AN1204

Microchip MiWi™ P2P 无线协议

作者： *Yifeng Yang*
Microchip Technology Inc.

简介

对无线通信的应用需求不断增长。

它的优点在于可降低成本且易于实现。无线通信不需要布线和其他硬件，以及相关的安装成本。在一些即使有可能布线，但仍有一定难度的区域，可以采用无线通信。

自从 IEEE 在 2003 年发布无线个人局域网 （Wireless Personal Area Network， WPAN）规范（IEEE 802.15.4™）以来，事实上这一规范已经成为低数据速率 WPAN（LR-WPAN）的行业标准。此规范适用于要求低功耗和低成本的低数据速率应用。

Microchip MiWi™ P2P 无线协议是 IEEE 802.15.4 的衍生协议，此无线协议采用了 Microchip 的 MRF24J40 2.4 GHz 收发器和带有串行外设接口（Serial Peripheral Interface，SPI）的Microchip 8位、16位或32位单片机。

通过易用的编程接口，该协议提供可靠直接的无线通信。该协议具有丰富的特性集，可以编入协议栈，也可以抽离于协议栈之外，以满足不同客户的需求——同时使协议栈的存储空间占用最小。

本应用笔记介绍了 Microchip 无线 （MiWi™）点对点（P2P）协议以及它与 IEEE 802.15.4 协议的区别。本文档详细介绍了其支持的特性以及如何实现这些特性。也介绍了简单的应用级数据结构和编程接口。

本应用笔记假设读者熟悉 C 编程语言。强烈建议读者在开始阅读本应用笔记或使用 MiWi P2P 无线协议之前，先阅读 IEEE 802.15.4 规范。

协议概述

MiWi P2P 协议修改了 IEEE 802.15.4 规范的介质访问控制（Media Access Control， MAC）层，增加了简化握手过程的命令。通过提供的辅助 MAC 命令，简化了链接断开和通道跳转操作。

不过，该协议未给出具体的应用决策，例如何时执行能量检测扫描或何时跳转通道。这些问题留给了应用开发人员。

协议特性

MiWi™ P2P 无线协议：

- 在 2.4 GHz 频带范围内提供 16 个通道 （采用 MRF24J40 收发器）
- 可工作在 Microchip PIC18、 PIC24、 dsPIC33 和 PIC32 平台上
- 支持 Microchip C18、 C30 和 C32 编译器
- 作为状态机的功能 （独立于 RTOS）
- 支持通信结束时器件处于休眠状态
- 使能能量检测 （Energy Detect， ED）扫描，以工作在噪声最小的通道
- 提供主动扫描，以检测已建立的连接
- 支持 IEEE 802.15.4 中定义的所有安全模式
- 使能频率捷变 （通道跳转）

协议注意事项

MiWi P2P 是 IEEE 802.15.4 的衍生协议，并支持点对点和星型拓扑结构。因为该协议没有路由机制，因此射频覆盖范围就是无线通信的范围。

由于不支持保证时隙 （Guaranteed Time Slot， GTS）和信标网络，因此通信两端的设备不能同时进入休眠状态。

如果应用需要无线路由而非 P2P 通信；或者需要与其他供应商的器件进行互操作；或者需要基于标准的解决方案以达到市场化，请参见 AN1066 《MiWi™ 无线网络协议栈》或 AN965 《Microchip ZigBee™ 协议栈》。

© 2009 Microchip Technology Inc.

IEEE 802.15.4™ 规范和 MIWI™ P2P 无线协议

在最初的 2003 版 IEEE 规范发布之后，又发布了 2006 修订版，其中阐明了一些问题。修订版称为 IEEE 802.15.4b 或 802.15.4-2006，修订版在 sub-GHz 频带内增加了两个 PHY 层定义，还修改了安全模块。

然而，目前市场上的大多数产品仍使用最初的 IEEE 802.15.4a 规范 —— 也称为 IEEE 802.15.4-2003 或 A 版。 Microchip MRF24J40 射频收发器支持该规范的 A 版本。

在本文档中，提及的 IEEE 802.15.4 规范指的就是规范的 A 版本。

PHY 层

MiWi P2P 协议栈仅使用了 IEEE 802.15.4 规范中丰富的 PHY 和 MAC 层定义的一部分。

该规范定义了三个 PHY 层，它们分别工作在以下频带下：868 MHz、915 MHz 和 2.4 GHz。MRF24J40 射频收发器工作在 2.4 GHz 的工业、科学和医疗（ISM）频带（可在全球范围内免费使用）。此频带有 16 个通道，最大的数据包长度为 127 个字节，其中包括了两字节的循环冗余校验（CRC）值。

理论上讲，IEEE 802.15.4 中 2.4 GHz ISM 的总带宽为 250 kbps。实际上，为了确保通信的可靠性，带宽通常为 20-30 kbps。

设备类型

根据 IEEE 的定义及设备在通信连接中的角色， MiWi P2P 协议对设备进行了分类（见表 1 和表 2）。

MiWi P2P 协议栈支持所有设备类型。

表 1： IEEE 802.15.4™ 设备类型 —— 根据功能分类

功能类型	电源	接收器空闲配置	数据接收方法
全功能设备（FFD）	主电源	打开	直接接收
精简功能设备（RFD）	电池	关闭	从相关设备查询

表 2： IEEE 802.15.4™ 设备类型 —— 根据角色分类

角色类型	功能类型	角色描述
个人局域网（PAN）协调器	FFD	首先启动此设备并等待连接。
终端设备	FFD 或 RFD	在 PAN 协调器开始建立连接之后，启动此设备。

© 2009 Microchip Technology Inc.

支持的拓扑结构

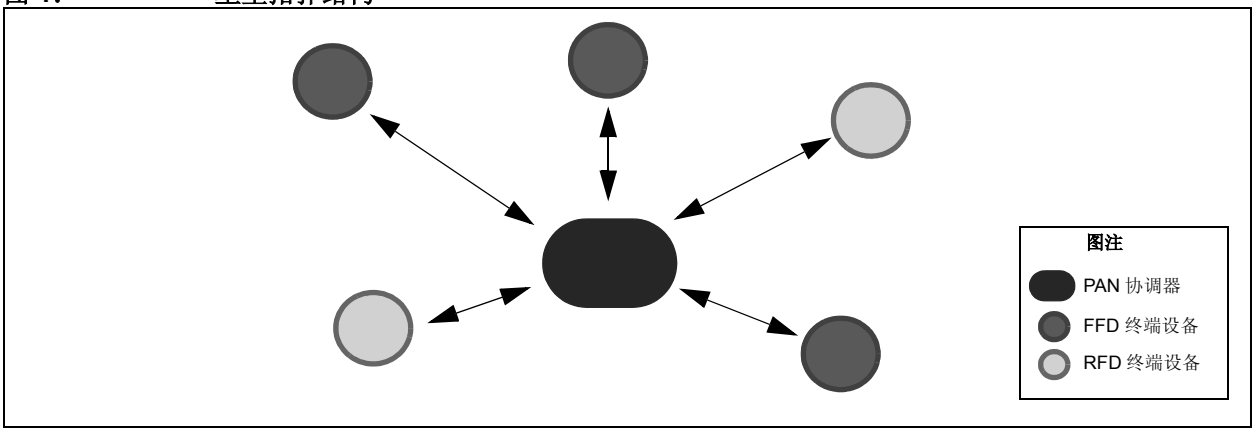
IEEE 802.15.4 和 MiWi P2P 协议栈支持两种拓扑结构：星型和点对点。

星型拓扑结构

图 1 为典型的星型拓扑结构。从设备角色的角度看，该拓扑结构有一个个人局域网（PAN）协调器，用于启动通信并能与其他设备建立连接。该拓扑结构有几个可加入通信的终端设备。终端设备只能与 PAN 协调器建立连接。

就功能类型而言，星型拓扑结构的 PAN 协调器是一个全功能设备（FFD）。射频一直处于打开状态的终端设备是全功能设备（FFD），在空闲时射频处于关闭状态的终端设备是精简功能设备（RFD）。无论终端设备为何种功能类型，终端设备只能与 PAN 协调器进行会话。

图 1：　　星型拓扑结构

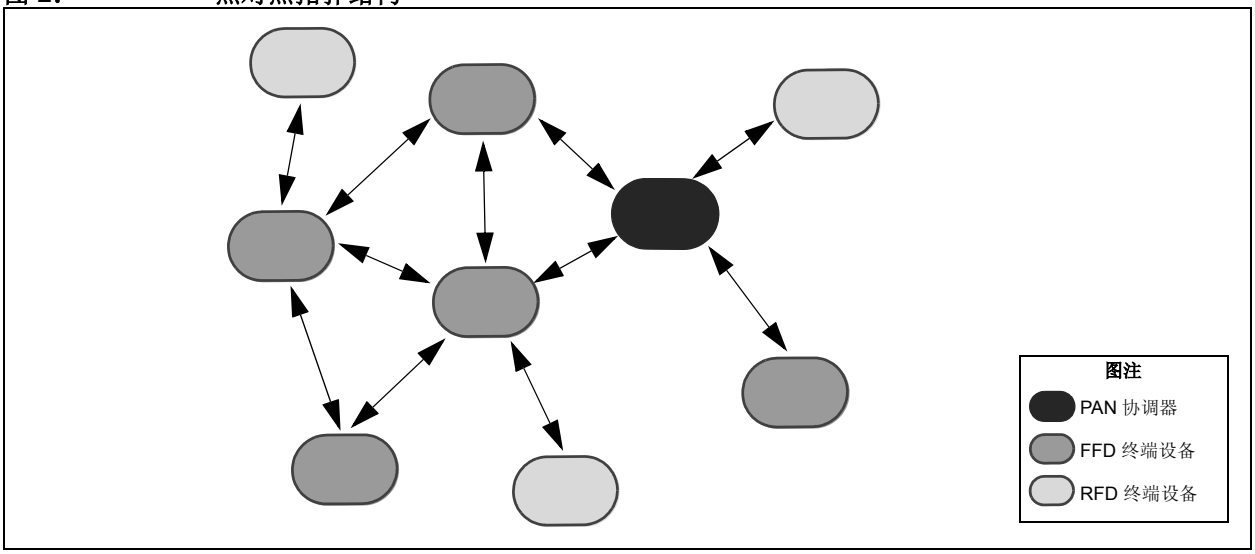


点对点（P2P）拓扑结构

图 2 为典型的 P2P 拓扑结构。从设备角色的角度来看，该拓扑结构也有一个用于启动通信的PAN协调器和多个终端设备。然而，当接入网络时，终端设备不必与 PAN 协调器建立连接。

就功能类型而言，PAN 协调器是一个 FFD，终端设备可为 FFD 或 RFD。但是，在此拓扑结构中，FFD 终端设备可有多个连接。但是，每个 RFD 终端设备只能与一个 FFD 相连，并且不能与其他 RFD 相连。

图 2：　　点对点拓扑结构



网络类型

IEEE 802.15.4 规范有两种网络类型：信标网络和无信标网络。

在信标网络中，设备只能在指定的时隙发送数据。PAN 协调器通过定期发出超帧（信标帧）来分配时隙。所有设备应与信标帧同步，且只能在指定的时隙发送数据。

在无信标网络中，只要能级 （噪声）低于预定义级别，所有设备在任何时间都能发送数据。

信标网络降低了所有设备的功耗，这是因为所有的设备都有机会定期关闭它们的射频。

无信标网络增加了 FFD 设备的功耗，这是因为它们必须使射频一直处于打开状态。但是，这类网络却降低了 RFD 设备的功耗，因为 RFD 不必频繁执行同步操作。

MiWi P2P 协议栈仅支持无信标网络。

网络寻址

IEEE 802.15.4 规范定义了两种寻址机制：

- 扩展组织惟一标识符 （EUI）或长地址 —— 每个设备所具有的全球惟一的八字节地址。

  前三个字节由推出产品的公司从 IEEE 购买。只要每个设备的完整 EUI 是全球惟一的，后五个字节便可由设备制造商指定。
- 短地址 —— 当此设备接入网路，由其父节点分配给设备的两字节地址。

  在网络内，短地址必须是惟一的。

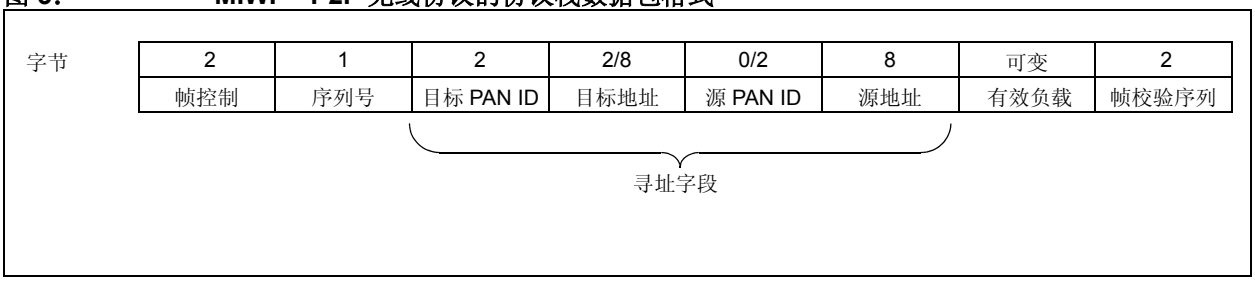
因为 MiWi P2P 协议栈仅支持单路程段 （one-hop）通信，因此它通过 EUI（或长地址）寻址来发送消息。短寻址仅在协议栈发送广播消息时使用。这是因为 IEEE 802.15.4 规范中没有预定义广播长地址。

© 2009 Microchip Technology Inc.

消息格式

MiWi P2P 协议栈的消息格式是 IEEE 802.15.4 规范的消息格式的子集。图 3 给出了协议栈的数据包格式及其字段。

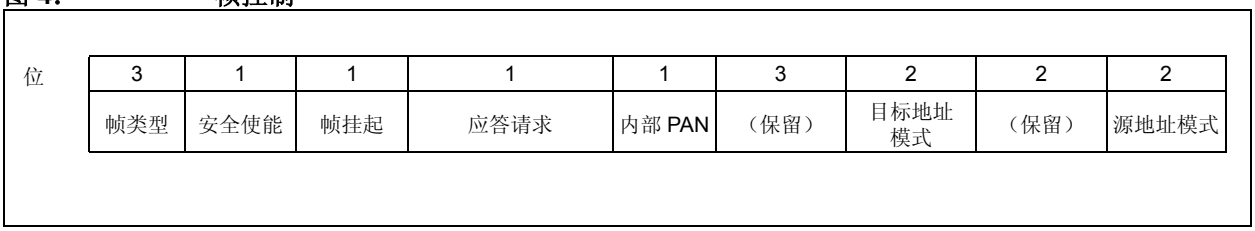
图 3：　　MIWI™ P2P 无线协议的协议栈数据包格式



帧控制

图 4 给出了两字节帧控制字段的格式。

图 4：　　帧控制



三位帧类型字段定义了数据包的类型，其值可为：

- 数据帧 = 001
- 应答 = 010
- 命令帧 = 011

但是，在此协议栈内，由于所有的应答数据包都由 MRF24J40 射频收发器内的硬件处理，所以没有使用应答帧。

安全使能位指示当前数据包是否加密。如果使用了加密，那么将需要一个额外的安全性报头，在后面关于安全特性的章节将详细介绍安全性报头。

帧挂起位只用在应答数据包内，该数据包由 MRF24J40 射频收发器硬件来处理。该位指示在从 RFD 终端设备接收到请求数据包之后，应答后是否需要跟随其他数据包。

内部 PAN 位指示消息是否在当前 PAN 内。如果该位置 1，那么在寻址字段内的源 PAN ID 字段将被省略。在此协议栈内，该位通常置为 1，但是如要使能内部 PAN 通信，可将其置为 0。如果需要，可在应用层中将该位复位为 0。

目标地址模式可为以下之一：

- 16 位短地址模式 = 10
- 64 位长地址模式 = 11

在 MiWi P2P 协议栈内，目标地址模式通常设为长地址模式。短地址模式只用于广播消息。对于广播消息，寻址字段内的目标地址字段固定为 0xFFFF。

MiWi P2P 协议栈的源地址模式只能为 64 位长地址模式。

序列号

序列号为八位。它从一个随机数字开始，每发送一个数据或命令数据包，该序列号就递增 1。该序列号用在应答数据包内，用来确认原始数据包。

原始数据包和应答数据包的序列号必须相同。

目标 PAN ID

这是目标设备的PAN标识符。在不知道或者不需要PAN标识符的情况下，可使用广播 PAN 标识符（0xFFFF）。

目标地址

目标地址可为 64 位的长地址，也可为 16 位的短地址。这个目标地址必须与帧控制字段中定义的目标地址模式保持一致。

如果使用 16 位的短地址，那么它必须是广播地址 0xFFFF。

源 PAN ID

源 PAN 标识符就是源设备的 PAN 标识符，必须与帧控制字段中的内部 PAN 定义相匹配。只有在内部 PAN 的值为 0 时，数据包中才存在源 PAN ID。

在当前 MiWi P2P 协议栈实现中，所有的通信都是内部 PAN 通信。因此，所有的数据包都没有源 PAN ID 字段。

但是，协议栈保留了应用层在内部 PAN 发送消息的功能。如果消息需要在内部 PAN 发送，则需使用源 PAN ID 。

源地址

源地址字段固定为使用源设备的 64 位扩展地址。

发送和接收

MiWi P2P 协议栈根据 IEEE 802.15.4 规范发送和接收数据包，个别例外情况除外。

发送消息

有两种发送消息的方式：广播和单播。

广播数据包把处于射频范围内的所有设备作为其广播目标。IEEE 802.15.4定义一个特殊短地址作为广播地址，但是没有对长地址做出定义。因此，使用广播的惟一情形是当 MiWi P2P 协议栈使用短地址时。

广播消息没有应答。

单播发送只有一个目标地址，并使用长地址作为目标地址。 MiWi P2P 协议栈要求应答所有的单播消息。

如果至少有一个发送设备因空闲而关闭了其射频，那么这个发送设备将把消息保存在 RAM 中，然后等待休眠设备唤醒并请求消息。此类数据发送称为间接消息传递。

如果休眠设备没有接收到这个间接消息，那么消息将超时并被丢弃。通常，间接消息超时时间要比休眠设备的轮询时间长一些。

接收消息

在MiWi P2P协议栈中，射频仅通知接收到消息的设备。如果接收到消息的设备在空闲时关掉了射频，那么它就只能从与它连接的设备接收消息。

对于关掉射频的空闲设备来说，如果它要接收消息，就必须向它的连接方发出数据请求命令。此时，如果存在间接消息，它将接收到该消息。

© 2009 Microchip Technology Inc.

握手过程的差异

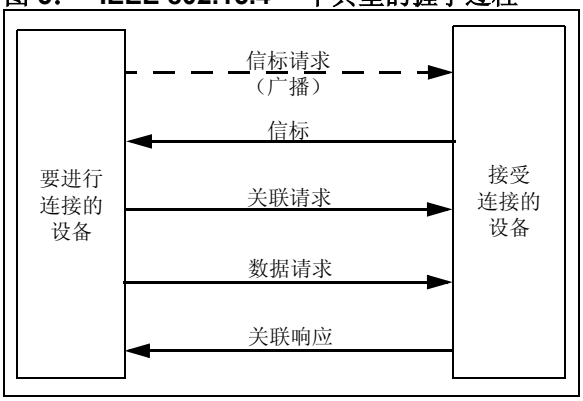
MiWi™ P2P 无线协议与 IEEE 802.15.4 规范的主要区别在于握手过程的不同。

在 IEEE 802.15.4 规范中，设备在上电后的第一步就是与其他设备进行握手。

图 5 给出了此规范的握手过程：

1. 寻求通信的设备发出一个信标请求。
2. 所有能够与其他设备连接的设备都会对信标消息做出应答。
3. 发起设备收集所有这些信标。（要容纳多个响应，设备应等待直到主动扫描请求超时。）设备决定用哪个信标来建立握手并发出一个关联请求命令。
4. 在预定义时间之后，发起设备发出一个数据请求命令，以从准备连接的另一方获得关联响应。
5. 连接的另一端设备发送关联响应。

图 5： IEEE 802.15.4™ 中典型的握手过程



握手是一个接入网络的复杂过程。设备只能连接一个设备作为其父节点，因此最初的握手其实是选择父节点的过程。

选择父节点需要：

1. 列出所有可能的父节点。
2. 选出一个合适的节点作为其父节点。

为了满足主动扫描超时的时序要求，在发送信标帧前，不使用 CSMA-CA 检测。因此，信标帧可能会因为数据包冲突而被丢弃。

MiWi P2P 协议设计为以星型和 P2P 通信拓扑结构实现简化的直接连接。某些 IEEE 802.15.4 要求阻碍了这种设计：

- 五步握手过程 —— 加上两个超时 —— 需要更加复杂的协议栈。
- 关联过程使用单连接通信，而不是点对点拓扑结构的多连接概念。

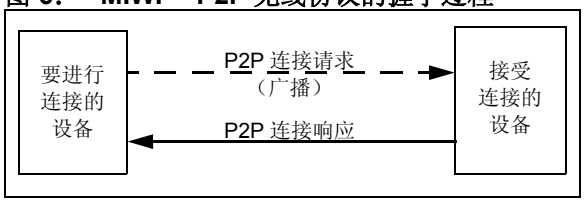
出于之前提到的原因，MiWi P2P 协议使用它自己的两步握手过程。图 6 给出了此过程：

1. 发起设备发出 P2P 连接请求命令。
2. 处于射频范围内的任何设备都会对要求建立连接的 P2P 连接响应命令做出响应。

这是一个可以建立多个连接的一对多的过程，如果可能，可建立点对点拓扑结构。由于这种握手过程使用了 MAC 层命令，因此每次数据发送都应使用 CSMA-CA。这就减少了数据包冲突的可能性。

RFD 可能会接收到来自多个 FFD 的连接请求命令，但是只能与一个 FFD 建立连接。 RFD 选择其接收到第一个 P2P 连接响应的 FFD 作为其连接方。

图 6： MIWI™ P2P 无线协议的握手过程



MiWi™ P2P 无线协议的定制 MAC 命令

通过使用移除两个设备间连接的定制 MAC 命令，MiWi P2P 协议扩展了 IEEE 802.15.4 规范的功能。表 3 中列出了该协议的所有定制 MAC 命令。

表 3： MIWI™ P2P 无线协议的定制 MAC 命令

命令标识符	命令名	说明
0x81	P2P 连接请求	请求建立 P2P 连接。在上电后，通常广播搜索 P2P 连接，而单播搜索单个连接。 也用于主动扫描功能。（请参见第 13 页的 **“ 主动扫描 ”**。）
0x82	P2P 连接移除请求	移除与其他终端设备的 P2P 连接。
0x83	数据请求	与 IEEE 802.15.4 规范的数据请求命令相似，如果本地节点的射频被关闭，那么将从 P2P 连接的另一端发出数据请求命令。为之前的休眠设备保存数据请求，以便用来请求另一个节点发送错失的消息（间接消息）。
0x84	通道跳转	请求把工作通道转换到其他通道。通常用在频率捷变特性中。
0x91	P2P 连接响应	对 P2P 连接请求的响应。也可用于主动扫描过程。
0x92	P2P 连接移除响应	对 P2P 连接移除请求的响应。

P2P 连接请求

在上电后，广播 P2P 连接请求 （0x81）以与另一个设备建立 P2P 连接。也可向特定设备单播请求以建立一个单独的连接。

当发送设备从另一端接收到 P2P 连接响应（0x91）时，就建立了 P2P 连接。

P2P 连接请求定制命令也可启动主动扫描，来确定附近有哪些设备可以使用。

当 P2P 连接请求命令以主动扫描的目的发出时，将不会附加功能信息和可选择的有效负载。接收设备根据附加信息或缺少的功能信息及可选择的有效负载，决定此命令是请求建立连接还是仅进行主动扫描。

MiWi P2P 协议栈可以使能或禁止设备与其他设备建立连接。一旦禁止设备建立连接，那么任何新的 P2P 连接请求都会被丢弃，除非满足以下条件：

- P2P 连接请求来自接收端已与其建立了连接的设备。
- P2P 连接请求是主动扫描。

P2P 连接请求命令帧的格式如图 7 所示。

图 7： P2P 连接请求命令格式

八位	15/21	1	1	1 （可选）	可变 （可选）
	MAC 报头	命令标识符（0x81）	工作通道	功能信息	用于标识节点的可选择的有效负载。非协议栈所必需的，但是对应用可能有用。

© 2009 Microchip Technology Inc.

工作通道用来回避来自其他通道的次谐波的影响。它将避免与运行在不同通道的设备之间的连接失败。

图 7 中的功能信息字节格式如图 8 所示。

图 8： 功能信息格式

位	0	1	2	3	4-7
	空闲时接收器开启	一旦醒来就进行数据请求	需要时间同步 （保留）	安全功能	（保留）

针对具体应用，提供 P2P 连接请求的可选有效负载。设备可能需要额外的信息来标识其本身——可以是其惟一的标识符，也可以是关于应用中所起功能的信息。通过可选择的有效负载，在建立连接后就不需要额外的数据包来介绍或标识设备了。

可选择的有效负载不可用于协议栈自身。

P2P 连接移除请求

P2P 连接移除请求 （0x82）被发送到连接的另一端来移除 P2P 连接。请求格式如图 9 所示。

图 9： P2P 连接移除请求格式

八位	15/21	1
	MAC 报头：发送到 P2P 连接的另一端来切断通信	命令标识符 （0x82）

数据请求

数据请求（0x83）命令与 IEEE 802.15.4 规范的数据请求 （0x04）命令相同。其格式如图 10 所示。

如果 P2P 连接节点的一端会在空闲时休眠，并且这个节点可以在休眠状态下接收消息，那么连接始终活动的一端必须把消息存储在其 RAM 中。当休眠设备唤醒并请求消息时，始终活动的一端将发送消息。

如果应用有这样的情况，那么需要激活 ENABLE_INDIRECT_MESSAGE 功能。休眠节点唤醒后，必须发送数据请求命令。

图 10： 数据请求格式

八位	21	1
	MAC 报头：从扩展源地址到扩展目标地址的单播	命令标识符 （0x83 或 0x04）

通道跳转

通道跳转命令（0x84）请求目标设备将工作通道转换为其他通道。命令格式如图 11 所示。

此命令通常由频率捷变启动器发出，它决定何时改变通道以及选择哪个通道。

通常广播此命令以通知所有在空闲时射频打开的设备切换通道。为了确保每个设备都接收到消息，频率捷变启动器将广播三次，而所有 FFD 设备将重新广播此消息。

当执行通道跳转序列以及所有 FFD 都跳转到新的通道时， RFD 必须执行重新同步以恢复与对应的 FFD 对等设备的连接。

图 11： 通道跳转格式

八位

15/21	1	1	1
MAC 报头：来自频率捷变启动器的广播或单播	命令标识符 （0x84）	当前运行的通道	跳转到的目标通道

P2P 连接响应

P2P 连接响应（0x91）命令用于对 P2P 连接请求做出响应。命令格式如图 12 所示。

P2P 连接响应命令可用于建立连接。此外，设备还可使用此命令来响应主动扫描，标识其自身为附近的活动设备。

如果接收到的 P2P 连接请求命令附加有功能信息字节和可选择的有效负载，则表明它正在请求连接。如果有功能信息和可选择的有效负载，它们会被附加到 P2P 连接响应上。

一旦连接的另一端接收到响应，就建立了 P2P 连接。连接的两端现在就可以交换数据包了。

如果接收到的P2P连接请求命令没有功能信息字节和可选择的有效负载，那么此命令便为主动扫描。因此，P2P 连接响应就不会附加功能信息或可选择的有效负载。

对于主动扫描连接请求，在消息交换后不会建立连接。

图 12： P2P 连接响应格式

八位

21	1	1	1 （可选）	可变 （可选）
MAC 报头：从扩展源地址到扩展目标地址的单播。	命令标识符（0x91）	状态位。 0x00 表示成功。所有其他值都是错误代码。	功能信息	用于标识节点的可选择的有效负载。非协议栈所必需的，但是对应用可能有用。

响应的功能信息格式如图 8 所示。

针对具体应用，提供可选的有效负载。其格式和用法与附加在 P2P 连接请求命令中的可选有效负载相同 （见第 9 页）。

P2P 连接移除响应

P2P 连接移除响应命令 （0x92）用于对 P2P 连接移除请求做出响应。它通知 P2P 连接的另一端早已接收到了 P2P 连接移除请求以及连接是否已移除。

命令格式如图 13 所示。

图 13： P2P 连接移除应答格式

八位

21	1	1
MAC 报头：从扩展源地址到扩展目标地址的单播	命令标识符 （0x92）	状态位 • 0x00 表示成功 • 所有其他值都是错误代码

© 2009 Microchip Technology Inc.

MIWI™ P2P 无线协议的特性

MiWi P2P 协议支持精简功能、点对点拓扑结构、直接连接以及丰富的特性集。根据无线应用的需要，所有的特性都可以使能或禁止，且可以编入协议栈，也可以从协议栈抽离出。

使用 Microchip 的 ZENA™ 软件应用程序可方便地实现协议栈的配置和报头文件的生成。欲知更多有关此软件的信息，请参见 *"ZENA™ Wireless Network Analyzer User's Guide"*（DS51606）。

这里介绍 MiWi P2P 协议的特性，包括：

- 小程序空间
- 支持设备空闲时关闭射频
- 间接消息传递
- 特定安全特性
- 用于寻找不同通道上现有 PAN 的主动扫描
- 用于寻找最小噪声通道的能量扫描
- 频率捷变 （通道跳转）

小程序空间

受许多无线应用成本的限制，MiWi P2P 协议栈应尽可能地小。使能以最小程序空间为目标的协议栈，可使代码长度减少至 3 KB 多一点。简单的应用程序可以很容易地装入到单片机仅 4 KB 的程序存储器中。

要激活这个特性，必须在 P2PDefs.h 文件中定义 "TARGET_SMALL"。 ZENA 软件可自动生成此文件，有经验的编程人员也可自己创建此文件。

此特性支持设备间的双向通信，但是禁止 PAN 之间的通信。如果使用了安全特性，那么将禁止更新检查。（欲知更多关于更新检查的信息，请参见第 12 页的 **" 安全特性 "**。）

设备空闲时关闭射频

这些设备由电池供电时，减小它们的功耗很重要。当这些设备不发送数据时，关闭它们的射频就可以降低功耗。MiWi P2P 协议栈具有使射频进入休眠模式以及从休眠中唤醒的特性。

要激活这个特性，必须在 P2PDefs.h 文件中定义 "ENABLE_SLEEP"。 ZENA 软件可自动生成此文件，有经验的用户也可自己创建此文件。

至于设备何时进入休眠模式由具体应用决定。可能的触发条件包括：

- 射频空闲的时间长度
- 接收到来自连接的 FFD 的数据包，请求设备进入休眠模式

唤醒设备的条件也由具体应用决定。可能的触发条件包括：

- 外部事件，比如按下按钮
- 预设定的定时器到期

当设备处于休眠模式时，其对等设备需要向它发送一条消息。如果没有发送消息，那么对等设备将无法使能其他特性。

如果对等设备需要发送消息给处于休眠模式的设备，那么对等设备必须把消息储存在它的易失性存储器中，直到休眠设备唤醒并获得此消息。由于消息不是直接发送给处于休眠模式的设备，因此该过程称为间接消息传递。

如果需要传递间接消息，那么休眠节点的对等设备需要在文件 P2PDefs.h 中定义 "ENABLE_INDIRECT_MESSAGE"。ZENA 工具可自动生成此文件，用户也可手动创建此文件。

如果使能了间接消息传递，就必须指定易失性存储器最多可存储的间接消息数。最大消息数取决于对等设备中可用的空闲 RAM 存储器以及连接至相同父节点 FFD 的 RFD 的个数。

最大间接消息数由文件 P2PDefs.h 中的 "INDIRECT_MESSAGE_SIZE" 定义。ZENA 软件可自动生成此文件，用户也可手动创建此文件。

要间接消息传递，还需要定义间接消息的超时周期。如果未定义超时周期且 RFD 设备不工作，那么此间接消息将永远保留在易失性存储器中。

间接消息的超时周期由文件 P2PDefs.h 中的 "INDIRCT_MESSAGE_TIMEOUT" 定义，测量单位为秒。

在一些应用中，广播很有用，但是它要求对等设备做更多的工作。当对等设备广播一条消息给 RFD 时，必须在 P2PDefs.h 文件中定义 "ENABLE_BROADCAST"。

安全特性

无线通信需要更多的安全性考虑，因为简单的监听器就可截取数据没有加密的消息。 MiWi P2P 协议栈提供了 IEEE 802.15.4 规范定义的七种安全模式，以满足各种安全性需求。这些模式可归为三类：

- AES-CTR 模式 —— 采用 16 字节安全密匙加密消息。该模式没有内置消息完整性检查或帧更新检查，因此该模式易受到重复攻击。
- AES-CBC-MAC 模式 —— 确保消息的完整性，每个数据包中附加有 4/8/16 字节的消息完整性代码（Message Integrity Code， MIC）。这保证了数据包 （包括报头和有效负载）在发送过程中不被修改。然而由于没有加密有效负载，因此消息是暴露的。

  MIC 的大小是由具体模式决定的：MIC 越大，提供的保护越强。
- AES-CCM 模式 —— 结合了前两种安全模式，即确保了消息的安全性也确保了其完整性。

表 4 给出了规范定义的七种安全模式。

表 4： IEEE 802.15.4™ 安全模式

安全模式		安全服务				MIC（字节）
标识符	名称	访问控制	数据加密	消息完整性	顺序更新	
01h	AES-CTR	X	X	—	X	0
02h	AES-CCM-128	X	X	X	X	16
03h	AES-CCM-64	X	X	X	X	8
04h	AES-CCM-32	X	X	X	X	4
05h	AES-CBC-MAC-128	X	—	X	—	16
06h	AES-CBC-MAC-64	X	—	X	—	8
07h	AES-CBC-MAC-32	X	—	X	—	4

要激活安全特性，必须在 P2PDefs.h 文件中定义 “ENABLE_SECURITY”。ZENA 工具可自动生成此文件，用户也可手动创建此文件。

当安全模式激活时，必须在 P2PDefs.h 中定义其他元素。这些元素包括 16 字节安全密匙、密匙序列号以及安全级。ZENA 软件帮助输入这些元素并把它们写入文件。或者，有经验的用户也可自己定义这些元素。

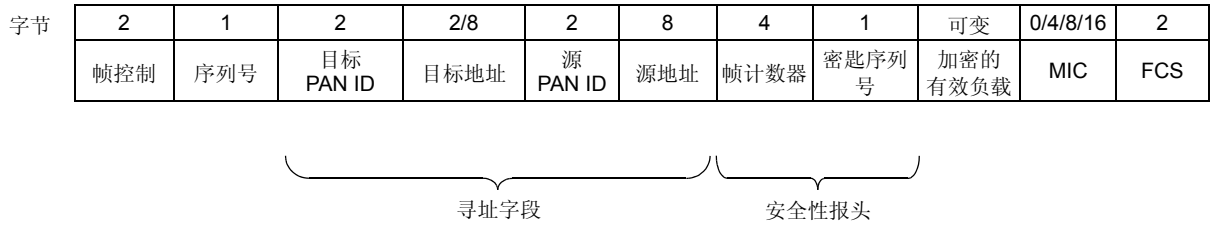
除了对数据进行加密和解密，安全模块也可使用帧计数器检查消息的更新。如果消息的帧计数器值小于记录值，那么丢弃消息。此特性可防止重复攻击。

如果希望占用的程序空间最小，可禁止消息的更新检查。通过定义 P2PDefs.h 文件中的 TARGET_SMALL 字段可实现此操作。

当使能安全性时，在标准 MAC 报头前增加了额外的安全性报头。安全性报头内含有四字节的帧计数器以及一字节的安全密匙序列号。

图 14 介绍了使能安全性时的数据包格式。

图 14： 使能安全性时 MIWI™ P2P 无线协议的数据包格式



© 2009 Microchip Technology Inc.

主动扫描

主动扫描就是获取本地个人局域网 （PAN）信息的过程。通过主动扫描得到以下信息：

- 设备的工作通道
- 在 PAN 中设备的信号强度
- PAN 标识码

在没有为本地设备预定义通道或 PAN ID 时，主动扫描尤其有用。

在协议栈中，ACTIVE_SCAN_RESULT_SIZE 定义了主动扫描可获取的最大 PAN 数量。

在主动扫描开始之前，需要确定扫描持续时间以及准备扫描的通道。

IEEE 802.15.4 规范定义了扫描持续时间，持续的时间长度（单位为 symbol）由公式 1 所示的公式计算得出。（1 秒等于 62,500 个 symbol。）

公式 1：

$$\text{扫描周期} \equiv 60 \cdot (2^{\text{ScanDuration}} + 1)$$

注： ScanDuration = 用户为扫描指定的输入参数。 1 到 14 之间的某一整数。

扫描持续时间为 10 将得到 61,500 个 symbol，即 1 秒的扫描周期。扫描持续时间为 9 得到约半秒的扫描周期。

扫描通道由位图定义，每个通道号由双字中可比较的比特数表示。通道 11 将由 b'0000 0000 0000 0000 0000 1000 0000 0000 表示。2.4GHz 频带支持的通道 11 到 26 由 b'0000 0111 1111 1111 1111 1000 0000 000 或 0x07FFF800 表示。

当主动扫描广播 P2P 连接请求命令时，它希望在射频范围内的所有设备都使用 P2P 连接响应命令做出应答。主动扫描将仅确定附近可连接的 PAN，而不是确定可用于新连接的设备数。这是因为每个设备都会响应扫描 —— 甚至那些不允许新连接的设备也会做出响应。

要激活主动扫描特性，必须在 P2PDefs.h 文件中定义 “ENABLE_ACTIVE_SCAN”。

能量扫描

IEEE 802.15.4 规范在 2.4 GHz 频带内定义了 16 个通道，但是 PAN 必须运行在一个通道上。使用的最佳通道就是能量或噪声最小的通道。

能量扫描用于扫描所有的可用通道并确定噪声最小的通道。

在执行能量扫描之前，需要确定扫描持续时间以及准备扫描的通道。

IEEE 802.15.4 规范定义了扫描持续时间，持续的时间长度（单位为 symbol）由公式 1 所示的公式计算得出。

请参见第 13 页 **“ 主动扫描 ”** 中关于测量的解释。

在完成扫描之后，返回噪声最小的通道标识符。

要激活能量扫描特性，必须在 P2PDefs.h 文件中定义 “ENABLE_ENERGY_SCAN”。

© 2009 Microchip Technology Inc.

频率捷变

如果运行条件要求转换通道，那么频率捷变将允许MiWi P2P PAN 转到其他通道。

在实现此特性时，受影响的设备可分为以下两种角色：

- 频率捷变启动器 —— 决定是否有必要进行通道跳转以及使用哪个新通道的设备
- 频率捷变跟随器 —— 得到指示时转换到其他通道的设备

频率捷变启动器

每个PAN可有一个或多个设备用作频率捷变启动器；启动器必须是 FFD。

每个启动器必须使能能量扫描特性。这是因为启动器必须执行能量扫描，以确定跳转到的最佳通道。然后，启动器向 PAN 上的其他设备广播通道跳转命令。

频率捷变跟随器

频率捷变跟随器可以是 FFD 设备或 RFD 设备。

FFD 通过执行以下两个操作之一，进行通道跳转：

- 从启动器接收通道跳转命令
- 如果数据发送不断失败，则重新同步连接

RFD 设备使用重新同步的方法（在通信失败时重新连接至 PAN）进行跳转。

实现并激活特性

何时执行频率捷变操作由应用程序决定。然而， MiWi P2P 协议栈提供了两个全局变量来帮助应用程序做出决定。

- “CCAFailureTimes”—— 定义了由于带冲突避免的载波侦听多路访问 （Carrier Sense Multiple Access with Collision Avoidance， CSMA-CA）失败而引起的数据发送连续失败的次数。

  这种情况通常意味着在当前通道上存在过多的噪声。
- “AckFailureTimes”—— 定义了由于没有应答而引起数据发送连续失败的次数。

  大的 AckFailureTimes 值通常意味着在接收端存在高噪声。或者，意味着无可用的目标设备，因为目标设备工作不正常或者已经跳转到其他通道。如果目标设备变换了通道，那么就有必要进行重新同步操作。

要激活频率捷变特性，必须在 P2PDefs.h 文件中定义“ENABLE_FREQUENCY_AGILITY”。要使能设备充当频率捷变启动器，必须在此文件中定义“FREQUENCY_AGILITY_STARTER”。

© 2009 Microchip Technology Inc.

应用程序接口 （API）

MiWi™ P2P 无线协议栈向应用程序开发人员提供了一组易用的 API。简单的 P2P 应用程序要求在应用层的代码少于 30 行。

协议栈通过使用附加的 API 调用扩展应用层，还可灵活地支持丰富的特性。

第 21 页的 **“ 应用程序接口 （API）清单 ”** 列出了这些 API—— 按函数名的字母顺序排列。此列表仅给出了每个 API 的基本功能特征。

本节给出了每个 API 的详细信息，通过以下基本步骤对它们进行分组：

创建点对点连接 16
接收数据包 17
发送数据包 18
进行主动扫描 19
进行能量扫描 19
使用频率捷变 20

简单的 P2P 应用程序

例 1 给出了一个简单的 P2P 应用程序代码。此代码实现了以下操作：建立连接，接收数据包以及发送数据包。

例 1 给出了 MiWi P2P 编程接口的关键部分。本文档的后续部分中，此示例程序代码将用于说明应用层如何使用协议栈实现对等设备的互连。

例 1： 简单的 P2P 应用程序

```
1.  void main(void)
2.  {
3.     Initiazation();// 初始化硬件平台和 P2P 协议栈
4.    SetChannel(myChannel);// 设置准备运行的通道
5.     EnableNewConnection();// 接受新的 P2P 连接
6.    CreateNewConnection(2);// 与另一个设备建立连接
7.
8.      while(1)
9.      {
10.      if( ReceivedPacket() )// 检查是否已接收到数据包
11.     {
12.       LED_1 = rxFrame.PayLoad[0]          // 处理数据包，根据接收到的数据包设置 LED
13.     DiscardPacket();          // 丢弃数据包，准备好接收下一数据包
14.       }
15.       else
16.       {
17.     BYTE PressedButton = ButtonPressed();          // 检查是否按下了按钮
18.     if (PressedButton)                             // 如果已经按下了按钮
19.     {
20.         FlushTx();    // 复位发送缓冲区
21.         for(i = 0; i < 65; i++)
22.         {
23.            WriteData(P2P[i]);// 填充发送缓冲区
24.         }
25.
26.       UnicastConnection(0, FALSE, TRUE);  // 单播数据包
27.          }
28.     }
29.  }
```

创建点对点连接

本节详细介绍用于创建点对点连接的 API。

注：	第 21 页的 **“ 应用程序接口 （API）清单 ”** 列出了所有 API，按函数名的字母顺序排列。此列表仅给出了基本功能元素。

在例 1 中，代码的第 5-6 行说明了建立连接的两种方法：

- 被动接受一个新连接
- 主动寻求一个新连接

要接受一个新连接，设备需要：

1. 等待 P2P 连接请求。
2. 响应 P2P 连接响应命令。

要寻求一个新连接，设备需要：

1. 发出一个 P2P 连接请求命令，搜索可连接的设备。
2. 接收对应的 P2P 连接响应命令，完成连接。

总之，在建立一个新连接之后，连接另一端的设备信息将被存储在两个设备的 P2P 连接项中。那些项代表现有连接并可用于向连接的对等设备发送数据包。（欲知更多信息，请参见第 18 页的 **“ 发送数据包 ”**。）

下面是建立点对点连接的 API。

void EnableNewConnection(void)

EnableNewConnection 函数调用通过使用 P2P 连接响应对 P2P 连接请求命令做出响应，从而允许协议栈接受新连接。

void DisableNewConnection(void)

DisableNewConnection通过对P2P连接请求命令做出响应来防止协议栈接受新连接。在以下两种情况下，将忽略此设置，即使禁止了新连接也如此：

- P2P 连接请求命令实际上是一种没有附加任何对等信息的主动扫描。
  将发出没有附加对等信息的 P2P 连接响应命令作为响应。
- P2P 连接请求命令来自于已与当前设备建立了连接的设备。
  在断电后当连接的一端用于重建原有连接时，通常使用此功能。

BYTE CreateNewConnection(BYTE RetryInterval)/BYTE CreateNewConnection(BYTE RetryInterval, DWORD ChannelMap)

CreateNewConnection 函数调用用于主动寻求新连接。在调用此函数之前，不必使能协议栈接受新连接，此函数会自动使能接受新连接。

当函数调用返回时，接受新连接的状态将被设置回以前的状态。

此函数调用按以下步骤进行：

1. 发出带有附加对等信息的 P2P 连接请求命令。
2. 协议栈等待 P2P 连接响应命令以建立连接。
3. 当建立了第一个连接后，函数调用返回 P2P 连接项中的连接索引。

CreateNewConnection 函数有两种格式。

- 当没有激活能量扫描特性时，此函数调用使用的惟一参数是发出 P2P 连接请求的时间间隔 （单位为秒）。
- 当激活能量扫描时，增加了另外一个参数 ChannelMap，该参数指定了搜索新连接的扫描通道。

ChannelMap 是双字参数，它利用位图指定要扫描的通道。每个通道号由双字中可比较的双字比特数表示。通道 11 由 b’0000 1000 0000 0000 表示。为了扫描 2.4 GHz 频带内的所有通道（11 至 26），ChannelMap 应设置为 0x07FFF800。

© 2009 Microchip Technology Inc.

接收数据包

本节详细介绍接收数据包的 API。

注：	第 21 页的“ 应用程序接口 （API）清单 ”列出了所有 API，按函数名的字母顺序排列。此列表仅给出了基本功能元素。

在例 1 中，代码的第 10-14 行说明了接收和处理数据包的方法。

Boolean ReceivedPacket(void)

ReceivedPacket 函数返回一个布尔表达式，以指示 MiWi P2P 协议栈是否已接收到一个数据包。如果调用此函数，那么在返回指示是否已接收到数据包的状态之前，整个协议栈在内部被调用。

如果协议栈已接收到数据包，那么 ReceivedPacket 的返回值为 TRUE。关于接收到的数据包的所有信息都存储在全局变量 rxFrame 中，它是一个 RECEIVED_FRAME 结构的变量。RECEIVED_FRAME 结构的定义如例 2 所示。

例 2： RECEIVED_FRAME 结构

```
1.  typedef struct _RECEIVED_FRAME
2.  {
3.      union _RECEIVED_FRAME_FLAG
4.      {
5.          BYTE    Val;
6.          struct _RECEIVED_FRAME_FLAG_BITS
7.          {
8.              BYTE    commandFrame    : 1; // 数据：0；命令： 1
9.              BYTE    security        : 1;
10.             BYTE    framePending    : 1;
11.             BYTE    intraPAN        : 1;
12.             BYTE    broadcast       : 1;
13.         } bits;
14.     } flags;
15.
16.     #ifndef TARGET_SMALL
17.         BYTE        PacketLQI;
18.         BYTE        PacketRSSI;
19.         WORD_VAL    SourcePANID;
20.     #endif
21.     BYTE        SourceLongAddress[8];
22.     BYTE        PayLoadSize;
23.     BYTE        *PayLoad;
24. } RECEIVED_FRAME;
```

此结构实际上说明，对接收到的数据包的所有有关信息进行了分类和存储。 RECEIVED_FRAME 的定义说明：

- 参数 PayLoadSize 和 PayLoad 仅用于 MAC 有效负载。不包括 MAC 报头。
- 如果协议栈配置为使用最小程序空间，那么 PacketLQI、 PacketRSSI 和 SourcePANID 等一些参数将无法使用。

  在这种情况下，不支持高信号质量和强度的内部 PAN 通信功能。
- 如果在发送过程中对数据包进行了加密，那么该信息也在 RECEIVED_PACKET 结构中解密。

  数据包原来是否进行了加密可在标志（flags.bits.security）的相应设置中找到。

在接收到数据包之后，接下来如何处理数据包由具体应用决定。数据包在处理完成后以及在返回到处理协议栈之前，需要调用 DiscardPacket 函数。函数移除当前接收到的数据包并准备接收下一数据包。

如果没有调用 DiscardPacket 函数，那么旧数据包将占用存储空间，就无法接收新的数据包。

发送数据包

本节详细介绍与发送数据包有关的 API。

注：	第 21 页的 **“ 应用程序接口 （API）清单 ”** 列出了所有 API，按函数名的字母顺序排列。此列表仅给出了基本功能元素。

在例 1 中，代码的第 20-26 行说明了一种发送数据包的方法。

当发送数据包时，协议栈将处理 MAC 报头。应用层仅关心 MAC 有效负载。

void FlushTx(void)

FlushTx 调用函数复位发送缓冲区，准备接收要发送的信息。

void WriteData(BYTE data)

WriteData 调用函数将向发送缓冲区每次填充一字节数据。如果数据包中有 20 个字节需要发送，那么将调用该函数 20 次。

发送缓冲区一旦填满了，就可发送数据包了。有三种发送数据包的方法：

- 以 P2P 连接项中的连接索引单播
- 以长地址单播
- 广播

BOOL UnicastConnection(BYTE ConnectionIndex, BOOL isCommand, BOOL SecurityEnabled)

在例 1 中，第 26 行的 UnicastConnection 函数使用了以连接索引的单播方法。

P2P 连接项中的记录包含了有关 P2P 连接对等设备的信息，协议栈利用此信息把数据包发送到目标地址。

该函数有三个参数：

- P2P 连接项中记录的索引
- 指示数据包是否是命令数据包的布尔表达式。
  正如前面讨论的， MiWi P2P 协议栈仅支持在 MAC 报头的帧控制字节中定义的数据和命令帧。（欲知更多信息，请参见第 5 页的 **“ 消息格式 ”** 或者 IEEE 802.15.4 规范。）
- 指示数据包是否需要加密的布尔表达式。
  必须使能安全特性，否则无法进行加密。

BOOL UnicastLongAddress(WORD_VAL DestinationPANID, BYTE *DestinationAddress, BOOL isCommand, BOOL SecurityEnabled)

通过调用 UnicastLongAddress 函数也可单播消息。通过指定长地址和 PAN 标识符，协议栈便可确定目标地址。

该函数有四个参数：

- 目标 PAN 标识符
- 目标长地址
- 指示数据包是否是命令数据包的布尔表达式
- 指示数据包是否需要加密的布尔表达式

BOOL BroadcastPacket(WORD_VAL DestinationPANID, BOOL isCommand, BOOL SecurityEnabled)

单播向单个目标地址发送数据包。当需要向射频范围内的所有设备发送消息时，就需要使用广播——调用函数 BroadcastPacket。

BroadcastPacket 函数有三个参数：

- 目标 PAN 标识符
  如果要向所有的 PAN 广播数据包，那么应用程序可选择使用 0xFFFF 标识符，这样可以消除内部 PAN 通信的限制。
- 指示数据包是否是命令数据包的布尔表达式
- 指示数据包是否需要加密的布尔表达式

© 2009 Microchip Technology Inc.

进行主动扫描

本节详细介绍主动扫描的 API。

注：	第 21 页的 **“ 应用程序接口 （API）清单 ”** 列出了所有 API，按函数名的字母顺序排列。此列表仅给出了基本功能元素。

知道本区域内是否有 MiWi P2P PAN 正在工作；如果有的话，它们的 PAN 标识符是什么，这些信息都很有用。这使得 PAN 协调器能够选择一个 PAN 加入，或使用惟一的 PAN 标识符启动一个新 PAN。

要完成此操作，MiWi P2P 协议栈提供了函数调用 `ActiveScan`。

BYTE ActiveScan(BYTE ScanDuration, DWORD ChannelMap)

只有在激活了主动扫描特性时，此函数调用才可用。

`ActiveScan` 函数有两个参数：

- 对每个通道进行扫描的持续时间。
  IEEE 802.15.4 规范给出了扫描持续时间的定义。欲知更多信息，请参见第 13 页的 **“ 主动扫描 ”**。
- 指定准备扫描的通道的位图。
  欲知更多关于通道位图设置的信息，请参阅第 13 页的 **“ 主动扫描 ”**。

`ActiveScan` 函数调用返回搜索到的 PAN 总数。所有的扫描结果存储在 `ACTIVE_SCAN_RESULT` 结构的全局变量 `ActiveScanResults` 中。`ACTIVE_SCAN_RESULT` 结构的定义如例 3 所示。

例 3： ACTIVE_SCAN_RESULT

```
1.  typedef struct _ACTIVE_SCAN_RESULT
2.  {
3.      BYTE            Channel;
4.      BYTE            RSSIValue;
5.      WORD_VAL        PANID;
6.  } ACTIVE_SCAN_RESULT;
```

如上例所示，主动扫描结果提供了通道数、信号强度以及 PAN 标识符。

在协议栈中，`ACTIVE_SCAN_RESULT_SIZE` 定义了主动扫描可获取的最大 PAN 数量。在 RAM 资源允许的情况下，应用程序开发人员可修改此设置。

默认值为 16。

进行能量扫描

本节详细介绍能量扫描的 API。

注：	第 21 页的 **“ 应用程序接口 （API）清单 ”** 列出了所有 API，按函数名的字母顺序排列。此列表仅给出了基本功能元素。

在 2.4 GHz 频带内有 16 个可用通道。由于 MiWi P2P PAN 可运行在任何一个通道中，因此它可选择噪声最小的通道。

能量扫描特性就是实现此功能。要能够调用编程接口运行能量扫描，必须激活能量扫描特性。

BYTE OptimalChannel(BYTE ScanDuration, DWORD ChannelMap, BYTE *RSSIValue)

`OptimalChannel` 函数需要三个参数：

- 对每个通道进行扫描的持续时间。
  扫描持续时间的定义遵循 IEEE 802.15.4 规范。欲知更多信息，请参见第 13 页的 **“ 能量扫描 ”**。
- 指定准备扫描的通道的位图。
  欲知更多关于通道位图设置的信息，请参见第 13 页的 **“ 能量扫描 ”**。
- `RSSIValue`—— 输出参数，它返回在返回的最佳通道上读取的最大能量。

`OptimalChannel` 返回在扫描期间噪声最小的通道。

使用频率捷变

频率捷变特性使 MiWi P2P 协议栈与无线环境相兼容。有时也称为通道跳转，此特性使 MiWi P2P PAN 能够在 2.4 GHz 频带通道之间移植，以避免噪声级的改变。

MiWi P2P PAN 上的设备可作为以下两种频率 / 捷变角色之一：启动器或跟随器。频率捷变启动器通过调用 `InitChannelHopping` 函数启动通道跳转过程。

BOOL InitChannelHopping (DWORD ChannelMap)

此函数需要一个参数：PAN 可能转移到的通道的位图。欲知更多关于此设置的信息，请参见第 14 页的 **“ 频率捷变 ”**。

`InitChannelHopping` 将返回一个布尔表达式，以指示操作是否成功。在检查其他通道之后，如果发现当前通道的噪声最小，那么初始化程序停止并返回值 FALSE。

如果变换了通道，在接收到来自频率捷变启动器的请求之后或在重新同步连接之后，频率 / 捷变跟随器跳到新的通道。

要重新同步连接，频率 / 捷变跟随器将调用 `Resync-Connection` 函数。

BOOL ResyncConnection (BYTE *DestinationAddress, DWORD ChannelMap)

`ResyncConnection` 函数有两个参数：

- 目标地址 —— 指向频率 / 捷变跟随器将与之重新同步的设备的长地址指针。

  此地址通常是频率捷变跟随器丢失与其连接的节点的地址。
- 通道位图 —— 可用新通道的位图。

  欲知更多信息，请参见第 14 页的 **“ 频率捷变 ”**。

`ResyncConnection` 将返回一个布尔表达式，以指示操作是否成功。如果重新同步连接在每个通道上尝试三次均不成功，那么函数调用将返回 FALSE。

© 2009 Microchip Technology Inc.

应用程序接口 （API）清单

ActiveScan

该函数执行主动扫描，搜索附近所有的 MiWi P2P PAN。

语法

```
BYTE ActiveScan(BYTE ScanDuration, DWORD ChannelMap)
```

输入

- BYTE – ScanDuration：对每个通道进行能量扫描的时间周期。
- DWORD – ChannelMap：进行能量扫描的通道的位图。

输出

BYTE —— 现有 MiWi P2P PAN 数量。

注

扫描结果将存储在全局变量数组 ActiveScanResults 中，最大结果大小为 ACTIVE_SCAN_RESULT_SIZE。

BroadcastPacket

该函数向射频范围内所有带有目标 PAN 标识符的设备广播消息。

语法

```
BOOL BroadcastPacket(WORD_VAL DestinationPANID, BOOL isCommand, BOOL SecurityEnabled)
```

输入

- WORD_VAL – DestinationPANID：目标设备的 PAN 标识符。
- BOOL – isCommand：指示发送的数据包是否是命令数据包的布尔表达式。
- BOOL – SecurityEnabled：指示发送的数据包是否需要加密的布尔表达式。

输出

BOOL —— 指示操作是否成功的布尔表达式。

注

在调用函数之前，应已经装入消息有效负载。

CreateNewConnection

该函数主动寻求新的 P2P 连接。

语法

```
BYTE CreateNewConnection(BYTE RetryInterval)
BYTE CreateNewConnection(BYTE RetryInterval, DWORD ChannelMap)
```

输入

- BYTE – RetryInterval：两次尝试请求新连接的时间间隔，单位为秒。
- DWORD – ChannelMap：尝试建立连接的通道的位图。要求使能能量扫描特性。

输出

BYTE ——P2P 连接项中新连接的索引。

注

仅在建立新 P2P 连接的情况下，有一个返回。

DisableNewConnection

该函数禁止协议栈接受新连接。

语法

```
void DisableNewConnection(void)
```

输入

无

输出

无

注

在禁止协议栈接受新连接后，如果请求来自已连接的设备或者请求为主动扫描，则发送响应。

DisableAcknowledgement

该函数禁止协议栈对每个单播数据包请求 MAC 应答。

语法

```
void DisableAcknowledgement(void)
```

输入

无

输出

无

注

无

DiscardPacket

该函数通知协议栈丢弃当前接收到的数据包，准备接收下一数据包。

语法

```
void DiscardPacket(void)
```

输入

无

输出

无

注

如果在处理当前数据包之后没有调用该函数，那么协议栈将永远不能接收下一个数据包。

© 2009 Microchip Technology Inc.

DumpConnection

该函数在超级终端打印有关连接的信息。

语法

```
void DumpConnection(BYTE index)
```

输入

BYTE Index：P2P 连接项中的连接索引。（值 0xFF 代表所有连接。）

输出

无

注

该函数通常在开发期间使用。

EnableAcknowledgement

该函数使能协议栈对每个单播数据包请求 MAC 应答。

语法

```
void EnableAcknowledgement(void)
```

输入

无

输出

无

注

无

EnableNewConnection

该函数使能协议栈开始接受新连接。

语法

```
void EnableNewConnection(void)
```

输入

无

输出

无

注

无

FlushTx

该函数复位发送缓冲区。

语法

```
void FlushTx(void)
```

输入

无

输出

无

注

应用程序启动后，在填充发送缓冲区之前，应调用该函数。

InitChannelHopping

频率捷变启动器调用该函数，以启动频率捷变（通道跳转）过程。

语法

```
BOOL InitChannelHopping( DWORD ChannelMap)
```

输入

DWORD ChannelMap：PAN 可能跳转到的候选通道的位图。

输出

BOOL—— 指示通道跳转请求成功的布尔表达式。

注

如果当前工作通道的噪声最小，那么不执行通道跳转操作并返回值 FALSE。

MRF24J40Sleep

该函数使 MRF24J40 射频收发器进入休眠模式。

语法

```
void MRF24J40Sleep(void)
```

输入

无

输出

无

注

无

© 2009 Microchip Technology Inc.

MRF24J40Wake

该函数把 MRF24J40 射频收发器从休眠模式唤醒。

语法

```
void MRF24J40Wake(void)
```

输入

无

输出

无

注

无

OptimalChannel

该函数用于扫描并寻找噪声最小的通道。

语法

```
BYTE OptimalChannel(BYTE ScanDuration, DWORD ChannelMap, BYTE *RSSIValue)
```

输入

- BYTE – ScanDuration：对每个通道能量进行扫描的时间周期。
- DWORD – ChannelMap：进行能量扫描的通道的位图。

输出

- BYTE – RSSIValue：指向最佳通道能级的指针。
- BYTE—— 噪声最小的最佳通道。

注

无

P2PInit

该函数初始化 MiWi P2P 协议栈。

语法

```
void P2PInit(void)
```

输入

无

输出

无

注

无

ReceivedPacket

该函数调用 MiWi P2P 协议栈并检查是否已接收到数据包。

语法

```
BOOL ReceivedPacket(void)
```

输入

无

输出

BOOL—— 指示 MiWi P2P 协议栈已接收到数据包的布尔表达式。

注

无

ResyncConnection

该函数用于重新同步连接。

语法

```
BOOL ResyncConnection(BYTE *DestinationAddress, DWORD ChannelMap)
```

输入

- BYTE – DestinationAddress：指向连接中准备与之重新同步的对等设备的长目标地址的指针。
- DWORD – ChannelMap：符合重新同步条件的通道的位图。

输出

BOOL—— 指示重新同步操作成功的布尔表达式。

注

无

SetChannel

该函数将设备设置为工作在 2.4 GHz 频带内可用的 16 通道之一。

语法

```
void SetChannel(BYTE channel)
```

输入

BYTE – Channel：设备当前使用的通道。

输出

无

注

无

© 2009 Microchip Technology Inc.

UnicastConnection

该函数向 P2P 连接项字段中连接的对等设备单播消息。

语法

```
BOOL UnicastConnection(BYTE ConnectionIndex, BOOL isCommand, BOOL SecurityEnabled)
```

输入

- BYTE – ConnectionIndex：P2P 连接项字段中对等设备的索引。
- BOOL – isCommand：指示数据包是命令数据包的布尔表达式。
- BOOL – SecurityEnabled：指示数据包是否需要加密的布尔表达式。

输出

BOOL—— 指示单播操作成功的布尔表达式的返回值。

注

无

UnicastLongAddress

该函数向具有输入长地址的设备单播消息。

语法

```
BOOL UnicastLongAddress(WORD_VAL DestinationPANID, BYTE *DestinationAddress, BOOL
isCommand, BOOL SecurityEnabled)
```

输入

- WORD_VAL – DestinationPANID：目标设备的 PAN 标识符。
- BYTE * – DestinationAddress：指向目标设备 8 字节长地址的指针。
- BOOL – isCommand：指示数据包是命令数据包的布尔表达式。
- BOOL – SecurityEnabled：指示数据包需要加密的布尔表达式。

输出

BOOL—— 指示操作成功的布尔表达式的返回值。

注

无

WriteData

该函数向发送缓冲区写入一字节数据。

语法

```
void WriteData(BYTE data)
```

输入

BYTE – Data：写入发送缓冲区的数据。

输出

无

注

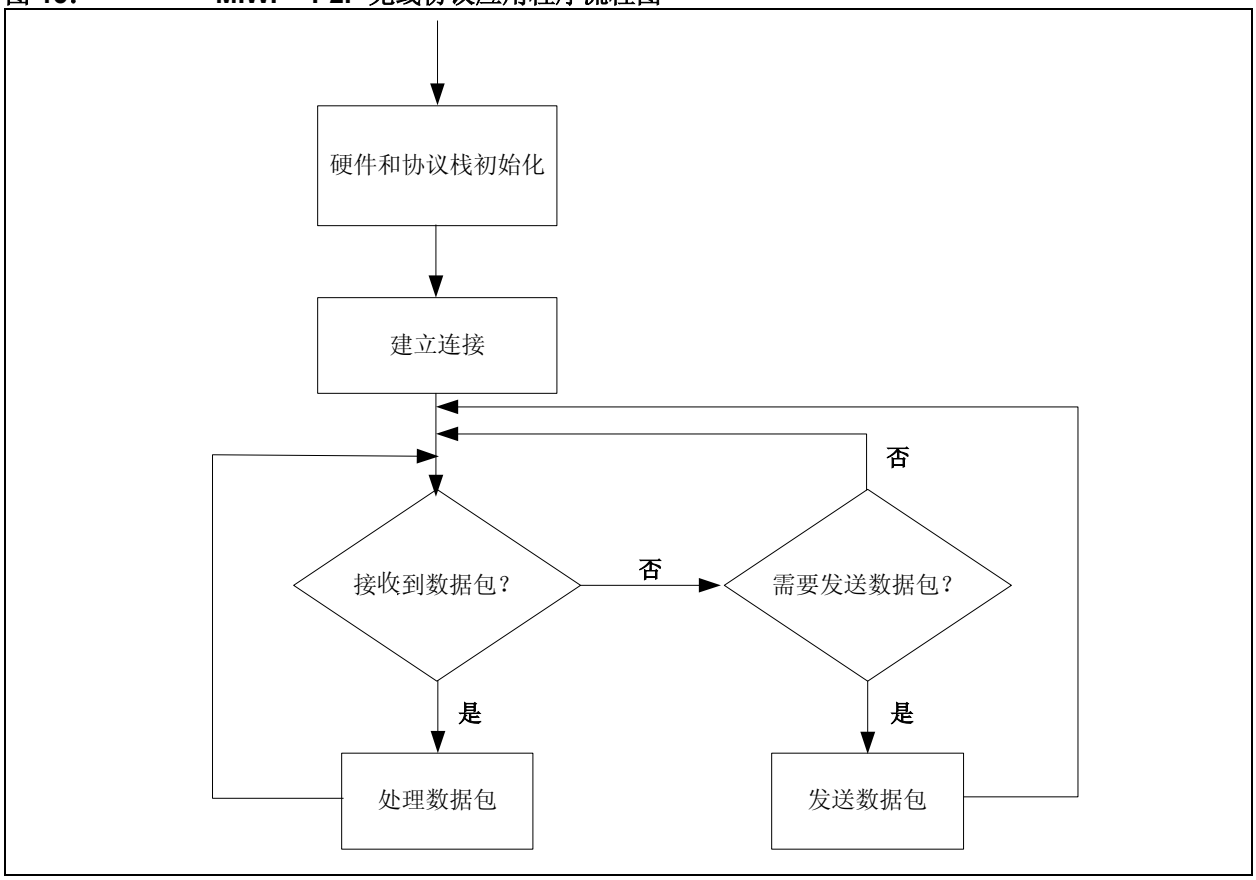
仅在 FlushTx 函数之后，调用该函数。

应用程序流程图

典型的 MiWi P2P 应用程序从初始化硬件和 MiWi P2P 协议栈开始。然后，尝试建立连接并进入接收和发送数据的正常工作模式。

图 15 给出了 MiWi P2P 应用程序的典型流程图。

图 15： MIWI™ P2P 无线协议应用程序流程图

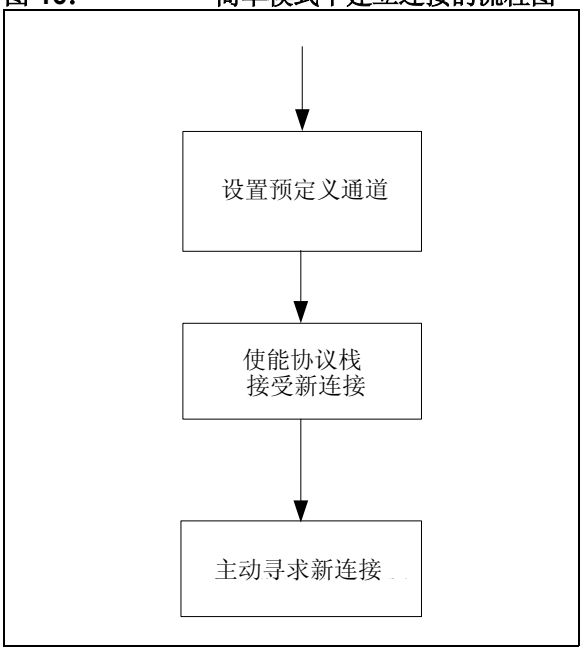


© 2009 Microchip Technology Inc.

在建立连接之后，大部分 MiWi P2P 应用程序的执行步骤相同。任何变化 —— 因不同的协议栈配置而不同 —— 都发生在连接建立期间。

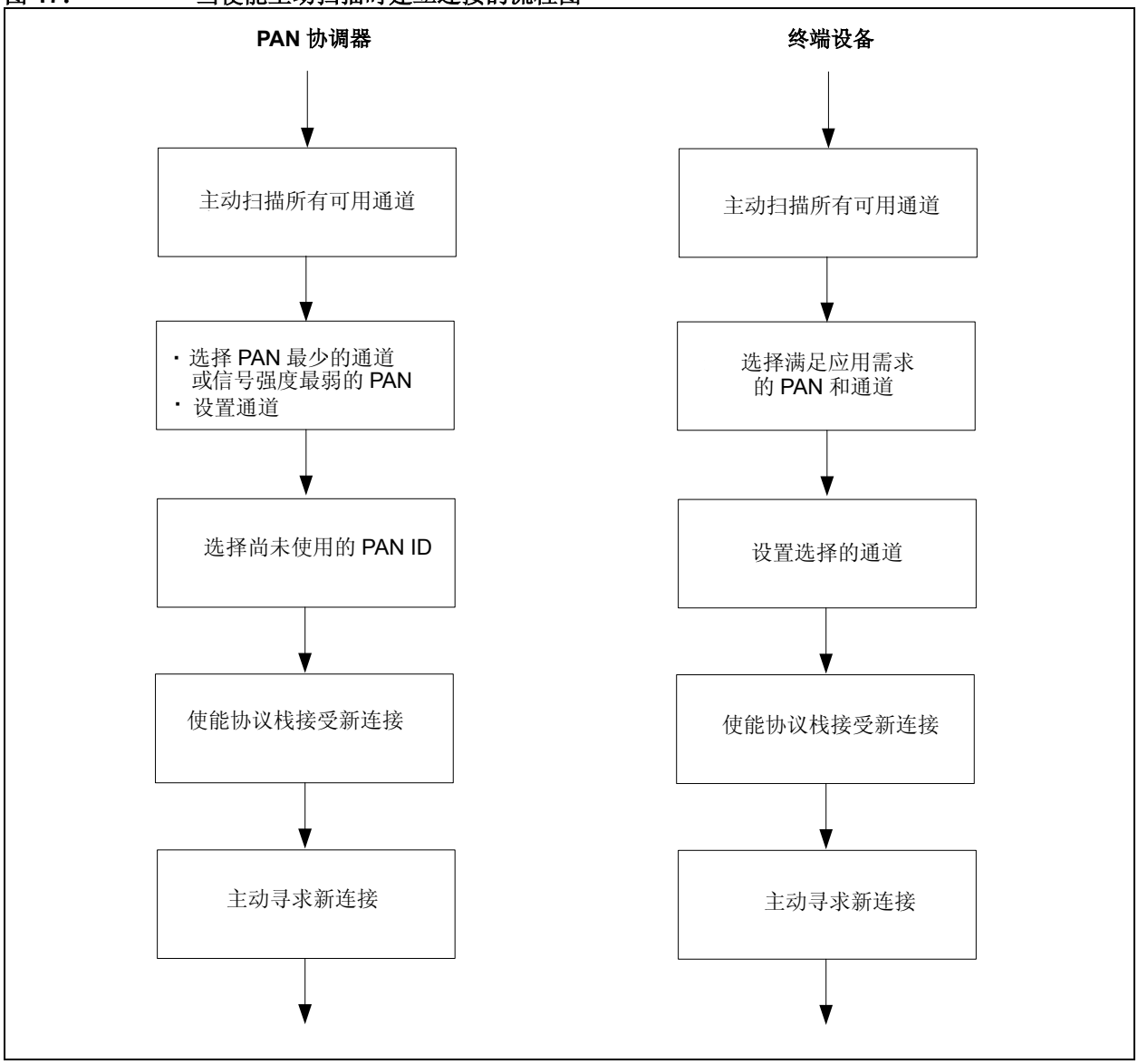
用于建立连接的最简单的 P2P 连接应用程序如图 16 所示。

图 16：　　简单模式下建立连接的流程图



在需要主动扫描功能的较复杂的应用程序中，PAN 协调器和终端设备建立连接的步骤各有不同，图 17 给出了这两类设备的主动扫描步骤。

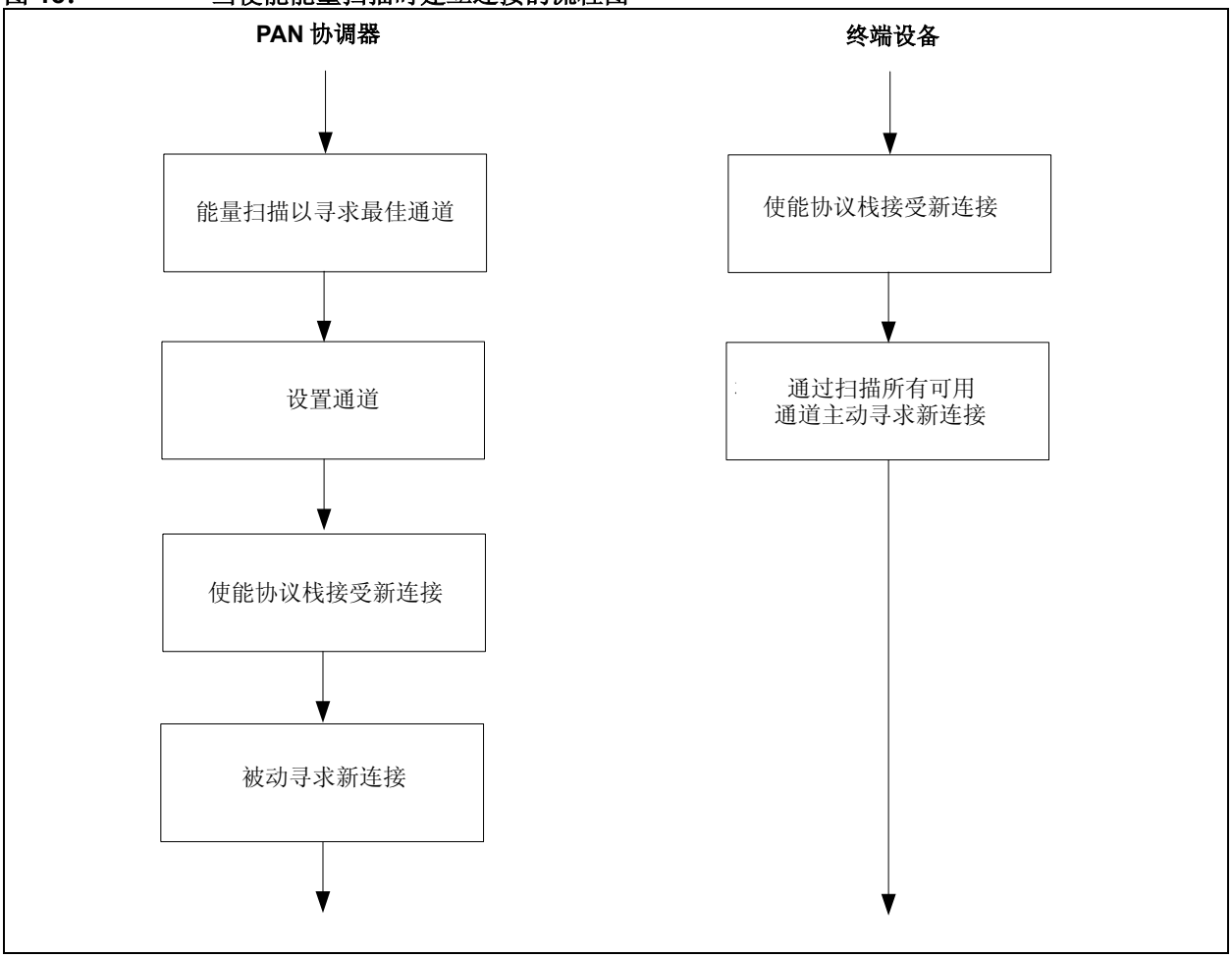
图 17：　　当使能主动扫描时建立连接的流程图



© 2009 Microchip Technology Inc.

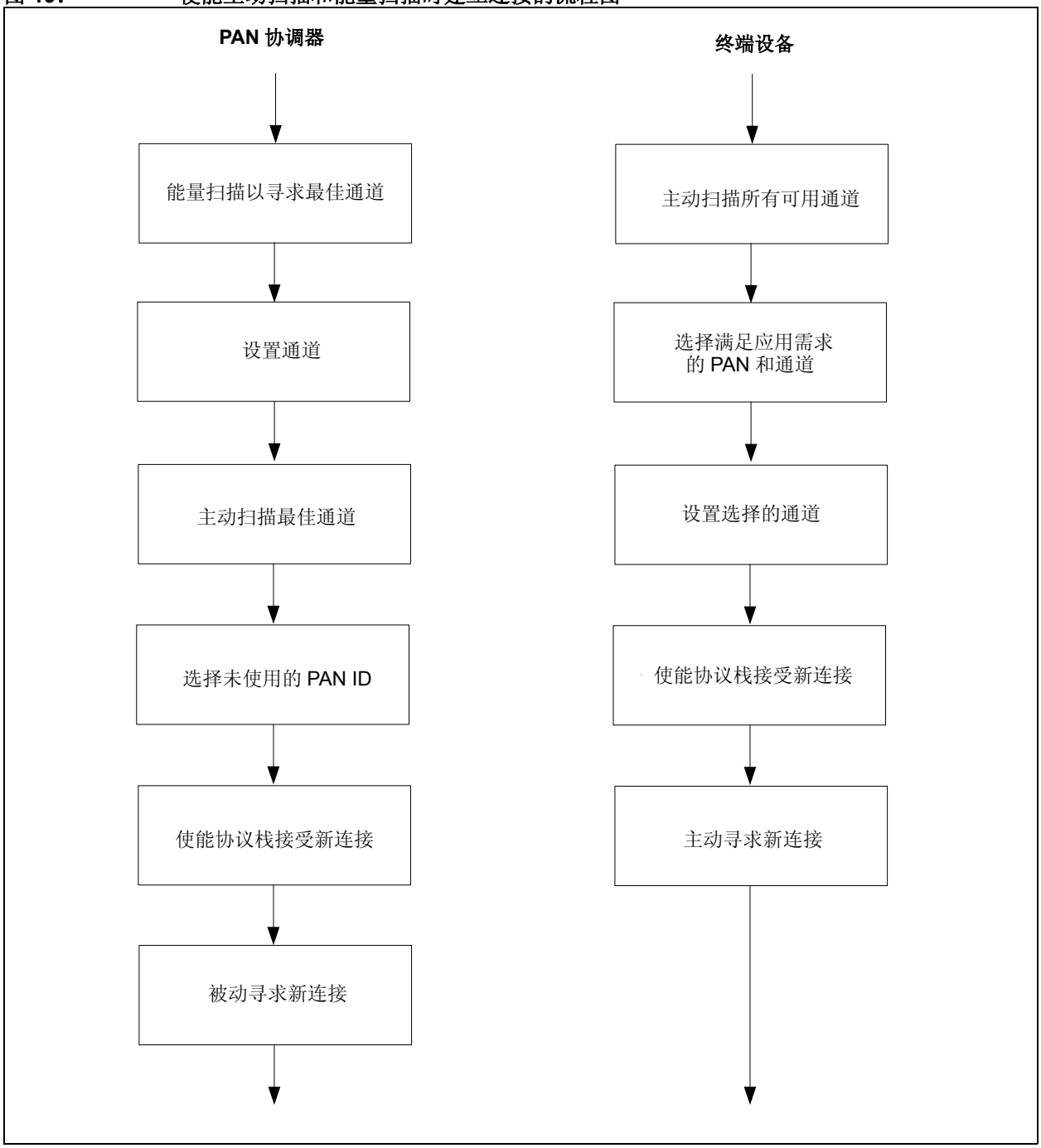
对使能能量扫描的应用程序，PAN 协调器和终端设备建立连接的步骤也有所不同（见图 18）。

图 18：　　当使能能量扫描时建立连接的流程图



建立连接的过程 —— 同时使能主动扫描和能量扫描 —
— 如图 19 所示。

图 19：　　使能主动扫描和能量扫描时建立连接的流程图



© 2009 Microchip Technology Inc.

系统资源要求

MiWi™ P2P 无线协议栈具有丰富的特性集。使能特性集将增加对单片机的系统要求。

表 5 给出基本的配置要求。

表 5： MIWI™ P2P 无线协议的基本协议栈对 PIC18 存储器的要求

配置	程序存储器（字节）	RAM （字节）
目标小协议栈大小	3,336	100 + 接收缓冲区大小 + 发送缓冲区大小 + (9 * P2P 连接大小)

其他 MiWi P2P 功能需要更大的程序存储器和 RAM。表 6 列出了对于高于基本配置的各特性的系统要求。

表 6： MIWI™ P2P 无线协议栈各特性对 PIC18 存储器的要求 †

配置	额外的程序存储器（字节）	额外的 RAM （字节）
使能内部 PAN 通信	462	0
使能休眠模式	186	0
使能安全性 （不带帧刷新检查）	500	48
使能安全性 （带帧刷新检查）	1,488	54
使能主动扫描	1,070	69
使能能量扫描	752	0
使能间接消息	950	间接消息大小 * 发送缓冲区大小
使能带有广播功能的间接消息	1,228	间接消息大小 * 发送缓冲区大小

† 这些要求是针对 PIC18 系列单片机的。该协议栈也支持 PIC24、 dsPIC33 和 PIC32 单片机，但是这些器件的要求可能会有所不同。

这些要求适用于该协议栈的最初版本，可能会更改。

结论

对于采用星型或点对点拓扑结构的无线应用来说，MiWi™ P2P 无线协议是一个很好的解决方案。该协议栈是一个简单而可靠的解决方案，具备了 IEEE 802.15.4 规范所有的优点。

如果应用比较复杂，可以考虑 Microchip MiWi™ 网路协议栈。该协议栈所支持的实际网络可最多经由 4 个路程段，最多有 1,024 个主动节点。欲知关于该协议的更多信息，请参见 AN1066《MiWi™ 无线网络协议栈》（DS01066A_CN）。

若需要更加复杂的网络或协同工作的能力， Microchip 实现的 ZigBee 协议规范也是一种选择。欲知关于该协议的更多信息，请参见 AN965《Microchip ZigBee™ 协议栈》（DS00965C_CN）。

参考文献

D. Flowers 和 Y. Yang，AN1066，《MiWi™ 无线网路协议栈》（DS01066A_CN），Microchip Technology Inc.，2007。

D. Flowers、K. Otten、Nilesh Rajbharti 和 Y. Yang，AN965，《Microchip ZigBee™ 协议栈》（DS00965C_CN），Microchip Technology Inc.，2007。

IEEE 标准 802.15.4-2003™，适用于低速率无线个人局域网（WPAN）的无线介质访问控制（MAC）和物理层（PHY）规范，IEEE，2006。

© 2009 Microchip Technology Inc.

请注意以下有关 Microchip 器件代码保护功能的要点：

- Microchip 的产品均达到 Microchip 数据手册中所述的技术指标。
- Microchip 确信：在正常使用的情况下， Microchip 系列产品是当今市场上同类产品中最安全的产品之一。
- 目前，仍存在着恶意、甚至是非法破坏代码保护功能的行为。就我们所知，所有这些行为都不是以 Microchip 数据手册中规定的操作规范来使用 Microchip 产品的。这样做的人极可能侵犯了知识产权。
- Microchip 愿与那些注重代码完整性的客户合作。
- Microchip 或任何其他半导体厂商均无法保证其代码的安全性。代码保护并不意味着我们保证产品是 “ 牢不可破 ” 的。

代码保护功能处于持续发展中。 Microchip 承诺将不断改进产品的代码保护功能。任何试图破坏 Microchip 代码保护功能的行为均可视为违反了《数字器件千年版权法案（Digital Millennium Copyright Act）》。如果这种行为导致他人在未经授权的情况下，能访问您的软件或其他受版权保护的成果，您有权依据该法案提起诉讼，从而制止这种行为。

提供本文档的中文版本仅为了便于理解。请勿忽视文档中包含的英文部分，因为其中提供了有关 Microchip 产品性能和使用情况的有用信息。Microchip Technology Inc. 及其分公司和相关公司、各级主管与员工及事务代理机构对译文中可能存在的任何差错不承担任何责任。建议参考 Microchip Technology Inc. 的英文原版文档。

本出版物中所述的器件应用信息及其他类似内容仅为您提供便利，它们可能由更新之信息所替代。确保应用符合技术规范，是您自身应负的责任。**Microchip 对这些信息不作任何明示或暗示、书面或口头、法定或其他形式的声明或担保，包括但不限于针对其使用情况、质量、性能、适销性或特定用途的适用性的声明或担保。** Microchip 对因这些信息及使用这些信息而引起的后果不承担任何责任。如果将 Microchip 器件用于生命维持和 / 或生命安全应用，一切风险由买方自负。买方同意在由此引发任何一切伤害、索赔、诉讼或费用时，会维护和保障 Microchip 免于承担法律责任，并加以赔偿。在 Microchip 知识产权保护下，不得暗中或以其他方式转让任何许可证。

商标

Microchip 的名称和徽标组合、 Microchip 徽标、 Accuron、 dsPIC、 KEELOQ、 KEELOQ 徽标、 MPLAB、 PIC、 PICmicro、 PICSTART、 rfPIC、 SmartShun 和 UNI/O 均为 Microchip Technology Inc. 在美国和其他国家或地区的注册商标。

FilterLab、 Linear Active Thermistor、 MXDEV、 MXLAB、 SEEVAL、SmartSensor 和 The Embedded Control Solutions Company 均为 Microchip Technology Inc. 在美国的注册商标。

Analog-for-the-Digital Age、 Application Maestro、 CodeGuard、 dsPICDEM、 dsPICDEM.net、 dsPICworks、 dsSPEAK、 ECAN、 ECONOMONITOR、 FanSense、 In-Circuit Serial Programming、 ICSP、 ICEPIC、 Mindi、 MiWi、 MPASM、 MPLAB Certified 徽标、 MPLIB、 MPLINK、 mTouch、 nanoWatt XLP、 PICkit、 PICDEM、 PICDEM.net、 PICtail、 PIC^{32} 徽标、 PowerCal、 PowerInfo、 PowerMate、 PowerTool、 REAL ICE、 rfLAB、 Select Mode、 Total Endurance、 TSHARC、 WiperLock 和 ZENA 均为 Microchip Technology Inc. 在美国和其他国家或地区的商标。

SQTP 是 Microchip Technology Inc. 在美国的服务标记。

在此提及的所有其他商标均为各持有公司所有。

© 2009, Microchip Technology Inc. 版权所有。

QUALITY MANAGEMENT SYSTEM
CERTIFIED BY DNV
ISO/TS 16949:2002

Microchip 位于美国亚利桑那州 Chandler 和 Tempe 与位于俄勒冈州 Gresham 的全球总部、设计和晶圆生产厂及位于美国加利福尼亚州和印度的设计中心均通过了 ISO/TS-16949:2002 认证。公司在 PIC® MCU 与 dsPIC® DSC、 KEELOQ® 跳码器件、串行 EEPROM、单片机外设、非易失性存储器和模拟产品方面的质量体系流程均符合 ISO/TS-16949:2002。此外， Microchip 在开发系统的设计和生产方面的质量体系也已通过了 ISO 9001:2000 认证。

全球销售及服务网点

美洲

公司总部 Corporate Office
2355 West Chandler Blvd.
Chandler, AZ 85224-6199
Tel: 1-480-792-7200
Fax: 1-480-792-7277
技术支持：
http://support.microchip.com
网址：www.microchip.com

亚特兰大 Atlanta
Duluth, GA
Tel: 678-957-9614
Fax: 678-957-1455

波士顿 Boston
Westborough, MA
Tel: 1-774-760-0087
Fax: 1-774-760-0088

芝加哥 Chicago
Itasca, IL
Tel: 1-630-285-0071
Fax: 1-630-285-0075

克里夫兰 Cleveland
Independence, OH
Tel: 216-447-0464
Fax: 216-447-0643

达拉斯 Dallas
Addison, TX
Tel: 1-972-818-7423
Fax: 1-972-818-2924

底特律 Detroit
Farmington Hills, MI
Tel: 1-248-538-2250
Fax: 1-248-538-2260

科科莫 Kokomo
Kokomo, IN
Tel: 1-765-864-8360
Fax: 1-765-864-8387

洛杉矶 Los Angeles
Mission Viejo, CA
Tel: 1-949-462-9523
Fax: 1-949-462-9608

圣克拉拉 Santa Clara
Santa Clara, CA
Tel: 408-961-6444
Fax: 408-961-6445

加拿大多伦多 Toronto
Mississauga, Ontario,
Canada
Tel: 1-905-673-0699
Fax: 1-905-673-6509

亚太地区

亚太总部 Asia Pacific Office
Suites 3707-14, 37th Floor
Tower 6, The Gateway
Harbour City, Kowloon
Hong Kong
Tel: 852-2401-1200
Fax: 852-2401-3431

中国 – 北京
Tel: 86-10-8528-2100
Fax: 86-10-8528-2104

中国 – 成都
Tel: 86-28-8665-5511
Fax: 86-28-8665-7889

中国 – 香港特别行政区
Tel: 852-2401-1200
Fax: 852-2401-3431

中国 – 南京
Tel: 86-25-8473-2460
Fax: 86-25-8473-2470

中国 – 青岛
Tel: 86-532-8502-7355
Fax: 86-532-8502-7205

中国 – 上海
Tel: 86-21-5407-5533
Fax: 86-21-5407-5066

中国 – 沈阳
Tel: 86-24-2334-2829
Fax: 86-24-2334-2393

中国 – 深圳
Tel: 86-755-8203-2660
Fax: 86-755-8203-1760

中国 – 武汉
Tel: 86-27-5980-5300
Fax: 86-27-5980-5118

中国 – 厦门
Tel: 86-592-238-8138
Fax: 86-592-238-8130

中国 – 西安
Tel: 86-29-8833-7252
Fax: 86-29-8833-7256

中国 – 珠海
Tel: 86-756-321-0040
Fax: 86-756-321-0049

台湾地区 – 高雄
Tel: 886-7-536-4818
Fax: 886-7-536-4803

台湾地区 – 台北
Tel: 886-2-2500-6610
Fax: 886-2-2508-0102

台湾地区 – 新竹
Tel: 886-3-572-9526
Fax: 886-3-572-6459

亚太地区

澳大利亚 Australia - Sydney
Tel: 61-2-9868-6733
Fax: 61-2-9868-6755

印度 India - Bangalore
Tel: 91-80-3090-4444
Fax: 91-80-3090-4080

印度 India - New Delhi
Tel: 91-11-4160-8631
Fax: 91-11-4160-8632

印度 India - Pune
Tel: 91-20-2566-1512
Fax: 91-20-2566-1513

日本 Japan - Yokohama
Tel: 81-45-471- 6166
Fax: 81-45-471-6122

韩国 Korea - Daegu
Tel: 82-53-744-4301
Fax: 82-53-744-4302

韩国 Korea - Seoul
Tel: 82-2-554-7200
Fax: 82-2-558-5932 或
82-2-558-5934

马来西亚 Malaysia - Kuala Lumpur
Tel: 60-3-6201-9857
Fax: 60-3-6201-9859

马来西亚 Malaysia - Penang
Tel: 60-4-227-8870
Fax: 60-4-227-4068

菲律宾 Philippines - Manila
Tel: 63-2-634-9065
Fax: 63-2-634-9069

新加坡 Singapore
Tel: 65-6334-8870
Fax: 65-6334-8850

泰国 Thailand - Bangkok
Tel: 66-2-694-1351
Fax: 66-2-694-1350

欧洲

奥地利 Austria - Wels
Tel: 43-7242-2244-39
Fax: 43-7242-2244-393

丹麦 Denmark-Copenhagen
Tel: 45-4450-2828
Fax: 45-4485-2829

法国 France - Paris
Tel: 33-1-69-53-63-20
Fax: 33-1-69-30-90-79

德国 Germany - Munich
Tel: 49-89-627-144-0
Fax: 49-89-627-144-44

意大利 Italy - Milan
Tel: 39-0331-742611
Fax: 39-0331-466781

荷兰 Netherlands - Drunen
Tel: 31-416-690399
Fax: 31-416-690340

西班牙 Spain - Madrid
Tel: 34-91-708-08-90
Fax: 34-91-708-08-91

英国 UK - Wokingham
Tel: 44-118-921-5869
Fax: 44-118-921-5820

02/04/09

© 2009 Microchip Technology Inc.