

Ting-01(M)

Ting 系列模块基于 LoRa(SX1278)扩频芯片，Ting-01 是单 SX1278 模组，Ting-01M 是 SX1278+MCU，两款型号做到了 PIN-to-PIN 兼容。使用 Ting 系列无线模组，可以让不带无线的设备具备远程、低功耗的无线通信能力。



特性

- 超低功耗处理器：STM8L051⁽¹⁾
- 通讯接口：SPI、UART⁽²⁾
- 接口电平：3.3V TTL
- 频率范围：410MHz-470MHz
- 中心频率：433MHz
- 最大功率：19.26dBm⁽³⁾
- 灵敏度：-148dBm
- 参考传输距离 10KM⁽⁴⁾
- 支持 FSK、GFSK、LoRa、OOK 等调制方式
- 小体积双列邮票孔贴片封装，带屏蔽罩
- 尺寸：14mm x 17 mm x 2.5 mm
- 生产工艺：无铅，防静电袋包装
- 工作温度范围：-40 ~ +85 摄氏度
- 工作湿度：10% ~ 90%相对湿度，无冷凝
- 贮存温度：-40 ~ +125 摄氏度

电源

- 供电电压：2.8V - 3.6V（建议 3.3V）
- 最大持续发射电流：93mA
- 持续接口模式电流：14mA
- 睡眠电流典型值：0.75mA（未优化）

适用场合

- 自动抄表
- 家庭和楼宇自动化
- 无线报警和安全系统
- 工业监控
- 远距离传感器通信

01M 额外特性

- 通信 UART：115200，8N1
- AT 命令控制接口
- 透传模式
- 地址过滤功能
- 低功耗睡眠模式

01M 内置工作模式

- 广播发射模式（多点对多点）
- 单点对单点
- 单点对多点（可配置 65535 个地址，便于组网）
- 其他模式待开发

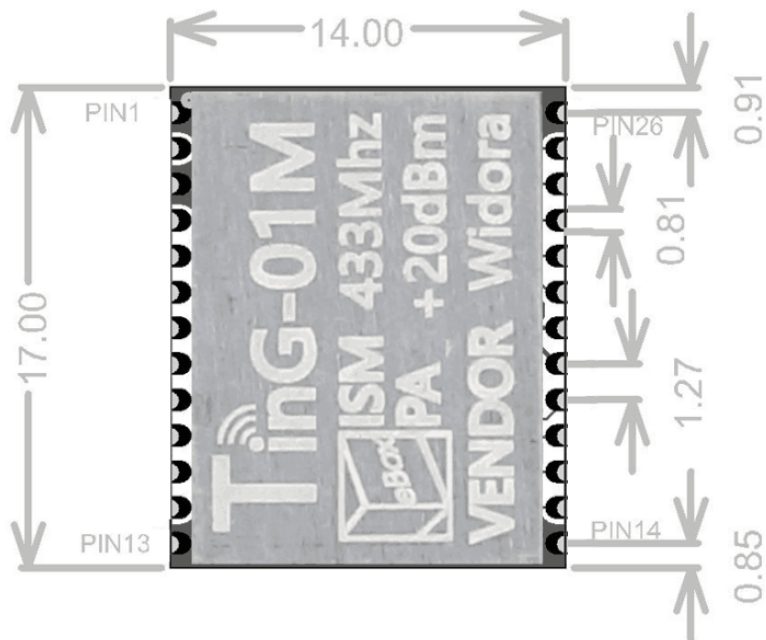
备注

- 1：仅仅 Ting-01M 具备特性
- 2：仅 Ting-01M 提供 UART 接口
- 3：升特官方提供参数为经过 RF 匹配后输出在 18.5 ~ 19.5 之间，Ting-01 实测为 19.26dBm

文档状态

- V0.6 版

尺寸数据（单位：毫米）



对外引脚

Ting-01/01M			
1	GND	GND	26
2	ANT	DIO5	25
3	GND	DIO4	24
4	VCC	DIO3	23
5	TR	DIO2	22
6	TX	DIO1	21
7	RX	NRST	20
8	SWIM	DIO0	19
9	CPURST	MISO	18
10	PD0	MOSI	17
11	PB0	SCK	16
12	PC4	NSS	15
13	GND	GND	14

引脚描述

引脚号	标识	功能
1	GND	电源地
2	ANT	RF 输出、输入
3	GND	电源地
4	VCC	电源正极
5	TR	1278 收发切换（悬空）
6	TX	模块 UART 数据输出端
7	RX	模块 UART 数据输入端
8	SWIM	量产刷固件（悬空）
9	CPURST	MCU 复位端（低有效）
10	PD0	GPIO D0
11	PB0	GPIO B0
12	PC4	GPIO C4
13	GND	电源地
14	GND	电源地
15	NSS	1278 SPI 片选
16	SCK	1278 SPI 时钟
17	MOSI	1278 SPI 数据输入
18	MISO	1278 SPI 数据输出
19	DIO0	1278 IO0
20	NRST	1278 复位控制（低有效）
21	DIO1	1278 IO1
22	DIO2	1278 IO2
23	DIO3	1278 IO3
24	DIO4	1278 IO4
25	DIO5	1278 IO5
26	GND	电路地

AT 命令

命令	发送数据格式	回复数据格式	备注
测试命令	AT\r\n	AT,OK\r\n	
复位命令	AT+RST\r\n	AT,OK\r\n	
读取版本命令	AT+VER\r\n	AT,V0.3,OK\r\n	版本 V0.3, x.x 格式
进入空闲模式	AT+IDLE\r\n	AT,OK\r\n	STM8 工作, SX1278 睡眠, 模块默认上电为此模式
进入睡眠模式	AT+SLEEP=1\r\n	AT,OK\r\n	睡眠模式, STM8 与 SX1278 都睡眠, 只允许 PC4 的下降沿唤醒
退出睡眠模式		AT,WakeUp\r\n	PC4 的下降沿唤醒睡眠中的 MCU
进入接收模式	AT+RX\r\n	AT,OK\r\n	进入接收模式, 接收模式为异步接收, 如果是单次接收, 接收完成后自动恢复到空闲模式; 如果是连续接收模式, 那将一直处于接收状态。
接收数据 (异步)		LR,XXXX,XX,ASFASDFASFD	XXXX 是源地址, 十六进制, 例如 FFCA XX 是两个字符, 十六进制数据长度, 范围为 (0x01~0xFB), 例如 5A, 代表 90 个字节 ASFASDFASFD 是任意数据
单次接收模式下的超时提醒 (异步)		AT,TimeOut\r\n	
查询 RSSI 值	AT+RSSI?\r\n	AT,-XXX,OK\r\n	十进制表示, 例如-63dB 返回为: AT,-063,OK\r\n
设置模块自身地址	AT+ADDR=XXXX\r\n	AT,OK\r\n	十六进制表示, 范围是 0000-FFFF, FFFF 是特殊地址, 如果一个模块设置自身地址为 FFFF, 那么它可以监听到同频率下所有通讯数据。
读取模块自身地址	AT+ADDR? \r\n	AT,XXXX,OK\r\n	十六进制表示, 范围是 0000-FFFF, 例如 D5AA:表示地址值为 0xD5AA
设置目标地址	AT+DEST=FF5A\r\n	AT,OK\r\n	十六进制表示, 范围是 0000-FFFF, FFFF 是特殊地址, 如果一个模块设置目标地址为 FFFF, 此时模块处于广播状态
读取目标地址	AT+DEST? \r\n	AT,XXXX,OK\r\n	十六进制表示, 范围是 0000-FFFF, 例如 FFAA:表示地址值为 0xFFAA
地址使能	AT+ADDREN=1 \r\n	AT ,OK\r\n	是否开启地址验证 1: 开启

			<p>0: 关闭</p> <p>本模块使用了软地址的协议，如果用户关闭软地址过滤则地址规则将失效，所有的模块可自由通讯。开启后服从地址的过滤规则。默认设置：0</p>
读取地址使能	AT+ADDREN? \r\n	AT ,X,OK\r\n	<p>X 代表是否开启地址验证，</p> <p>1: 开启</p> <p>0: 关闭</p>
配置参数	AT+CFG=433000000,20,6,10,1,1,0,0,0,0,3000,8,4\r\n	OK\r\n	<p>字段顺序依次为：</p> <p>载波频率(433000000)，功率(20)，带宽(6)，扩频因子(10)，纠错码(1)，CRC 校验(1)，隐式报头(0)，单次接收(0)，调频(0)，调频周期(0)，接收超时时间(3000)，用户数据长度(8)，前导码长度(4)。</p> <p>详细介绍见《参数配置命令表》。</p>
保存命令	AT+SAVE\r\n	AT,OK\r\n	<p>将配置参数、自身的地址，目标地址三个变量保存至 EEPROM。下次开机默认使用。因 EEPROM 固有特性，不要频繁调用。</p>
发送数据命令	AT+SEND=XX\r\n	AT,OK\r\n AT,SENDING\r\n AT,SENDED\r\n	<p>参数：XX 代表发送数据长度，范围为 1-250，比如要传输 25 字节数据，发送 AT+SEND=25WrWn，模块返回“AT,OKWrWn”，此时用户可以通过串口传输 25 个字节的任意数据。多于的数据将会被抛弃，模块接收完 25 个字节后，会回复“AT,SENDINGWrWn”，表示模块进入发送状态，此时，用户需等待模块回复“AT,SENDEDWrWn”表示此次数据发送完成。</p>
进入透传命令	AT+TSP\r\n	AT,OK\r\n	<p>设备将进入透传模式，之后所有串口的数据将会直接被发送出去。</p> <p>注意：发送完成后模块并不会提示任何信息！需要用户评估一下所需要的时间做好两次发送的间隔延时。如果模块正在发送上一次的数</p>

			据，又接收到用户新要求发送的数据，则会报 “AT,busy...” 错误。
退出透传	+++	AT,OK\r\n AT,busy...\r\n	<p>设备成功退出透传模式进入标准 AT 命令。</p> <p>设备忙于发送数据，请稍后再试</p> <p>用户必须保证模块在空闲的状态下（不进行发送数据的状态）发送 “+++” 命令才能正确的退出透传模式，所以建议用户在发送 “+++” 之前加一个合适的延时。</p>

参数配置详解

配置参数命令字段	描述	范围	示例
载波频率	模块工作时的载波频率，十进制，用 9 个字符表示， <u>如果当地某个频点受干扰比较严重就可以切换下载波频率</u>	410MHz-470MHz	433000000
功率	发射功率，十进制，用 2 个字符表示， <u>发射功率越大功耗越大，发射距离越远</u>	5dBm-20dBm	20
调制带宽	<u>发射占用信道的带宽，带宽越大发送数据越快，但灵敏度也就越低，传输距离越近。</u> 配置命令中仅使用带宽的代号，不用出现实际带宽数。	7.8K-500K，代号与带宽对应关系如下： 0: 7.8KHz 1: 10.4KHz 2: 15.6KHz 3: 20.8KHz 4: 31.2KHz 5: 41.6KHz 6: 62.5KHz 7: 125KHz 8: 250KHz 9: 500KHz	6
扩频因子	<u>扩频通讯的关键参数，扩频因子越大发送数据越慢，但灵敏度也就越高，传输距离越远。</u> 配置命令中仅使用扩频因子的代号，不用出现实际扩频因子。	64-4096，代号与扩频因子的对应关系如下： 6: 64 7: 128 8: 256 9: 512 10: 1024 11: 2048 12: 4096	10
纠错码	扩频通讯的关键参数，配置命令中仅使用纠错码的代号，不用出现实际纠错码。	4/5-4/8，代号与纠错码对应关系如下： 1: 4/5 2: 4/6 3: 4/7 4: 4/8	1
CRC 校验	用户数据 CRC 校验，SX1278 内部的 CRC 校验工具开启或者关闭。	0: 关闭 1: 开启	1
隐式报头	在用应用在固定帧长度的场合，并正确的设置了用户所需要的用户数据长度，可以设置隐士报头为 TRUE（1）。这样可以减少模块传输的数据，进而提高用户的传输速度。	0: 显式 1: 隐式	0

单次接收	接收模式设置，此功能只能在 AT 命令模式下工作，透传模式只支持连续接收，用户如果设置了单次接收模块将无法正常工作。	0: 连续 1: 单次	0
跳频设置	不支持！设置任何都可，模块会忽略这个设置。	0: 不支持 1: 支持	0
跳频周期	每次跳频间隔时间，模块会忽略这个设置。	保留	0
接收数据的超时时间	接收数据超时时间，在单次接收模式下，当超过此时间还没接收到数据软件，模块报超时错误，并自动进入 IDLE 模式，十进制表示，单位为毫秒	1-65535	3000
用户数据长度	用户数据长度，十进制表示作用：应用在隐式报头模式下，指定模块发送和接收数据的长度（此长度=实际用户数据长度+4）。显示报头下无效。	5-255	8
前导码长度	前导码长度，十进制表示	4-65535	4

错误命令含义表

错误命令	含义
ERR:CMD\r\n	命令错误，发送的命令格式错误。
ERR:CPU_BUSY\r\n	CPU 忙错误。
ERR:RF_BUSY\r\n	SX1278 忙错误，当用户发送数据时，SX1278 没有完成上次的发送任务会返回此错误。
ERR:SYMBLE\r\n	命令的后面只能识别=或者?，如果某个命令不支持标识符或者用户发送了错误的标识符，会返回此错误。
ERR:PARA\r\n	参数错误，如果用户在=标识后面输入了错误的参数，返回此错误。
AT,busy...\r\n	透传模式下，如果模块正在处于无线发送状态，用户再次尝试通过串口发送数据，则会出现此错误。



用户使用过程参考 (Ting-01M)

概述

本模块上电后自动加载上一次保存的参数，并直接进入透传模式，用户可直接通过串口收发数据。如果用户想更改配置参数，则需要发送“+++”退出透传命令，正确的进入 AT 命令模式后更改参数，并保存。之后用户可以有三种操作方法

- 1、通过复位命令 (AT+RST\r\n) 复位模块，等待模块复位成功后，可直接进行收发数据。
- 2、退出透传模式，模块将自动进入接收状态，可以异步的接收数据，用户可直接在 AT 命令模式下进行通讯，发送 AT+RX\r\n 则进入接收状态；发送 AT+SEND=10\r\n 则进入发送状态，然后发送 10 (和前面的命令对应) 个用户数据；发送 AT+IDLE\r\n 则 SX1278 进入空闲状态，持续电流将降低至 0.7mA。
- 3、通过 AT+TSP\r\n 命令直接进入透传模式。进入透传模式后，如果模块在 IDLE 模式，则自动进入接收模式。

地址过滤功能是本模块的一个特色，他可以灵活的开关，在关闭情况下所有模块只要无线参数配置一样即可通讯，也可兼容其他品牌的透传模块，如果开启地址过滤功能，则模块会自动屏蔽不是发给自己的讯息，即使在透传模式下也可以实现地址过滤，将不关心的数据过滤掉，降低用户多模块通讯信息需要用户过滤其他模块信息的开发难度。

一、硬件连接

1、在电脑端连接模块：

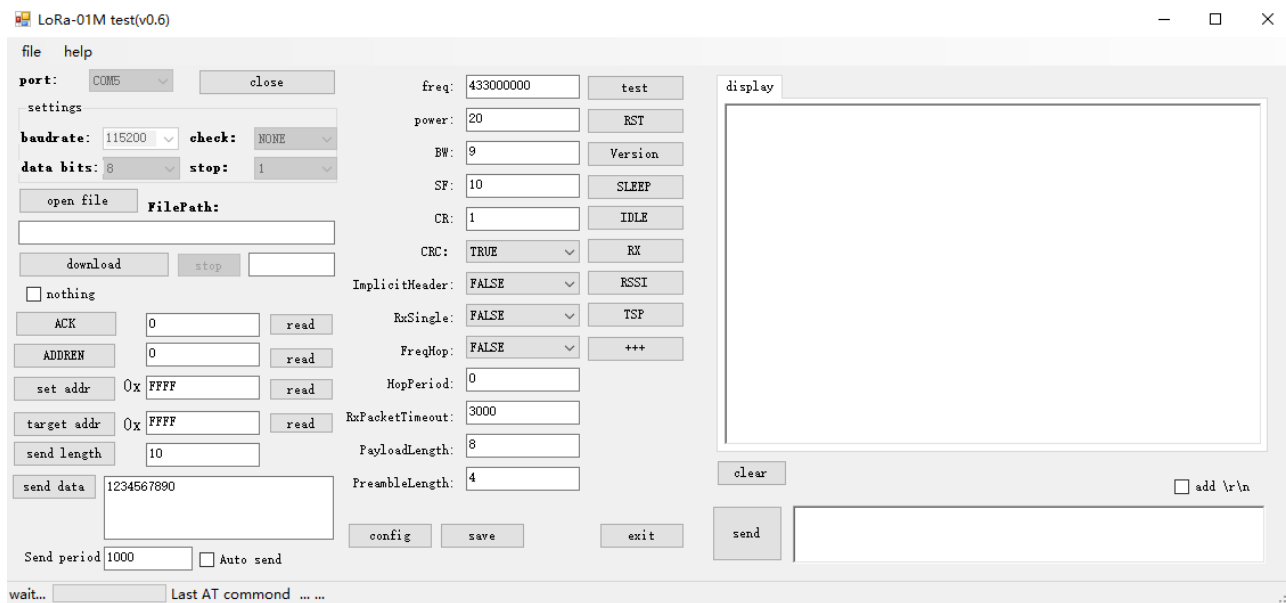
用户需要通过 USB 转串口模块的 RX, TX 交叉链接模块的 RX,TX, 并链接 GND 和 VCC, 一定要保证 VCC 是 3.3V 电压, 电压如果是 5V 则会烧毁模块。

2、单片机连接模块：

用户需要通过 MCU 串口和模块的 RX, TX 交叉连接。然后连接 GND 和 VCC, 一定要保证 VCC 是 3.3V 电压, 电压如果是 5V 则会烧毁模块。另外用户可以将“CPU_RST”引脚连接到 MCU 的 IO 上, 用户可以控制模块强行复位。将“PC4”连接到 IO 上, 在开启 ACK 的情况下, 用户可以通过此 IO 检测是否接收到新的数据。也可将“PC4”连接到 LED 上, 作为一个接收指示灯。

二、软件测试（通过上位机测试模块）

用户需要通过 USB 转串口模块的 RX, TX 交叉链接模块的 RX,TX, 并链接 GND 和 VCC, 一定要保证 VCC 是 3.3V 电压, **电压如果是 5V 则会烧毁模块。**



发送端：打开 LoRa-01M test 程序选择正确的串口打开, 如果需要修改配置, 则点击“+++”退出透传模式后, 更新设置并保存, 点击“RST”或者“TSP”进入透传模式。可在右下角的空白区域填写用户数据, 并点击“发送”可将数据发送出去。用户必须保证再次点击发送的时候, lora 模块已经成功的将数据发送完成, 否则则会返回“AT,busy...”错误。用户接收到的数据自动的显示在接收区。

接收端：打开 LoRa-01M test 程序选择正确的串口打开, 如果需要修改配置, 则点击“+++”退出透传模式后, 更新设置并保存, 点击“复位”或者“TSP”进入透传模式。用户将接收到发送端发送的数据并显示接收区, 即使用户在非透传模式也可以接收到数据。

注意：

1、如果用户在透传模式退出后, LoRa 状态会自动进入接收状态, 所以即使不在透传模式用户也可以接收到发送端的数据。如果不想接收数据请将模块设置为 IDLE 模式。

2、用户必须保证两端的无线参数完全一致，并且设置合适的地址规则，才能正确的接收到数据。

3、发送端关闭 ADDREN，接收端开启 ADDREN，在这种情况下，如果接收方地址为 0xFFFF，或者发送方目标地址为 0xFFFF，则会将无地址协议的数据按照有地址协议的数据格式进行解析，而 0xFFFF 代表不过滤任何地址数据。所以会导致解析出错误的源地址和数据长度（将用户的有效负载数据的前 1, 2 个字节解析为源地址, 3, 4 字节解析为目标地址，数据长度还是用户的数据长度，输出结果会丢弃前 4 个字节，并在用户数据结束后增加 4 个不确定的数据）。

例子：

发送：“1234567890”

接收方，在接收完成后并通过串口将此帧数据发给主机“LR,3132,0A,567890xxxx”

31:代表发送方第一个字节‘1’的 ASCII 码 31;

32:代表发送方第二个字节‘2’的 ASCII 码 32;

0A:代表发送方总共的数据长度。

xxxx:是由于用户数据 1234 被错误的认为是源地址和目的地址，模块再转发出来的时候跳过前 4 个字节，所以后面会出现 xxxx 四个不确定的数据。

如果发送方和接收方的源地址和目标地址都不是 0xFFFF 广播地址，则会按照严格的过滤方法，如果无法通过地址匹配，则数据不会输出。

4、发送端开启 ADDREN，接收端关闭 ADDREN，如果接收方地址为 0xFFFF，或者发送方目标地址为 0xFFFF，在这种情况下，会出现接收方多接收到前 4 个字节数据是发送端的源地址和目地址信息，之后为用户数据。

5、用户可以在上位尝试**命令模式**，**透传模式**，**地址使能**，**各种地址配置**的任意组合，测试出想要的结果。

6、单片机控制状态下和上位机的思路一致。

三、关于冲突

LoRa 模块为半双工工作模式，两个模块最好工作在一问一答的状态，类似于 485 总线的工作状态，这样能保证数据不会丢失，否则产生冲突无法保证数据能正确的发送出去。而且模块可能无法报出此错误。用户如果使用“多主动发送模式”产生此冲突时必然的，用户需要通过软件协议来保证数据的正常通讯。

如果在当地已经有人使用了 LoRa 模块并且占用了某个频点，也会给模块造成干扰。这样的话，用户需要切换频点以避免不必要的干扰。

四、关于无线参数

详见：参数配置详解

五、更新固件

通过“open file”选择对应的 bin 文件。如果模块工作于透传模式，点击“+++”按钮使模块正确的退出透传模式，然后点击“download”按钮，等待下方进度条完成 100%完成更新固件程序，模块自动复位并加载上次保存的参数（更新固件不会清除之前的设置），之后进入透传模式。

file help

port: COM5 close

settings

baudrate: 115200 check: NONE

data bits: 8 stop: 1

open file FilePath:

C:\Ting01M_V0.6.bin

download stop

☐ nothing

ACK 0 read

ADDREN 0 read

set addr 0x FFFF read

target addr 0x FFFF read

send length 10

send data 1234567890

Send period 1000 ☐ Auto send

wait... Last AT command ...

最新资料:

https://github.com/eboxmaker/STM8L_Lora