



RF-BM-4044B4 硬件规格书

(CC2640R2FRSM)

深圳市信驰达科技有限公司
更新日期：2020 年 05 月 15 日

目录

● 选型表	3
● 概述	4
● 模块参数	5
● 模块尺寸与引脚定义	5
● 硬件设计注意事项	7
● 常见问题	8
➢ 传输距离不理想	8
➢ 易损坏—异常损坏	8
➢ 误码率太高	8
● 回流焊条件	8
● 静电放电警示	9
● 版本更新记录	9
● 联系我们	10



RF-STAR

● 选型表

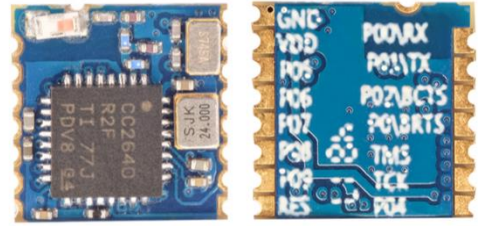
芯片型号	内核	FLASH (KB)	RAM (KB)	发射功率 (dBm)	模块型号	天线形式	模块尺寸 (mm)	通信距离 (M)	模块照片 (点击可访问)
CC2640 R2FRSM	M3	128	28	2	RF-BM-4044B2	PCB	11.2*16.6	300	
					RF-BM-4044B3	IPEX	11.2*15.2	500	
					RF-BM-4044B4	CHIP	8*8	150	
CC2640 R2FRGZ	M3	128	28	5	RF-BM-4077B1	PCB	17*23.5	500	
					RF-BM-4077B2	PCB	17*23.5	500	
CC2642R	M4F	352	80	5	RF-BM-2642B1	PCB	17*23.5	500	
CC2652R	M4F	352	80	5	RF-BM-2652B1	PCB	17*23.5	BLE: 500 Zigbee: 300	
CC2652P	M4F	352	80	20	RF-BM-2652P1	Half-hole	16.4*25	BLE: 500 Zigbee: 300	
CC1352R	M4F	352	80	5/14	RF-TI1352B1	IPEX	16.8*26.5	BLE: 500 Zigbee: 300 868MHz: 1500	
CC1352P	M4F	352	80	20	RF-TI1352P1	Half-hole	16.4*25	BLE: 500 Zigbee: 300 868MHz: 1500	

注:

- 1、通信距离为以模块最大发射功率在晴朗天气下空旷无干扰环境下测试所得最远距离。
- 2、点击图片可跳转至购买链接。

● 概述

深圳信驰达蓝牙模块 RF-BM-4044B4 是基于 TI 公司 CC2640R2FRSM 芯片研发的低功耗蓝牙 (BLE) 射频模块，可广泛应用于短距离无线通信领域。具有功耗低、体积小、传输距离远、抗干扰能力强等特点。



该模块可用于开发基于蓝牙5 (BLE 5, 低功耗蓝牙) 的消费类电子产品, 手机外设产品等, 为客户产品与智能移动设备通讯提供快速的 BLE 解决方案。软件开发可参考基于 TI 提供的标准 BLE 协议栈, LIB 底层库以及 API 调用接口。源码级 profile, APP Demo 等资料可有效缩短开发投入时间。系列模块含有一个32 位 ARM Cortex-M3 处理器, 与主处理器工作频率同为 24 MHz, 具有丰富的外设功能集, 包括一个独特的超低功耗传感器控制器, 适用于在系统处于休眠模式时连接外部传感器和/或自主采集模拟和数字数据。

凭此特性, 信驰达 CC2640R2F 系列模块成为注重电池使用寿命、小型尺寸和简便实用性的各类应用的理想选择。Bluetooth 低能耗控制器嵌入在 ROM 中, 并在 ARM Cortex-M0 处理器上单独运行, 此架构可改善整体系统性能和功耗, 并释放闪存以供应用。

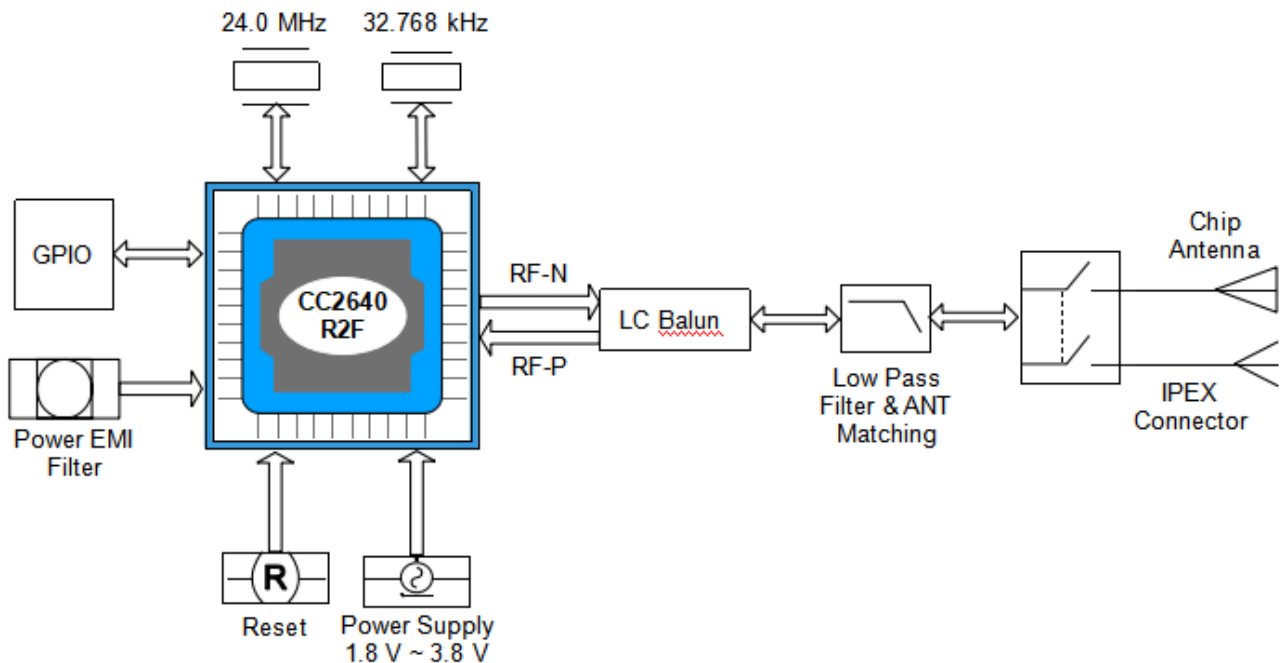


图 1. RF-BM-4044B4原理框图

● 模块参数

表 1. RF-BM-4044B4参数

芯片型号	CC2640R2FRSM
工作电压	1.8 ~ 3.8 V, 推荐为 3.3V
工作频段	2402 MHz ~ 2480 MHz
发射功率	-21 dBm ~ +2 dBm (正常为0 dBm输出)
接收灵敏度	- 97 dBm
RAM	20 KB
FLASH	128 KB
GPIO数量	10 个
晶振频率	24 MHz, 32.768 KHz
封装方式	SMT 封装 (邮票半孔)
通讯接口	UART, I2S, I2C, SPI, ADC
模块尺寸	8 x 8 mm (超小尺寸)
工作温度	-40 °C ~ +85 °C
储存温度	-40 °C ~ +125 °C

● 模块尺寸与引脚定义

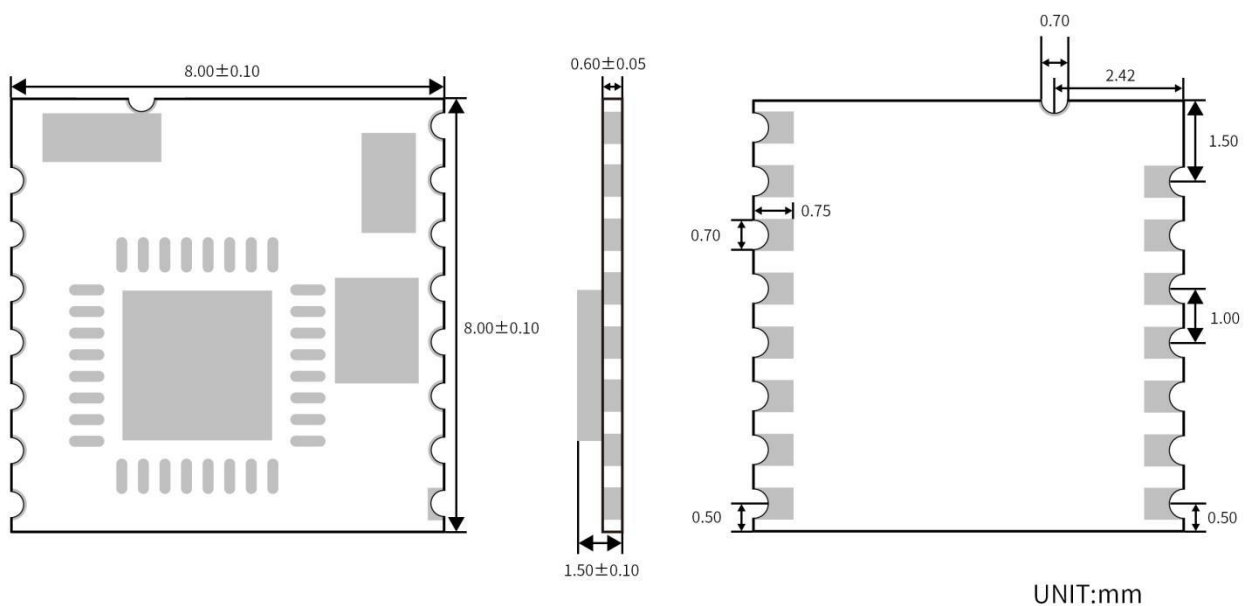


图 2. RF-BM-4044B4尺寸图

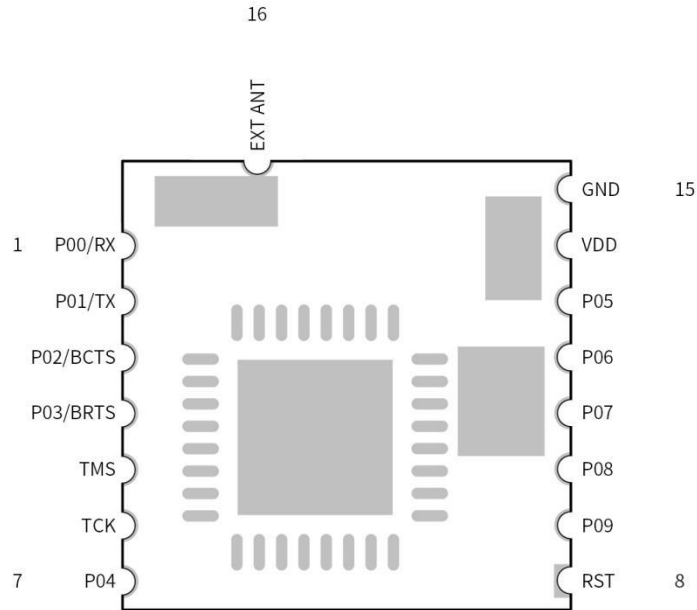


图 3. RF-BM-4044B4 引脚定义图

表 2. RF-BM-4044B4 模块引脚定义

引脚序号	名称	功能	备注
1	P00\RX	I/O	
2	P01\TX	I/O	
3	P02\BCTS	I/O	
4	P03\BRTS	I/O	
5	TMS	—	接 XDS 仿真器 TMS
6	TCK	—	接 XDS 仿真器 TCK
7	P04	I/O	
8	RES	复位脚	低电平有效，不需要外部 RC 电路
9	P09	I/O	
10	P08	I/O	
11	P07	I/O	
12	P06	I/O	
13	P05	I/O	
14	VCC	电源正极输入	模块电源，可接电池(1.8 V to 3.8 V)，推荐为 3.3V
15	GND	模块地	模块地
16	EXT ANT	—	外接天线输出（可选，默认使用陶瓷天线）

● 硬件设计注意事项

- 1、推荐使用直流稳压电源对模块进行供电，电源纹波系数尽量小，模块需可靠接地；请注意电源正负极的正确连接，如反接可能会导致模块永久性损坏；
- 2、请检查供电电源，确保在推荐供电电压之间，如超过最大值会造成模块永久性损坏；请检查电源稳定性，电压不能大幅频繁波动；
- 3、在针对模块设计供电电路时，往往推荐保留 30% 以上余量，有利于整机长期稳定地工作；模块应尽量远离电源、变压器、高频走线等电磁干扰较大的部分；
- 4、高频数字走线、高频模拟走线、电源走线必须避开模块下方，若实在不得已需要经过模块下方，假设模块焊接在 Top Layer，在模块接触部分的 Top Layer 铺地铜（全部铺铜并良好接地），必须靠近模块数字部分并走线在 Bottom Layer；
- 5、假设模块焊接或放置在 Top Layer，在 Bottom Layer 或者其他层随意走线也是错误的，会在不同程度影响模块的杂散以及接收灵敏度；
- 6、假设模块周围有存在较大电磁干扰的器件也会极大影响模块的性能，跟据干扰的强度建议适当远离模块，若情况允许可以做适当的隔离与屏蔽；
- 7、假设模块周围有存在较大电磁干扰的走线（高频数字、高频模拟、电源走线）也会极大影响模块的性能，跟据干扰的强度建议适当远离模块，若情况允许可以做适当的隔离与屏蔽；
- 8、通信线若使用5V电平，必须使用电平转换电路；
- 9、尽量远离部分物理层亦为 2.4 GHz 频段的TTL 协议，例如：USB 3.0；
- 10、模块天线布局请参考下图：

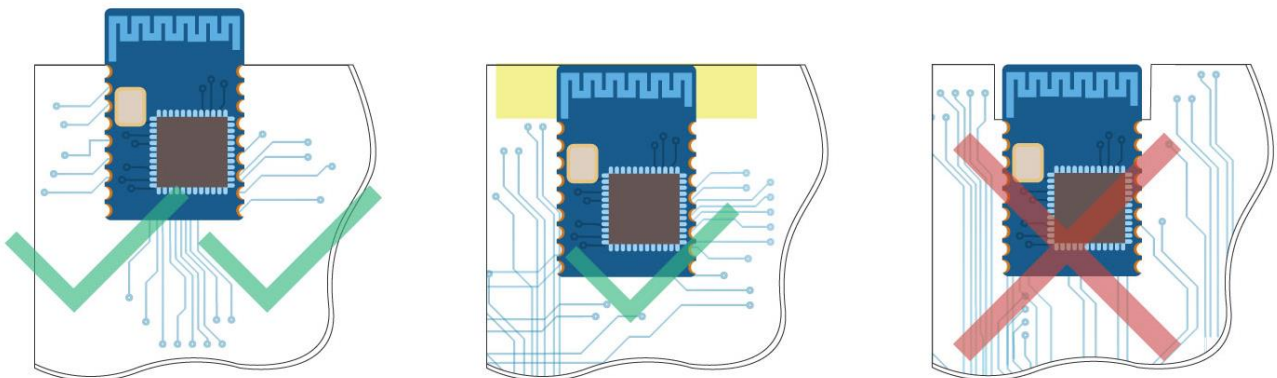


图 4. 布局建议

● 常见问题

➤ 传输距离不理想

- 1、当存在直线通信障碍时，通信距离会相应的衰减；温度、湿度，同频干扰，会导致通信丢包率提高；地面吸收、反射无线电波，靠近地面测试效果较差；
- 2、海水具有极强的吸收无线电波能力，故海边测试效果差；
- 3、天线附近有金属物体，或放置于金属壳内，信号衰减会非常严重；
- 4、功率寄存器设置错误、空中速率设置过高（空中速率越高，距离越近）；
- 5、室温下电源低压低于推荐值，电压越低发功率越小；
- 6、使用天线与模块匹配程度较差或天线本身品质问题。

➤ 易损坏—异常损坏

- 1、请检查供电电源，确保在推荐供电电压之间，如超过最大值会造成模块永久性损坏；请检查电源稳定性，电压不能大幅频繁波动；
- 2、请确保安装使用过程防静电操作，高频器件静电敏感性；
- 3、请确保安装使用过程湿度不宜过高，部分元件为湿度敏感器件；如果没有特殊需求不建议在过高、过低温度下使用。

➤ 误码率太高

- 1、附近有同频信号干扰，远离干扰源或者修改频率、信道避开干扰；
- 2、电源不理想也可能造成乱码，务必保证电源的可靠性；
- 3、延长线、馈线品质差或太长，也会造成误码率偏高。

● 回流焊条件

- 1、加热方法：常规对流或 IR 对流；
- 2、允许回流焊次数：2 次，基于以下回流焊(条件)(见图 5)；
- 3、温度曲线：回流焊应按照下列温度曲线(见图 5)；
- 4、最高温度：245°C。

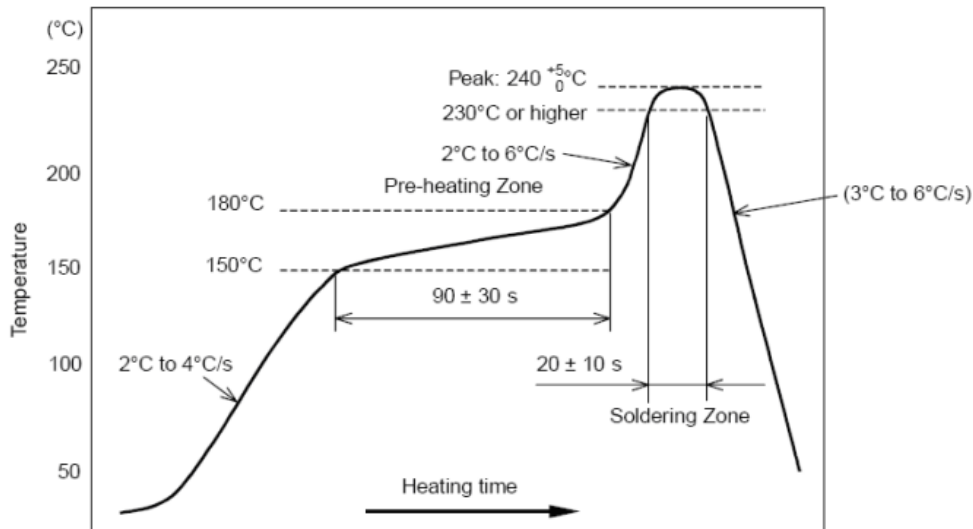


图 5. 部件的焊接耐热性温度曲线(焊接点)

● 静电放电警示

模块会因静电释放而被损坏，RF-star 建议所有模块应在以下 3 个预防措施下处理：

- 1、必须遵循防静电措施，不可以裸手拿模块。
- 2、模块必须放置在能够预防静电的放置区。
- 3、在产品设计时应该考虑高电压输入或者高频输入处的防静电电路。

静电可能导致的结果为细微的性能下降到整个设备的故障。由于非常小的参数变化都可能导致设备不符合其认证要求的值限，从而模块会更容易受到损害。

● 版本更新记录

版本号	文档日期	更新内容
V1.0	2018/01/23	第一次发布
V1.1	2018/04/25	更新模块参数 更新模块照片
V1.1	2018/08/02	更新公司地址
V1.1	2018/12/10	更新参数说明
V1.1	2020/01/13	更换模块封装尺寸图
V1.1	2020/01/19	添加模块选型图
V1.1	2020/05/15	更新选型表

● 联系我们

深圳市信驰达科技有限公司

Shenzhen RF-star Technology Co., Ltd.

Tel(Sales): 0755-8632 9829

Tel(FAE): 0755-3695 3756

E-mail: sales@szrfstar.com

Web: www.szrfstar.com

地址：深圳市南山区高新园科技南一道创维大厦 C 座 601 室

Add: Room 601,Block C,Skyworth Building,Nanshan High-Tech Park,Shenzhen.





低功耗蓝牙 (BLE) 模块及蓝牙 5.0 从协议 (CC2640R2FRSM)

协议版本: V4.50u(透传)



深圳市信驰达科技有限公司
更新日期: 2020 年 10 月 12 日

前言

如何快速低成本地开发智能手机新外设

—论低功耗蓝牙技术在智能移动设备中的应用—

USB 协议的产生，让个人电脑的外设如雨后春笋般地涌现。同样，做为智能手机最新开放的低功耗蓝牙 (BLE) 无线应用技术，也有异曲同工之妙。BLE 技术给电子产品桥接智能手机提供了可能。相对 Wi-Fi, Bluetooth 2.0 等无线技术，有着能耗低，连接迅速，通讯距离更远等优势，让智能手机的外围电子设备有了更开阔的发展前景。

做为国际蓝牙联盟 (BT-SIG) 成员之一，德州仪器 (TI) 于 2015 年强势推出 CC2640R2F 系列单芯片 (SOC) 低功耗蓝牙收发器，强大的 ARM® Cortex® -M3 内核，最强优势在于丰富的外围：最多达 31 个 IO (7mm*7mm 封装)、UART、SPI、PWM、ADC、analog comparator、op-amp，超宽的工作电压 (1.8V - 3.8V)，极低的能耗 (0.1μA)。

为方便低功耗蓝牙 (BLE) 应用技术在各个行业产品中的移植和使用，在 2018 年中国 TI 无线领域战略合作伙伴信驰达科技特别推出了 5.0 的低功耗蓝牙透传模块 **RF-BM-4044B2**、**RF-BM-4044B3**、**RF-BM-4044B4**。

此类型模块适应市场需求于 5.0 协议栈开发，相比于 4.2 模块提高到 2 倍的通信速率、4 倍的通信距离、8 倍的广播数据传输量。并且模块做为智能手机外设的桥梁，使得主机端应用开发异常简单。在桥接模式下 (串口)，用户的现有产品或者方案配合此透传模块，能十分方便地和移动设备 (需支持蓝牙 5.0 及以下设备) 相互通讯，实现超强的智能化控制和管理。

RF-BM-4044B2、RF-BM-4044B3、RF-BM-4044B4 低功耗蓝牙模块，采用 TI 的 CC2640R2F 作为核心处理器。模块运行在 2.4 GHz ISM band，GFSK 调制方式 (高频移键控)，40 频道最高 2 MHz 的通道间隙，自定义高达 232 字节的大广播包，3 个固定的广播通道，37 个自适应自动跳频数据通道，物理层可以和经典蓝牙 RF 组合成双模设备，可调的 1MHz 和 2 MHz 间隙能更好地防止相邻频道的干扰。宽输出功率调节 (-21 dBm~+2dBm)，-97 dBm 高增益接收灵敏度。

此模块的设计目的是迅速桥接电子产品和智能移动设备，可广泛应用于有此需求的各种电子设备，如仪器仪表、物流跟踪、健康医疗、智能家居、运动计量、汽车电子、休闲玩具等。**随着安卓 4.3 智能设备对 BLE 技术的集成，智能手机标配 BLE 必将成为时尚，手机外设的市场需求将成级数倍增。**用户可借此模块，以最短的开发周期整合现有方案或产品，以最快的速度占领市场，同时为企业的发展注入崭新的技术力量。

版本更新记录

版本号	文档日期	更新内容
V4.50	2019/02/25	<ul style="list-style-type: none"> ✓ 第一次发布 ✓ 支持 AT 指令自定义 232 字节的大广播包数据内容，支持实时更新，掉电保存； ✓ 支持手动调整物理层带宽，默认 1M； ✓ 支持 AT 指令调整发射功率，修改广播间隔； ✓ 支持 AT 指令附加自定义广播数据，支持实时更新，掉电保存； ✓ 支持 AT 指令自定义设备识别码，掉电保存，会包含在广播数据中。 ✓ 如果是 TTM:开头的包，视为 AT 指令，否则做透传数据处理。如果以 TTM:开头，而后面的字串无法解析，输出"TTM:ERP\r\n0" ✓ 默认波特率为 115200，可自修改为高速波特率，掉电保存，最低波特率支持 4800； ✓ 发射功率，产品识别码，自定义广播内容，广播周期服务通道设置可以通过 APP 对这些系统参数进行设置； ✓ 从 TX 输出蓝牙正常断线和超时断线字串提示； ✓ 自定义广播内容掉电保存功能。 ✓ 连接指示反馈设备的 MTU 大小和连接间隔；
V4.50	2019/03/19	<ul style="list-style-type: none"> ✓ 修复名称修改只到 15 字节的问题 ✓ 修复防劫持秘钥 20S 断开功能
V4.50	2019/04/22	<ul style="list-style-type: none"> ✓ 修正文档中部分描述错误 ✓ 加入基本的硬件规格说明；
V4.50	2019/06/20	<ul style="list-style-type: none"> ✓ 更新文档速率和发包说明 ✓ 更新硬件规格说明；
V4.50	2019/07/05	<ul style="list-style-type: none"> ✓ 新增 AT 指令查询版本号； ✓ 新增 AT 指令修改密码，删除密码；
V4.50	2019/11/18	<ul style="list-style-type: none"> ✓ 新增 AT 指令设定数据延时
V4.50	2020/10/12	<ul style="list-style-type: none"> ✓ 新增 AT 指令设定 Service UUID ✓ 新增外围参考设计 ✓ 新增功耗测试截图

注：

- 1、文档会不定期优化更新，在使用此文档前，请确保是最新版本；
- 2、获取最新协议或文档，请到信驰达科技官方网址下载：

目录

前言.....	2
版本更新记录.....	3
目录.....	4
● 概述.....	6
● 工作模式示意图.....	8
● 封装尺寸及脚位定义.....	9
➢ RF-BM-4044B2、RF-BM-4044B3.....	9
➢ RF-BM-4044B4.....	12
● 串口透传协议说明(桥接模式).....	14
● 串口 AT 指令.....	16
➢ 连接间隔设定.....	16
➢ 获取模块名称.....	16
➢ 模块重命名.....	16
➢ 获取波特率.....	17
➢ 波特率设定.....	17
➢ 获取物理地址 MAC.....	17
➢ 设置模块 MAC 地址.....	17
➢ 获取模块版本号.....	17
➢ 获取模块连接密码.....	18
➢ 设置模块连接密码.....	18
➢ 模块复位—上电复位.....	18
➢ 模块复位—浅恢复.....	18
➢ 模块复位—深度恢复.....	18
➢ EN 脚内部使能.....	18
➢ 广播周期设定.....	19
➢ 修改透传服务 UUID.....	19
➢ 附加自定义广播内容.....	20
➢ 定义产品识别码.....	20
➢ 发射功率设定.....	20
➢ 自定义大广播包数据.....	21
➢ 设置物理层数据率.....	21
➢ 数据延时设定.....	21

AT 指令表.....	22
● BLE 协议说明(APP 接口).....	24
➢ 蓝牙数据通道【服务 UUID: 0xFFE5】.....	24
➢ 串口数据通道【服务 UUID: 0xFFE0】.....	24
➢ 模块参数设置【服务 UUID: 0xFF90】.....	24
➢ 设备信息【服务 UUID: 0x180A】.....	27
➢ 防劫持密钥【服务 UUID: 0xFFC0】.....	27
● 广播数据设置.....	30
● 系统复位与恢复.....	31
● 用 APP 测试透传功能.....	32
● 主机参考代码(透传).....	34
● IOS APP 编程参考.....	35
● 联系我们.....	37
附录 A: BLE 模块应用方案提示.....	38
附录 B: BLE 模块硬件规格说明.....	39
附录 C: 功耗测试截图.....	42

● 概述

模块暂时只工作在桥接模式(透传模式)。

模块启动后会自动进行广播,已打开特定 APP 的手机会对其进行扫描和对接,成功之后便可以通过 BLE 协议对其进行监控。

桥接模式下,用户 CPU 可以通过模块的通用串口和移动设备进行双向通讯,用户也可以通过特定的串口 AT 指令,对某些通讯参数进行管理控制。用户数据的具体含义由上层应用程序自行定义。移动设备可以通过 APP 对模块进行写操作,写入的数据将通过串口发送给用户的 CPU。模块收到来自用户 CPU 串口的数据包后,将自动转发给移动设备。此模式下的开发,用户必须负责主 CPU 的代码设计,以及智能移动设备端 APP 代码设计。

主要特点:

- 1、5.0 协议栈开发,与 4.2 相比有 2 倍的速率,4 倍的通信距离,8 倍的广播包内容长度;
- 2、支持手动调整物理层带宽,提升通信距离和速率;
- 3、连接指示设备的 MTU 大小,灵活分配数据包大小;
- 4、此模块使用简单,无需任何蓝牙协议栈应用经验;
- 5、用户接口使用通用串口设计,全双工双向通讯,最低波特率支持 4800bps;
- 6、默认 20ms 连接间隔,连接快速;
- 7、支持 AT 指令软件复位模块,获取 MAC 地址;
- 8、支持 AT 指令修改 Service UUID;
- 9、串口数据包长度,可以是 1024Byte 以下(含 1024)的任意长度(大包自动分发);
- 10、支持 AT 指令调整蓝牙连接间隔,控制不同的转发速率(动态功耗调整);
- 11、支持 AT 指令调整发射功率、修改广播间隔、自定义广播数据、自定义设备识别码、修改串口波特率,修改模块名;
- 12、高速透传转发(主从模块之间),可以达到最快 16K/S 的稳定速率;
- 13、支持移动设备 APP 修改模块名称、修改串口波特率、产品识别码、自定义广播内容、广播周期;
- 14、支持移动设备 APP 对模块进行远程复位,设置发射功率;
- 15、支持移动设备 APP 调节蓝牙连接间隔(动态功耗调整);
- 16、支持连接状态、广播状态提示脚;
- 17、支持移动设备 APP 对模块进行远程恢复初始化;

18、支持从 TX 串口获取蓝牙连接状态(连接，正常断线和超时断线)字串提示；

19、极低功耗的待机模式，CC2640R2F 芯片官方数据睡眠电流 0.1 μ A，模块实测功耗如下：

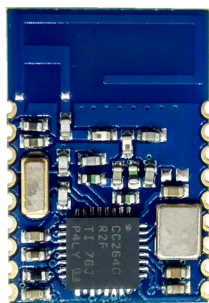
事件	平均电流(小壁虎测量*3 默认)		测试条件/备注
	开启 EN 内部上拉	关闭 EN 内部上拉	
模块睡眠功耗	2.02 μ A		EN 为高电平
广播	808.64 μ A	642.05 μ A	广播周期 200 ms (默认)
广播	328.62 μ A	108.76 μ A	广播周期 1000 ms
广播	231.95 μ A	25.71 μ A	广播周期 5000 ms
连接事件	710.97 μ A	497.41 μ A	连接间隔 30 ms
模块接收串口数据并 发送给 APP	3.7mA	3.5mA	(20 bytes,10 次/秒) 连接间隔 100 ms
模块接收 APP 数据并 发送给 MCU	3.99mA	3.78mA	(20 bytes,10 次/秒) 连接间隔 100 ms

*1 注：官方测试方式：在电源回路上串一个 10R 的电阻，使用示波器截取压降波形，进行积分计算。

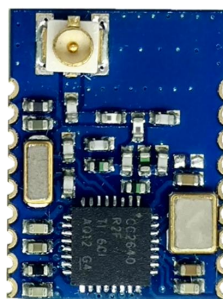
*2 注：万用表测试方式：用万用表 μ A 或 mA 档串在电池与模块之间查看显示值。测试电压为 3.3V。

*3 注：小壁虎测试方式：把小壁虎（Silicon Labs）工具接入到模块电源回路中，打开官方软件 Simplicity Studio 进行监测功耗波形数据，自动积分计算功耗值。

以上数据为信驰达模块 **RF-BM-4044B2** 抽样实测数据，仅供参考。如果希望得到更低功耗，可适当增大连接间隔或者广播周期，详见[《模块参数设置》](#)和[《串口 AT 指令》](#)相关章节。



[RF-BM-4044B2](#)



[RF-BM-4044B3](#)



[RF-BM-4044B4](#)

● 工作模式示意图

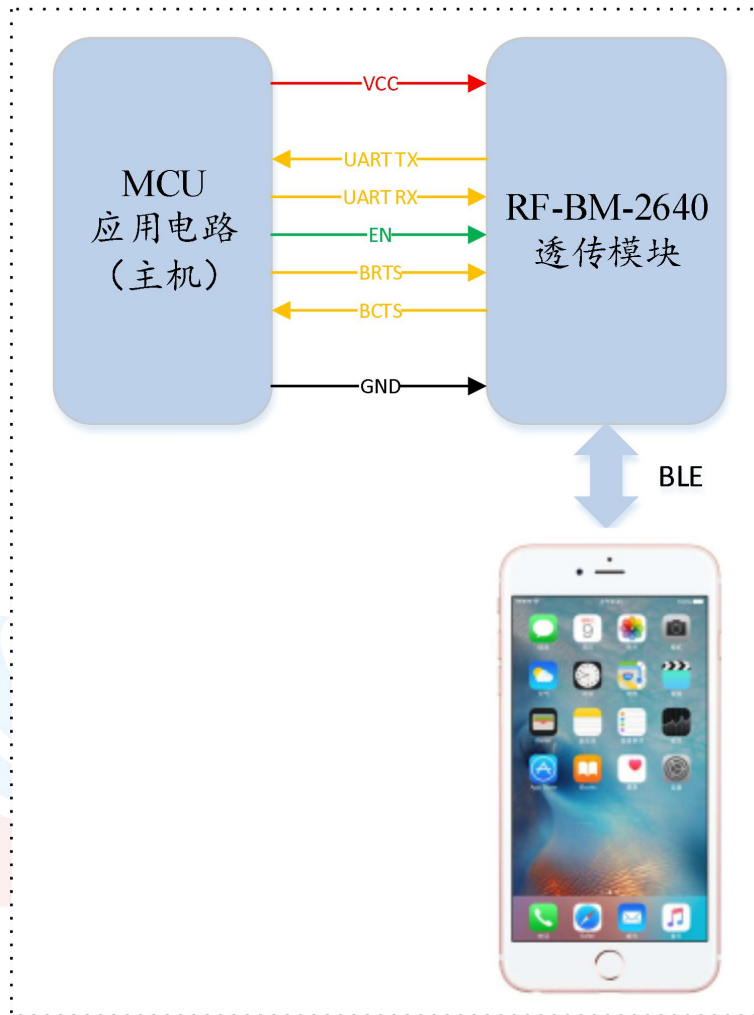


图1 模块桥接模式示意图

注：为避免用户 MCU 的 IO 和模块 IO 的输出电平差异导致大电流，建议在模块的输出信号线 TX，BCTS 上串入一小额隔离电阻。

● 封装尺寸及脚位定义

➤ RF-BM-4044B2、RF-BM-4044B3

RF-BM-4044B2 和 RF-BM-4044B3 是引脚兼容的,两者的区别是 RF-BM-4044B2 使用的是板载天线,而 RF-BM-4044B3 使用的是带 IPEX 天线座的外置天线,RF-BM-4044B2 的尺寸比 RF-BM-4044B3 的尺寸稍微长一些(天线部分)。

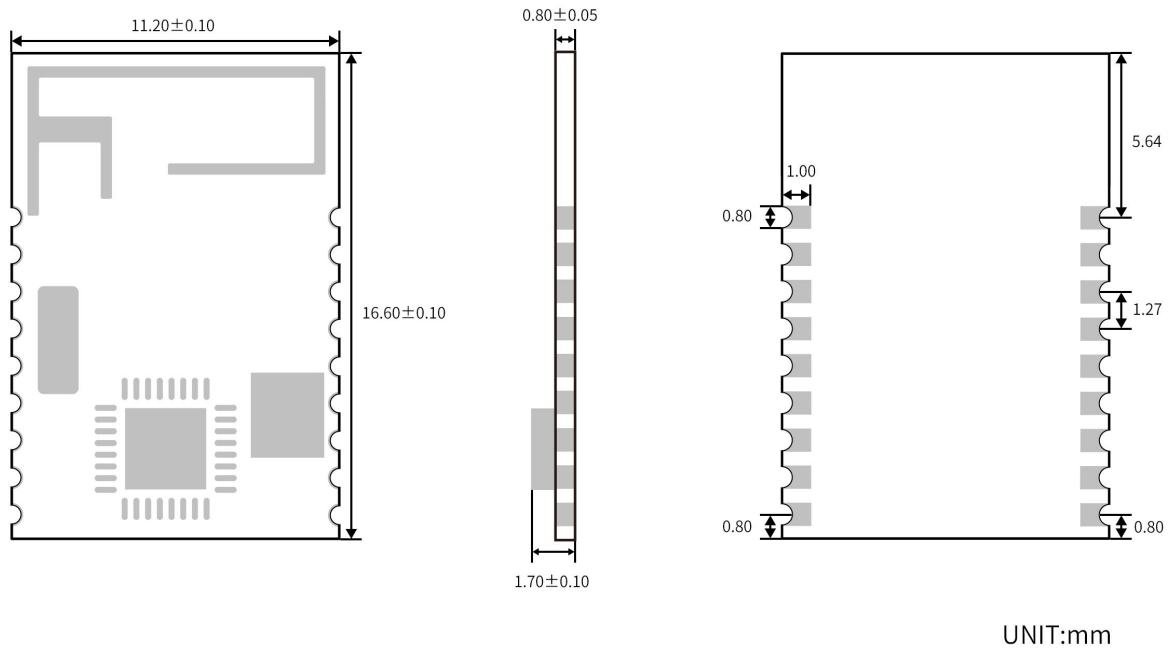


图2 RF-BM-4044B2尺寸图

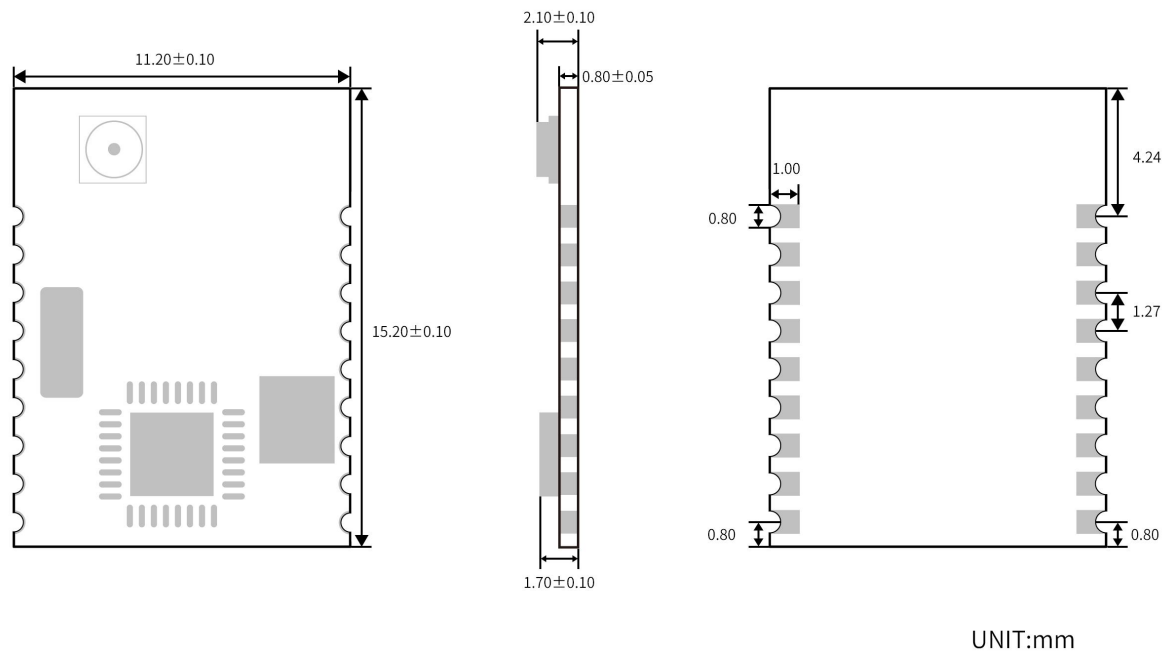
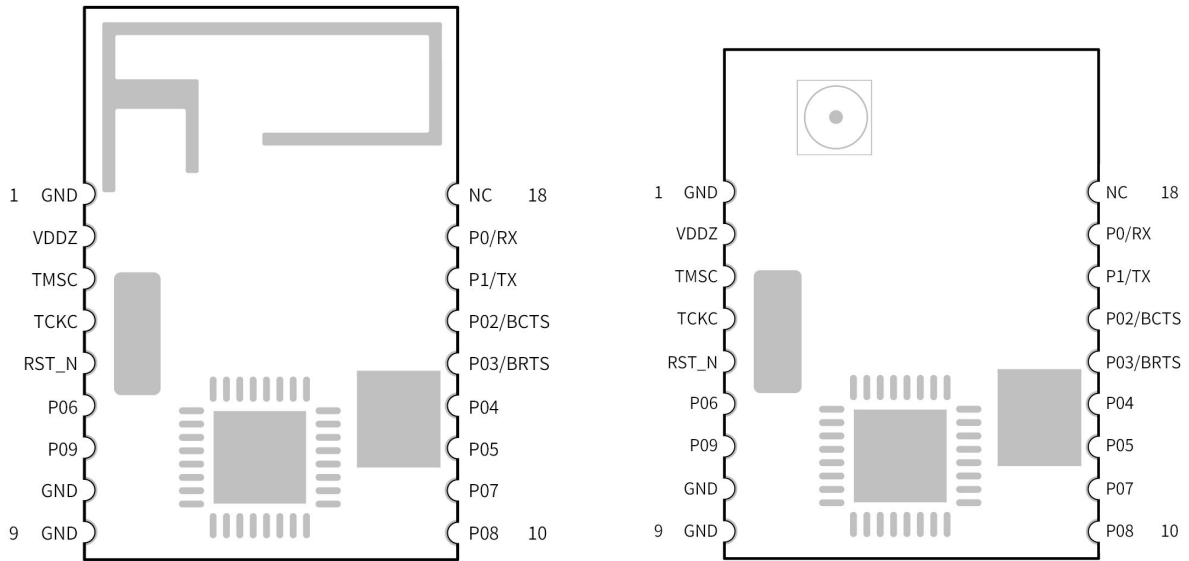
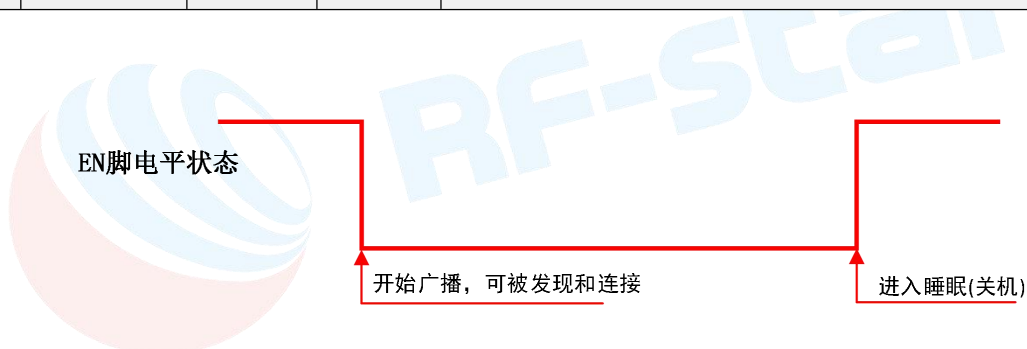


图3 RF-BM-4044B3尺寸图


图4 模块引脚图
引脚定义表

模块脚位序号	模块脚位名称	芯片脚位名称	输入/输出	说明
Pin1	GND	—	—	模块地 GND
Pin2	VCC	—	—	模块电源正极 1.8V~3.8V
Pin3	TMS	—	I/O	接 XDS 仿真器 TMS
Pin4	TCK	—	I/O	接 XDS 仿真器 TCK
Pin5	RST	—	—	复位输入脚，低电平有效，无内部上拉
Pin6	EN	P06	I	模块使能控制线(低电平有效) 0: 模块开始广播，直到连接到移动设备 1: 无论模块当前状态，立即进入完全睡眠状态 (0.1μA)
Pin7	—	P09	I/O	
Pin8	GND	—	—	模块地 GND
Pin9	GND	—	—	模块地 GND
Pin10	RESTORE	P08	I/O	恢复出厂设置触发 模块上电后保持此引脚低电平 5s，系统会恢复部分参数（浅恢复），若保持 20s 以上则将会恢复全部参数（深度恢复）（见《系统复位与恢复》系统复位与恢复章节）
Pin11	—	P07	I/O	
Pin12	链路指示	P05	O	链路指示 0: 蓝牙已连接 1: 蓝牙未连接

Pin13	广播指示	P04	O	广播指示 0: 开启广播 1: 关闭广播
Pin14	BRTS	P03	I	作为数据发送请求(用来唤醒模块) 0: 主机有数据发送, 模块将等待接收来自主机的数据, 此时模块不睡眠 1: 主机无数据发送, 或主机数据发送完毕之后, 应该将此信号线置 1
Pin15	BCTS	P02	O	数据输入信号(用来唤醒主机, 可选) 0: 模块有数据发送到主机, 主机接收模块数据 1: 模块无数据发送到主机, 或模块数据发送完毕之后, 会将此信号置 1
Pin16	TX	P01	O	模块串口发送端
Pin17	RX	P00	I	模块串口接收端
Pin18	NC	—	—	

说明 1:


模块广播时(此状态下, 可被发现, 被连接)有以下特性:

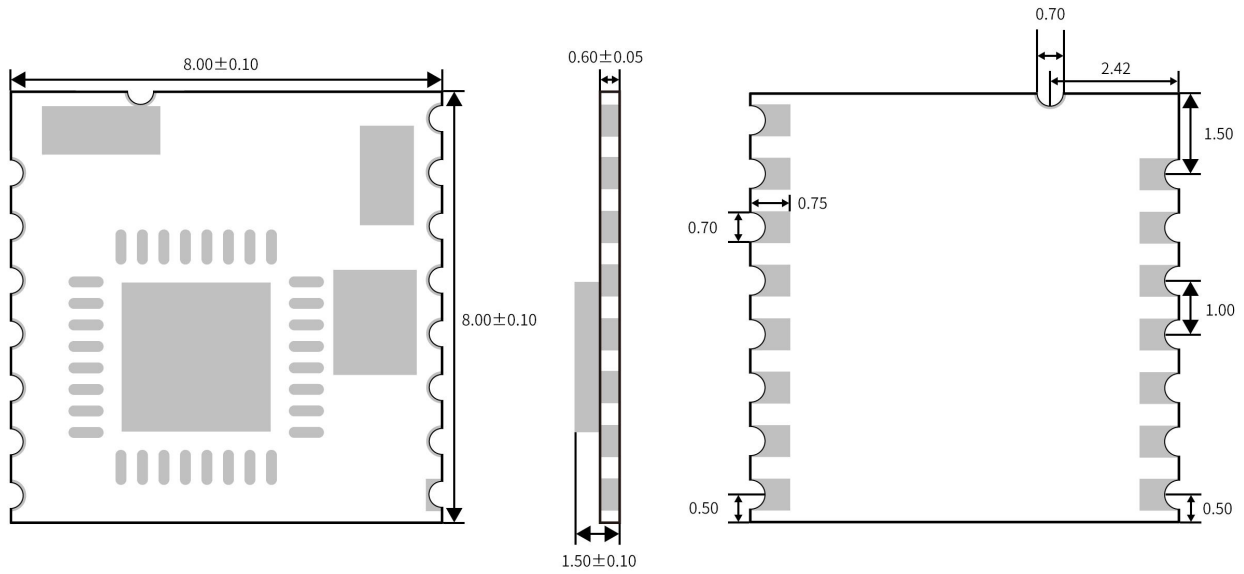
- 1、如果 EN 脚(Pin 6)被使能后(置低), 模块会保持一直广播, 直到被连接, 或者 EN 被置高。
- 2、正常断开或者超时断开后, 只要 EN 置低, 模块总会保持广播, 直到再次被连接。

说明 2:

P04 作为广播指示脚, 当模块开启广播输出低电平, 当模块关闭广播输出高电平; P05 做为链路指示脚, 当蓝牙已连接输出低电平, 当蓝牙未连接或者断开(超时断开和 APP 主动断开)后处于未连接状态时, 输出高电平。

模块状态	使能后未连接		连接		正常断开		超时断开	
	P04 提示方式	P05 提示方式	P04 提示方式	P05 提示方式	P04 提示方式	P05 提示方式	P04 提示方式	P05 提示方式
电平使能模式	低电平广播	高电平未连接	高电平未广播	低电平连接	低电平广播	高电平未连接	低电平广播	高电平未连接

➤ **RF-BM-4044B4**



UNIT:mm

图5 RF-BM-4044B4尺寸图

16

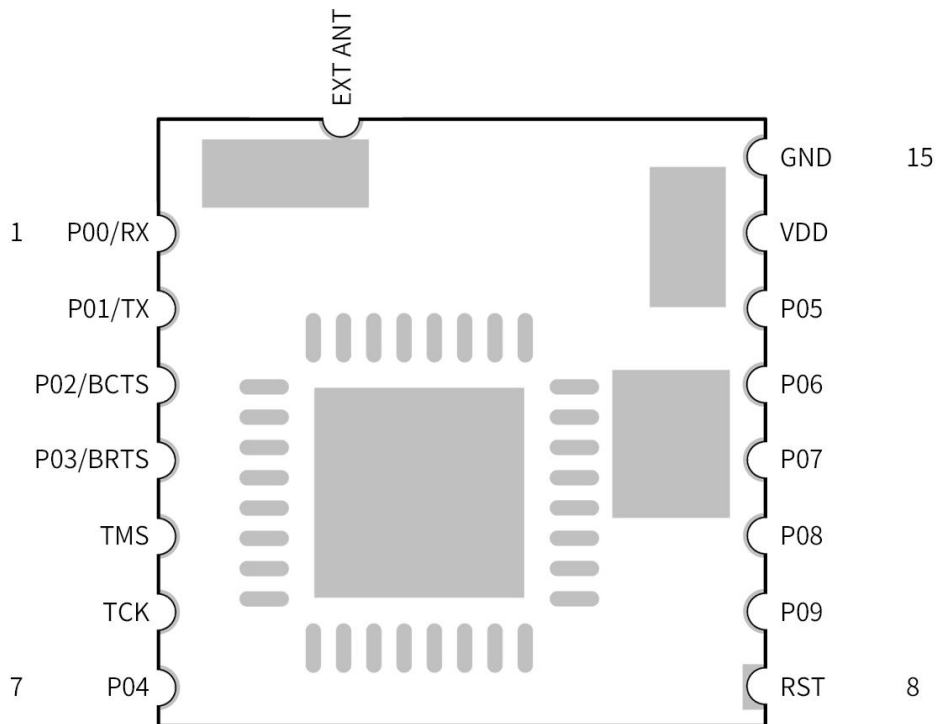


图6 模块引脚图

引脚定义表

模块脚位序号	模块脚位名称	芯片脚位名称	输入/输出	说明
Pin1	RX	P00	I	模块串口接收端
Pin2	TX	P01	O	模块串口发送端
Pin3	BCTS	P02	O	数据输入信号(用来唤醒主机, 可选) 0: 模块有数据发送到主机, 主机接收模块数据 1: 模块无数据发送到主机, 或模块数据发送完毕之后, 会将此信号置 1
Pin4	BRTS	P03	I	作为数据发送请求(用来唤醒模块) 0: 主机有数据发送, 模块将等待接收来自主机的数据, 此时模块不睡眠 1: 主机无数据发送, 或主机数据发送完毕之后, 应该将此信号线置 1
Pin5	TMS	—	I/O	接 XDS 仿真器 TMS
Pin6	TCK	—	I/O	接 XDS 仿真器 TCK
Pin7	广播指示	P04	O	广播指示 0: 开启广播 1: 关闭广播
Pin8	RES	—	—	复位输入脚, 低电平有效, 无内部上拉
Pin9	—	P09	I/O	
Pin10	RESTORE	P08	I/O	恢复出厂设置触发 模块上电后保持此引脚低电平 5s , 系统会恢复部分参数(浅恢复), 若保持 20s 以上则将会恢复全部参数(深度恢复)(见《系统复位与恢复》系统复位与恢复章节)
Pin11	—	P07	I/O	
Pin12	EN	P06	I	模块使能控制线(低电平有效) 0: 模块开始广播, 直到连接到移动设备 1: 无论模块当前状态, 立即进入完全睡眠状态 (0.1μA)
Pin13	链路指示	P05	O	链路指示 0: 蓝牙已连接 1: 蓝牙未连接
Pin14	VCC	—	—	模块电源正极, 1.8V~3.8V, 推荐 3.3V
Pin15	GND	—	—	模块地 GND
Pin16	EXT ANT	—	—	外接天线输出

● 串口透传协议说明(桥接模式)

模块的桥接模式是指，通过通用串口和用户 CPU 相连，建立用户 CPU 和移动设备之间的双向通讯。用户可以通过串口，使用指定的 AT 指令对串口波特率，BLE 连接间隔进行重设置(详见后面《串口 AT 指令》章节)。针对不同的串口波特率以及 BLE 连接间隔，以及不同的发包间隔，模块将会有不同的数据吞吐能力。

此型号模块和信驰达 CC2640 蓝牙 5.0 主模块之间设定波特率为 921600 bps、最小连接间隔为 10ms 时，模块能达到最高 16K/S 的有效转发速率；此模块和手机通信，在波特率为 921600 连接间隔为 30ms 时，有效转发速率为 5K/S。

模块可以根据获取到的已连接设备的 MTU 自定义串口包，模块会根据数据包大小自动分包发送，每个无线包最大载荷为 MTU 减去 3 个字节（例如：模块和苹果手机连接，获取 MTU 为 185，则模块发送数据给手机每个最大的包为 182 个字节）。移动设备方发往模块的数据包，必须自行分包（每包 1 个字节到 (MTU-3) 个字节之间）发送。模块收到无线包后，会依次转发到主机串口接收端。

- 1、串口硬件协议：115200 bps，8，无校验位，1 停止位。
- 2、EN 为高电平，蓝牙模块处于完全睡眠状态。EN 置低时，模块会以 200ms 的间隔开始广播，直到和手机对接成功。当 EN 从低到高跳变，不论模块状态，会立即进入睡眠。
- 3、连接成功之后，主机(MCU)如有数据发送至 BLE 模块，需将 BRTS 拉低，主机可在约 50ms 后开始发送数据。发送完毕之后主机应主动抬高 BRTS，让模块退出串口接收模式。要注意的是，数据发送完毕后也要延时一段时间，抬高 BRTS 之前请确认串口数据完全发送完毕，否则会出现数据截尾现象。
- 4、当模块有数据上传请求时，模块会置低 BCTS，开始发送，直到数据发送完毕。数据发送完毕，模块会将 BCTS 置高。
- 5、如若主机的 BRTS 一直保持低电平，则蓝牙模块会一直处于串口接收模式，会有较高的功耗。
- 6、在模块连接成功后，会从 TX 给出 "TTM:CONNECTED" 字串，可以根据此字串来确定是否可以进行正常转发操作。当然也可以使用连接状态提示脚，也可以通过手机发送一个特定的确认字串到模块，主机收到后即可确认已经连接。当连接被 APP 端主动断开后，会从 TX 给出 "TTM:DISCONNECT" 字串提示，如果是非正常断开，会从 TX 给出 "TTM:DISCONNECT

FOR TIMEOUT” 字串提示。

7、模块的蓝牙默认连接间隔为 **20 ms**，如果需要节省功耗采用低速转发模式，需通过 AT 指令调整连接间隔（最长连接间隔 2000ms），每个连接间隔最多传输 248 个字节，连接间隔为 T(单位：ms)，那么每秒最高转发速率 V（单位 byte/s）为：

$$V = 248 * 1000 / T \quad (V \text{ 只和 } T \text{ 有关})$$

如果模块的蓝牙连接间隔为 **10ms**，而每个间隔最多传输 **248 byte** 有效数据，因此理论最高传输能力(转发速率)为 **248*1000/10= 24.8K Byte/s**。测试表明，转发速率在 **16K/s** 以下，在信号较好时，基本上无漏包情况。安全起见，无论是低速或者高速转发应用，都建议在上层做校验重传处理。

注：Android 的 MTU 为 251 字节，IOS 的为 185 字节，发送时每包大小为 MTU-3 个字节。

8、串口数据包的大小可以不定长，长度可以是 1K 字节以下的任意值，同样满足以上条件即可。但为最大效率地使用通讯的有效载荷，同时又避免通讯满负荷运行，推荐使用 **20, 80, 248** 字节长度的串口数据包，包间间隔取大于 20ms。

注：经测试，在 IOS 中，调用对 **Characteristic** 的写函数使用

CBCharacteristicWriteWithResponse 参数，使用带回应写模式，这种模式会降低部分转发效率，但可保证单个数据包的正确性，而使用 **CBCharacteristicWriteWithoutResponse** 参数，使用不带回应写模式，这种模式会有利于提高转发效率，但数据包的正确性需要 APP 上层去校验。

● 串口 AT 指令

以"TTM"开头的字符串会当成 AT 指令进行解析并执行，之后会追加输出执行结果，"TTM:OK\r\n\r\n"或 "TTM:ERP\r\n\r\n"等。不以“TTM”开头的串口数据包，将被视为透传数据。

➤ 连接间隔设定

向串口 RX 输入以下字符串，设定 BLE 连接间隔：

"TTM:CIT-Xms"

其中 X="20", "30", "50", "100", "200", "300", "400", "500", "1000", "1500", "2000", 单位 ms（以上数据格式都为 ASCII 码）。如“TTM:CIT-30ms”表示设定连接间隔为 30ms。

在执行完此指令之后，会从串口 TX 得到以下确认：

"TTM:TIMEOUT" 表示更改超时，修改失败；

"TTM:OK" 表示更改成功，正以新的连接间隔在运行。

注：只有在连接成功后更改此指令才有效。

➤ 获取模块名称

向串口 RX 输入以下字符串：

"TTM:NAM-?"

会从 TX 收到：

" TTM:NAM-xxxxxxxxxxxx\r\n\r\n"

字符串后面"xxxxxxxxxxxx"为蓝牙模块名称。

➤ 模块重命名

向串口 RX 输入以下字符串，其中“Name”为模块名称，长度为 16 个字节以内，ASCII 码格式，

" TTM:REN-" + Name

如“TTM:REN-ABC123”表示将模块重命名为“ABC123”。

若修改成功则会从 TX 收到 "TTM:OK" 确认，如果指令格式不对，则会返回：

"TTM:ERP"

测试表明，设备名称修改在 iOS 系统中无法立即变更，在 android 系统中可以实现立即变更。用户可以通过 PC 进行设置后使用，也可以通过移动设备的 BLE APP 接口进行设置。见 [《模块参数设置【服务 UUID: 0xFF90】》](#)。

➤ 获取波特率

向串口 RX 输入以下字符串，设定波特率：

"TTM:BPS-?"

会从 TX 收到：

"TTM:BPS-X"

其中 X="4800", "9600", "19200", "38400", "57600", "**115200**", "256000", "512000", "921600"（以上数据格式都为 ASCII 码）。

➤ 波特率设定

向串口 RX 输入以下字符串，设定波特率：

"TTM:BPS-X"

其中 X="4800", "9600", "19200", "38400", "57600", "**115200**", "256000", "512000", "921600",（以上数据格式都为 ASCII 码）。如“TTM:BPS-115200”表示设定波特率为 115200bps。在执行完此指令之后，会从串口 TX 得到以下确认：“TTM:BPS SET AFTER 2S ...”，如果设置值不在选项中，或者指令格式不对，则返回：“TTM:ERP\r\n\r\n0”

➤ 获取物理地址 MAC

向串口 RX 输入以下字符串：

"TTM:MAC-?"

会从 TX 收到：

" TTM:MAC-xxxxxxxxxxxx"

字符串后面"xxxxxxxxxxxx"为 6 字节模块蓝牙地址。

➤ 设置模块 MAC 地址

向串口 RX 输入以下字符串：

"TTM:MAC-xxxxxxxxxxxx"，其中"xxxxxxxxxxxx"为待设置的 6 字节模块蓝牙地址。

会从 TX 脚收到 "TTM:OK\r\n\r\n0" 确认，如果指令格式不对，则会返回：

"TTM:ERP\r\n\r\n0"

设定掉电保存，重启模块后，模块将按照新的 MAC 地址进行工作。

➤ 获取模块版本号

向串口 RX 输入以下字符串：

"TTM:VER-? "

会从 TX 脚收到 "TTM:VER-XXXXXX" ， X 为模块的版本号，如果指令格式不对，则会返回："TTM:ERP\r\n0"

➤ 获取模块连接密码

向串口 RX 输入以下字串：

"TTM:PWD-?"

会从 TX 脚收到 "TTM:PWD-XXXXXX" ， X 为 6 个字节的连接密码，如果指令格式不对，则会返回：

"TTM:ERP\r\n0"

➤ 设置模块连接密码

向串口 RX 输入以下字串，可以设置或者更改模块的连接密码，长度为 6 个字节的数字格式。

"TTM:PWD-xxxxxx"

会从 TX 脚收到 "TTM:OK-xxxxxx\r\n0" 确认，如果指令格式不对，则会返回：

"TTM:ERP\r\n0"

➤ 模块复位—上电复位

向串口 RX 输入以下字串：

"TTM:RST-SYSTEMRESET"

会迫使模块软复位一次。

➤ 模块复位—浅恢复

向串口 RX 输入以下字串：

"TTM:RST-RSTPWD" 会迫使模块浅复位一次。这次浅复位是恢复模块的密码参数。

➤ 模块复位—深度恢复

"TTM:RST-RESET" 会迫使模块深度复位一次。这次深度复位是恢复模块所有修改过的参数，使其恢复到出厂状态。

➤ EN 脚内部使能

向串口 RX 输入以下字串，设置 EN 脚开启内部上拉（默认状态）："TTM:EUP-ON\r\n0"。

向串口 RX 输入以下字串，设置关闭 EN 脚开启内部上拉，EN 脚拉低使能广播时，能节

省 120 多 μ A 电流: "TTM:EUP-OFF\r\n0".

设置成功后 TX 脚收到 "TTM:OK\r\n0" 确认, 如果指令格式不对, 则会返回: "TTM:ERP\r\n0".

注: 此参数掉电不保存, 并且掉电后默认恢复上拉状态。

➤ 广播周期设定

向串口 RX 输入以下字串, 设置模块的广播周期: "TTM:ADP-(X)".

其中 X = "20", "50", "100", "**200**", "500", "1000", "1500", "2000", "2500", "3000", "4000", "5000"之一 (以上数据格式都为 ASCII 码)。如 "TTM:ADP-(200)" 表示设定广播周期为 200ms。若修改成功则会从 TX 脚收到 "TTM:OK\r\n0" 确认, 如果指令格式不对, 则会返回: "TTM:ERP\r\n0".

重启模块后, 模块将按照新的广播周期进行广播。

➤ 修改透传服务 UUID

向串口 RX 输入以下字串, 设置模块的透传服务 UUID:

"TTM:UIDS-x1,x2,x3,x4,x5,x6", 其中

x1: UUID 位数选择。取值范围: 0~1。(0: 使用 16bit UUID, 1: 使用 128bit UUID, 重启设备生效)

x2: 设备透传发送 service UUID。取值范围: 0000~FFFF。(若使用 128bit 模式时, 该值为 128bit 模式的第 3、4 字节)

x3: 设备透传发送 characteristic UUID。取值范围: 0000~FFFF。(若使用 128bit 模式时, 该值为 128bit 模式的第 3、4 字节)

x4: 设备透传接收 service UUID。取值范围: 0000~FFFF。(若使用 128bit 模式时, 该值为 128bit 模式的第 3、4 字节)

x5: 设备透传接收 characteristic UUID。取值范围: 0000~FFFF。(若使用 128bit 模式时, 该值为 128bit 模式的第 3、4 字节)

x6 : 128bit 基础 UUID 值。取值范围: 00000000000000000000000000000000~FFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFF。(基础 UUID 的第 3、4 字节替换为以上参数的 UUID 构成设备 128bit 的实际 UUID)

说明:

1、使用 16bit UUID 功能时, 参数 x6 无作用, 可以不设置。

2、参数使用十六进制数值表示, 参数 x2 到参数 x5 使用 2 个字节, 参数 x6 使用 16 字节, 如参数 x5 为 "0002", 参数 x6 为 "00112233445566778899AABBCCDDEEFF", 指令参考使用如下:

- TTM:UIDS-0,FFE0,FFE4,FFE5,FFE9,FFEEFFE0BBAA99887766554433221100\r\n
(默认值)
 - TTM:UIDS-0,BB00,BB01,BB02,BB03\r\n
 - TTM:UIDS-1,BB00,BB01,BB02,BB03,11223344556677889900112233445566\r\n
 - TTM:UIDS-1,0000,0001,0000,0002,00112233445566778899AABBCCDDEEFF\r\n
- 若修改成功则会从 TX 脚收到 "TTM:OK\r\n\r\n0" 确认，如果指令格式不对，则会返回：
"TTM:ERP\r\n\r\n0"。
- 指令重启后生效，掉电保存。

➤ 附加自定义广播内容

向串口 RX 输入以下字串，自定义广播内容：

"TTM:ADD-"+ Data

其中 Data 为准备附加的广播的数据，长度 $0 < L \leq 16$ ，以 ASCII 码格式输入。例如向串口 RX 输入 "TTM:ADD-Advertisement!"，会从 TX 脚收到 "TTM:OK\r\n\r\n0" 确认，如果指令格式不对，则会返回："TTM:ERP\r\n\r\n0"。

此指令设置后立即生效，可以通过此功能广播一些自定义内容，数据掉电保存。如果设置为 16 个全 0 (0x00) 数据，则认为不使用自定义广播数据，而是使用默认广播内容。

➤ 定义产品识别码

向串口 RX 输入以下字串，自定义产品识别码：

"TTM:PID-"+ Data

其中 Data 为两个字节的识别码，范围 0x0000~0xFFFF (L = 2)，每个字符以 ASCII 码格式向串口 RX 输入。例如向串口 RX 输入 "TTM:PID-RS" ("RS 对应的十六进制为 0x5253")，会从 TX 脚收到 "TTM:OK\r\n\r\n0" 确认，如果指令格式不对，则会返回：TTM:ERP\r\n\r\n0"。

此识别码掉电保存，会出现在广播中，可以以此来过滤设备或判断是否是特定的产品。

➤ 发射功率设定

向串口 RX 输入以下字串，设置相应的发射功率，单位 dBm。

"TTM:TPL-(X)"

其中 X="2", "1", "0", "-3", "-6", "-9", "-12", "-15", "-18", "-21" (以上数据格式都为 ASCII 码)。如 "TTM:TPL-(2)" 表示设定发射功率为 +2dBm。之后会从 TX 脚收到 "TTM:OK\r\n\r\n0" 确认，并且模块立即使用新的发射功率进行通讯，如果指令格式不对，则会返回："TTM:ERP\r\n\r\n0"。

➤ 自定义大广播包数据

向串口 RX 输入以下字串，设置相应的物理层数据率：**"TTM:ADL-"+Data**。

其中 Data 为准备附加的广播的数据，长度 $0 < L \leq 232$ ，以 ASCII 码格式输入。例如向串口 RX 输入“TTM:ADL-Advertisement012345.....9”，会从 TX 脚收到 **"TTM:OK\r\n0"** 确认，如果指令格式不对，则会返回：**"TTM:ERP\r\n0"**。

此指令设置后立即生效，可以通过此功能设置蓝牙 5.0 的大广播包数据内容，此数据掉电保存。

➤ 设置物理层数据率

向串口 RX 输入以下字串，设置相应的物理层数据率：**"TTM:PHY-(x)"**。

其中 $x = "1","2","3","4","5","6"$ 表示连接时使用的物理层模式分别代表"1M PHY","2M PHY","1M + 2M PHY","CODED PHY (Long range)","1M + 2M + CODED PHY","Auto PHY"。设置成功会从 TX 脚收到**"TTM:OK\r\n0"**并返回所设置的对应参数值，并且接下来的通信会按照这个数据率进行传输。如果指令的格式不对则会返回：**"TTM:ERP\r\n0"**。

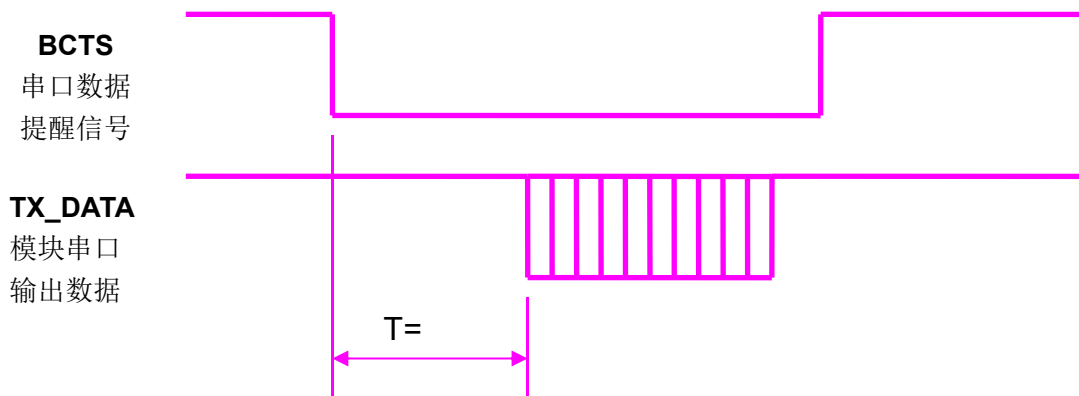
➤ 数据延时设定

向串口 RX 输入以下字串，设置 BCTS 输出低到串口 TX 输出数据之间的延时，单位 ms。

"TTM:CDL-Xms"

其中 $X = "0","2","5","10","15","20","25"$ 之一（以上数据格式都为 ASCII 码）。如“TTM:CDL-2ms”表示设定延时为 2ms，如果指令无误，会从 TX 收到 **"TTM:OK\r\n0"** 确认，如果指令格式不对，则会返回：**"TTM:ERP\r\n0"**。

为了让用户 CPU 有足够的时间从睡眠中唤醒，到准备接收，模块提供了这个延时(X)设定，在模块串口有数据发出之前会置低 BCTS，而 BCTS 输出低到模块 TX 输出数据之间的延时由此参数设定。可以保证最小延时不小于 X，实际延时会是 $T = (X + Y)$ ms，其中 $500\mu s < Y < 1ms$ 。此参数掉电保存。



模块串口输出数据延时设定示意图

AT 指令表

AT 指令格式	掉电保存	参数说明	可能的回应	含义
TTM:CIT-Xms (连接成功后才有效)	否	X="20", "30", "50", "100", "200", "300", "400", "500", "1000", "1500", "2000" 设置相应的 BLE 连接间隔, 单位 ms	TTM:TIMEOUT\r\n\r\n0 TTM:OK\r\n\r\n0 TTM:ERP\r\n\r\n0	设置超时 设置成功 错误参数
TTM:NAM-?	是	获取模块名称	TTM:NAM-xxxxxxxxxxx xx, "xxxxxxxxxxxxx" 为模块名称	返回模块名称
TTM:REN-Name	是	Name, 新模块名, 长度为 16 字节以内的任意字串。	TTM:OK\r\n\r\n0 TTM:ERP\r\n\r\n0	设置成功 错误参数
TTM:BPS-?	—	获取波特率	TTM:BPS-X, "X" 为波特率	返回波特率
TTM:BPS-X	是	X="4800", "9600", "19200", "38400", "57600", "115200", "256000", "512000", "921600"。 设置相应的波特率	TTM:BPS SET AFTER 2S ... \r\n\r\n0 TTM:ERP\r\n\r\n0	设置成功, 会在两秒后使用新的波特率 错误参数
TTM:MAC-?	—	获取 MAC 地址	TTM:MAC-xxxxxxxxxxx xx, "xxxxxxxxxxxxx" 为模块 MAC 地址	返回 MAC 地址
TTM:MAC-X	是	X 为 12 位 MAC 字符, 比如 123456789ABC	TTM:OK\r\n\r\n0 TTM:ERP\r\n\r\n0	设置成功 错误参数
TTM:VER-?	—	获取版本号	"TTM:VER-XXXXXXX 为模块版本号	返回版本号
TTM:PWD-?	—	获取密码	"TTM:PWD-XXXXXXX"	返回密码
TTM:PWD-xxxxxx	是	X 为 6 位纯数字, 比如 123456	"TTM:OK-xxxxxx\r\n\r\n0" "TTM:ERP\r\n\r\n0"	返回 MAC 地址 错误参数
TTM:RST-SYSTEMRESET	—	让模块系统复位	Module is working!	复位模块
TTM:RST-RSTPWD	—	浅恢复	Module is working!	恢复出厂密码
TTM:RST-RESET	—	深度恢复	Module is working!	恢复所有参数

TTM:EUP-ON	否	开启 EN 脚上拉使能	TTM:OK\r\n\r\n0	EN 脚内部上拉开启
TTM:EUP-OFF	否	关闭 EN 脚上拉使能	TTM:OK\r\n\r\n0	EN 脚内部上拉关闭
TTM:ADP-(X)	是	X = "20", "50", "100", " 200 ", "500", "1000", "1500", "2000", "2500", "3000", "4000", "5000"之一.	TTM:OK\r\n\r\n0 TTM:ERP\r\n\r\n0	设置广播周期, 如 设 为 "TTM:ADP-(200)"表示设定广播周期为 200ms。
TTM:UIDS-x1,x2,x3,x4,x5,x6	是	修改透传服务 UUID	TTM:OK\r\n\r\n0 TTM:ERP\r\n\r\n0	设置成功 错误参数
TTM:ADD- Data	是	Data 为自定义广播数据,数据长度 L <= 16 Byte;	TTM:OK\r\n\r\n0 TTM:ERP\r\n\r\n0	设置自定义广播内容
TTM:PID- Data	是	Data 为自定义产品识别码,数据长度 L = 2 , 默认为 5246 ;	TTM:OK\r\n\r\n0 TTM:ERP\r\n\r\n0	设置自定义产品识别码
TTM:TPL-(X)	否	X=" 2 ", "1", "0", "-3", "-6", "-9", "-12", "-15", "-18", "-21" 设置相应的发射功率, 单位 dBm	TTM:OK\r\n\r\n0 TTM:ERP\r\n\r\n0	设定发射功率
TTM:ADL-Data	是	Data 为自定义大广播包内容,数据长度 L <=232 , 默认全为 0;	TTM:OK\r\n\r\n0 TTM:ERP\r\n\r\n0	设定 5.0 特有的大广播包自定义内容
TTM:PHY-(x)	否	x = "1","2","3","4","5","6" 设连接时物理层的模式分别表示: 1: 1M PHY 2: 2M PHY 3: 1M + 2M PHY 4:CODED PHY (Long range) 5: 1M + 2M + CODED PHY 6: Auto PHY	"TTM:OK\r\n\r\n0" "TTM:ERP\r\n\r\n0"	设置物理层数据率
TTM:CDL-Xms	是	X=" 0 ","2","5","10","15","20","25", 设置 BCTS 输出低到串口输出数据之间的延时,单位 ms	TTM:OK\r\n\r\n0 TTM:ERP\r\n\r\n0	最小延时不于 X, 实际延时会是 X+Y ms , 500μs<Y<1ms.

* 注: 蓝色粗体为默认设置。

● BLE 协议说明(APP 接口)

➤ 蓝牙数据通道【服务 UUID: 0xFFE5】

特征值 UUID	可执行的操作	字节数	默认值	备注
FFE9	Write	MTU-3	无	写入的数据将会从串口 TX 输出

说明： 蓝牙输入转发到串口输出。APP 通过 BLE API 接口向此通道写操作后，数据将会从串口 TX 输出。详细操作规则见《串口透传协议说明(桥接模式)》章节。

➤ 串口数据通道【服务 UUID: 0xFFE0】

特征值 UUID	可执行的操作	字节数	默认值	备注
FFE4	Notify	MTU-3	无	从串口 RX 输入的数据将会在此通道产生通知发给移动设备

说明： 串口输入转发到蓝牙输出。如果打开了 FFE4 通道的通知使能开关，主 CPU 通过串口向模块 RX 发送的合法数据后，将会在此通道产生一个 notify 通知事件，APP 可以直接在回调函数中进行处理和使用。详细操作规则见《串口透传协议说明(桥接模式)》章节。

注： MTU 指 最大传输单元 (Maximum Transmission Unit, MTU)，通信协议所能通过的最大数据包大小 (以字节为单位)。

➤ 模块参数设置【服务 UUID: 0xFF90】

特征值 UUID	可执行的操作	字节数	默认值	备注
FF91	Read/Write	16	Tv450u-xxxxxx xx(带结束符的 ASCII 字串)	设备名称, xxxxxxxx 为物理地址的后四个字节

FF92	Read/write	1	0	蓝牙通讯连接间隔： 0: 10ms 1: 20ms 2: 30ms 3: 50ms 4: 100ms 5: 200ms 6: 300ms 7: 400ms 8: 500ms 9: 1000ms 10: 1500ms 11: 2000ms
FF93	Read/write	1	5	设定串口波特率： 0: 4800 bps 1: 9600 bps 2: 19200 bps 3: 38400 bps 4: 57600 bps 5: 115200 bps 6: 256000bps 7: 512000bps 8: 921600bps
FF94	write	1	无	远程复位恢复控制通道： ➤ 远程复位控制，写入 0x55 对模块进行复位 ➤ 远程深度恢复控制，写入 0x36 对模块进行深度恢复（让模块所有参数回到出厂设置），并复位
FF95	Read/write	1	3	设定广播周期： 0: 20 ms, 1: 50 ms, 2: 100 ms, 3: 200 ms, 4: 500 ms, 5: 1000 ms, 6: 1500 ms, 7: 2000 ms, 8: 2500 ms, 9: 3000 ms, 10: 4000 ms, 11: 5000 ms.

FF96	Read/write	2	0x5246	设定产品识别码
FF97	Read/write	1	7	设定发射功率: 0: -21 dBm 1: -18 dBm 2: -15 dBm 3: -12 dBm 4: -9 dBm 5: -6 dBm 6: -3 dBm 7: 0 dBm 8: 1 dBm 9: 2 dBm
FF98	Read/write	16	默认广播内容 (详见《广播数据设置》章节)	设定自定义广播数据 自定义广播数据, $0 < n \leq 16$
FF99	Read/write	232	默认广播内容 (详见《广播数据设置》章节)	设定自定义大广播包数据 自定义大广播数据, $0 < n \leq 232$

说明: 模块信息配置通道。

FF91 为设备名称设置通道。可以通过对此通道进行读写操作, 来获取和设定模块名称。设置的名称长度 L, 必须满足 $0 < L < 17$, **建议以结束符结尾 ('0')**。默认为 "Tvvvvu-xxxxxxx\0" (16 byte), vvvv 为固件版本号, xxxxxxxx 为 MAC 地址后四个字节。

FF92 为模块连接间隔设置通道。可以通过对此通道进行写操作, 来设定移动设备和模块之间的连接间隔, 借此可以灵活控制设备功耗及数据吞吐量。在对此通道进行写操作之后, 会从串口 TX 得到以下确认:

"TTM:TIMEOUT" 表示更改超时, 修改失败;

"TTM:OK" 表示更改成功, 正以新的连接间隔在运行。

测试表明, 使用 iPhone 4s (IOS 5.1.1) 从连接间隔为 500ms 修改为其他连接间隔, 需要大约 30s 的等待时间。相反从高频度的连接间隔 (如 50ms) 进行变更, 会有很高的执行效率 (BLE 协议导致)。

FF93 为模块串口波特率设置通道。可以通过对此通道进行读写操作, 来设定模块通用串口波特率, 两秒后开始启用新的波特率。出厂设置默认为 5 (115200 bps)。

FF94 为远程复位恢复控制通道, 通过写入不同值, 可以实现不同的控制功能。

- 1、对此通道写入 **0x55**, 对模块进行软件复位。
- 2、过此通道写入 **0x36**, 对模块进行深度恢复, 所有系统参数将恢复到出厂设置控制, 之后会

复位模块。

FF95 为模块广播周期设置通道。可以通过对此通道进行读写操作，来设定模块广播周期。出厂设置默认为 3 (200ms)。

FF96 为模块产品识别码设置通道。可以通过对此通道进行读写操作，来设定模块识别码，APP 端可以通过此识别码来进行过滤和连接指定的产品类型。出厂设置默认为 0x5246。

FF97 为模块发射功率设置通道。可以通过对此通道进行写操作，来设定模块发射功率。出厂设置默认为 7 (0 dBm)。

FF98 为模块一般广播内容设置通道。可以通过对此通道进行写操作，来自定义模块的广播数据。当数据为全 0 (16 Byte) 时，认为不使用自定义广播数据，而使用默认的广播数据，[详见《广播数据设置》章节](#)。

FF99 为模块大广播内容设置通道。可以通过对此通道进行写操作，来自定义模块的大广播数据。当数据为全 0 (232 Byte) 时，认为不使用自定义广播数据，而使用默认的广播数据，[详见《广播数据设置》章节](#)。

➤ 设备信息【服务 UUID: 0x180A】

特征值 UUID	可执行的 操作	字节数	默认值	备注
2A23	Read	8	xxxxxx0000xxxxxx (Hex)	系统 ID, xxxxxxxxxxxx 为模块芯片物理地址, 低字节在前
2A26	Read	6	V4.50u (ASCII)	模块软件版本号

说明：模块信息读取通道。

2A23 为模块信息获取通道，可以通过对此通道进行读操作，来获取此模块 ID。格式如 xxxxxx0000xxxxxx，其中 xx 部分为模块芯片的物理地址 MAC，六个字节，低字节在前。

2A26 为模块软件版本号读取通道，可以通过对此通道进行读操作，来获取模块软件版本，格式为 Vx.xx。x.xx 为固件版本号。

➤ 防劫持密钥【服务 UUID: 0xFFC0】

模块支持防劫持加密，此服务可以有效防止被非授权移动设备(手机)连接到此模块。模块的初始密码为 000000 (ASCII)，此情况下 APP 无需提交密码，视为不使用密码，任何安装指定 APP 的移动设备可以对其发起连接。

新密码（非全 0）的设置和备份保存由 APP 完成，如果设置了新密码（非全 0），开始

启用防劫持密码。在 APP 对此模块进行连接后，必须在蓝牙连接后的 **20 秒内**向模块提交一次曾经设置的连接密码，否则模块会断开连接。在 APP 提交正确密码到模块之前，无法对服务通道进行任何除提交密码之外的写操作。

为了安全起见，模块不提供密码读操作，密码的记忆由 APP 来负责。

协议提供了密码通道来实现密码的提交，修改，和取消密码服务。同样也提供了密码事件通知服务来通知 APP 对密码操作的结果，其中包括密码正确，密码错误，密码修改成功，取消使用密码四个事件。

特征值 UUID	可执行的 操作	字节 数	举例	备注
FFC1 (handle: 0x0045)	write (掉电 保存)	1 2	"123456123456"(ASCII)	提交当前密码 123456 ，新密码和旧密码必须一致
			"123456888888"(ASCII)	把旧密码 123456 修改为新密码 888888 ，旧密码必须正确
			"888888000000"(ASCII)	取消密码，新密码修改为 000000 ，旧密码必须正确
FFC2 (handle: 0x0048)	notify	1	0 (PWD_RIGHT_EVENT)	提交密码正确
			1 (PWD_ERROR_EVENT)	提交密码错误
			2 (PWD_UPDATED_EVENT)	密码修改成功
			3 (PWD_CANCEL_EVENT)	取消密码

说明：

- 1、密码结构为 12 字节 ASCII 码，红色部分为当前密码，蓝色部分为新密码；
- 2、当前密码在被 APP 修改之前，默认为“000000”；
- 3、通过打开通道 **FFC2** 的通知使能（如果使用 **BTTool** 操作，需向 $0x0048+1=0x0049$ 写入 **0100**），将会在此通道产生有关密码操作的执行结果通知。
- 4、当 APP 提交密码“123456123456”，新密码和当前密码相同，APP 会在 FFC2 通道得到通知 `notify:0(PWD_RIGHT_EVENT)`，表示提交密码正确；
- 5、当 APP 提交密码（红色部分）和当前密码不一致，如：“123455xxxxxx”，x 部分不论是何值，APP 会在 FFC2 通道得到通知 `notify:1(PWD_ERROR_EVENT)`，表示密码提交错误；
- 6、当 APP 提交密码“123456888888”，新密码为“888888”，当前密码为“123456”，APP

会在 FFC2 通道得到通知 notify:2(PWD_UPDATED_EVENT), 表示密码修改成功;

7、当 APP 提交密码“888888000000” , 新密码被修改为全 0, 则表示取消使用密码, APP 会在 FFC2 通道得到通知 notify:3(PWD_CANCEL_EVENT)。



● 广播数据设置

默认广播数据：当模块的 EN 脚被置低后，模块将会进行间隔为 200ms 的广播，在广播数据中的 GAP_ADTYPE_MANUFACTURER_SPECIFIC (IOS 编程中官方定义宏) 域中包含了以下内容，默认广播内容为 3 个字节：

```
{  
    0x00,0x00,          自定义设备类型编码，默认为 00 00，可由 AT 指令进行设定；  
    0x00,              模块供电电量百分比，2.0V = 0%。  
}
```

自定义广播数据：如果使用 AT 指令自定义了广播内容，最大长度为 16 字节(蓝色部分)，在广播数据中的 GAP_ADTYPE_MANUFACTURER_SPECIFIC 域中将包含了以下内容，长度为 2+n 个字节：

```
{  
    0x00,0x00,          自定义设备类型编码，默认为 00 00 ，可由 AT 指令进行设定；  
    Data [n],          自定义广播数据，n <= 16 ;  
}
```

自定义大广播数据：如果使用 AT 指令自定义了大广播内容，最大长度为 232 字节(蓝色部分)。此大广播包为 5.0 蓝牙特点，最大自定义字节可达 232，需要支持 5.0 的设备或者 APP 才能看到其广播的内容：

注：自定义广播数据可通过 AT 指令修改，并且掉电保存。重新上电后，将会使用最后自定义的广播数据。如果自定义广播数据为全 0，则认为不使用自定义广播，而使用系统默认的广播内容。为避免广播数据过长带来多余的功耗，也可以通过设置自定义广播数据为 1 字节的任意值。

● 系统复位与恢复

让模块复位有三种方法，其中第三种方法可以恢复系统参数

- 1、使用 AT 指令复位模块(详见《[串口 AT 指令](#)》章节)；
- 2、使用服务通道接口，用 APP 对模块进行远程复位。(详见《[BLE 协议说明 \(APP 接口\) — 模块参数设置](#)》章节)；
- 3、使用硬件 RESTORE 脚位（见脚位定义表），模块上电后任意时刻，将此脚位拉低 **5 秒**后，模块的系统参数会恢复用户级参数（浅恢复，释放此脚位后立即复位），如果持续拉低 **20 秒**后会将模块的所有系统参数恢复到出厂设置（深度恢复），并立即复位。此脚位带内部上拉，默认不会进入此模式。

浅恢复中被恢复的系统参数包括：

- a) 防劫持密码，恢复到“000000”，默认不使用密码；

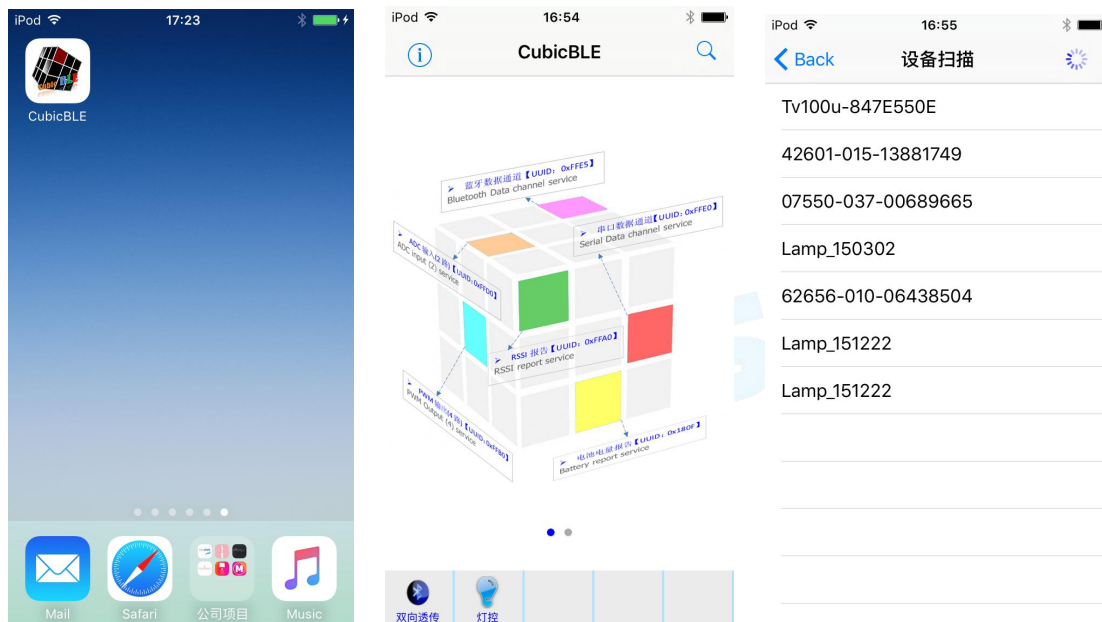
深度恢复中被恢复的系统参数包括：

- a) 防劫持密码，恢复到“000000”，默认不使用密码；
- b) 串口波特率，恢复到 115200bps；
- c) 设备名称，恢复到"Tv450u-XXXXXXXXX"，X 是 MAC 的后四个字节；
- d) 广播周期，恢复到 3(200ms)；
- e) 连接间隔，恢复到 20ms；
- f) 发射功率，恢复到 0dBm；
- g) 产品识别码，恢复到 0x5246；
- h) 自定义广播长度，恢复到 0；
- i) 自定义广播数据，恢复到全 0，不使用自定义广播数据，使用默认广播数据。

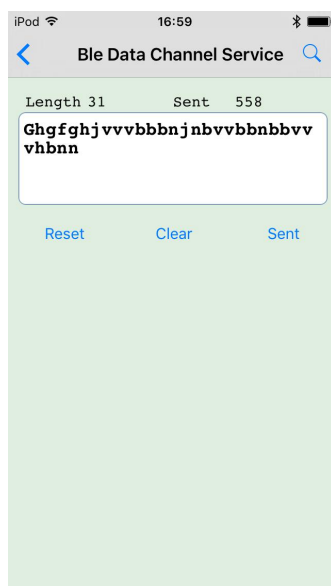
● 用 APP 测试透传功能

模块 IOS 平台的测试工具 (APP) 可以在 AppStore 下载到。打开 iPhone 5S, iPhone 6, iPhone 6s 或者 iPad4 等中的 AppStore, 搜索 CubicBLE (如需源码可向业务索取), 下载安装, 进行测试。

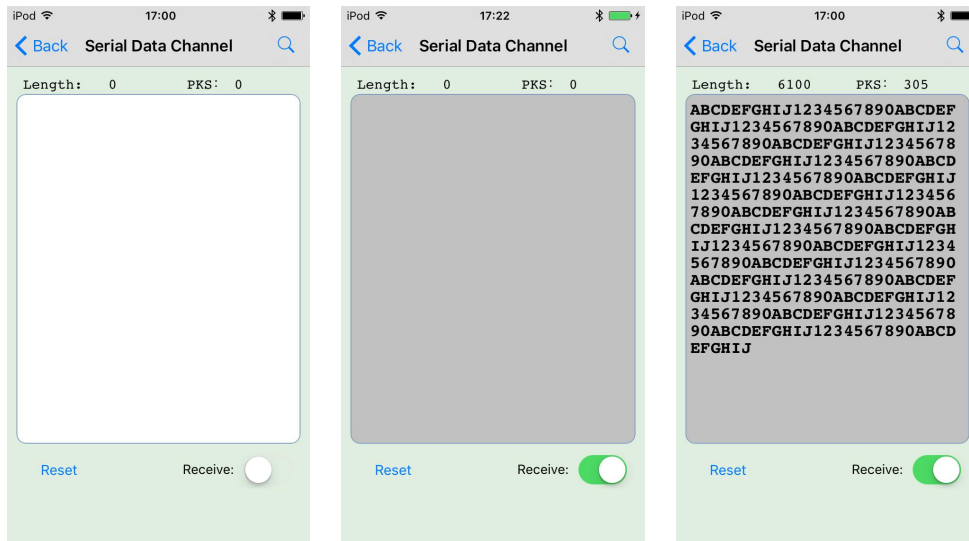
下载 APP 后, 打开蓝牙, 然后打开 APP, 选择页面右上角扫描蓝牙按钮。选择需要连接的设备, 进行连接。接着选择 Back 退回到主界面。



此时如果模块串口已经就绪 (连接了主 CPU, 或者串口终端), 即可以开始工作可进行收发测试。选择蓝牙数据通道, 输入需要发送的数据点击 “Sent” 即可发送数据到模块。



选择串口数据通道，点击“Receive”，即可接收到模块发送到 APP 的数据。



注：如果使用串口终端进行测试，串口终端的数据要发到手机，**必须保持 BRTS 置低**，防止模块进入睡眠。

关于 IOS 编程，根据低功耗蓝牙协议，移动设备发送数据可以通过 **B 通道**(发送)的对应服务 (UUID) 进行写操作。模块数据到移动设备的数据传送，是通过通知的形式进行，因此在 APP 启动后需要打开 **A 通道**(接收)对应服务 (UUID) 的通知 (Notification) 使能，之后模块串口收到的数据包会自动发送到移动设备。具体操作可以参考信驰达科技提供的基于 IOS 的透传模块测试 APP 源码。此软件不定期更新，敬请关注信驰达官方网站 (<http://www.szrfstar.com/>) 动态。

● 主机参考代码(透传)

逻辑关系：模块间是用 BCTS, BRTS 两个 IO 口进行发送接收的通知和控制。这两个 IO 常态高位，置低触发，如果模块有数据要发，置低 BCTS 通知单片机接收，如果单片机有数据要发，置低 BRTS 通知模块接收。示意性代码如下：

```

void main(void)
{
    EN = 0; //使能 EN，开始广播

    while(!BLEMoudleAck("TTM:OK")); //等待手机端扫描，连接
                                     //等待连接成功，也可加入限时等待
                                     //也可判断连接提示信号线的电平

    BRTS = 0; //BRTS 置低通知 CC2640R2F 模块准备接收
    halMcuWaitMs(50); //延迟 50ms
    UARTWrite( HAL_UART_PORT_0, "TTM:CIT-100ms", 14);
                                     //修改连接间隔，从串口得到确认：
    halMcuWaitMs(50); //延迟 50ms,确保数据已经发出
    BRTS = 1; //RTS 置高，发送完毕
    while(!BLEMoudleAck("TTM:OK")); //等待设置成功，也可加入限时等待

    while(1){ //循环收发测试
        while(1){
            if(BCTS == 0){ //检测，若 BCTS 置低则准备接收
                while(BCTS==0); //等待发送完毕，也可限时等待
                if(UARTRead(uartBuffer) == SUCCESS) //串口读取数据
                    {... ..} //使用数据
            }
            BRTS = 0; //RTS 置低通知 CC2640R2F 模块准备接收
            halMcuWaitMs(50); //延迟 50ms
            send_TX("1234567890"); //发送任意数据(200byte 以内)
            halMcuWaitMs(50); //延迟 50ms,确保数据已经发出
            BRTS = 1; //RTS 置高，发送完毕
            halMcuWaitMs(20); //延迟再发下一个包，延时视包大小而定
        }
    }
}
}

```

● IOS APP 编程参考

模块总是以从模式进行广播，等待智能移动设备做为主设备进行扫描，以及连接。这个扫描以及连接通常是由 APP 来完成，由于 BLE 协议的特殊性，在系统设置中的扫描蓝牙连接没有现实意义。智能设备必须负责对 BLE 从设备的连接，通讯，断开等管理事宜，而这一切通常是在 APP 中实现。

有关 BLE 在 IOS 下的编程，最关键的就是对特征值 (Characteristic, 本文叫通道) 的读、写，以及开启通知开关。通过对通道的读写即可实现对模块直驱功能的直接控制，无需额外的 CPU。典型函数说明摘抄如下：

```
/*!
 * @method writeValue:forCharacteristic:withResponse:
 * @param data The value to write.
 * @param characteristic The characteristic on which to perform the write operation.
 * @param type The type of write to be executed.
 * @discussion Write the value of a characteristic.
 * The passed data is copied and can be disposed of after the call finishes.
 * The relevant delegate callback will then be invoked with the status of the request.
 * @see peripheral:didWriteValueForCharacteristic:error:
 */
- (void)writeValue:(NSData *)data forCharacteristic:(CBCharacteristic *)characteristic type:(
CBCharacteristicWriteType)type;
```

说明：对某个特征值进行写操作。

```
NSData *d = [[NSData alloc] initWithBytes:&data length:mdata.length];
    [p writeValue:d
    forCharacteristic:c
    type:CBCharacteristicWriteWithoutResponse];
```

```
/*!
 * @method readValueForCharacteristic:
 * @param characteristic The characteristic for which the value needs to be read.
 * @discussion Fetch the value of a characteristic.
 * The relevant delegate callback will then be invoked with the status of the request.
```

```
* @see peripheral:didUpdateValueForCharacteristic:error:
```

```
*/
```

```
- (void)readValueForCharacteristic:(CBCharacteristic *)characteristic;
```

说明：读取某个特征值。

```
[p readValueForCharacteristic:c];
```

```
/*!
```

```
* @method setNotifyValue:forCharacteristic:
```

```
* @param notifyValue The value to set the client configuration descriptor to.
```

```
* @param characteristic The characteristic containing the client configuration.
```

```
* @discussion Ask to start/stop receiving notifications for a characteristic.
```

```
* The relevant delegate callback will then be invoked with the status of the request.
```

```
* @see peripheral:didUpdateNotificationStateForCharacteristic:error:
```

```
*/
```

```
- (void)setNotifyValue:(BOOL)notifyValue forCharacteristic:(CBCharacteristic *)characteristic;
```

说明：打开特征值通知使能开关。

```
[self setNotifyValue:YES forCharacteristic:c];//打开通知使能开关
```

```
[self setNotifyValue:NO forCharacteristic:c];//关闭通知使能开关
```

```
/*
```

```
* @method didUpdateValueForCharacteristic
```

```
* @param peripheral Peripheral that got updated
```

```
* @param characteristic Characteristic that got updated
```

```
* @error error Error message if something went wrong
```

```
* @discussion didUpdateValueForCharacteristic is called when CoreBluetooth has updated a
```

```
* characteristic for a peripheral. All reads and notifications come here to be processed.
```

```
*
```

```
*/
```

```
- (void)peripheral:(CBPeripheral *)peripheral didUpdateValueForCharacteristic:(CBCharacteristic *)characteristic error:(NSError *)error
```

说明：每次执行完读取操作后，会执行到这个回调函数。应用层在此函数内保存读取到的数据。

有关设备的扫描，连接以及其他通讯细节，可以参考信驰达科技提供的基于 IOS 的透传模块测试 APP 源码。里面实现了对 FFE9 和 FFE4 转发蓝牙数据到串口，转发串口数据到蓝牙两个通道(特征值)的操作(通知和写操作)。(相关源码请向业务索取)

● 联系我们

深圳市信驰达科技有限公司

SHENZHEN RF STAR TECHNOLOGY CO.,LTD.

Tel: 0755-8632 9829(Sales) 0755-3695 3756(FAE) Web: www.szrfstar.com

Fax: 0755-8632 9413 E-mail: sales@szrfstar.com

地址：深圳市南山区高新园科技南一道创维大厦 C 座 601 室

Add: Room 601,Block C,Skyworth Building,Nanshan High-Tech Park,Shenzhen.



附录 A: BLE 模块应用方案提示

计数采集(计步器, 弹跳球, 心率计), 86 盒插座改造, 遥控开关, 调光照明, 环境渲染背景光, 医疗检测〔血压, 血氧, 体温〕, 互动遥控玩具〔开关量, 模拟量, 输入, 输出〕, 机器人, 直升飞机, 玩具车, 防丢寻物, 电量采集, 充电管理, 外置 gps, 温湿度计, 蓝牙手表, 飞镖机, 保龄球等娱乐设备新接口, (智能设备)遥控接口, 报警器, 门禁考勤(蓝牙锁), 巡逻寻根器, 反控(智能设备)应用(紧急拨号, 遥控拍摄), 蓝牙打印, 空调控制器, 机顶盒控制器, 物流统计管理, 胎压检测, 汽车自动锁, 遥控按摩器, 车位记录, 户外点阵广告, 运动计量〔跑步, 自行车, 高尔夫〕, 定时开关, 宠物监管, 婴儿儿童护理(实时体温检测, 防丢失), 运动健身玩具(手机配合), 距离感应触发应用, 调速应用, 智能家居(遥控类), 仪器仪表无线接口, 设备无线配置接口, 景点定位, 区域软禁控制, 定量计时, 可穿戴设备, 蓝牙读卡器, 便携仪表, 设备固件远程升级接口

附录 B: BLE 模块硬件规格说明

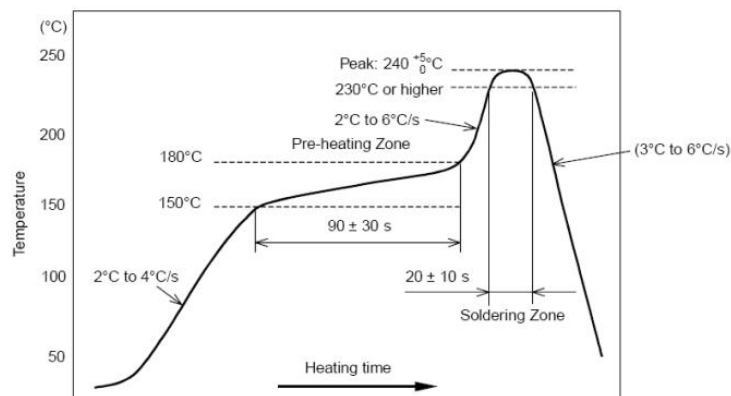
一、模块参数

- 工作电压: 2.0V~3.6V, 推荐工作电压: 3.3V
- 工作频段: 2402MHz~2480MHz
- 最大发射功率: 为+2dBm (-21dBm~+2dBm, 可编程)
- 接收灵敏度: -97dbm
- 频率误差: $\pm 40\text{kHz}$
- 工作温度: $-40^{\circ}\text{C} \sim +85^{\circ}\text{C}$
- 储存温度: $-40^{\circ}\text{C} \sim +125^{\circ}\text{C}$
- 接收电流 RX: 5.9mA
- 发射电流 TX(0dBm): 6.1mA
TX (+5dBm): 9.1 mA
- 功耗模式:
 - Active-Mode MCU: 61 $\mu\text{A}/\text{MHz}$
 - Active-Mode MCU: 48.5 CoreMark/mA
 - Active-Mode Sensor Controller: 0.4mA + 8.2 $\mu\text{A}/\text{MHz}$
 - Standby: 1.1 μA (RTC Running and RAM/CPU Retention)
 - Shutdown: 100 nA (Wake Up on External Events)

二、注意事项

使用本模块注意事项:

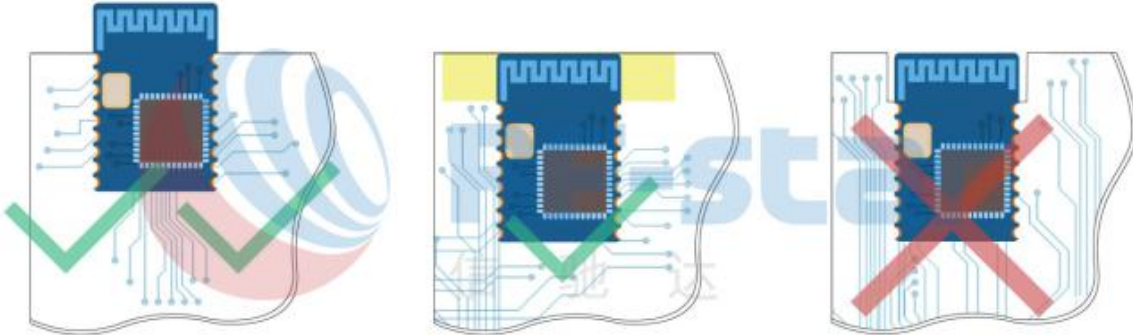
- 1、在运输、使用过程中要注意防静电。
- 2、器件接地要良好，减少寄生电感。
- 3、尽量手工焊接，如需机贴，**请控制回流焊温度不要超过 245 摄氏度**，如下图所示。
- 4、模块天线下面不要铺铜，最好挖空，以防止阻抗改变。
- 5、天线应远离其他电路，防止辐射效率变低和影响其他电路正常使用。
- 6、模块的接入电源建议使用 2.2uF+0.1uF 滤波电容对地。



部件的焊接耐热性温度曲线(焊接点)

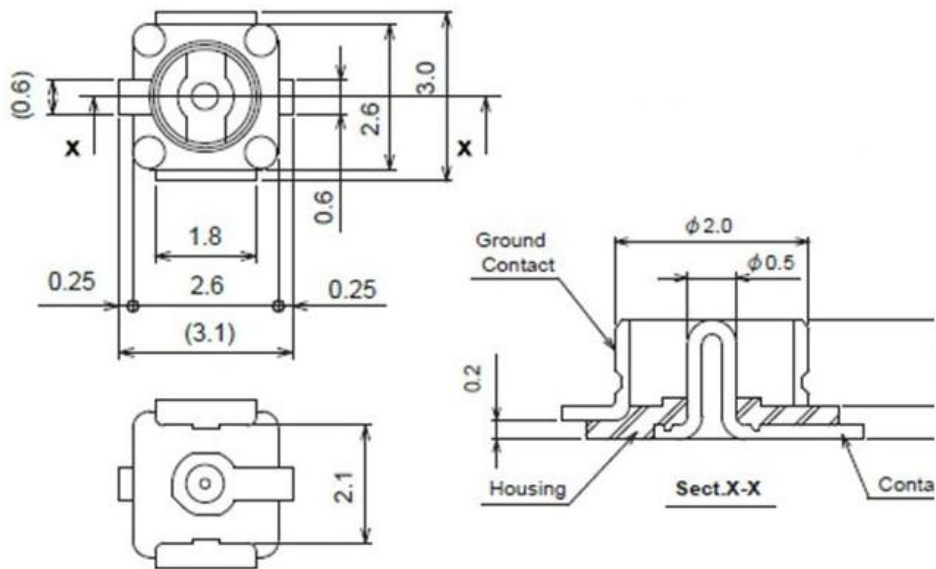
三、天线布局建议

模块天线布局请参考下图。天线的放置直接影响天线的辐射效率，建议客户在实际的产品上在进行一次天线调试，黄色区域为 keep out ，距离建议大于 10 mm：

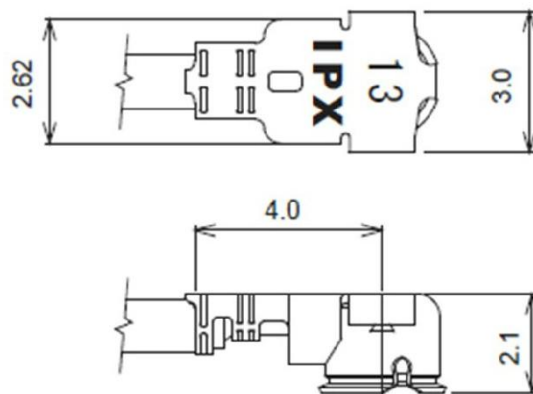


四、天线规格

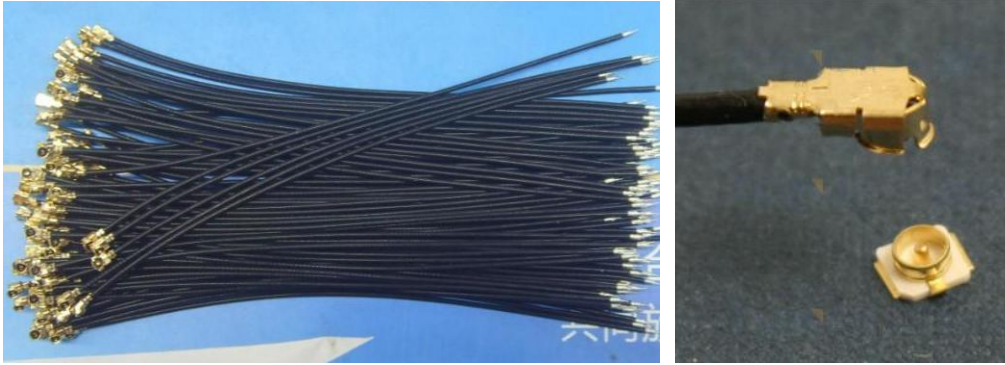
1、IPEX 天线座的规格如下图所示：



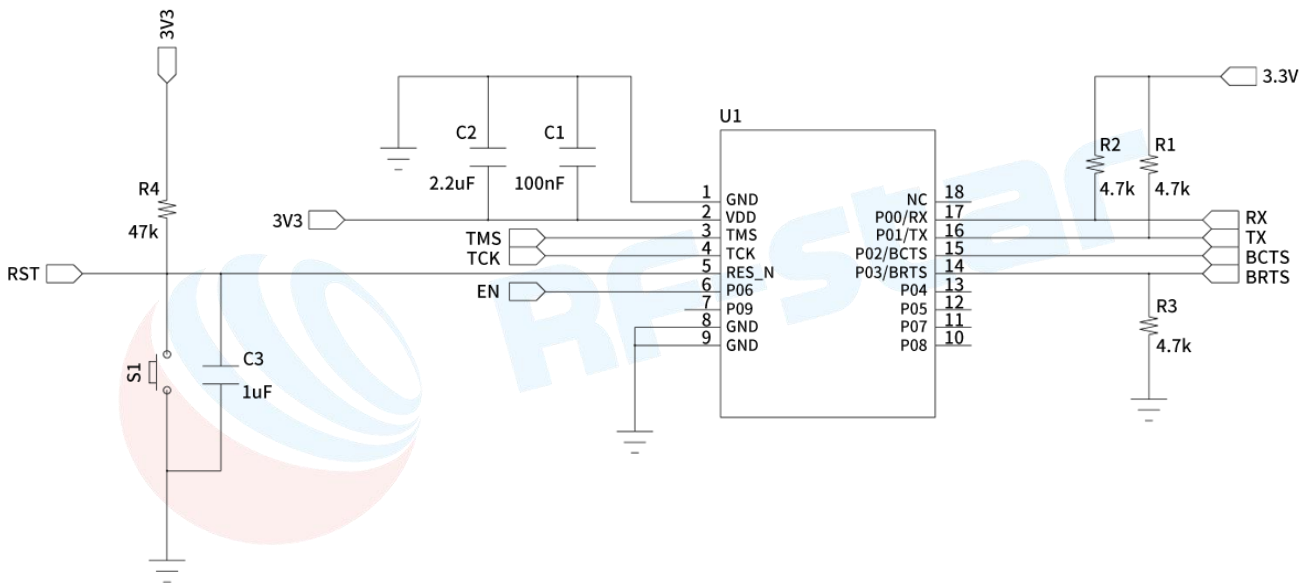
2、IPEX 线端的规格如下图所示：



3、常用天线：



五、外围参考设计



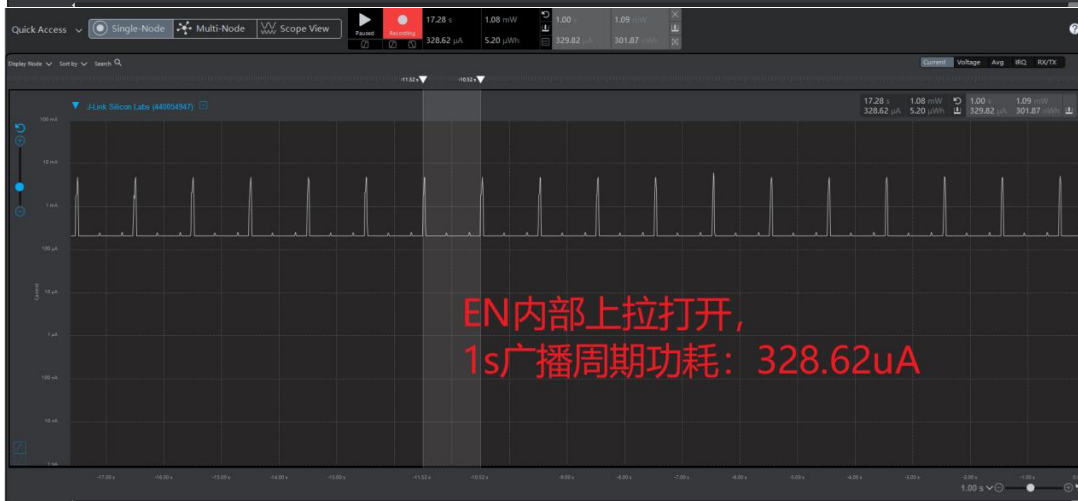
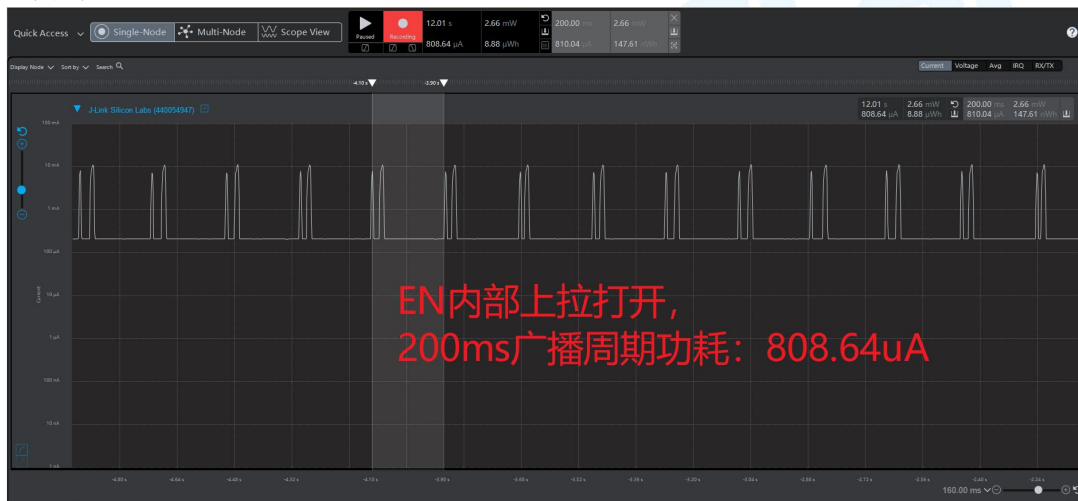
附录 C：功耗测试截图

1、睡眠功耗。



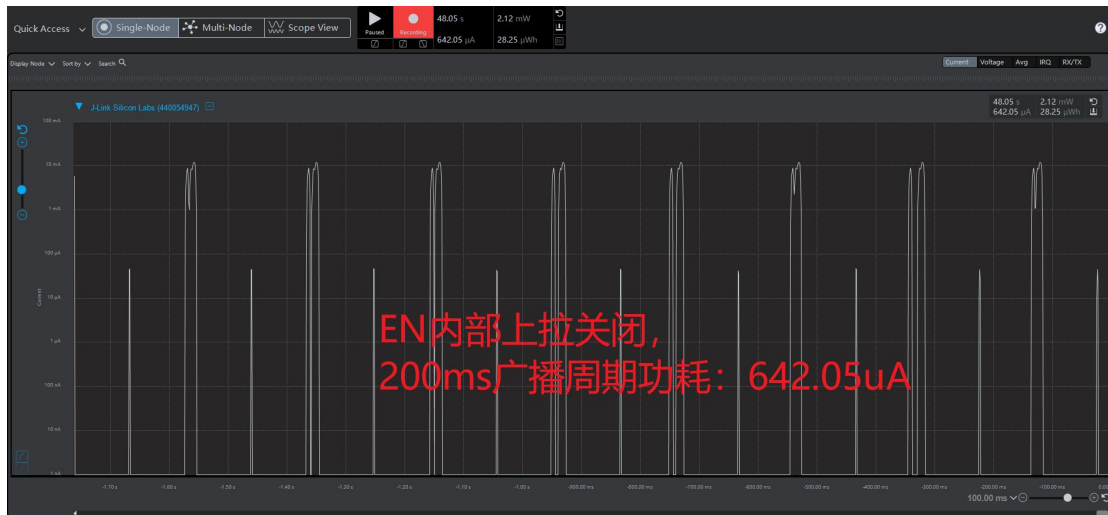
2、广播功耗。

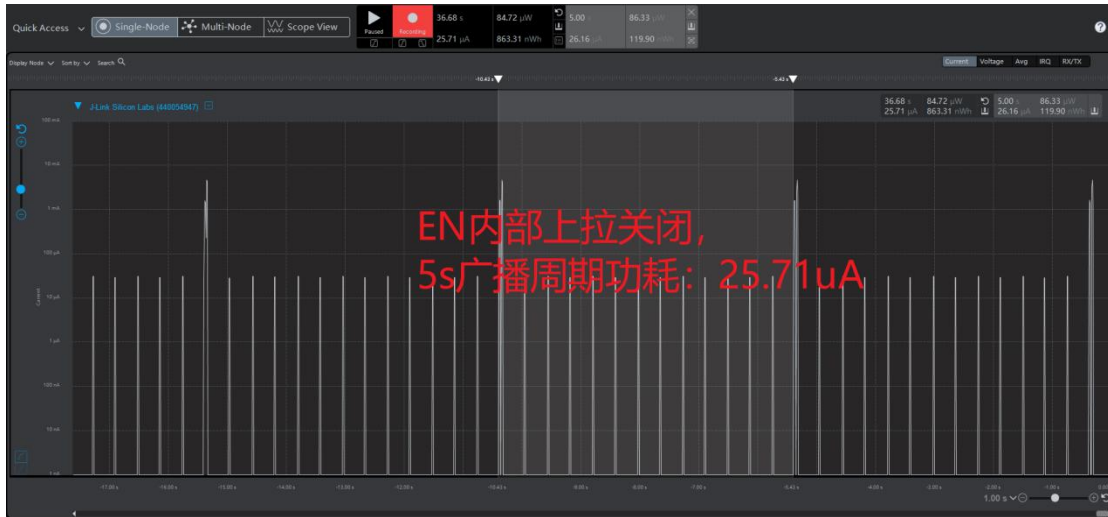
(1) EN 内部上拉打开:





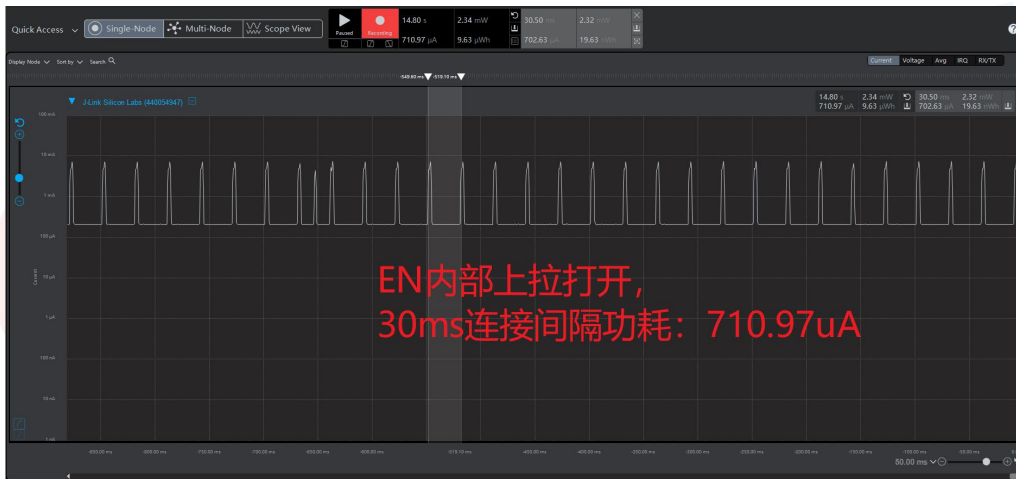
(2) EN 内部上拉关闭:





3、连接功耗。

(1) EN 内部上拉打开:



(2) EN 内部上拉关闭:

