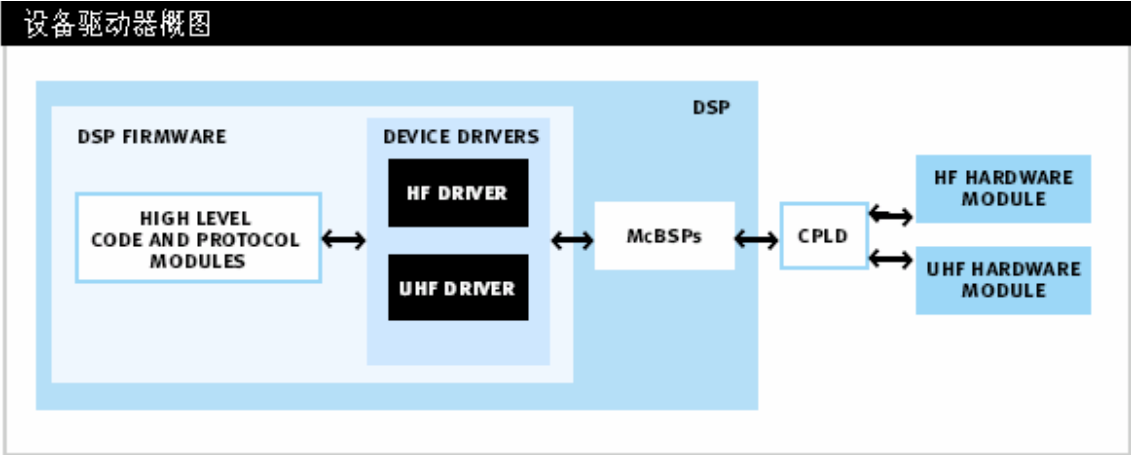


图8 设备驱动概图

设备驱动器码把对设备的功能命令转换成硬件操作来执行所要求的功能。设备驱动器通过管理片上DSP外围电路（如，串行端口和DMA控制器）和外部硬件设备低层零件将硬件接口抽象出来。硬件接口的细节将在第4章论述。RF设备驱动器的设计目标是提供以不依赖某种协议的方式访问硬件，以便使硬件所支持的全部协议模块可以在一小部分设备驱动器上运行。设备驱动器应用程序接口（API）由一系列用于从设备写或读数据或构造设备参数的C语言呼叫功能组成。API与POSIX 文件 I/O接口极其相似，也对这些过程使用写，读，和ioctl功能。设备驱动器通过“打开（open）”命令激活并且用“关闭（Close）”命令将其释放。

从设备读和写的确切含义取决于设备的性能，在某些情况下可能是非法操作（例如，写入RX设备或从TX设备读取）。一些设备（如LED前显示板设备）不提供读写功能。这些设备完全由ioctl功能控制。

UHF TX和RX的配置功能表	
UHF TX设备	
IOCTL 功能	作用
set_passthru	当passthru设置为1时，写到设备中的位是通过射频，就像开关键控芯片一样，以Ioctl功能中设定的芯片传输率传输。当路经设置为0时，射频输出稳定在开或关状态就像具体通过设置射频状态一样并且写入到设备中的数据被忽略。

set_rate	设置写到设备中的位射频芯片传输率
set_RF_state	设置射频场为开或关。这个状态只用在路径设置为0时。
freq_hop	用在跳频
set_RF_power	设置射频输出场能
UHF RX设备	
IOCTL功能	作用
set_rate	为设备读操作的样本采集设置取样率

写入TX驱动器数据特别会引起数据在输出场调制。从RX驱动器读取到的数据是用来自ADC的采样值装进输入缓冲区。通常这些采样值将表示需要进行信号处理的部分已解调数据流。操作流程如图9所述。

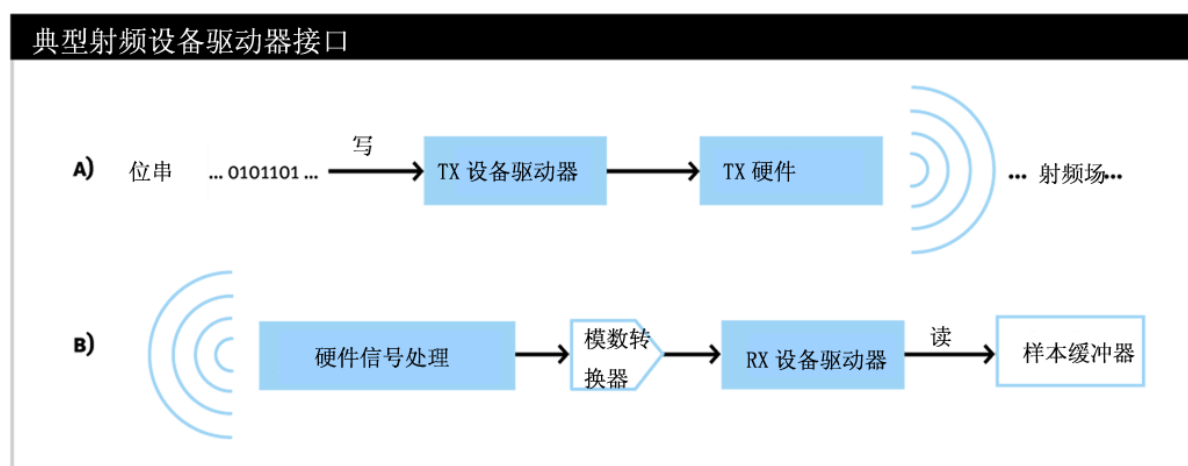


图9 典型射频设备驱动器

因为不同的设备一起分享硬件资源（例如，HF和UHF TX设备通过相同的硬件串行端口发送数据），因此某种形式的资源管理是必要的。在这个系统中高层固件管理资源避免发生碰撞。一般说来，这通过一次仅激活一个设备驱动器来完成。对于多个设备驱动器必须同时被激活的情况（例如，发送和接收的时间调节），应详细说明安全联结设备驱动器功能要求。

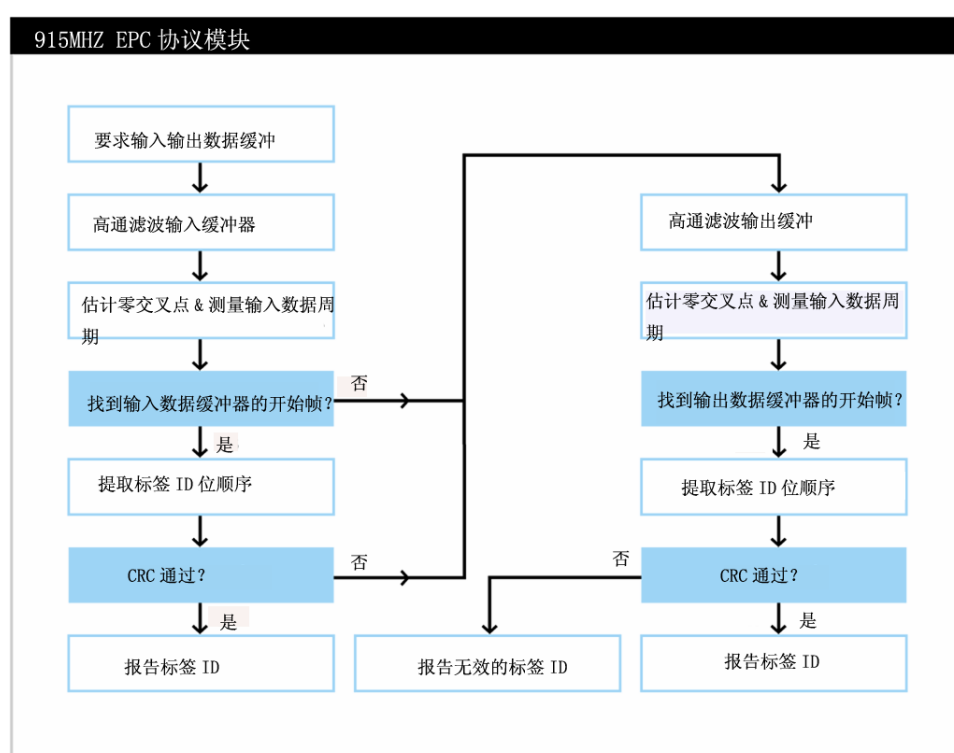
5.3 UHF 软件模块

5.3.1 命令结构

UHF软件模块执行在整个DSP中。它支持如下要求：

1. AC_search_start（反碰撞（简称AC）搜索启动）初始化反碰撞搜索。通过在标签数据库中注册来报告电磁场中的所有标签。由Bamboo查询数据库。

2. AC_search_stop命令终止了AC_start命令初始化的AC搜索。
 3. AC_search_step命令一次进行AC搜索一个标签查询命令的一步。
 4. Ping是使DSP处理Ping 0和Ping 1命令，这个命令作用于场中能响应的标签。
 5. scroll_start使DSP可连续的发出Global Scroll命令。任何找到的标签都向标签数据库报告。
 6. scroll_stop命令终止由scroll_start命令所启动的scroll。
 7. set_ping_threshold命令设置对ping命令的标签响应进行探测的电源下限。
- 不同命令和标签响应的详细情况在（1）中给出说明。



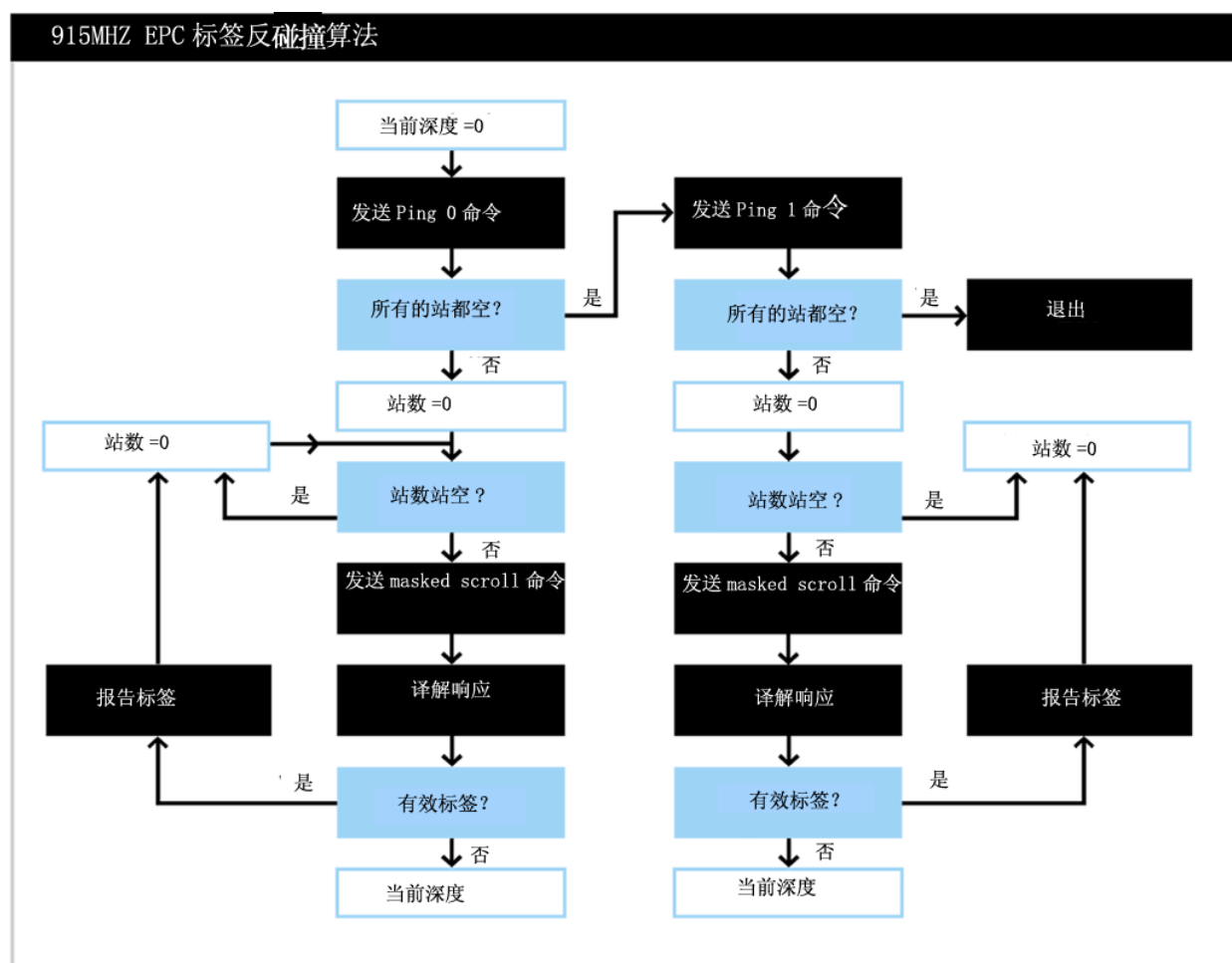
5. 3. 2 反碰撞算法

UHF EPC标签可响应三个命令：Ping, Masked Scroll和Global Scroll。。如果不止一个标签，Global Scroll命令将使工作场内的所有标签同时响应、引起碰撞。因此有必要进行一个反碰撞的搜索（AC搜索），通过系统的使用Ping和Masked Scroll命令每次可仅使某几个标签响应。

对场内多标签进行解码的AC算法是通过Ping和Masked Scroll命令的结合完成的。按照深度优先的方法中沿着八进制树搜索。表11展示了AC搜索对变量current_depth设置为零的AC搜索流程图。这种变量表明了正在查询中的标签ID的位长和搜索树的当前深度。current_depth为零表明正在查询2到4位。current_depth为1表明正在被查询5到7位等

等。

AC搜索通过发出一个Ping 0命令开始。作为对Ping 0命令的响应，带有至少一个最高零位的标签将会响应。响应被八个站（bin）中的一个接收。接收标签响应的站数据表明了标签ID接下来的三位。例如，如果发出Ping 0命令并且站3接收了响应，那么标签ID的前四位：0 011。然后Masked Scroll命令处理所有已对标签响应记录的站。



Masked Scroll命令仅在搜索中利用了标签ID固定位。标签对整个标签ID 响应。一旦对标签ID 解码失败，就可以断定不止一个标签响应并且有必要通过发出更多位的Ping命令进一步区分标签ID。在这种情况下变量current_depth增加。如果一个有效的标签ID被解码了，那么标签ID被输入数据库并且剩余的含有Ping响应的站通过Masked Scroll命令探查。

当变量current_depth增加时，发出在位mask中带有额外三位的Ping命令。这个过程不断重复，直到在一个给定current_depth没有标签响应或者current_depth达到标签ID最大值才结束。在完成了Ping 0树之后，发出Ping 1命令并且Ping 1树的搜索方式同Ping

0树。

5.3.3 信号流和解调

发到标签的每个命令都有一个具体如(1)所述的唯一的命令位码，DSP基于发送的命令码和命令参数对位序格式化。位序包括标签所要求的、使命令有效的正确的CRC值。组成的位序经由DMA和MCBSP从DSP发送到CPLD。CPLD使数据轮流通过调制输入到RF板。

在返回通道中反变换RF信号由ADC进行数字化处理并且通过CPLD送给DSP。数据有12位的动态范围。DSP进行信号滤波，然后进一步处理数据抽取数据位。

5.3.4 Scroll处理

标签通过发送编码在载波频率中的标签ID对Scroll命令做出响应。由于标签锁住识读器信号的位周期的变化很大。位周期也可以在标签发出响应的同时变换。实际的位周期是在搜索预先已知的标签发出的位格式的过程中由DSP决定。一旦位周期已定，就通过把接收到的信号和位1、位0所期望的格式相配对标签ID进行解码。在这个过程中一直实时地估计位周期以计算位周期变动。信号处理步骤具体在图10中。

一旦接收到所需的位数，标签ID就会通过检查CRC生效。一旦CRC检查通过，标签ID就会被报送数据库。

5.3.5 Ping处理

标签对ping命令的响应是通过发送相应的ping站中的8位数据完成的(站1到8)。DSP软件模块通过和参考帧比较估计出每帧的能量。如果信号能量超过了无响应参考信号中能量与可调门限的乘积，那么我们就可以得出结论目前站中至少有一个标签。

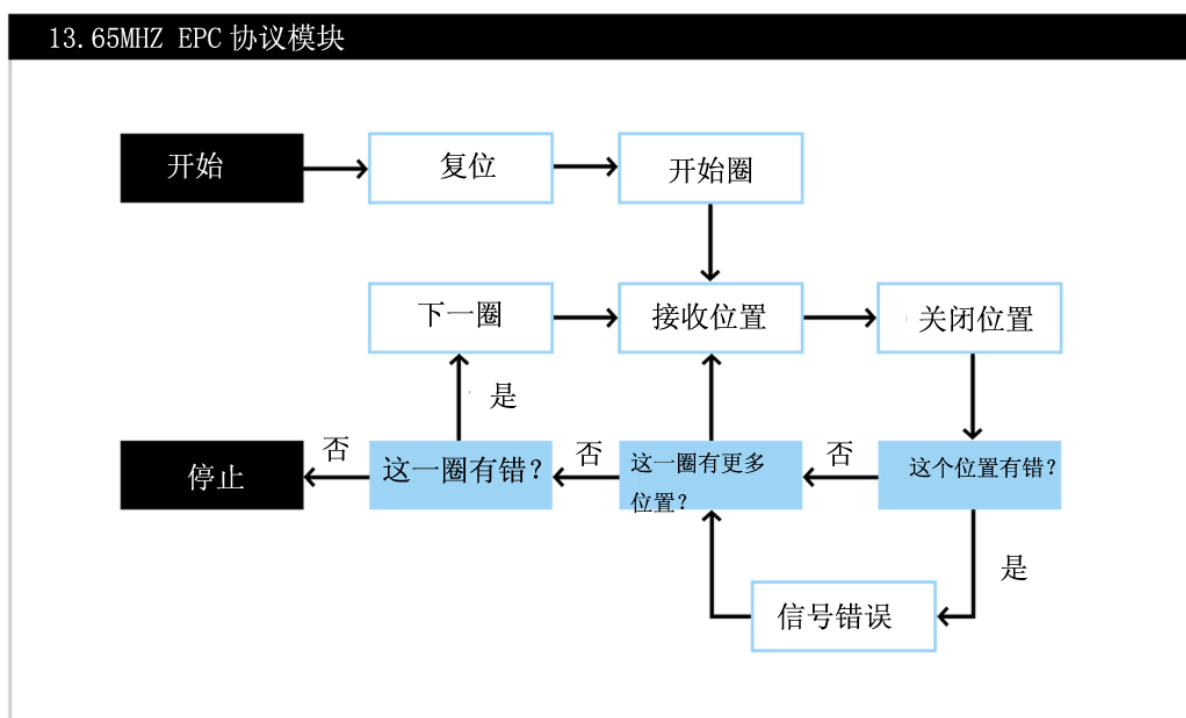
5.4 HF软件模块

与EPC UHF协议不同，EPC HF协议及其反碰撞方案是基于伪随机时隙的观点，例如标签在不同的时间响应，这样可以避免碰撞。EPC HF协议的基本事务就是“时隙(slot)”，简单的说，这就是期望在一个时间间隙中只有单个标签响应。时隙合成组就叫做“圈(round)”，每一个圈都有固定数目的时隙，这在圈开始的时候会声明。标签搜索是由一系列圈组成的，一个接一个，直到所有可利用的标签都接收了为止。

在搜索循环开始之前，Reset命令使所有的标签都处在一个已知的开始状态。随之搜索就以一个Begin Round的命令开始。

搜索过程的内圈只处理一个时隙(slot)。首先，识读器了解时隙的持续性和试着将来自接收信号的标签响应解码。这一过程处理完之后，通过发送Close Slot信号关闭时

隙。



此外,如果识读器认为前一个时隙有标签响应但却不能正确对其进行解码则发出错误信号。这个错误信号通知标签其响应没有被成功接收并告之稍候重复其响应。(在下一圈中)

在当前圈的内环重复每一个时隙(在圈开始之前预定的数目应该选择理论上减少标签碰撞并将未用时隙的数量降到最低)。

当圈完成时,识读器根据圈中是否有碰撞(解码错误)决定下一圈是否有必要进行。碰撞暗示了至少有一个标签发生了响应,但是不能被识读器成功收到。在这种情况下, next round命令使新一圈开始并且也开始循环。

6. 总结

这篇文章上下贯穿,提出了多频段EPC标签识读器的概念。它进一步概括了基本的设计原理以及实现这样一个识读器的可供参考的选择因素,最后详细说明了实施细则。除了这份白皮书之外,在Auto-ID中心的许可下也可以使用一系列硬件图解和固件资源代码。

这些文献和这份白皮书共同构建了EPC识读器参考设计模型,一个熟练的工程师可以根据这个设计模型再加工并最终生产出可实际工作的识读器,我们也可以用这个参考设

计模型启发另一个EPC识读器的设计。

在接下来的几个月中将进行识读器的系统性能评测。我们将检测不同天线设计和频率下的识读范围和识读方式、不同几何形状所带来的识读率的改变，反碰撞搜索的数量和现实环境中识读的可靠性。

双频段识读器提供了一个直接并且是同时对HF和UHF RFID技术进行比较的绝妙机会。有后端软件接口和数字识读器基础设施就可以比较前端模拟电路和空中接口的行为区别。作者坚信这种测试将会肯定供应链上部署RFID会存在众多频段的需求，从而肯定了会有多频段RFID识读器的需求。